
Sistemas Distribuídos

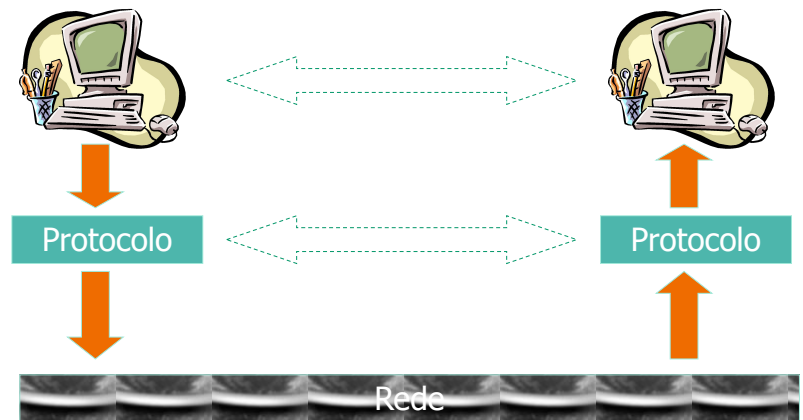
Modelo Cliente Servidor
Sockets

Protocolos

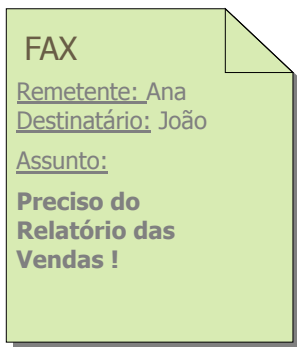
Introdução

- O que é um protocolo de comunicação?
 - Conjunto de regras e procedimentos para que duas entidades distintas possam estabelecer um canal de comunicação
 - Etapas de estabelecimento e finalização de conexão
 - Cabeçalhos indicativos de numeração e ordem dos pacotes
 - Endereço do emissor e receptor

Protocolos na Internet



Comparação



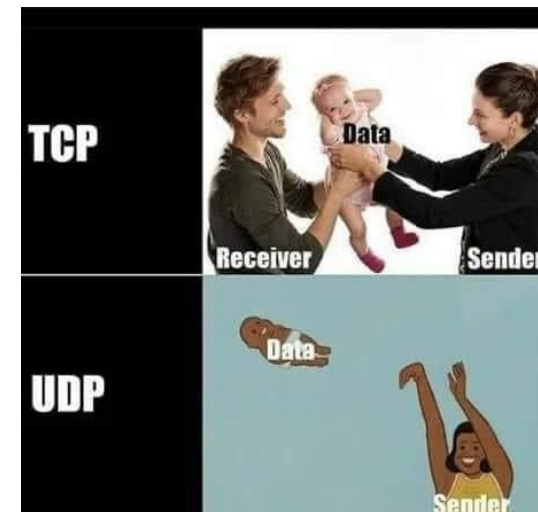
Vida Real



Source	200.18.92.1
Destination	187.231.97.2
Flags	1 0 1 1 0 1
Data	"Preciso do Relatório das Vendas"

Pacote de Dados

TCP x UDP

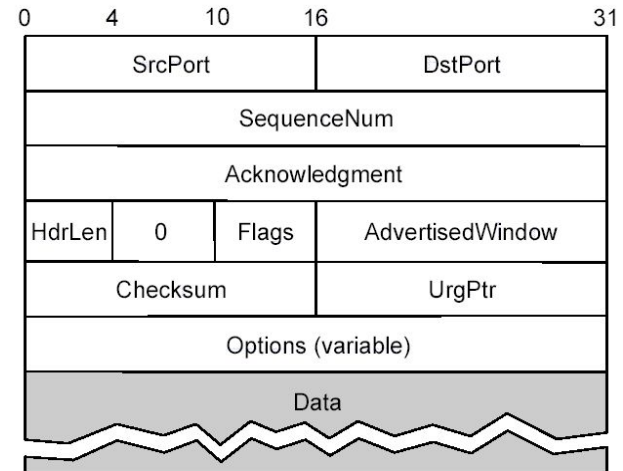


Protocolo TCP/IP

- Protocolo atualmente utilizado na Internet.
- Baseado no modelo OSI da ISO (7 camadas)
- TCP – protocolo de transporte que utiliza os serviços do protocolo IP para garantir estabelecimento de conexões e integridade de dados
- IP – protocolo de rede responsável pelo endereçamento das máquinas (endereço IP) e rotas entre dispositivos.

Protocolo TCP

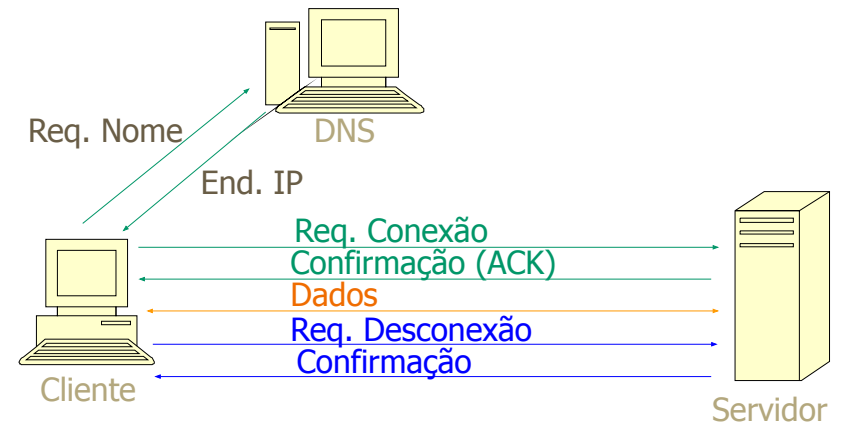
- **Header TCP:**



Protocolos TCP

- **Transmission Control Protocol (TCP):**
- Protocolo orientado à conexão:
 - Exige o estabelecimento de um canal lógico para iniciar a transmissão de dados, em 3 fases:
 - Fase de conexão
 - Fase de dados
 - Fase de desconexão
- Exemplos de aplicação:
 - TELNET, Web Browser, ...

