



REDES DE COMPUTADORES  
Laboratorio

**Práctica 2: ANEXO Planificación redes y subredes**

## 1. Subneteo de una Red.

La función del Subneteo o Subnetting es dividir una red IP física en subredes lógicas (redes más pequeñas) para que cada una de ellas trabaje a nivel envío y recepción de paquetes como una red individual, aunque todas pertenezcan a la misma red física y al mismo dominio.

El Subneteo permite una mejor administración, control del tráfico y seguridad al segmentar la red por función. También mejora el rendimiento general, al reducir el tráfico de broadcast de nuestra red.

- **Porción de Red:**

En el caso que la máscara sea por defecto, una dirección con Clase, la cantidad de bits "1" en la porción de red, indican la dirección de red, es decir, la parte de la dirección IP que va a ser común a todos los hosts de esa red.

- **Porción de Host:**

La cantidad de bits "0" en la porción de host de la máscara, indican que parte de la dirección de red se usa para asignar direcciones de host, es decir, la parte de la dirección IP que va a variar según se vayan asignando direcciones a los hosts.

- **Calcular la Cantidad de Subredes y Hosts por Subred:**

Cantidad de Subredes es igual a  $2^N$ , donde "N" es el número de bits "robados" a la porción de Host.

Cantidad de Hosts por Subred es igual a:  $2^M - 2$ , donde "M" es el número de bits disponible en la porción de host y "-2" es debido a que toda subred debe tener su propia dirección de red y su propia dirección de *broadcast*. Actualmente para obtener la cantidad de subredes se utiliza la fórmula  $2^M$ , que permite utilizar tanto la subred *cero* como la subnet *broadcast* para ser asignadas.

Entre los tipos de subneteo más elementales, encontramos:

- **Por clase**, genera una máscara común (*fija*) y una cantidad de hosts iguales a todas las subredes. Desaprovecha un gran número de direcciones.
- **Por VLSM (Variable Length Subnet Mask)**, toma una dirección de red o subred y la divide en subredes más pequeñas adaptando las máscaras según las necesidades de hosts de cada subred, generando una máscara diferente para las distintas subredes de una red. Esto permite no desaprovechar un gran número de direcciones, sobre todo en los enlaces seriales.

### 1.1 Ejemplo de subneteo por clase de una Red Clase C.

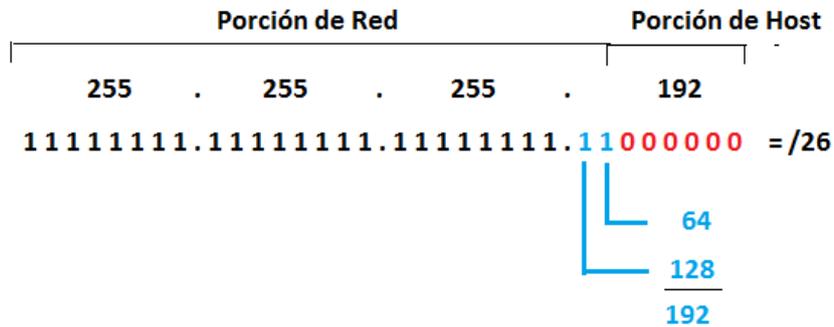
*Nos dan la dirección de red Clase C 192.168.1.0 /24 para realizar mediante subneteo 4 subredes con un mínimo de 50 hosts por subred.*

a) **Adaptar la Máscara de Red por Defecto a Nuestras Subredes.**

La máscara por defecto para la red 192.168.1.0 es 255.255.255.0

Usando la fórmula  $2^N$ , donde N es la cantidad de bits que tenemos que robarle a la porción de host, adaptamos la máscara de red por defecto de la subred.

Nos piden 4 subredes, es decir que el resultado de  $2^N$  tiene que ser mayor o igual a 4. Para crear 4 subredes debemos robar 2 bits a la porción de host. Agregamos los 2 bits robados reemplazándolos por "1" a la máscara Clase C por defecto, y obtenemos la máscara adaptada 255.255.255.192.



