

Web Application Using Ansible

Objective

To create an automation script to deploy an application using Ansible and Jinja2 template

Problem Statement and Motivation

Real-time scenario:

You have joined as a DevOps engineer in XYZ Pvt. Ltd. It is a platform where individuals can create their profile and start blogging on various topics. The application is ready to be hosted on a server.

You are tasked with implementing an Ansible script to deploy this application on a remote Nginx server.

Tasks

The following tasks outline the process of deploying web application on a remote server:

1. Create an inventory file to define the remote server(s)
2. Write a YAML playbook with tasks for installing Nginx, copying web application files, deploying the Nginx configuration, and enabling the site
3. Create a directory for templates and a Jinja2 template for the Nginx configuration
4. Define variables in the playbook for application details and Nginx configuration
5. Include tasks in the playbook for installing Nginx, copying application files, deploying Nginx configuration, and enabling the Nginx site
6. Execute the playbook to deploy the web application on the remote server

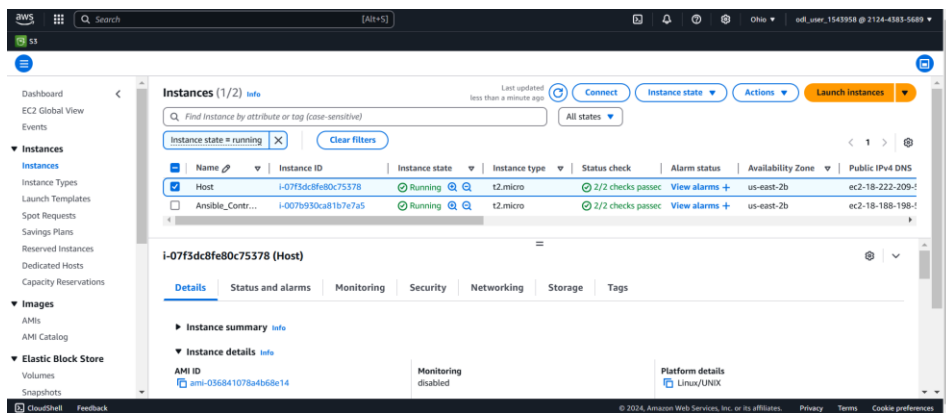
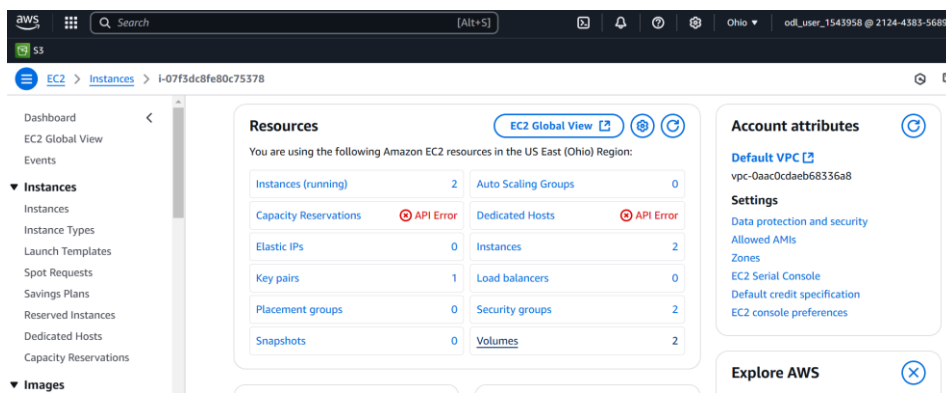
Procedure

1. Log in to Amazon Web Services

2. Launching 2 EC2 Instances

1. Ansible Controller

2. Host



3.Connecting to instances through Command prompt and SSH

```
ubuntu@ip-172-31-19-125:~$ client_loop: send disconnect: Connection reset

C:\Users\HP\Downloads>ssh -i "Project_ansible.pem" ubuntu@ec2-18-188-198-52.us-east-2.compute.amazonaws.com
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1018-aws x86_64)

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

System information as of Thu Dec 12 18:46:03 UTC 2024

System load:  0.0          Processes:      123
Usage of /:   35.3% of 6.71GB Users logged in: 1
Memory usage: 33%         IPv4 address for enx0: 172.31.19.125
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

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

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Thu Dec 12 18:29:47 2024 from 152.59.212.126
ubuntu@ip-172-31-19-125:~$
```

