
TP3: – Nível de Ligação Lógica: Redes Ethernet e Protocolo ARP

TRABALHO REALIZADO POR:

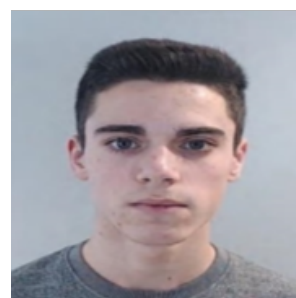
BEATRIZ RIBEIRO MONTEIRO
PEDRO PEREIRA SOUSA
TELMO JOSÉ PEREIRA MACIEL



A95437
Beatriz Monteiro



A95826
Pedro Sousa



A96569
Telmo Maciel

Índice

| | | |
|----------|--|----------|
| 1 | Captura e análise de Tramas <i>Ethernet</i> | 1 |
| 1.1 | Pergunta 1 | 1 |
| 1.2 | Pergunta 2 | 1 |
| 1.3 | Pergunta 3 | 1 |
| 1.4 | Pergunta 4 | 1 |
| 1.5 | Pergunta 5 | 2 |
| 1.6 | Pergunta 6 | 2 |
| 1.7 | Pergunta 7 | 3 |
| 2 | Protocolo ARP | 3 |
| 2.1 | Pergunta 8 | 3 |
| 2.2 | Pergunta 9 | 3 |
| 2.3 | Pergunta 10 | 4 |
| 2.4 | Pergunta 11 | 4 |
| 2.5 | Pergunta 12 | 4 |
| 2.6 | Pergunta 13 | 4 |
| 2.6.1 | Alínea a | 4 |
| 2.6.2 | Alínea b | 4 |
| 2.7 | Pergunta 14 | 5 |
| 3 | Domínios de colisão | 6 |
| 3.1 | Pergunta 15 | 6 |
| 3.2 | Pergunta 16 | 7 |
| 4 | Conclusão | 8 |

1 Captura e análise de Tramas *Ethernet*

1.1 Pergunta 1

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

O endereço MAC do destino é 00:d0:03:ff:94:00 e na origem é e0:d4:e8:38:aa:54.

| | | | | | | |
|-----|--------------|---------------|---------------|------|-----|---|
| 187 | 10.259239452 | 172.26.58.253 | 193.137.9.150 | TCP | 74 | 49176 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1713756704 TSecr=0 WS=128 |
| 188 | 10.253431628 | 193.137.9.150 | 172.26.58.253 | TCP | 74 | 80 → 49176 [SYN, ACK] Seq=0 Ack=1 Win=12500 Len=0 MSS=1250 WS=4 SACK_PERM=1 TSval=1373696196 TSecr=1713756704 |
| 189 | 10.253522421 | 172.26.58.253 | 193.137.9.150 | TCP | 66 | 49176 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1713756707 TSecr=1373696196 |
| 190 | 10.253975691 | 172.26.58.253 | 193.137.9.150 | HTTP | 407 | GET / HTTP/1.1 |
| 191 | 10.269236474 | 193.137.9.150 | 172.26.58.253 | HTTP | 198 | HTTP/1.0 302 Moved Temporarily |
| 192 | 10.269292347 | 172.26.58.253 | 193.137.9.150 | TCP | 66 | 49176 → 80 [ACK] Seq=342 Ack=133 Win=64128 Len=0 TSval=1713756714 TSecr=1373696200 |

Figure 1: Pacote TCC de acesso ao *https://elearning.uminho.pt*.

```
▶ Frame 190: 407 bytes on wire (3256 bits), 407 bytes captured (3256 bits) on interface wlo1, id 0
▼ Ethernet II, Src: IntelCor_38:aa:54 (e0:d4:e8:38:aa:54), Dst: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
  ▶ Destination: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
  ▶ Source: IntelCor_38:aa:54 (e0:d4:e8:38:aa:54)
  └─ Type: IPv4 (0x0800)
▶ Internet Protocol Version 4, Src: 172.26.58.253, Dst: 193.137.9.150
▶ Transmission Control Protocol, Src Port: 49176, Dst Port: 80, Seq: 1, Ack: 1, Len: 341
▶ Hypertext Transfer Protocol
```

Figure 2: Trama capturada

1.2 Pergunta 2

Identifique a que sistemas se referem. Justifique.

O endereço MAC de origem refere-se ao endereço físico da nossa máquina, enquanto o endereço de destino se refere ao endereço físico do router.

