

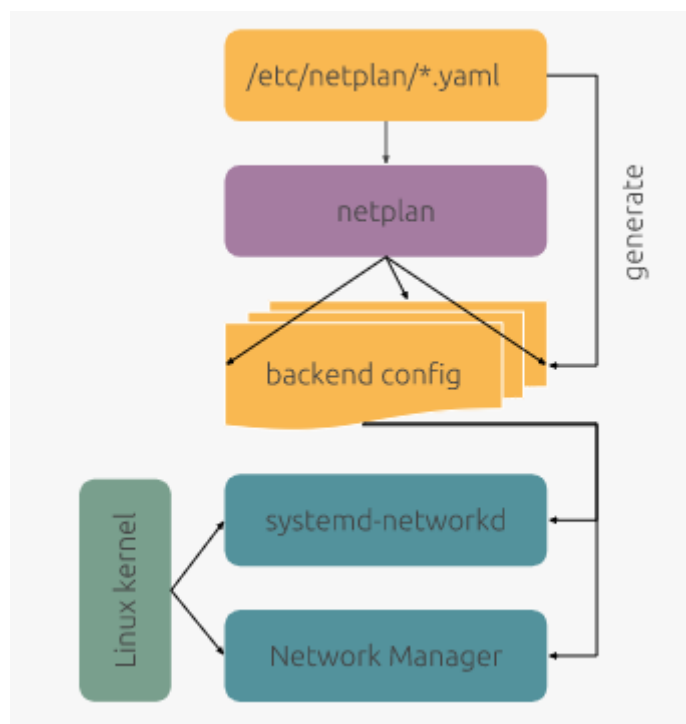


PRACTICA 7 – CONEXIÓN EN RED MODO NAT DE LOS EQUIPOS

OBJETIVOS

- Configurar cada uno de los adaptadores de red
- Cada alumno tendrá sus equipos en la red 192.168.x.nn /24 . Cambiando x por su nº de clase. Mascara 255.255.255.0
- Paso previo: Definir la red NAT : le pondremos de nombre Red-ASO. No DHCP
- Captura pantalla configuración
- Configuración UbuntuServer (NetPlan)

INTRODUCCIÓN CONFIGURACIÓN DE RED UBUNTU SERVER 24.



¿Qué es NETPLAN?

Netplan se usa en Ubuntu para configurar la red. Proporciona una manera simplificada de definir la configuración de red mediante archivos YAML. Con Netplan, puedes configurar interfaces de red, definir direcciones IP, puertas de enlace, DNS y más, facilitando la gestión de la red en sistemas basados en Ubuntu.



Establecer una dirección IP estática en Ubuntu Server 24.04

Antes de 2017, Ubuntu Server basaba su configuración de red en los parámetros almacenados en el archivo de configuración **/etc/network/interfaces**, una estructura heredada directamente de Debian.

Sin embargo, a partir de la versión 17.10, Canonical introduce una nueva herramienta de configuración de red llamada **NetPlan**, que pretende facilitar el trabajo a desarrolladores y usuarios en general.

El funcionamiento de **NetPlan** se basa en un archivo con la descripción de los adaptadores de red que necesitemos definir. El archivo estará escrito en **YAML** (siglas de YAML Ain't Markup Language), un archivo de texto plano con un formato específico para la serialización que resulta fácilmente comprensible para las personas.

Por lo tanto, a partir de la versión 17.10, deberemos utilizar archivos de configuración con extensión **yaml**, almacenados en la ruta **/etc/netplan**.

De forma predeterminada, sólo tenemos un archivo que, en nuestro caso, se llama **50-cloud-init.yaml**.

La configuración inicial de NetPlan

Para comprobar el valor inicial que tiene el archivo **50-cloud-init.yaml**, basta con escribir una orden como esta:

```
sudo cat /etc/netplan/50-cloud-init.yaml
```

En versiones anteriores de Ubuntu, podíamos ver (aunque no cambiar) el contenido de este archivo sin usar privilegios administrativos. Sin embargo, ahora, sí que los necesitamos. Por eso hemos escrito la orden **sudo** por delante de **cat**.

Contenido del archivo 50-cloud-init.yaml.

En el caso de Ubuntu Server, la configuración predeterminada de NetPlan será similar a esta:

```
network:  
ethernets:  
  enp0s3:  
    dhcp4: true  
version: 2
```

```
root@ubsrv04:~# cat /etc/netplan/50-cloud-init.yaml  
network:  
  version: 2  
  ethernets:  
    enp0s3:  
      dhcp4: true  
root@ubsrv04:~# _
```

Explicado de forma sencilla, lo que establece es que el adaptador de red que estamos definiendo es **enp0s3** y que utilizará DHCP. Es decir, que su dirección IP será asignada desde otro dispositivo de nuestra red, normalmente el router que nos facilita



el acceso a Internet, que actuará como servidor DHCP.

