

ROUTER

Interfaz ens18

Propósito: Conexión al exterior de la red (generalmente la salida hacia Internet o redes externas).

1. **dhcp4: false**
 - No usa DHCP, por lo que su dirección IP y otros parámetros son configurados manualmente.
2. **addresses:**
 - Tiene una dirección IP estática 100.77.20.38/24. Esto significa que pertenece a la red 100.77.20.0 con máscara de subred /24 (255.255.255.0).
3. **nameservers:**
 - Falta configuración de servidores DNS en esta sección. Esto podría ser un error o simplemente no está especificado aquí.
4. **routes:**
 - Una ruta predeterminada se define con to: default, indicando que el tráfico que no tiene una ruta específica en la tabla de enrutamiento se enviará a través del gateway 100.77.20.1.

Conclusión:

La interfaz ens18 está configurada como la salida hacia el exterior (posiblemente Internet), y todo el tráfico saliente que no pertenece a la red local será dirigido al gateway 100.77.20.1.

Interfaz ens19

Propósito: Conexión hacia una red interna.

1. **dhcp4: false**
 - Configuración manual (sin uso de DHCP).
2. **addresses:**
 - Se le asigna la dirección IP estática 10.20.40.1/24. Esto indica que pertenece a la red interna 10.20.40.0 con máscara de subred /24 (255.255.255.0).
3. **nameservers:**
 - Define dos servidores DNS:
 - 10.20.40.1: Generalmente un servidor DNS interno.
 - 8.8.8.8: DNS público de Google.
4. **routes:**
 - Aunque aparece el atributo routes, no hay rutas adicionales específicas configuradas para esta interfaz.

Conclusión:

La interfaz ens19 está configurada para servir como la conexión hacia la red interna, posiblemente para comunicación con dispositivos o servicios locales en la red 10.20.40.0/24.

```

GNU nano 6.2 /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
# Interfaz de ens18 para comunicase al exterior de la Red
network:
  version: 2
  ethernets:
    ens18:
      dhcp4: false
      addresses:
        - 100.77.20.38/24
      nameservers:
        addresses:
          - 8.8.8.8
      routes:
        - to: default
          via: 100.77.20.1

#####
# Interfaz de ens19 para comunicase a la Red Interna

    ens19:
      dhcp4: false
      addresses:
        - 10.20.40.1/24
      nameservers:
        addresses:
          - 10.20.40.1
          - 8.8.8.8
#      routes:
#        - to: default
#          via: 100.77.20.1

```

Este código configura un servidor DHCP para asignar automáticamente direcciones IP en el rango 10.20.40.10 - 10.20.40.100 dentro de la subred 10.20.40.0/24, especificando parámetros como máscara de subred, gateway (10.20.40.1), DNS (10.20.40.17), dominio (rbooks.com), y tiempos de concesión de IP.

#}

A slightly different configuration for an internal subnet.

```
subnet 10.20.40.0 netmask 255.255.255.0 {  
    range 10.20.40.10 10.20.40.100;  
    option domain-name-servers 10.20.40.17;  
    option domain-name "rbooks.com";  
    option subnet-mask 255.255.255.0;  
    option routers 10.20.40.1;  
    option broadcast-address 10.20.40.255;
```

```
    default-lease-time 600;  
    max-lease-time 7200;  
    ddns-update-style interim;  
    ddns-updates on;  
    ddns-domainname "rbooks.com.";   
    ddns-rev-domainname "in-addr.arpa.";   
    update-static-leases on;  
    ignore client-updates;
```

```
    authoritative;  
    option domain-name "rbooks.com";
```

}

Iremos a cada MV para ir al `/etc/netplan/00-installer-config.yaml` para poder configurar las interfaces de red en los sistemas que usan Netplan:

DNS

```
GNU nano 6.2 /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  version: 2
  renderer: networkd
  ethernets:
    ens18:
      dhcp4: true
      # addresses: [10.20.40.11/24]
      # nameservers:
      #   addresses: [10.20.40.11, 9.9.9.9, 8.8.8.8]
      #   search: [rbooks.local]
      routes:
        - to: default
          via: 10.20.40.1
```

Cliente

```
GNU nano 6.2 /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    ens18:
      dhcp4: true
#   addresses:
#     - 10.20.40.10/24
#   nameservers:
#     addresses:
#       - 8.8.8.8
#       - 8.8.4.4
#     search: []
#   routes:
#     - to: default
#       via: 10.20.40.1
  version: 2
```

Web

```
GNU nano 6.2 /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    ens18:
      dhcp4: true
#     addresses: [10.20.40.13/24]
#     nameservers:
#       addresses: [10.20.40.13, 9.9.9.9, 8.8.8.8]
#       search: [rbooks.local]
      routes:
        - to: default
          via: 10.20.40.1
  version: 2
```

BBDD

```
GNU nano 6.2 /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    ens18:
      dhcp4: true
#     addresses: [10.20.40.12/24]
#     nameservers:
#       addresses: [10.20.40.12, 9.9.9.9, 8.8.8.8]
#       search: [rbooks.local]
      routes:
        - to: default
          via: 10.20.40.1
  version: 2
```

En el HW del Router le ponemos el dispositivo de red que creamos el vmbr1. En netplan creamos la red interna y externa.

```
GNU nano 6.2 /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
# Interfaz de ens18 para comunicase al exterior de la Red
network:
  version: 2
  ethernets:
    ens18:
      dhcp4: false
      addresses:
        - 100.77.20.38/24
      nameservers:
        addresses:
          - 8.8.8.8
      routes:
        - to: default
          via: 100.77.20.1

#####
# Interfaz de ens19 para comunicase a la Red Interna

    ens19:
      dhcp4: false
      addresses:
        - 10.20.40.1/24
      nameservers:
        addresses:
          - 10.20.40.1
          - 8.8.8.8
#      routes:
#        - to: default
#          via: 100.77.20.1
```

Y le hacemos un "netplan apply" para guarda los cambios que hicimos.

Luego añadimos la dirección nano /etc/sysctl.conf y Quitamos el "#" en la línea donde se encuentra "net.ipv4.ip_forward=1". Y le ponemos los cabios con el "sysctl -p"

Después agregamos sudo apt install iptables.

Cuando termine podemos comprobar que no tenemos ninguna regla habilitada, por supuesto: iptables -L iptables -t nat -L

Ahora configuramos una regla de iptables como se muestra a continuación. iptables -t nat -A POSTROUTING -o ens18 -j MASQUERADE

Hacemos esta comprobación:

```
root@router:~# iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target     prot opt source                destination

Chain INPUT (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source                destination
MASQUERADE all  --  anywhere              anywhere
root@router:~#
```

Luego le hacemos un "sudo iptables-save" y luego le instalamos "sudo apt install iptables-persistent -y"

Redirige el tráfico entrante en el puerto 80 (HTTP) al servidor Nginx en la IP interna

- `iptables -t nat -A PREROUTING -p tcp -d 100.77.20.38 --dport 80 -j DNAT --to-destination 10.20.40.16:80`

Del mismo modo redirigimos el tráfico entrante en el puerto 443 (HTTPS) al servidor Nginx en la IP interna

- `iptables -t nat -A PREROUTING -p tcp -d 100.77.20.38 --dport 443 -j DNAT --to-destination 10.20.40.16:443`