

Redes de Computadores

Arquitetura de redes

Aula 02

HTTP: Mensagem de requisição (formato e exemplo)

Requisição
Cabeçalhos
[nome; valor]
....
[nome; valor]
Linha em branco (CR/LF)
Corpo (presente apenas em algumas mensagens)

```
GET /~asc/redes/http.html HTTP/1.1
Host: www.someschool.edu
Connection: close
User-agent: Mozilla/4.0
Accept-language: fr
```

Mensagem de requisição

- Método GET não possui corpo
- Método POST envia aqui as informações

Motivação: Exemplo de sessão http

- Acesso a página <http://www.inf.ufrgs.br/~asc/redes/http.html>



Problemas:

1. Como pedir a página? Como receber/interpretar a página?
2. E se as máquinas tiverem sistema operacional e hardware diferentes? Como vão "se entender"?
3. Servidores web (Tomcat, Apache, IIS...): qual está em uso? Browser (Firefox, Chrome, Opera...): qual está em uso?
4. Vários outros: como localizar o servidor web? Como chegar lá? Como tratar erros?

Criando regras a serem respeitadas por todos, independente do sistema
= Protocolos

HTTP: Mensagens de resposta (formato e exemplo)

Status
Cabeçalhos
[nome; valor]
...
[Nome; valor]
Linha em branco (CR/LF)
Corpo (presente apenas em algumas mensagens)

```
HTTP/1.1 200 OK
Connection: close
Date: Thu, 06 Aug 1998 12:00:15 GMT
Server: Apache/1.3.0 (Unix)
Last-Modified: Mon, 22 jun 1998 09:23:24 GMT
Content-Length: 6821
Content-Type: text/html
```

Mensagem de resposta

(data data data Data data data data)

Motivação: Exemplo de sessão http

- ❑ Acesso a página
 - <http://www.inf.ufrgs.br/~asc/redes/http.html>
- ❑ Sessão wireshark
 - [Arquivo http 1.1 exemplo](#)

Redes de Computadores

5



Introdução

- ❑ Comunicação de dados envolve diferentes entidades
 - Entidade=qualquer dispositivo com capacidade de enviar/receber dados
 - Heterogeneidade tanto em nível de software como de hardware
- ❑ Solução para heterogeneidade: uso de convenções (protocolo)
- ❑ Protocolo
 - Conjunto de regras e de convenções para permitir a troca de informações entre entidades

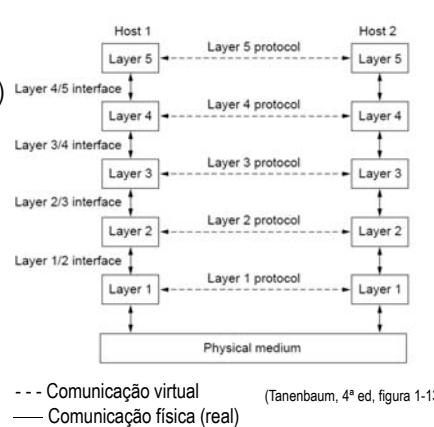
Instituto de Informática - UFRGS

Redes de Computadores

6

Camadas, protocolos e interfaces

- ❑ Objetivo:
 - Reduzir complexidade do projeto (“Dividir para vencer” -Júlio Cesar)
 - Tarefa de comunicação é quebrada em módulos



