Redes de Computadores

https://github.com/AdriCri22/Xarxes-Computadors-XC-FIB

Tema 1 – Introducción

Cuando se habla de transmisión de datos, se habla del **Bitrate**:

$$v_t = 1/t_b$$

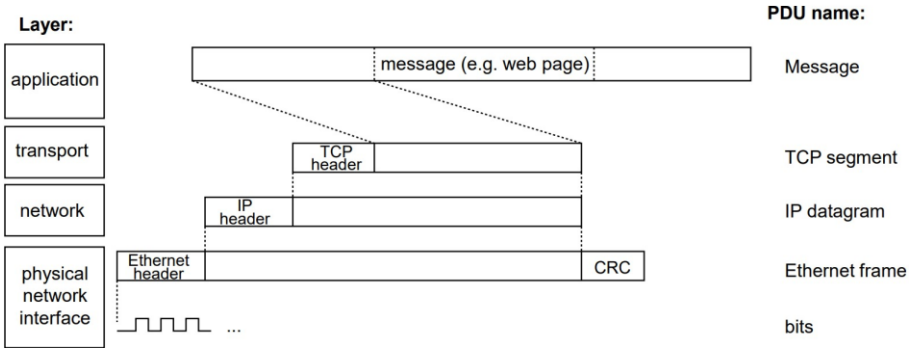
t_b = Tiempo de transmisión de 1 bit

v_t = Tasa de bits por segundo

$k, kilo: 10^3$
$M, Mega: 10^6$
$G, Giga: 10^9$
$T, Tera: 10^{12}$
$P, Peta: 10^{15}$

Tema 2 – Redes IP

Encapsulamiento de la información en niveles o capas



En la transmisión de información a través de la red, cada nivel o capa añade una cabecera con la información necesaria para comunicarse con el siguiente nivel.

La cabecera de una dirección IP contiene la @IP de origen y la @IP de destino.

Fragmentación IP

Un datagrama que se envía por una red se puede fragmentar en el caso de que:

- El router al que le encaminamos el datagrama tiene una MTU (Maximum Transfer Unit), menor que el router al que se lo envía.
- Cuando una aplicación hace una escritura mayor de un datagrama que la MTU de la red.

Inconvenientes:

- Ralentiza los routers.
- Puede reducir la eficiencia debido a fragmentos pequeños.
- Si se pierde un fragmento se descartan todos.

$$Fragment\ size = \frac{Max\ data\ size}{offset\ size} = \frac{\#bytes\ MTU - header\ of\ datagram}{offset\ size}$$

Offset: Posición del primer byte del fragmento (en el primer fragmento el offset es 0), se cuenta en unidades de 8 bytes.

Direcciones IP

`x.x.x.x` → cada x son 8 bits (1 byte)

La IP está dividida en 2 partes:

`Network id`	`Host id`

`x.x.x.x/n` → el /n indica cuántos bits ocupa la netid (también puede contener el #bits del netid + subnetid).

Propiedades:

$2^{\#bits\ host}$ → IPs disponibles en una red

$2^{\#bits\ host} - 2$ → IPs que se pueden asignar, quitando las direcciones:

Netid y el hostid todo 0s → dirección de la red o prefijo.

Netid y el hostid todo 1s → dirección de broadcast.

$2^{\#bits\ host} - 3$ → hosts posibles, restamos uno porque que existe mínimo un router en la red.

Las **direcciones privadas**: se usan para redes (o hosts) que utilizan TCP/IP y no tienen dirección pública, se han reservado las siguientes direcciones (no son enrutables a internet):

- 1 dirección de clase A: 10.0.0.0 ~ 10.255.255.255
- 16 direccione de clase B: 172.16.0.0 ~ 172.31.255.255
- 256 direcciones de clase C: 192.168.0.0 ~ 192.168.255.255

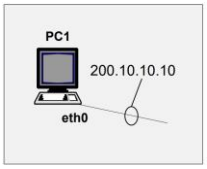
Router

Un **Router** conecta redes IP. Contiene cómo mínimo dos redes IP, ya que un IP conectado a una red sólo no sirve de nada, ya que los hosts que comparten esa red pueden comunicarse sin necesidad de un router. Si función es indicar, con ayuda de las tablas de enrutamiento, dónde enviar un datagrama.

