GUÍA CUCA

COMANDOS más USUALES de Configuración y Administración (routers y switches) para el CCNA

Una copia gratuita de este documento puede obtenerse en la sgte. Dirección:

https://stoneheads.wordpress.com/2016/11/22/guia-comandos-cisco-routing-switching

o contactando directamente con el autor del mismo en el correo indicado.

Autor: Carlos Felipe Alvarez González

carlos_felipe@nodo50.org

Fecha: 22 de Noviembre de 2016

Título: guia_cuca

Versión: 2.0

Condiciones de uso

¡Advertencia!

Esta guía es simplemente un resumen de los comandos más habituales de configuración de los routers y switches CISCO. Está pensada como referencia práctica y como apoyo para la preparación del CCNA. Incorpora muy poco de teoría , por lo que **no sustituye el conocimiento teórico de los temas tratados**. Se supone al usuario pues, una base suficiente de los temas que se tratan.

No es ni completa ni exhaustiva, ya que ni se incluyen todos los comandos, ni se ven en profundidad los que aparecen, sólo las opciones más comunes.

Aunque se ha hecho un esfuerzo para que sea lo más precisa posible, **no se da ninguna garantía de que no contenga omisiones, gazapos y errores** (algunos incluso graves) por lo que se declina cualquier responsabilidad en cualquier daño o perjuicio que la utilización de esta guía pueda provocar ya sea en la configuración de equipos o realización de pruebas o exámenes o cualquier otro uso. Vamos, que si sigue esta guía y la caga, las reclamaciones, al maestro armero.

Su uso y abuso puede provocar y provocará seguramente prepotencia y alopecia.

Indice

Conexión

Consola: Cable azul RS232 - COM. También hay con adaptador USB (se necesitan los drivers).

Nos conectamos al puerto de consola usando un cable especial, que se conecta al puerto COM1 del ordenador o a un USB con un adaptador.

Usamos un programa como TeraTerm, Putty o SecureCRT (Linux). Los parámetos de configuración son:

- Puerto COM adecuado (mirar en la conf. hardware del equipo).
- 9600 baudios.
- 8 bits de datos.
- Sin paridad.
- 1 bit de parada.
- Sin control de flujo.

Teclas de Función

?	Muestra los comandos y las opciones que tenemos para completarlos.
CTRL+Z	Vuelve directamente al modo privilegiado.
CTRL+B	Ini linea.
CTRL+E	Fin linea.
CTRL+MAY+6	Aborta el comando en ejecución.
CTRL+R	Vuelve a mostrar la línea de comandos cuando nos interrumpe un mensaje de error.
CTRL+C	Cancela el comando que estás escribiendo.

Modos

Al iniciar, si no hay conf. en la NVRAM, entra en **modo Setup** (te pide establecer toda la conf.l ini.) -> decir que NO.

Pasa al **User EXEC Mode**. Modo usuario. Prompt: *router>*

enable: **Pasa al Privileged EXEC Mode**. (como root) Prompt: *router#*

disable: Sale.

end: Vuelve directamente al modo EXEC.

Vista de configuraciones, comandos de conectividad, etc...

Global Conf. Mode. Prompt: (config)#

```
router#configure terminal
router#conf t
```

router(config)#exit	Sale del modo en que estemos y vuelve al anterior, p.e. modo conf. interfaz → modo conf. global.

Conf. Globales y acceso a las configuraciones de las terminales, interfaces, etc...

Con un **no** delante del comando, lo anula o deshace lo hecho.

Con un **do** permite ejecutar un comando aunque no esté en el modo adecuado.

#undebug all	Quita todos los debug que tengamos puestos

Acceso

Clave de root:

(config)#enable secret miclave

La quita:

(config)#no enable secret

Con password usa una encriptación propia de Cisco. Con secret, usa md5. Sólo puede tener una u otra.

Nombre del equipo:

(config)#hostname Liuva	Max. 64 car. Distingue MAY y min (a-z|0-9|-_\)
(config)#no hostname	Quita el nombre.

Mensaje de entrada:

Se pone siempre como medida de protección legal.

(config)#banner motd 'ACCESO RESTRINGIDO'

Consola.

Conexión mediante cable. Acceso físico al equipo.

(config)#line console 0

Evita que nos interrumpan los mensajes de consola:

(config-line)#logging synchronous

Terminales.

Acceso remoto (telnet/ssh). Podemos configurar todos los que hay de una tacada:

(config)#line vty 0 15

Ponemos la clave para acceder al terminal y habilitamos el acceso con login (podemos hacer lo mismo para la consola. En los terminales, si no ponemos clave no se puede entrar, en consola, sí):

(config-line)#password miclave
(config-line)#login

En algunos IOS se puede usar secret en consola y vty.

Muestra las sesiones de Telnet activas abiertas desde el equipo y los usuarios conectados desde fuera:

#show sessions

#show users

Para ver los usuarios creados localmente, hay que hacer un **show run**

Desactiva una sesión indeseada:

#clear line vty 5

Conf. claves y loggin

Encriptamos las contraseñas actuales y futuras:

(config)#service password-encryption

Crea un usuario con nombre en la BD local de usuarios:

(config)#username usuario secret clave

Lo quita:

(config)#no username usuario

Long. min. de las claves (passwords, no secret):

(config)#security passwords min-length 15

Bloqueamos los intentos de acceso 120s si hay 3 intentos fallidos en 60s:

(config)#login block-for 120 attempts 3 within 60

Desconectamos al usuario si está inactivo >10m (para consola y vty):

(config-vty)#exec-timeout 10

Reloj (NTP):

(config)#clock set 19:50:00 25 June 2016

#show clock	Con la opción **detail** muestra la fuente.

NTP

Para sincronizar los relojes por red:

(config)#ntp master 1	Fuente local, prioridad 1 (por defecto vale 8).
(config)#ntp server 10.1.1.1	Fuente serv. internet.
#show ntp associations	IP , stratum y refs. de los hosts conectados a nuestro equipo.
#show ntp status	Info. del estado NTP. (nuestro stratum y referencia, o sea, el servidor al que estamos conectados)

Logs

Si nos conectamos por terminal, no veremos los mensajes de log de consola. Para verlos y quitarlos hacemos:

```
#terminal monitor

#terminal no monitor
```

Muestra en los logs el t. que lleva funcionando el disp. o el t. del evento (tiene que estar conf. el reloj)

```
(config)#service timestamps log {uptime | datetime}  msec
```

#show logging	Muestra las configuraciones de los logs y los logs del buffer. Admite pipelines para filtrar la salida.
(config)#logging console	Manda los logs a consola (por defecto)
(config)#logging buffered	Manda los logs al buffer

Para enviar logs a un servidor:

(config)# logging 192.168.1.3	Def. la IP del servidor Syslog.
(config)# logging trap 4	Def. el nivel max. de log (aquí manda logs de nivel 0, más grave, hasta 4). Podemos poner el nombre en vez del nº.
(config)# logging source-interface g0/0	Opcional. Los logs. llevarán la IP del int. dado, independientemente de por qué int. salgan.

SSH

#show ip ssh	Ver caract. Ssh
#show ssh	Ver usuarios conectados.

Si nos muestra la versión 1.99 podemos usar ssh2:

```
(config)#ip ssh version 2
```

(config)#ip domain-name cisco.com	Dominio que tenga registrado la org.
(config)#crypto key generate rsa	Metemos 1024 como valor.

Creamos un usuario con el nivel max. de privilegios (administrador):

```
(config)#username netadmin privilege 15 secret clave
```

Hacemos que el acceso por vty sea por ssh:

(config-vty)#login local	Valida el usuario contra la tabla local de usuarios (username). Obliga a identificarse al usuario. Si ya habíamos habilitado el acceso normal, primero hay que poner **no login**.

```
(config-vty)#transport input ssh
```

Cambia el timeout y el n.º de intentos de login por ssh:

```
(config)#ip ssh time-out 75
(config)#ip ssh authentication-retries 2
```

Desconecta una sesión ssh por n.º de sesión o terminal:

```
#disconnect ssh 5
#disconnect ssh vty 3
```

Recuperación de contraseñas:

Para recuperar la contraseña en un switch, nos conectamos con el cable de consola y hacemos como en los routers caseros (pulsar reset un rato). Si el modelo no lo permite o se trata de un router, hacemos:

1-Entrar en el ROMON mode.

