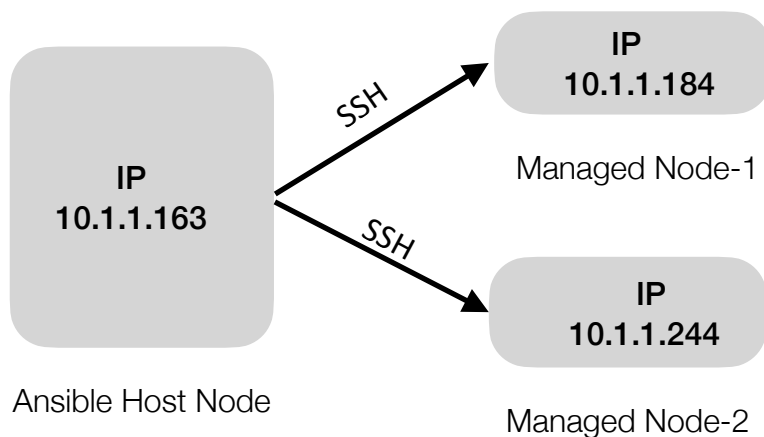


Installation of nginx web-server on managed nodes using Ansible

Overview:

Ansible is an open-source automation tool used for IT tasks such as configuration management, application deployment, intra-service orchestration, and provisioning. It helps automate repetitive tasks, thereby speeding up processes. In this project let's use it as configuration management tool.



Steps to configure:

1. Let's create 3 virtual machines and deploy it (I have deployed on AWS cloud), you can use any cloud providers. Given the names accordingly.

Instances (1/3) Info							
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>				All states ▾			
<input checked="" type="checkbox"/>	Name ↗	Instance ID	Instance state ▾	Instance type ▾	Status check	Alarm status	
<input checked="" type="checkbox"/>	linux-host	i-08ea7ec4bb357793b	Running	t2.micro	2/2 checks passed	View alarms	
<input type="checkbox"/>	linux-node-1	i-076050bcf46081624	Running	t2.micro	2/2 checks passed	View alarms	
<input type="checkbox"/>	linux-node-2	i-040fee900d8230ad9	Running	t2.micro	2/2 checks passed	View alarms	

2. Login to the host machine and make password less authentication to managed nodes.
3. To make password less run "ssh-keygen" you will get a pair of public key (id_rsa.pub) and private key (id_rsa).

```
[root@ip-10-1-1-163 ~]# cd .ssh
[root@ip-10-1-1-163 .ssh]# ls
authorized_keys  id_rsa  id_rsa.pub  known_hosts
[root@ip-10-1-1-163 .ssh]#
```

4. Copy the public key and paste it on the managed host under “.ssh/authorised_keys”. Do the same thing with another node.
5. Create a inventory.ini file which contains the ip's of the nodes.

```
[root@ip-10-1-1-163 Ansible]# ls
inventory.ini  playbook.yml
[root@ip-10-1-1-163 Ansible]# cat inventory.ini
10.1.1.188
10.1.1.244
```

6. Before proceeding further test the the password-less authentication is working fine.
7. From the host server run the command below.
#ssh <IP of node>
8. Install ansible on the host server “yum install ansible”. Depending upon the os you are working with, use software package management tool accordingly.
9. Now create a playbook.yml file which contains the steps to install the nginx server on the nodes.

```
---
- name: Install and start Nginx server
  hosts: webservers
  become: yes # Execute tasks with sudo

  tasks:
    - name: Ensure EPEL repository is enabled
      yum:
        name: epel-release
        state: present

    - name: Ensure Nginx is installed
      yum:
        name: nginx
        state: present
      notify:
        - Start Nginx

    - name: Ensure Nginx is running and enabled on boot
      service:
        name: nginx
        state: started
        enabled: yes

  handlers:
    - name: Start Nginx
      service:
        name: nginx
        state: restarted
```

