

Práctica 1: Configuración de TCP/IP en Windows 10 y Linux

1. Introducción

Para hacer esta práctica necesitas dos entornos distintos: Windows 10 y Linux.

WINDOWS 10:

En el laboratorio, selecciona el arranque Windows 10. En casa, si no dispones de este sistema operativo, puedes instalar la máquina virtual Windows 10, a partir de un fichero .ova, que puedes obtener directamente de Microsoft desde el enlace: <https://developer.microsoft.com/es-es/microsoft-edge/tools/vms/>. La contraseña, como dice la página, es “Passw0rd!”. La máquina virtual está configurada de forma predeterminada en idioma inglés, pero puede cambiarse mediante el menú “Settings” > “Time & Language” > “Language”, añadiendo el “Local Experience Pack” adecuado y reiniciando la máquina virtual.

LINUX:

En el laboratorio, descarga e instala sobre VirtualBox la máquina virtual **Redes-2021**, disponible en [\zuria\disca\asignaturas\alumnos\gii-red\](#).

Esta práctica está dedicada a revisar el procedimiento básico de instalación y configuración de los protocolos TCP/IP en Windows 10 y Linux. Se muestra el uso de algunas herramientas útiles a la hora de resolver problemas con estos protocolos: configurar el software de red, verificar su funcionamiento y ajustar los parámetros relacionados con TCP/IP.


2. Configuración de TCP/IP en Windows 10

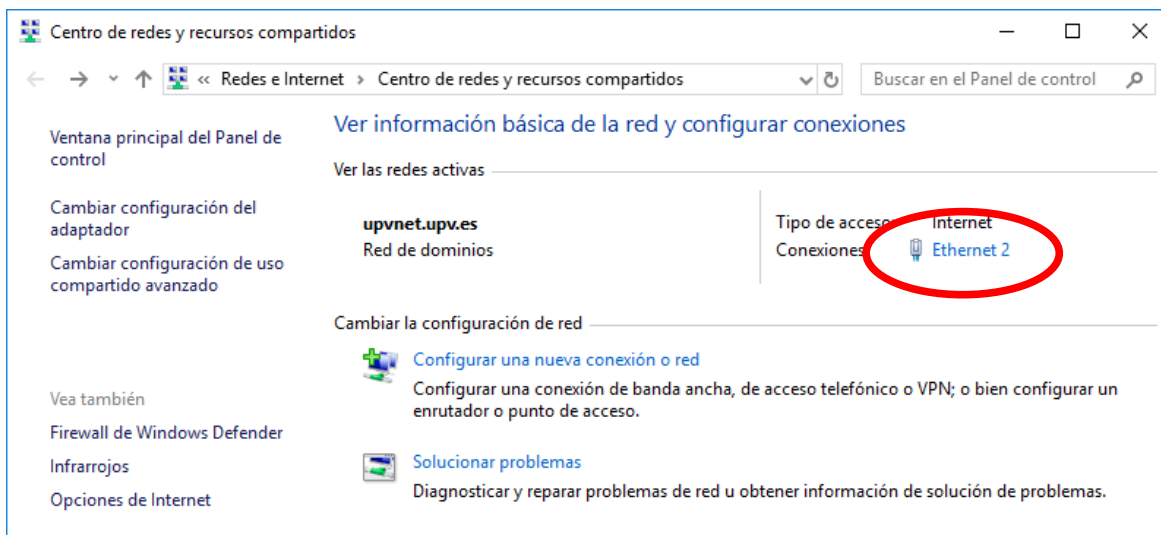
Este apartado requiere, evidentemente, iniciar en tu ordenador un entorno Windows 10.

Para utilizar los protocolos TCP/IP desde una máquina Windows 10 conectada a una red de área local es necesario tener instalada una tarjeta adaptadora o NIC (*Network Interface Adapter*). En nuestros computadores esta tarjeta ya suele estar instalada y configurada.

