

# Introducción al CLI en Routers y Switches CISCO

(Versión 2.3)

Puedes descargar la última versión de este documento de:

[http://blog.unlugarenelmundo.es/?page\\_id=127](http://blog.unlugarenelmundo.es/?page_id=127)

José María Morales Vázquez

[josemaria@morales-vazquez.com](mailto:josemaria@morales-vazquez.com)



Este documento se encuentra bajo una Licencia [Creative Commons Atribución-CompartirIgual 3.0 Unported](https://creativecommons.org/licenses/by-sa/3.0/).

## CONTENIDO

1. INTRODUCCIÓN.....	3
Modos de operación.....	3
Otros trucos útiles en el CLI.....	4
2. CONTRASEÑA DE ACCESO AL MODO PRIVILEGIADO Y OTROS COMANDOS....	4
El comando show.....	4
Historial de comandos.....	4
Contraseña de acceso al modo privilegiado.....	4
El comando no.....	5
El comando do.....	5
Otros comandos básicos.....	5
3. INTERFACES Y LÍNEAS. CONFIGURACIÓN BÁSICA DE UN ROUTER.....	6
Identificación de interfaces y líneas.....	6
Configuración de interfaces ethernet.....	6
Configuración de rutas estáticas.....	7
4. ACCESO REMOTO.....	7
Acceso por telnet.....	8
Acceso por ssh.....	8
5. CONFIGURACIÓN DE UN SERVIDOR DHCP.....	9
6. INTRODUCCIÓN A LOS PROTOCOLOS DE ENCAMINAMIENTO DINÁMICO.....	10
7. RIP, ROUTING INFORMATION PROTOCOL.....	11
8. EIGRP, ENHANCED INTERIOR GATEWAY ROUTING PROTOCOL.....	12
9. OSPF, OPEN SHORTEST PATH FIRST.....	14
10. SWITCHES.....	15
Asignación de una IP al switch para acceder de forma remota.....	16
11. REDES VIRTUALES (VLANs).....	16
Enrutamiento entre VLANs.....	18
12. OTROS COMANDOS.....	19
13. INTRODUCCIÓN A LA SEGURIDAD EN LOS SWITCHES CISCO.....	19
Desconexión de puertos no usados o sospechosos.....	19
DHCP Snooping.....	20
Control de las MAC conectadas a los puertos del Switch.....	21
Storm Control.....	23

## 1. INTRODUCCIÓN

Los routers CISCO son como pequeños ordenadores. Tienen un procesador, se pueden ampliar con nuevos interfaces, su software se actualiza como un verdadero sistema operativo, etc. Una de sus características más importantes son sus cuatro diferentes tipos de memoria:

- **ROM** – Es la memoria que viene de serie y contiene el código indispensable para que el router arranque. No puede modificarse.
- **Flash** – Es la memoria donde se carga el software IOS de CISCO que es el que nos va a permitir configurar y operar el router. Es actualizable y CISCO proporciona nuevas versiones periódicamente corrigiendo errores y proporcionando nuevas funcionalidades.
- **NVRAM** – Es una pequeña memoria donde se guarda la configuración del router que queremos que se cargue inicialmente cada vez que este arranca.
- **RAM** – Es la memoria de trabajo del router donde se cargan las tablas de rutado, las configuraciones que hemos aplicado pero que no hemos salvado, etc. Es la única memoria volátil, es decir, que se pierde cuando se reinicia el router.

La primera vez que vamos a configurar un router nos tenemos que conectar con un cable serie a la entrada de consola y nos saltará un script de configuración inicial que podemos saltarnos (pulsando **Ctrl+c** en cualquier momento o diciendo que no a la primera pregunta). Packet Tracer nos simula este proceso cuando entramos en la pestaña CLI de un router o cuando lo conectamos a un PC mediante un terminal serie al puerto de consola.

Los routers de CISCO suelen llevar también un pequeño servidor web al que podemos conectarnos a través de un navegador para configurarlos. CISCO también proporciona aplicaciones Java para hacer esto, pero la opción más popular y más buscada entre los profesionales es que sepan configurar un router CISCO a través del CLI (Command Line Interface).

### Modos de operación

Los routers CISCO tienen tres modos de operación en su línea de comando:

- **modo normal** de usuario, el inicial con un prompt así >
- **privilegiado** o de administración, con prompt #
- **de configuración**, con la palabra config antes del prompt #
  
- Para pasar de modo normal a privilegiado se usa el comando **enable**
- Para pasar de privilegiado a configuración se usa el comando **configure terminal**
- Para volver atrás un nivel se usa el comando **exit**. **Ctrl+Z** también nos devuelve al modo privilegiado desde el modo de configuración. **disable** también nos devuelve al modo normal desde el modo privilegiado.

## Otros trucos útiles en el CLI

- Para autocompletar comandos se usa el tabulador
- Para pedir ayuda de comandos disponibles u opciones de los mismos se usa el signo ?
- No hace falta escribir los comandos completos. Basta con las letras suficientes para que no haya confusión con otras alternativas. Es válido **ena** por **enable**, **config term** por **configure terminal**, etc.
- **Ctrl+Shift+6** interrumpe la ejecución de un comando que no responde y que se ha quedado pillado.

## 2. CONTRASEÑA DE ACCESO AL MODO PRIVILEGIADO Y OTROS COMANDOS

### El comando show

El comando show es básico para obtener información acerca del router. Iremos viendo otras utilidades pero, por ejemplo, tenemos también las siguientes:

- **show version** muestra la versión de IOS que estamos usando, la información de copyright de CISCO que aparece en el arranque y un resumen de las características e interfaces de nuestro router.
- **show running-config** lista la configuración del router que está actualmente en ejecución (¡en RAM y no en NVRAM!). Es muy útil para guardar “en papel” la configuración de referencia de cada uno de nuestros routers.

