

Camada de Internet



Prof. Dr. Bruno Rodrigues



Politica de alocação de endereços IP- ICANN



Política de alocação de endereços

A **ICANN** assim como a **IANA** são responsáveis por manter o registro de blocos alocados e não alocados de endereços **IPv4** e **IPv6**, **ASNs** e Nomes de domínios ; e é responsável por alocar grandes blocos de endereços IP e ASNs aos cinco RIRs de acordo com as políticas globais.

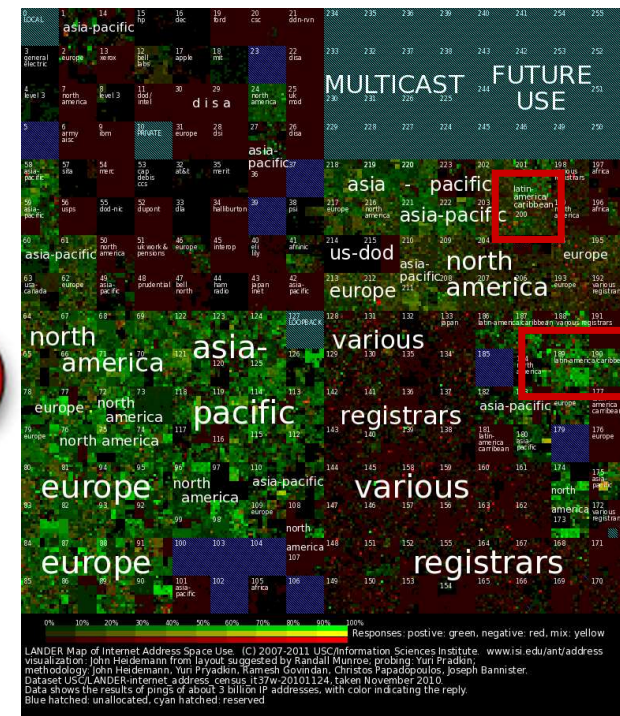
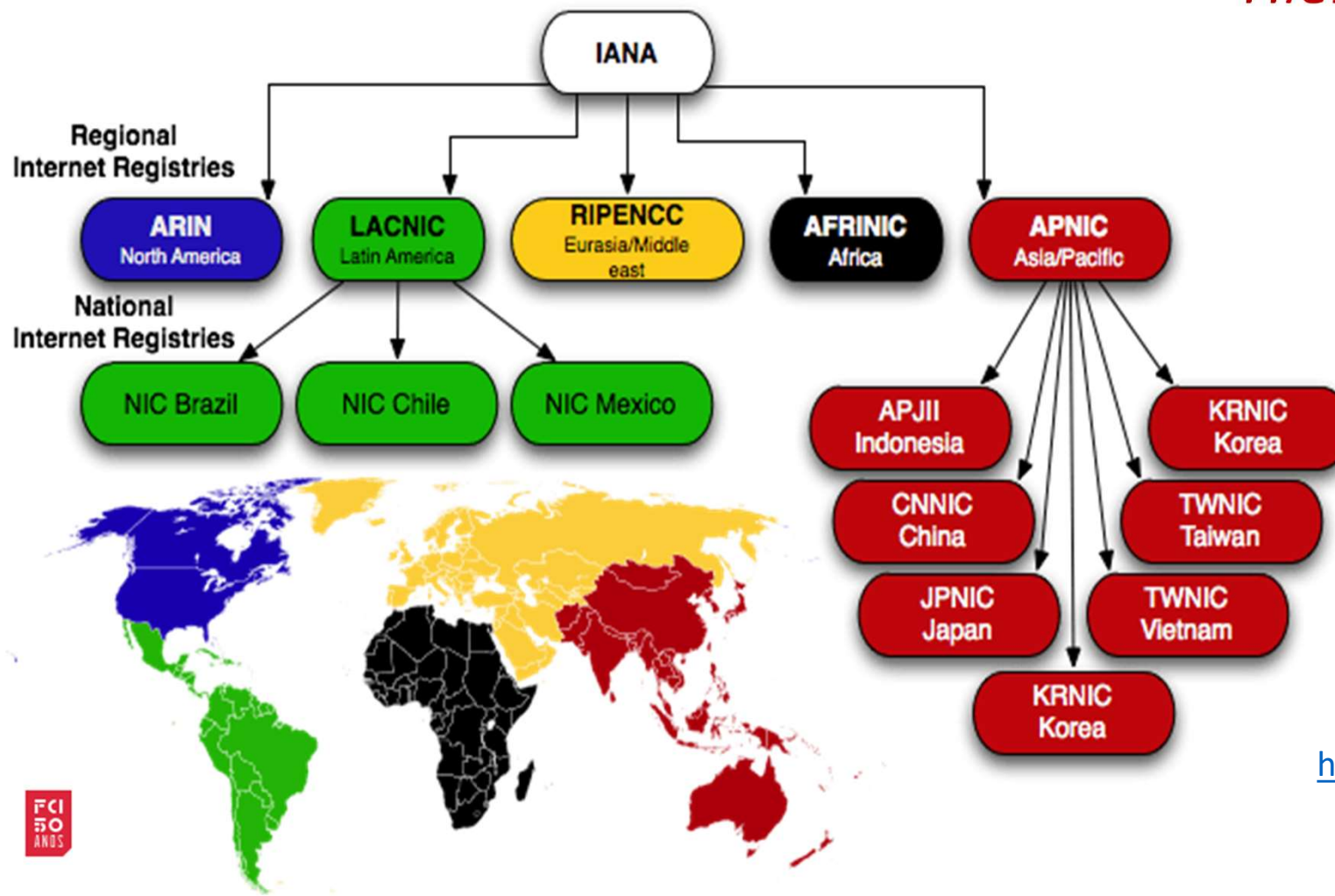


Regional Internet Registry (RIR)



Política de alocação de endereços

Hierarquia



<https://ant.isi.edu/address/index.html>

Algoritmos de Roteamento



Algoritmos de roteamento

- Em geral um hospedeiro está ligado diretamente a um roteador, o **roteador default** para esse hospedeiro.
- O problema de rotear um pacote do hospedeiro de origem até o hospedeiro de destino se reduz, claramente, ao **problema de direcionar o pacote do roteador de origem ao roteador de destino**.

Algoritmos de roteamento

E = conjunto de enlaces = $\{ (u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) \}$

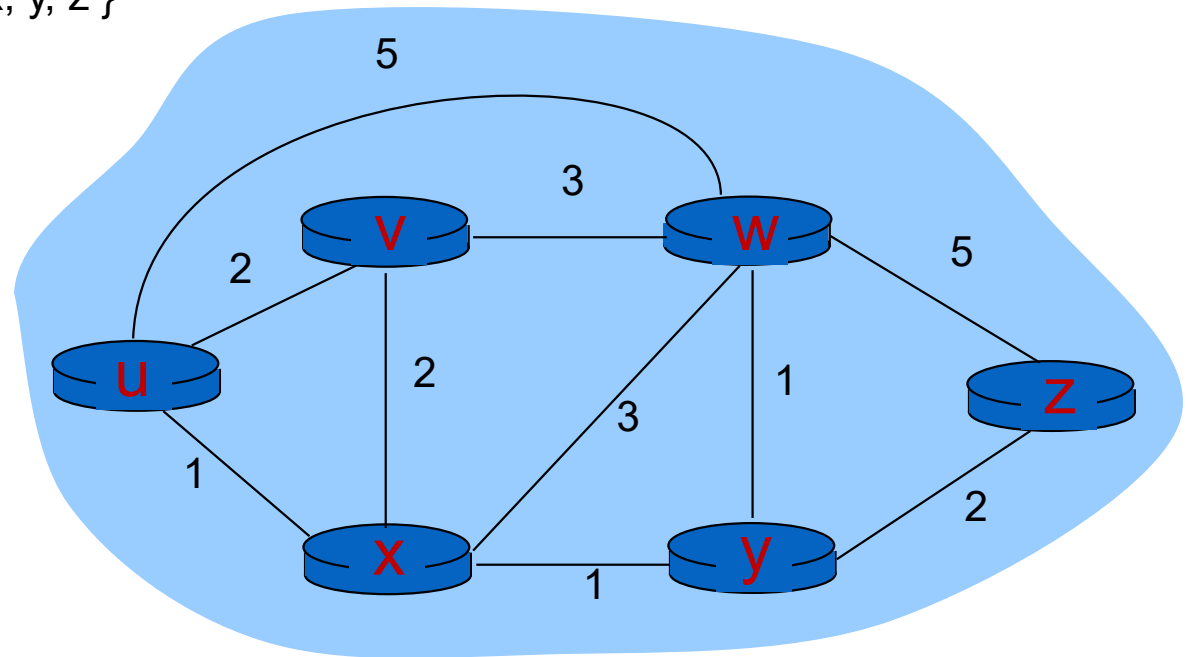
N = conjunto de roteadores = $\{ u, v, w, x, y, z \}$

Grafo: $G = (N,E)$

- $c(x,x')$ = custo do enlace (x,x')

- eg., $c(w,z) = 5$

- custo poderia ser sempre 1, ou inversamente relacionado à largura ou inversamente relacionado ao congestionamento



Algoritmos de roteamento

E = conjunto de enlaces = $\{ (u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) \}$

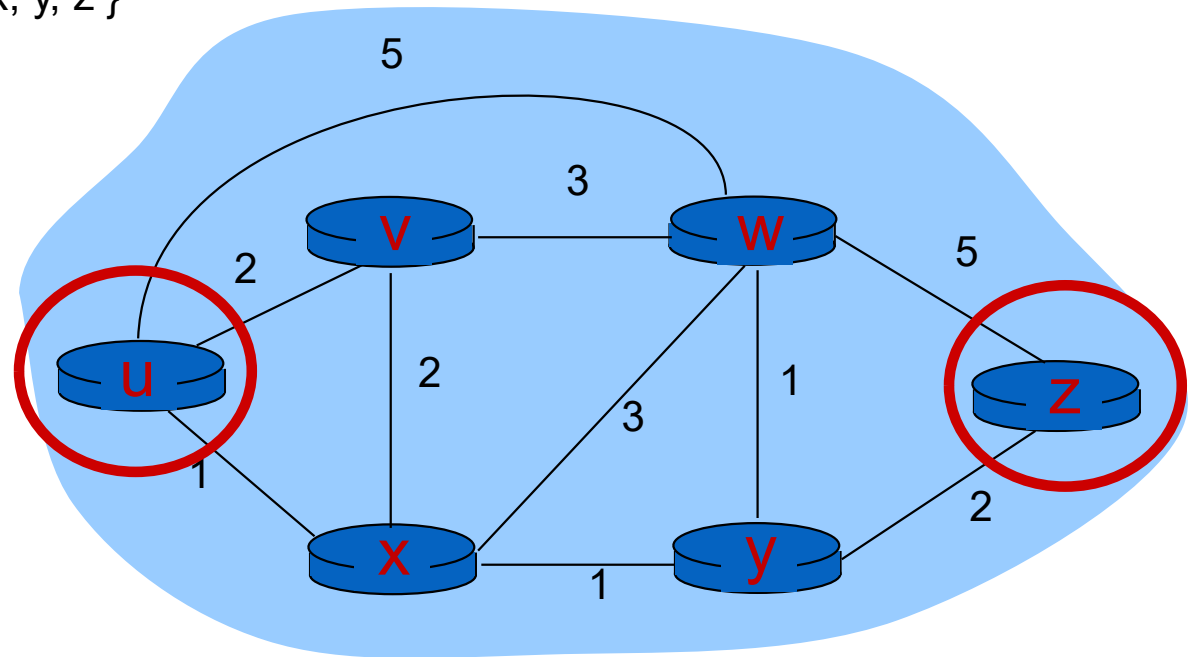
N = conjunto de roteadores = $\{ u, v, w, x, y, z \}$

Grafo: $G = (N,E)$

- $c(x,x')$ = custo do enlace (x,x')

- eg., $c(w,z) = 5$

- custo poderia ser sempre 1, ou inversamente relacionado à largura ou inversamente relacionado ao congestionamento



Pergunta: Qual é o caminho de menor custo entre u e z?