**b) Obtener Cantidad de Hosts por Subred.**

Ya tenemos nuestra máscara de red adaptada, que va a ser **común** a todas las subredes y hosts que componen la red. Ahora queda obtener los hosts, para lo cual vamos a trabajar con la dirección IP de red, específicamente con la porción de host (*rojo*).

El ejercicio nos pedía un mínimo de 50 hosts por subred. Para esto utilizamos la fórmula  $2^M - 2$ , donde M es el número de bits "0" disponibles en la porción de host y - 2 porque la primer y última dirección IP de la subred no se utilizan por ser la dirección de la subred y broadcast respectivamente. En nuestro caso,  $2^6 - 2 = 62$  hosts por subred.

Los 6 bits "0" de la porción de host, son los que vamos a utilizar según vayamos asignando los hosts a las subredes.

**c) Obtener Rango de Subredes.**

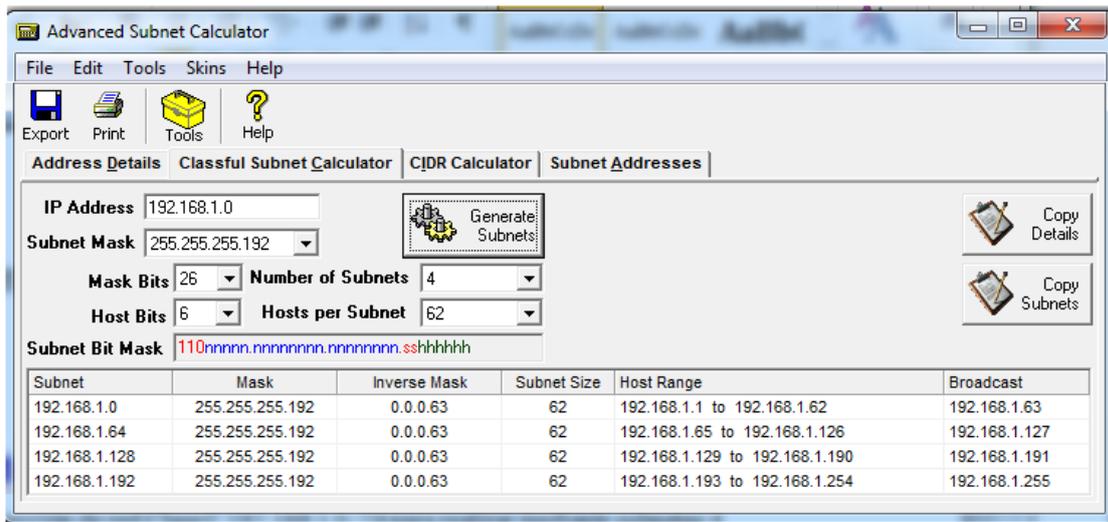
Para obtener el rango subredes, utilizamos la porción de red de la dirección IP que fue modificada al adaptar la máscara de red. A la máscara de red se le agregaron 2 bits en el cuarto octeto, entonces se van a tener que modificar esos mismos bits pero en la dirección IP.

Los 2 bits "0" de la porción de red son los modificaremos según vayamos asignando las subredes. Para obtener el rango la forma más sencilla, se restará a 256 el número de la máscara de subred adaptada. En este caso sería:  $256 - 192 = 64$ , obtenemos 64 que va a ser el rango entre cada subred.

Nº de Subred	Rango IP		Host asignables por Subred
	Desde	Hasta	
<b>0</b>	<b>192.168.1.0</b>	<b>192.168.1.63</b>	<b>62</b>
<b>1</b>	<b>192.168.1.64</b>	<b>192.168.1.127</b>	<b>62</b>
<b>2</b>	<b>192.168.1.128</b>	<b>192.168.1.191</b>	<b>62</b>
<b>3</b>	<b>192.168.1.192</b>	<b>192.168.1.255</b>	<b>62</b>

