

Trabalho Prático Nº3 - Nível de Ligação Lógica: Redes Ethernet e Protocolo ARP

Simão Cunha^[a93262], Duarte Leitão^[a100550], and Diogo Barros^[a100600]

Universidade do Minho - Campus de Gualtar, R. da Universidade, 4710-057 Braga Portugal

Redes de Computadores (2022/2023) - PL10 - Grupo 7

Captura e análise de Tramas Ethernet

1 Anote os endereços MAC de origem e de destino da trama capturada. Identifique a que sistemas se referem. Justifique.

De acordo com a figura 1, o endereço MAC da origem da trama capturada é 48:a4:72:f0:84:0d e o endereço MAC do destino é 00:d0:03:ff:94:00

No.	Time	Source	Destination	Protocol	Length	Info
115	1.416287508	172.26.96.29	193.137.9.171	HTTP	414	GET / HTTP/1.1
116	1.420235643	193.137.9.171	172.26.96.29	HTTP	195	HTTP/1.0 302 Moved Temporarily
52	0.673739040	172.26.96.29	194.210.238.81	OCSP	489	Request
53	0.683385784	194.210.238.81	172.26.96.29	OCSP	955	Response
84	0.801878397	172.26.96.29	194.210.238.81	OCSP	489	Request
85	0.816882432	194.210.238.81	172.26.96.29	OCSP	954	Response
167	1.525661712	172.26.96.29	194.210.238.81	OCSP	489	Request
176	1.535502161	194.210.238.81	172.26.96.29	OCSP	955	Response

Frame 115: 414 bytes on wire (3312 bits), 414 bytes captured (3312 bits) on interface wlo1, id 0

Ethernet II, Src: IntelCor_f0:84:0d (48:a4:72:f0:84:0d), Dst: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)

Internet Protocol Version 4, Src: 172.26.96.29, Dst: 193.137.9.171

Transmission Control Protocol, Src Port: 33126, Dst Port: 80, Seq: 1, Ack: 1, Len: 348

Hypertext Transfer Protocol

GET / HTTP/1.1\r\n

Host: alunos.uminho.pt\r\n

User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/112.0\r\n

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8\r\n

Accept-Language: en-US,en;q=0.5\r\n

Accept-Encoding: gzip, deflate\r\n

Connection: keep-alive\r\n

Upgrade-Insecure-Requests: 1\r\n

\r\n

[Full request URI: <http://alunos.uminho.pt/>]

[HTTP request 1/1]

[Response in frame 116]

Figura 1: Captura de tramas

O endereço MAC de origem refere-se à máquina nativa onde se efetuou a captura do tráfego e o endereço MAC destino refere-se ao router presente na sala de aula. Isto justifica-se pelo facto de este último ser o endereço que encaminha a trama para fora da rede da sala de aula.

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

De acordo com a figura 2, o valor hexadecimal do campo Type da trama Ethernet é 0x0800, o que nos indica que estamos a utilizar o protocolo IPv4 ao nível da rede.

```

Frame 115: 414 bytes on wire (3312 bits), 414 bytes captured (3312 bits) on interface wlo1, id 0
Ethernet II, Src: IntelCor_f0:84:0d (48:a4:72:f0:84:0d), Dst: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
  Destination: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
  Source: IntelCor_f0:84:0d (48:a4:72:f0:84:0d)
  Type: IPv4 (0x0800)

```

Figura 2: Visualização do valor *type*

3 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, no caso de HTTPS)? Calcule e indique, em percentagem, a sobrecarga (overhead) introduzida pela pilha protocolar

Para responder a esta questão, temos de verificar os seguintes valores:

- overhead no endereço Ethernet
- overhead no endereço IP
- e overhead no endereço TCP

Sabemos de antemão que o overhead do endereço Ethernet é 14 bytes, por este ser um endereço físico, e, de acordo com as figuras abaixo, concluimos que o overhead no endereço IP é 20 e o overhead do TCP é 32.

```

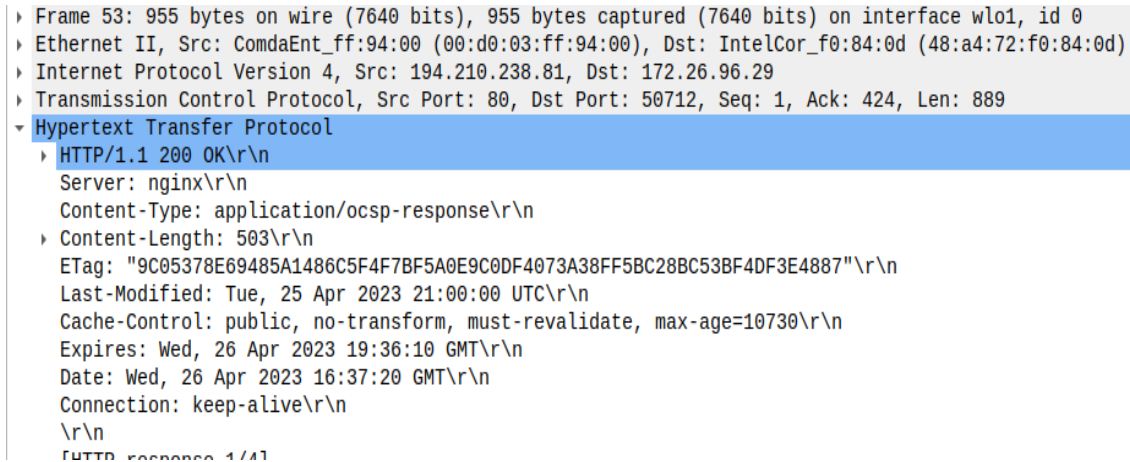
Internet Protocol Version 4, Src: 172.26.96.29, Dst: 193.137.9.171
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
Transmission Control Protocol, Src Port: 33126, Dst Port: 80, Seq: 1, Ack: 1, Len: 348
  Source Port: 33126
  Destination Port: 80
  [Stream index: 0]
  [Conversation completeness: Incomplete, DATA (15)]
  [TCP Segment Len: 348]
  Sequence Number: 1 (relative sequence number)
  Sequence Number (raw): 3250100701
  [Next Sequence Number: 349 (relative sequence number)]
  Acknowledgment Number: 1 (relative ack number)
  Acknowledgment number (raw): 3667946352
  1000 .... = Header Length: 32 bytes (8)
Frame 115: 414 bytes on wire (3312 bits), 414 bytes captured (3312 bits) on interface wlo1, id 0

