

Camada de Rede

Redes de Computadores

Profa. Kalinka Castelo Branco

Universidade de São Paulo

Abril de 2020

Camada de Rede

Profa.
Kalinka
Branco

Introdução

Roteamento

Endereços IP

1 Introdução

2 Roteamento

3 Endereços IP

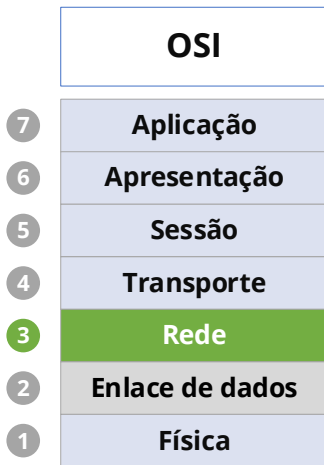
Camada de Rede

Profa.
Kalinka
Branco

Introdução

Roteamento

Endereços IP



Função da Camada de Rede

- Transportar pacotes entre os sistemas finais da rede.
- A camada de rede deve ter uma entidade em cada sistema final ou roteador da rede.

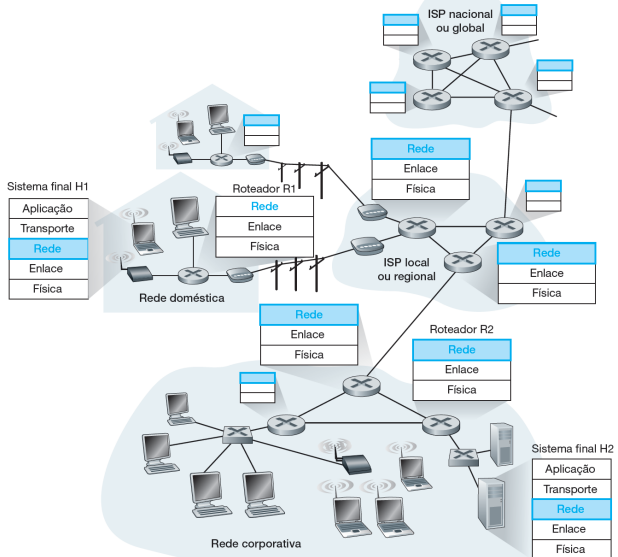
Camada de Rede

Profa.
Kalinka
Branco

Introdução

Roteamento

Endereços IP



- Três funções importantes:
 - **Determinação de caminhos:** rota escolhida pelos pacotes entre a origem e o destino (algoritmos de roteamento);
 - **Comutação:** mover pacotes entre as portas de entrada e de saída dos roteadores;
 - **Estabelecimento de conexão:** algumas arquiteturas de rede exigem o estabelecimento de circuitos virtuais antes da transmissão de dados.

Camada de Rede

Profa.
Kalinka
Branco

Introdução

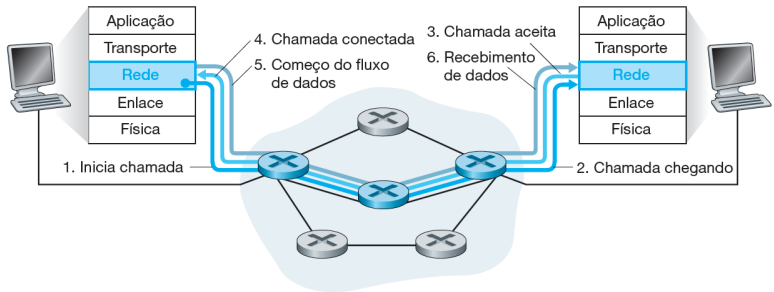
Roteamento

Endereços IP

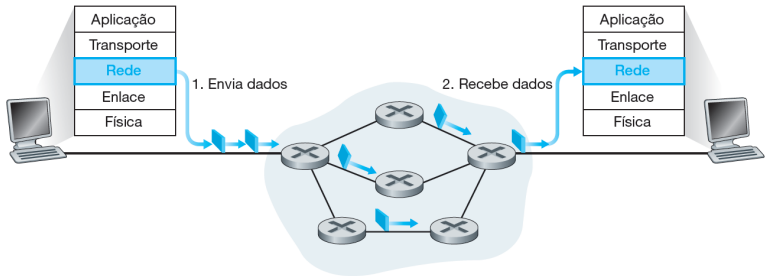
- Existem dois serviços possíveis para entregar pacotes a seus respectivos destinos:
 - Redes de Circuitos Virtuais;
 - Redes de Datagramas.

