

## 1. Que es una dirección IP?

Una dirección IP esta conformada por 4 octetos, o 32 bits. Es usualmente representada en formato decimal como este: 131.107.2.205. Cada número representa un Octeto. Un octeto es un grupo de 8 bits. Como tenemos 4 octetos en una dirección IP, entonces tenemos  $8 \times 4 = 32$  bits en una dirección IP.

Las computadoras no entienden la notación decimal, ya que ellas solo funcionan en binario. Todo lo que las computadoras entienden es 1 y 0. Por lo tanto, debemos buscar una manera de transferir una dirección IP del formato decimal al binario. Vamos a hacerlo octeto por octeto.

Cada BIT en un octeto tiene un valor decimal asociado:

BIT	1	2	3	4	5	6	7	8
Valor BIT	128	64	32	16	8	4	2	1

Vamos a tomar un ejemplo. La dirección IP 131.107.2.4. ¿Que es en binario?

Vamos a hacerlo octeto por octeto:

131:

Valor BIT	128	64	32	16	8	4	2	1
Binario	1	0	0	0	0	0	1	1

Las columnas con un "1" en binario significan que tenemos que contar el correspondiente valor del BIT. Si sumamos todos los valores de BIT marcados con uno, entonces tenemos  $128 + 2 + 1 = 131$ .

107:

Valor BIT	128	64	32	16	8	4	2	1
Binario	0	1	1	0	1	0	1	1

Las columnas con un "1" en binario significan que tenemos que contar el correspondiente valor del BIT. Si sumamos todos los valores de BIT marcados con uno, entonces tenemos  $64 + 32 + 8 + 2 + 1 = 107$

2:

Valor BIT	128	64	32	16	8	4	2	1
Binario	0	0	0	0	0	0	1	0

Las columnas con un "1" en binario significan que tenemos que contar el correspondiente valor del BIT. Si sumamos todos los valores de BIT marcados con uno, entonces el resultado es 2

4:

Valor BIT	128	64	32	16	8	4	2	1
Binario	0	0	0	0	0	1	0	0

Las columnas con un "1" en binario significan que tenemos que contar el correspondiente valor del BIT. Si sumamos todos los valores de BIT marcados con uno, entonces el resultado es 4

Ahora sabemos que existe otra manera de escribir nuestra dirección IP de ejemplo 131.107.2.4 es:

10000011.01101011.00000010.00000100

Una dirección IP esta conformada por dos partes: el **Network ID** y el **Host ID**. Cuando tratas de darle ping a una dirección IP, IP en capa 4 necesita determinar si la IP destino es local o remota en tu subnet. Para explicar esto, me gusta hacerles a mis estudiantes la siguiente pregunta:

"Digamos que yo vivo en la calle Hidalgo. Tú dices que también vives en la calle Hidalgo. ¿Somos vecinos? Bueno, quizás si o quizás no. No tenemos suficiente información para responder a esta pregunta. Específicamente, no sabemos si vivimos en la misma Ciudad. Si nosotros viviésemos en la misma Ciudad y nuestras calles tuviesen nombres similares, entonces si seríamos vecinos. Si no vivimos en la misma ciudad, no importa si nuestras calles tengan el mismo nombre: **no somos vecinos**"

Lo mismo aplica para el direccionamiento IP. Antes de que yo pueda encontrar cual es tu Host ID – ejemplo: el nombre de tu calle – Tengo primero que averiguar cual es tu Network ID – ejemplo: Tu Ciudad.

Por tanto, ¿Como el direccionamiento IP determina cual es el Host ID y el Network ID? Ese es el rol de la mascara de subred (subnet mask).

Nota: Ten en cuenta que ni el Network ID, ni el Host ID pueden ser todos Ceros o todos Unos. Veremos esto mas adelante....

## 2. Que es la Mascara de Subred (subnet mask)?

La Mascara de Subred permite a IP en la capa 3 el determinar si la dirección IP destino que estas tratando de contactar es **remota** o **local**. Esa es su principal función. Ella ayuda a determinar que parte de la dirección IP es el Network ID y cual es el Host ID. Pero, ¿Como hace esto?