### Historial de comandos

Como casi todos los interfaces en línea de comandos, el CLI de CISCO guarda un historial de las últimas órdenes que hemos ejecutado desde la consola y que podemos visualizar con la siguiente orden que se puede ejecutar desde el modo normal o privilegiado:

- **show history**

Por defecto se guardan los últimos 10 comandos ejecutados. Si queremos incrementar ese número lo hacemos desde el modo privilegiado con la siguiente orden:

- **terminal history size 50**

Si quisiéramos desactivar el historial de comandos ejecutamos lo siguiente:

- **terminal history size 0**

### Contraseña de acceso al modo privilegiado

Como hemos visto, al sacar el router de la caja este no está protegido con contraseña. Existen diferentes formas de proteger el acceso al router pero la primera que debemos de usar es la de asegurarnos que nadie entra al modo privilegiado sin una contraseña cifrada. Para ello, desde el modo de configuración tenemos que ejecutar el comando **enable secret** y a continuación la contraseña que deseamos.

Existe otra opción mediante el comando **enable password**, pero en este caso la contraseña se listará en claro al hacer **show running-config**. Podemos cifrar esta contraseña mediante el comando **service password-encryption**, pero aún así se trata de una protección muy débil que puede saltarse mediante herramientas comunes como la que hay en esta web:

<http://www.ibeast.com/content/tools/CiscoPassword/>

Veremos más adelante otras formas de proteger el acceso a nuestro router: con usuario y contraseña, etc.

### El comando no

El comando no antes puesto a otro comando sirve, en las ocasiones en las que esto se puede hacer, para eliminar, desactivar, volver atrás o deshacer el efecto de dicho comando. Por ejemplo, para eliminar la contraseña de acceso al modo privilegiado usamos **no enable secret** o para borrar el nombre que le hemos puesto a nuestro router podemos usar el comando **no hostname**.

### El comando do

Si queremos ejecutar un comando del modo privilegiado desde el modo configuración (algo típico si necesitamos ejecutar algún comando show mientras estamos configurando algo) y no queremos estar cambiando continuamente de modo, podemos hacerlo anteponiendo el comando **do**. Por ejemplo, escribiendo **do sh ip route** desde el modo de configuración nos ejecutaría el comando como si estuviéramos en el modo privilegiado. El único inconveniente de ejecutar comandos con **do** es que no tendremos disponibles ni las funciones de autocompletar ni la ayuda con la tecla ?

### Otros comandos básicos

- **hostname <nombre>** desde el modo de configuración, nos permite cambiar el nombre distintivo del router.
- **banner motd #** También desde el modo de configuración, nos permite personalizar el mensaje de bienvenida que recibimos al conectarnos al router. Podemos escribir lo que queramos usando varias líneas, etc. El mensaje finaliza cuando volvamos a escribir **#** al principio de una línea y pulsemos la tecla **INTRO**. Si lo deseamos podemos usar otro carácter como finalizador cambiándolo en el comando.
- **reload** Desde el modo privilegiado reinicia el router. Los cambios no salvados se pierden.
- **write memory** o sólo **write** desde el modo privilegiado salva la configuración actualmente en ejecución como configuración por defecto.
- **copy running-config startup-config** Idem al anterior.
- **write erase** Limpia toda la configuración del router y lo deja como recién salido de la caja.
- **write startup-config** Idem al anterior.

### 3. INTERFACES Y LÍNEAS. CONFIGURACIÓN BÁSICA DE UN ROUTER

#### Identificación de interfaces y líneas

Los routers CISCO tienen dos tipos de conexiones: interfaces y líneas. Las interfaces son aquellas conexiones que usamos para conectarlos entre sí o a otros equipos para que desempeñen las funciones de rutado propiamente dichas. Las líneas, por otro lado, son conexiones que usamos sólo para configurarlos o manipularlos.

Las interfaces pueden ser serie, ethernet, fast ethernet, giga ethernet, atm, etc. Las líneas pueden ser console, aux o vty.

Las interfaces se identifican por una secuencia de números separados por barras del tipo **0/0/0** donde el primero representa la bahía, el segundo el slot y el tercero el puerto. Si sólo hubiera dos serían slot y puerto y si aparece uno sólo es que se identifica con el puerto sin más (aunque esto sólo suele ocurrir en las líneas y no en los interfaces). Se añade además, al principio de la secuencia, una letra que indica el tipo de interface (**e** por ethernet, **f** por fast ethernet, **g** por gigabit ethernet, **s** por serial, etc.). Por ejemplo **e0/1/0** o **f0/0**, **s0/0/0** o **g7/0**.

Podemos listar los interfaces que tiene nuestra máquina con todos sus detalles mediante el comando **show interfaces** desde el modo privilegiado.

Si queremos información sólo de un determinado interface lo concretamos en el comando. Por ejemplo **show interface f0/1**

Para añadir una dirección IP secundaria a un interface (en Packet Tracer no está implementado esto y no funciona) usamos la siguiente sintaxis:

- **ip address 192.168.10.1 255.255.255.0 secondary**

#### Configuración de interfaces ethernet

La configuración de un interface se realiza en tres pasos, siempre desde el modo de configuración: seleccionar la interface a configurar, asignarle una ip y una máscara a ese interface y, finalmente, activarla. Para las interfaces serie necesitaríamos, además, activar una señal de reloj en uno (y sólo uno) de los extremos de la comunicación, pero esto no vamos a verlo por el momento.

Supongamos que queremos configurar el interfaz fast ethernet identificado como 0/1 con la IP 192.168.0.1 de la subred 192.168.0.0/25. La secuencia de comandos a aplicar (desde el modo de configuración) sería la siguiente:

- **interface f0/0** para seleccionar la configuración del interface
- **ip address 192.168.0.1 255.255.255.128** para asignarle IP y máscara
- **no shutdown** para activarlo
- **exit** para salir del modo de configuración de interface y volver al modo de configuración normal.

Observa que en realidad no existe ningún comando para activar un interface. Sólo existe el comando **shutdown** que lo desactiva. Al usarlo junto con el comando **no** conseguimos el efecto contrario.

Observa también que una vez seleccionado el interface a configurar con el comando **interface f0/1** el prompt vuelve a cambiar y pone **(config-if)#** en lugar del **(config)#** habitual para que distingas que estás configurando un interface de red.