- Estabelece-se uma conexão antes do envio de dados;
- Libera-se a conexão após troca de dados;
- Cada pacote transporta um identificador do VC, não transporta o endereço completo do destino;
- Cada roteador na rota mantém informações de estado para a conexão que passa por ele.
- Vantagens:
 - Orientado ao desempenho;
 - A banda passante e os recursos do roteador podem ser alocados por VC;
 - Controle de Qualidade de Serviço (QoS) por VC.

- Sinalização é usada para estabelecer, manter e encerrar Circuitos Virtuais;
- Usados em ATM, Frame Relay e X-25, mas não na Internet.



- Não estabelece conexões;
- Não há informação de estado de conexão nos roteadores;
- Pacotes tipicamente transportam o endereço de destino;
 - Pacotes para o mesmo destino podem seguir diferentes rotas.
- Usado na Internet;



Redes de Datagrama (Internet):

- Dados trocados entre computadores:
 - Serviço elástico;
 - Requisitos de atraso não críticos.
- Sistemas finais inteligentes:
 - Podem adaptar-se, realizar controle e recuperação de erros;
 - A rede é simples;
 - Complexidade nos sistemas finais;
- Muitos tipos de enlaces:
 - Características diferentes;
 - Difícil obter um serviço uniforme.

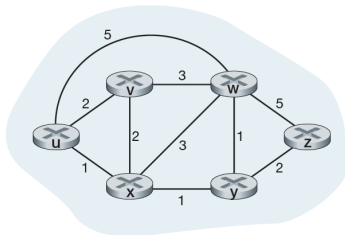
Redes de Circuito Virtual (ATM):

- Originário da telefonia;
- Ótimo para conversação humana:
 - Tempos estritos, exigências de confiabilidade;
 - Necessário para serviço garantido.
- Sistemas finais mais simples:
 - Telefones;
 - Complexidade dentro da rede.

Roteamento

Determinar “bons” caminhos (sequência de roteadores) através da rede da fonte até o destino.

- Algoritmos de roteamento são descritos por **grafos**;
- Os **nós** do grafo são **roteadores**;
- As **arestas** do grafo são **enlaces**:
 - Custo do enlace: atraso, preço ou nível de congestionamento.
- “Bons” caminhos:
 - Caminhos de menor custo;
 - Caminhos redundantes.



- **Informação Global:**

- Todos os roteadores tem informações completas da topologia e do custo dos enlaces;
- Algoritmos de estado de enlace (“*Link state*”).

- **Informação Descentralizada:**

- Roteadores só conhecem informações sobre seus vizinhos e os enlaces para chegar até eles;
- Processo de computação iterativo, troca de informações com os vizinhos;
- Algoritmos de vetor de distância (“*Distance vector*”).

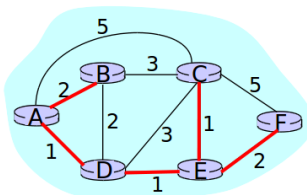
- **Estático:**
 - As rotas mudam lentamente ao longo do tempo;
 - Muitas vezes dependem de mudanças feitas por um administrador de rede.
- **Dinâmico:**
 - As rotas mudam mais rapidamente:
 - Atualizações periódicas;
 - Podem responder a mudanças no custo dos enlaces.

- *Shortest path*;
- **Estático e Global**: topologia de rede e custo dos enlaces são conhecidos por todos os nós antecipadamente para o cálculo das tabelas de roteamento.
- **Funcionamento**: calcula-se o caminho mais curto pelo Algoritmo de Dijkstra.

- Topologia de rede e custo dos enlaces são conhecidos por todos os nós.
 - Implementado via “link state broadcast”.
 - Todos os nós têm a mesma informação
- Computa caminhos de menor custo de um nó (fonte) para todos os outros nós.
 - Fornece uma tabela de roteamento para aquele nó.
- Convergência: após k iterações, conhece o caminho de menor custo para k destinos.

- $C(i,j)$: custo do enlace do nó i ao nó j . Custo é infinito se não houver ligação entre i e j
- $D(v)$: valor atual do custo do caminho da fonte ao destino v
- $P(v)$: nó predecessor ao longo do caminho da fonte ao nó v , isto é, antes do v
- N : conjunto de nós cujo caminho de menor custo é definitivamente conhecido.

