

Capítulo 1

PROTOCOLOS FUNDAMENTAIS DA INTERNET

protocolos TCP/IP – são um conjunto de standards que permitem que computadores de todas marcas e feitos comuniquem entre si

Algumas datas históricas:

1968 - Advanced Research Projects Agency (ARPA) e Dept of Defense (DOD) criam a ARPANET

1973 - TCP/IP versão 1

1978 - TCP/IP versão 4

1982 - Transição de ARPANET para IP

1990 - HTML

1991 - HTTP

ARQUITECTURA TCP/IP — LAYERS (CAMADAS)

- os protocolos TCP/IP descrevem várias camadas de software:

Application	telnet, ftp, HTTP, SMTP
Transport	TCP, UDP
Network	IP, ICMP
Link	Ethernet
physical	

LAYERS

Physical layer - descreve o nível físico: níveis de tensão, modulação, cablagem, conectores etc. Exemplos: ethernet, token ring, ADSL, wireless Wi-Fi 802.11b/g

Link layer - corresponde ao device driver incluído no sistema operativo. É o software que faz a formatação de dados mais apropriada para o nível físico. Exemplo: ethernet, ppp, pppoe

Network layer - (internet layer) responsável pelo transporte dos pacotes na rede: encaminhamento (routing)

Transport layer - responsável pelo fluxo de dados—integridade, fraccionamento e re-agrupamento dos dados

LAYERS (2)

Application layer - software que faz a interface com o utilizador.

Exemplos:

telnet (login remoto)

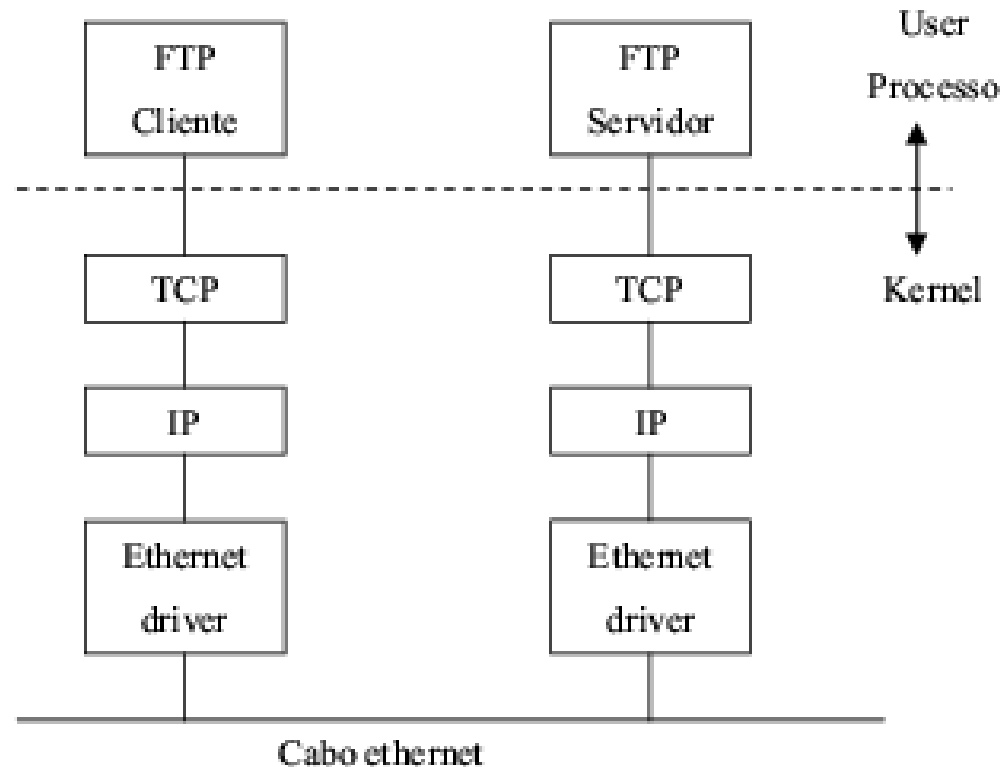
ftp (file transfer protocol)

ssh (secure shell)

SMTP e POP3 (mail transfer protocols)