Todos hemos visto una Mascara de subred antes. Usualmente es algo parecido a esto:

255.255.255.0

Esta Mascara de Subred es obviamente mostrada en formato de dotación decimal. Como ya conocemos, las computadoras no entienden este formato. Por lo tanto vamos a aplicar lo que ya conocemos sobre Binario a nuestra Mascara de Subred:

255

Valor BIT	128	64	32	16	8	4	2	1
Binary	1	1	1	1	1	1	1	1

Las columnas con un "1" en binario significan que tenemos que contar el correspondiente valor del BIT. Si sumamos todos los valores de BIT marcados con uno, entonces tenemos:  $128+64+32+16+8+4+2+1=255$

Por lo tanto, 255 es "Todos Uno's", como puedes haber oído antes.

0:

Valor BIT	128	64	32	16	8	4	2	1
Binary	0	0	0	0	0	0	0	0

Las columnas con un "1" en binario significan que tenemos que contar el correspondiente valor del BIT. Si sumas todos los valores de Bits con un UNO, tu obtendrás solo CERO. Por lo tanto el resultado seria, "todos Ceros".

En otras palabras, nuestra mascara de subred 255.255.255.0 en binario seria:

11111111.11111111.11111111.00000000

Observa que la mascara de subred es también de 32 BIT, un paquete de 4 octetos que concuerdan con la estructura de nuestra dirección IP.

Si nosotros sobreponemos la dirección IP con la mascara de subred que hemos traducido hasta ahora, obtendremos esto:

131.107.2.4	10000011.	01101011.	00000010.	00000100
255.255.255.0	11111111.	11111111.	11111111.	00000000

He marcado los bits de Network en rojo, y los bits del Host en Azul. ¿Notan algo? All the network bits are the bits that had a value of "1" in the subnet mask, and all the host bits are the bits that had a value of "0" in the subnet mask. Easy, isn't it? In our example, the Network ID is 131.107.2, and the host ID is 4. If I change the subnet mask to 255.255.0.0, what happens? Well, let's apply the same logic:

131.107.2.4	10000011.	01101011.	00000100	00000100
255.255.0.0	11111111.	11111111.	00000000	00000000

¿Que pasa ahora? Mi network ID es ahora 131.107 y mi Host ID es 2.4! Esta es la explicación de porque una dirección IP por si sola ¡no puede existir! Ese es el porque un Host en nuestra red necesita **al menos** una dirección IP ¡Y una mascara de subred!

Vamos a suponer que tienes dos direcciones IP:

131.107.2.4 y 131.107.5.6

Estas ips son locales una con la otra o son remotas?

*Tu no podrías responder esa pregunta, porque esta incompleta! Necesito darte una mascara de subred también! Veamos porque...*

Digamos que la mascara de subred es 255.255.255.0. Entonces tenemos:

131.107.2.4	10000011.	01101011.	00000010.	00000100
131.107.5.6	10000011.	01101011.	00000101.	00000110
255.255.255.0	11111111.	11111111.	11111111.	00000000

Son iguales los network Ids? No! Miremos el 3er octeto:

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

**Si el network IDs no concuerda, entonces las direcciones son remotas una de la otra.** Ellas estarán en diferentes subredes por lo que necesitaras un router para que se comuniquen entre ellas.

Vamos a tomar el mismo ejemplo pero con una mascara de subred diferente.  
Ahora será 255.255.0.0:

131.107.2.4	10000011.	01101011.	00000010.	00000100
131.107.5.6	10000011.	01101011.	00000101.	00000110
255.255.0.0	11111111.	11111111.	00000000.	00000000

Ahora el Network IDs coincide? Si!

**Si el Network IDs es el mismo, entonces las dos direcciones estan en la misma red.** No necesitaras un router para comunicar una ip con la otra en este escenario, ya que ambas Ips son locales.

Vamos a resumir lo que hemos aprendido: Todos hemos visto que tener solo la dirección IP no es suficiente, y como dos direcciones pueden ser locales o remotas dependiendo de la mascara de subred que estemos empleando. Esta es la base de la búsqueda de Fallas y detección de problemas en el subneteo IP.

### 3. Que son las clases (IP classes) ?

Es posible que hayas escuchado las diferentes clases de direcciones IP, nombradas clase A, B y C. Que significa esto? Echémosles una mirada a la siguiente tabla:

Clase A	1-127*	0xxxxxxx.
Clase B	128-191	10xxxxxx.
Clase C	192-223	110xxxxx.