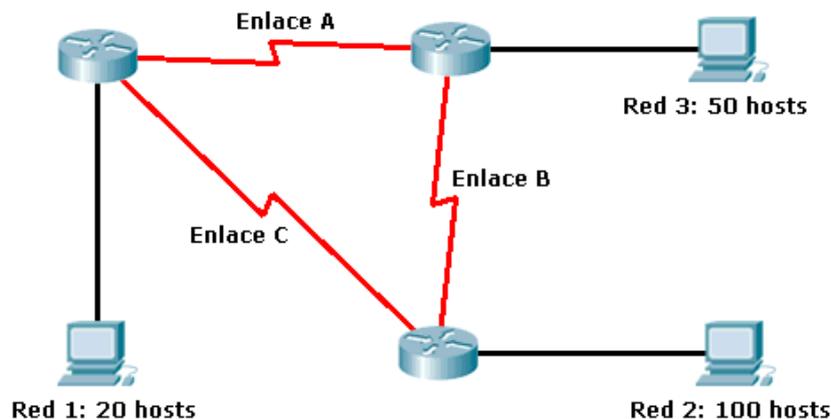
**Nota:** La primera y última dirección de cada rango no se tendrían presentes, ya que corresponden a la dirección de Red y la dirección de Broadcast.

Si lo resolvemos ayudándonos de la calculadora de subredes, tenemos:



## 1.2 Ejemplo de subneteo con VLSM de una Red Clase C.

Dada la siguiente topología y la dirección IP 192.168.1.0/24, se nos pide que por medio de subneteo con VLSM obtengamos direccionamiento IP para los hosts de las 3 subredes, las interfaces Ethernet de los routers y los enlaces seriales entre los routers.



### a) Calcular Cantidad de Direcciones IP para toda la Topología.

Lo primero que tenemos que hacer es organizar la cantidad de hosts de cada subred de **mayor a menor**, sumarle a los hosts de cada subred 2 direcciones (una dirección de red y broadcast) y 1 dirección más para la interfaz Ethernet del router.

**Red 2:** 100 host + 2 (red y broadcast) + 1(Ethernet) = **103 direcciones**

**Red 3:** 50 host + 2 (red y broadcast) + 1(Ethernet) = **53 direcciones**

**Red 1:** 20 host + 2 (red y broadcast) + 1(Ethernet) = **23 direcciones**

**Total Redes:** 103 + 53 + 23 = **179 direcciones**

Por cada enlace serial necesitamos 4 direcciones, 2 para las interfaces serial, y 2 para dirección de red y broadcast.

**Enlace A:**  $2 + 2$  (red y broadcast) = **4 direcciones**

**Enlace B:**  $2 + 2$  (red y broadcast) = **4 direcciones**

**Enlace C:**  $2 + 2$  (red y broadcast) = **4 direcciones**

**Total Enlaces:**  $4 + 4 + 4 = 12$  direcciones

Sumamos todas las direcciones y obtenemos la totalidad de direcciones IP que vamos a necesitar para la topología.

**Total Redes + Total Enlaces:**  $179 + 12 = 191$  direcciones

Una vez que sabemos la cantidad de direcciones que vamos a necesitar, tenemos que asegurarnos que con la dirección IP dada, se pueda alcanzar ese número sin importar el número de subredes que necesitemos. Para ello tomamos la máscara de la dirección IP 192.168.1.0/24, la convertimos a binario y diferenciamos la porción de red y host. Con los 8 bits de la porción de host podemos obtener 256 direcciones ( $2^8 = 256$ ), como nosotros necesitamos **191 direcciones**, es por tanto *es viable*.

#### b) Obtener Direccionamiento IP para las Subredes.

Para obtener las subredes siempre se comienza **de mayor a menor** según la cantidad de direcciones. Entonces vamos a empezar primero por la **Red 2** (103 direcciones), luego por la **Red 3** (53 direcciones), luego por la **Red 1** (23 direcciones) y por **último los 3 enlaces seriales** (4 direcciones cada uno).

#### I. Obtener Direccionamiento IP para la Red 2 = 103 Direcciones.

Para obtener la **Red 2**, lo primero que tenemos que hacer es adaptar la máscara de red de la dirección IP 192.168.1.0 /24, que como ya vimos permite 256 direcciones ( $2^8 = 256$ ).