Esto se hace accediendo dir. con consola y pulsando una secuencia de escape mientras el router arranca (en Putty es Ctrl+Break) o quitando la flash antes de encenderlo.

2-Cambiar el registro de configuración a 0x2142 para que ignore la startup config. y reiniciar el router

>confreg 0x2142	Ignora los contenidos en la NVRAM.
>reset	

3-Hacer los cambios necesarios en la startup config original

#copy startup-config running-config	Esto carga la conf. de arranque. Inicialmente la conf. de ejecución está vacía. ¡Ojo! No hacerlo al revés que nos cargamos la conf. almacenada.
(config)# enable secret cisco	Cambiamos la clave.
(config)# config-register 0x2102	Volvemos a poner el reg. de conf. normalmente.

4-Salvamos la nueva configuración

 # copy running-config startup-config

#reload

Conectividad

Resolución nombres de host

Muestra los nombres de host conf. y sus direcciones:

#show hosts

(config)#ip host Servidor_Web 192.168.1.1

Deshabilita la resolución de nombres en línea de comandos:

(config)#no ip domain-lookup

Ping y Traceroute

#ping 192.168.0.1 **!** Ok **.** No se recibió o t. agotado **U** Inalcanzable

#traceroute 192.168.0.1
* No hay respuesta
"time exceeded" Recibido por algún router, pero descartado.
"destination unreachable" Idem, pero no se pudo reenviar.

Ping y Traceroute extendidos:

#ping Nos va preguntando todas las opciones.

#traceroute

Tablas MAC y de enrutamiento:

#show mac-address-table	Para switches
#show arp	Para routers
#show ip route	Para routers

CDP.

Muestra info. de los dispositivos conect. dir. (sólo para disp. Cisco)

#show cdp neighbors

#show cdp neighbors detail	Incluye más detalles, como la IP.
#show cdp interfaces	Muestra los interfaces conf. con CDP
(config)#cdp run	Lo habilita globalmente.
(config)#no cdp run	Lo deshabilita globalmente.
(config-if)#cdp enable	Idem localmente, en cada interfaz.

LLDP

Idem con LLDP (standard)

(config)#lldp run	Lo habilita globalmente.
(config-if)#lldp transmit	Habilita transmisión en un interfaz.
(config-if)#lldp receive	Idem recepción.
#show lldp neighbors	Muestra info. de los dispositivos conect. Dir.
#show lldp neighbors detail	Incluye más detalles, como la IP.

Configuración arranque/ejecución

Muestra la conf. actual:

#show running-config
#show run

Muestra la conf. que se guarda en la NVRAM. (La que se aplicará en el arranque):

#show startup-config

Muestra el contenido de la flash, incluyendo la memoria total y la disponible:

#show flash:

Muestra el historial de comandos:

#show history

Pipelines:

#show run | include nome	Líneas con la palabra nome.
#show run | section nome	La sección desde nome.
#show run | begin nome	Desde nome hasta el final.
#show run | exclude nome	Líneas sin la palabra nome.

Copias respaldo.

Copia la configuración actual en la de inicio:

#copy running-config startup-config

Copia la configuración inicial en la de flash:

#copy startup-config flash:

Copia la conf. a un servidor TFTP y a la inversa

#copy running-config tftp:

#copy tftp: running-config

Copia a un llavero

#copy running-config usbflash0:/copia1
(La / sólo se pone si a continuación va un nombre de archivo)

Copia un archivo de la flash a un TFTP (nos pedirá los datos)

#copy flash: tftp:

Reset

Borra la conf. de inicio. ¡Ojo! No hacer a no ser que tengamos acceso físico al equipo:

#erase startup-config

Reiniciar el equipo. Cambia la conf. de ejecución por la de inicio (si la hemos borrado, usa la de fábrica)

#reload

Configuración del sistema

#show version

Version 15,2(4) M1	Versión del IOS
...	
ROM System Bootstrap, version 15.0(r1) M15	Versión del Bootstrap
…	
System image file is "flash0:……..bin"	Imagen del IOS. Un K9 en el nombre, indica que soporta criptografía.
…	
Cisco CISCO1941/K9	CPU (modelo) del router
with 44444/7784 Kbytes of memory	RAM + RAM paquetes = Total RAM
…	
2 Gigabit Ethernet interfaces	Interfaces
2 Serial …	
…	
255 Kbytes of non-volatile conf. memory	NVRAM
250880 Kbytes of ATA System CompactFlash	Flash
…	
Configuration register is 0x2102	Predet. Indica cómo se arranca. 2142 Ignora el contenido de la NVRAM. 2120 Arranca en modo ROM.

#show file systems Muestra los sistemas de archivos. * Predeterminado. # De arranque.

#cd nvram: Entra en el sistema de archivos dado.

#dir Listado del directorio.

#pwd Muestra el dir. actual.

#rename flash:config.txt flash:config.old Cambia el nombre de un fichero.

Si queremos cargar otra imagen del IOS, podemos copiarla en la flash y borrar la antigua. También podemos tener varias y escoger con cuál arrancar. En este caso arrancaría con la de la flash y en caso de que algo fuera mal, intentaría arrancar con la del tftp y luego la de la ROM:

(config)#boot system flash://nombre-imagen

(config)#boot system tftp://nombre-imagen

(config)#boot system rom

Licencias

#show license [feature] Muestra las licencias, tipo y n.º y estado y opcionalmente también las de los paquetes tecnológicos.

#show license udi Muestra el Unique Device Id.

#license install flash://fic_licencia Instala una licencia.
#reload

Acepta el EULA para una lic. de evaluación.

(config)#license accept end user agreement

(config)#license boot module c1900 technology-package ipbasek9

#reload

Salva y recupera un archivo de licencia.

#license save flash://fic_licencia

#license install

Elimina una licencia.

(config)#license boot module c1900 technology-package seck9 disable

#reload
…

#license clear seck9

(config)#no license boot module c1900 technology-package seck9

#reload

Configuración interfaces

Mostrar conf.

#show ip interface brief	Resumen de los interfaces.
#show interface f0/0	Si el interfaz está "admin. down", es que se ha hecho un shutdown. Si lo levantamos y aparece "line protocol down", probablemente exista un problema de capa física.
#show controllers interface s0/0/0	Muestra info. carac. hardware interfaz

Configurar

Dar IP a un interfaz y descripción:

```
(config)#interface g0/0
(config-if)#ip address 192.168.10.2 255.255.255.0
(config-if)#description 'enlace a LAN1'
(config-if)#no shutdown
```

Para configurar **varios** interfaces a la vez (p.e apagar todos los interfaces de 1 a 5):

```
(config)#interface range f0/1 – 5

(config-if-range)#shutdown
```

Podemos definir el rango así también:

```
(config)#interface range f0/1 – 5 , g0/1 - 2
```

IPv6

Necesario SIEMPRE para trabajar con IPv6:

```
(config)#ipv6 unicast-routing
```

Direcciones en los interfaces:

```
(config-if)#ipv6 address 2001:DB8:ACAD:1::1/64
```

Hay que poner también el Link-local. Se suele poner el mismo en todos los interfaces de un router. Este es el gateway que hay que poner en los hosts.

```
(config-if)#ipv6 address FE80::1 link-local
```

Resto, igual que IPv4, pero poniendo **ipv6** en los comandos.

