

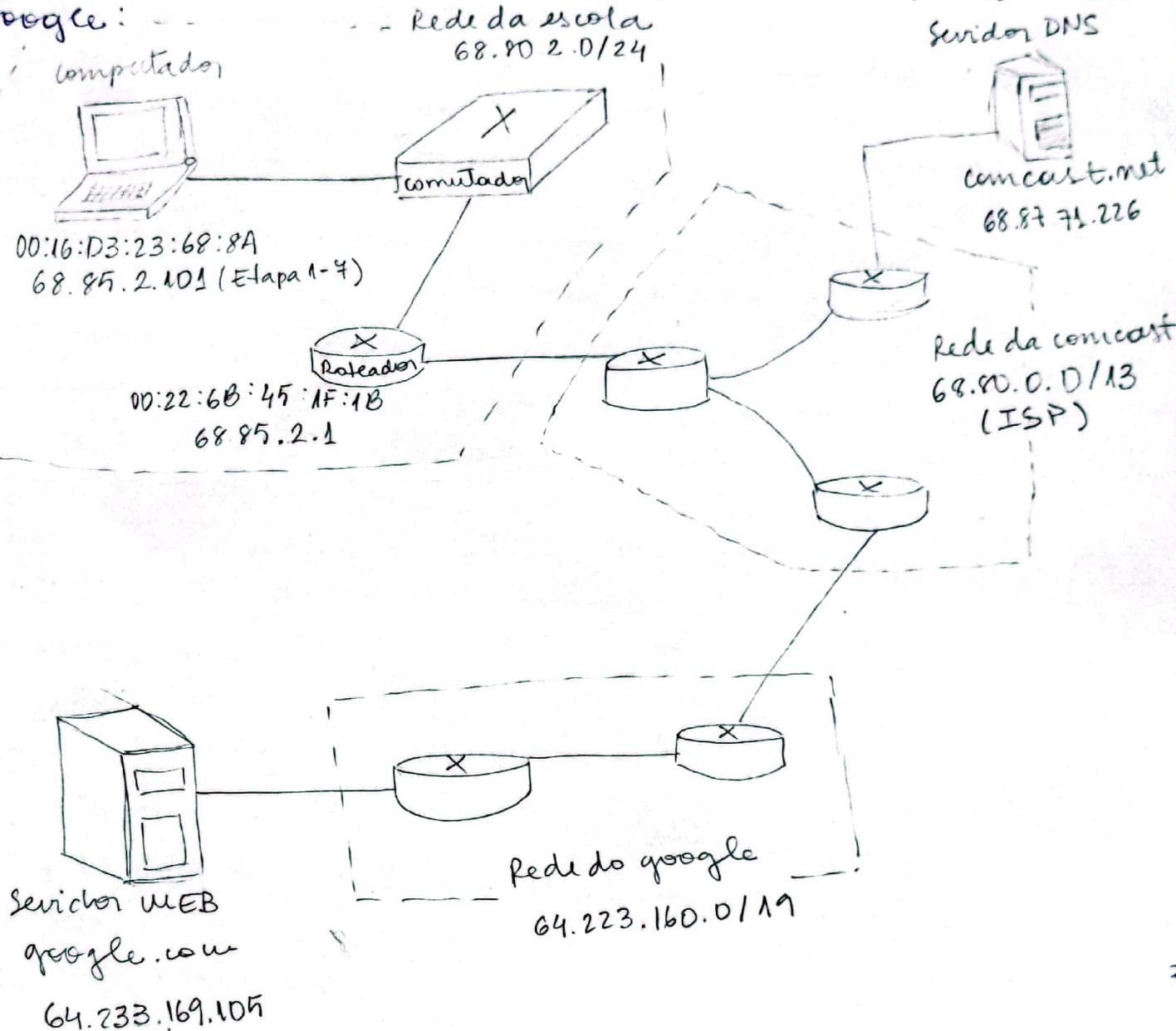
Redes de computadores - Prova. I.

Assuntos:

- Fundamentos
- Arquiteturas TCP/IP
- Redes Celulares
- Redes de sensores
- Internet das Coisas
- Redes industriais

5.7. Um dia na vida de uma solicitação de página WEB:

- Caso: estudante se conecta ao computador da escola e acessa o google:

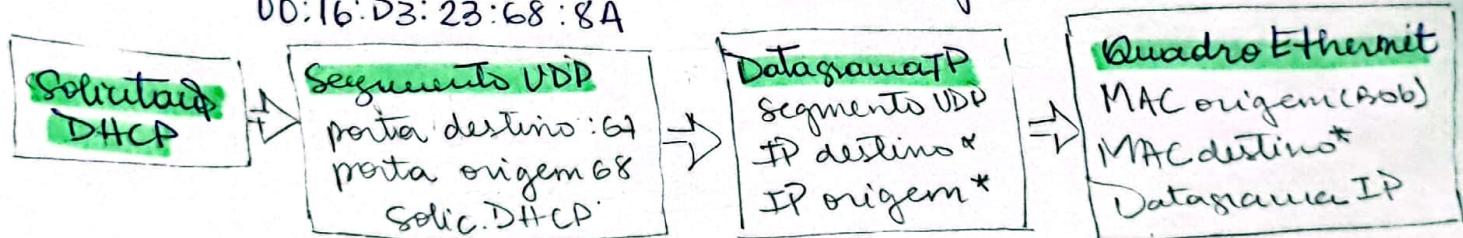


*) No primeiro momento, Bob não consegue fazer o download da página WEB porque não tem endereço IP, portanto, a primeira ação relacionada à rede, tomada pelo notebook, é executar o protocolo DHCP:

PRIMEIRA AÇÃO: Executar o protocolo DHCP para a obtenção do endereço IP e outras infos do servidor DHCP local.

