

Universidade do Minho

Departamento de Informática

TP3 - Nível de Ligação Lógica: Ethernet e
Protocolo ARP

Redes de Computadores

Grupo 37

Catarina Pais Vieira (a89524)

José Duarte Pereira de Castro Alves (a89563)

Leonardo de Freitas Marreiros (a89537)

Conteúdo

Questões e Respostas	3
3. Captura e análise de Tramas Ethernet	3
Exercício 1	3
Exercício 2	3
Exercício 3	3
Exercício 4	3
Exercício 5	4
Exercício 6	4
Exercício 7	4
Exercício 8	4
4. Protocolo ARP	5
Exercício 9	5
Exercício 10	5
Exercício 11	6
Exercício 12	6
Exercício 13	6
Exercício 14	7
5. ARP Gratuito	8
Exercício 15	8
6. Domínios de colisão	8
Exercício 16	8
Conclusão	11

Questões e Respostas

3. Captura e análise de Tramas Ethernet

Exercício 1

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



```
▼ Ethernet II, Src: Apple_62:de:d1 (88:e9:fe:62:de:d1), Dst: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
```

Figura 1 - endereço MAC

Pela Figura 1 observamos que o endereço MAC de destino é 00:d0:03:ff:94:00 e o endereço MAC de origem é 88:e9:fe:62:de:d1.

Exercício 2

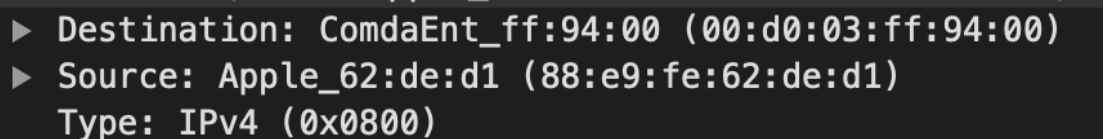
Identifique a que sistemas se referem. Justifique.

Sistema destino de endereço 00:d0:03:ff:94:00, que corresponde à interface da rede local do router, pois trata-se de um endereço MAC e o *request* tem como destino final um sistema fora da rede local.

Sistema origem de endereço 88:e9:fe:62:de:d1, que corresponde ao *host*, pois este fez o *request*.

Exercício 3

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



```
► Destination: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
► Source: Apple_62:de:d1 (88:e9:fe:62:de:d1)
Type: IPv4 (0x0800)
```

Figura 2 - trama Ethernet

Como se pode ver pela Figura 2, o campo *type* tem o valor 0x0800. Indica que o protocolo IP utilizado ao nível da rede é IPv4.

Exercício 4

Quanto bytes são usados desde o início da trama até ao caractere ASCII “G” do método HTTP GET? Calcule e indique, em percentagem, a sobrecarga (*overhead*) introduzida pela pilha protocolar no envio do HTTP GET.

0000	00 d0 03 ff 94 00 88 e9 fe 62 de d1 08 00 45 02b....E.
0010	01 ff 00 00 40 00 40 06 8f f0 ac 1a 31 cd c1 89	...@.@...1...
0020	09 96 c7 a2 00 50 bc c6 49 62 8b 71 7d 89 80 18	...P..Ib.q}...
0030	08 02 22 01 00 00 01 01 08 0a 27 93 b3 a6 d7 f7	...".....'
0040	2a f4 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 41 63 63	g.uminho .pt..Acc
0070	65 70 74 3a 20 74 65 78 74 2f 68 74 6d 6c 2c 61	ept: text/html,a
0080	70 70 6c 69 63 61 74 69 6f 6e 2f 78 68 74 6d 6c	pplicati on/xhtmll
0090	2b 78 6d 6c 2c 61 70 70 6c 69 63 61 74 69 6f 6e	+xml,app lication

Figura 3- bytes até ao G

Pela Figura 3, vemos que até ao caractere ASCII “G” são utilizados 66 bytes. No total são usados 525 bytes, logo, a percentagem da sobrecarga é: $66/525 * 100 = 12.57\%$.

Exercício 5

Através de visualização direta ou construindo um filtro específico, verifique se foram detetadas tramas com erros (por verificação do campo FCS (Frame Check Sequence)).

Segundo a documentação do Wireshark a maior parte dos sistemas operativos não suportam a captura do FCS em Ethernet, logo o Wireshark não deteta os FCS das tramas, o que não permite verificar se foram detetadas tramas com erros.

