

Manual de instalación de paquetes necesarios para el funcionamiento del sistema mediante

Ansible

Máximo Castro, Federico González, Santiago Hornos, Wenten Viere

Escuela Superior de Informática

Sistemas Operativos III

Profesor Walter Domínguez

28 de junio de 2024



Índice

Resumen	3
¿Qué es Ansible?	3
Requisitos	3
Descarga y configuración	3
WSL	3
Ansible	4
SSH	4
Python	5
Instalar ssh Rocky Linux	5
Ansible	8
Apache web	9
Configurar ip estática por Ansible	15
Referencias	19

Resumen

Este documento está destinado al profesor de Sistemas Operativos III, Walter Domínguez, de la Escuela Superior de Informática.. Explicaremos el proceso a la hora de la instalación y configuración de los paquetes para la máquina virtual, junto con la manera de realizarlos mediante ansible para automatizar el proceso en futuras máquinas.

¿Qué es Ansible?

Ansible es una plataforma de automatización (automation platform) soportada por Red Hat. Permite configurar sistemas, desplegar software y ejecutar tareas de informática más avanzadas, como despliegues continuos o actualizaciones permanentes sin tiempo de inactividad.

Requisitos

- Sistema Operativo
- Python
- SSH
- WinRM (para nodos Windows)

Descarga y configuración

WSL

En mi caso la workstation en la que administrare Ansible estará instalada en un windows 10 por lo que necesito WSL (Windows Subsystem for Linux) donde instalare ubuntu para administrar Ansible y en el futuro Docker.

Para instalar WSL accederemos a Windows powershell con permisos de administrador y ejecutaremos el siguiente comando:

```
PS C:\WINDOWS\system32> wsl --install
```

wsl --install

Luego de instalar WSL necesitaremos una distribución para esta en mi caso utilizaré ubuntu, lo descargo desde microsoft store y lo inicio.

Ansible

Antes de empezar a instalarlo nos aseguramos de que esté actualizado con los siguientes comandos.

```
maximo@DESKTOP-OG1FK9Q:~$ sudo apt upgrade
[sudo] password for maximo:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
maximo@DESKTOP-OG1FK9Q:~$ sudo apt install software-properties-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
software-properties-common is already the newest version (0.99.22.8).
software-properties-common set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

sudo apt upgrade

sudo apt install software-properties-common

Ahora añadimos el Personal Package Archive (PPA) de Ansible a la lista de repositorios de software de tu sistema y lo descargamos.

```
maximo@DESKTOP-OG1FK9Q:~$ sudo add-apt-repository --yes --update ppa:ansible/ansible
```

sudo add-apt-repository --yes --update ppa:ansible/ansible

```
maximo@DESKTOP-OG1FK9Q:~$ sudo apt install ansible
```

sudo apt install ansible

Asegúrate de presionar "y" para aceptar.

SSH

Ahora instalaremos el servicio ssh para conectar nuestra estación de trabajo con el servidor.

```
maximo@DESKTOP-OG1FK9Q:~$ sudo apt update
```

sudo apt update

```
maximo@DESKTOP-OG1FK9Q:~$ sudo apt install openssh-server
```

sudo apt install openssh-server

Habilitamos el servidor ssh.

```
maximo@DESKTOP-OG1FK9Q:~$ sudo systemctl enable ssh
```

sudo systemctl enable ssh

Y lo iniciamos

```
maximo@DESKTOP-OG1FK9Q:~$ sudo systemctl start ssh
```

sudo systemctl start ssh

Python

Simplemente tenemos que verificar si nuestra distribución tiene instalado python.

```
maximo@DESKTOP-OG1FK9Q:~$ sudo apt update
```

sudo apt update

```
maximo@DESKTOP-OG1FK9Q:~$ sudo apt install python3
```

sudo apt install python3

Instalar ssh Rocky Linux

Utilizamos sudo dnf update para actualizar los paquetes y procedemos a instalar el servidor ssh.

```
[administrador@192 ~]$ sudo dnf install -y openssh-server
```

sudo dnf install -y openssh-server

Habilitamos el servicio ssh.

```
[administrador@192 ~]$ sudo systemctl enable sshd
```

sudo systemctl enable sshd

Lo iniciamos.

```
[administrador@192 ~]$ sudo systemctl start sshd
```

sudo systemctl start sshd

Y verificamos.