Passo	início N	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
→ 0	A	2,A	5,A	1,A	infinito	infinito
→ 1	AD	2,A	4,D		2,D	infinito
→ 2	ADE	2,A	3,E			4,E
→ 3	ADEB		3,E			4,E
→ 4	ADEBC					4,E
5	ADEBCF					



- 1 **Inicialização:**
- 2 $N = \{A\}$
- 3 para todos os nós v
- 4 se v é adjacente a A
- 5 então $D(v) = c(A,v)$
- 6 senão $D(v) = \infty$
- 7
- 8 **Loop**
- 9 ache w não em N tal que $D(w)$ é um mínimo
- 10 acrescente w a N
- 11 atualize $D(v)$ para todo v adjacente a w e não em N :
- 12 $D(v) = \min(D(v), D(w) + c(w,v))$
- 13 */* novo custo para v é ou o custo anterior para v ou o menor*
- 14 *custo de caminho conhecido para w mais o custo de w a v */*
- 15 **até que todos os nós estejam em N**

- *Flooding*;
- **Estático e Descentralizado**;
- **Funcionamento**: os pacotes que chegam são reencaminhados para todas as linhas, exceto pela que chegou.
- Para reduzir o número de pacotes, é possível acrescentar um contador de saltos no cabeçalho.

- *Distance Vector* (DV);
- Também conhecido como Algoritmo de Roteamento de Bellman-Ford ou Algoritmo de Ford-Fulkerson;
- **Dinâmico e Descentralizado;**
- **Funcionamento:**
 - ① Cada nó mantém um vetor com a distância até seu vizinho;
 - ② Os nós enviam uns para os outros seus vetores de distância;
 - ③ Quando recebe um vetor, o nó compara com os custos que conhece;
 - ④ Se o custo até o vizinho mais o custo no vetor recebido for menor do que o valor no seu próprio vetor, esse custo é atualizado e a nova rota é atualizada na tabela de roteamento.

Algoritmo Vetor de Distância

Camada de Rede

Profa. Kalinka Branco

Introdução

Roteamento

Endereços IP

Tabela do nó x

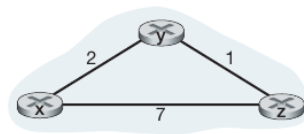
		Custo até					Custo até					Custo até		
		x	y	z			x	y	z			x	y	z
De	x	0	2	7	De	x	0	2	3	De	x	0	2	3
	y	∞	∞	∞		y	2	0	1		y	2	0	1
	z	∞	∞	∞		z	7	1	0		z	3	1	0

Tabela do nó y

		Custo até					Custo até					Custo até		
		x	y	z			x	y	z			x	y	z
De	x	∞	∞	∞	De	x	0	2	7	De	x	0	2	3
	y	2	0	1		y	2	0	1		y	2	0	1
	z	∞	∞	∞		z	7	1	0		z	3	1	0

Tabela do nó z

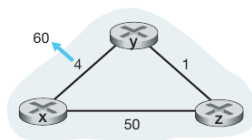
		Custo até					Custo até					Custo até		
		x	y	z			x	y	z			x	y	z
De	x	∞	∞	∞	De	x	0	2	7	De	x	0	2	3
	y	∞	∞	∞		y	2	0	1		y	2	0	1
	z	7	1	0		z	3	1	0		z	3	1	0



- **Iterativo:**
 - Continua até que os nós não troquem mais informações;
 - *Self-terminating*: não há sinal de parada.
- **Assíncrono:**
 - Os nós não precisam trocar informações simultaneamente.
- **Distribuído:**
 - Cada nó se comunica apenas com os seus vizinhos, diretamente conectados.
- **Estrutura de dados da Tabela de Distância:**
 - Cada nó tem sua própria tabela;
 - Linha para cada possível destino;
 - Coluna para cada roteador vizinho.

- **Problema da contagem até infinito:**

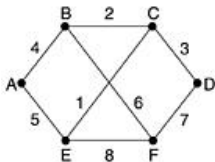
- Aumento do custo ou falhas são propagadas muito lentamente;



- Os valores para y eram até de 4 até x e de 1 até z e para z eram de 1 até y e de 5 até x.
- O nó y detecta a mudança e atualiza o valor até x no seu vetor comparando 60 com o caminho passando por z $5 + 1$, escolhendo 6 passando por z.
- O nó z recebe a mudança de y e atualiza o valor até x no seu vetor comparando 50 com o caminho passando por y $6 + 1$, escolhendo 7 passando por y.
- Esse processo é repetido por 44 vezes até chegarem nos valores corretos.