Switches: Acceso remoto

Dar una IP a un switch, para acceder remotamente, p.e.

```
(config)#interface vlan1
(config-if)#ip address 192.168.10.2 255.255.255.0
(config-if)#no shutdown

(config)#ip default-gateway 192.168.10.1
```

¡Ojo! Sólo puede haber un gateway en un switch y sólo una de las vlan puede tener IP (vlan administrativa)

Switches: Hardware

Hacer que el switch detecte auto. el tipo de cable (podemos usar cualquier cable):

(config-if)#mdix auto

Modo duplex:

(config-if)#duplex { auto | full | half }

Velocidad (en Mb/s, a partir de 1000, sólo funciona en full-duplex):

(config-if)#speed { 100 | auto }

Switch capa 3. Activa un puerto en modo de enrutamiento:

(config-if)#no switchport

Routers

En los interfaces serial, en el que pongamos el extremo DCE del cable, hay que conf:

(config-if)#clock rate 129000 Valor arbitrario.

En el que sea DTE, hay que quitarlo:

(config-if)#no clock rate

Configurar Loopback:

(config)#interface loopback 100 N.º arbitrario

En IPv4 no hay lista de reyes godos, pero podemos poner su nombre al dispositivo, p.e.:

(config)#hostname Chindasvinto

VLANs

Aumentan el número de los dominios de broadcast, añaden seguridad, dividen en subredes y facilitan el añadir y cambiar hosts en ellas. Además usando switches ¿qué más se puede pedir?

Creación

(config)#vlan 99	N.º que queramos (El rango normal es hasta 1005. La 1, 1002-1005 ya existen y no se pueden borrar)
(config-vlan)#name Informatica	
(config)#no vlan 99	La borra. Los puertos que tuviera asignados siguen hab. pero ya no se pueden comunicar.

Eliminar la conf. de las vlan:

#delete vlan.dat	
#erase startup-config	Sólo es nec. en switches Catalyst.

Mostrar conf.

#show vlan	
#show vlan brief	Muestra sólo n.º, nombre, estatus y puertos asoc.
#show interface f0/1 switchport	Ver datos de la vlan asoc. al puerto, modo acceso, si está protegido…

Asignación

Cada interfaz donde se va a conectar un host, se asigna a una vlan. Para cada interfaz donde vayamos a usar una vlan:

(config-if)#switchport mode access	
(config-if)#switchport access vlan 10	Si no existe la vlan, la crea auto.
(config-if)#no shutdown	
(config-if)#no switchport access vlan	Quita el puerto de la vlan donde esté y lo pone en la 1.

Troncales (Vlan Nativa)

Las conexiones entre switches (o entre switch y router) son las troncales, por las que pasa el tráfico de varias vlan (y el tráfico que no va por ninguna vlan). La vlan nativa es la que se asigna al troncal. Por defecto, la 1, aunque conviene cambiarla. Tiene que ser la misma en ambos extremos, si no se pierde el tráfico de control, aunque no el de datos. No poner hosts en enlaces troncales ni asignados a la vlan nativa.

(config-if)#switchport mode trunk	
(config-if)#switchport trunk native vlan 99	Vlan nativa del troncal.
(config-if)#switchport trunk alowed vlan 10,20,30,99	Vlans que pueden cruzar el troncal (no olvidar la nativa)
(config-if)#no switchport trunk allowed vlan	Quita las vlan permitidas.
(config-if)#no switchport trunk native vlan	Permite todas las vlan y pone de nativa a la 1.
(config-if)#switchport mode access	Quita el trunk.

Muestra los interfaces en modo trunk:
#show interfaces trunk

SVI (Vlan Administrativa)

Es la que se asigna a la interfaz virtual del switch (SVI) para acceder remotamente. Por defecto la 1, aunque conviene cambiarla. Sólo puede haber una.

DTP

Dynamic Trunking Protocol. Negocia el modo del puerto. Propietario de Cisco. Desactivar si conectamos con switches no Cisco.

Desactivar:

```
(config-if)#switchport mode trunk
(config-if)#switchport nonegotiate
```

Muesta el modo DTP:

```
#show dtp interface f0/1
```

Seleccionar:

Por defecto. Trunk sólo si el otro es trunk o desirable

```
(config-if)#switchport mode dynamic auto
```

Trunk salvo que el otro sea access

```
(config-if)#switchport mode dynamic desirable
```

DHCP Snooping

Es para evitar el DHCP Spoofing (sí, suena raro). Determina qué puertos del switch pueden responder a pet. DHCP. Se ponen en modo Trusted sólo los de entrada que estén en la ruta desde el servidor DHCP hasta los clientes.

```
(config)#ip dhcp snooping
(config)#ip dhcp snooping vlan 10,20          Lo habilita para ciertas VLAN.

(config)#interface f0/1
(config-if)#ip dhcp snooping trust            Define el puerto como trusted.
```

Opcional. Limita el ratio de pet. DHCP que se pueden hacer por un puerto untrusted:

```
(config-if)#ip dhcp snooping limit rate 5
```

Port Security

Se configura para cada interfaz.

Activación:

```
(config-if)#switchport mode access
(config-if)#switchport port-security
```

Limita el n.º de MAC permitidas en un puerto:

```
(config-if)#switchport port-security maximum 10
```

Guarda las MAC aprendidas dinámicamente en la configuración actual:

```
(config-if)#switchport port-security  mac-address sticky
```

Modos de violación de seguridad:

Protect: Descarta y no avisa.
Restrict: Descarta y avisa.
Shutdown: Descarta, avisa y deshabilita (para levantarlo hacer shutdown y no shutdown)

```
(config-if)#switchport port-security violation {protect | restrict | shutdown}
```

Ver estado:

```
#show port-security interface f0/1
```

#show port-security address	Muestra la tabla de MAC seguras para todas las interfaces.

Pvlan Edge

Hace que no se envíe tráfico (except. de control) entre puertos protegidos.

```
(config-if)#switchport protected
```

```
(config-if)#no switchport protected
```

Enrutamiento

Vemos el método Router on Stick (Router pinchao en un palo). Dividimos un interfaz físico del router en varios lógicos, uno para cada VLAN.

Crear subinterfaces

R1(config)#interface g0/0.10	Dividimos el interfaz g0/0 en subinterfaces. El n.º (.10) puede ser cualquiera, pero se recomienda que coincida con el n.º de VLAN.
R1(config)#no interface g0/0.10	Lo marca como borrado.

Asignar subinterfaz a VLAN

Asigna la VLAN 10. Opcionalmente podemos indicar que es la nativa:

```
R1(config-subif)#encapsulation dot1q 10 [native]
```

Esto sería el gateway por defecto y la máscara de red de los hosts de la VLAN:

```
R1(config-subif)#ip address 192.168.10.1 255.255.255.0
```

Activamos los subinterfaces:

```
R1(config)#interface g0/0
R1(config-if)#no shutdown
```

VLAN Nativa

Si usamos una VLAN nativa distinta de la de por defecto (la 1), hay que configurarla también:

```
R1(config)#interface g0/0.99
R1(config-subif)#encapsulation dot1q 99 native
R1(config-subif)#ip address 192.168.99.1 255.255.255.0
```

¡Ojo! El puerto del Switch que conecta con el router, debemos conf. en modo trunk.

```
Sw(config-if)#switchport mode trunk
Sw(config-if)#switchport trunk native vlan 99
Sw(config-if)#switchport trunk alowed vlan 10,20,30,99
```

Vlan nativa del troncal.