```
[administrador@192 ~]$ sudo systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: ena>
   Active: active (running) since Sun 2024-06-09 16:50:14 -03; 14min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
  Main PID: 968 (sshd)
    Tasks: 1 (limit: 10900)
   Memory: 1.5M
      CPU: 18ms
   CGroup: /system.slice/sshd.service
           └─968 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

jun 09 16:50:13 localhost.localdomain systemd[1]: Starting OpenSSH server daemo>
jun 09 16:50:14 localhost.localdomain sshd[968]: Server listening on 0.0.0.0 po>
jun 09 16:50:14 localhost.localdomain sshd[968]: Server listening on :: port 22.
jun 09 16:50:14 localhost.localdomain systemd[1]: Started OpenSSH server daemon.
lines 1-16/16 (END)
```

sudo systemctl status sshd

Si tienes un firewall habilitado en tu sistema, necesitarás permitir el tráfico SSH. Puedes hacerlo con

```
[administrador@192 ~]$ sudo firewall-cmd --permanent --add-service=ssh
Warning: ALREADY_ENABLED: ssh
success
```

```
sudo firewall-cmd --permanent --add-service=ssh
```

```
[administrador@192 ~]$ sudo firewall-cmd --reload
success
```

```
sudo firewall-cmd --reload
```

Antes de conectar las máquinas es recomendable crear llaves ssh para la seguridad de nuestros sistemas.

```
maximo@DESKTOP-OG1FK9Q:~$ ssh-keygen -t rsa -b 4096 -C "maximocastroalzugaray@gmail.com"
```

ssh-keygen -t rsa -b 4096 -C "(email identificador sirve cualquier nombre)"

```
Generating public/private rsa key pair.
Enter file in which to save the key (/home/maximo/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/maximo/.ssh/id_rsa
Your public key has been saved in /home/maximo/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:er2tyd3gGP9cLmJ/jV5EN+nmAYVw8fw52wLIDSL0a+Y maximocastroalzugaray@gmail.com
The key's randomart image is:
+---[RSA 4096]---+
  | .oo. |
  | ..+ .|
  | .    . =O|
  | o . . . o.=|
  | o S o . *O|
  | o = .O.=|
  | . = + . .+=|
  | = . Xo+O=O|
  | E *O==*O.|
+----[SHA256]-----+
```

Ahora copiaremos la clave a el servidor Rocky Linux

```
maximo@DESKTOP-OG1FK9Q:~$ ssh-copy-id administrador@192.168.222.129
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/maximo/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
administrador@192.168.222.129's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'administrador@192.168.222.129'"
and check to make sure that only the key(s) you wanted were added.

maximo@DESKTOP-OG1FK9Q:~$
```

ssh-copy-id (nombre del usuario)@(ip de la maquina)

Para asegurarte de que puedes conectarte al servidor Rocky Linux mediante SSH, intenta conectarte desde otra máquina (o desde la misma máquina si tiene la funcionalidad de cliente SSH) usando:

```
ssh usuario@direccion_ip_rocky_linux
```

En mi caso lo haré desde el WSL.

```
maximo@DESKTOP-OG1FK9Q:~$ ssh administrador@192.168.222.129
The authenticity of host '192.168.222.129 (192.168.222.129)' can't be established.
ED25519 key fingerprint is SHA256:3vml2xJaP9zAW3o6moH0WCUEI3TdAkSc8w0MUUGnB0.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.222.129' (ED25519) to the list of known hosts.
administrador@192.168.222.129's password:
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Sun Jun  9 16:51:01 2024
[administrador@192 ~]$
```

ssh (nombre usuario)@(ip)

Conexion exitosa

Ahora crearemos una clave ssh exclusivamente para el uso de ansible. Hay que tener cuidado a la hora de seleccionar la ruta de la llave porque podemos acabar sobrescribiendo la anterior.

```
maximo@DESKTOP-OG1FK9Q:~$ ssh-keygen -t ed25519 -C "ansible"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/maximo/.ssh/id_ed25519): /home/maximo/.ssh/ansible
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/maximo/.ssh/ansible
Your public key has been saved in /home/maximo/.ssh/ansible.pub
The key fingerprint is:
SHA256:ISi4vGofi2yfw3fbFLAraGuMqxXRPwWjaPjF67siebo ansible
The key's randomart image is:
+--[ED25519 256]--+
|      o           |
| o + o o          |
| o = * ..o        |
|. + + o oo.       |
|..o . o.S.        |
| .o.  .. .        |
| o=+.. . .        |
|====o..o          |
|EO=B=.....       |
+----[SHA256]-----+
```

ssh-keygen -t ed25519 -C "ansible"

Ahora agregamos la llave al servidor

```
maximo@DESKTOP-OG1FK9Q:~$ ssh-copy-id -i ~/.ssh/ansible.pub administrador@192.168.222.129
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/maximo/.ssh/ansible.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
Enter passphrase for key '/home/maximo/.ssh/id_rsa':

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'administrador@192.168.222.129'"
and check to make sure that only the key(s) you wanted were added.

maximo@DESKTOP-OG1FK9Q:~$
```

ssh-copy-id i ~/.ssh/ansible.pub (nombre usuario)@(ip)

Si queremos evitar tener que ingresar la passphrase de la ssh key del servidor cada vez que ingresemos podemos utilizar estos comandos para acelerar el proceso.

