

# Nível de Ligação Lógica: Redes Ethernet e Protocolo ARP

Filipa Rebelo and Joana Oliveira

Universidade do Minho, Departamento de Informática, 4710-057 Braga, Portugal  
email: {a90234,a87956}@alunos.uminho.pt

## Captura e análise de Tramas Ethernet

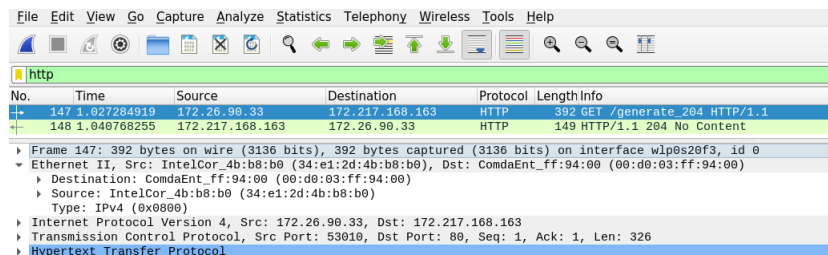
### Exercício 1

**Anote os endereços MAC de origem e de destino da trama capturada.**

Uma vez que este se encontra encriptado, o acesso ao endereço indicado foi recusado. No entanto, é ainda possível saber quais os endereços MAC de origem e de destino.

Endereço MAC origem: 34:e1:2d:4b:b8:b0

Endereço MAC destino: 00:d0:03:ff:94:00



**Fig. 1.** Endereços MAC

### Exercício 2

**Identifique a que sistemas se referem. Justifique.**

O endereço *Source* refere-se à interface da nossa máquina, ou seja de onde é enviada a trama. O endereço *Destination* refere-se à interface do router da rede local a que estamos ligados.

### Exercício 3

Qual o valor hexadecimal do campo Type da trama Ethernet? O que significa?

O valor hexadecimal do campo Type é 0x0800, que representa o protocolo IPv4.

### Exercício 4

Quantos bytes são usados no encapsulamento protocolar, i.e. desde o início da trama até ao início dos dados do nível aplicacional (Application Data Protocol: http-over-tls)? Calcule e indique, em percentagem, a sobrecarga (overhead) introduzida pela pilha protocolar.

Observando a Figura 2 verifica-se que temos 66 bytes até ao início dos dados do nível aplicacional.

Assim, tendo sido já usados 392 bytes até ao momento, a sobrecarga é de:

$$\frac{66}{392} * 100 = 16.84\%$$

No.	Time	Source	Destination	Protocol	Length	Info
147	392 bytes on wire (3136 bits), 392 bytes captured (3136 bits) on interface wlp0s20f3, id 0					
▼	Ethernet II, Src: IntelCor_4b:b8:b0 (34:e1:2d:4b:b8:b0), Dst: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)					
▶	Destination: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)					
▶	Source: IntelCor_4b:b8:b0 (34:e1:2d:4b:b8:b0)					
▶	Type: IPv4 (0x0800)					
▶	Internet Protocol Version 4, Src: 172.26.90.33, Dst: 172.217.168.163					
▶	Transmission Control Protocol, Src Port: 53810, Dst Port: 80, Seq: 1, Ack: 1, Len: 326					
▼	Hypertext Transfer Protocol					
▼	GET /generate_204 HTTP/1.1					
▶	[Expert Info (Chat/Sequence): GET /generate_204 HTTP/1.1]					
▶	Request Method: GET					
▶	Request URI: /generate_204					
▶	Request Version: HTTP/1.1					
▶	Host: www.gstatic.com					
▶	Connection: keep-alive					
▶	Pragma: no-cache					
▶	Cache-Control: no-cache					
▶	User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/100.0.4896.127 Safari/537.36					
▶	Accept-Encoding: gzip, deflate					
▶	Accept-Language: pt-BR,pt;q=0.9,en-US;q=0.8,en;q=0.7					
▶	[Full request URI: http://www.gstatic.com/generate_204]					
▶	[HTTP request 1/1]					
▶	[Response in frame: 148]					
0000	00 d0 03 ff 94 00 34 e1	2d 4b b8 b0 08 00 45 00	.....4.-K....E-			
0010	01 7a ed 23 40 00 40 06	f0 a1 ac 1a 5a 21 ac d9	.z#@@...Z!...			
0020	a8 a3 cf 12 00 50 74 47	17 b7 98 3b bc 47 80 18	....PtG...;G--			
0030	01 f6 5d 25 00 00 01 01	08 0a f0 7d 1f 88 3c 14	..]%. ....<..			
0040	b2 75 57 45 20 2f 67	65 6e 65 72 61 74 65 5f	u57/generate_			
0050	32 30 34 20 48 54 50	2f 31 2e 31 0d 0a 48 6f	204 HTTP /1.1..Ho			
0060	73 74 3a 20 77 77 77 2e	67 73 74 61 74 69 63 2e	st: www.gstatic.			
0070	63 6f 6d 0d 0a 43 6f 6e	6e 65 63 74 69 6f 6e 3a	com..Con nectio:			
0080	20 6b 65 65 70 2d 61 6c	69 76 65 0d 0a 50 72 61	keep-al ive..Pra			
0090	67 6d 61 3a 20 6e 6f 2d	63 61 63 68 65 0d 0a 43	gma: no- cache..C			
00a0	61 63 68 65 2d 43 6f 6e	74 72 6f 6c 3a 20 6e 6f	ache-Con trol: no			

Fig. 2. Trama que contém o pedido HTTP

### Exercício 5

Qual é o endereço Ethernet da fonte? A que sistema de rede corresponde? Justifique.

O endereço Ethernet da fonte encontra-se no campo *Source* com o valor 00:d0:03:ff:94:00. Este corresponde ao *gateway* da rede local, uma vez que o servidor não se encontra na rede local e apenas conhecemos os IPs das redes locais e do *gateway*.

No.	Time	Source	Destination	Protocol	Length	Info
147	1.027284919	172.26.90.33	172.217.168.163	HTTP	392	GET /generate_204 HTTP/1.1
148	1.049768255	172.217.168.163	172.26.90.33	HTTP	149	HTTP/1.1 204 No Content

▶	Frame 148: 149 bytes on wire (1192 bits), 149 bytes captured (1192 bits) on interface wlp8s20f3, id 0
▼	Ethernet II, Src: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00), Dst: IntelCor_4b:b8:b0 (34:e1:2d:4b:b8:b0)
▶	Destination: IntelCor_4b:b8:b0 (34:e1:2d:4b:b8:b0)
▶	Source: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
▶	Type: IPv4 (0x0800)
▶	Internet Protocol Version 4, Src: 172.217.168.163, Dst: 172.26.90.33
▶	Transmission Control Protocol, Src Port: 80, Dst Port: 53010, Seq: 1, Ack: 327, Len: 83
▶	Hypertext Transfer Protocol

0000	34 e1 2d 4b b8 b0 00 d0 03 ff 94 00 08 00 45 00	4--K----	E-
0010	00 87 57 fd 00 00 76 06 90 bb ac d9 a8 a3 ac 1a	--W--V-	-----
0020	5a 21 00 50 cf 12 98 3b bc 47 74 47 18 fd 80 18	Z!P---	-GtG----
0030	01 05 64 46 00 00 01 01 08 0a 3c 14 b4 84 f0 7d	..dF----	-<-----}
0040	1f 88 48 54 54 50 2f 31 2e 31 20 32 30 34 20 4e	..HTTP/1	.1 204 N
0050	6f 20 43 6f 6e 74 65 6e 74 8d 0a 43 6f 6e 74 65	o Conten t	: Conte
0060	6e 74 2d 4c 65 6e 67 74 68 3a 20 39 0d 0a 00 01	nt-Lengt h: 0	.D
0070	74 65 4a 20 40 72 69 2c 20 32 32 20 41 70 72 20	t: 11, 22 Apr	
0080	32 30 32 32 20 31 33 3a 35 39 3a 33 33 20 47 4d	2022, 13: 59:33 GM	
0090	54 0d 0a 0d 0a	T...	