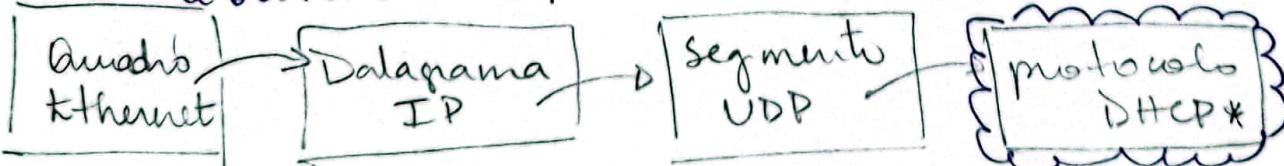
1// O sistema operacional do PC de Bob cria uma mensagem de solicitação DHCP e coloca dentro do segmento UDP com porta de destino 67 servidor DHCP e origem porta 68 cliente DHCP, o segmento então é encapsulado dentro de um datagrama IP de destino por difusão (255.255.255.255) e IP de destino origem 0.0.0.0 - já que o PC de Bob ainda não tem IP.

2// O datagrama IP contendo a solicitação DHCP é colocado dentro de um quadro Ethernet que possui como endereço de destino MAC FF:FF:FF:FF:FF:FF de modo que o quadro seja transmitido para todos os dispositivos conectados ao comutador (que se espera que tenha um servidor DHCP) e quadro de origem o MAC de Bob: 00:16:D3:23:68:8A



3// Após ser montado, o quadro Ethernet é enviado pelo PC do Bob para o comutador Ethernet, que envia o quadro para todas as portas de saída, incluindo a porta do roteador.

4// O roteador recebe o quadro Ethernet transmitido que possui endereço MAC de destino: 00:22:6B:45:1F:1B (roteador) e executa o desempacotamento do quadro Ethernet.



5// De posse da mensagem DHCP, o roteador utilizando

o endereço IP 68.85.2.0/24 dentro dele aloca o IP do bloco como:
na escola estão dentro do bloco os endereços IP usados
5.1. Supondo que o servidor DHCP designe o IP 68.85.2.101
ao PC de Bob
5.2. O servidor DHCP monta um novo quadro Ethernet:

A) Mensagem DHCP ACK:

• Continua:

- IP do PC do Bob: 68.85.2.101
- IP do DNS: 68.85.2.101
- IP do gateway: 68.85.2.1

B) A máscara (subrede): 68.85.2.0/24

C) O segmento UDP é colocado em um segmento UDP

D) O datagrama IP é colocado em um datagrama IP

net que tem MAC de origem e do roteador (00:22:6B:45:1F:
1B) e MAC de destino (00:16:D3:23:68:8A) PC de Bob.

Mensagem
DHCP
IP PC Bob
IP DNS
IP gateway
IP subrede

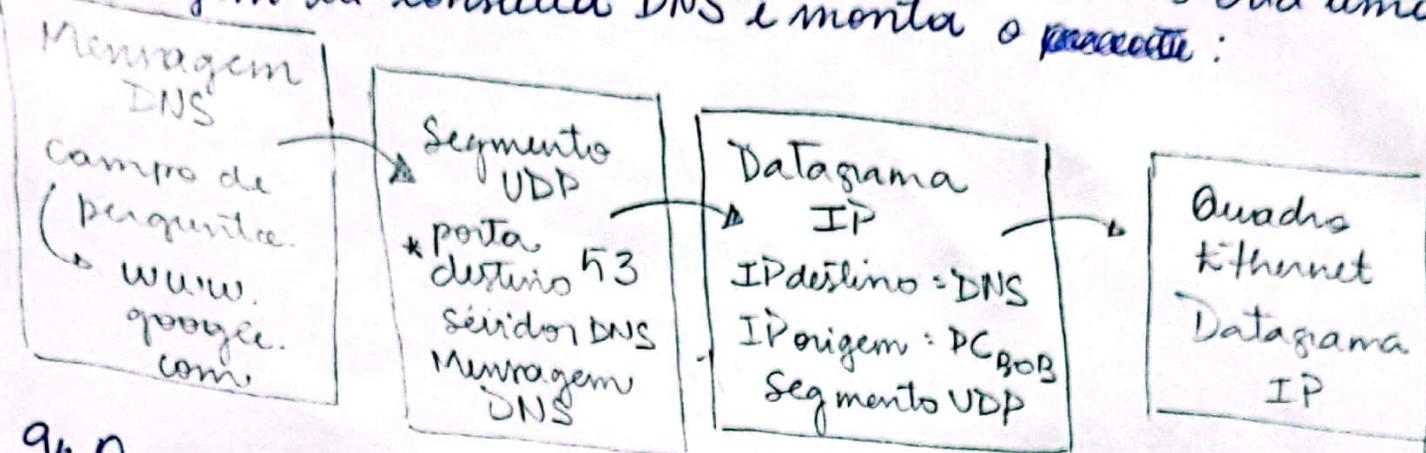


6// O quadro Ethernet contendo o ACK DHCP é enviado do roteador para o comutador que sabe como enviar para o PC de Bob porque realizou a autoaprendizagem a partir da mensagem de solicitação anterior

7// O notebook de Bob recebe o quadro Ethernet e envia a ACK DHCP, o cliente DHCP do computador de Bob grava o endereço IP próprio e o IP do servidor DNS, bem como o endereço IP do roteador de borda (68.85.2.1).

