C.F.G.S. DESARROLLO DE APLICACIONES WEB

MÓDULO:

Despliegue de Aplicaciones Web

Unidad 2

Configuración de una red.

Protocolo TCP/IP

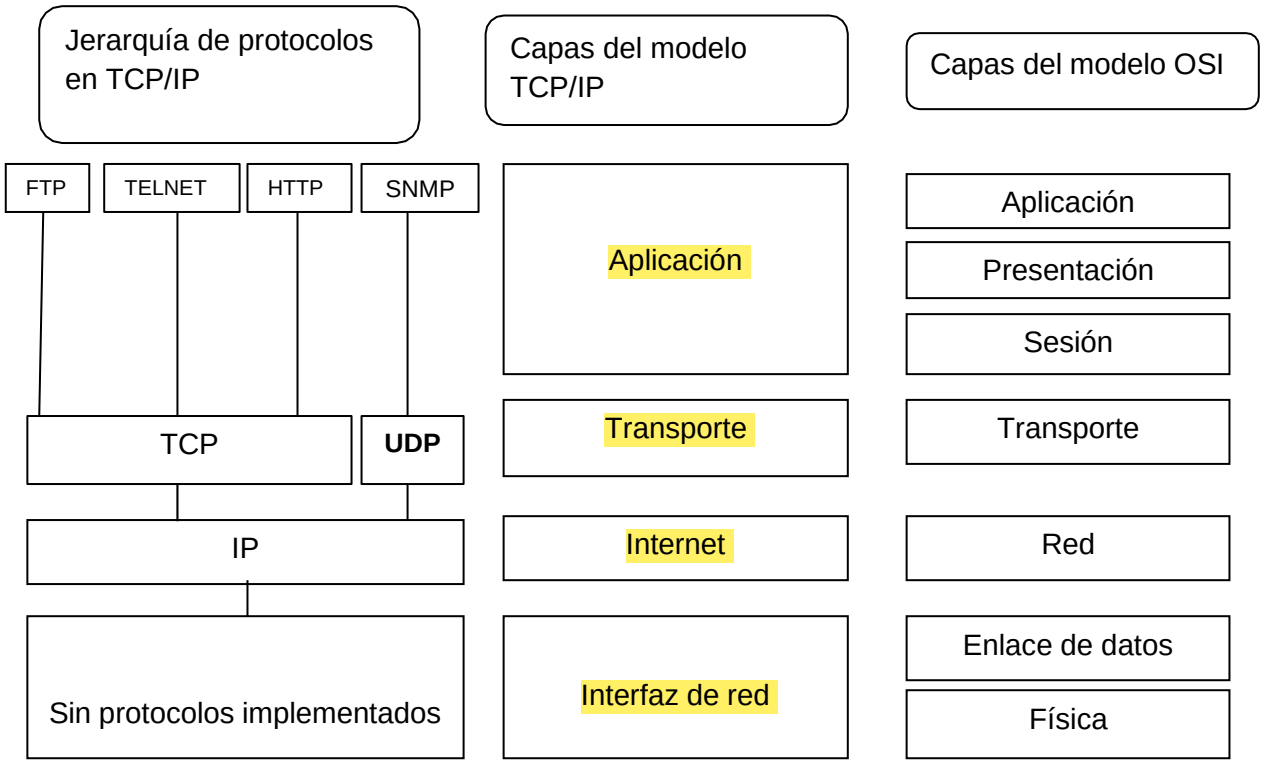
ÍNDICE DE CONTENIDOS

1.-La arquitectura TCP/IP, el modelo Cliente/Servidor y los servicios de red.

1.1.- La arquitectura TCP/IP.

La arquitectura TCP/IP nos proporciona una estructura y una serie de normas de funcionamiento para poder interconectar sistemas. La complejidad de esta tarea ha necesitado una subdivisión del trabajo en niveles o capas coordinadas de manera que, cada capa realiza una labor concreta y así la integración modular y jerárquica de todas ellas hace posible la comunicación.

En cada capa existen una serie de protocolos que ofrecen unas normas estrictas a seguir para el diálogo entre los sistemas. Cada protocolo se apoya en los protocolos de las capas inferiores para realizar su labor y, a su vez, ofrece sus servicios a las capas superiores. Esta es una de las características fundamentales de la arquitectura TCP/IP.



Equivalencia entre los niveles OSI y TCP/IP

1.2.- Protocolos de la Arquitectura TCP/IP.

TCP/IP organiza el proceso de comunicación en 4 capas:

1.- Aplicaciones
2.- Transporte
3.- Internet
4.- Interfaz de red

En cada una de las capas opera uno o varios protocolos. Cada protocolo desempeña una función específica en el mantenimiento de la comunicación, dentro del propósito de la capa a la que pertenece.
Esta estructura existe tanto en la máquina emisora como receptora de la comunicación.
La información del emisor desciende desde su capa de Aplicaciones hasta la Interfaz de red, y cuando llega al receptor, esta información asciende desde la interfaz de red hasta la capa de Aplicaciones, dirigiéndose a una aplicación en particular que se identifica con un número de puerto.

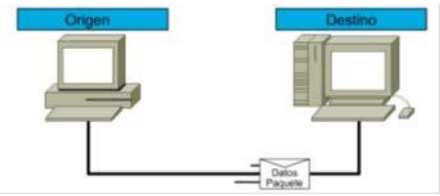
- **CAPA DE APLICACIONES.**
  Incluye todas las aplicaciones que pretenden acceder a la red.

  - HTTP: Protocolo de transferencia de hipertexto. La Web.
  - FTP: Protocolo de transferencia de archivos.

  Y otros más como: SMTP, POP3, DNS, etc.

- **CAPA DE TRANSPORTE.**
  Se encarga de preparar los datos para el transporte; los fragmenta para su envío y los ordena cuando son recibidos en su destino.

  - **TCP: Protocolo de control de transmisión. Entrega de datos con acuse de recibo.**
    Protocolo de Control de Transmisión. Proporciona un servicio de entrega de datos fiable y orientado a conexión entre dos hosts. La conexión debe establecerse antes de que los datos puedan ser transmitidos entre los dos equipos. Esta conexión se denomina sesión.



- Es una comunicación fiable, ya que garantiza la entrega de los datos al destino.
- En el equipo origen, TCP organiza en paquetes los datos que deben transmitirse.
- En el equipo destino, TCP reorganiza los paquetes reconstruyendo el mensaje de origen.
- El equipo de destino devuelve un acuse de recibo por cada paquete o grupo de paquetes.
- Si transcurrido un tiempo de espera el equipo origen no recibe respuesta de acuse de recibo repite la última transmisión.

- **UDP. Protocolo de datagramas de usuario. Sistema rápido pero no seguro.**
  Este protocolo también pertenece a la capa de transporte. Proporciona un servicio de entrega rápido de paquetes sin conexión. Tampoco ofrece fiabilidad ya que no requiere acuse de recibo de los datos recibidos. Este protocolo se utiliza en transmisiones de multidifusión, o en envíos de información de poca importancia.

  Para utilizar UDP, la aplicación origen debe suministrar su número de puerto UDP y el puerto de la aplicación de destino. Los puertos UDP son distintos de los puertos TCP aunque pueden coincidir en los mismos números.

- **CAPA DE INTERNET**
  Se encarga del enrutamiento de los datos a través de la red.

  - **IP. Protocolo de Internet. Direcciona los datos a su destino.**
    Es un protocolo no fiable y no orientado a conexión. Se encarga de direccionar los paquetes y enrutarlos entre los equipos de la red. IP no se responsabiliza de los paquetes que se puedan perder o retrasar en su entrega. Cada paquete lleva la IP del equipo emisor y la IP del equipo destino.
    Si IP identifica una dirección de destino como una dirección del mismo segmento de red transmite el paquete directamente a ese equipo. Si la dirección IP de destino no se encuentra en el mismo segmento, IP debe utilizar un enrutador para enviar la información.
    A cada paquete le asigna un tiempo de vida (TTL: Time To Live) para que no permanezca viajando por la red.

  - **ARP. Protocolo de resolución de direcciones.**
    Se encarga de identificar la dirección MAC del adaptador de red del equipo destino.
    Protocolo de resolución de direcciones. Se encarga de asociar la dirección MAC (escrita en el adaptador de red) a la IP de la máquina correspondiente.

- **ICMP. Protocolo de mensajes de control en Internet.**
  Informa de errores en la entrega incorrecta de datos. Protocolo de mensaje de control de Internet. Ofrece información sobre la red y permite evaluar las rutas de acceso hacia otros hosts.
  Entre enrutadores se envían mensajes ICMP para informar de anomalías en la red. Proporciona facilidades de diagnóstico e información de errores para los paquetes que no pueden entregarse. Si IP no puede entregar un paquete al equipo de destino, ICMP envía un mensaje de: Destino Inaccesible al equipo de origen.