Fig. 3. Trama que contém a resposta HTTP

### Exercício 6

Qual é o endereço MAC do destino? A que sistema corresponde?

O endereço MAC do destino, presente no campo *Destination* da figura anterior, é 34:e1:2d:4b:b8:b0 e corresponde à interface ethernet da nossa máquina.

### Exercício 7

Atendendo ao conceito de desencapsulamento protocolar, identifique os vários protocolos contidos na trama recebida.

Os protocolos contidos na trama são IPv4, Ethernet, TCP e HTML.

## Protocolo ARP

### Exercício 8

Observe o conteúdo da tabela ARP. Diga o que significa cada uma das colunas.

A coluna *Address* contém os endereços IP ou nome dos hosts e neste caso apenas contém o *gateway* da rede local. A coluna *HWtype* permite indicar que as

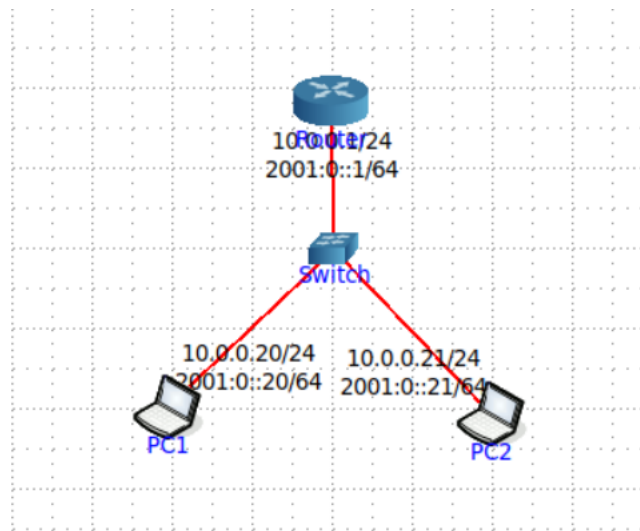
conexões são conexões Ethernet. A coluna *HWaddress* contém os endereços MAC de cada dispositivo. A coluna *Flags* permite saber, por exemplo, se um endereço MAC foi aprendido ou introduzido manualmente. Neste caso, como o seu valor é C é possível saber que este registo foi aprendido dinamicamente através do protocolo ARP. A coluna *Mask* corresponde à máscara de sub-rede. Por fim, a coluna *Iface* representa a interface de rede, que neste caso corresponde a wlo1.

```
~$ arp
```

Address	Hwtype	Hwaddress	Flags	Mask	Iface
_gateway	ether	00:d0:03:ff:94:00	C		wlo1

**Fig. 4.** Tabela ARP

Para a realização das perguntas seguintes, e devido à impossibilidade de fazer ping para outros hosts da sala de aula, foi-nos proposto que fizéssemos uma simulação no Core através do uso de um router, um switch e dois portáteis. A topologia criada pode ser observada na figura seguinte.



**Fig. 5.** Rede auxiliar

```

vcmd
root@PC1:/tmp/pycore.36071/PC1.conf# ping 10.0.0.21
PING 10.0.0.21 (10.0.0.21) 56(84) bytes of data.
64 bytes from 10.0.0.21: icmp_seq=1 ttl=64 time=0.103 ms
64 bytes from 10.0.0.21: icmp_seq=2 ttl=64 time=0.139 ms
64 bytes from 10.0.0.21: icmp_seq=3 ttl=64 time=0.152 ms
^C
--- 10.0.0.21 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2032ms
rtt min/avg/max/mdev = 0.103/0.131/0.152/0.020 ms
root@PC1:/tmp/pycore.36071/PC1.conf#

```

Fig. 6. Comando ping

### Exercício 9

Qual é o valor hexadecimal dos endereços origem e destino na trama Ethernet que contém a mensagem com o pedido ARP (ARP Request)? Como interpreta e justifica o endereço destino usado?