Exercício 6

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

```
▼ Ethernet II, Src: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00), Dst: Apple_62:de:d1 (88:e9:fe:62:de:d1)
  ► Destination: Apple_62:de:d1 (88:e9:fe:62:de:d1)
  ► Source: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
```

Figura 4 - trama Ethernet

Endereço de origem: 00:d0:03:ff:94:00, como se trata de um endereço MAC e de uma resposta que vem de fora da rede local, este endereço corresponde ao router dessa rede.

Exercício 7

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

Endereço destino: 88:e9:fe:62:de:d1, que corresponde à interface Ethernet da máquina nativa.

Exercício 8

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

```
[Protocols in frame: eth:ethertype:ip:tcp:http]
```

Figura 5 - protocolos da trama recebida

Os protocolos contidos na trama da Figura 4 são: Ethernet, IPv4, TCP e HTTP

4. Protocolo ARP

Exercício 9

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

```
Interface: 192.168.56.1 --- 0x7
  Internet Address      Physical Address      Type
  192.168.56.255        ff-ff-ff-ff-ff-ff    static
  224.0.0.22            01-00-5e-00-00-16    static
  224.0.0.251           01-00-5e-00-00-fb    static
  224.0.0.252           01-00-5e-00-00-fc    static
  239.255.255.250       01-00-5e-7f-ff-fa    static
  255.255.255.255       ff-ff-ff-ff-ff-ff    static

Interface: 172.26.91.121 --- 0x14
  Internet Address      Physical Address      Type
  172.26.254.254        00-d0-03-ff-94-00    dynamic
  172.26.255.255        ff-ff-ff-ff-ff-ff    static
  224.0.0.22            01-00-5e-00-00-16    static
  224.0.0.251           01-00-5e-00-00-fb    static
  224.0.0.252           01-00-5e-00-00-fc    static
  239.255.255.250       01-00-5e-7f-ff-fa    static
  255.255.255.255       ff-ff-ff-ff-ff-ff    static
```

Figura 6 - tabela ARP

A tabela ARP mapeia o endereço IP para o endereço MAC dos sistemas que comunicaram recentemente. A primeira coluna representa o endereço IP do *host*, a segunda coluna representa o MAC *address* e a terceira o tipo do endereço usado.

Exercício 10

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?

1278 4.175797	IntelCor_c4:c3:b3	Broadcast	ARP	42 Who has 172.26.254.254? Tell 172.26.91.121
> Frame 1278: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface \Device\NPF_{B36B90C9-7245-4092-B18E-804B7CA4D870}, id 0				
▼ Ethernet II, Src: IntelCor_c4:c3:b3 (7c:2a:31:c4:c3:b3), Dst: Broadcast (ff:ff:ff:ff:ff:ff)				
Destination: Broadcast (ff:ff:ff:ff:ff:ff)				
Address: Broadcast (ff:ff:ff:ff:ff:ff)				
.....1. = LG bit: Locally administered address (this is NOT the factory default)				
.....1. = IG bit: Group address (multicast/broadcast)				
▼ Source: IntelCor_c4:c3:b3 (7c:2a:31:c4:c3:b3)				
Address: IntelCor_c4:c3:b3 (7c:2a:31:c4:c3:b3)				
.....0. = LG bit: Globally unique address (factory default)				
.....0. = IG bit: Individual address (unicast)				
Type: ARP (0x0806)				
> Address Resolution Protocol (request)				

Figura 7 - pedido ARP

O valor hexadecimal dos endereços de origem é 7c:2a:31:c4:c3:b3 e o valor hexadecimal dos endereços destino é ff:ff:ff:ff:ff:ff.

É usado o endereço ethernet do *broadcast* (da camada 2) para poder ser recebido por todos os *hosts* da rede.

Exercício 11

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

O valor hexadecimal do campo tipo da trama Ethernet pode-se ver na Figura 7. Tem o valor 0x0806. Isto indica que o campo de dados pertence ao ARP.

Exercício 12

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

```
▼ Address Resolution Protocol (request)
  Hardware type: Ethernet (1)
  Protocol type: IPv4 (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: request (1)
  Sender MAC address: IntelCor_c4:c3:b3 (7c:2a:31:c4:c3:b3)
  Sender IP address: 172.26.91.121
  Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
  Target IP address: 172.26.254.254
```

Figura 8 - pedido ARP

Pela observação da Figura 8 verificamos que o *opcode* tem o valor 1 pelo que se trata de um pedido ARP (RFC 826, RFC 5227). Nesta mensagem estão contidos endereços MAC e IP do remetente e do alvo. Se um *host* está a comunicar com outro *host* na mesma rede IP, o destino do pedido ARP é o endereço IP do outro *host*. Se um *host* estiver a comunicar com outro *host* numa rede IP diferente, o destino do pedido ARP será o endereço IP do *gateway* padrão. Neste caso, o *host* com IP 172.26.91.121 e MAC 7c:2a:31:c4:c3:b3 pretende saber qual o MAC do *host* com IP 172.26.254.254 pelo que o MAC alvo é o endereço de *broadcast*.