⇒ Assim, o PC de Bob está apto para iniciar a navegação WEB.

8) O sistema operacional do notebook de Bob envia uma mensagem de consulta DNS e monta o pacote:



9) O computador de Bob não sabe o endereço MAC do roteador da escola, e para descobrir ele precisará usar o protocolo ARP. O notebook insere a consulta ARP direcionada ao IP do roteador (68.85.2.1) no pacote Ethernet que será transmitido por difusão ao endereço MAC de destino (FF:FF:FF:FF:FF). Ele manda para o comutador que entrega a todos os dispositivos, incluindo o roteador de borda.

10) O roteador de borda recebe o quadro Ethernet, que contém a mensagem ARP, verifica que o IP de destino é o seu próprio e monta uma mensagem ARP contendo seu próprio endereço MAC (00:22:6B:45:1F:1B) insere em um quadro Ethernet com endereço de destino o MAC de Bob e envia para o comutador, que envia para o PC de Bob.

11) O PC de Bob extrai da resposta ARP o endereço MAC do roteador de borda

12) Finalmente, o PC de Bob pode enviar o quadro Ethernet com a consulta ao DNS, igual ao mostrado na 8:

Datagrama IP → IP destino : DNS

Quadro Ethernet → MAC destino : Roteador de borda

* Após a saída de descobrir o MAC do roteador, comece a saída de descobrir o IP do google:

4// O roteador de borda recebe o quadro ethernet e extrai o datagrama IP que contém o IP de destino (o servidor DNS) consulta a tabela de repasse e determina que deve ser enviado para o roteador da esquerda da rede Comcast, coloca o datagrama IP num quadro de enlace adequado e envia p/ o próximo roteador.

5// O roteador esquerdo da Comcast recebe o quadro e extrai o datagrama IP, examina o IP de destino e determina a interface de saída pela qual enviará o datagrama para o servidor DNS, consultando sua tabela de repasse.

6// Finalmente, o datagrama IP chega ao servidor DNS que entra na consulta DNS, procura o nome www.google.com na sua base de dados e encontra o REGISTRO DE RECURSO DNS que contém o IP 64.233.169.105 p/ o site da google.

O servidor DNS forma uma mensagem DNS resposta com o mapeamento entre nome de hospedeiro e IP e envia um datagrama, envia para o roteador, roteador, comutador, PC

14// O notebook de Bob finalmente tem acesso ao IP do google.

* Finalmente, Bob tem acesso ao google:

→ Interação Cliente - Servidor WEB (TCP e HTTP)

15// Agora que o computador de Bob possui o IP do google ele está pronto para enviar um socket TCP. Quando Bob cria um socket TCP, o seu notebook precisa primeiro executar um dos três vias com o TCP (página 208)

1. Notebook cria um segmento TCP SYN com portas 80 (HTTP) coloca o segmento TCP dentro de um datagrama IP com endereço IP de destino 64.233.169.105 (IP google) e coloca o datagrama em um quadro com endereço MAC de destino 00:22:6B:45:1F:1B (roteador de borda) envia para o comutador.

19// O quadro passa do roteador da escola → roteador Comcast → roteador do google, usando as tabelas de repasse de cada roteador

20// O datagrama contendo o TCP SYN com porto de destino 80 chega ao google.com. Um socket de conexão é criado para a conexão TCP entre servidor HTTP do google e o notebook de Bob. Um segmento TCP SYN ACK é gerado e colocado dentro de um datagrama e enviado ao PC de Bob.
21// O datagrama que contém o TCP SYN ACK é enviado através da rede do Google, Comcast e da escola, finalmente chegando ao cartão Ethernet de Bob e o socket TCP entra em estado de conexão.

22// Agora com o socket dentro do notebook de Bob pronto para enviar bytes, o navegador cria uma mensagem HTTP GET contendo a URL a ser procurada.