10. Before running the .yml file you can make a dry run.

```
#ansible-playbook --check playbook.yml
```

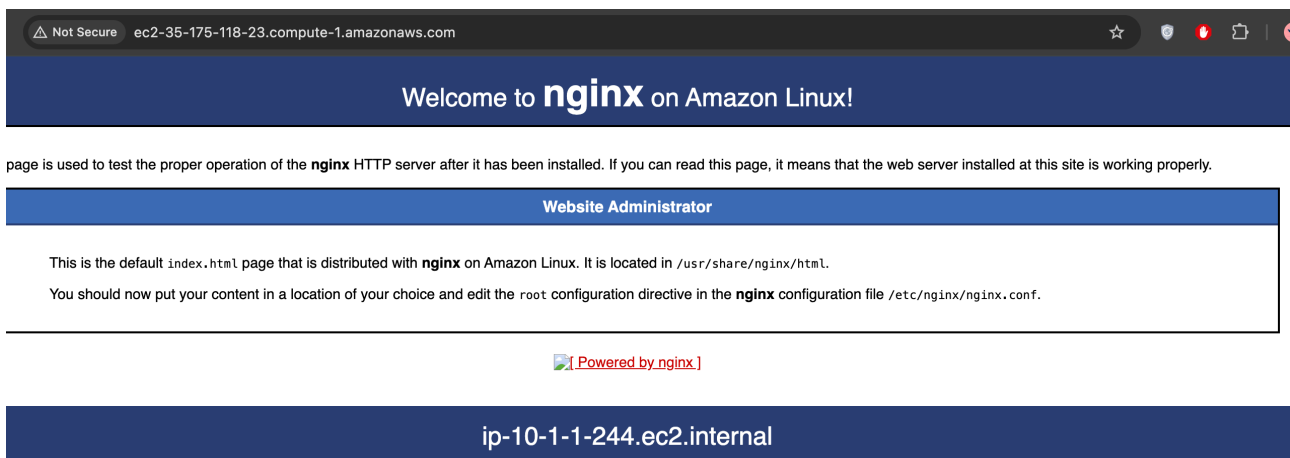
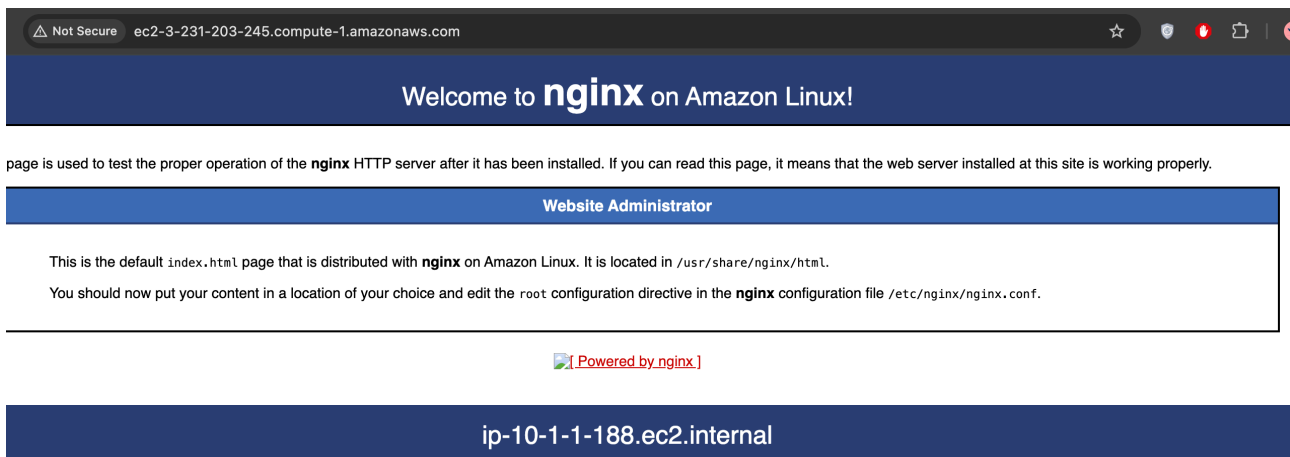
11. You can also check if there are any syntax errors in the file.

```
#ansible-playbook --syntax-check playbook.yml
```

12. Let's run the playbook and test it.

```
#ansible -i inventory.ini all playbook.yml
```

13. Copy the public dns of the server and paste it in browser.



14. It's perfectly working.

15. We have successfully installed nginx-web server on managed nodes using Ansible.