El router recibe por el ip_input un datagrama y a partir de la tabla de enrutamiento decide a que interfaz (ip_output) ha de enviar datagrama, a esto se le llama **ip_forwarding**.

La **tabla de enrutamiento** es una tabla que contiene las redes de destino a las que router sabe llegar, por cada red de destino la tabla dice por cual interfaz ha de enviarse el **datagrama o paquete**.

- Enrutamiento indirecto → Si el que recibe el datagrama NO es el host de destino.
- Enrutamiento directo → Si el que recibe el datagrama es el host de destino.



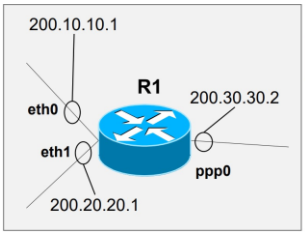
Net Destination	Netmask	Gateway	Interface
200.10.10.0	255.255.255.0	0.0.0.0	eth0
0.0.0.0	0.0.0.0	200.10.10.1	eth0

Comunica el PC1 con el resto de los dispositivos de la misma red

La entrada por defecto comunica el PC1 con el router para enviar paquetes hacia fuera de la red (con quien soy capaz de hablar, que a su vez, sea capaz de hablar con otros de otra red)

Un host ha de tener mínimo un número de reglas: 1 por interfaz y 1 entrada por defecto

- El **Gateway** contiene la dirección IP que se usa para enrutar el datagrama hacia el destino. Cuando vale 0.0.0.0 indica que no se ha hecho uso de ningún router ya que se ha enrutado directamente.
- La **Interfaz** indica la interfaz (la salida) por dónde se ha enviado el datagrama.
- La **máscara** indica la dirección de una red o una subred. Se puede poner en formato /n o 255.255…



Net Destination	Netmask	Gateway	Interface
200.10.10.0	255.255.255.0	0.0.0.0	eth0
200.20.20.0	255.255.255.0	0.0.0.0	eth1
0.0.0.0	0.0.0.0	200.30.30.1	ppp0

Un router que recibe un datagrama dirigido a una dirección desconocida, este router descartará el datagrama, por ello las tablas de enrutamiento del router deben estar inicializadas antes de enviar un datagrama.

Congestión → El router tiene un buffer que en el caso de que se llene, empezara a descartar datagramas.

Un datagrama tiene un TTL (Time To Live), que los routers van decrementando y lo acaban descartando si llega a 0, para evitar que debido a un bug un datagrama se quede rebotando de router en router.

Rutas agregadas: se usa para reducir el tamaño de las tablas de enrutamiento al usar máscaras en vez de clases. Ejemplo:

