



Unidad 3 – Clase 2:

Redes



Direcciones de red y host

La máscara indica qué parte de una dirección corresponde al número de red (y host).

	red	host
Dirección:	0011	0010110
Máscara de red:	1111	00000000 = /4
Dirección de red:	0011	00000000 <i>todos ceros</i>
Dirección de host mínima en la red:	0011	00000001 <i>todos ceros y un uno</i>
Dirección de host máxima en la red:	0011	11111110 <i>todos unos y un cero</i>
Dirección de difusión en la red: (broadcast)	0011	11111111 <i>todos unos</i>
Número máximo de hosts en la red:	$2^7 - 2 = 126$	

Direcciones IPv4

Las direcciones IPv4 (Internet Protocol Version 4) son números de 32 bits. Ejemplo:

110000001010100000000000100000001

Puede verse como cuatro bytes:

110000001010100000000000100000001

Pueden escribirse como cuatro números decimales (entre 0 y 255) separados por puntos:

192 . 168 . 1 . 1

ipcalc

```
black@black: ~
Archivo Edición Pestañas Ayuda
black@black:~$ ipcalc 192.168.0.1/24
Address: 192.168.0.1      11000000.10101000.00000000. 00000001
Netmask: 255.255.255.0 = 24 11111111.11111111.11111111. 00000000
Wildcard: 0.0.0.255      00000000.00000000.00000000. 11111111
=>
Network: 192.168.0.0/24   11000000.10101000.00000000. 00000000
HostMin: 192.168.0.1     11000000.10101000.00000000. 00000001
HostMax: 192.168.0.254   11000000.10101000.00000000. 11111110
Broadcast: 192.168.0.255 11000000.10101000.00000000. 11111111
Hosts/Net: 254           Class C, Private Internet

black@black:~$
black@black:~$
black@black:~$ ipcalc 186.125.0.1/24
Address: 186.125.0.1     10111010.01111101.00000000. 00000001
Netmask: 255.255.255.0 = 24 11111111.11111111.11111111. 00000000
Wildcard: 0.0.0.255      00000000.00000000.00000000. 11111111
=>
Network: 186.125.0.0/24  10111010.01111101.00000000. 00000000
HostMin: 186.125.0.1     10111010.01111101.00000000. 00000001
HostMax: 186.125.0.254   10111010.01111101.00000000. 11111110
Broadcast: 186.125.0.255 10111010.01111101.00000000. 11111111
Hosts/Net: 254           Class B

black@black:~$ █
```