```

maximo@DESKTOP-OG1FK9Q:~$ eval $(ssh-agent)
Agent pid 3343
maximo@DESKTOP-OG1FK9Q:~$ ps aux | grep 3343
maximo      3343  0.0  0.0   7972  1076 ?        Ss   17:47   0:00 ssh-agent
maximo      3345  0.0  0.0   4024  2176 pts/0    R+   17:47   0:00 grep --color=auto 3343
maximo@DESKTOP-OG1FK9Q:~$ ssh-add
Enter passphrase for /home/maximo/.ssh/id_rsa:
Identity added: /home/maximo/.ssh/id_rsa (maximocastroalzugaray@gmail.com)
maximo@DESKTOP-OG1FK9Q:~$ ssh administrador@192.168.222.129
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Sun Jun  9 17:45:44 2024 from 192.168.222.1
[administrador@192 ~]$

```

```

eval $(ssh-agent)
ps aux | grep 3343
ssh-add
ssh (nombre)@(ip)

```

Ansible

Luego de todo eso podemos empezar a utilizar ansible, primero debemos crear el inventario de ansible.

El archivo de inventario de Ansible es donde defines los hosts que deseas gestionar. Crea un archivo llamado **hosts** o **inventory** en tu directorio de trabajo.

```
maximo@DESKTOP-OG1FK9Q:~$ nano ~/inventory
```

```
nano ~/inventory
```



```

maximo@DESKTOP-OG1FK9Q: ~
GNU nano 6.2 /home/maximo/inventory *
[rocky_servers]_
rocky ansible_host=192.168.222.129 ansible_user=administrador

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Un
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line M-E Re

```

[rocky_servers]

rocky ansible_host=(ip maquina) ansible_user=(usuario)

Luego de escribir esa configuración usamos ctrl+x para salir y enter para guardar.

Y utilizamos nuestro primer comando de ansible que es un ping para comprobar la conexión con el servidor.

```

maximo@DESKTOP-OG1FK9Q:~$ ansible -i ~/inventory rocky_servers -m ping
rocky | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
maximo@DESKTOP-OG1FK9Q:~$

```

ansible -i ~/inventory rocky_servers -m ping

Apache web

Hacemos un archivo yum que contiene las instrucciones para descargar e iniciar apache.

Podemos verificar cualquier playbook de ansible con este comando por si tenemos dudas de la sintaxis: ansible-playbook --check playbook.yml

```
maximo@DESKTOP-OG1FK9Q:~$ nano install_apache.yml
```

```
nano install_apache.yml
```

```
---
- name: Instalar Apache HTTP Server Rocky Linux
  hosts: rocky_servers
  become: yes
  tasks:
    - name: Instalar Apache HTTPS Server
      ansible.builtin.package:
        name: httpd
        state: present

    - name: Iniciar servicio Apache
      ansible.builtin.service:
        name: httpd
        state: started
        enabled: yes
```

```
---
- name: Instalar Apache HTTP Server Rocky Linux
  hosts: rocky_servers
  become: yes
  tasks:
    - name: Instalar Apache HTTPS Server
      ansible.builtin.package:
        name: httpd
        state: present

    - name: Iniciar servicio Apache
      ansible.builtin.service:
        name: httpd
        state: started
        enabled: yes
```

Creo un archivo que contiene configuraciones para el uso de ansible, una de ellas que cuando use un comando sudo me pida antes la contraseña.

```
maximo@DESKTOP-OG1FK9Q:~$ nano ansible.cfg
```

```
nano ansible.cfg
```

```

maximo@DESKTOP-OG1FK9Q: ~
GNU nano 6.2
[defaults]
inventory = inventory

[privilege_escalation]
become = true
become_method = sudo
become_ask_pass = true

```

[defaults]
inventory = inventory

[privilege_escalation]
become = true
become_method = sudo
become_ask_pass = true

Ejecuto el archivo para instalar apache.