200.1.10.0/24
200.1.10.11.0/24 → 200.1.10.0/23

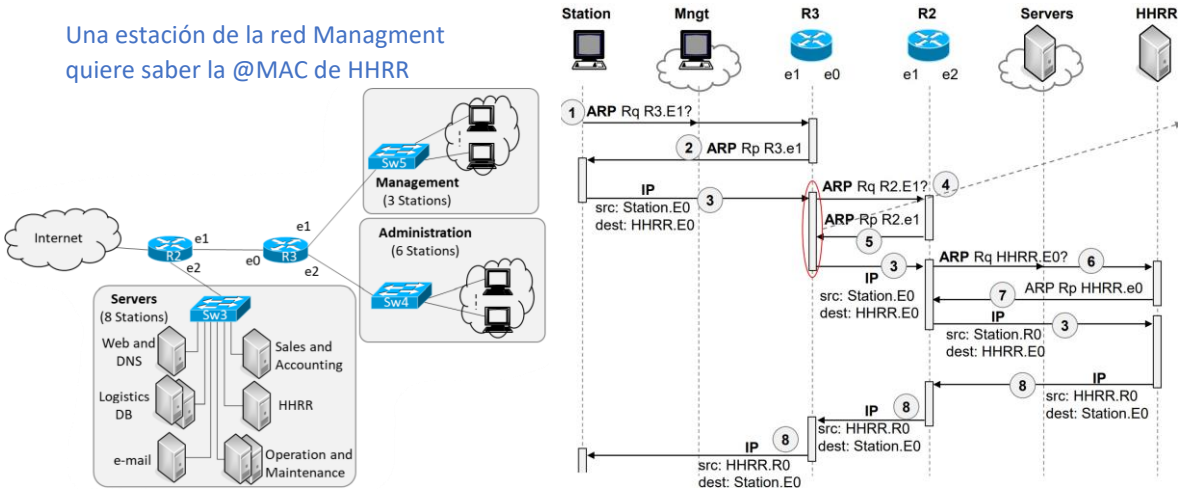
ARP (Address Resolution Protocol)

A veces, después de mirar la tabla de enrutamiento, el nivel IP puede necesitar hacer una conversión para mirar la dirección física (MAC), esto no siempre es necesario, por ejemplo en un enlace de punto a punto, ya que sabemos con seguridad que hay un único receptor, en este caso el datagrama se pasa al driver para hacer la transmisión. En caso contrario el protocolo ARP convierte una @IP en una @MAC, mientras se hace esta conversión, el datagrama se guarda en el buffer que tiene un time-out, si en ese tiempo no se ha hecho la conversión el datagrama es descartado.

El protocolo ARP envía un mensaje broadcast, por lo que los hosts interrumpen la CPU para recibir el mensaje, preguntando quien tiene cierta @IP a todas las máquinas de la red (ARP request):

- Si una máquina tiene asignada esa @IP → Devolverá la @MAC (ARP reply).
- Si no → descartan el ARP request.

Las máquinas implicadas actualizan la cache de la ARP (@IP : eth), sólo se actualizan las caches de las máquinas implicadas debido a que estas caches son pequeñas, para un funcionamiento más rápido. La cache de la ARP se ha de consultar cada vez que se envía un datagrama.



Gratuitous ARP: consiste en que un host envía un ARP request con su propia @IP, los motivos son:

- Detectar al hacer el boot si existe en una misma red otro host con su misma @IP.
- Para actualizar las caches ARP en el caso de cambiar la @IP de un host A a un host B, de esta manera las caches de las otras máquinas actualizarán la @física relacionada con esa @IP, es decir cambiarían la @física de la @IP del host A al host B.

ICMP (Internet Control Message Protocol)

Es usada por los dispositivos de la red para indicar mensajes de error o atención.

- Es un protocolo de la capa IP (no TCP/UDP).
- Los pueden generar una IP/TCP/UDP o un proceso de usuario, pero no pueden ser generados por otro ICMP.
- Pueden ser de dos tipos:
  - Query: este tipo de mensajes tienen un identificador para corresponder un request-reply
  - Error: devuelven la cabecera IP del datagrama que ha causado el mensaje y sus 8 primeros bytes del payload, ya que si el datagrama lleva encapsulado un segmento TCP o UDP, los puertos origen y destino están en estos 8 bytes, a la dirección origen. En el caso del MTU Path Discovery, la TCP necesita conocer los puertos de origen y destino para reducir el tamaño de un segmento, ya que por defecto selecciona el tamaño máximo de segmento, para evitar sobrescribir el header.

Traceroute: envía paquetes con TTL = 1, 2,… y utiliza el ICMP error para ver las distintas IPs que se usan para ir a una determinada IP destino.

DHCP (Dynamic Host Configuration Protocol)

Configura de manera automática los parámetros de configuración de una red (IP, máscara, ruta por defecto, hostname, DNS del dominio y servidores, …).