  Dentro de este protocolo opera el comando ping.

  ```
  ping IP_destino
  ```

  El comando envía un grupo de paquetes a la máquina con IP indicada y ésta a su vez devuelve acuse de recibo. Algunos firewall obstaculizan el funcionamiento de este protocolo.

- **IGMP. Protocolo de administración de grupos de Internet. Gestiona la multidifusión.**
  Protocolo de administración de grupos de Internet. La multidifusión IP es un proceso por el cual un mensaje se transmite a un grupo seleccionado de receptores o grupo de multidifusión. IGMP mantiene la lista de miembros suscritos a cada grupo de multidifusión. Todos los miembros de un grupo de multidifusión escuchan el tráfico IP dirigido a una dirección de multidifusión IP específica y reciben los paquetes enviados a esa dirección IP. Los envíos se realizan mediante el protocolo UDP, que no garantiza la entrega.

- **CAPA DE INTERFAZ DE RED**
  Se responsabiliza del envío de datos en el medio físico y recibirlos desde el mismo. Cada adaptador de red dispone de un número exclusivo en hexadecimal de 12 caracteres (dirección MAC). Aquí no existen protocolos pero sí se definen tecnologías como son Ethernet o ATM.

1.3.- El modelo Cliente/Servidor.

Para la comunicación de aplicaciones a través de una red se emplean fundamentalmente tres paradigmas:

- El modelo **cliente/servidor** es el más extendido y utilizado. En él se distingue entre un proceso cliente (que generalmente solicita servicios) y un proceso servidor (que presta servicio al cliente).

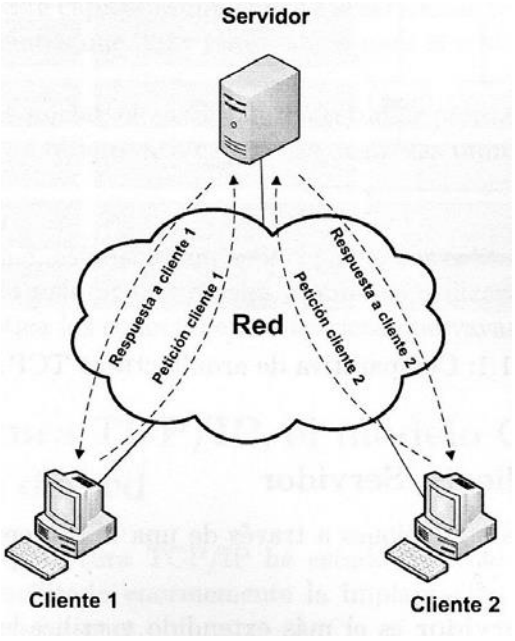
- El modelo **entre pares o P2P** (Point to Point) de difusión creciente y donde todos los nodos de la red son responsables por igual en la comunicación de las aplicaciones y no existe un elemento que centralice la comunicación.
- El modelo **híbrido** que resulta de la combinación de los dos anteriores y donde el servidor no presta el servicio como tal, sino que generalmente pone en contacto a los clientes para que estos se comuniquen entre sí.

El modelo **cliente/servidor** es el más extendido y el que se aplica en la mayoría de los diferentes servicios y será por tanto, en el que vamos a centrar preferentemente nuestra atención.

El cliente y el servidor son procesos inseparables y uno carece de sentido sin la presencia del otro.

El modelo **cliente/servidor** define tanto la estructura de las aplicaciones que se comunican entre sí como su sincronización. Está formado por dos procesos que interactúan entre sí, el proceso cliente y el proceso servidor:

- El **cliente** es el proceso que habitualmente inicia la comunicación por lo que su papel suele ser activo en la misma, envía una petición a un proceso servidor y a continuación queda a la espera de la respuesta.
- El **servidor** suele ser un proceso que inicialmente permanece a la espera escuchando las posibles conexiones de los potenciales clientes. En este punto, su papel es pasivo en la comunicación. Aún así, son sistemas complejos, difíciles de diseñar y programar, y suelen requerir privilegios del sistema. Deben ser robustos, pues ofrecen servicios que es posible que deban estar activos permanentemente. Deben tenerse en cuenta en su diseño cuestiones como la autenticación, autorización, seguridad y privacidad de la información. También es habitual que el proceso servidor pueda gestionar peticiones simultaneas de varios clientes.



Para realizar su función los protocolos del nivel de aplicación emplean los protocolos de niveles TCP/IP inferiores para funcionar. Así, por ejemplo, el protocolo de aplicación Telnet emplea el protocolo TCP en la capa de transporte, mientras que el protocolo de aplicación DHCP emplea el protocolo UDP en la capa de transporte.

1.4.- Los servicios de red.

Un **servicio de red** es, o puede considerarse, una función o prestación que ofrecen las **aplicaciones** y los **protocolos** a los **usuarios** o a otras **aplicaciones**.

En este sentido, las **aplicaciones** son sistemas software que se comunican e intercambian información con otras aplicaciones, con ayuda de los **protocolos de la arquitectura TCP/IP** tanto de nivel de aplicación como de niveles inferiores.

Es importante no confundir los protocolos del nivel de aplicación con las aplicaciones que los utilizan:

- Las **aplicaciones** son los diferentes programas, instalados por el usuario o que son parte del sistema operativo, que se sirven de los protocolos de la arquitectura TCP/IP para comunicarse. Ejemplos de aplicaciones son Mozilla Firefox, Google, Chrome, etc.
- Los **protocolos** son normas concretas, descritas formalmente, que detallan cómo se produce la comunicación entre sistemas para ofrecer los servicios de red. Ejemplos de protocolos del nivel de aplicación son IMAP y HTTP.

2.- Nivel de red en TCP/IP – El protocolo IP.

A nivel de red se realiza el **direccionamiento** de los dispositivos y el **encaminamiento** de la información a través de la red. Todo ello se lleva a cabo con el protocolo IP que es el principal de este nivel en la arquitectura TCP/IP. El esquema de direccionamiento utilizado en cada nodo de la red y los procesos de encaminamiento, que se ejecutan en los dispositivos que interconectan las redes, son las funciones principales de este protocolo. La comunicación a nivel IP se hace mediante unidades de datos llamadas datagramas que siguen el formato especificado en el protocolo IP.

2.1.- Direccionamiento IP.

El protocolo IP proporciona conectividad extremo a extremo en la comunicación. Esto supone que debe ser capaz de direccionar de forma única todos los dispositivos que tengamos conectados en nuestra red y; por extensión, en todo Internet. Este direccionamiento es abstracto, de forma que es independiente del dispositivo físico al que se asigna y puede ser modificado vía software.

Para saber más

Una dirección IP no identifica a un ordenador en la red, sino que identifica **a un interfaz de red** de un ordenador en la red. Por eso es posible que un mismo equipo pueda tener varias direcciones IP, una por interfaz, y eso hace posible que pueda estar contacto a redes diferentes de manera simultánea. Incluso es posible que un interfaz pueda tener varias direcciones IP denominándose entonces direccionamiento virtual.

- **Formato de direcciones IP.**

Una dirección IP es un número binario de 32 bits. Esto permite un espacio de direcciones de 2^{32} (4.294.967.296) direcciones diferentes posibles. Habitualmente, la notación empleada para facilitar la legibilidad de las direcciones IP es la notación decimal con puntos. Así se dividen los 32 bits en 4 grupos de 8 bits, escribiendo cada uno de ellos en base decimal, separando por puntos los cuatro grupos resultantes. Por tanto, una dirección IP estará formada por cuatro números entre 0 y 255 separados por puntos.

Ejemplo de dirección IP:

1 1 0 0 1 1 0 0 0 0 1 1 0 0 1 1 1 0 1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 1

↓

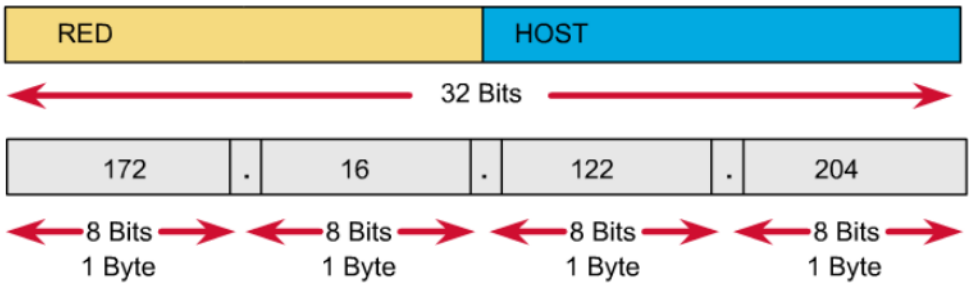
1 1 0 0 1 1 0 0. 0 0 1 1 0 0 1 1. 1 0 1 0 1 0 1 0. 0 1 0 1 0 1 0 1

↓

204. 51. 170. 85

A efectos de direccionamiento y de encaminamiento las direcciones IP constan de dos partes:

- **Identificador de red**: determina la red en la que se encuentra del dispositivo. Es decir identifica el segmento de red.
- **Identificador del host**: identifica el equipo dentro de ese segmento de la red.