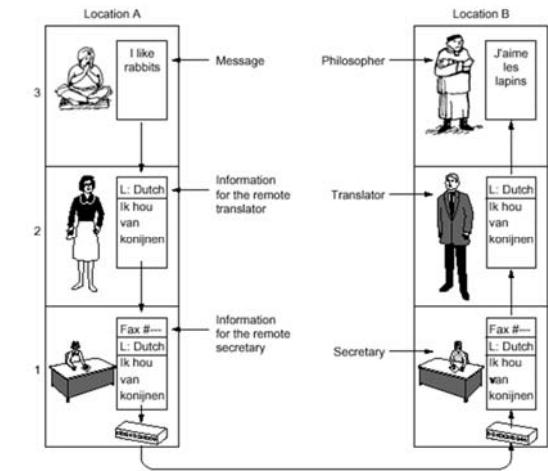
Redes de Computadores

7

Instituto de Informática - UFRGS

Redes de Computadores

Princípio de funcionamento (analogia): um clássico



8

Conceitos básicos

- ❑ Hierarquia de camadas
 - Atividades diferentes, com responsabilidades diferentes realizadas e organizadas em uma determinada ordem → camadas funcionais
- ❑ Serviços
 - Procedimentos que uma camada funcional oferece para uma outra camada
 - Emissor : camada N usa serviços da camada N-1
 - Receptor : camada N-1 gera eventos para a camada N (aciona serviços)
- ❑ Interface
 - Descrição dos serviços providos por uma camada
- ❑ Peer (pares)
 - Entidades, em máquinas distintas, que correspondem a uma mesma camada de protocolo

Arquitetura de redes

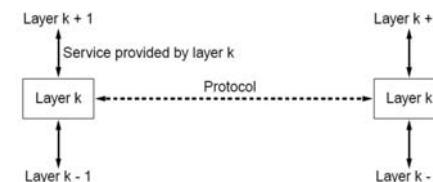
- ❑ Conjunto de camadas e de protocolos
 - Arquitetura TCP/IP
 - Aplicação (http, https, dns, smtp, pop, imap, ftp, telnet, etc...)
 - Transporte (tcp, udp, rsvp, sctp, dccp ...)
 - Rede (IPv4, IPv6, IPsec, igmp, icmp, ...)
 - Enlace (IEEE 802.3, IEEE 802.11,...)
- ❑ Conceito de pilha de protocolos
 - Lista de protocolos usados por um sistema

Modelo de referência OSI (MR-OSI)

- ❑ Open Systems Interconnection (OSI)
- ❑ Desenvolvido pela International Organization for Standardization (ISO)
- ❑ Organizado em sete camadas (ou níveis)
 - Físico, enlace, rede, transporte, sessão, apresentação e aplicação
 - É um modelo: define funcionalidades sem se preocupar com a implementação
 - Protocolos ISO vieram depois do modelo
- ❑ Três conceitos básicos:
 - Serviços
 - Interfaces
 - Protocolo

Serviço

- ❑ Implementados por uma camada cuja função é oferecer primitivas e eventos para a camada superior
- ❑ A camada N de uma entidade se comunica com a camada N de outro entidade utilizando os serviços da camada N-1
- ❑ Fornece a semântica de funcionamento de uma camada
 - Informa o que a camada faz e não como é feito
 - Camada é similar a um objeto (métodos = serviços oferecidos)



(Tanenbaum, 4^a ed, figura 1-19)

Tipos de serviços (semântica de associação)

- ❑ Orientados a conexão (*connection oriented*)
- ❑ Não orientados a conexão (*connectionless*)

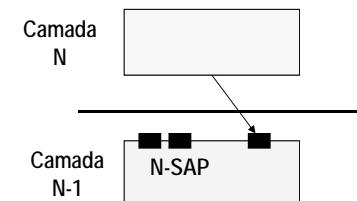
O que é uma conexão (confiável)?

É um tipo de serviço que oferece garantia de:

- ① entrega dos dados;
- ② ordenamento (ordem de recepção é igual a ordem da emissão);
- ③ não haver duplicação de dados;

Interface

- ❑ Forma de identificar o serviço e fornecer parâmetros
 - Formalmente: SAP + Estrutura de dados
- ❑ *Service Access Point (SAP)*
 - Situado na “fronteira” das camadas
 - Ponto de entrada para acessar o serviço
 - Identificador único de um serviço N



Exemplo: biblioteca de *sockets*

Serviço orientado a conexão (STREAM): mapeia para TCP
Serviço não orientado a conexão (DGRAM): mapeia para UDP

