

# Relatório TP3

## Grupo 102

Diogo Cunha a100481

Rui Cerqueira a100537

Guilherme Rio a100898

### 3. Captura e análise de Tramas Ethernet

1	0.000000	20.54.232.160	172.26.71.17	TCP	60 443 → 53347 [ACK] Seq=1 Ack=1 Win=2050 Len=0
2	0.051717	20.54.232.160	172.26.71.17	TLSv1.2	544 Application Data
3	0.095375	172.26.71.17	20.54.232.160	TCP	54 53347 → 443 [ACK] Seq=1 Ack=491 Win=510 Len=0
4	0.086033	172.26.71.17	35.186.224.25	TCP	55 53156 → 443 [ACK] Seq=1 Ack=1 Win=500 Len=1 [TCP segment of a reassembled PDU]
5	0.703839	35.186.224.25	172.26.71.17	TCP	66 443 → 53156 [ACK] Seq=1 Ack=2 Win=261 Len=0 SLE=1 SRE=2
6	0.866073	162.159.133.234	172.26.71.17	TLSv1.2	137 Application Data
7	0.918818	172.26.71.17	162.159.133.234	TCP	54 53141 → 443 [ACK] Seq=1 Ack=84 Win=511 Len=0
8	0.999750	162.159.133.234	172.26.71.17	TLSv1.2	220 Application Data
9	1.042081	172.26.71.17	162.159.133.234	TCP	54 53141 → 443 [ACK] Seq=1 Ack=250 Win=511 Len=0
10	1.346711	172.26.71.17	34.223.124.45	HTTP	678 GET /online/ HTTP/1.1
11	1.376034	172.26.71.17	35.186.224.47	TLSv1.2	97 Application Data
12	1.405200	35.186.224.47	172.26.71.17	TCP	60 443 → 51939 [ACK] Seq=1 Ack=44 Win=267 Len=0
13	1.424120	35.186.224.47	172.26.71.17	TLSv1.2	94 Application Data
14	1.473205	172.26.71.17	35.186.224.47	TCP	54 51939 → 443 [ACK] Seq=44 Ack=41 Win=510 Len=0
15	1.558438	34.223.124.45	172.26.71.17	TCP	60 80 → 53339 [ACK] Seq=1 Ack=625 Win=220 Len=0
16	1.561960	34.223.124.45	172.26.71.17	TCP	1304 80 → 53339 [ACK] Seq=1 Ack=625 Win=220 Len=1250 [TCP segment of a reassembled PDU]
17	1.561960	34.223.124.45	172.26.71.17	HTTP	350 HTTP/1.1 200 OK (text/html)
18	1.561996	172.26.71.17	34.223.124.45	TCP	54 53339 → 80 [ACK] Seq=625 Ack=1547 Win=512 Len=0

```
0000 28 d0 ea 5f e5 42 00 d0 03 ff 94 00 08 00 45 00 (.._.B.. ..E-
0010 00 ce d5 c4 40 00 34 06 54 b0 a2 9f 85 ea ac 1a ...@.4. T.....
0020 47 11 01 bb cf 95 61 1b 51 6a 9d f5 c5 b0 50 18 G.....a. Qj....P.
0030 00 08 2b 1a 00 00 17 03 03 00 a1 88 b6 a9 7f 0e ..+.....
0040 f6 f6 9e f0 15 52 31 45 82 5c 49 35 54 f6 b2 c3 .....R1E .\IST...
0050 6d b2 86 22 53 cc a0 5e ae 94 d3 0d 64 c6 2f 20 m.."S..^ ....d./
0060 f7 a2 ca ab 73 39 59 24 00 33 ec 49 ec f9 9e ef ....$9Y$ .3.I....
0070 20 6a 8e 92 c5 8b 11 12 ed 55 16 67 62 47 fb 83 j.....-U.gbG..
0080 88 e4 b8 41 d4 c5 53 b0 69 cb 6a ee 40 0f 3a 6f ...A..S. i.j.@.:o
0090 c5 e4 93 f2 71 f9 f6 a3 a9 99 f2 ce b3 4e fc 4c ...q... ..N.L
00a0 bb 65 15 7b 40 f7 6f 85 98 23 2a 47 22 1e 2b 34 .e-{@.o- .#*G"+4
00b0 05 80 40 09 9f 2a 1d 24 ac 71 fb ad d6 a2 e2 31 .@..*.$ .q.....1
00c0 ac 1d 5f 54 bc b5 7d d5 8f f8 72 18 37 48 ec da .._T..}..r.7H..
00d0 3c 15 b2 a1 a5 2d ca 64 cf ae 9e a6 <.....d ....
```

20 bytes (internet protocol) + 20 bytes (TCP header) + 14 bytes(Ethernet II).

