Escuela Superior de Cómputo

Ingeniero en Sistemas Computacionales

Simbología

Router

Switch Multicapa

Línea Serial

LAN Switch

Red/Internet

Hub

PC

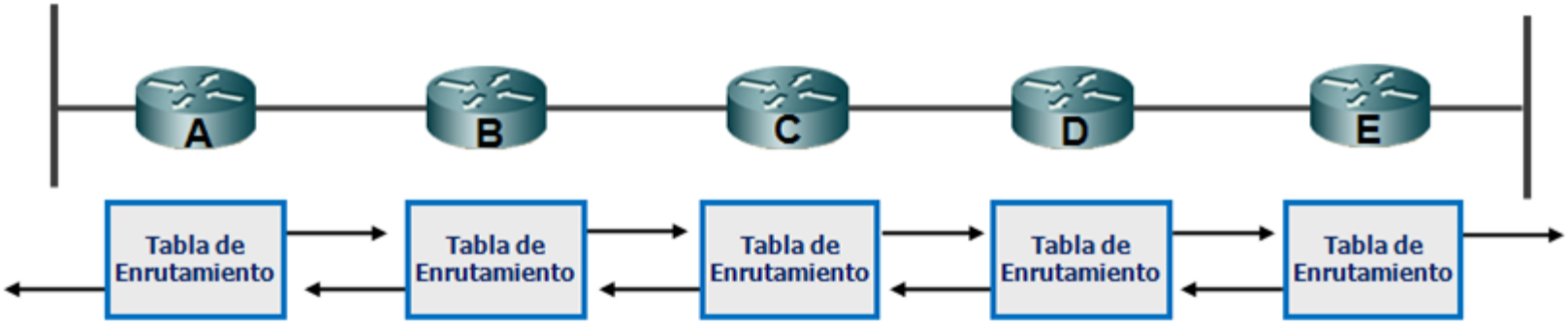
Protocolos de Enrutamiento Vector Distancia

Algoritmo de Enrutamiento Vector Distancia

Algoritmo de Enrutamiento Vector Distancia

Los protocolos vector distancia utilizan el algoritmo **Bellman-Ford** para calcular las mejores rutas.

- Cada router recibe una tabla de enrutamiento de sus vecinos conectados directamente.
- El *router* añade un valor de vector de distancia (número de saltos) y pasa esta nueva tabla de enrutamiento a sus *routers* vecinos.
- Este proceso se lleva a cabo en todas las direcciones entre *routers* vecinos directamente conectados.



Actualización de las Tablas de Enrutamiento

por Vector Distancia

Actualizaciones de las Tablas de Enrutamiento por Vector Distancia

Actualizaciones de las Tablas de Enrutamiento por Vector Distancia

- Se producen al haber cambios en la topología.
- Al recibir una tabla de enrutamiento completa de un vecino, un *router* puede verificar todas las rutas conocidas.
- El *router* hace cambios en su tabla de enrutamiento local con la tabla de enrutamiento recibida de un *router* vecino.

Actualizaciones de las Tablas de Enrutamiento por Vector Distancia

En las actualizaciones de enrutamiento, a una entrada de un destino nuevo o modificado, el *router* agrega 1 al valor de la métrica indicado en la actualización y lo incluye en la tabla de enrutamiento.

A este proceso se le conoce como “enrutamiento por rumor”; el conocimiento que tiene un *router* de la red está basado en la perspectiva de la tabla de enrutamiento de un *router* vecino.

Definición de Cuenta Máxima

Definición de Cuenta Máxima

Definición de Cuenta Máxima

Para prevenir *loops* de enrutamiento, es implementado un límite en el número de saltos en una ruta desde el origen hacia el destino.

- El número máximo de saltos en una ruta por vector distancia es 15.
- En una actualización de enrutamiento recibida por un *router* que contenga una entrada nueva o modificada, y que al incrementar el valor de la métrica en 1 causa que sea infinita (esto es 16), el destino de red se considera inalcanzable.
- Una desventaja de ésta característica es que se limita el diámetro de la red a menos de 16 saltos.

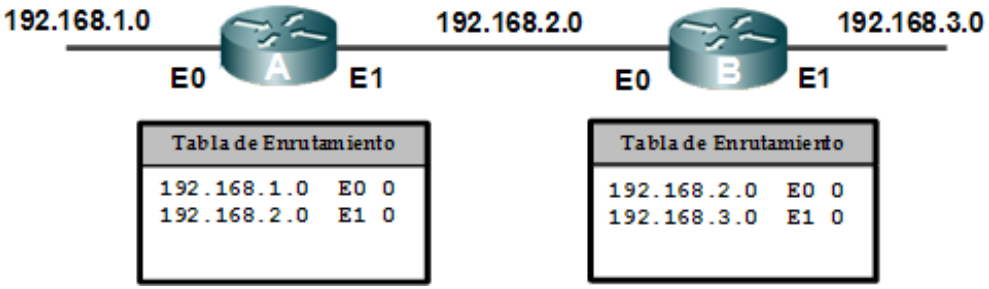
Descubrimiento de la Red por Vector Distancia

Descubrimiento de la Red por Vector Distancia

Descubrimiento de la Red por Vector Distancia

Cada router que utiliza el enrutamiento por vector-distancia comienza por identificar sus propios vecinos.

La interfaz que conduce a las redes conectadas directamente tiene una distancia de 0.

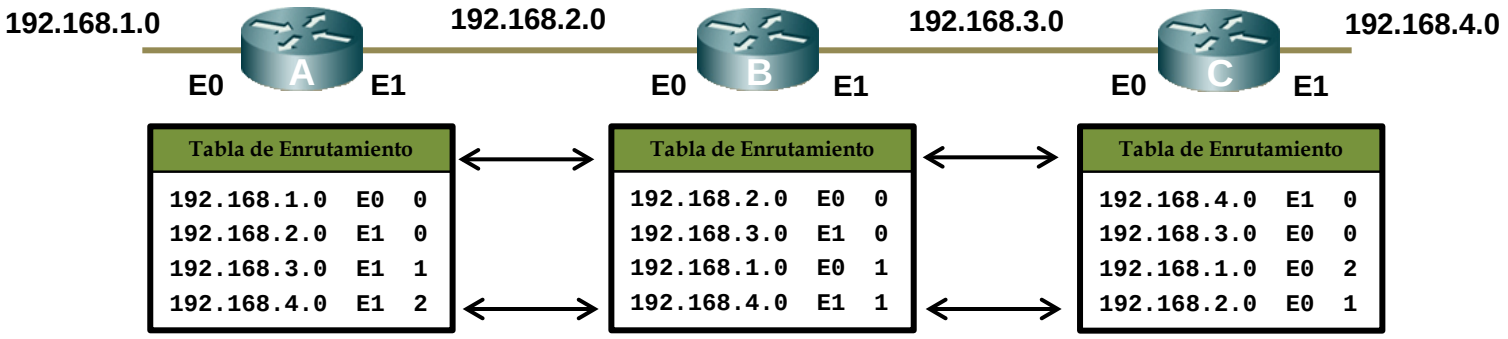


A medida que el proceso de descubrimiento de la red avanza, los *routers* descubren la mejor ruta hacia las redes de destino, de acuerdo a la información de vector-distancia que reciben de cada vecino.

Descubrimiento de la Red por Vector Distancia

Por ejemplo, el *router* A aprende acerca de otras redes según la información que recibe del *router* B.

Cada una de las redes de destino en la tabla de enrutamiento tiene una cifra total de vector-distancia, la cual indica la distancia a la que se encuentra dicha red por una ruta determinada.