Averiguamos cuantos bits "0" se necesitan en la porción de host de la máscara de red para obtener un mínimo de 103 direcciones, vemos que con  $2^7$  obtenemos 128 direcciones, es decir que de los 8 bits "0" de la máscara de red original solo necesitamos 7 bits "0" (de derecha a izquierda) para las direcciones. A la porción de host le robamos ese bit "0" restante y lo reemplazamos por un bit "1", haciéndolo parte de la porción de red. La máscara de red adaptada va a quedar  $255.255.255.128 = /25$ , nos permite 2 subredes con 128 direcciones cada una.

Sabemos que la **subred cero** es la 192.168.1.0 /25 y que va a ser para la **Red 2**. Ahora nos restaría obtener el rango de la **subred uno**. Para obtener el rango entre subredes la forma más sencilla es restarle al número 256 el número de la máscara de subred adaptada:  $256 - 128 = 128$ . Entonces el rango entre las subredes va a ser 128, es decir, que la **subred uno** va a ser 192.168.1.128 /25.

Nº de Subred	Rango IP		Dir. por Subred	Máscara	Asignada
	Desde	Hasta			
0	192.168.1.0	192.168.1.127	128	/25	Red 2
1	192.168.1.128	192.168.1.255	128	...	...

**Nota:** La primera y última dirección de cada rango no se tendrían presentes, ya que corresponden a la dirección de Red y la dirección de Broadcast.

## II. Obtener Direccionamiento IP para la Red 3 = 53 Direcciones.

Para obtener la **Red 3** se necesita un mínimo de 53 direcciones, para lo cual vamos trabajar con la **subred uno** que hemos generado (192.168.1.128 /25), que permite 128 direcciones. La máscara de red adaptada va a quedar 255.255.255.192 = /26, permite 2 subredes con 64 direcciones cada una. Entonces la dirección IP 192.168.1.128 /26 con 64 direcciones va a ser la dirección de la **Red 3**.

Ahora nos restaría obtener la dirección de la siguiente subred de 64 direcciones. Volvemos a utilizar el método de resta para obtener el rango entre subredes: 256 - 192 = 64. Entonces el rango entre las subredes va a ser 64, y la **subred dos** va a ser 192.168.1.192 /26.

Nº de Subred	Rango IP		Dir. por Subred	Máscara	Asignada
	Desde	Hasta			
0	192.168.1.0	192.168.1.127	128	/25	Red 2
1	192.168.1.128	192.168.1.191	64	/26	Red 3
2	192.168.1.192	192.168.1.255	64	...	...

**Nota:** La primera y última dirección de cada rango no se tendrían presentes, ya que corresponden a la dirección de Red y la dirección de Broadcast.

## III. Obtener Direccionamiento IP para la Red 1 = 23 Direcciones.

Con la dirección de la **subred dos** generada 192.168.1.192 /26 que permite 64 direcciones, tenemos que obtener la **Red 1** que necesita un mínimo de 23 direcciones. La máscara de red adaptada va a quedar 255.255.255.224 = /27, permite 2 subredes con 32 direcciones cada una. La dirección IP 192.168.1.192 /27 con 32 direcciones va a ser para la **Red 1**.

Nos restaría obtener la subred siguiente de 32 direcciones. Hacemos la resta para rango entre subredes y obtenemos la **subred tres** con la dirección 192.168.1.224 /27.

Nº de Subred	Rango IP		Dir. por Subred	Máscara	Asignada
	Desde	Hasta			
0	192.168.1.0	192.168.1.127	128	/25	Red 2
1	192.168.1.128	192.168.1.191	64	/26	Red 3
2	192.168.1.192	192.168.1.223	32	/27	Red 1
3	192.168.1.224	192.168.1.255	32	...	...

**Nota:** La primera y última dirección de cada rango no se tendrían presentes, ya que corresponden a la dirección de Red y la dirección de Broadcast.