```

Figura 3: Cálculo dos overheads

Assim, o overhead na pilha protocolar é $14 + 20 + 32 = 66$ bytes a mais, traduzindo-se em 15.94% $((66/414)*100)$.

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



```

Frame 53: 955 bytes on wire (7640 bits), 955 bytes captured (7640 bits) on interface wlo1, id 0
Ethernet II, Src: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00), Dst: IntelCor_f0:84:0d (48:a4:72:f0:84:0d)
Internet Protocol Version 4, Src: 194.210.238.81, Dst: 172.26.96.29
Transmission Control Protocol, Src Port: 80, Dst Port: 50712, Seq: 1, Ack: 424, Len: 889
Hypertext Transfer Protocol
  HTTP/1.1 200 OK\r\n
    Server: nginx\r\n
    Content-Type: application/ocsp-response\r\n
  Content-Length: 503\r\n
  ETag: "9C05378E69485A1486C5F4F7BF5A0E9C0DF4073A38FF5BC28BC53BF4DF3E4887"\r\n
  Last-Modified: Tue, 25 Apr 2023 21:00:00 UTC\r\n
  Cache-Control: public, no-transform, must-revalidate, max-age=10730\r\n
  Expires: Wed, 26 Apr 2023 19:36:10 GMT\r\n
  Date: Wed, 26 Apr 2023 16:37:20 GMT\r\n
  Connection: keep-alive\r\n
  \r\n
  HTTP/1.1 200 OK 1/41

```

Figura 4: Endereço Ethernet

De acordo com a figura 4, o endereço Ethernet da fonte é 00:d0:03:ff:94:00 e corresponde ao router da rede local ao qual a máquina nativa em uso está conectada, considerando que este encaminha as respostas de fora da rede local para dentro da mesma.

5 Qual é o endereço MAC do destino? A que sistema (host) corresponde?

Como se pode observar na figura 4, o endereço MAC do destino é 48:a4:72:f0:84:0d, correspondendo à interface em utilização na nossa máquina nativa.

6 Atendendo ao conceito de encapsulamento protocolar, identifique os vários protocolos contidos na trama recebida. Justifique, indicando em que campos dos cabeçalhos capturados se baseou.

Tal como podemos observar na figura 4, os protocolos contidos na trama recebida são:

- HTTP (Hypertext Transfer Protocol)
- IPv4 (Internet Protocol Version 4)
- Ethernet II
- TCP (Transmission Control Protocol)

Protocolo ARP

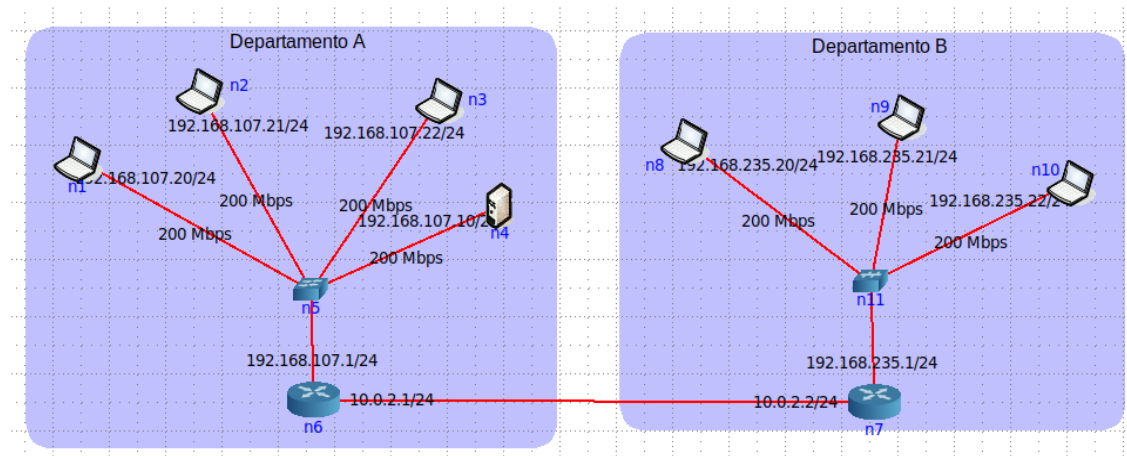


Figura 5: Topologia criada para este trabalho prático

7 Abra uma consola no PC onde efetuou o ping. Observe o conteúdo da tabela ARP com o comando `arp -a`.

7.1 Com a ajuda do manual ARP (`man arp`), interprete o significado de cada uma das colunas da tabela.

Neste exercício iremos escolher o PC n1, que irá fazer ping para n9 e n10. De acordo com a figura abaixo, a tabela ARP obtida apresenta a estrutura [endereço IP] at [endereço físico] [tipo de endereçamento utilizado] on [interface].