El comando **show ip interface brief** nos muestra un listado resumen de todos los interfaces de nuestro router y el estado en el que se encuentran.

### Configuración de rutas estáticas

Las rutas estáticas se introducen en el router a través de los tres mismos parámetros que usábamos en las ventanas de asistencia del packet tracer. Un ejemplo del comando a usar (desde el modo de configuración) para introducir una ruta estática es el siguiente:

```
■ ip route 192.168.3.0 255.255.255.0 192.168.0.2
```

Donde **192.168.3.0** es la dirección de la red a la que queremos llegar, **255.255.255.0** su máscara y **192.168.0.2** el siguiente salto, es decir, la dirección IP de otro router, conocido por el que estamos configurando, donde hay que entregar los mensajes para encaminarlos hacia la red a la que queremos llegar.

Si necesitamos rutas alternativas, podemos añadir al final otro parámetro con una métrica que simbolice la distancia o preferencia de ruta. Cuanto menor sea el número, mayor preferencia se dará a esa ruta. Por defecto si no ponemos nada se configura como 1:

```
■ ip route 192.168.3.0 255.255.255.0 192.168.0.5 2
```

El comando **show ip route** nos muestra todas las redes que nuestro router conoce, distinguiendo si está conectado directamente a ellas, si tiene alguna ruta estática configurada o cualquier otra condición. Si sólo queremos ver las redes directamente conectadas a nuestro router usaremos el comando **show ip route connected**

## 4. ACCESO REMOTO

Hasta ahora hemos trabajado en modo CLI con nuestros routers CISCO, pero en ningún momento nos hemos planteado como realizamos ese acceso.

¿Cómo accedemos a la pantalla del CLI de un dispositivo CISCO realmente? Tenemos varias formas de hacerlo. Una de ellas, la que deberíamos de usar inicialmente cuando sacamos el dispositivo de su caja, es conectarnos

directamente con un PC mediante un emulador de terminal y a través del puerto de consola que traen todos ellos. Pero una vez instalado, configurado y emplazado en un armario de comunicaciones junto con otras decenas de dispositivos y, a lo mejor, en otra planta u otro edificio diferente al de nuestro lugar de trabajo esta forma ya no resulta cómoda. Afortunadamente podemos realizar un acceso remoto mediante telnet o ssh para lo cual tenemos que realizar una configuración previa.

**NOTA:** Algunos dispositivos CISCO disponen, además, de una dirección IP configurada por defecto que nos permiten, nada más enchufados, conectarnos a ellos a través de un navegador web y realizar una primera configuración de forma gráfica mediante una interfaz web.

### Acceso por telnet

La configuración del acceso mediante telnet es la más simple, pero hemos de recordar que el acceso por telnet se hace siempre “en claro” y cualquiera podría escuchar a través de la red cualquier comando o contraseña que escribamos. Desde el modo de configuración, ejecutamos lo siguiente:

- **line vty 0 4** vamos a configurar hasta cinco accesos simultáneos (desde el 0 al 4) remotos. El prompt cambia a (config-line)#
- **password smr** definimos la contraseña de entrada por telnet como smr
- **login** habilitamos el acceso

Con esto nos deja conectarnos por telnet pero no nos va a dejar entrar al modo privilegiado o al de configuración porque no está protegido con contraseña. Para que se nos permita usar el modo privilegiado en una conexión remota tenemos que protegerlo mediante contraseña como ya sabemos hacer:

- **enable secret arboleda**

**NOTA:** El procedimiento es el mismo para poder acceder a un router o a un switch.

### Acceso por ssh

El acceso por ssh es el más común y altamente recomendado. Todo el tráfico entre nuestro terminal y el dispositivo de CISCO se realiza cifrado y es indescifrable. Todos los routers admiten acceso por ssh pero sólo los switches de gama alta lo soportan. Recuerda, además, que tanto unos como otros deben de tener configurada una dirección IP para que podamos acceder a ellos por ssh. Los comandos necesarios desde el modo de configuración son:

- **hostname mirouter**
- **ip domain-name arboleda.net** Es obligatorio definir un nombre y un nombre de dominio para el dispositivo. Lo hacemos con el comando anterior (que ya conocemos) y con este.
- **crypto key generate rsa** para configurar la fortaleza de la clave de cifrado que usaremos. Se nos pedirá un número que puede ser 512, 1024 o 2048. A



más alto, mayor fortaleza pero también mayor consumo de CPU. 1024 es el valor recomendado.

- **line vty 0 1** Vamos a configurar hasta dos accesos simultáneos por ssh. El prompt cambia a (config-line)#
- **transport input ssh** Para habilitar el acceso por ssh
- **login local** Para habilitar el acceso mediante usuario y contraseña en lugar de sólo con contraseña como hasta ahora.
- **username josemaria privilege 15 password arboleda** donde creamos al usuario josemaria con contraseña arboleda y nivel de privilegios 15

Los niveles de privilegios van entre 0 y 15 y definen los comandos que tendrá permiso para ejecutar el usuario. Un nivel 1 corresponde con el modo normal (prompt >) y un nivel 15 con el de máximos privilegios (prompt #).

Y ya está. Ahora podemos acceder por ssh desde cualquier equipo a este dispositivo.

- **show ip ssh** o **show ssh** nos muestran información acerca del servicio
- **show privilege** nos dice el nivel de privilegio con el que estamos trabajando.

**NOTA:** El procedimiento es el mismo para poder acceder a un router o a un switch salvo que aún no sabemos como configurar una IP en un switch. Eso lo veremos más adelante.