Protocolos

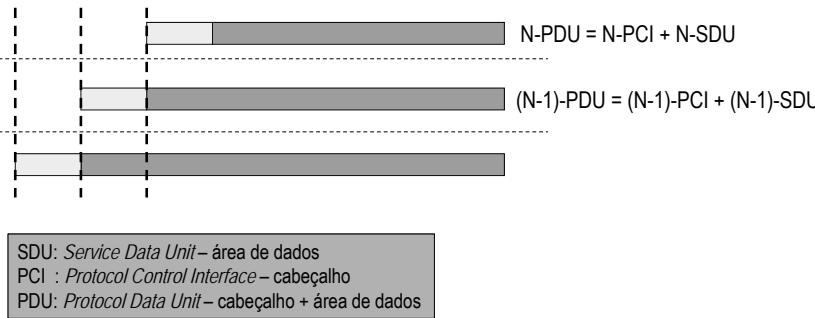
- ❑ Elementos chaves são
 - Sintaxe:
 - Estrutura ou formato dos dados e a ordem com que são apresentados
 - e.g.: endereço de destino + endereço fonte + mensagem
 - Semântica
 - Como deve ser interpretado um determinado conjunto de bits e qual a ação a ser executada
 - Temporização
 - Determinação de quando que os dados podem ser enviados e quanto rápido isso pode ser feito
- ❑ *Protocol Data Unit (PDU)*
 - Formado por um cabeçalho e uma área de dados

Funções genéricas de um protocolo

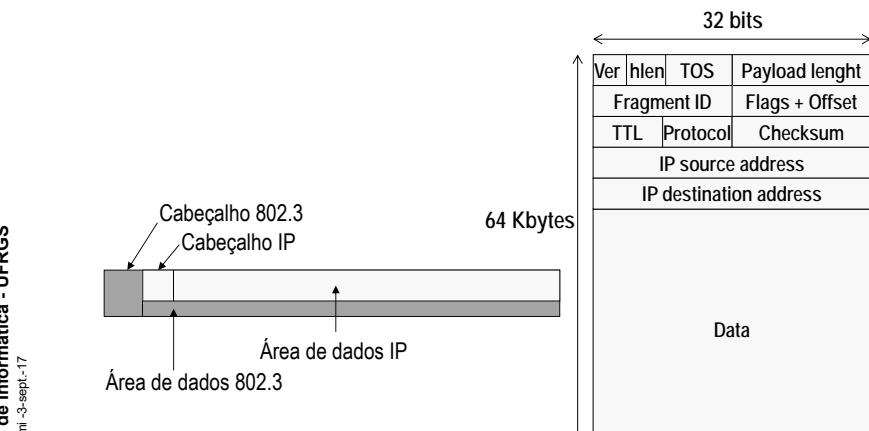
- ❑ Encapsulamento
- ❑ Segmentação (fragmentação) e remontagem
- ❑ Controle de conexão
- ❑ Controle de fluxo
- ❑ Controle de erro
- ❑ Endereçamento
- ❑ Multiplexação

Encapsulamento

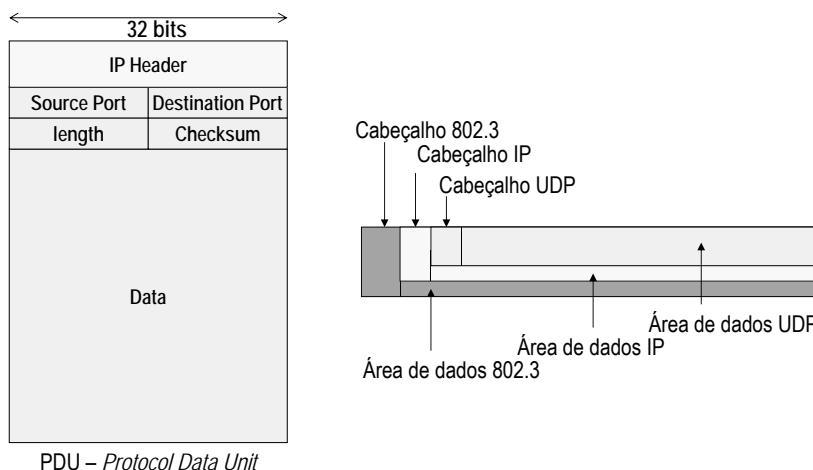
- ❑ Inserção de informações de controle de uma camada N junto aos dados
 - Exemplo: endereços, códigos para detecção de erro, bits de controle etc