```
maximo@DESKTOP-OG1FK9Q:~$ ansible-playbook -i inventory install_apache.yml
```

ansible-playbook -i inventory install_apache.yml

```

maximo@DESKTOP-OG1FK9Q:~$ ansible-playbook -i inventory install_apache.yml
BECOME password:

PLAY [Instalar Apache HTTP Server Rocky Linux] *****
TASK [Gathering Facts] *****
Enter passphrase for key '/home/maximo/.ssh/id_rsa':
ok: [rocky]

TASK [Instalar Apache HTTPS Server] *****
ok: [rocky]

TASK [Iniciar servicio Apache] *****
ok: [rocky]

PLAY RECAP *****
rocky                : ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

maximo@DESKTOP-OG1FK9Q:~$

```

ansible-playbook -i inventory install_apache.yml

Creo un archivo para verificar el apache server



```

GNU nano 6.2                                check_ap
---
name: Verificar apache server
hosts: rocky_servers
become: yes
tasks:
  - name: Verificar el estado del servicio apache
    ansible.builtin.command:
      cmd: systemctl status httpd
    register: apache_status

  - name: Mostrar estado de apache
    ansible.builtin.debug:
      var: apache_status.stdout_

```

```

---
- name: Verificar apache server
  hosts: rocky_servers
  become: yes
  tasks:
    - name: Verificar el estado del servicio apache
      ansible.builtin.command:
        cmd: systemctl status httpd
      register: apache_status

    - name: Mostrar estado de apache
      ansible.builtin.debug:
        var: apache_status.stdout

```

Desde el rocky linux cambio las configuraciones del firewall para que permita la comunicación con el puerto 80 que es el puerto del apache web server.

```

Last login: Sun Jun  9 23:13:14 2024 from 192.168.222.1
[administrador@192 ~]$ sudo firewall-cmd --zone=public --add-port=80/tcp --permanent
[sudo] password for administrador:
success
[administrador@192 ~]$ sudo firewall-cmd --reload
success
[administrador@192 ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Active: active (running) since Sun 2024-06-09 22:43:01 -03; 33min ago
     Docs: man:httpd.service(8)
  Main PID: 6666 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec:
    Tasks: 177 (limit: 10900)
   Memory: 34.2M
      CPU: 1.211s
   CGroup: /system.slice/httpd.service
           └─6666 /usr/sbin/httpd -DFOREGROUND
             └─6739 /usr/sbin/httpd -DFOREGROUND
               └─6740 /usr/sbin/httpd -DFOREGROUND
                 └─6741 /usr/sbin/httpd -DFOREGROUND

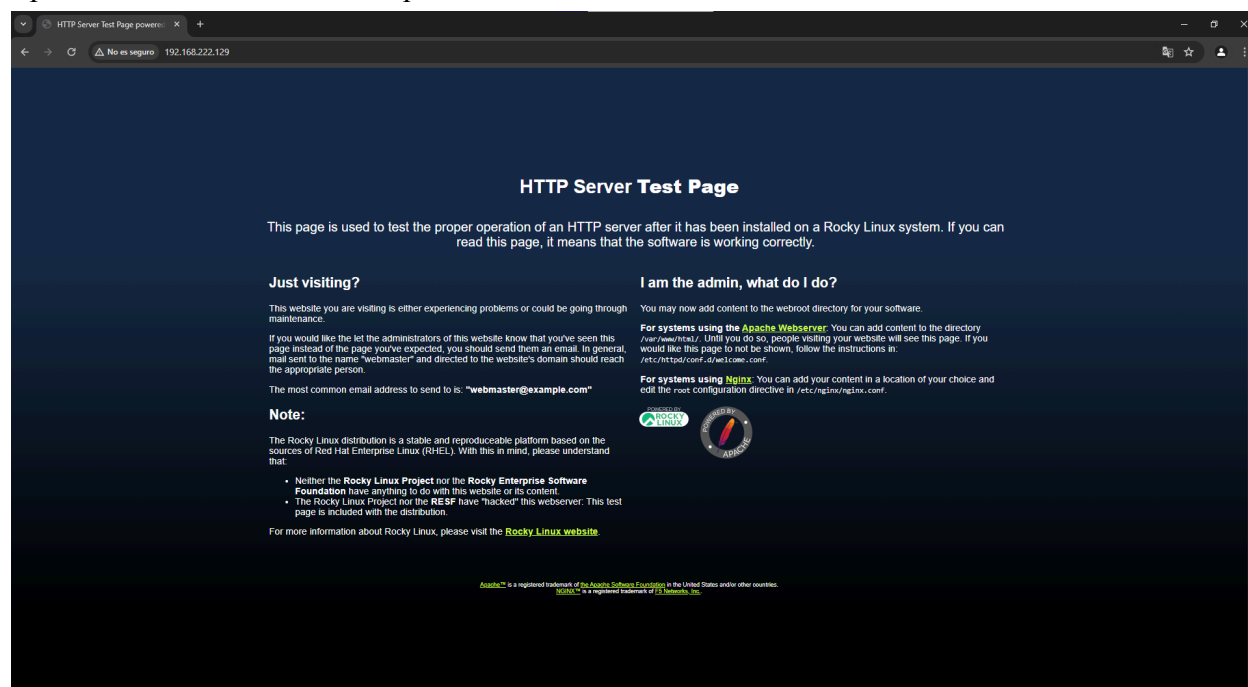
```

sudo firewall-cmd --zone=public --add-port=80/tcp --permanent

sudo firewall-cmd --reload

sudo systemctl status httpd

Accedo desde una web a la ip del servidor rocky linux y esta la pestaña predeterminada del Apache web server indicando que la instalación fue correcta.



My sql