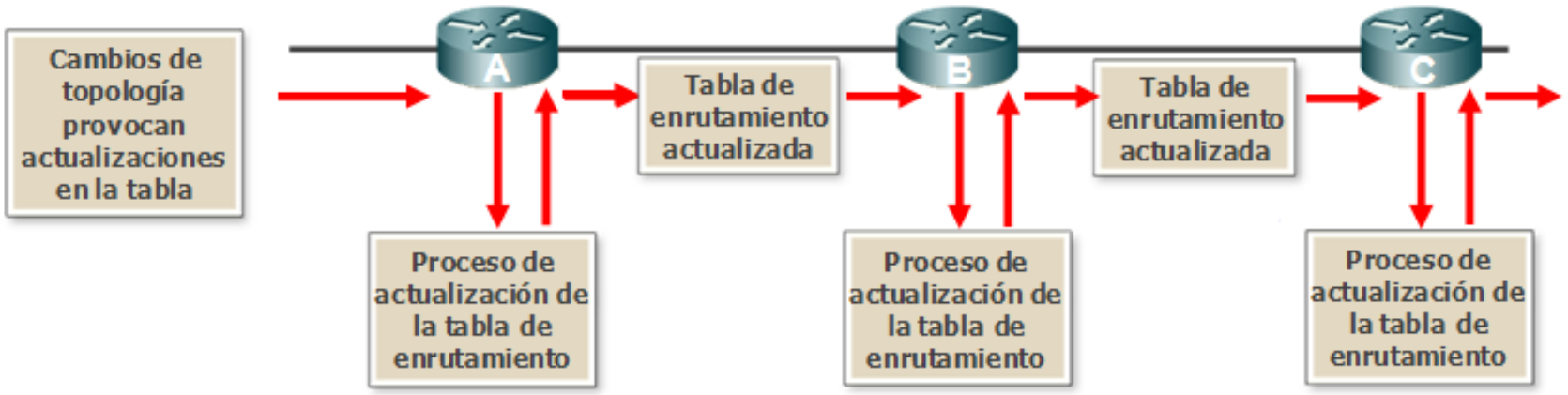
Propagación de los Cambios de Topología a Través de la Red

Propagación de los Cambios de Topología a Través de la Red

Al igual que en el proceso de descubrimiento de la red, las actualizaciones de cambios de topología avanzan paso a paso, de un *router* a otro.

Cada *router* envía su tabla de enrutamiento completa a cada uno de sus vecinos adyacentes.

Las tablas de enrutamiento incluyen información acerca del costo total de la ruta (definido por su métrica).

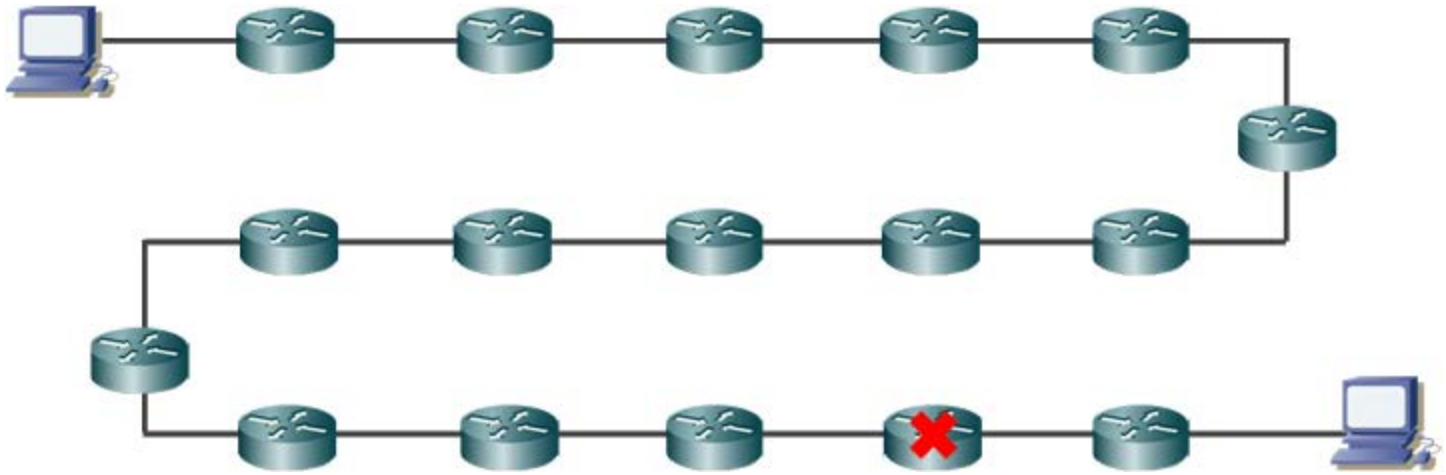


Componentes de la Métrica de Enrutamiento

Componentes de la Métrica de Enrutamiento

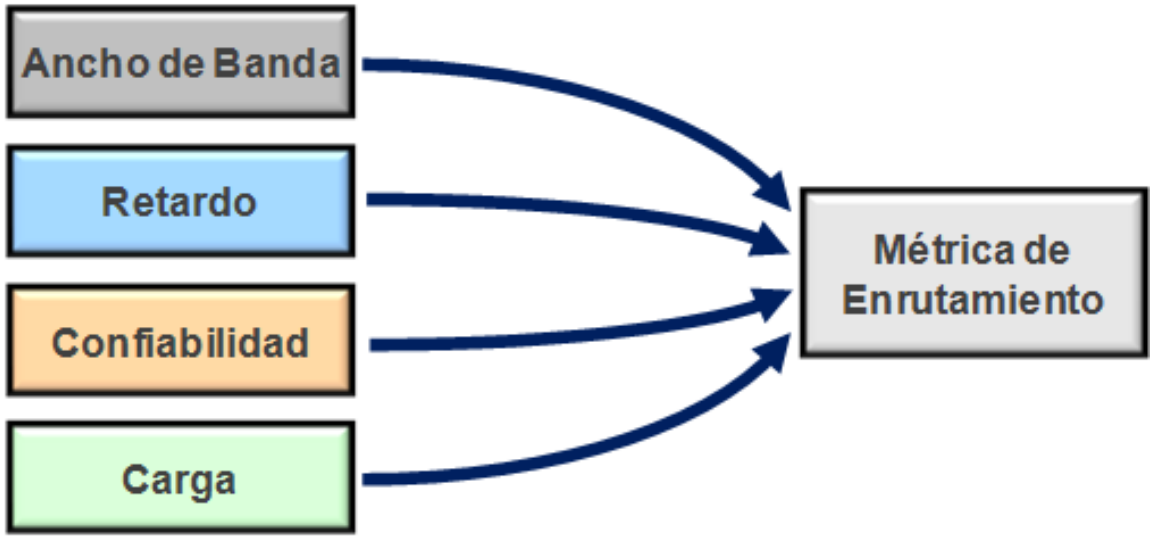
Algunos protocolos utilizan la cuenta de saltos para determinar la dirección y la distancia a cualquier red o enlace.

- Se selecciona la ruta con menos saltos.
- Es la única métrica de enrutamiento utilizada.
- No selecciona necesariamente la ruta más rápida a un destino.
- Es una métrica fija. Aunque pueden ser configuradas, es por naturaleza una métrica estática.



Componentes de la Métrica de Enrutamiento

IGRP y EIGRP utilizan un métrica compuesta, que se calcula como función del ancho de banda, el retardo, la carga y la confiabilidad.



