Capa d'Internet

Maria dels Àngels Cerveró Abelló

30 de desembre de 2015

Índex

1 Introducció

La Capa d'Internet del Model TCP/IP (Capa de Xarxa en el Model OSI) s'encarrega d'enrutar les dades entre el dispositiu d'origen i el dispositiu de destí. Rep el Segment de la Capa de Transport i

l'encapsula dins del Paquet, la capçalera del qual conté les dades que permetran que la informació viatge dins de la mateixa xarxa o entre xarxes.

2 Procés de comunicació entre dispositius finals

Per tal de poder enviar un Paquet entre els dos dispositius finals que s'estan comunicant, la Capa d'Internet utilitza 4 funcions:

1. **Adreçament:** la Capa d'Internet s'encarrega d'identificar cada dispositiu de la xarxa amb una adreça única dins d'aquella xarxa. Aquesta adreça és la IP.
2. **Encapsulació:** quan la Capa d'Internet rep el Segment (o Datagrama) de la Capa de Transport, l'encapsula dins de la seva PDU, que s'anomena Paquet. Per tal de fer-ho, afegeix la capçalera de la capa d'Internet a aquest Segment. Aquesta capçalera conté les IP dels dispositius origen i destí de la comunicació.
   Un cop fet això, el Paquet s'envia a la Capa d'Accés a la Xarxa per tal de poder ser enviat físicament al seu destí.
3. **Enrutament:** la Capa d'Internet gestiona tot un conjunt de serveis que permeten que els Paquets viatgin per les xarxes fins a arribar al seu destí.
   Els routers són els dispositius intermedis que connecten les diferents xarxes i, per tant, són els encarregats de fer l'enrutament, és a dir, de
   (a) seleccionar la ruta i
   (b) enviar el Paquet al seu destí.

   A mesura que el Paquet va viatjant per la xarxa, travessa múltiples routers. Cara duta que agafa el Paquet per tal d'arribar al següent dispositiu s'anomena **salt**.
   El contingut, la porció d'informació, del Paquet (el Segment de la Capa de Transport) sempre queda intacte fins que arriba a destí.
4. **Desencapsulació:** un cop el Paquet arriba al destí, el dispositiu comprova que realment és per a ell (comprova la IP). Si és així, en desencapsula les dades (elimina la capçalera de la Capa d'Internet) i entrega el Segment a la Capa de Transport.

La Capa d'Internet ignora, en tot moment, les dades que transporta dins del seu Paquet (ignora el contingut del Segment). Això li permet funcionar independentment del significat dels bits que transporta. Funciona de la mateixa manera siguin les dades que siguin (vídeo, text, veu...).

3 Protocols

- IPv4 (Protocol d'Internet versió 4) i IPv6 (Protocol d'Internet versió 6): són els protocols estàndard que pertanyen a la Suite TCP/IP
- IPX: pertany a la Suite IPX/SPX, la qual està basada en les MAC i té el problema de què no escala bé.
- Appletalk: pertany a a la Suite d'Apple.
- Servei de xarxa sense connexió: CLNS/DECNet

4 IPv4

La Suite TCP/IP té dos protocols definits per la Capa d'Internet: IPv4 i IPv6. Tot i que IPv6 s'utilitza localment en alguns àmbits i, segurament, serà el protocol que s'acabi imposant a causa del gran creixement d'Internet, actualment el protocol utilitzat per enrutar Paquets a través de les xarxes és IPv4.

Característiques bàsiques d'IPv4

- No està orientat a connexió.
- Entrega amb "Màxim Esforç" (no és confiable).
- És independent del medi físic que transmet les dades (funciona sobre qualsevol tipus de medi).

El fet que siguin un protocol no orientat a connexió i no confiable fa que **no sobrecarregui** la xarxa enviant paquets de control. Si es necessita confiabilitat, seran les capes superiors les encarregades de facilitar els mecanismes corresponents (per exemple, la Capa de Transport por utilitzar el protocol TCP per garantir aquesta confiabilitat). Això també fa que la capçalera del protocol IPv4 sigui molt **petita**, la qual cosa també facilita la **baixa sobrecàrrega** del medi i millora l'**eficiència** en l'entrega de paquets.