```
maximo@DESKTOP-OG1FK9Q:~$ nano install_mysql.yml
```

nano install_mysql.yml

```
maximo@DESKTOP-OG1FK9Q: ~
GNU nano 6.2                                install_mysql.yml
---
- name: Instalar MySQL en Rocky
  hosts: rocky_servers
  become: yes
  tasks:
    - name: Instalar repositorio mysql
      ansible.builtin.yum:
        name: mysql-server
        state: present

    - name: Instalar Mysql server
      ansible.builtin.yum:
        name: mysql-server
        state: present

    - name: Iniciar y habilitar Mysql
      ansible.builtin.service:
        name: mysqld
        state: started
        enabled: yes

    - name: Obtener contraseña temporal de root
      ansible.builtin.shell:
        cmd: grep 'temporary password' /var/log/mysqld.log | awk '{print $NF}'
        register: mysql_root_temp_password

    - name: Mostrar contraseña temporal de root
      ansible.builtin.debug:
        var: mysql_root_temp_password.stdout

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo     M-A Set Mark M-] To Bracket
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E Redo     M-G Copy     ^Q Where Was
[ Read 29 lines ]
```

```
---
- name: Instalar MySQL en Rocky
  hosts: rocky_servers
  become: yes
  tasks:
    - name: Instalar Mysql server
      ansible.builtin.yum:
        name: mysql-server
        state: present

    - name: Iniciar y habilitar Mysql
      ansible.builtin.service:
        name: mysqld
        state: started
        enabled: yes
```

- name: Obtener contraseña temporal de root
 ansible.builtin.shell:
 cmd: grep 'temporary password' /var/log/mysql.log | awk '{print \$NF}'
 register: mysql_root_temp_password
- name: Mostrar contraseña temporal de root
 ansible.builtin.debug:
 var: mysql_root_temp_password.stdout

```
maximo@DESKTOP-OG1FK9Q:~$ ansible-playbook -i inventory install_mysql.yml
BECOME password:

PLAY [Instalar MySQL en Rocky] *****
TASK [Gathering Facts] *****ok: [rocky]
TASK [Instalar repositorio mysql] *****ok: [rocky]
TASK [Instalar Mysql server] *****ok: [rocky]
TASK [Iniciar y habilitar Mysql] *****ok: [rocky]
TASK [Obtener contraseña temporal de root] *****changed: [rocky]
TASK [Mostrar contraseña temporal de root] *****ok: [rocky] => {
  "mysql_root_temp_password.stdout": ""
}

PLAY RECAP *****rocky
: ok=6  changed=1  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0

maximo@DESKTOP-OG1FK9Q:~$
```

ansible-playbook -i inventory install_mysql.yml

Configurar ip estática por Ansible

Antes accederemos al servidor de rocky y utilizaremos el comando

ip a

Esto mostrará todas las interfaces de red disponibles. Busca la interfaz de red que está conectada y que tiene una dirección IP asignada.

```

[administrador@192 ~]$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens160: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:0c:29:0b:99:fb brd ff:ff:ff:ff:ff:ff
    altname enp3s0
    inet 192.168.222.129/24 brd 192.168.222.255 scope global dynamic noprefixroute ens160
        valid_lft 1751sec preferred_lft 1751sec
    inet6 fe80::20c:29ff:fe0b:99fb/64 scope link noprefixroute
        valid_lft forever preferred_lft forever

```

En mi caso se llama ens160.

```

maximo@DESKTOP-OG1FK9Q:~$ nano configure_static_ip.yml

```

nano configure_static_ip.yml

```

GNU nano 6.2                                configure_static_ip.yml *
- name: Configurar IP fija en Rocky Linux
  hosts: rocky_servers
  become: yes
  tasks:
    - name: Instalar NetworkManager si no está presente
      ansible.builtin.yum:
        name: NetworkManager
        state: present

    - name: Configurar IP fija
      ansible.builtin.copy:
        dest: /etc/sysconfig/network-scripts/ifcfg-ens160 #Pondremos el nombre de la interfaz que vimos antes con ip
        content: |
          TYPE=Ethernet
          PROXY_METHOD=none
          BROWSER_ONLY=no
          BOOTPROTO=none
          DEFROUTE=yes
          IPV4_FAILURE_FATAL=no
          IPV6INIT=yes
          IPV6_AUTOCONF=yes
          IPV6_DEFROUTE=yes
          IPV6_FAILURE_FATAL=no
          IPV6_ADDR_GEN_MODE=stable-privacy
          NAME=ens160 #Aqui ponemos el nombre de la interfaz de rocky
          DEVICE=ens160 #Lo mismo aqui
          ONBOOT=yes
          IPADDR=192.168.222.129
          PREFIX=24
          GATEWAY=192.168.222.1
          DNS1=8.8.8.8
          DNS2=8.8.4.4