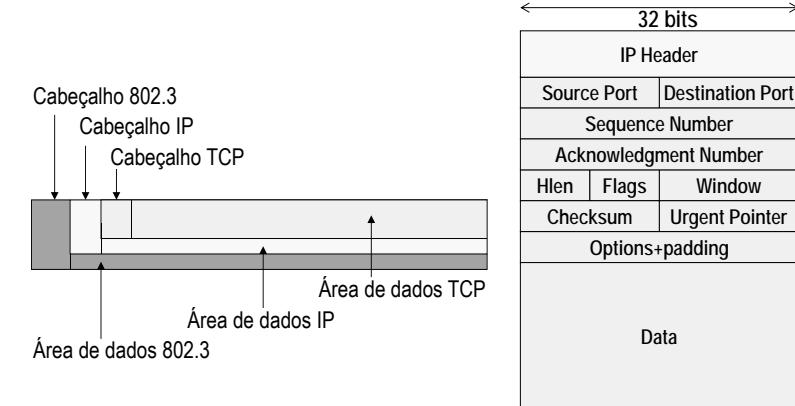
Encapsulamento e datagrama IP



Datagrama UDP e encapsulamento



Segmento TCP e encapsulamento



Fragmentação (segmentação) e remontagem

- ❑ PDU possuem um tamanho máximo pré-definido

- Restrições de tecnologia (e.g. quadros IEEE 802.3 tem 1500 bytes de dados)
- Fornecer acesso equitável a rede
- Executar um controle de erro mais eficiente
- Facilitar dimensionamento e gerenciamento de buffers

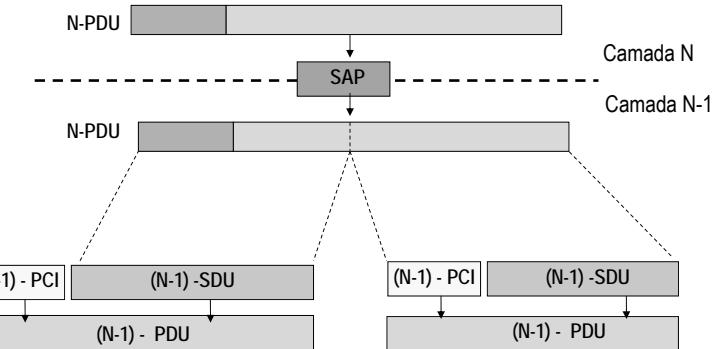
- ❑ O que fazer quando $N\text{-PDU} > (N-1)\text{-PDU}$?

- Solução: quebrá-la em várias (fragmentação)

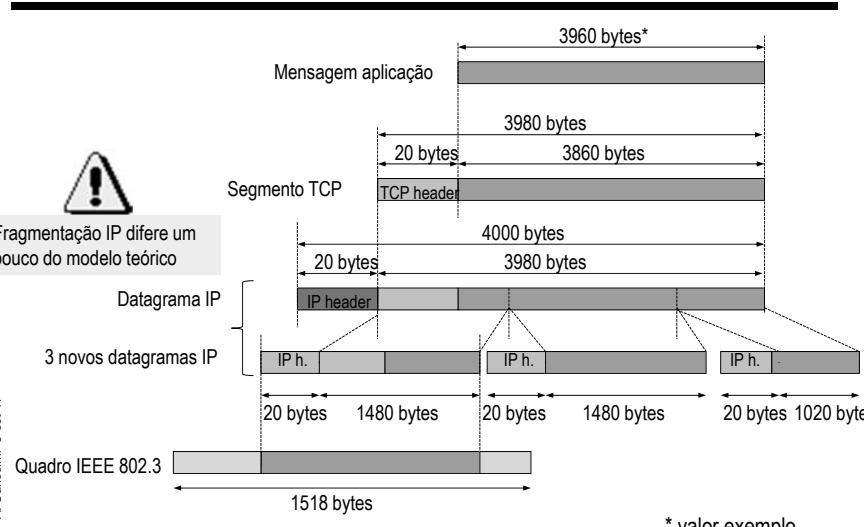
- ❑ Fragmentação deve ser transparente à camada N