Isso é uma função do Algoritmo de Roteamento !!!

Algoritmos de roteamento

Estático ou dinâmico?

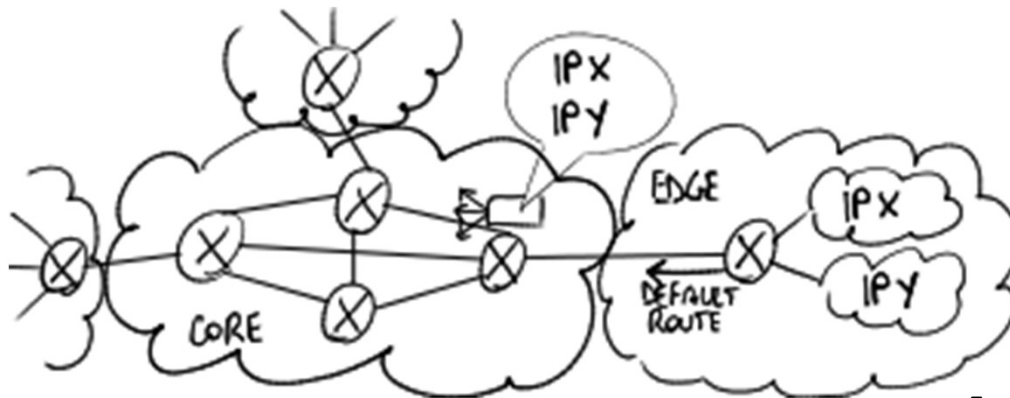
Classificação

Estático ou não adaptativos:

- ✓ Todas as rotas são calculadas previamente
- ✓ Rotas mudam lentamente com o tempo
- ✓ Não responde bem a falhas

Dinâmico ou adaptativos:

- ✓ Rotas mudam mais rapidamente
 - ❖ atualização periódica
 - ❖ em resposta a mudanças no custo do enlace e topologias



Fonte: wikibooks

Algoritmos de roteamento

Adaptativos ou Dinâmicos: As rotas remotas são aprendidas automaticamente através de mensagens trocadas entre os roteadores. Esses algoritmos alternam nas decisões de roteamento conforme mudanças na topologia, assim como no tráfego da rede.

Global:

- Todos os roteadores têm topologia completa, informação de custo do enlace
- Algoritmos de “estado do enlace”

Descentralizada:

- Roteador conhece vizinhos conectados fisicamente, custos de enlace para vizinhos
- Processo de computação iterativo, troca de informações com vizinhos
- Algoritmos de “vetor de distância”

Algoritmos de Estado do enlace

Link state(LS)



Algoritmos de roteamento

Algoritmo de Estado do Enlace

Algoritmo de Dijkstra

- Nova topologia, custos de enlace conhecidos de todos os nós
 - Realizado por “broadcast de estado do enlace”
 - Todos os nós têm a mesma informação
- Calcula caminhos de menor custo de um nó (“origem”) para todos os outros nós
 - Da **tabela de repasse** para esse nó
- Iterativo: após k iterações, sabe caminho de menor custo para k destinos

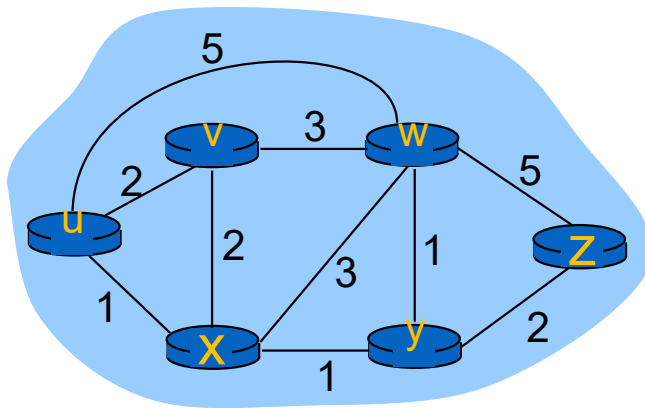
Notação:

- $c(x,y)$: Custo do enlace do nó x até y ; $= \infty$ se não forem vizinhos diretos
- $D(v)$: Valor atual do custo do caminho da origem ao destino v
- $p(v)$: Nó predecessor ao longo do caminho da origem até v
- N' : Conjunto de nós cujo caminho de menor custo é definitivamente conhecido

Algoritmos de roteamento

Algoritmo de Estado do Enlace

Etapa	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					



Saindo do roteador U como chegar com menor custo ao roteador W

Algoritmos de roteamento

Algoritmo de Estado do Enlace

Etapa	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

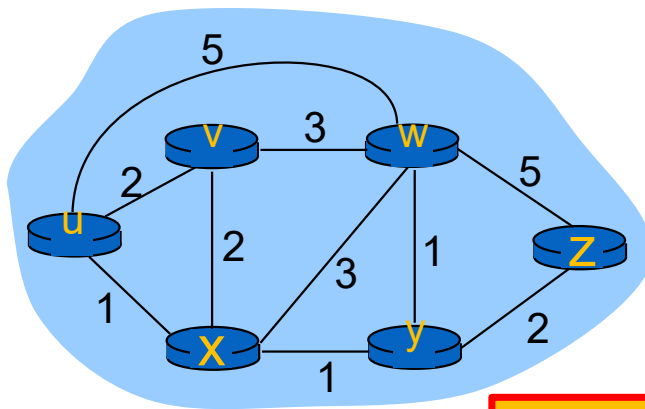


tabela de repasse resultante em u:

destino	enlace
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

Saindo do roteador U como chegar com menor custo ao roteador W

Algoritmos de Vetor de distância



Algoritmos de roteamento

Algoritmo de Vetor de distância

Equação de Bellman-Ford (programação dinâmica)

Defina:

$d_x(y) :=$ custo do caminho de menor custo de uma rede x para rede y

Depois:

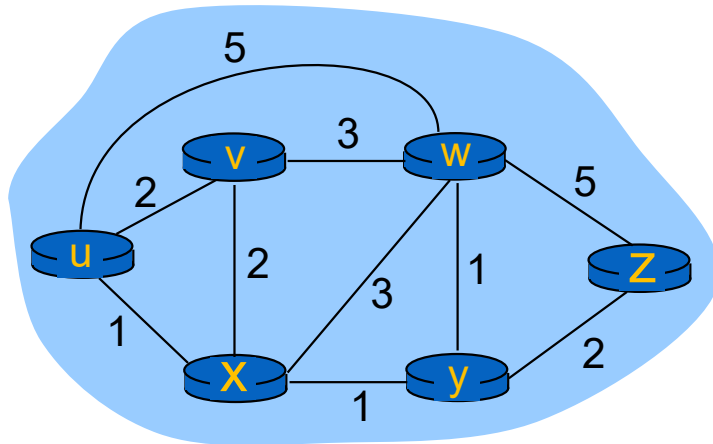
$$d_x(y) = \min \{c(x,v) + d_v(y) \}$$

onde min assume todos os vizinhos v de x

Algoritmos de roteamento

Algoritmo de Vetor de distância

Exemplo de Bellman-Ford



claramente, $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$

equação B-F diz:

$$\begin{aligned} d_u(z) &= \min \{ c(u,v) + d_v(z), \\ &\quad c(u,x) + d_x(z), \\ &\quad c(u,w) + d_w(z) \} \\ &= \min \{ 2 + 5, \\ &\quad 1 + 3, \\ &\quad 5 + 3 \} = 4 \end{aligned}$$

nó que alcança mínimo é o próximo salto
no caminho mais curto → tabela de repasse



Roteamento hierárquico



Algoritmos de roteamento

Algoritmo de roteamento hierárquico

Os algoritmos de roteamento **Link-state** e **Distance vetor** exigem que **todos os roteadores executem o mesmo algoritmo**, saibam da existência uns dos outros, serem de complexidade quadrática e gerarem tráfego adicional na rede. **Na prática isso não é viável**, uma rede mundial de computadores nesses termos seria impossível.

Escala: com 200 milhões de destinos:

- Não pode armazenar todos os destinos nas tabelas de roteamento!
- Troca de tabela de roteamento atolaria os enlaces!



Fonte: Google

Algoritmos de roteamento

Algoritmo de roteamento hierárquico



Fonte: Google

Escala: com 200 milhões de destinos:

- Não pode armazenar todos os destinos nas tabelas de roteamento!
- Troca de tabela de roteamento atolaria os enlaces!

Autonomia administrativa

- Internet = rede de redes
- Cada administrador de rede pode querer controlar o roteamento em sua própria rede'

Algoritmos de roteamento

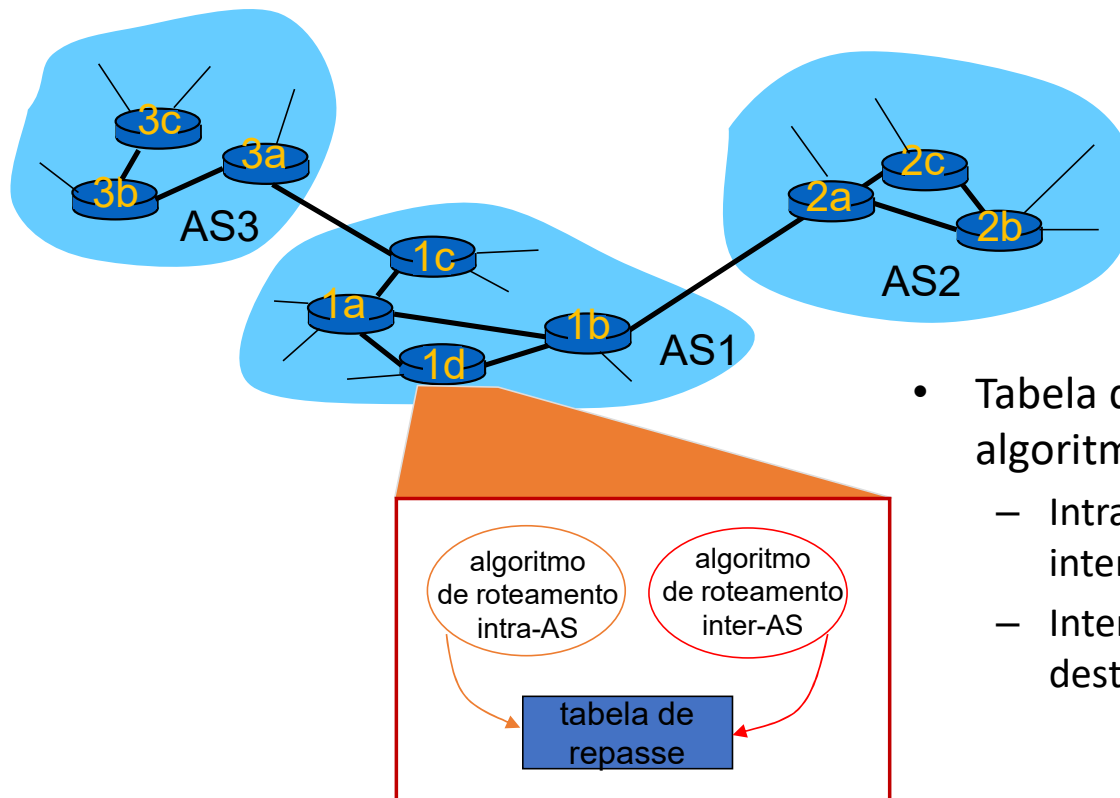
Algoritmo de roteamento hierárquico

- Roteadores agregados em regiões, “sistemas autônomos” (AS)
- Roteadores no mesmo AS rodam o mesmo protocolo de roteamento
 - Protocolo de roteamento “intra-AS”
 - Roteadores em ASes diferentes podem executar protocolo de roteamento intra-AS diferente



