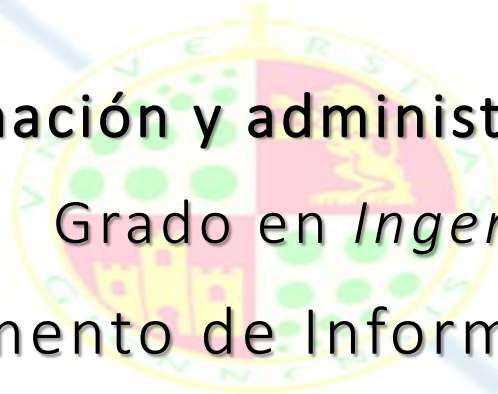


Protocolos de enrutamiento I

Programación y administración de redes - Semana 9

Grado en *Ingeniería Informática*

Departamento de Informática. Universidad de Jaén



Objetivos

General

Obtener una visión general sobre el funcionamiento de la técnica de enrutamiento IP y conocer los algoritmos de enrutamiento interno

Específicos

- Saber en qué consiste el **enrutamiento IP**
- Interpretar la información alojada en una **tabla de enrutamiento**
- Diferenciar entre los enrutamientos **interno y externo**
- Identificar las **características generales** de los algoritmos de enrutamiento
- Conocer el funcionamiento del **protocolo RIP**

Enrutamiento - Introducción

Definición

El enrutamiento IP es el encargado de reenviar los datagramas desde el host de origen hasta el de destino a través de los equipos de conmutación

Conceptos

- Son los **equipos de conmutación**, principalmente *routers*, los más implicados en el proceso de enrutamiento
- Cada equipo cuenta con una **tabla de enrutamiento** que le permite saber qué direcciones puede alcanzar por cada una de sus salidas
- Las tablas de rutas son mantenidas por protocolos de la capa de aplicación, pero **las utiliza el protocolo IP** al nivel de la capa de red
- Nos interesa conocer la **estructura** de las tablas de enrutamiento y los **protocolos** que las mantienen

Protocolos de enrutamiento

Finalidad

Crear y mantener las tablas de enrutamiento en los router para que el protocolo IP pueda realizar el trabajo de reenvío

Detalles

– Tipos de protocolos

- **Internos:** se usan entre redes pertenecientes a un sistema autónomo
- **Externos:** facilitan la comunicación entre redes de distintos sistemas autónomos

RIP
Routing Information Protocol

– Protocolos de enrutamiento interno

- **RIP:** sencillo, para redes pequeñas
- **OSPF:** complejo, para redes más grandes

OSPF
Open Shortest Path First

– Protocolos de enrutamiento externo

- **BGP:** compuesto de dos subalgoritmos llamados iBGP y eBGP

BGP
Border Gateway Protocol

Sistemas autónomos

Definición

Un SA (Sistema Autónomo) es un conjunto de routers bajo un control técnico y administrativo común, por ejemplo un operador o gran empresa
















Conceptos

- Cada SA se considera una **gran WAN** formada por otras WAN y LAN y se identifica de manera única mediante un **ASN** (*AS Number*)
- Los SA tienen **interconexiones con otros SA** vecinos, facilitando el intercambio de paquetes entre ellos
- Cada SA es responsable de un **conjunto de direcciones** que administra entre sus clientes
- Los **clientes de un SA** pueden ser otros SA más pequeños, que dependen de él, o bien directamente empresas y personas
- **Internet** actualmente es una red formada por todos los SA existentes en el mundo

Sistemas autónomos - Actividad

Conceptos

- Accede a asrank.caida.org para conocer el ranking de SA en la actualidad
- Busca el **ASN 15810**:
 - ¿Cuántos clientes tiene?
 - ¿Qué tipo de empresa es?
 - ¿De qué otro ASN depende?
- Busca el **ASN 12956**:
 - ¿Qué tipos de clientes tiene?
 - ¿Cuántas direcciones gestiona?