Componentes de la Métrica de Enrutamiento

Estos protocolos utilizan el siguiente cálculo para la métrica:

métrica=[K1*Ancho de Banda + (K2*Ancho de Banda)/(256 – carga) + (K3*Retardo)]
*[K5/(confiabilidad+K4)]

Los siguientes son valores por defecto de las constantes:

K1=1, K2=0, K3=1, K4=0, K5=0

métrica = ancho de banda + retardo.

Cuando K4 y K5 son 0, la porción [K5/(confiabilidad+K4)] de la ecuación no forma parte del cálculo de la métrica.

Se usan las siguientes ecuaciones para determinar los valores en el cálculo de la métrica:

- Ancho de Banda para IGRP = 10000000/ancho de banda
- Ancho de Banda para EIGRP = (10000000/ancho de banda)*256
- Retardo para IGRP = retardo/10
- Retardo para EIGRP = (retardo/10)*256

Loops de Enrutamiento

Loops de Enrutamiento

Loops de Enrutamiento

- Resultado de tablas de enrutamiento incongruentes.
- Son debidos a la lenta convergencia de una red sujeta a cambios.
- Un *loop* de enrutamiento ocurre cuando dos o más *routers* tienen información de enrutamiento que indican incorrectamente que una ruta hacia un destino inalcanzable existe a través de los otros routers.

Loops de Enrutamiento

Hay técnicas disponibles para eliminar los *loops* de enrutamiento:

- **Horizonte dividido.**
- **Envenenamiento de rutas y envenenamiento inverso.**
- **Actualizaciones activadas por eventos.**
- **Temporizadores de espera.**

Loops de Enrutamiento

Este proceso describe como se produce un *loop* de enrutamiento:

1. Antes de la falla de la red 192.168.4.0, todos los *routers* tienen un conocimiento consistente de la red y tablas de enrutamiento correctas. Se dice que la red ha logrado convergencia.

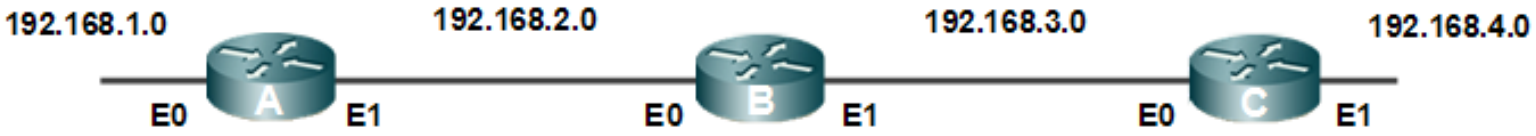


Tabla de Enrutamiento		
192.168.1.0	E0	0
192.168.2.0	E1	0
192.168.3.0	E1	1
192.168.4.0	E1	2

Tabla de Enrutamiento		
192.168.2.0	E0	0
192.168.3.0	E1	0
192.168.1.0	E0	1
192.168.4.0	E1	1

Tabla de Enrutamiento		
192.168.4.0	E1	0
192.168.3.0	E0	0
192.168.1.0	E0	2
192.168.2.0	E0	1

Loops de Enrutamiento

2. Cuando la red 192.168.4.0 falla, el *router* C lo detecta y detiene el enrutamiento de paquetes hacia esa red. Los *routers* A y B no han sido notificados, por lo que aún reflejan el acceso hacia la red 192.168.4.0 en sus tablas de enrutamiento.

   El *router* A aún indica que puede alcanzar a la red 192.168.4.0 a través del *router* B.

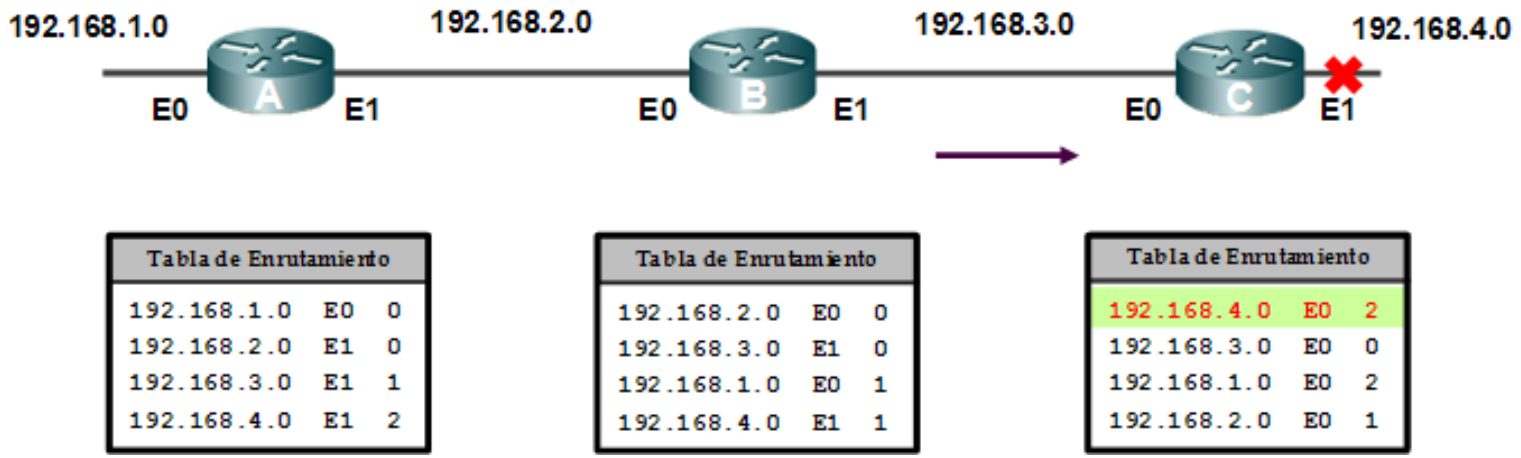
Tabla de Enrutamiento		
192.168.1.0	E0	0
192.168.2.0	E1	0
192.168.3.0	E1	1
192.168.4.0	E1	2

Tabla de Enrutamiento		
192.168.2.0	E0	0
192.168.3.0	E1	0
192.168.1.0	E0	1
192.168.4.0	E1	1

Tabla de Enrutamiento		
192.168.4.0	E1	D
192.168.3.0	E0	0
192.168.1.0	E0	2
192.168.2.0	E0	1

Loops de Enrutamiento

3. Cuando el *router* B envía una copia periódica de su tabla de enrutamiento hacia el *router* C, éste actualiza su tabla de enrutamiento, ahora reflejando una ruta hacia la red 192.168.4.0 a través del *router* B con una cuenta de 2 saltos.



Loops de Enrutamiento

4. En este ejemplo, un paquete destinado hacia la red 192.168.4.0 llega al *router* A y de acuerdo a su tabla de enrutamiento reenvía el paquete hacia el *router* B sobre su interfaz E1. Al llegar el paquete hacia el *router* B, reenvía el paquete hacia su interfaz E1 como lo indica su tabla de enrutamiento. El paquete es recibido por el *router* C que al revisar su tabla, lo reenvía sobre su interfaz E0, regresando al *router* B, quien nuevamente lo reenvía sobre su interfaz E1 hacia el *router* C.