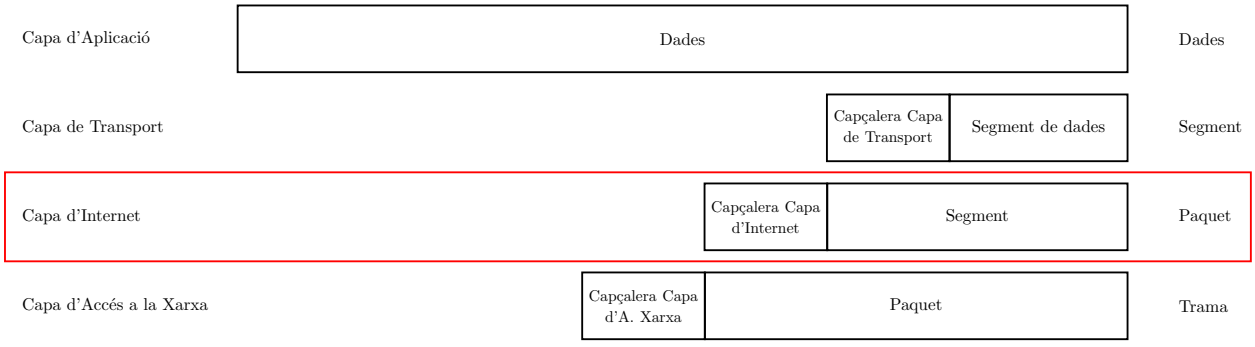
Una de les característiques bàsiques d'IPv4 és que és independent del medi. No obstant això, sí que té en compte la mida màxima de PDU que pot transportar cada medi. Aquesta mida és la **MTU** (Unitat Màxima de Transmissió). Per tal de conèixer aquesta dada, la Capa d'Internet i la Capa d'Accés a la Xarxa s'envien missatges de control. D'aquesta manera, la Capa d'Internet sap quina mida han de tenir els seus Paquets.

Es pot donar la situació que un router comuniqui una xarxa amb una altra un la MTU sigui més petita. En aquest cas, el router haurà de trencar el Paquet original en trossos. D'això se'n diu **Fragmentació de Paquets** o **Fragmentació** directament.

L'RFC del Protocol IPv4 és el RFC-791.

5 Encapsulació de la Capa d'Internet: Empaquetatge

Com ja s'ha dit, la Capa d'Internet rel el Segment (PDU de la Capa de Transport) i l'encapsula dins del Paquet.



Un dels beneficis de l'encapsulació és que ens permet que cada capa tracti la PDU de la capa superior com un bloc de dades hermètic (com un conjunt de bits). Això significa que, per exemple, podem afegir nous protocols a la Capa de Transport sense afectar a les altres capes: la Capa d'Internet continuarà funcionant igual sigui quin sigui el protocol escollit en la Capa de Transport.

A meś a més, en cap moment del procés, la Capa d'Internet modificarà el Segment (PDU de la Capa de Transport) que s'ha empaquetat (excepció: quan s'ha de produir una Fragmentació).

6 Capçalera IPv4

- Adreça IP destí (32 bits)
- Adreça IP origen (32 bits)
- Temps de vida (TTL) (8 bits): temps de vida del Paquet, en segons. Indica el màxim nombre de salts que pot fer el paquet abans d'arribar al seu destí. Cada cop que el Paquet és processat per un router (a cada salt), el seu TTL disminueix en una unitat, com a mínim ja que, qualsevol temps inferior a 1 segon s'arrodoneix a 1. Per exemple, si per anar d'un router a un altre, el Paquet ha tardat 0.3 segons, el TTL disminueix en 1 segon; si ha tardat 2.4 segons, el TTL disminueix en 2 segons (sempre arrodoneix a l'alça).
  Quan el TTL arriba a zero el router elimina el Paquet, el descarta. Això evita que els Paquets quedin viatjant eternament si entren en un "bucle" i no poden arribar al destí. Si això passés, la xarxa es congestionaria.
- Protocol (8 bits): identifica el protocol de la capa superior (TCP, UDP, ICMP...)
- Tipus de servei (8 bits): indica la prioritat del Paquet. Permet gestionar la QoS, ja que el router pot conèixer les prioritats i, si està configurat, decidir quin Paquet s'ha d'enviar abans.
- Desplaçament dels fragments i Senyalització: quan s'ha de produir la fragmentació d'un Paquet IP, el router utilitza la Senyalització **MF** (More Fragments) per indicar que hi ha diversos fragments que s'han d'unir al destí. A més a més, indica l'ordre d'aquests fragments mitjançant el camp de **Desplaçament dels Fragments**, per tal que es puguin reacoblar. La Senyalització **MF** ocupa 1 bit (és un flag) i funciona de la següent manera:
  - Quan MF=1 vol dir que hi ha més fragments. Per tant, el dispositiu de destí va ordenant els fragments segons el que indiqui el cap de Desplaçament.
  - Quan MF=0 i Desplaçament$\neq$0 significa que el fragment és l'últim i, per tant, el dispositiu de destí el posa al final de tots els fragments rebuts.
  - Quan MF=0 i Desplaçament=0, el Paquet no ha estat fragmentat

  Així doncs, MF=1 simplement indica que hi ha més fragments d'un mateix paquet.