\*127 es parte del rango de la clase A, pero no puedes asignarle la dirección 127.x.x.x a ningún nodo ya que el rango entero esta reservado por InterNIC para direcciones loop back..

Que significa esto? Cuando hablemos de clases en direcciones IP, solo debemos mirar el primer octeto para determinar a que clase IP pertenece la dirección.

Para la clase A, InterNIC decidió que el 1er octeto debía comenzar con un valor de BIT de 0. Por lo tanto, el binario menor del 1er octeto es 0000001, y el más alto es 01111111 (en decimal, esto sería de 1 a 127)

Para la clase B, InterNIC decidió que el 1er octeto debía comenzar con un valor de BIT de 10.

Por cuanto, el binario mas bajo del 1er octeto es 1000000, y el más alto es 10111111 (en decimal, sería de 128 a 191)

Para la clase C, InterNIC decidió que el 1er octeto debía comenzar con un valor de BIT de 110.

Por cuanto, el binario mas bajo del 1er octeto es 1100000, y el más alto es 11011111 (en decimal, sería de 192 hasta 223)

También hay otras clases, D y E, pero no son usadas ahora además de ser ilegales en la Internet. La clase D es usada en Multicast y la E es experimental.

Las mascarar de subred también están definidas por clase. Puedes combinar cualquier clase de mascara con cualquier clase de dirección IP. Las clases de mascarar están definidas en la siguiente tabla:

Clase A	255.0.0.0
Clase B	255.255.0.0
Clase C	255.255.255.0

En otras palabras, es totalmente posible tener una dirección IP clase B y una mascara de subred clase C.... esperen.... ¿No es un buen ejemplo el que habíamos citado al principio?. Recordemos nuestra dirección IP 131.107.2.4 con la mascara 255.255.255.0?

#### 4. Que es subnetear (subnetting)?

Subnetear es la acción de tomar un rango de direcciones IP donde todas las IPS sean locales unas con otras y dividir las en diferentes rangos, o subnets, donde las direcciones IPS de un rango serán remotas de las otras direcciones.

Si tú quieres determinar cuantos hosts tú tienes en un rango IP, primero debes determinar cuantos hosts bits tenemos. Vamos a tomar el ejemplo anterior de 131.107.2.4 y 255.255.255.0. Ya establecimos anteriormente que el network ID es 131.107.2 y el host ID es 4. En otras palabras, tenemos 3 octetos para el Network ID y uno – un octeto – para el Host ID. Ahora que hemos determinado la cantidad de hosts bits que tenemos, aplicar este número a la siguiente formula:

$(2^N) - 2$  = numero de hosts, donde N es el numero de Host bits

Esto nos da:  $((2^8) - 2) = 254$  hosts.

Esto quiere decir que en nuestro ejemplo, tenemos la red 131.107.2.x, que contiene 254 direcciones IP posibles, todas locales unas con otras.

Que pasaría si escogemos una mascara clase A? Entonces seria:

$(2^{24}) - 2 = 16,777,214$  direcciones IP validas en ese rango!

Que pasaría si tu no necesitas tantos Hosts y decides dividir ese rango en otros mas pequeños y mas administrables?. Bueno, en este caso necesitamos subnetear.

**Nota: Tenemos que substraer 2 porque perdemos todos los valores "todos Ceros" y "todos Unos"**

Si solo tenemos 2 bits, en binario, tendríamos  $2^2=4$  posibilidades:

00

01

10

11

No obstante, **todos Ceros** en IP significa que es la Network en si, y **todos unos** son el Broadcast ID, ninguna de estas opciones podrá ser asignada a un Host. Este es el porque siempre perdemos 2 y tenemos que sustraer 2 de  $(2^N)$ ...

Echémosle una mirada a la siguiente tabla:

Valor BIT	128	64	32	16	8	4	2	1
Mascara Subred		192	224	240	248	252	254	255
Numero de Subnets		2	6	14	30	62	126	254

Esta tabla es la única tabla que tienes que aprenderte para entender el subneteo y direccionamiento IP! Tres pequeñas líneas!

Miremos línea por línea. La primera línea ya sabemos de donde sale, así que no perdamos tiempo en ella.

