

1.O modelo OSI

O modelo de referência OSI (*Open System Interconnections*) fornece uma extensa lista de funções e serviços que podem ocorrer em camada. Esse tipo de modelo fornece consistência em todos os tipos de protocolos e serviços de rede, descrevendo o que deve ser feito em uma camada específica, mas não prescrevendo como deve ser realizado. Ele também descreve a interação de cada camada com as camadas diretamente acima e abaixo.

Enquanto as camadas do modelo TCP/IP são referidos pelo nome, as sete camadas do modelo OSI são mais frequentemente referidas pelo número do que o nome.

2.Camada física/camada de transmissão/primeira camada

Os protocolos aqui descrevem o mecânico, elétrico, meios funcionais e processuais para ativar, manter e desativar conexões físicas para uma transmissão de bits de, e para uma rede ou dispositivo. A camada responsável pela transmissão de bits entre computadores lida com as características físicas e elétricas da rede. Algumas de suas principais características incluem topologia física, bits de sincronização, estratégias de multiplexação, entre outras.

No que se refere ao estado de corrente, a representação binária ocorre da seguinte forma: 0V equivale a 0 (binário) e $\pm 5V$ equivale a 1 (binário). Já no estado de transição, se houver mudança durante um ciclo de clock, o valor representado será 1; caso contrário, será 0.

A camada 1 enxerga a rede sob uma perspectiva física, analisando como os cabos estão conectados. No caso da transmissão de bits, os métodos podem ser assíncronos ou síncronos. Bits assíncronos utilizam bits de início e parada para indicar quando as transmissões devem ocorrer entre receptor e transmissor, enquanto bits síncronos utilizam um sinal de clock para coordenar as transmissões entre ambas as partes.

A transmissão de dados pode ser feita de duas formas principais: **broadband** e **baseband**. O método broadband divide a largura de banda em canais separados, como ocorre na TV. Já o método baseband utiliza todos os canais disponíveis para transmitir os dados em uma determinada frequência, como na Ethernet.

Existem diferentes estratégias de multiplexação para compartilhamento de recursos:

- **TDM (Time-Division Multiplexing)**: cada sessão utiliza um intervalo de tempo específico para transmissão, permitindo o compartilhamento ordenado entre todos os usuários.

- **StatTDM (Statistical Time-Division Multiplexing)**: uma versão mais eficiente do TDM, que aloca dinamicamente os intervalos de tempo de acordo com a necessidade de cada usuário, em vez de fazer uma alocação fixa.
- **FDM (Frequency-Division Multiplexing)**: divide o espectro em diferentes canais com base em frequências, permitindo que cada sessão seja transmitida em um canal distinto, como no broadband.

Os meios físicos de transmissão incluem cabos Ethernet e fibra óptica. Já as transmissões por meio de frequências de rádio englobam tecnologias como Wi-Fi e Bluetooth.

Por fim, os dispositivos de infraestrutura fundamentais nessa camada incluem hubs, pontos de acesso sem fio (wireless access points) e conversores de dispositivos.

3.Camada de enlace/segunda camada

A camada 2 é responsável pelo empacotamento de dados em frames, pela transmissão desses frames para a rede, pela detecção e correção de erros, pela identificação de dispositivos de rede por meio do endereço MAC e pelo controle de fluxo de dados.

O **MAC (Media Access Control)** é um identificador físico dos dispositivos de rede e um meio para a transmissão de dados entre eles. Cada endereço MAC possui 48 bits, sendo que os primeiros 24 bits correspondem ao identificador do fabricante e os outros 24 bits são exclusivos para cada dispositivo. Todo dispositivo eletrônico conectado a uma rede possui um endereço MAC.

Diferentemente da camada 1, que enxerga a rede de forma física, a **camada 2** vê a rede de maneira lógica. Dentro dessa camada, o **LLC (Logical Link Control)** é responsável por confirmar o recebimento das mensagens, ajudar no controle de fluxo (limitando a quantidade de dados que um transmissor pode enviar) e gerenciar erros, permitindo que o receptor informe problemas no envio de pacotes de dados.