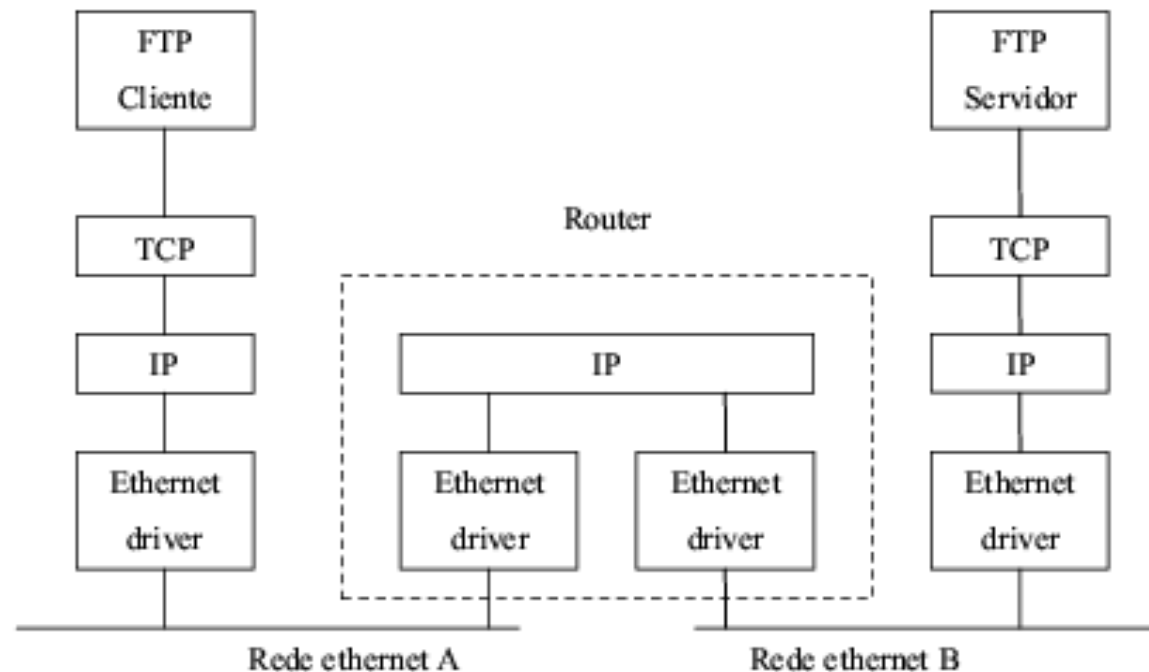
HTTP

EXEMPLO: FTP entre dois hosts (mesma rede)



- A maior parte das aplicações funcionam em modo cliente servidor
- O servidor oferece um serviço—é um programa que corre em background (daemon) e está permanentemente à espera que o cliente se ligue

EXEMPLO: FTP entre dois hosts (redes diferentes)



- A Internet é feita de redes independentes e fisicamente diferentes
- As redes A e B podem ser de tipos físicos diferentes (ethernet, token ring, adsl, wireless etc)

ENDEREÇOS IP

- 32 bits agrupados 8 a 8. Todos o equipamento na Internet tem que ter um numero único
- Exemplo

