

---

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

---

TRABALHO REALIZADO POR:

BEATRIZ RIBEIRO MONTEIRO  
CARLOS EDUARDO CULOLO DANTAS DA COSTA  
HUGO RICARDO MACEDO GOMES



A95437  
Beatriz Monteiro



A88551  
Carlos Costa



A96842  
Hugo Gomes

# Índice

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Captura e análise de Tramas Ethernet</b>	<b>1</b>
2.1	Exercício 1 . . . . .	1
2.2	Exercício 2 . . . . .	1
2.3	Exercício 3 . . . . .	2
2.4	Exercício 4 . . . . .	3
2.5	Exercício 5 . . . . .	3
2.6	Exercício 6 . . . . .	3
<b>3</b>	<b>Protocolo ARP</b>	<b>4</b>
3.1	Exercício 1 . . . . .	5
3.2	Exercício 2 . . . . .	5
3.3	Exercício 3 . . . . .	6
3.4	Exercício 4 . . . . .	8
3.5	Exercício 5 . . . . .	8
3.6	Exercício 6 . . . . .	8
<b>4</b>	<b>Domínios de Colisão</b>	<b>9</b>
4.1	Exercício 1 . . . . .	9
4.2	Exercício 2 . . . . .	9
<b>5</b>	<b>Conclusão</b>	<b>10</b>

## List of Figures

1	Tráfego . . . . .	1
2	Captura da trama que contém a mensagem de acesso ao servidor . . . . .	1
3	Captura da trama 135 . . . . .	2
4	Protocolo TCP . . . . .	2
5	Protocolo IPv4 . . . . .	3
6	Captura da trama 137 que contém o primeiro <i>byte</i> da resposta HTTP . . . . .	3
7	Protocolos contidos na trama 137 . . . . .	4
8	Topologia a ser utilizada . . . . .	4
9	Comando <i>arp</i> . . . . .	5
10	Trama Ethernet com o ARP Request . . . . .	5
11	Trama Ethernet com o ARP Reply . . . . .	6
12	ifconfig feito . . . . .	7
13	netstat -rn e arp feitos . . . . .	7
14	Diagrama com as mensagens ARP e ICMP trocadas . . . . .	8
15	Ping e TCPdump no A . . . . .	9
16	Tabela de Comutação do Switch . . . . .	9

# 1 Introdução

O objetivo deste trabalho é explorar a camada de ligação lógica, focando o uso da tecnologia Ethernet e do protocolo ARP (*Address Resolution Protocol*), bem como termos uma melhor sensibilidade em funcionalidades de serviço como detecção e correção de erros, protocolos de acesso de controlo de ligação, endereços MAC e interligação de redes locais.

## 2 Captura e análise de Tramas Ethernet

Responda às perguntas seguintes com base no conteúdo da trama Ethernet que contém a mensagem de acesso ao servidor (HTTP GET encriptada).

### 2.1 Exercício 1

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

O endereço MAC de origem é 0e:0a:f6:54:31:23 e o endereço MAC de destino é 00:d0:03:ff:94:00. O primeiro é um endereço físico da máquina onde foi executado o que foi pedido no enunciado, enquanto o segundo refere-se ao endereço físico do *router*.

