



**Universidade do Minho**

**LICENCIATURA EM CIÊNCIAS DA COMPUTAÇÃO**  
**SISTEMAS DE COMUNICAÇÕES E REDES**

**Ensaio Escrito**

**Nível de Ligação Lógica - Ethernet e Protocolo  
ARP; Redes Sem Fios (IEEE 802.11)**

**Grupo 28**

Davide Santos (A102938)

Edgar Araújo (A102946)

Pedro Augusto Camargo (A102504)

Outubro 2023

# Contents

1	Analise da fragmentação de pacotes IP	2
---	---------------------------------------	---

# 1 Análise da fragmentação de pacotes IP

Foi capturado o tráfego gerado pelo comando: `ping -s 6328 marco.uminho.pt`

No.	Time	Source	Destination	Protocol	Length	Info
17	6.896640340	172.26.57.176	193.136.9.240	ICMP	450	Echo (ping) request id=0x4d4c, seq=1/256, ttl=64 (reply in 22)
22	6.902078810	193.136.9.240	172.26.57.176	ICMP	1514	Echo (ping) reply id=0x4d4c, seq=1/256, ttl=61 (request in 17)
52	14.336102714	172.26.57.176	193.137.16.65	ICMP	128	Destination unreachable (Port unreachable)
53	14.336129153	172.26.57.176	193.137.16.65	ICMP	248	Destination unreachable (Port unreachable)
59	14.381506542	172.26.57.176	193.136.9.240	ICMP	450	Echo (ping) request id=0x4d4c, seq=2/512, ttl=64 (reply in 64)
64	14.391508599	193.136.9.240	172.26.57.176	ICMP	1514	Echo (ping) reply id=0x4d4c, seq=2/512, ttl=61 (request in 59)
94	19.421088600	172.26.57.176	193.136.9.240	ICMP	450	Echo (ping) request id=0x4d4c, seq=3/768, ttl=64 (reply in 99)
99	19.428025982	193.136.9.240	172.26.57.176	ICMP	1514	Echo (ping) reply id=0x4d4c, seq=3/768, ttl=61 (request in 94)

Figure 1: Ping

## a. Localize a primeira mensagem ICMP.

i) Porque é que houve necessidade de fragmentar o pacote inicial?

9	5.029830901	172.26.57.176	193.136.9.240	ICMP	150	Standard query response id=0x4d4c, seq=1/256, ttl=64 (reply in 18)
10	5.029833055	172.26.57.176	193.136.9.240	IPv4	1514	Fragmented IP protocol (proto=ICMP 1, off=1480, ID=015e) [Reassembled in #13]
11	5.029822411	172.26.57.176	193.136.9.240	IPv4	1514	Fragmented IP protocol (proto=ICMP 1, off=2960, ID=015e) [Reassembled in #13]
12	5.029828633	172.26.57.176	193.136.9.240	IPv4	1514	Fragmented IP protocol (proto=ICMP 1, off=4440, ID=015e) [Reassembled in #13]
13	5.029831157	172.26.57.176	193.136.9.240	ICMP	450	Echo (ping) request id=0x4d4c, seq=1/256, ttl=64 (reply in 18)

Figure 2: Fragment

A necessidade surge de o facto de o tamanho do pacote ser superior ao MTU da rede, ou seja, não cabe num único pacote. O MTU da rede é de 1500 bytes, e o tamanho do pacote é de 6328 bytes, logo é necessário fragmentar o pacote.

ii) Em que equipamento da rede ocorreu essa fragmentação?

A fragmentação ocorreu no computador que enviou o pacote.

## b. Imprima o primeiro fragmento do datagrama IP.

i) Que informação no cabeçalho indica que o datagrama foi fragmentado?

A opção MF, que indica que o pacote foi fragmentado, pode ser observada no Wireshark, dentro do campo Flags, no cabeçalho IP.

Internet Protocol Version 4, Src: 172.26.57.176, Dst: 193.136.9.240	
0100	.... = Version: 4
....	0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)	
Total Length: 1500	
Identification: 0x015e (350)	
001. .... = Flags: 0x1, More fragments	
...0 0000 0000 0000 = Fragment Offset: 0	
Time to Live: 64	
Protocol: ICMP (1)	

Figure 3: More Fragments

ii) Que informação no cabeçalho IP indica que se trata do primeiro fragmento?