La asignación de una dirección IP puede ser:

- Dinámica: tiempo limitado.
- Automática: tiempo ilimitado.
- Manual (static): Las direcciones IP son asignadas a una dirección MAC específica.

DCHP está basado en UDP, pero el cliente y el servidor DHCP deben estar en la misma red IP. El protocolo funciona de la siguiente manera:

1. El cliente envía un mensaje DHCPDISCOVER con dirección IP origen 0.0.0.0 y destinación de broadcast (255.255.255.255), por el puerto origen 68 y puerto destino 67. Este mensaje puede sugerir las opciones que desee el cliente, como la dirección IP y el tiempo de leasing.
2. Cada servidor responde con destinación broadcast con un mensaje DHCPOFFER con la oferta de los parámetros de configuración, uno de estos parámetros es el identificador del servidor.
3. El cliente responde con un mensaje DHCPREQUEST broadcast (para que lo reciban todos los servidores) con el identificador del servidor que había en el mensaje DCHPOFFER que ha escogido.
4. El servidor escogido confirma la configuración enviando un mensaje DHCPACK a la dirección broadcast y a partir de este momento la dirección del cliente deja de ser 0.0.0.0 y pasa a ser la dirección de IP pactada con el servidor.

En el caso de que el cliente recuerde la IP de la sesión anterior, puede hacer directamente un DCHPREQUEST.

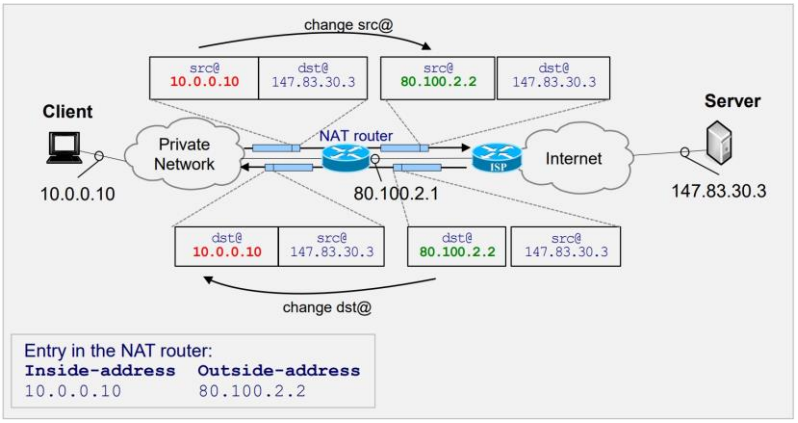
Para prolongar el tiempo de leasing el cliente puede enviar un DHCPREQUEST.

NAT (Network Address Translation)

El NAT es una técnica (no protocolo) que cambia una dirección privada por una pública al momento de acceder a internet, esto se hace porque las direcciones públicas son pocas y cuestan un dinero. El router NAT hace lo siguiente:

- A los datagramas que salen de la red privada, les cambia la dirección origen privada por una dirección pública (una dirección contratada de la ISP).
- A los datagramas que entran a la red privada, como respuesta a los datagramas anteriores, deshace el cambio, es decir, cambia la dirección de destinación pública por la dirección de destinación privada del host.

El host ni siquiera sabe que dirección pública le ha tocado al salir a internet, por lo que para saber los cambios que ha de hacer mantiene una "tabla NAT" con las direcciones públicas y privadas.