    - name: Reiniciar NetworkManager
      ansible.builtin.systemd:
        name: NetworkManager
        state: restarted

```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark M-I To Bracket
 ^X Exit ^R Read File ^V Replace ^U Paste ^D Justify ^_ Go To Line M-E Redo M-G Copy ^J Where Was

- name: Configurar IP fija en Rocky Linux

hosts: rocky_servers

become: yes

tasks:

- name: Instalar NetworkManager si no está presente

ansible.builtin.yum:

name: NetworkManager

state: present

- name: Configurar IP fija

ansible.builtin.copy:

dest: /etc/sysconfig/network-scripts/ifcfg-ens160 # NOMBRE DE MI INTERFAZ

content: |

TYPE=Ethernet

PROXY_METHOD=none

BROWSER_ONLY=no

BOOTPROTO=none

DEFROUTE=yes

IPV4_FAILURE_FATAL=no

IPV6INIT=yes

IPV6_AUTOCONF=yes

IPV6_DEFROUTE=yes

```

IPV6_FAILURE_FATAL=no

IPV6_ADDR_GEN_MODE=stable-privacy

NAME=ens160 #ATENCIÓN NOMBRE INTERFAZ ROCKY VERIFICA EL SUYO
DEVICE=ens160 #ATENCIÓN NOMBRE DE MI INTERFAZ VERIFICA EL SUYO
ONBOOT=yes

IPADDR=192.168.222.129

PREFIX=24

GATEWAY=192.168.222.1

DNS1=8.8.8.8

DNS2=8.8.4.4

```

- name: Reiniciar NetworkManager

```
ansible.builtin.systemd:
```

```
  name: NetworkManager
```

```
  state: restarted

```

Explicación:

NetworkManager es una utilidad que facilita la administración de conexiones de red en sistemas operativos Linux y otros sistemas basados en Unix.

```
ansible.builtin.copy:
```

```
  dest: /etc/sysconfig/network-scripts/ifcfg-enp0s3

```

Esto crea un archivo y lo pega en esa dirección

Cada línea dentro de **content** tiene un propósito específico para la configuración de la interfaz de red:

- **TYPE=Ethernet:** Especifica que el tipo de conexión es Ethernet.
- **PROXY_METHOD=none:** Indica que no se usará un proxy para esta conexión.
- **BROWSER_ONLY=no:** Especifica que esta conexión no es solo para navegación web.
- **BOOTPROTO=none:** Deshabilita el protocolo de arranque automático, ya que estamos configurando una IP estática.
- **DEFROUTE=yes:** Esta interfaz debe ser utilizada como la ruta por defecto.
- **IPV4_FAILURE_FATAL=no:** Indica que si la configuración IPv4 falla, no se debe considerar como un fallo crítico.
- **IPV6INIT=yes:** Habilita IPv6 para esta interfaz.
- **IPV6_AUTOCONF=yes:** Permite la autoconfiguración de IPv6.
- **IPV6_DEFROUTE=yes:** Esta interfaz debe ser utilizada como la ruta por defecto para IPv6.
- **IPV6_FAILURE_FATAL=no:** Indica que si la configuración IPv6 falla, no se debe considerar como un fallo crítico.
- **IPV6_ADDR_GEN_MODE=stable-privacy:** Usa direcciones IPv6 de privacidad estable.
- **NAME=enp0s3:** Nombre de la interfaz.
- **DEVICE=enp0s3:** Nombre del dispositivo de red.
- **ONBOOT=yes:** Habilita la interfaz para activarse al inicio del sistema.
- **IPADDR=192.168.222.129:** Dirección IP estática que se asignará a la interfaz.

- **PREFIX=24:** Longitud del prefijo de la máscara de subred, equivalente a una máscara de subred de **255.255.255.0**.
- **GATEWAY=192.168.222.1:** Dirección IP de la puerta de enlace predeterminada.
- **DNS1=8.8.8.8:** Dirección IP del servidor DNS primario (en este caso, uno de los servidores DNS públicos de Google).