Na tabela ARP são indicadas as correspondências entre endereços IP e endereços MAC que estabeleceram ligação, juntamente com a interface do nó origem.

```
root@n1:/tmp/pycore.43935/n1.conf# arp -a
? (192.168.107.1) at 00:00:00:aa:00:08 [ether] on eth0
root@n1:/tmp/pycore.43935/n1.conf#
```

Figura 6: Execução do comando `arp -a`

7.2 Indique, justificando, qual o equipamento da intranet em causa que poderá apresentar a maior tabela ARP em termos de número de entradas.

O equipamento em causa será o router de acesso ao departamento A, denominado n6. Isto ocorre porque este router precisa de manter informações de routing de tráfego entre os nós do departamento (n1, n2, n3 e n4) e a todas as redes exteriores (internet) às quais há comunicação, que neste caso são todos os nós do departamento B.

8 Observe a trama Ethernet que contém a mensagem com o pedido ARP (ARP Request).

17	24.186910444	00:00:00_aa:00:00	Broadcast	ARP	42	Who has 192.168.107.1? Tell 192.168.107.20
18	24.187121574	00:00:00_aa:00:08	00:00:00_aa:00:00	ARP	42	192.168.107.1 is at 00:00:00_aa:00:08

▶ Frame 17: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface veth1.0.34, id 0 ▼ Ethernet II, Src: 00:00:00_aa:00:00 (00:00:00_aa:00:00), Dst: Broadcast (ff:ff:ff:ff:ff:ff) ▶ Destination: Broadcast (ff:ff:ff:ff:ff:ff) ▶ Source: 00:00:00_aa:00:00 (00:00:00_aa:00:00) 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:00:00 (00:00:00_aa:00:00) Sender IP address: 192.168.107.20 Target MAC address: 00:00:00_00:00:00 (00:00:00_00:00:00) Target IP address: 192.168.107.1

Figura 7: Trama Ethernet que contém o ARP request com a execução do comando `ping n1 → n9`

8.1 Qual é o valor hexadecimal dos endereços MAC origem e destino? Como interpreta e justifica o endereço destino usado?

De acordo com a figura 7, o valor hexadecimal do endereço MAC de origem é `00:00:00_aa:00:00` e o valor hexadecimal do endereço MAC de destino é `ff:ff:ff:ff:ff:ff`.

O endereço de destino é o endereço de broadcast. Este é utilizado quando o endereço IP que está a ser procurado não se encontra na tabela ARP do host. Isso permite que todos os hosts da rede em questão recebam a mensagem com o pedido ARP. Assim, logo que a máquina de origem receba a resposta ao seu pedido com o endereço MAC da máquina destino, esta adiciona este endereço à sua tabela ARP.

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

De acordo com a figura 7, o valor hexadecimal para o campo Type da trama Ethernet é `0x0806`. Este valor indica que os dados em questão pertencem ao protocolo ARP.

8.3 Observando a mensagem ARP, como pode saber que se trata efetivamente de um pedido ARP? Refira duas formas distintas de obter essa informação.

Como se pode observar na figura 7, o campo *ARP opcode* possui dois valores: a tag de request e valor 1, indicando que estamos perante um pedido ARP.

No pedido ARP, é possível identificar três tipos de endereços:

- Endereço MAC de origem
- Endereço IP de origem

- Endereço IP de destino

Tendo em conta estes tipos, podemos concluir que existe um host (que possui um endereço IP de origem e um endereço MAC de origem) pretende saber qual o endereço MAC do host a que se quer conectar (endereço MAC de destino), cujo endereço IP é o endereço IP de destino (192.168.107.1).

8.4 Explícite, em linguagem comum, que tipo de pedido ou pergunta é feita pelo host de origem à rede?

Segundo a figura 7, o pedido feito pelo host é: `Who has 192.168.107.1? Tell 192.168.107.20`, ou seja, pergunta quem tem o endereço MAC 192.168.107.1 e pede-se para que a resposta seja entregue a 192.168.107.20.