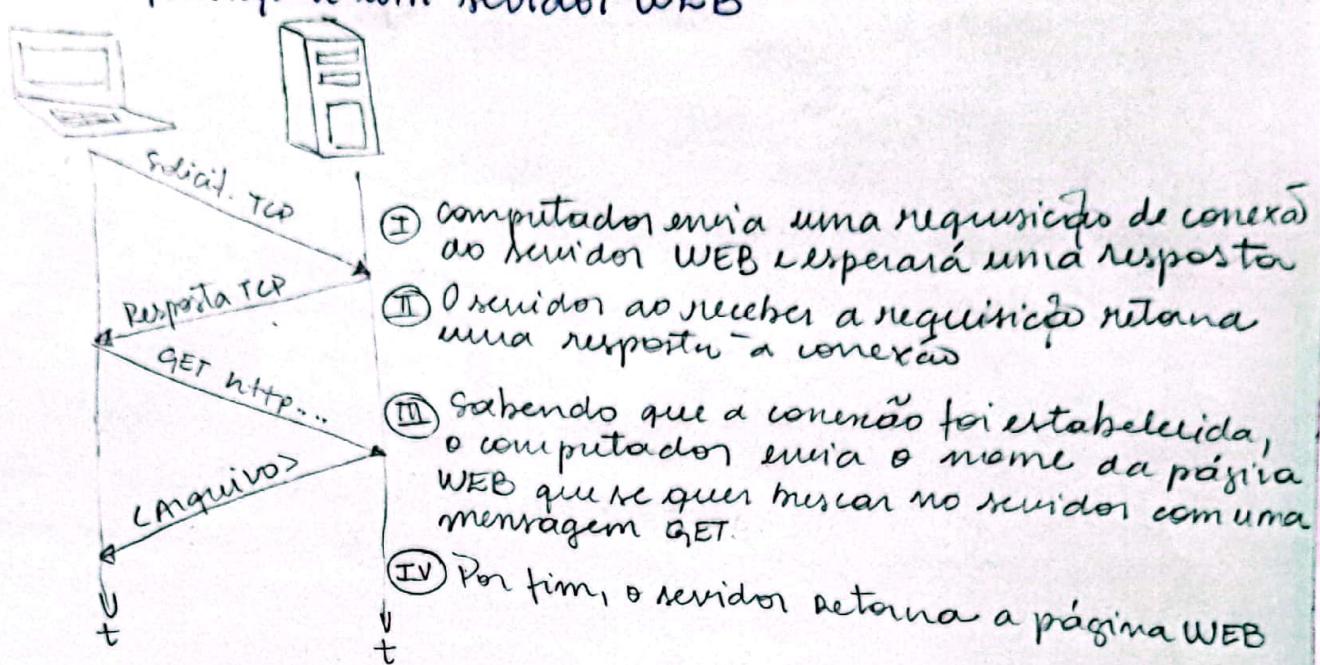
→ A mensagem GET é enviada ao socket retornando carga útil do segmento TCP.

23// O www.google.com lê a mensagem HTTP GET do socket TCP e cria uma resposta HTTP com o conteúdo da página Web requisitada no corpo da mensagem e envia pelo socket TCP

24// Finalmente, o datagrama contendo a resposta HTTP chega ao PC de Bob, o programa de navegação lê a resposta http e extrai o HTML do corpo da resposta

I Capítulo I:

① Protocolo: Define o formato e a ordem das mensagens trocadas entre duas ou mais entidades comunicantes, bem como as ações realizadas na transmissão e/ou recebimento de uma mensagem / evento
 RX: Requisição a um servidor WEB



② Comutação de pacotes:

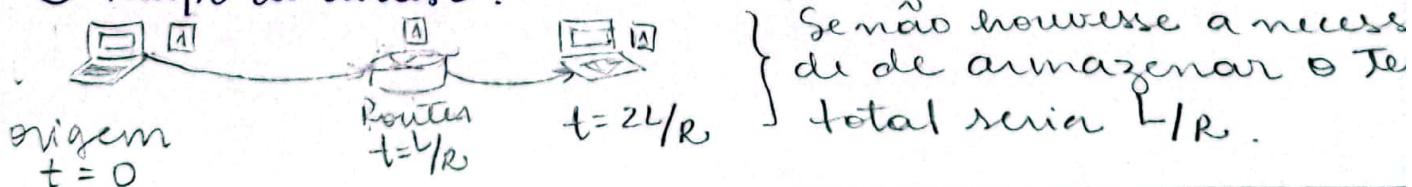
- comutação: Processo de interligar dois ou mais pontos entre si
- O sistema de origem fragmenta uma mensagem em pacotes e cada pacote percorre enlaces de comunicação e comutadores de pacotes (roteadores e comutadores da camada de en-

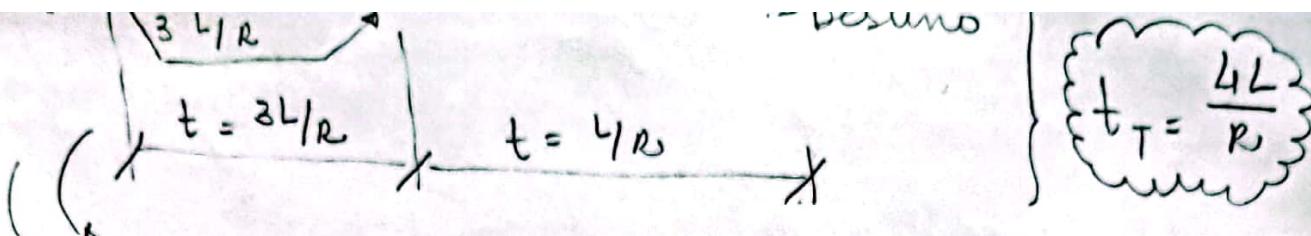
$$t_{transmissão} = \frac{L}{R} = \frac{\text{bits por pacote}}{\text{bits por segundo}}$$

A) Transmissão Armazena-e-reenvia (store and forward):

- O comutador espera receber todos os bits do pacote para poder começar a transmitir o pacote bit por bit isso porque os roteadores precisam armazenar e processar o pacote inteiro antes de transmitir