En ocasiones, también puede aparecer una línea más, parecida a esta:

renderer: networkd

Esto indicará que el demonio ***systemd-networkd*** es el encargado de administrar la red.

Establecer una dirección IP fija en NetPlan

La mayoría de las veces, un servidor necesita disponer de una dirección IP fija. Imagina, por ejemplo, que quieres redireccionar algún puerto desde el router, usar SSH para administrar el servidor de forma remota, o alguna tarea por el estilo.

En estos casos, cuando utilicemos Ubuntu Server 24.04 LTS, deberemos editar el archivo de configuración correspondiente (en nuestro ejemplo, 50-cloud-init.yaml) y realizar las modificaciones oportunas.

Para lograrlo, puedes usar el editor de textos nano:

sudo nano /etc/netplan/50-cloud-init.yaml

Como de costumbre, escribimos el comando y pulsamos la tecla Intro.

Una vez que nos encontramos en la ventana de trabajo de nano, sustituiremos su valor por algo como esto:

```
network:
  ethernets:
    enp0s3:
      addresses:
        - 192.168.1.10/24
      nameservers:
        addresses:
          - 208.67.222.222
      routes:
        - to: default
          via: 192.168.1.1
  version: 2
```

Obviamente, los valores concretos tendrán que adaptarse a tus necesidades pero, para facilitarte las cosas, a continuación te explicamos cada una de sus partes:

- **addresses:** Establece que, en mi red, la dirección IP de equipo será siempre 192.168.1.10.

También indica que la máscara de red utiliza 24 bits (lo que equivale a 255.255.255.0). Este valor también puede cambiar según las características de tu red.

- **nameservers:** Configura la dirección (o direcciones) de Internet donde se



encuentra el servidor DNS que usaremos para convertir los nombres de dominio en direcciones IP. En este caso utilizaremos el servidor primario de OpenDNS.

Si necesitáramos incluir más de una dirección, bastaría con duplicar la línea, como muestra este ejemplo:

nameservers:

addresses:

- 208.67.222.222

- 208.67.220.220

A partir de Ubuntu 22.04 LTS se implementó una nueva versión de NetPlan, que propuso la desaparición de gateway4 y gateway6, aunque seguirán funcionando por un tiempo. Sin embargo, lo recomendable es que, desde ahora, en su lugar comencemos a utilizar routes que, por cada ruta definida establece dos valores:

- to, para indicar a ruta que debe utilizarse.
- via, para establecer la dirección IP por la que hacerlo.

Cuando hayamos completado la configuración, estaremos listos para salir de la edición del archivo.

Para lograrlo, sólo tenemos que pulsar la combinación de teclas **Control + X**.

Aplicar los cambios

De vuelta en el prompt del sistema, estaremos listos para aplicar los cambios del archivo de configuración. Algo que podemos conseguir ejecutando el siguiente comando:

sudo netplan apply

Escribimos el comando y pulsamos la tecla Intro.

Si el comando no ofrece ninguna información de salida es que todo ha funcionado correctamente. No obstante, para comprobar que los cambios han surtido efecto, podemos consultar la configuración actual de la red:

ip addr

Como antes, escribimos el comando y pulsamos la tecla Intro.

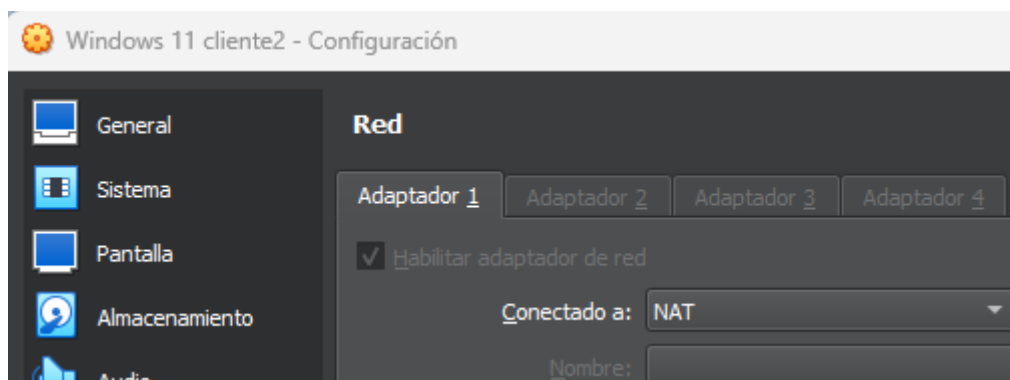
```
root@ubsrv04:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq
    link/ether 08:00:27:a4:25:b4 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.10/24 brd 192.168.1.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fea4:25b4/64 scope link
        valid_lft forever preferred_lft forever
```



