

Anotações dos slides de InfraCom

‘Módulo 4’

Objetivos :

- Compreender os **princípios** por trás dos serviços da camada de rede:
 - **Modelos de serviço** da camada de rede.
 - **Encaminhamento (forwarding)** versus **Roteamento**.
 - **Como um roteador funciona**.
 - **Roteamento** (seleção de caminho).
 - Lidando com a **escalabilidade**.
 - Tópicos avançados: **IPv6**, **mobilidade**.
- **Instanciação, Implementação na Internet**.

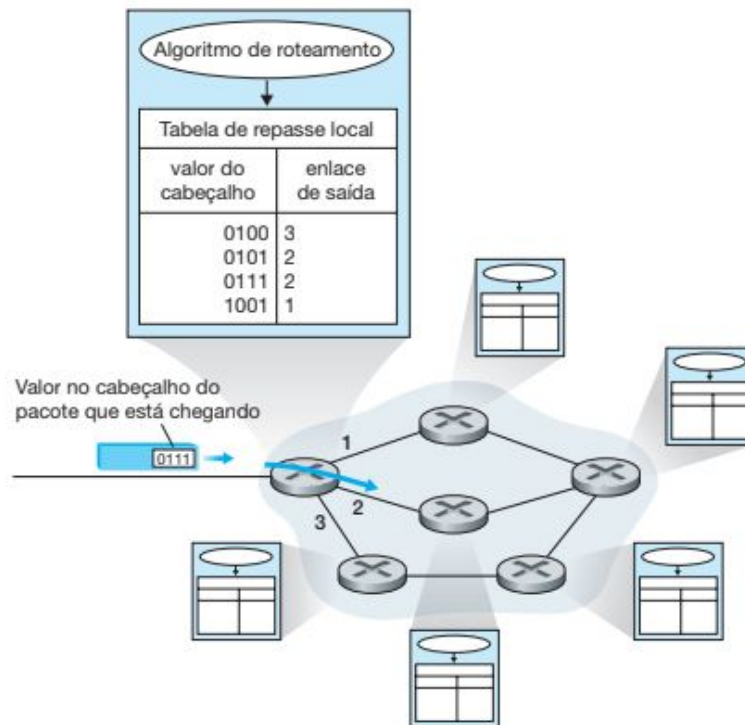
4.1 Introdução

- A camada de rede transporta o segmento do **hospedeiro emissor** para o **hospedeiro receptor**.
 - No lado emissor, a camada de rede **encapsula** segmentos em datagramas.
 - No lado receptor, a camada de rede **desencapsula** os datagramas, e entrega os segmentos para a camada de transporte.
- Protocolos da **camada de rede** estão presentes em **todos os hospedeiros e roteadores**.
- Roteador **examina campos do cabeçalho** dos **datagramas IP** que passam por ele.

4.1.1 Repasse e roteamento

- Principais funções da camada de rede:
 - **Roteamento**: **determina rota** a ser tomada por pacotes da **fonte ao destino**.
 - **Forwarding(repasse)**: **encaminha pacotes** que chegam ao roteador **para a saída apropriada** do roteador.

- Analogia:
 - Roteamento: processo de **planejamento** de uma viagem da **origem ao destino**.
 - Forwarding: processo de **finalização completa de uma "escala"** na viagem para o destino.



- **Estabelecimento de conexão**
 - 3ª função importante de algumas arquiteturas de redes:
 - por exemplo: redes ATM, X.25 e frame relay.
 - **Antes do envio do fluxo de datagramas**, os dois hosts com auxílio dos roteadores intermediários **estabelecem uma conexão virtual**.
 - **Roteadores participam explicitamente** do processo.
 - Serviço de conexão da camada de rede e da camada de transporte
 - **Rede:** entre **2 hospedeiros**
 - Transporte: entre **2 processos**

4.1.2 Modelos de serviço de rede

O modelo de serviço de rede **define as características** do transporte de dados fim a fim entre uma borda da rede e a outra, isto é, **entre sistemas finais remetente e destinatário**.

Qual o modelo de serviço para transporte direcionado dos datagramas do emissor para o receptor ?

- Exemplos de serviço para datagramas individuais:
 - **Entrega garantida**
 - **Entrega garantida com limite de atraso**
- Exemplos de serviços para fluxo de datagramas:
 - **Entrega de datagramas na ordem correta**
 - **Largura de banda mínima garantida**
 - Restrições sobre mudança de **espaçamento entre pacotes sucessivos** (**jitter** = variação de atraso)

Arquitetura de rede	Modelo de serviço	Garantia de largura de banda	Garantia contra perda	Ordenação	Temporização	Indicação de congestionamento
Internet	Best effort	Nenhuma	Nenhuma	Qualquer ordem possível	Não mantida	Nenhuma
ATM	CBR	Taxa constante	Sim	Na ordem	Mantida	Não haverá congestionamento
ATM	VBR	Taxa garantida	Sim	Na ordem	Mantida	Não haverá congestionamento
ATM	ABR	Mínima garantida	Nenhuma	Na ordem	Não mantida	Indicação de congestionamento
ATM	UBR	Nenhuma	Nenhuma	Na ordem	Não Mantida	Nenhuma

4.2 Redes de Circuitos Virtuais e de Datagramas

- **Redes de datagrama** provêem serviço de camada de rede **não orientado à conexão**.
- **Redes de circuitos virtuais (VC)** provêem serviço de rede **orientado à conexão**.
- Embora os **serviços de camada** de rede tenham algumas semelhanças com serviços oferecidos pela camada de transporte, **há diferenças cruciais**:
 - **Serviço**: host para host.
 - **Sem escolha**: a rede provê ou um serviço não orientado à conexão ou um serviço orientado à conexão, **não ambos**.
 - **Implementação**: nos roteadores no núcleo da rede, bem como nos sistemas finais.

4.2.1 Redes de circuitos virtuais (VC)

- “caminho da fonte para o destino se comporta de forma semelhante ao circuito da **linha telefônica**”
 - Em termos de **performance**
 - **Ações da rede** ao longo do caminho da fonte-destino
- Estabelecimento (teardown-encerramento) para cada chamada antes (depois) que os dados possam ser (foram) enviados.
- Cada pacote carrega um **identificador de circuito virtual** (não de endereço do host de destino)
- Todos os roteadores no caminho fonte-destino **mantêm “estado” para cada conexão passando**.
- **Recursos do roteador** (banda, buffers) podem ser **alocados** para o circuito virtual.

- **Implementação do VC**

- ◆ Um VC consiste em:

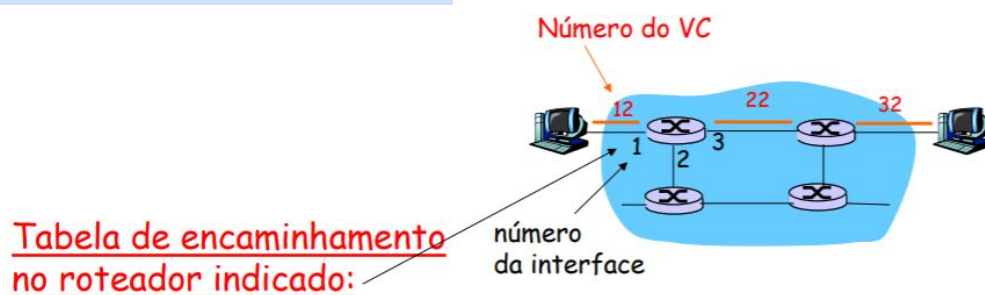
1. Caminho da fonte ao destino.
2. **Números de VCs**: 1 número para cada enlace ao longo do caminho.
3. Entradas nas **tabelas de encaminhamento** dos roteadores ao longo do caminho.

- ◆ Pacote pertencendo a um VC **carrega consigo um número de VC**.

- ◆ O número de VC deve ser **mudado em cada enlace**.

- Novo número de VC **obtido da tabela de encaminhamento**.