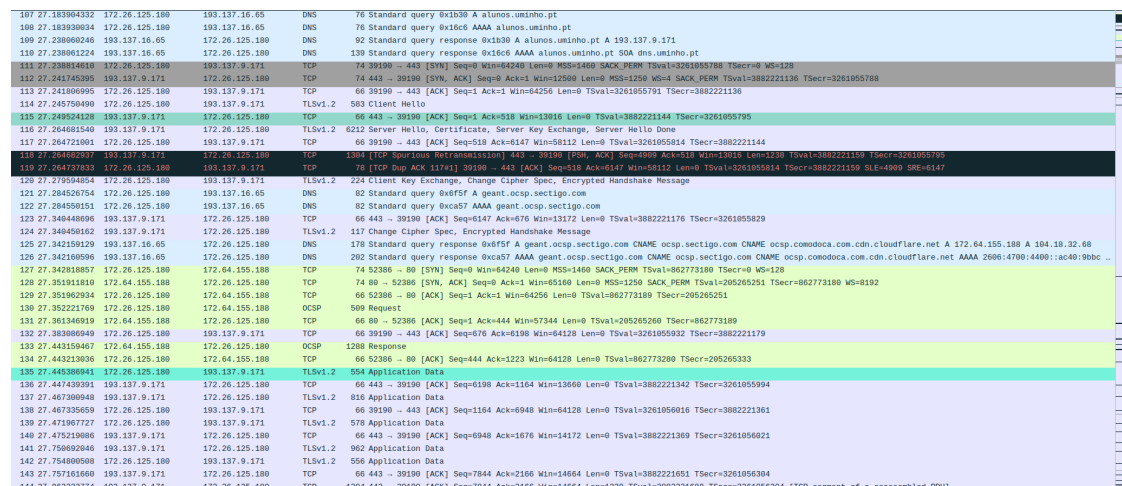


Figure 1: Tráfego

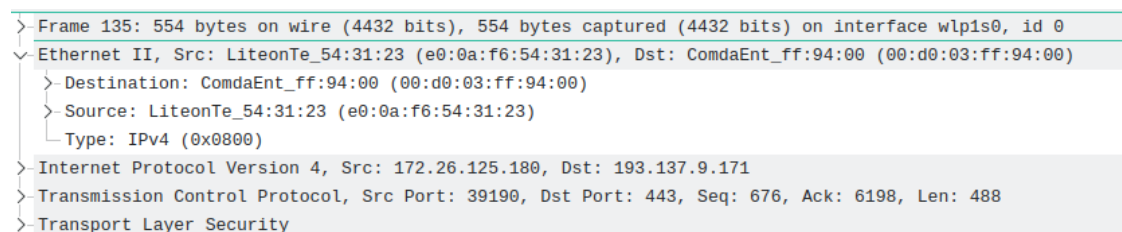


Figure 2: Captura da trama que contém a mensagem de acesso ao servidor

### 2.2 Exercício 2

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

---

O valor hexadecimal do campo *Type* da trama Ethernet é, como é possível visualizar na figura 2 é 0x0800. O campo *Type* de uma trama Ethernet indica qual é o protocolo que está a ser utilizado na camana superior, neste caso, o protocolo é o IPv4.

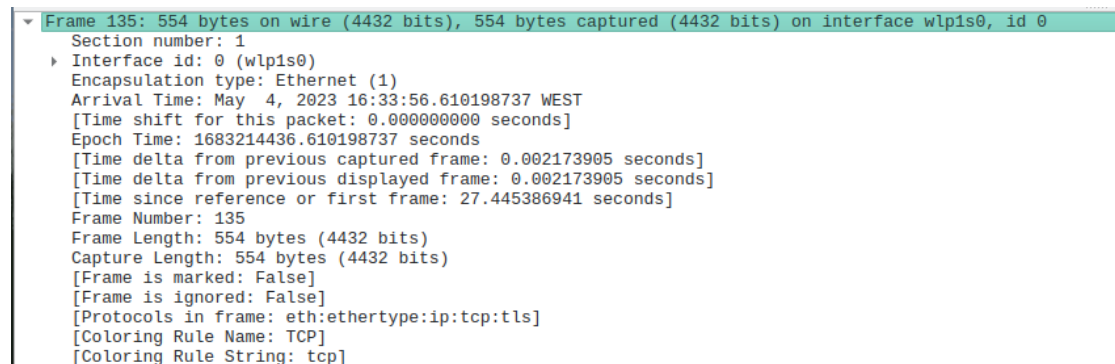
### 2.3 Exercício 3

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.

O tamanho do cabeçalho cabeçalho Ethernet é de 14 *bytes*, o cabeçalho do IPv4 tem 20 *bytes*, enquanto o cabeçalho do TCP é de 32 *bytes*. Podemos concluir estes 2 últimos pela observação das imagens 4 e 5.

