

CMPE 272 - Enterprise Software Platforms

Name: Gautam Santhanu Thampy

SJSU ID:018222477

Github: https://github.com/gautamthampy/CMPE-272-Enterprise-Software-Platforms/tree/7726b697e6d58857c29013f1712c47515adce05c/Assignment_1/ansible-webserver

Homework #1 - Deployment of Web Server using Ansible on two VMs

Problem Statement:

- Configure two VMs, **VM1** and **VM2** either on your own hardware, or in a cloud environment. Configure Ansible to deploy a web server on **VM1** and **VM2** on port 8080 with a web page that is accessible from a web browser, and displays the message: “Hello World from SJSU-X” where X is 1 or 2 depending on which web server instance, VM1 or VM2.
- Include in the Ansible playbook, plays to **deploy** and **undeploy** the web server resources
- Submit a Word document, with screenshots showing your work, and a demo, and all ansible code/scripts via github

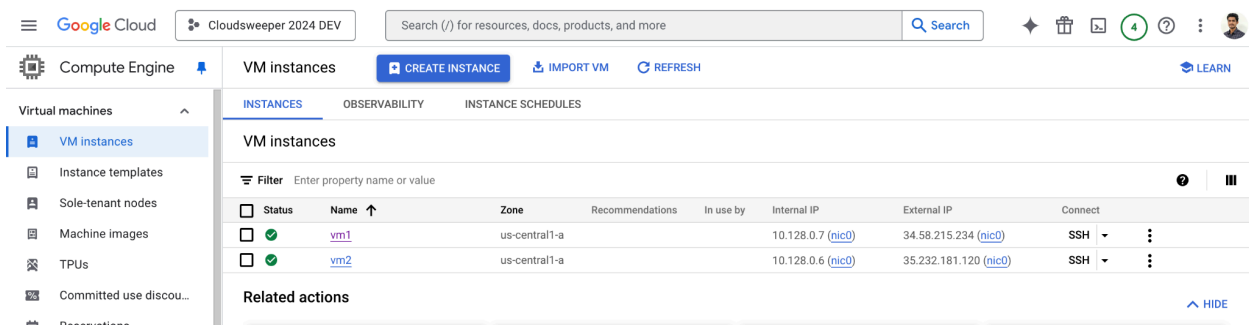
Solution:

- **Step 1: Setup VMs**

I have used the gcloud command to setup 2 virtual machines on Google Cloud Platform.

- VM specifications:

I have used an e2-micro (2 vCPUs, 1 GB Memory) instance-type for both the virtual machines. Ubuntu is the preferred OS used for this homework.



The screenshot shows the Google Cloud Platform interface for VM instances. The left sidebar lists navigation options like 'Virtual machines', 'Instance templates', 'Sole-tenant nodes', 'Machine images', 'TPUs', 'Committed use discount', and 'Reservations'. The main panel is titled 'VM instances' and includes tabs for 'INSTANCES', 'OBSERVABILITY', and 'INSTANCE SCHEDULES'. A table lists two instances, 'vm1' and 'vm2', both in the 'us-central1-a' zone. The table columns include Status, Name, Zone, Recommendations, In use by, Internal IP, External IP, and Connect. Both instances are shown with a green checkmark status and have their respective internal and external IP addresses listed. Below the table, there is a 'Related actions' section with a 'HIDE' link.

Status	Name	Zone	Recommendations	In use by	Internal IP	External IP	Connect
✓	vm1	us-central1-a			10.128.0.7 (nic0)	34.58.215.234 (nic0)	SSH
✓	vm2	us-central1-a			10.128.0.6 (nic0)	35.232.181.120 (nic0)	SSH

Fig 1: vm1 and vm2 successfully hosted on GCP

Gcloud command for creating 2 VMs

```
gcloud compute instances create vm1 vm2 \  
  --machine-type=e2-micro \  
  --zone=us-central1-a \  
  --image-family=ubuntu-2004-lts \  
  --image-project=ubuntu-os-cloud \  
  --metadata-from-file ssh-keys=PATH_TO_YOUR_PUBLIC_KEY
```

- Configure SSH for Persistent Access:

Added the SSH key to project-wide metadata in GCP in order to prevent the resetting of ssh keys when the vm is refreshed or reset.