1.3 Pergunta 3

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

O endereço é 0x0800, como dá para ver na Figura 2. Significa que a camada superior está a utilizar o protocolo IPv4.

1.4 Pergunta 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.

Na Figura 2 podemos observar que cada *frame* tem 407 *bytes*. Dá para ver que há $8 \times 2 \times 4 + 2 = 66$ *bytes* desde o início da trama até ao início dos dados do nível aplicacional. Como sabemos que a trama tem 407 *bytes*, significa que o overhead em percentagem é $(66/407) \times 100 = 16.22\%$.

| | | |
|------|---|----------------------|
| 0000 | 00 d0 03 ff 94 00 e0 d4 e8 38 aa 54 08 00 45 00 |8.T.E. |
| 0010 | 01 89 d4 30 40 00 40 06 b3 07 ac 1a 3a fd c1 89 | ...0@.@...:... |
| 0020 | 09 96 c0 18 00 50 3e fe ad 57 c2 64 6c 24 80 18 |P>..W.dl\$. . |
| 0030 | 01 f6 b3 b2 00 00 01 01 08 0a 66 25 da 24 51 e0 |f% \$Q. |
| 0040 | f0 c4 47 45 54 20 2f 20 48 54 54 50 2f 31 2e 31 | ..GET / HTTP/1.1 |
| 0050 | 0d 0a 48 6f 73 74 3a 20 65 6c 65 61 72 6e 69 6e | ..Host: elearnin |
| 0060 | 67 2e 75 6d 69 6e 68 6f 2e 70 74 0d 0a 55 73 65 | g.uminho .pt..Use |
| 0070 | 72 2d 41 67 65 6e 74 3a 20 4d 6f 7a 69 6c 6c 61 | r-Agent: Mozilla |
| 0080 | 2f 35 2e 30 20 28 58 31 31 3b 20 4c 69 6e 75 78 | /5.0 (X1 1; Linux |
| 0090 | 20 78 38 36 5f 36 34 3b 20 72 76 3a 39 39 2e 30 | x86_64; rv:99.0 |
| 00a0 | 29 20 47 65 63 6b 6f 2f 32 30 31 30 30 31 30 31 |) Gecko/ 20100101 |
| 00b0 | 20 46 69 72 65 66 6f 78 2f 39 39 2e 30 0d 0a 41 | Firefox /99.0..A |
| 00c0 | 63 63 65 70 74 3a 20 74 65 78 74 2f 68 74 6d 6c | ccept: t ext/html |
| 00d0 | 2c 61 70 70 6c 69 63 61 74 69 6f 6e 2f 78 68 74 | , applica tion/xht |
| 00e0 | 6d 6c 2b 78 6d 6c 2c 61 70 70 6c 69 63 61 74 69 | ml+xml,a pplicati |
| 00f0 | 6f 6e 2f 78 6d 6c 3b 71 3d 30 2e 39 2c 69 6d 61 | on/xml;q =0.9,ima |
| 0100 | 67 65 2f 61 76 69 66 2c 69 6d 61 67 65 2f 77 65 | ge/avif, image/we |
| 0110 | 62 70 2c 2a 2f 2a 3b 71 3d 30 2e 38 0d 0a 41 63 | bp, /*/*; q =0.8..Ac |
| 0120 | 63 65 70 74 2d 4c 61 6e 67 75 61 67 65 3a 20 65 | cept-Lan guage: e |
| 0130 | 6e 2d 55 53 2c 65 6e 3b 71 3d 30 2e 35 0d 0a 41 | n-US,en; q=0.5..A |
| 0140 | 63 63 65 70 74 2d 45 6e 63 6f 64 69 6e 67 3a 20 | ccept-En coding: |
| 0150 | 67 7a 69 70 2c 20 64 65 66 6c 61 74 65 0d 0a 43 | gzip, de flate..C |
| 0160 | 6f 6e 6e 65 63 74 69 6f 6e 3a 20 6b 65 65 70 2d | onnectio n: keep- |
| 0170 | 61 6c 69 76 65 0d 0a 55 70 67 72 61 64 65 2d 49 | alive..U pgrade-I |
| 0180 | 6e 73 65 63 75 72 65 2d 52 65 71 75 65 73 74 73 | nsecure- Requests |
| 0190 | 3a 20 31 0d 0a 0d 0a | : 1.... |