- **Tabela de encaminhamento**

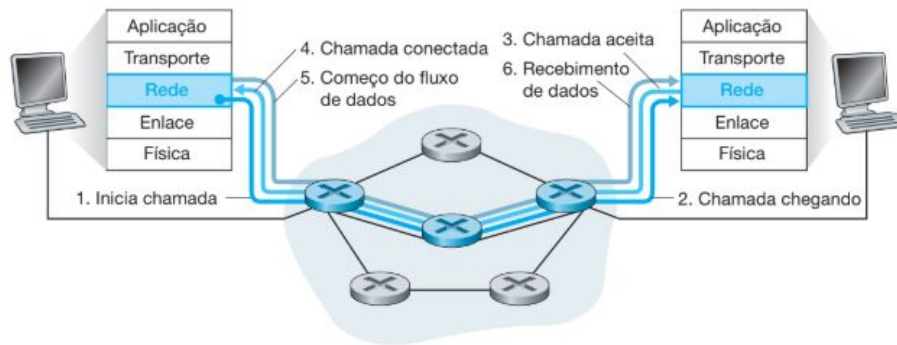


Roteadores mantêm informação de estado da conexão!

Interface de entrada	Nº do VC de entrada	Interface de saída	Nº do VC de saída
1	12	3	22
2	63	1	18
3	7	2	17

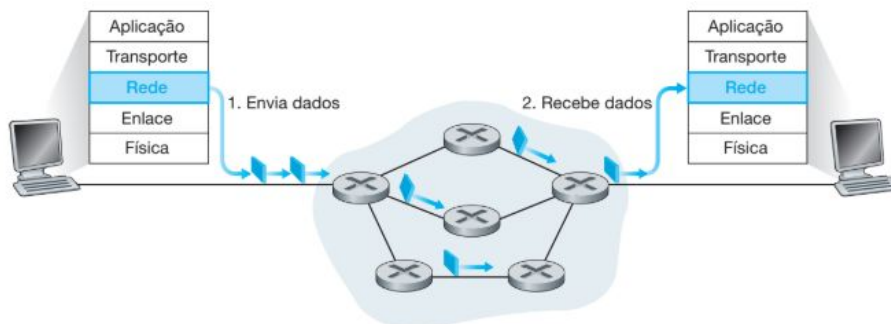
- **Protocolos de sinalização**

- Usados para estabelecer(setup), manter e encerrar(teardown) um VC
- Usados em redes ATM, frame-relay e X.25
- Não usados na internet atual



4.2.2 Redes de datagramas

- Não há estabelecimento de conexão/chamada na camada de rede.
- Os roteadores **não mantêm estado** sobre conexões fim-a-fim.
 - Não existe o conceito de "conexão" na rede de datagramas.
- Os pacotes são encaminhados com base no **endereço de destino**.
 - Pacotes entre mesmo par fonte-destino podem tomar **caminhos diferentes**.



- **Tabela de encaminhamento**

Faixa de endereços de destino	Interface de enlace
11001000 00010111 00010000 00000000 até 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 até 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 até 11001000 00010111 00011111 11111111 senão	2 3

+4 bilhões de entradas possíveis

- Prefixo mais longo correspondente

Prefixo do endereço	Interface de enlace
11001000 00010111 00010	0
11001000 00010111 00011000	1
11001000 00010111 00011	2
senão	3

Por qual interface o endereço será encaminhado?

- 11001000 00010111 00010110 10100001
 - Resposta: XXXXXXXXXX.
- 11001000 00010111 00011000 10101010
 - Resposta: XXXXXXXXXX.

4.2.3 Redes de datagramas ou VCs: Por quê?

✧ Internet (Datagramas)

- Dados trocados entre computadores
 - Serviço “elástico”, nenhum requisito de tempo estrito.
- End systems “*inteligentes*” (computadores)
 - Pode se adaptar, realizar controle, recuperação de erros...
 - Simplicidade dentro da rede, complexidade nas “bordas”.
- Diferentes tipos de enlace
 - Características diferentes.
 - Serviço uniforme difícil.

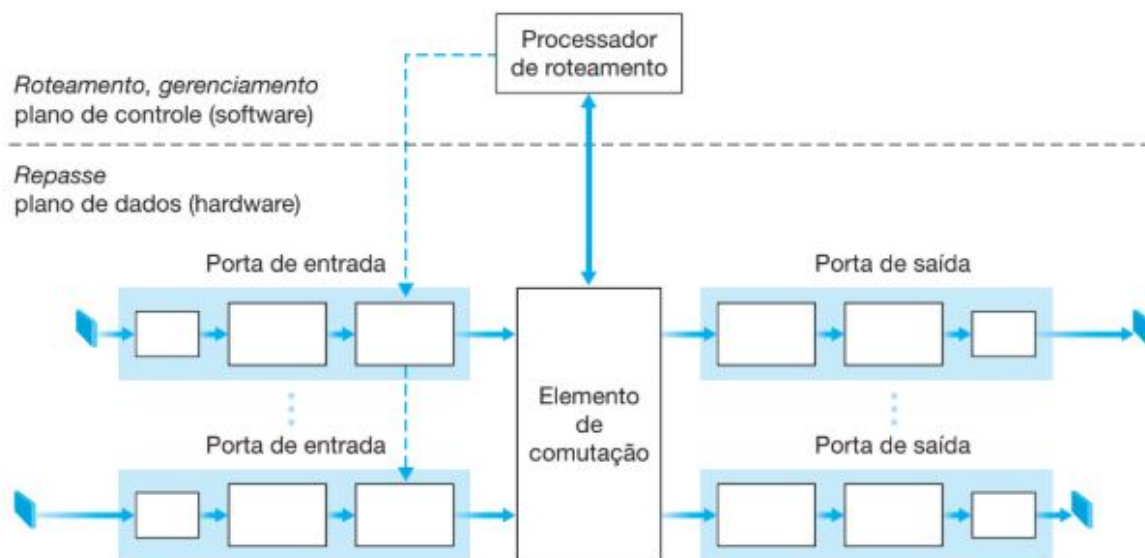
✧ ATM (VC)

- Evolução do telefone.
- Conversação humana:
 - Tempo estrito, requisitos de confiabilidade.
 - Necessário para serviços garantidos.
- End systems “*burros*”
 - Telefones
 - Complexidade dentro da rede.

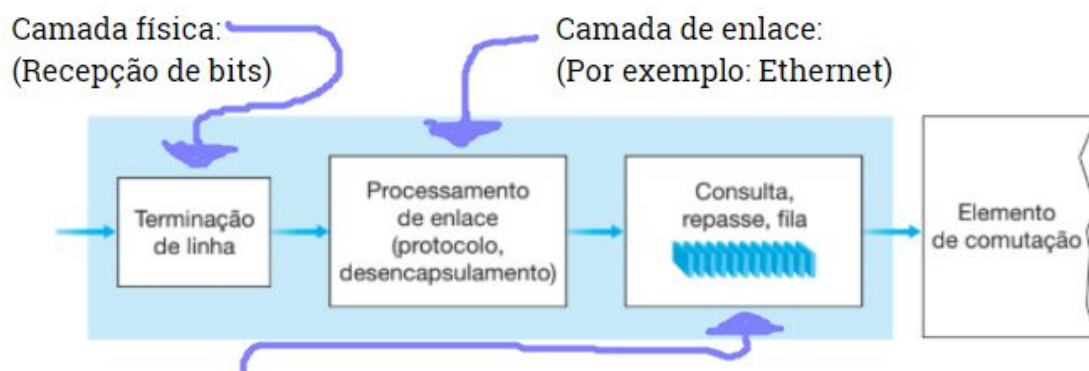
4.3 O que há dentro de um roteador?

2 funções principais do roteador:

- Executar algoritmos/protocolos de roteamento (RIP, OSPF, BGP).
- Encaminhar (repassar) datagramas do enlace de entrada para o enlace de saída.