Gcloud command for persistent ssh access

```
gcloud compute project-info add-metadata \  
  --metadata-from-file ssh-keys=PATH_TO_YOUR_PUBLIC_KEY
```

- **Step 2: Download and setup ansible on your machine**

Ansible was installed using Homebrew command: ***brew install ansible***

- **File setup:**

inventory.ini

```
[webservers]  
vm1 ansible_host=34.58.215.234  
vm2 ansible_host=35.232.181.120  
  
[all:vars]  
ansible_user=<Username>  
ansible_ssh_private_key_file=~/.ssh/id_ed25519  
ansible_become=yes  
ansible_become_method=sudo  
ansible_become_pass=NOPASSWD
```

site.yml

- name: Deploy webserver

hosts: webserver

become: yes

vars:

http_port: 8080

tasks:

- name: Install Python3 and pip

apt:

name:

- python3

- python3-pip

state: present

update_cache: yes

- name: Install required packages

apt:

name:

- nginx

state: present

- name: Create nginx sites-available configuration

template:

src: nginx.conf.j2

dest: /etc/nginx/sites-available/default

notify: restart nginx

- name: Create web content directory

file:

path: /var/www/html

state: directory

mode: '0755'

- name: Deploy index page

template:

src: index.html.j2

```
    dest: /var/www/html/index.html
    mode: '0644'
    notify: restart nginx
```

handlers:

- name: restart nginx
- ```
 service:
 name: nginx
 state: restarted
```

### undeploy.yml

```

- name: Undeploy webservers
 hosts: webservers
 become: yes

 tasks:
 - name: Stop nginx service
 service:
 name: nginx
 state: stopped

 - name: Remove nginx package
 apt:
 name: nginx
 state: absent
 purge: yes

 - name: Remove web content
 file:
 path: /var/www/html
 state: absent
```

## nginx.conf.j2

```
server {
 listen {{ http_port }};
 server_name _;

 root /var/www/html;
 index index.html;

 location / {
 try_files $uri $uri/ =404;
 }
}
```

## index.html.j2

```
<!DOCTYPE html>
<html>
<head>
 <title>SJSU Webserver</title>
 <style>
 body {
 font-family: Arial, sans-serif;
 display: flex;
 justify-content: center;
 align-items: center;
 height: 100vh;
 margin: 0;
 background-color: #f0f0f0;
 }
 .container {
 text-align: center;
 padding: 20px;
 background-color: white;
 border-radius: 8px;
 box-shadow: 0 2px 4px rgba(0,0,0,0.1);
 }
 h1 {
```

```

 color: #2c3e50;
 }
</style>
</head>
<body>
 <div class="container">
 <h1>Hello World from SJSU-{% if inventory_hostname == 'vm1' %}1{%
else %}2{% endif %}</h1>
 </div>
</body>
</html>

```

- **Step 3: Test the connection and deploy the web servers:**

I tested the ansible connection with: ***ansible -i inventory.ini webserver -m ping***

```

(base) gautamthampy@Gautams-Laptop ~ % ansible -i inventory.ini webserver -m ping

[WARNING]: Platform linux on host vm1 is using the discovered Python
interpreter at /usr/bin/python3.10, but future installation of another Python
interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-
core/2.18/reference_appendices/interpreter_discovery.html for more information.
vm1 | SUCCESS => {
 "ansible_facts": {
 "discovered_interpreter_python": "/usr/bin/python3.10"
 },
 "changed": false,
 "ping": "pong"
}
[WARNING]: Platform linux on host vm2 is using the discovered Python
interpreter at /usr/bin/python3.10, but future installation of another Python
interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-
core/2.18/reference_appendices/interpreter_discovery.html for more information.
vm2 | SUCCESS => {
 "ansible_facts": {
 "discovered_interpreter_python": "/usr/bin/python3.10"
 },
 "changed": false,
 "ping": "pong"
}

```

Fig 2: Output of connection test

- To deploy the web servers, we run the ansible playbook as follows:

***ansible-playbook -i inventory.ini site.yml***

```
(base) gautamthampy@Gautams-Laptop ansible-webserver % ansible-playbook -i inventory.ini site.yml

PLAY [Deploy webserver] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host vm2 is using the discovered Python interpreter at /usr/bin/python3.8, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.15/reference_appendices/interpreter_discovery.html for more information.
ok: [vm2]
[WARNING]: Platform linux on host vm1 is using the discovered Python interpreter at /usr/bin/python3.8, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.15/reference_appendices/interpreter_discovery.html for more information.
ok: [vm1]

TASK [Install Python3 and pip] *****
ok: [vm2]
ok: [vm1]

TASK [Install required packages] *****
changed: [vm2]
changed: [vm1]

TASK [Create nginx sites-available configuration] *****
changed: [vm2]
changed: [vm1]

TASK [Create web content directory] *****
ok: [vm1]
ok: [vm2]

TASK [Deploy index page] *****
changed: [vm1]
changed: [vm2]

RUNNING HANDLER [restart nginx] *****
changed: [vm2]
changed: [vm1]

PLAY RECAP *****
vm1 : ok=7 changed=4 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
vm2 : ok=7 changed=4 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

Fig 3: Output of ansible playbook - web server deployment

- To undeploy the web servers, run the command:

***ansible-playbook -i inventory.ini undeploy.yml***

```
(base) gautamthampy@Gautams-Laptop ansible-webserver % ansible-playbook -i inventory.ini undeploy.yml