Ventajas:

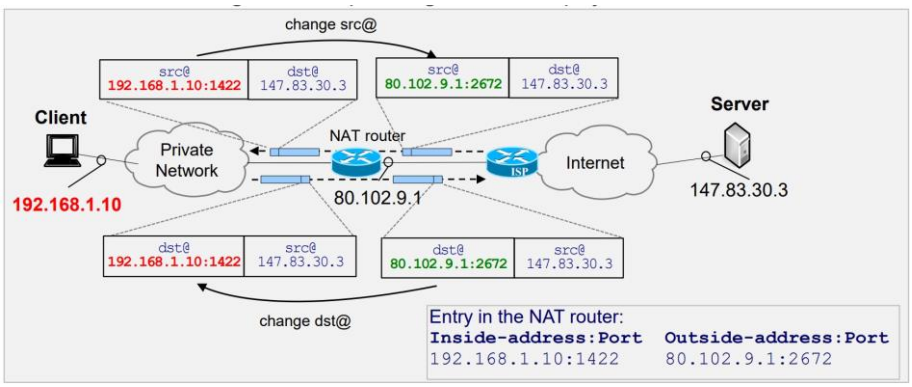
- Ahorro de direcciones públicas, porque hay que contratar menos direcciones públicas.
- No hay que esperar a tener un ISP para asignar a las direcciones privadas y tampoco hay que reasignarlas en caso de cambiar de ISP.
- Añade seguridad: la entrada y salida de la red está controlada por un router NAT (los hosts a los que el router no cambie la dirección no podrán acceder a internet)

Tipo de tablas NAT:

- Estáticas: manualmente se decide que direcciones internas pueden salir a internet.
- Dinámicas: hay un conjunto (pool) de direcciones públicas que se asignan dinámicamente, cada una de estas tiene un timeout, de manera que cuando un host deja de utilizar esa dirección durante un tiempo, la dirección es asignada a otro host.

PNAT (Network Address and Port Translation)

Se usa para que más de un host utilice una misma @pública simultáneamente para acceder a internet. Funciona de la siguiente manera: un host en la tabla NAT se guarda como: (@privada, puerto local), (@pública, puerto externo), de esta manera identificamos un host por una @ y un puerto, por lo que al salir un datagrama (@privada, puerto local) o entrar (@pública, puerto externo), cada par de (@, puerto) a ser diferente.



DNAT (Destination Network Address Translation)

Este es el mecanismo que explica cómo hacer que un servidor sea accesible desde Internet.

En el caso de la DNAT la iniciativa es de un cliente externo, primero se cambia la dirección de destinación, para que los datagramas que envía el cliente puedas entrar a la red interna, dado que antes de que lleguen los datagramas del cliente, la tabla NAT ya está configurada para saber a qué servidores de la red interna deja acceder desde Internet, por lo que la configuración del DNA ha de ser estática.

Algoritmos de enrutamiento

Tienen como objetivo añadir entradas a las tablas de enrutamiento, estas pueden ser:

- Estáticas: se añaden manualmente, mediante scripts o protocolos de configuración cómo DHCP, una vez establecidas las entradas no cambian de valor.
- Dinámicas: hay un "protocolo de enrutamiento" (routing protocol) que se encarga de calcular y añadir entradas a las tablas de manera automática. Tipos de routing protocols:
  - IGP (Interior Gateway Protocol): se establece entre los routers de un mismo AS.
    - RIP (Routing Information Protocol):

      Para saber si ha cambiado la topología de la red, el protocolo envía cada 30 segundos un mensaje RIP con destinaciones y métricas (número de saltos hasta una destinación) conocidas, este mensaje se envía con UDP con una dirección broadcast para cada interfaz dónde se quiera usar RIP. Si se deja de recibir un mensaje RIP de un router vecino durante 180 segundos, se asume que el router vecino ha caído y las entradas que se usan cómo gateway se marcan como inaccesibles (métrica infinita = 16, es decir, que el máximo número de hops (saltos) son 15).

      RIP v2: añade la máscara a las destinaciones y opcionalmente usar dirección de destino multicast.

      RIP NO sabe la topología de la red, sabe únicamente los saltos que hay que dar de un router a otro (y no las conexiones de todos los routers entre estos), para escoger el camino más corto se usa el algoritmo de Bellman-Ford (o Distance Vector Routing), dónde cada router durante el camino indica al datagrama por dónde ir.

