

# Aula 13

# Conteúdo do Capítulo 3

□ UDP: Transporte não orientado a conexão

□ Transporte orientado a conexão: TCP

- transferência confiável
- controle de fluxo
- gerenciamento de conexões

# UDP: User Datagram Protocol [RFC 768]

- ❑ Protocolo de transporte da Internet mínimo, "sem frescura",
- ❑ Serviço "melhor esforço", segmentos UDP podem ser:
  - perdidos
  - entregues à aplicação fora de ordem do remesso
- ❑ *sem conexão:*
  - não há "apresentação" UDP entre remetente, receptor
  - tratamento independente de cada segmento UDP
- ❑ Exemplo: DNS - protocolo da camada de aplicação que usa o UDP.

## Por quê existe um UDP?

- ❑ elimina estabelecimento de conexão (o que pode causar retardo)
- ❑ simples: não se mantém "estado" da conexão no remetente/receptor
- ❑ pequeno cabeçalho de segmento
- ❑ sem controle de congestionamento: UDP pode transmitir o mais rápido possível

# Aplicações populares da Internet e seus protocolos de transporte

aplicação	Protocolo da camada de aplicação	Protocolo de transporte
Correio eletrônico	SMTP	TCP
Acesso a terminal remoto	Telnet	TCP
Web	HTTP	TCP
Transferência de arquivo	FTP	TCP
Servidor de arquivo remoto	NFS	Tipicamente UDP
Recepção de multimídia	proprietário	UDP ou TCP
Telefonia por Internet	proprietário	UDP ou TCP
Gerenciamento de Rede	SNMP	Tipicamente UDP
Protocolo de roteamento	RIP	Tipicamente UDP
Tradução de nome	DNS	Tipicamente UDP

# UDP: User Datagram Protocol

- ❑ Muitas aplicações importantes executam sobre o UDP
- ❑ O UDP é usado para atualização das tabelas de roteamento com o protocolo RIP
- ❑ O UDP também é usado para levar dados de gerenciamento de rede - são aplicações usadas quando a rede está sobrecarregada.

# Mais sobre UDP

- ❑ muito utilizado para apls. de meios contínuos (voz, vídeo)
  - tolerantes de perdas
  - sensíveis à taxa de transmissão
- ❑ outros usos de UDP (por quê?):
  - DNS (nomes)
  - DHCP (gerenciamento dos endereços IP)
- ❑ transferência confiável com UDP: incluir confiabilidade na camada de aplicação
  - recuperação de erro específica à apl.!

# UDP e TCP

❑ O UDP e o TCP também são comumente usados para aplicações de multimídia, como telefone por internet, videoconferência em tempo real e áudio e vídeo armazenados.

❑ O TCP está sendo utilizado cada vez mais para transporte de mídia.