- Remontagem antes de entregá-la a camada N no destino

Fragmentação e encapsulamento de PDUs



Encapsulamento e fragmentação no TCP/IP e IEEE802.3



* valor exemplo

Endereçamento

- ❑ Identificação de entidades, serviços e abstrações empregadas na comunicação
 - Endereço da placa de rede (MAC)
 - Endereço IP
 - Identificador do processo (porta)
- ❑ Escopo: local versus global
- ❑ Modos de endereçamento:
 - Unicast: uma única entidade
 - Multicast: um sub-conjunto das entidades
 - Broadcast: todas entidades

Multiplexação e demultiplexação

- ❑ Situação:
 - Uma determinada camada pode oferecer mais de um serviço
 - Um serviço da camada N-1 pode atender mais de um serviço da camada N
- ❑ Questão: como encaminhar corretamente as PDUs?
 - Multiplexação e demultiplexação
 - Ex.: O IP carrega PDUs dos protocolos TCP e UDP (transporte)
 - Identificadores de protocolo: TCP =6; UDP =17; ICMP=1

Controle de fluxo

- ❑ Mecanismo empregado para o receptor limitar a quantidade de informação enviada por um transmissor
- ❑ O objetivo é evitar que o transmissor sature o receptor de dados fazendo com que ele perca dados
- ❑ Exemplos: stop-and-wait, go back-N, créditos etc
 - Na arquitetura TCP/IP apenas o TCP executa controle de fluxo

Controle de erro

- ❑ Forma de evitar perdas ou alterações por erros de transmissão
- ❑ Baseado em dois mecanismos:
 - Detecção do erro seguida de sua correção
 - Detecção do erro seguido por um pedido de retransmissão (ou retransmissão por time-out)
- ❑ Na arquitetura TCP/IP, apenas o TCP possui controle de erro

Controle de conexão

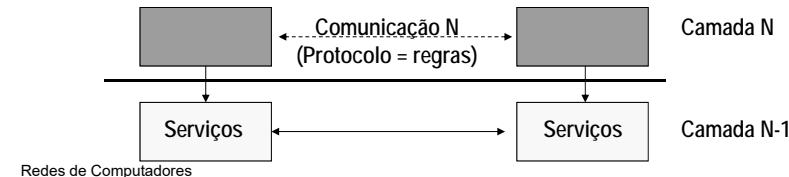
- ❑ Conexão é um serviço que visa:
 - Garantia de entrega de PDUs (sem perdas)
 - Ordenamento (PDUs são entregues na mesma ordem que foram emitidas)
 - Sem duplicação de PDUs
- ❑ Implementando com controles negociados em três etapas:
 - Estabelecimento da conexão: negociação dos parâmetros de controle
 - Transferência de dados: troca de dados garantindo a entrega, não duplicação e ordenamento
 - Encerramento
- ❑ Na arquitetura TCP/IP, apenas o TCP efetua controle de conexão
 - IP e UDP são protocolos não orientados a conexão

Leituras adicionais

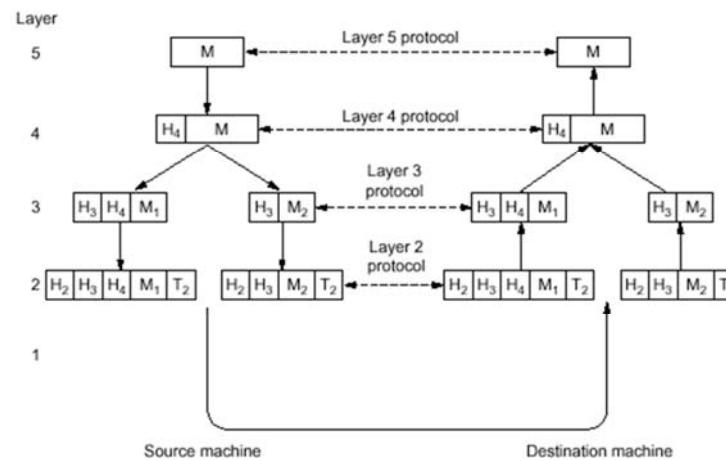
- Tanenbaum, A.; Wethreall, D. *Redes de Computadores* (5^a edição), Editora Pearson Education, 2011.
 - Capítulo 1
- Carissimi, A.; Rochol, J; Granville, L.Z; *Redes de Computadores*. Série Livros Didáticos. Bookman 2009.
 - Capítulo 2
- Kurose, J.F.; Ross, K.W. *Redes de Computadores e a Internet: uma abordagem top-down*. 5^a edição. Addison-Wesley. São Paulo. 2010.
 - Capítulo 1 (exceto 1.4)