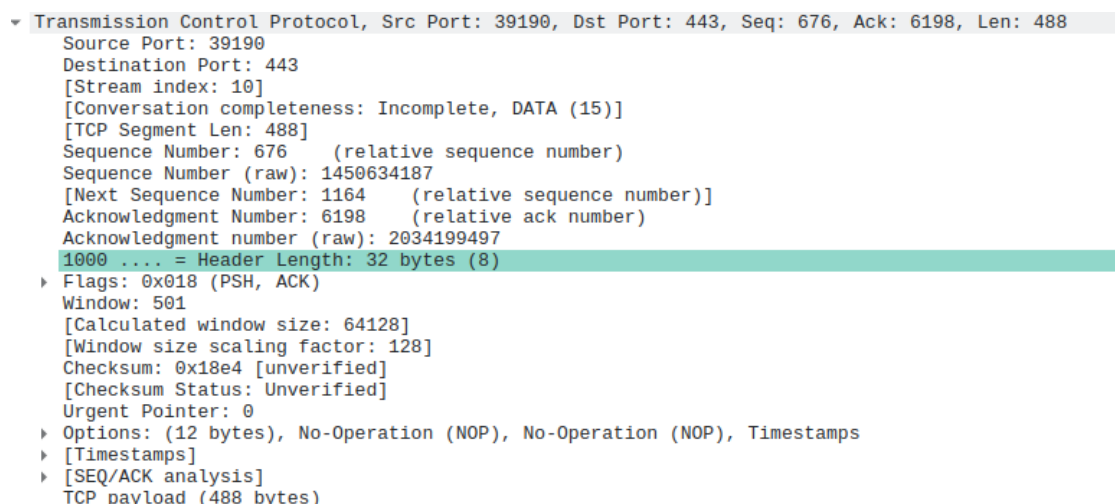
O nº de *bytes* usados no encapsulamento protocolar é  $14+20+32 = 66$  *bytes*.

O tamanho total total do pacote é de 554 *bytes*, como podemos ver na figura 3. Logo, o *overhead* introduzido pela pilha protocolar é  $\frac{66}{554} = 11.91\%$ .



```
▼ Frame 135: 554 bytes on wire (4432 bits), 554 bytes captured (4432 bits) on interface wlp1s0, id 0
  Section number: 1
  ▶ Interface id: 0 (wlp1s0)
  Encapsulation type: Ethernet (1)
  Arrival Time: May  4, 2023 16:33:56.610198737 WEST
  [Time shift for this packet: 0.000000000 seconds]
  Epoch Time: 1683214436.610198737 seconds
  [Time delta from previous captured frame: 0.002173905 seconds]
  [Time delta from previous displayed frame: 0.002173905 seconds]
  [Time since reference or first frame: 27.445386941 seconds]
  Frame Number: 135
  Frame Length: 554 bytes (4432 bits)
  Capture Length: 554 bytes (4432 bits)
  [Frame is marked: False]
  [Frame is ignored: False]
  [Protocols in frame: eth:ethertype:ip:tcp:tls]
  [Coloring Rule Name: TCP]
  [Coloring Rule String: tcp]
```

Figure 3: Captura da trama 135



```
▼ Transmission Control Protocol, Src Port: 39190, Dst Port: 443, Seq: 676, Ack: 6198, Len: 488
  Source Port: 39190
  Destination Port: 443
  [Stream index: 10]
  [Conversation completeness: Incomplete, DATA (15)]
  [TCP Segment Len: 488]
  Sequence Number: 676 (relative sequence number)
  Sequence Number (raw): 1450634187
  [Next Sequence Number: 1164 (relative sequence number)]
  Acknowledgment Number: 6198 (relative ack number)
  Acknowledgment number (raw): 2034199497
  1000 .... = Header Length: 32 bytes (8)
  ▶ Flags: 0x018 (PSH, ACK)
  Window: 501
  [Calculated window size: 64128]
  [Window size scaling factor: 128]
  Checksum: 0x18e4 [unverified]
  [Checksum Status: Unverified]
  Urgent Pointer: 0
  ▶ Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
  ▶ [Timestamps]
  ▶ [SEQ/ACK analysis]
  TCP payload (488 bytes)
```

Figure 4: Protocolo TCP

---

```

▼ Internet Protocol Version 4, Src: 172.26.125.180, Dst: 193.137.9.171
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  ▶ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 540
    Identification: 0x730f (29455)
  ▶ 010. .... = Flags: 0x2, Don't fragment
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 64
    Protocol: TCP (6)
    Header Checksum: 0xd0c9 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 172.26.125.180
    Destination Address: 193.137.9.171

```

Figure 5: Protocolo IPv4

A seguir responda às seguintes perguntas, baseado no conteúdo da trama Ethernet que contém o primeiro *byte* da resposta HTTP proveniente do servidor.

```

> Frame 137: 816 bytes on wire (6528 bits), 816 bytes captured (6528 bits) on interface wlp1s0, id 0
▼ Ethernet II, Src: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00), Dst: LiteonTe_54:31:23 (e0:0a:f6:54:31:23)
  > Destination: LiteonTe_54:31:23 (e0:0a:f6:54:31:23)
  > Source: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
  └ Type: IPv4 (0x0800)
> Internet Protocol Version 4, Src: 193.137.9.171, Dst: 172.26.125.180
> Transmission Control Protocol, Src Port: 443, Dst Port: 39190, Seq: 6198, Ack: 1164, Len: 750
> Transport Layer Security

```

Figure 6: Captura da trama 137 que contém o primeiro *byte* da resposta HTTP

## 2.4 Exercício 4

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

Pela figura 6 podemos constatar que o endereço da fonte é 00:d0:03:ff:94:00 que se trata de um endereço físico de uma interface do *router*.