9 Localize a mensagem ARP que é a resposta ao pedido ARP efetuado

17	24.186910444	00:00:00_aa:00:00	Broadcast	ARP	42	Who has 192.168.107.1? Tell 192.168.107.20
18	24.187121574	00:00:00_aa:00:08	00:00:00_aa:00:00	ARP	42	192.168.107.1 is at 00:00:00_aa:00:08
▶ Frame 18: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface veth1.0.34, id 0 ▼ Ethernet II, Src: 00:00:00_aa:00:08 (00:00:00_aa:00:08), Dst: 00:00:00_aa:00:00 (00:00:00_aa:00:00) ▶ Destination: 00:00:00_aa:00:00 (00:00:00_aa:00:00) ▶ Source: 00:00:00_aa:00:08 (00:00:00_aa:00:08) 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:00:08 (00:00:00_aa:00:08) Sender IP address: 192.168.107.1 Target MAC address: 00:00:00_aa:00:00 (00:00:00_aa:00:00) Target IP address: 192.168.107.20						

Figura 8: Trama Ethernet que contém o ARP reply com a execução do comando `ping n1 → n9`

9.1 Qual o valor do campo ARP opcode? O que especifica?

Tal como observado na figura 8, o valor do campo ARP opcode é `reply (2)`, indicando que estamos perante uma resposta a um pedido no protocolo ARP.

9.2 Em que posição da mensagem ARP está a resposta ao pedido ARP efetuado?

Tal como observado na figura 8, a resposta ao pedido ARP encontra-se no encontra-se entre os bytes 6 e 11 (inclusive).

0000	00 00 00 aa 00 00	00 00 00 aa 00 08	08 06 00 01
0010	08 00 06 04 00 02	00 00 00 aa 00 08	c0 a8 6b 01k
0020	00 00 00 aa 00 00	c0 a8 6b 14		k

Figura 9: Endereço do MAC de destino oriundo na resposta

9.3 Identifique a que sistemas correspondem os endereços MAC de origem e de destino da trama em causa, recorrendo aos comandos `ifconfig`, `netstat -rn` e `arp` executados no PC selecionado.

De forma a sabermos o endereço MAC de origem com a interação via *ping* entre *n1* e *n9*, executamos o comando `ifconfig`.

```
root@n1:/tmp/pycore.45783/n1.conf# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.107.20 netmask 255.255.255.0 broadcast 0.0.0.0
    inet6 fe80::200:ff:feaa:0 prefixlen 64 scopeid 0x20<link>
    inet6 2001::20 prefixlen 64 scopeid 0x0<global>
    ether 00:00:00:aa:00:00 txqueuelen 1000 (Ethernet)
    RX packets 153 bytes 14868 (14.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 20 bytes 1732 (1.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@n1:/tmp/pycore.45783/n1.conf#
```

Figura 10: Identificação do endereço MAC de origem com `ifconfig`

Aqui, podemos identificar que o endereço MAC de origem é 00:00:00:aa:00:00. Este comando mostra informações sobre as interfaces de rede do PC, incluindo o endereço MAC da interface.

De seguida, para obtermos o endereço MAC de destino na mesma interação, executamos os comandos `arp` (figura 12) e `netstat -nr` (figura 11).

```
root@n1:/tmp/pycore.45783/n1.conf# netstat -nr
Kernel IP routing table
Destination    Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0        192.168.107.1  0.0.0.0         UG      0 0        0 eth0
192.168.107.0  0.0.0.0        255.255.255.0   U       0 0        0 eth0

root@n1:/tmp/pycore.45783/n1.conf#
```

Figura 11: Tabela de encaminhamento de *n1*

Na figura 11, conseguimos identificar que todo o tráfego a sair do departamento A vai para o nó 192.168.107.1.

```
root@n1:/tmp/pycore.45783/n1.conf# arp
Address          HWtype  HWaddress          Flags Mask          Iface
192.168.107.1    ether   00:00:00:aa:00:08  C                 eth0

root@n1:/tmp/pycore.45783/n1.conf#
```

Figura 12: Identificação do endereço MAC de destino com `arp`

Já observando a figura 12, conseguimos identificar o *next hop* observada na tabela de encaminhamento de n1 (192.168.107.1) e identificamos que o endereço MAC de destino é 00:00:00:aa:00:08.

9.4 Justifique o modo de comunicação (unicast vs. broadcast) usado no envio da resposta ARP (ARP Reply)

A comunicação unicast consiste na comunicação ponto a ponto onde é enviada uma mensagem de uma única fonte para um único destino; já a comunicação broadcast consiste no envio de uma mensagem de uma só fonte para vários destinos.

Na resposta ARP reply, esta mensagem é unicast e não feita em broadcast, uma vez que a fonte que está a enviar a resposta ARP para o destino sabe o seu endereço Ethernet/MAC.

10 Verifique se o ping feito ao segundo PC originou pacotes ARP. Justifique a situação observada.

31	25.811669696	00:00:00_aa:00:08	00:00:00_aa:00:00	ARP	42 Who has 192.168.107.20? Tell 192.168.107.1
32	25.811680766	00:00:00_aa:00:00	00:00:00_aa:00:08	ARP	42 192.168.107.20 is at 00:00:00:aa:00:00
▶ Frame 31: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface veth1.0.65, id 0 ▶ Ethernet II, Src: 00:00:00_aa:00:08 (00:00:00:aa:00:08), Dst: 00:00:00_aa:00:00 (00:00:00:aa:00:00)					
▼ 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:00:08 (00:00:00:aa:00:08)					
Sender IP address: 192.168.107.1					
Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)					
Target IP address: 192.168.107.20					

Figura 13: Visualização de mensagens ARP com o segundo ping

De acordo com a figura 13, apareceram mensagens ARP no segundo ping, mas desta vez já não ocorreu *broadcast* para saber o endereço MAC de destino pois ele já consta na tabela ARP do router n6.