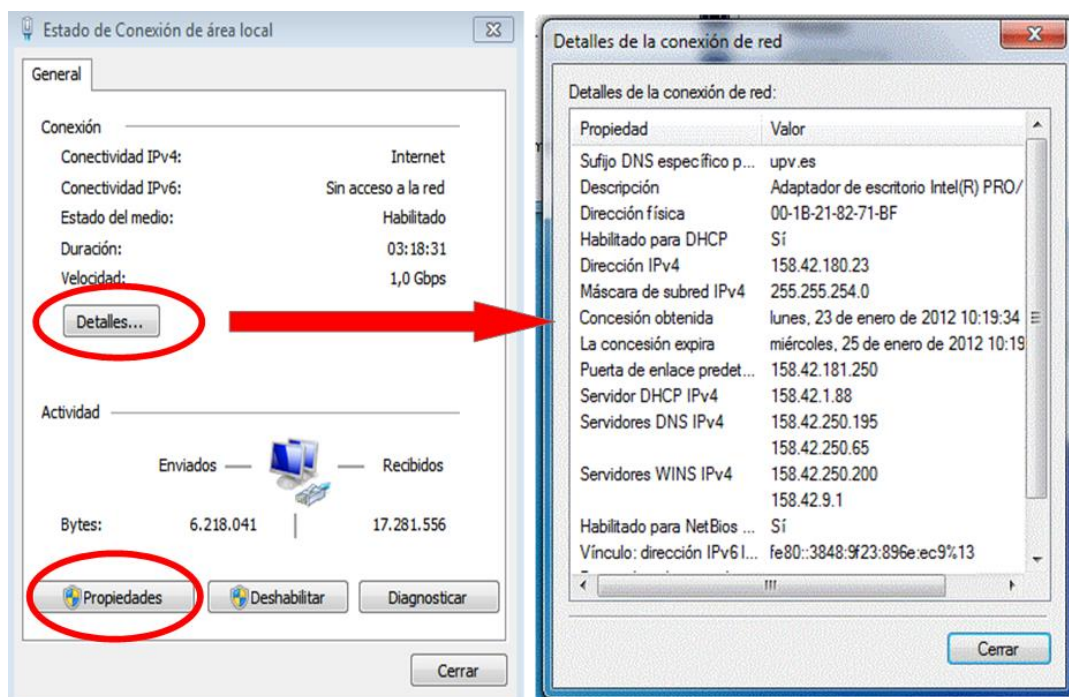
Dependiendo de si tienes una tarjeta de red Ethernet o Wifi la forma de ver la configuración puede ser ligeramente distinta. Vamos a ver los dos casos:

Para ver la configuración de la tarjeta adaptadora Ethernet:

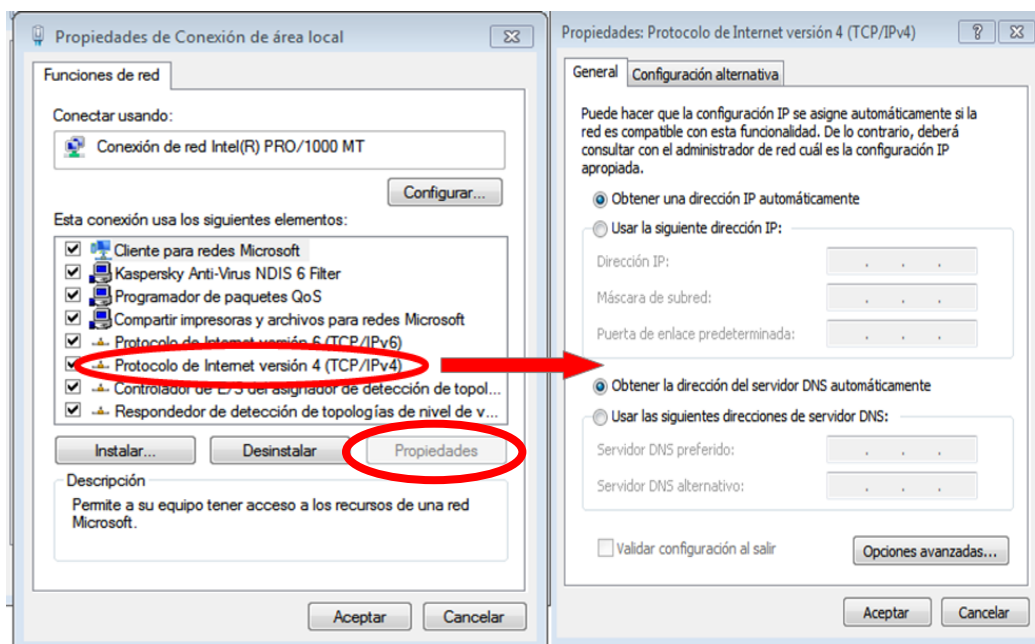
1. Pulsa el botón **Inicio**, y después **Configuración** (o  icono)→ **Red e Internet**.
2. En el apartado **Centro de redes y recursos compartidos** aparece, entre otras informaciones, la red activa en la que se encuentra tu PC y el tipo de acceso a esa red. En los computadores directamente conectados a una red local, en **conexiones** aparece el adaptador Ethernet que se está empleando. Haz clic sobre el enlace **Ethernet** (en nuestro ejemplo **Ethernet2**)



3. Se abrirá la ventana de **Estado** del adaptador Ethernet. En ella, al pulsar el botón **Detalles**, podemos ver las principales propiedades de la conexión de red asociada al adaptador: dirección física del adaptador de red, dirección IP asociada a ese interfaz, máscara de red, etc.




Además del adaptador de red, para poder utilizar aplicaciones Internet es necesario que estén instalados los protocolos TCP/IP. De nuevo, estos protocolos ya están instalados en nuestros ordenadores. Un administrador podría comprobarlo siguiendo los pasos siguientes. Si tú no lo eres, puedes observarlo en la figura siguiente.