Através da figura seguinte é possível observar que o endereço origem é 00:00:00:aa:00:01 e o endereço destino é ff:ff:ff:ff:ff:ff (broadcast).

O endereço destino é o de broadcast, o que permite enviar a mensagem para todos os hosts da rede. O host que tenha o IP destino correspondente irá então responder com o seu endereço MAC e este será adicionado à tabela ARP.

No.	Time	Source	Destination	Protocol	Length	Info
16	12.863451786	00:00:00_aa:00:01	Broadcast	ARP	42	Who has 10.0.0.21? Tell 10.0.0.20
17	12.863521593	00:00:00_aa:00:02	00:00:00_aa:00:01	ARP	42	10.0.0.21 is at 00:00:00:aa:00:02
29	17.938431307	00:00:00_aa:00:02	00:00:00_aa:00:01	ARP	42	Who has 10.0.0.20? Tell 10.0.0.21
31	17.938476322	00:00:00_aa:00:01	00:00:00_aa:00:02	ARP	42	10.0.0.20 is at 00:00:00:aa:00:01

▶ Frame 16: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface veth3.0.b9, id 0  
 ▼ Ethernet II, Src: 00:00:00\_aa:00:01 (00:00:00:aa:00:01), Dst: Broadcast (ff:ff:ff:ff:ff:ff)  
   ▶ Destination: Broadcast (ff:ff:ff:ff:ff:ff)  
   ▶ Source: 00:00:00\_aa:00:01 (00:00:00:aa:00:01)  
     Type: ARP (0x0806)  
   ▼ Address Resolution Protocol (request)  
     Hardware type: Ethernet (1)  
     Protocol type: IPv4 (0x0800)  
     Hardware size: 6  
     Protocol size: 4  
     Opcode: request (1)  
     Sender MAC address: 00:00:00\_aa:00:01 (00:00:00:aa:00:01)  
     Sender IP address: 10.0.0.20  
     Target MAC address: 00:00:00\_00:00:00 (00:00:00:00:00:00)  
     Target IP address: 10.0.0.21

Fig. 7. Pedido ARP

### Exercício 10

Qual o valor hexadecimal do campo tipo da trama Ethernet? O que indica?

O valor hexadecimal é 0x0806 e indica que se trata de uma mensagem ARP.

### Exercício 11

Como pode confirmar que se trata efetivamente de um pedido ARP? Identifique que tipo de endereços estão contidos na mensagem ARP? Que conclui?

Tal como visto na alínea anterior esta trata-se de uma mensagem ARP. Analisando o campo *Opcode* vemos que este contém valor 1, juntamente com a palavra *request*, o que nos permite confirmar que se trata de um pedido ARP. Os tipos de endereços contidos na mensagem ARP são endereços IP e MAC, tanto de origem como de destino.

### Exercício 12

Explicita o tipo de pedido ou pergunta é feita pelo host de origem.

O pedido feito pelo host de origem é "Who has 10.0.0.21? Tell 10.0.0.20". Este pedido é então enviado a todos os hosts da rede com o objetivo de saber o endereço MAC do endereço IP especificado, que deverá ser depois enviado para o host origem.

### Exercício 13

No.	Time	Source	Destination	Protocol	Length	Info
16	12.863451786	00:00:00_aa:00:01	Broadcast	ARP	42	Who has 10.0.0.21? Tell 10.0.0.20
17	12.863421593	00:00:00_aa:00:02	00:00:00_aa:00:01	ARP	42	10.0.0.21 is at 00:00:00_aa:00:02
29	17.938431397	00:00:00_aa:00:02	00:00:00_aa:00:01	ARP	42	Who has 10.0.0.20? Tell 10.0.0.21
31	17.938476322	00:00:00_aa:00:01	00:00:00_aa:00:02	ARP	42	10.0.0.20 is at 00:00:00_aa:00:01

```

Frame 17: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface veth3.0.b9, id 0
  Ethernet II, Src: 00:00:00_aa:00:02 (00:00:00_aa:00:02), Dst: 00:00:00_aa:00:01 (00:00:00_aa:00:01)
    Destination: 00:00:00_aa:00:01 (00:00:00_aa:00:01)
    Source: 00:00:00_aa:00:02 (00:00:00_aa:00:02)
    Type: ARP (0x0806)
  Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: 00:00:00_aa:00:02 (00:00:00_aa:00:02)
    Sender IP address: 10.0.0.21
    Target MAC address: 00:00:00_aa:00:01 (00:00:00_aa:00:01)
    Target IP address: 10.0.0.20
  
