

CAMADA DE APLICAÇÃO

COMUNICAÇÃO ENTRE PROCESSOS

- É a camada que define o motivo da existência das redes, de acordo com a sua época:
 - **Década de 80:** Necessidade de correio eletrônico, acesso remoto a certas coisas e transferência de arquivos.
 - **Década de 90:** WEB.
 - **Final do Milênio:** Vídeo em tempo real, rádio e telefonia, mensagens instantâneas, compartilhamento de arquivos P2P.
- É a camada responsável pela comunicação entre os aplicativos do usuário e a camada de transporte.
- Tipos de protocolos que operam nessa camada:
 - **HTTP** (HyperText Transfer Protocol)
 - **SMTP** (Simple Mail Transfer Protocol)
 - **FTP** (File Transfer Protocol)
 - **DNS** (Domain Name System)
 - **DHCP** (Dynamic Host Configuration Protocol)

ARQUITETURA DE APLICAÇÕES MODERNAS

- Cliente - Servidor
- P2P
- Híbrida: Cliente - Servidor / P2P.

CLIENTE - SERVIDOR

Um host (Servidor) sempre está em funcionamento e sempre que um cliente chegar com um pedido, ele irá mandar a resposta.

EXEMPLO: Browser <-> Servidor Web

P2P

Não tem um servidor sempre funcionando, na realidade existem pares de hosts (máquinas), os chamados peers, que se comunicam entre si (par-a-par), removendo a necessidade de precisar ter uma máquina sempre funcionando.

- Um host participante pode mudar seu endereço IP sempre que for ligado a conexão.
- Pares geram trabalho, requisitando arquivos e também entregam capacidade de serviço ao sistema.

HÍBRIDA: CLIENTE - SERVIDOR / P2P

Alguns serviços funcionam de acordo com a arquitetura P2P, como troca de arquivos, enquanto outros funcionam de acordo com a arquitetura Cliente - Servidor, como consultas ao servidor para saber que pares possuíam o arquivo para a troca.

COMUNICAÇÃO ENTRE PROCESSOS:

Os programas se comunicam através de processos, um processo é uma troca de mensagem por meio da rede.

PROCESSOS CLIENTE:

- Processo que faz a requisição e inicia a comunicação.

PROCESSOS SERVIDOR:

- Processo que espera ser contatado para iniciar a sessão.

ENDEREÇAMENTO DE PROCESSOS:

Para enviar uma mensagem, o processo originador precisa identificar o processo destinatário. Necessitando duas informações: O endereço do host de destino e o ID do processo.

- Para fazer essa identificação, são utilizados **endereços IP** (IPv4 ou IPv6) que **identificam exclusivamente um host**.
- Para identificar o processo que está rodando no outro host, é utilizado o **número de porta**, que atende essa finalidade através de números que podem estar relacionados com aplicações padrão:
 - **FTP: 20 e 21**
 - **STMP: 25**
 - **HTTP: 80**
 - **DHCP: 67**
- Através desses números, é possível saber o tipo de conteúdo do pacote de dados que está sendo transmitido.
- Cada número de porta é um número de 16 bits (podem ir de 0 a 65535).
- Além disso, para identificar os protocolos de transporte (TCP/UDP), existem um números de protocolos, que são apenas um byte simples, ou arquivos de texto.

Naturalmente, alguns processos que precisarem de uma porta também alocarão dinamicamente, usando as que estiverem livres.

SOCKETS:

Funciona como se fosse a porta de uma casa, um processo envia mensagens para a rede e recebe mensagens por meio do socket. (Interface entre a camada de aplicação e de transporte) ou simplesmente API (Application Programming

Interface).

- É chamado de socket a combinação de um endereço IP com um número de porta.
- Identificador único, como falado anteriormente.
- Um par de sockets define uma conexão para protocolos orientados à conexão, como o TCP.

PROTOSCOLOS DA CAMADA DE APLICAÇÃO:

- Define os **tipos de mensagens trocadas**: De requisição e de resposta.
- **A sintaxe dos vários tipos de mensagens**: Quais os campos da mensagem.
- **A semântica dos campos**: O significado da informação nos campos.
- **Regras**: Determina como deve ser feito o envio e resposta das mensagens.
- São apenas uma parte da aplicação de rede.

NECESIDADES DE SERVIÇO DE UMA APLICAÇÃO:

- **Transferência Confiável de Dados**
 - Não permitir perda de pacotes na transferência de dados.
- **Largura de Banda**
 - Velocidade de transmissão de dados.
- **Temporização**
 - Temporização para entrega de dados (Time Limit Exceeded).