- *Link State* (LS);
- **Dinâmico**;
- **Global**: topologia de rede e custo dos enlaces são conhecidos por todos os nós:
 - Implementado via “*link state broadcast*”;
 - Todos os nós têm a mesma informação.
- **Funcionamento**:
 - ① Descobre seus vizinhos e seus endereços de rede e calcula o custo até cada um deles;
 - ② Cria um pacote que informa tudo o que foi descoberto e calculado e envia para todos os outros roteadores;
 - ③ Calcula o caminho mais curto até cada um dos roteadores (Algoritmo de Dijkstra).
- **Convergência**: após k iterações, conhece o caminho de menor custo para k destinos.

- Exemplo de sub-rede e pacotes do algoritmo de roteamento para cada um dos nós:



(a)

	Link	State	Packets		
A	B	C	D	E	F
Seq.	Seq.	Seq.	Seq.	Seq.	Seq.
Age	Age	Age	Age	Age	Age
B 4	A 4	B 2	C 3	A 5	B 6
E 5	C 2	D 3	F 7	C 1	D 7
	F 6	E 1		F 8	E 8

(b)

- Seq*: sequência usada para identificar pacotes duplicados;
- Age*: idade decrementada após cada transmissão para identificar/descartar pacotes antigos.

- Problemas do mundo real:
 - Roteadores não são todos idênticos;
 - As redes não são homogêneas na prática.
- Escala:
 - Uma projeção afirma que até 2020 teremos 50 bilhões de dispositivos conectados à Internet;
 - Não é possível armazenar todos os destinos em uma única tabela de rotas;
 - Mudanças na tabela de rotas congestionariam os enlaces.
- Autonomia administrativa:
 - Internet = rede de redes;
 - Cada administração de rede pode querer controlar o roteamento na sua própria rede.

- Agrega roteadores em regiões chamadas de Sistemas Autônomos (AS);
- Roteadores no mesmo AS rodam o mesmo protocolo de roteamento:
 - Protocolo de roteamento Intra-AS;
- Roteadores em diferentes AS podem rodar diferentes protocolos de roteamento:
 - Protocolo de roteamento Inter-AS;

Roteamento Intra-AS e Inter-AS

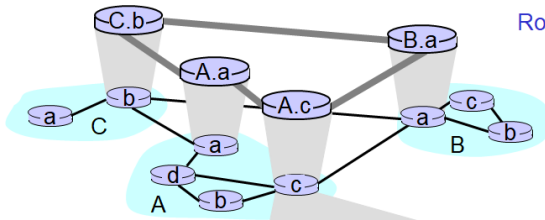
Camada de Rede

Prof. Kalinka Branco

Introdução

Roteamento

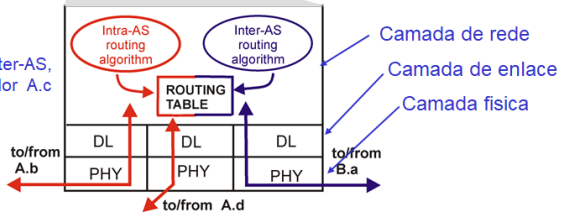
Endereços IP



Roteadores de Borda

- realizam roteamento inter-AS entre si
- realizam roteamento intra-AS com outros roteadores do mesmo AS

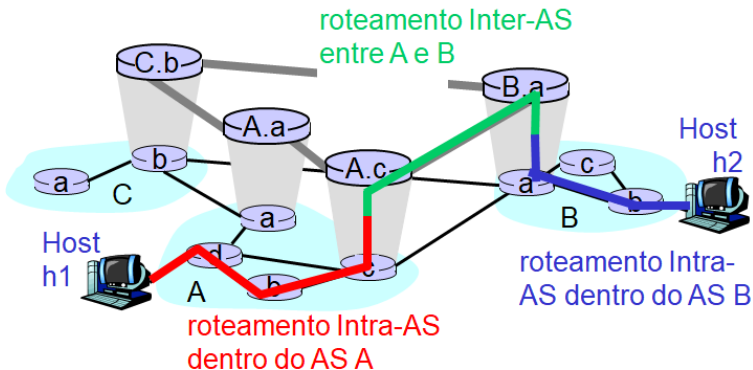
Roteamento inter-AS, intra-AS no roteador A.c