Direcciones IP públicas y privadas

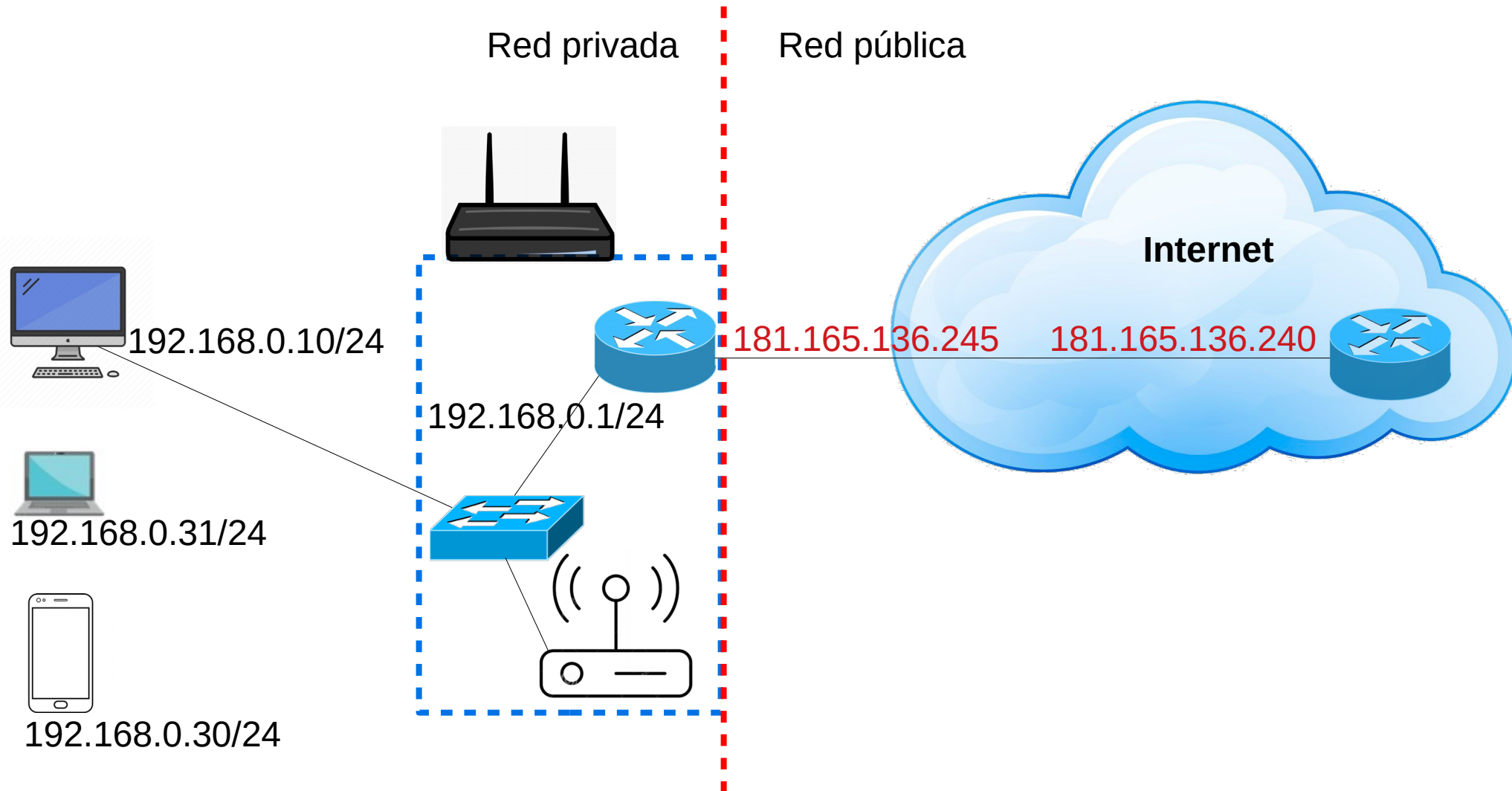
Existen direcciones IP que son para uso en redes privadas. Estas direcciones no pueden circular por Internet y los routers públicos ignoran los paquetes que tienen una dirección de destino privada.

Rangos de direcciones privadas:

10.0.0.0	a	10.255.255.255	/8
172.16.0.0	a	172.31.255.255	/12
192.168.0.0	a	192.168.255.255	/16