## IV. Obtener Direccionamiento IP para los Enlaces.

Para los enlaces se necesitan siempre 4 direcciones, (2 para enlaces, y 2 para dirección de red y broadcast), usamos para todos la misma máscara de red 255.255.255.252 = /30, permite 8 subredes con 4 direcciones cada una.

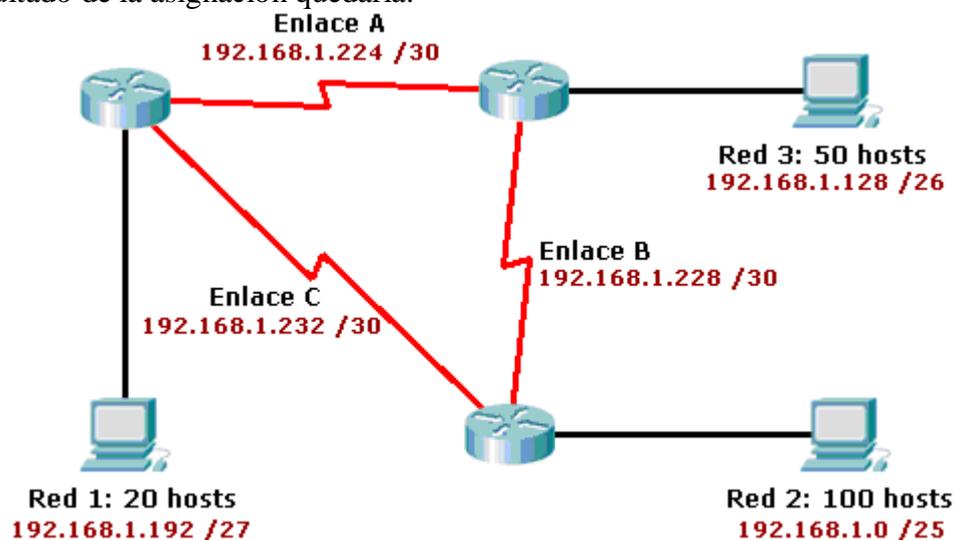
La dirección IP 192.168.1.224 /30 con 4 direcciones va a ser para el **Enlace A**, nos restaría obtener las 2 subredes para los **Enlaces B y C**.

Hacemos la resta para rango entre subredes y obtenemos las 2 direcciones restantes:  
**Enlace B** es 192.168.1.228 /30 y **Enlace C** es 192.168.1.232 /30.

Nº de Subred	Rango IP		Dir. por Subred	Máscara	Asignada
	Desde	Hasta			
0	192.168.1.0	192.168.1.127	128	/25	Red 2
1	192.168.1.128	192.168.1.191	64	/26	Red 3
2	192.168.1.192	192.168.1.223	32	/27	Red 1
3	192.168.1.224	192.168.1.227	4	/30	Enlace A
4	192.168.1.228	192.168.1.231	4	/30	Enlace B
5	192.168.1.232	192.168.1.235	4	/30	Enlace C

**Nota:** La primera y última dirección de cada rango no se tendrían presentes, ya que corresponden a la dirección de Red y la dirección de Broadcast.

En resultado de la asignación quedaría:



Para la comprobación de resultados de los cálculos realizados, podemos servirnos ayudándonos de calculadoras online de subredes VLSM. A continuación se presentan resultados con dos de ellas.

- Disponemos de la Web <http://www.vlsm-calc.net/>. (En este ejemplo, el algoritmo escoge para las 3 últimas subredes una máscara de /29 en vez de /30).

### Subnetting Successful

Major Network: 192.168.1.0/24  
 Available IP addresses in major network: 254  
 Number of IP addresses needed: 191  
 Available IP addresses in allocated subnets: 236  
 About 97% of available major network address space is used  
 About 81% of subnetted network address space is used