  El cap de Senyalització està format per un conjunt de 3 flags (3 bits). Un d'ells és l'MF. UN altre és el DF (Don't Fragment). Si el DF està activat, és a dir DF=1, significa que el Paquet **no** es pot fragmentar. Si un router necessita fragmentar un Paquet per poder-lo fer passar per medi però es troba amb el flag DF activat (DF=1), aleshores descarta aquest Paquet.
- Versió: la versió del protocol (4 o 6).

- Longitud de la capçalera (IHL): mida de la capçalera del paquet. Es mesura en paraules de 4B. Per tant, si tenim un IHL=5, en realitat, la capçalera ocupa $5 \cdot 4 = 20$B.
- Identificació: identificador del Paquet. Quan un Paquet s'ha de fragmentar, tots els fragments tenen el mateix identificador que el Paquet original, per tant, aquest camp permet que el dispositiu destí pugui identificar tots els fragments d'un mateix Paquet.
- Checksum: permet controlar que no hi hagi errors en la capçalera.
- Opcions: gairebé mai s'utilitza. Són dades que es poden utilitzar per donar més capacitats al protocol.

7 Divisió en subxarxes

Si una xarxa creix massa té problemes de gestió, d'eficiència, d'enrutament i d'adreçament. Per aquesta raó, dividim les xarxes en subgrups de dispositius o subxarxes.

Durant la planificació per crear subxarxes cal decidir com agrupar els dispositius. Els factors determinants són:

- Localització geogràfica: millora l'administració i l'operabilitat (eficiència) de la xarxa. Per exemple: un pis, un edifici, etc.
- Objectiu: els usuaris que tenen tasques similars normalment utilitzen les mateixes aplicacions i tenen patrons d'accés a la xarxa comuns. El trànsit de Paquets es pot reduir si els servidors que necessiten aquests usuaris estan dins de la mateixa xarxa que els seus dispositius. A més a més, això també facilita la gestió de permisos d'accés.
- Propietat: defineix una xarxa dins d'una mateixa empresa o organització. Permet gestionar els permisos.

Amb la divisió en subxarxes aconseguim atenuar els 3 problemes bàsics de les xarxes grans:

1. Millorem el rendiment: en una subxarxa amb menys dispositius disminueix el trànsit de trames i baixa la sobrecàrrega dels dispositius i del medi.
   Les trames que ocasionen més trànsit són les de control i, sobretot, les de broadcast. Com més petita sigui una xarxa, menys problemes de trànsit de Paquet tindrà. Per aquesta raó, a les xarxes i subxarxes també se'ls anomena **Domini de Broadcast**.
2. Millor gestió de la seguretat: gràcies a les subxarxes podem prohibir, permetre i/o monitoritzar el trànsit dins d'aquesta subxarxa i l'accés als seus recursos.
   Podem controlar tant el trànsit intern com el que intenta entrar des de fora de la xarxa o el que intenta sortir-ne.
   La seguretat entre xarxes s'implementa en el router o el firewall.
3. Bona gestió de les adreces: si tots els dispositius haguessin de conèixer les adreces IP de la resta de dispositius necessitarien molta memòria i temps de processament en el moment de buscar-ne una. A més a més, les IP haurien de ser més grans de 32 bits. Amb la divisió en subxarxes s'aconsegueix que els dispositius de cadascuna d'elles només hagin de gestionar les adreces dels dispositius de la mateixa xarxa. En el moment en què necessiten saber l'adreça i

l'enrutament cap a un dispositiu fora de la xarxa, deleguen la feina al router. Aquest router és el *gateway*.

8 Adreçament jeràrquic

Per tal de poder fer divisions en subxarxes necessitem l'adreçament jeràrquic.

Amb la jerarquització es permet que cada pas que es doni per fer arribar les trames al destí (cada salt) sigui eficient, ja que no s'ha de tractar tota l'adreça IP sencera, sinó només la part corresponent a la xarxa.

Així doncs, la jerarquització es fa a través de la màscara de xarxa. Per tant, els routers enruten els Paquets tenint en compte només els bits de xarxa de l'adreça. Un cop el Paquet arriba a la xarxa de destí sí que s'utilitzen els bits de host per entregar el Paquet al dispositiu correcte.