# Encapsulamento segmento UDP

**Camada de aplicação**

Mensagem da Aplicação

**Camada de transporte**

Cabeçalho  
UDP

Area de dados do segmento  
UDP

**Camada  
de rede**

Cabeçalho  
IP

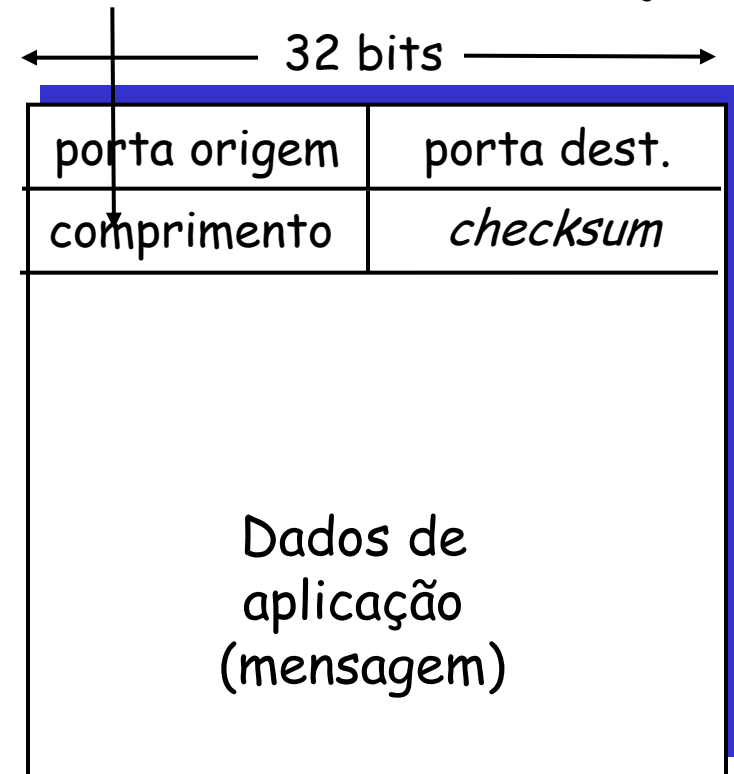
Area de dados do datagrama IP



# Estrutura do segmento UDP

Comprimento em bytes do segmento UDP, incluindo cabeçalho

- ❑ Os dados da aplicação ocupam o campo de dados do segmento UDP
- ❑ O cabeçalho tem apenas 4 campos, cada um consistindo de 2 bytes
- ❑ O campo comprimento especifica o número de bytes (cabeçalho + dados)
- ❑ O checksum - soma de verificação - é usada pelo hospedeiro receptor para verificar se foram introduzidos erros no segmento.



Formato do segmento UDP

# exercício

1. Qual o serviço oferecido pelo protocolo UDP as aplicações de rede?
2. Qual a vantagem do protocolo UDP?
3. Dê um exemplo de um protocolo da camada de aplicação que utiliza o UDP na camada de transporte?
4. Desenhe a estrutura do segmento UDP.

# Conteúdo do Capítulo 3

❑ UDP: Transporte não orientado a conexão

❑ Transporte orientado a conexão: TCP

- transferência confiável
- controle de fluxo
- gerenciamento de conexões

# TCP: Visão geral RFCs: 793, 1122, 1323, 2018, 2581

- ❑ **orientado a conexão:**
  - handshaking (troca de msgs de controle) inicia estado de remetente, receptor antes de trocar dados
- ❑ **Provê serviço full duplex:**
  - fluxo de dados bi-direcional na mesma conexão
- ❑ **ponto a ponto:**
  - 1 remetente, 1 receptor

# TCP: Visão geral

- ❑ Suponha que um processo que roda em um host queira iniciar a conexão com outro processo em outro host.
- ❑ O processo que inicia a conexão é o processo cliente, o outro é denominado processo servidor.
- ❑ O processo de aplicação cliente primeiramente informa à camada de transporte no cliente que ele quer estabelecer a conexão com um processo servidor.
- ❑ A camada de transporte no cliente então passa a estabelecer uma conexão TCP-servidor.
- ❑ O cliente envia um segmento TCP especial;
- ❑ O servidor responde com um segmento TCP especial;
- ❑ E por fim, o cliente responde novamente com um terceiro segmento especial.
- ❑ Esse procedimento é denominado apresentação de três vias (3-way handshake)

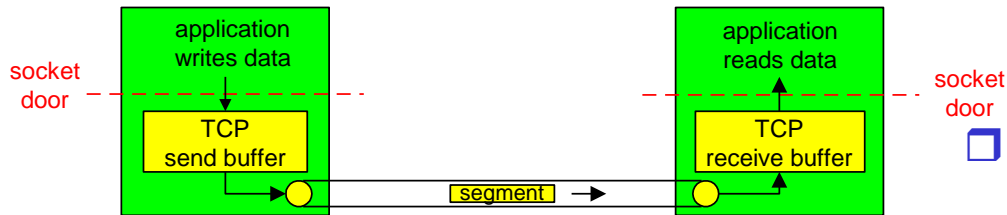
# TCP: Visão geral

- ❑ Os dois primeiros segmentos não contém nenhuma carga útil - ou seja, nenhum dado da camada de aplicação!
- ❑ O terceiro desses segmentos pode carregar uma carga útil.
- ❑ Uma vez estabelecida uma conexão TCP, os dois processos de aplicação podem enviar dados um para o outro.
- ❑ Vamos considerar o envio de dados do cliente para o servidor!
- ❑ O processo cliente passa uma cadeia de dados através do socket.
- ❑ Tão logo passem pelo socket, os dados estão nas mãos do TCP que está rodando no cliente.
- ❑ O TCP direciona seus dados para o **buffer de envio** da conexão, que é reservado durante a apresentação de três vias inicial.