11 Identifique na mensagem ARP os campos que permitem definir o tipo e o tamanho dos endereços das camadas de rede e de ligação lógica que se pretendem mapear. Justifique os valores apresentados nesses campos.

Para respondermos a esta alínea, iremos utilizar o ARP Request para analisarmos o tipo e o tamanho dos endereços das camadas de rede e de ligação lógica que se pretendem mapear.

No.	Time	Source	Destination	Protocol	Length	Info
21	20.681817849	00:00:00_aa:00:00	Broadcast	ARP	42	Who has 192.168.107.1? Tell 192.168.107.20
32	25.811680766	00:00:00_aa:00:00	00:00:00_aa:00:00	ARP	42	192.168.107.20 is at 00:00:00_aa:00:00
22	20.682204522	00:00:00_aa:00:00	00:00:00_aa:00:00	ARP	42	192.168.107.1 is at 00:00:00_aa:00:00
31	25.811669696	00:00:00_aa:00:00	00:00:00_aa:00:00	ARP	42	Who has 192.168.107.20? Tell 192.168.107.1

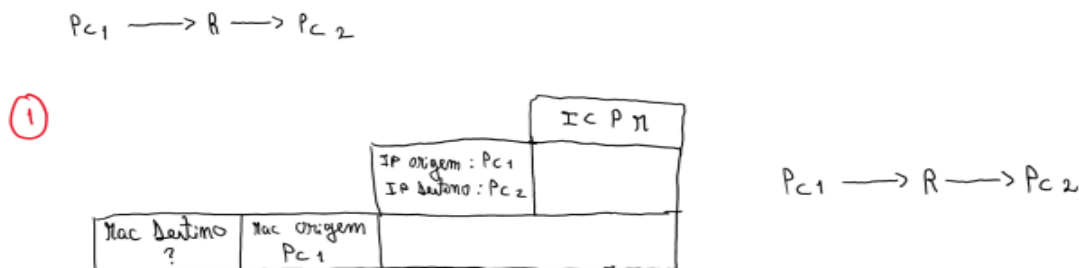
▶ Frame 21: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface veth1.0.65, id 0
 ▶ Ethernet II, Src: 00:00:00_aa:00:00 (00:00:00_aa:00:00), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
 ▼ Address Resolution Protocol (request)
 Hardware type: Ethernet (1)
 Protocol type: IPv4 (0x0800)
 Hardware size: 6
 Protocol size: 4
 Opcode: request (1)
 Sender MAC address: 00:00:00_aa:00:00 (00:00:00_aa:00:00)
 Sender IP address: 192.168.107.20
 Target MAC address: 00:00:00_00:00:00 (00:00:00_00:00:00)
 Target IP address: 192.168.107.1

Figura 14: Tipo e tamanho de endereços na mensagem ARP

Através da figura 14, podemos observar que:

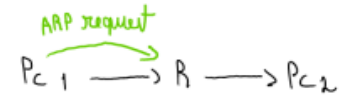
- O tipo de endereços da camada de rede é dado pelo campo *Protocol Type* é IPv4 (0x0800);
- O tamanho dos endereços da camada de rede é dado pelo campo *Protocol Size* é 4, que representa o nº de octetos para representar um endereço IPv4 ($4 \times 8 = 32$ bits);
- O tipo de endereços da camada de ligação lógica é dado pelo campo *Hardware type* é Ethernet (1);
- O tamanho dos endereços da camada de ligação lógica é dado pelo campo *Hardware Size* é 6, que representa o nº de octetos para representar um endereço Ethernet ($6 \times 8 = 48$ bits);

- 12 Na situação em que efetua um ping a um PC não local à sua sub-rede, 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 sistema destino (represente apenas os nós intervenientes). Assuma que todas as tabelas ARP se encontram inicialmente vazias.



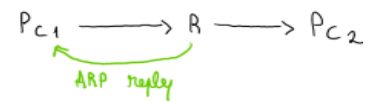
②

Mac destino FF:FF:FF:FF:FF:FF (broadcast)	Mac Origem PC1	Mac origem PC1	Mac destino 00:00:00:00:00:00
		IP origem PC1	IP destino R



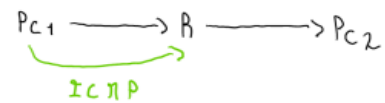
③

		Mac origem R	Mac destino PC1
		IP origem R	IP destino PC1
Mac destino PC1	Mac origem R		



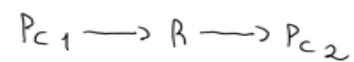
④

		ICMP	
		IP origem: PC ₁ IP destino: PC ₂	
Mac destino A	Mac origem PC ₁		

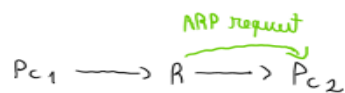
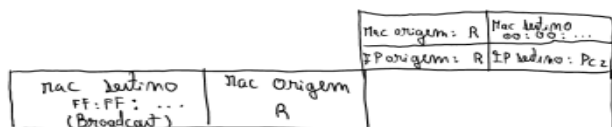


⑤

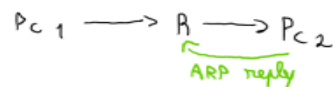
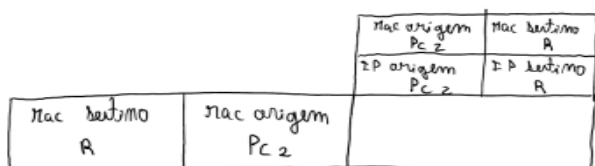
		ICMP	
		IP origem: PC ₁	
		IP destino: PC ₂	
mac destino ?	mac origem A		