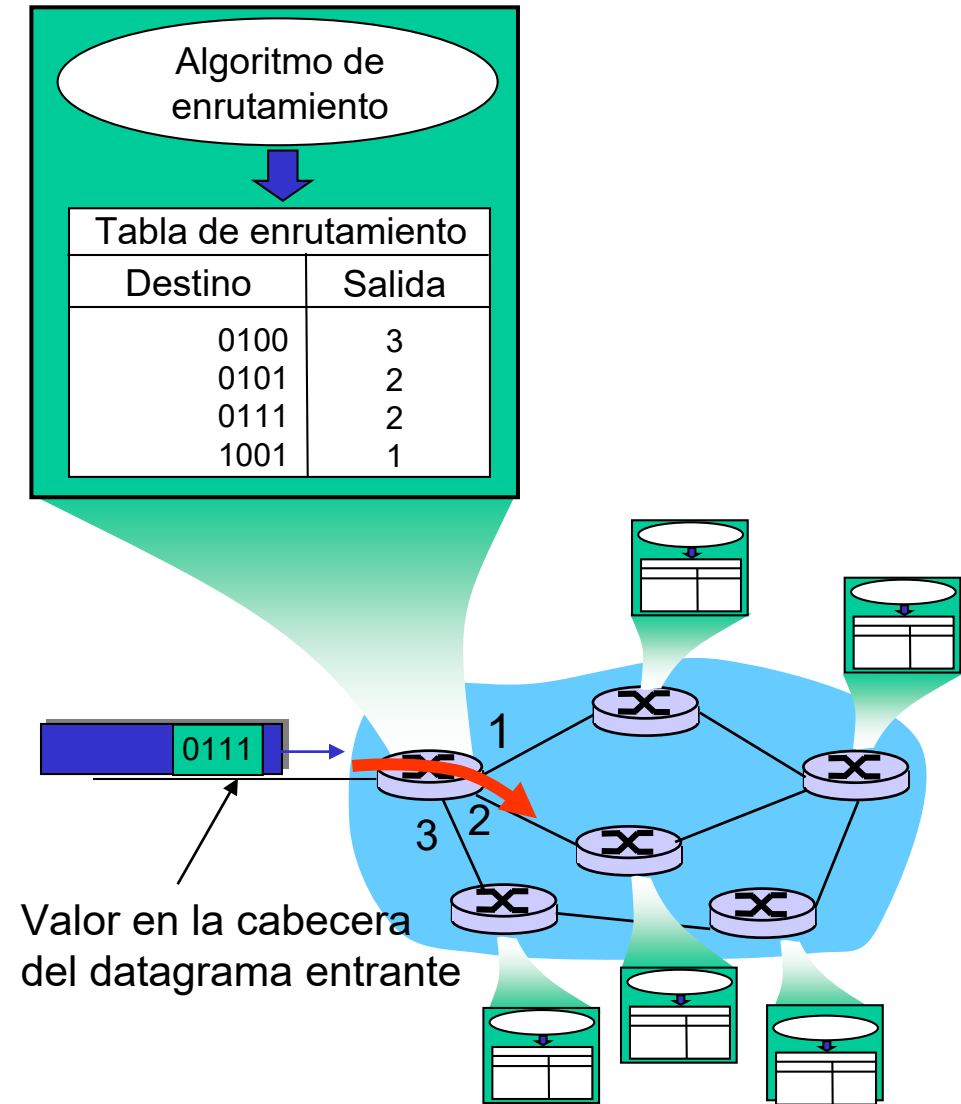
ASN name or number					search
1	2	3	4	..	1779
AS Rank ▲	AS Number ▼	Organization		cone size (ASes) ▼	
1	3356	Level 3 Parent, LLC		46698	
2	3257	GTT Communications Inc.		35756	
3	1299	Telia Company AB		35422	
4	174	Cogent Communications		30142	
5	2914	NTT America, Inc.		20538	
6	6762	Telecom Italia S.p.A.		18838	
7	6939	Hurricane Electric LLC		15495	
8	6453	TATA COMMUNICATIONS ...		14624	
9	6461	Zayo Bandwidth		14140	
10	3549	Level 3 Parent, LLC		13209	
11	3491	PCCW Global, Inc.		12580	
12	1273	Vodafone Group PLC		8138	
13	9002	RETN Limited		7155	
14	4637	Telstra International Limited		5467	
15	12956	TELEFONICA GLOBAL SOL...		4259	

Enrutamiento de datagramas IP



Enrutamiento - Conceptos

- **Enrutamiento:** reenvío de paquetes que llegan por unas líneas de entrada hacia unas líneas de salida usando la tabla de enrutamiento
- **Tabla de enrutamiento:** tabla que contiene las asociaciones para obtener la línea de salida por la que se enviará un paquete dado. Al menos tiene que tener las columnas **Red destino** y **Línea de salida**. **Interpretación:** para alcanzar este destino se te enviará por esta línea de salida
- **Estructura:** el enrutamiento hoy en día suele ser jerárquico, es decir no se guardan destinos que son *host* concretos, sino redes en general o destinos por defecto
- **Algoritmo:** el algoritmo de enrutamiento tiene por objetivo obtener una tabla de rutas para un *router*
 - En ciertos contextos también es común administrar o rellenar estas tablas manualmente



Enrutamiento - Funcionamiento

Esquema general

- El enrutamiento de un datagrama se va haciendo **router a router** ya que el protocolo IP no es orientado a conexión
- Cada *router* decide en función de su tabla de enrutamiento cuál es el **siguiente router** al que se envía
- El datagrama no se altera en todo su recorrido a excepción del **campo TTL** o **salvo que haya fragmentación** o enmascaramiento

Proceso para encaminamiento

1. Se extrae del datagrama la dirección IP de destino, a la que denominamos DD_D
2. Para cada entrada de la tabla de enrutamiento (que tiene una dirección de red destino DR_T y una máscara de red asociada M_T):
 - Se hace un AND bit a bit entre DD_D y la M_T obteniéndose el identificador de red (de DD_D) DR_D
 - Si $DR_D = DR_T$ el datagrama se encaminará por donde indique la tabla
 - Si $DR_D \neq DR_T$ se pasa a la siguiente entrada
3. Si no se encuentra coincidencia en la tabla de rutas se envía por la **ruta por defecto**. Si no existe esta ruta se genera un error

Enrutamiento - Ejemplo 1

Cabecera del datagrama enviado

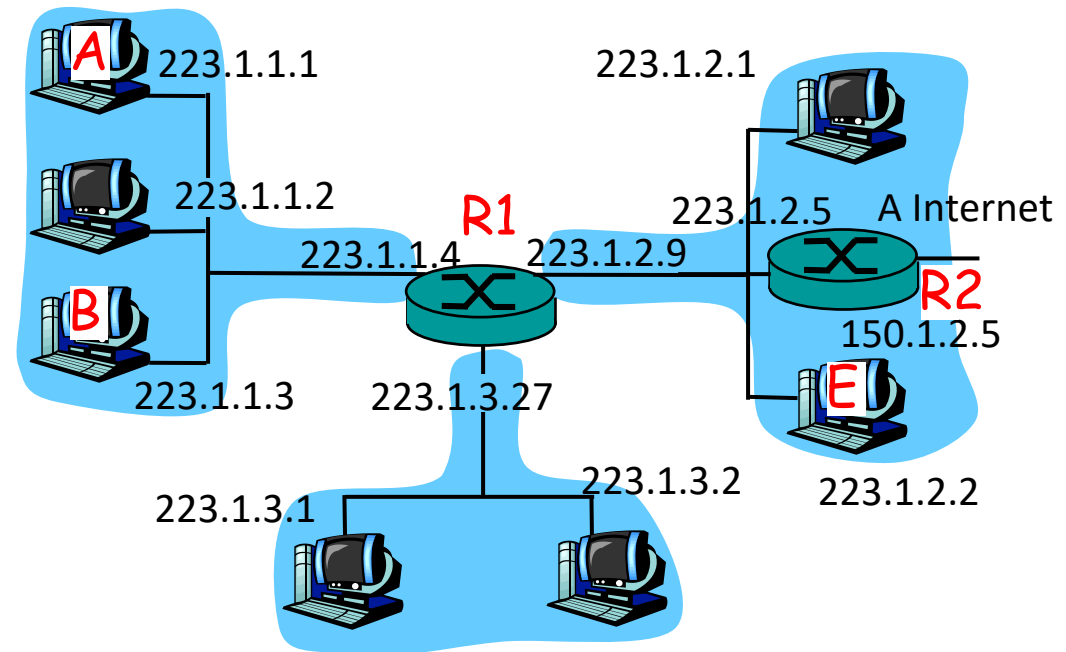
Campos misc.	223.1.1.1	223.1.1.3	Datos
--------------	-----------	-----------	-------

