Subredes.-

Cuando se trabaja con una red pequeña, con pocos host conectados, el administrador de red puede fácilmente configurar el rango de direcciones IP usado para conseguir un funcionamiento óptimo del sistema. Pero conforme la red va creciendo se hace necesaria una división en partes de la misma.

En primer lugar, porque conforme se va extendiendo la red va aumentando de forma pareja el dominio de colisión, llegando un momento en el que el rendimiento de la red se ve afectado seriamente. Esto se puede mitigar segmentando la red, dividiendo la misma en una serie de segmentos significativos, de tal forma que mediante switches podremos limitar estos dominios de colisión, enviando las tramas tan sólo al segmento en el que se encuentra el host destino.

En segundo lugar, y aunque segmentemos la red, conforme aumenta el número de host aumenta también el número de transmisiones de broadcast (cuando un equipo origen envía datos a todos los dispositivos de la red), llegando un momento que dicho tráfico puede congestionar toda la red de forma inaceptable, al consumir un ancho de banda excesivo. Esto es así porque todos los host están enviando de forma constante peticiones de este tipo: peticiones ARP, envíos RIP, peticiones DNS, etc.

Para solventar este hecho es preciso dividir la red primaria en una serie de subredes, de tal forma que cada una de ellas va a funcionar luego, a nivel de envío y recepción de paquetes, como una red individual, aunque todas pertenezcan a la misma red principal (y por lo tanto, al mismo dominio). De esta forma, aunque la red en su conjunto tendrá una dirección IP única, administrativamente, a nivel administrativo podremos considerar subredes bien diferenciadas, consiguiendo con ello un control del tráfico de la red y una limitación de las peticiones de broadcast que la atraviesan.

En las explicaciones siguientes vamos a considerar una red pública, es decir, formada por host con direcciones IP públicas, que pueden ser vistas por todos las máquinas conectadas a Internet. Pero el desarrollo es igualmente válido para redes privadas, por lo que su aplicación práctica es válida para toda red corporativa. Y para hacer más claro el desarrollo, vamos a parir de una red con dirección IP real.

Vamos a tomar como ejemplo una red de clase C, teniendo claro que lo que expliquemos va a ser útil para cualquier tipo de red, sea de clase A, B o C. Entonces, tenemos nuestra red, con dirección IP **210.25.2.0**, por lo que tenemos para asignar a los host de la misma todas las direcciones IP del rango 210.25.2.1 al 210.25.2.254, ya que la dirección 210.25.2.0 será la de la propia red y la 210.25.2.255 será la dirección de broadcast general.

Si expresamos nuestra dirección de red en binario tendremos:

210.25.2.0 => 11010010.00011001.00000010.00000000

Con lo que tenemos 24 bits para identificar la red (en rojo) y 8 bits para identificar los host (en azul).

La máscara de red será:

11111111.11111111.11111111.00000000 = 255.255.255.0

Para crear subredes a partir de una dirección IP de red padre, la idea es "robar" bits a los host, pasándolos a los de identificación de red. ¿Cuántos? Bueno, depende de las subredes que queramos obtener, teniendo en cuenta que cuántas más bits robemos, más subredes obtendremos, pero con menos host cada una. Por lo tanto, el número de bits a robar depende de las necesidades de funcionamiento de la red final.

Máscara de subred.-

Otro elemento que deberemos calcular para cada una de las subredes es su máscara de subred, concepto análogo al de máscara de red en redes generales, y que va a ser la herramienta que utilicen luego los routers para dirigir correctamente los paquetes que circulen entre las diferentes subredes.

Para obtener la máscara de subred basta con presentar la dirección propia de la subred en binario, poner a 1 todos los bits que dejemos para la parte de red (incluyendo los robados a la porción de host), y poner a 0 todos los bits que queden para los host. Por último, pasaremos la dirección binaria resultante a formato decimal separado por puntos, y ésa será la máscara de la subred.

Por ejemplo, si tenemos la dirección de clase B:

150.10.x.x = 10010110.00001010.hhhhhhhh.hhhhhhhh

y le quitamos 4 bits a la porción de host para crear subredes:

10010110.00001010.rrrrhhhh.hhhhhhhh

la máscara de subred será:

11111111.11111111.11110000.000000

que pasada a decimal nos queda:

255.255.240.0

Las máscaras de subred, al igual que ocurre con las máscaras de red, son muy importantes, resultando imprescindibles para el trabajo de enrutamiento de los routers.

Creando las subredes.-

Vamos pues a partir de nuestra dirección IP de la red padre y vamos a ir quitando bits sucesivos a la porción de host, calculando en cada caso las subredes obtenidas, sus direcciones IP, sus máscaras de subred y el rendimiento de la partición obtenida.