4.3.1 Processamento de entrada

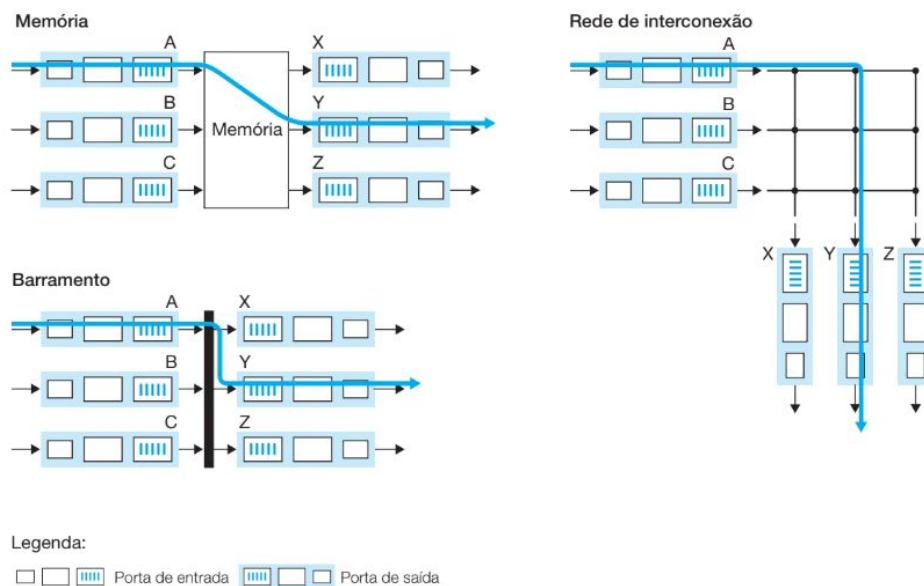


Comutação descentralizada :

- Dado o destino do datagrama, **procurar porta de saída** usando a **tabela de encaminhamento** na memória da porta de entrada
- Objetivo: completar processamento na porta de entrada na **"velocidade da linha"**
- **Fila (queuing):** para o caso de datagramas chegarem **mais rápido que a taxa de encaminhamento** no elemento de comutação (switch fabric)

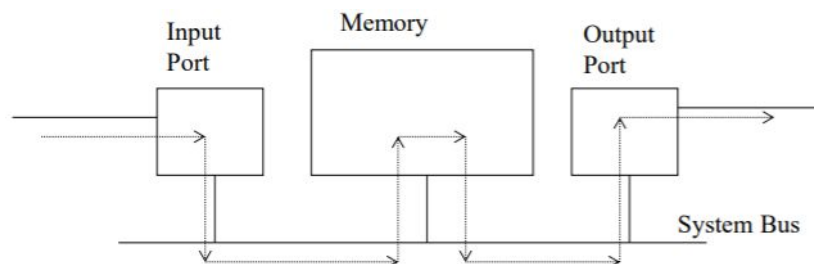
4.3.2 Elemento de comutação

É por meio do elemento de comutação que os pacotes são comutados (isto é, repassados) de uma porta de entrada para uma porta de saída. A comutação pode ser realizada de inúmeras maneiras.



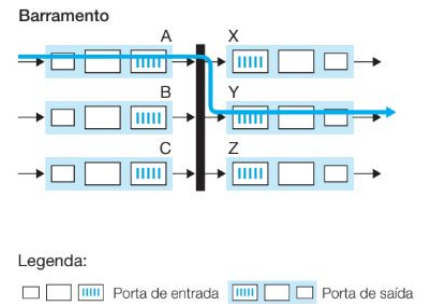
♦ Comutação via Memória

- Roteadores de 1ª geração:
 - Eram **computadores tradicionais** nos quais a **comutação** entre as portas de entrada e de saída era **realizada sob o controle direto da CPU** (processador de roteamento).
 - Pacote copiado para a **memória do processador**. O processador de roteamento então **extraía o endereço de destino** do cabeçalho, **consultava a porta de saída apropriada** na tabela de repasse e **copiava o pacote para os buffers da porta de saída**.
 - **Velocidade limitada** pela **banda passante da memória** (cada datagrama cruza dois barramentos).



◆ Comutação via Barramento

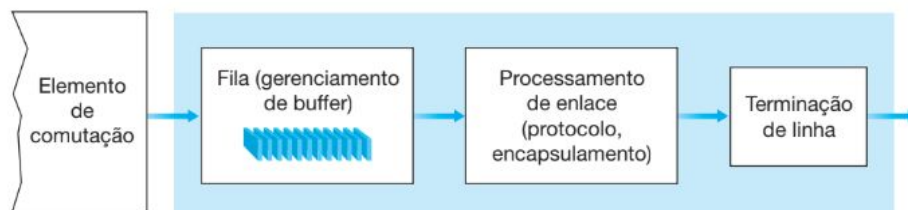
- As portas de entrada transferem um pacote diretamente para a porta de saída por um **barramento compartilhado** sem a intervenção do processador de roteamento.
- Para isso, **um rótulo é inserido antes do pacote** indicando a porta de saída e o pacote é transmitido para o barramento. Ele é **recebido por todas as portas** de saída, mas somente a porta correta manterá o pacote. **O rótulo é então removido na porta de saída**, pois só é usado dentro do comutador para atravessar o barramento.
- **Bus contention**: velocidade de comutação **limitada pela banda passante do barramento**, pois cada pacote precisa atravessar o único barramento.
- Barramento de 1 Gbps, Cisco 1900: velocidade suficiente para roteadores de acesso e de empresas (inadequado para acesso regional e para o backbone).



◆ Comutação via uma rede de Interconexão (interna ao roteador)

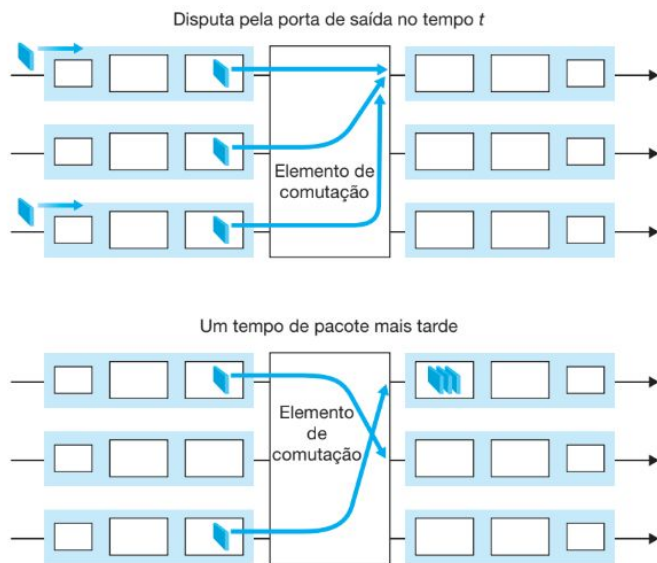
- “Vencer” limitações de banda do barramento.
- **Complexa, cara, porém rápida.**
- Redes Banyan, redes de interconexão inicialmente desenvolvidas para conectar processadores em sistemas multiprocessados.
- Design avançado: fragmentação de datagramas em células de tamanho fixo, comuta células através do equipamento.
- Cisco 12000: comuta Gbps através de rede de interconexão.

4.3.3 Processamento de saída



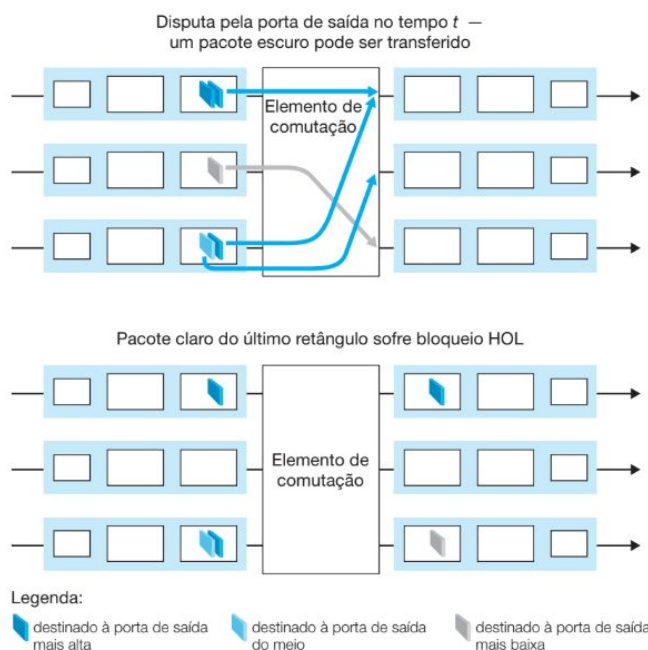
- **Buffering (fila)** requerido quando datagramas chegam do elemento de comutação (switch fabric) **mais rápido que a taxa de transmissão**.
- **Disciplina de escalonamento** escolhe datagramas da fila a serem transmitidos (prioridades).