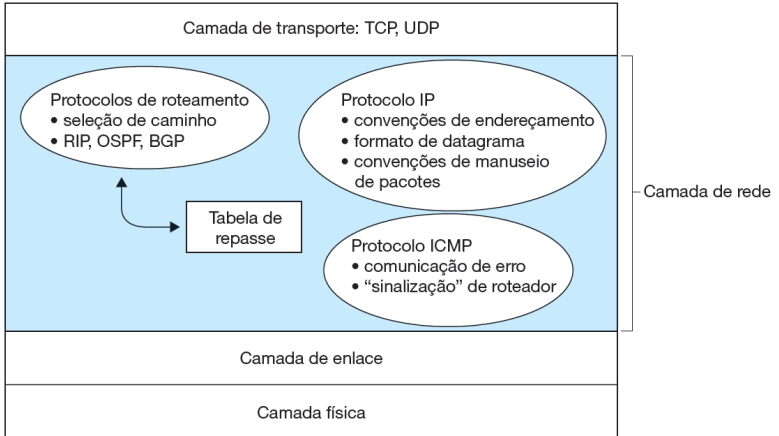


Roteadores de borda

- Roteadores de interface de um AS;
- Rodam protocolos de roteamento intra-AS com os outros roteadores do AS;
- Responsáveis por enviar mensagens para fora do AS:
 - Rodam protocolo de roteamento inter-AS com outros roteadores de borda.

Exemplo: envio de mensagem do *host* h1 para o *host* h2.





- **Endereço IP:** identificador de 32 bits (IPv4) para interfaces de roteadores e *hosts*.

Um endereço IPv4 (notação decimal com pontos)

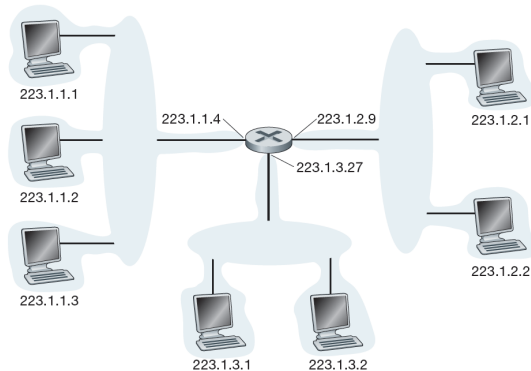
172 . 16 . 254 . 1

↓ ↓ ↓ ↓
 10101100 .00010000 .11111110 .00000001

Um byte=Oito bits

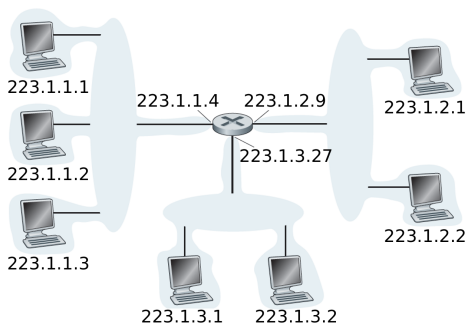
Trinta e dois bits (4 x 8) ou 4 bytes

- **Interface:** conexão entre roteador ou *host* e enlace físico.
 - Roteador tem tipicamente múltiplas interfaces;
 - *Hosts* podem ter múltiplas interfaces;
 - Endereços IP são associados com interfaces, não com o *host* ou com o roteador.

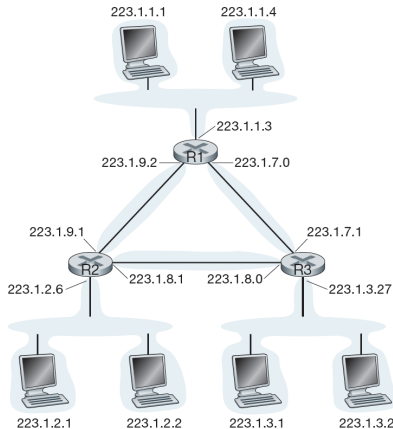


- **Endereço IP:**
 - Parte de rede (bits mais significativos);
 - Parte de *host* (bits menos significativos).
- **O que é uma rede?** (na perspectiva do endereço)
 - As interfaces de dispositivos com a mesma parte de rede no endereço IP podem fisicamente se comunicar sem o auxílio de um roteador.