VLAN Administrativa

Igualmente con la VLAN administrativa. Es necesaria para gestionar el switch. Puede ser la misma que la nativa. En cualquier caso, la ip del switch y su gateway deben configurarse y pertenecer a la VLAN administrativa.

```
R1(config)#interface g0/0.88
R1(config-subif)#encapsulation dot1q 88
R1(config-subif)#ip address 192.168.88.1 255.255.255.0
```

En el Switch haríamos (recordar ésto sólo se puede hacer en una VLAN del Sw, la administrativa):

```
Sw(config)#interface vlan 88
Sw(config-if)#ip address 192.168.88.10 255.255.255.0

Sw(config)#ip default-gateway 192.168.88.1
```

Consejos de seguridad

- No usar la vlan 1
- Desactivar los puertos que no se usen y ponerlos en una vlan agujero negro (una que no se usa nunca, no otra cosa, malpensado).
- Usar una vlan con un ID no obvio para administración solamente.
- Usar SSH.
- Usar la vlan nativa sólo para eso, no conectar hosts.
- Hacer los trunk explícitamente, no usar los modos dinámicos.
- Poner vlans para VoIP aparte.

VTP

Nos aseguramos que los Sw tienen la conf. por defecto.

Primero conf. el servidor, luego los clientes y añadimos las VLAN en el servidor. Comprobamos que los clientes las hayan recibido.

Cuando añadamos un Sw nuevo al dominio, resetear primero el nº de revisión y comprobar la conf.

#show vtp status Versión VTP, nombre dominio, prunning, modo VTP, nº VLANs, nº revisión.

#show vtp password Muestra la clave.

Establece el modo VTP. Podemos ponerlo a transparente y luego vuelta para resetear el nº revisión.

(config)#vtp mode {server | client | transparent | off}

Nombre del dominio. Podemos cambiarlo y volverlo a poner como estaba para resetear el nº revisión.

(config)#vtp domain nomedominio

(config)#vtp password cisco12345 Clave del dominio (igual para todos los SW)

Enrutamiento

Si a un router le llegan varias rutas al mismo destino, sólo guarda la de AD más baja.

Estático

Muestra la **tabla de enrutamiento**. [Dist. Administrativa, AD/Métrica]

L, local, es la IP de los interfaces, /32 (/128 en IPv6) es una IP, no una red.
C, directamente conectada [0/0], si no tienen IP o están shutdown, no aparecen.
S, estática [1/-]
D, EIGRP [90/-]
O, OSPF [110/-]
R, RIP [120/-]
Inalcanzable [255/-] Nunca llegaré a Córdoba.

Opcionalmente podemos hacer que muestre sólo las dir. conectadas, las estáticas o las sumarizadas.

```
#show ip route [connected | static | summary]
```

Añade una ruta estática next-hop o recursiva (forma recomendada). Red y máscara de destino e IP del sgte. salto (del router sgte.)

```
(config)#ip route 192.168.1.0 255.255.255.0 192.168.0.2
```

Ruta dir. conectada, idem pero ponemos el interfaz de salida (del propio router). Si ponemos un loopback, tiene que ser así.

```
(config)#ip route 192.168.1.0 255.255.255.0 s0/0/0
(config)#ip route 192.168.1.0 255.255.255.0 loopback 1
```

Ruta completamente especificada. Ponemos ambas cosas:

```
(config)#ip route 192.168.1.0 255.255.255.0 s0/0/0  192.168.0.2
```

La elimina:

```
(config)#no ip route 192.168.1.0 255.255.255.0 192.168.0.2
```

```
#clear ip route *                    Borra todas las rutas.
#clear ip route 172.31.20.0          Sólo las rutas a la red de destino especificada.
```

Añade la ruta por defecto (quad-zero, ::/0 en IPv6):

```
(config)#ip route  0.0.0.0    0.0.0.0    192.168.0.2
```

```
(config)#ip default-network 0.0.0.0      Usa el gateway predeterminado.
```

Ruta Flotante: ruta alt. en caso de que la ppal falle. Tiene una AD más alta que 1.

```
(config)#ip route 192.168.1.0  255.255.255.0  192.168.0.2   5
```

En IPv6 es igual, excepto que si usamos link-local como next-hop, hay que poner también el interfaz.

RIP

Es un protocolo de enrutamiento dinámico de vector distancia. Sin clase (sólo en la v2 y no completamente). AD 120. Métrica: distancia (nº saltos)

Mostrar conf.

#show run

#show ip route rip Muestra las rutas aprendidas por RIP (R).

#show ip protocols

#debug ip rip
#undebug ip rip

Configurar

(config)#router rip

(config)#no router rip Lo desactiva.

(config-router)#version 2 Sólo envía y recibe act. de RIP2. RIP2 soporta VLSM (subredes sin clase) pero NO al sumarizar.

Anunciar subredes

(config-router)#network 192.168.2.0 Anuncia la subred. Toma la máscara del interfaz correspondiente.

(config-router)#no network 192.168.2.0

(config-router)#passive-interface g0/0 No envía act. por ese interfaz. Util para ahorrar recursos y por seguridad, si por ese interfaz no hay routers (o routers RIP al menos).

(config-router)#no passive-interface g0/0

(config-router)#passive-interface default Pone todos los interfaces pasivos por defecto.

(config-router)#no auto-summary Conviene desact. la sumarización auto. ya que sólo funciona bien en caso de que sean redes contiguas de la misma clase y además todas por el mismo lado. Sería correcto en routers de borde.

(config-router)#default-information originate Difunde el gateway de último recurso (ruta por defecto).

(config-router)#ipclassless Viene en routers antiguos para que pueda anunciar redes sin clase. Si lo quitamos (o no lo ponemos), no funciona.

RIPng (RIP IPv6)

Activar:

(config)#ipv6 router rip ID-RIP

(config-if)#ipv6 rip ID-RIP enable Se hace por interfaz. Se usa un ID arbitrario.

No hay comando network.

(config)#ipv6 rip ID-RIP default-information originate

OSPF

Es un protocolo de enrutamiento dinámico de link-state. Sin clase. AD 110. A diferencia de RIP, no da problemas de sumarización. Métrica: coste (basado en el ancho de banda)

Mostrar conf.

#show run	
#show ip route ospf	Muestra las rutas aprendidas por OSPF (O).
#show ip protocols	Muestra PID, RID, Area(s) y networks.
#show ip ospf	Muestra PID, RID, Area(s), Ancho de Banda de referencia, interfaces en cada área.
#show ip ospf interface [s0/0/0]	Muestra PID, RID, Area(s), DR/BDR, Temporizadores y Seguridad.
#show ip ospf neighbor	Para cada vecino, muestra el RID, la prioridad, el estado, si es DR o BDR, el intervalo de dead y la interfaz nuestra por la que lo vemos. Muy útil para ver la adyacencia y DR/BDR.
#show ip ospf database	Muestra la LSDB para cada área (si es un ABR). Los Router Link States son los de nuestra área y los Summary Net Link States son las redes aprendidas de otras áreas y el vecino que nos la envió. (Pero sólo Jose Hilario Viñeiredo, el Unico, lo vió y lo sabe todo, y no duerme.)

#debug ip ospf spf
#no debug ip ospf spf

Configurar

(config)#router ospf 1	Usamos un n.º arbitrario para el PID. Conviene que sea el mismo para todos los routers de la misma área, aunque esto no es necesario para formar adyacencias.

En la elección de DR/BDR, se mira primero la prioridad, luego el router ID, después la IP de loopback si existe y por último la mayor IP de los interfaces activos. Cuanto más alto los valores, más prioridad.

(config-router)#router-id 3.3.3.3	Ponemos el RID (Router ID). Sólo se suele poner en DR/BDR y candidatos. Es único. Si dos routers tienen el mismo ID, no serán vecinos.
(config-if)#ip ospf priority 20	Establece la prioridad del interfaz (0-255). Por defecto vale 1. Si ponemos 0, no participa en el proceso de elecc. DR/BDR, se queda fuera.
#clear ip ospf process	Vuelve al estado Init y reinicia el enrutamiento OSPF. Hay que hacerlo en todos los routers. No borra los datos OSPF. Si necesitamos cambiar el RID.
(config-if)#ip ospf hello-interval 20	Frecuencia (en s) con la que envía saludos. Por defecto vale 10 (30 en Frame Relay). Tiene que ser la misma entre vecinos.
(config-if)#ip ospf dead-interval 80	T. que espera hasta declarar a un vecino KO. Es 4x el de saludo. Aunque en principio lo pone el router auto. conviene hacerlo explícitamente.
(config-if)#no ip ospf {hello-interval | dead-interval}	Pone los valores por defecto.