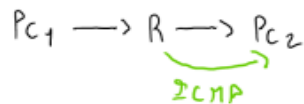
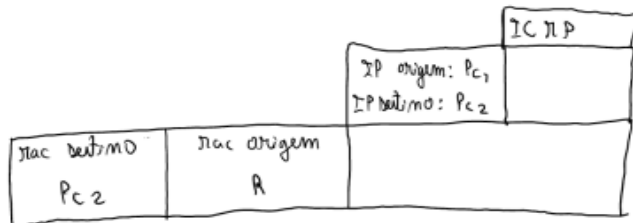
⑥



⑦



⑧



Notas:

- com o objetivo de tornar o esquema mais legível, optamos por não incluir o comprimento/tipo na parte da Ethernet (DST + Origem).
- PC1 corresponde a um dos nós do departamento A (ex: n1), R corresponde ao router de acesso do departamento A (n6) e PC2 corresponde a um nó do departamento B (ex: n8).

Domínios de colisão

- 13 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 é gerado 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.

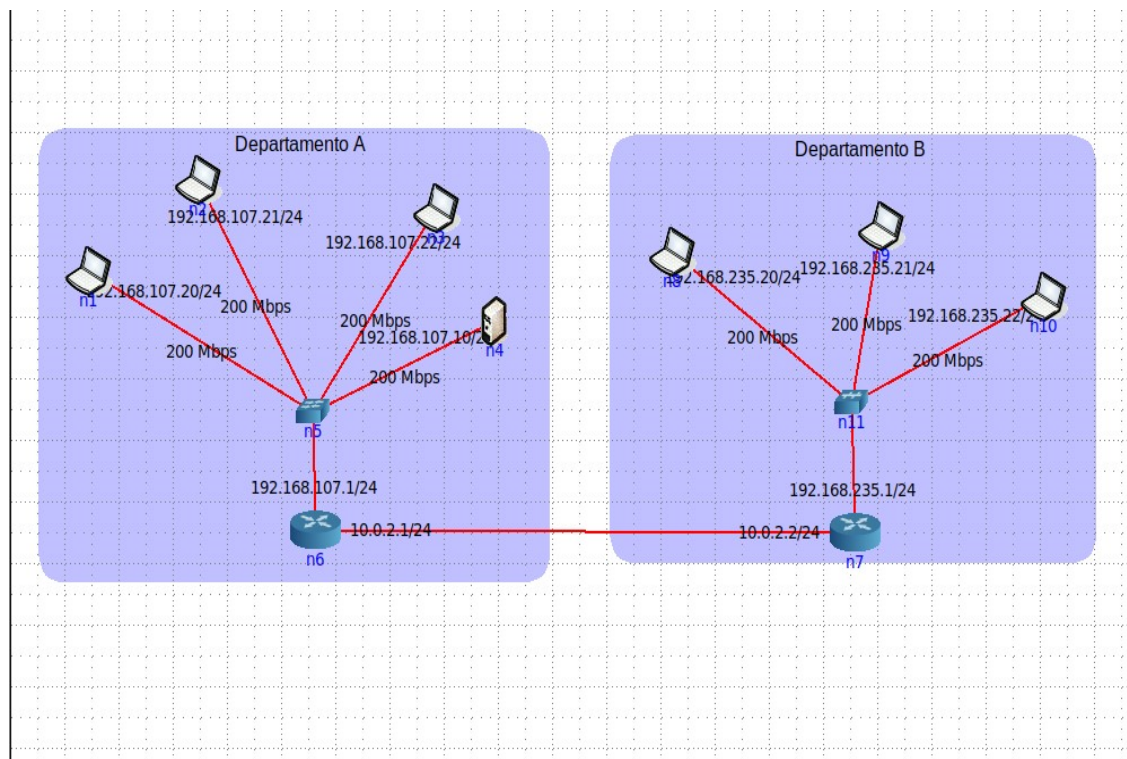
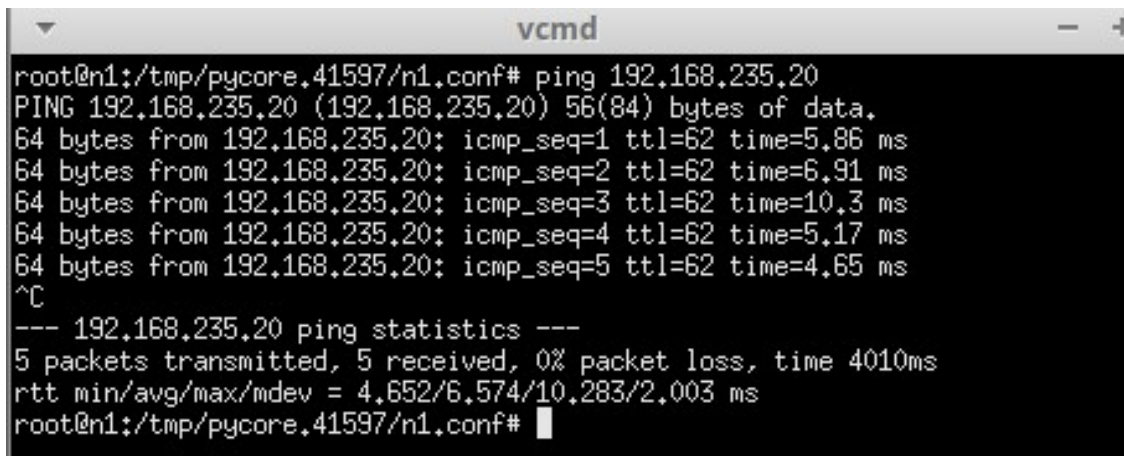