Tabla de enrutamiento en el *host A*

Destino	Router	Máscara	Interfaz
223.1.1.0	*	255.255.255.0	eth0
default	223.1.1.4	0.0.0.0	eth0

A envía un datagrama a B

1. Consultar la dirección de red de **B** en la tabla de enrutamiento
2. La dirección de **B** se encuentra en la misma red que **A**
3. La capa de enlace enviará el datagrama directamente a **B** dentro de un trama de la capa de enlace:
 - **B** y **A** están directamente conectados
4. El *host B* recibe y procesa el datagrama



Enrutamiento - Ejemplo 2 - Paso 1

Cabecera del datagrama enviado

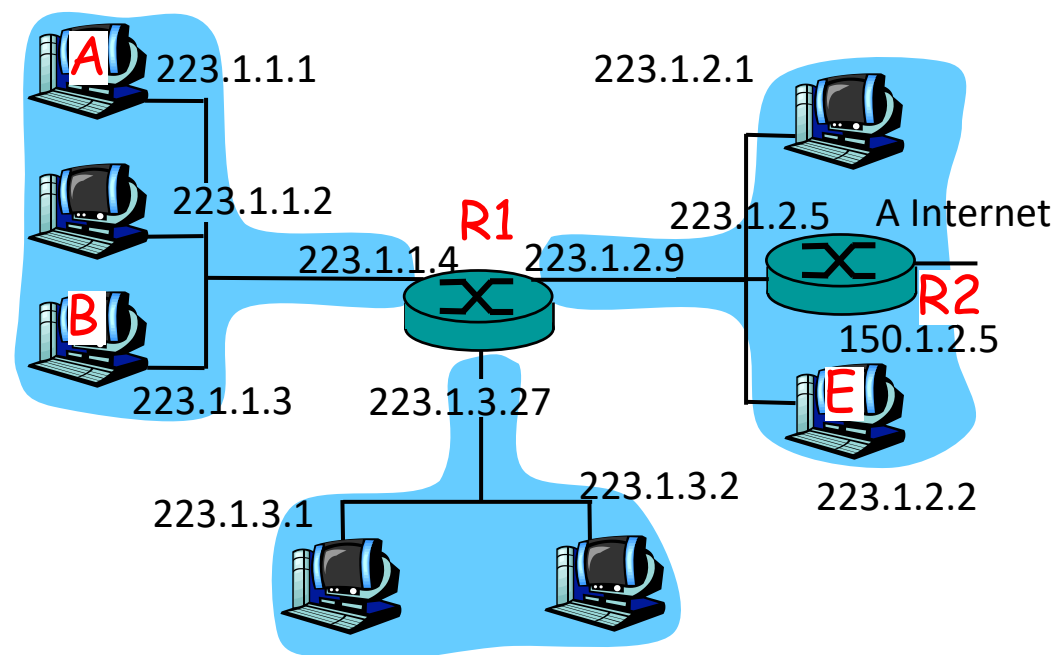
Campos misc.	223.1.1.1	223.1.2.2	Datos
--------------	-----------	-----------	-------

Tabla de enrutamiento en el *host A*

Destino	Router	Máscara	Interfaz
223.1.1.0	*	255.255.255.0	eth0
default	223.1.1.4	0.0.0.0	eth0

A envía un datagrama a E

1. Consultar la dirección de red de **E** en la tabla de enrutamiento
2. El *host E* está en una red distinta
3. Tabla de enrutamiento: el próximo *router* a **E** es **223.1.1.4**
4. La capa de enlace envía el datagrama al *router* **223.1.1.4** encapsulado en una trama de la capa de enlace
5. El datagrama llega al *router* **R1** por su interfaz **223.1.1.4**



Enrutamiento - Ejemplo 2 - Paso 2

Cabecera del datagrama enviado

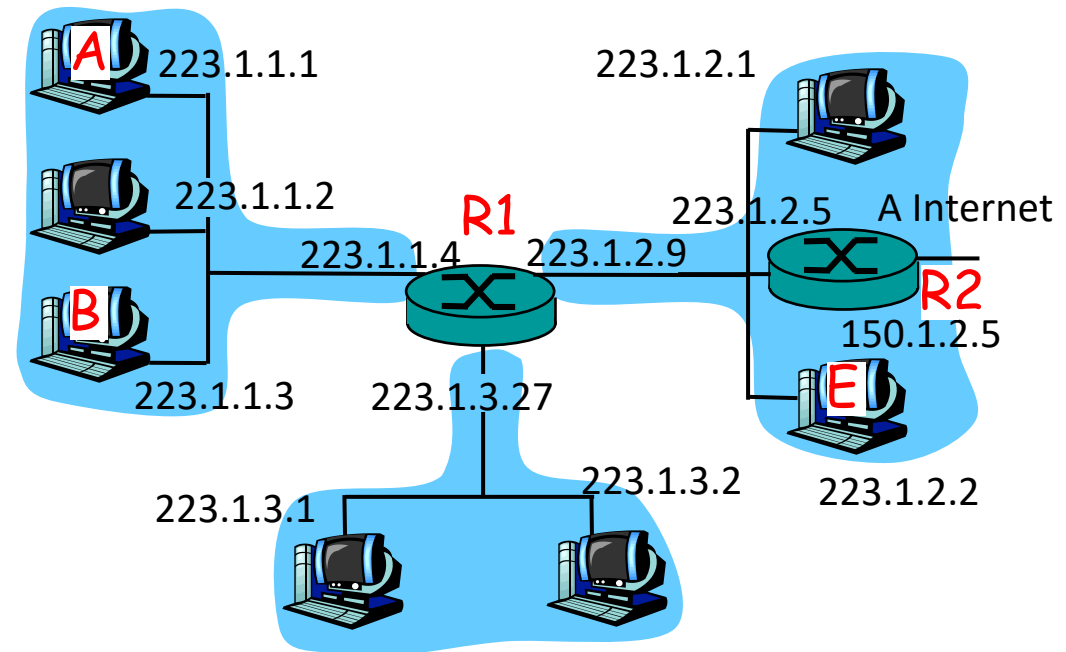
Campos misc.	223.1.1.1	223.1.2.2	Datos
--------------	-----------	-----------	-------