```

Fig. 8. Resposta ARP

Alínea 13a) Qual o valor do campo ARP opcode? O que especifica?

O valor do campo opcode é 2, o que significa que se trata de de uma resposta (ARP Reply) a um pedido anterior.

Alínea 13b) Em que campo da mensagem ARP está a resposta ao pedido ARP?

A resposta encontra-se no campo *Sender MAC adress*.

#### Exercício 14

Na situação em que efetua um ping a outro host, assuma que este está diretamente ligado ao mesmo router, mas noutra subrede, e que todas as tabelas ARP se encontram inicialmente vazias. Esboce um diagrama em que indique claramente, e de forma cronológica, todas as mensagens ARP e ICMP trocadas, até à recepção da resposta ICMP do host destino.

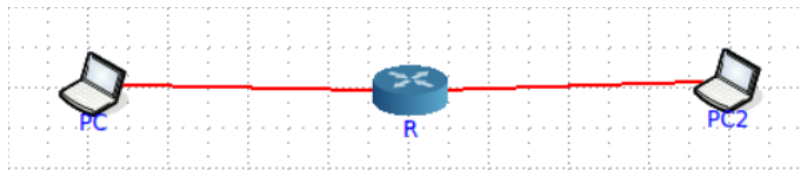


Fig. 9. Exemplo de rede

			IP Origem: IP PC	ICMP
			IP Destino: IP PC2	
AC Destino: ?	MAC Origem: MAC PC	Tipo: 0x800 (IPv4)		Payload
			MAC Origem: MAC PC	MAC Destino: 00:00:00:00:00:00
			IP Origem: IP PC	IP Destino: IP R
AC Destino: ff:ff:ff:ff:ff:ff	MAC Origem: MAC PC	Tipo: 0x806 (ARP)		Payload
			MAC Origem: MAC R	MAC Destino: MAC PC
			IP Origem: IP R	IP Destino: IP PC
AC Destino: MAC PC	MAC Origem: MAC R	Tipo: 0x806 (ARP)		Payload
				ICMP
			IP Origem: IP PC	
			IP Destino: IP PC2	
AC Destino: MAC R	MAC Origem: MAC PC	Tipo: 0x800 (IPv4)		Payload

Fig. 10. Diagrama das mensagens trocadas entre PC e Router

				ICMP
			IP Origem: IP PC	
			IP Destino: IP PC2	
MAC Destino: ?	MAC Origem: MAC R	Tipo: 0x800 (IPv4)		Payload
			MAC Origem: MAC R	MAC Destino: 00:00:00:00:00:00
			IP Origem: IP R	IP Destino: IP PC2
MAC Destino: ff:ff:ff:ff:ff:ff	MAC Origem: MAC R	Tipo: 0x806 (ARP)		Payload
			MAC Origem: MAC PC2	MAC Destino: MAC R
			IP Origem: IP PC2	IP Destino: IP R
MAC Destino: MAC R	MAC Origem: MAC PC2	Tipo: 0x806 (ARP)		Payload
				ICMP
			IP Origem: IP PC	
			IP Destino: IP PC2	
MAC Destino: MAC PC2	MAC Origem: MAC R	Tipo: 0x800 (IPv4)		Payload

Fig. 11. Diagrama das mensagens trocadas entre Router e PC2

## Domínios de colisão

### Exercício 15

Através da opção `tcpdump` verifique e compare como flui o tráfego nas diversas interfaces do dispositivo de interligação no departamento A (LAN partilhada) e no departamento B (LAN comutada) quando se gera tráfego intra-departamento (por exemplo, fazendo ping IPaddr da Bela para Monstro, da Jasmine para o Alladin, etc.) Que conclui?