* tempo de atraso: $V = R \text{ bits/segundo}$

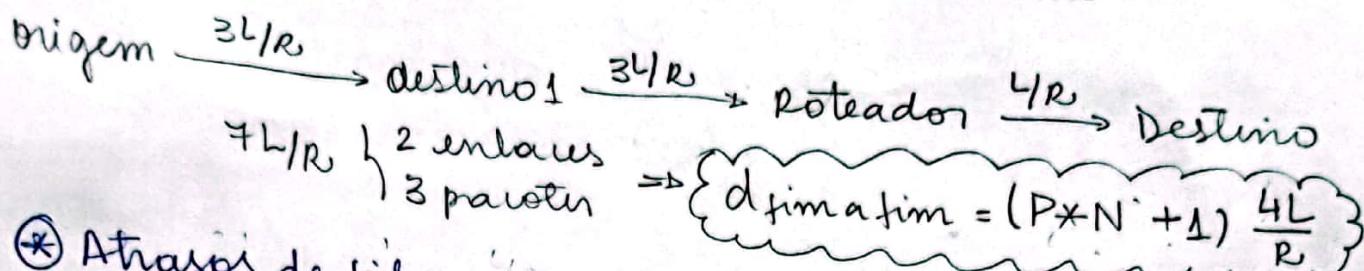




Demanda geral: $d_{\text{dimatim}} = N \frac{L}{R}$ ou

Ex: 2 enlaces, 3 paquetes

Enlaces



④ Atrasos de fila e perda de pacote:

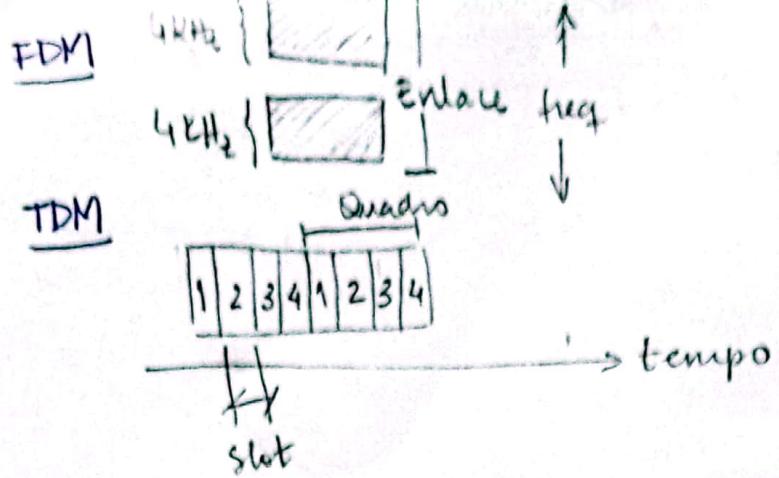
- Caso um pacote esteja chegando para ser transmitido mas encontra o enlace ocupado, este deve aguardar esteja lotado, o que levará a perda de pacote.

⑤ Tabelas de Repasse e protocolos de roteamento: (capítulo 4)

③ Comutação de circuitos: Um caminho dedicado é estabelecido entre o remetente e o destinatário antes da transmissão de dados (o caminho fica reservado exclusivamente para essa comunicação), também fica reservada a taxa de transmissão constantes nos enlaces, nodendo garantir a taxa constante de transmissão.

⑥ Multiplexação em redes de comutação por circuito:

- Um circuito é implementado em um enlace a partir da multiplexação por divisão de frequência (FDM) ou multiplexação por divisão de tempo (TDM)
- FDM: O espectro de frequência de um enlace é dividido entre as conexões do enlace
- TDM: O tempo é dividido em quadros de duração fixa e cada quadro é dividido em um número de compartimentos (slots)



* Comutação de pacotes X Comutação de circuitos:

Vantagens:

1// Eficiência:

O compartilhamento de recursos de rede permite uma utilização mais eficiente da largura de banda.

2// Tolerância a falhas:

Se um link/nó falhar existem caminhos alternativos.

3// Adequado para tráfego variável:

Funciona bem para tráfego de dados com diferentes tamanhos.

Desvantagens:

1// Atrasos variáveis:

Devido à natureza independente dos pacotes, pode haver atrasos na entrega.

2// Complexidade de gerenciamento

3// Overhead: os pacotes possuem overheads

Vantagens:

1// Baixa latência:

Possui baixa latência por conta do caminho dedicado.

2// Previsibilidade:

A largura de banda é garantida em toda a sessão, prevendo o desempenho.

3// Adequados para comunicações em tempo real:

Por possuir baixa latência

Desvantagens:

1// Ineficiência:

A largura de banda reservada não é usada eficientemente.

2// Menos tolerante a falhas:

Se o circuito dedicado falhar, a comunicação é interrompida.

3// Menos flexibilidade:

Não é adequado para tráfego variável ou intermitente.

④ Atrasos :

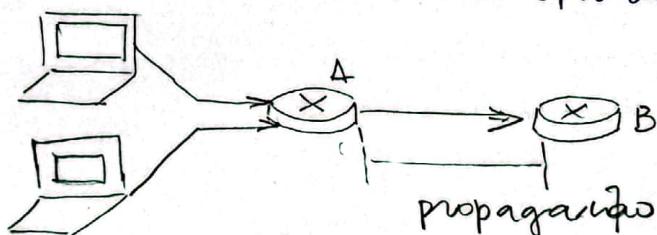
$$\text{Atrazo Nodal} = \text{Atrazo de processamento} + \text{Atrazo de fila} + \text{Atrazo de transmissão} + \text{Atrazo de propagação}$$

A) Atrazo de processamento: tempo exigido para decodificar o cabeçalho do pacote e determinar para onde direcioná-lo, além do tempo para verificar erros em bits do pacote

B) Atrazo de fila: O pacote sofre atraso de fila quando espera ser transmitido pelo enlace e depende da quantidade de outros pacotes que chegaram antes e que já estiveram na fila esperando pela transmissão pelo enlace.

