

¡Estupendo! Hemos concluido la administración avanzada de dispositivos de almacenamiento del Examen 204 de LPIC-2. Ahora, volvemos al ámbito de las redes, profundizando en los fundamentos vistos en LPIC-1. El "Examen 205" según tu lista se centra en la configuración de red.

Comenzaremos con la configuración de red básica a un nivel LPIC-2, revisando y detallando los métodos de configuración persistente en las diferentes ramas de distribución y con las herramientas modernas. Destacaremos las importantes diferencias en los archivos y comandos de configuración.

Comencemos con el objetivo 205.1.

LPIC-2 / Examen 205 - Configuración de Red

Este examen (en la nomenclatura oficial de LPI, estos objetivos están bajo el Examen 201) cubre la configuración de red, la resolución de problemas y la configuración avanzada.

205.1 Configuración de red básica

Teoría

La configuración de red en LPIC-2 va más allá de asignar una IP temporalmente. Implica comprender los diferentes métodos para configurar interfaces de red (cableadas, inalámbricas), direcciones IP (IPv4, IPv6), máscaras de subred, puertas de enlace, servidores DNS y nombres de host, asegurando que la configuración sea persistente entre reinicios.

Conceptos Clave (Revisión LPIC-1 + Profundización LPIC-2):

- **Nombres de Interfaces de Red:** Reafirmar la transición de nombres predictivos (ej: `enpXsY`, `wlpXsY`) gestionados por `udev`, en lugar de los antiguos `ethX`, `wlanX`. Aunque los nombres antiguos aún pueden verse o configurarse.
- **Direccionamiento IP:** IPv4 e IPv6. Direcciones unicast, multicast, broadcast. Clases de red (histórico), subredes y notación CIDR.
- **Puerta de Enlace Predeterminada:** El router por defecto para alcanzar redes remotas.
- **Servidores DNS:** Para la resolución de nombres a direcciones IP.

Herramientas y Archivos de Configuración Persistente (LPIC-2):

La principal diferencia entre distribuciones y la evolución del sistema de inicio (`systemd`) ha llevado a la coexistencia de varios métodos para configurar la red de forma persistente. Es crucial identificar cuál usa un sistema en particular.

1. Métodos Tradicionales (Basados en Archivos de Texto y Scripts):

- Estos métodos dependen de que los scripts de inicio (`ifup`, `ifdown`) o servicios `SysVinit/systemd` lean archivos de configuración estáticos.
- **Rama Debian/Ubuntu:**
 - Archivo principal: `/etc/network/interfaces`.

- Define "stanzas" para cada interfaz (`auto <interfaz>, allow-hotplug <interfaz>, iface <interfaz> inet <método>`).
- Métodos comunes: `dhcp`, `static`, `manual`, `loopback`.
- Opciones comunes: `address`, `netmask`, `gateway`, `dns-nameservers`, `network`, `broadcast`. También `up` y `down` para scripts personalizados al levantar/bajar la interfaz.
- Ejemplo estático:

```
auto enp3s0
iface enp3s0 inet static
    address 192.168.1.100
    netmask 255.255.255.0
    gateway 192.168.1.1
    dns-nameservers 8.8.8.8 8.8.4.4
```

- Ejemplo DHCP:

```
auto enp3s0
iface enp3s0 inet dhcp
```

- Aplicar cambios: `sudo ifdown <interfaz>` luego `sudo ifup <interfaz>`, o reiniciar servicio `networking.service` (si se usa) o reiniciar el sistema.

- **Rama Red Hat/CentOS/Fedora:**

- Archivos por interfaz en `/etc/sysconfig/network-scripts/` (ej: `ifcfg-enp3s0`).
- Formato `clave=valor`.
- Parámetros comunes: `DEVICE`, `BOOTPROTO` (`none` para estático, `dhcp`), `ONBOOT` (`yes/no`), `TYPE`, `IPADDR`, `NETMASK`, `GATEWAY`, `DNS1`, `DNS2`, `UUID`, `HWADDR`, `PEERDNS` (`yes/no` - si DHCP debe actualizar `/etc/resolv.conf`), `NM_CONTROLLED` (`yes/no` - si NetworkManager debe gestionar este archivo).
- Ejemplo estático (`/etc/sysconfig/network-scripts/ifcfg-enp3s0`):

```
TYPE=Ethernet
BOOTPROTO=none
DEFROUTE=yes
IPADDR=192.168.1.100
NETMASK=255.255.255.0
GATEWAY=192.168.1.1
DNS1=8.8.8.8
DNS2=8.8.4.4
ONBOOT=yes
```

- Ejemplo DHCP: Cambiar `BOOTPROTO=dhcp`, eliminar líneas `IPADDR`, `NETMASK`, `GATEWAY`, `DNS*`.

