

TEMA 07 – Protocolo TCP e UDP

Habilidades

- Identificar as principais diferenças entre TCP e UDP
 - Configurar e gerenciar *sockets* TCP e UDP em aplicações
 - Compreender o modelo de três vias (*Three-Way Handshake*) do TCP
 - Diagnosticar problemas de rede relacionados ao TCP e UDP
 - Avaliar o desempenho de aplicativos usando TCP ou UDP
 - Implementar medidas de segurança para proteger a comunicação TCP e UDP
-

Introdução

Assim como duas pessoas precisam falar a mesma língua para se entender, **computadores também precisam “falar” o mesmo protocolo** para se comunicarem em rede.

Entre os diversos protocolos existentes, **dois se destacam por sua importância e uso:**

 **TCP (Transmission Control Protocol)** e

 **UDP (User Datagram Protocol).**

Ambos atuam na **camada de transporte** do modelo TCP/IP, garantindo (em maior ou menor grau) que as mensagens cheguem aos seus destinos.

1 Protocolo TCP (Transmission Control Protocol)

O TCP oferece uma **transmissão confiável e orientada à conexão** — ou seja, garante que os dados cheguem **na ordem correta, sem perdas e sem duplicação**.

Características principais

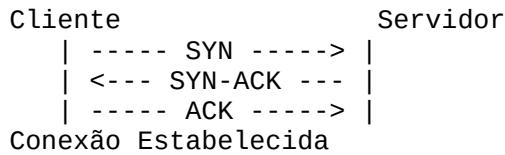
- Comunicação **ponto a ponto** (cliente-servidor).
 - Usa **ACK (bit de reconhecimento)** para confirmar pacotes recebidos.
 - **Reordena pacotes e retransmite** os que se perdem.
 - Controla o **fluxo** e evita congestionamentos.
 - Utiliza **checksum** para verificar integridade dos dados.
-

🔗 Etapas da Conexão TCP – Three-Way Handshake

A conexão TCP é estabelecida em **três passos** (“aperto de mãos”):

Etapa	Ação	Descrição
1	SYN	O cliente solicita a conexão.
2	SYN + ACK	O servidor aceita o pedido e responde.
3	ACK	O cliente confirma a resposta e a conexão é criada.

📊 Representação Simplificada:



⬅ END Finalização da Conexão – Four-Way Handshake

O encerramento também é controlado, garantindo que ambos os lados saibam que a comunicação terminou.

Etapa	Ação	Descrição
1	FIN (Cliente)	Cliente deseja encerrar a conexão.
2	ACK (Servidor)	Servidor reconhece o pedido.
3	FIN (Servidor)	Servidor também encerra.
4	ACK (Cliente)	Cliente confirma o fim.

📊 Representação Simplificada:



2 Protocolo UDP (User Datagram Protocol)

O UDP é um protocolo **sem conexão (não confiável)** — ele apenas **envia os dados** para o destino, sem confirmar se chegaram ou não.

⚙️ Características principais

- Envia **datagramas** sem controle de entrega.
- **Não há verificação de erros nem confirmação (ACK).**
- **Mais rápido e leve** que o TCP.
- Ideal para **aplicações em tempo real**.

Aplicações comuns

- Streaming de áudio e vídeo
- Chamadas de voz (VoIP)
- Jogos online
- Transmissões em tempo real

 Se um pacote se perder, o usuário dificilmente percebe — e o fluxo continua.

Comparação: TCP vs UDP

Característica	TCP	UDP
Tipo de conexão	Orientado à conexão	Sem conexão
Confiabilidade	Alta – confirma entrega	Baixa – sem confirmação
Ordem dos pacotes	Garantida	Pode chegar fora de ordem
Velocidade	Menor (controle rigoroso)	Maior (sem verificação)
Controle de fluxo	Sim	Não
Retransmissão	Automática	Não há
Aplicações típicas	Web, e-mail, FTP	Streaming, jogos, VoIP

Portas TCP e UDP

Cada serviço de rede utiliza uma **porta** para identificar o tipo de comunicação. As portas são números de **0 a 65535** (16 bits), definidas pela **IANA**.

Faixa	Tipo	Descrição
0 – 1023	Bem conhecidas	Usadas por serviços padrão (HTTP, FTP etc.)
1024 – 49151	Registradas	Para aplicativos específicos
49152 – 65535	Dinâmicas/privadas	Uso temporário ou particular

Principais portas e protocolos

Porta	Serviço	Protocolo
20–21	FTP	TCP
22	SSH	TCP
23	Telnet	TCP
25	SMTP	TCP
53	DNS	TCP/UDP
67–68	DHCP	UDP
80	HTTP	TCP
110	POP3	TCP
123	NTP	UDP
143	IMAP	TCP
161	SNMP	UDP

Porta Serviço Protocolo

443	HTTPS	TCP
3389	RDP	TCP/UDP
5060	SIP	TCP/UDP

4 Diagnóstico de Problemas em TCP e UDP

Problemas comuns:

- **Alta latência** (demora na resposta)
- **Perda de pacotes**
- **Congestionamento**
- **Configuração incorreta de portas**

Ferramentas úteis:

- `ping` → testa conectividade
- `traceroute` → identifica o caminho e gargalos
- **Analisadores de pacotes** (Wireshark, tcpdump)

 *Lembre-se: TCP tenta se recuperar de falhas automaticamente. UDP não.*

5 Desempenho e Latência

- **TCP** → mais seguro, mas tem **overhead** (retransmissões e verificações).
- **UDP** → mais rápido, ideal para **baixa latência**.

 **Latência** = tempo entre o envio de uma solicitação e a resposta.

Quanto menor, mais “tempo real” a experiência (como em chamadas e jogos).

6 Segurança em TCP e UDP

Medidas recomendadas

- **Criptografia:** usar TLS/SSL
- **Autenticação:** senhas, tokens, certificados
- **Firewalls:** controlar tráfego por porta e protocolo
- **VPNs:** proteger dados em redes públicas
- **Controle de acesso:** restringir conexões indevidas
- **Monitoramento:** detectar ameaças em tempo real

Resumo Rápido

Aspecto	TCP	UDP
Confiável	<input checked="" type="checkbox"/> Sim	<input checked="" type="checkbox"/> Não
Verifica erros	<input checked="" type="checkbox"/> Sim	<input type="triangle-down"/> Parcial
Ordem garantida	<input checked="" type="checkbox"/> Sim	<input checked="" type="checkbox"/> Não
Uso principal	Web, e-mail, transferência de arquivos	Streaming, jogos, chamadas
Velocidade	 Menor	 Maior
Controle de fluxo	<input checked="" type="checkbox"/> Sim	<input checked="" type="checkbox"/> Não

Dica de Estudo

- ◊ Quando a prioridade é **confiabilidade**, use **TCP**.
- ◊ Quando a prioridade é **velocidade e tempo real**, use **UDP**.
- ◊ Em aplicações modernas (ex.: *streaming seguro*), **ambos podem coexistir** com criptografia TLS.