## 5. CONFIGURACIÓN DE UN SERVIDOR DHCP

Los router CISCO, al igual que casi todos los routers, pueden configurarse de forma que además realicen las funciones de servidor dhcp. La forma de configurarlos es muy simple y tiene cuatro pasos: crear un pool de direcciones, asociarlo a una determinada red (indicando la dirección de la misma y su máscara), indicar la dirección del router por defecto que entregarán a sus clientes y, por último, excluir las direcciones que usaremos para asignaciones manuales. Los comandos concretos a usar, desde el modo de configuración, son los siguientes:

- **ip dhcp pool laarboleda** crea un pool de direcciones para administrar por dhcp y le asigna el nombre distintivo **laarboleda**
- **network 192.168.0.0 255.255.255.128** le asigna al pool anterior la red delimitada por la dirección de red **192.168.0.0** con máscara **255.255.255.128**
- **default-router 192.168.0.1** asigna la dirección **192.168.0.1** como la del router por defecto que entregará a los clientes junto con la ip correspondiente,
- **exit** para salir del modo de configuración del dhcp y volver al modo de configuración normal.

El router puede tener más de un interface de red y también más de un pool de direcciones dhcp. El asocia los pools con los interfaces a través de los cuales

entregará las direcciones porque la dirección del interface debe de corresponder con la de la red asociada al pool y, como ya sabemos, cada interface del router debe de pertenecer a una red diferente.

Observa también que, al igual que ocurría con la configuración de los interfaces, el prompt también cambia aquí por **dhcp-config** para advertirnos del modo en el que nos encontramos.

Para excluir un rango de direcciones usamos el siguiente comando:

- **ip dhcp excluded-address 192.168.0.1 192.168.0.20** para excluir las direcciones entre la 192.168.0.1 y la 192.168.0.20. El resto de las direcciones de la red (en este caso desde la 192.168.0.21 hasta la 192.168.126) serían usadas por el servicio dhcp

Podemos inspeccionar la tabla de asignación de direcciones del servicio dhcp mediante el siguiente comando:

- **sh ip dhcp binding**

El servicio se detiene simplemente usando el comando no sobre la primera orden:

- **no ip dhcp pool laarboleada**

Existen muchas más opciones a la hora de configurar el servidor dhcp de un router CISCO: el tiempo de vida de las direcciones entregadas, asignar direcciones fijas por MAC a ciertos equipos, asignar las direcciones de los servidores DNS a los clientes, etc. (aunque algunas de estas opciones no vienen aún implementadas en packet tracer). Para quien quiera información exhaustiva sobre esto tiene la siguiente guía (en inglés) elaborada por CISCO:

[http://www.cisco.com/en/US/docs/ios/12\\_2/ip/configuration/guide/1cfdhcp.html](http://www.cisco.com/en/US/docs/ios/12_2/ip/configuration/guide/1cfdhcp.html)

## 6. INTRODUCCIÓN A LOS PROTOCOLOS DE ENCAMINAMIENTO DINÁMICO

Los routers pueden utilizar dos tipos de modos de funcionamiento a la hora de realizar su trabajo de encaminamiento: usar tablas de encaminamiento estático (lo que hemos visto hasta ahora) o usar protocolos de encaminamiento dinámico.

En redes sencillas en las que sólo existe un único camino para comunicar entre dos puntos los protocolos de encaminamiento dinámico no tienen razón de ser. En las redes complejas en las que podemos tener diferentes rutas entre dos puntos y, además, la topología puede cambiar o existe el riesgo de que ciertos tramos se colapsen en determinadas circunstancias, los protocolos de encaminamiento dinámico ofrecen un mejor resultado.

Se usan dos estrategias en los protocolos dinámicos: las denominadas de vector-distancia y las de estado de enlace. A grandes rasgos, la primera trata el problema como las indicaciones de carretera. Para llegar a un destino sólo tenemos que seguir los carteles pero no tenemos una información total de la ruta: sólo el número de kms. que nos separa de nuestro destino. Si tenemos dos rutas alternativas veremos los kms. que faltan según cojamos una u otra, pero no tendremos casi ninguna otra información. Los protocolos de estado de enlace serían comparables a viajar con un GPS y un mapa de carreteras: tenemos información completa de toda la ruta hasta nuestro destino y podemos tomar decisiones no sólo por el número de kms.

Los tres protocolos de encaminamiento dinámico más utilizados en la actualidad son RIP, EIGRP y OSPF. Todos ellos pueden usarse en routers CISCO. Los dos primeros son de vector-distancia y el tercero de estado de enlace.

### 7. RIP, ROUTING INFORMATION PROTOCOL

Routing Information Protocol. Protocolo de encaminamiento dinámico estándar muy simple. Es un protocolo de los denominados de **vector-distancia** y es muy utilizado en redes pequeñas o medianas.

La configuración de RIP es muy sencilla: el router sólo necesita información de las redes conectadas directamente a él. El resto lo descubrirá por la información que le envíen los demás routers. RIP elige siempre la ruta con mínimo número de saltos, pero si esta se cae y existe otra disponible se recupera y la sustituye (aunque lo hace de forma bastante lenta comparada con otros protocolos). Existen dos versiones de RIP. La segunda, que es la que veremos, es de configuración "sin clase" (classless), y nos permitirá el uso de subnetting y supernetting en nuestras redes sin tener, además, que indicar las máscaras de las mismas.

En los protocolos de vector distancia los routers no tienen una tabla completa de la red, sino sólo de los destinos adyacentes (de forma muy parecida a las rutas estáticas). RIP está restringido a redes con más de 15 saltos entre dos destinos. De forma predeterminada actualiza la información de sus tablas comunicando con los demás routers de la red cada 30 segundos. Esto es así incluso cuando no hay cambios en la red en varios días.

Los routers con RIP en su versión 1 se comunican entre ellos usando broadcast y la dirección especial **255.255.255.255**. En la versión 2, más recomendable, usan multicast y la dirección IP **224.0.0.9** (recuerda: una dirección clase D que están reservadas para usos especiales como este).

Los comandos básicos para configurar un router CISCO con la versión 2 del protocolo RIP son:

- **router rip** desde el modo de configuración activa el modo de configuración del protocolo RIP. El prompt cambia a (config-router).
- **version 2** habilita el uso de la versión 2 del protocolo, que es con la que trabajaremos.