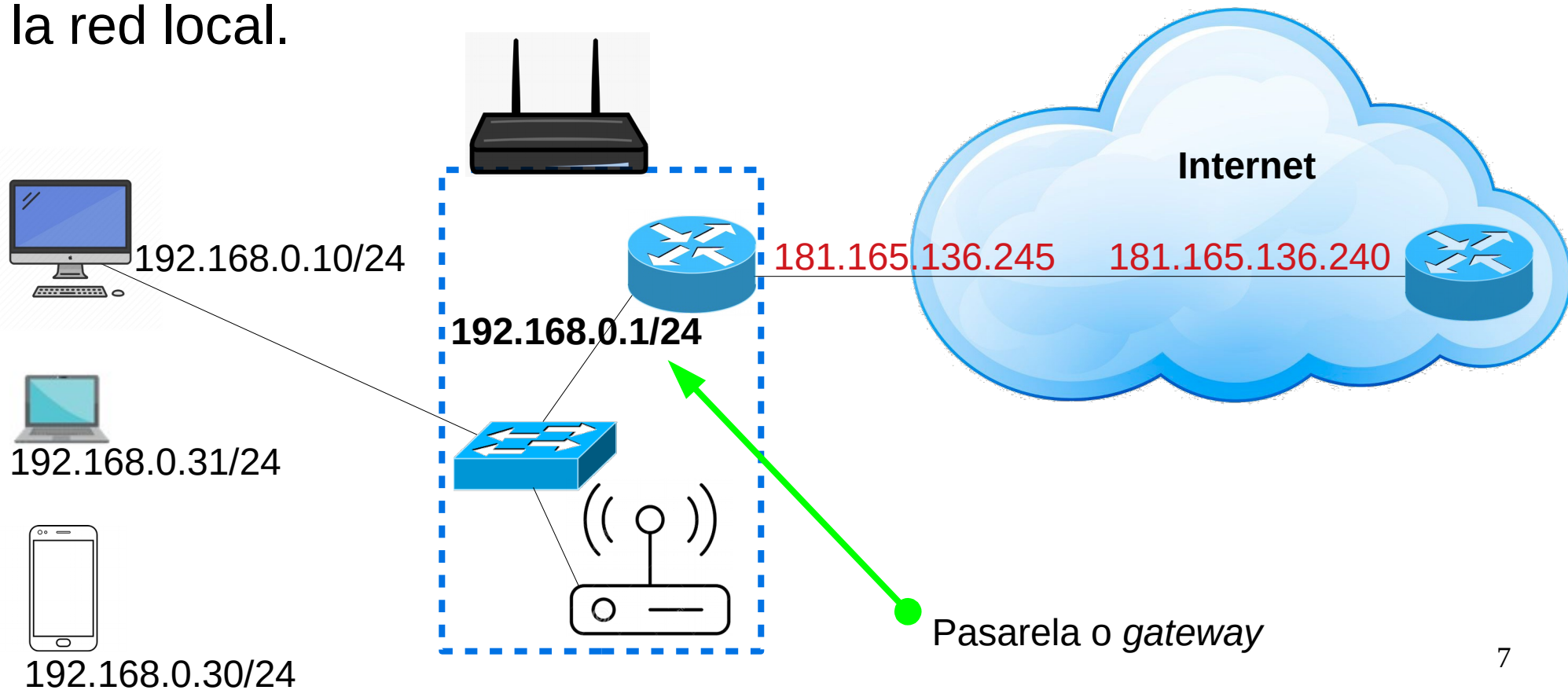
Esta diferenciación hace posible la construcción de redes privadas sin necesidad de ocupar direcciones de Internet públicas para hosts que no lo requieren. Ej: Red interna de la universidad, red hogareña, etc.

Redes públicas y privadas



Pasarela o gateway

En una tabla de reenvío, la **pasarela** indica, para cada entrada en la tabla, la dirección del router que debe ocuparse de reenviar el paquete. En caso de que el paquete no necesite atravesar un router, la pasarela tendrá todos sus bits en 0; este es el caso en que el paquete se dirige a un host de la red local.



route

Los comandos **route -n** o **netstat -rn** permiten ver la tabla de ruteo (o reenvío) de un nodo. Se muestran el campo destino, pasarela, máscara (Genmask), e interfaz.

```
javier@javier-ThinkPad-X121e: ~  
javier@javier-ThinkPad-X121e:~$ route -n  
Tabla de rutas IP del núcleo  
Destino      Pasarela      Genmask      Indic Métric Ref      Uso Interfaz  
0.0.0.0      192.168.1.1   0.0.0.0      UG     600    0        0 wlp1s0  
169.254.0.0  0.0.0.0       255.255.0.0  U      1000   0        0 wlp1s0  
192.168.1.0  0.0.0.0       255.255.255.0 U      600    0        0 wlp1s0  
javier@javier-ThinkPad-X121e:~$
```