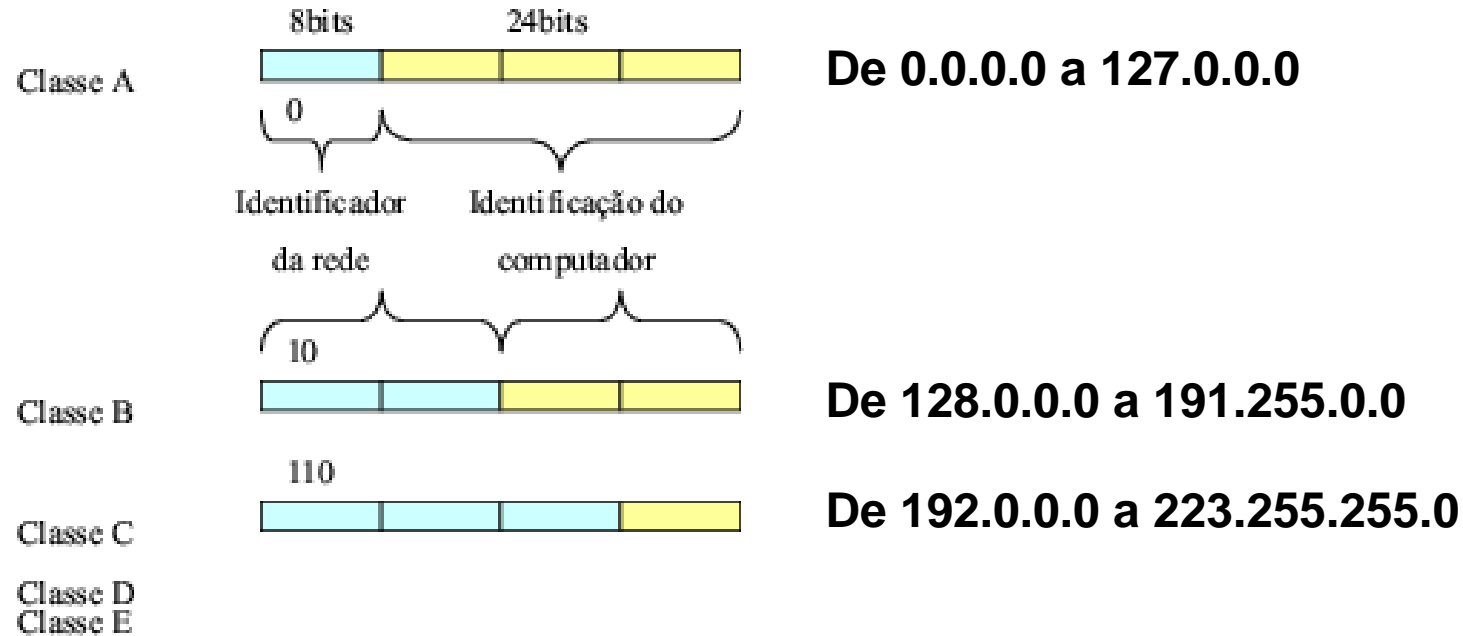
140.252.13.33 decimal

1000 1100 . 1111 1100 . 0000 1101 . 0010 0001 binário

8C.FC.0D.21 hexadecimal

CLASSES

- Os endereços IP dividem-se em classes



- Classe A - 8 bits identificam a rede
- Classe B - 16 bits identificam a rede
- Classe C - 24 bits identificam a rede

MÁSCARA DE REDE

- É um conjunto de 32 bits
 - os bits a 1 identificam a rede
 - os bits a 0 identificam o host
- Todos os hosts na mesma rede tem os mesmos bits de identificação da rede
- Faz-se uma operação AND da máscara com o endereço IP para encontrar a rede

Mascara de rede.

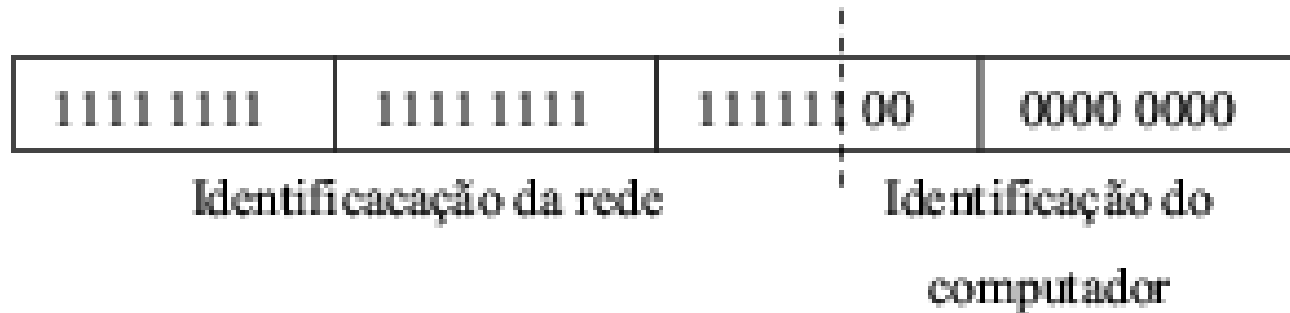
Classe A – máscara 255.0.0.0

Classe B – máscara 255.255.0.0

Classe C – máscara 255.255.255.0

MÁSCARA DE REDE GENERICA

- Numero de bits que identificam a rede não precisa de ser um multiplo de 8 bits
- Classless network
- Exemplo: mascara 255.255.252.0



MÁSCARA DE REDE GENERICA (2)

- Todos os bits de identificação do host a zero -> endereço da rede
- Todos os bits de identificação do host a um -> endereço de broadcast

EXEMPLO

- Qual o endereço de rede e o endereço de broadcast?

IP 10.10.23.27/255.255.252.0

IP	0000 1010.0000 1010.0001 01	11.0001 1011
Mascara	1111 1111.1111 1111.1111 11	00.0000 0000

AND	0000 1010.0000 1010.0001 01	00.0000 0000
-----	-----------------------------	--------------

Endereço da rede: 10.10.20.0

Endereço de broadcast: 10.10.23.255

ENDEREÇOS IP RESERVADOS (PRIVADOS)

10.x.x.x 127.x.x.x	Classe A loopback
172.16.x.x 172.31.x.x	Classe B
192.168.0.x 192.168.255.x	Classe C

DOMAIN NAME SYSTEM (DNS)

