

Práctica 2:

2.1.- Configuración básica del router con RIPv1 y RIPv2

2.2.- Recuperación de contraseñas.

Material necesario: maqueta de routers, cables de red y consola y ordenadores de consola.

Práctica 2.1: Configuración básica del router con RIPv1 y RIPv2

PARTE 1

Objetivo

En esta primera parte de la práctica se va a configurar el router de distintas maneras, primero usando el setup y posteriormente usando el interface de comandos del IOS. Antes de cambiar la configuración del router se realizara una simulación de la configuración.

Destacar que en los ejemplos utilizados se hace mención a uno de los routers de la maqueta, “**Lab-A**>”, pero que puede generalizarse al resto “**Lab-B**>”, “**Lab-C**>”, ...

Parte 1.1: Configurar un router mediante el setup.

Si se consulta el fichero de configuración que se ha capturado en la práctica anterior (debe tener una configuración parecida a la de la tabla inferior).

1. Mirar el mapa de topología y completar la tabla.

Fichero de Configuración	Indicar su función:
hostname _____	
enable secret genios	
enable password class	
interface Fastethernet0 ip address ____ . ____ . ____ . ____ 255.255.255.0 no shutdown	
interface Serial0 ip address ____ . ____ . ____ . ____ 255.255.255.252 clockrate 56000 //solo los DCE no shutdown	
interface Serial1	

ip address _____.____.____.____ 255.255.255.0 no shutdown	
router rip network _____.____.____.0 network _____.____.____.0 network _____.____.____.0	
banner login C Entrando en el router C banner motd C Router del U.V.E.G C	Banner son mensajes aparecidos en la conexión bien por login o por consola delimitados por el mismo carácter
line con 0 password netacad login	
line aux 0	
line vty 0 4 password cisco login	

2. Poner todo el cableado de los routers excepto los cables serie (azules)
3. Iniciamos la conexión al router mediante el *minicom* -s en el PC.

Se supone que el router no está configurado y por lo tanto al arrancar se podrá acceder al modo privilegiado sin necesidad de introducir un password tal y como se indica a continuación:

```
Router> enable
Router#
```

Probar con **show run**, que en efecto no hay una configuración introducida. Si hubiera una configuración introducida o nos pidiera un password de *enable* se debe borrar la configuración del *startup-config* de la siguiente forma:

4. Borramos la configuración del router y **Router# erase start.** y
5. Reiniciamos el router (todos los routers del laboratorio simultáneamente) **Router # reload.**
6. Cuando el router arranca, detecta que no hay ninguna configuración en la NVRAM y lo indica con el mensaje: “*NVRAM invalid*”. Por esto pide entrar en modo de configuración.
7. Con el mensaje “*Would you like enter the initial configuration dialog? [yes/no]*” debemos contestar **yes**

NOTA: Si se quiere salir del setup: pulsar control + c.

NOTA: En modo administrativo se puede entrar en el setup con el comando Router # setup y entre corchetes aparecerán los valores que el router tenga configurados

8. Ahora pide configurar mínimamente el router para que se pueda configurar a través de la red conectada a un interface del router. “*Would you like enter basic management setup? [yes/no]*” diremos que **no** para poder configurar todo el router. (no sólo un interface).
9. Vamos introduciendo las líneas de configuración:

- a) Como vemos, nos pide unos passwords de cada tipo de conexión, pondremos los indicados en la tabla de arriba.
 - b) No usamos configuración remota via SNMP, por eso en la pregunta: “*Configure SNMP Network Management?[yes]*” contestaremos **NO**
 - c) Introducir los datos que se pidan según la tabla que se ha completado anteriormente.
 - d) Se usa IP y RIP (mirar la topología para saber las IPs de cada)
10. Si nos pregunta sobre guardar la configuración “[2] *Save this configuration to NVRAM?*” responderemos que **NO** (si ponemos **YES** después tendremos que borrarla mediante: ***Router# erase start***)

NOTA: Cuando se escriben algunos comandos en el fichero de configuración después no se ven igual (con *Router#show run*) a lo que hemos escrito, como por ejemplo ocurre con:

(Router # enable secret genios → enable secret 4ht6e39w73hts72js91k39ks93).

(Router # banner motd C Router de U.V.E.G C → banner motd C Router de U.V.E.G C).