tracpath / traceroute

El comando **tracpath -n ip** o **traceroute -n ip** permite identificar (si es posible) el camino que siguen nuestros paquetes desde nuestra máquina local hacia otra máquina.

```
javier@javier-ThinkPad-X121e: ~  
javier@javier-ThinkPad-X121e:~$ tracpath -n 170.210.81.15  
1?: [LOCALHOST] pmtu 1500  
1: 192.168.1.1 2.018ms  
1: 192.168.1.1 1.760ms  
2: 192.168.0.1 6.043ms  
3: no reply  
4: no reply  
5: no reply  
6: no reply  
7: no reply  
8: no reply  
9: 181.96.72.101 33.613ms asymm 11  
10: 181.88.168.59 36.407ms  
11: 181.88.70.74 36.917ms asymm 10  
12: 170.210.4.1 33.444ms asymm 11  
13: 200.32.34.149 79.243ms asymm 18  
14: 200.70.52.14 58.063ms asymm 17  
15: 200.26.75.241 52.623ms asymm 16  
16: 170.210.4.94 59.065ms asymm 15  
17: 170.210.81.15 83.661ms reached  
Resume: pmtu 1500 hops 17 back 16  
javier@javier-ThinkPad-X121e:~$
```

Modelo OSI de la ISO

Las redes pueden estudiarse y comprenderse mediante modelos jerárquicos compuestos por capas, donde cada pieza de hardware o de software pertenece a una capa o nivel.

ISO: International Organization for Standardization

OSI: Open System Interconnection

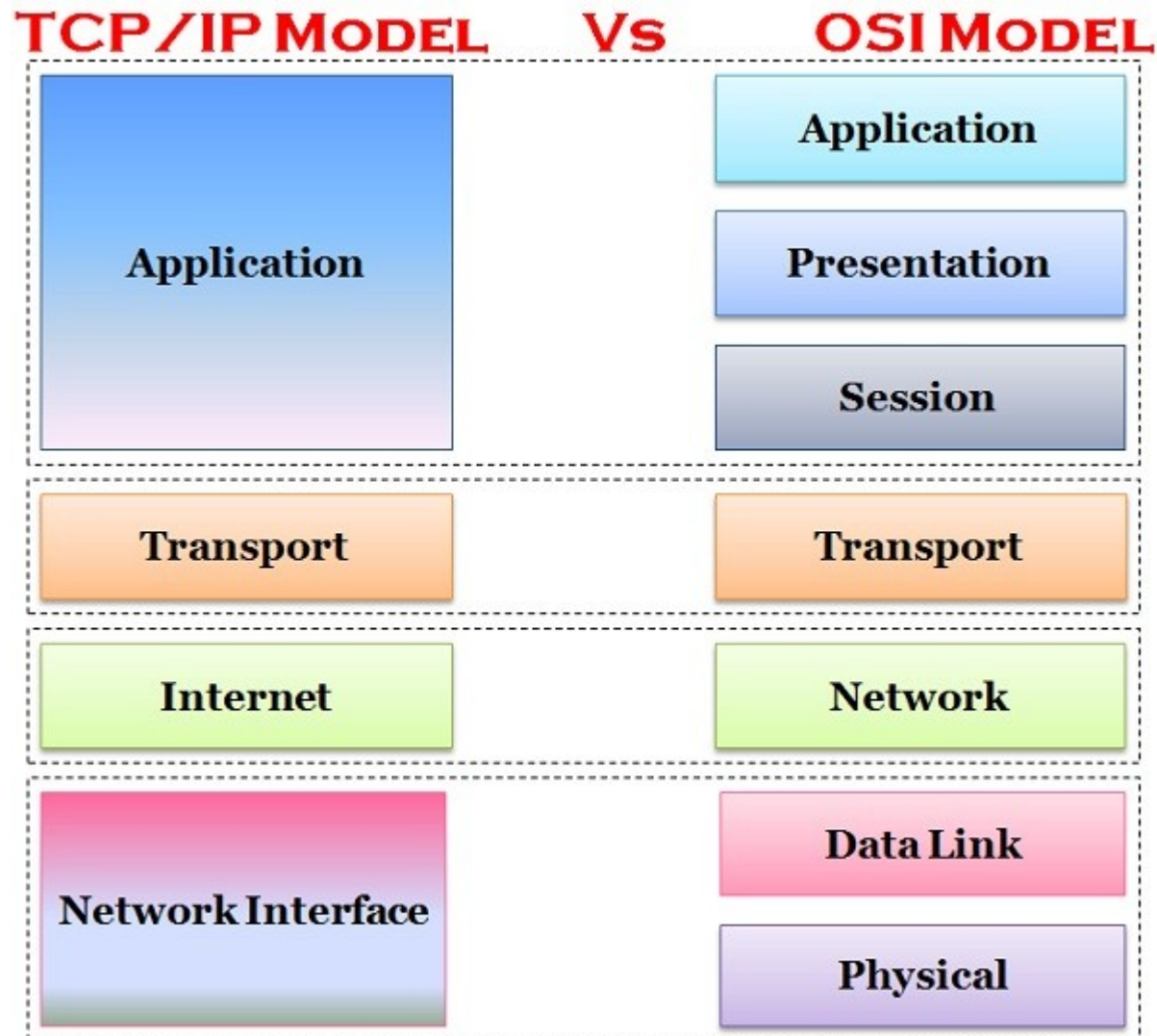
A finales de la década del '70, la ISO propone que las redes de computadoras sean organizadas utilizando una arquitectura de 7 capas, la cual se denominó modelo OSI.