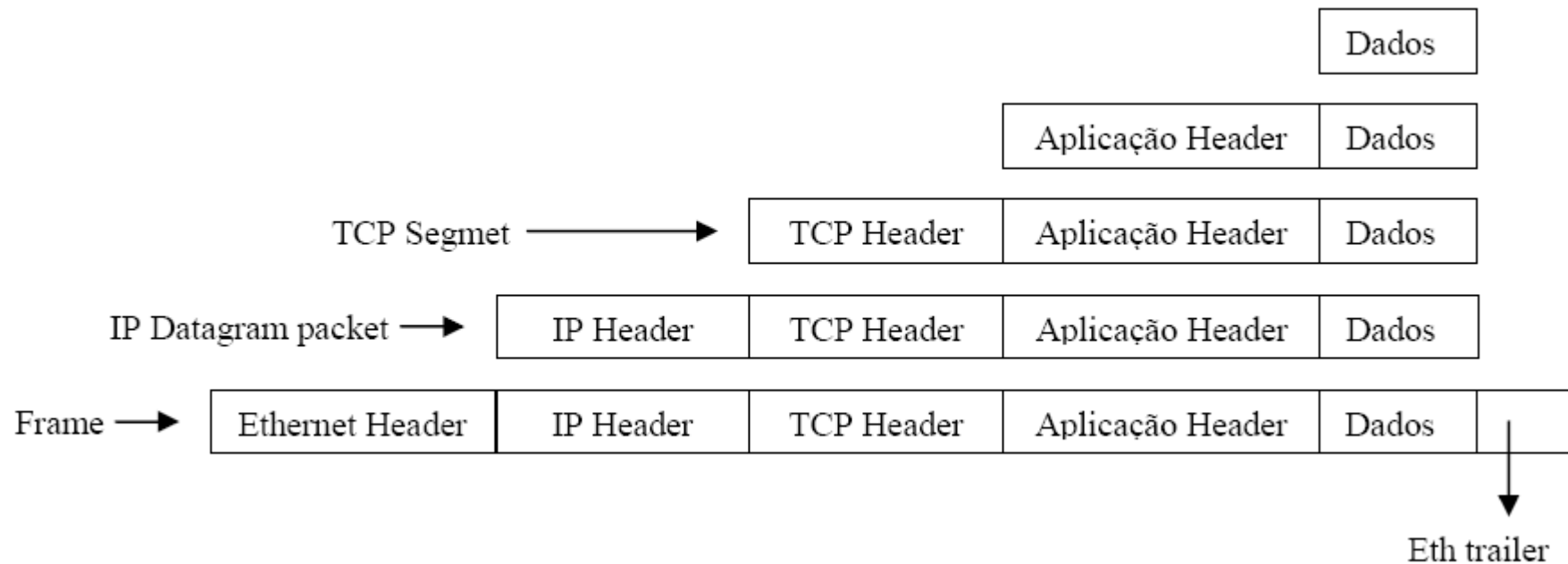
- Endereços IP não são práticos para nós (humanos!)
- As pessoas preferem recordar nomes
- Exemplo

193.136.224.30 www.ualg.pt

- O servidor DNS converte IPs em nomes e vice-versa
- As aplicações web são clientes de DNS, mas também há clientes dedicados (por exemplo nslookup)

ENCAPSULAMENTO

- Cada nível no protocolo coloca um header e um trailer (cabeçalho e terminação) nos dados que lhe são transmitidos



- Importante: cada nível NÃO altera os dados que recebe

ENCAPSULAMENTO (2)

- Application header: informação sobre os dados (conteúdo, formato, compressão etc)
- TCP header: identifica a aplicação que gera os dados
- IP header: informação de routing (IP origem, IP destino)
- Ethernet header: endereços físicos de origem e destino

MODELO CLIENTE-SERVIDOR

- A maior parte das aplicações na Internet funcionam no modelo cliente-servidor
- O servidor é concorrente – quando o cliente se liga ao servidor, este gera imediatamente uma cópia de si mesmo para servir o cliente
- Assim, o servidor está sempre disponível para aceitar pedidos de ligação do cliente