Subnet Name	Needed Size	Allocated Size	Address	Mask	Dec Mask	Assignable Range	Broadcast
Red 2	103	126	192.168.1.0	/25	255.255.255.128	192.168.1.1 - 192.168.1.126	192.168.1.127
Red 3	53	62	192.168.1.128	/26	255.255.255.192	192.168.1.129 - 192.168.1.190	192.168.1.191
Red 1	23	30	192.168.1.192	/27	255.255.255.224	192.168.1.193 - 192.168.1.222	192.168.1.223
Enlace A	4	6	192.168.1.224	/29	255.255.255.248	192.168.1.225 - 192.168.1.230	192.168.1.231
Enlace B	4	6	192.168.1.232	/29	255.255.255.248	192.168.1.233 - 192.168.1.238	192.168.1.239
Enlace C	4	6	192.168.1.240	/29	255.255.255.248	192.168.1.241 - 192.168.1.246	192.168.1.247

- También tenemos esta otra <https://subnettingpractice.com/vlsm.html>. (En este ejemplo, el algoritmo escoge para las 3 últimas subredes una máscara de /29 en vez de /30)

How many subnets do you need to create?

What is the network IP prefix in slash notation (i.e. 192.168.1.0/24)?

Subnet Names:	Number of hosts:
Host1	103
Host2	53
Host3	23
Host4	4
Host5	4
Host6	4

The network 192.168.1.0/24 has 254 hosts.  
Your subnets need 191 hosts.

Name	Hosts Needed	Hosts Available	Unused Hosts	Network Address	Slash	Mask	Usable Range	Broadcast	Wildcard
Host1	103	126	23	192.168.1.0	/25	255.255.255.128	192.168.1.1 - 192.168.1.126	192.168.1.127	0.0.0.127
Host2	53	62	9	192.168.1.128	/26	255.255.255.192	192.168.1.129 - 192.168.1.190	192.168.1.191	0.0.0.63
Host3	23	30	7	192.168.1.192	/27	255.255.255.224	192.168.1.193 - 192.168.1.222	192.168.1.223	0.0.0.31
Host4	4	6	2	192.168.1.224	/29	255.255.255.248	192.168.1.225 - 192.168.1.230	192.168.1.231	0.0.0.7
Host5	4	6	2	192.168.1.232	/29	255.255.255.248	192.168.1.233 - 192.168.1.238	192.168.1.239	0.0.0.7
Host6	4	6	2	192.168.1.240	/29	255.255.255.248	192.168.1.241 - 192.168.1.246	192.168.1.247	0.0.0.7

**NOTA:**

Para ampliar conocimientos sobre la materia tratada, podemos descargarnos de forma gratuita, el documento de muestra **Chapter 3: IP Subnetting and Variable Length Subnet Masks (VLSMs)** del libro **CCNA™: Cisco® Certified Network Associate Study Guide** proporcionado **SYBEX Sample Chapter CCNA™**.



**Chapter 3  
IP Subnetting and Variable Length Subnet Masks (VLSMs)**

THE CCNA EXAM TOPICS COVERED IN THIS CHAPTER INCLUDE THE FOLLOWING:

- ✓ **PLANNING & DESIGNING**
  - Design a simple LAN using Cisco Technology
  - Design an IP addressing scheme to meet design requirements
  - Design a simple internetwork using Cisco technology
- ✓ **TECHNOLOGY**
  - Evaluate TCP/IP communication process and its associated protocols
- ✓ **TROUBLESHOOTING**
  - Perform LAN and VLAN troubleshooting
  - Troubleshoot IP addressing and host configuration
  - Troubleshoot a device as part of a working network

## 2. Configuración básica de un router.

### 2.1 Modos de acceso.

El Cisco IOS (*Internetworking Operating System*) es el sistema operativo empleado por los routers de Cisco Systems. El Cisco IOS CLI (*Command-Line Interface*) es el método principal para la configuración, monitorización y mantenimiento de equipos Cisco.

El Cisco IOS CLI ofrece diferentes modos de comandos. Cada modo de comandos ofrece un conjunto de comandos diferentes y con diferente objetivo (configuración, mantenimiento, monitorización...). Los comandos disponibles en cada momento dependen del modo en el cual se encuentre el usuario. En cualquiera de estos modos se puede emplear el interrogante (?) para obtener una lista de los comandos disponibles en ese modo.