La segunda línea nos dice cual es la mascara de subred. Como obtenemos estos resultados? Si miramos en la línea del Valor BIT y agregamos los valores de bits, veremos lo siguiente:

$$128+64=192$$

$$192+32=224$$

$$224+16=240$$

$$240+8=248$$

$$248+4=252$$

$$252+2=254$$

$$254+1=255$$

Fácil No? ☺

Ahora la tercera línea. Esta nos dirá cuantas mascaras de subred tendremos si usamos la correspondiente mascara. En otras palabras, si usamos 192 en nuestra mascara de subred, tendremos dos subredes. Si usamos 224, tendremos 6 subredes, etc.... Pero como obtenemos esos números?. Bueno, vamos a tomar 192 como un ejemplo. Cuantos bits usamos para obtener 192? Bueno, agregamos 128 y 64, así que serian 2 bits, no? Veamos a nuestra bien conocida formula:

$$((2^2)-2)= 2$$

Es de aquí de donde salen los valores de la 3ra línea. Quieres checar otra? Veamos ahora 248. Cuantos bits usamos para obtener el ?  
 $128+64+32+16+8=248$ , o un total de 5 bits.  $((2^5)-2)=30$

así que ahora todos sabemos como construir esta tabla y también entendemos como es construida. Ya tenemos casi todo lo que necesitamos para subnetear!

Digamos que mi jefe me encuentra un día en la mañana y dice:

"Alexis, tengo un rango IP 131.107.0.0 y una mascara de subred 255.255.0.0. Yo quiero obtener 6 subnets." Que tengo que hacer ahora?

Yo miro mi tabla, y veo que para obtener 6 subnets, necesitaría una mascara de 224. Ya yo tengo una subnet de 255.255.0.0, que en binario luciría así:

```
11111111.11111111.00000000.00000000
```

Yo no podemos usar mas bits de los primeros 2 octetos, porque son el Network ID – representados por el valor binario de 1-. Por lo tanto solo podemos usar este "224" a mi mascara de subred en el área de los host ID. Tomaríamos el próximo octeto disponible – el tercero en nuestro ejemplo– y terminaríamos con una mascara de subred 255.255.224.0, que en binario seria:

```
11111111.11111111.11100000.00000000
```

Nota que nosotros usamos 3 host bits – valor binario de 0 – en el tercer octeto y los convertimos en network bits – valor binario de 1. También observa que esos 3 bits es lo que necesito para lograr el 224:  
 $128+64+32=224$ .

Ahora que tenemos la mascara de subred, ya tendríamos solo 13 host bits, lo que significa que tenemos  $((2^{13})-2)=8190$  direcciones IP validas por rango  
☺

Así que vuelvo a donde esta mi Jefe y le digo "Jefe, nuestra nueva mascara de subred para la red 131.107.x.x será 255.255.224.0, y tendremos 6 subredes con 8190 direcciones IP en cada rango" Cual será la siguiente pregunta de mi jefe?

"Alexis, cuales son esos rangos?" OH si...quizás olvide esa parte ☺



## 5. Calculando rangos IP en ambientes subneteados

Valor BIT	128	64	32	16	8	4	2	1
Mascara de Subred		192	224	240	248	252	254	255
Numero de Subnets		2	6	14	30	62	126	254

Aquí esta nuestra pequeña tabla de subneteo nuevamente. En esta sección, aprenderemos como derivar los rangos de IP actuales de las direcciones de red y la mascara.

En nuestro ejemplo, tenemos:

Rango IP original: 131.107.x.x

Mascara de subred original: 255.255.0.0

Mascara de subred subneteadas: 255.255.224.0

Numero de subredes: 6

El primer rango valido será de 131.107.32.1 a 131.107.63.254. Como llegue a esta conclusión? Yo use nuestra tabla de arriba.

Una vez que yo determine que 224 es mi subnet BIT, me pregunto a mi mismo: cual es el valor de BIT mas bajo para obtener 224?" La respuesta es : 32 (128+64+32=224, y 32 es el mas bajo de los 3 valores de BIT.) Si miramos la tabla, veremos esto:

BIT value	32
Subnet Mask	224
Number of subnets	6

Visualmente, es fácil ver que tenemos 6 subnets. Hemos usado 224 subnet BIT y mi rango comenzara en 32. Entonces incrementaremos cada rango por el mismo valor de 32. Yo adoro esta tabla!