Es un modelo de referencia teórico.

Cada capa provee un servicio que se construye encima de la capa inferior, y está más cerca de las necesidades de la aplicación.



Equivalencia entre el modelo OSI y el de Internet

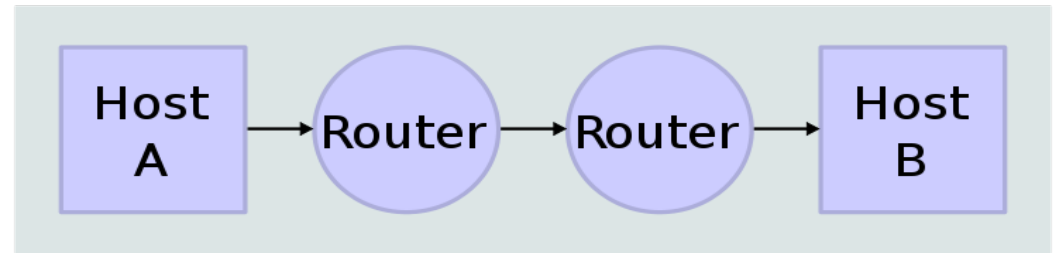


Capa de red y capa de enlace

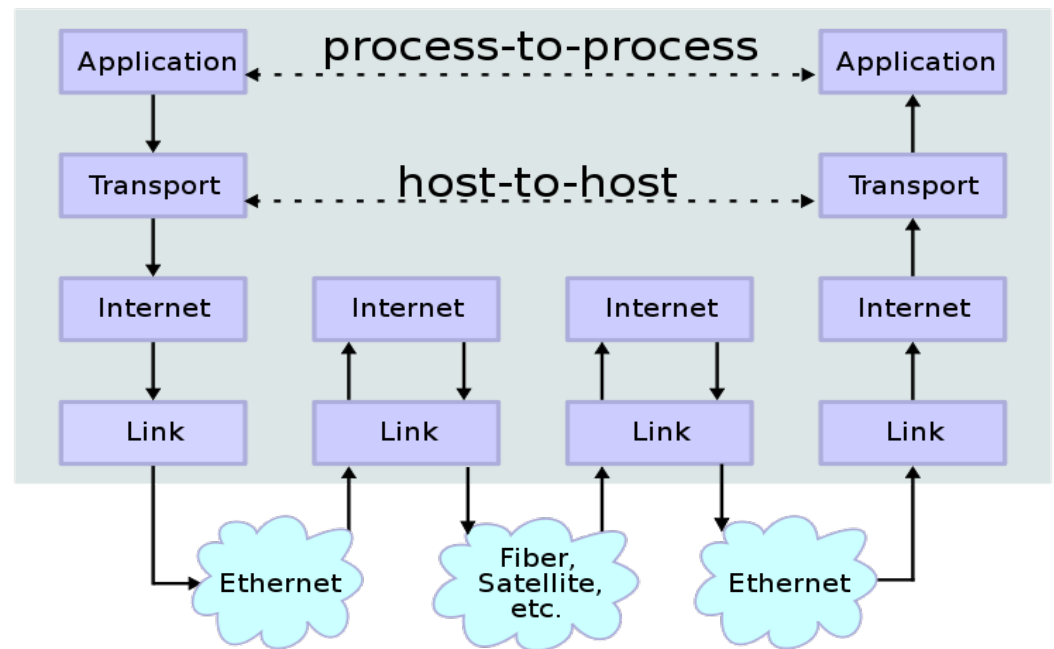
La **capa de red** de Internet encamina un paquete a través de una serie de routers entre el origen y el destino. Para trasladar un paquete de un nodo (host o router) al siguiente nodo de la ruta, la capa de red confía en los servicios de la **capa de enlace**.