# TCP: Visão geral

- ❑ A quantidade máxima de dados que pode ser retirada e colocada em um segmento é limitada pelo **tamanho máximo do segmento (MSS)**.
- ❑ O MSS é a quantidade máxima de dados de camada de aplicação no segmento.
- ❑ O TCP combina cada porção da dados do cliente com um cabeçalho TCP, formando assim **segmentos TCP**.
- ❑ Os segmentos são passados para baixo, para a camada de rede, onde são encapsulados separadamente dentro dos datagramas IP de camada de rede.
- ❑ Os datagramas IP são então enviados para dentro da rede.
- ❑ Quando o TCP recebe um segmento na outra extremidade, os dados do segmentos são colocados no buffer de recepção da conexão.

# TCP: Visão geral



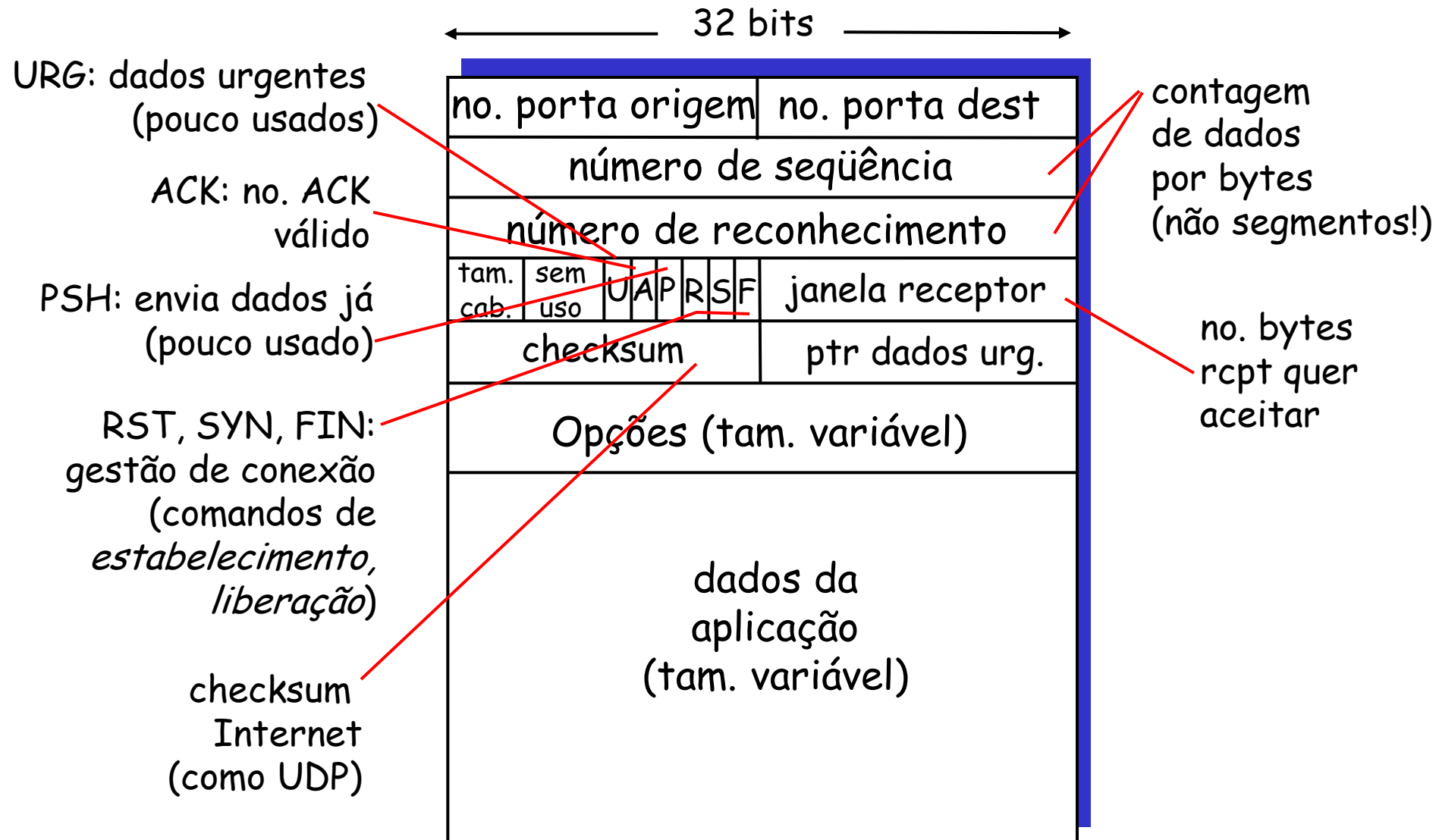
- conforme a figura, a aplicação lê a cadeia de dados desse buffer.
- Cada lado da conexão tem seus próprios buffers de envio e seu próprio buffer de recepção.