- 11. Conectar los cables serie (azules)
- 12. Para ver si la configuración que hemos hecho coincide con la topología propuesta (gráfica), comprobamos la conectividad con los comandos

Router > show interface

Router > ping _._._._._

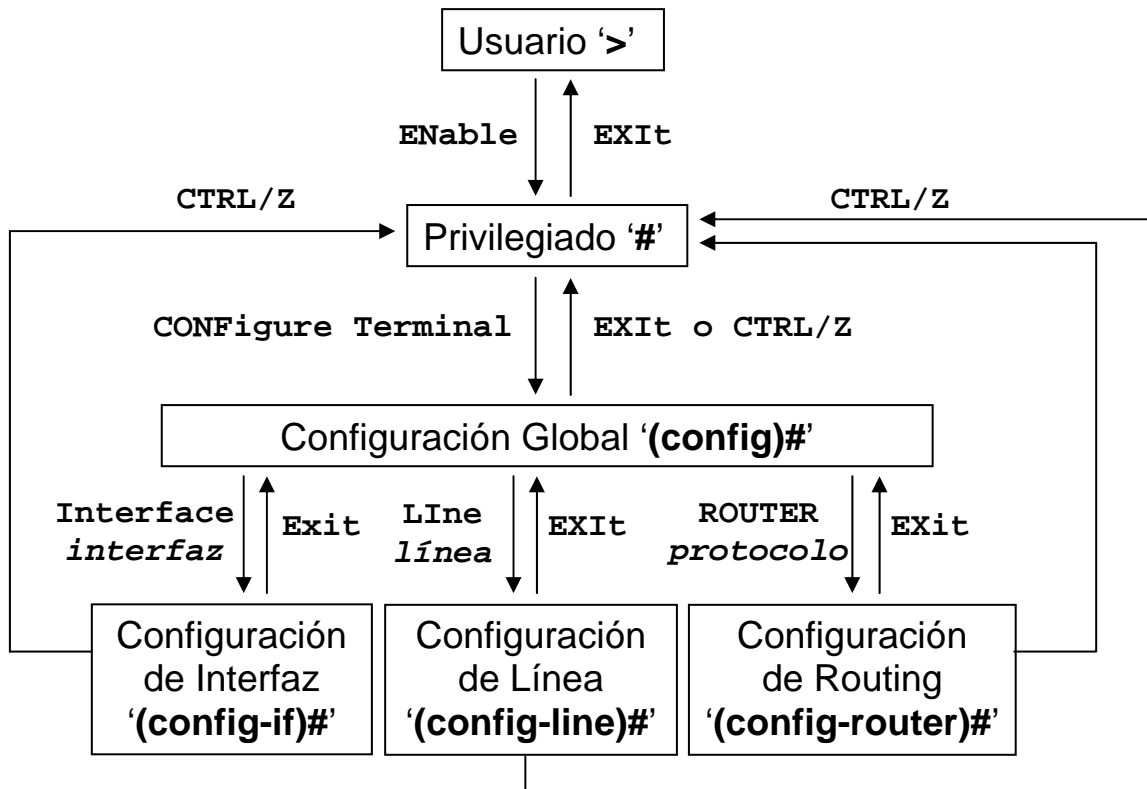
Parte 1.2: Modificar el fichero de configuración mediante comandos del IOS en el router.

Ahora habrá que configurar el router mediante comandos del IOS

- 1. Iniciamos la conexión al router mediante el *minicom -s*
- 2. Como se supone que no tenemos guardada una configuración en el router, sólo necesitamos apagarlos y volver a encenderlos para que arranquen sin ninguna configuración. Esto debe hacerse con los 5 routers de la maqueta a la vez o quitando los cables serie.
- 3. Cuando el router arranca, detecta que no hay ninguna configuración en la NVRAM y lo indica con el mensaje: “*NVRAM invalid*”. Por esto pide entrar en modo de configuración.
- 4. Con el mensaje “*Would you like enter the initial configuration dialog? [yes/no]*” debemos contestar **NO**

NOTA: Si por error se entra en el setup, pulsando [ctrl.]+C se sale del setup.

- 5. Ahora pide confirmación mediante la pregunta: “*Terminate autoinstall [yes]*” aquí se debe contestar **YES**



6. Entramos en modo privilegiado y vemos la configuración actual con el comando *Router#show run*. Los modos en los que vamos a entrar en configuración los podemos ver en la figura superior.

NOTA: Los valores por defecto que tiene el router si se quieren cambiar se ha de indicar anteponiendo a esa orden el comando *NO* (borra lo que le sigue a continuación), como por ejemplo *no shutdown*.

7. Vamos introduciendo las líneas de configuración para modificar y dejar bien configurado el router:

```
Router# config terminal
Router (config)#...
```

NOTA: Algunas líneas escritas en la configuración, con el *router#show run* no se muestran igual a las que hemos escrito. Como por ejemplo ocurre con:

```
(Lab-A# enable secret genios → enable secret 4ht6e39w73hts72js91k39ks93).
(Lab-A# banner motd C Router de U.V.E.G C → banner motd C Router de U.V.E.GC).
```

- 8 Comprobamos la conectividad con los comandos [*router#show interface*], [*router#ping _._._._*]

9. Mirar si todo funciona adecuadamente, para ello mirar la tabla de rutas:

```
Lab-A# show ip route →
```

¿Qué rutas ves, con qué distancia administrativa, con qué métrica y de dónde las ha aprendido el router (estáticas, directamente conectadas, RIP, IGRP, ...)? ¿Vés todas las rutas?