Algoritmos de roteamento

Algoritmo de roteamento hierárquico



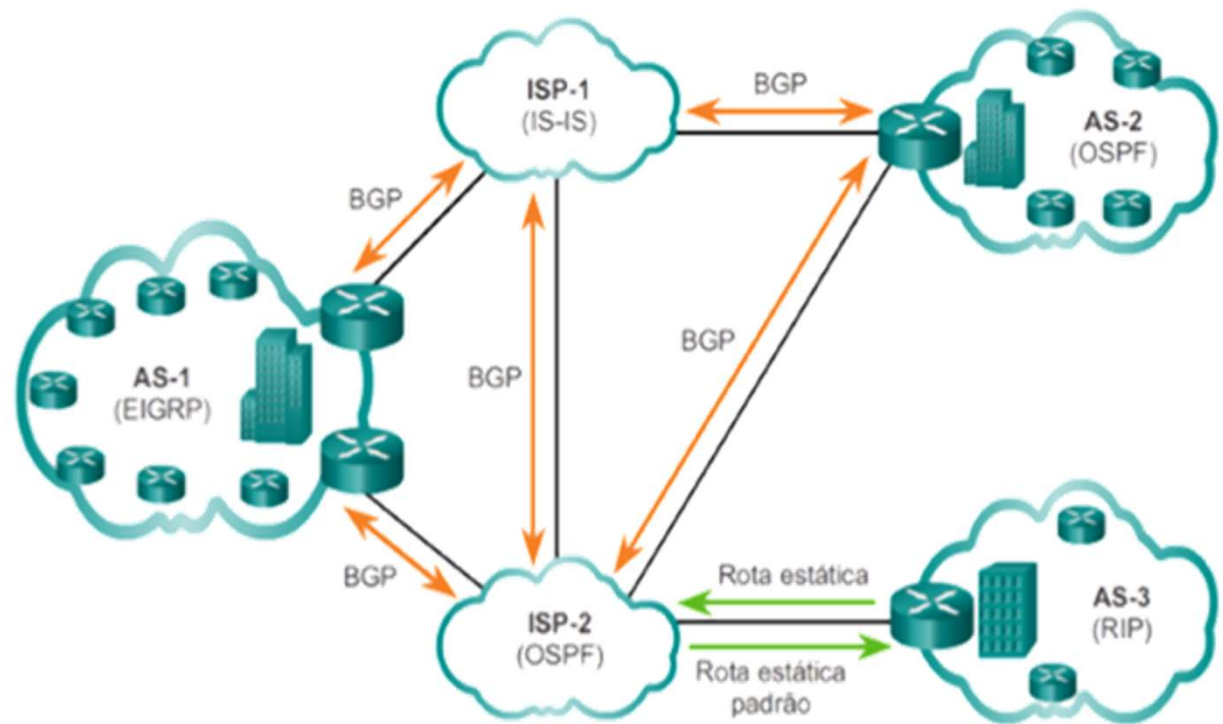
- Tabela de repasse configurada por algoritmo de roteamento intra e inter-AS
 - Intra-AS define entradas para destinos internos
 - Inter-AS & intra-AS definem entradas para destinos externos

Algoritmos de roteamento

Algoritmo de roteamento hierárquico

Um Autonomous System (AS), é um grupo de roteadores que estão sob um mesmo controle administrativo;

- ❖ Dentro do AS a empresa executa o seu protocolo de preferência e entre ASs haverá necessidade de um mesmo protocolo;
- ❖ Roteador que conecta a outro AS é um **roteador de borda (Border Gateway)**.



Algoritmos de roteamento

Algoritmo de roteamento hierárquico

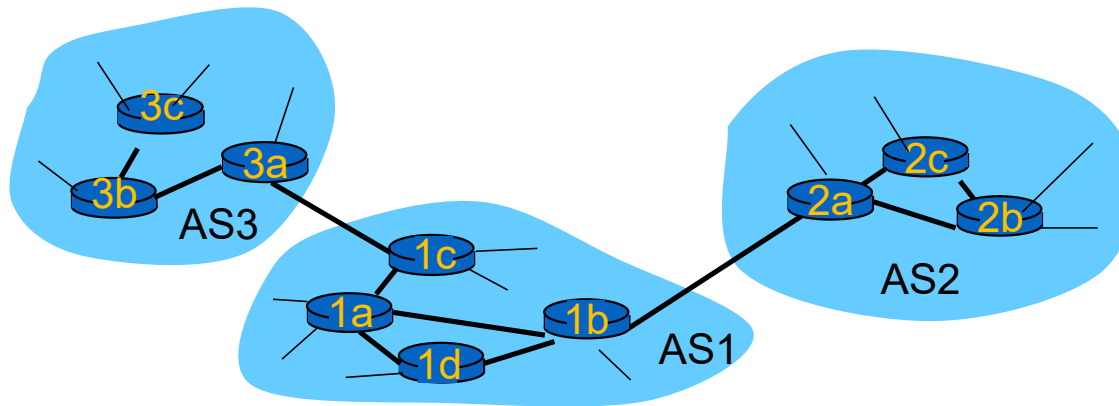
Tarefas inter-AS

- Suponha que roteador no AS1 recebe datagrama destinado para fora do AS1:
 - roteador deve encaminhar pacote ao roteador de borda, mas qual?

AS1 deve:

1. Descobrir quais destinos são alcançáveis por AS2 e quais por AS3
2. Propagar essa informação de acessibilidade a todos os roteadores no AS1

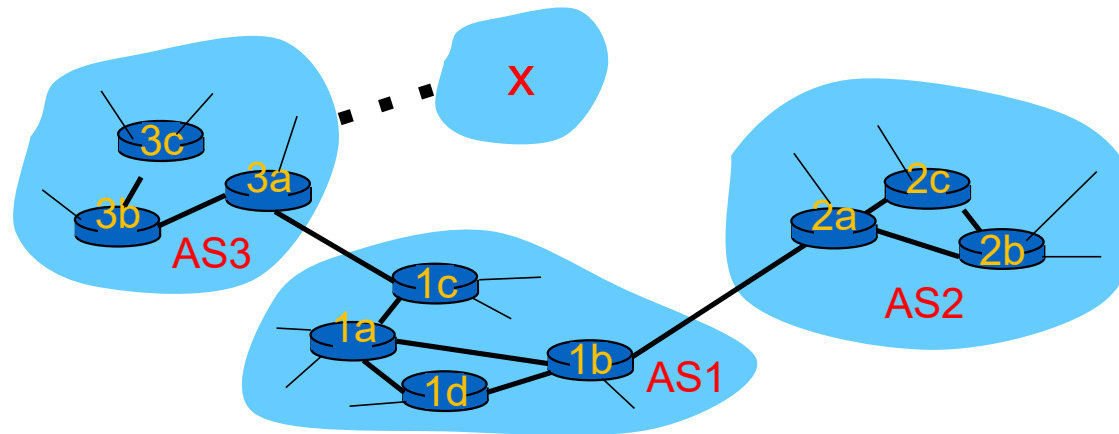
Tarefa do roteamento inter-AS!



Algoritmos de roteamento

Algoritmo de roteamento hierárquico

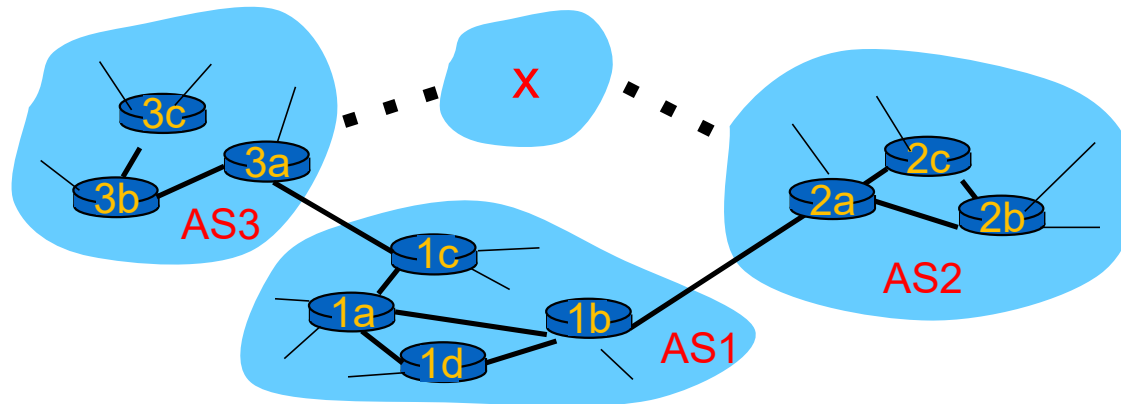
- Suponha que AS1 descubra (pelo protocolo inter-AS) que a sub-rede **x** é alcançável via AS3 (gateway 1c), mas não via AS2.
- Protocolo inter-AS propaga informação de acessibilidade a todos os roteadores internos.
- Roteador 1d determina pelo roteamento intra-AS informação de que sua interface **/** está no caminho de menor custo para 1c.
 - Instala entrada da tabela de repasse (**x, /**)



Algoritmos de roteamento

Algoritmo de roteamento hierárquico

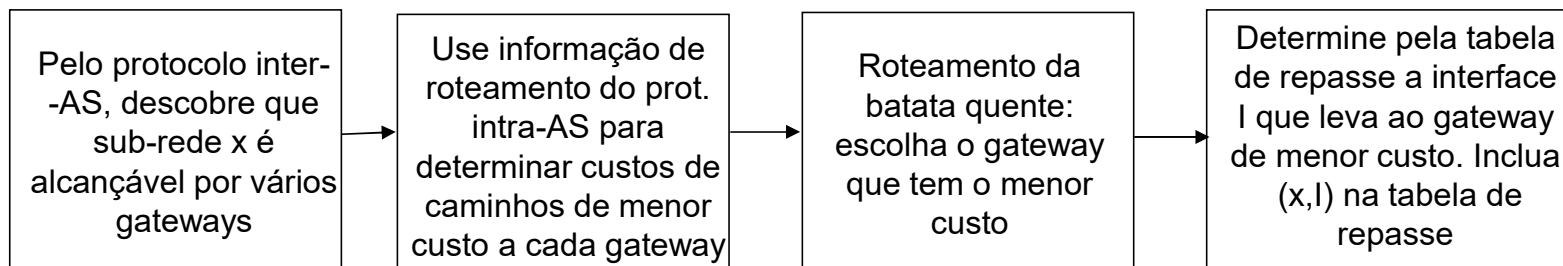
- Agora suponha que o AS1 descubra pelo protocolo inter-AS que a sub-rede **x** pode ser alcançada por AS3 e por AS2.
- Para configurar a tabela de repasse, roteador 1d deve determinar para que gateway ele deve repassar os pacotes para o destino **x**.
 - Isso também é tarefa do protocolo de roteamento inter-AS!