A continuación tenemos que ir agregando una a una todas las redes que tienen conexión directa con el router. Lo hacemos añadiendo sólo las direcciones de red (sin máscara) a través del siguiente comando:

- **network 192.168.20.0**

En redes con subnetting y supernetting complejas tenemos que decirle al router que no queremos que agrupe las rutas de forma automática con el siguiente comando:

- **no auto-summary**

El auto summary de rutas ahorra mucha memoria a los routers pero puede dar lugar a errores en algunas topologías complejas.

Tendremos que ejecutar un comando como el anterior por cada red conectada al router.

Por último, para comprobar el correcto funcionamiento del protocolo podemos activar el modo debug de forma que veremos en la consola todos los mensajes generados. Lo hacemos con el siguiente comando:

- **debug ip rip**

Para desactivar este modo y volver a la operativa normal:

- **undebug all**

Seguramente no podrás escribirlo de una vez porque te verás interrumpido por los mensajes del protocolo. No eches cuenta de ello, escríbelo como si no pasara nada y pulsa intro al final.

Más información y opciones extendidas sobre este protocolo:

[http://www.cisco.com/en/US/docs/ios/12\\_1/iproute/configuration/guide/1cdrrip.html](http://www.cisco.com/en/US/docs/ios/12_1/iproute/configuration/guide/1cdrrip.html)

## 8. EIGRP, ENHANCED INTERIOR GATEWAY ROUTING PROTOCOL

Enhanced Interior Gateway Routing Protocol. Se trata de un protocolo propietario de CISCO. A veces se lo considera un protocolo híbrido que usa **vector-distancia** y **estado de enlace**, pero en realidad es un protocolo de vector-distancia que usa

algunas de las características de los protocolos de estado de enlace y, a la hora de transmitir información a los routers vecinos, no tiene en cuenta sólo el número de saltos para llegar a destino sino que también considera otros parámetros.

EIGRP usa 5 métricas para establecer las rutas óptimas: el ancho de banda, el retraso de la red, la confiabilidad, la carga de la red y el mtu o **unidad máxima de transferencia**, aunque tiene un modo por defecto que sólo toma el ancho de banda y el retraso. Si todos los enlaces tienen el mismo ancho de banda y el mismo retraso, es el número de saltos lo que determina el camino. En este caso se comporta igual que RIP a la hora de escoger el camino (minimizando el número de saltos) pero recompone la ruta mucho más rápido si se cae algún enlace. En una ruta con varios saltos se toma como ancho de banda el del salto que lo tenga menor y como retraso la suma de todos los de la ruta.

Los parámetros pueden ser diferentes en uno y otro extremo de un salto porque se supone la existencia de líneas asimétricas (como el ADSL) con características diferentes en cada extremo. El ancho de banda y retraso que se configura en el interfaz corresponde con los parámetros de salida de la línea. Los de la entrada se configuran en el otro extremo.

EIGRP no envía actualizaciones periódicas y sólo lo hace cuando ha habido algún cambio en la red. Para ello usa la dirección multicast **224.0.0.10**.

Los comandos básicos para configurar un router CISCO con EIGRP son (desde el modo de configuración):

- **router eigrp 1** para activar el protocolo eigrp. El prompt pasa a modo (config-router). El número 1 que aparece en el comando define el llamado "sistema autónomo" y debe de ser el mismo en todos los routers que queremos que intercambien información entre si con este protocolo. Puede variar entre 1 y 65535.
- **network 192.168.1.0** para definir las redes conectadas directamente al router. EIGRP no es tan bueno como RIP detectando las máscaras cuando aplicamos subnetting o supernetting, así que en estos casos hay que especificar la máscara, pero lo hacemos usando lo que se llama "wildcard mask" o máscara de comodines que es el resultado de invertir los valores de la máscara común. Es decir, una máscara 255.255.255.128 quedaría como 0.0.0.127:
- **network 192.168.10.128 0.0.0.127**

Además, en redes con subnetting y supernetting complejas también tenemos que decirle al router que no queremos que agrupe las rutas de forma automática con el siguiente comando:

- **no auto-summary**

Para caracterizar las líneas con determinados anchos de banda y retrasos, tenemos que entrar en el modo de configuración de cada una de ellas. Por ejemplo, si queremos caracterizar la interfaz f0/0 vamos al modo de configuración y ejecutamos lo siguiente:

- **interface f0/0**
- **bandwidth 64000** para configurar el ancho de banda de la línea correspondiente al interface f0/0 en bps (en 64Kbps en el ejemplo)
- **delay 1000** para configurar el retraso de la línea con una cantidad en décimas de microsegundo (en 10000 microsegundos en el ejemplo).

Para que tomen efecto los cambios en estos parámetros hay que bajar la línea (**shutdown**) y volver a levantarla luego (**no shutdown**).

Existen otros tres parámetros más: **load**, **reliability** y **mtu**, pero en el modo por defecto no se tienen en cuenta. Además, podemos asignar más importancia a unos u otros parámetros, pero cuanto más complejo hagamos el cálculo más sufrirá con estos la CPU del router.

Puesto que las líneas pueden ser asimétricas y tener valores diferentes de entrada y de salida (como ocurre con el ADSL) los valores que configuramos en un interface corresponden a los de salida de la línea. Los de entrada se configurarían en el otro extremo.

Diversos comandos para obtener información sobre la configuración y funcionamiento de EIGRP:

- **sh ip eigrp interfaces**
- **sh ip eigrp neighbors**
- **sh ip eigrp topology**
- **sh ip eigrp traffic**

Más información:

[http://www.cisco.com/en/US/docs/ios/12\\_1/iproute/configuration/guide/1cdeigrp.html](http://www.cisco.com/en/US/docs/ios/12_1/iproute/configuration/guide/1cdeigrp.html)

## 9. OSPF, OPEN SHORTEST PATH FIRST

Open Shortest Path First. Es un protocolo estándar de los denominados de **estado de enlace** que nos garantiza el uso del camino más corto entre dos puntos. Se trata, posiblemente, del protocolo más usado en redes grandes y complejas. A grandes rasgos, cada router construye una topología de la red que conoce y la comparte con sus vecinos de forma que entre todos completan un plano total de la red en la que trabajan y de forma que cada uno calcula sus rutas de forma

independiente a los demás.