Figure 3: Valores do bytes da Trama em estudo

1.5 Pergunta 5

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

Como podemos ver na figura que se segue verificamos que o endereço da fonte *Ethernet* é 00:d0:03:ff:94:00. Corresponde ao endereço físico da interface ativa do *router* com que estamos a comunicar.

| | |
|---|---|
| ▶ | Frame 191: 198 bytes on wire (1584 bits), 198 bytes captured (1584 bits) on interface wlo1, id 0 |
| ▼ | Ethernet II, Src: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00), Dst: IntelCor_38:aa:54 (e0:d4:e8:38:aa:54) |
| ▶ | Destination: IntelCor_38:aa:54 (e0:d4:e8:38:aa:54) |
| ▶ | Source: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00) |
| └ | Type: IPv4 (0x0800) |
| ▶ | Internet Protocol Version 4, Src: 193.137.9.150, Dst: 172.26.58.253 |
| ▶ | Transmission Control Protocol, Src Port: 80, Dst Port: 49176, Seq: 1, Ack: 342, Len: 132 |
| ▶ | Hypertext Transfer Protocol |

Figure 4: Resposta HTTP

1.6 Pergunta 6

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

O endereço MAC destino é e0:d4:e8:38:aa:54, correspondente ao endereço físico da nossa máquina.

1.7 Pergunta 7

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

Como podemos analisar na figura 4, os protocolos contidos na trama são Ethernet, IPv4 (*Internet Protocol Version 4*), TCP (*Transmission Control Protocol*) e HTTP (*Hypertext Transfer Protocol*).

2 Protocolo ARP

2.1 Pergunta 8

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

A primeira coluna mostra os endereços *IP* e a segunda coluna representa os respetivos endereços MAC.

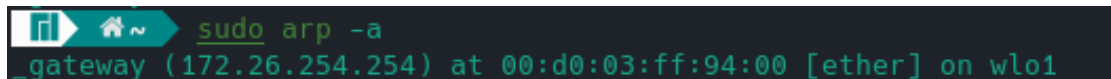


Figure 5: Tabela *arp*

| No. | Time | Source | Destination | Protocol | Length Info |
|-----|--------------|-------------------|-------------------|----------|--|
| 153 | 46.446433981 | IntelCor_38:aa:54 | Broadcast | ARP | 42 Who has 172.26.254.254? Tell 172.26.109.132 |
| 154 | 46.449091065 | ComdaEnt_ff:94:00 | IntelCor_38:aa:54 | ARP | 60 172.26.254.254 is at 00:d0:03:ff:94:00 |

Figure 6: Tráfego ARP

2.2 Pergunta 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?

O valor hexadecimal do endereço origem é e0:d4:e8:aa:54 e o endereço de destino é ff:ff:ff:ff:ff:ff (*Broadcast*). Este endereço de destino é utilizado uma vez que o nosso dispositivo não está diretamente conectado ao dispositivo para o qual queremos enviar uma mensagem e, por isso, é necessário enviar uma mensagem para todos os dispositivos da rede para que assim o dispositivo pretendido possa responder ao seu endereço MAC.