- Internet Protocol Version 4, Src: 162.159.133.234, Dst: 172.26.71.17
  - 0100 .... = Version: 4
  - .... 0101 = Header Length: 20 bytes (5)
- Transmission Control Protocol, Src Port: 443, Dst Port: 53141, Seq: 84, Ack: 1, Len: 166
  - Source Port: 443
  - Destination Port: 53141
  - [Stream index: 2]
  - [Conversation completeness: Incomplete (12)]
  - [TCP Segment Len: 166]
  - Sequence Number: 84 (relative sequence number)
  - Sequence Number (raw): 1629180266
  - [Next Sequence Number: 250 (relative sequence number)]
  - Acknowledgment Number: 1 (relative ack number)
  - Acknowledgment number (raw): 2650129840
  - 0101 .... = Header Length: 20 bytes (5)

.1 Destino: 28:d0:ea:5f:e5:42

Origem: 00:d0:03:ff:94:00

A origem é o nosso computador e o destino o router da rede local.

Ethernet II, Src: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00), Dst: IntelCor_5f:e5:42 (28:d0:ea:5f:e5:42)					
Destination: IntelCor_5f:e5:42 (28:d0:ea:5f:e5:42)					
Address: IntelCor_5f:e5:42 (28:d0:ea:5f:e5:42)					
....0. .... = LG bit: Globally unique address (factory default)					
....0. .... = IG bit: Individual address (unicast)					
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: IPv4 (0x0800)					
10 1.346711	172.26.71.17	34.223.124.45	HTTP	678 GET /online/ HTTP/1.1	

.2 Valor hexadecimal do campo Type: 0x0800, que representa o tipo de protocolo IPv4 utilizado.

.3

Ethernet II = 14 bytes, IPv4 = 20 bytes, TCP = 20 bytes

Total = 54 bytes.

$54 / 678 * 100 = 8\%$

00 d0 03 ff 94 00 28 d0 ea 5f e5 42 08 00 45 00	.....( . _ .B .E .
02 98 64 44 40 00 80 06 00 00 ac 1a 47 11 22 df	..dD@... ..G-".
7c 2d d0 5b 00 50 1d dd 77 4e 07 dc c6 db 50 18	--[-P...wN....P.
02 00 94 c2 00 00 47 45 54 20 2f 6f 6e 6c 69 6e	.....GE T /onlin
65 2f 20 48 54 54 50 2f 31 2e 31 0d 0a 48 6f 73	e/ HTTP/ 1.1..Hos
74 3a 20 71 75 69 65 74 6f 6c 64 67 6c 6f 77 69	t: quiet oldglowi
6e 67 76 65 72 73 65 2e 6e 65 76 65 72 73 73 6c	ngverse. neverssl
2e 63 6f 6d 0d 0a 43 6f 6e 6e 65 63 74 69 6f 6e	.com..Co nnection
3a 20 6b 65 65 70 2d 61 6c 69 76 65 0d 0a 43 61	: keep-a live..Ca
63 68 65 2d 43 6f 6e 74 72 6f 6c 3a 20 6d 61 78	che-Cont rol: max
2d 61 67 65 3d 30 0d 0a 55 70 67 72 61 64 65 2d	-age=0.. Upgrade-
49 6e 73 65 63 75 72 65 2d 52 65 71 75 65 73 74	Insecure -Request
73 3a 20 31 0d 0a 55 73 65 72 2d 41 67 65 6e 74	s: 1..Us er-Agent
3a 20 4d 6f 7a 69 6c 6c 61 2f 35 2e 30 20 28 57	: Mozill a/5.0 (v
69 6e 64 6f 77 73 20 4e 54 20 31 30 2e 30 3b 20	indows NT 10.0;
57 69 6e 36 34 3b 20 78 36 34 29 20 41 70 70 6c	Win64; x 64) AppI
65 57 65 62 4b 69 74 2f 35 33 37 2e 33 36 20 28	eWebKit/ 537.36 (
4b 48 54 4d 4c 2c 20 6c 69 6b 65 20 47 65 63 6b	KHTML, l ike Geck
6f 29 20 43 68 72 6f 6d 65 2f 31 31 32 2e 30 2e	o) Chrom e/112.0.
30 2e 30 20 53 61 66 61 72 69 2f 35 33 37 2e 33	0.0 Safa ri/537.3