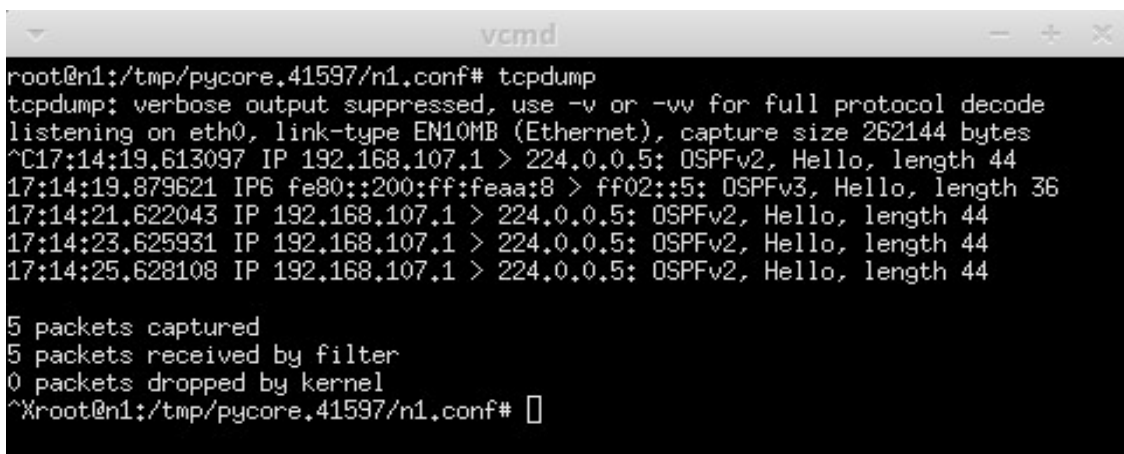
Figura 15: Topologia usada

Foi feito um ping entre os dois departamentos, para verificar se toda esta topologia estava bem conectada.



```
vcmd
root@n1:/tmp/pycore.41597/n1.conf# ping 192.168.235.20
PING 192.168.235.20 (192.168.235.20) 56(84) bytes of data:
64 bytes from 192.168.235.20: icmp_seq=1 ttl=62 time=5.86 ms
64 bytes from 192.168.235.20: icmp_seq=2 ttl=62 time=6.91 ms
64 bytes from 192.168.235.20: icmp_seq=3 ttl=62 time=10.3 ms
64 bytes from 192.168.235.20: icmp_seq=4 ttl=62 time=5.17 ms
64 bytes from 192.168.235.20: icmp_seq=5 ttl=62 time=4.65 ms
^C
--- 192.168.235.20 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4010ms
rtt min/avg/max/mdev = 4.652/6.574/10.283/2.003 ms
root@n1:/tmp/pycore.41597/n1.conf#
```

Figura 16: Ping entre n1 (departamento A) e n8 (departamento B)



```
vcmd
root@n1:/tmp/pycore.41597/n1.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
^C17:14:19.613097 IP 192.168.107.1 > 224.0.0.5: OSPFv2, Hello, length 44
17:14:19.879621 IP6 fe80::200:ff:feaa:8 > ff02::5: OSPFv3, Hello, length 36
17:14:21.622043 IP 192.168.107.1 > 224.0.0.5: OSPFv2, Hello, length 44
17:14:23.625931 IP 192.168.107.1 > 224.0.0.5: OSPFv2, Hello, length 44
17:14:25.628108 IP 192.168.107.1 > 224.0.0.5: OSPFv2, Hello, length 44

5 packets captured
5 packets received by filter
0 packets dropped by kernel
^Xroot@n1:/tmp/pycore.41597/n1.conf#
```

Figura 17: Comando "tcpdump"feito no departamento A

```

root@n9:/tmp/pycore.36433/n9.conf#
root@n9:/tmp/pycore.36433/n9.conf#
root@n9:/tmp/pycore.36433/n9.conf#
root@n9:/tmp/pycore.36433/n9.conf#
root@n9:/tmp/pycore.36433/n9.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
^C18:35:30.869906 IP 192.168.235.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:35:32.871572 IP 192.168.235.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:35:34.876985 IP 192.168.235.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:35:35.729179 IP 192.168.235.22 > 192.168.107.20: ICMP echo request, id 37, seq 1, length 64
18:35:35.729891 IP 192.168.107.20 > 192.168.235.22: ICMP echo reply, id 37, seq 1, length 64
18:35:36.739457 IP6 fe80::200:ff:feaa:7 > ff02::5: OSPFv3, Hello, length 36
18:35:36.741427 IP 192.168.235.22 > 192.168.107.20: ICMP echo request, id 37, seq 2, length 64
18:35:36.742896 IP 192.168.107.20 > 192.168.235.22: ICMP echo reply, id 37, seq 2, length 64
18:35:36.878200 IP 192.168.235.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:35:37.744558 IP 192.168.235.22 > 192.168.107.20: ICMP echo request, id 37, seq 3, length 64
18:35:37.744932 IP 192.168.107.20 > 192.168.235.22: ICMP echo reply, id 37, seq 3, length 64
18:35:38.878999 IP 192.168.235.1 > 224.0.0.5: OSPFv2, Hello, length 44

12 packets captured
12 packets received by filter
0 packets dropped by kernel
root@n9:/tmp/pycore.36433/n9.conf#

