

Para el examen LPIC-2, objetivo 205.1, "Configuración básica de red", se espera que domines la configuración de interfaces de red para establecer conectividad tanto a redes locales (cableadas e inalámbricas) como a redes de área amplia (WAN). Además, debes ser capaz de configurar la comunicación entre diferentes subredes utilizando tanto IPv4 como IPv6.

A continuación, se presenta un documento de estudio detallado que cubre las áreas clave y las utilidades mencionadas:

Documento de Estudio: LPIC-2 Objetivo 205.1 - Configuración Básica de Red

****Peso del Objetivo:** 3**

Descripción General

Este objetivo se enfoca en la capacidad de un administrador de sistemas Linux para configurar interfaces de red para la conectividad local y de área amplia. Esto incluye el manejo de configuraciones IPv4 e IPv6, la gestión de tablas de enrutamiento y la configuración de redes inalámbricas básicas.

Áreas de Conocimiento Clave

1. ****Utilidades para configurar y manipular interfaces de red Ethernet:****

- * Comprender la diferencia y el uso de `ip` (la herramienta moderna y preferida) y las herramientas "legacy" como `ifconfig` y `route`.

- * Configuración de direcciones IP estáticas y dinámicas (DHCP).

- * Configuración de máscaras de red, gateways predeterminados y servidores DNS.

- * Habilitar/deshabilitar interfaces.

- * Gestión de la tabla de enrutamiento.

- * Configuración de subinterfaces (interfaces virtuales).

2. ****Configurar el acceso básico a redes inalámbricas:****

- * Identificar redes inalámbricas disponibles.

- * Conectarse a redes inalámbricas con y sin cifrado.

- * Configurar interfaces inalámbricas con utilidades de línea de comandos.

3. ****Comunicación entre subredes (IPv4 e IPv6):****

- * Comprender el concepto de subredes y cómo el enrutamiento permite la comunicación entre ellas.

- * Configurar rutas estáticas para alcanzar subredes remotas.

- * Activación del reenvío de paquetes (IP forwarding) para que un sistema actúe como un router.

Lista Parcial de Archivos, Términos y Utilidades

A continuación, se detalla el uso y la relevancia de cada uno:

1\ Herramientas de Configuración de Red (Ethernet e IP)

* **`ip` (IP utility - Preferred)**

* La herramienta moderna y unificada para la configuración de redes.

* **Sintaxis general:** `ip [OPTIONS] OBJECT { COMMAND | help }`

* **Objetos comunes:**

* `link`: Gestiona dispositivos de red (interfaces).

* `ip link show`: Muestra información sobre las interfaces de red.

* `ip link set <device> up/down`: Habilita/deshabilita una interfaz.

* `ip link set <device> promisc on/off`: Activa/desactiva el modo promiscuo.

* `address` (o `a`): Gestiona direcciones IP.

* `ip a show`: Muestra direcciones IP asignadas a interfaces.

* `ip a add <IP/MASK> dev <device>`: Asigna una dirección IP a una interfaz.

* `ip a del <IP/MASK> dev <device>`: Elimina una dirección IP de una interfaz.

* `route` (o `r`): Gestiona la tabla de enrutamiento.

* `ip r show`: Muestra la tabla de enrutamiento.

* `ip r add default via <gateway_IP> dev <device>`: Configura el gateway por defecto.

* `ip r add <network/mask> via <gateway_IP> dev <device>`: Añade una ruta estática a una red específica.

* `ip r del default`: Elimina el gateway por defecto.

* `neigh` (o `n`): Gestiona la tabla ARP/NDP (vecinos).

* `ip n show`: Muestra la tabla de vecinos.

* **`ifconfig` (Interface Configuration - Legacy)**

* Más antigua, pero aún ampliamente utilizada en muchos scripts y sistemas.

* **Uso:**

* `ifconfig`: Muestra la configuración de todas las interfaces.

* `ifconfig <interface>`: Muestra la configuración de una interfaz específica.

* `ifconfig <interface> <IP_address> netmask <netmask> up/down`: Asigna una IP y máscara, y habilita/deshabilita la interfaz.

* `ifconfig <interface> mtu <value>`: Establece la MTU (Maximum Transmission Unit).

* **`route` (IP Routing Table Utility - Legacy)**

* Se utiliza para ver y manipular la tabla de enrutamiento del kernel.

* **Uso:**

* `route -n`: Muestra la tabla de enrutamiento numéricamente (útil para evitar resolución DNS).

* `route add default gw <gateway_IP>`: Añade el gateway por defecto.