En su modo completo se trata de un protocolo complejo que permite dividir la red en diferentes áreas y tratarlas de forma diferente. Nosotros veremos aquí el protocolo OSPF de área única.

Se le considera como el sucesor de RIP. Para la comunicación usa las direcciones multicast **224.0.0.5** y **224.0.0.6**. Al igual que ocurrí ya con EIGRP, sólo envían nuevas comunicaciones cuando detectan cambios en la topología de la red.

Los principales problemas de OSPF (y de todos los protocolos de estado de enlace en general) son los siguientes:

- Necesitamos routers más potentes, con más memoria para almacenar los mapas de red y más CPU para hacer cálculos con un algoritmo más complejo.
- La información que se transmite es mayor, por lo tanto existe más riesgo de que saturen la red con sus mensajes en el momento inicial del arranque (cuando empiezan a cambiar información entre ellos para construir los mapas de red) o en redes inestables en las que tenemos muchos y frecuentes cambios.

La configuración básica de un router con OSPF se realiza con los siguientes comandos:

- **router ospf 1** para habilitar el protocolo OSPF en el router
- **network 192.168.1.0 0.0.0.255 area 0** para dar de alta la red dada, con máscara en formato de comodines y especificando que dicha red pertenece al área 0.

La mayoría de los comandos disponibles para controlar el estado de ospf no están implementados en packet tracer. El más útil de los que disponemos es este:

- **sh ip ospf neighbor**

Más información:

[http://www.cisco.com/en/US/docs/ios/12\\_1/iproute/configuration/guide/1cdospf.html](http://www.cisco.com/en/US/docs/ios/12_1/iproute/configuration/guide/1cdospf.html)

## 10. SWITCHES

Los switches de CISCO usan el mismo sistema de configuración en línea de comando que los routers y, por tanto, muchas de las órdenes que hemos visto hasta ahora nos sirven también para los switches. La forma de entrar en modo privilegiado o de configuración, la ayuda, los comandos para salvar la configuración, etc.

## Asignación de una IP al switch para acceder de forma remota

El primer paso para realizar un acceso remoto es que el dispositivo cuente con una dirección IP configurada. En los routers ya sabemos hacerlo. Para los switches el procedimiento de configuración de una IP es ligeramente diferente. Desde el modo de configuración ejecutamos lo siguiente:

- **int vlan 1** entramos a configurar la vlan 1. Un switch de CISCO puede tener hasta 1005 VLANs diferentes, pero la número 1 es especial para administración (porque la ip que asignemos será "escuchada" por cualquiera de sus puertos) y entre la 1002 y la 1005 están reservadas. Más adelante veremos en detalle que son las VLAN y para que otras cosas sirven. El prompt cambia al modo (config-vlan)#
- **ip address 192.168.1.2 255.255.255.0** le asignamos la ip y máscara especificada
- **no sh** para activar el interface

Si queremos acceder a él desde una red diferente a la que se encuentra instalado tenemos, además, que definirle un router por defecto:

- **ip default-gateway 192.168.1.1**

## 11. REDES VIRTUALES (VLANs)

Las Redes Virtuales constituyen una técnica útil para dividir una red en diferentes subredes separadas entre si sin necesidad de usar routers y permitiendo una separación funcional de los equipos en diferentes grupos de trabajo sin importar la ubicación donde se encuentran conectados. No se trata de una tecnología propietaria de CISCO y puede implementarse igualmente con switches de otros fabricantes.

Como hemos dicho antes, los switches CISCO permiten hasta 1005 VLANs de las cuales la 1 es especial y las existentes entre la 1002 y la 1005 están reservadas. Podemos ver información de las VLAN's activas en nuestro switch con el comando siguiente:

- **show vlan brief**

Crear una VLAN es tan fácil como ejecutar lo siguiente (desde el modo de configuración):

- **vlan 2** para crear una vlan con el identificador 2. El prompt cambia a (config-vlan)#
- **name estudiantes** para asignarle el nombre estudiantes (esto es opcional y no indispensable para que funcione. Si no se asigna nombre se pone uno de forma automática)
- **int vlan 2** activa la vlan y la pone como activa.

Para asignar un puerto del switch, por ejemplo el f0/10, a una determinada VLAN, la 2 en este ejemplo:



- **int f0/10**
- **switchport mode access**
- **switchport access vlan 2**

Para desasociar un puerto de la VLAN usamos el comando no:

- **int f0/10**
- **no switchport access vlan**

Una VLAN se elimina completamente con todos sus puertos asociados también con el comando no:

- **no vlan 2**

Si no queremos eliminarla sino desactivarla temporalmente:

- **int vlan 2**
- **shutdown**

Para ver toda la información asociada a una VLAN, por ejemplo la 10:

- **show vlan id 10**

Cada puerto de un switch admite estar asociado a una única VLAN. Entonces ¿como enlazamos diferentes switches para que comuniquen entre si y transmitan la información de todas las VLAN de nuestra red? Pues lo hacemos definiendo los enlaces entre switches como troncales. Un enlace troncal transmitirá el tráfico de todas las VLANs disponibles. A su vez, tenemos que asociarle un identificador a los enlaces troncales que, en el ejemplo siguiente, es el 99:

- **vlan 99**
- **name troncal**
- **int vlan 99** para crear y activar la vlan 99 que será la que usemos para los enlaces troncales. Si tenemos varios enlaces troncales, esto sólo hay que hacerlo una vez por cada switch.
- **int f0/5**
- **switchport mode trunk**
- **switchport trunk native vlan 99**

Para mostrar la información de todos los enlaces troncales de un switch:

- **show interfaces trunk**

Para comprobar la configuración de un enlace definido como troncal:

- **show interfaces f0/5 switchport**

Los enlaces troncales se eliminan también usando el comando no.