- Aplicar cambios: `sudo ifdown <interfaz>` luego `sudo ifup <interfaz>`, o reiniciar servicio `network.service` (si se usa) o reiniciar el sistema.

2. Métodos Modernos (Gestionados por Servicios Dinámicos):

- Estos servicios gestionan conexiones de red de forma más dinámica, a menudo interactuando con `udev` y respondiendo a eventos (conexión/desconexión). Su configuración anula o integra los métodos tradicionales.
- **NetworkManager:** Común en escritorios y muchos servidores. Demonio `NetworkManager.service`.
 - Herramientas: `nmcli` (línea de comandos, potente), `nmtui` (interfaz de texto).
 - Configuración: Almacena conexiones en archivos de configuración (ej: `/etc/NetworkManager/system-connections/`) o usa plugins para leer formatos tradicionales.
 - Aplicar cambios: Los cambios con `nmcli` o `nmtui` suelen aplicarse inmediatamente o al activar/desactivar la conexión (`nmcli connection up <nombre>`).
- **systemd-networkd:** Común en servidores o contenedores. Demonio `systemd-networkd.service`.
 - Herramienta: `networkctl`.
 - Configuración: Archivos `.network`, `.link`, `.netdev` en `/etc/systemd/network/`.
 - Aplicar cambios: `sudo systemctl restart systemd-networkd.service` o `sudo networkctl reload`.
- **netplan (Solo Ubuntu/Debian modernos):** Una capa de abstracción basada en YAML. Configuras la red en archivos `.yaml` en `/etc/netplan/`. `netplan` lee estos archivos y *genera* la configuración para el back-end (`NetworkManager` o `systemd-networkd`).
 - Ejemplo (`/etc/netplan/01-netcfg.yaml`):

YAML

```
network:
  version: 2
  renderer: networkd # o networkmanager
  ethernets:
    enp3s0:
      dhcp4: yes
      # Para estático:
      # dhcp4: no
      # addresses: [192.168.1.100/24]
      # gateway4: 192.168.1.1
      # nameservers:
      #   addresses: [8.8.8.8, 8.8.4.4]
```

- Aplicar cambios: `sudo netplan generate` (genera la config para el backend) luego `sudo netplan apply` (aplica la config, a menudo sin reiniciar).
- **Identificar el método activo:** Visto en 9.2.1, verificar qué servicio de systemd (NetworkManager, `systemd-networkd`) está activo. Si ninguno lo está, o si los archivos tradicionales tienen `ONBOOT=yes` y no `NM_CONTROLLED=yes`, es probable que se usen scripts tradicionales. La presencia de archivos en `/etc/netplan/` indica que se usa `netplan`.

3. Configuración del Hostname:

- Archivo estándar: `/etc/hostname`. Contiene solo el nombre del host.
- Comando: `hostname`. Muestra el nombre actual.
- Con systemd: `hostnamectl status` muestra más detalles. `sudo hostnamectl set-hostname <nuevo_nombre>`. Esto modifica `/etc/hostname`.
- Archivo Red Hat antiguo: `/etc/sysconfig/network` (puede contener `HOSTNAME=`).

4. Configuración del Cliente DNS (`/etc/resolv.conf`):

- Revisado en 109.4. Contiene `nameservers` y opciones `domain/search`.
- A menudo gestionado automáticamente por DHCP clients, NetworkManager o `systemd-resolved`.
- Configurar persistentemente los DNS se hace en el archivo de configuración de la interfaz correspondiente al método activo (ej: `dns-nameservers` en `/etc/network/interfaces`, `DNS1/DNS2` en `ifcfg`, `nameservers` en `netplan`, `nmcli` o `networkctl` configuración).

Verificación de la Configuración de Red:

- `ip addr show <interfaz>`: Dirección IP, máscara, estado (UP/DOWN).
- `ip route show`: Tabla de enrutamiento, puerta de enlace predeterminada.
- `cat /etc/resolv.conf`: Servidores DNS configurados.
- `ping <ip>` o `ping <hostname>`: Prueba de conectividad.
- `ss -tulnp` o `netstat -tulnp`: Puertos a la escucha.