* `route add -net <network> netmask <netmask> gw <gateway_IP>`: Añade una ruta a una red específica.

* `route del default gw <gateway_IP>`: Elimina el gateway por defecto.

* **`arp` (Address Resolution Protocol utility - Legacy)**

* Muestra y manipula la caché ARP del sistema. ARP mapea direcciones IP a direcciones MAC.

* **Uso:**

* `arp -a`: Muestra todas las entradas en la caché ARP.

* `arp -d <IP_address>`: Elimina una entrada específica de la caché ARP.

2\. Configuración de Redes Inalámbricas

* **`iw` (Wireless Devices - Preferred)**

* La herramienta moderna para configurar y manipular dispositivos inalámbricos basados en nl80211`.

* **Sintaxis general:**`iw [OPTIONS] COMMAND`

* **Comandos comunes:**

* `iw dev`: Muestra dispositivos inalámbricos.

* `iw dev <wlan_device> scan`: Escanea redes inalámbricas disponibles.

* `iw dev <wlan_device> link`: Muestra el estado de la conexión.

* `iw dev <wlan_device> connect <SSID> [key 0:abcde]`: Conecta a una red WEP (no recomendada).

* `iw dev <wlan_device> disconnect`: Desconecta de la red actual.

* **`iwconfig` (Wireless Configuration - Legacy)**

* Similar a `ifconfig` pero para interfaces inalámbricas.

* **Uso:**

* `iwconfig`: Muestra la configuración de todas las interfaces inalámbricas.

* `iwconfig <interface> essid <SSID>`: Establece el SSID de la red.

* `iwconfig <interface> key <hex_key> / open / restricted`: Configura la clave WEP.

* `iwconfig <interface> mode Ad-Hoc / Managed`: Establece el modo de operación.

* **`iwlist` (List Wireless Parameters - Legacy)**

* Se utiliza para escanear redes y obtener información detallada sobre las capacidades inalámbricas.

* **Uso:**

* `iwlist <interface> scan`: Escanea redes inalámbricas y muestra detalles (SSID, canal, cifrado, etc.).

* `iwlist <interface> frequency`: Muestra las frecuencias disponibles.

* `iwlist <interface> channel`: Muestra los canales.

* `iwlist <interface> rate`: Muestra las tasas de bits disponibles.

3\ Archivos de Configuración de Red Persistente

Es crucial entender que la configuración de red realizada con comandos como `ip` o `ifconfig` es **temporal** y se pierde al reiniciar el sistema. Para una configuración persistente, se deben editar archivos de configuración específicos de la distribución:

* **Debian/Ubuntu (y derivados):**

* `/etc/network/interfaces`: Archivo principal para la configuración manual de interfaces (en versiones más antiguas o si no se usa Netplan).

* `/etc/netplan/*.yaml`: Archivos de configuración YAML utilizados por Netplan (el gestor de red por defecto en versiones recientes de Ubuntu).

* Ejemplo básico de Netplan para una interfaz estática:

```
``yaml
```

```
network:
```

```
  version: 2
```

```
  renderer: networkd
```

```
  ethernet:
```

```
    enp0s3:
```

```

    dhcp4: no
    addresses: [192.168.1.100/24]
    gateway4: 192.168.1.1
    nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
    ...

```

* Para subinterfaces en Netplan:

```

``yaml
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: no
      addresses: [192.168.1.10/24]
  bonds:
    mybond:
      interfaces: [enp0s3] # Ejemplo, subinterfaces se manejan como alias

```

scope de la interfaz

```

      parameters:
        mode: active-backup

```

Para una subinterfaz virtual (alias) en Netplan, se declara en la sección de la interfaz

principal:

```

# enp0s3:
#   addresses:
#     - 192.168.1.10/24
#     - 192.168.1.11/24 # Esto actúa como un alias
...

```

*** **Red Hat/Rocky Linux/CentOS (y derivados):****

* `/etc/sysconfig/network-scripts/ifcfg-<interface_name>`: Archivos de configuración para interfaces individuales.

* Ejemplo para una interfaz estática (`ifcfg-eth0`):

```

...
TYPE=Ethernet
BOOTPROTO=none
NAME=eth0
DEVICE=eth0
ONBOOT=yes
IPADDR=192.168.1.100
PREFIX=24
GATEWAY=192.168.1.1
DNS1=8.8.8.8
...