- Uma conexão TCP consiste em buffers, variáveis e um socket de conexão de um processo em um hospedeiro e outro conjunto de buffers, variáveis e um socket de conexão de um processo em outro hospedeiro.
- *Nenhum buffer nem variáveis são alocados à conexão nos elementos da rede existentes entre os hospedeiros (roteadores, comutadores e repetidores).*



# TCP: estrutura do segmento

Pág 172



# TCP: nos. de seq. e ACKs

## Número de sequencia:

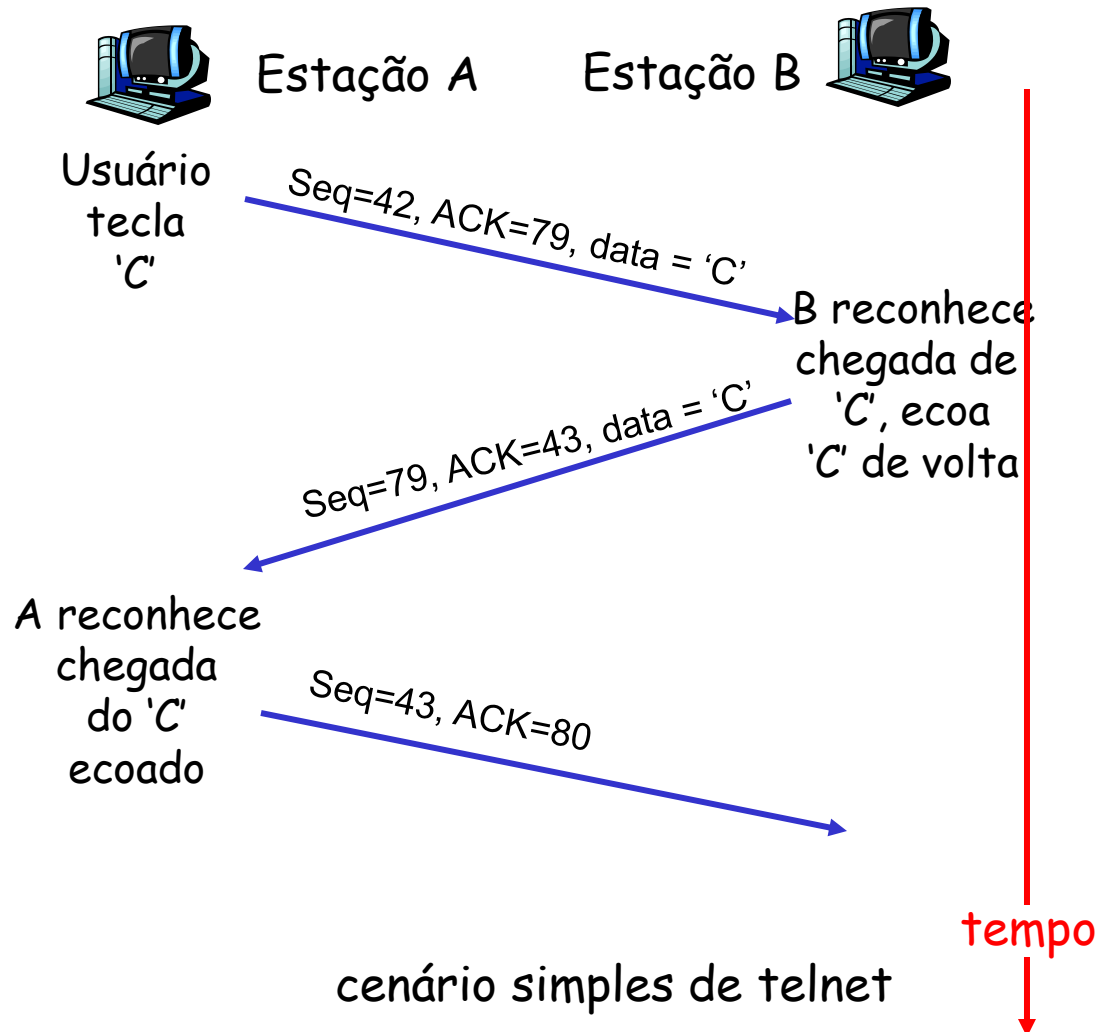
- "número" dentro do fluxo de bytes do primeiro byte de dados do segmento