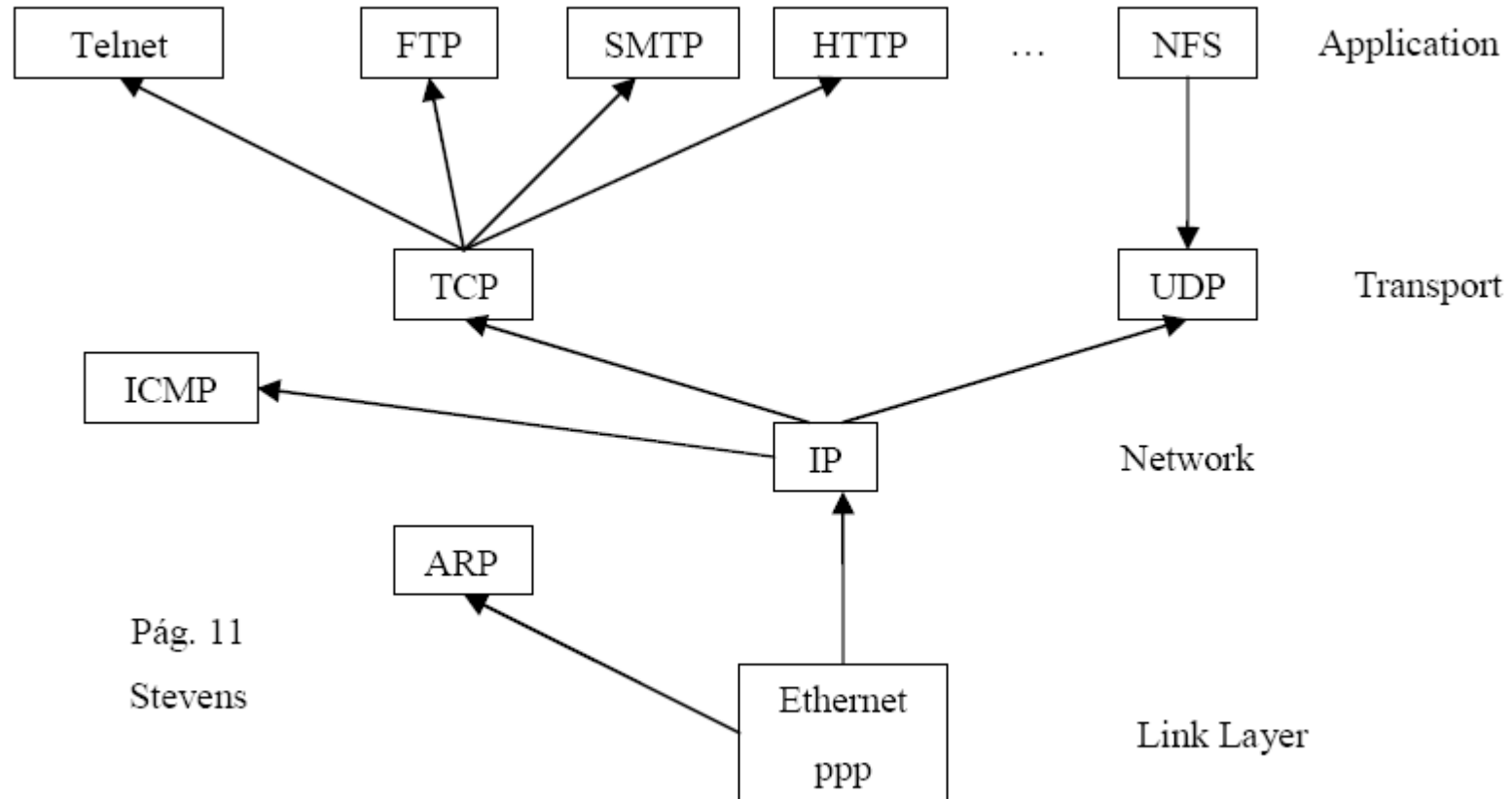
PORTAS (PORTOS)

- As aplicações “servidor” são identificados por números standard – portas. As portas 1-1023 são reservadas a serviços que pertencem a root
- Exemplos:

telnet	TCP	Porta 23
FTP	TCP	Porta 21
SMTP	TCP	Porta 25
TFTP	UDP	Porta 69
HTTP	TCP	Porta 80

- A combinação protocolo-porta identifica univocamente a aplicação no servidor (exemplo TCP porta 80 -> servidor web)
- A aplicação cliente é igualmente identificada por um par protocolo-porta – agora a porta assume um número aleatório (ainda não utilizado) acima de 1024

PROTOCOLOS FUNDAMENTAIS



Pág. 11
Stevens

REQUEST FOR COMMENTS (RFCs)