      Count to infinity: el problema de RIP es el tiempo de convergencia, es decir el tiempo que pasa des de que hay un cambio en la topología de la red, hasta que las tablas de enrutamiento cambian.

      Split horizon: Al enviar un update a una interfaz, elimina las entradas cuyo getway sean los de la misma interfaz.

    - OSPF (Open Shortest Path First): se usa para redes más grandes y que cambian la topología más frecuentemente.
  - EGP (External Gateway Protocol): se establece entre routers de diferentes AS.
    - BGP (Border Gateway Protocol)

Nota: AS (Autonomous System): conjunto de operadores red que tienen UNO y CLARAMENTE DEFINIDO protocolo de enrutamiento.

Nota: Topología: conjunto de redes interconectados entre sí. (Es como un grafo con nodos (routers) y aristas (conexiones)). Una topología cambia si se añade una conexión o si se quita una conexión (por que ha caído).

Seguridad IP

Objetivos:

- Confidencialidad: quien puede accedes a los datos.
- Integridad: quien puede modificar los datos.
- Disponibilidad: garantizar el poder acceder.

Vulnerabilidades:

- Tecnológicos: protocolos, SO, equipos de la red…
- De configuración: servidores "demasiado permisivos", cuentas no seguras, listas de acceso mal configuradas…
- Falta de políticas de seguridad: aplicaciones poco seguras, contraseñas fáciles, firewalls…

Tipos de ataques:

- De reconocimiento: busca vulnerabilidades de la red (es el paso previo a un ataque), para ello:
  - Se descubren las direcciones IP de la red accesibles (con pings).
  - Descubren los puertos y servidores disponibles.
  - Conectarse al servidor para descubrir el SO y la versión.
  - Eavesdropping: interceptar en tiempo real los datagramas y decodificar el contenido.
- Acceso: acceso de un intruso a una cuenta o servicio.
- Denegación de servicio (Denia lof Service, DoS): deshabilitar o corromper un servicio, red o sistema para que no se pueda acceder.
- Virus, worms, trojan horses… : software malicioso que se autorreplica corrompiendo el sistema atacado.

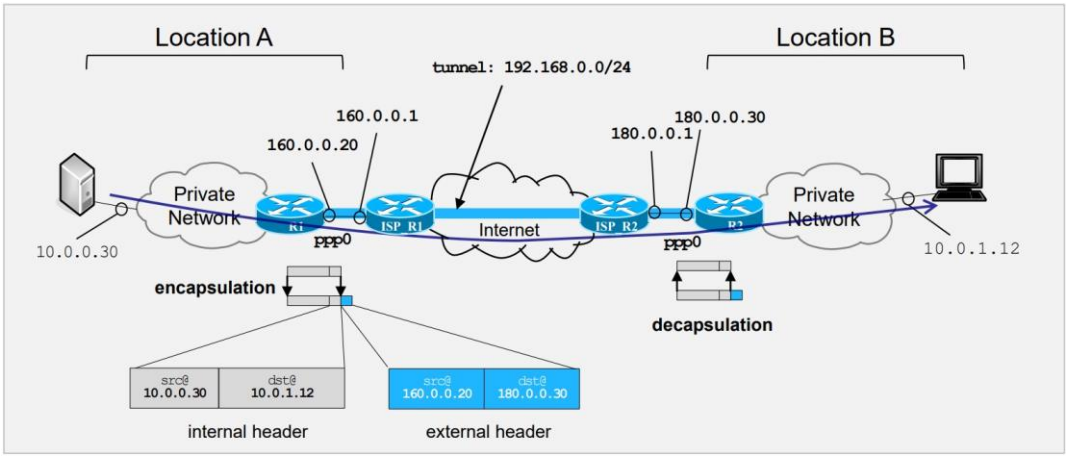
Soluciones básicas:

Firewalls: dispositivos que separa (hace de barrera) entre los posibles intrusos (todo internet) y la red interna. Funciona de la siguiente manera:

- La red interna usa direcciones privadas, el firewall usa NAT para que los hosts de la red interior puedan acceder al exterior.
- El firewall filtra los paquetes que vienen del exterior y no cumplen ciertas condiciones, este filtrado se hace mediante ACL (Acces Control List), que se aplican a la entrada o salida de una interfaz (eth0, eth1, …), estas reglas se miran de manera secuencial y si alguna de las reglas se cumple se deja de mirar.