```

* Para subinterfaces (alias) en Red Hat/Rocky, se crea un archivo `ifcfg-eth0:0` (o similar):

```

...
TYPE=Ethernet
BOOTPROTO=none
NAME=eth0:0
DEVICE=eth0:0
ONBOOT=yes

```

```
IPADDR=192.168.10.1
```

```
PREFIX=24
```

```
# No se especifica GATEWAY en subinterfaces a menos que sea una ruta específica.  
...
```

* `NetworkManager`: El gestor de red por defecto. Sus configuraciones se guardan en `/etc/NetworkManager/system-connections/` y se pueden manipular con `nmcli` o `nmtui`.

4\ Reenvío de Paquetes (IP Forwarding)

Para que un sistema Linux actúe como un router y reenvíe tráfico entre diferentes subredes, el reenvío de paquetes debe estar habilitado.

* **Activación temporal (hasta el reinicio):**

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

O para IPv6:

```
echo 1 > /proc/sys/net/ipv6/conf/all/forwarding
```

* **Activación permanente:**

* Editar `/etc/sysctl.conf` o añadir un archivo en `/etc/sysctl.d/` (ej. `/etc/sysctl.d/99-ip-forward.conf`).

* Añadir o descomentar las siguientes líneas:

```
...
```

```
net.ipv4.ip_forward = 1
```

```
net.ipv6.conf.all.forwarding = 1
```

* Aplicar los cambios sin reiniciar con:

```
sudo sysctl -p
```

Escenarios de Configuración Típicos

1. **Configuración de IP estática (IPv4):**

* Identificar la interfaz (`ip link show`).

* Asignar dirección IP y máscara (`sudo ip a add 192.168.1.10/24 dev eth0`).

* Configurar gateway (`sudo ip r add default via 192.168.1.1 dev eth0`).

* Configurar DNS (editar `/etc/resolv.conf` temporalmente, o configurarlo en los archivos persistentes del gestor de red).

* Habilitar la interfaz (`sudo ip link set eth0 up`).

* Hacerlo persistente según la distribución.

2. **Configuración de DHCP (IPv4):**

- * En Debian/Ubuntu: Asegurarse de que la interfaz en `/etc/network/interfaces` esté configurada como `dhcp` (o `dhcp4: yes` en Netplan).
- * En Red Hat/Rocky: `BOOTPROTO=dhcp` en `ifcfg-eth0`.
- * Reiniciar el servicio de red o la interfaz.

3. **Configuración de IP estática (IPv6):**

- * Asignar dirección IPv6 y máscara (`sudo ip a add 2001:db8::10/64 dev eth0`).
- * Configurar gateway IPv6 (`sudo ip -6 r add default via fe80::1 dev eth0`).
- * Hacerlo persistente.

4. **Configuración de subinterfaces (alias):**

- * **Con `ip` (temporal):** `sudo ip a add 192.168.10.1/24 dev eth0:0` (el `eth0:0` es una convención, `ip` simplemente lo trata como una IP adicional en la interfaz).
- * **Persistente:** Ver ejemplos en la sección de archivos de configuración para Netplan y `ifcfg` files.

5. **Configuración de red inalámbrica básica:**

- * Identificar la interfaz inalámbrica (`iw dev` o `iwconfig`).
- * Escanear redes (`sudo iw dev wlan0 scan` o `sudo iwlist wlan0 scan`).
- * Conectar (si es una red abierta): `sudo ip link set wlan0 up && sudo iw dev wlan0 connect <SSID>`.
- * Obtener IP via DHCP: `sudo dhclient wlan0`.
- * Para WPA/WPA2, generalmente se usa `wpa_supplicant` junto con las utilidades `ip` o `iw`. `iw` por sí solo no gestiona las complejas autenticaciones de WPA/WPA2.

Puntos Clave para Recordar

- * **`ip` vs. `ifconfig`/`route`:
- * Aunque `ifconfig` y `route` aún funcionan, `ip` es la herramienta preferida y más potente para la configuración de red en Linux. Familiarízate con `ip`.
- * **Persistencia:** Siempre piensa en cómo hacer que los cambios de red sean persistentes después de un reinicio. Esto implica editar los archivos de configuración adecuados para la distribución (`/etc/network/interfaces`, Netplan, `ifcfg-*`, NetworkManager).
- * **Enrutamiento:** Comprende cómo la tabla de enrutamiento dirige el tráfico y cómo habilitar el reenvío de IP para que un sistema actúe como un router.
- * **Diagnóstico:** Después de cualquier cambio, verifica la configuración con `ip a`, `ip r`, `ping` y `traceroute` para asegurarte de que la conectividad es la esperada.

Este documento proporciona una base sólida para el objetivo 205.1. Practicar con una máquina virtual es esencial para consolidar estos conocimientos.