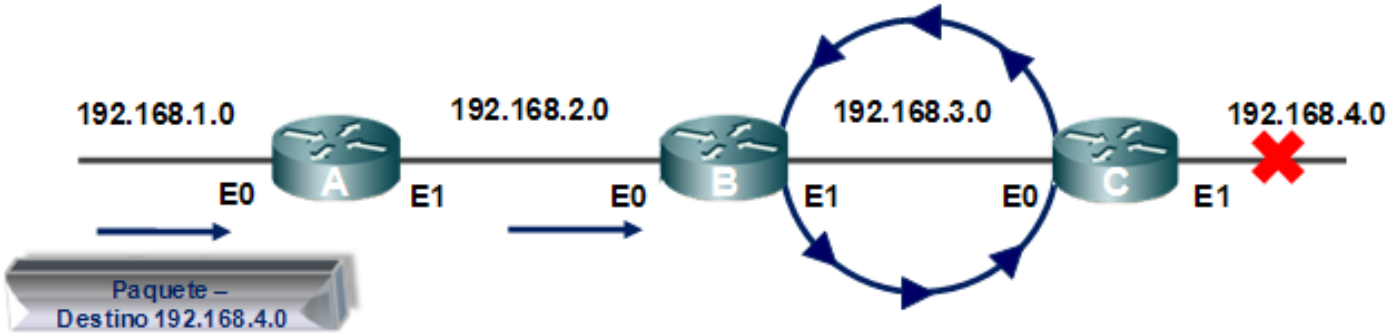


Tabla de Enrutamiento		
192.168.1.0	E0	0
192.168.2.0	E1	0
192.168.3.0	E1	1
192.168.4.0	E1	2

Tabla de Enrutamiento		
192.168.2.0	E0	0
192.168.3.0	E1	0
192.168.1.0	E0	1
192.168.4.0	E1	1

Tabla de Enrutamiento		
192.168.4.0	E0	2
192.168.3.0	E0	0
192.168.1.0	E0	2
192.168.2.0	E0	1

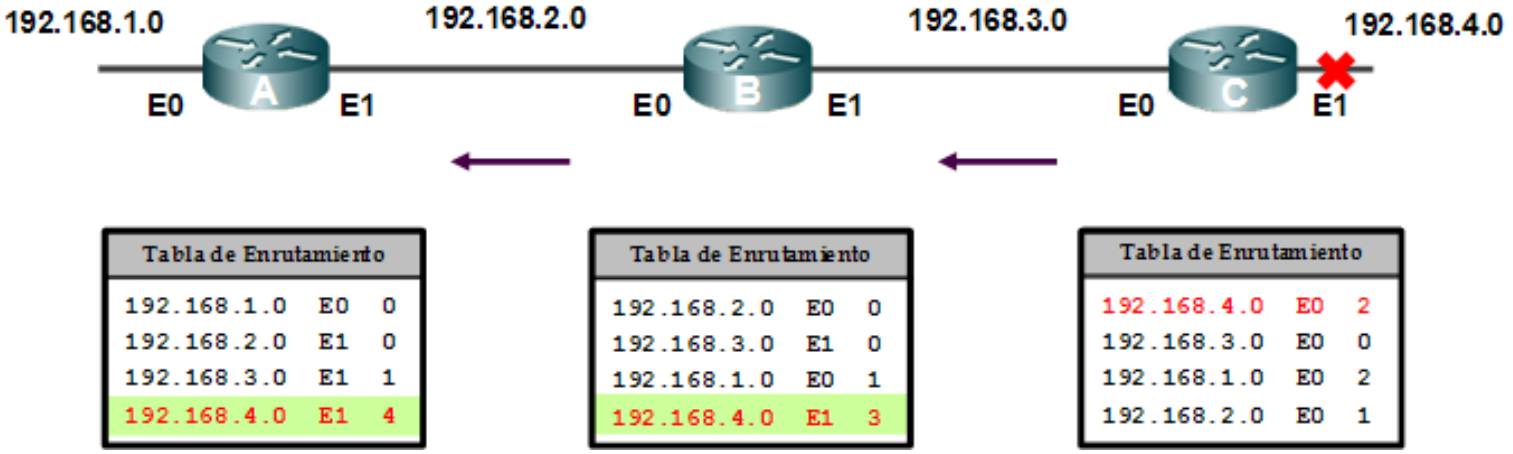
Cuenta al Infinito

Cuenta al Infinito

Continuando con el ejemplo de *loops* de enrutamiento, los *routers* B y C incluían en sus tablas rutas erróneas hacia la red 192.168.4.0:

1. El *router* B recibe una nueva actualización desde el *router* C, actualizando la ruta con un nuevo costo de 3 saltos.

2. El *router* B envía una nueva tabla al *router* A, que detecta la modificación de la distancia del vector hacia la red 192.168.4.0, y re-calcula la distancia hacia esa red con un valor de 4.

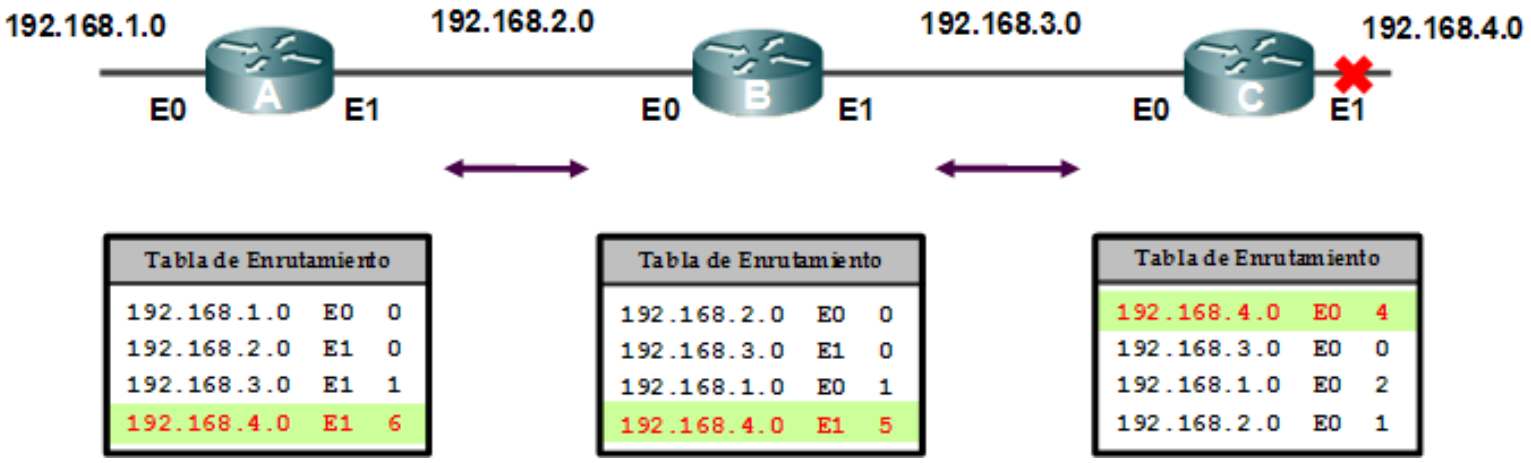


Cuenta al Infinito

3. En este punto, las tablas de los tres routers son incorrectas, mostrando rutas inexistentes hacia la red 192.168.4.0, con cuentas de saltos que no tienen sentido.

4. Las actualizaciones de tablas de enrutamiento continuarán entre los *routers*, con lo que las cuentas de saltos serán cada vez más grandes (contando al infinito).

Los paquetes hacia la red 192.168.4.0 nunca alcanzarán su destino, moviéndose continuamente entre los *routers* creando un *loop* de enrutamiento.



Horizonte Dividido

Horizonte Dividido

Horizonte Dividido (Split Horizon)

Es una técnica para eliminar *loops* de enrutamiento y para acelerar el proceso de convergencia.