4.3.4 Fila da porta de saída



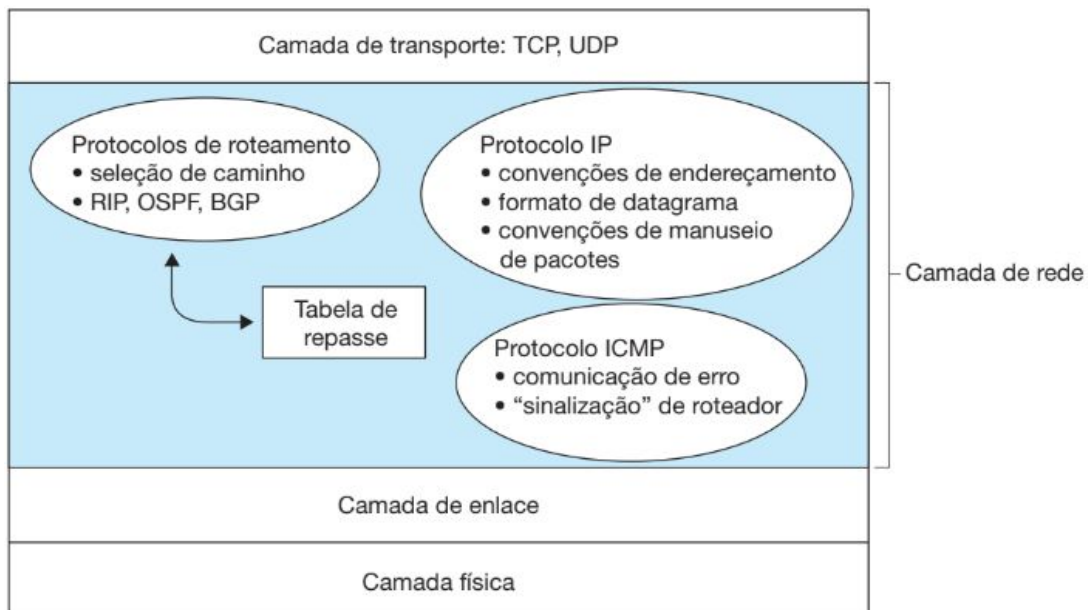
- “**Buferização**” quando taxa de chegada através do comutador excede a velocidade da linha de saída.
- **Queueing/enfileiramento** (atraso) e perda devido ao transbordamento (**overflow**) do **buffer** da porta de saída!

4.3.5 Fila da porta de entrada



- Elemento de comutação (switch fabric) é **mais lento** que as **portas de entrada** combinadas → enfileiramento pode ocorrer nas portas de entrada.
- **Head-of-the-Line (HOL) blocking**: datagrama enfileirado na frente da fila **previne** outros de **serem encaminhados**.
- **Atraso de fila e perda** devido ao transbordamento (**overflow**) do **buffer** da porta de entrada.

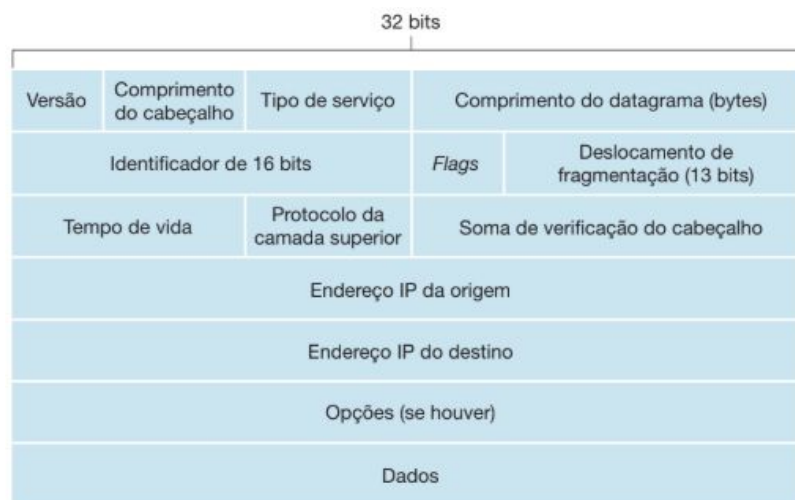
4.4 IP: Internet Protocol (Protocolo da Internet)



Três componentes mais importantes da camada de rede.

- O primeiro é o [protocolo IP](#).
- O segundo é o [componente de roteamento](#), que determina o caminho que um datagrama segue desde a origem até o destino.
- O componente final da camada de rede é um dispositivo para [comunicação de erros](#) em datagramas e para [atender requisições](#) de certas informações de camada de rede.

4.4.1 Formato do datagrama

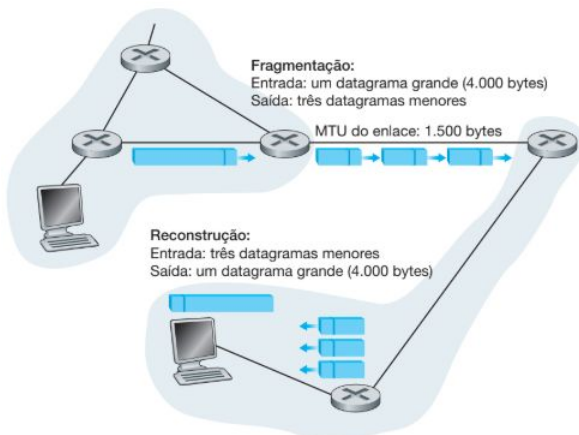


Datagrama IPv4

Principais campos do IPv4:

- **Número da versão:** 4 bits que especificam a versão do protocolo IP do datagrama. Com isso o roteador sabe como tratar o restante do datagrama.
- **Comprimento do cabeçalho:** 4 bits que informam o tamanho do cabeçalho (**em bytes**). No geral o tamanho do cabeçalho é de 20 bytes, porém pode ser adicionado um número **variável de opções** (no cabeçalho do datagrama IPv4).
- **Tipo de serviço:** Usados para diferenciar alguns tipos de datagramas IP, como aqueles que requerem baixo atraso, alta vazão, etc.
- **Comprimento do datagrama:** Comprimento total do datagrama (**cabeçalho + dados**) medido **em bytes**. Tem 16 bits de comprimento, gerando até 65.535 bytes de tamanho, mas dificilmente passa de 1500 bytes.
- **Identificador (16 bits), flags (3 bits), deslocamento de fragmentação (13 bits):** Campos relacionados com a **fragmentação do IP**.
- **Tempo de vida (TTL: Time to Live):** é incluído para garantir que datagramas não fiquem **circulando infinitamente pela rede**. Esse campo é decrementado de uma unidade cada vez que o datagrama é processado por um roteador. **Se o campo TTL chegar a 0, o datagrama deve ser descartado.**
- **Protocolo:** Usado quando o datagrama chega ao destino final. Ele serve para informar a camada de transporte qual o **protocolo de transporte** que esse datagrama será destinado.
- **Soma de verificação de cabeçalho (Checksum):** Usada para **identificar erros de bits** no datagrama IP recebido. Após fazer o processo de comparar se o valor calculado é igual ao valor recebido, o resultado é armazenado no campo de soma de verificação, caso não, normalmente se descarta esse pacote.
- **Endereço IP de origem e destino:** Quando uma origem cria um datagrama, insere-se seu endereço IP no campo de endereço de origem e insere o endereço de destino no campo de endereço de destino.
- **Opções:** Campo que permite que o cabeçalho IP seja estendido. Por variar o tamanho do datagrama não há como saber a priori onde os dados começam, além de **variar o tempo de processamento dos roteadores**.
- **Dados(carga útil):** Contém o **segmento da camada de transporte** (TCP ou UDP) a ser entregue ao destino **ou outro tipos de dados como mensagens ICMP**.

♦ Fragmentação e Remontagem IP



- Enlace da rede possui **MTU (tamanho máximo de transferência)** – maior quadro possível no enlace.
 - MTUs variam conforme o enlace (Ex. Ethernet: 1500 Bytes, ATM: 53 Bytes...)
- Datagrama IP “grande” **dividido (“fragmentado”)** dentro da rede.
 - 1 Datagrama torna-se **vários datagramas**.
 - “Remontados” somente **no destino final**.
 - Bits do cabeçalho IP usados para identificar, ordenar fragmentos relacionados.