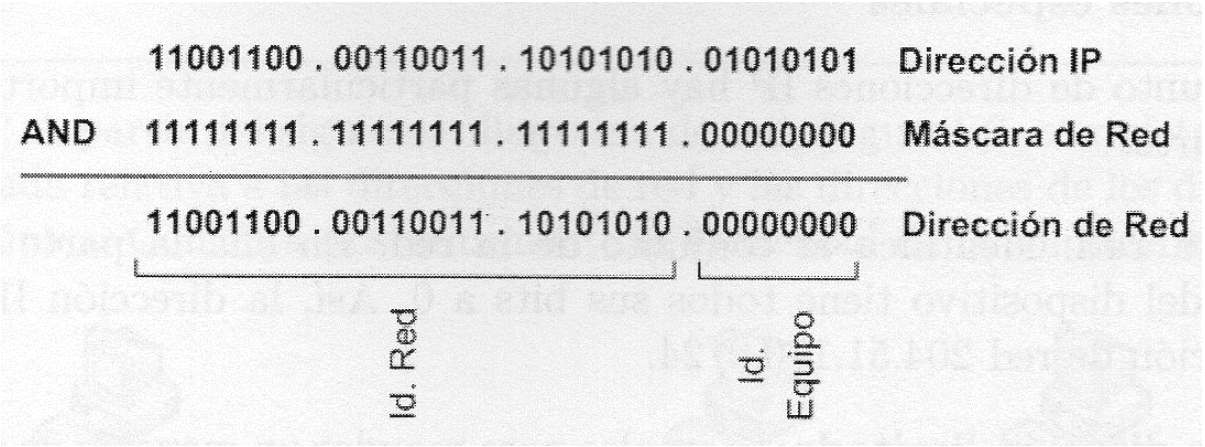


De esta forma todos los host dentro de una misma red comparten la parte de identificador de red. Además, el identificador de red podrá tener el valor que se quiera en función del tamaño de la red. Así, las redes grandes tendrán un identificador de red pequeño y las redes pequeñas tendrán un identificador de red grande.

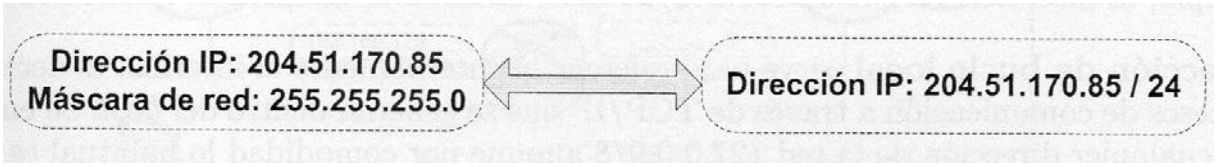
- **Máscara de red.**

La máscara de red se emplea para diferenciar el prefijo de la dirección IP correspondiente al **identificador de red**, de la parte correspondiente al identificador del host. La máscara de red es un número de 32 bits que define en las posiciones a '1' el prefijo o **identificador de red**, y en las posiciones a '0' el sufijo o **identificador del host**.

El host compara esta máscara con la dirección IP, de tal manera que todos los bits de la dirección IP que coincidan con un uno en la máscara pertenecen a la dirección de la subred, mientras que los bits de la dirección IP que coincidan con un cero de la máscara corresponden a identificar el equipo.



La máscara de red también puede expresarse mediante la **notación CIDR** (Classless Inter-Domain Routing) consistente en situar un sufijo a continuación de la dirección IP que indica cuantos bits de la máscara de red están a 1.



Notación CIDR

NOTACIÓN CIDR	MÁSCARA SUBRED
/8	255.0.0.0
/16	255.255.0.0
/24	255.255.255.0

- **Clases de direcciones IP.**

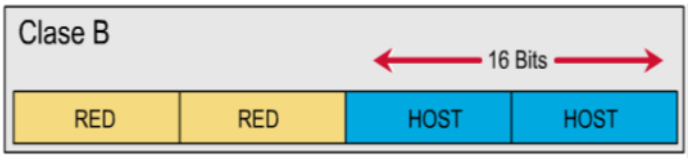
  En un principio se predeterminaron una serie de máscaras de red concretas para facilitar el proceso de encaminamiento. Surge así el concepto de clases de direcciones. Así:

  - **Clase A**: Un byte para el identificador de red y tres bytes para el identificador de host (redes grandes).

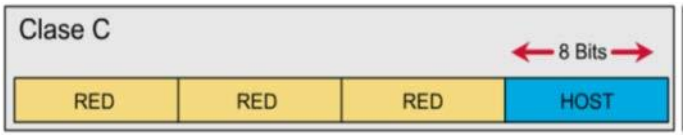
    La máscara de red equivalente es 255.0.0.0

- **Clase B**: Dos bytes para el identificador de red y dos bytes para el identificador del host (redes medianas).



La máscara de red equivalente es 255.255.0.0

- **Clase C**: Tres bytes para el identificador de red y un byte para el identificador del host (redes pequeñas).



La máscara de red equivalente es 255.255.255.0

- **Clase D**: Se emplea para multicast, es decir, envío de datagramas a un grupo de equipos de la red. No diferencian entre identificador de red y de host.
- **Clase E**: Reservadas para uso experimental en proyectos de investigación.

■ **IPv6.**

Actualmente se está utilizando la versión IPv4.

IPv4 establece 4 bytes para una dirección. 4 bytes = 32 bits. Pero la demanda mundial de direcciones ha excedido las posibilidades de este formato. El rango debe incrementarse de alguna manera. En este sentido ha surgido IPv6.

Con este nuevo formato cada dirección se compone de 128 bits. Una dirección se representa en 8 grupos de 4 dígitos cada uno en notación hexadecimal. Los grupos se separan entre sí mediante dos puntos (:).

Ejemplo: 1101:0db8:85a3:08d3:1319:3a2f:0370:7234

■ **Direcciones especiales.**

Dentro del conjunto de direcciones IP hay algunas particularmente importantes que merecen una explicación aparte:

- **Dirección de red**: Identifica al conjunto de la red. En ella la parte correspondiente al identificador del dispositivo tiene todos los bits a 0. Así, la dirección IP 204.51.170.85/24 tiene la dirección de red 204.51.170.0/24
- **Dirección de difusión limitada**: Se emplea para mandar un mensaje de difusión o broadcast al conjunto de dispositivos de la propia red. Es la misma para todas las redes (255.255.255.255)

- **Dirección de difusión dirigida**: Se emplea para mandar un mensaje de difusión o broadcast al conjunto de dispositivos de una red. Por tanto, no puede asignarse a un interfaz de red en concreto. Viene dado por el identificador de la red en la que queramos hacer la difusión a la izquierda y los bits correspondientes a la dirección del dispositivo todos 1 a la derecha. Por ejemplo, la dirección IP 204.51.170.255/24 hace difusión dirigida en la red 204.51.170.0/25

- **Dirección de bucle local**: Sirve para referenciar internamente al interfaz, es decir, para los procesos de comunicación a través de TCP/IP que se generan dentro del host. Se emplea para ello cualquier dirección de la red 127.0.0.0/8. Aunque por comodidad lo habitual es utilizar la dirección 127.0.0.1/8

■ **Direcciones públicas y privadas**

Dentro del espacio de direcciones hay algunas que se han reservado para un uso privado, es decir, son direcciones que no deben tener acceso a Internet. Se distingue así entre:

- **Direcciones públicas**: identifican un dispositivo conectado a Internet.

- **Direcciones privadas**: son rangos de direcciones reservados para redes privadas o intranets y no pueden emplearse en Internet. Son las pertenecientes a las siguientes redes: 10.0.0.0/8, 172.16.0.0/16 y 192.168.0.0/16. Los rourters conectados a redes públicas (Internet) descartan el tráfico dirigido a direcciones privadas como medida adicional de la seguridad.