O campo Fragment Offset indica que se trata do primeiro fragmento, uma vez que o seu valor é 0.

iii) Qual é o tamanho deste fragmento?

O tamanho do fragmento é de 1500 bytes, tal como indica o campo Total Length.

## c. Imprima o segundo fragmento do datagrama IP original. Que informação do cabeçalho IP indica que não se trata do primeiro fragmento? Há mais fragmentos? O que nos permite afirmar isso?

O campo Fragment Offset indica que não se trata do primeiro fragmento, uma vez que o seu valor é 1480. O campo MF indica que há mais fragmentos, uma vez que o seu valor é 1. Logo basta que:  $\text{Fragment Offset} \neq 0 \wedge \text{MF} == 1$  para sabermos que não se trata do primeiro fragmento.

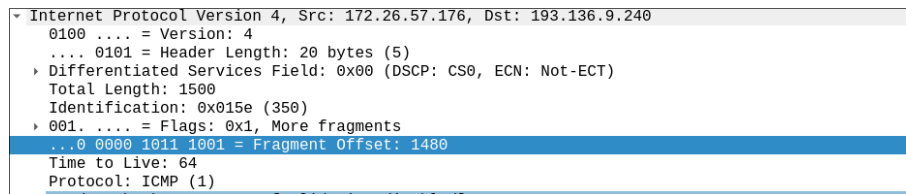


Figure 4: More Fragments Com Offset

d. Indique os campos que mudam no cabeçalho IP entre os diferentes fragmentos, e explique a forma como essa informação permite reconstruir o datagrama original.

Os campos que mudam são:

- Flag More Fragments (MF)
- Fragment Offset
- Total Length

Os campos que permitem reconstruir o datagrama original, são o Fragment Offset, e a flag MF, pois estes permitem saber a ordem exata dos fragmentos, de forma a reconstruir tal e qual como antes de ser fragmentado.

e. Como se deteta o último fragmento correspondente ao datagrama original? Estabeleça um filtro no Wireshark que permita listar o último fragmento do primeiro datagrama IP segmentado.

`ip.flags.mf == 0`

f. Identifique o equipamento onde o datagrama IP original é reconstruído a partir dos fragmentos. A reconstrução poderia ter ocorrido noutro equipamento diferente do identificado? Porquê?

O equipamento onde o datagrama IP original é reconstruído é o servidor de IP: 193.136.9.240, ou seja, o servidor de marco.uminho.pt. A reconstrução poderia ter ocorrido noutro equipamento desde que a MTU fosse superior a MTU da ligação anterior, ou seja maior que 1500 bytes, e que o equipamento tivesse a capacidade de reconstruir o datagrama original.

g. Por que razão apenas o primeiro fragmento de cada pacote é identificado como sendo um pacote ICMP?

Apenas o primeiro fragmento de cada pacote é identificado como sendo um pacote ICMP, pois o conceito de ICMP tem por base os cabeçalhos IP, e o conceito de fragmentação de datagramas IP é relativo ao cabeçalho IP.

h. Determine o valor máximo de SIZE sem que ocorra fragmentação do pacote? Justifique o valor obtido, relacionando-o com o MTU (Maximum Transmission Unit) da rede.

Após observar o output do comando ping no Linux: Reparei que houve um acréscimo de 28 bytes

```

$ ping -s 6328 marco.uminho.pt
PING marco.uminho.pt (193.136.9.240) 6328(6356) bytes of data.
6336 bytes from marco.uminho.pt (193.136.9.240): icmp_seq=1 ttl=61 time=8.

```

Figure 5: Comando Ping no Linux

na informação enviada, para acomodar todos os cabeçalhos essenciais na transmissão do pacote. Logo o valor máximo de SIZE sem que ocorra fragmentação do pacote é de 1472 bytes, pois  $1472 + 28 = 1500$  bytes, que é o MTU da rede.