Para ello, pasamos la dirección IP a binario, tomamos los bits robados de la porción de host y vamos variando de las 16 formas posibles:

0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111,.....1100, 1101, 1110, 1111.

en el caso de 4 bits robados a la porción del host en la red clase B anterior; luego calculamos las IP's de los host correspondientes a cada una de las variaciones hallando los márgenes de las mismas, ya que estarán entre el valor mínimo y el máximo al variar los bits de la porción de host entre todos 0 (dirección de subred) y todos 1 (dirección de broadcast correspondiente).

Por sencillez en el análisis, se va a ejemplificar sobre la red clase C mencionada anteriormente, es decir con la IP = 210.25.2.0, pudiendo luego generalizarse las conclusiones a cualquier clase de red.

Robo de 1 bit:

Si quitamos un sólo bit a la parte de host:

Parte de red: 11010010.00011001.00000010.r

Parte de host: hhhhhhh

Permutando los bits de host robados para obtener las subredes obtenidas: $2^1=2$

Es decir, 2 subredes (las 11010010.00011001.00000010.0 y 11010010.00011001.00000010.1). Pero resulta que no podemos disponer de la subred que toma el 0, ya que entonces contendría la IP de la red padre, ni de la que toma el 1, ya que contendría la dirección de broadcast de la red padre. **Es decir, robando 1 sólo bit no podemos crear subredes**.

Como regla general, el número de subredes obtenidas al quitar n bits a la porción de host serán 2^n-2, y el número de host disponibles en cada subred será $2^{(8-n)}$-2, (en clase C), ya que toda subred debe tener su propia dirección de red y su propia dirección de broadcast.

Si vamos aumentando el número de bits robados a la porción de host obtenemos:

Robo de 2 bits:

Parte de red: 11010010.00011001.00000010.rr

Parte de host: hhhhhh

Número de subredes válidas: 2^2-2=2

Número de host válidos por cada subred: 2^6-2=62

Las direcciones de subred las obtenemos haciendo las combinaciones posibles con los 2 bits robados:

11010010.00011001.00000010. **00** 000000 a 11010010.00011001.00000010. **00** 111111 =210.25.2.0 a 210.25.2.63 (**no valida**, al contener la dirección de red de la red padre).

11010010.00011001.00000010.**01**000000 a 11010010.00011001.00000010.**01**111111 = 210.25.2.64 a 210.25.2.127
Subred válida, con dirección de red = 210.25.2.64, broadcast =210.25.2.127 y 62 direcciones IP para host, que son las comprendidas entre las dos anteriores (de la 210.25.2.65 a la 210.25.2.126).

11010010.00011001.00000010.**10** 000000 a 11010010.00011001.00000010.**10** 111111 = 210.25.2.128 a 210.25.2.191

Subred válida, con dirección de red=210.25.2.128, broadcast=210.25.2.191 y 62 direcciones IP para host, que son las comprendidas entre las dos anteriores (de la 210.25.2.129 a la 210.25.2.190).

11010010.00011001.00000010.**11** 000000 a 11010010.00011001.00000010. **11**111111 = 210.25.2.192 a 210.25.2.225 (**no valida**, al contener la dirección de broadcast de la red padre).

Máscara de subred de todas ellas:

11111111.11111111.11111111.11000000 = 255.255.255.192

Como se ve, la máscara de subred es la misma para todas las subredes obtenidas robando 2 bits a la porción de host.

Resumiendo: para evitar ambigüedades entre las direcciones de subredes y las IP de red padre e IP broadcast, la norma evita las subredes 00 y 11, con lo que obtenemos dos subredes válidas, con 62 direcciones IP válidas para host cada una, es decir, en definitiva desperdiciamos:

IP inútiles para host = (256-2)-(62+62)=130 direcciones IP que no se utilizan.

por lo que el ***rendimiento de la partición en subredes*** será:

R = (IP útiles host)/(IP útiles totales)=124/254=0.488= 48%

Algo similar ocurre para el robo de otro número de bits.

Robo de 3 bits:

Parte de red: 11010010.00011001.00000010.rrr

Parte de host: hhhhh

Número de subredes válidas: 2^3-2=6

Número de host válidos por cada subred: 2^5-2=30

Las direcciones de subred las obtenemos haciendo las combinaciones posibles con los 3 bits robados:

11010010.00011001.00000010. **000**00000 a 11010010.00011001.00000010.**000** 11111 (**no valida**, al contener la dirección de red de la red padre).