- El horizonte dividido reduce la información de enrutamiento errónea y el coste derivado del enrutamiento.

- La regla del horizonte dividido dice que nunca es útil retornar información acerca de una ruta hacia la dirección en la que la información original provino.

Horizonte Dividido

- El *router* B tiene acceso a la red 192.168.4.0 a través del *router* C. No tiene sentido que el *router* B anuncie al *router* C que tiene acceso a esa red por el *router* C.
- Debido a que el *router* B pasó el anuncio de su ruta hacia la red 192.168.4.0 al *router* A, no tiene sentido que el *router* A anuncie su distancia a la red 192.168.4.0 al *router* B.
- Cuando el *router* C anuncia que su conexión a la red 192.168.4.0 está caída, el *router* B ve que no hay ruta alternativa hacia esa red destino, concluyendo que es inaccesible. El router C no usa al *router* B para alcanzar la red 192.168.4.0.

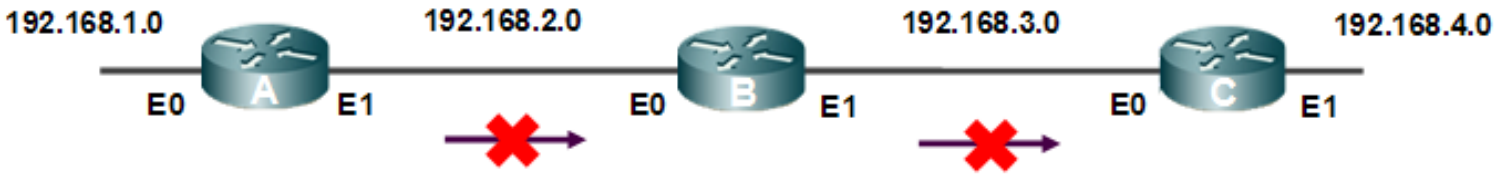


Tabla de Enrutamiento		
192.168.1.0	E0	0
192.168.2.0	E1	0
192.168.3.0	E1	1
192.168.4.0	E1	2

Tabla de Enrutamiento		
192.168.2.0	E0	0
192.168.3.0	E1	0
192.168.1.0	E0	1
192.168.4.0	E1	1

Tabla de Enrutamiento		
192.168.4.0	E1	0
192.168.3.0	E0	0
192.168.1.0	E0	2
192.168.2.0	E0	1

Envenenamiento de la Ruta

Envenenamiento de la Ruta

Envenenamiento de la Ruta (Route Poisoning)

El envenenamiento de rutas es otro mecanismo que ayuda a prevenir *loops* de enrutamiento.

El *router* establece una entrada de la tabla que mantiene consistente el estado de la red mientras otros *routers* convergen gradualmente.

Envenenamiento de la Ruta

Cuando la red 192.168.4.0 no se encuentra disponible, el *router* C envenena su enlace a esta red enviando una actualización indicando que tiene una métrica infinita y una cuenta de saltos de 16 (inalcanzable).

Debido al envenenamiento de la ruta hacia la red 192.168.4.0, el *router* B no es susceptible de efectuar actualizaciones incorrectas de la ruta hacia dicha red.

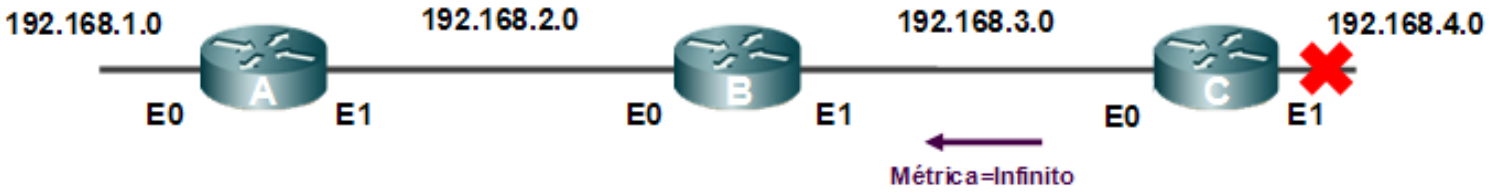


Tabla de Enrutamiento		
192.168.1.0	E0	0
192.168.2.0	E1	0
192.168.3.0	E1	1
192.168.4.0	E1	2

Tabla de Enrutamiento		
192.168.2.0	E0	0
192.168.3.0	E1	0
192.168.1.0	E0	1
192.168.4.0	E1	1

Tabla de Enrutamiento		
192.168.4.0	E1	D
192.168.3.0	E0	0
192.168.1.0	E0	2
192.168.2.0	E0	1

Envenenamiento de la Ruta

Envenenamiento Inverso (Poison Reverse)

Ofrece información explícita cuando una red no es accesible.

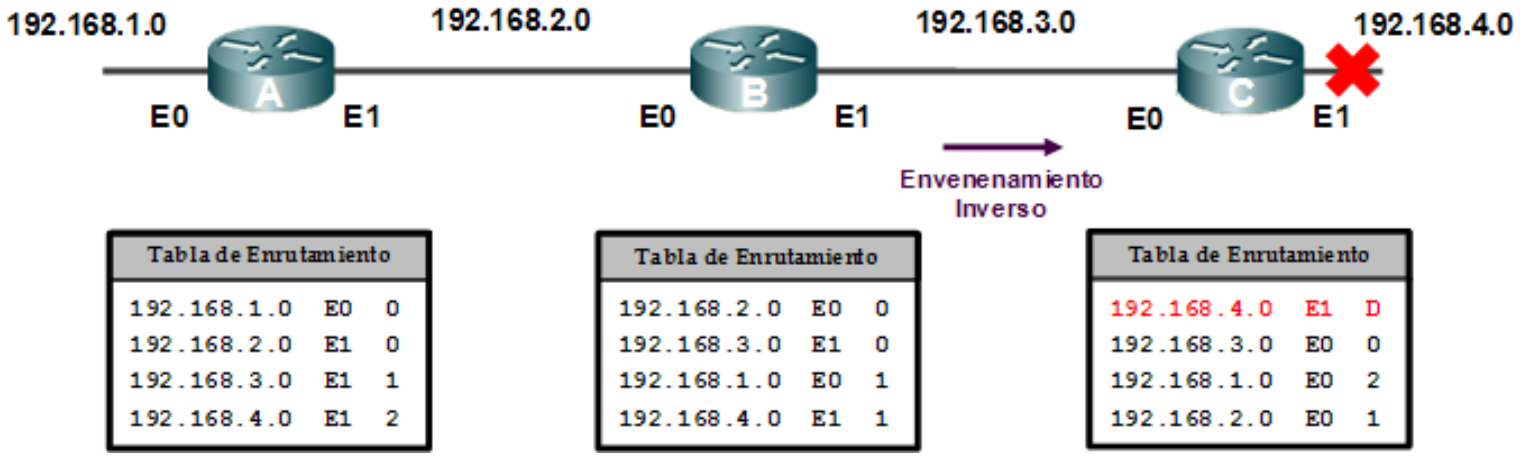
Es una circunstancia específica que hace caso omiso del horizonte dividido.

La regla del envenenamiento inverso dice: “Una vez que aprenda una ruta a través de una interfaz, publíquela hacia atrás como inalcanzable a través de esa misma interfaz”.

Envenenamiento de la Ruta

Cuando el *router* B recibe el envenenamiento de ruta desde el *router* C, envía un mensaje llamado actualización de envenenamiento inversa de vuelta al *router* C, de tal forma que publica esa red como inalcanzable a través de ese enlace.