- RFCs – documentos oficiais que descrevem os protocolos

RFC 791 – protocolo IP

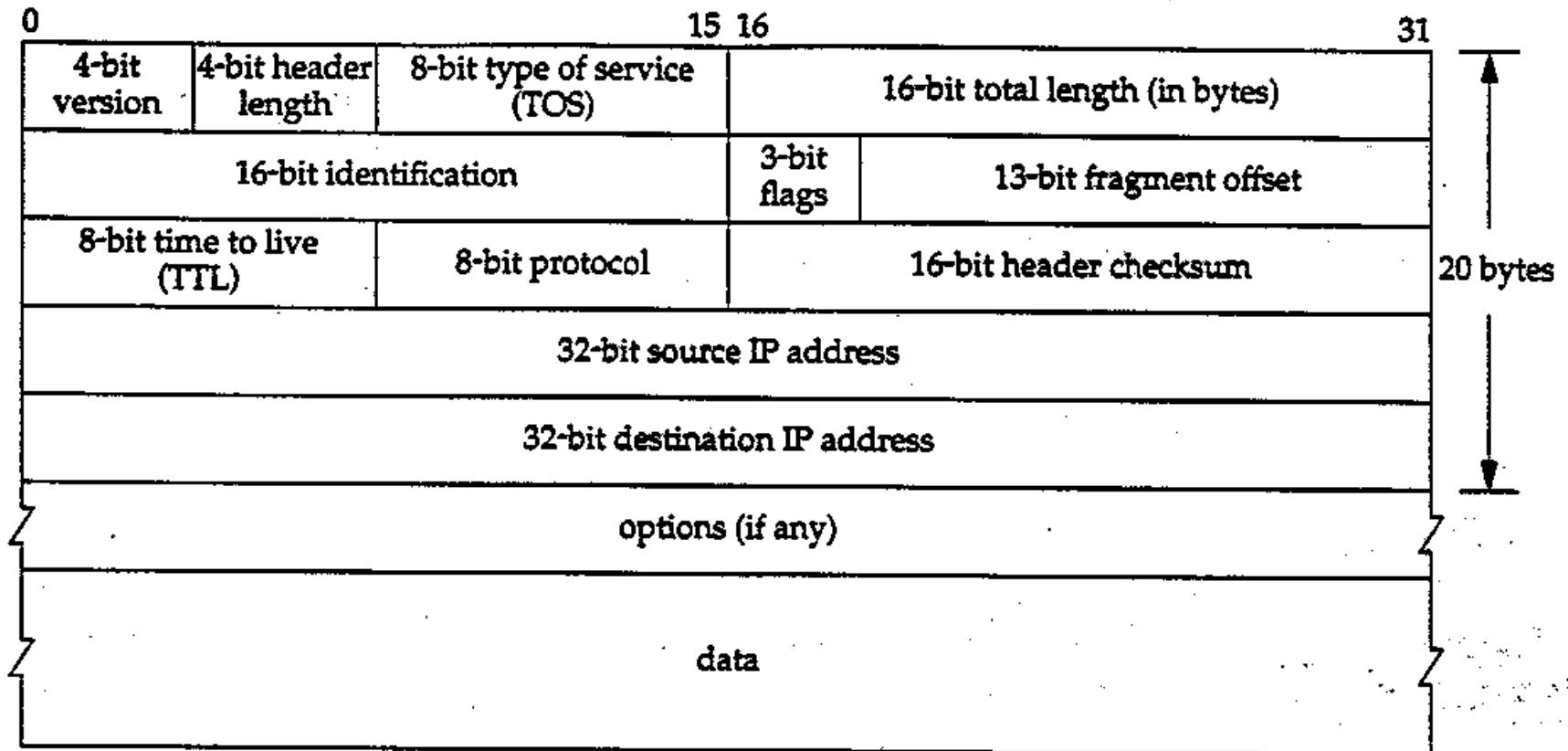
RFC 793 – protocolo TCP

RFC 768 – protocolo UDP

PROTOCOLO IP

- Responsável pelo encaminhamento dos dados
- **Simplex:**
 - Não estabelece uma ligação (**connectionless**)
 - Não seguro (**unreliable**)
- **Connectionless** - apenas fornece a informação para enviar dados de um ponto para outro, não verifica que a ligação é possível
- **Unreliable** - não garante a entrega dos dados

IP HEADER

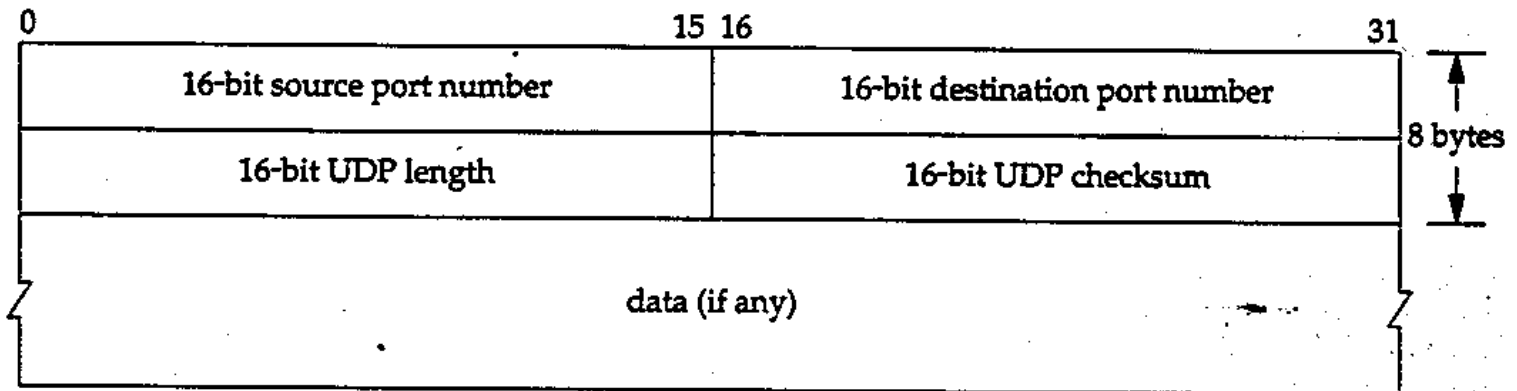


PROTOCOLO UDP

- **Objectivo:** transportar datagramas de uma porta (da camada de aplicações) para outra
- **não estabelece uma ligação**
- **não seguro:** não há garantia de entrega