Algoritmos de roteamento

Algoritmo de roteamento hierárquico

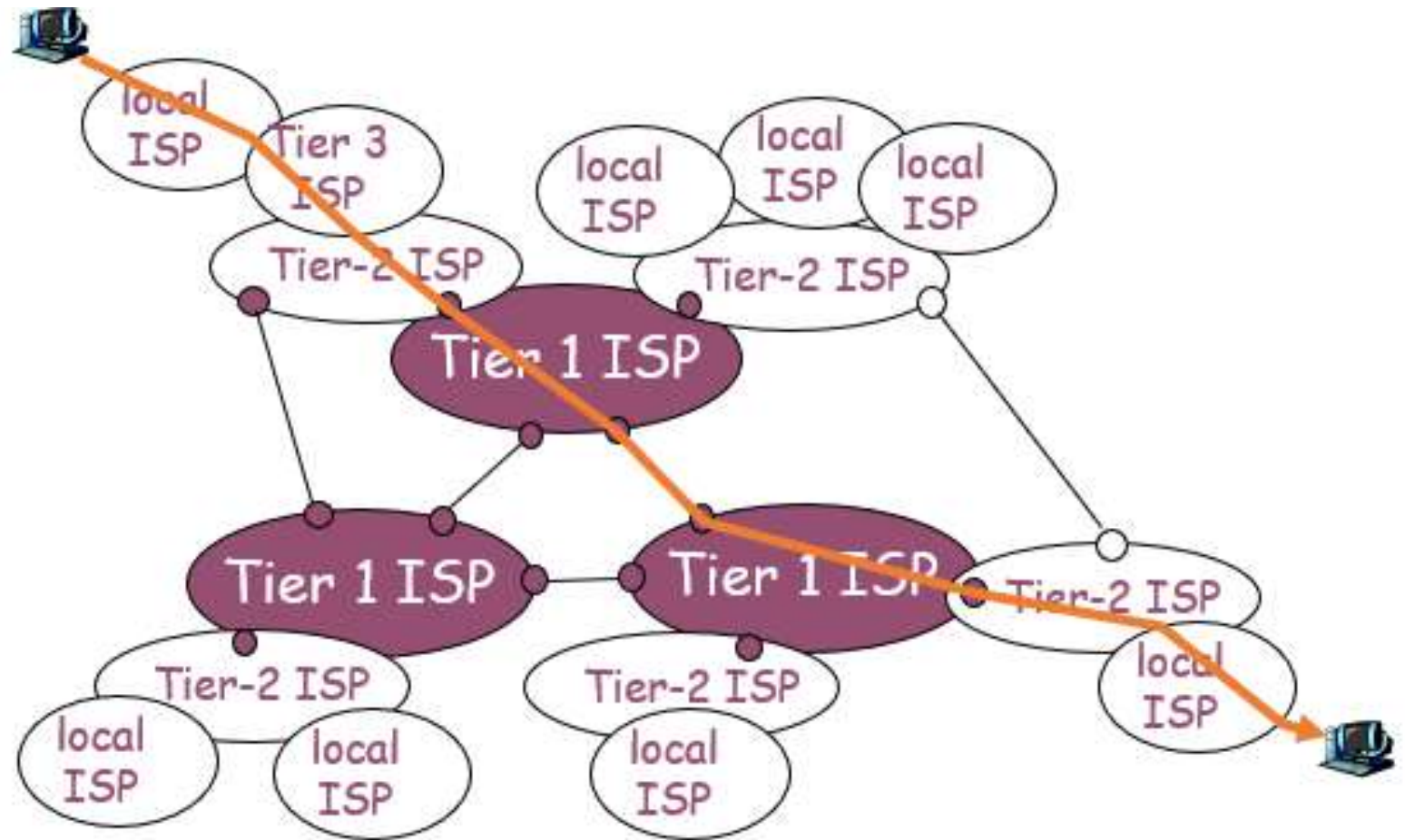
- Agora suponha que AS1 descubra pelo protocolo inter-AS que sub-rede **x** pode ser alcançada por AS3 e por AS2.
- Para configurar a tabela de repasse, o roteador 1d deve determinar para qual gateway deve repassar pacotes para destino **x**.
 - isso também é tarefa do protocolo de roteamento inter-AS!
- ***Envia pacote para o mais próximo dos dois roteadores.***





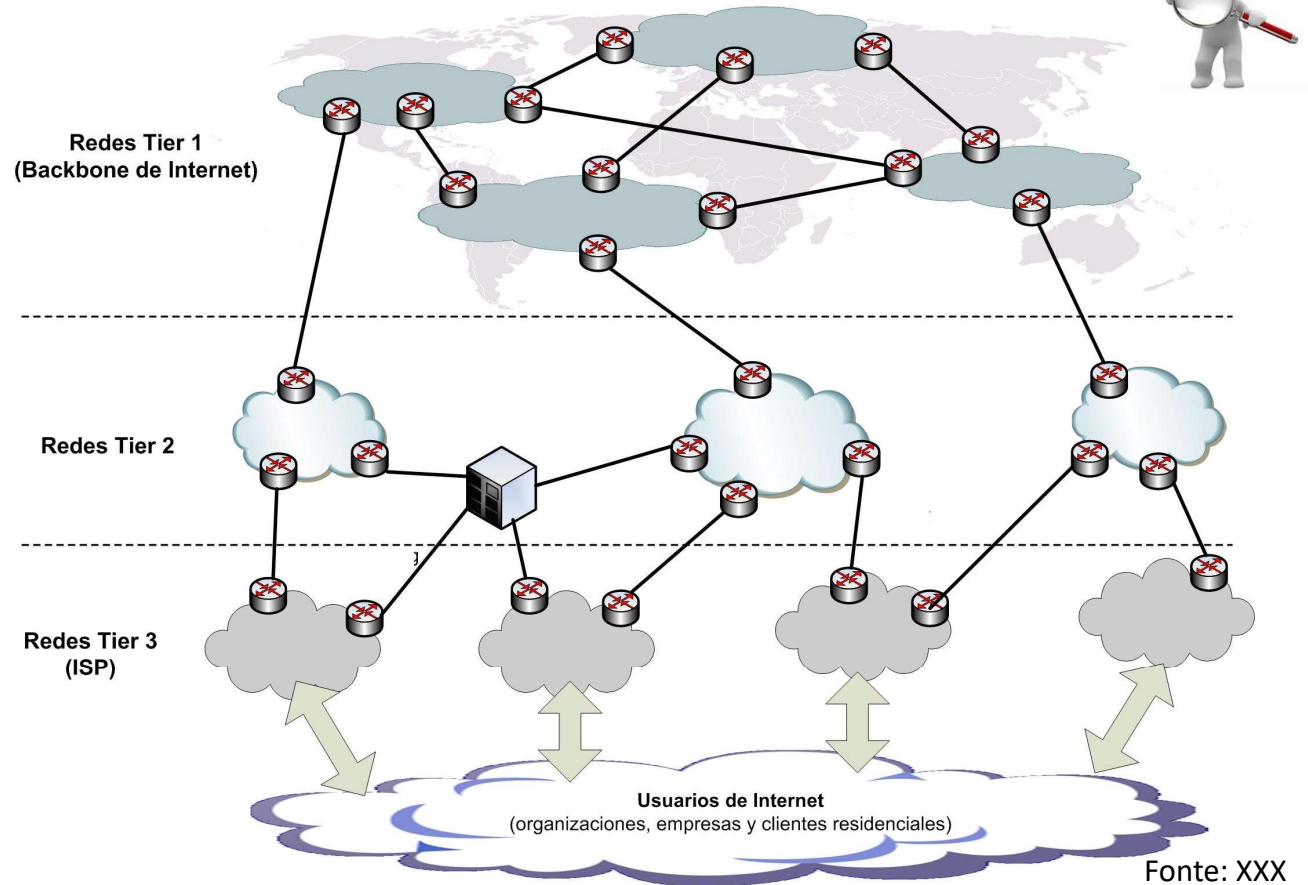
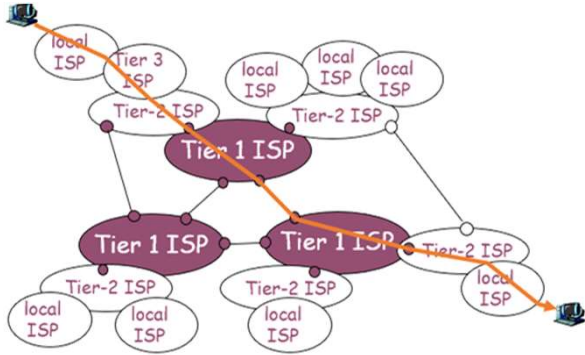
Algoritmos de roteamento