On Host Machine

```
Ansible_controller  x  Host  x  +  v

C:\Users\HP\Downloads>ssh -i "Project_ansible.pem" ubuntu@ec2-18-222-209-50.us-east-2.compute.amazonaws.com
The authenticity of host 'ec2-18-222-209-50.us-east-2.compute.amazonaws.com (18.222.209.50)' can't be established.
ED25519 key fingerprint is SHA256:1GoBnmtBhbrRDJLyr8a4yS29Qpt0XNFooNHgdHbotXk.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-18-222-209-50.us-east-2.compute.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1018-aws x86_64)

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

System information as of Thu Dec 12 14:47:09 UTC 2024

System load:  0.0          Processes:      104
Usage of /:   24.7% of 6.71GB Users logged in:  0
Memory usage: 21%         IPv4 address for enx0: 172.31.20.172
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

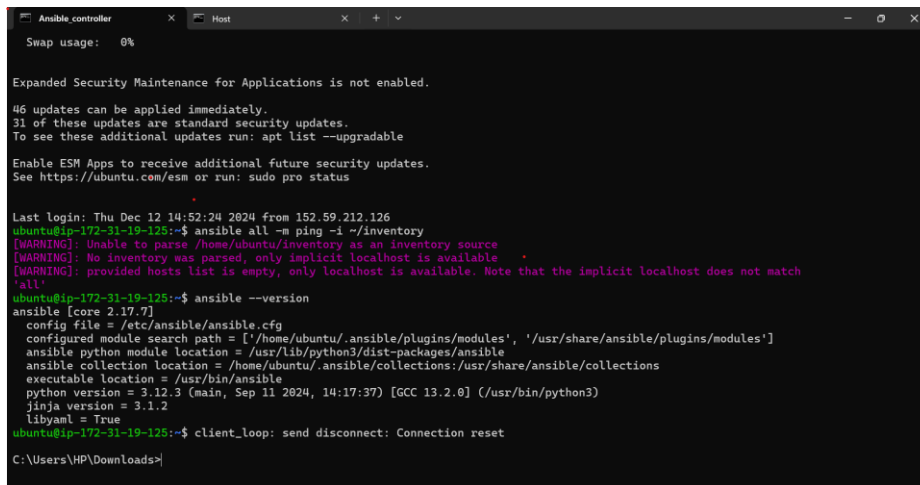
The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
```

4.Install Ansible on 1 machine i.e on Ansible Controller

```
sudo apt update
sudo apt install -y software-properties-common
sudo add-apt-repository --yes --update ppa:ansible/ansible
sudo apt install -y ansible
```

Verify version



```
Ansible_controller  x  Host  x  +  x
Swap usage:  0%

Expanded Security Maintenance for Applications is not enabled.

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

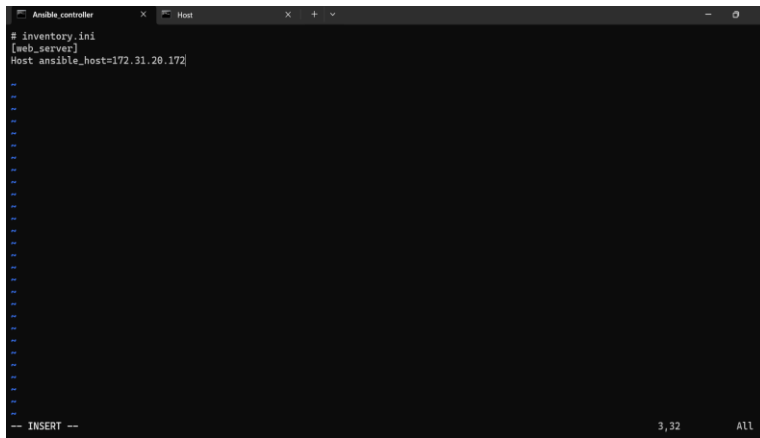
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Thu Dec 12 14:52:24 2024 from 152.59.212.126
ubuntu@ip-172-31-19-125:~$ ansible all -m ping -i ~/inventory
[WARNING]: Unable to parse /home/ubuntu/inventory as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
ubuntu@ip-172-31-19-125:~$ ansible --version
ansible [core 2.17.7]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/ubuntu/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.12.3 (main, Sep 11 2024, 14:17:37) [GCC 13.2.0] (/usr/bin/python3)
  jinja2 version = 3.1.2
  libyaml = True
ubuntu@ip-172-31-19-125:~$ client_loop: send disconnect: Connection reset
C:\Users\HP\Downloads>
```