VPN (Virtual Private Networks): el objetivo es proveer conectividad a los usuarios remotos (Remote Access VPNs) o redes remotas (LAN-to.LAN VPNs) como si estuvieran conectados a la red interna. Para ello se procura que la VPN sea segura, que sólo accedan los usuarios o redes autorizadas, para ello se usan las siguientes técnicas:

- Autentificación: para reconocer al usuario autorizado.
- Encriptación: para evitar eavesdropping (espionaje).
- Túneles: aíslan los usuarios remotos de internet, existen diferentes tipos de túneles.



La cabecera está dividida en 2, la cabecera interna con la dirección origen y destino que referencia las IPs de las máquinas internas y la cabecera externa que indican las direcciones IPs origen y destino de las entradas/salidas de los routers que conectan con la red interna.

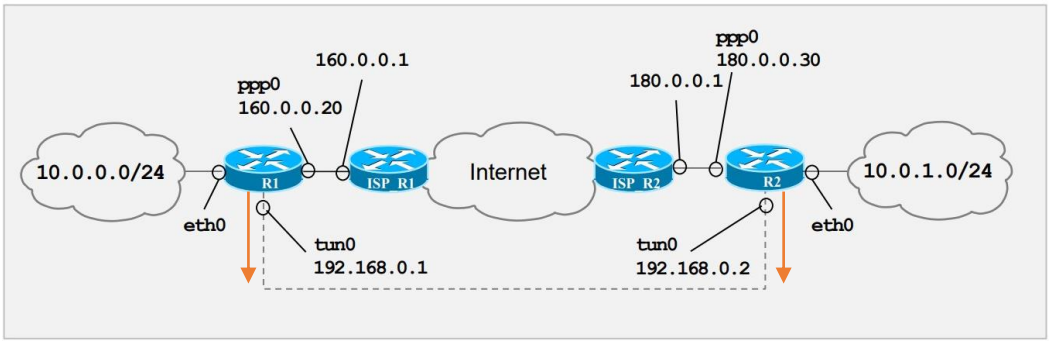


Tabla de enrutamiento de R1		
Dest/mask	**Gw**	**Iface**
160.0.0.1/32	0.0.0.0	ppp0
10.0.0.0/24	0.0.0.0	eth0
192.168.0.0/24	0.0.0.0	tun0
10.0.1.0/24	192.168.0.2	tun0
0.0.0.0/0	160.0.0.1	ppp0

Tabla de enrutamiento de R2		
Dest/mask	**Gw**	**Iface**
180.0.0.1/32	0.0.0.0	ppp0
10.0.1.0/24	0.0.0.0	eth0
192.168.0.0/24	0.0.0.0	tun0
10.0.0.0/24	192.168.0.2	tun0
0.0.0.0/0	160.0.0.1	ppp0

Comunica R1 con las redes a las que está conectado.

Configura el router para que el tráfico que vaya a la red interna que deseamos vaya a través del túnel.

Entrada por defecto para salir por internet e indica cómo llegar al otro extremo del router por el túnel.

Problemas:

- Fragmentación: al añadir la cabecera externa de IP se reduce la MTU.
- ICMP: Si hay un problema dentro del túnel y se genera un error de ICMP, este error llegará al comienzo del túnel, pero no llegará al origen del paquete.
- MTU: puede fallar en el descubrimiento de TCP.

Soluciones:

- La entrada del túnel del router mantiene un “tunnel state”, para luego saber generar paquetes ICMP hacía el origen.
- Procuran hacer la fragmentación en el tramo del túnel y así evitar que lo haga el host de destino.