Roteamento hierárquico



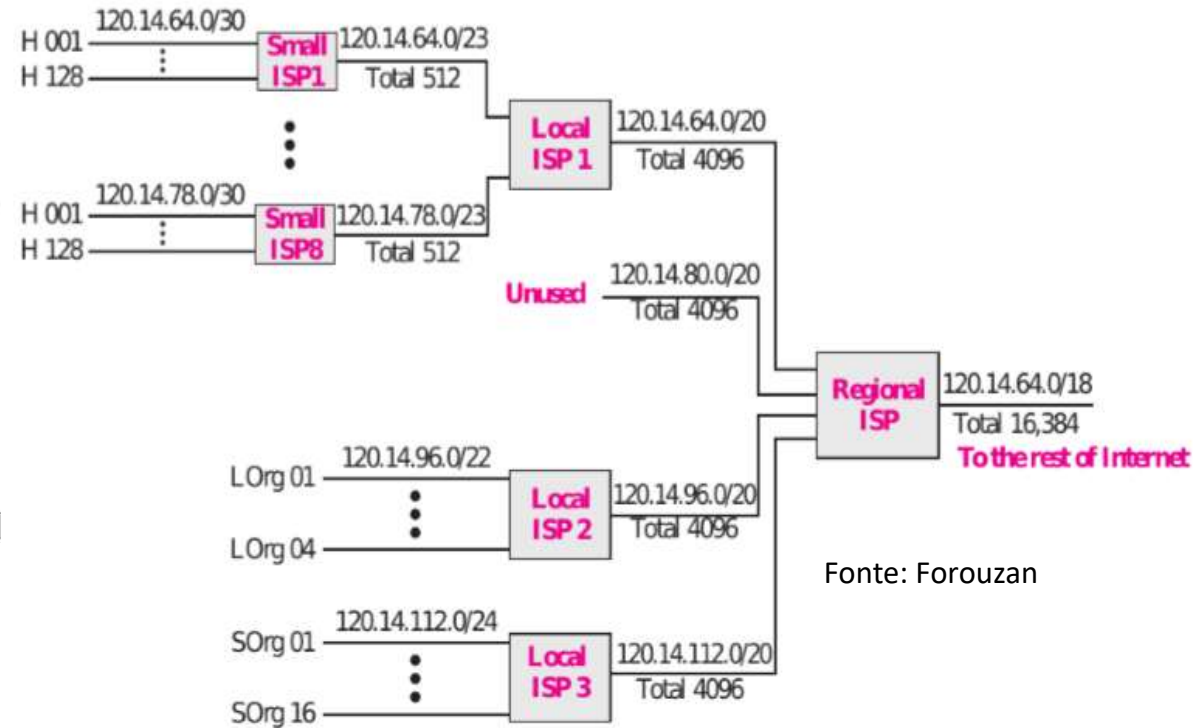
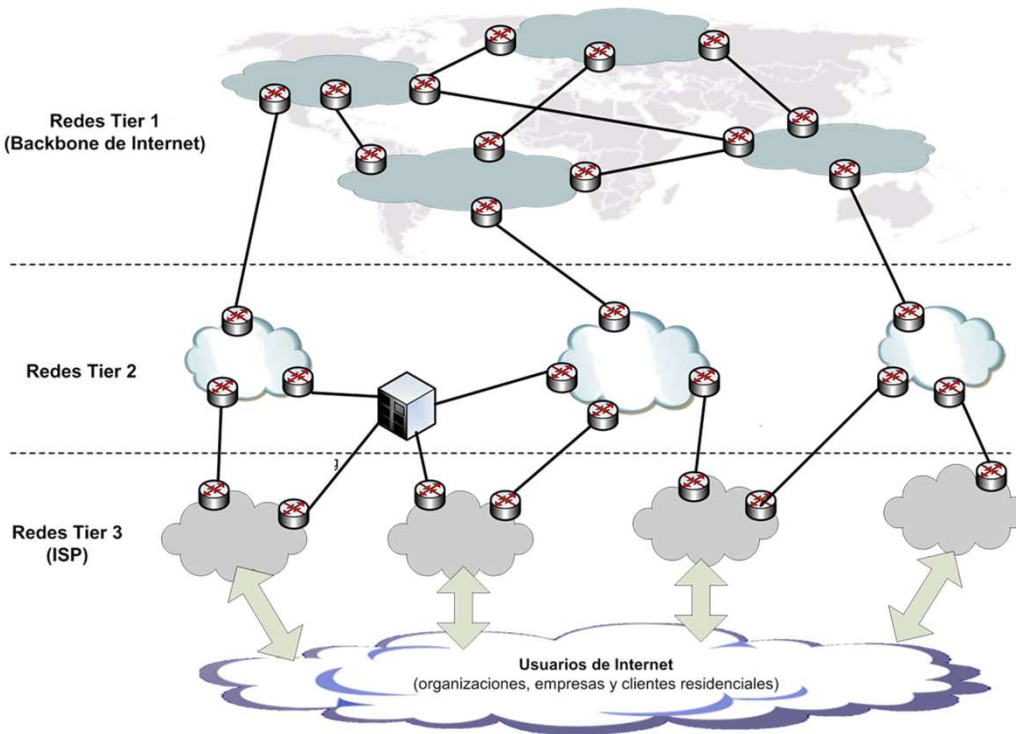
Algoritmos de roteamento

Roteamento hierárquico



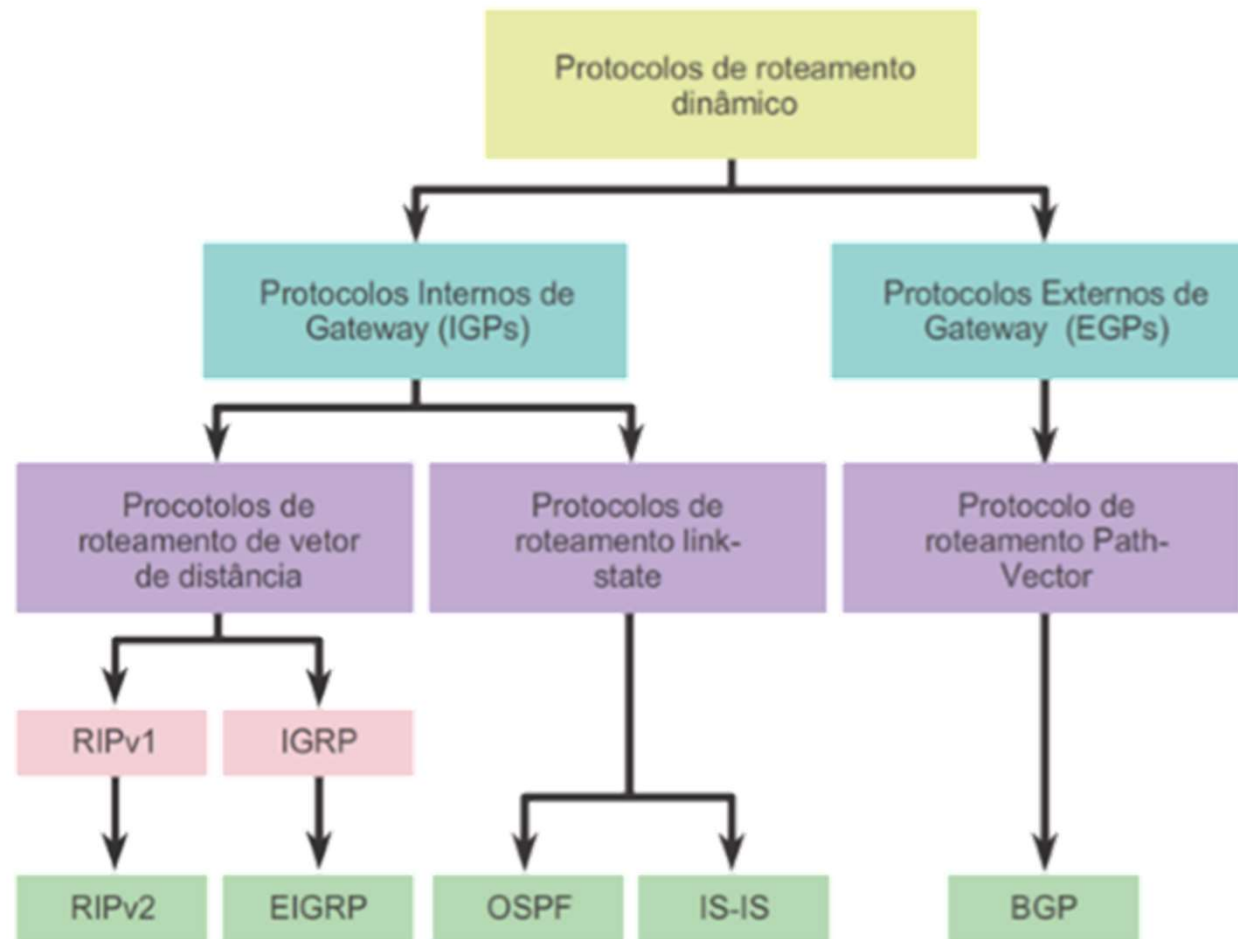
Algoritmos de roteamento

Roteamento hierárquico



Fonte: Forouzan

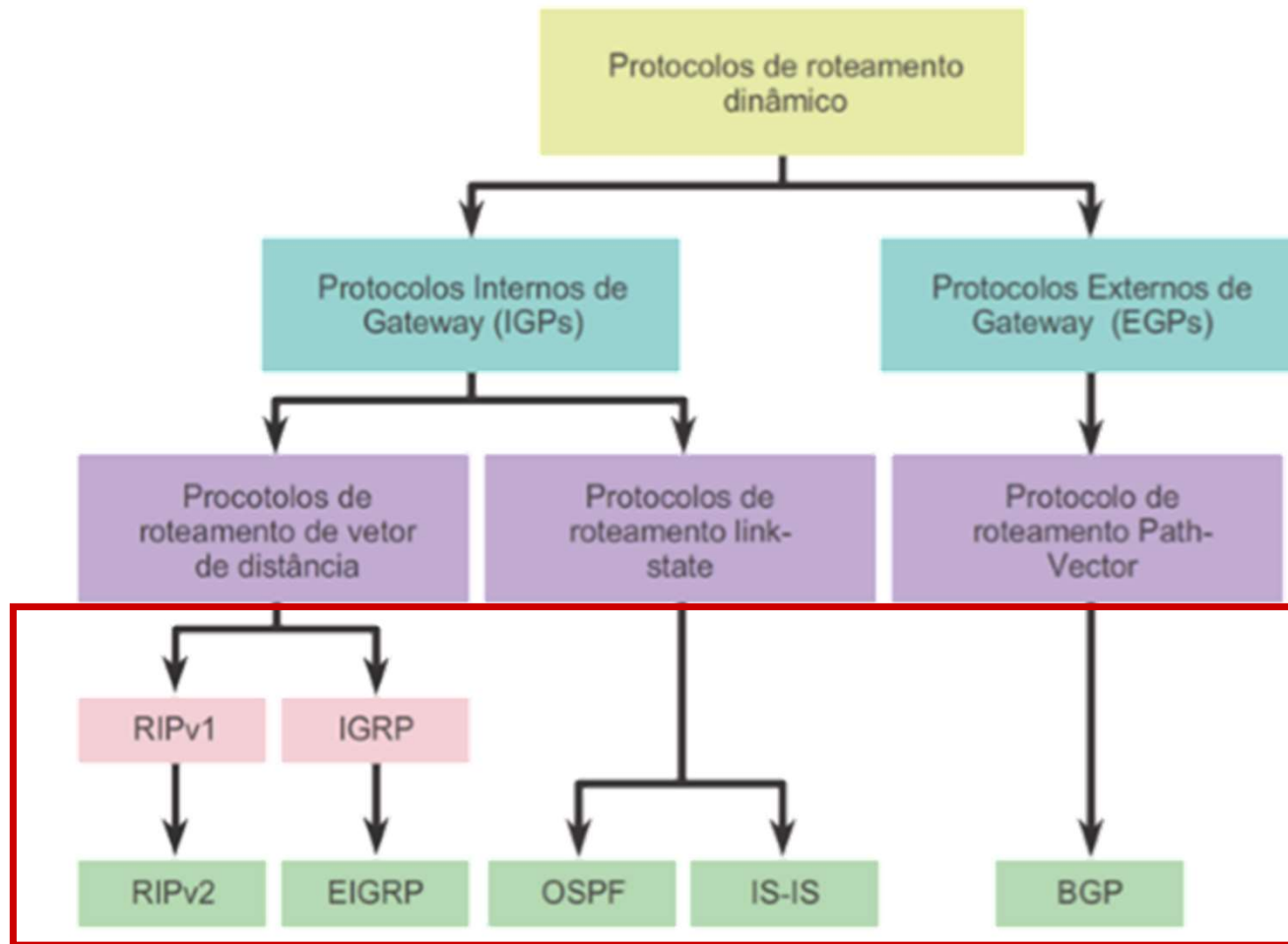
Algoritmos de roteamento



Protocolos de roteamento



Protocolos de roteamento



Protocolos de roteamento

Os Protocolos de Roteamento são divididos em 2 grupos:

✓ **Protocolos de Roteamento Interno**

Protocolos onde as informações são trocadas dentro de Sistemas Autônomos (SA).

