PROTOCOLOS FUNDAMENTAIS INTERNET

AULA 6 UDP - USER DATAGRAM PROTOCOL RFC 768

IP - FRAGMENTAÇÃO

UDP é um protocolo da camada de transporte com o único objectivo de transportar um datagrama de uma porta (da camada de aplicações) para outra.

NÃO é um protocolo seguro - não há nenhuma garantia que o datagrama chegue ao destino

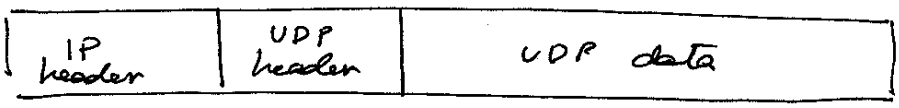
NÃO há controlo/fluxo de dados - o protocolo nas se preocupa se o destino está pronto a receber os dados

No entanto há varias aplicações que preferenciam simplicidade e velocidade: por exemplo envio de voz e imagem

Outro exemplo: aplicações que fazem broadcast e multicast (é um overhead mto grande garantir que os dados chegam a todos os destinos)

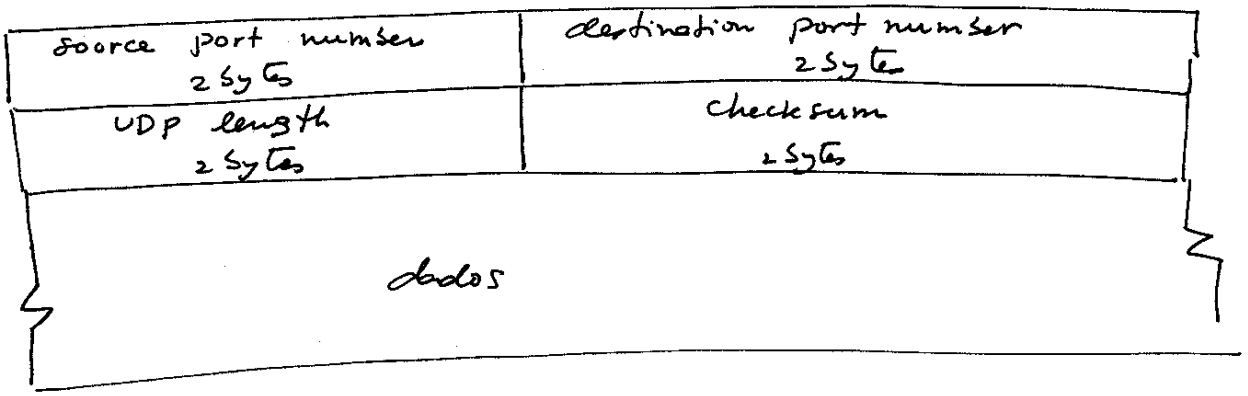
No entanto se uma aplicaçaõ necessitar de garantias na entrega dos dados, ela própria pode implementar os mecanismos necessários (confirmações, retransmissaõ, time outs etc)

UDP encapsulação



o datagrama UDP é encapsulado num datagrama IP

UDP header stevens pag 144



- source port number } identificam as aplicações (cliente/servidor)
- dest port number }
- UDP length — comprimento do UDP datagrama (header + dados)
  mínimo 8 Sytes (zero dados)

- checksum (aritmética complemento para 1) = 12 BYTE PSEUDO HEADER
  source IP address + dest IP address + protocolo + UDP length ← IP header campos seleccionados
  + source port number + destination port number + UDP length + dados

CHECKSUM — opcional MAS deve ser sempre implementado!

LAB:
sock -v -u -i -n4 carneiro discard
tcpdump host [illegible] carneiro

FRAGMENTAÇÃO IP

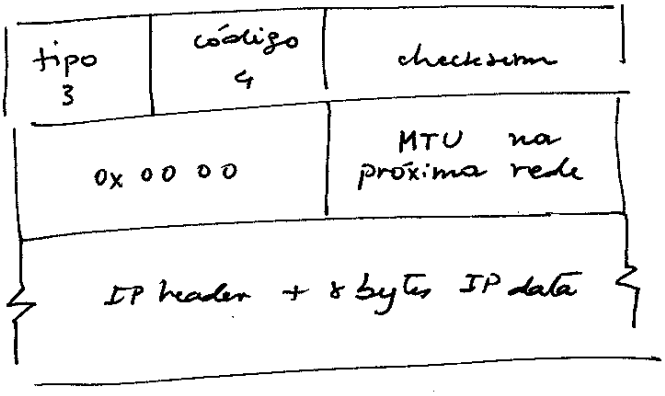
. A camada física (buffers nas interfaces) impõe um comprimento máximo para o datagrama

Se o datagrama enviado é maior que o MTU em qualquer parte do percurso, o IP datagrama tem que ser segmentado
Isto acontece, quase sempre, com datagramas UDP.