LPIC-2 / Examen 205 - Configuración de Red - Ejercicios

*Nota: Estos ejercicios se centran en identificar y visualizar configuraciones. **Realizar cambios en la configuración de red puede interrumpir la conectividad.** Hazlo SIEMPRE en una VM de prueba con un plan de recuperación (ej: acceso a la consola) y guarda snapshots.*

Ejercicio 5.1.1: Identificando el Método de Configuración de Red Activo

- **Objetivo:** Determinar qué servicio o archivos gestionan la configuración de red persistente.
- **Requisitos:** Acceso a la línea de comandos.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal.
 2. **Verifica qué servicio de gestión de red está activo:** Ejecuta `systemctl status NetworkManager.service systemd-networkd.service`. Uno de ellos debería estar `active (running)`. Si ninguno lo está, es probable que se usen scripts tradicionales o `netplan` con otro backend.
 3. **Verifica si se usa Netplan (Solo Debian/Ubuntu modernos):** Ejecuta `ls -l /etc/netplan/`. Si existen archivos `.yaml`, se está usando Netplan. Visualiza un archivo `.yaml`: `cat /etc/netplan/*.yaml`. Busca la línea `renderer:` para ver si usa `networkd` o `NetworkManager` como backend.
 4. **Examina archivos de configuración tradicionales (si no usas NM o networkd, o para entender la coexistencia):**
 - En Debian/Ubuntu: `ls -l /etc/network/interfaces`.
 - En Red Hat/CentOS/Fedora: `ls -l /etc/sysconfig/network-scripts/ifcfg-*`.
 - Busca en los archivos tradicionales si tienen `auto (interfaces)` u `ONBOOT=yes` y `NM_CONTROLLED=no (ifcfg-*)`.

Ejercicio 5.1.2: Explorando Archivos de Configuración Persistente (Basado en el Método)

- **Objetivo:** Localizar y visualizar los archivos de configuración de red relevantes para tu sistema.
- **Requisitos:** Identificar el método activo (Ej. 5.1.1). Privilegios de superusuario (`sudo`).
- **Desarrollo Paso a Paso:**
 1. Abre una terminal.
 2. **Si usas NetworkManager:**
 - Explora el directorio de conexiones: `sudo ls -l /etc/NetworkManager/system-connections/`.
 - Visualiza un archivo de conexión: `sudo less /etc/NetworkManager/system-connections/<nombre_conexion>`. Busca secciones `[ipv4]` o `[ipv6]` y los parámetros `method`, `address1`, `gateway`, `dns`.
 3. **Si usas systemd-networkd:**
 - Explora el directorio de configuración: `ls -l /etc/systemd/network/`. Busca archivos `.network`.
 - Visualiza un archivo `.network`: `cat /etc/systemd/network/<nombre_interfaz>.network`. Busca secciones `[Network]` y `[Match]`, y los parámetros `DHCP=yes`, `Address`, `Gateway`, `DNS`.
 4. **Si usas Netplan:**

- Explora el directorio de configuración: `ls -l /etc/netplan/`. Busca archivos `.yaml`.
 - Visualiza un archivo `.yaml`: `cat /etc/netplan/<nombre_archivo>.yaml`. Observa la estructura YAML y los parámetros para tus interfaces.
5. **Si usas métodos tradicionales:**
- En Debian/Ubuntu: Visualiza `/etc/network/interfaces` (`cat /etc/network/interfaces`).
 - En Red Hat/CentOS/Fedora: Visualiza el archivo `ifcfg` para tu interfaz principal (`cat /etc/sysconfig/network-scripts/ifcfg-<interfaz>`).

Ejercicio 5.1.3: Viendo la Configuración de Hostname

- **Objetivo:** Identificar cómo está configurado el nombre del host.
- **Requisitos:** Acceso a la línea de comandos.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal.
 2. **Muestra el nombre de host actual:** Ejecuta `hostname`.
 3. **Visualiza el archivo estándar de hostname:** Ejecuta `cat /etc/hostname`.
 4. **Con systemd, usa `hostnamectl`:** Ejecuta `hostnamectl status`. Muestra el nombre de host estático, transitorio y bonito, así como el estado de `systemd-timesyncd`.
 5. **(En Red Hat, si usas un sistema antiguo):** Ejecuta `cat /etc/sysconfig/network` y busca la línea `HOSTNAME=`.