Anunciar subredes.

Indica la IP de red y la máscara inversa (wildcard, 255-normal, todas acaban en impar ó 0) y el área a la que pertenece el interfaz. (En el CCNA usan 0 para conf. de una sola área, pero en la práctica ésa sería la backbone, habría que usar otro número 1, p.e.)

(config-router)#network 192.168.2.0 0.0.0.255 area 0

En IOS nuevos se puede poner la IP del interfaz y 0.0.0.0 de máscara:

(config-router)#network 192.168.2.1 0.0.0.0 area 0

(config-router)#passive-interface g0/0 No envía act. por ese interfaz. Util para ahorrar recursos y por seguridad, si por ese interfaz no hay routers (o routers OSPF al menos).

(config-router)#no passive-interface g0/0

(config-router)#passive-interface default Pone todos los interfaces pasivos por defecto.

(config-router)#default-information originate Difunde el gateway de último recurso (ruta por defecto).

Configurar coste

Para Fastethernet (100Mbps) y superior el coste es 1. Si tenemos enlaces más rápidos en la LAN, para tener una referencia más precisa, se pone esto en el router que anuncia la red donde está el enlace:

(config-router)#auto-cost reference-bandwidth 10000 Mbps. En este caso es un 10-Gigabit.

(config-if)#bandwidth 64 Kbps. Sirve sólo para calcular el coste. Normalmente sólo se usa en los serial (64, 1544…) si no se ajusta el real al que viene por defecto. Hay que ajustarlo en ambos interfaces de la conexión.

(config-if)#no bandwidth Vuelve al ancho de banda por defecto.

(config-if)#ip ospf cost 15625 Mete dir. el coste.

Seguridad

Recomendada MD5. No se encripta el envío de las act. de rutas, sólo se firma una parte para autentificarlas.

1-Opcional. Fuerza la autentificación en todos los interfaces OSPF. Si uno no lo está y el del vecino sí, no podrá formar adyacencias.

(config-router)#area 0 authentication message-digest

2-Se hace por interfaz. La clave tiene que ser la misma entre vecinos. Se poner también un num. que viene a ser el ID de la clave.

(config-if)#ip ospf message-digest-key 1 md5 micontraseña

3-Opcional. Configura la autentificación en un interfaz. Sobreescribe las pref. globales.

(config-if)#ip ospf authentication message-digest

Multiarea

El área 0 es la Backbone. Sólo hay una por AS. Intercambia áreas.

Router interno: Router que está dentro de un área.

ABR: Router que comunica 2 o más áreas.

ASBR: Router que comunica con otro AS (Sistema Autónomo). Normalmente, también conecta con el área 0.

O- Ruta OSPF Intra-area
O-IA Ruta OSPF que viene de otra área (OI)
O E1 Ruta OSPF externa tipo 1 (la mandan los ASBR)
O E2 Ruta OSPF externa tipo 2 (la mandan los ASBR)

Interárea no sumariza por defecto. Hay que hacerlo manualmente.

Envía la red del Area 1 sumarizada. Ojo, que la máscara es normal. Crea un interfaz Null para evitar bucles:

(config-router)#area 1 range 10.1.0.0 255.255.252.0

En OSPFv3 es igual, poniendo la dir de la forma: **2001:DB8:ACAD:1::/22**

Para anunciar una ruta externa sumarizada:

(config-router)#summary-address 10.1.0.0 255.255.252.0

#show ip ospf database	Muesta el contenido del LSDB para cada Area (si es un ABR)
...	
Router Link States…	Lo de nuestra area.
...	
Summary Net Link States…	Rutas aprendidas de otras areas y qué vecino nos la envió.
...	

OSPFv3

Es la versión de OSPF para IPv6. Se puede tener OSPFv2 y v3 simultáneamente en un interfaz (simplemente repetimos la conf. para ambos).

Salvo que se indique, es lo mismo que OSPFv2, pero cambiando ip por ipv6 en los comandos.

(config)#ipv6 unicast-routing

(config)#ipv6 router ospf 10

Se configura los parámetros para cada interfaz. No hay comando network:

(config-if)#ipv6 ospf 10 area 0

La elección de DR/BDR, es igual que OSPFv2, de hecho, se usa IPv4, no IPv6 para los RID. Como última instancia, lo pide por consola.

Usa autentificación IPSec.

(config-if)#ipv6 ospf authentication ipsec spi

EIGRP

Es un protocolo de enrutamiento dinámico de vector distancia. Sin clase. AD 5 (redes sumarizadas), 90 (redes internas) y 170 (redes externas). Al igual que RIP, puede dar problemas de sumarización. Era de Cisco, pero lo liberó en 2013. Métrica: compuesta (ancho de banda, delay, [fiabilidad, carga])

Mostrar conf.

#show run	
#show ip route eigrp	Muestra las rutas aprendidas por EIGRP (D).
#show ip protocols	Muestra PID (AS), RID, Métricas(AD y K) y networks, n.º de rutas del mismo coste y si está act. la autosumarización.
#show interface [s0/0/0]	Muestra PID, RID, AB, Temporizadores.
#show ip eigrp interfaces	Muestra los interfaces que participan en EIGRP.
#show ip eigrp neighbors	Para cada vecino, muestra la IP, la prioridad, el intervalo de hold (t. para declarar a un vecino kaput), el t. que lleva en la tabla y la interfaz nuestra por la que lo vemos.
#show eigrp topology [all-links]	Muestra la tabla de topología. Para cada ruta muestra el estado, el FD, y para cada sucesor y FS (si hay) el FD y RD y la interfaz por la que llegamos a él. Opcionalmente, puede mostrar todos los vecinos que no son FS porque no cumplen el FC. (EC - ¿Está Claro?)
#debug eigrp fsm **#no debug eigrp fsm**	Muestra logs de DUAL cuando se cambia una ruta.

Configurar

(config)#router eigrp 1	Usamos un n.º arbitrario para el PID. Conviene que sea el mismo para todos los routers de la misma área. Se denomina AS aunque no tiene que ver con un AS (sí, es confuso)
(config)#no router eigrp 1	Termina el proceso y borra todas las conf. de EIGRP

En la asignación RID, se mira primero el router ID, después la IP más alta de loopback si existe y por último la IP más alta de los interfaces.

(config-router)#eigrp router-id 3.3.3.3	Ponemos el RID (Router ID). Tiene que ser único. No vale 0.0.0.0 ni 255.255.255.255. Sirve para saber qué router nos manda las rutas ext.
(config-router)#eigrp log-neighbor-changes	Activado por defecto. Muestra los logs de las adyacencias.
(config-if)#ip hello-interval eigrp 1 20	Frecuencia (en s) con la que envía saludos. Por defecto vale 5 (60 en Frame Relay).
(config-if)#ip hold-time eigrp 1 150	T. que espera hasta declarar a un vecino KO. Tiene que ser >= que el Hello.

Anunciar subredes.

Indica la IP de red y la máscara inversa (wildcard, 255-normal, todas acaban en impar ó 0). Sólo usa las interfaces que pertenezcan a la subred:

(config-router)#network 192.168.2.0 0.0.0.255

De esta forma coge todas las subredes de los interfaces del router. A diferencia de RIP, usa las máscaras de cada interfaz:

(config-router)#network 192.168.2.0

(config-router)#passive-interface g0/0 No envía act. por ese interfaz. Util para ahorrar recursos y por seguridad, si por ese interfaz no hay routers (o routers EIGRP al menos).

(config-router)#no passive-interface g0/0

(config-router)#passive-interface default Pone todos los interfaces pasivos por defecto.

(config-router)#no auto-summary Conviene desact. la sumarización auto. ya que sólo funciona bien en caso de que sean redes contiguas de la misma clase. Sería correcto en routers de borde.