- En concreto, en cada nodo, la capa de red pasa el paquete a la capa de enlace, que entrega el paquete al siguiente nodo existente a lo largo de la ruta. En el siguiente nodo, la capa de enlace pasa el paquete a la capa de red.

Network Topology



Data Flow



Capa de Enlace

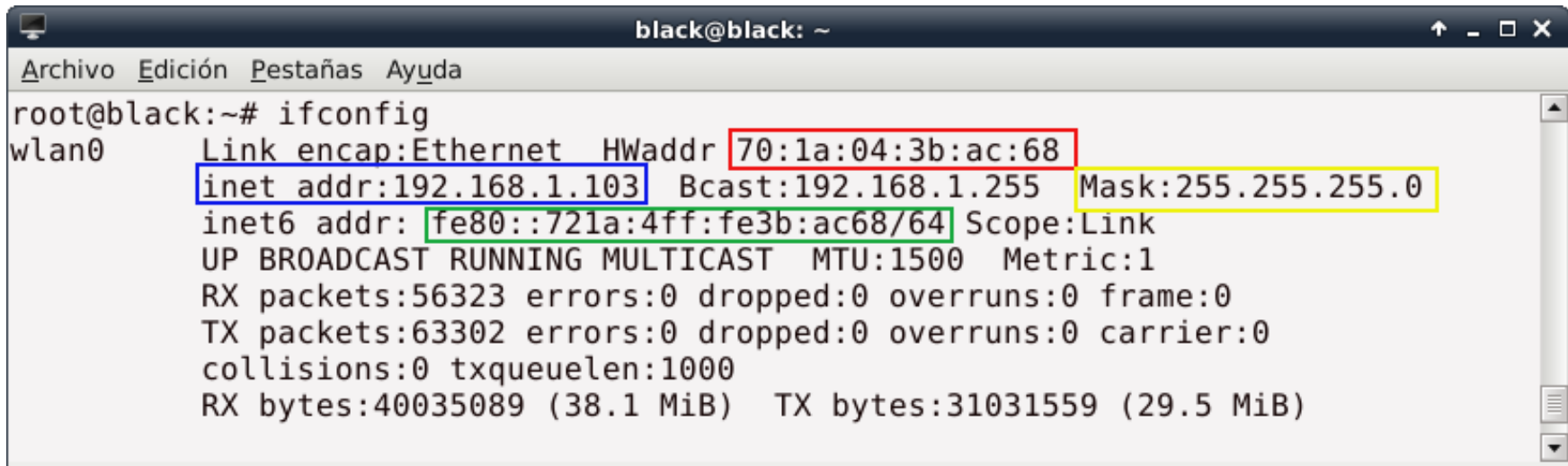
Se encarga de mover paquetes entre nodos adyacentes de la misma red, es decir, entre hosts accesibles sin atravesar un router. Por lo tanto, el switch y el punto de acceso son dispositivos que operan en capa de enlace.