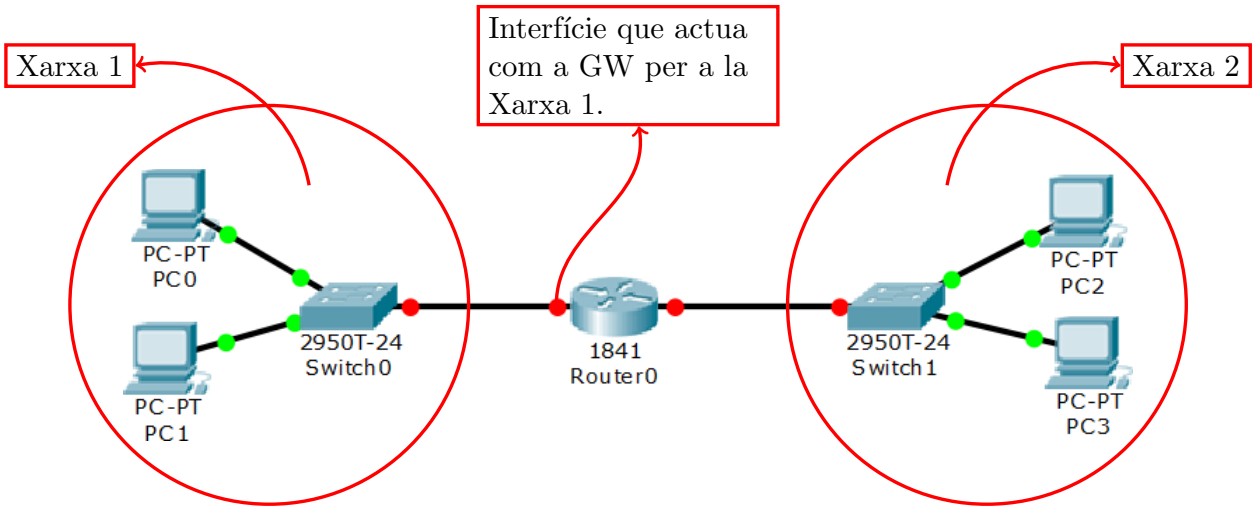
El router, de cara enfora, només necessita saber com arribar a cada xarxa. De cara endins, tracta amb la IP completa per tal del poder entregar les dades als dispositius correctes.

Les subxarxes estan gestionades per la parella IP-Màscara de Xarxa (MX). La màscara de xarxa indica quants bits de la IP estan destinats a definir la xarxa. Si necessitem fer més subxarxes només cal que augmentem el número de bits a 1 de la MX. Ara però, com més subxarxes fem, menys dispositius podrem tenir per xarxa, ja que el número de bits destinats als hosts disminueix.

9 Enrutament

Recordem: un **switch** és un dispositiu intermedi que treballa sobre la Capa d'Accés a la Xarxa (amb les MAC), per tant, només pot manipular trames si aquestes tenen l'origen i el destí dins de la mateixa xarxa (les MAC d'origen i destí són de 2 dispositius de la mateixa xarxa que el switch). En canvi, un **router** és capaç de treballar amb la informació de la Capa d'Internet. Per tant, pot enrutar els Paquets entre xarxes.

El router es coneix com el **gateway** des de la nostra xarxa cap a les xarxes exteriors.



Quan un dispositiu vol enviar informació a un dispositiu d'una xarxa diferent, envia la trama al

seu gateway. És aquest gateway, o router, qui té l'adreça del següent dispositiu a on ha d'enviar les dades (del següent salt). Ens podem trobar, doncs, 2 situacions:

1. Que la xarxa de destí estigui directament connectada al router, amb la qual cosa, aquest router ja podrà entregar el Paquet directament.
2. Que la xarxa de destí no estigui connectada al router, amb la qual cosa, aquest router enviarà el Paquet al següent router que tingui connectat.

Per detectar si el dispositiu de destí està o no està en la nostra mateixa xarxa només cal que analitzem els bits de xarxa de la seva IP (recordem l'adreçament jeràrquic). A més a més, també cal tenir en compte dos punts:

- El gateway (GW) ha d'estar a la mateixa xarxa (IP de la mateixa xarxa) que els dispositius que el tenen com a GW.
- El GW s'ha de configurar a cada dispositiu, ja sigui de manera **manual**, a través dels fitxers de configuració de la xarxa, o **dinàmicament**, a través del servidor DHCP.

Comandes

```
$ifconfig: IP, MX i MAC del PC local
$ip route: taula d'enrutament amb el GW
$route: taula d'enrutament amb el GW
```

10 Taula d'enrutament

La taula d'enrutament d'un dispositiu final només conté dues coses:

1. L'adreça del GW
2. La ruta a la seva pròpia xarxa

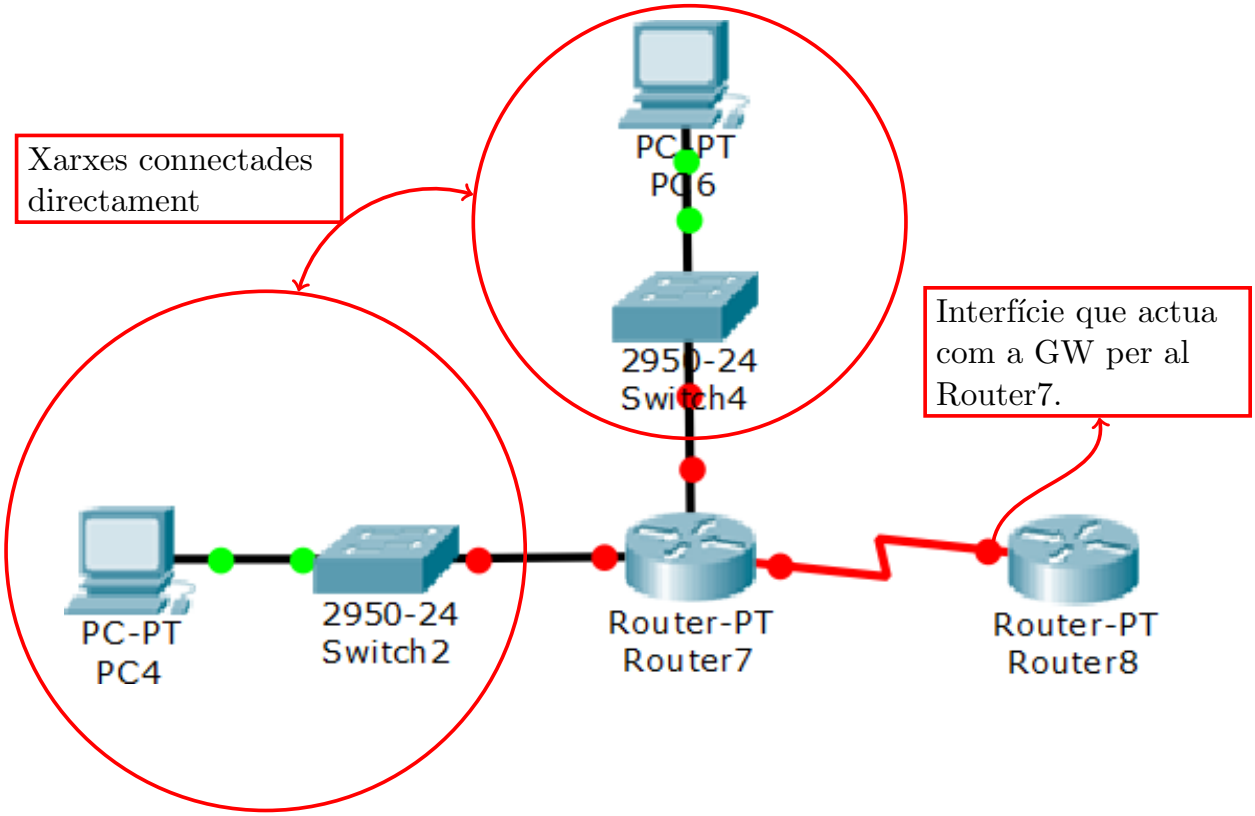
Ara però, manualment s'hi poden afegir més adreces i rutes. Per tal de consultar-la, s'utilitza la comanda

```
$ip route
```

D'aquesta manera,

1. Si el dispositiu de destí està dins de la nostra mateixa xarxa, empaquetarem les dades, utilitzarem la nostra MAC i la MAC del PC de destí i enviarem directament la trama (normalment a través del switch).
2. Si el dispositiu destí no està en la nostra xarxa, empaquetarem les dades, utilitzarem la nostra MAC i la MAC del GW i enviarem la trama. En aquest cas, el switch enviarà les dades al GW i aquest les enrutarà.

El GW també té una taula d'enrutament amb la informació de quina xarxa té connectada a cada interfície i la seva pròpia GW (el següent router).



A més a més, un router pot tenir diferents routers connectats a ell mateix. És a dir, pot tenir diverses GW. Per tal que pugui discernir per quin router ha d'enviar quin paquet perquè arribi a la xarxa correcta es poden configurar més entrades a la taula d'enrutament del router: manualment o dinàmicament (mitjançant protocols d'enrutament dinàmic).