(config-router)#redistribute static Difunde el gateway de último recurso (ruta por defecto). D*EX AD 170.

Sumarización manual. AD 90. La envía por el interfaz dado:

(config-if)#ip summary-address eigrp 1 192.168.0.0 255.255.252.0

Configurar coste, AB y balanceo

(config-if)#bandwidth 64 Kbps. Sirve sólo para calcular el coste. Normalmente sólo se usa en los serial (64, 1544…) si no se ajusta el real al que viene por defecto.

(config-if)#no bandwidth Vuelve al ancho de banda por defecto.

(config-if)#ip bandwidth-percent eigrp 1 40 Det el % max. de ancho de banda que usará EIGRP (por defecto 50%)

(config-router)#maximun paths 8 N.º de rutas con igual coste por las que balancea (por defecto 4, max. 32, 1 lo deshabilita)

Seguridad

Recomendada MD5. No se encripta el envío de las act. de rutas, sólo se firma una parte para autentificarlas.

1- Crea un key chain y una clave (la misma y con el mismo ID para todos los routers)

(config)#key chain MILLAVERO

(config-keychain)#key 3

(config-keychain-key)#key-string miclave123

2- Se hace por interfaz. La clave tiene que ser la misma entre vecinos.

(config-if)#ip authentication mode eigrp 1 md5

(config-if)#ip authentication key-chain eigrp 1 MILLAVERO

EIGRP IPv6

Salvo que se indique, es lo mismo que EIGRP IPv4, pero cambiando ip por ipv6 en los comandos.

(config)#ipv6 unicast-routing

(config)#ipv6 router eigrp 10

(config-rtr)#eigrp router-id 1.1.1.1	Aquí es obligatorio ponerlo para que pueda haber convergencia.
(config-rtr)#no shutdown	¡Ojo! Esto es necesario en IPv6.

Se activa para cada interfaz. No hay comando network:

(config-if)#ipv6 eigrp 10

No tiene sumarización auto. pero sí manual.

ACL

Filtran paquetes en las capas 3 y 4. Sólo puede haber una lista por protocolo (p.e. IPv4 e IPv6), dirección e interfaz (Pero en un interfaz podemos tener asociadas varias listas, siempre que se aplique a protocolos o direcciones distintas y una lista puede aplicarse a varios interfaces). P.e. en un router con 3 interfaces e IPv4 y v6 habilitados, podríamos tener hasta 12 ACL (3x2x2).

La dirección (IN, inbound, OUT, outbound) es desde el punto de vista del router.

Es importante el orden de las sentencias de la lista. Primero se ponen las condiciones más específicas y luego las generales.

No controlan el tráfico que se genera en el propio router (como sesiones de telnet).

Implícitamente, la última instrucción es un **deny any**, pero podemos ponerlo explícitamente para marcar el fin de la lista y para que se contabilicen los accesos denegados.

Usan máscara inversa (wildcard) para las redes.

ACLs Standard

Se basan sólo en la IP de origen. Se ponen lo más cerca posible del destino. Numeradas de 1-99 y de 1300-1999.

(config)#access-list 1 permit any	Permite todos.
(config)#access-list 1 permit 0.0.0.0 255.255.255.255	Idem.
(config)#access-list 1 permit host 192.168.10.1	Permite sólo a un host.
(config)#access-list 1 permit 192.168.10.1 0.0.0.0	Idem.
(config)#access-list 1 permit 172.16.16.0 0.0.0.255	Permite a la red 172.16.16.0/24

Podemos controlar un rango de redes, haciendo como en la sumarización. P.e. podemos denegar el rango de subredes entre la 172.16.16.0/24 y la 172.16.31.0/24

(config)#access-list 1 deny 172.16.16.0 0.0.15.255

Revisión y edición

(config)#access-list 1 remark Esto es un comentario (Max. 100 car.)

Si ponemos **log** al final de una sentencia, manda mensajes de log (nº ACL, permitido o denegado, IP org y nº de paquetes) a la consola para el primer paquete que coincida y luego cada 5 min.

#show access-lists **#show access-list 1**	Muestra las listas de acceso y el nº de coincidencias.
#show ip interface g0/0	Muestra las listas del interfaz.
#clear access-list counters 1	Borra contadores de la lista 1.
(config)#no access-list 10	Elimina la lista. No olvidar quitarla también del interfaz.

Modo de configuración de ACL

```
(config)#ip access-list standard 1

      (config-std-nacl)#permit 200.200.10.64 0.0.0.7
      (config-std-nacl)#deny ...
      (config-std-nacl)#remark tal y cual

      (config-std-nacl)#no 10                    Borra la línea 10.
      (config-std-nacl)#no permit …              Elimina la instrucción dada.
      (config-std-nacl)#10 deny ...              Modifica la línea 10.
      (config-std-nacl)#deny … sequence 10       Idem
```

Nombradas

```
(config)#ip access-list [standard | extended] NOME
```

Se introducen las entadas en el modo de configuración de ACL.

Lista extendida que deniega todo el tráfico ftp:

```
(config-ext-nacl)#deny tcp any any eq 21
```

Aplicación

```
(config-if)#ip access-group 1 out          Aplica en el int. la ACL 1 en la salida.
(config-if)#ip access-group NOME out       Idem ACL NOME.

(config-line)#access-class 21 in           Aplica la ACL a VTY. Las conex. de telnet y
                                           SSH deben controlarse así.

(config-if)#no ip access-group 1 out       Quita la lista.
```

ACLs Extendidas

Permiten más criterios (IP de origen y destino, puertos, protocolos, etc...) Se ponen lo más cerca posible del origen. Numeradas de 100-199 y de 2000-2699.

```
(config)#access-list 114     {permit | deny | remark} {tcp | udp | icmp | ip}
                             ip org. mask org.
                             [operador nº puerto o nombre servicio org.]
                             ip dest. mask dest.
                             [operador nº puerto o nombre servicio dest.]
                             [established]
```

¡Ojo! Aquí hay que poner siempre un protocolo. Si ponemos **ip**, se aplica a cualquiera.

Operador y puerto son opcionales. Operador puede ser **eq** (=), **neq** (<>), **gt** (>), **lt** (<), **range**

La opción **established** sólo vale para **tcp**. Permite sólo respuestas a consultas enviadas a través de una conexión establecida. En la práctica sólo funciona bien con **www**.

```
(config)#access-list 101 permit ip any any      Permite todo.
```

Niega el acceso del host dado a la web y permite todo al resto:

```
(config)#access-list 101 deny tcp host 204.204.10.1 any eq 80
(config)#access-list 101 permit ip any any
```

Niega el acceso por udp a todos los hosts de la red 200.20.10.64/29:

```
(config)#access-list 114 deny udp 200.20.10.64 0.0.0.7 any
```

Si queremos aplicar la ACL al puerto origen (aquí a conex. a telnet):

```
...deny tcp any eq telnet ip dest. …
```

permit icmp …. echo	Permite ping. (sin poner eq)
permit icmp … echo-reply	Permite respuesta a ping.

ACL IPv6

Equivalen a las extendidas de IPv4.

Sólo son nombradas. No pueden tener el mismo nombre que una IPv4.

No hay máscara wildcard.

Sustituimos **ip** por **ipv6**, incluida la especificación de protocolos.

Ponemos **traffic-filter** en vez de **access-group**:

```
(config-if)#ipv6 traffic-filter NOME out
```

Terminan implícitamente con:

```
permit icmp any any nd-na
permit icmp any any nd-ns
deny ipv6 any any
```

Para permitir ICMP Neighbor Discovery.

```
(config)#ipv6 access-list CCNA
(config-ipv6-acl)#deny tcp any any gt 5000
(config-ipv6-acl)#deny ::/0 lt 5000 ::/0
```

DHCP

Viene hab. por defecto en los IOS que lo soportan.

Excluye dir.