Depura las configuraciones para que ello ocurra.

10. Mirar el funcionamiento del RIP activando el debug con el comando:

Lab-A# debug ip rip → muestra los mensajes de actualización que envía y recibe el RIP

Lab-A# no debug ip rip → deja de mostrar los mensajes del RIP

PARTE 2: Compatibilidad RIP v1 y RIP v2

Objetivo

En esta parte de la práctica trataremos de configurar RIP v1 y RIP v2. Éste último será configurado para poder aceptar actualizaciones de tipo RIP v1. Posteriormente haremos uso de una de las características de RIP v2: autenticación.

Escenario

En la maqueta original tratemos de establecer que sólo los routers LAB_C, LAB_D y LAB_E sean capaces de configurarse con RIP v2 y por lo tanto manejar VLSM (máscaras de longitud variable) y autenticación.

Por este motivo, LAB_C, LAB_D y LAB_E necesitarán configurarse para recibir actualizaciones RIP v1 para que tanto LAB_A como LAB_B puedan comunicarse con el resto de la red.

Paso 1

Comprobar antes de pasar al siguiente paso la conectividad (si no lo habéis hecho aún) entre todos los elementos de la red, haciendo uso de RIP v1.

Paso 2

Hacer uso de los siguientes comandos para verificar el protocolo de enrutamiento que está ejecutándose:

NOTA: NO TENGAIS EN CUENTA EL DIRECCIONAMIENTO USADO EN LOS EJEMPLOS MOSTRADOS. SON SOLO EJEMPLOS ACLARATORIOS.

Router#**show ip protocols**

Routing Protocol is "rip"

Sending updates every 30 seconds, next due in 14 seconds

Invalid after 180 seconds, hold down 180, flushed after 240

Outgoing update filter list for all interfaces is

Incoming update filter list for all interfaces is

Redistributing: rip

Default version control: send version 1, receive any version

```
Interface Send Recv Triggered RIP Key-chain
Serial0/0 1 1 2
Routing for Networks:
192.168.1.0
192.168.224.0
Routing Information Sources:
Gateway Distance Last Update
192.168.224.2 120 00:00:22
Distance: (default is 120)
```

Ahora verificar el tipo de actualizaciones que el router está recibiendo.

```
Router#debug ip rip
RIP protocol debugging is on
00:21:28: RIP: sending v1 update to 255.255.255.255 via Serial0/0
(192.168.224.1)
00:21:28: RIP: sending v1 update to 255.255.255.255 via Loopback0
(192.168.1.3)
00:21:28: RIP: build update entries
00:21:28: network 192.168.3.0 metric 2
00:21:28: network 192.168.224.0 metric 1
00:21:28: network 192.168.240.0 metric 2
00:21:28: network 192.168.252.0 metric 3
00:21:41: RIP: received v1 update from 192.168.224.2 on Serial0/0
00:21:41: 192.168.3.0 in 1 hops
00:21:41: 192.168.240.0 in 1 hops
00:21:41: 192.168.252.0 in 2 hops
```

Fijaros que todas las actualizaciones RIP v1 son enviadas usando **direcciones broadcast** a través de todas las interfaces. Los anuncios son también *classfull*, es decir ninguna información sobre mascarar es enviada (no hay soporte para VLSM). Desactivad **debug** si está habilitado y comprobar las rutas mediante la observación de las tablas de enrutamiento con **show ip route**

Paso 3

El siguiente paso es configurar RIP v2 sobre los routers que lo admiten (LAB_C, LAB_D y LAB_E)

Ejemplo:

```
LAB_C(config)#router rip
LAB_C (config-router)#version 2
```

Mediante el comando **show running-config** en estos routers comprobar que la versión 1 ha sido reemplazada por la RIP v2. Teclea el comando **clear ip route *** para “vaciar” la tabla de enrutamiento.

Comprobar las tablas de enrutamiento en estos routers. ¿Qué rutas se conocen ahora?

Por defecto, RIP v1 aceptará actualizaciones RIP v2. Sin embargo, RIP v2 no aceptará RIP v1. Por este motivo, RIP v2 debe ser configurado para aceptarlas.

Para observar esto, utilizaremos el comando **debug ip rip** sobre los diferentes routers, para ver que los mensajes que aparecen utilizando RIP v1 permiten la aceptación de actualizaciones RIP v2, mientras que los routers que utilizan RIP v2 ignoran actualizaciones RIP v1.

Ejemplo.