## 2.5 Exercício 5

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

Como é possível na figura 6, o endereço MAC do destino, desta vez, é 0e:0a:f6:54:31:23, pois trata-se do endereço da nossa máquina.

## 2.6 Exercício 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.

Na figura 7 podemos ver que foram utilizados os seguintes protocolos:

**Ethernet**

**IPv4** - Internet Protocol Version 4

**TCP** - Transmission Control Protocol

**TLS** - Transport Layer Security

**HTTP** - Hypertext Transfer Protocol (que é encriptado pelo TLS na camada superior)

```

▶ Frame 137: 816 bytes on wire (6528 bits), 816 bytes captured (6528 bits) on interface wlp1s0, id 0
▼ Ethernet II, Src: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00), Dst: LiteonTe_54:31:23 (e0:0a:f6:54:31:23)
  ▶ Destination: LiteonTe_54:31:23 (e0:0a:f6:54:31:23)
  ▶ Source: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)
  Type: IPv4 (0x0800)
▼ Internet Protocol Version 4, Src: 193.137.9.171, Dst: 172.26.125.180
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  ▶ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 802
  Identification: 0x11ea (4586)
  ▶ 010. .... = Flags: 0x2, Don't fragment
  ...0 0000 0000 0000 = Fragment Offset: 0
  Time to Live: 252
  Protocol: TCP (6)
  Header Checksum: 0x74e8 [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 193.137.9.171
  Destination Address: 172.26.125.180
▶ Transmission Control Protocol, Src Port: 443, Dst Port: 39190, Seq: 6198, Ack: 1164, Len: 750
▼ Transport Layer Security
  ▼ TLSv1.2 Record Layer: Application Data Protocol: Hypertext Transfer Protocol
    Content Type: Application Data (23)
    Version: TLS 1.2 (0x0303)
    Length: 745
    Encrypted Application Data: 4be9afb4b1649b7a6a4c5564b6006ca9f2927fd505b3839dbfa300507e7b5979df7cdaa7...
    [Application Data Protocol: Hypertext Transfer Protocol]
```

Figure 7: Protocolos contidos na trama 137

Conseguimos perceber que eram estes os protocolos que estão contidos na trama que se encontra na imagem acima, tanto pela observação direta dos campos que contém, bem como através de uma observação mais cuidadosa dos diferentes campos do protocolos que se encontram destacados na imagem da trama 137. Por vezes, os protocolos indicam qual o protocolo que será usado na camada abaixo, como é o caso do Ethernet, IPv4, TCP e TLS.