Arquitetura de redes

- Conjunto de camadas e de protocolos
- Objetivo:
 - Reduzir complexidade do projeto (“Dividir para vencer” Júlio Cesar)
 - Tarefa de comunicação é quebrada em módulos
- Cada módulo:
 - Implementado por uma camada cuja função oferecer serviços e primitivas de comunicação as camadas superiores.
 - A camada N de uma máquina se comunica com a camada N da outra, utilizando os serviços da camada N-1.



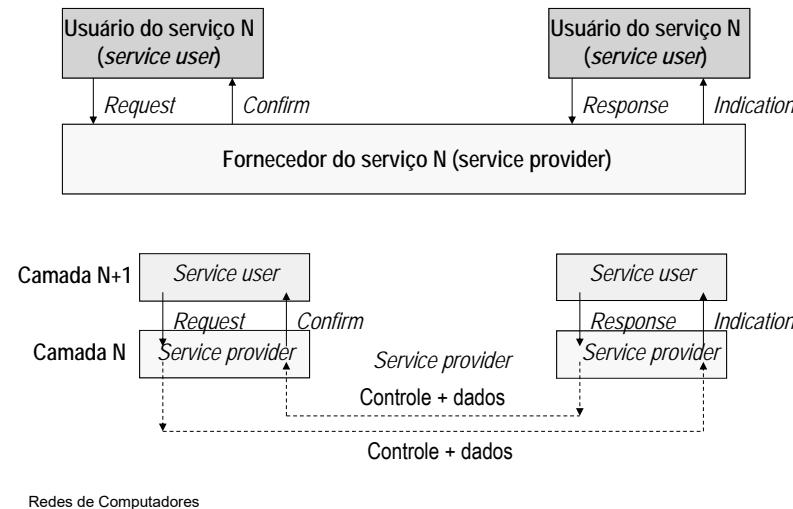
Fragmentação e encapsulamento de PDUs



Serviço

- Conjunto de primitivas e eventos disponibilizados por uma camada a outra
- Fornece a semântica de funcionamento camada
 - Informa o que a camada faz e não como é feito
- Formalmente possui quatro primitivas:
 - Emissor: *request* e *confirmation*
 - Receptor: *indication* e *reponse*
- Define:
 - Serviços confirmados: seqüência *request-indication-reponse-confirmation*
 - Serviços não-confirmados: seqüência *request-indication*

Modelo de base de serviços

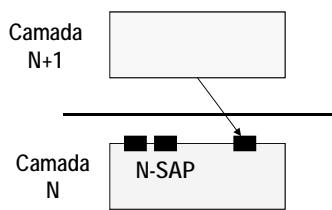


Tipos de serviços (semântica de associação)

- ❑ Orientados a conexão (*connection oriented*)
 - Analogia ao sistema telefônico
- ❑ Não orientados a conexão (*connectionless*)
 - Analogia ao sistema postal (correio)
- ❑ O que é uma conexão ?
 - É um tipo de serviço que oferece garantia de (1) entrega dos dados; (2) ordem de recepção é igual a da emissão e; (3) não haver duplicação

Interface

- ❑ Forma de identificar o serviço e fornecer parâmetros
 - Formalmente: SAP + Estrutura de dados
- ❑ Service Access Point (SAP)
 - Situado na “fronteira” das camadas
 - Ponto de entrada para acessar o serviço
 - Identificador único de um serviço N
- ❑ Estrutura de dados
 - Passagem de parâmetros entre camadas



Estrutura de dados de serviços

- ❑ Service Data Unit (SDU)
- ❑ Protocol Control Information (PCI)
- ❑ Protocol Data Unit (PDU)
- ❑ Interface Data Unit (IDU)
- ❑ Interface Control Information (ICI)

Esquema de estrutura de dados