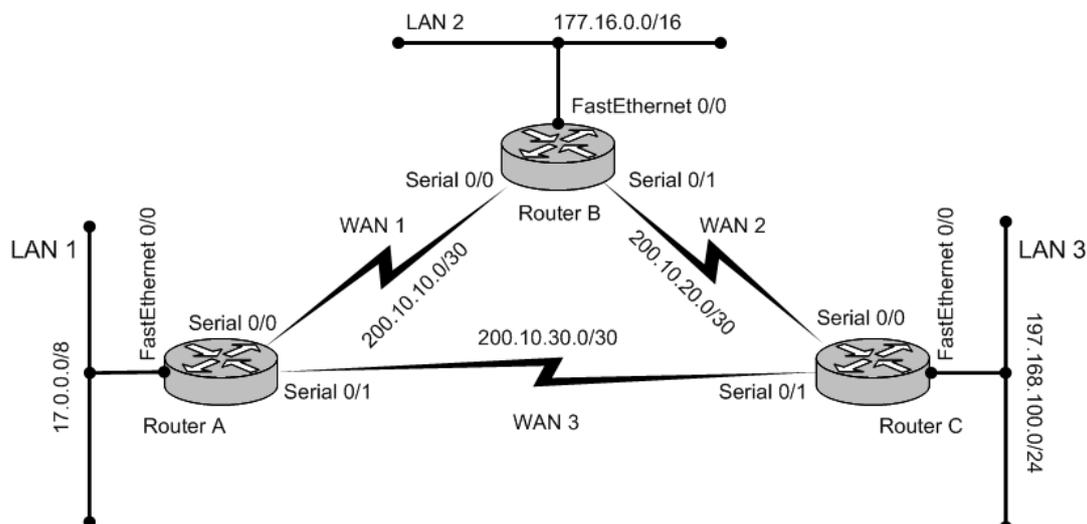
Los niveles de ejecución de comandos son:

- **Modo Usuario** (*es en el que nos encontramos por defecto al arrancar el router*):
  - Permite realizar tareas comunes como verificar el estado del equipo.
  - No permite cambiar la configuración del equipo.
- **Modo Privilegiado**:
  - Permite realizar tareas que pueden cambiar la configuración del equipo.

### 2.2 Encaminamiento estático.

El encaminamiento estático implica establecer manualmente las tablas de encaminamiento de cada router, hacia las rutas de las redes destino que dicho router no conoce.

Disponemos de esta red, a modo de ejemplo:



Donde tenemos en la siguiente tabla, las definiciones para el **Router A**:

Interfaz	Enlace	Dirección	Máscara de subred
FastEthernet	LAN1	17.0.0.1	255.0.0.0
Serial 0/0	WAN1	200.10.10.5	255.255.255.252
Serial 0/1	WAN3	200.10.30.5	255.255.255.252

Para el **Router B** tenemos:

Interfaz	Enlace	Dirección	Máscara de subred
FastEthernet	LAN2	177.16.0.1	255.255.0.0
Serial 0/0	WAN1	200.10.10.6	255.255.255.252
Serial 0/1	WAN2	200.10.20.5	255.255.255.252

Y para el **Router C**:

Interfaz	Enlace	Dirección	Máscara de subred
FastEthernet	LAN3	197.168.100.1	255.255.255.0
Serial 0/0	WAN2	200.10.20.6	255.255.255.252
Serial 0/1	WAN3	200.10.30.6	255.255.255.252

Queremos que cualquier paquete de datos originado, por ejemplo, en un host de la red **LAN1** pueda llegar a su destino, por ejemplo, un host en las redes **LAN2** o **LAN3**. Para lo cual debemos establecer tablas de encaminamiento en cada router.

- **Rutas estáticas en el Router A.**

La forma de hacerle saber al **Router A** cómo encaminar correctamente esos datagramas es incorporando a su tabla de encaminamiento información sobre esas redes y cómo llegar hasta ellas, es decir, debemos hacerle saber de la existencia de las redes 177.16.0.0 y 197.168.100.0 y hacia dónde reenviar los datagramas para llegar a las mismas

Veamos ahora que comandos tendríamos que aplicar en los routers para que nuestro ejemplo pueda funcionar. La primera vez que se conecte al router lo hará en el modo de usuario *no privilegiado*. Desde ahí, puede acceder al modo con privilegios tecleando `enable`, y para salir del modo se utiliza `exit`.