Objetivo: enviar pacotes de forma mais eficiente possível da origem ao destino.

Ex: *RIP, OSPF e IGRP (Cisco)*

✓ **Protocolos de Roteamento Externo**

Protocolos que trocam informações entre SA's.

Objetivo: permitir que políticas de roteamento sejam executadas entre SA's.

Ex: *BGP e EGP*

Routing Information Protocol (RIP)



Protocolos de roteamento

RIP (Routing Information Protocol)

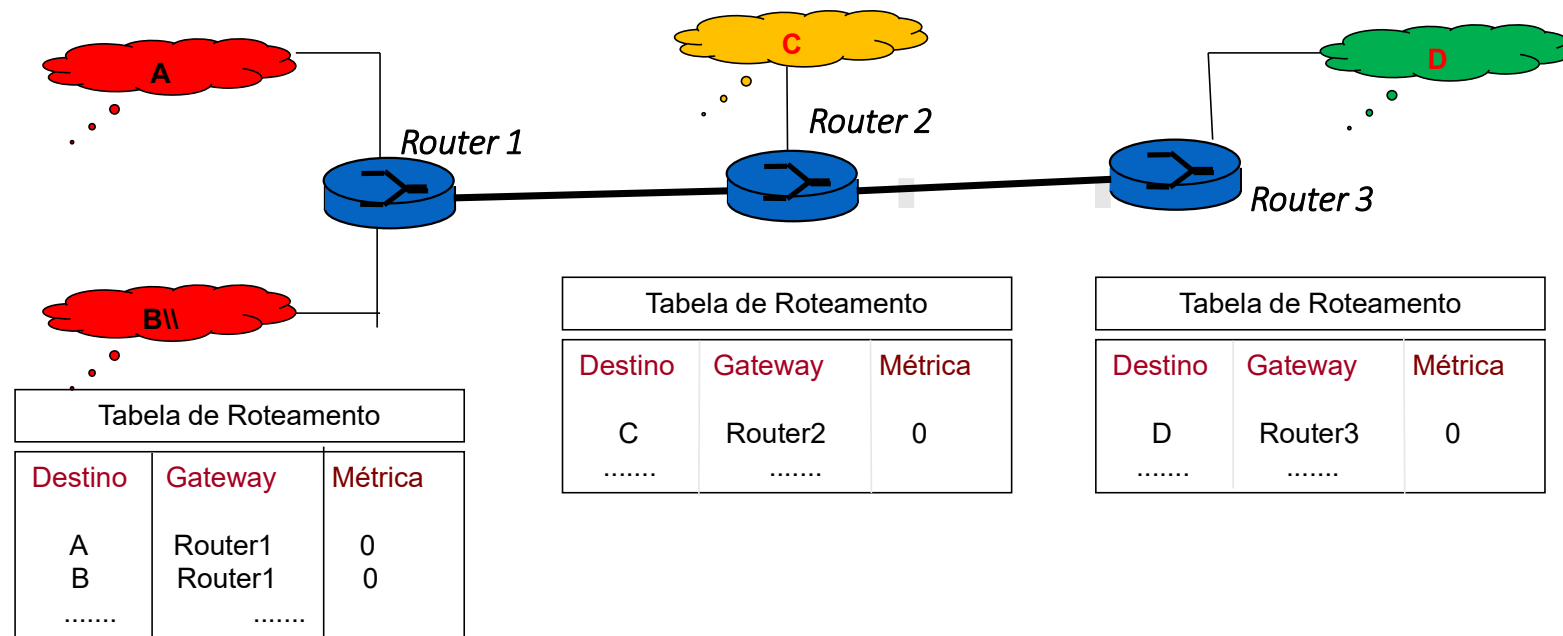
Um dos protocolos internos mais amplamente usados em redes IP. Baseado no Algoritmo com *Vetor de Distâncias* e utiliza a métrica do números de hops, ou seja, escolhe o caminho que percorre o menor número de gateways.

Características:

- ✓ Facilidade de configuração;
- ✓ Seu algoritmo não necessita de grande poder de computação e capacidade de memória nos roteadores;
- ✓ Funciona bem em ambiente pequenos;
- ✓ Roteamento Classful
- ✓ Troca de mensagens em UDP

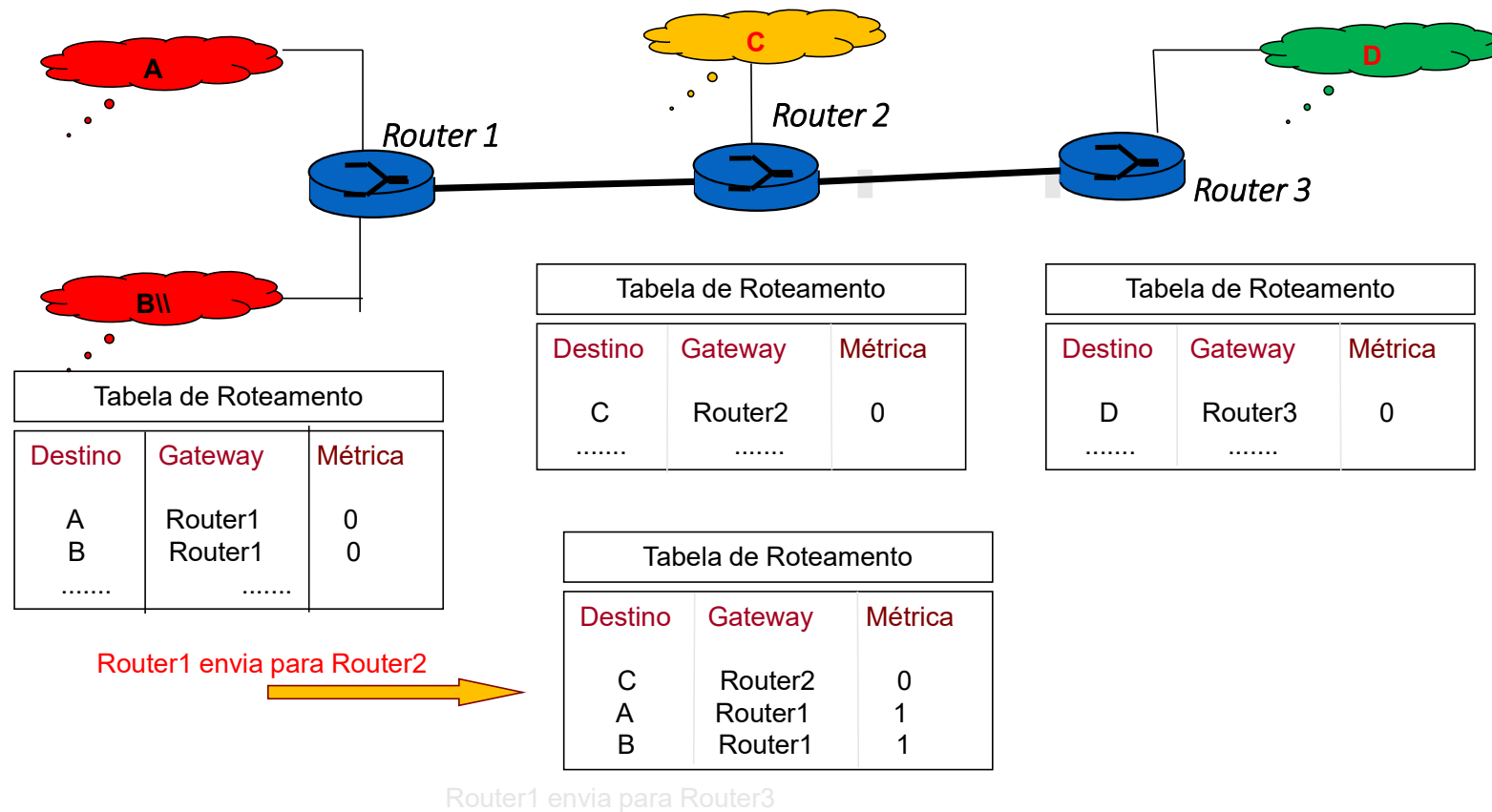
Protocolos de roteamento

RIP (Routing Information Protocol)



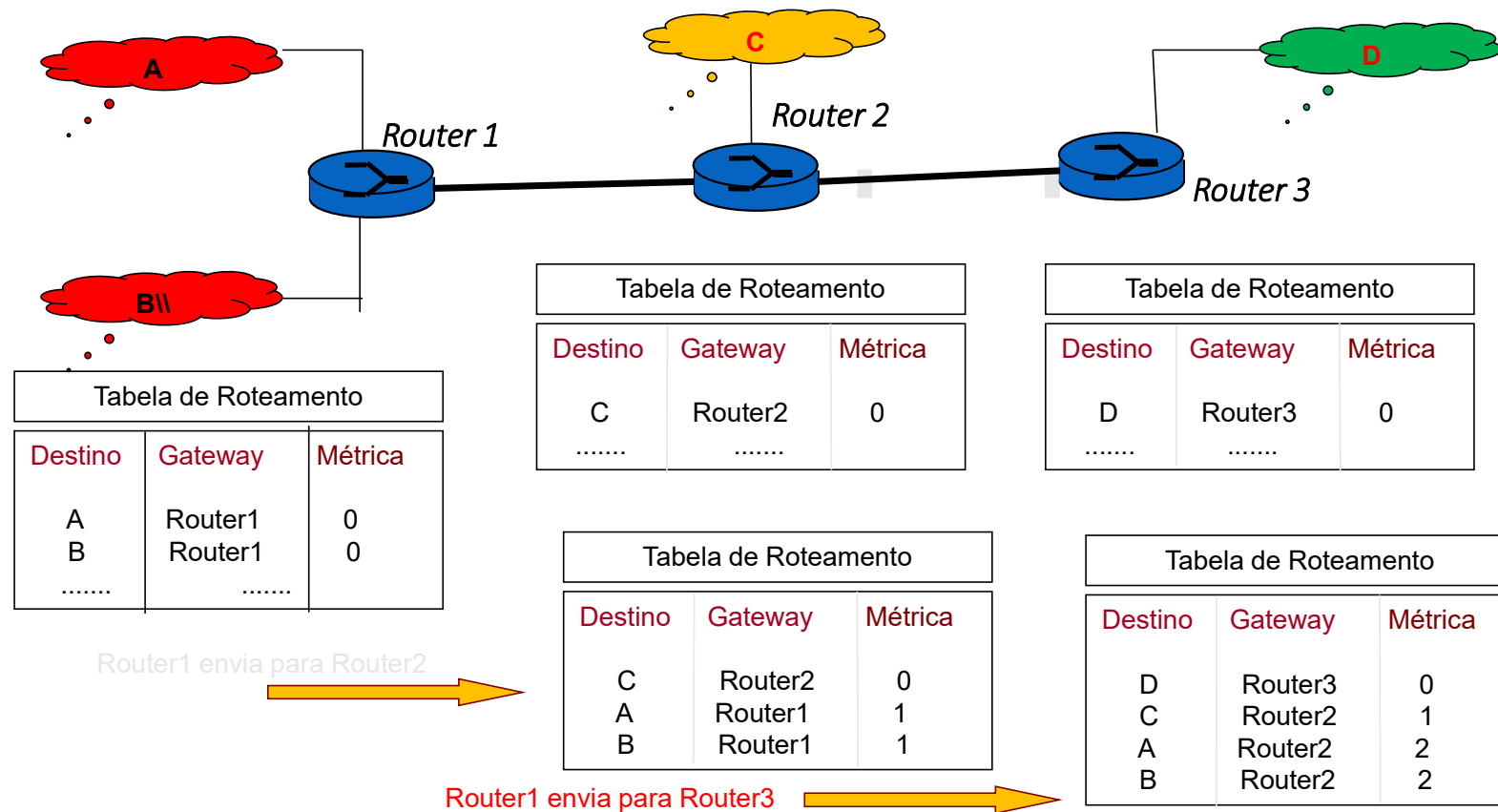
Protocolos de roteamento

RIP (Routing Information Protocol)