Esto asegura que todas las rutas del segmento hayan recibido la información del envenenamiento de la ruta.



Envenenamiento de la Ruta

Actualizaciones Activadas por Eventos (Triggered Updates)

Las actualizaciones activadas es una técnica para acelerar el proceso de convergencia.

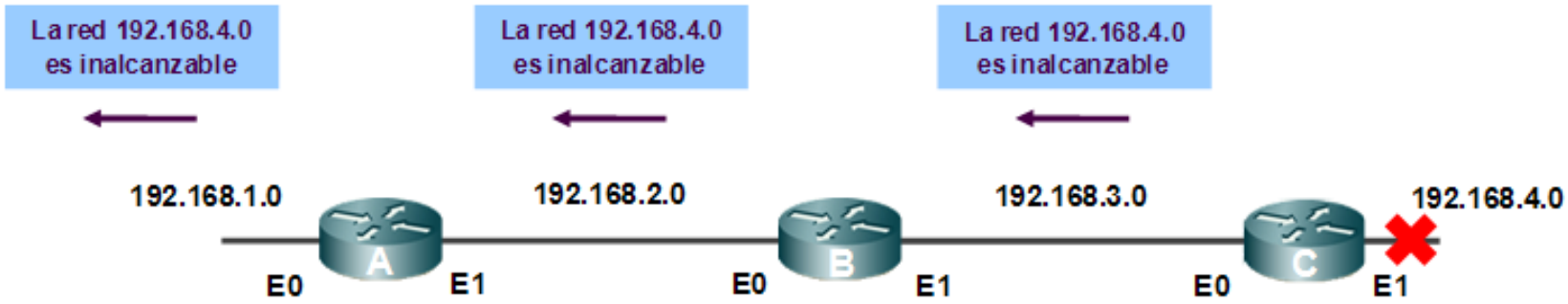
Las actualizaciones de la tabla de enrutamiento son enviadas normalmente a los *routers* vecinos en intervalos regulares.

Una actualización activada por un evento permite al *router* anunciar los cambios en los valores de la métrica casi inmediatamente, en vez de esperar al siguiente anuncio periódico.

Envenenamiento de la Ruta

El activador es un cambio en la métrica en una entrada de la tabla de enrutamiento.

Las actualizaciones generadas por eventos combinadas con el envenenamiento de rutas, agilizan el tiempo de convergencia ya que los routers vecinos no tienen que esperar 30 segundos antes de publicar la ruta envenenada



Si las actualizaciones activadas fueran enviadas inmediatamente por todos los *routers*, podrían causar una cascada de *broadcast* a través de la red.

Las actualizaciones activadas por eventos mejoran el tiempo de convergencia, pero con el costo de tráfico adicional de *broadcast* en la propagación de las actualizaciones.

Temporizadores de Espera

Temporizadores de Espera

Temporizadores de Espera (Hold–Down Timers)

Las actualizaciones activadas por evento no ocurren inmediatamente. Es posible que otro *router* haya transmitido una actualización periódica antes de haber sido recibida la actualización activada, causando la inserción de una ruta errónea, generando un *loop* por cuenta al infinito.

El problema de la cuenta al infinito se puede prevenir utilizando temporizadores de espera.

Los temporizadores de espera son usados para prevenir que mensajes de actualización regulares reinserten rutas que pueden ser incorrectas.

Temporizadores de Espera

El funcionamiento de estos temporizadores es:

1. Cuando un router recibe una actualización de un vecino que indica que una red previamente accesible es ahora inaccesible, el *router* marca esa ruta como “posible inaccesible” e inicia un temporizador de espera.

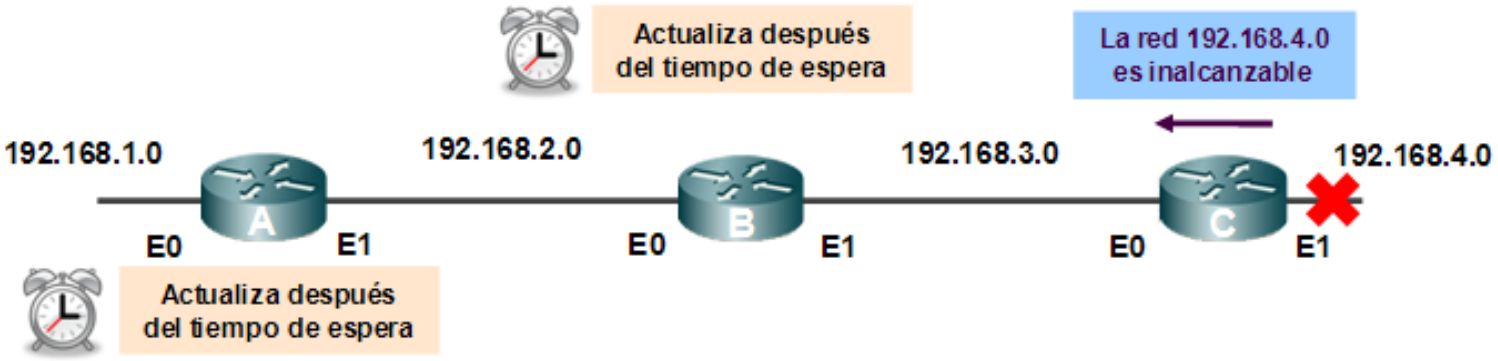
   Si se recibe una actualización del mismo vecino indicando que la red es nuevamente accesible antes que finalice el temporizador de espera, el *router* marca la red como accesible y elimina el temporizador.

2. Si una actualización llega de un *router* vecino diferente con una métrica mejor que la registrada originalmente para esa red, el *router* marca a la red como accesible y remueve el temporizador.

Temporizadores de Espera

3. Si se recibe una actualización de un *router* vecino con una métrica mayor (m=3, m=5) antes que expire el temporizador, la actualización es ignorada. Esto permite que haya más tiempo para que el cambio se propague por toda la red.

   Durante el periodo del temporizador de espera, las rutas aparecen en la tabla de enrutamiento como posiblemente inaccesibles. El *router* intentará aún enrutar paquetes hacia la red posiblemente inaccesible (en el caso de que la red tenga problemas de conectividad intermitentes; *flapping*).



Proceso de Enrutamiento RIP

Proceso de Enrutamiento RIP

RIP es un protocolo estándar abierto de enrutamiento.

- Descrito en el RFC 1058 y en el Estándar de Internet 56 (STD 56).
- Las características claves de RIP son:
  - Es un protocolo de enrutamiento por vector distancia.
  - Utiliza la cuenta de saltos como métrica para la selección de rutas.
  - La cuenta de saltos máxima permisible es 15. Si el número es mayor el paquete se descarta.
  - Las actualizaciones son a través de broadcast, por defecto cada 30 segundos.