RouterX#debug ip rip

RIP protocol debugging is on

01:12:28: RIP: received v2 update from 192.168.224.2 on Serial0/0

01:12:28: 192.168.3.0/24 via 0.0.0.0 in 1 hops

01:12:28: 192.168.240.0/24 via 0.0.0.0 in 1 hops

01:12:28: 192.168.252.0/24 via 0.0.0.0 in 2 hops

01:12:32: RIP: sending v1 update to 255.255.255.255 via Serial0/0(192.168.224.1)

01:12:32: RIP: build update entries

01:12:32: network 192.168.1.0 metric 1

01:12:32: RIP: sending v1 update to 255.255.255.255 via Loopback0 (192.168.1.3)

01:12:32: RIP: build update entries

01:12:32: network 192.168.3.0 metric 2

01:12:32: network 192.168.224.0 metric 1

01:12:32: network 192.168.240.0 metric 2

01:12:32: network 192.168.252.0 metric 3

RouterY#debug ip rip

RIP protocol debugging is on

01:49:40: RIP: sending v2 update to 224.0.0.9 via Serial0/0 (192.168.224.2)

01:49:40: RIP: build update entries

01:49:40: 192.168.3.0/24 via 0.0.0.0, metric 1, tag 0

01:49:40: 192.168.240.0/24 via 0.0.0.0, metric 1, tag 0

01:49:40: 192.168.252.0/24 via 0.0.0.0, metric 2, tag 0

01:49:40: RIP: sending v2 update to 224.0.0.9 via Serial0/1 (192.168.240.1)

01:49:40: RIP: build update entries

01:49:40: 192.168.3.0/24 via 0.0.0.0, metric 1, tag 0

01:49:40: 192.168.224.0/24 via 0.0.0.0, metric 1, tag 0

01:49:40: RIP: sending v2 update to 224.0.0.9 via Loopback0 (192.168.3.1)

01:49:40: RIP: build update entries

01:49:40: 192.168.224.0/24 via 0.0.0.0, metric 1, tag 0

01:49:40: 192.168.240.0/24 via 0.0.0.0, metric 1, tag 0

01:49:40: 192.168.252.0/24 via 0.0.0.0, metric 2, tag 0

01:49:40: RIP: ignored v2 packet from 192.168.3.1 (sourced from one of our addresses)

01:49:45: RIP: received v2 update from 192.168.240.2 on Serial0/1

01:49:45: 192.168.252.0/24 via 0.0.0.0 in 1 hops

01:40:45: RIP: ignored v1 packet from 192.168.224.1 (illegal version)

Paso 4

Configurar RIP v2 para aceptar actualizaciones RIP v1.

Ejemplo:

```
LAB_C(config-if)#interface serial 0  
LAB_C( (config-if)#ip rip receive version 1
```

Al igual que antes, ejecutar clear ip route * y de Nuevo comprobar las tablas de enrutamiento. ¿Qué rutas se conocen?

Habilitar **debug ip rip** sobre los routers LAB_C, LAB_D y LAB_E (RIP v2) y comprobar que ahora aceptan actualizaciones RIP v1

Recordar que las actualizaciones RIP v1 son enviadas a través de broadcast. ¿Cómo son enviadas las actualizaciones RIP v2?

Llegados a este punto trataremos de establecer autenticación entre los routers, y esto solo va a ser posible mediante el protocolo de enrutamiento RIP v2. Procederemos por tanto a eliminar el protocolo RIP v1 en aquellos routers en los que estaba configurado. (uso del comando “**no**” en la configuración).

Paso 5

Configuración de la autenticación (sobre todas las interfaces serie):

Por ejemplo, sobre el router LAB_A quedaría:

```
LAB_A(config)#key chain private  
LAB_A(config-keychain)#key 1  
LAB_A(config-keychain-key)#key-string 234
```

....

Habilitar la autenticación sobre las interfaces:

```
LAB_A(config)#interface serial 0  
LAB_A(config-if)#ip rip authentication key-chain private
```

Ejecutar el comando **debug ip rip** de nuevo sobre los routers y analizar la salida obtenida:

Deberá aparecer alguna línea como la siguiente:

```
01:10:35: RIP: received packet with text authentication 234
```

Paso 6.- Ojo. Esta parte puede que no funcione

La implementación de CISCO de RIP v2 soporta autenticación MD5. Esto provee un alto nivel de seguridad por encima del uso de texto plano. Todos los routers necesitarán configurarse con MD5.

La clave numérica y la de cadena debe coincidir entre cada par de routers o la autenticación fallará.

Ejemplo:

```
LAB_A(config)#interface serial0/0  
LAB_A(config-if)#ip rip authentication mode md5
```

Examinar de nuevo, haciendo uso de debug.

Observar si hay problemas con las rutas ahora y en el caso de que los haya, haced uso de comandos como **show ip rip database** y de otros ya vistos con anterioridad.