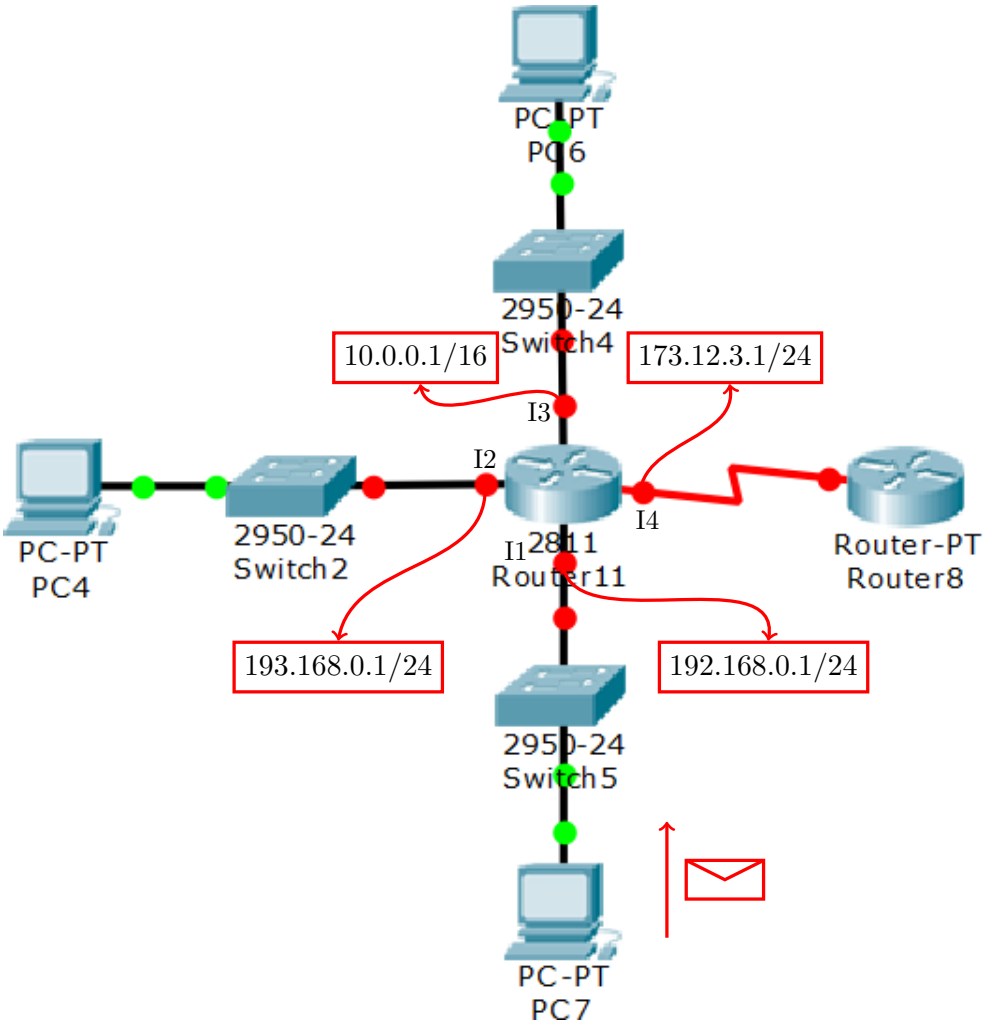
Cada entrada de les taules d'enrutament dels routers té 3 dades bàsiques:

1. Xarxa de destí

2. Pròxim salt per aquesta xarxa de destí (GW per aquesta xarxa)

3. Mètrica (a quants sals està aquesta xarxa)

Si hi ha múltiples rutes possibles, la mètrica ens ajuda a decidir quin camí agafar.



Cada cop que el Router11 ha de tractar un paquet, contrasta les xarxes amb la IP destí. Així doncs, imaginem un paquet que surt de la xarxa 192.168.0.0/24 i va al dispositiu 10.0.0.27 (recordar que a la capçalera de la Capa d'Internet només s'hi indica la IP i que la MX mai hi surt). El Router11 actuarà de la següent manera:

1. Contrastarà la IP de destí amb la màscara de la xarxa connectada a la interfície 2

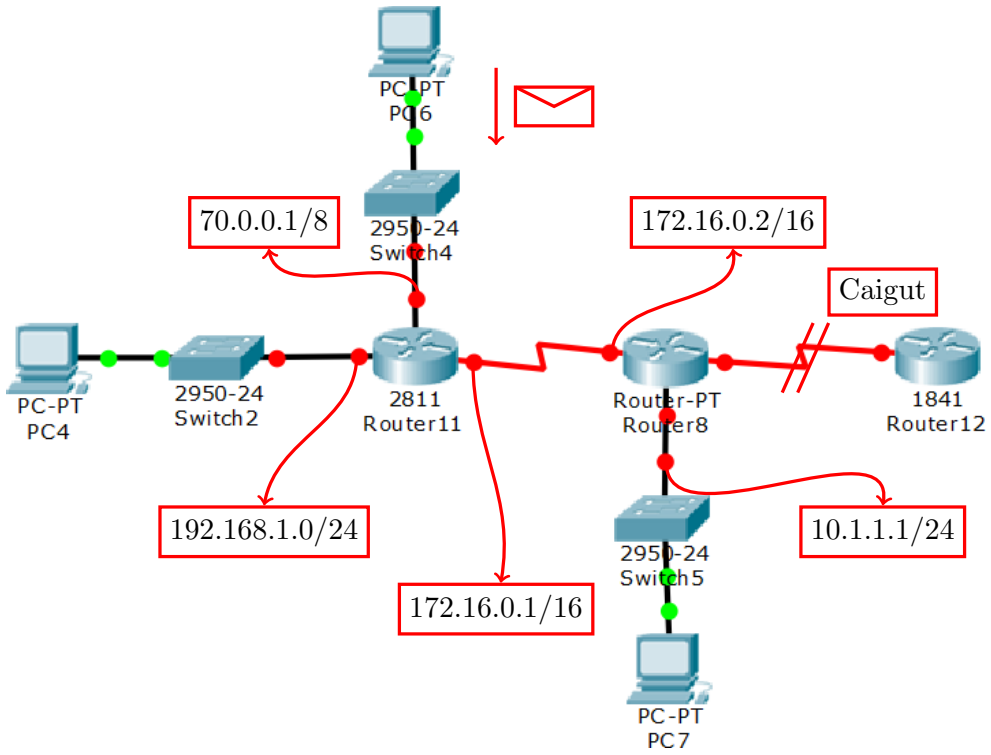
$$10.0.0.27 \text{AND} 255.255.255.0 == 193.168.0.1 \text{AND} 255.255.255.0?$$

2. Com que el resultat del punt anterior és negatiu, passarà a contrastar la IP de destí amb la màscara de la xarxa connectada a la interfície 3

$$10.0.0.27 \text{AND} 255.255.0.0 == 10.0.0.1 \text{AND} 255.255.0.0?$$

La segona comprovació donarà positiva i, per tant, el Router11 enviarà el paquet a través de la interfície 3. En canvi, si la IP de destí hagués estat la 10.10.0.27, tant la interfície 2 com la 3 haurien donant resultats negatius. Per tant, el Router11 hauria decidit enviar el paquet a la seva GW, és a dir al Router8 a través de la interfície 4. El Router8 actua com a GW o *Ruta per Defecte* del Router11.

Imaginem ara la situació següent



Quan el Router11 rebi el paquet només el podrà enviar a la seva GW o *Ruta per Defecte*, és a dir, al Router8. Quan el paquet arribi al Router8, aquest router veurà que no és per cap de les 2 xarxes que té connectades i que, a més a més, la seva GW o *Ruta per Defecte*, el Router12, té un problema i ha caigut de la xarxa. L'única possibilitat que tindria seria tornar a enviar el paquet al Router11, però això **MAI** es fa, perquè podria ocasionar un bucle entre el Router11 i el Router8 i col·lapsar la seva comunicació.

Així doncs, un paquet **MAI** torna per on ha vingut. Si un router no pot enrutar un paquet en concret, el descarta i envia un missatge d'error al router que li havia entregat aquest paquet. Per tant, en l'exemple anterior el Router8 descartarà el paquet i enviarà un missatge d'error al Router11.

Quan un paquet entra al router, la IP de destí es compara amb les entrades de la seva taula d'enrutament de tal manera que el pròxim salt serà aquell que sigui més específic. Per exemple, imaginem una situació en la qual el paquet té com a destí el dispositiu 10.1.1.55 i que la taula d'enrutament del router és la següent:

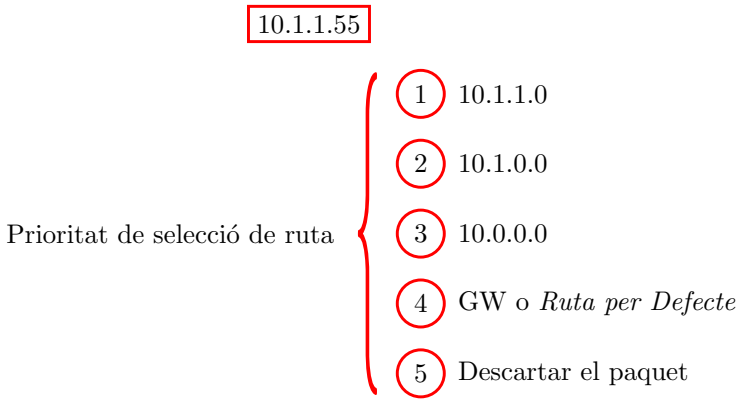
La xarxa 10.0.0.0/24 està dividida en dues subxarxes

10.1.1.0, salt per 192.168.1.2

10.1.2.0, salt per 192.168.1.2

El paquet, primer escolliria la ruta a 10.1.1.0. Si no existís, buscaria la ruta a 10.1.0.0. Si aquesta tampoc hi fos, saltaria per la 10.0.0.0. Finalment, si no en trobés cap, agafaria la GW o *Ruta per*

Defecte sempre i quan això no impliqués tornar enrere. Si la GW no pogués ser, descartaria el paquet i enviaria un missatge d'error (d'aquesta manera evita sobrecàrrega i ineficiència).