## ACKs(reconhecimentos):

- no. de seq do próx. byte esperado do outro lado
- ACK cumulativo

**P:** como receptor trata segmentos fora da ordem?

- Não impõem nenhuma regra e deixa a decisão ao implementador do TCP.

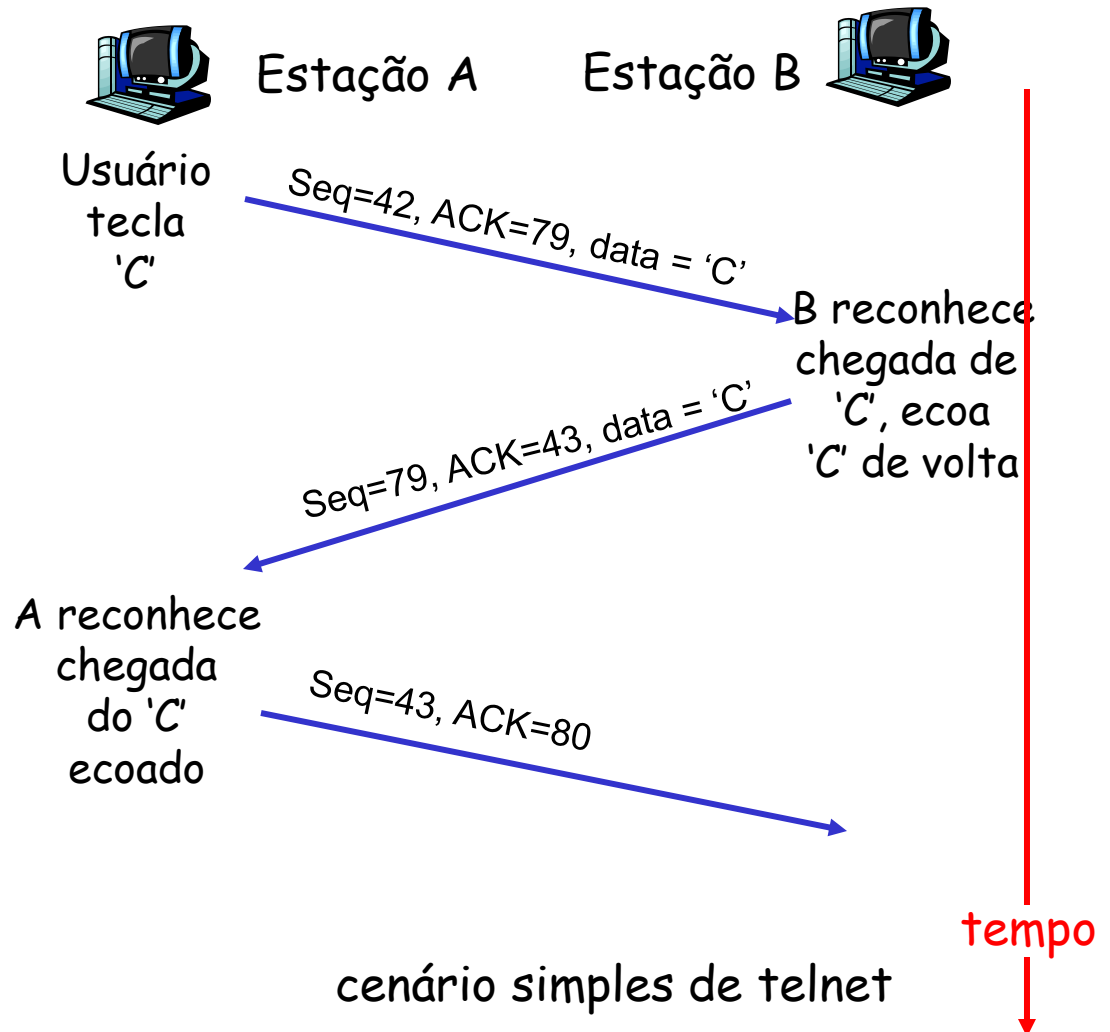