A envía un datagrama a E

6. El datagrama llega a 223.1.1.4 camino de 223.1.2.2
7. Consultar la dirección de red de E en la tabla de enrutamiento del *router R1*
8. El *host E* está en la misma red que la interfaz 223.1.2.9 del *router*
9. La capa de enlace envía el datagrama al *router* 223.1.2.2 encapsulado en una trama de la capa de enlace por la interfaz 223.1.2.9 de R1
10. El datagrama llega a 223.1.2.2

Tabla de encaminamiento para el router R1

Destino	Router	Máscara	Interfaz
223.1.1.0	*	255.255.255.0	eth0
223.1.2.0	*	255.255.255.0	eth1
223.1.3.0	*	255.255.255.0	eth2
default	223.1.2.5	0.0.0.0	eth1



Enrutamiento - Actividad

Cabecera del datagrama enviado

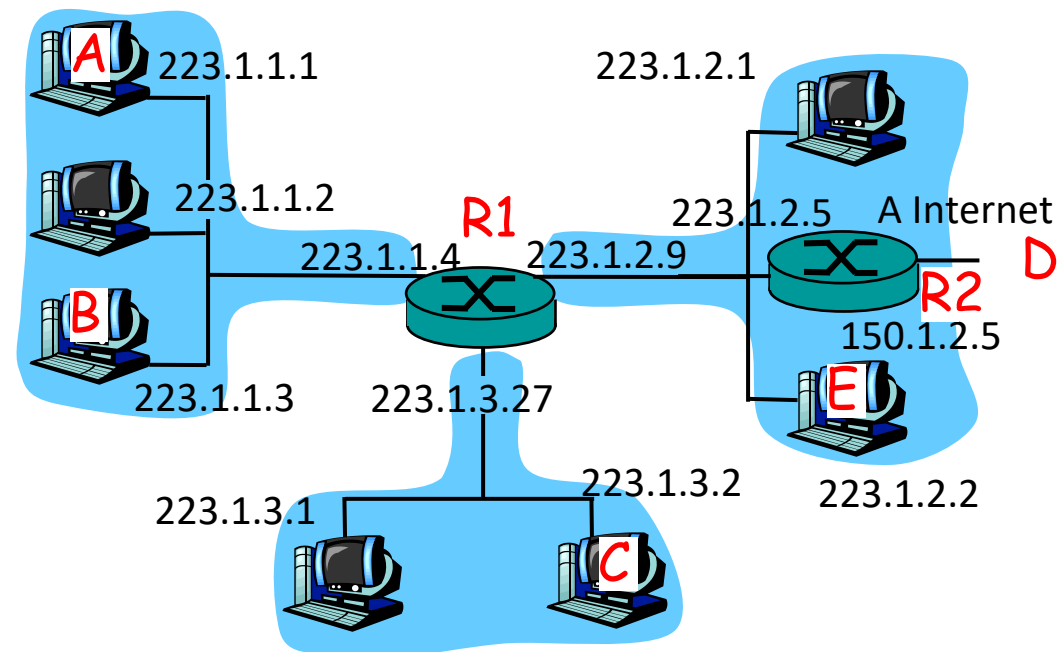
Campos misc.	223.1.1.1	223.1.3.2	Datos
--------------	-----------	-----------	-------

A envía un datagrama a C/D

1. Completa la tabla de rutas que corresponde al *router R2* deduciendo sus entradas a partir del esquema de red mostrado a la derecha
2. Enumera los pasos que habría de seguir el datagrama enviado desde **A** para llegar al host **C**
3. Haz lo mismo pero en este caso para enviar el datagrama a un equipo **D** externo a esta red, ubicado en algún lugar de Internet

Tabla de encaminamiento para el router R2

Destino	Router	Máscara	Interfaz



Algoritmos y protocolos de enrutamiento

Perspectiva general



Algoritmos de enrutamiento - Función y propiedades

Función

Recuerda que el objetivo de estos algoritmos es crear y mantener las tablas de rutas que usan los router

- **Principio de optimalidad:** “Si **B** está en la ruta óptima de **A** a **C**, entonces el camino óptimo de **B** a **C** está incluido en esa ruta”
- **Consecuencia:** de esto se deriva que todas las rutas óptimas para llegar a un nodo de la red forman un árbol con raíz en ese nodo, al cual se denomina *spanning tree*
- **Propiedades** de un algoritmo de enrutamiento:
 - **Robustez:** el algoritmo debe adaptarse a los problemas de la red
 - **Estabilidad:** convergencia a un equilibrio en cuanto al tráfico
 - **Sencillez:** implementación ligera y rápida
 - **Imparcialidad:** no debe favorecer unos paquetes sobre otros
 - **Optimalidad:** debe seguir la vía más óptima en cada caso
- Las propiedades pueden entrar en **conflicto**, p.e.: las dos últimas