Por defecto, usa todas las dir. de la red como pool, excepto la de red y broadcast, que las excluye auto. Para el resto, hay que indicar explícitamente las que se excluyen.

Excluye un rango de direcciones IP. También se pueden excluir una a una.

```
(config)# ip dhcp excluded-address 192.168.0.1 192.168.0.9
```

Conf. pool de dir.

Puede haber varios. En este caso, escogería el adecuado según de dónde le venga la petición. En el pool podemos poner cualquier red de la topología.

`(config)# ip dhcp pool NOME`	Crea el pool.
`(dhcp-config)# network 192.168.1.0 255.255.255.0`	Red del pool.
`(dhcp-config)# default-router 192.168.1.1`	Gateway de la red del pool. Puede haber hasta 8.
`(dhcp-config)# dns-server 209.165.200.225`	Opcional. DNS del pool.
`(dhcp-config)# domain-name ccna-lab.com`	Opcional.
`(dhcp-config)# lease 30 24 60`	Opcional. Días, horas y min. que dura la concesión. Por defecto 1 día.
`(config)# no service dhcp`	Deshabilita DHCP.
`#clear ip dhcp binding 192.168.0.1`	Quita la IP asignada y obliga a que se escoja otra.
`#clear ip dhcp pool NOME binding *`	Idem, pero con todas las IP del pool.

DHCP Relay

Por defecto, los clientes y el servidor tienen que estar en la misma red. Si queremos que el servidor sirva a clientes de otra red, conf. un router para que reenvíe las peticiones y mensajes.

`(config-if)# ip helper-address 192.168.2.254`	Ponemos la IP del servidor remoto. Hace lo mismo para otros servicios udp (DNS, FTP, etc...)

Cliente

`(config-if)# ip address dhcp`	Router como cliente.

Comprobación

`#show run`	
`#show ip dhcp binding`	Muestra las IP y MAC asignadas.
`#show ip dhcp server statistics`	Memoria, nº mensajes, etc...
`#show ip dhcp pool`	
`#debug ip dhcp server events`	

DHCPv6

SLACC (Stateless Address Autoconfiguration)

Por defecto en Cisco. No usa un servidor. El cliente tiene que det. su ID de interfaz (los últimos 64 bits de la dir. IPv6), ya sea mediante el método EUI-64 (por defecto en Cisco) o aleatoriamente.

```
(config-if)#no ipv6 nd managed-config-flag
(config-if)#no ipv6 nd other-config-flag
```
Servidor.

```
(config-if)# ipv6 address autoconfig
```
Cliente.

Stateless DHCPv6 (SLACC+DHCPv6)

Igual que antes, sólo que ahora sí que se usa el servidor, pero sólo para otros datos como el DNS, gateway, etc...

1-Configuramos el servidor como en DHCPv4 (acordarse de ip → ipv6) pero sin usar network ni default-router.

2-En el interfaz por donde el servidor escuchará las peticiones:

```
(config-if)# ipv6 dhcp server NOME_POOL

(config-if)# ipv6 nd other-config-flag
```

3-El cliente, igual que en SLACC.

Stateful DHCPv6

Usa el servidor para todo. Paso 1 igual que anteriormente, pero añadiendo:

Asigna el prefijo IPv6 y opcionalmente el t. de lease o el t. válido y preferido en seg.

```
(config-dhcpv6)#address prefix 2001:db8:acad:a::/64 [lifetime infinite | 6000 7000]
```

2-Igual que antes, pero la última instrucción es:

```
(config-if)# ipv6 nd managed-config-flag
```

3-En el cliente:

```
(config-if)# ipv6 address dhcp
```

Comprobación

```
#show ipv6 interface g0/1
```
Se mira en el cliente.

```
#show ipv6 dhcp pool
```
En el servidor.

Relay

```
(config-if)#ipv6 dhcp relay destination 2001:DB8:CAFE:1::6/64
```

NAT

Traduce dir. IP privadas a dir. IP públicas para que los hosts puedan salir a Internet. Dicho de otro modo, relaciona dir. Inside Local con Inside Global. Se pueden usar varios de los métodos sgtes. en un mismo router, siempre que no mezclemos las IP públicas disponibles.

NAT Estático

Rel. 1 a 1. Se conf. manualmente y no cambia. Conveniente para servidores.

```
(config)# ip nat inside source static 192.168.1.20 209.165.200.225
```

Con un **no** delante, se anula.

Interfaces:

(config-if)# ip nat inside	En todos los interfaces Inside del router NAT.
(config-if)# ip nat outside	En el interfaz outside.
(config-if)# no ip nat inside	Para cambiarlos, primero los quitamos.

NAT Dinámico

Rel N a N. Usa un pool de dir. públicas. A cada IP privada, le asigna una pública del pool según van llegando. Por defecto, las concesiones duran 24h.

1-Definimos el pool:

```
(config)# ip nat pool NOME_POOL 209.165.200.242 209.165.200.254
                        {netmask 255.255.255.224 | prefix-length  27 }
```

A la hora de reservar rangos, tenemos que controlar que no se pisen ni solapen con otras dir. usadas por NAT estático o PAT. Si el rango es discontinuo, lo def. en la misma instrucción separado por comas:

```
(config)# ip nat pool NOME_POOL ip_global_ini   ip_global_fin netmask máscara,
                                ip_global_ini2 ip_global_fin2 netmask máscara2
```

2-Definimos qué IPs privadas pueden usar el pool mediante una ACL estándard numerada:

```
(config)# access-list 20 permit 192.168.1.0 0.0.0.255
```

3-Rel. la ACL con el pool

```
(config)# ip nat inside source list 20 pool NOME_POOL
```

Los interfaces se conf. igual.

Control

(config)#ip nat translation timeout 3600	Nº de seg. que dura la concesión.
(config)#clear ip nat translation *	Borra todas.

```
(config)#clear ip nat translation inside global_ip local_ip
                                    [outside local_ip global_ip]
```

PAT (Port Address Translation o NAT overloading)

Rel. N a 1. Es la más común. Todos salen con la misma IP pública. Usa el nº de puerto del cliente para identificarlo, si ya está en uso, usa el sgte. disponible; para protocolos sin puerto, usa otro id que figure en la cabecera.

1-Def. pool igual que en NAT Dinámico. Si usamos sólo una IP pública, ip_global_ini = ip_global_fin

2-Igual que NAT Dinámico.

3- Si usamos sólo una IP pública, podemos usar la del interfaz outside:

(config)#ip nat inside source list 3 [pool NOME_POOL | interface s0/1] overload

Los interfaces se conf. igual.

Port Forwarding (Tunneling)

Permite a usuarios externos acceder a servidores internos usando un puerto dado. Es como NAT estático especificando el puerto:

(config)# ip nat inside source static {tcp | udp} 192.168.1.20 1200
209.165.200.225 8080

Comprobación

#show ip nat translations	Si no se usa NAT, no sale nada. Las estáticas siempre aparecen (las inside), si hay una conex. también muestra las outside. Muestra los puertos (ojo, para NAT estático y dinámico también)
#show ip nat statistics	Nº de traslaciones, interfaces, etc..
#clear ip nat statistics	

Serial (WAN)

Vemos los distintos protocolos de encapsulamiento (capa 2) de los datos en tramas antes de cruzar el enlace WAN.

HDLC

Por defecto en Cisco. Propietario. Síncrono. Usa un flag para marcar el ini y fin de cada trama.

(config-if)#encapsulation hdlc

PPP

Estándard. Síncrono/Asíncrono. Usa un flag para marcar el ini y fin de cada trama. Controla la calidad del enlace. Incluye seguridad, compresión y multilink (combinar 2 ó más enlaces).

Configuración

(config-if)#encapsulation ppp

(config-if)#compress {predictor | stac} Establece el sist. de compresión. No usar cuando el tráfico consiste en archivos ya comprimidos.

(config-if)#ppp quality 80 Establece el % de calidad mín. del enlace. Si no llega, lo cierra.