- Exemplo: uma rede consistindo de 3 redes IP (para endereços IP começando com 223, os primeiros 24 bits são o endereço de rede):



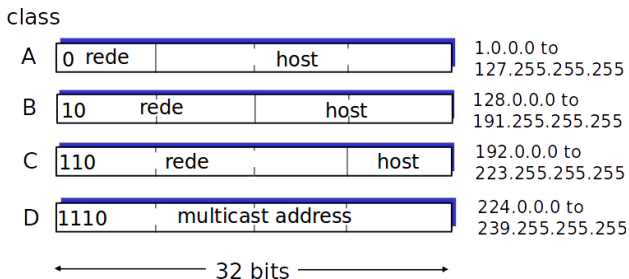
- Como encontrar as redes:
 - Separe cada interface de roteadores e *hosts*;
 - Crie ilhas de redes isoladas;
 - Use a técnica de nuvens.
- Sistema com seis redes interconectadas:



- Endereços especiais:

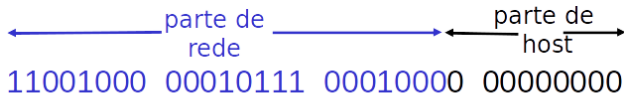
0 0		Este host		
0 0	...	0 0	Host	Um host nesta rede
1 1				Broadcast na rede local
Rede	1 1 1 1	...	1 1 1 1	Broadcast em uma rede distante
127	(Qualquer coisa)			Loopback

Endereçamento *Classful*:



- Classe A endereça $2^7 = 128$ redes e $(2^{24}) - 2 = 16777214$ *hosts* por rede;
- Classe B endereça $2^{14} = 16384$ redes e $(2^{16}) - 2 = 65534$ *hosts* por rede;
- Classe C endereça $2^{21} = 2097152$ redes e $(2^8) - 2 = 254$ *hosts* por rede.

- Endereçamento *Classful*:
 - Uso ineficiente do espaço de endereçamento, exaustão do espaço de endereços;
 - Ex.: rede de Classe B aloca endereços para 65000 *hosts*, mesmo se só existem 2000 *hosts* naquela rede.
- CIDR (*Classless Interdomain Routing*):
 - A porção de endereço de rede tem tamanho arbitrário;
 - Formato do endereço: $A.B.C.D/x$, onde x é o número de bits na parte de rede do endereço.



200.23.16.0/23

Hosts:

- **Endereço fixo:** definido pelo administrador;
- **Endereço dinâmico por DHCP** (*Dynamic Host Configuration Protocol*): permite a atribuição dinâmica de endereços IP.
 - ① O *host* envia (por *broadcast*) mensagem “*DHCP discover*”;
 - ② O servidor DHCP responde com mensagem “*DHCP offer*”;
 - ③ O *host* solicita um endereço IP com mensagem “*DHCP request*”;
 - ④ O servidor DHCP envia um endereço com a mensagem “*DHCP ack*”.

Servidor DHCP:
223.1.2.5



Cliente recém-chegado



Descoberta DHCP

Origem: 0.0.0.0, 68
Destino: 255.255.255.255, 67
DHCPDISCOVER
Internet: 0.0.0.0
ID transação: 654

Oferta DHCP

Origem: 223.1.2.5, 67
Destino: 255.255.255.255, 68
DHCPOFFER
Internet: 223.1.2.4
ID transação: 654
ID servidor DHCP: 223.1.2.5
Vida útil: 3.600 s

Requisição DHCP

Origem: 0.0.0.0, 68
Destino: 255.255.255.255, 67
DHCPREQUEST
Internet: 223.1.2.4
ID transação: 655
ID servidor DHCP: 223.1.2.5
Vida útil: 3600 s

ACK DHCP

Origem: 223.1.2.5, 67
Destino: 255.255.255.255, 68
DHCPACK
Internet: 223.1.2.4
ID transação: 655
ID servidor DHCP: 223.1.2.5
Vida útil: 3.600 s

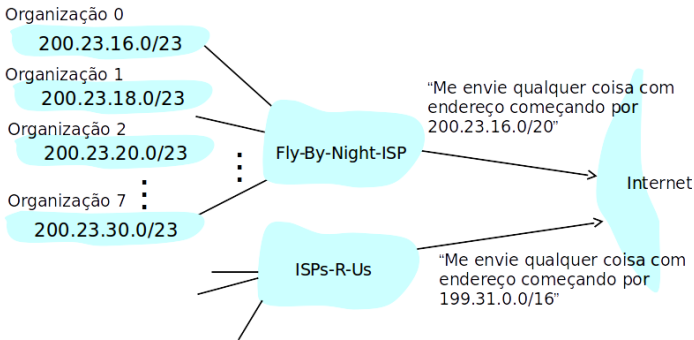
Rede (porção de rede):