### 3 Protocolo ARP

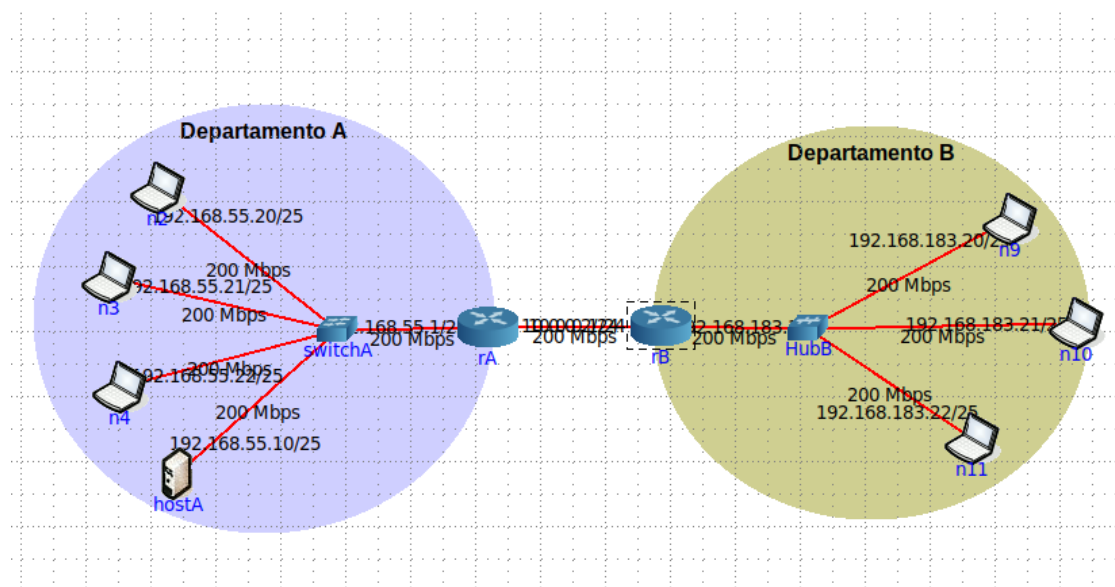


Figure 8: Topologia a ser utilizada

---

### 3.1 Exercício 1

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

```
root@n2:/tmp/pycore.42121/n2.conf# arp -a
? (192.168.55.1) at 00:00:00:aa:00:00 [ether] on eth0
```

Figure 9: Comando *arp*

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

Ao executarmos o comando *arp -a* obtivemos a tabela ARP apresentada na figura 9. Na primeira coluna temos o endereço IP do *router* que permite o departamento A com o exterior, na segunda coluna temos o seu endereço MAC.

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

Considerando toda a topologia, o equipamento que deverá apresentar um maior número de entradas na tabela ARP deverá ser o *router* rB, pois foi este o *router* que teve que comunicar com mais equipamentos, isto é, depois de executar os 2 *pings* para 2 PCs do departamento B, o n2 comunica unicamente com o rA, o rA com o n2 e o rB, enquanto este último comunica com o rA, o n10 e o n11.

### 3.2 Exercício 2

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

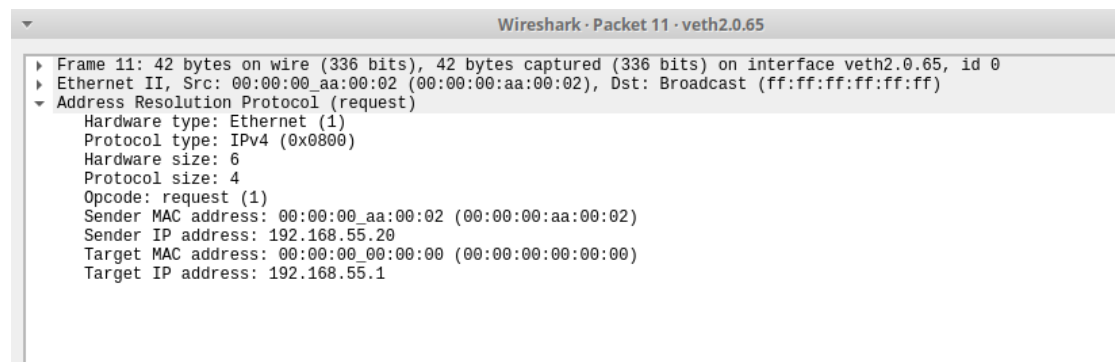


Figure 10: Trama Ethernet com o ARP Request

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

A fonte é 192.168.55.20 (00:00:00\_aa:00:02), e o destino é o ff:ff:ff:ff:ff:ff (Broadcast). Utiliza-se a rede como destino, pois não sabendo o equipamento ao qual corresponde o endereço IP que procuramos, o host de origem questiona todos os equipamentos na rede, esperando que o equipamento procurado lhe responda.

- 
- b. Qual o valor hexadecimal do campo Tipo da trama Ethernet? O que indica?

O campo *type* tem valor (0x0806), o que indica o uso do protocolo ARP.

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

A trama tem o campo *type* ARP (0x0806) e também endereça a mensagem a todos os equipamentos na rede, realiza *broadcast*.

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

O *host* de origem pergunta à rede quem "é" o equipamento com o endereço IP indicado. Todos os equipamentos na rede recebem então essa mensagem e o equipamento com o endereço IP procurado responde com o seu endereço MAC diretamente ao *host* de origem.