PLAY [Undeploy webserver] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host vm2 is using the discovered Python interpreter at /usr/bin/python3.8, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.15/reference_appendices/interpreter_discovery.html for more information.
ok: [vm2]
[WARNING]: Platform linux on host vm1 is using the discovered Python interpreter at /usr/bin/python3.8, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.15/reference_appendices/interpreter_discovery.html for more information.
ok: [vm1]

TASK [Stop nginx service] *****
changed: [vm1]
changed: [vm2]

TASK [Remove nginx package] *****
changed: [vm1]
changed: [vm2]

TASK [Remove web content] *****
changed: [vm2]
changed: [vm1]

PLAY RECAP *****
vm1 : ok=4 changed=3 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
vm2 : ok=4 changed=3 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

Fig 4: Output of ansible playbook - undeploy

## Verification:

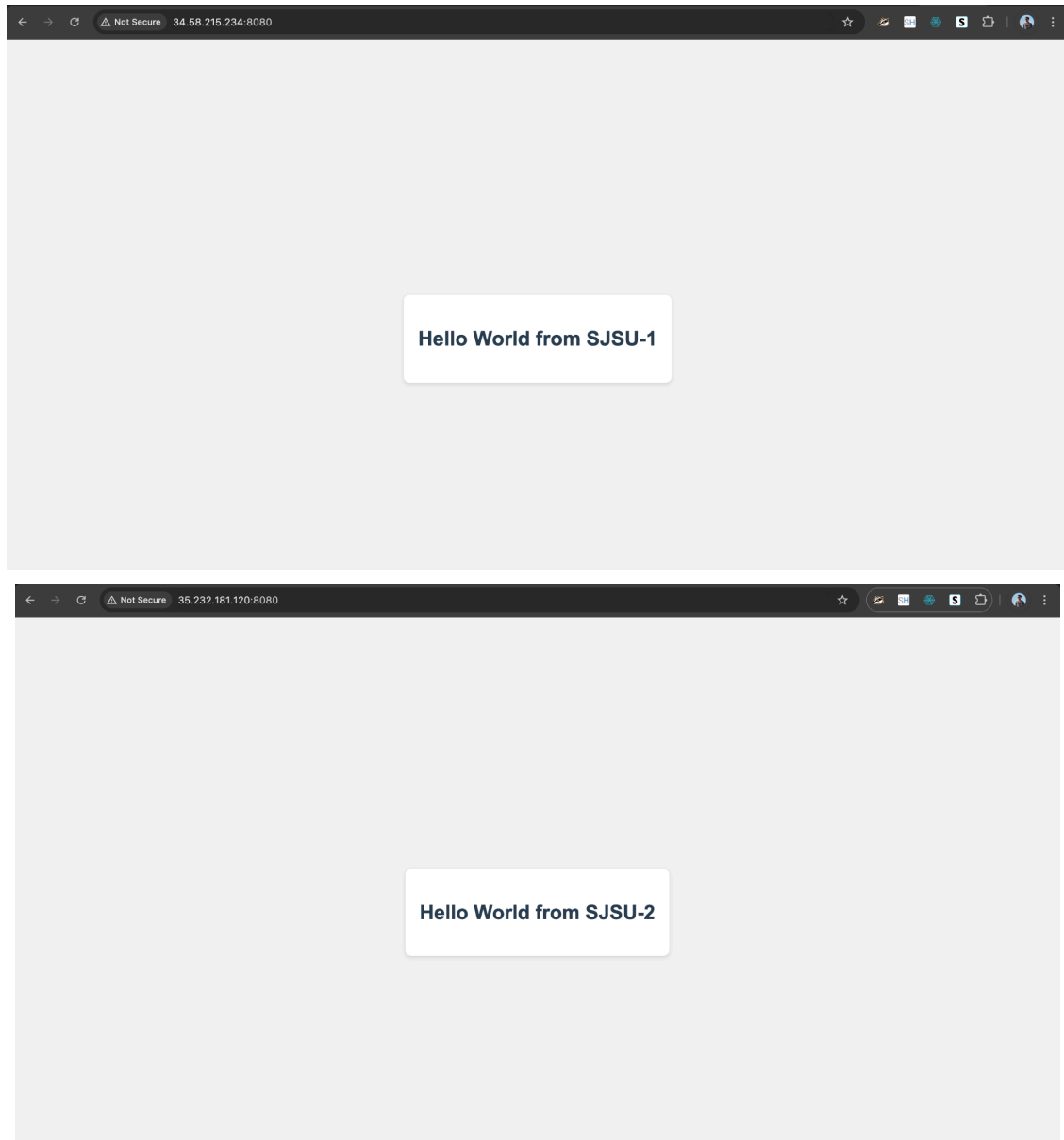


Fig 5 and 6: Web pages when accessing the URL

## References:

- Ansible documentation: <https://docs.ansible.com/>
- Gcloud cli cheatsheet: <https://cloud.google.com/sdk/docs/cheatsheet>
- Claude by Anthropic:  
<https://claude.ai/chat/b4f39a8b-c6a4-43da-a473-a028f62450cb>