## Enrutamiento entre VLANs

La comunicación entre las diferentes VLANs se puede hacer de varias formas. Una de ellas consiste en usar switches de capa 3 que incluyen funciones de enrutamiento básicas (un switch puede perfectamente permitir la configuración de vlans sin necesidad de que incorpore funciones de capa 3). La segunda forma consiste en usar un router común conectado a uno de los switches que realice las funciones de enrutado.

Para configurar un router de capa 3 de forma que realice funciones de rutado se procede como sigue:

- La configuración de vlans es exactamente igual que como hemos visto hasta ahora salvo que en los switches de capa 3 al configurar un interface como troncal debemos usar un comando adicional. Imagina que vas a configurar como troncal el interface f0/10. La vlan troncal ya está definida como la vlan 25. Tienes que proceder como sigue:

```
int f0/10
```

```
switchport trunk encapsulation dot1q
```

```
switchport mode trunk
```

```
switchport trunk native vlan 25
```

- Asignamos una ip que actuara como la ip del router por defecto a cada vlan. **Si tenemos varios switches conectados mediante troncales este paso sólo tienes que hacerlo en uno de ellos.** Imagina que tenemos dos vlans (la 10 y la 20). La vlan 10 usa la red 192.168.10.0/ 24 y queremos que la ip del router por defecto sea la 192.168.10.1. Para la vlan 20 asignamos la red 192.168.20.0/24 y la ip 192.168.20.1 como router por defecto.

```
int vlan 10
```

```
ip add 192.168.10.1 255.255.255.0
```

```
int vlan 20
```

```
ip add 192.168.20.1 255.255.255.0
```

- Activamos las características de rutado en el switch en el que hemos configurado las ip en las vlans:

```
ip routing
```

- Podemos hacer pruebas de que el rutado funciona correctamente usando el comando ping (similar a como lo usamos en cualquier PC) o pidiendo al switch que nos enseñe su tabla de rutas de igual forma a como lo haríamos en un router:

```
show ip route
```

En el siguiente enlace, en inglés, te explican algo más acerca de las funciones de rutado en vlans con switches de capa 3:

[http://www.cisco.com/en/US/tech/tk389/tk815/technologies\\_configuration\\_exam ple09186a008019e74e.shtml](http://www.cisco.com/en/US/tech/tk389/tk815/technologies_configuration_exam ple09186a008019e74e.shtml)

## 12. OTROS COMANDOS

Por defecto, y al revés de lo que ocurre con los routers, todos los interfaces de un switch vienen activos. Si queremos deshabilitar uno de ellos (por ejemplo porque vemos una actividad de tráfico sospechoso) ejecutamos lo siguiente desde el modo de configuración:

- **int f0/12**
- **sh** para deshabilitarlo

El comando que nos permite ver las direcciones MAC asociadas a cada puerto del switch es este:

- **show mac-address-table** para visualizar la tabla de direcciones MAC asociadas a cada puerto del switch.

## 13. INTRODUCCIÓN A LA SEGURIDAD EN LOS SWITCHES CISCO

Los switches son elementos centrales de la configuración de nuestras redes. Todo el tráfico pasa por ellos y todos los equipos deben, de una forma directa o indirecta, estar conectados a un switch para tener acceso a nuestra red. Introducir mecanismos de seguridad en los mismos es, por tanto, una buena forma de controlar de forma preventiva algunos problemas importantes de seguridad. En este documento veremos algunas de estas medidas y su implementación en switches CISCO.

### Desconexión de puertos no usados o sospechosos

Los puertos no usados de nuestros switches o sospechosos de tener problemas deberían desconectarse de la red. Por defecto todos los puertos de los switches aparecen conectados pero su desconexión es bien sencilla. En las siguientes líneas desconectamos el puerto f0/4

```
ena
config term
int f0/4
shutdown
```

También es posible seleccionar un rango completo de puertos para aplicar una misma configuración sobre todos ellos. Los siguientes comandos desactivan todos los puertos entre el f0/5 y el f0/10

```
ena
```

```
config term
int range f0/5-10
shutdown
```

## DHCP Snooping

EL DHCP Snooping comprende un conjunto de técnicas encaminadas a hacer más seguro el funcionamiento del protocolo DHCP. Ya vimos en su momento que se trataba de un protocolo muy cómodo en redes grandes pero con evidentes problemas de seguridad.

CISCO implementa algunas técnicas encaminadas a este fin. La primera de ellas nos permite prohibir el tráfico de un servidor DHCP desde todos los puertos del switch salvo de los que nosotros autorizemos. Para decir, por ejemplo, que nuestro servidor DHCP está en el puerto f0/2, ejecutaríamos lo siguiente:

```
ena
config term
ip dhcp snooping
int f0/2
ip dhcp snooping trust
```

Para retirar la autorización del interfaz usaríamos el comando no:

```
int f0/2
no ip dhcp snooping trust
```

Si nuestro switch detectara un servidor DHCP conectado a cualquier otro puerto lo desconectaría de forma automática.

El segundo mecanismo de seguridad nos permite limitar el número de peticiones a un servidor DHCP que se realiza desde determinados puertos. Esto nos ayuda a mitigar que alguien trate de saturar un servidor DHCP cambiando de MAC y enviando peticiones al servidor. En el siguiente ejemplo se limitan a 5 paquetes por segundo las peticiones que pueden realizarse desde los puertos 10 al 24:

```
ena
config term
ip dhcp snooping
int range f0/10-24
ip dhcp snooping limit rate 5
```

El siguiente comando mostrará información sobre la configuración de DHCP snooping en nuestro switch (puertos habilitados, velocidades máximas permitidas, etc.):

```
show ip dhcp snooping
```

## Control de las MAC conectadas a los puertos del Switch

Todos sabemos que la principal diferencia del switch frente al hub es que el primero “memoriza” los puertos a los que están conectados los distintos equipos de forma que la comunicación entre un equipo y otro sólo se envía por los segmentos de red adecuados. Esto mejora la calidad de las comunicaciones en nuestra red local (tenemos un mayor ancho de banda disponible) y mejora la seguridad en la red (alguien con un sniffer lo tiene más difícil para leer la información que no va destinada a él.)