PARTE 3. Limitación de RIP v1

Objetivo

En este punto vamos a tratar de entender una de las limitaciones que caracterizan a RIP v1 en cuanto a que no es capaz de transmitir en sus actualizaciones información sobre las máscaras de subredes, dando lugar a problemas de convergencia.

Escenario

Para lograr nuestro objetivo, estableceremos un direccionamiento distinto al usado en la maqueta anterior: Haremos uso de la dirección de red de clase C 192.168.1.0 /24 y partiendo de ella crearemos subredes, según los siguientes requerimientos:

- Se requiere que como mínimo hayan 25 direcciones de host, disponibles en cada LAN, pero a su vez se requiere que se conserven el máximo número de direcciones para futuro crecimiento de la red.

Por lo que para soportar 25 hosts en cada subred se necesitarán un mínimo de 5 bits en cada porción de red relativa a los hosts. Por ello tendríamos 3 bits más para direccionamiento de red (/27)

Para maximizar este direccionamiento, la subred 192.168.1.0/ 27 podemos dividirla de nuevo haciendo uso de una máscara de 30 bits. MIRAR TABLA ADJUNTA. De esta forma estableceremos una gestión mucho mejor de nuestras direcciones, ya que cada

subred sólo permite 2 posibles direcciones de hosts (las necesarias para direccionar cada uno de los extremos de la línea serial que conecta cada par de routers).

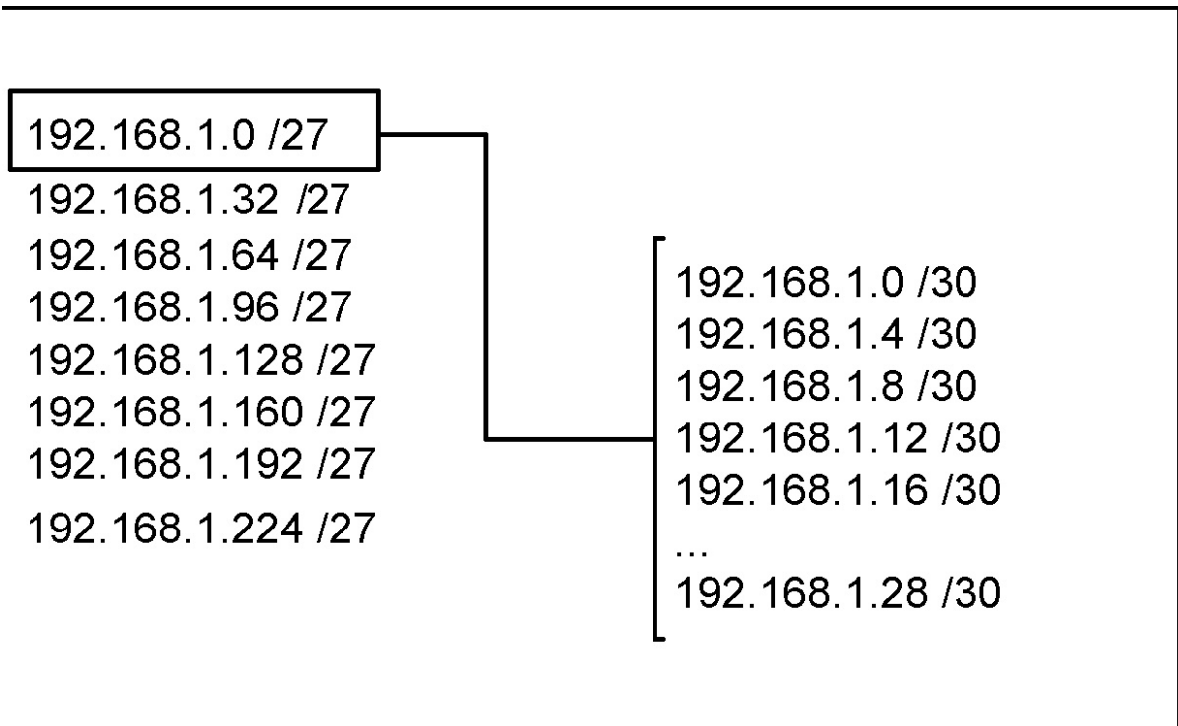


Tabla: Direccionamiento de subredes.

Por ejemplo, la línea S1 que une los routers LAB_A y LAB_E podría utilizar las direcciones 192.168.1.1 /30 y 192.168.1.2/30 para cada uno de los extremos respectivamente. La S0 que une los routers LAB_A y LAB_B las direcciones 192.168.1.5/30 y 192.168.1.6/ 30 y siguiendo este orden proseguir con el resto de la maqueta.