SECUENCIA/DESARROLLO

1. W10Clixx:

1. W10Clixx debe tener configurado 1 tarjeta de red Nat



2. Configuración IP: 192.168.0.50. Mascara + Puerta de enlace

Captura pantalla de detalles de la conexión de red

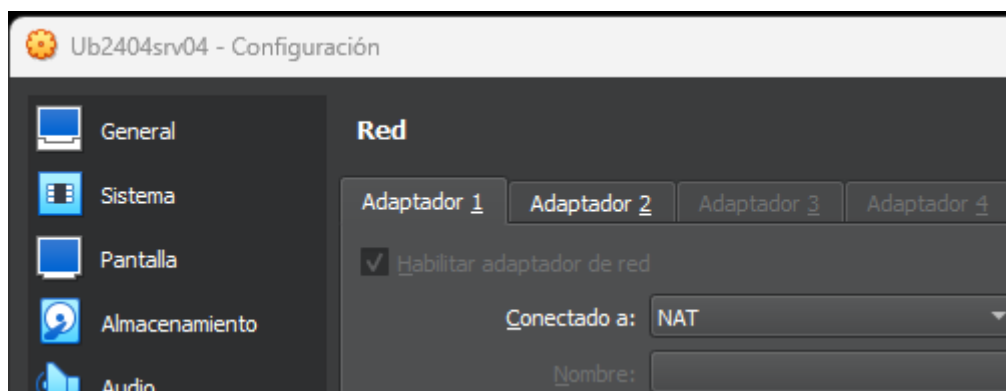
3. El comando para comprobar la configuración de la red es **ipconfig**

```
Adaptador de Ethernet Ethernet:

Suíjo DNS específico para la conexión. . . :
Vínculo: dirección IPv6 local. . . : fe80::6171:502d:abd7:39f2%4
Dirección IPv4. . . . . : 192.168.0.50
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . . : 192.168.0.11
```

2. **Ubserverxx:** *En usr/share/doc/netplan/examples → Hay ejemplos de configuración*

1. Ubserver debe tener configurado 1 tarjeta de red Nat



2. Configuración IP: 192.168.0.11. Mascara + Puerta de enlace

Captura pantalla del archivo de configuración de la red.

3. El comando para comprobar la configuración de la red es **ip a** **Captura pantalla**



```

root@ubsrv04:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue
    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 noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500
    link/ether 08:00:27:a4:25:b4 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.10/24 brd 192.168.1.255 scope glob
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fea4:25b4/64 scope link
        valid_lft forever preferred_lft forever

```

4. La utilidad para configurar fácilmente la red en ubuntu es **netplan** Y se basa en archivos **50-cloud-init.yaml**
5. ¿Que comando utilizo para que una vez cambiada la configuración de la red, se apliquen los nuevos valores?

netplan apply

3. Probar conectividad mediante comandos Pings:

1. Prueba conectividad:hacer ping de Wserverxx al equipo cliente. (y viceversa) .
Anota cambios que hay que hacer para que funcione

```

root@ubsrv04:~# netplan apply
root@ubsrv04:~# ping 192.168.0.50
PING 192.168.0.50 (192.168.0.50) 56(84) bytes of data.
From 192.168.0.11 icmp_seq=1 Destination Host Unreachable
From 192.168.0.11 icmp_seq=2 Destination Host Unreachable
From 192.168.0.11 icmp_seq=3 Destination Host Unreachable
^C

```

no funciona porque dos tarjetas redes de los equipos son de modo NAT, está configuración va dividir dos equipos en diferente red. Entonces necesitamos que configurar a modo Red Interna or al menos en misma red.

2. Captura pantallas con respuestas de estos pings.

Ping desde ubuntu server al Windows cliente:

```

root@ubsrv04:~# ping 192.168.0.50
PING 192.168.0.50 (192.168.0.50) 56(84) bytes of data.
64 bytes from 192.168.0.50: icmp_seq=1 ttl=128 time=0.891 ms
64 bytes from 192.168.0.50: icmp_seq=2 ttl=128 time=0.833 ms
64 bytes from 192.168.0.50: icmp_seq=3 ttl=128 time=0.871 ms
64 bytes from 192.168.0.50: icmp_seq=4 ttl=128 time=0.833 ms
64 bytes from 192.168.0.50: icmp_seq=5 ttl=128 time=0.848 ms
^C64 bytes from 192.168.0.50: icmp_seq=6 ttl=128 time=0.846 ms
64 bytes from 192.168.0.50: icmp_seq=7 ttl=128 time=2.27 ms
^C
--- 192.168.0.50 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6013ms
rtt min/avg/max/mdev = 0.833/1.056/2.272/0.496 ms
root@ubsrv04:~#

```



Desde Windows cliente al ubuntu server:

```
C:\Users\usuario>ping 192.168.0.11

Haciendo ping a 192.168.0.11 con 32 bytes de datos:
Respuesta desde 192.168.0.11: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.0.11: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.0.11: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.0.11: bytes=32 tiempo=1ms TTL=64

Estadísticas de ping para 192.168.0.11:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 1ms, Media = 0ms

C:\Users\usuario>
```