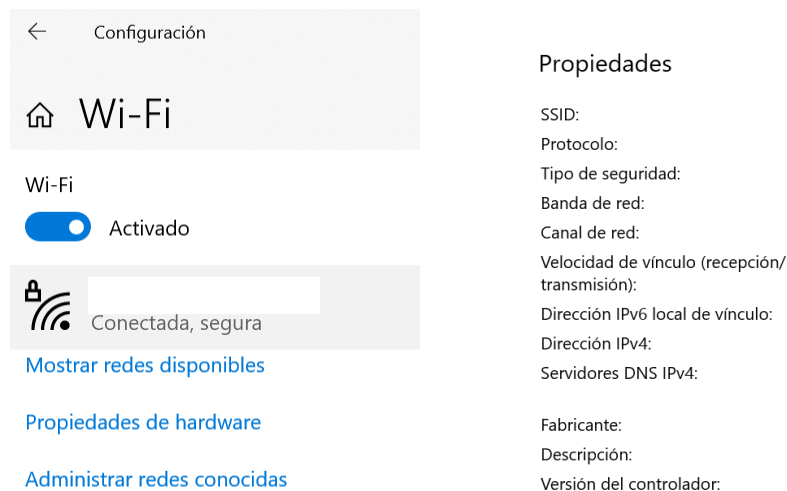
4. Desde la ventana **Estado** del adaptador Ethernet (que se muestra en el gráfico anterior a la izquierda) si se pulsa sobre el botón **Propiedades**, aparece la ventana de **Propiedades**, donde se muestran todos los elementos que emplean este adaptador de red. Entre ellos destaca el protocolo de Internet versión 4, que representa la pila TCP/IP. Tras seleccionarlo, el botón **Propiedades** permite acceder a una nueva ventana que muestra los parámetros de funcionamiento.

Como podemos observar, la configuración más habitual consiste en que la mayoría de los parámetros necesarios para el funcionamiento de TCP/IP (¡incluyendo la propia dirección IP!) no se configuran manualmente, sino que se obtienen automáticamente durante el proceso de arranque de la máquina. Esto es posible gracias al protocolo DHCP (*Dynamic Host Configuration Protocol*) que permite a un cliente solicitar al servidor una dirección IP. Este protocolo, de empleo frecuente, se estudiará en una práctica posterior durante este cuatrimestre. Además de la dirección IP, el servidor DHCP proporciona información adicional necesaria para el funcionamiento de los protocolos TCP/IP (dirección IP del servidor DNS, dirección IP del *router*, etc.).

Para ver la configuración de la tarjeta adaptadora WiFi:

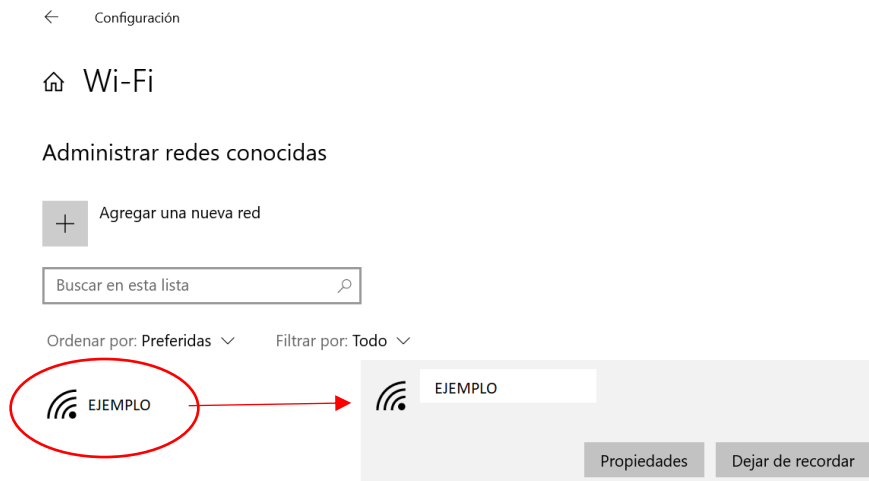
1. Pulsa el botón **Inicio**, y después **Configuración** (o el icono ) → **Red e Internet**.
2. En el apartado **Wi-Fi** aparece, entre otras informaciones, la red activa en la que se encuentra tu PC (parte izquierda de la figura)

3. En **propiedades de hardware** (parte derecha de la figura) podemos ver las principales propiedades de la conexión de red asociada: dirección física del adaptador de red, dirección IP asociada a ese interfaz, etc.



Además del adaptador de red, para poder utilizar aplicaciones Internet es necesario que estén instalados los protocolos TCP/IP. De nuevo, estos protocolos ya están instalados en nuestros ordenadores.

4. Desde la opción **Administrar redes conocidas** si se pulsa sobre nuestra red Wi-Fi (EJEMPLO), aparece la ventana de **Propiedades**, donde se muestran los elementos que relacionados con este adaptador de red.



5. Entre ellos destaca el protocolo de Internet versión 4, que representa la pila TCP/IP. Tras seleccionar en **Configuración de IP**, el botón **Editar**, si no tenemos una configuración automática puesta por defecto, podemos acceder a una nueva ventana que muestra los parámetros de funcionamiento.