Distintos tipos de enlaces (Ethernet, WiFi, etc.) usan distintos protocolos.

Control de flujo: Evitar que un equipo rápido desborde a uno lento.

Direcciones de capa de enlace: cada interfaz de red tiene una **dirección MAC** propia, asignada por el fabricante y única en el mundo. La dirección MAC (Media Access Control) es un identificador de 48 bits mostrados en 6 bloques de dos caracteres hexadecimales (8 bits) cada uno. Ejemplo: **00:21:6a:10:a8:96**

ifconfig



```
black@black: ~
Archivo Edición Pestañas Ayuda
root@black:~# ifconfig
wlan0      Link encap:Ethernet  HWaddr 70:1a:04:3b:ac:68
          inet addr:192.168.1.103  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::721a:4ff:fe3b:ac68/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:56323 errors:0 dropped:0 overruns:0 frame:0
          TX packets:63302 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:40035089 (38.1 MiB)  TX bytes:31031559 (29.5 MiB)
```

Dirección MAC o de hardware (la especifica el fabricante de la placa de red)

Dirección IP v4 (la especifica el administrador de la red o usuario)

Máscara IP v4 (la especifica el administrador de la red o usuario)

Dirección IP v6 (la especifica el administrador de la red o usuario)

El comando **ip a** muestra información similar a **ifconfig**, y cada vez se está utilizando más.

Capa de transporte

Esta capa se encarga de transportar mensajes de las aplicaciones entre nodos terminales de la red.

En Internet, existen dos protocolos de transporte: TCP y UDP.

TCP: para aplicaciones que requieren entrega confiable de datos.