- Exemplo:

Datagrama de 4000 Bytes (Dados + Cabeçalho)

Comprimento = 4000	ID = 777	FragFlag = 0	Deslocamento = 0
--------------------	----------	--------------	------------------

MTU = 1500 Bytes

Fragmentos :

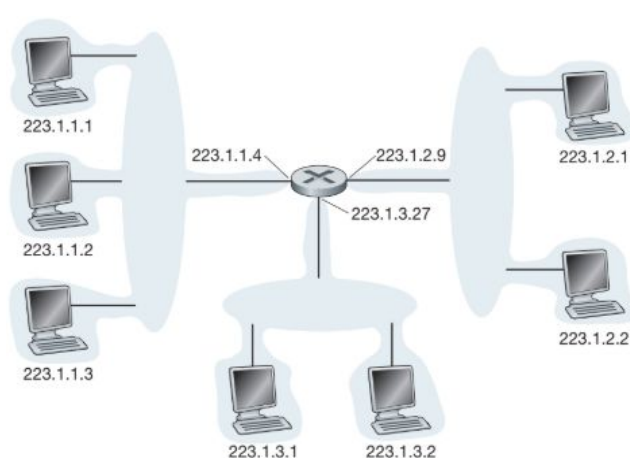
Comprimento = 1500 (Dados = 1480 bytes)	ID = 777	FragFlag = 1	Deslocamento = 0
Comprimento = 1500 (Dados = 1480 bytes)	ID = 777	FragFlag = 1	Deslocamento = 185 $1480 \div 8 = 185$
Comprimento = 1040 (Dados = 1020 bytes)	ID = 777	FragFlag = 0	Deslocamento = 370 $2960 \div 8 = 370$

- O campo de deslocamento indica a partir de qual byte (dividido por 8) os dados devem ser inseridos
- O campo FragFlag indica se haverá mais fragmentos do mesmo datagrama.

4.4.2 Endereçamento IPv4

A hierarquia de endereçamento IP:

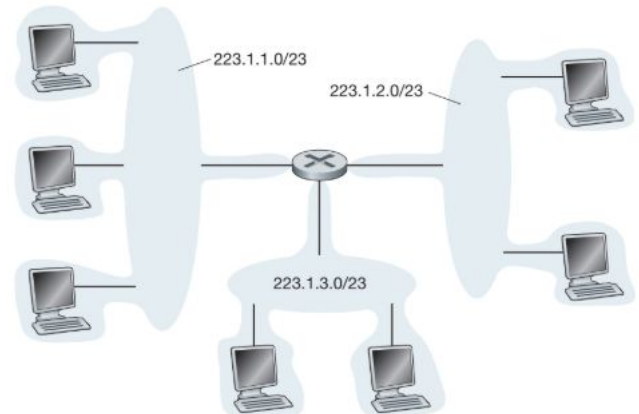
- Por questões de eficiência, o endereço IP é definido por duas partes, prefixo e sufixo:
 - O **prefixo** é responsável por determinar **a rede que o computador está acoplado**.
 - O **sufixo** é responsável por identificar **um computador acoplado em cada rede**.
- **Endereço IP: identificador de 32 bits** para interface do host/roteador
- **Interface:** conexão entre host/roteador e enlace físico
 - **Roteadores** tipicamente **possuem múltiplas interfaces**.
 - **Host** tipicamente possui **uma única interface**.
 - **1 endereço IP por interface**.



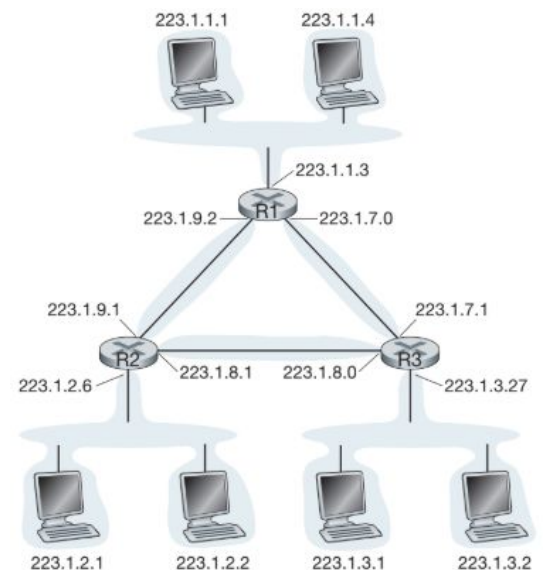
$$223.1.1.1 = \underbrace{11011111}_{223} \underbrace{00000001}_1 \underbrace{00000001}_1 \underbrace{00000001}_1$$

◆ Subredes

- Endereço IP:
 - Parte da subrede (bits de ordem mais elevada).
 - Parte do host (bits de ordem mais baixa).
- O que é uma subrede ?
 - Interfaces de dispositivos com **mesma parte de subrede do endereço IP**.
 - Dispositivos podem fisicamente alcançar os outros sem ajuda de um roteador.



- **RECEITA:**
 - Para determinar a subrede, retirar cada interface do seu host ou roteador, criando ilhas de redes isoladas. Cada rede isolada é chamada de subrede.
 - Quantas subredes tem na imagem ao lado?
 - Resposta: XXXXXXXXXX.



◆ Classes

- Classe A: 8 bits para a subrede
- Classe B: 16 bits para a subrede
- Classe C: 24 bits para a subrede

← 32 Bits →				
Class				Range of host addresses
A	0	Network	Host	1.0.0.0 to 127.255.255.255
B	10	Network	Host	128.0.0.0 to 191.255.255.255
C	110	Network	Host	192.0.0.0 to 223.255.255.255
D	1110	Multicast address		224.0.0.0 to 239.255.255.255
E	1111	Reserved for future use		240.0.0.0 to 255.255.255.255

- **Desperdício de endereços IP**

- ↳ Classe C permite até $2^8 - 2 = 254$ hosts/interfaces
- ↳ Classe B permite até 65534 hosts/interfaces
- ✗ Para organizações com até 2000 hosts é necessário alocar um endereço de rede classe B levando ao desperdício de mais de 63000 endereços que não serão usados !

- ◆ **CIDR (Classless InterDomain Routing)**

- A porção no endereço que representa a subrede tem **tamanho arbitrário**.
- Formato do endereço: a.b.c.d/x, onde x é o número de bits na porção do endereço que **representa a subrede**.

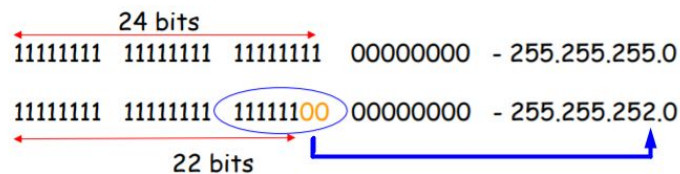
200.23.16.0/23

Parte da Subrede

Parte do Host

11001000 00010111 00010000 00000000

- Exemplos
 - 192.168.0.0/24 representa os 256 endereços IPv4 de 192.168.0.0 até 192.168.0.255 (este último é o endereço de broadcast da rede)
 - 192.168.0.0/22 representa 1024 endereços IPv4 de 192.168.0.0 até 192.168.3.255 (este último é o endereço de broadcast da rede)
- Representação alternativa na forma decimal com pontos
 - 192.168.0.0/24 pode ser escrito como 192.168.0.0/255.255.255.0
 - 192.168.0.0/22 pode ser escrito como 192.168.0.0/255.255.252.0



- ◆ **Obtenção de um endereço IP**

Como os hosts obtêm endereço IP?

- Manualmente:
 - Colocado em um arquivo de configuração
 - Wintel: control-panel → network → configuration → tcp/ip → properties
 - UNIX: /etc/rc.config
- Automaticamente:

- ◆ **DHCP (Dynamic Host Configuration Protocol)**

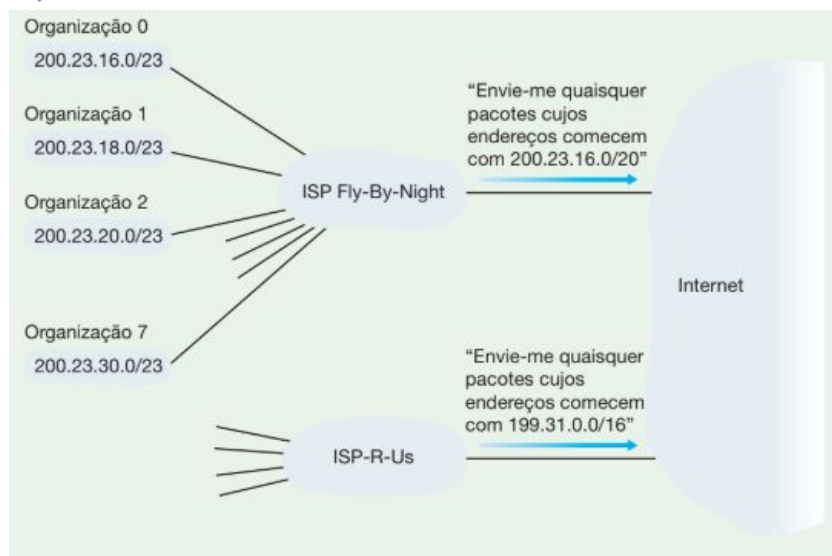
- Protocolo que permite que um hospedeiro obtenha um endereço IP de maneira automática (**plug and play**).
- Obtém **dinamicamente** um endereço **através de um servidor**.
- Endereço pode ser **temporário** ou **fixo**.