(config-if)#no ppp quality Lo deshabilita.

(config)#interface multilink 1 Establece un multilink.

(config-if)#ip address …. No olvidarse de poner la ip del enlace.
(config-if)#encapsulation ppp
(config-if)#ppp multilink
(config-if)#ppp multilink group 1

(config)#no ppp multilink Desactiva el multilink.

#show ppp multilink Muestra los enlaces multilink.

Seguridad

Puede ser PAP ó CHAP. PAP es sin encriptación, two-way. Sólo se usa si no queda otro remedio y en simulaciones. CHAP usa md5 para autenticar los routers participantes. Three-way. Re-autentica periódicamente.

El usuario tiene que ser el hostname del router. El otro router tiene que tener nuestro hostname en su BD local de usuarios. Las contraseñas de ambos tienen que ser la misma, claro.

(config-if)#ppp authentication {pap | chap} Podemos poner ambos por orden de preferencia.

Esto sólo es necesario en PAP:

(config-if)#ppp pap sent-username usuario password miclave

#show int s0/0/0 Esto lo tenemos que hacer en el router receptor, para ver lo que envía el emisor.
#show run Es la única manera de ver nuestros usuarios locales creados.

#debug ppp [authentication]

Frame Relay

Más rápido. Usa PVC (Circuitos permanentes virtuales) o SVC (Circuitos intercambiados virtuales) que se establecen dinámicamente. Cada usuario tiene una línea dedicada al nodo FR del proveedor, que le conecta con otros clientes. Comparte el AB entre clientes.

FR Switch: equipo DCE del proveedor que conecta los clientes a la red FR.

DLCI (Data Link Connection ID), identifica el enlace con el destino de la conexión. Es un valor local en cada router y no tiene necesariamente que ser único en la WAN FR. Pueden existir varias conex. lógicas en un mismo enlace físico.

LMI (Local Management Interface), es un mecanismo que informa del estado entre los DTE y los DCE. Tiene que ser del mismo tipo entre ambos. Se conf. auto. a partir del IOS 11.2

La configuración es muy dependiente de la topología que usemos.

No hay gestión de errores.

Conf. y comprobación básica

(config)# interface s0/0/0

Configuramos la ip...

(config-if)#encapsulation frame-relay [cisco | ietf] Si no ponemos nada, usa cisco por defecto. Para conectar con routers no cisco hay que usar ietf.

(config-if)#bandwidth 64 Kbps. Opcional. Se ajusta en ambos interfaces de la conexión.
(config-if)#no shutdown

(config-if)#no encapsulation frame-relay Lo deshabilita.

#show interfaces s0/0/0 Muestra la IP, encapsulación, LMI y DLCI.
#show frame-relay pvc [102] Estado de un (o todos) PVC, interfaz, FECN, BECN.
#show frame-relay lmi Estadísticas de la conex. y tipo de LMI.
#show frame-relay map Interfaz, IP next-hop, DLCI, si es dinámico o estático, encapsulación y parámetros.

Dynamic Mapping

Se basa en IARP. Resuelve una IP next-hop a un DLCI. Necesita una conex. directa entre los extremos. Habilitada por defecto en los equipos Cisco para todos los protocolos. Si la deshabilitamos, hacemos:

#clear frame-relay inarp Limpia los mapas dinámicos.
...
(config-if)#frame-relay inverse-arp ip 102 Habilita IARP en el int. para IP en el DLCI 102.

Static Mapping

Lo conf. a mano. No se puede usar con el dinámico para el mismo DLCI y protocolo. Es necesario si un router no soporta IARP o usamos una topología Hub&Spoke (estrella) entre los DTE. Acordarse de mapear las conexiones en ambos sentidos. Después de hacer la conf. básica, hacemos:

(config-if)#no frame-relay inverse-arp Es necesario si un router no soporta IARP.

Define el DLCI que se usa (102) para todos los paquetes que van a la IP de destino dada. Permite el tráfico broadcast. La opción ietf se pone cuando conectamos a un router no Cisco:

(**config-if)#frame-relay map ip 10.1.1.2 102 broadcast [cisco | ietf]**

(config-if)#no shutdown

En IPv6 es similar, aunque aquí la opción broadcast sólo es necesaria en link-local:

```
(config-if)# frame-relay map ipv6 FE80::2  102 broadcast
```

Subinterfaces

La configuración de FR más común es usar subinterfaces de tipo punto a punto, que permiten evitar el problema del horizonte dividido (no reenvía paquetes por la interfaz que le llegan, causando problemas en los proto. de enrutamiento de vector distancia). También se puede usar con multipunto.

Punto a punto

Cada par de routers están en su propia subred (normalmente /30). Sólo se usa un DLCI en cada subinterfaz. No le afecta el problema del horizonte dividido.

Conf. en R1. Primero nos aseguramos que el interfaz no tiene IP configurada:

```
R1(config)#interface s0/0/0
R1(config-if)#no ip address
R1(config-if)#encapsulation frame-relay
R1(config-if)#no shutdown
```

Definimos los subinterfaces. No tienen porqué corresponderse con el DLCI, pero así es más claro:

```
R1(config)#interface s0/0/0.102 point-to-point
R1(config-subif)#ip address 10.1.1.1 255.255.255.252
R1(config-subif)#frame-relay interface-dlci 102
R1(config-subif)#description CONEX. CON R2
```

En R2 haríamos algo similar con ip 10.1.1.2/30 y un DLCI distinto (201 p.e.)

```
R1(config)#interface s0/0/0.103 point-to-point
R1(config-subif)#ip address 10.1.3.2 255.255.255.252
R1(config-subif)#frame-relay interface-dlci 103
R1(config-subif)#description CONEX. CON R3
```

Idem en R3.

Si nos equivocamos, hacemos shutdown en el físico, hacemos los cambios y lo volvemos a levantar.

Multipunto

En principio, todos los routers pueden estar en la misma subred, aunque también se puede conf. como en punto a punto. Se podría poner varios DLCI en la misma subinterfaz. Le afecta el problema del horizonte dividido.

Aparte de lo ya comentado, se haría similarmente, pero poniendo **multipoint** al def. el subinterfaz.

Para conf. el FR Switch (DCE) haríamos:

```
(config)#frame-relay switching

(config)#interface s0/0/1
(config-if)#no ip address
(config-if)#encapsulation frame-relay
(config-if)#clock rate 56000
(config-if)#frame-relay intf-type dce
```

Mapeamos las conex. del cliente que accede por este interfaz (en este caso, R1). Especificamos el DLCI del cliente y el interfaz y DLCI correspondientes del destino.

```
(config-if)#frame-relay route 102 interface serial 0/0/0 201
(config-if)#frame-relay route 103 interface serial 0/0/2 301
(config-if)#no shutdown
```

#show frame-relay route Muestra las rutas mapeadas.

GRE

Generic Routing Encapsulation. Es un protocolo de VPN básico, no seguro (encripta, pero la clave viaja junto con los paquetes). Crea un túnel IP (enlace virtual punto a punto). No tiene estado, ni mecanismos de control de flujo. Es un carrier (proto. que lleva tráfico de diversos tipos sobre un proto. de transporte, en este caso IP).

(config)#interface tunnel 0	Crea el int. del túnel. El nº no tiene porqué ser el mismo en ambos extremos.
(config-if)#tunnel mode gre ip	
(config-if)#ip address 10.10.10.1 255.255.255.252	IP y subred virtual para el túnel.
(config-if)#tunnel source s0/0/0	Podemos poner también la IP del interfaz físico del túnel.
(config-if)#tunnel destination 209.165.122.2	¡Ojo! Es la Ip del interfaz físico de destino.
(config-if)#no shutdown	

#show interface tunnel 0

A la hora de hacer rutas dinámicas o estáticas, podemos usar como IP de destino la virtual del túnel.
Para enrutamiento dinámico, anunciamos la subred del túnel.