```

Figura 18: Comando "tcpdump"feito no departamento B

Com o comando "tcpdump", podemos verificar como flui o tráfego. Como podemos verificar temos um switch, no departamento A e um hub no departamento B.

O switch verifica menos pacotes do que um hub porque, numa LAN comutada, cada porta do switch é um domínio de colisão que está separado e os pacotes são direcionados apenas para a porta de destino correta. O switch verifica a sua tabela de endereços para determinar a porta ao qual o pacote deve ser encaminhado e, de seguida, envia o pacote apenas para essa porta.

Por outro lado, numa LAN partilhada com um hub, este simplesmente retransmite todos os pacotes que recebe em todas as portas, independentemente do destino, o que significa que todos os dispositivos conectados ao hub têm que verificar todos os pacotes, mesmo aqueles que não são destinados a eles. Isso pode levar a um congestionamento de rede desnecessário e atrasos na entrega de pacotes.

Dessa forma, o switch é mais eficiente em sua operação, pois envia os pacotes apenas para a porta de destino correta, reduzindo a quantidade de tráfego desnecessário na rede e tornando a transmissão de dados mais rápida e eficiente. Já o hub é menos eficiente, pois transmite todos os pacotes para todas as portas, aumentando a quantidade de tráfego e diminuindo a eficiência da rede.

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

Para a elaboração da tabela de comutação obtivemos os seguintes endereços MAC do Departamento A:

```

root@n6:/tmp/pycore.34851/n6.conf# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.107.1 netmask 255.255.255.0 broadcast 0.0.0.0
    inet6 fe80::200:ff:feaa:8 prefixlen 64 scopeid 0x20<link>
    inet6 2001::1 prefixlen 64 scopeid 0x0<global>
    ether 00:00:00:aa:00:08 txqueuelen 1000 (Ethernet)
    RX packets 113 bytes 10427 (10.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 413 bytes 33674 (33.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

Figura 19: Endereço MAC do router n6

22	15.747144664	00:00:00_aa:00:00	00:00:00_aa:00:08	ARP	42 Who has 192.168.107.1? Tell 192.168.107.20
▶ Frame 22: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface veth1.0.ab, id 0					
▶ Ethernet II, Src: 00:00:00_aa:00:00 (00:00:00:aa:00:00), Dst: 00:00:00_aa:00:08 (00:00:00:aa:00:08)					
▶ Address Resolution Protocol (request)					

Figura 20: Endereço MAC de n1

18	8.448727055	00:00:00_aa:00:01	00:00:00_aa:00:08	ARP	42 Who has 192.168.107.1? Tell 192.168.107.21
▶ Frame 18: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface veth2.0.ab, id 0					
▶ Ethernet II, Src: 00:00:00_aa:00:01 (00:00:00:aa:00:01), Dst: 00:00:00_aa:00:08 (00:00:00:aa:00:08)					
▶ Address Resolution Protocol (request)					

Figura 21: Endereço MAC de n2

4	4.720165037	00:00:00_aa:00:02	Broadcast	ARP	42 Who has 192.168.107.1? Tell 192.168.107.22
▶ Frame 4: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface veth3.0.ab, id 0					
▶ Ethernet II, Src: 00:00:00_aa:00:02 (00:00:00:aa:00:02), Dst: Broadcast (ff:ff:ff:ff:ff:ff)					
▶ Address Resolution Protocol (request)					

Figura 22: Endereço MAC de n3

4	5.156703828	00:00:00_aa:00:03	Broadcast	ARP	42 Who has 192.168.107.1? Tell 192.168.107.10
▶ Frame 4: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface veth4.0.ab, id 0					
▶ Ethernet II, Src: 00:00:00_aa:00:03 (00:00:00:aa:00:03), Dst: Broadcast (ff:ff:ff:ff:ff:ff)					
▶ Address Resolution Protocol (request)					

Figura 23: Endereço MAC de n4

Obtivemos a seguinte tabela:

Interface	Endereço MAC	TTL
n1	00:00:00:aa:00:00	63
n2	00:00:00:aa:00:01	63
n3	00:00:00:aa:00:02	63
n4	00:00:00:aa:00:03	63
n6	00:00:00:aa:00:08	63

Conclusões

A realização deste trabalho prático #3 permitiu que consolidássemos assuntos abordados nas aulas teóricas e práticas, nomeadamente os endereços MAC e Ethernet, o protocolo ARP e domínios de colisão. Permitiu, também, ganhar experiência (tal como em trabalhos práticos anteriores) na utilização do Wireshark e CORE, onde analisámos várias tramas Ethernet capturadas e analisamos a utilização de switches e hubs na diminuição de colisões de tramas Ethernet.