

Proyecto: despliegue automatizado de servidores web con Ansible

Resumen

Este proyecto consiste en la automatización del despliegue y configuración de servidores web Apache en dos nodos Linux utilizando Ansible desde un nodo controlador. Se configura el firewall, se garantiza que Apache esté activo y se personaliza la página web en cada nodo. El objetivo es facilitar la gestión y el mantenimiento de múltiples servidores con un solo playbook.

Objetivos

Desplegar servidores web Apache en dos nodos Linux mediante Ansible.

Configurar el firewall para permitir el tráfico necesario.

Asegurar que el servicio Apache esté activo y habilitado.

Personalizar la página de inicio en cada servidor.

Tecnologías Utilizadas

Sistemas Operativos: Ubuntu Zorin OS en nodos y controlador

Herramientas: Ansible, Apache, UFW

Redes: Configuración IP estática para comunicación entre nodos

Automatización: Playbooks y roles de Ansible

1. Preparación del entorno

Crear tres máquinas virtuales Ubuntu Zorin OS (un nodo controlador y dos nodos gestionados).

Configurar Red interna NAT en VirtualBox para que las VMs se comuniquen.

Asignar IPs estáticas a las máquinas:

Esto lo hago editando el archivo /etc/netplan/01-static.yml con nano.

Controlador: 10.0.2.15

Nodo1: 10.0.2.16

Nodo2: 10.0.2.17

2. Instalación de Ansible en el nodo controlador

Actualizar paquetes:

```
sudo apt update
```

Instalar Ansible:

```
sudo apt install ansible -y
```

3. Configuración de acceso SSH sin contraseña

Generar claves SSH en el nodo controlador:

```
ssh-keygen -t rsa
```

Copiar clave pública a nodos gestionados:

```
ssh-copy-id usuario@nodo1
```

```
ssh-copy-id usuario@nodo2
```

4. Creación del inventario y estructura de Ansible

Crear directorios y archivos del proyecto:

```
mkdir -p ansible-infra/inventory ansible-infra/roles/web/tasks  
touch ansible-infra/inventory/hosts ansible-infra/site.yml ansible-infra/group_vars/all.yml ansible-  
infra/roles/web/tasks/main.yml
```

Definir hosts en inventory/hosts:

```
[webservers]  
nodo1 ansible_host=10.0.2.16 ansible_user=un usuario que haya en el nodo1  
nodo2 ansible_host=10.0.2.17 ansible_user=un usuario que haya en el nodo2
```

Definir variables globales en group_vars/all.yml:

```
firewall_ports:  
- 22  
- 80
```

5. Definición del rol web para instalar y configurar Apache

En roles/web/tasks/main.yml, incluir las tareas:

Instalar Apache

Asegurar que Apache esté activo y habilitado

Configurar UFW para abrir puertos 22 y 80

Crear página de bienvenida personalizada

```
sudo nano roles/web/tasks/main.yml
```

```
- name: Instalar Apache
```

```
apt:
```

```
name: apache2
```

```
state: present
```

```
update_cache: yes
```

```
- name: Asegurar que Apache está activo
```

```
systemd:
```

```
name: apache2
```

```
enabled: yes
state: started
- name: Abrir puertos en UFW
ufw:
rule: allow
port: "{{ item }}"
proto: tcp
loop: "{{ firewall_ports }}"
- name: Crear página de bienvenida
copy:
dest: /var/www/html/index.html
content: "<h1>Servidor {{ inventory_hostname }} desplegado automáticamente
por Ansible</h1>"
```

6. Creación del playbook principal

En site.yml, definir el playbook que aplica el rol web a los nodos webservers con privilegios de superusuario.

```
sudo nano site.yml
- name: Desplegar servidores web
hosts: webservers
become: true
roles:
- web
```

7. Ejecución del playbook

Desde el nodo controlador ejecutar:

```
ansible-playbook -i inventory/hosts site.yml
```

8. Verificación del despliegue

Acceder desde un navegador a las IPs de los nodos:

<http://10.0.2.16>

<http://10.0.2.17>

Verificar que la página muestre el mensaje personalizado indicando que el servidor fue configurado por Ansible.

Lecciones Aprendidas

Automatización: El proyecto me permitió entender cómo usar Ansible para automatizar la instalación y configuración de servicios en múltiples servidores, mejorando la eficiencia y la consistencia en los despliegues.

Gestión de infraestructura: Aprendí a definir inventarios, roles, y variables para gestionar configuraciones complejas de forma organizada y reproducible.

Resolución de problemas: Durante la ejecución, enfrenté desafíos con la conexión SSH y la configuración del firewall, lo que me ayudó a mejorar mis habilidades en diagnóstico y solución en entornos automatizados. También al hacerlo todo el VirtualBox tuve que crear una Red NAT interna y asignar una IP a cada nodo personalizada editando el archivo `/etc/netplan/01-static.yml`