5.Setup Host and inventories

Create an inventory

Create a file vi ~/inventory

A screenshot of a terminal window titled 'Ansible controller'. The window shows the content of the 'inventory.ini' file. The text displayed is: '# inventory.ini', '[web_server]', and 'Host ansible_host=172.31.20.172'. The terminal has a dark background with light blue text. The status bar at the bottom shows '-- INSERT --' on the left and '3,32 All' on the right.

```
# inventory.ini
[web_server]
Host ansible_host=172.31.20.172
```

5.Connecting Ansible and host machine

Setup SSH connection between Ansible and host machines

- On the control machine
 - Check is a SSH keypair exists already
 - ll ~/.ssh
 - If SSH key-pair don't exist, generate SSH keys
 - ssh-keygen -t rsa
 - (just press enter for any input prompts)
 - Print SSH public Key
 - cat ~/.ssh/id_rsa.pub
 - Copy the above printed key

- On each Host machine

- a. Connect to each host instance
- b. Add public key (copied from the master machine) to the end of below file as a new line (Please DO NOT delete any existing data from the below file)

i.nano ~/.ssh/authorized_keys

6.Verify the SSH connection to hosts

On Ansible machine

Commented [NN1]: Pinging all hosts

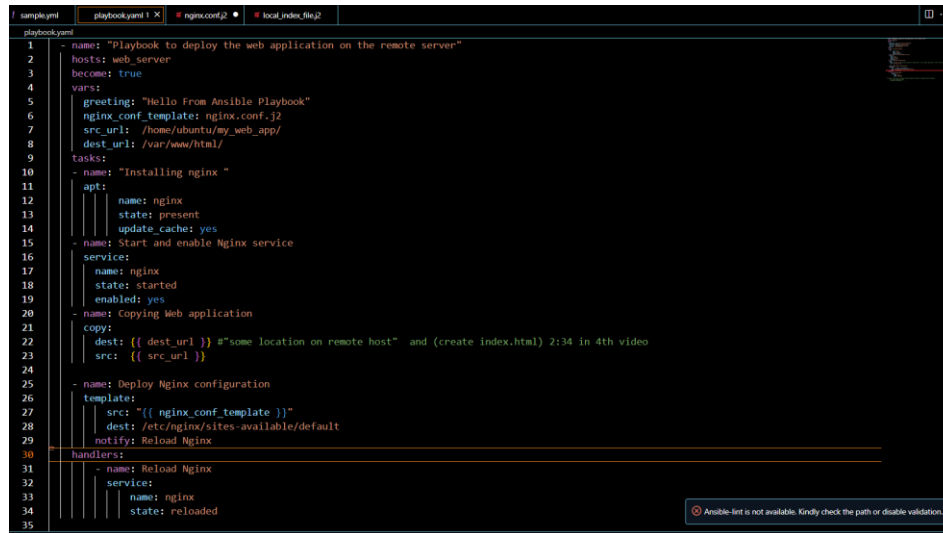
```
ansible all -m ping -i ~/inventory
```

Expected output

```
ubuntu@ip-172-31-19-125:~$ vi ~/inventory
ubuntu@ip-172-31-19-125:~$ ansible all -m ping -i ~/inventory
The authenticity of host '172.31.20.172 (172.31.20.172)' can't be established.
ED25519 key fingerprint is SHA256:1Go8nmt8hbrRDJLyr8a4ySZ9Qpt8XNFooNHgdHbotXk.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
[WARNING]: Platform linux on host Host is using the discovered Python interpreter at /usr/bin/python3.12, but future installation of
another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
Host | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.12"
  },
  "changed": false,
  "ping": "pong"
}
ubuntu@ip-172-31-19-125:~$ |
```

Write a YAML Playbook (Playbook.yaml)

Next, write the playbook with tasks for installing Nginx, copying web application files, deploying the Nginx configuration, and enabling the site.



```
1 - name: "Playbook to deploy the web application on the remote server"
2   hosts: web_server
3   become: true
4   vars:
5     greeting: "Hello From Ansible Playbook"
6     nginx_conf_template: nginx.conf.j2
7     src_url: /home/ubuntu/my_web_app/
8     dest_url: /var/www/html/
9   tasks:
10    - name: "Installing nginx "
11      apt:
12        name: nginx
13        state: present
14        update_cache: yes
15    - name: Start and enable Nginx service
16      service:
17        name: nginx
18        state: started
19        enabled: yes
20    - name: Copying Web application
21      copy:
22        dest: "{{ dest_url }}" s"some location on remote host" and (create index.html) 2:34 in 4th video
23        src: "{{ src_url }}"
24    - name: Deploy Nginx configuration
25      template:
26        src: "{{ nginx_conf_template }}"
27        dest: /etc/nginx/sites-available/default
28        notify: Reload Nginx
29    handlers:
30      - name: Reload Nginx
31        service:
32          name: nginx
33          state: reloaded
34
35
```