### 3.3 Exercício 3

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

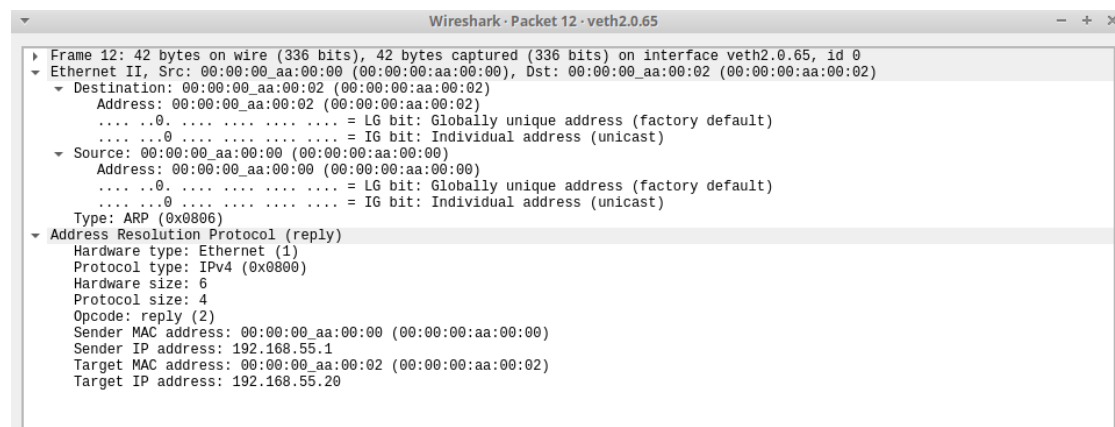


Figure 11: Trama Ethernet com o ARP Reply

- a. Qual o valor do campo ARP *opcode*? O que especifica?

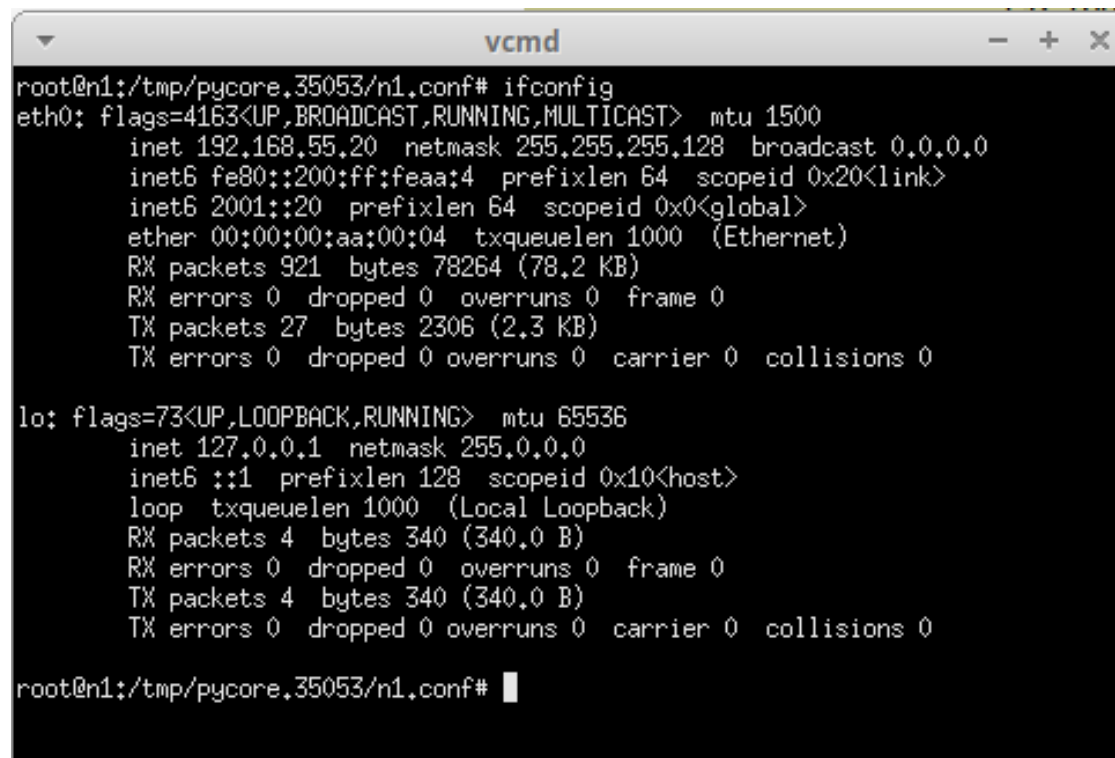
O valor é 2, valor este que indica uma *reply*.

- b. Em que posição da mensagem ARP está a resposta ao pedido ARP efetuado?