# TCP: nos. de seq. e ACKs

São enviados três segmentos;

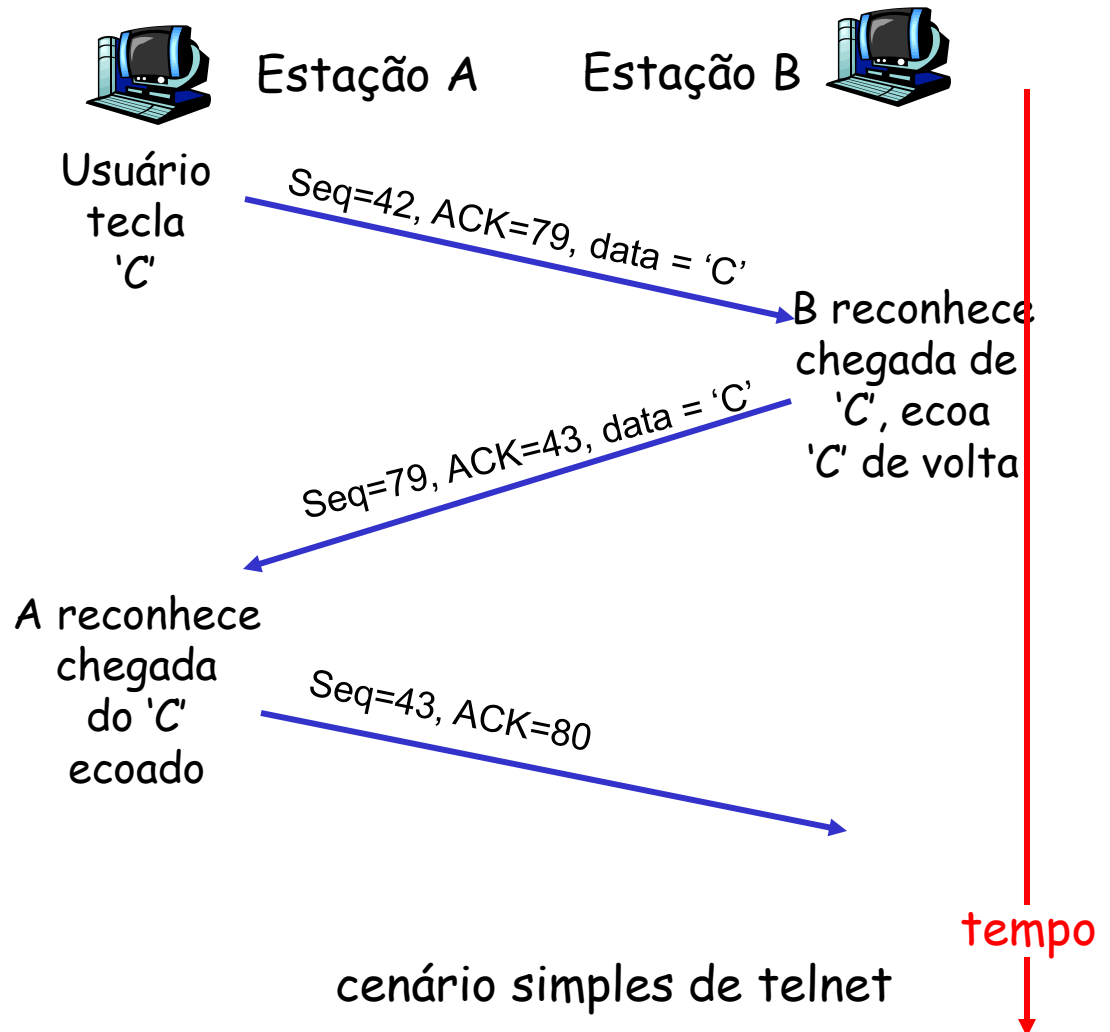
O primeiro é enviado do cliente ao servidor, contendo em seu campo de dados "C". Tem 42 em seu campo de número de sequência e 79 em seu campo de número de reconhecimento;