UDP HEADER

UDP Header

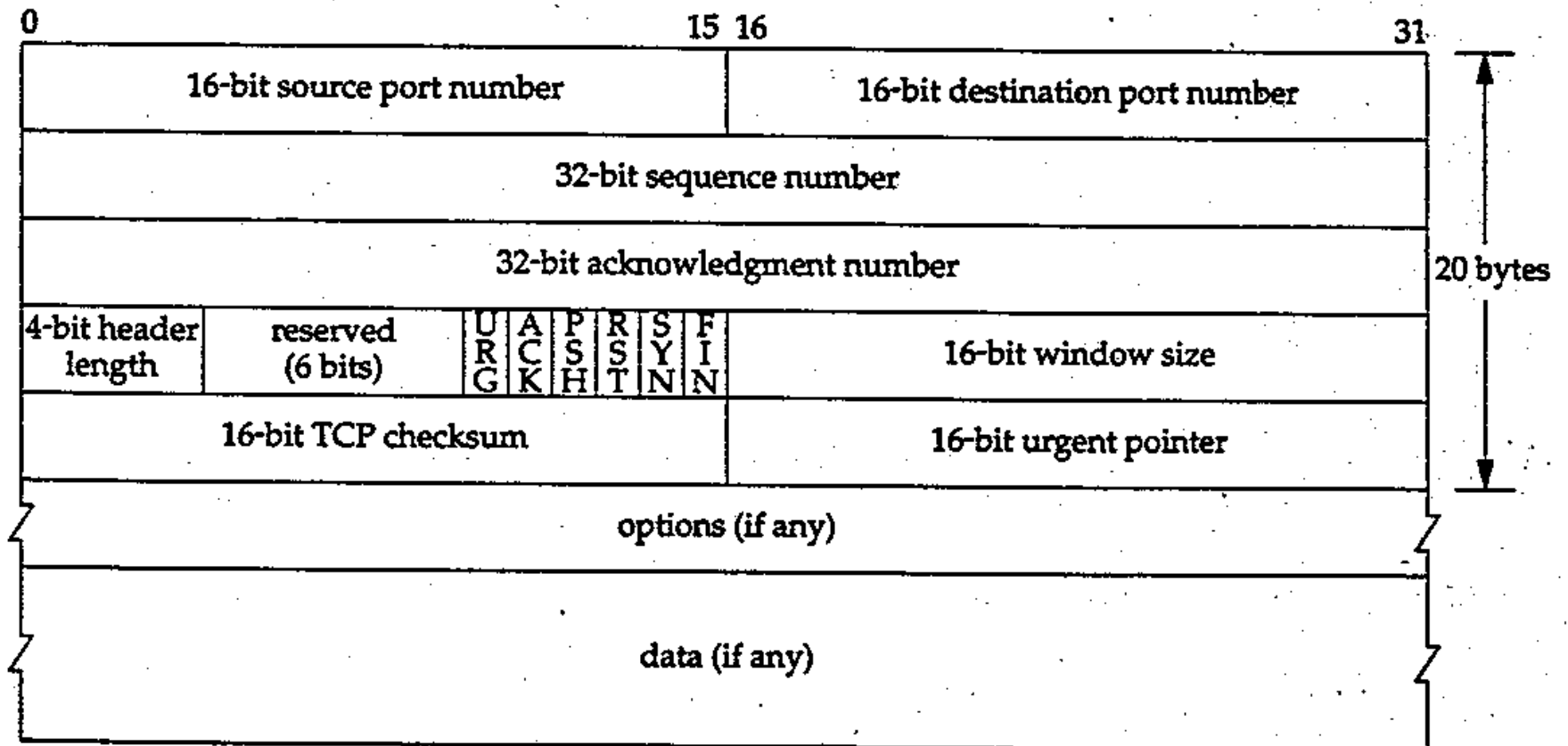


PROTOCOLO TCP

- **Objectivo:** transportar datagramas de uma porta (da camada de aplicações) para outra
- **estabelece uma ligação**
- **seguro:** há garantia de entrega