Algoritmos de enrutamiento - Clasificación

- Según su actualización pueden ser:
 - **Estáticos:** las tablas de rutas se calculan a priori y no cambian
 - **Dinámicos o adaptativos:** las tablas de rutas se van recalculando
- Para el cálculo de las tablas de rutas pueden utilizar **información:**
 - **Global:** conocen toda la topología de la red e incluso pueden conocer capacidad de los enlaces, carga de estos, etc. Pueden calcular rutas óptimas. Ejemplo: OSPF
 - **Local:** conocen solo información local como vecinos e información que le envían estos. Las rutas no suelen ser óptimas y se pueden producir problemas. Ejemplo: RIP
- Existen múltiples **algoritmos:**
 - Algoritmo del camino más corto (Dijkstra, Bellman-Ford, etc.)
 - Algoritmo de aprendizaje hacia atrás
 - Algoritmo de la patata caliente
 - Algoritmo de inundación
 - Algoritmo distribuido o vector-distancia (RIP)
 - Algoritmo del estado del enlace (OSPF)
 - Algoritmo óptimo o *spanning tree*

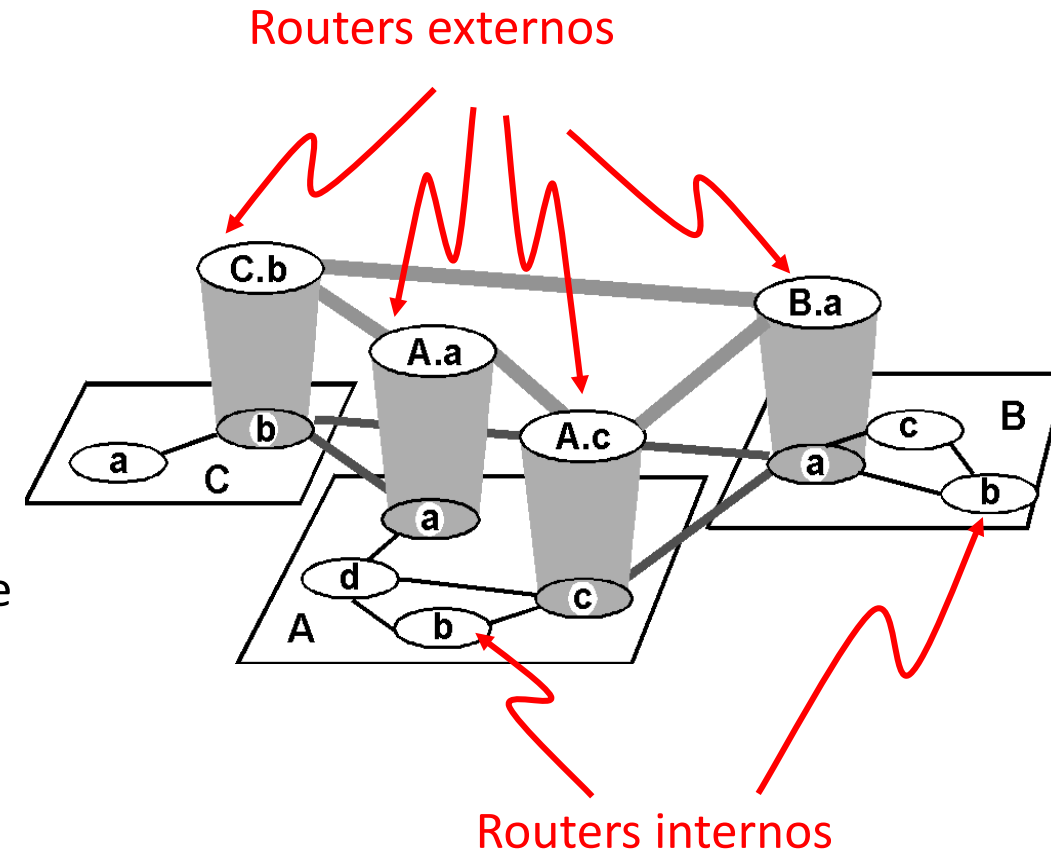
Protocolos de enrutamiento en Internet

Dos niveles de enrutamiento

- **Interno a un SA:** el administrador es el responsable de la elección del algoritmo de enrutamiento en la red. Hay varias alternativas:
 - RIP: *Routing Information Protocol*
 - OSPF: *Open Shortest Path First*
 - HELLO
- **Externo, entre diferentes SA:** hay que garantizar la comunicación e intercambio de rutas entre los SA, por lo que solo se considera un algoritmo:
 - BGP: *Border Gateway Protocol*

Ubicación en la arquitectura TCP/IP

- **Operación:** los algoritmos de enrutamiento operan en la capa de aplicación, siendo su objetivo construir y mantener las tablas de rutas. En el RFC 1812 (<https://tools.ietf.org/html/rfc1812>) se detalla la ubicación de estos protocolos
- **Mensajes:** los paquetes de datos de estos protocolos se envían mediante TCP, UDP o bien directamente encapsulados en un datagrama IP, según los casos



El protocolo RIP

Routing Information Protocol



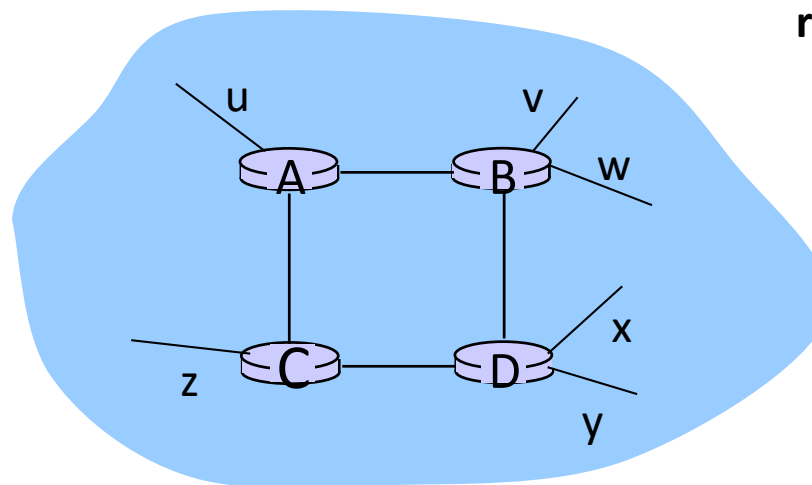
El protocolo RIP

Introducción

Algoritmo de tipo vector de distancias incluido en BSD-UNIX en 1982

Características

- Medida de la distancia hacia un destino: **número de saltos**
- Cada enlace entre *routers* tiene **coste 1** (máximo = 15 saltos)
- **Anuncio**: vectores de distancias intercambiados entre los vecinos cada 30 segundos
 - Cada anuncio lista hasta **25 redes** de destino
- Un *host* construye su tabla de enrutamiento eligiendo al **vecino por el que tarda menos** en llegar a un destino determinado