Configuración de IP

Asignación de IP:

Editar

Editar configuración de IP

Manual

IPv4

☒ Activado

Dirección IP

Longitud del prefijo de subred

Puerta de enlace

DNS preferido

DNS alternativo

Como en el caso del adaptador Ethernet, la configuración más habitual emplea DHCP para obtener la información necesaria para el funcionamiento de los protocolos TCP/IP (dirección IP del servidor DNS, dirección IP del *router*, etc.).



2.1 La orden `ipconfig`

Una vez que los protocolos TCP/IP están instalados, la orden **`ipconfig`**, (se ejecuta desde una ventana de DOS – Símbolo del sistema – a la que puede accederse desde el menú “Inicio” tecleando `cmd` en la ventana de texto inferior), proporciona información sobre la configuración de la red en nuestra máquina (para cada uno de los adaptadores de red instalados).

Si ejecutamos **`ipconfig /?`** obtendremos información sobre la orden, que nos permite, entre otras cosas, obtener información para cada uno de los adaptadores de red instalados en nuestro ordenador sobre:

- **Dirección IPv4 e IPv6:** direcciones IP asignadas a nuestra máquina, en nuestro caso de forma dinámica mediante el protocolo DHCP.
- **Máscara de subred:** indica qué parte de la dirección IPv4 identifica la red, y qué parte identifica al computador (a un adaptador de red). La red de la UPV globalmente tiene asignado el bloque de direcciones IPv4 158.42.0.0/16, que se ha desglosado en una serie de subredes. La máscara de subred (255.255.254.0) indica que, en el caso de la subred del laboratorio, los 23 bits más significativos de cada dirección IPv4 (bits a 1 en la máscara) deben considerarse identificador de red, y los 9 últimos (bits a 0 en la máscara) identificador de *host*.
- **Puerta de enlace predeterminada:** dirección IP del router que conecta nuestra subred con el resto de la red de la UPV y con el exterior (Internet).

Ejercicio 1

Desde el intérprete de ordenes de Windows ( > Sistema de Windows >  Símbolo del sistema) ejecuta **ipconfig /?** para visualizar sus opciones. A continuación, identifica cuál es la forma de averiguar los parámetros explicados en párrafos anteriores (**direcciones IPv4 e IPv6, máscara de subred, puerta de enlace predeterminada**) usando esta orden.

En la respuesta obtenida, identifica cuál de todos los adaptadores que aparecen es el que te conecta a Internet y por qué.

Si ejecutamos la orden **ipconfig /all** obtenemos información adicional, entre la cual destaca:

- **Dirección física:** es la dirección que corresponde a la tarjeta adaptadora de red (Ethernet en nuestro caso) que está instalada en nuestro computador y nos permite el acceso a la red.
- **Servidores DNS:** la dirección IP de la(s) máquina(s) que realiza(n) las traducciones de nombres a direcciones IP (servidor de nombres).
- **Servidor DHCP:** dirección IP de la máquina que nos ha asignado la dirección IP y la mayoría de parámetros que aparecen en esta ventana.
- **Concesión obtenida (la concesión expira):** fecha en la que fue obtenida (caducará) la dirección IP actual. Aplicable únicamente en el caso de información obtenida por DHCP.

Las órdenes **ipconfig /release** e **ipconfig /renew** permiten liberar y renovar la dirección IPv4 obtenida mediante DHCP.

Ejercicio 2

Ejecuta la orden **ipconfig /all** y completa la información siguiente relativa al adaptador que tenga asociada una dirección IP.

Dirección física del adaptador Ethernet conexión de área local	
Dirección IPv4	
Máscara de subred	
Dirección IP del router (puerta de enlace)	
Servidores DNS	
Servidor DHCP	

Según la información obtenida:

- ¿Cuál es la dirección IP de la red a la que está conectado tu equipo?
- Los servidores DNS y DHCP, ¿están en la misma subred que tu ordenador? ¿Cómo lo has averiguado?

Ejercicio 3

Comprueba el contenido de la caché DNS. Anota en la tabla los valores de uno de los registros que aparecen que sea de tipo 1:

Tipo registro	
Nombre registro	
Valor registro (un registro <host>)	

2.2 La orden ping

Mediante la orden **ping** (que se ejecuta desde una ventana DOS) se obtiene una estimación del tiempo de ida y vuelta de un paquete (RTT, *Round Trip Time*), desde la estación donde se ejecuta la orden a la estación destino que se especifica. El funcionamiento de la orden **ping** se basa en el uso de mensajes ICMP de eco, que se estudiarán en una práctica posterior.

Ejemplo: ejecuta la orden **ping www.rediris.es** y observa lo que hace.

La orden **ping** admite una serie de opciones, la única que nos interesa de momento se muestra a continuación:

```
ping [-n cantidad] destino
```

-n cantidad número de solicitudes de eco a enviar.