■ **Direcciones de enlace local.**

Hay un conjunto de direcciones reservado para ser asignado en redes que no dispongan de direccionamiento estático ni dinámico, es decir, son direcciones que se asignan los dispositivos a si mismos cuando nadie lo hace. Se emplean las direcciones de red 169.254.0.0**/**16. Estas direcciones tampoco pueden circular por Internet.

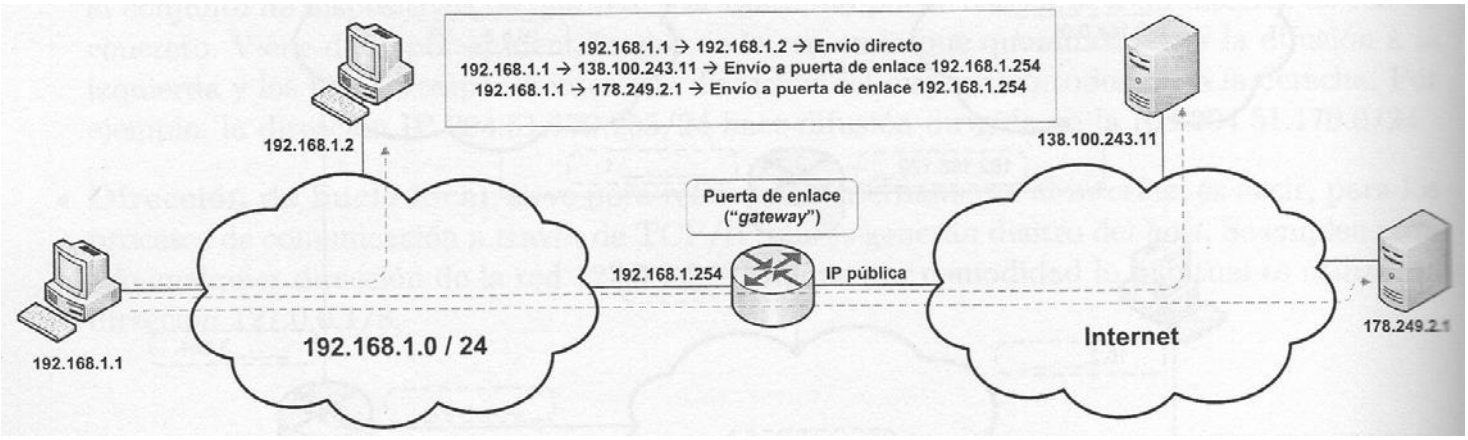
2.2.- Encaminamiento IP

El **encaminamiento** a nivel IP puede definirse como el proceso de llevar un datagrama desde la máquina origen a la máquina destino, independientemente de si ambas máquinas se encuentran en la misma o diferentes redes. El protocolo IP es el responsable de este encaminamiento.

■ **Encaminadores.**

Los **encaminadores o routers** son dispositivos de nivel 3 que enlazan las diferentes redes que forman parte de una “red de redes” y van a desempeñar un papel crucial en el encaminamiento de los datagramas. Un router está conectado al menos a dos redes y realizará el encaminamiento de todo el tráfico de datagramas que pase por él. Para ello se servirá de las tablas de encaminamiento que veremos mas adelante.

Además de los routers, los propios **equipos** también van a participar en el encaminamiento. De hecho, el encaminamiento se inicia incluso antes de que un equipo coloque un datagrama en la red. Es importante apreciar que a nivel de dirección IP y desde el punto de vista de un equipo, el mundo se divide en dos, las direcciones que están en su misma red y luego todas las demás, independientemente de en que red se encuentren. Por tanto, cuando un equipo desee poner un datagrama en la red debe decidir si la dirección IP de destino está en su misma red, con lo que el envío seria **directo**, o bien, no está en su misma red y en este caso se enviaría a un **encaminador (puerta de enlace),** para que lo encamine hacia su destino (envío **indirecto**).



Un matiz importante es que los equipos solamente encaminan el tráfico saliente, a diferencia de los routers que encaminan todo el tráfico que pasa por ellos.

El router, una vez recibe el datagrama, debe encaminarlo. Pueden darse dos situaciones:

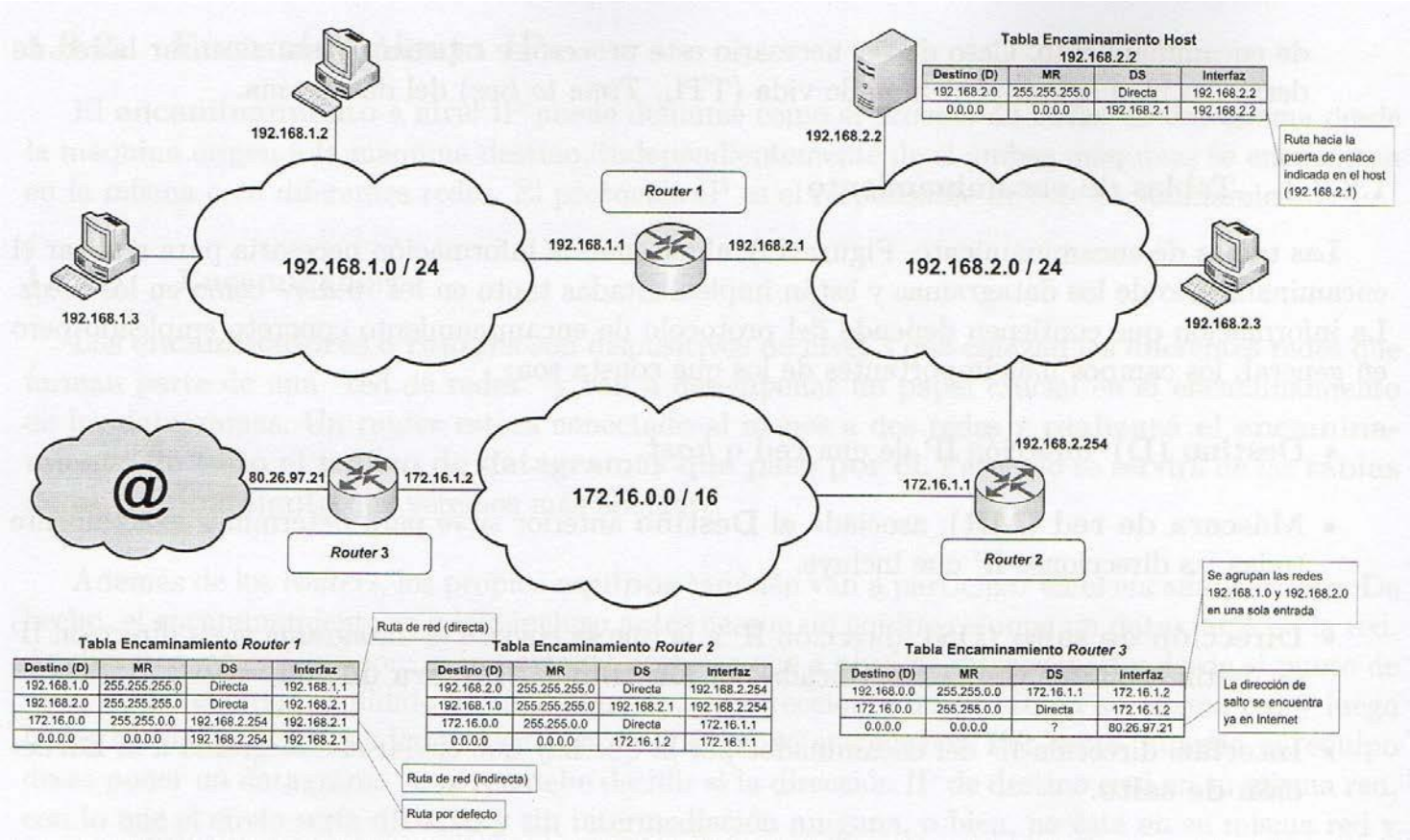
- Que el datagrama vaya dirigido a una dirección que pertenezca a una red conectada al router directamente, en cuyo caso la entrega es directa.
- Que el datagrama no vaya dirigido a una red conectada directamente al router, en cuyo caso reenviará el datagrama hacia otro encaminador siguiendo lo especificado en su tabla de encaminamiento. Caso de ser necesario este proceso se repetiría hasta alcanzar la red de destino o bien agotar el tiempo de vida (TTL, Time to live) del datagrama.

■ **Tablas de encaminamiento**

Las tablas de encaminamiento almacenan la información necesaria para realizar el encaminamiento de los datagramas y están implementadas tanto en los routers como en los host.