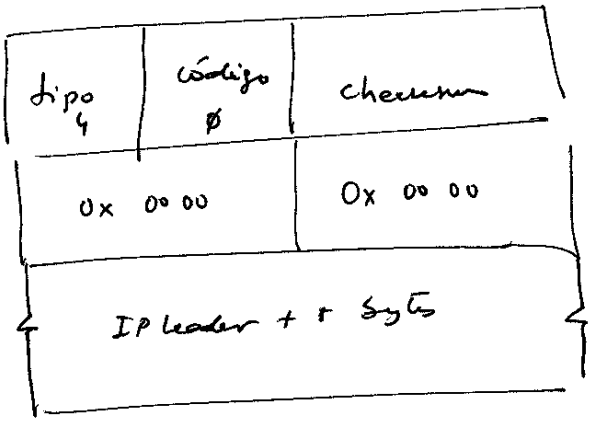
LAB ping australia (com flag dont fragment set
router 1 com MTU = 512)
-s 1024

traceroute . pmtu australia

ICMP fragmentation required error message



ICMP source quench error message



LAB sock -u -i -w -n 100 machine porta

LAB (criação de UDP socket)

servidor: sock -s -u -v -E -R 256 -r256 -P30 6666

cliente sock -u -v servidor 6666
111111 (terminar com newline)
2222222 (terminar com newline)

tcpdump host cliente
ethereal

O IP header tem campos apropriados para o efeito:

identificação (2 Syte) identifica univocamente o datagrama

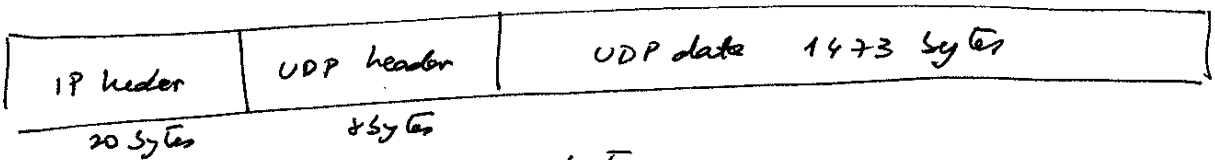
flag dont fragment (1 bit)

flag (more fragments) (1 bit)

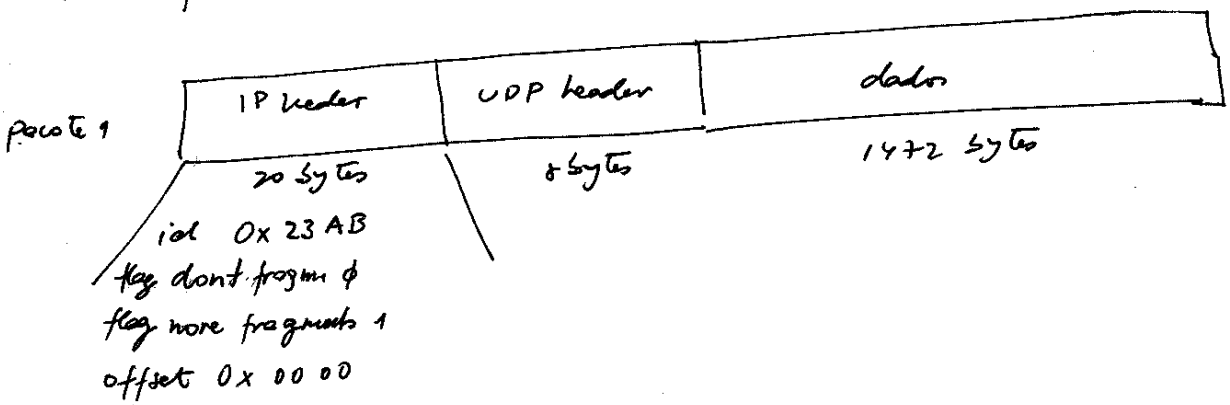
fragment offset (13 bit) - offset para este segmento a partir do início do IP datagrama (8 byte jumps)

IP total lenght alterado em acordo

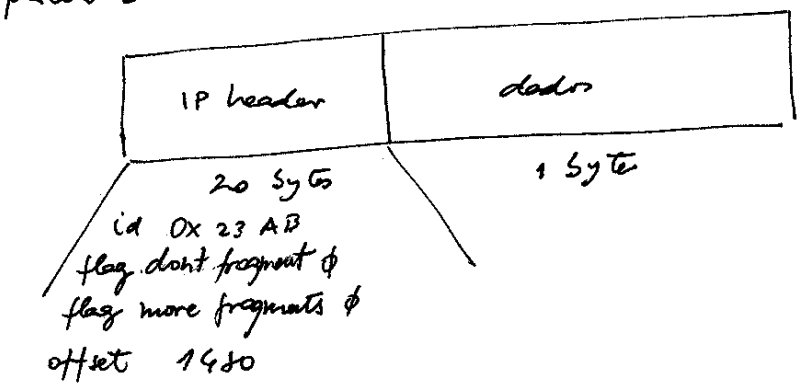
exemplo



meio físico ethernet : MTU 1500 bytes



pacote 2



LAB: sock -u -i -n1 -w 1471 carneiro discard
-w 1472
-w 1473
-w 1474