C) Atrasos de Transmissão: atraso relacionado à natureza armazena e reenvia. (4/8)

D) Atrazo de propagação: é o tempo necessário para a propagar um bit desde início do enlace até o roteador B e depende do meio físico (fio de cobre trançado / fib. ótica)



⑤ Protocolos : - Protocolo da internet

- Camadas
 - * Aplicação
 - * Transporte
 - * Rede
 - * Enlace
 - * Físico
- Top-down.

I) Camada de aplicação:

definição: Camada de abstração que engloba protocolos que realizam comunicação tim-a-tim entre aplicações, é a camada mais próxima do usuário. (diferentes sistemas finais)

Existem 3 arquiteturas:

1. Cliente - Servidor

2. P2P

3. Híbrida

① Cliente - Servidor: O servidor é o hospedeiro sempre ativo que atende a requisição de muitos outros hospedeiros, são chamados de clientes, que podem apresentar conexões intermitente, IP dinâmicos e não se comunicam diretamente entre si. EX: Banco de Dados, servidores de e-mail

② Peer-to-Peer (P2P): O servidor não está sempre online, pois cada participante da rede funciona como um servidor e ajuda a manter o sistema funcionando

* Autoscalabilidade: novos pontos (peer) dão à estrutura maior capacidade de serviço assim como novas demandas de serviço.

EX: VoIP descentralizado, Bit Torrent

Cliente - Servidor X

- 1) Controle Centralizado
- 2) Escalabilidade Vertical
- 3) Desenvolvimento mais simples
- 4) Segurança Aprimorada

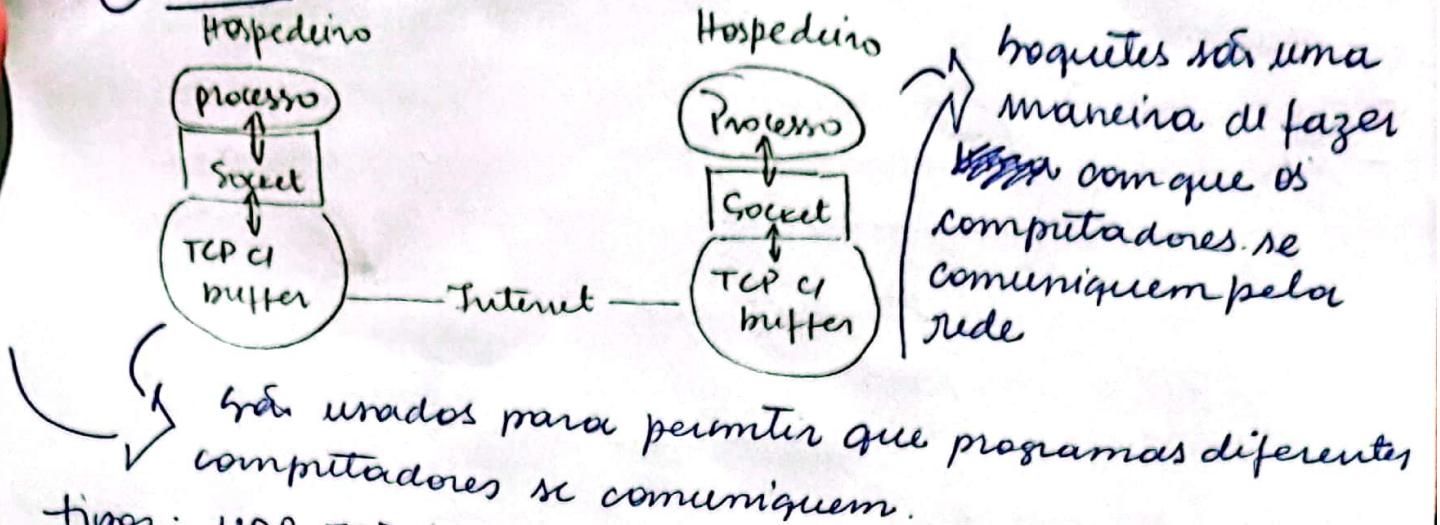
Desvantagem:

- 1) Ponto único de falha
- 2) Custos de infraestrutura
- 3) Latência

Peer-to-Peer

- 1) Descentralização
 - 2) Escalabilidade distribuída
 - 3) Eficiência de largura de banda (Recursos são compartilhados em pares)
 - 4) Independência Geográfica
-
- 1) Controle distribuído
 - 2) Variação de qualidade (variação de desempenho compartilhado)
 - 3) Questão legal:

③ Socket: Soquetes:



tipos: UDP, TCP (Serviços de transporte provisórios pela internet).

I) TCP (Transmission Control Protocol): / confiável: se esforça p/ garantir que os dados enviados sejam entregues corretamente

↳ Serviço Orientado para Conexão: / garantir que os dados enviados sejam entregues corretamente

- O TCP estabelece uma conexão entre cliente e servidor antes de iniciar a troca de dados.
- O serviço orientado para conexão garante que ambas as partes estejam clientes uma das outras e estejam montas para a troca de dados antes que os dados reais sejam enviados.