Etapas em uma Conexão TCP



Sockets

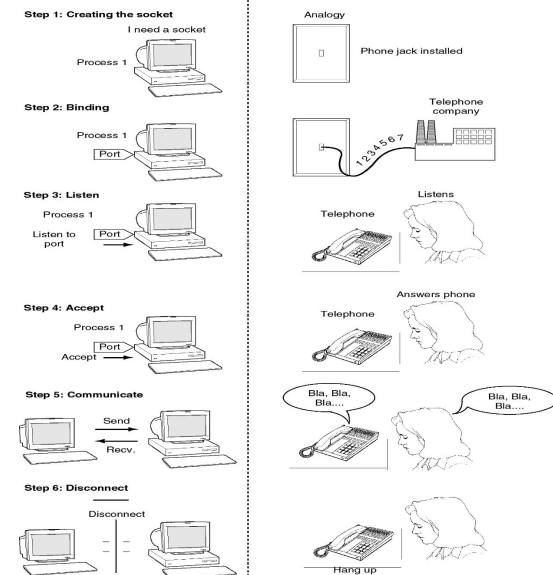
Introdução: Sockets

- Para estabelecer a Comunicação Interprocesso nos Sistemas Distribuídos e para permitir que processos se comuniquem na troca de dados ou acessos a recursos ou serviços em processadores remotos, se faz necessário o uso de um mecanismo de serviços de transporte;
- Um dos mecanismos mais utilizado é o *Socket*;
- *Sockets* é a maneira mais popular de utilizar as funcionalidades de comunicação TCP/IP;
- Todos os mecanismos *Sockets* são gerenciados pela camada de transporte;
- Existem diversas APIs *Sockets* (*Application Program Interface*) e as mais populares são do ambiente Unix, bem como a *WinSock* do Windows.

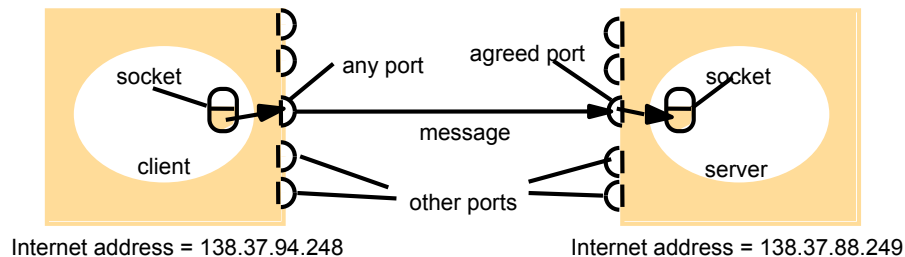
Socket: Definição

- Um *Socket* é um ponto final (*endpoint*) de um canal bidirecional de comunicação entre dois programas rodando em uma rede;
- Cada *Socket* tem os seguintes endereços de *endpoint*:
 - **Endereço local** (número da porta) que refere-se ao endereço da porta de comunicação para camada de transporte;
 - **Endereço global** (nome *host*) que refere-se ao endereço do computador (*host*) na rede.

Socket: Uma analogia



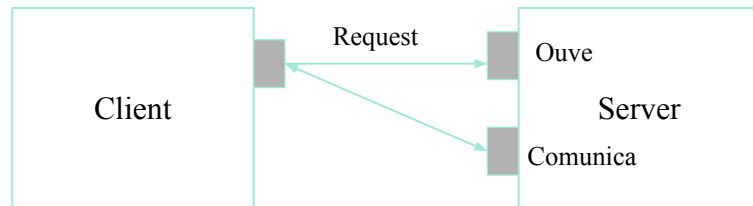
Sockets e Portas



Socket: Conexão

- O servidor apenas fica “ouvindo” o *Socket* aguardando um pedido de conexão do cliente;
- O cliente sabe o nome do *host* e qual porta está associada à aplicação servidora;
- Assim que o servidor aceitar a conexão, este cria um novo *Socket* (e conseqüentemente o associa a uma nova porta) e pode ficar esperando novas conexões no *Socket* original enquanto atende às requisições do cliente pelo novo *Socket*.

Socket: Conexão

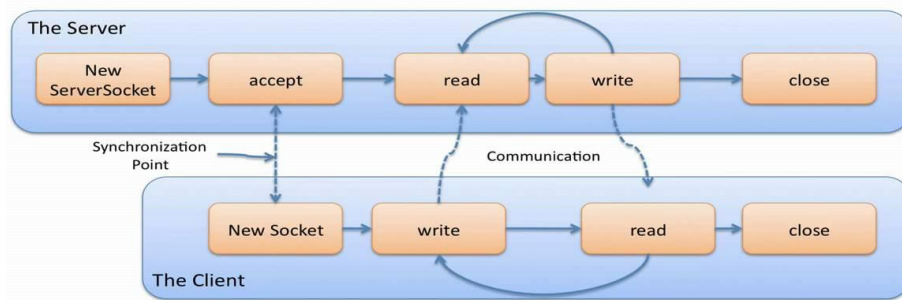


Sockets en JAVA



Modelo

Java Socket Overview



Tipos de Sockets em Java

- Pacote *java.net*
 - *java.net.Socket*
 - *java.net.ServerSocket*
- *Socket* - usado em cada lado do canal de comunicação bidirecional.
- *ServerSocket* - responsável por ficar aguardando pedidos de conexão dos clientes