- Obter uma parte do espaço de endereços do seu ISP (*Internet Service Provider*):

Bloco do ISP	<u>11001000</u>	<u>00010111</u>	<u>00010000</u>	00000000	200.23.16.0/20
Organização 0	<u>11001000</u>	<u>00010111</u>	<u>00010000</u>	00000000	200.23.16.0/23
Organização 1	<u>11001000</u>	<u>00010111</u>	<u>00010010</u>	00000000	200.23.18.0/23
Organização 2	<u>11001000</u>	<u>00010111</u>	<u>00010100</u>	00000000	200.23.20.0/23
...
Organização 7	<u>11001000</u>	<u>00010111</u>	<u>00011110</u>	00000000	200.23.30.0/23

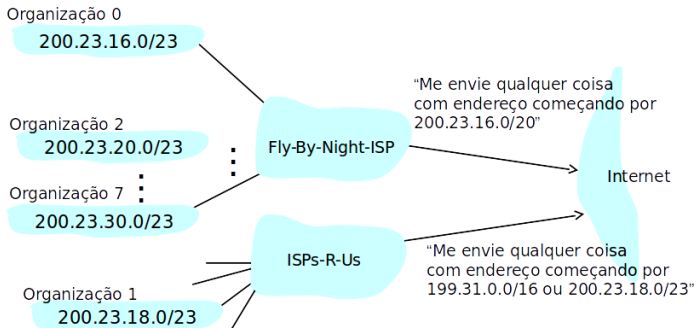
O endereçamento hierárquico permite uma propagação de rotas mais eficiente:

- Obter uma parte do espaço de endereços do seu ISP:



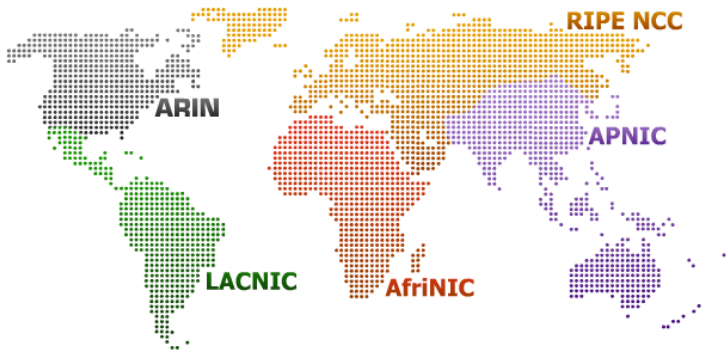
ISPs-R-Us tem uma rota mais específica para a organização 1:

- Obter uma parte do espaço de endereços do seu ISP:

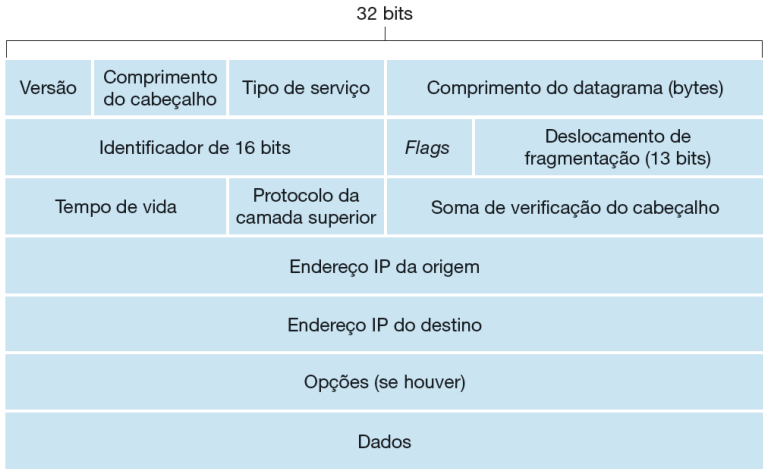


- Como o ISP obtém seu bloco de endereço?
- Por meio do **ICANN** (*Internet Corporation for Assigned Names and Numbers*):
 - Aloca endereços;
 - Gerencia DNS;
 - Atribui nomes de domínios.

- Alocação de endereços é realizada pela **IANA** (departamento do ICANN): *The Internet Assigned Numbers Authority*;
- Por meio dos **RIRs**: *Regional Internet Registry* (Registros Regionais da Internet).