Esta orden se analizará con detalle en la práctica destinada al estudio del protocolo ICMP. Hasta ese momento la emplearemos únicamente con el propósito de saber si un destino determinado puede alcanzarse o no a través de la red y cuál es su dirección IP.

2.3 La orden tracert

La orden **tracert** (se ejecuta desde una ventana DOS) permite conocer el camino (secuencia de routers) que debe atravesar un paquete para llegar desde la estación origen a la estación destino. Se basa en el empleo de mensajes ICMP y, por lo tanto, también se estudiará en la práctica destinada a este protocolo.

Ejemplo: ejecuta la orden **tracert www.rediris.es** y observa lo que hace. Cuando finalice, ejecuta **ping www.rediris.es** e intenta justificar el valor del campo TTL de las respuestas.

2.4 La orden netstat

La orden **netstat** (desde **Símbolo del sistema**) ofrece diversa información sobre el estado y estadísticas de los protocolos de red. Se pueden obtener datos sobre los principales sucesos Ethernet, IP, ICMP, UDP y TCP. Ejecuta **netstat -h**, y observa las diferentes opciones de uso. Emplearemos varias de ellas a continuación, aunque puedes experimentar con el resto.

Mediante la orden **netstat -r** obtenemos información sobre la tabla de encaminamiento (produce la misma salida que la orden **route print**).

Recuerda que, cuando hay que encaminar un datagrama, para averiguar la ruta se sigue el proceso de reenvío tal como se estudió en las clases de teoría. En concreto, el mecanismo es el siguiente:

1. Para cada línea de la tabla de encaminamiento, se realiza un AND lógico entre la **dirección IP destino** del datagrama y la **máscara de red**. IP compara el resultado con la **Red destino** y marca todas las rutas en las que se produce coincidencia.
2. De la lista de rutas coincidentes IP selecciona la ruta que tiene más bits en la máscara. Esta es la ruta más específica y se conoce como la **ruta de máxima coincidencia** (*longest matching*).
3. Si hay varias rutas de máxima coincidencia, se usa la ruta con menor **métrica**. Si hay varias con la misma métrica se usa una cualquiera de ellas.

Ejercicio 4:

Visualiza la tabla de encaminamiento (apartado IPv4) del ordenador en el que estás trabajando. Averigua y anota las direcciones IP de los destinos siguientes (recuerda los ejercicios anteriores) y analiza qué ruta de la tabla se seleccionaría para cada uno de ellos:

- a) Un paquete destinado a **zoltar.redes.upv.es**
- b) Un paquete destinado a **www.upv.es**
- c) Un paquete destinado a **www.usc.edu**

Ejercicio 5:

La orden **netstat -e** proporciona estadísticas sobre el número de bytes y tramas enviadas y recibidas por el adaptador de red. Se detalla el número de tramas unicast (un solo destino), no unicast (múltiples destinos y difusiones), paquetes erróneos y descartados.

Ejecuta esta orden y anota los resultados en la tabla siguiente:

	Recibidos	Enviados
Paquetes de unidifusión (unicast)		
Paquetes no de unidifusión (no unicast)		
Descartados		
Errores		

Comprueba también la ejecución de **netstat -es**. Indica las diferencias que observas entre el formato de salida de las dos ejecuciones.

Ejercicio 6:

La orden **netstat -sp IP** produce estadísticas sobre el tráfico IP. Ejecuta esta orden y anota los resultados en la tabla siguiente:

	Cantidad
Paquetes recibidos	
Errores de encabezado recibidos	
Errores de dirección recibidos	
Datagramas reenviados	
Protocolos desconocidos recibidos	
Datagramas correctamente fragmentados	

Ejercicio 7:

Análogamente la orden **netstat -sp TCP** produce estadísticas sobre el tráfico TCP (también se pueden solicitar estadísticas sobre los protocolos ICMP y UDP). Ejecuta esta orden y anota los resultados en la tabla siguiente:

	Cantidad
Activos abiertos	
Pasivos abiertos	
Intentos de conexión erróneos	
Conexiones actuales	

¿A qué hacen referencia las dos primeras filas de la tabla (“Activos abiertos” y “Pasivos abiertos”)?

La orden **netstat** sin argumentos ofrece información sobre las conexiones activas en nuestra máquina. Si se utiliza con la opción **-a**, además de la información anterior se indica también la relación de puertos TCP y UDP en los que hay alguna aplicación escuchando (dispuesta a aceptar conexiones TCP o datagramas UDP).

2.5 La orden arp

Esta orden también resulta de mucha utilidad para la configuración y diagnóstico de problemas en redes. Para analizarla de forma detallada, se dedicará una práctica al protocolo ARP más adelante durante este cuatrimestre.

Ejemplo: ejecuta la orden **arp -a** para observar las direcciones IP de las máquinas con las cuales ha interactuado tu PC, y su dirección física asociada.