En cuanto a las LANs , por ejemplo para el primero de los routers (LAB_A):

Haciendo uso de las direcciones de red 192.168.1.32 / 27 y 192.168.1.64/27

Interfaz E0: 192.168.1.33/ 27

Interfaz F0: 192.168.1.65/27

Host ET_1: 192.168.66/27

“MIRAR LA MAQUETA ANEXA AL FINAL DE ESTE ENUNCIADO”

Paso 1

1. Como se supone que no tenemos guardada una configuración en el router, solo necesitamos apagarlos y volver a encenderlos para que arranquen sin ninguna configuración. Esto debe hacerse con los 5 routers de la maqueta a la vez o

quitando los cables serie. Si hubieses guardado una configuración acuérdate de hacer *erase start* y luego *reload*

2. Cuando el router arranca, detecta que no hay ninguna configuración en la NVRAM y lo indica con el mensaje: “*NVRAM invalid*”. Por esto pide entrar en modo de configuración.
3. Con el mensaje “*Would you like enter the initial configuration dialog? [yes/no]*” debemos contestar **no**
4. **Entrar en modo “configuración global” y posteriormente configurar cada una de las interfaces en cada uno de los routers.** Cuando se hace subnetting, la primera y última subred quedan inutilizadas. Eso ocurre porque la dirección de subred de la primera subred coincide con la dirección de subred de la red mayor (o subneteada) y la dirección broadcast de la última subred coincide con la dirección broadcast de la red mayor (o subneteada). **Para que los routers puedan trabajar con la primera subred y con la última el IOS activa por defecto el comando “ip subnet zero” (“no ip subnet zero” para desactivar la opción).** Esto depende de la versión de la IOS. **IMPORTANTE: COMPROBAR CON SHOW RUN QUE ESA LINEA EXISTE.**

Una vez establecido el direccionamiento, estableceremos de nuevo en todos los routers el RIP v1 (RIP por defecto), y comprobaremos otra vez la conectividad entre elementos de la red.

Paso 2

Haced uso del comando **show ip route** sobre cada router y comprobar que rutas no conoce el router en cuestión.

¿A qué creéis que es debido este desconocimiento?

Paso 3

Habilitar en todos los routers RIP v2

Ejemplo:

```
LAB_A(config)#router rip
```

```
LAB_A (config)#network 192.168.1.0 (en el caso en que RIP haya sido eliminado)
```

```
LAB_A (config-router)#version 2
```

```
LAB_A (config-router)#no auto-summary
```

Configurando **no auto-summary** inhabilitamos la sumarización automática de rutas de subred en sus direcciones de red base, permitiendo enviar información de subredes en las actualizaciones.

Haced uso **show ip route** de nuevo y comprobar el conocimiento que ahora han adquirido los distintos routers.

Práctica 2.2: Recuperación de contraseñas.

Explicación: Resumen de la teoría referente a los valores del registro:

El registro de configuración de la NVRAM tiene 16bits, que se agrupan de 4 en 4 y se traducen a hexadecimal:

$$16 \text{ bits} = [_ _ _ _] + [_ _ _ _] + [_ _ _ _] + [_ _ _ _] \text{ bits} = \quad \begin{matrix} [_][_][_][_] \text{ en hexadecimal.} \\ [2][1][0][2] \text{ (valor usual 2102)} \end{matrix}$$

1. Si el bit nº 6 es 1 se ignora la configuración de la NVRAM.

Ejemplos:

[0010][0001][0000][0010] bin → [2][1][0][2] Hex → (2102) Permite la carga de la configuración de la NVRAM (start-config)

[0010][0001][0100][0010] bin → [2][1][4][2] Hex → (2142) Inhabilita la configuración que hay en la NVRAM (detectara los interfaces y los pondrá shutdown por defecto)

2. Los 4 bits de menor peso indican de el lugar de carga del IOS.

[2][1][0][0000] → [2][1][0][0] → igual que [control]+[pausa] (entra en **monitor ROM**). Pausa es break

[2][1][0][0001] → [2][1][0][1] → Carga el IOS reducido de la ROM.

[2][1][0][0010] → [2][1][0][2]

..... → secuencia de inicio estándar:

[2][1][0][1111] → [2][1][0][F]

SECUENCIA DE BÚSQUEDA DEL IOS ESTANDAR:

- 1º) nvram
- 2º) flash.
- 3º) tftp.
- 4º) ROM

Explicación: Como modificar el registro de configuración.