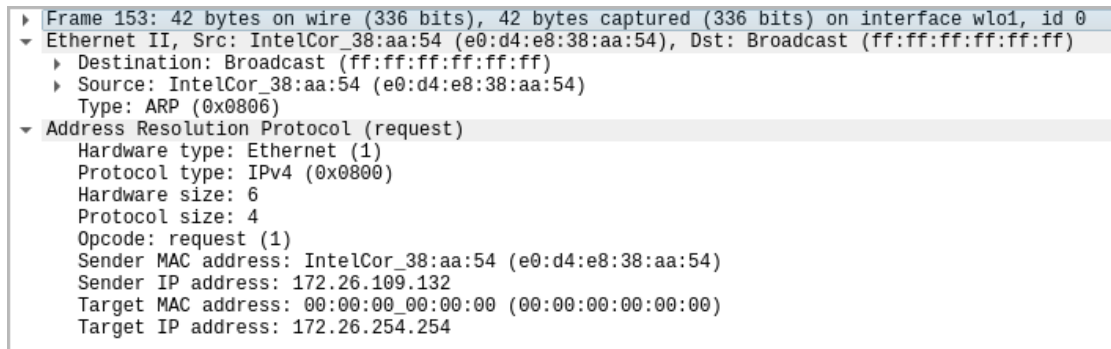


Figure 7: Trama com o pedido ARP

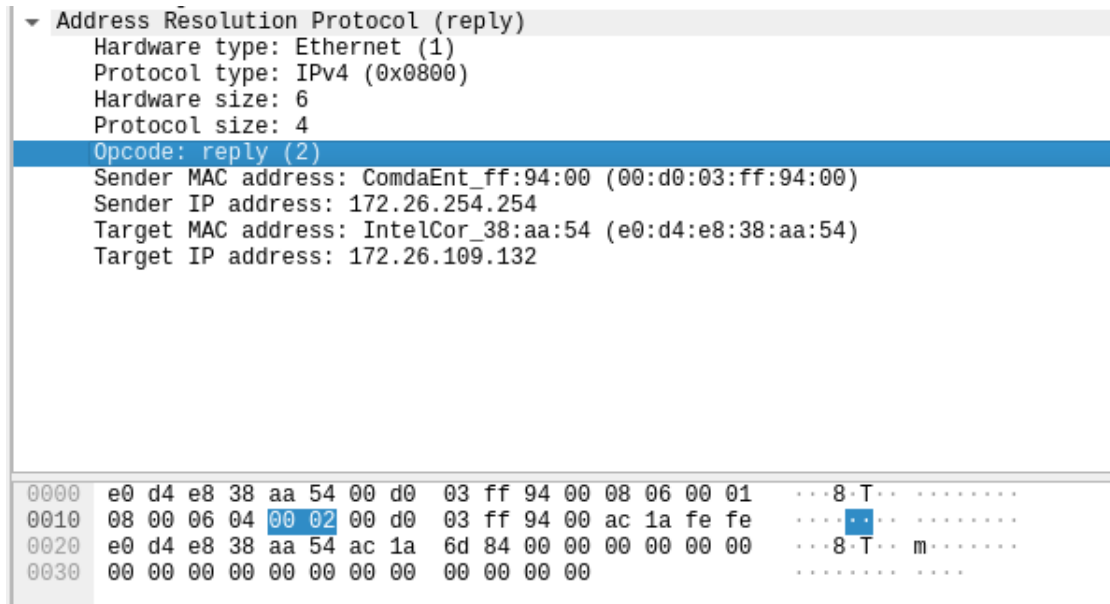


Figure 8: Trama com a resposta ARP

2.7 Pergunta 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.

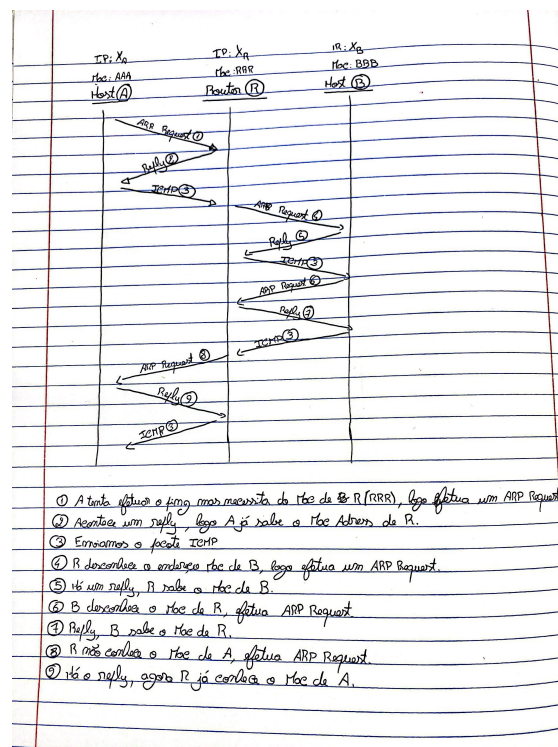


Figure 9: Diagrama das mensagens ARP e ICMP trocada

Figure 11: Tráfego nas interfaces no departamento A (*hub*)

3.2 Pergunta 16

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

Na figura 12, podemos verificar que no departamento com o switch (departamento B), ao contrário do departamento com o *hub* (departamento A), a mensagem entre os laptops não são partilhadas para o servidor, tendo este apenas recebido mensagens OSPF.

| MAC Address | Porta | |
|-------------------|-------|----------|
| 0:00:00::aa:00:27 | 1 | Jasmine |
| 0:00:00:aa:00:29 | 2 | Alladin |
| 0:00:00:aa:00:31 | 3 | Router B |