```
maximo@DESKTOP-OG1FK9Q:~$ ansible-playbook -i inventory configure_static_ip.yml --ask-become-pass
BECOME password:

PLAY [Configurar IP fija en Rocky Linux] *****
TASK [Gathering Facts] *****
Enter passphrase for key '/home/maximo/.ssh/id_rsa':
ok: [rocky]
TASK [Instalar NetworkManager si no está presente] *****
ok: [rocky]
TASK [Configurar IP fija] *****
changed: [rocky]
TASK [Reiniciar NetworkManager] *****
changed: [rocky]
PLAY RECAP *****
rocky                : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

ansible playbook -i inventory configure_static_ip.yml --ask-become-pass

Ahora vamos a verificar en el rocky

```
[administrador@192 ~]$ sudo reboot
```

sudo reboot

```
[administrador@localhost ~]$ ip addr show ens160
2: ens160: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group d
fault qlen 1000
    link/ether 00:0c:29:0b:99:fb brd ff:ff:ff:ff:ff:ff
    altname enp3s0
    inet 192.168.222.129/24 brd 192.168.222.255 scope global noprefixroute ens1
0
        valid_lft forever preferred_lft forever
    inet6 fe80::f11d:9f59:33ba:f63b/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
[administrador@localhost ~]$
```

ip addr show (nombre interfaz configurada)

```
maximo@DESKTOP-OG1FK9Q:~$ ping -c4 192.168.222.129
PING 192.168.222.129 (192.168.222.129) 56(84) bytes of data.
64 bytes from 192.168.222.129: icmp_seq=1 ttl=63 time=0.405 ms
64 bytes from 192.168.222.129: icmp_seq=2 ttl=63 time=0.396 ms
64 bytes from 192.168.222.129: icmp_seq=3 ttl=63 time=0.381 ms
64 bytes from 192.168.222.129: icmp_seq=4 ttl=63 time=0.479 ms

--- 192.168.222.129 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3139ms
rtt min/avg/max/mdev = 0.381/0.415/0.479/0.037 ms
maximo@DESKTOP-OG1FK9Q:~$
```

Finalmente hacemos un ping desde la estación de trabajo a el server y es exitoso.

Automatización de ansible

Con todos los playbook creados podemos usarlos para facilitar la configuración a futuro de otras máquinas virtuales, hay que tener en cuenta que se puede dificultar el mantenimiento a futuro por eso es importante que esté bien documentado.

Playbook de ansible que instala y configura todos los paquetes necesarios:

```
maximo@DESKTOP-OG1FK9Q:~$ nano configure_rocky.yml
```

nano configure_rocky.yml

```
GNU nano 6.2                                configure_rocky.yml
--
- name: Configurar Rocky Linux
  hosts: rocky_servers
  become: yes
  tasks:
    - name: Instalar Apache HTTPS Server
      ansible.builtin.package:
        name: httpd
        state: present

    - name: Iniciar servicio Apache
      ansible.builtin.service:
        name: httpd
        state: started
        enabled: yes

    - name: Instalar MySQL server
      ansible.builtin.yum:
        name: mysql-server
        state: present

    - name: Iniciar y habilitar MySQL
      ansible.builtin.service:
        name: mysqld
        state: started
        enabled: yes

    - name: Obtener contraseña temporal de root
      ansible.builtin.shell:
        cmd: grep 'temporary password' /var/log/mysqld.log | awk '{print $NF}'
      register: mysql_root_temp_password

    - name: Mostrar contraseña temporal de root
      ansible.builtin.debug:
        var: mysql_root_temp_password.stdout

    - name: Configurar IP fija
      ansible.builtin.copy:
        dest: /etc/sysconfig/network-scripts/ifcfg-ens160
        content: |
          TYPE=Ethernet
          PROXY_METHOD=none
          BROWSER_ONLY=no
          BOOTPROTO=none
          DEFROUTE=yes
          IPV4_FAILURE_FATAL=no
          IPV6INIT=yes
          IPV6_AUTOCONF=yes
          IPV6_DEFROUTE=yes
          IPV6_FAILURE_FATAL=no
          IPV6_ADDR_GEN_MODE=stable-privacy
          NAME=ens160
          DEVICE=ens160
          ONBOOT=yes
          IPADDR=192.168.222.129
          PREFIX=24
          GATEWAY=192.168.222.1
```

```
          GATEWAY=192.168.222.1
          DNS1=8.8.8.8

    - name: Reiniciar NetworkManager
      ansible.builtin.systemd:
        name: NetworkManager
        state: restarted
```

```
---
```

- name: Configurar Rocky Linux

hosts: rocky_servers

become: yes

tasks:

- name: Instalar Apache HTTP Server

ansible.builtin.package:

name: httpd

state: present

- name: Iniciar y habilitar servicio Apache

ansible.builtin.service:

name: httpd

state: started

enabled: yes

- name: Instalar MySQL server

ansible.builtin.yum:

name: mysql-server

state: present

- name: Iniciar y habilitar MySQL

ansible.builtin.service:

name: mysqld

state: started

enabled: yes

- name: Obtener contraseña temporal de root

ansible.builtin.shell:

cmd: grep 'temporary password' /var/log/mysqld.log | awk '{print \$NF}'

register: mysql_root_temp_password

- name: Mostrar contraseña temporal de root

ansible.builtin.debug:

var: mysql_root_temp_password.stdout

- name: Configurar IP fija

ansible.builtin.copy:

dest: /etc/sysconfig/network-scripts/ifcfg-ens160

content: |

TYPE=Ethernet

PROXY_METHOD=none

BROWSER_ONLY=no

BOOTPROTO=none

DEFROUTE=yes

IPV4_FAILURE_FATAL=no

IPV6INIT=yes


```
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens160
DEVICE=ens160
ONBOOT=yes
IPADDR=192.168.222.129
PREFIX=24
GATEWAY=192.168.222.1
DNS1=8.8.8.8
```

- name: Reiniciar NetworkManager

ansible.builtin.systemd:

name: NetworkManager

state: restarted

```
maximo@DESKTOP-OG1FK9Q:~$ ansible-playbook -i inventory configure_rocky.yml --ask-become-pass
```

ansible-playbook -i inventory configure_rocky.yml --ask-become-pass

```

maximo@DESKTOP-OG1FK9Q:~$ ansible-playbook -i inventory configure_rocky.yml --ask-become-pass
BECOME password:

PLAY [Configurar Rocky Linux] *****
TASK [Gathering Facts] *****
ok: [rocky]
TASK [Instalar Apache HTTPD Server] *****
ok: [rocky]
TASK [Iniciar servicio Apache] *****
ok: [rocky]
TASK [Instalar MySQL server] *****
ok: [rocky]
TASK [Iniciar y habilitar MySQL] *****
ok: [rocky]
TASK [Obtener contraseña temporal de root] *****
changed: [rocky]
TASK [Mostrar contraseña temporal de root] *****
ok: [rocky] => {
  "mysql_root_temp_password.stdout": ""
}
TASK [Configurar IP fija] *****
changed: [rocky]
TASK [Reiniciar NetworkManager] *****
changed: [rocky]
PLAY RECAP *****
rocky                : ok=9   changed=3   unreachable=0   failed=0   skipped=0   rescued=0   ignored=0

maximo@DESKTOP-OG1FK9Q:~$

```

```

#!/bin/bash
#Esto es para que se interprete por Bash

set -eu -o pipefail # -e hace que el script falle si un comando devuelve error, -u falla el script si se inten>

#verifico permisos de sudo
if ! sudo -n true 2>/dev/null; then
    echo " Debes tener permisos de sudo para usar el script, usa su -"
    exit 1
fi

echo "Bienvenido al script de instalacion de Mysql,SSH y Apache Web"
sleep 3
echo "Actualizaremos los paquetes del sistema antes de instalar los otros"
sleep 3
sudo dnf update -y
while read -r paquete ; do sudo dnf install $paquete -y ; done < <(cat << "FIN"
    mysql-server
    openssh-server
    httpd
FIN
)
echo Instalando prerequisites...
echo Espere a que se instale o utilice Ctrl+C para cancelar
sleep 5
echo Instalacion completa

[ línea 13/42 (30%), col 1/8 (12%), car 473/1290 (36%) ]
^G Ayuda      ^O Guardar    ^W Buscar     ^K Cortar     ^T Ejecutar   ^C Ubicación  M-U Deshacer
^X Salir      ^R Leer fich. ^\ Reemplazar ^U Pegar      ^J Justificar ^_ Ir a línea  M-E Rehacer

```

```

while read -r paquete ; do sudo dnf install $paquete -y ; done < (cat << "FIN"
mysql-server
openssh-server
httpd
FIN
)
echo Instalando prerequisites...
echo Espere a que se instale o utilize Ctrl+C para cancelar
sleep 5
echo Instalacion completa
sleep 3
echo Iniciar servicio ssh automaticamente al iniciar el sistema
sleep 6
sudo systemctl start sshd
sudo systemctl enable sshd
echo Iniciar servicio mysql automaticamente al iniciar el sistema
sleep 7
sudo systemctl start mysqld
sudo systemctl enable mysqld
echo Iniciar Apache Web server
sleep 4
sudo systemctl start httpd
sudo systemctl enable httpd
echo "Script Finalizado puede verificar las descargas con sudo status sshd,mysqld,httpd"
sleep 10

```

^G Ayuda ^O Guardar ^W Buscar ^K Cortar ^T Ejecutar ^C Ubicación M-U Deshacer
 ^X Salir ^R Leer fich ^\ Reemplazar ^J Pegar ^I Justificar ^_ Ir a línea M-F Rehacer

Referencias

“Getting started with Ansible”. Learn Linux TV. Obtenido el 10 de junio de 2024 de

<https://www.youtube.com/playlist?list=PLT98CRI2KxKEUHie1m24-wkyHpEsa4Y70>