- A rede obtém a parte de subrede do endereço IP pegando a porção alocada do espaço de endereço do seu ISP

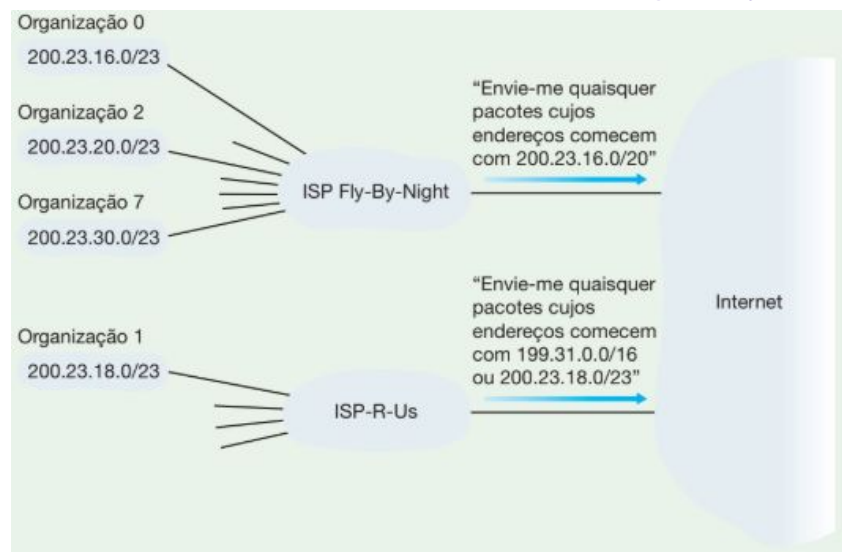
Bloco do ISP 11001000 00010111 00010000 00000000 200.23.16.0/20

Organização 0	<u>11001000 00010111 0001</u> 0000 00000000 200.23.16.0/23
Organização 1	<u>11001000 00010111 0001</u> 0010 00000000 200.23.18.0/23
Organização 2	<u>11001000 00010111 0001</u> 0100 00000000 200.23.20.0/23
...
Organização 7	<u>11001000 00010111 0001</u> 1110 00000000 200.23.30.0/23

- Endereçamento Hierárquico
 - Agregação de rotas
 - Endereçamento hierárquico permite uma eficiente divulgação de informações de roteamento:

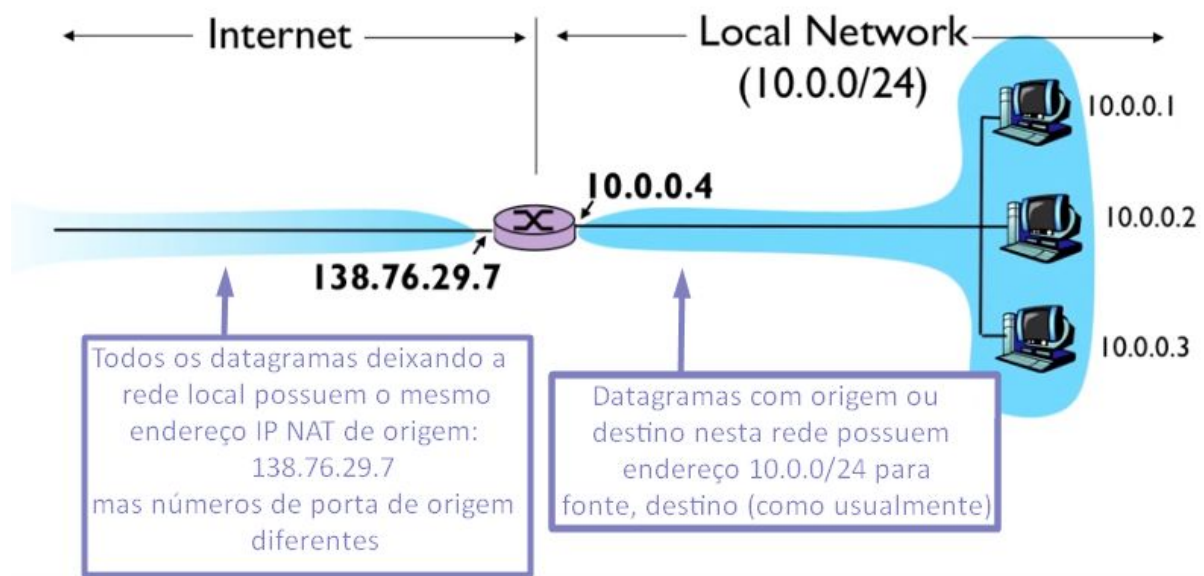


- Rotas mais específicas
 - ISPs-R-Us possui uma rota mais específica para a Organização 1



- ICANN: Internet Corporation for Assigned Names and Numbers
 - Uma autoridade global que tem responsabilidade de gerenciar os espaços de endereços IP.
 - Fornece ao ISP blocos de endereços.
 - Aloca endereços.
 - Gerencia DNS.
 - Atribui nomes de domínio, resolve disputas

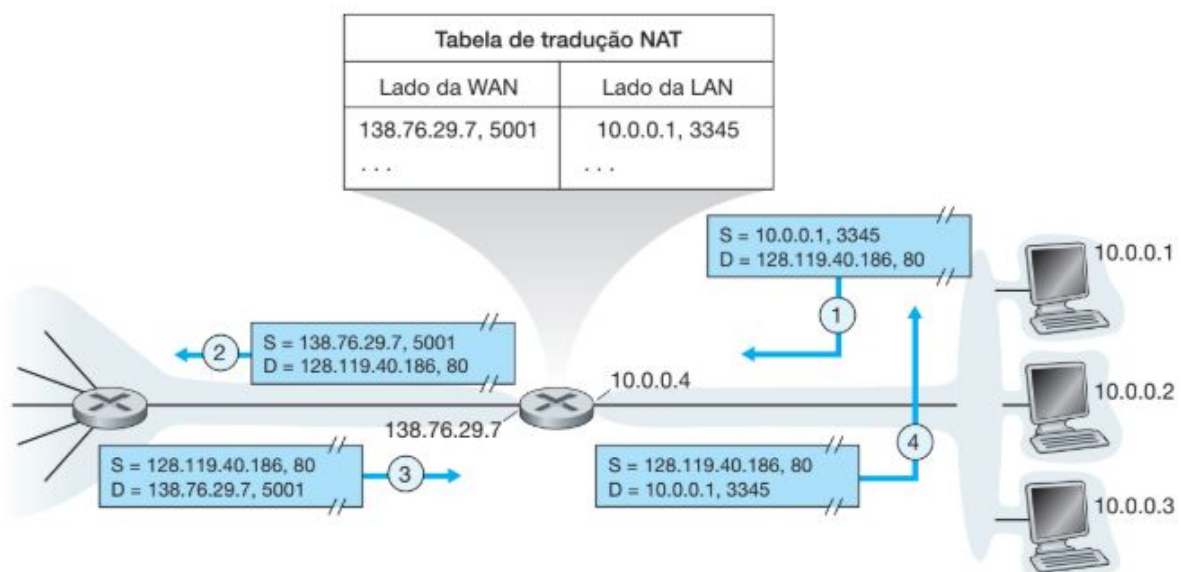
♦ NAT: Tradução de Endereços na Rede



- Motivação: rede local usa somente **um único endereço IP** quando há necessidade de falar com o **mundo externo**:
 - Range de endereços não são necessários do ISP: **apenas um endereço IP para todos os dispositivos**.
 - Pode **mudar endereço** de dispositivos na rede local **sem necessidade de notificar o mundo externo**.
 - Pode **mudar o ISP** **sem mudar o endereço dos dispositivos na rede local**.
 - **Dispositivos dentro da rede local** não são explicitamente endereçáveis, ou seja, **não são visíveis pelo mundo externo** (um pouco **mais de segurança**).