11010010.00011001.00000010.**001** 00000 a 11010010.00011001.00000010. **001**11111 = 210.25.2.32 a 210.25.2.63. **Subred válida**, con dirección de red = 210.25.2.32, broadcast = 210.25.2.63 y 30 direcciones IP para host, que son las comprendidas entre las dos anteriores (de la 210.25.2.33 a la 210.25.2.62).

11010010.00011001.00000010.**010** 00000 a 11010010.00011001.00000010.**010** 11111 = 210.25.2.64 a 210.25.2.95. **Subred válida**, con dirección de red = 210.25.2.64, broadcast =

210.25.2.95 y 30 direcciones IP para host, que son las comprendidas entre las dos anteriores (de la 210.25.2.65 a la 210.25.2.94).

11010010.00011001.00000010.**011** 00000 a 11010010.00011001.00000010.**011** 11111 = 210.25.2.96 a 210.25.2.127. **Subred válida**, con dirección de red = 210.25.2.96, broadcast = 210.25.2.127 y 30 direcciones IP para host, que son las comprendidas entre las dos anteriores (de la 210.25.2.97 a la 210.25.2.126).

11010010.00011001.00000010.**100** 00000 a 11010010.00011001.00000010.**100** 11111 = 210.25.2.128 a 210.25.2.159. **Subred válida**, con dirección de red = 210.25.2.128, broadcast = 210.25.2.159 y 30 direcciones IP para host, que son las comprendidas entre las dos anteriores (de la 210.25.2.129 a la 210.25.2.158).

11010010.00011001.00000010.**101** 00000 a 11010010.00011001.00000010.**101** 11111 = 210.25.2.160 a 210.25.2.191. **Subred válida**, con dirección de red = 210.25.2.160, broadcast = 210.25.2.191 y 30 direcciones IP para host, que son las comprendidas entre las dos anteriores (de la 210.25.2.161 a la 210.25.2.190).

11010010.00011001.00000010.**110** 00000 a 11010010.00011001.00000010.**110** 11111 = 210.25.2.192 a 210.25.2.223. **Subred válida**, con dirección de red = 210.25.2.192, broadcast = 210.25.2.223 y 30 direcciones IP para host, que son las comprendidas entre las dos anteriores (de la 210.25.2.193 a la 210.25.2.222).

11010010.00011001.00000010. **111**00000 a 11010010.00011001.00000010. **111**11111 = 210.25.2.224 a 210.25.2.255 (**no valida**, al contener la dirección de broadcast de la red padre).

Máscara de subred para todas ellas:

11111111.11111111.11111111.11100000 = 255.255.255.224

Como se ve, la máscara de subred es la misma para todas las subredes obtenidas robando 3 bits a la porción de host.

Resumiendo: obtenemos 6 subredes válidas, con 30 direcciones IP válidas para host cada una, es decir, desperdiciamos:

IP inútiles para host = (256-2)-(30+30+30+30+30+30)= 74 direcciones IP que no se utilizan.

con lo que el ***rendimiento de la partición en subredes*** será:

R = (IP útiles host)/(IP útiles totales)=180/254=0.708= 70.8%

Lo mismo que en los casos anteriores, hacemos en el caso de robar 4, 5 y 6 bits.
Notar que **7 bits no podemos robar**, ya que entonces las subredes resultantes sólo podrían tener 2 direcciones IP, una para la subred y otra de broadcast, con lo que no podrían tener host.

Cada vez que se pide prestado otro bit del campo de host, la cantidad de subredes totales posibles se duplica, mientras que la cantidad de direcciones de host totales que se pueden asignar se reduce a la mitad (aunque la cantidad de redes y host útiles varía un poco de esta regla: 2 menos en todo caso).

Un patrón de equivalencia decimal-binario a la hora de calcular máscaras de subred es el siguiente:

patrones decimal-binario

128	64	32	16	8	4	2	1	
1	0	0	0	0	0	0	0	= 128
1	1	0	0	0	0	0	0	= 192
1	1	1	0	0	0	0	0	= 224
1	1	1	1	0	0	0	0	= 240
1	1	1	1	1	0	0	0	= 248
1	1	1	1	1	1	0	0	= 252
1	1	1	1	1	1	1	0	= 254
1	1	1	1	1	1	1	1	= 255

En cualquier caso, y una vez realizada la partición, la primera dirección IP válida de la misma se suele asignar al router que unirá las diferentes subredes.

Optimizando la partición.-

Es tarea del diseñador de la red o del administrador de la misma el obtener la partición en subredes más acertada de acuerdo con las necesidades actuales y futuras, con objeto de optimizar el número de IP's utilizadas, sobre todo en el caso de que la red sea pública.