Exercício 13

Explícite que tipo de pedido ou pergunta é feita pelo *host* de origem?

A pergunta feita pelo *host* de origem é do tipo “Who has 172.26.254.254? Tell 172.26.92.121”. Isto significa que perguntamos ao *host* da rede qual o MAC e quem tem o IP 192.26.92.254, e pedimos para enviar a resposta para 172.26.92.121.

Localize a mensagem ARP que é a resposta ao pedido ARP efetuado. **a.** Qual o valor do campo ARP *opcode*? O que especifica? **b.** Em que posição da mensagem ARP está a resposta ao pedido ARP?

Figura 9 - campo ARP opcode

- ```

1279 4.179442 ComdaEnt_ff:94:00 IntelCor_c4:c3:b3 ARP 60 172.26.254.254 is at 00:d0:03:ff:94:00
> Frame 1279: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF_{B36B90C9-7245-4092-B18E-804B7CA4D870}, id 0
v Ethernet II, Src: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00), Dst: IntelCor_c4:c3:b3 (7c:2a:31:c4:c3:b3)
 > Destination: IntelCor_c4:c3:b3 (7c:2a:31:c4:c3:b3)
 v Source: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
 Address: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
 0. = LG bit: Globally unique address (factory default)
 0. = IG bit: Individual address (unicast)
 Type: ARP (0x0806)
 Padding: 00
 v Address Resolution Protocol (reply)
 Hardware type: Ethernet (1)
 Protocol type: IPv4 (0x0800)
 Hardware size: 6
 Protocol size: 4
 Opcode: reply (2)
 Sender MAC address: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
 Sender IP address: 172.26.254.254
 Target MAC address: IntelCor_c4:c3:b3 (7c:2a:31:c4:c3:b3)
 Target IP address: 172.26.91.121

0000 7c 2a 31 c4 c3 b3 00 d0 03 ff 94 00 08 06 00 01 |*1...:.....
0010 08 00 06 04 00 02 00 d0 03 ff 94 00 ac 1a fe fe |.....[y....
0020 7c 2a 31 c4 c3 b3 ac 1a 5b 79 00 00 00 00 00 00 |*1.....[y....
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |.....

```

Figura 10 - pedido ARP

## 5. ARP Gratuito

### Exercício 15

Identifique um pacote de pedido ARP gratuito originado pelo seu sistema. Analise o conteúdo de um pedido ARP gratuito e identifique em que se distingue dos restantes pedidos ARP. Registe a trama Ethernet correspondente. Qual o resultado esperado face ao pedido ARP gratuito enviado?

```
438 22.964954 IntelCor_78:fa:0a Broadcast ARP 42 ARP Announcement for 172.26.91.28
> Frame 438: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface \Device\NPF_{177160AE-3B1D-469B-A8D2-4BC81DB1D662}, id 0
v Ethernet II, Src: IntelCor_78:fa:0a (34:c9:3d:78:fa:0a), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
 v Destination: Broadcast (ff:ff:ff:ff:ff:ff)
 Address: Broadcast (ff:ff:ff:ff:ff:ff)
 1. = LG bit: Locally administered address (this is NOT the factory default)
 1. = IG bit: Group address (multicast/broadcast)
 v Source: IntelCor_78:fa:0a (34:c9:3d:78:fa:0a)
 Address: IntelCor_78:fa:0a (34:c9:3d:78:fa:0a)
 0. = LG bit: Globally unique address (factory default)
 0. = IG bit: Individual address (unicast)
 Type: ARP (0x0806)
 v Address Resolution Protocol (ARP Announcement)
 Hardware type: Ethernet (1)
 Protocol type: IPv4 (0x0800)
 Hardware size: 6
 Protocol size: 4
 Opcode: request (1)
 [Is gratuitous: True]
 [Is announcement: True]
 Sender MAC address: IntelCor_78:fa:0a (34:c9:3d:78:fa:0a)
 Sender IP address: 172.26.91.28
 Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
 Target IP address: 172.26.91.28
```

Figura 11 - pedido ARP gratuito

Analisando a Figura 11 verificamos que este pedido ARP diferencia dos anteriores pela presença da *flag* – “Is gratuitous”, que apresenta o valor True, o que significa que se trata de um pedido ARP gratuito. Além disso, os endereços IP alvo e remetente são iguais. O ARP gratuito é enviado como um *broadcast*, como forma de um nó anunciar ou atualizar seu mapeamento IP para MAC para toda a rede. **Nota:** Foram também enviados 3 pedidos ARP de sonda (*Probe*) antes do pedido ARP de anúncio (*Announcement*).