Os métodos de sincronização utilizados nesta camada incluem:

- **Isochronous**: utiliza um clock referencial para a transmissão, reduzindo a sobrecarga.
- **Asynchronous**: cada dispositivo utiliza seu próprio clock interno para iniciar e parar transmissões.
- **Synchronous**: ambos os dispositivos sincronizam seus clocks para transmissão e recepção de dados.

Os dispositivos operantes na camada 2 incluem **switches**, **bridges** e **NICs (Network Interface Cards)**. Alguns switches possuem a capacidade de operar em diferentes

camadas da rede, sendo chamados de **multi-layer switches**. Entretanto, essa característica só deve ser considerada quando explicitamente necessário.

4. Camada de rede/terceira camada

A camada 3 é responsável pelo encaminhamento de pacotes através do endereço IP, que funciona como um identificador lógico dos dispositivos na rede. Antes da adoção do IP, outros protocolos, como **AppleTalk** e **Internetwork Packet Exchange (IPX)**, foram utilizados para esse propósito. Atualmente, os protocolos padrão são **IPv4** e **IPv6**.

Existem três principais métodos de **switching** utilizados para a transmissão de dados:

- **Packet Switching**: os dados são divididos em pacotes que são enviados através da rede. O dispositivo receptor tem a responsabilidade de remontar os pacotes conforme os recebe.
- **Circuit Switching**: um link de comunicação direto é estabelecido entre os dois dispositivos que estão se comunicando, garantindo uma conexão dedicada.
- **Message Switching**: os dados são divididos em mensagens, que podem ser armazenadas e enviadas posteriormente. Esse método é raramente utilizado.

Para garantir o encaminhamento correto dos pacotes, os **roteadores** precisam descobrir a existência de outros roteadores na rede e manter um caminho eficiente de transmissão com base no **IP de destino**. Essa descoberta pode ocorrer de duas formas:

- **Estática**: os caminhos são configurados manualmente pelo administrador da rede.
- **Dinâmica**: utiliza protocolos como **RIP (Routing Information Protocol)**, **OSPF (Open Shortest Path First)** e **EIGRP (Enhanced Interior Gateway Routing Protocol)** para atualizar automaticamente as rotas disponíveis.

Os **serviços de conexão da camada 3** expandem as funcionalidades da camada 2 para aumentar a confiabilidade da transmissão de dados. Além disso, essa camada gerencia o **controle de fluxo**, garantindo que pacotes sejam distribuídos de maneira eficiente entre diferentes links e rotas para um melhor desempenho.

Os **multilayer switches** possuem a capacidade de realizar o roteamento tanto com base em **endereços IP** quanto em **endereços MAC**, oferecendo maior flexibilidade e eficiência no encaminhamento de pacotes dentro da rede.

5. Camada de transporte/quarta camada

A camada de transporte atua como uma divisão entre as camadas superiores e inferiores do modelo de rede. É nela que estão os protocolos **TCP (Transmission**

Control Protocol) e **UDP (User Datagram Protocol)**, sendo responsável pelo envio de dados em **segmentos**.

O **TCP** é um protocolo confiável e orientado à conexão. Ele garante a entrega correta dos segmentos, detectando e reenviando qualquer segmento perdido. Para estabelecer uma conexão, o TCP utiliza o **three-way handshake**, que segue os seguintes passos:

1. O transmissor envia um pacote **SYN (synchronize)**.
2. O receptor responde com um **SYN-ACK (synchronous acknowledgment)**.
3. O transmissor finaliza o processo enviando um **ACK (acknowledgment)**, confirmando o recebimento.

Já o **UDP** é um protocolo **não orientado à conexão** e não garante a retransmissão de segmentos perdidos. Sua prioridade é a **velocidade**, tornando-o ideal para transmissões em tempo real, como **áudio e vídeo**, onde a latência baixa é mais importante do que a correção de pacotes perdidos.