La información que contienen depende del protocolo de encaminamiento concreto empleado pero en general, los campos más importantes de los que consta son:

- **Destino (D):** dirección IP de una red o host
- **Máscara de red (MR):** asociada al **Destino** anterior sirve para determinar exactamente todas las direcciones IP que incluye.
- **Dirección de salto (DS):** dirección IP a la que se enviará el datagrama si su dirección IP de destino coincide con la especificada por **Destino** y **Máscara de red**.
- **Interfaz:** dirección IP del encaminador por la que hay que enviar el datagrama a la Dirección de salto.



Cada registro de la tabla encamina a ese destino en concreto. En general se distinguen tres tipos de destino diferentes:

- **Ruta de red**: cuando la entrada de la tabla de encaminamiento se refiere a toda una red.

- **Ruta de host:** cuando la entrada de la tabla de encaminamiento se refiere a un host o equipo.
- **Ruta por defecto**: cuando ninguna entrada de los tipos anteriores es aplicable. Se puede disponer de una ruta por defecto para todas las redes no consideradas previamente.

■ **Protocolos de encaminamiento.**

Al arrancar los equipos, tanto routers como host, las tablas de encaminamiento se inicializan con las rutas correspondientes a las redes adyacentes al mismo. A partir de este momento, se pueden distinguir dos estrategias para configurar las tablas de encaminamiento:

- **Encaminamiento estático**: la configuración de las tablas de encaminamiento se hace de forma manual. Es una estrategia no adaptativa, es decir que cualquier cambio que se produzca en la topología de la red debe ser supervisado por el administrador para evitar rutas imposibles o bucles indeseados. Por esto mismo es muy sensible a fallos y solo recomendable en redes de pequeño tamaño y topología fija.
- **Encaminamiento dinámico**: El propio encaminador actualiza sus tablas gracias a la utilización de protocolos específicos como **RIP** (Routing Information Protocol) **OSPF** (Open Shortest Path First) y **BGP** (Border Gateway Protocol) que permiten que los encaminadores intercambien información de encaminamiento para mantener sus tablas lo más actualizadas posibles.

3.- Nivel de transporte en TCP/IP – Protocolos TCP y UDP.

En los sistemas operativos multitarea y en red actuales es habitual que las comunicaciones impliquen, de forma simultanea, a varios procesos dentro de una misma máquina y a varias máquinas dentro de la red.

Hasta ahora, hemos visto que el protocolo IP nos permite comunicar dos máquinas remotas haciendo que los datagramas puedan ir del origen al destino, pero al disponer únicamente de las direcciones IP de origen y destino como mecanismo de diferenciación en la comunicación, se plantea el problema de que no nos permite, por ejemplo, mantener varias comunicaciones simultaneas entre dos mismos equipos, ya que a nivel IP no podríamos diferenciar los datagramas pertenecientes a unas u otras.

El nivel de transporte nos provee de elementos para diferenciar y gestionar, de forma simultánea, múltiples orígenes y destinos en una comunicación y múltiples comunicaciones en cada equipo. También permite identificar los extremos finales en la comunicación y nuevos servicios orientados a la conexión.

3.1.- Puertos de comunicaciones.

Los protocolos del nivel de transporte implementan el concepto de puerto de comunicaciones que nos permite identificar los procesos del nivel de aplicación entre los que está establecida una comunicación.

Así, cada proceso del nivel de aplicación tiene asociado uno o varios puertos a través de los cuales es accesible. Cada puerto se identifica por un número binario de 16 bits que en notación decimal puede variar entre 0 y 2^{16} - 1= 65535.

En el nivel de transporte disponemos de dos protocolos completamente independientes (TCP Y UDP) y ambos manejan el concepto de puerto, pero es importante destacar que los puertos TCP y UDP son totalmente independientes.
Existen varias clases de puertos en función del uso que se hace de ellos:

- **Puertos conocidos (0-1023):** se conocen como well known ports y están reservados para aplicaciones y servicios estándar como HTTP, FTP, etc. Las aplicaciones clientes se conectan a estos puertos para acceder a los servicios.
- **Puertos registrados (1024-49151):** para aplicaciones no estándar instaladas por el usuario que no tienen un puerto well known preasignado. Estos puertos pueden asignarse dinámicamente a clientes si ningún servicio está haciendo uso de ellos.
- **Puertos dinámicos (49152-65535):** habitualmente se emplean para iniciar conexiones desde el cliente. No suelen emplearse en procesos servidores.

La correspondencia entre procesos y puertos se hace de dos formas distintas:

- **Asignación estática**: **los well known ports** están reservados para aplicaciones estándar y solo pueden ser empleados por estos procesos.
- **Asignación dinámica**: cuando un proceso necesita un puerto y este no se asigna estáticamente, el sistema operativo le asigna uno que esté disponible (1024-65535).

3.2.- Protocolo UDP.

El protocolo UDP (User Datagram Protocol) proporciona un servicio no orientado a la conexión (al igual que IP) con todo lo que esto supone: sin establecimiento de conexión previo a la transmisión, sin control de flujo (existirá la posibilidad de entrega de segmentos duplicados o desordenados), etc.
Al tratarse de un protocolo muy básico suele emplearse en casos en los que prevalece más la velocidad de la transmisión respecto a la fiabilidad de la misma o bien en aplicaciones con requerimientos sencillos del tipo petición respuesta como DHCP, DNS y voz IP.

3.3.- Protocolo TCP

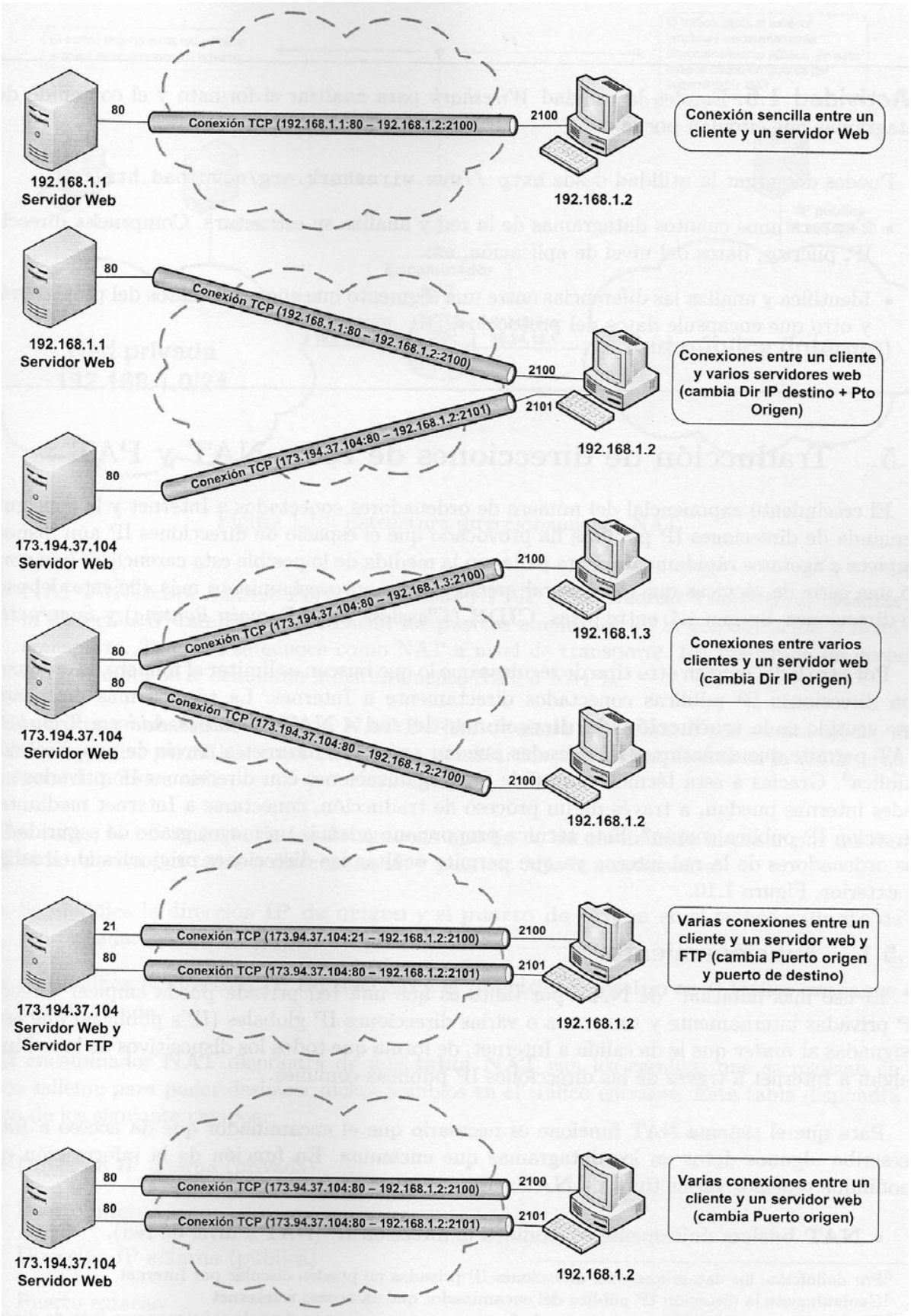
El protocolo TCP (Transmission Control Protocol) proporciona un servicio orientado a conexión con lo que existen grandes diferencias respecto a UDP. TCP obliga al establecimiento previó de una conexión antes de empezar a transmitir y ofrece control de flujo y de errores con lo que garantiza al nivel de aplicación un servicio fiable.