Table 1: Tabela de comutação de *switch*

```

vcmid
root@Alladin:/tmp/pycore.44405/Alladin.conf# ping 192.168.32.146
PING 192.168.32.146 (192.168.32.146) 56(84) bytes of data:
64 bytes from 192.168.32.146: icmp_seq=1 ttl=64 time=0.510 ms
64 bytes from 192.168.32.146: icmp_seq=2 ttl=64 time=1.65 ms
64 bytes from 192.168.32.146: icmp_seq=3 ttl=64 time=0.242 ms
64 bytes from 192.168.32.146: icmp_seq=4 ttl=64 time=0.318 ms
64 bytes from 192.168.32.146: icmp_seq=5 ttl=64 time=3.72 ms
^C
--- 192.168.32.146 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4055ms
rtt min/avg/max/mdev = 0.242/1.288/3.723/1.319 ms
root@Alladin:/tmp/pycore.44405/Alladin.conf#

vcmid
root@SB:/tmp/pycore.44405/SB.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
23:25:08.921284 IP 192.168.32.145 > 224.0.0.5: OSPFv2, Hello, length 44
23:25:10.922043 IP 192.168.32.145 > 224.0.0.5: OSPFv2, Hello, length 44
23:25:10.935561 IP6 fe80::200:ff:feaa:30 > ff02::5: OSPFv3, Hello, length 36
3 packets captured
3 packets received by filter
0 packets dropped by kernel
root@SB:/tmp/pycore.44405/SB.conf#

vcmid
23:25:07.716577 IP 192.168.32.147 > 192.168.32.146: ICMP echo request, id 24, seq 3, length 64
23:25:07.716608 IP 192.168.32.146 > 192.168.32.147: ICMP echo reply, id 24, seq 3, length 64
23:25:08.732672 IP 192.168.32.147 > 192.168.32.146: ICMP echo request, id 24, seq 4, length 64
23:25:08.732713 IP 192.168.32.146 > 192.168.32.147: ICMP echo reply, id 24, seq 4, length 64
23:25:08.921287 IP 192.168.32.145 > 224.0.0.5: OSPFv2, Hello, length 44
23:25:09.756934 IP 192.168.32.147 > 192.168.32.146: ICMP echo request, id 24, seq 5, length 64
23:25:09.757031 IP 192.168.32.146 > 192.168.32.147: ICMP echo reply, id 24, seq 5, length 64
23:25:10.809728 ARP, Request who-has 192.168.32.147 tell 192.168.32.146, length 28
23:25:10.809895 ARP, Reply 192.168.32.147 is-at 00:00:00:aa:00:23 (oui Ethernet), length 28
23:25:10.922092 IP 192.168.32.145 > 224.0.0.5: OSPFv2, Hello, length 44
23:25:10.935562 IP6 fe80::200:ff:feaa:30 > ff02::5: OSPFv3, Hello, length 36
14 packets captured
14 packets received by filter
0 packets dropped by kernel
root@Jasmine:/tmp/pycore.44405/Jasmine.conf#

vcmid
root@RB:/tmp/pycore.44405/RB.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
23:25:06.920769 IP 10.0.3.2 > 224.0.0.5: OSPFv2, Hello, length 48
23:25:07.201802 IP 10.0.3.1 > 224.0.0.5: OSPFv2, Hello, length 48
23:25:08.921072 IP 10.0.3.2 > 224.0.0.5: OSPFv2, Hello, length 48
23:25:09.201694 IP 10.0.3.1 > 224.0.0.5: OSPFv2, Hello, length 48
23:25:10.692344 IP6 fe80::200:ff:feaa:2e > ff02::5: OSPFv3, Hello, length 40
23:25:10.732567 IP6 fe80::200:ff:feaa:2f > ff02::5: OSPFv3, Hello, length 40
23:25:10.921908 IP 10.0.3.2 > 224.0.0.5: OSPFv2, Hello, length 48
23:25:11.207440 IP 10.0.3.1 > 224.0.0.5: OSPFv2, Hello, length 48
8 packets captured
8 packets received by filter
0 packets dropped by kernel
root@RB:/tmp/pycore.44405/RB.conf#

```

Figure 12: Tráfego nas interfaces no departamento B (*switch*)

4 Conclusão

Com este trabalho conseguimos aprofundar conhecimentos sobre a *Ethernet* e a sua organização e conhecer melhor o protocolo ARP, utilizando o *wireshark* e o CORE para que fosse possível responder às questões propostas.

Também conseguimos aprofundar melhor a diferença entre um *switch* e um *hub* e os impactos que estes têm no tráfego de rede.