3. Configuración de TCP/IP en Linux

*Para este apartado debes emplear la máquina virtual **Redes2021**.*

En Linux podemos encontrar utilidades equivalentes a las que se han estudiado en el apartado anterior. Para verificar la conectividad con una determinada máquina disponemos de las órdenes **tracert**, y **ping**, que cumplen la misma misión, respectivamente, que **tracert** y **ping** de Windows 10. A diferencia de Windows, **ping** en Linux sigue enviando paquetes hasta que se aborta la ejecución (**Ctrl + c**) si no se especifica el parámetro **-c <numero>**, que limita el número de intentos.

Las nuevas versiones del sistema operativo emplean una única herramienta, **ip**, para el acceso a la configuración local, proporcionando funcionalidades superiores a las órdenes **ipconfig**, **arp** y **netstat**, que hemos estudiado para Windows. Como anécdota, en versiones anteriores sí existen las órdenes **arp** y **netstat**, e incluso **ifconfig**, que resulta similar a **ipconfig**.

3.1 La orden ip

La orden **ip**, que puede ejecutarse **desde un terminal de órdenes**, permite configurar y obtener información sobre diversos aspectos de la configuración de red. Al ejecutar **ip** sin parámetros se obtiene una descripción de sus posibilidades:

```

usulocal@rdevm:~$ ip
Usage: ip [ OPTIONS ] OBJECT { COMMAND | help }
       ip [ -force ] -batch filename
where  OBJECT := { link | address | addrlabel | route | rule | neigh | ntable |
                  tunnel | tuntap | maddress | mroute | mrule | monitor | xfrm |
                  netns | l2tp | fou | macsec | tcp_metrics | token | netconf | ila |
                  vrf | sr }
       OPTIONS := { -V[ersion] | -s[tatistics] | -d[etails] | -r[esolve] |
                   -h[uman-readable] | -iec |
                   -f[amily] { inet | inet6 | ipx | dnet | mpls | bridge | link } |
                   -4 | -6 | -I | -D | -B | -O |
                   -l[oops] { maximum-addr-flush-attempts } | -br[ief] |
                   -o[neline] | -t[imestamp] | -ts[hort] | -b[atch] [filename] |
                   -rc[vbuf] [size] | -n[etns] name | -a[ll] | -c[olor]}

usulocal@rdevm:~$

```

Como puede verse, **ip** tiene muchas más posibilidades que **ipconfig** de Windows 10. Permite trabajar sobre varios elementos de la pila de protocolos: mediante **ip link** se accede a los adaptadores de red; **ip address** se refiere a las direcciones IPv4 e IPv6 del adaptador; **ip route** accede a la tabla de reenvío mientras que **ip neighbour** permite acceder a las tablas de caché de vecinos (protocolos ARP en IPv4 y ND en IPv6), mientras que **ip ntables** permite gestionar estas tablas. Existen más posibilidades que se comentarán en prácticas posteriores.

3.1.1 La orden **ip address**

Para obtener la información relativa a los adaptadores de red del sistema y sus direcciones IP asociadas, la orden adecuada es **ip address list** o **ip address show**. También valdría emplear **ip address**, puesto que **show** es la opción por defecto. Además, pueden abreviarse los parámetros siempre que no exista ambigüedad; **ip address** puede escribirse como **ip addr** e incluso **ip a**.

```

usulocal@rdevm:~$ ip address show
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: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
    link/ether 08:00:27:84:e5:1b brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 85360sec preferred_lft 85360sec
    inet6 fe80::a00:27ff:fe84:e51b/64 scope link
        valid_lft forever preferred_lft forever

```

Como puede verse, la máquina empleada para las pruebas tiene dos interfaces de red. El interfaz **lo** corresponde a una tarjeta de red virtual que implementa los accesos a *local loopback*, es decir, recibe los datagramas IP enviados a direcciones del tipo 127.x.x.x y los devuelve como si se hubieran recibido desde la red. Por el contrario, el interfaz **enp0s3** corresponde a la tarjeta de red del computador que permite su conexión a la red local, y, a través de ella, a Internet.

Para cada interfaz, **ip** muestra cuáles son sus direcciones IP asociadas, tanto IPv4 como IPv6, su prefijo de red y su rango: En el caso de las direcciones IPv4, todas las direcciones son globales, mientras que la dirección IPv6 asignada pertenece a un ámbito más restringido (direcciones locales).

También aparece la información relativa al adaptador a nivel de enlace de datos: Su dirección física (que veremos en prácticas sucesivas), la MTU de la red (fundamental para la siguiente práctica de fragmentación), y si el interfaz está activo (UP) o no.

Ejercicio 8:

Utiliza la orden **ip** para averiguar cuántos adaptadores de red existen en tu computador, cual es la dirección IP y máscara de red de cada uno de ellos. Interpreta si permiten difusiones (broadcast), y multidifusión (multicast) y cuál es la MTU de cada red accesible a través de cada uno. Verifica, además, si son direcciones permanentes o deben ser renovadas periódicamente, así como si son globales o locales.