Ejercicio 5.1.4: Viendo la Configuración del Cliente DNS

- **Objetivo:** Revisar los servidores DNS configurados y cómo se gestiona `/etc/resolv.conf`.
- **Requisitos:** Acceso a la línea de comandos.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal.
 2. **Visualiza `/etc/resolv.conf`:** Ejecuta `cat /etc/resolv.conf`. Identifica las líneas `nameserver` y los comentarios que indican quién gestiona el archivo.
 3. **Verifica si es un enlace simbólico:** Ejecuta `ls -l /etc/resolv.conf`. Si apunta a un archivo bajo `/run/` o `/usr/`, es gestionado dinámicamente.
 4. **Si 127.0.0.53 está en `resolv.conf`:** Esto indica que `systemd-resolved` está activo. Verifica su estado: `systemctl status systemd-resolved.service`. Y ve los servidores DNS reales que usa: `systemd-resolve --status`.

Ejercicio 5.1.5: Usando Comandos `ip` para Verificar Configuración (Runtime)

- **Objetivo:** Usar las herramientas de línea de comandos para ver la configuración de red tal como la ve el kernel.
- **Requisitos:** Acceso a la línea de comandos.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal.
 2. **Verifica direcciones IP y estado de interfaz:** Ejecuta `ip addr show`. Compara con la configuración persistente.
 3. **Verifica la tabla de enrutamiento:** Ejecuta `ip route show`. Comprueba la puerta de enlace predeterminada y las rutas de red local.
 4. **Verifica el estado de la interfaz:** Ejecuta `ip link show <interfaz>` (ej: `ip link show enp3s0`).

Ejercicio 5.1.6: (Conceptual) Configurando una IP Estática vs. DHCP Persistente

- **Objetivo:** Entender cómo realizar la configuración persistente usando los diferentes métodos.
- **Requisitos:** Identificar el método activo. Privilegios de superusuario (sudo). VM de prueba.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal.
 2. **Si usas /etc/network/interfaces (Debian/Ubuntu):**
 - Para estático: Edita `/etc/network/interfaces`, cambia `dhcp` por `static`, añade `address`, `netmask`, `gateway`, `dns-nameservers`.
 - Para DHCP: Edita `/etc/network/interfaces`, cambia `static` por `dhcp`, elimina las líneas de IP, `gateway`, DNS.
 - Aplicar: `sudo ifdown <interfaz> && sudo ifup <interfaz>`.
 3. **Si usas /etc/sysconfig/network-scripts/ifcfg-* (Red Hat/CentOS/Fedora):**
 - Para estático: Edita `ifcfg-<interfaz>`, cambia `BOOTPROTO=dhcp` a `BOOTPROTO=none` o `static`, añade `IPADDR`, `NETMASK`, `GATEWAY`, `DNS1`.
 - Para DHCP: Edita `ifcfg-<interfaz>`, cambia `BOOTPROTO=none` o `static` a `BOOTPROTO=dhcp`, elimina las líneas de IP, `gateway`, DNS.
 - Aplicar: `sudo systemctl restart network.service` (si usa ese servicio) o `sudo ifdown <interfaz> && sudo ifup <interfaz>`.
 4. **Si usas NetworkManager (nmcli):**
 - Identifica el nombre de conexión (suele ser el mismo que la interfaz o un UUID). Ejecuta `nmcli connection show`.
 - Para estático: `sudo nmcli connection modify <nombre_conexion> ipv4.method manual ipv4.addresses`

```
"192.168.1.100/24" ipv4.gateway "192.168.1.1"
ipv4.dns "8.8.8.8 8.8.4.4".
```

- Para DHCP: `sudo nmcli connection modify <nombre_conexion> ipv4.method auto.`
- Aplicar: `sudo nmcli connection up <nombre_conexion>` (puede que necesites down primero).

5. **Si usas systemd-networkd:**

- Edita el archivo `.network` para tu interfaz en `/etc/systemd/network/`.
- Para estático: Comenta o elimina `DHCP=yes`. Añade `Address=192.168.1.100/24, Gateway=192.168.1.1, DNS=8.8.8.8 8.8.4.4.`
- Para DHCP: Elimina líneas de IP, gateway, DNS. Añade `DHCP=yes`.
- Aplicar: `sudo systemctl restart systemd-networkd.service.`

6. **Si usas Netplan:**

- Edita el archivo `.yaml` en `/etc/netplan/`.
- Para estático: Debajo de la interfaz, añade `dhcp4: no, addresses: [IP/Máscara], gateway4: Gateway, nameservers: {addresses: [DNS1, DNS2]}.`
- Para DHCP: Debajo de la interfaz, asegúrate de que `dhcp4: yes` está presente y comenta/elimina las líneas de estático.
- Aplicar: `sudo netplan apply.`

7. **(MUY IMPORTANTE en VM de prueba):** Después de realizar cambios, verifica inmediatamente la conectividad. Si pierdes la red, usa la consola de la VM para revertir los cambios.