- Formato do datagrama IPv4:

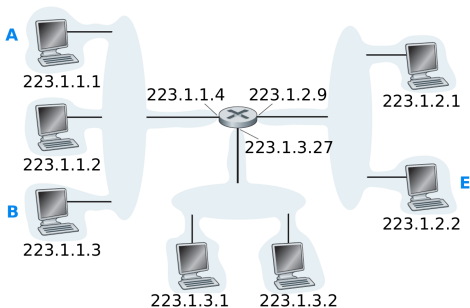


Outros campos	223.1.1.1	223.1.1.3	Dados
---------------	-----------	-----------	-------

- Envio do datagrama:
- Os endereços de origem e destino não mudam durante a transmissão.

Tabela de roteamento de A.

Rede destino	Próximo roteador	Número de hops
223.1.1		1
223.1.2	223.1.1.4	2
223.1.3	223.1.1.4	2

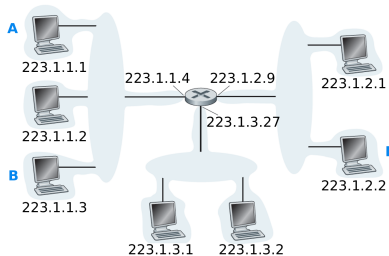


Outros campos	223.1.1.1	223.1.1.3	Dados
---------------	-----------	-----------	-------

- Envio do datagrama:
 - A verifica o endereço de rede de B e descobre que está na mesma sub-rede;
 - Repassa o pacote para a camada de enlace de dados;
 - Se o endereço físico de B não é conhecido, usa o ARP para descobri-lo;
 - A camada de enlace de A envia o quadro com o datagrama para B;

Tabela de roteamento de A.

Rede destino	Próximo roteador	Número de hops
223.1.1		1
223.1.2	223.1.1.4	2
223.1.3	223.1.1.4	2

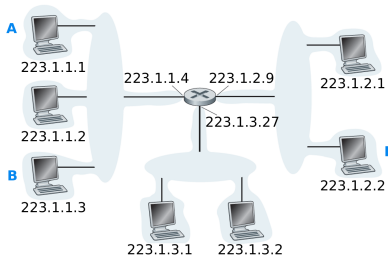


Outros campos	223.1.1.1	223.1.2.2	Dados
---------------	-----------	-----------	-------

- Envio do datagrama:
 - A verifica o endereço de rede de E e descobre que está em uma sub-rede diferente;
 - A consulta sua tabela de roteamento e descobre que o pacote deve ser enviado para o roteador no endereço 223.1.1.4;
 - A repassa o pacote para a camada de enlace, que envia o datagrama em um quadro para o roteador;

Tabela de roteamento de A.

Rede destino	Próximo roteador	Número de hops
223.1.1		1
223.1.2	223.1.1.4	2
223.1.3	223.1.1.4	2

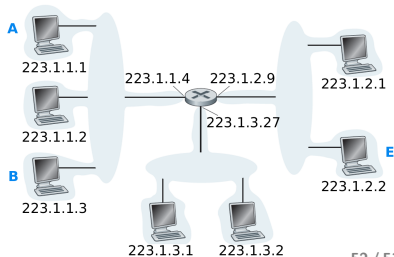


Outros campos	223.1.1.1	223.1.2.2	Dados
---------------	-----------	-----------	-------

- Recebimento do datagrama:
 - Roteador recebe o datagrama da sua camada de enlace;
 - Roteador consulta sua tabela de roteamento e descobre que o pacote deve ser enviado para o E;
 - Roteador verifica o endereço de rede de E e descobre que está na mesma sub-rede de sua interface com endereço 223.1.2.9;
 - Roteador repassa o pacote para a camada de enlace, que envia o pacote em um quadro para E;

Tabela de roteamento do roteador.

Rede destino	Endereço interface	Número de hops
223.1.1	223.1.1.4	1
223.1.2	223.1.2.9	1
223.1.3	223.1.3.27	1



Exercício

Considere a rede mostrada a seguir e admita que cada nó inicialmente conheça os custos até cada um de seus vizinhos. Como ficariam as tabelas de distâncias do nó z para o algoritmo Vetor de Distância? E como ficariam os pacotes do algoritmo Estado de Enlace? Encontre também o menor caminho de acordo com o algoritmo de Dijkstra do nó u para o nó z.