## 6. Domínios de colisão

### Exercício 16

Através da opção tcpdump verifique e compare como flui o tráfego nas diversas interfaces dos vários dispositivos no departamento A (LAN comutada) e no departamento B (LAN partilhada) quando gera tráfego intra-departamento (por exemplo, através do comando ping). Que conclui? Comente os resultados obtidos quanto à utilização de *hubs* e *switches* no contexto de controlar ou dividir domínios de colisão. Documente as suas observações e conclusões com base no tráfego observado/capturado.



```

root@n11:/tmp/pycore.36715/n11.conf# tcpdump
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
^C13:35:15.691835 IP 130.39.104.3 > 130.39.104.2: ICMP echo request, id 27, seq
5, length 64
13:35:15.691858 IP 130.39.104.2 > 130.39.104.3: ICMP echo reply, id 27, seq 5, l
ength 64
13:35:16.463808 IP 130.39.104.1 > 224.0.0.5: OSPFv2, Hello, length 44
13:35:16.489512 IP6 fe80::200:ff:feaa:14 > ff02::5: OSPFv3, Hello, length 36
13:35:16.623765 ARP, Request who-has 130.39.104.3 tell 130.39.104.2, length 28
13:35:16.623859 ARP, Request who-has 130.39.104.2 tell 130.39.104.3, length 28
13:35:16.623897 ARP, Reply 130.39.104.2 is-at 00:00:00:aa:00:15 (oui Ethernet),
length 28
13:35:16.623902 ARP, Reply 130.39.104.3 is-at 00:00:00:aa:00:16 (oui Ethernet),
length 28
13:35:16.716177 IP 130.39.104.3 > 130.39.104.2: ICMP echo request, id 27, seq 6,
length 64
13:35:16.716207 IP 130.39.104.2 > 130.39.104.3: ICMP echo reply, id 27, seq 6, l
ength 64
13:35:17.747317 IP 130.39.104.3 > 130.39.104.2: ICMP echo request, id 27, seq 7,
length 64
13:35:17.747346 IP 130.39.104.2 > 130.39.104.3: ICMP echo reply, id 27, seq 7, l
ength 64
13:35:17.747355 IP 130.39.104.3 > 130.39.104.2: ICMP echo request, id 27, seq 8,
length 64
13:35:17.747365 IP 130.39.104.2 > 130.39.104.3: ICMP echo reply, id 27, seq 8, l
ength 64
14 packets captured
14 packets received by filter
0 packets dropped by kernel

```

Figura 12 - Host departamento B com Hub

```

root@RB:/tmp/pycore.36715/RB.conf# tcpdump -i eth3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth3, link-type EN10MB (Ethernet), capture size 262144 bytes
13:35:15.691835 IP 130.39.104.3 > 130.39.104.2: ICMP echo request, id 27, seq 5, length 64
13:35:15.691858 IP 130.39.104.2 > 130.39.104.3: ICMP echo reply, id 27, seq 5, length 64
13:35:16.463781 IP 130.39.104.1 > 224.0.0.5: OSPFv2, Hello, length 44
13:35:16.489499 IP6 fe80::200:ff:feaa:14 > ff02::5: OSPFv3, Hello, length 36
13:35:16.623856 ARP, Request who-has 130.39.104.3 tell 130.39.104.2, length 28
13:35:16.623860 ARP, Request who-has 130.39.104.2 tell 130.39.104.3, length 28
13:35:16.623903 ARP, Reply 130.39.104.3 is-at 00:00:00:aa:00:16 (oui Ethernet), length 28
13:35:16.623906 ARP, Reply 130.39.104.2 is-at 00:00:00:aa:00:15 (oui Ethernet), length 28
13:35:16.716179 IP 130.39.104.3 > 130.39.104.2: ICMP echo request, id 27, seq 6, length 64
13:35:16.716226 IP 130.39.104.2 > 130.39.104.3: ICMP echo reply, id 27, seq 6, length 64
13:35:17.747318 IP 130.39.104.3 > 130.39.104.2: ICMP echo request, id 27, seq 7, length 64
13:35:17.747365 IP 130.39.104.2 > 130.39.104.3: ICMP echo reply, id 27, seq 7, length 64
16 packets captured
16 packets received by filter
0 packets dropped by kernel