O segundo segmento é enviado do servidor ao cliente. Esse segmento tem dupla finalidade - reconhecimento dos dados recebidos pelo servidor; e ecoar a letra "C".



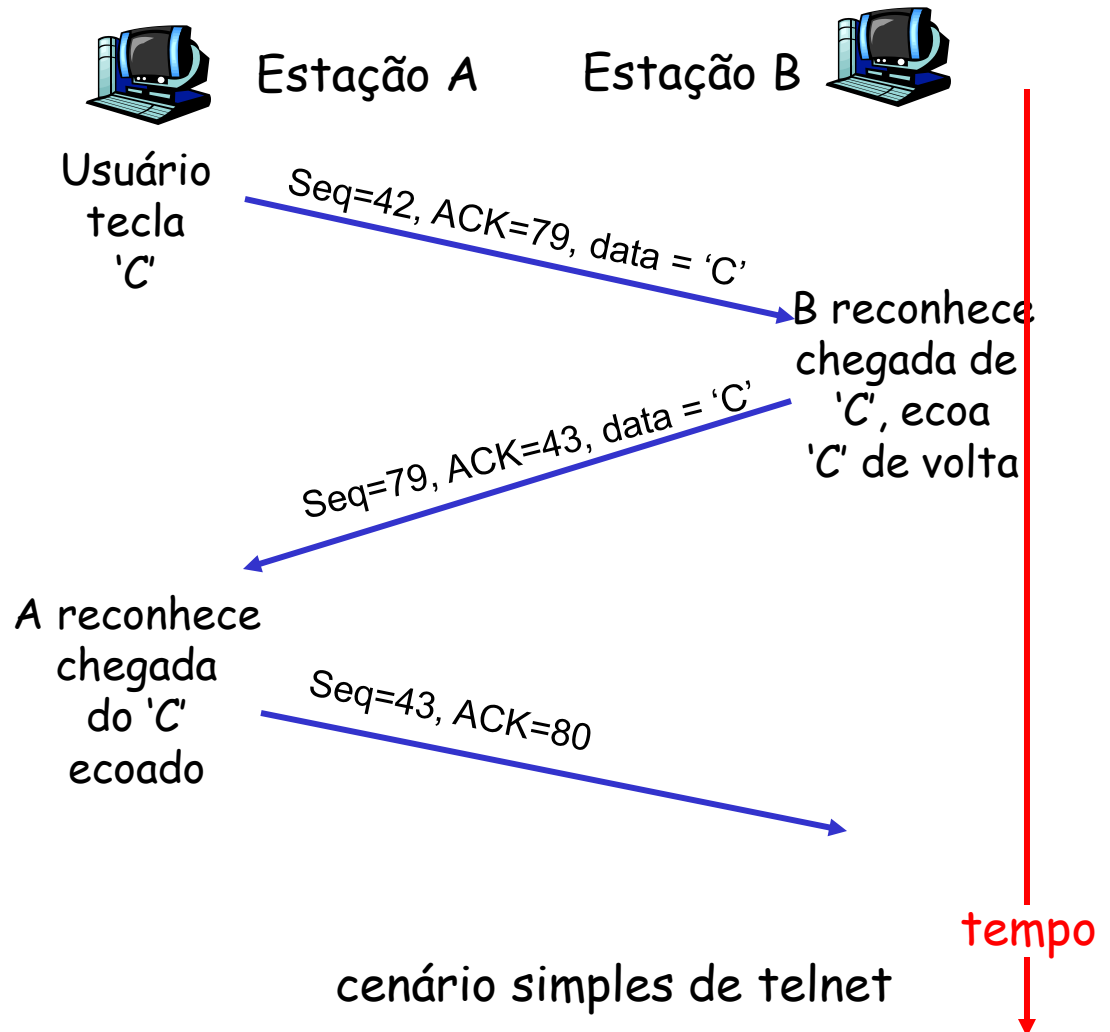
# TCP: nos. de seq. e ACKs

Ao colocar ACK=43 o servidor esta dizendo que recebeu com sucesso tudo até o byte 42 e agora está aguardando os bytes de 43 em diante; o segundo segmento tem "C" em seu campo de dados. Ele tem o número de sequencia 79 que o número inicial do fluxo de dados do servidor para cliente dessa conexão TCP, pois este é o primeiro byte de dados que o servidor está enviando.



# TCP: nos. de seq. e ACKs

O terceiro segmento é enviado do cliente ao servidor. Seu propósito é reconhecer os dados que recebeu do servidor. Esse segmento tem um campo de dados vazio. Tem o número 80 no campo de reconhecimento porque o cliente recebeu a cadeia de dados até o byte com número de sequência 79 e esta aguardando os bytes de 80 em diante.



# Transferência de dados confiável do TCP

- ❑ O TCP cria um serviço de transferência confiável de dados sobre o serviço não confiável do IP
- ❑ O IP (camada de rede) é um serviço não confiável.
  - Não garante a entrega de datagramas na ordem correta, nem a integridade desses dados;