Um roteador NAT deve:

- Datagramas **saindo** para o mundo externo:
 - **substituir** [endereço IP fonte, número de porta] de todo datagrama sendo enviado para o mundo externo por [endereço IP NAT, novo número de porta]
 - Clientes/servidores remotos responderão usando [endereço IP NAT, novo número de porta] como **endereço de destino**.
 - **Recordar** (na tabela de tradução NAT) os pares de equivalência [endereço IP fonte, número de porta] \equiv [endereço IP NAT, novo número de porta]
- Datagramas **chegando** do mundo externo:
 - **substituir** [endereço IP NAT, novo número de porta] no campo “destino” de todos os pacotes provenientes do mundo externo com a informação correspondente [endereço IP fonte, número de porta] armazenada na tabela NAT



1. Host 10.0.0.1, 3345 Envia datagrama para 128.119.40.186, 80	2. Roteador NAT muda endereço de origem do datagrama de 10.0.0.1, 3345 para 138.76.29.7, 5001, e atualiza tabela
3. Resposta chega com endereço de destino: 138.76.29.7, 5001	4. Roteador NAT muda endereço de destino do datagrama de 138.76.29.7, 5001 para 10.0.0.1, 3345

- Campo “número de porta” de 16 bits:

- +60.000 conexões simultâneas com um único endereço IP externo!
- NAT é controverso:
 - Roteadores devem processar somente até a camada 3
 - Viola o argumento “fim-a-fim”
 - Possibilidade do uso de NAT deve ser levada em conta por projetistas de aplicações (ex. aplicações P2P)
 - Esgotamento de endereços deve ser resolvido pelo IPv6

4.4.3 Protocolo de Mensagens de Controle da Internet (ICMP)

- Usado por hosts e roteadores para comunicar informações do nível rede
 - Reportagem de erro: host, rede, porta, protocolo inalcançável
 - Echo request/reply (usado pelo ping)
- Camada de rede “acima do” IP:
 - ICMP mensagens transportadas em datagramas IP
- Mensagem ICMP: tem um campo de tipo e um campo de código. Além disso, contém o cabeçalho e os primeiros 8 bytes do datagrama IP que causou o erro (de modo que o remetente pode determinar o datagrama que causou o erro).

Tipo ICMP	Código	Descrição
0	0	resposta de eco (para ping)
3	0	rede de destino inalcançável
3	1	hospedeiro de destino inalcançável
3	2	protocolo de destino inalcançável
3	3	porta de destino inalcançável
3	6	rede de destino desconhecida
3	7	hospedeiro de destino desconhecido
4	0	repressão da origem (controle de congestionamento)
8	0	solicitação de eco
9	0	anúncio do roteador
10	0	descoberta do roteador
11	0	TTL expirado
12	0	cabeçalho IP inválido

♦ Traceroute e ICMP

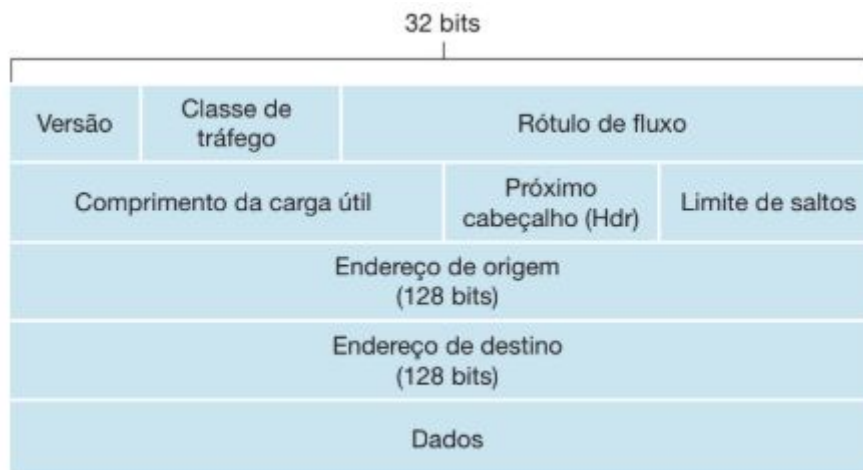
- Fonte envia uma série de segmentos UDP para o destino
 - O 1º possui TTL = 1
 - O 2º possui TTL= 2, e assim por diante...
 - Número de porta improvável
- Quando enésimo datagrama chega ao enésimo roteador:
 - Roteador descarta datagrama

- E envia para a fonte um mensagem ICMP (tipo 11, código 0)
- Mensagem inclui nome do roteador e endereço
- Quando a mensagem ICMP chega, fonte calcula o RTT
- Traceroute faz isto 3 vezes
- **Critério de parada**
 - Segmento UDP eventualmente **chegará ao host de destino**
 - O destino **retorna a mensagem ICMP** “porta de destino inalcançável” (tipo 3, código 3)
 - Quando **a fonte recebe esta mensagem ICMP**, para.

4.4.4 IPv6

- **Motivação inicial:** espaço de endereçamento de 32 bits estará completamente alocado em pouco tempo.
- **Motivação adicional:**
 - Formato do cabeçalho ajuda a fazer **processamento/encaminhamento mais rápido**
 - Mudanças no cabeçalho para **facilitar QoS**
 - **Formato do datagrama IPv6:**
 - Cabeçalho de **tamanho fixo de 40 bytes**
 - **Fragmentação não é permitida!**

♦ Cabeçalho IPv6



- **Classe de tráfego (Priority):** Identifica **prioridade** dos datagramas
- **Rótulo de fluxo (flow label):** Identifica datagramas no mesmo “fluxo.” (conceito de “fluxo” não é bem definido).
- **Próximo cabeçalho (Next header):** identifica protocolo da camada superior para “data” → dados (possui outras funções também)

♦ **Outras mudanças com relação ao IPv4**

- **Checksum**: inteiramente **removido** para reduzir tempo de processamento em cada salto
- **Options**: permitido, mas **fora do cabeçalho**, indicado pelo campo "Next Header"
- **ICMPv6: nova versão do ICMP**
 - Tipos **adicionais de mensagens**, ex. "Pacote muito grande".
 - Funções de **gerenciamento de grupos multicast**.

◆ Endereçamento IPv6 - Notação

★ Cada um formado por **8 grupos com 4 dígitos hexadecimais**

- 2001:0db8:85a3:08d3:1319:8a2e:0370:7334

★ 4 zeros em um grupo podem ser omitidos e substituídos por ::

- 2001:0db8:85a3:**0000**:1319:8a2e:0370:7334 equivale à
2001:0db8:85a3::**1319:8a2e:0370:7334**

★ Exemplo de endereços válidos e equivalentes

- 2001:0db8:0000:0000:0000:0000:1428:57ab (end. original)
- 2001:0db8:0000:0000:0000::1428:57ab (grupo de 4 zeros subst.)
- 2001:0db8:**0:0:0:0**:1428:57ab ("0000" por "0")
- 2001:0db8:0:0::1428:57ab (:: são equivalentes a 0000:0000)
- 2001:0db8::1428:57ab (:: são equivalentes a 0000:0000:0000:0000)
- 2001:**db8**::1428:57ab (zero à esquerda pode ser retirado)

★ As simplificações não podem gerar **ambiguidades** !