```

Figura 13 - Router departamento B com Hub

Foram feitos pings de um *host* do departamento B para um outro *host* do mesmo departamento (Figura 12 - tcpdump host recetor, Figura 13 - tcpdump da interface da rede B do respetivo router). Assim, como se pode ver pelas imagens, no departamento B (com *hub* em vez de *switch*), tudo passa no router (tanto o ping, como a resposta, como até tramas ARP), o que suporta o facto de existir um domínio de colisão entre o *hub* e a interface do router neste tipo de redes.

```

root@n10:/tmp/pycore.36715/n10.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
^C13:40:45.487418 IP 130.39.96.2 > 130.39.96.3: ICMP echo request, id 27, seq 6,
length 64
13:40:45.487433 IP 130.39.96.3 > 130.39.96.2: ICMP echo reply, id 27, seq 6, len
gth 64
13:40:45.578878 ARP, Request who-has 130.39.96.2 tell 130.39.96.3, length 28
13:40:45.578938 ARP, Reply 130.39.96.2 is-at 00:00:00:aa:00:09 (oui Ethernet), 1
length 28
13:40:46.523916 IP 130.39.96.2 > 130.39.96.3: ICMP echo request, id 27, seq 7, 1
length 64
13:40:46.523943 IP 130.39.96.3 > 130.39.96.2: ICMP echo reply, id 27, seq 7, len
gth 64
13:40:46.655730 IP 130.39.96.1 > 224.0.0.5: OSPFv2, Hello, length 44
13:40:46.846069 IP6 fe80::200:ff:feaa:b > ff02::5: OSPFv3, Hello, length 36
13:40:47.532495 IP 130.39.96.2 > 130.39.96.3: ICMP echo request, id 27, seq 8, 1
length 64
13:40:47.532527 IP 130.39.96.3 > 130.39.96.2: ICMP echo reply, id 27, seq 8, len
gth 64

10 packets captured
10 packets received by filter
0 packets dropped by kernel

```

Figura 14 - Host departamento A com switch

```

root@RA:/tmp/pycore.36715/RA.conf# tcpdump -i eth2
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth2, link-type EN10MB (Ethernet), capture size 262144 bytes
^C
0 packets captured
0 packets received by filter
0 packets dropped by kernel
root@RA:/tmp/pycore.36715/RA.conf#

```

Figura 15 - Router departamento A com switch

O mesmo método foi utilizado para a observação da rede do departamento A que tem um *switch* em vez de um *hub* (Figura 14 - tcpdump do host recetor, Figura 15 - tcpdump da interface da rede A do respetivo router). Como é observável nas imagens, ao router não chega qualquer informação, o que mostra que é o *switch* que está a controlar o fluxo da rede local e que suporta o facto de que apenas as portas do *switch* são potencialmente domínios de colisão (dado que, em oposição a “half-duplex”, se a comunicação for “full-duplex” não existe domínio de colisão).

## Conclusão

Com este trabalho, foram colocados em prática conhecimentos referentes ao capítulo *Link Layer* adquiridos durante as aulas teóricas anteriores, o que nos levou a consolidar melhor a matéria. Podemos dividir este trabalho em três partes: análise de tramas Ethernet, protocolo ARP e domínios de colisão.

Com a ajuda do Wireshark e o Core conseguimos capturar e analisar tramas de Ethernet, esta informação foi essencial para a realização deste trabalho, e ajudou-nos a aprimorar os conhecimentos referidos anteriormente.

A primeira parte foi baseada na utilização de uma conexão por Ethernet. Na segunda parte focamo-nos nos pacotes ARP e na terceira parte o nosso foco virou para a comparação entre *Hubs* e *Switches*.

Com este trabalho prático abordamos melhor a camada de ligação lógica, percebendo melhor como funciona a interconexão de redes locais baseado no envio de pacotes.

Utilizando a ferramenta anteriormente referida, Core, simulamos um ambiente que nos permitiu analisar a maneira de funcionamento dos domínios de colisão e o modo como eles são corrigidos.

Com isto, basicamente todo este capítulo de *Link Layer* foi abrangido e consolidado.