Anuncio difundido desde router A a vecinos

<u>Subred</u>	<u>Salto</u>
u	1
v	2
w	2
x	3
y	3
z	2

El protocolo RIP - Ejemplo - Paso 1

Situación inicial

El router D tiene en su tabla de rutas la configuración actual de la red

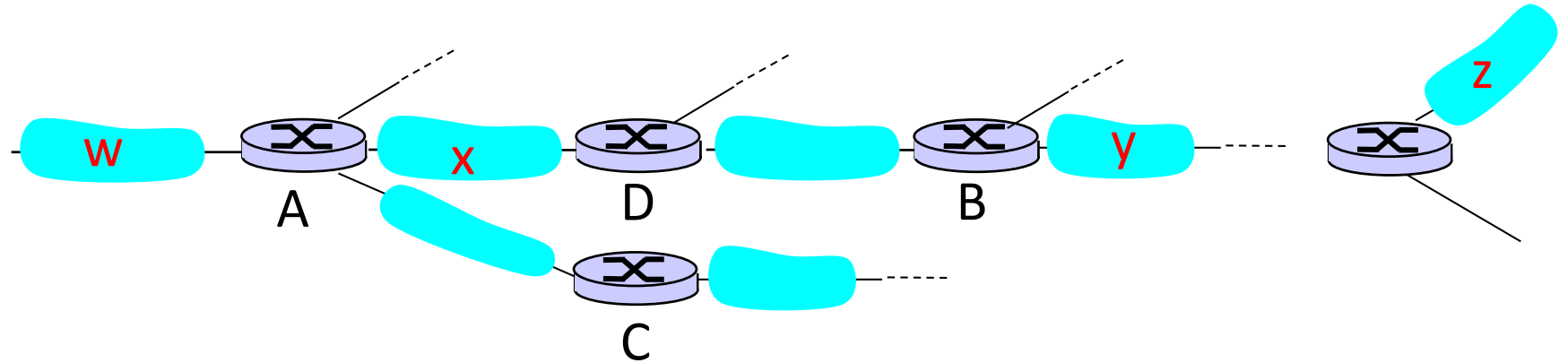


Tabla de enrutamiento en el router D

- Puede alcanzar la red **x** directamente
- Puede alcanzar la red **w** a través de A en 2 saltos
- Puede alcanzar la red **y** a través de B en 2 saltos
- Puede alcanzar la red **z** a través de B en 7 saltos

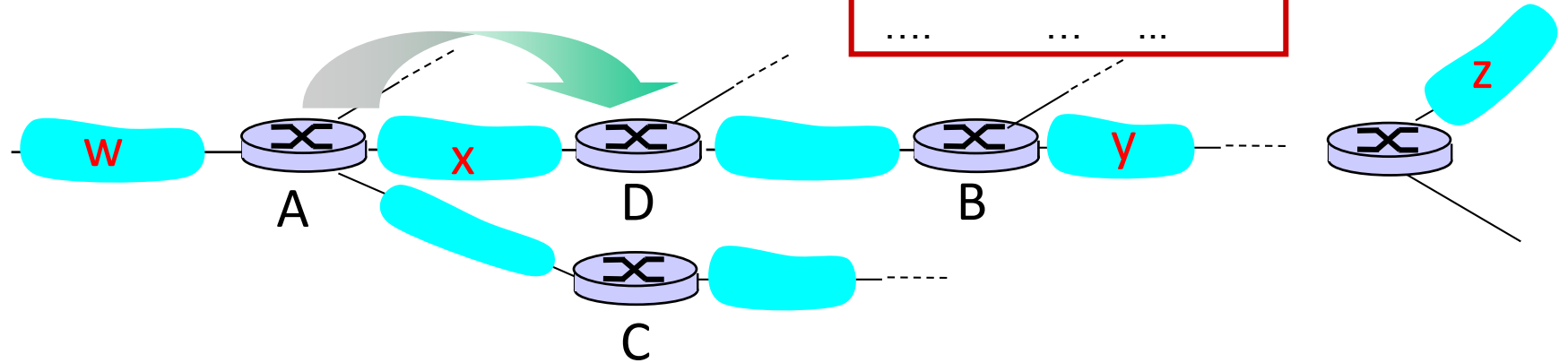
Destino	Sig. router	# Saltos
w	A	2
y	B	2
z	B	7
x	--	1
....

El protocolo RIP - Ejemplo - Paso 2

Se recibe un anuncio

El *router* D recibe un anuncio del *router* A con

Dest	Sig. r	Salto
w	-	1
x	-	1
z	C	4
....



El *router* **A** hace saber a **D** lo siguiente:

- Que puede llegar a **w** en 1 salto
- Que puede llegar a **x** en 1 salto
- Que puede llegar a **z** en 4 saltos a través del router C

Tabla de enrutamiento en el router D

Destino	Sig. router	# Saltos
w	A	2
y	B	2
z	B	7
x	--	1
....