TCP HEADER

TCP Header



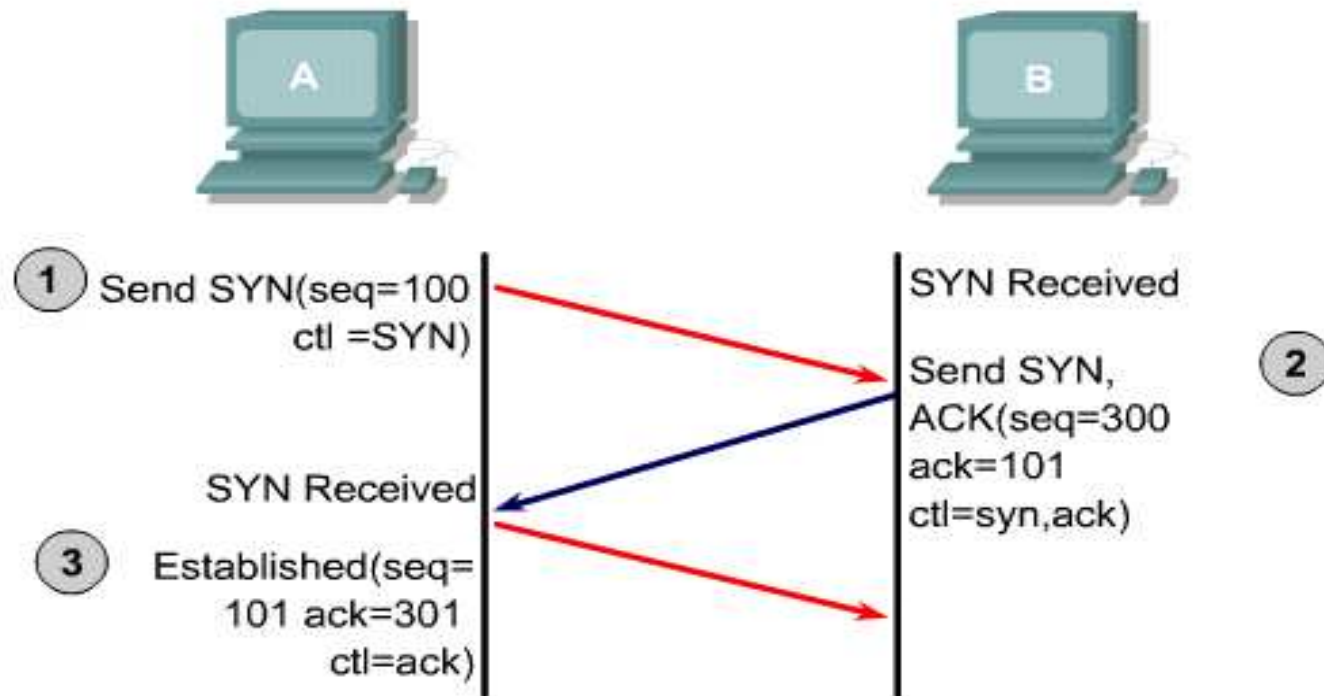
TCP Handshake

TCP Three-Way Handshake/Open Connection

FIGURES

1

2



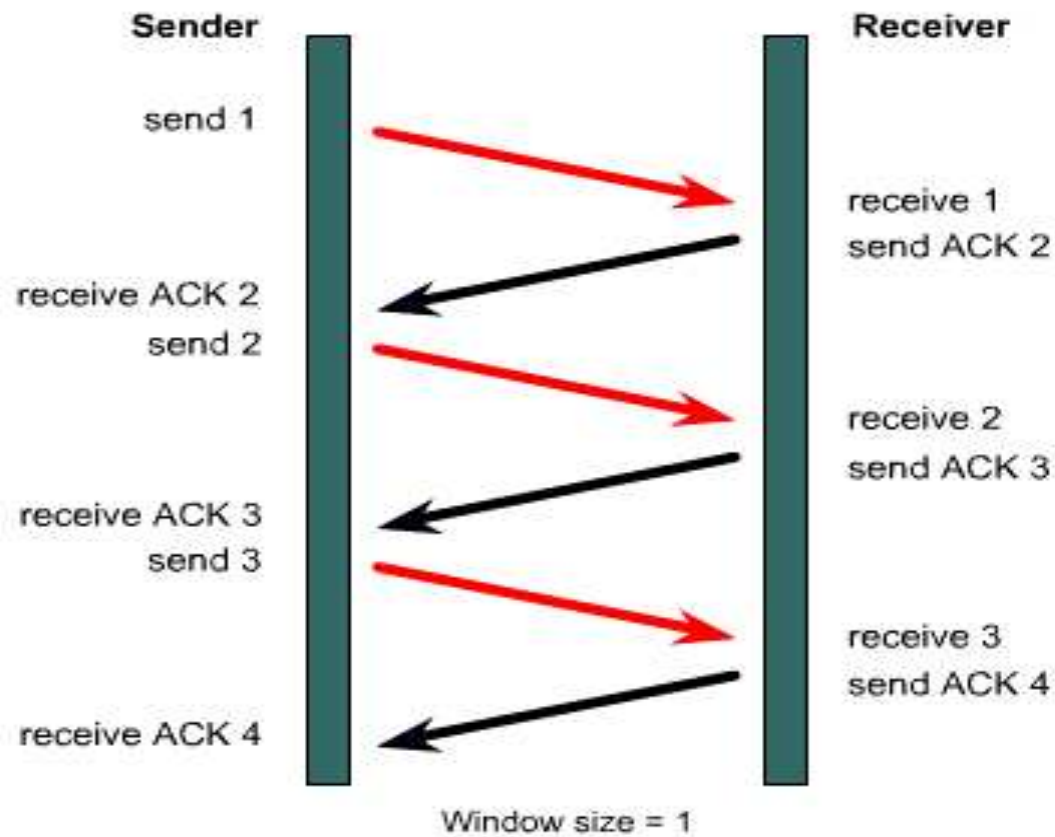
TCP Acknowledge (1)

TCP Basic Window

FIGURES

1

2



TCP Acknowledge (2)

TCP Sliding Window

FIGURES

1

2