Protocolos de roteamento

RIP (Routing Information Protocol)



Protocolos de roteamento

RIP (Routing Information Protocol)

Desvantagens:

- **Limita o número de hops em 15**, sendo assim inadequado para redes grandes - Lenta convergência;
- Leva relativamente **muito tempo para que alterações** na rede fiquem sendo conhecidas por todos os roteadores, podendo causar loops de roteamento devido a falta de sincronia nas informações nos roteadores ;
- **Grande consumidor de largura de banda**, pois, a cada **30s** faz um broadcast de sua tabela de roteamento
- Determina o melhor caminho entre dois pontos levando em conta somente o número de saltos entre eles, **ignorando** outros fatores como: **velocidade e tráfego da rede**, entres outras métricas.

Protocolos de roteamento

RIP (Routing Information Protocol)

A primeira versão do RIP foi descrita em 1988 (RFC 1058). Com o surgimento de protocolos mais robustos e as deficiências apresentadas pelo RIP, em 1993 um no versão **RIPv2** (RFC 1388). Esta nova versão tem como características:

- ✓ Anúncios RIPv2 são baseados em tráfego multicast;
- ✓ Suporte a CIDR;
- ✓ Autenticação MD5;

Open Shortest Path First (OSPF)



Protocolos de roteamento

OSPF - Open Shortest Path First

O protocolo **Open Shortest Path First (OSPF)**, definido no RFC 2328 , é um protocolo IGP utilizado para distribuir a informação de roteamento em um único Sistema Autônomo. Baseado no algoritmo de **Estado do Enlace** (Link-State), o qual foi especificamente projetado para operar com redes grandes.

- “open”: publicamente disponível
- Usa algoritmo Link State
 - disseminação de pacote LS
 - mapa de topologia em cada nó
 - cálculo de rota usando algoritmo de **Dijkstra**
- Anúncio OSPF transporta uma entrada por roteador vizinho
- Anúncios disseminados ao AS **inteiro** (com inundação)
 - transportados nas mensagens OSPF diretamente por IP (em vez de TCP ou UDP)

Protocolos de roteamento

OSPF - Open Shortest Path First

O protocolo **Open Shortest Path First (OSPF)** possui as seguintes características:

- Protocolo Aberto (RFC1247 e RFC2328)
- Usa algoritmo Link State
- OSPF usa datagramas IP diretamente (89 protocolo superior)
- Anúncios disseminados ao AS **inteiro** (com inundação)
- Todas as mensagens OSPF **autenticadas** - Senhas ou hash MD-5

Protocolos de roteamento

OSPF - Open Shortest Path First

ospf								
No.	Time	Source	Destination	Host	Protocol	Length	Destinat	Info
21	3.624424	192.168.0.1	224.0.0.5		OSPF	98		Hello Packet
22	3.823114	192.168.0.2	224.0.0.5		OSPF	98		Hello Packet

> Frame 21: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)

> Ethernet II, Src: Vmware_dd:4c:54 (00:0c:29:dd:4c:54), Dst: IPv4mcast_05 (01:00:5e:00:00:05)

> Internet Protocol Version 4, Src: 192.168.0.1, Dst: 224.0.0.5

> Open Shortest Path First

> OSPF Header

Version: 2

Message Type: Hello Packet (1)

<Hello: True>

Packet Length: 48

Source OSPF Router: 10.0.0.1

Area ID: 0.0.0.0 (Backbone)

Checksum: 0x0000 (None)

Auth Type: Cryptographic (2)

Auth Crypt Key id: 1

Auth Crypt Data Length: 16

Auth Crypt Sequence Number: 1185822602

Auth Crypt Data: a8bebf4068271bc8691fe905634986c1

> OSPF Hello Packet

Mensagens em Multicast

Autenticação do roteador!!!!

Entalecendo conexão com os Vizinhos

Protocolos de roteamento

OSPF - Open Shortest Path First

ospf								
No.	Time	Source	Destination	Host	Protocol	Length	Destinat	Info
21	3.624424	192.168.0.1	224.0.0.5		OSPF	98		Hello Packet
22	3.823114	192.168.0.2	224.0.0.5		OSPF	98		Hello Packet

> Frame 21: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)

> Ethernet II, Src: Vmware_dd:4c:54 (00:0c:29:dd:4c:54), Dst: IPv4mcast_05 (01:00:5e:00:00:05)

> Internet Protocol Version 4, Src: 192.168.0.1, Dst: 224.0.0.5

> Open Shortest Path First

> OSPF Header

Version: 2

Message Type: Hello Packet (1)

<Hello: True>

Packet Length: 48

Source OSPF Router: 10.0.0.1

Area ID: 0.0.0.0 (Backbone)

Checksum: 0x0000 (None)

Auth Type: Cryptographic (2)

Auth Crypt Key id: 1

Auth Crypt Data Length: 16

Auth Crypt Sequence Number: 1185822602

Auth Crypt Data: a8bebf4068271bc8691fe905634986c1

> OSPF Hello Packet

Mensagens em Multicast

Autenticação do roteador!!!!

Entalecendo conexão com os Vizinhos

Protocolos de roteamento

Os Protocolos de Roteamento são divididos em 2 grupos:

✓ ~~Protocolos de Roteamento Interno~~

~~Protocolos onde as informações são trocadas dentro de Sistemas Autônomos (SA).~~

~~**Objetivo:** enviar pacotes de forma mais eficiente possível da origem ao destino.~~

~~**Ex:** *RIP, OSPF e IGRP (Cisco)*~~

✓ Protocolos de Roteamento Externo

Protocolos que trocam informações entre SA's.

Objetivo: permitir que políticas de roteamento sejam executadas entre SA's.

Ex: *BGP e EGP*

Protocolos de roteamento

BGP - Border Gateway Protocol

- ✓ BGP é um protocolo de roteamento do tipo interdomínio que transmite informações de prefixos
- ✓ BGP é um protocolo do tipo “*path vector*”
 - ✓ Similar ao “*distance vector*”
- ✓ BGP percebe a Internet como uma coleção de autonomous systems (AS)
- ✓ BGP suporta CIDR
- ✓ Roteadores BGP trocam informações de roteamento entre “peers”

Protocolos de roteamento

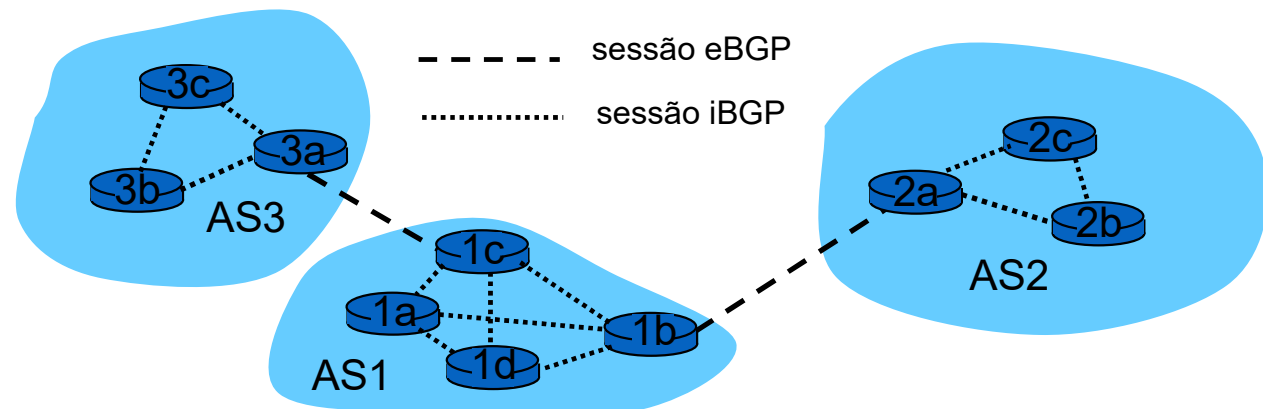
BGP - Border Gateway Protocol

- BGP oferece a cada AS um meio de:
 1. Obter informação de acessibilidade da sub-rede a partir de ASs vizinhos.
 2. Propagar informação de acessibilidade a todos os roteadores internos ao AS.
 3. Determinar rotas “boas” para sub-redes com base na informação e política de acessibilidade.
- Permite que a sub-rede anuncie sua existência ao resto da Internet.

Protocolos de roteamento

BGP - Border Gateway Protocol

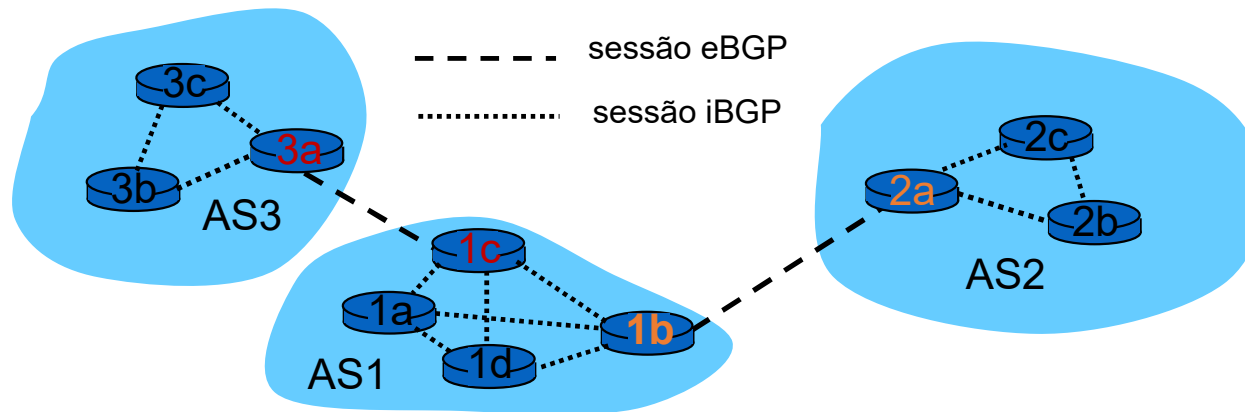
- **Sessões BGP** - Pares de roteadores (BGP peers) trocam informações de roteamento por conexões **TCP**.
- Quando AS2 comunica um prefixo ao AS1, AS2 está prometendo que irá encaminhar todos os datagramas destinados a esse prefixo em direção ao prefixo.
- AS2 pode agregar prefixos em seu comunicado.