No cabeçalho Ethernet, incluem-se 3 informações - o destinatário, o emissor da mensagem, e o tipo. A segunda, o emissor é o equipamento procurado pelo *host* que emitiu o pedido ARP, que por sua vez, este envia o seu endereço ao *host* que o procurava.



- c. 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.

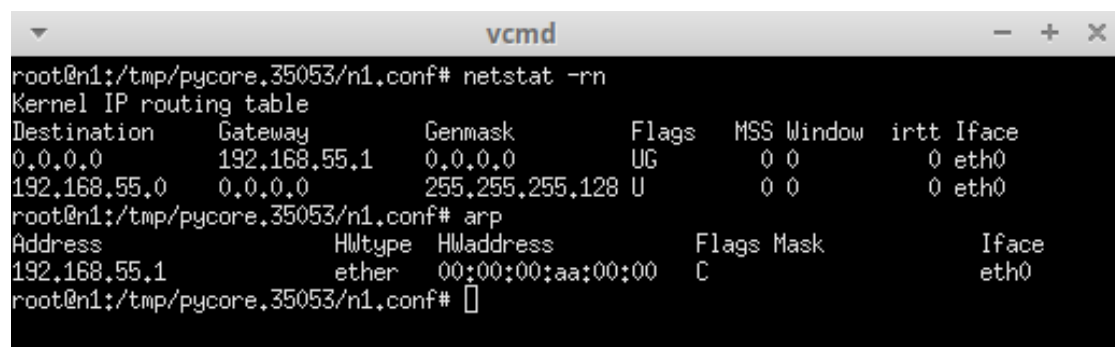


```
root@n1:/tmp/pycore.35053/n1.conf# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.55.20 netmask 255.255.255.128 broadcast 0.0.0.0
    inet6 fe80::200:ff:feaa:4 prefixlen 64 scopeid 0x20<link>
    inet6 2001::20 prefixlen 64 scopeid 0x0<global>
    ether 00:00:00:aa:00:04 txqueuelen 1000 (Ethernet)
    RX packets 921 bytes 78264 (78,2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 27 bytes 2306 (2,3 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 4 bytes 340 (340,0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 4 bytes 340 (340,0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@n1:/tmp/pycore.35053/n1.conf#
```

Figure 12: ifconfig feito



```
root@n1:/tmp/pycore.35053/n1.conf# netstat -rn
Kernel IP routing table
Destination    Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0        192.168.55.1   0.0.0.0         UG      0 0        0 eth0
192.168.55.0   0.0.0.0        255.255.255.128 U        0 0        0 eth0
root@n1:/tmp/pycore.35053/n1.conf# arp
Address          HWtype  HWaddress      Flags Mask    Iface
192.168.55.1     ether   00:00:00:aa:00:00 C              eth0
root@n1:/tmp/pycore.35053/n1.conf#
```

Figure 13: netstat -rn e arp feitos

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

Como no *ARP Request* o emissor envia o seu próprio endereço na trama para todos os equipamentos na rede (*broadcast*), o equipamento a ser procurado pelo emissor já sabe a quem enviar o seu próprio endereço. Sendo assim, é desnecessário enviar o seu endereço por *broadcast*, o endereço pode enviar diretamente por (*unicast*).

### 3.4 Exercício 4

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

Como o primeiro ping criou entrada na tabela ARP para o *router*, não é necessário criar mais pacotes com *pings* sucessivos porque o *host* já sabe o caminho a se dirigir ao *router*.

### 3.5 Exercício 5

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.

Pela figura 11, podemos ver que para a camada de rede o protocolo utilizado é o IPv4 e o tamanho dos endereços é de 4, enquanto o protocolo utilizado para a camada de ligação lógica é o Ethernet com tamanho dos endereços 6.

### 3.6 Exercício 6

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.