Na topologia representada na Figura 12 foi substituído o switch existente no departamento A por um hub e adicionado um novo portátil, tanto no departamento A como no B, de modo a permitir correr o comando `tcpdump` e ver o como flui o tráfego em ambos os departamentos.

No Departamento A, departamento onde ocorreu a substituição do switch para hub, ao ser realizado um ping de Bela para Monstro verifica-se que todos os hosts da interface recebem os pacotes enviados pela Bela. Isto é comprovado pelo facto de o PC1 capturar tramas tais como *echo request* e *echo reply* enviadas entre os dois outros portáteis.

No departamento B, departamento em que o switch ainda é utilizado, ao efetuar um ping de Jasmine para Alladin verifica-se que apenas o Alladin recebe os pacotes enviados por Jasmine e apenas Jasmine recebe a resposta enviada por Aladin. Isto é confirmado pelo facto de o PC2 não conseguir capturar quaisquer tramas enviadas entre os dois portáteis, sendo que as tramas existentes na Figura 16 nada têm a ver com o ping efetuado.

Assim conclui-se que, para diminuir as colisões, os switches são a melhor opção, uma vez que este apenas envia os pacotes para o host indicado, ao contrário do hub envia para todos os dispositivos a que esteja ligado.



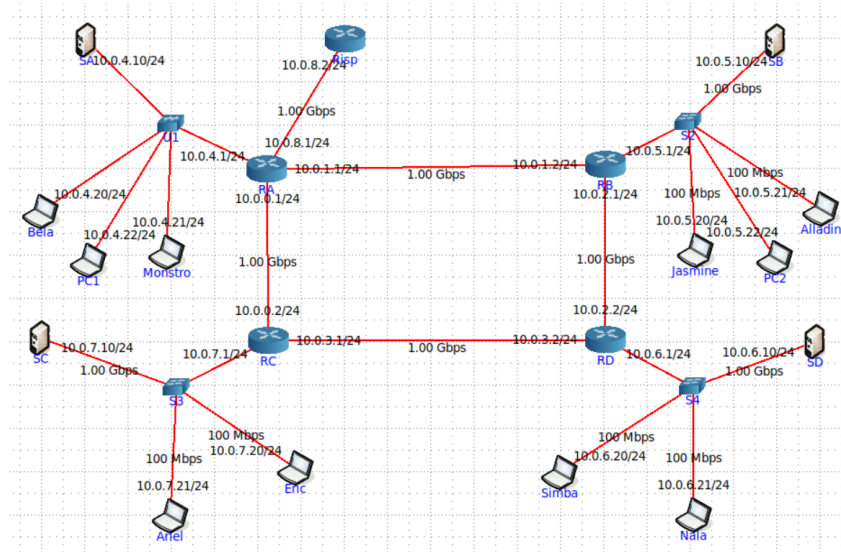


Fig. 12. Topologia Core

```

vcmd
root@Bela:/tmp/pycore.45177/Bela.conf# ping 10.0.4.21
PING 10.0.4.21 (10.0.4.21) 56(84) bytes of data:
64 bytes from 10.0.4.21: icmp_seq=1 ttl=64 time=0.147 ms
64 bytes from 10.0.4.21: icmp_seq=2 ttl=64 time=0.160 ms
64 bytes from 10.0.4.21: icmp_seq=3 ttl=64 time=0.167 ms
64 bytes from 10.0.4.21: icmp_seq=4 ttl=64 time=0.065 ms
^C
--- 10.0.4.21 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3026ms
rtt min/avg/max/mdev = 0.065/0.134/0.167/0.040 ms
root@Bela:/tmp/pycore.45177/Bela.conf#