Protocolos de roteamento

BGP - Border Gateway Protocol

- Usando sessão **eBGP** entre **3a** e **1c**, AS3 envia informação de tangibilidade do prefixo a AS1.
 - **1c** pode então usar **iBGP** para distribuir nova informação de prefixo a todos os roteadores em AS1
 - **1b** pode então reanunciar nova informação de atingibilidade para AS2 por sessão **eBGP 1b-para-2a**
- Quando roteador descobre novo prefixo, ele cria entrada para prefixo em sua tabela de repasse.



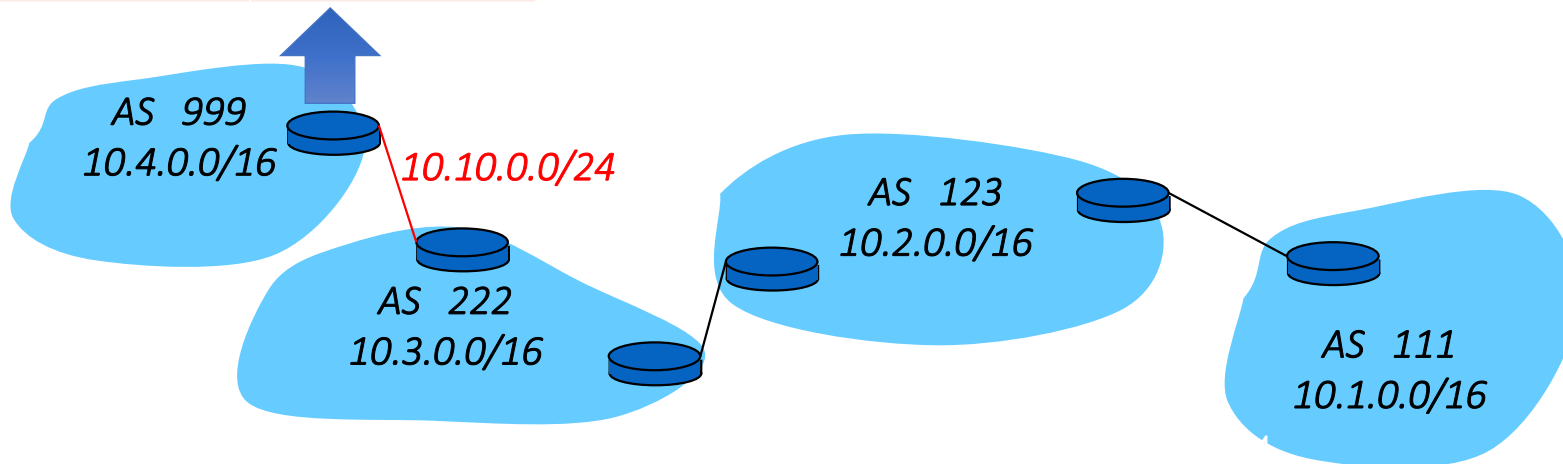
Protocolos de roteamento

BGP - Border Gateway Protocol

- Dois atributos importantes:

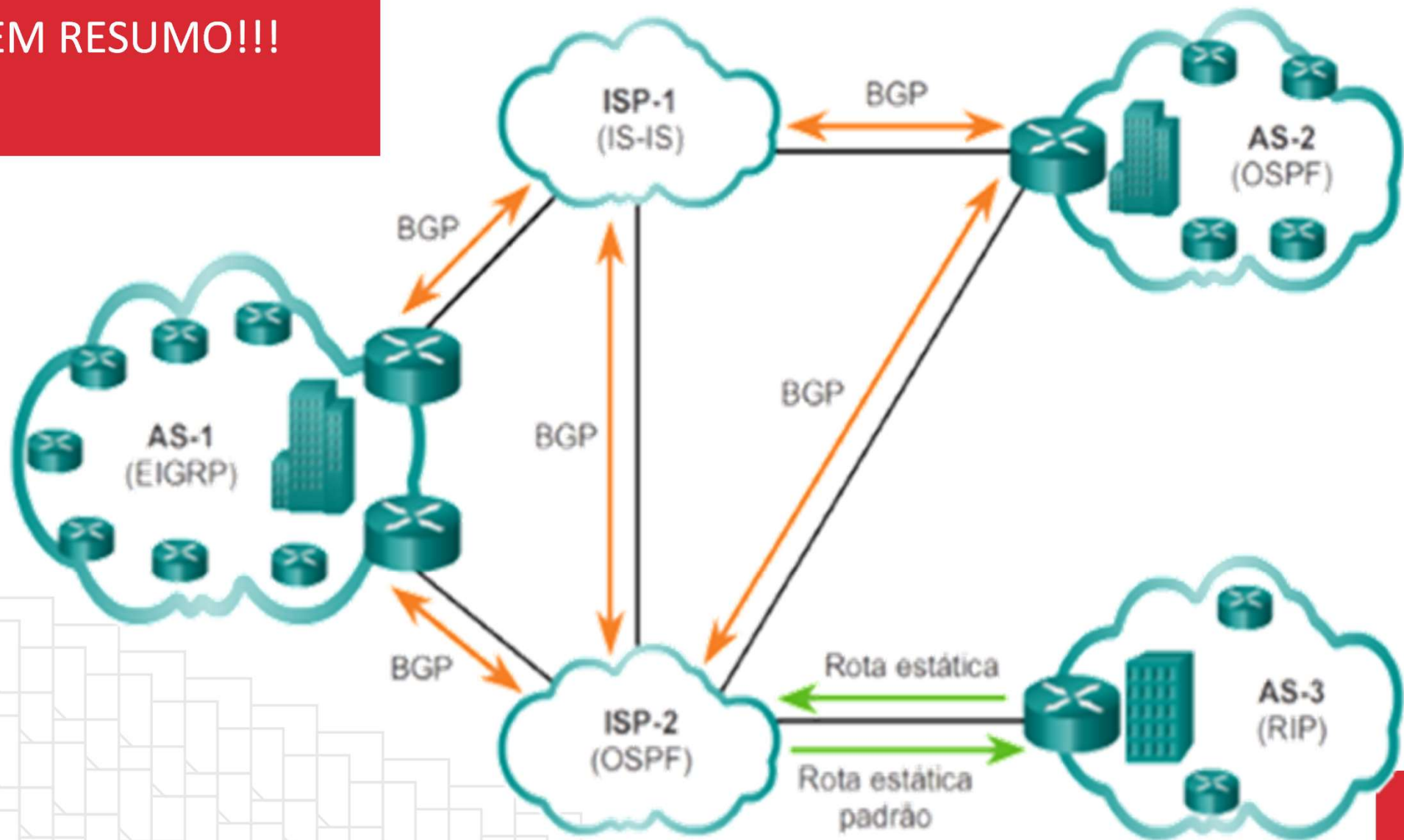
Prefixo	Next-hop	AS-Path
10.1.0.0/16	10.10.0.1	222, 123, 111
10.2.0.0/16	10.10.0.1	222, 123
10.3.0.0/16	10.10.0.1	222
10.4.0.0/16		

- **AS-PATH**: contém ASs através dos quais o anúncio do prefixo passou: p. e., AS 67, AS 17
- **NEXT-HOP**: indica roteador específico do AS interno para AS do próximo salto (podem ser múltiplos enlaces para AS atual até AS do próximo salto)





EM RESUMO!!!





Obrigado!



Referências :

Algoritmos de roteamento – link state e Distance vector

Capítulo 4 - Páginas de 268 à 279

Roteamento hierárquico

Capítulo 4 – Página 280

Protocolos de Roteamento

Capítulo 4 – Páginas de 283 à 294



Referências :

Algoritmos de roteamento – link state e Distance vector

Capítulo 5 - Páginas de 226 à 236

Roteamento hierárquico

Capítulo 5 – Página 237

Protocolos de Roteamento

Capítulo 5 – Páginas de 296 à 302



Referências :

COMER, D. E. **Redes de computadores e internet**. 6. ed. Porto Alegre: Bookman, 2016.

The Internet Engineering Task Force (IETF®) - <https://www.ietf.org/>

Apostilas Cert Br <https://cartilha.cert.br/downloads/>

Notas de curso - **Cisco Routing & Switching**

Notas de Aulas - Ana Cristina Benso da Silva



Algoritmos de roteamento

Algoritmo de Vetor de distancia

tabela nó x

custo para
x y z

x	0	2	7
y	∞	∞	∞
z	∞	∞	∞

tabela nó y

custo para
x y z

x	∞	∞	∞
y	2	0	1
z	∞	∞	∞

tabela nó z

custo para
x y z

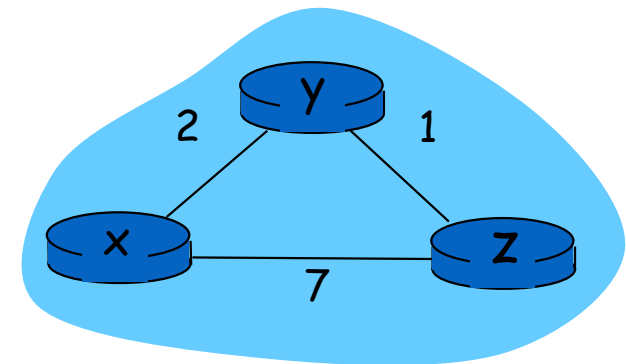
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0

custo para
x y z

x	0	2	3
y	2	0	1
z	7	1	0

$$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\} \\ = \min\{2+0, 7+1\} = 2$$

$$D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\} \\ = \min\{2+1, 7+0\} = 3$$



Algoritmos de roteamento

Algoritmo de Vetor de distancia

tabela nó x

	custo para		
	x	y	z
x	0	2	7
de y	∞	∞	∞
z	∞	∞	∞

tabela nó y

	custo para		
	x	y	z
x	∞	∞	∞
de y	2	0	1
z	∞	∞	∞

tabela nó z

	custo para		
	x	y	z
x	∞	∞	∞
de y	∞	∞	∞
z	7	1	0

	custo para		
	x	y	z
x	0	2	3
de y	2	0	1
z	7	1	0

	custo para		
	x	y	z
x	0	2	7
de y	2	0	1
z	7	1	0

	custo para		
	x	y	z
x	0	2	7
de y	2	0	1
z	3	1	0

	custo para		
	x	y	z
x	0	2	3
de y	2	0	1
z	3	1	0

	custo para		
	x	y	z
x	0	2	3
de y	2	0	1
z	3	1	0

	custo para		
	x	y	z
x	0	2	3
de y	2	0	1
z	3	1	0

$$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\} \\ = \min\{2+0, 7+1\} = 2$$

$$D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\} \\ = \min\{2+1, 7+0\} = 3$$