Comencemos por ver el contenido actual de la tabla de encaminamiento del **Router A**. Para ello utilizamos el comando de modo *Privilegiado* `show ip route`:

```
RouterA> enable
```

```
RouterA# show ip route
```

```
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
C 17.0.0.0/8 is directly connected, FastEthernet0/0
C 200.10.30.0/30 is directly connected, Serial0/1
C 200.10.10.0/30 is directly connected, Serial0/0
```

Como podemos ver, el **Router A** tiene información únicamente sobre las tres redes a las cuales está directamente conectado. En consecuencia, si desde **LAN1** el router recibe un paquete de datos destinado a la **LAN2** o a la **LAN3**, no sabrá hacia donde encaminarlo, simplemente porque desconoce la existencia de esas redes.

Para entrar en el modo *configuración global* teclearemos `configure terminal`, desde el que se realizará la configuración.

```
RouterA# configure terminal
```

```
RouterA(config)#
```

A continuación, debemos indicar la *dirección de la red de destino, su máscara de red, y a través de qué dirección IP accederemos al siguiente router*. Comencemos por la red **LAN 2**, cuya dirección IP es 177.16.0.0, y su máscara de red 255.255.0.0.

```
RouterA(config)# ip route 177.16.0.0 255.255.0.0 200.10.10.6
```

Sigamos por la red **LAN 3**, cuya dirección IP es 197.168.100.0, y su máscara de red 255.255.255.0

```
RouterA(config)# ip route 197.168.100.0 255.255.255.0 200.10.30.6
```

Una vez realizado esto, podemos nuevamente ejecutar el comando `show ip route` para comprobar que las nuevas rutas han quedado guardadas en la tabla de encaminamiento de dicho router.

Y para terminar.

```
RouterA(config)# end
```

De igual modo, este proceso se puede repetir para el resto de routers a configurar.

## **2.3 Archivos de Configuración de IOS.**

En los routers de Cisco encontramos dos archivos muy importantes para su funcionamiento: el archivo de *configuración de arranque* y el *archivo de configuración en ejecución*.

El archivo de *configuración en ejecución*, denominado `RUNNING-CONFIG`, contiene la configuración actual con la cual el router está ejecutándose. Este archivo se mantiene en la memoria RAM y se modifica cada vez que se ejecuta un comando de Configuración.

El archivo de *configuración de arranque*, denominado `STARTUP-CONFIG`, contiene la configuración que IOS utiliza inicialmente cuando se enciende el router. Este archivo se almacena en la memoria NVRAM, cuyo contenido no se borra cuando se apaga el router.

Durante el funcionamiento normal del router, estos dos archivos suelen ser iguales. Sin embargo, cuando se ejecuta algún comando de configuración para cambiar el valor de

algún parámetro de la configuración del router, este cambio se refleja en el archivo RUNNING-CONFIG. Si después de estos cambios se apaga el router, los mismos se pierden puesto que el archivo RUNNING-CONFIG se mantiene en la memoria RAM.

Si se desea que los cambios realizados en la configuración se mantengan después de reiniciar el router, debemos copiar el contenido del archivo RUNNING-CONFIG en el archivo STARTUP-CONFIG, ya que es desde este archivo donde IOS toma la configuración inicial cuando el router se enciende o se reinicializa. Para realizar esta copia se utiliza el comando de modo *Privilegiado* `copy running-config startup-config`. Lo ejecutaríamos como sigue:

```
Router# copy running-config startup-config
```

```
Destination filename [startup-config]?  
Building configuration...
```

IOS puede tardar uno o dos minutos en generar el archivo de configuración y salvarlo en la memoria NVRAM. Una vez que la configuración haya sido salvada, aparece el siguiente mensaje:

```
[OK]
```

```
Router#
```