Mis 6 rangos serán:

131.107.32.1 to 131.107.63.254

131.107.64.1 to 131.107.95.254

131.107.96.1 to 131.107.127.254

131.107.128.1 to 131.107.159.254

131.107.160.1 to 131.107.191.254

131.107.**192**.1 to 131.107.**223**.254

Como podemos ver, para obtener el siguiente rango, simplemente incremente mi valor original hasta el siguiente valor (miren los números en **ROJO**) Comenze con 32, entonces incremente 5 veces por el mismo valor de 32.

Si miras los valores en **AZUL** veras que siempre será el siguiente valor rojo menos 1. Por ejemplo, si miramos el primer rango x.x.32.1 a x.x.63.254, el siguiente valor rojo de la otra línea será 64, pero cuando le extraemos 1 tenemos 63. Esto es todo lo que tenemos que hacer!

Debes notar que mi ultimo octeto en el comienzo del rango siempre es 1– no puede ser 0, o sino mi host ID serán todos 0s – y el ultimo octeto siempre será 254 en la ultima IP de cada rango – no puede ser 255, o sino todos serian 1s, que ya sabemos es la dirección de broadcast, y por lo tanto no esta disponibles para los hosts.

En este ejemplo, hemos subneteado una red clase B network en 6 subredes. Ten en cuenta que hay un detalle diferente si hacemos este procedimiento con subnets clase A y Clase C. Hagamos un ejemplo con la clase A.

Rango IP original: 10.x.x.x

Mascara de subred original: 255.0.0.0

Numero de subredes requeridas: 14

Para obtener 14 subredes necesitamos un subnet BIT de 240, por lo tanto mi nueva mascara de subred seria 255.240.0.0. Observa que mi subnet BIT es ahora el segundo octeto, no el tercero como en el ejemplo anterior. Esto tendrá relevante importancia cuando vayamos a crear nuestros rangos.

Cual es el valor más pequeño que necesitamos para tener 240? 16. Por lo tanto comenzare mis rangos con 16 e incrementaremos por el mismo valor de 16.

Mis rangos entonces son:

10.16.0.1 a 10.31.255.254	10.128.0.1a 10.143.255.254
10.32.0.1 a 10.47.255.254	10.144.0.1a 10.159.255.254
10.48.0.1 a 10.63.255.254	10.160.0.1 a 10.175.255.254
10.64.0.1 a 10.79.255.254	10.176.0.1 a 10.191.255.254
10.80.0.1 a 10.95.255.254	10.192.0.1a 10.207.255.254
10.96.0.1 a 10.111.255.254	10.208.0.1a 10.223.255.254
10.112.0.1a 10.127.255.254	10.224.0.1a 10.239.255.254

Ahora hagámoslo con la clase C. Recuerden, clase C es la mas difícil, así que observen cuidadosamente!

Rango IP original: 192.168.2.x

Mascara de subred original: 255.255.255.0

Numero de subredes requeridas: 6

Para obtener 6 subredes necesitamos un subnet BIT de 224, por lo tanto mi nueva mascara de subred será 255.255.255.224. Observa que mi subnet BIT ahora esta en el 4to octeto, no en el tercero o en el segundo como en los ejemplos anteriores. Esto tendrá gran importancia cuando creemos los rangos.

Cual es el valor de BIT más pequeño cuando queremos 224? 32. Por lo tanto comenzare mis rangos con 32 e incrementare por el mismo valor de 32

Mis rangos serian:

192.168.2.33 a 192.168.2.62	192.168.2.129 a 192.168.2.158
192.168.2.65 a 192.168.2.94	192.168.2.161a 192.168.2.190
192.168.2.97a 192.168.2.126	192.168.2.193 a 192.168.2.222

Entonces tú dices, porque no comenzamos en 32?!? En su lugar el primer rango comienza en 33! Bueno, recordamos los otros ejemplos? Nosotros siempre empezábamos en .1, no? Excepto aquí, porque ya estamos subneteando en el 4to octeto, no tenemos un 5to para agregar el .1, así que lo tenemos que incorporar al último octeto. Porque no podemos usar 192.168.2.32 con una mascara de subred de 255.255.255.224? Veamos porque:

192.168.2.32	11000000.	10101000.	00000010.	00100000
255.255.255.224	11111111.	11111111.	11111111.	11100000

Mirándolo en binario, es obvio que el **Host ID** son todos ceros, lo cual sabemos que no es posible...

Y eso es todo. Si tú conoces y entiendes esto, entonces ya puedes subnetear. Disfruta practicando.