```

Fig. 13. Ping de Bela para Monstro

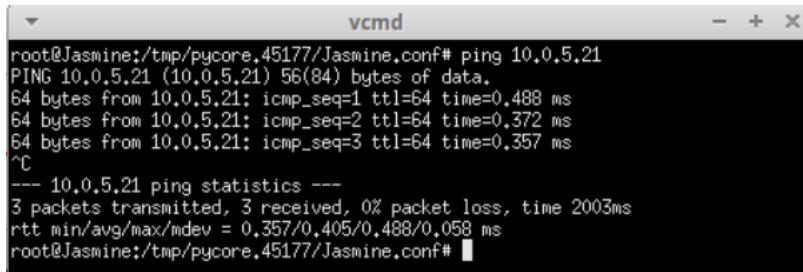
```

vcmd
root@PC1:/tmp/pycore.45177/PC1.conf# tcpdump
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
^C22:30:25.910891 IP 10.0.4.1 > 224.0.0.5: OSPFv2, Hello, length 44
22:30:26.306796 IP 10.0.4.20 > 10.0.4.21: ICMP echo request, id 55, seq 1, length 64
22:30:26.306869 IP 10.0.4.21 > 10.0.4.20: ICMP echo reply, id 55, seq 1, length 64
22:30:27.308676 IP 10.0.4.20 > 10.0.4.21: ICMP echo request, id 55, seq 2, length 64
22:30:27.308743 IP 10.0.4.21 > 10.0.4.20: ICMP echo reply, id 55, seq 2, length 64
22:30:27.911666 IP 10.0.4.1 > 224.0.0.5: OSPFv2, Hello, length 44
22:30:28.309143 IP 10.0.4.20 > 10.0.4.21: ICMP echo request, id 55, seq 3, length 64
22:30:28.309217 IP 10.0.4.21 > 10.0.4.20: ICMP echo reply, id 55, seq 3, length 64
22:30:29.332975 IP 10.0.4.20 > 10.0.4.21: ICMP echo request, id 55, seq 4, length 64
22:30:29.333003 IP 10.0.4.21 > 10.0.4.20: ICMP echo reply, id 55, seq 4, length 64
22:30:29.653066 IP6 fe80::200:ff:feaa:17 > ff02::5: OSPFv3, Hello, length 36
22:30:29.912460 IP 10.0.4.1 > 224.0.0.5: OSPFv2, Hello, length 44
22:30:31.541019 ARP, Request who-has 10.0.4.20 tell 10.0.4.21, length 28
22:30:31.541031 ARP, Request who-has 10.0.4.21 tell 10.0.4.20, length 28
22:30:31.541103 ARP, Reply 10.0.4.20 is-at 00:00:00:aa:00:18 (oui Ethernet), length 28
22:30:31.541109 ARP, Reply 10.0.4.21 is-at 00:00:00:aa:00:19 (oui Ethernet), length 28
22:30:31.912642 IP 10.0.4.1 > 224.0.0.5: OSPFv2, Hello, length 44
22:30:32.308814 IP6 fe80::a859:8ff:fee7:c78d > ip6-allrouters: ICMP6, router solicitation, length 16

18 packets captured
18 packets received by filter
0 packets dropped by kernel
root@PC1:/tmp/pycore.45177/PC1.conf#