Por un lado, se pueden precisar subredes con unas necesidades de host predeterminadas (p.e. 50 host por subred, 120, etc.), por otro se debe procurar que el número de IP's desperdiciadas sea mínimo, y por otro lado se deben limitar al máximo el ancho de banda absorbido por las peticiones de broadcast.

Por lo tanto, se hace preciso un cálculo exacto de las diferentes opciones disponibles, buscando que el rendimiento de la partición sea máximo, dentro de las necesidades exigidas a la partición. Un resumen de los rendimientos (direcciones) se tiene en la siguiente tabla:

rendimientos en clase C

Cantidad de bits prestados	Cantidad de subredes creadas	Cantidad de hosts por subred	Cantidad total de hosts	Porcentaje utilizado
2	2	62	124	49%
3	6	30	180	71%
4	14	14	196	77%
5	30	6	180	71%
6	62	2	124	49%

De todas formas, el caso más normal con el que nos encontraremos será una empresa u organización con una o varias direcciones IP públicas, asignadas por su ISP (Proveedor de Servicios de Internet), que serán usadas por los routers/firewalls, encargados de dar salida a Internet a todos los host internos. Tras los routers habrá normalmente uno o más servidores Proxi, que serán los que se encargarán de gestionar las peticiones de servicios externos de los host, y tras él tendremos una red interna, privada, formada por diferentes host, servidores de

aplicaciones, servidores de datos, impresoras, etc. En estos casos, el administrador o diseñador de la red interna dispondrá de todo un rango de IP's disponibles para realizar las particiones, pudiendo usar la clase IP privada (clase A, B o C) que más le convenga. No obstante, es muy importante también el cálculo óptimo de la partición, a fin de limitar al máximo los dominios de colisión y el ancho de banda consumido por los broadcast.

Existen para ello **direcciones IP reservadas**, para las llamadas **redes privadas**, para usos internos, que se establecieron por convenio. (RFC 1597-*Distribución de direcciones para redes privadas*) Estas direcciones no son vistas desde el exterior, no son públicas, y sus rangos son:

- Clase A: 10.0.0.0 (una sola red)
- Clase B: 172.16.0.0 a 172.31.0.0 (16 redes contiguas)
- Clase C: 192.168.X.0 (con X variando). (256 redes contiguas)

En pocas palabras, relaja la regla de que las direcciones IP han de ser unívocas globalmente al reservar parte del espacio de direcciones para redes que se usan exclusivamente dentro de una sola organización y que no requieren conectividad IP con Internet

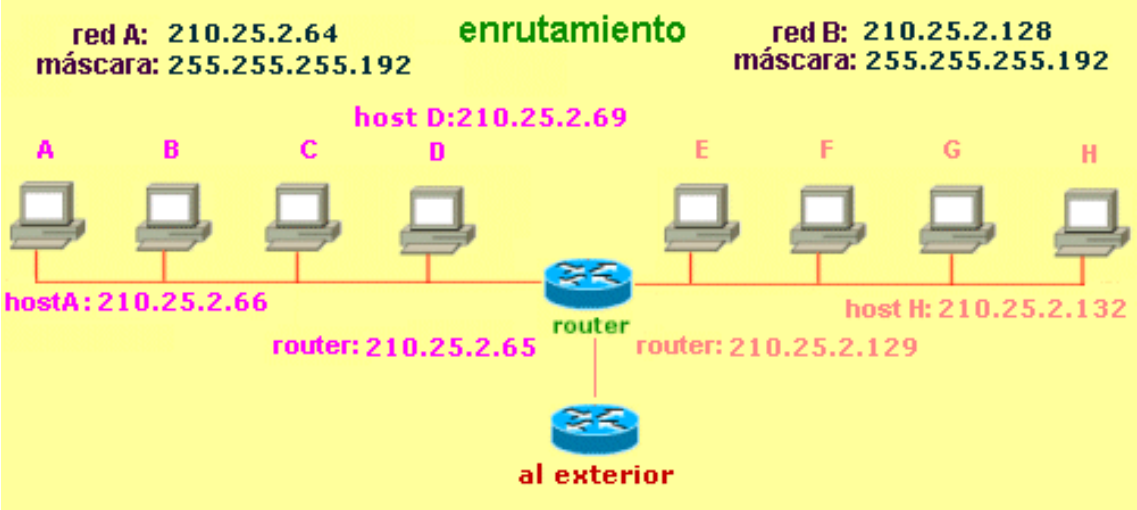
Cualquier organización puede usar cualquier dirección en estos rangos si no hace referencia a ninguna otra organización. Sin embargo, debido a que estas direcciones no son unívocas a nivel global, no pueden ser direccionadas por hosts de otras organizaciones y no están definidas para los "routers" externos. Se supone que los "routers" de una red que no usa direcciones privadas, particularmente aquellos operados por proveedores de servicios de Internet, han de desechar toda información de encaminamiento relativa a estas direcciones. Los "router" de una organización que utiliza direcciones privadas deberían limitar todas las referencias a direcciones privadas a los enlaces internos; no deberían hacer públicas las rutas a direcciones privadas ni enviar datagramas IP con estar direcciones a los "routers" externos. Estas redes (Intranet) tienen la ventaja de ser mucho menos accesibles a ataques desde el exterior.Los hosts que sólo tienen una dirección IP privada carecen de conexión IP con Internet. Esto puede ser deseable y a lo mejor puede ser una razón para emplear direcccionamiento privado. Toda la conectividad con host externos de Internet la deben proporcionar pasarelas de aplicación.