Los switches realizan esta función de forma simple almacenando en su memoria unas tablas con las direcciones MAC de los equipos y los puertos a los que están conectados los mismos. Estas tablas deben de tener capacidad para guardar más de una MAC por puerto (podemos tener un switch de 96 puertos conectado en cascada al puerto de otro switch) y, lógicamente, tienen un tamaño finito.

Si un switch recibe un mensaje para un equipo cuya MAC no tiene registrado en sus tablas envía el mensaje a todos sus puertos como si fuese un hub. Cuando el equipo responde registra el puerto desde el que lo ha hecho y, a partir de ese momento, ya le enviará siempre los mensajes directamente.

Pero ¿cómo se comporta un switch si estas tablas se llenan del todo y tiene que enviarle un mensaje a un equipo cuya MAC no está incluida en ellas y por tanto no sabe en que puerto está conectado? Difundiendo el mensaje a través de todos sus puertos como si se tratase de un hub. El problema ocurre cuando este equipo contesta. Puesto que el switch no tiene espacio no incluirá el puerto desde el que lo hace y seguirá siempre enviándole los mensajes como si se tratase de un hub.

Un ataque de saturación o inundación de direcciones MAC (MAC flooding o ARP flooding) está encaminado a esto: inundar un switch con peticiones de diferentes direcciones MAC ficticias para que el switch sature sus tablas y comience a comportarse como un hub con mensajes legítimos de forma que podamos capturarlos mediante el uso de un sniffer. CISCO incluye funcionalidades que nos permiten limitar el número de MAC diferentes que pueden conectarse en los puertos de sus switches de forma que nos permiten solucionar este problema. Además, también evita los ataques de saturación a un servidor DHCP.

Si queremos limitar el puerto de un switch de forma que por este sólo pueda conectarse un equipo con una MAC conocida ejecutaríamos lo siguiente:

```
ena  
config term  
int f0/5  
switchport mode trunk  
switchport port-security  
switchport port-security mac-address 0016.E650.45E2
```

De esta forma, el único equipo legítimo reconocido para estar conectado al puerto f0/ sería el que tiene la MAC indicada. Fíjate que la forma de escribir la

MAC es ligeramente diferente a como estamos acostumbrados. En lugar de ponerla como **00:16:E6:50:45:E2** lo hacemos **0016.E650.45E2**

Si el switch detectara un equipo con una MAC diferente conectada a ese puerto desconectaría el puerto de forma automática.

Si queremos que en lugar de desactivar el puerto simplemente no permita el tráfico de equipos con una MAC diferente ejecutaríamos también lo siguiente:

**switchport port-security violation restrict**

Y si queremos volver al modo anterior:

**switchport port-security violation shutdown**

En algunas ocasiones no podemos limitar a una única MAC las reconocibles en determinados puertos, pero queremos poner un máximo permitido. Tampoco queremos o nos es posible escribir a mano las diferentes direcciones MAC que vamos a permitir. En este caso podemos limitar el número de MAC y decirle además al switch que "aprenda" cuales serán estas de forma automática. Esto lo hacemos con los siguientes comandos:

```
ena
config term
int 0/1
switchport mode access
switchport port-security
switchport port-security maximum 30
switchport port-security mac-address sticky
```

El puerto f0/1 de nuestro switch admitirá, como máximo, 30 direcciones diferentes que el irá aprendiendo de forma automática (las 30 primeras que detecte). A partir de ahí, si existe un intento de comunicación por parte de alguna otra dirección, desactivará el puerto. El modo de respuesta lo podemos modificar también con el comando visto antes (**switchport port-security violation restrict**).

Por último, los comandos que nos permiten ver la configuración que hemos realizado de estas medidas son los siguientes:

```
show port-security
show port-security int f0/5
```

Si, además, queremos ver las direcciones MAC autorizadas para cada puerto ejecutaríamos esto:

```
show port-security address
```

Una variante de este ataque se denomina ARP spoofing o envenenamiento de ARP. En este caso no hace falta llenar la tabla ARP del switch sino que se lo "engaña" haciéndole pensar que un equipo está en un puerto diferente. El switch manda los mensajes a este puerto y el equipo que realiza el engaño los lee y

luego los remite a la máquina auténtica para que esta no sospeche. Si esto mismo se hace simultáneamente con dos equipos se “espía” todas las conversaciones entre ambos constituyendo lo que se conoce como ataque de hombre en el medio (o Man in the Middle). Existen muchas herramientas para hacer esto de forma automática pero tal vez la más popular y fácil de usar sea ettercap, disponible tanto para plataformas Linux como Windows.

### Storm Control

A veces, y generalmente por averías hardware o software, algunas tarjetas de red crean una cantidad de tráfico tan grande que saturan al switch al que están conectadas, deteriorando las comunicaciones y llegando incluso al punto de hacer caer total o parcialmente la red. Los switches CISCO tienen la posibilidad de controlar esto mediante un subconjunto de comandos.

```
ena
config term
int f0/1
storm-control broadcast level 20
```

En el ejemplo anterior habilitamos el storm control para el puerto 1 de nuestro switch de forma que el tráfico broadcast del mismo no supere el 20% del ancho de banda nominal del mismo.

Podemos controlar de igual forma el tráfico unicast y el multicast con, por ejemplo, las siguientes instrucciones:

```
storm-control multicast level 50
storm-control unicast level 90
```

No debemos de olvidar cuando configuramos estos parámetros que los tres tipos de tráfico son necesarios y que, en particular, el multicast se usa por todos los protocolos de rutado dinámico (entre otros) y el unicast constituiría el tráfico común de un equipo.

Si quisiéramos bloquear totalmente el tráfico broadcast de un equipo podríamos poner lo siguiente:

```
storm-control broadcast level 0
```

El tráfico multicast y el unicast puede bloquearse también, además de con el comando anterior, con el siguiente:

```
switchport block unicast
switchport block multicast
```

Y si por defecto quisiéramos deshabilitar el puerto cuando se pase del nivel indicado en el comando:

```
storm-control action shutdown
```