Convergencia RIP

Convergencia RIP

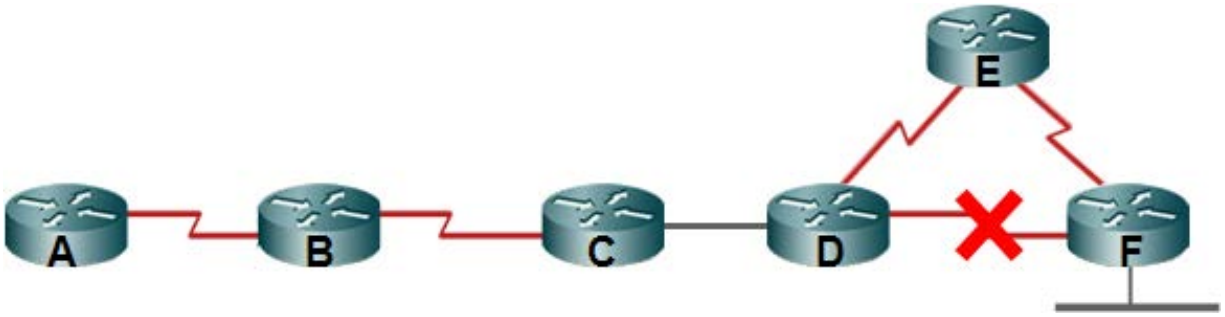
La convergencia es uno de los problemas asociados a los protocolos de enrutamiento de vector distancia.

RIP incluye diversas características las cuales están presentes en otros protocolos de enrutamiento.

Por ejemplo, RIP implementa los mecanismos de espera y horizonte dividido para prevenir la propagación de información de enrutamiento errónea.

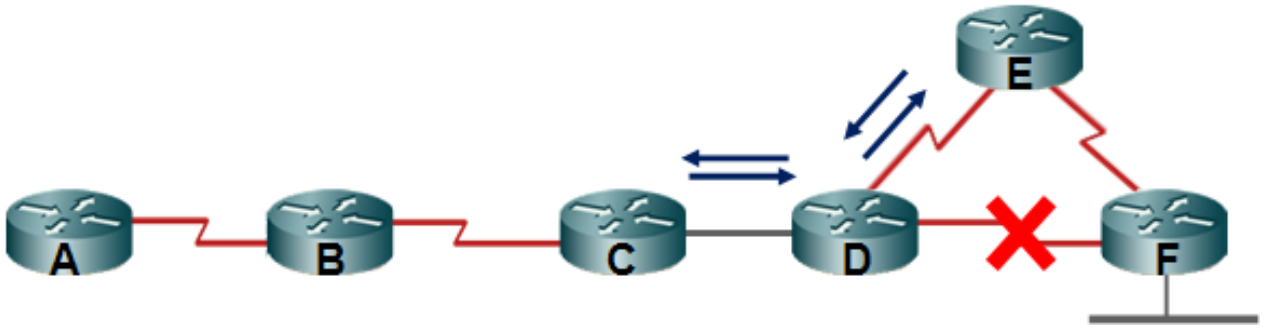
Convergencia RIP

La siguiente lista muestra los eventos de convergencia cuando un problema ocurre, en un proceso de convergencia del protocolo RIP.



1. El enlace entre los *routers* D y F falla.
2. El *router* D envía una ruta envenenada a los *routers* C y E. El *router* C informa al *router* B, que a su vez informa al *router* A. El *router* C elimina la entrada del enlace caído en su tabla de enrutamiento.

Convergencia RIP



4. El *router* D envía un "*broadcast*" de consulta. El *router* C le responde con un envenenamiento inverso, y el *router* E responde con una ruta hacia esa red. El *router* D ingresa esta ruta a su tabla de enrutamiento.

5. El *router* D no mantiene inactiva esta ruta debida a que había eliminado la entrada de la ruta hacia este enlace.

6. El *router* D anuncia la nueva ruta al *router* C, pero el *router* C la ignora debido a que la tiene como posiblemente inactiva. El *router* C encía otro envenenamiento inverso al *router* D.

Convergencia RIP

7. Los temporizadores de espera finalizan en los *routers* C, B y A, lo que causa que sus entradas en las tablas de enrutamiento puedan ser actualizadas.

El tiempo requerido para la convergencia del *router* A es el tiempo de detección, más el tempo de espera (*hold-down*), más dos tiempos de actualización, mas otro tiempo de actualización. En este caso la convergencia completa en el *router* A puede tomar alrededor de 240 segundos.

Uso de los Comandos de Configuración de RIP

Uso de los Comandos de Configuración de RIP

Estos comandos son utilizados para la configuración de RIP en el equipo SecureStack C3 de Enterasys.

router rip

Habilita o deshabilita el modo de configuración de RIP. La forma “no” de éste comando deshabilita RIP.

Se ejecuta desde el modo de configuración global.

```
router rip
no router rip
```

El siguiente ejemplo muestra la ejecución de éste comando.

```
diplomado(su)->router#configure
diplomado(su)->router(Config)#router rip
diplomado(su)->router(Config-router)#
```

Uso de los Comandos de Configuración de RIP

ip rip enable

Habilita RIP en una interfaz. La forma “no” de éste comando deshabilita RIP en una interfaz. Por defecto, RIP está deshabilitado en todas las interfaces.

Se ejecuta desde el modo de configuración de interfaz.

```
ip rip enable
no ip rip enable
```

Como ejemplo, para habilitar RIP en la interfaz VLAN 20:

```
diplomado(su)->router#configure
diplomado(su)->router(Config)#interface vlan 20
diplomado(su)->router(Config-if(vlan20))#ip rip enable
```

Uso de los Comandos de Configuración de RIP

ip rip send version

Establece la versión o las versiones de RIP para los paquetes de actualización transmitidos en una interfaz. La forma “no” de éste comando restablece la versión de los paquetes de actualización que fueron transmitidos por el router RIP. Por defecto, la versión de RIP es v1.

Se ejecuta desde el modo de configuración global.

```
ip rip send version {1 | 2 | r1compatible}
no ip rip send version
```

Las siguientes líneas muestran como establecer la versión de envío de RIP a la versión 2 para los paquetes transmitidos en la VLAN 20:

```
diplomado(su)->router#configure
diplomado(su)->router(Config)#interface vlan 20
diplomado(su)->router(Config-if(vlan20))#ip rip send version 2
```

Uso de los Comandos de Configuración de RIP

ip rip receive version

Establece la versión o las versiones de RIP para los paquetes de actualización aceptados en una interfaz. La forma “no” de éste comando restablece la versión por defecto de los paquetes de actualización del router RIP que fueron aceptados en la interfaz. Por defecto, la versión de RIP es v1.

Se ejecuta desde el modo de configuración de interfaz.

```
ip rip receive version {1 | 2 | 1 2 | none}
no ip rip receive version
```

El ejemplo muestra como establecer la versión de recepción de RIP a la versión 2 para los paquetes recibidos en la VLAN 20:

```
diplomado(su)->router#configure
diplomado(su)->router(Config)#interface vlan 20
diplomado(su)->router(Config-if(vlan20))#ip rip receive version 2
```

Uso de los Comandos de Configuración de RIP

no auto-summary

Deshabilita la sumarización automática de rutas. Al deshabilitar la sumarización automática de rutas, se habilita CIDR, permitiendo que RIP anuncie la información de enrutamiento de todas las subredes y hosts en el SecureStack C3.

Se ejecuta desde el modo de configuración de router.

`no auto-summary`

`auto-summary`

Para deshabilitar la sumarización de rutas automática de RIP:

```
diplomado(su)->router(Config)#router rip
diplomado(su)->router(Config-router)#no auto-summary
```

Uso de los Comandos de Configuración de RIP

split–horizon poison

Habilita o deshabilita el modo horizonte dividido y envenenamiento inverso (split horizon poison-reverse) para los paquetes RIP. La forma “no” de éste comando deshabilita el horizonte dividido y envenenamiento inverso .