Para modificar el registro de configuración se deberemos usar los siguientes comandos:

Comandos a introducir	Descripción
<i>Lab-a#Show version</i>	En la última línea que aparece se muestra el valor del registro de configuración actual. (normalmente: 0x2102)
<i>Lab-a#config terminal</i>	
<i>Lab-a(config)# config-register 0x2142</i>	Este comando cambia el valor del registro y se graba directamente en la NVRAM (no hace falta salvar con <i>copy</i>).
<i>Lab-a(config)# exit</i>	
<i>Lab-a# Show version</i>	Vemos que ahora el registro de configuración es (0x2142)

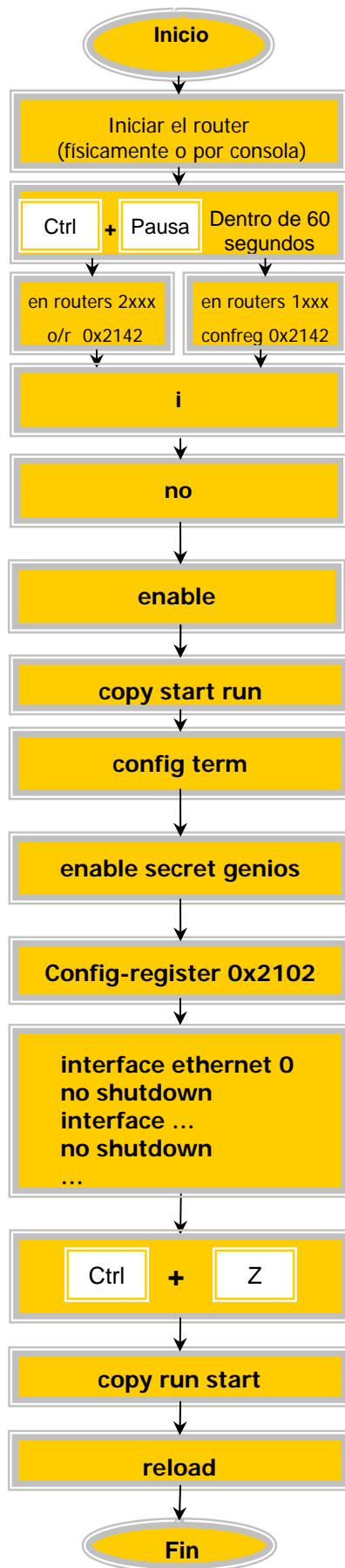
Cual es la diferencia entre los dos registros (0x2102) y (0x2142):

- a) A nivel de bits: _____
- b) A nivel de función: _____

NOTA: Este procedimiento sólo se puede realizar mediante una conexión al puerto de consola. (ni mediante telnet ni mediante el puerto AUX).

Explicación: Procedimiento de recuperación de la contraseña.

Destacar que el “Ctrl.+ Pausa” exige de una combinación diferente de teclas según el terminal y sistema operativo. Ver tabla siguiente.



- Debemos tener acceso físico al router y estar conectados al router por el puerto de consola.
- Reiniciamos el router
- Usando Ctrl+a+f se detiene la secuencia de inicio y entra en **monitor ROM** (para el minicom).
- En el **monitor ROM** cambiamos el registro de configuración por el **0x2142** (ignora la **NVRAM**)
- **(init)**: inicia la carga normal del Bootstrap-configuración-IOS. (La configuración de la **NVRAM** no es leída).
- **(no)**: Al no cargar el fichero de configuración el router muestra el acceso al **setup** para configurarse.
- Permite la entrada en modo administrador sin password (Router sin configuración). Hacer **show run** para ver que hay
- Recuperamos la configuración que hay en la NVRAM
- Al estar en modo administrador podemos modificar el fichero de configuración donde esta el password que hemos olvidado. Introduciendo el nuevo password de administrador, de consola, o cualquier otro: **enable secret (password) genios**
- Cambiamos el registro de configuración por el inicial **(0x2102)** (para que pueda cargar el **startup**)
- Hay que levantar todas las interfaces que se utilicen (el arranque automáticamente las ha desactivado)
- Salimos del modo configuración del router
- Copiamos los cambios en el fichero de inicio
- Volvemos a iniciar la secuencia de carga del (Bootstrap-configuración-IOS)