El protocolo RIP - Ejemplo - Paso 3

Se actualiza la tabla de rutas

El *router* D evalúa el anuncio de A e introduce en su tabla de rutas los cambios apropiados

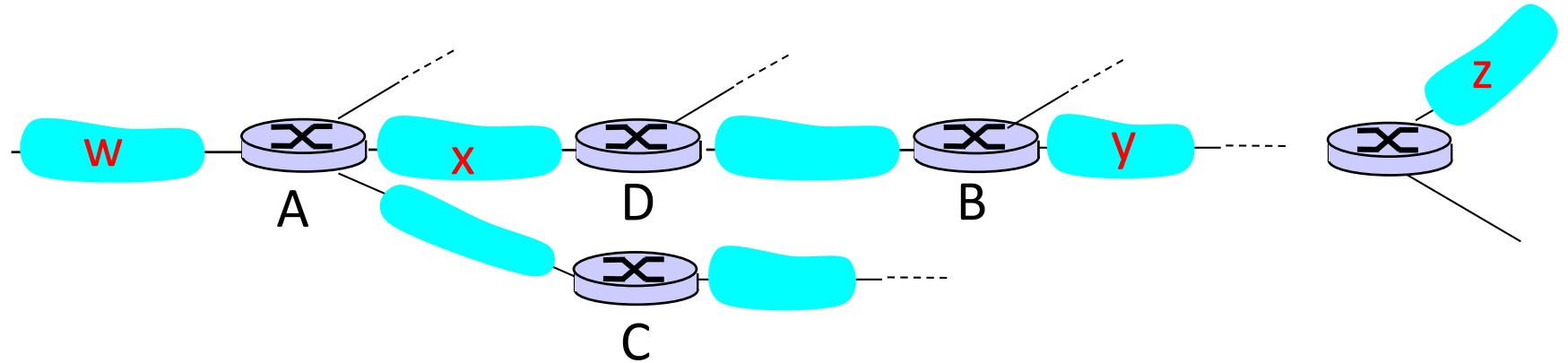


Tabla de enrutamiento en el router D

Con la información recibida de **A** el *router* **D** sabe que:

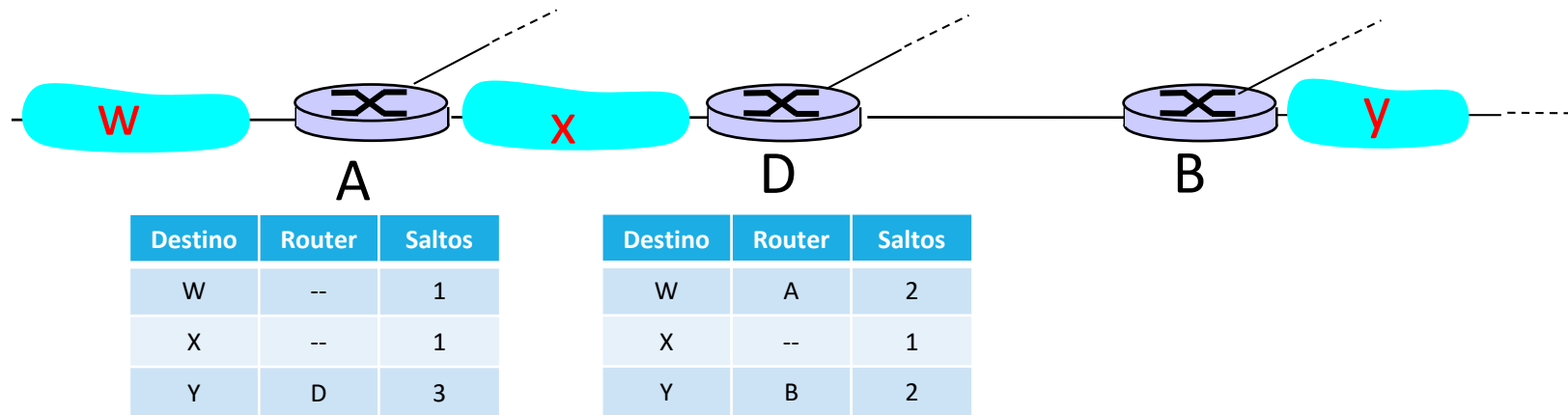
- Podría llegar a w en 2 saltos: no se mejora
- Podría llegar a x en 2 saltos: se empeora
- Podría llegar a z en 5 saltos: se mejora respecto a 7

D actualiza la entrada de **z** para ir a través de **A**

Destino	Sig. router	# Saltos
w	A	2
y	B	2
z	B → A	7 → 5
x	--	1
....

El protocolo RIP - Actividad

Esquema y situación inicial de las tablas en A y D



A partir de esta situación:

- Suponer que el enlace existente entre **D** y **B** se interrumpe
- Transcurrido el tiempo establecido el *router* **A** envía su actualización a **D**. Actualizar la tabla de rutas de **D**
- Transcurrido el tiempo establecido el *router* **D** envía su actualización a **A**. Actualizar la tabla de rutas de **A**
- Repetir los dos pasos previos durante varias iteraciones. ¿Qué problema surge?

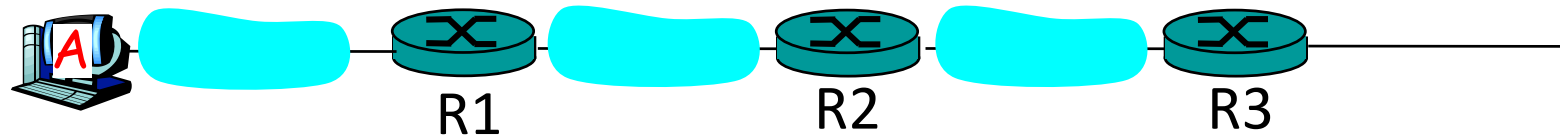
El protocolo RIP - Problemas y limitaciones

Convergencia lenta

- Los paquetes de aviso de cambios de topología en la red se propagan lentamente por el funcionamiento de RIP (periodicidad de los anuncios)
- Esta convergencia dependerá de la distancia máxima entre *routers*

Cuenta al infinito

- Inicialmente la distancia de **R1** a **A** es 1 desde **R2** es 2 (a través de **R1**) y desde **R3** es 3 (por **R2**)
- Supongamos que cae el enlace entre **R1** y la red donde está **A**
- En la próxima actualización de tablas, **R2** le manda a **R1** que puede llegar a **A** en 2 saltos
- **R1** actualizará su tabla y creará que puede llegar a **A** en 3 saltos (por **R2**), lo cual es falso
- En siguientes actualizaciones los *router* incrementarán su distancia hasta el “infinito”, determinándose que la red es inalcanzable