→ Handshake three-ways (Aperto de mão de três vias):
• usado para estabelecer uma conexão confiável entre dois dispositivos. Permite que haja a sincronização entre os dispositivos:

1^a Etapa: SYN (Synchronize) cliente envia um segmento TCP com flag $SYN = 1$ (ativada) para o servidor

2^a Etapa: SYN - ACK (Synchronize - acknowledge): O servidor recebe o segmento SYN, reconhece a solicitação de conexão e responde um segmento TCP com flags $SYN = 1$ e $ACK = 1$

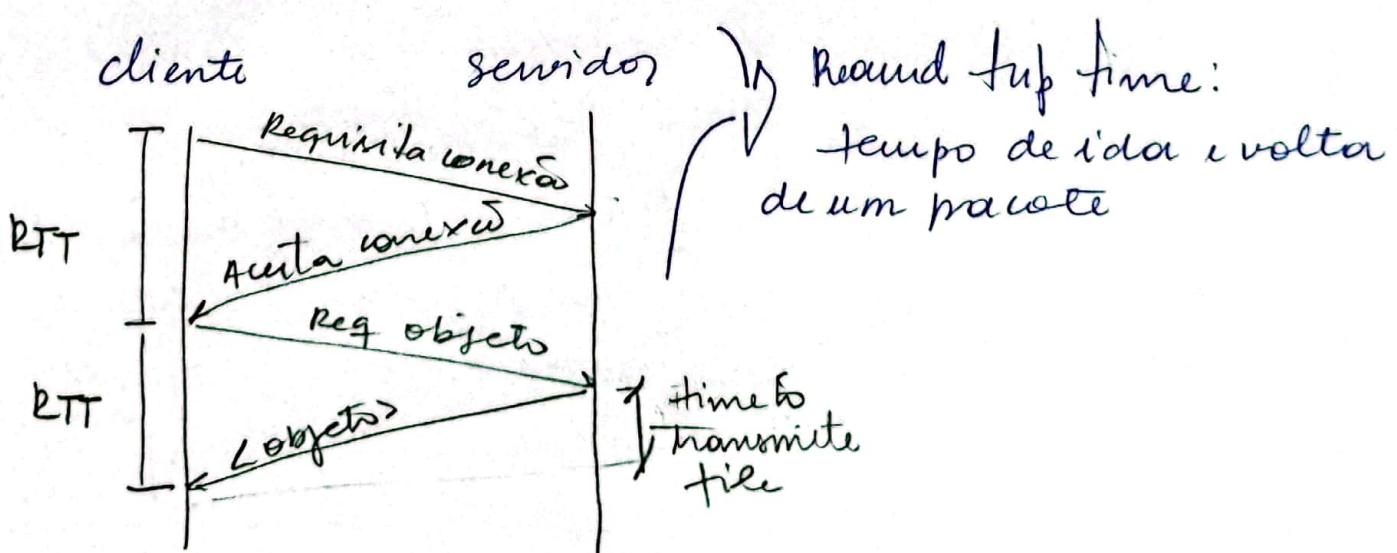
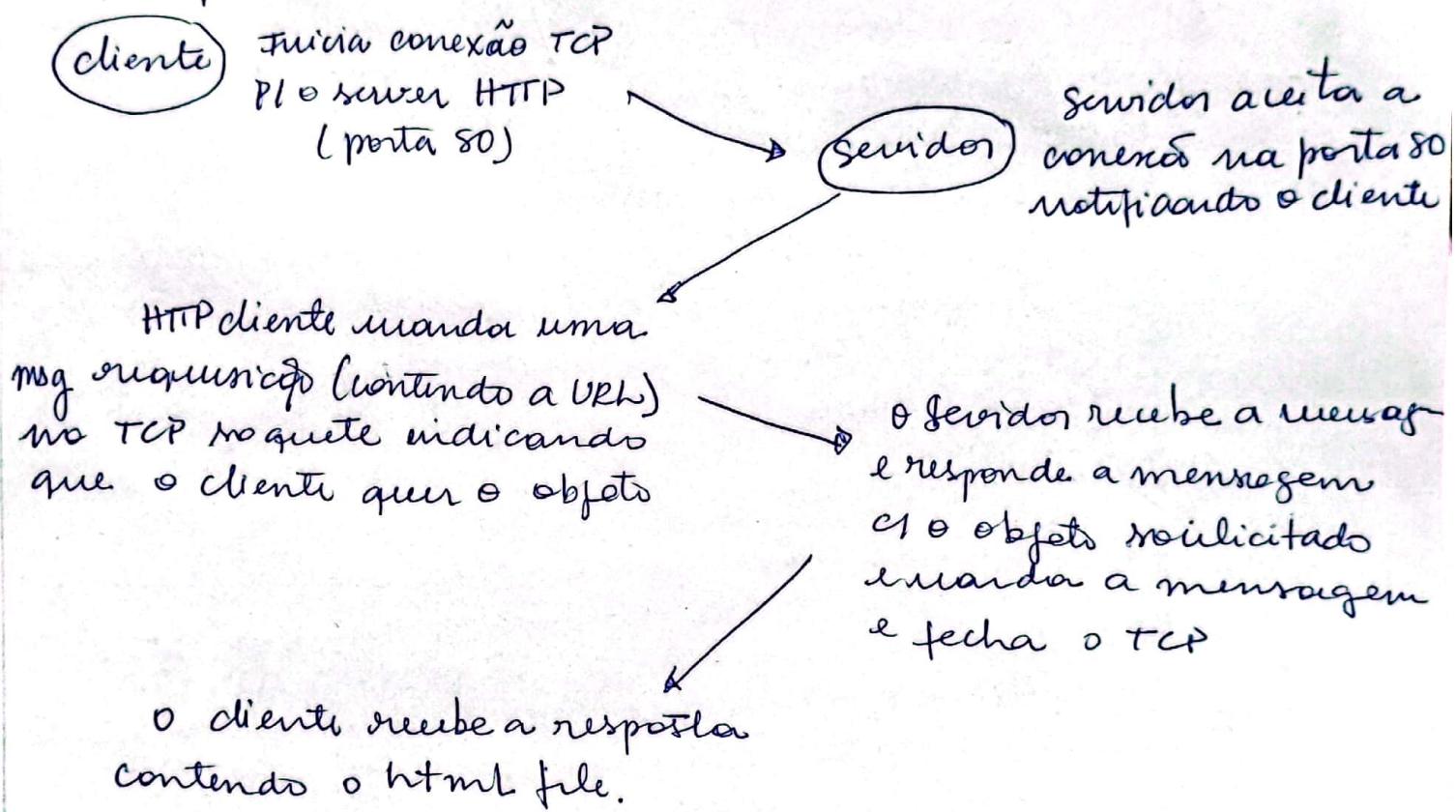
3^a Etapa: ACK (Acknowledge): O cliente recebe a resposta SYN-ACK do servidor e responde com um $ACK = 1$, confirmando que a conexão foi estabelecida.