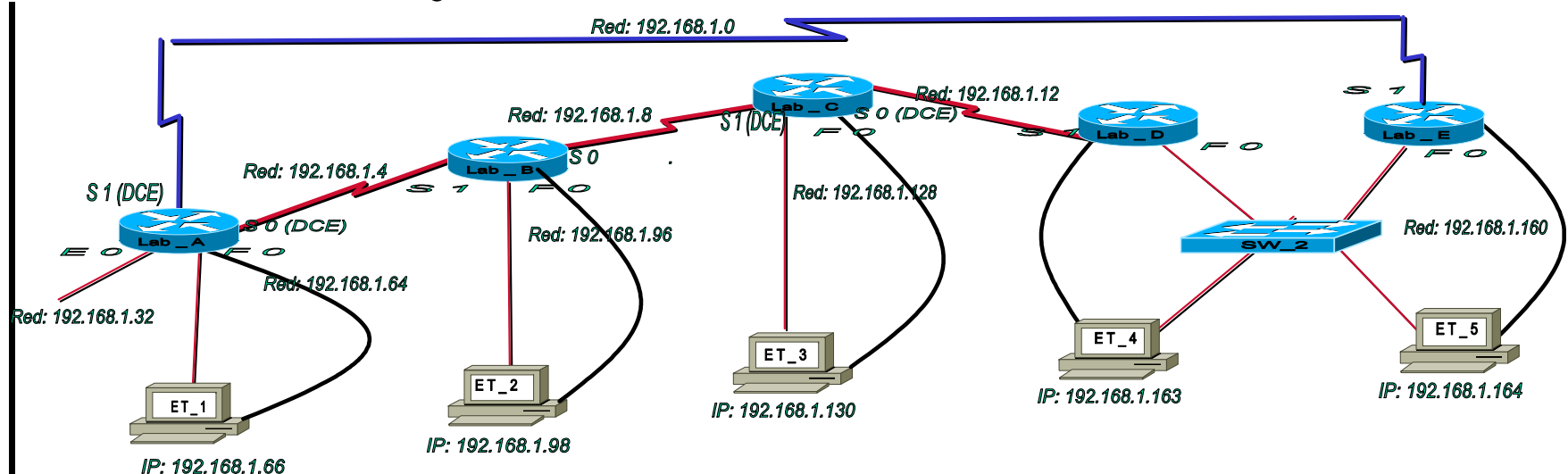
Una vez terminada la práctica borrar la configuración (si se ha guardado en memoria) con erase start y apagar los routers y recogerlo todo.

ANEXOS:

Secuencia de generación de “break” según software y sistema operativo.

Software	Plataforma	S.O.	Secuencia
Hyperterminal	IBM Compatible	Windows 2000	Ctrl-Break
Hyperterminal (version 595160)	IBM Compatible	Windows 95	Ctrl-F6-Break
Kermit	Sun Workstation	UNIX	Ctrl-\l
			Ctrl-\b
MicroPhone Pro	IBM Compatible	Windows	Ctrl-Break
Minicom	IBM Compatible	Linux	Ctrl-a f
ProComm Plus	IBM Compatible	DOS or Windows	Alt-b
Telix	IBM Compatible	DOS	Ctrl-End
Telnet to Cisco	IBM Compatible	N/A	Ctrl-]
Teraterm	IBM Compatible	Windows	Alt-b
Terminal	IBM Compatible	Windows	Break
			Ctrl-Break
Tip	Sun Workstation	UNIX	Ctrl-], then Break or Ctrl-c
			~#
VT 100 Emulation	Data General	N/A	F16
Windows NT	IBM Compatible	Windows	Break-F5
			Shift-F5
			Shift-6 Shift-4 Shift-b (^\$B)
Z-TERMINAL	Mac	Apple	Command-b
N/A	Break-Out Box	N/A	Connect pin 2 (X-mit) to +V for half a second
	Cisco to aux port	N/A	Control-Shft-6, then b
	IBM Compatible	N/A	Ctrl-Break

MAQUETA CON NUEVO DIRECCIONAMIENTO DE SUBREDES



Router Name= Lab_A
 EO = 192.168.1.33
 F0 = 192.168.1.65
 SO = 192.168.1.5
 S1 = 192.168.1.1
 MS(EO)= 255.255.255.224
 MS(F0)= 255.255.255.224
 MS(S0)= 255.255.255.252
 MS(S1)= 255.255.255.252

Router Name= Lab_B
 FO = 192.168.1.97
 SO = 192.168.1.9
 S1 = 192.168.1.6
 MS(FO)= 255.255.255.224
 MS(S0)= 255.255.255.252
 MS(S1)= 255.255.255.252

Router Name= Lab_C
 FO = 192.168.1.129
 SO = 192.168.1.13
 S1 = 192.168.1.10
 MS(FO)= 255.255.255.224
 MS(S0)= 255.255.255.252
 MS(S1)= 255.255.255.252

Router Name= Lab_D
 FO = 192.168.1.161
 S1 = 192.168.1.14
 MS(FO)= 255.255.255.224
 MS(S1)= 255.255.255.252

Router Name= Lab_E
 FO = 192.168.1.162
 S1 = 192.168.1.2
 MS(FO)= 255.255.255.224
 MS(S1)= 255.255.255.252

Name= ET_1
 IP= 192.168.1.66
 MS= 255.255.255.224
 GW = 192.168.1.65

Name= ET_2
 IP= 192.168.1.98
 MS= 255.255.255.224
 GW = 192.168.1.97

Name= ET_3
 IP= 192.168.1.130
 MS= 255.255.255.224
 GW = 192.168.1.129

Name= ET_4
 IP= 192.168.1.163
 MS= 255.255.255.224
 GW = 192.168.1.161

Name= ET_5
 IP= 192.168.1.164
 MS= 255.255.255.224
 GW = 192.168.1.162

PASSWORDS:

CONSOLA: NETACAD

TELNET: CISCO

ENABLE: GENIOS