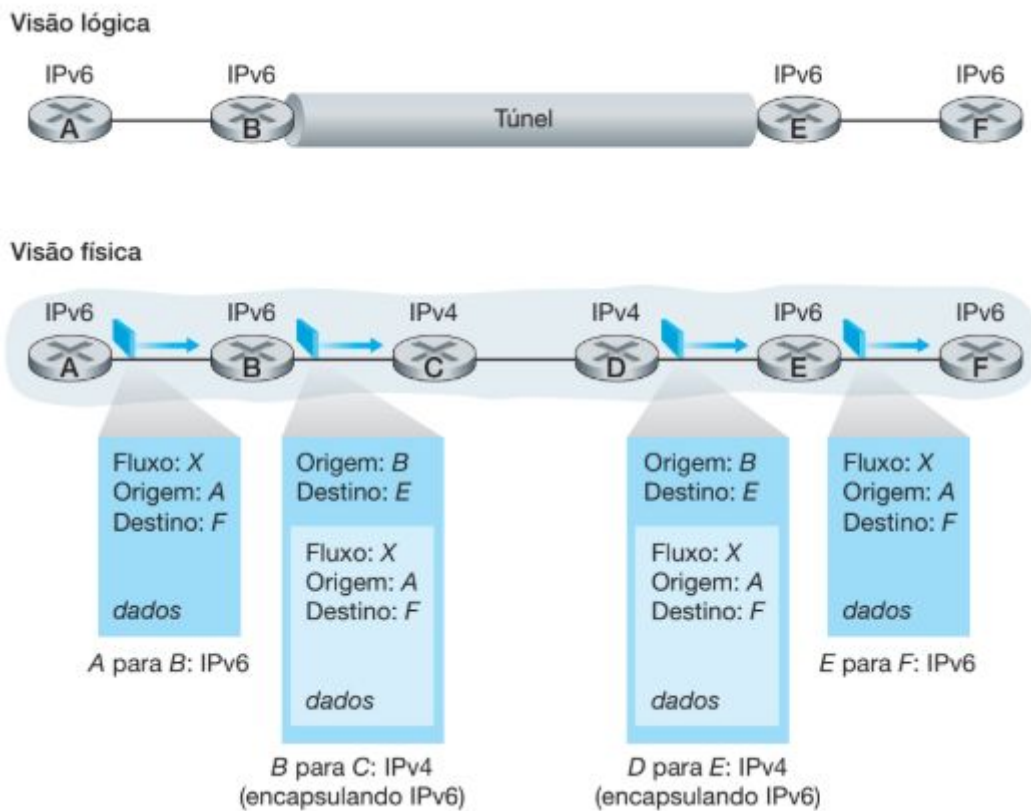
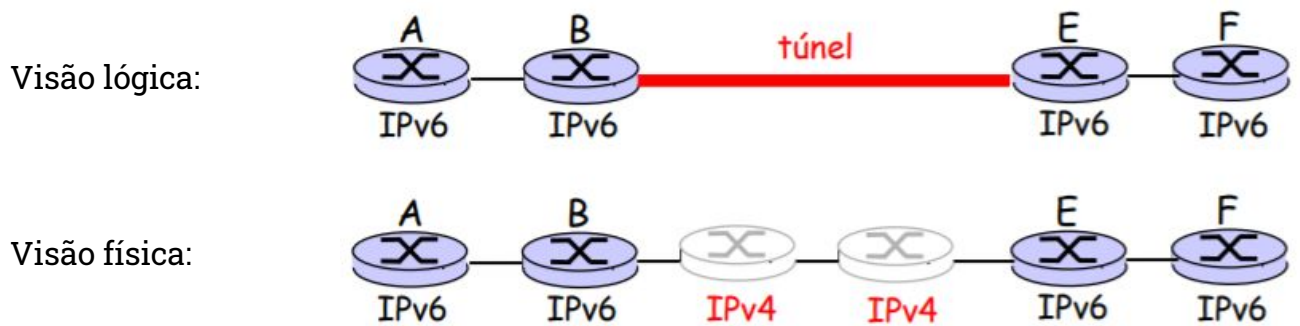
- Exemplo: 2001:C00:0:0:5400:0:0:9 → 2001:C00::5400::9
 - Pode significar:
 - 2001:C00:0:0:5400:0:0:9
 - 2001:C00:**0:0:0**:5400:**0**:9
 - 2001:C00:**0**:5400:**0:0:0**:9

★ Em uma URL devemos fazer como segue

- http://[2001:0db8:85a3:08d3:1319:8a2e:0370:7344]/
 - Qual a razão do **uso de colchetes** na URL ?
http://[2001:0db8:85a3:08d3:1319:8a2e:0370:7344]:443/
- Rede
 - Segue notação CIDR
2001:0db8:1234::/48 significa que rede possui faixa de endereços de
2001:0db8:1234:0000:0000:0000:0000:0000 até
2001:0db8:1234:FFFF:FFFF:FFFF:FFFF:FFFF

♦ Transição do IPv4 para o IPv6

- Todos os roteadores não podem ser atualizados simultaneamente
 - dia do mutirão de atualização é impossível.
 - Como a rede irá operar com roteadores IPv4 e IPv6 misturados dentro dela?
- **Tunelamento:** IPv6 transportado como “payload” em datagramas IPv4 entre roteadores IPv4



Questões

1) No roteador, a função roteamento é encaminhar pacotes que chegam ao roteador para a saída apropriada do roteador.

Resposta:

Falso, essa é a função forwarding, o roteamento determina a rota tomada pelos pacotes.

2) Nenhum modelo de serviço da camada de rede implementa o serviço de entrega garantida.

Resposta:

Falso, o ATM CBR tem entrega garantida.

3) Os serviços da camada de rede não são orientados a conexão.

Resposta:

Falso, redes de VC provêem serviço de rede orientado à conexão.

4) Restrição de jitter é um exemplo de modelo de serviço para datagramas individuais.

Resposta:

Falso, restrição de jitter afeta o fluxo dos datagramas.

5) Os roteadores usados em um VC mantêm estado da conexão.

Resposta:

Verdadeiro.

6) Em um VC, cada pacote carrega um identificador de circuito virtual e endereço do host destino.

Resposta:

Falso, cada pacote carrega apenas um identificador de circuito virtual.

7) O número de VC de um pacote deve ser mudado a cada enlace, sendo obtido através da tabela de encaminhamento.

Resposta:

Verdadeiro.

8) Os campos TTL (time to live) e opções sempre estão presentes em um datagrama IP.

Resposta:

Falso, o campo opções é opcional.

9) As vezes um datagrama IP é muito grande para passar por um enlace, uma das soluções para esse problema é fragmentar esse datagrama em vários datagramas.

Resposta:

Verdadeiro.

10) Em um enlace com MTU de 500 bytes, o máximo valor do campo dados em um pacote TCP/IP é 480 bytes, já que o cabeçalho do IP é 20 bytes.

Resposta:

Falso, o cabeçalho do TCP de 20 bytes também deve ser levado em consideração.

11) Em um endereço IP, o prefixo é responsável por determinar a rede a qual o computador está acoplado.

Resposta:

Verdadeiro.

12) O padrão de endereçamento IP CIDR (Class InterDomain Routing) propõe o uso de classes de endereçamento.

Resposta:

Falso, o CIDR (Class**LESS** InterDomain Routing) propõe exatamente o oposto.

13) A classe B do endereçamento IP reserva 16 bits para subrede e 16 bits para o host, já a classe A reserva 24 bits para subrede e 8 bits para o host.

Resposta:

Falso, a classe A reserva 8 bits para subrede e 24 bits para o host.

14) 192.168.0.0/24 pode ser escrito como 192.168.0.0/255.255.255.0.

Resposta:

Verdadeiro, os últimos 8 bits são setados para 0.

15) 192.168.0.0/20 pode ser escrito como 192.168.0.0/255.255.252.0.

Resposta:

Falso, o correto seria 192.168.0.0/255.255.240.0.

16) Em uma rede NAT vários computadores podem acessar a internet usando um mesmo endereço IP.

Resposta:

Verdadeiro.

17) Um roteador NAT consegue garantir 2^{16} conexões simultâneas sem violar o princípio fim-a-fim.

Resposta:

Falso, embora ele garanta esse número de conexões, ele viola o princípio fim-a-fim que os roteadores devem processar somente até a camada 3.

18) A motivação da implantação do IPv6 é apenas suprir a necessidade de mais endereços de internet com o esgotamento do IPv4.

Resposta:

Falso, mudanças no formato do cabeçalho ajudam a fazer processamento e encaminhamento mais rápidos e facilitam o QoS.

19) No IPv6 quando ocorre fragmentação do datagrama, uma mensagem ICMP é enviada para o próximo hop informando que os próximos pacotes estão fragmentados. Por isso não existem campos para denotar fragmentação no cabeçalho IPv6.

Resposta:

Falso, não existe fragmentação no IPv6, uma mensagem ICMP é enviada informando que o pacote está muito grande.

20) O endereço IPv6 2001:0db8:00b5:0000:0000:0000:1428:57ab pode ser abreviado para 2001:db8:b5::1428:57ab

Resposta:

Verdadeiro.

21) O endereço IPv6 2001:C010:0000:0000:54ba:0000:0000:912a pode ser abreviado para 2001:C010::54ba::912a

Resposta:

Falso, essa abreviação gera ambiguidades, uma abreviação correta seria 2001:C010:0:0:54ba::912a

22) O mutirão para troca do IPv4 para IPv6 está marcado para 21/12/2020 onde todos os usuários da internet deverão trocar seus aparelhos.

Resposta:

Falso, a troca para o IPv6 é progressiva, uma das soluções, é o tunelamento, onde o datagrama IPv6 é encapsulado como dados em um datagrama IPv4 para passar por uma rede IPv4.