- **Conexiones TCP.**

  La conexión TCP es el paso previo imprescindible para iniciar una comunicación. Una vez se ha establecido la conexión cualquiera de los extremos de la misma puede empezar a transmitir y también terminar la conexión en el momento que lo desee. La conexión TCP se define de forma única por los datos relativos a los puntos extremos de la comunicación, es decir, por estos cuatro elementos:

  (**Dirección IP origen**, **Puerto TCP origen)=>** (**Dirección IP destino**, **Puerto TCP destino**).

  No puede haber dos conexiones TCP que tengan en común esos cuatro elementos. Se aprecia claramente en la figura siguiente:


80
Conexión TCP (192.168.1.1:80 – 192.168.1.2:2100)
2100
192.168.1.1
Servidor Web
192.168.1.2
Conexión sencilla entre un cliente y un servidor Web
80
Conexión TCP (192.168.1.1:80 – 192.168.1.2:2100)
2100
192.168.1.1
Servidor Web
2101
Conexión TCP (173.194.37.104:80 – 192.168.1.2:2101)
192.168.1.2
Conexiones entre un cliente y varios servidores web (cambia Dir IP destino + Pto Origen)
80
173.194.37.104
Servidor Web
Conexión TCP (173.194.37.104:80 – 192.168.1.3:2100)
2100
192.168.1.3
80
Conexión TCP (173.194.37.104:80 – 192.168.1.2:2100)
2100
173.194.37.104
Servidor Web
192.168.1.2
Conexiones entre varios clientes y un servidor web (cambia Dir IP origen)
21
Conexión TCP (173.94.37.104:21 – 192.168.1.2:2100)
2100
80
Conexión TCP (173.94.37.104:80 – 192.168.1.2:2101)
2101
173.194.37.104
Servidor Web y
Servidor FTP
192.168.1.2
Varias conexiones entre un cliente y un servidor web y FTP (cambia Puerto origen y puerto de destino)
80
Conexión TCP (173.94.37.104:80 – 192.168.1.2:2100)
2100
80
Conexión TCP (173.94.37.104:80 – 192.168.1.2:2101)
2101
173.194.37.104
Servidor Web
192.168.1.2
Varias conexiones entre un cliente y un servidor web (cambia Puerto origen)

4.- Traducción de direcciones de red NAT y PAT.

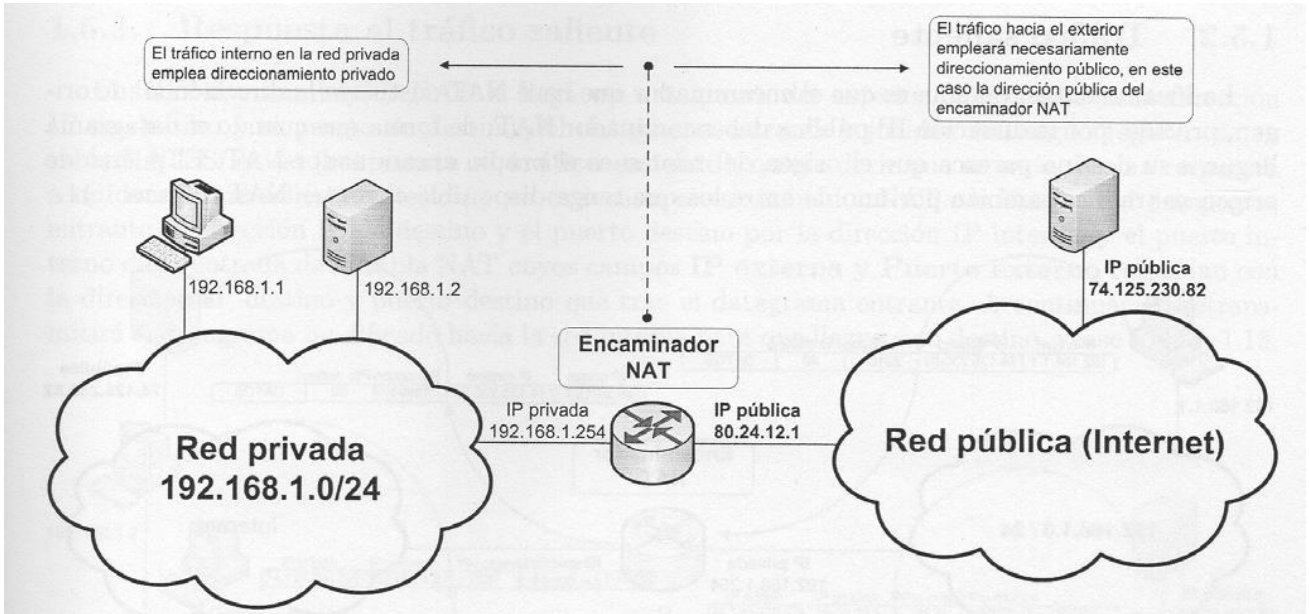
El crecimiento exponencial del número de ordenadores conectados a Internet y la consiguiente demanda de direcciones IP públicas ha provocado que el espacio de direcciones IP aún disponible empiece a agotarse rápidamente. Para paliar en la medida de lo posible esta carencia se han empleado una serie de técnicas que, en general, persiguen un aprovechamiento más eficiente del espacio de direcciones. Surgen así, entre otras, **CIDR** (Classless Inter-Domain Routing) y **Supernetting.**

Por otro lado, existen otro tipo de técnicas que lo que buscan es limitar el número de ordenadores con direcciones IP públicas conectados directamente a Internet. La técnica más destacada en este sentido es la **traducción de direcciones de red –NAT** (Network Address Translation). **NAT** permite que direcciones IP privadas puedan acceder a Internet a través de una dirección IP pública. Gracias a esta técnica es posible que organizaciones con direcciones IP privadas en sus redes internas puedan, a través de un proceso de traducción, conectarse a Internet mediante una dirección IP pública común. Esta técnica proporciona además un mayor grado de seguridad para los ordenadores de la red interna ya que permite ocultar las direcciones originales al circular por el exterior.

4.1.- Funcionamiento.

El uso más habitual de **NAT** por tanto es que una red privada pueda emplear direcciones IP privadas internamente y tener una o varias direcciones IP globales (IP's públicas) que estarán asignadas al router que da salida a Internet, de forma que todos los dispositivos de la red interna salgan a internet a través de las direcciones IP públicas comunes.

Para que el sistema **NAT** funcione es necesario que el encaminador que da acceso a Internet reescriba algunos datos en los datagramas que encamina.



En función de la información que se modifique tenemos varios tipos de **NAT**:

- **NAT básico**: únicamente se modifica la dirección IP (NAT a nivel de red).
- **NAPT** (Network Address Port Translation)**/ PAT** (Port Address Translation): además de la dirección IP también se modifican los puertos empleados en la comunicación a nivel de transporte. También se conoce como **NAT a nivel de transporte**. Ha sustituido de hecho a NAT y ahora se denomina indistintamente **NAT** o **NAPT**.