Enrutamiento en subredes.-

El enrutamiento en subredes es análogo al que se produce en el caso de redes, si bien ahora podemos considerar un caso más, el enrutamiento indirecto en la propia red. Partamos para la explicación que sigue de una red dividida en dos subredes (210.25.2.64 y 210.25.2.128), enlazadas mediante un sólo router (210.25.2.65 en A y 210.25.2.129 en B), que además es el gateway por defecto, es decir, el encargado de sacar fuera de la red padre las tramas externas. (Figura de la página siguiente)

Cuando el host A se quiere comunicar con otro, lo primero que hace es consultar su tabla ARP, para ver si tiene en la misma la entrada que le de la equivalencia IP-MAC del host destino. Si es así, construye sus tramas completas y las envía al medio, esperando que el destinatario las reciba directamente. Si no encuentra la entrada correspondiente en la tabla, lanza una petición ARP querry, de tipo broadcast, esperando que el host destino le devuelva su dirección física. Si el host destino es el D, que se encuentra en la misma subred, responderá a la petición ARP con su MAC o recogerá directamente las tramas a él destinadas. Este direccionamiento se conoce con el nombre de **enrutamiento directo**. En este proceso, el router recoge las tramas y hace una operación AND con la dirección IP destino que en ellas figura y con la máscara de subred del host que ha enviado los datos:

210.25.2.69 AND 255.255.255.192 = 210.25.2.64



Con lo que "sabe" que el host destino se encuentra en la misma subred que el origen de datos, dejando pasar las tramas sin intervenir.

Ahora bien, si el host destino fuera el H, que no se encuentra en la misma subred, el router, al hacer la operación AND lógica obtendrá:

210.25.2.132 AND 255.255.255.192 = 210.25.2.128

Con lo que "sabe" que el host destino no se encuentra en la misma subred que el host A. Entonces, recoge él mismo las tramas enviadas por A y las pasa a la subred de H, con lo que se puede realizar la entrega. En este caso nos encontramos con un **enrutamiento indirecto interno**.

Un tercero y último caso se producirá cuando el host A quiera enviar datos a un host externo a las subredes que une el router. Éste, al hacer la operación AND lógica, descubre que las tramas no van a ningún host de las subredes que une, por lo que cambia la dirección MAC de las mismas por la suya propia, de la subred a la que pertenece al host origen, y dejando la dirección IP del host destino, sacando los datos entonces al exterior de las subredes, enviándolas al router externo que crea que puede proseguir mejor el enrutamiento. En este caso hablamos de **enrutamiento indirecto externo**. Los routers poseen sus correspondientes tablas de enrutamiento dinámicas, que son las que van a fijar el router externo al que se envían las tramas.

Las tramas así enviadas van viajando por diferentes routers, hasta llegar a la red/subred destino. Cuando el host que recibe las tramas responde al origen, los datos viajan en sentido opuesto (aunque no tienen porqué hacerlo por el mismo camino), y al llegar de nuevo al router de nuestras subredes, las tramas tendrán como dirección física (dirección MAC) la del router, y como dirección lógica (dirección IP),la del host A. Entonces el router vuelve a hacer la operación lógica AND entre la dirección IP de las tramas y las de las diferentes subredes que une, obteniendo la subred a la que pertenece el host A, con lo que le envía los datos a éste, finalizando el proceso.

Es decir, en el enrutamiento indirecto externo el router funciona como un intermediario, lo mismo que los diferentes routers que van enrutando las tramas hasta el destino, usando para ello sus propias direcciones MAC, que van cambiando, permaneciendo siempre fijas las IP de los host destino.