La orden **ip address** ofrece a un administrador del sistema la posibilidad de añadir o eliminar direcciones de red sobre un adaptador. Pueden asignarse varias direcciones IP al mismo interfaz. Así, ejecutando **ip address add 192.168.1.153/255.255.255.0 dev enp0s3** se asigna una segunda dirección al interfaz:

```

sulocal@rdcvm:~$ sudo ip address add 192.168.1.153/255.255.255.0 dev enp0s3
[sudo] contraseña para usulocal:
usulocal@rdcvm:~$ 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: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
    link/ether 08:00:27:84:e5:1b brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 70735sec preferred_lft 70735sec
    inet 192.168.1.153/24 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe84:e51b/64 scope link
        valid_lft forever preferred_lft forever

```

Como es habitual, la primera vez que se emplea **sudo** es necesario repetir la contraseña. También podría indicarse una dirección de difusión para el mismo mediante **ip address add broadcast 192.168.1.255 dev enp0s3**, o bien realizar ambas simultáneamente mediante **ip address add 192.168.1.153/24 broadcast + dev enp0s3**. El “+” indica que para obtener la dirección de difusión debe sustituir cada uno de los bits de *host* de la IP proporcionada por “1”. De forma análoga, sustituir “**add**” por “**del**” permite eliminar esta asignación.

Esta funcionalidad permite, por ejemplo, comprobar localmente un programa cliente sobre un servidor de IP fija, asignando dicha IP al adaptador **lo**. Por ejemplo:

```

usulocal@rdcvm:~$ sudo ip address add 158.41.1.1/255.255.0.0 dev lo
usulocal@rdcvm:~$ ping 158.41.1.1
PING 158.41.1.1 (158.41.1.1) 56(84) bytes of data.
 64 bytes from 158.41.1.1: icmp_seq=1 ttl=64 time=0.020 ms
 64 bytes from 158.41.1.1: icmp_seq=2 ttl=64 time=0.032 ms
 64 bytes from 158.41.1.1: icmp_seq=3 ttl=64 time=0.030 ms
^C
--- 158.41.1.1 ping statistics ---
 3 packets transmitted, 3 received, 0% packet loss, time 2053ms
 rtt min/avg/max/mdev = 0.020/0.027/0.032/0.006 ms
usulocal@rdcvm:~$ traceroute 158.41.1.1
traceroute to 158.41.1.1 (158.41.1.1), 30 hops max, 60 byte packets
 1  158.41.1.1 (158.41.1.1)  0.024 ms  0.009 ms  0.008 ms

```

Ejercicio 9:

Accede a la página web principal de la UPV (www.upv.es) empleando un navegador como Firefox. A continuación, utiliza la orden **ip** para asignar la dirección IP correspondiente (158.42.4.23/16) a tu adaptador **lo**. Puedes instalar tu propio “servidor web” para ver la petición del cliente empleando “**sudo nc -l 80**” en otro terminal. Repite el acceso y comprueba las diferencias. Revierte el cambio, eliminando esta dirección IP de tu adaptador, y comprueba nuevamente el acceso.

3.1.2 La orden *ip link*

Es posible obtener una relación de los interfaces de red existentes en el computador ejecutando **ip link**. Sin embargo, esta orden no proporciona información acerca de la configuración relativa a protocolos por encima de nivel de enlace.

```
usulocal@rdcvm:~$ ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group
default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode
DEFAULT group default qlen 1000
    link/ether 08:00:27:84:e5:1b brd ff:ff:ff:ff:ff:ff
```

También se proporcionan estadísticas de uso de los interfaces mediante el parámetro **-s**. Se indican tanto los bytes/paquetes transmitidos y recibidos correctamente como el número de paquetes que han sufrido algún error.

Ejercicio 10:

Utiliza la orden **ip -s link show** para averiguar cuántos adaptadores de red existen y el número de bytes transmitidos y recibidos a través de cada uno de ellos. Comprueba que las cifras se incrementan tras ejecutar **ping -c 2 localhost** y **ping -c 2 www.rediris.es**

Esta orden permite a los administradores (en nuestro caso, anteponiendo **sudo**) añadir y eliminar interfaces de red de forma manual, si bien esta funcionalidad supera las expectativas de esta práctica. Sí puede resultar útil la capacidad de modificar algunos parámetros del mismo. Por ejemplo, es posible desactivar el interfaz mediante la orden **ip link set <interfaz> down**, impidiendo pues su utilización. Para restaurar su funcionalidad se emplea **ip link set <interfaz> up**.

Ejercicio 11:

Desactiva uno de los interfaces de tu computador, y comprueba el funcionamiento de los *pings* anteriores. A continuación, repite la prueba tras activar el primero y desactivar otro. Observa los resultados.

Otra de las opciones interesantes de **ip link set** consiste en la posibilidad de manipular casi cualquier parámetro del enlace. Por ejemplo:

- **ip link set dev <adaptador> arp (on|off)** permite habilitar y deshabilitar las respuestas ARP –que comentaremos en prácticas siguientes – desde el interfaz.

