Steps to Add HTTPS Support Using TLS Certificates

1. Generate Self-Signed TLS Certificates

Generate a self-signed certificate for testing purposes, or use a certificate from a trusted Certificate Authority (CA) for production.

mkdir -p files/certs

openssl req -x509 -nodes -days 365 -newkey rsa:2048 \

-keyout files/certs/nginx.key \

-out files/certs/nginx.crt \

-subj "/C=US/ST=State/L=City/O=Organization/OU=Department/CN=localhost

2. Update the NGINX Configuration

Create a custom NGINX configuration file that includes HTTPS settings.

Example configuration (`files/nginx.conf`):

server {

listen 443 ssl;

server\_name localhost;

ssl\_certificate /etc/nginx/certs/nginx.crt;

ssl\_certificate\_key /etc/nginx/certs/nginx.key;

location / {

root /usr/share/nginx/html;

index index.html;

}

}

3. Update Your Ansible Playbook

Modify your playbook to:

- Copy the TLS certificates and custom NGINX configuration to the container.

- Run the NGINX container with HTTPS support.

Here's the updated playbook:

yaml name=deploy\_website\_tls.yaml

- name: Deploy and validate a website using Docker NGINX with HTTPS

hosts: remotes

become: yes

vars\_files:

- vault.yaml

vars:

website\_path: /opt/website

site\_url: https://localhost

test\_file\_path: /tmp/test-index.html

expected\_text: "Hello from Ansible"

tasks:

- name: Create website directory

file:

path: "{{ website\_path }}"

state: directory

mode: '0755'

- name: Copy website files to remote host

copy:

src: files/index.html

dest: "{{ website\_path }}/index.html"

- name: Copy TLS certificates to remote host

copy:

src: files/certs/

dest: "{{ website\_path }}/certs/"

mode: '0600'

- name: Copy custom NGINX configuration to remote host

copy:

src: files/nginx.conf

dest: "{{ website\_path }}/nginx.conf"

- name: Run NGINX container with mounted website directory and TLS

community.docker.docker\_container:

name: nginx-site

image: nginx

state: started

restart\_policy: always

ports:

- "443:443"

volumes:

- "{{ website\_path }}:/usr/share/nginx/html:ro"

- "{{ website\_path }}/certs:/etc/nginx/certs:ro"

- "{{ website\_path }}/nginx.conf:/etc/nginx/conf.d/default.conf:ro"

- name: Display success message

debug:

msg: "Website with HTTPS deployed and validated successfully!"

4. Install Required Collections

Ensure the `community.docker` collection is installed:

ansible-galaxy collection install community.docker

5. Run the Playbook

Execute the playbook using:

bash

ansible-playbook -i inventory.ini deploy\_website\_tls.yaml --ask-vault-pass

---

Notes:

- Self-Signed Certificate: For production, replace the self-signed certificate with one issued by a trusted CA, such as Let's Encrypt.

- Verify HTTPS: Use a browser or a tool like `curl` to verify the HTTPS setup:

bash

curl -k https://localhost

This setup ensures your website is served securely over HTTPS.