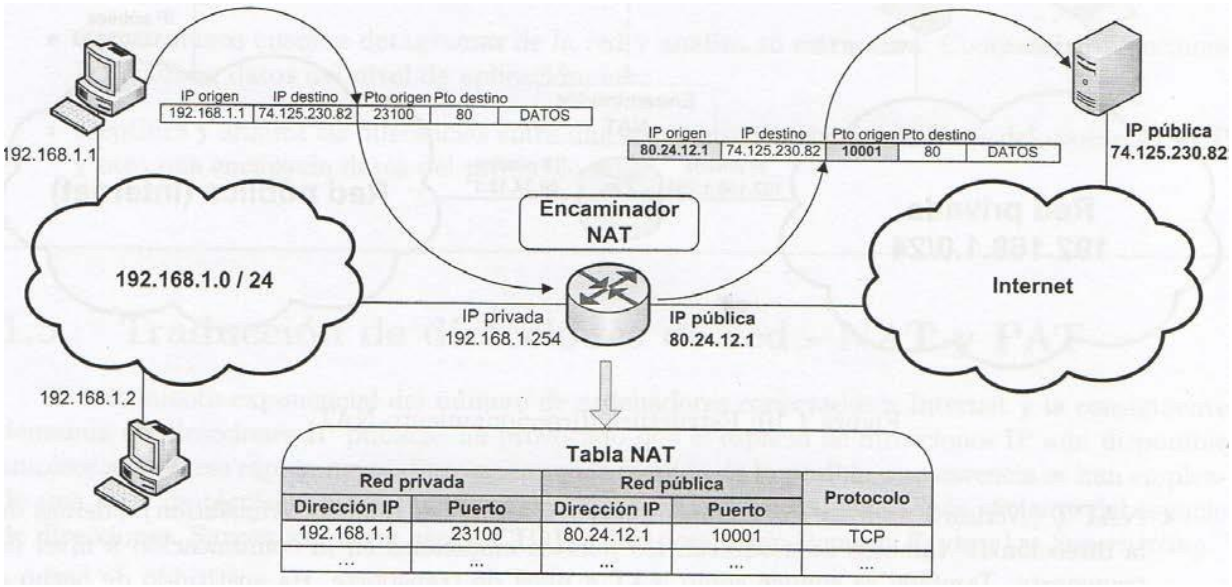
Por tanto, la información de los datagramas que ha de ser modificada al pasar por un router NAT es la relativa por tanto a direcciones IP y puertos. Más concretamente:

- Se modifica la **dirección IP de origen** y el **puerto de origen** en el tráfico **saliente** de la red privada.
- Se modifica la **dirección IP de destino** y el **puerto de destino** en el tráfico **entrante** en la red privada.

El encaminador **NAT** dispondrá de una tabla **NAT** con los cambios que se realizan en el tráfico saliente para poder deshacer dichos cambios en el tráfico entrante. Esta tabla dispondrá al menos de los siguientes campos:

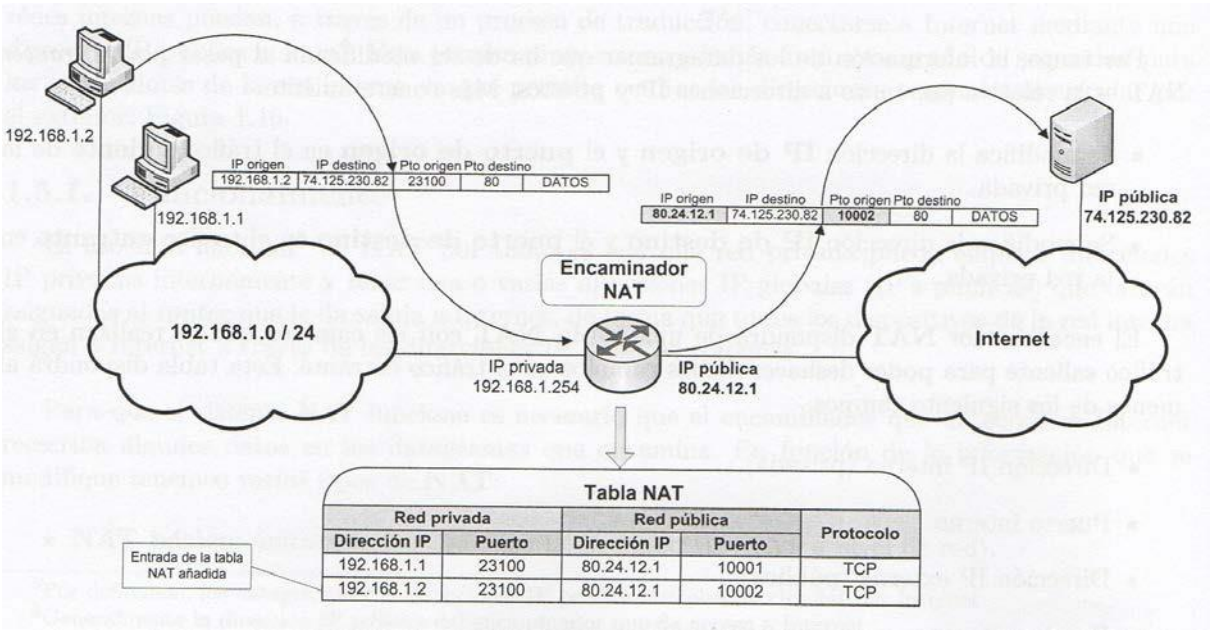
- Dirección IP interna (privada).
- Puerto interno.
- Dirección IP externa (pública).
- Puerto externo.
- Protocolo a nivel de transporte (TCP o UDP).

4.2.- Tráfico saliente.



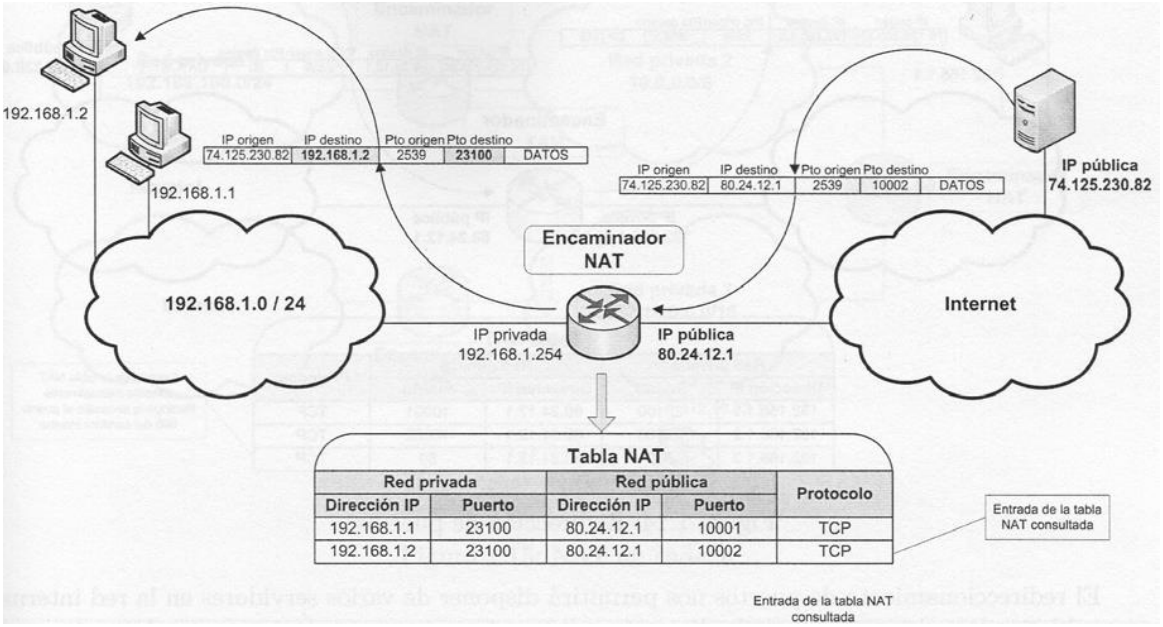
La idea fundamental aquí es que el encaminador que hace NAT sustituya la dirección IP de origen, privada, por la dirección IP pública del encaminador NAT, de forma que cuando el datagrama llegue a su destino parezca que el origen del mismo es el propio encaminador NAT. El puerto de origen se traduce también por uno de entre los que tenga disponible el router NAT.

El encaminador NAT empleará los puertos que tenga disponibles para seguir asignándolos al tráfico saliente.



4.3.- Respuesta al tráfico saliente.

Cuando al encaminador NAT le llega un datagrama del exterior, comprueba si la dirección IP destino y el puerto de destino del datagrama entrante coinciden con algún registro de su tabla NAT.