- ❑ Os datagramas podem transbordar dos buffers dos roteadores e jamais alcançar seu destino;
- ❑ Os datagramas também podem chegar fora de ordem;
- ❑ Os bits dos datagramas podem ser corrompidos (passar de 0 para 1 e vice-versa);
- ❑ Logo, os segmentos da camada de transporte são carregados pela rede por datagramas IPs, podem sofrer os mesmos problemas anteriores.

# Exercício - 10

1. Por que o TCP é orientado para conexão?
2. A conexão TCP é sempre **ponto a ponto**, o que isso significa?
3. Como a conexão TCP é estabelecida?
4. O que é o tamanho máximo do segmento- MSS?
5. O que é a unidade máxima de transmissão - MTU?
6. O que é o número de sequência para um segmento?
7. Cite 5(cinco) aplicações populares da Internet seus protocolos de camada de aplicação e de transporte ?
8. **Assista o vídeo no youtube - Protocolos TCP e UDP do professor Paulo Kretcheu.**

# Capítulo 3: Resumo

- ❑ Princípios atrás dos serviços da camada de transporte:
  - multiplexação/demultiplexação
  - transferência confiável de dados
- ❑ Protocolos de Transporte na Internet
  - UDP
  - TCP

## Próximo capítulo:

- ❑ saímos da “borda” da rede (camadas de aplicação e transporte)
- ❑ entramos no “núcleo” da rede