L'adreça de la *Ruta per Defecte* és la 0.0.0.0/0.

L'enrutament es fa paquet per paquet i salt a salt (cada paquet es tracta de manera independent en cada router).

Quan un router rep una trama:

1. Examinació: desencapsula la trama per llegir el paquet, és a dir, elimina la capçalera de la Capa d'Accés a Xarxa. En aquest pas comprova que la MAC de destí és la MAC de la interfície per on ha rebut la trama.
2. Selecció: llegeix la capçalera de la Capa d'Internet i selecciona la ruta a partir de la IP de destí i la seva taula d'enrutament.
3. Enviament: envia el paquet al següent salt. Per fer-ho necessita tornar a encapsular el paquet dins d'una trama, per tant, afegeix la capçalera de la Capa d'Accés a Xarxa. Depenent del següent salt, aquesta capçalera contindrà:
   (a) El dispositiu destí està dins d'una xarxa connectada directament al router:
   $$\text{MAC}_\text{O} = \text{MAC}_\text{router} \qquad \text{MAC}_\text{D} = \text{MAC}_\text{dispositiu}$$
   (b) El dispositiu no està connectat directament i, per tant, la trama ha de saltar a un altre router:
   $$\text{MAC}_\text{O} = \text{MAC}_\text{router} \qquad \text{MAC}_\text{D} = \text{MAC}_\text{router següent salt}$$

Cada router només coneix el següent salt, no coneix la ruta completa que haurà de fer el paquet fins arribar al seu destí. A més a més, no tots els paquets amb mateix origen i destí han de seguir la mateixa ruta. Això és així perquè els routers, mitjançant missatges de control, aprenen noves rutes i saben quines rutes han caigut i quines no. Per tant, poden prendre decisions diferents per cada paquet.

10.1 Creació de Taules d'Enrutament

Ja sabem que la taula d'enrutament d'un router és l'eina que permet al dispositiu prendre les decisions necessàries per poder enrutar un paquet. La informació bàsica que ha de tenir és:

1. Les xarxes que el router té connectades directament
2. La GW o *Ruta per Defecte*

Ara però, també pot contenir informació addicional que li permeti conèixer quin és el millor salt per arribar a una xarxa remota. Com més completa i actualitzada sigui aquesta informació, més eficient serà l'entrega de paquets i menys dades es perdran.

La configuració d'aquestes rutes addicionals es pot fer de manera manual o dinàmicament a partir d'un conjunt de protocols que permeten que els routers comparteixin dades entre ells.

10.1.1 Enrutament Estàtic

L'enrutament estàtic fa referència a la configuració manual de la taula d'enrutament d'un router amb les entrades referents a xarxes remotes.

Inconvenient
Cada Canvi que tingui lloc (noves xarxes, noves rutes, caiguda d'una ruta...) s'ha d'introduir manualment. Això fa que sigui més complicat mantenir la taula d'enrutament actualitzada i completa i, per tant, es poden produir retards i pèrdues de dades.

10.1.2 Enrutament Dinàmic

L'enrutament dinàmic utilitza els protocols d'enrutament per tal de configurar, de manera dinàmica, les taules d'enrutament dels routers. Aquests protocols permeten enviar missatges entre routers connectats directament. Cada cop que un router detecta una canvi en alguna de les xarxes que té connectades o en els routers que té connectats, canvia la seva taula d'enrutament i envia un missatge als seus router veïns amb la informació actualitzada. Aquests veïns canvien les seves taules i tornen a enviar la informació actualitzada als seus propis veïns, i així successivament. Amb això aconseguim:

1. Mantenir les taules actualitzades de manera dinàmica
2. Que els routers aprenguin quin és el millor camí per arribar a xarxes remotes

Alguns dels protocols d'enrutament dinàmic són:

- RIP: Protocol d'Informació d'Enrutament
- EIGRP: Protocol d'Enrutament de GW Interior Millorat
- OSPF: Open Shortest Path First

Inconvenients

1. Sobrecàrrega de la xarxa amb els missatges entre routers
2. Els routers necessitem una major capacitat de processament i de memòria per tal de poder enrutar els paquets i gestionar les taules d'enrutament de manera eficaç i sense perdre informació ni causar retards.

Molts cops es combina l'enrutament dinàmic amb l'estàtic per tal d'obtenir el màxim benefici de tots dos i reduir al mínim els inconvenients.