- `ip link set dev <adaptador> multicast (on|off)` permite el tráfico *multicast* a través del mismo.
- `ip link set dev <adaptador> promisc (on|off)` cambia el adaptador en modo promiscuo, es decir, el adaptador recibe todo el tráfico aunque no vaya dirigido a él. Es especialmente útil para la monitorización de redes, por ejemplo, empleando **Wireshark**.
- `ip link set dev <adaptador> address <dirección>` permite cambiar manualmente la dirección física del adaptador.
- `ip link set dev <adaptador> mtu <tamaño>` modifica el tamaño máximo del campo de datos de la unidad de enlace de datos (trama).

3.1.3 La orden *ip route*

- La orden **ip route** permite acceder a las tablas de reenvío del sistema. Linux emplea varias tablas para gestionar las rutas. En general, las rutas se almacenan en la tabla 254, también denominada **main**, y el sistema únicamente emplea esta tabla al calcular rutas. La tabla 255, denominada local, almacena las rutas hacia direcciones locales y de difusión. Esta distribución en tablas resulta especialmente interesante al aplicar distintas políticas de reenvío.

La forma más sencilla de utilizarla es **ip route show**:

```
usulocal@rdcvm:~$ ip route show
default via 10.0.2.2 dev enp0s3 proto dhcp src 10.0.2.15 metric 100
10.0.2.0/24 dev enp0s3 proto kernel scope link src 10.0.2.15
10.0.2.2 dev enp0s3 proto dhcp scope link src 10.0.2.15 metric 100
```

Al no indicar la tabla, se muestra la tabla principal (**main**). La interpretación de esta tabla de reenvío es sencilla. La ruta por defecto – default – consiste en transmitir a 10.0.2.2 (el router por defecto) empleando el adaptador de red **enp0s3**. Esta configuración ha sido obtenida mediante DHCP, el protocolo de configuración automático de parámetros de IP, que se estudiará en prácticas sucesivas. Finalmente, la IP origen de los datagramas IP enviados por defecto será 10.0.2.15 – nuestra dirección IP – y la métrica asociada a este enlace es 100.

Esta métrica sirve como parámetro adicional para la selección de rutas. Como se ha visto en las clases de teoría, si varias entradas de la tabla de reenvío son aplicables se elige la ruta con el prefijo de red más largo. En igualdad de longitud, se empleará esta métrica, que representa el coste de uso del enlace, para seleccionar la ruta óptima (de menor coste). La segunda línea se aplica para las entregas directas, es decir, el reenvío de los datagramas IP cuyo destino se encuentra en la misma red IP. En este caso, la información ha sido incluida en la tabla a instancias del propio sistema operativo (*kernel*).

Es posible acceder a la tabla de reenvío para el protocolo IPv6 empleando el parámetro **-6** :

```
usulocal@rdcvm:~$ ip -6 route show
fe80::/64 dev enp0s3 proto kernel metric 256 pref medium
```

Ejercicio 12:

Utiliza la orden **ip route show table all** para visualizar todas las reglas de reenvío de tu sistema, e interpreta el resultado. Prueba también con las tablas **main** y **local** sustituyendo **all** en la orden anterior.

Para un usuario administrador (anteponiendo **sudo**), **ip route add default via 192.168.1.150** permitirá modificar la ruta por defecto del sistema. También pueden añadirse rutas específicas en esta tabla (**ip route add 172.16.32.0/24 via 192.168.1.150 dev enp0s3**) e incluso rutas particulares para direcciones individuales (**ip route add 172.16.32.32 via 192.168.1.150 dev enp0s3**). Al igual que en el caso anterior, “**del**” permite anular el efecto de “**add**”.

En este apartado emplearemos la posibilidad de modificar las tablas para ver el efecto sobre el encaminamiento de los paquetes de las entradas más importantes. En particular, analizaremos la entrada por defecto, que nos permite alcanzar el resto de Internet.

Ejercicio 13:

- 1) Visualiza la tabla de encaminamiento de tu ordenador (orden **ip route show**).
- 2) Encuentra y anota la dirección del router de salida de la red. Nos referiremos a ella como **dir_IP_de_tu_router**.
- 3) Elimina la entrada de la dirección de red por defecto (**sudo ip route del default**) y visualiza la tabla de encaminamiento.
- 4) Intenta acceder a un destino fuera de tu red IP. Por ejemplo, mediante la orden
ping -c 2 www.rediris.es

Explica que pasa.

- 5) Vuelve a probar el ping empleando ahora la dirección IP de www.rediris.es (130.206.13.20) en vez del nombre del servidor.
- 6) Intenta acceder a un destino que esté en la misma red IP que tu ordenador. Por ejemplo, al router por defecto mediante la orden **ping -c 2 dir_IP_de_tu_router**.
- 7) Restaura la línea de la tabla de encaminamiento que habías eliminado (**sudo ip route add default via dir_IP_de_tu_router**).
- 8) Comprueba el estado de la tabla de encaminamiento. Debe ser el mismo que era antes de eliminar la ruta. Verifica también que ahora sí funcionan los pings a computadores fuera de tu red.