Create a Directory for Templates and a Jinja2 Template

Create a directory for templates and add a Jinja2 template for the Nginx configuration

\$ mkdir templates

Create the Jinja2 template file nginx.conf.j2 in the templates directory:

\$ Vi nginx.conf.j2

```

1 # templates/nginx.conf.j2
2 server { listen 80;
3     server_name http://18.222.209.50/;
4     root {{ dest_url }};
5     index index.html index.htm;
6     location / {
7         try_files $uri $uri/ =404;
8     } }
9

```

Create sample Web app

Created an index.html file in /home/ubuntu/my_web_app/index.html

Executed Command (Output)

Ansible-playbook -i ~/inventory Playbook.yaml

```

Ansible_controller  x  Host  x  +  v
To see these additional updates run: apt list --upgradable
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Thu Dec 12 17:03:02 2024 from 152.59.212.126
ubuntu@ip-172-31-19-125:~$ ansible-playbook -i ~/inventory Playbook.yaml

PLAY [Playbook to deploy the web application on the remote server] *****

TASK [Gathering Facts] *****
ok: [Host]

TASK [Installing nginx] *****
ok: [Host]

TASK [Start and enable Nginx service] *****
ok: [Host]

TASK [Copying Web application] *****
ok: [Host]

TASK [Deploy Nginx configuration] *****
ok: [Host]

PLAY RECAP *****
Host : ok=5  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0

```


Steps To check NGINX Server deployed on webserver

To verify that your web application has been deployed successfully on your web server

1. Verify Nginx Status

sudo systemctl status nginx

```
Expanded Security Maintenance for Applications is not enabled.
51 updates can be applied immediately.
31 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Thu Dec 12 18:04:18 2024 from 172.31.19.125
ubuntu@ip-172-31-20-172:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Active: active (running) since Thu 2024-12-12 17:19:19 UTC; 45min ago
     Docs: man:nginx(8)
   Process: 2480 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Process: 2482 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Process: 6710 ExecReload=/usr/sbin/nginx -g daemon on; master_process on; -s reload (code=exited, status=0/SUCCESS)
  Main PID: 2483 (nginx)
    Tasks: 2 (limit: 1130)
   Memory: 1.8M (peak: 3.0M)
      CPU: 16ms
   CGroup: /system.slice/nginx.service
           └─2483 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─6712 "nginx: worker process"

Dec 12 17:19:19 ip-172-31-20-172 systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
Dec 12 17:19:19 ip-172-31-20-172 systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
Dec 12 18:03:59 ip-172-31-20-172 systemd[1]: Reloading nginx.service - A high performance web server and a reverse proxy server...
Dec 12 18:03:59 ip-172-31-20-172 nginx[6710]: 2024/12/12 18:03:59 [notice] 6710#6710: signal process started
Dec 12 18:03:59 ip-172-31-20-172 systemd[1]: Reloaded nginx.service - A high performance web server and a reverse proxy server.
ubuntu@ip-172-31-20-172:~$
```

2. Check the Web Root Directory

Ensure that the web application files have been copied to the correct directory (/var/www/html/):

ls -l /var/www/html/

```
ubuntu@ip-172-31-20-172:~$ ls -l /var/www/html/
total 12
-rw-r--r-- 1 root root 551 Dec 12 18:03 index.html
-rw-r--r-- 1 root root 615 Dec 12 17:19 index.nginx-debian.html
-rw-r--r-- 1 root root 411 Dec 12 18:03 styles.css
```

We should see Our web application files listed here.

3. Check Nginx Configuration

Verify that the Nginx configuration is set up correctly. You can test the configuration with the following command:

```
sudo nginx -t
```

```
ubuntu@ip-172-31-20-172:~$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
ubuntu@ip-172-31-20-172:~$
```

If the configuration is correct, you should see a message indicating that the test was successful.

4. Access the Web Application

<http://18.222.209.50/>