O segmento ACK recebe o número de sequência do servidor acrescido de 1, indicando que o próximo byte de dados esperado do servidor é um byte do que o de fe-

II) UDP (user datagram protocol): não orientado à conexão

- não garante que os dados enviados chegarão ao destino
 - não é orientado para conexão: não requer handshake
 - transfere rapidamente: simplicidade e sem overhead
- ↳ Velocidade e simplicidade
perda ocasional.

④ HTTP: hypertext transfer protocol: protocolo de camadas de aplicação:



II) Camada de Transporte:

MAP

definição : A camada de Transporte é responsável ~~pe~~ pela comunicação entre os dispositivos em redes diferentes (TCP, UDP...), serviços de multiplexação, segmentação, controle de fluxo e controle de erros.

Cliente: quebra a aplicação em segmentos

Receptor: Monta os segmentos em uma montagem

⇒ Demultiplexing: Processo de direcionar os pacotes de dados para os aplicativos corretos em um dispositivo de base nas intenções contidas no cabeçalho dos pacotes

⇒ Encamada de Rede: Roteamento de pacotes para seus destinos corretos em base nos endereços IP de origem e destino contidos no cabeçalho

TCP (RTT):

↳ como saber o tempo }
 too long: reacq lenta p/ a
 perda do segmento
 too short: premature timeout
 retransmissões desnecessárias

⇒ Sample RTT:

- Estimativa do tempo da transmissão de segmentos até o ack. aapt
- Ignora ~~retransmissões~~ retransmissões

TCP Slow Start : Exponencial: 1, 2, 4...

III) Camada de Rede:

definição: lida como o roteamento e encaminhamento de dados ~~de~~.

- Envelopa segmentos em datagrama

- Forwarding (Encaminhamento): Envia pacotes de dados para seu destino imediato
- Roteamento: ~~encaminhamento~~ Enviar através de seleção de rotas melhores.

IP:	A	8 bits rede : 1.000 até 127.255.255.255
Classes	B	16 " : 128.000.000 até 191.255.255.255
	C	24 " : 192.0.0.0 - 223.255.255.255
	D	32 " 224.0.0.0 até 239.255.255.255

SDN:

=> NAT: network address translation
 CIDR: classless interdomain routing (cidr)

200.23.16.0/23:

0 11001000 00010111 00010000 0 00000000

 23 (Subnet) _____ 9 (host part)

Ex: 200.129.68.0/27 =>

rede	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰
0.129.68.	0.0.0.0	0.0.0.0	0.0.0.0	0.0.0.0	0.0.0.0

 1ª Rede: 200.129.68.32 Host

1ª Máquina: 200.129.68.33

Última máquina: 200.129.68.62 (32+16+8+4+2)

200.129.68.62	128	64	32	32
	2 ⁷	2 ⁶	2 ⁵	2 ⁴

Broadcast: 200.129.68.63

L. todos os 1 no host.

2ª Subrede: 200.129.68.64

128	64	32	32
160	192	224	224

$$\frac{192}{8} \quad \frac{168}{8} \quad \frac{8}{8} \quad \frac{8}{8}$$

4 rede 4 host

$$\underbrace{2^7 2^6 2^5 2^4}_{\text{4 rede}} \underbrace{2^3 2^2 2^1 2^0}_{\text{4 host}} + \underbrace{2^4}_{15} \rightarrow 2^5 \quad 255$$

1ª Rede: 192.168.0.0

1ª Máquina: 192.168.0.1

Última máquina: 192.168.15.254

Broadcast: 192.168.15.255

~~2ª e 3ª máquinas 192.168.0.0~~

2ª Rede: 192.168.32.0

1ª Máquina: 192.168.32.1

Última: 192.168.47.254

Broadcast: 192.168.47.255

$$2^4 = 128 \quad 2^5 = 32 \quad 2^3 = 8 \quad 2^1 = 2 \\ 2^6 = 64 \quad 2^4 = 16 \quad 2^2 = 4 \quad 2^0 = 1$$

$$T = 255$$



II) Camada de enlace: Gerencia a transferência confiável de dados entre dispositivos encapsula dados em quadros, adiciona MAC.

- Detecção de erro:
 - Ruidos
 - Detector erros, ruido ou perda de quadros