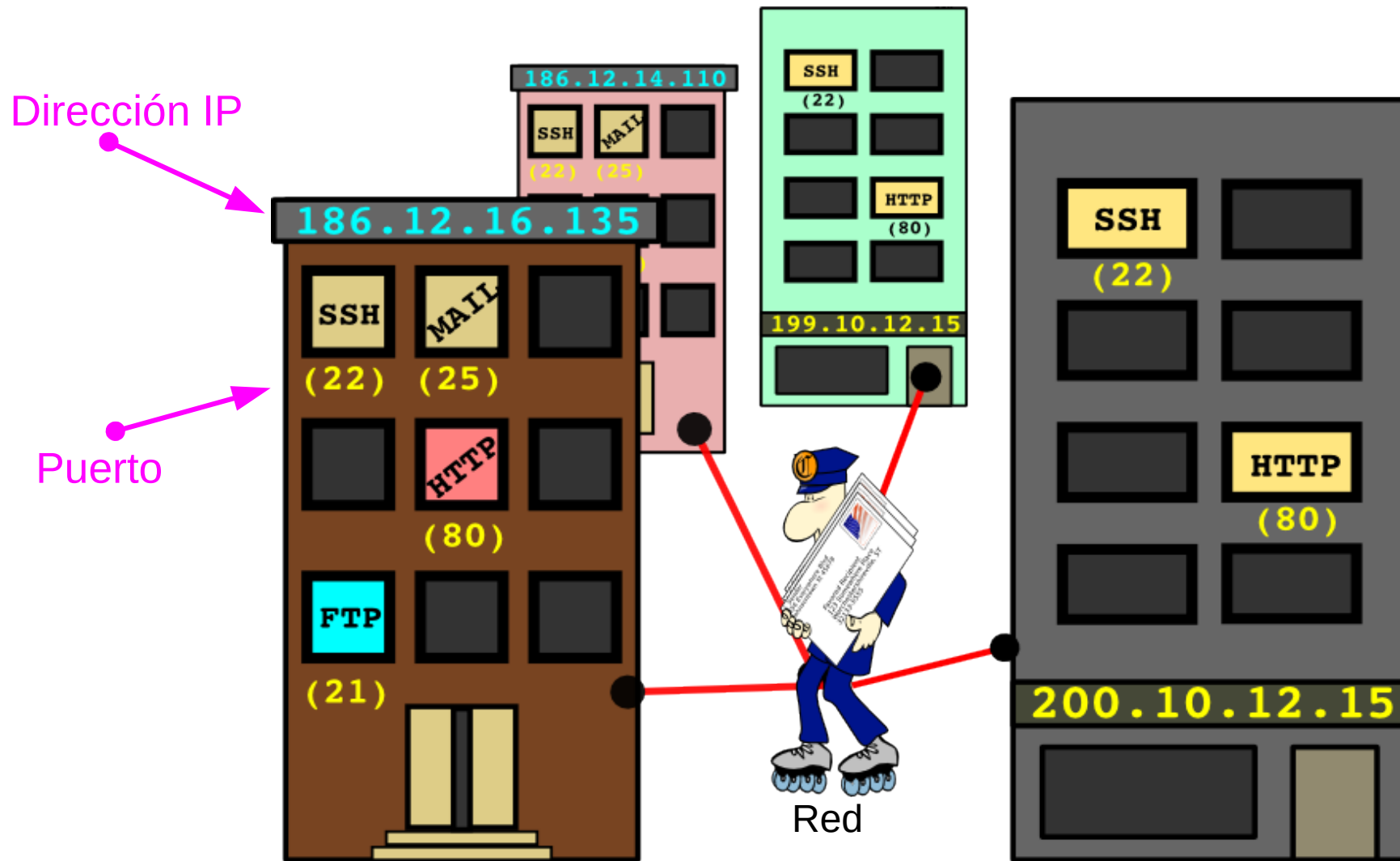
- Servicio confiable: se detectan los paquetes perdidos y se retransmiten.
- Entrega ordenada: los paquetes se reciben en orden.
- Control de congestión: reduce la transferencia de datos cuando la red está saturada.

UDP: para aplicaciones que no requieren entrega confiable de datos.

- Servicio NO confiable
- Entrega NO ordenada
- Sin control de congestión.

Capa de transporte: Puertos

¿Cómo pueden coexistir varias aplicaciones en un mismo nodo? Cada aplicación se asocia con un número de **puerto** diferente. Así, para comunicarse con una aplicación de un cierto nodo, es necesario conocer la dirección **IP y puerto**.



Escaneo de puertos TCP: nmap

Es posible descubrir (escanear) los puertos TCP abiertos de otras máquinas con el comando **nmap <ip>**

```
javier@javier-ThinkPad-X121e: ~  
javier@javier-ThinkPad-X121e:~$ nmap 216.58.202.35  
  
Starting Nmap 7.01 ( https://nmap.org ) at 2019-10-22 17:32 -03  
Nmap scan report for eze04s05-in-f3.1e100.net (216.58.202.35)  
Host is up (0.14s latency).  
Not shown: 998 filtered ports  
PORT      STATE SERVICE  
80/tcp    open  http  
443/tcp   open  https  
  
Nmap done: 1 IP address (1 host up) scanned in 49.29 seconds  
javier@javier-ThinkPad-X121e:~$
```



Bibliografía



Capítulo 1 del libro **“Redes de Computadoras un enfoque descendente”**. James Kurose, Keith Ross

“Computer Networks”. Andrew S. Tanenbaum