Se ejecuta desde el modo de configuración de router.

```
split–horizon poison
no split–horizon poison
```

Las líneas muestran como habilitar split horizon y poison reverse:

```
diplomado(su)->router(Config)#router rip
diplomado(su)->router(Config-router)#split–horizon poison
```

Uso de los Comandos de Configuración de RIP

passive–interface

Previene que RIP transmita paquetes de actualización en una interfaz. La forma “no” de éste comando deshabilita una interfaz pasiva. Este comando no evita que RIP monitoree actualizaciones en una interfaz.

Se ejecuta desde el modo de configuración de router.

`passive–interface vlan` *`vlan-id`*

`no passive–interface vlan` *`vlan-id`*

Este ejemplo muestra como establecer la VLAN 10 como una interfaz pasiva. Las actualizaciones de RIP no serán transmitidas en ésta interfaz:

```
diplomado(su)->router(Config)#router rip
diplomado(su)->router(Config-router)#passive-interface vlan 10
```

Uso de los Comandos de Configuración de RIP

receive–interface

Permite que RIP reciba paquetes de actualización en una interfaz. La forma “no” de éste comando deniega la recepción de actualizaciones RIP. Por defecto, la recepción está habilitada en todas las interfaces de enrutamiento. Este comando no afecta el envío de actualizaciones RIP en una interfaz.

Se ejecuta desde el modo de configuración de router.

`receive–interface vlan` *`vlan-id`*

`no receive–interface vlan` *`vlan-id`*

Para denegar la recepción de actualizaciones RIP en la interfaz VLAN 10, el ejemplo muestra la ejecución de los comandos:

```
diplomado(su)->router(Config)#router rip
diplomado(su)->router(Config-router)# no receive-interface vlan 10
```

Uso de los Comandos Show para Verificar la Operación de RIP

Uso de los Comandos Show para Verificar la Operación de RIP

show ip rip

Muestra el estado de la configuración del router RIP.

Se ejecuta desde el modo EXEC privilegiado.

`show ip rip`

En la ejecución de este comando tenemos :

```
diplomado(su)->router#show ip rip
RIP Admin Mode                          - Enable
Split Horizon Mode                      - Poison Reverse
Auto Summary Mode                       - Disable
passive-interface vlan                  - 10
receive-interface vlan                  - 100
diplomado(su)->router#
```

Uso de los Comandos Show para Verificar la Operación de RIP

show ip rip vlan

Muestra la configuración RIP por interfaz.

Se ejecuta desde el modo EXEC privilegiado.

show ip rip vlan *vlan-id*

En la ejecución de este comando tenemos:

```
diplomado(su) ->router#show ip rip vlan 10
                             Send        Receive       RIP
Vlan       IP Address       Version      Version      Mode
---------  ---------------  -----------  -----------  ---------
Vlan 10    10.10.100.1      None         None         Disable
diplomado(su)->router#show ip rip vlan 20
                             Send        Receive       RIP
Vlan       IP Address       Version      Version      Mode
---------  ---------------  -----------  -----------  ---------
Vlan 20    10.16.16.1       RIP-2        RIP-2        Enable
```

Balanceo de Cargas con RIP

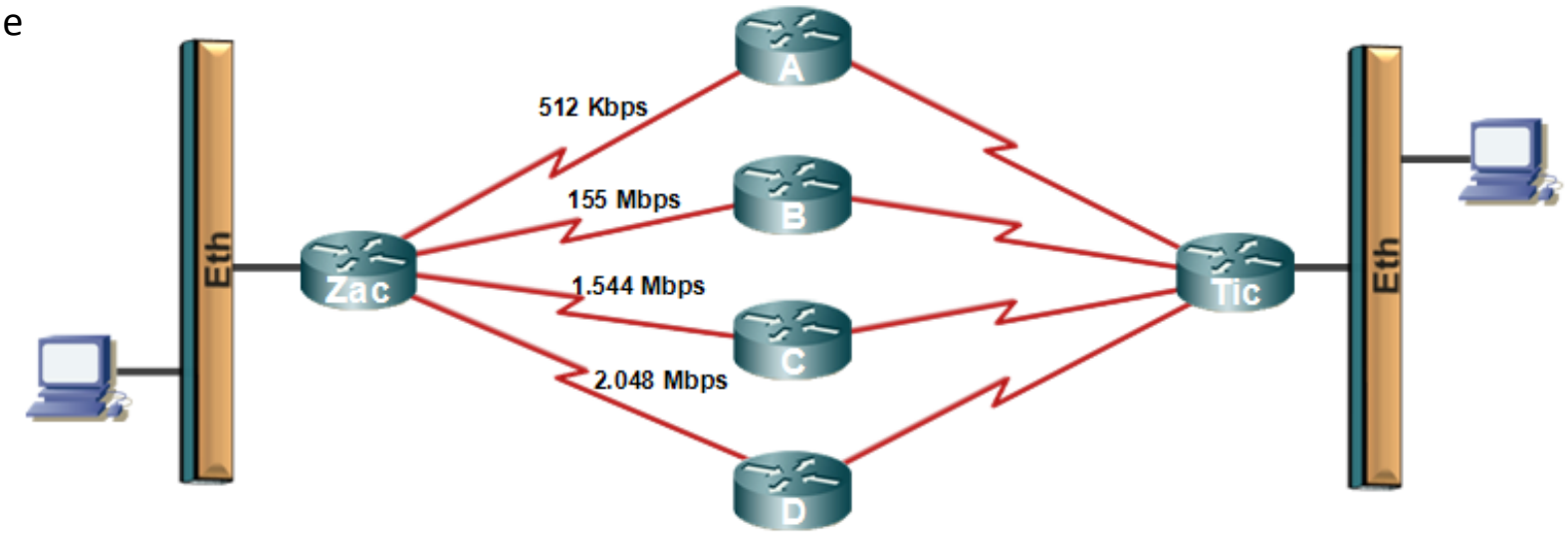
Balanceo de Cargas con RIP

RIP tiene la capacidad de balancear la carga hasta con seis rutas de igual costo.

El balanceo de cargas predeterminado es con cuatro rutas.

El balanceo realizado por RIP se conoce como balanceo de cargas "por turnos" o "en cadena" (*round robin*). Significa que RIP, envía los paquetes por turnos a través de las rutas paralelas.

Como la métrica de RIP es el número de saltos, no se toma en cuenta la velocidad de los e



Diferencias entre RIP v1 y RIP v2.

Diferencias entre RIP v1 y RIP v2

Comparación entre RIP v1 y RIP v2

Característica	RIP v1	RIP v2
Tipo de Protocolo	Classful	Classless
Soporte de VLSM	No	Si
Envía máscaras de subred en las actualizaciones	No	Si
Tipo de direccionamiento	Broadcast	Multicast
Soporta sumarización de rutas manual	No	Si
Soporte de autenticación	No	Si
Definido en…	RFC 1058	RFCs 1721, 1722 y 2453

Diferencias Entre RIP v1 y RIP v2

Referencias y Fuentes de Información

Guía del Primer Año. CCNA 1 y 2.

Tercera Edición.

Cisco Press.

IP Routing Fundamentals

Cisco Press - Macmillan Publishing USA.

SecureStack C3 Stackable Switches

Configuration Guide

Firmware Version 1.1.xx

Enterasys Networks

Enterasys Matrix™
DFE-Platinum and Diamond Series
Configuration Guide
Firmware Version 1.1.xx
Enterasys Networks

CCNP Routing Study Guide.

Cisco Systems – Sybex Inc.

www.sybex.com

Internetworking Technologies Handbook

Cisco Systems.

http://www.microsoft.com/technet/prodtechnol/windows2000serv/reskit/intwork/inae_ips_vzbs.mspx?mfr=true