Outros conceitos importantes desta camada incluem:

- **Windowing**: permite ao cliente ajustar dinamicamente a quantidade de dados enviados em cada segmento. Se houver muitas retransmissões, o tamanho da janela diminui. Se houver poucas ou nenhuma, o tamanho aumenta, otimizando a transmissão.
- **Buffering**: ocorre quando dispositivos como **roteadores** alocam memória para armazenar segmentos temporariamente caso a largura de banda não esteja disponível. Quando há disponibilidade, os dados armazenados no buffer são transmitidos. Se houver um **buffer overflow**, pacotes podem ser descartados.

Exemplos de dispositivos e serviços que operam nesta camada incluem **load balancers** e **firewalls de camada 4**, que fazem o balanceamento de tráfego e a filtragem de pacotes com base em informações de transporte, como números de porta TCP/UDP.

6. Camada de sessão/quinta camada

A camada de sessão é responsável por **configurar, manter e encerrar sessões** entre dispositivos de rede, garantindo que os dados de diferentes sessões não sejam misturados durante a transmissão.

Suas principais funções incluem:

- **Verificação de credenciais de usuários** para garantir que apenas usuários autorizados possam acessar a sessão.
- **Atribuição de números únicos a cada sessão** para identificação e controle.

- **Negociação de serviços**, determinando os recursos necessários para a comunicação.
- **Definição de quem inicia o envio dos dados**, organizando a troca de informações entre os dispositivos.
- **Gerenciamento da transferência de dados**, assegurando que a comunicação ocorra corretamente.
- **Restabelecimento de conexões perdidas**, caso haja uma interrupção inesperada.
- **Confirmação do recebimento dos dados**, garantindo que as informações chegaram ao destino.
- **Encerramento da sessão**, que pode ocorrer por acordo entre as partes ou quando uma delas decide se desconectar.

Exemplos de protocolos da camada de sessão incluem **H.323**, utilizado para chamadas de **VoIP**, e **NetBIOS**, que permite o compartilhamento de arquivos em redes locais.

7.Camada de apresentação/sexta camada

A **camada de apresentação** é responsável pelo **formatamento dos dados** e pela **criptografia**, garantindo que as informações transmitidas possam ser corretamente interpretadas pelo sistema receptor.

Suas principais funções incluem:

- **Formatar os dados** em padrões reconhecidos, como **ASCII**, **GIF**, **JPG**, **MPG**, **MOV**, entre outros.
- **Assegurar a compatibilidade** dos dados entre sistemas diferentes, garantindo que a informação seja interpretada corretamente.
- **Fornecer estrutura apropriada** para a apresentação dos dados.
- **Negociar a sintaxe de transferência de dados** para a camada de aplicação.
- **Aplicar criptografia** para garantir a confidencialidade, como no caso do **SSL (Secure Sockets Layer)** e do **TLS (Transport Layer Security)**, que protegem a comunicação entre um website e um computador.

Exemplos de protocolos e formatos de dados nesta camada incluem **HTML**, **XML**, **GIF**, **JPG**, **MPG**, **MOV**, **SSL** e **TLS**.

8.Camada de aplicação/sétima camada

A **camada de aplicação** é responsável por fornecer **serviços de aplicação** e permitir a comunicação entre os usuários e outros computadores ou sistemas. Essa camada é onde ocorrem as interações diretas com os serviços de rede.

Suas principais funções incluem:

- **Transferência de arquivos**, e-mails, acesso remoto e comunicação entre processos cliente/servidor.
- **Envio de anúncios de disponibilidade**, como ocorre no **Active Directory** no Windows.
- **Facilitar a comunicação entre softwares e a rede**, garantindo que aplicativos possam utilizar os protocolos apropriados.

Exemplos de **protocolos** desta camada incluem:

- **POP3, IMAP, SMTP** – utilizados para envio e recebimento de e-mails.
- **HTTP, HTTPS** – responsáveis pela navegação na web.
- **DNS** – converte nomes de domínio em endereços IP.
- **FTP** – permite a transferência de arquivos.
- **TELNET** – possibilita acesso remoto a sistemas.
- **SNMP** – gerencia dispositivos de rede.

Esta camada é essencial para a interação entre usuários e serviços de rede, tornando possível a utilização da internet, o envio de mensagens e a transferência de arquivos.