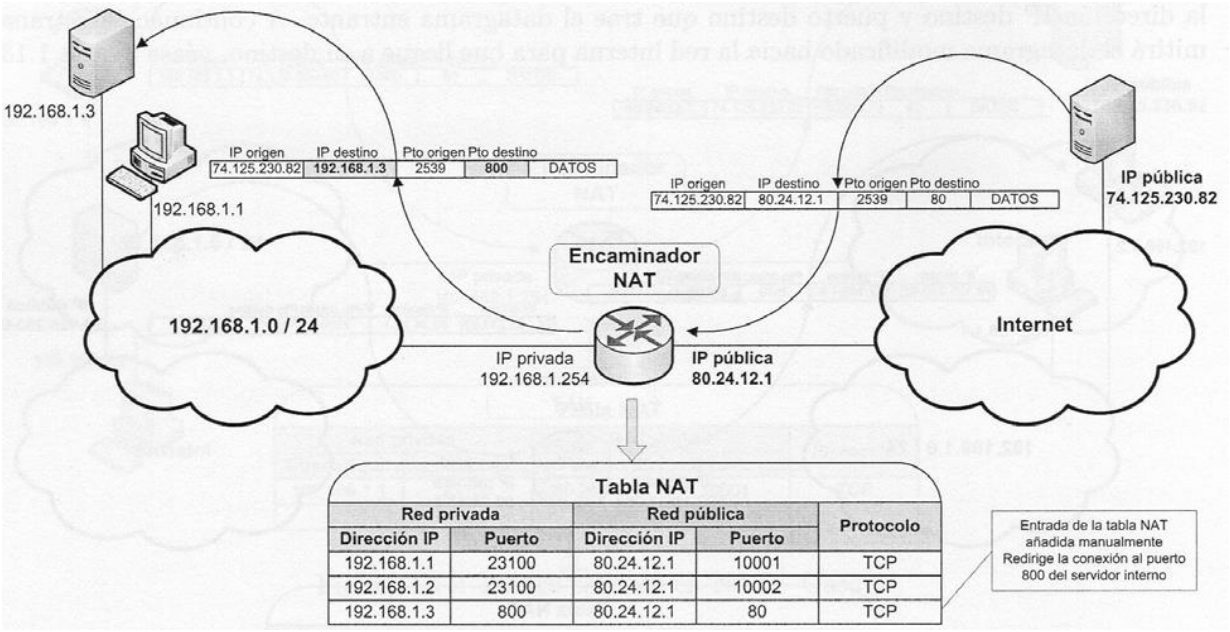
Si es así (**respuesta al tráfico saliente**), quiere decir que se trata de una respuesta a un datagrama saliente anterior, y en este caso, el encaminador NAT cambiará en el datagrama entrante la dirección IP de destino y el puerto de destino por la dirección IP interna, y el puerto interno de la entrada de la tabla NAT cuyos campos IP externa y Puerto externo coincidan con la dirección IP destino y puerto destino que trae el datagrama entrante. A continuación retransmitirá el datagrama modificado hacia la red interna para que llegue a su destino.

Si en cambio, el datagrama que le llega al encaminador NAT no se corresponde con un datagrama saliente previo (**tráfico entrante nuevo**), el encaminador NAT no sabrá a que dirección IP interna y puerto redirigirlo y, en general, **descartará** el datagrama. Existen excepciones a este caso como por ejemplo, que el router NAT tenga un servidor escuchando en un puerto, en cuyo caso sí que aceptará los datagramas entrantes nuevos que vayan dirigidos a ese puerto de escucha. **Por ejemplo, un servidor web escuchando en el puerto 80 del router NAT.**

4.4.- Soluciones al tráfico entrante nuevo. Redirección de puertos.

El hecho de descartar por defecto las conexiones entrantes puede suponer una gran limitación en nuestra red interna, ya que si en ella disponemos de algún tipo de servidor nos va a resultar imposible permitir que clientes de la red externa se conecten a nosotros. Para subsanar esta carencia y a pesar del peligro potencial que supone dejar que se inicien conexiones desde la red exterior, es posible permitir las conexiones entrantes nuevas mediante el mecanismo de **redirección de puertos**.

La redirección de puertos (Port Forwarding) consiste en indicar al encaminador NAT una dirección IP de la red interna a la que redirigir todo el tráfico entrante nuevo. Para ello, deberemos añadir manualmente nuevas entradas en la tabla NAT que redirijan el tráfico entrante nuevo en función del puerto al que vaya dirigido.



De esta forma todo el tráfico entrante dirigido, por ejemplo, al puerto 80 será reenviado a una dirección IP privada (y hacia el puerto que queramos) de nuestra red interna.

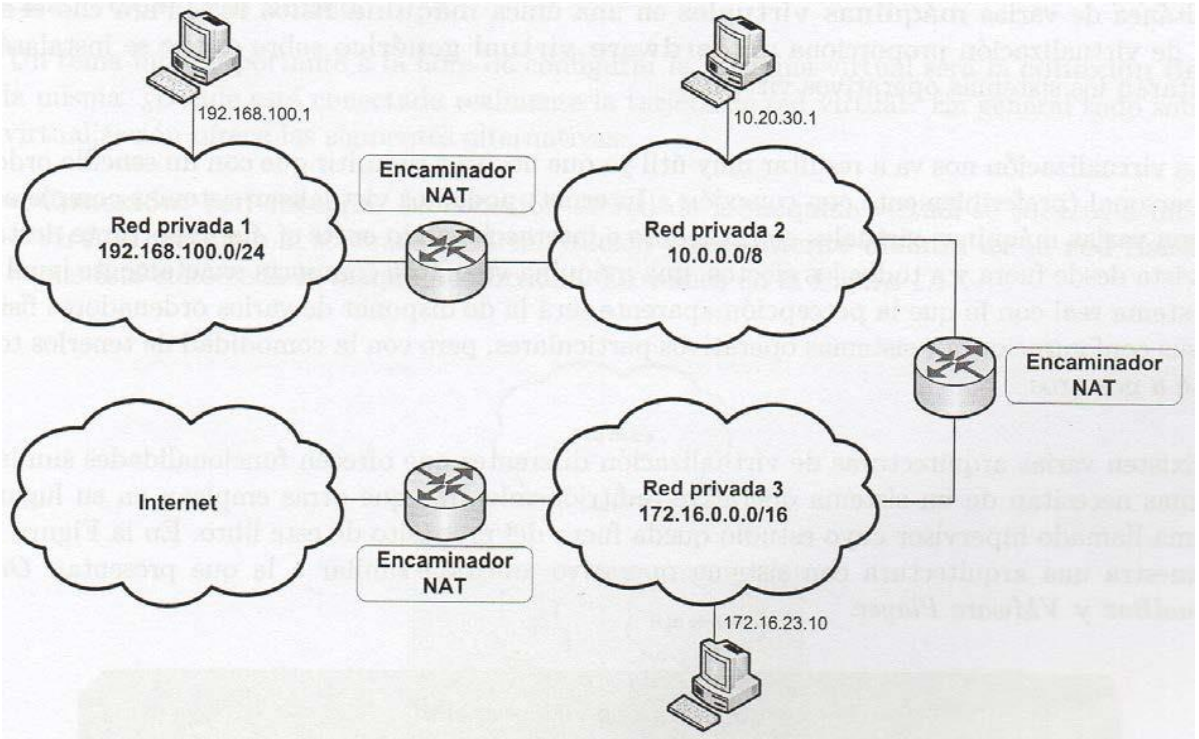
El redireccionamiento de puertos nos permitirá disponer de varios servidores en la red interna, accesibles todos ellos a través de la dirección pública de nuestra red. Únicamente obligará a que, para cada servicio, se acceda a un puerto diferente.

4.5.- Limitaciones de NAT.

Fundamentalmente nos interesa el hecho de que hay protocolos de nivel de aplicación que incluyen las direcciones IP o los puertos empleados en la comunicación en el campo de datos de un datagrama. Como el encaminador NAT solo actúa sobre las cabeceras de los protocolos IP, TCP y UDP pueden darse incongruencias en los datagramas que, por un lado, indiquen una dirección IP o un puerto de cabecera y que luego esta no se corresponda con lo indicado en el campo de datos. Si esto sucede y la máquina de destino utiliza los valores que viajan en la parte de datos no se identificará correctamente al origen.

Para subsanar esto los encaminadores NAT deberían conocer el protocolo concreto de nivel de aplicación encapsulado en cada datagrama para poder modificar el campo de datos del mismo.

Las últimas implementaciones de NAT van incorporando poco a poco los protocolos más habituales que encapsulan direcciones IP y puertos en el campo de datos, poniendo remedio a esta limitación aún a costa de añadir un retardo mayor en el procesamiento de cada datagrama.



Por otro lado, el uso de NAT es escalable, es decir, podemos añadir tantas etapas de enrutadores NAT y redes privadas intermedias como queramos antes de alcanzar la red pública. Esto permitirá un mayor ahorro de direcciones IP públicas.

Para saber más

Los proveedores de acceso a Internet (ISP, Internet Service Provider) emplean NAT para aumentar el número de direcciones IP disponibles para sus clientes. En la mayoría de los países, y siempre según la legislación vigente en materia de protección de datos, el proceso de NAT realizado por los ISP debe limitarse al transporte de datos entre la red externa e interna y, si no existe el consentimiento expreso del usuario final, no autoriza a realizar registros de uso o implementar filtros de contenido. En cualquier caso es un tema controvertido a día de hoy por el conflicto de intereses existente, por un lado, entre los ISP y servidores web que desean un registro de sus usuarios, y por otro lado, el derecho a la privacidad de aquellos que utilizan estos servicios.