```

Fig. 14. Tcpdump no PC1

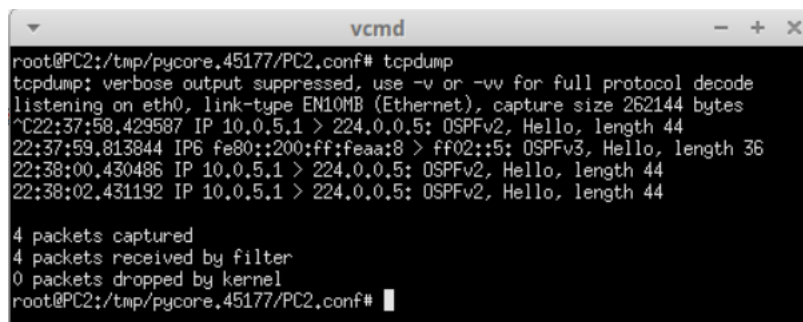


```

vcmd
root@Jasmine:/tmp/pycore.45177/Jasmine.conf# ping 10.0.5.21
PING 10.0.5.21 (10.0.5.21) 56(84) bytes of data:
64 bytes from 10.0.5.21: icmp_seq=1 ttl=64 time=0.488 ms
64 bytes from 10.0.5.21: icmp_seq=2 ttl=64 time=0.372 ms
64 bytes from 10.0.5.21: icmp_seq=3 ttl=64 time=0.357 ms
^C
--- 10.0.5.21 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.357/0.405/0.488/0.058 ms
root@Jasmine:/tmp/pycore.45177/Jasmine.conf#

```

Fig. 15. Ping de Jasmine para Alladin



```

vcmd
root@PC2:/tmp/pycore.45177/PC2.conf# tcpdump
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
^C22:37:58.429587 IP 10.0.5.1 > 224.0.0.5: OSPFv2, Hello, length 44
22:37:59.813844 IP6 fe80::200:ff:feaa:8 > ff02::5: OSPFv3, Hello, length 36
22:38:00.430486 IP 10.0.5.1 > 224.0.0.5: OSPFv2, Hello, length 44
22:38:02.431192 IP 10.0.5.1 > 224.0.0.5: OSPFv2, Hello, length 44

4 packets captured
4 packets received by filter
0 packets dropped by kernel
root@PC2:/tmp/pycore.45177/PC2.conf#

```

Fig. 16. Tcpdump no PC2

### Exercício 16

Construa manualmente a tabela de comutação do switch do Departamento B, atribuindo números de porta à sua escolha

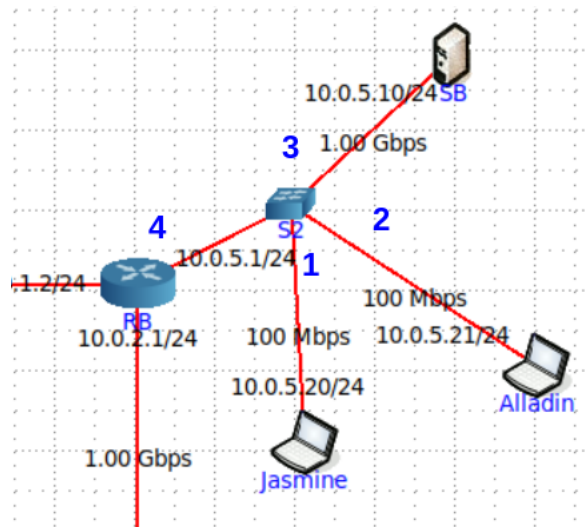


Fig. 17. Numeração das interfaces no Departamento B

MAC Address	Interface	TTL
00:00:00:aa:00:0e	1	64
00:00:00:aa:00:0f	2	64
00:00:00:aa:00:09	3	64
00:00:00:aa:00:08	4	64

Fig. 18. Tabela de comutação

## Conclusão

Através da realização deste trabalho foi possível consolidar os conhecimentos obtidos nas aulas teóricas sobre o **Nível de Ligação Lógica**, nomeadamente os conceitos de **Ethernet** e **Protocolo ARP**.

Para tal foi então utilizado o **Wireshark**, que nos permitiu capturar e visualizar o conteúdo dos diversos pacotes e o **CORE** que nos permitiu verificar a diferença entre switches e hubs e o seu impacto na deteção e correção de erros.

Foi ainda abordado o protocolo ARP, nomeadamente as suas tabelas e o conteúdo das mensagens enviadas através do mesmo.

Por fim, acreditamos ter cumprido o objetivo deste trabalho e obtido conhecimento importante em conceitos como captura e análise de tramas ethernet, protocolo ARP e domínios de colisão.