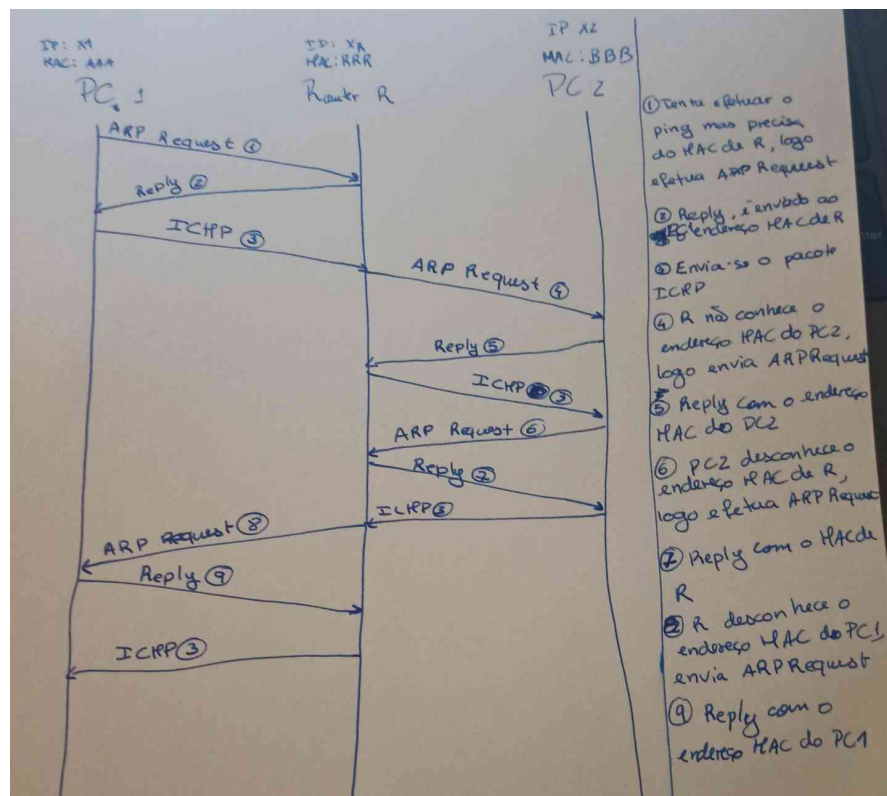


Figure 14: Diagrama com as mensagens ARP e ICMP trocadas

## 4 Domínios de Colisão

### 4.1 Exercício 1

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.

Foi feito um ping do n1 para o n3, e fez-se o *tcpdump* no *host* do departamento A. Verifica-se que a mensagem não é partilhada pelo switch.

No departamento B, fez se um ping do n9 para o n10 e fizemos o *tcpdump* no n11. Pelas mensagens ICMP *echo request* e ICMP *echo reply*, concluímos que a mensagem é partilhada pelo *hub* para o n11, pois não é um "interveniente direto" na conversa entre o n9 e o n10. Com isto, podemos verificar que um *hub* envia todas as mensagens que recebe para todas as interfaces da sua subrede enquanto que o *switch* faz isso apenas uma vez quando a sua tabela não está preenchida, ou seja, o *switch* "aprende" enquanto que o *hub* não.

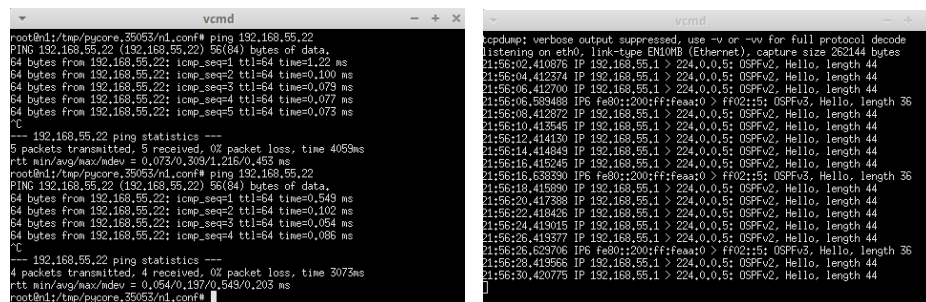


Figure 15: Ping e TCPdump no A

### 4.2 Exercício 2

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

MAC Address	Porta	Nome
2001:0::20/64		1 n2
2001:0::21/64		2 n3
2001:0::22/64		3 n4
2001:0::10/64		4 hostA
2001:0::1/64		5 rA

Figure 16: Tabela de Comutação do Switch

---

## 5 Conclusão

Neste trabalho prático foi-nos possível aprofundar o nosso conhecimento do conteúdo lecionado nas aulas e consolidar a nossa aprendizagem teórica. Pudemos conhecer melhor a Ethernet e o protocolo ARP e pôr em prática o que efetivamente absorvemos nas aulas teóricas. Na parte das colisões, conseguimos perceber a diferença entre *switches* e *hubs* e as implicações associadas ao uso de cada um desses equipamentos.