El protocolo RIP - Soluciones

- **Horizonte dividido** (*Split horizon*)
 - No se propaga información de encaminamiento correspondiente a un destino dado sobre la interfaz por la que atraviesa la ruta hacia dicho destino
- **Retención** (*Hold down*)
 - Se ignora durante un tiempo (60s) todos los mensajes de encaminamiento correspondientes a un destino dado para el que se detecta inaccesibilidad
- **Reversa envenenada** (*Poison reverse*)
 - Cuando quieres que no se acceda a través de ti a una red mandas una distancia muy grande o infinita. (distancia infinita = 16 saltos)
- Información del ***router* vecino de salida**
 - El nodo que envía un anuncio debe de incluir en la información de la ruta no solo el destino y la distancia, sino el *router* vecino de salida (Esto ya se hace en la versión 2)
- Implementar **multidifusión**
 - Para informar a todo el mundo del problema

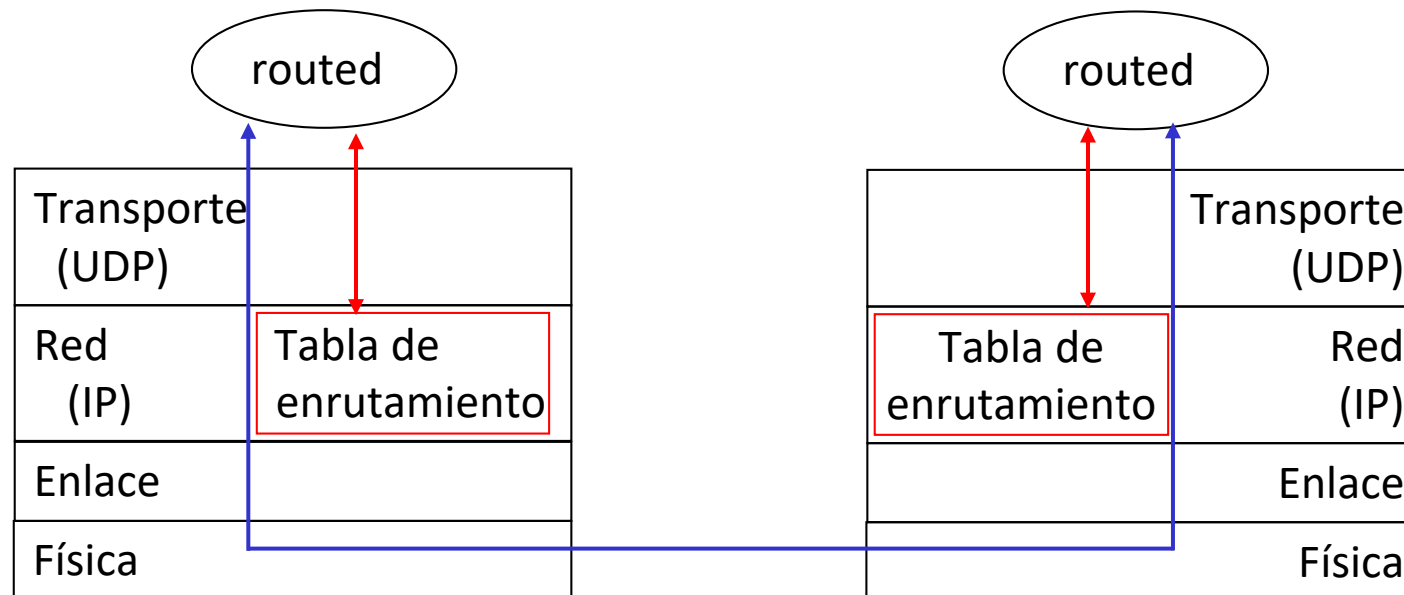
El protocolo RIP - Fallo y recuperación de enlace

- Si no se recibe ningún anuncio procedente de un vecino tras 180 segundos → el vecino/enlace se **da por muerto**:
 - **Invalidación de rutas**: las rutas a través del vecino se invalidan mediante el uso de la reversa envenenada (*poison reverse*) para prevenir los bucles de *ping-pong*
 - **Propagación**: los vecinos envían nuevos anuncios por turnos (si cambian las tablas), para intentar propagar la información de fallo lo más rápidamente por la red

El protocolo RIP - Funcionamiento

Procesamiento en los equipos

- En los *host* el protocolo RIP se administra mediante el demonio `routed` y opera en la **capa de aplicación**
- Los paquetes RIP son enviados en **segmentos UDP**, empleando el puerto 520 de este protocolo de transporte
- La versión de RIP actualmente en uso es RIPv2 que, respecto a la anterior, incluye envío con autenticación y multidifusión a la dirección **224.0.0.9**



Cuestiones clave de este tema



Cuestiones clave

Qué deberías saber

*Al inicio de este tema se planteaban unos objetivos específicos que deberían permitirte **responder a las siguientes cuestiones clave***

Cuestiones

- ¿Cómo funciona el enrutamiento IP, en la capa de red?
- ¿Cuál es la estructura de una tabla de enrutamiento?
- ¿Qué información usan los *host* y *router* para reenviar los datagramas a partir de la información almacenada en la tabla de rutas?
- ¿Cómo se agrupan los protocolos de enrutamiento atendiendo a sus características?
- ¿Cuál es el funcionamiento general del algoritmo RIP?
- ¿Qué limitaciones tiene RIP y qué técnicas se emplean para superarlas?

Material adicional

Descripción

Para ampliar tus conocimientos sobre los contenidos de esta semana te recomendamos que consultes los recursos indicados a continuación.

Recursos

- **Capítulo 5** La capa de red: el plano de control, del libro Redes de computadoras 7ED disponible en [formato digital](#) en la BUJA (recuerda identificarte para poder acceder a leerlo desde tu navegador), concretamente hasta la **sección 5.3**
- **TCP/IP Routing protocols** en el recurso electrónico [The TCP/IP Guide](#), en cuya primera parte encontrarás información general sobre el enrutamiento IP y detalles sobre el protocolo RIP
- **Requirements for IP Version 4 Routers** en el [RFC 1812](#), en cuyo índice encontrarás la ubicación de los protocolos de enrutamiento y, en particular, la [sección F-2](#) dedicada al protocolo RIP y sus aspectos específicos