.4 O endereço Ethernet da fonte é o endereço MAC 00:d0:03:ff:94:00, este corresponde com a NIC do servidor do website pois este pacote é a resposta ao HTTP GET, ou seja, trata-se do pacote enviado pelo servidor para a máquina que requisitou acesso pelo navegador.

17	1.561960	34.223.124.45	172.26.71.17	HTTP	350 HTTP/1.1 200 OK (text/html)
18	1.561996	172.26.71.17	34.223.124.45	TCP	54 53339 → 80 [ACK] Seq=625 Ack=1547 W
19	1.992721	162.159.133.234	172.26.71.17	TLSv1.2	142 Application Data
20	2.022261	172.26.71.17	162.159.133.234	TCP	54 53141 → 443 [ACK] Seq=1 Ack=338 Win

Ethernet II, Src: ComdaEnt\_ff:94:00 (00:d0:03:ff:94:00), Dst: IntelCor\_5f:e5:42 (28:d0:ea:5f:e5:42)

> Destination: IntelCor\_5f:e5:42 (28:d0:ea:5f:e5:42)

> Source: ComdaEnt\_ff:94:00 (00:d0:03:ff:94:00)

Type: IPv4 (0x0800)

Internet Protocol Version 4, Src: 34.223.124.45, Dst: 172.26.71.17

.5 O endereço MAC do destino é 28:d0:ea:5f:e5:42, este corresponde a máquina usada para acessar ao website, neste caso o nosso computador.

17	1.561960	34.223.124.45	172.26.71.17	HTTP	350 HTTP/1.1 200 OK (text/html)
18	1.561996	172.26.71.17	34.223.124.45	TCP	54 53339 → 80 [ACK] Seq=625 Ack=
19	1.992721	162.159.133.234	172.26.71.17	TLSv1.2	142 Application Data
20	2.022261	172.26.71.17	162.159.133.234	TCP	54 53141 → 443 [ACK] Seq=1 Ack=

> Frame 17: 350 bytes on wire (2800 bits), 350 bytes captured (2800 bits) on interface \Device\NPF\_{58B78EF0-7BDE-4DE}

> Ethernet II, Src: ComdaEnt\_ff:94:00 (00:d0:03:ff:94:00), Dst: IntelCor\_5f:e5:42 (28:d0:ea:5f:e5:42)

> Destination: IntelCor\_5f:e5:42 (28:d0:ea:5f:e5:42)

.6

## Identificadores do protocolo Ethernet II:

Endereço MAC destino:

Frame 17: 350 bytes on wire (2800 bits), 350 bytes captured (2800 bits) on interface \Device\NPF_{58B78EF0-7BDE-4DE}																
Ethernet II, Src: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00), Dst: IntelCor_5f:e5:42 (28:d0:ea:5f:e5:42)																
Destination: IntelCor_5f:e5:42 (28:d0:ea:5f:e5:42)																
Source: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)																
0000	28	d0	ea	5f	e5	42	00	00	03	ff	94	00	00	45	00	
0010	01	50	9d	ac	40	00	cf	06	7a	c3	22	df	7c	2d	ac	1a
0020	47	11	00	50	d0	5b	07	dc	cb	bd	1d	dd	79	be	50	18
0030	00	dc	ca	2f	00	00	04	b1	3e	7a	90	65	d0	12	01	5b

Endereço MAC fonte:

> Frame 17: 350 bytes on wire (2800 bits), 350 bytes captured (2800 bits) on interface \Device\NPF_{58B78EF0-7BDE-4DE}	0000	28	d0	ea	5f	e5	42	00	d0	03	ff	94	00	08	00	45	00
> Ethernet II, Src: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00), Dst: IntelCor_5f:e5:42 (28:d0:ea:5f:e5:42)	0010	01	50	9d	ac	40	00	cf	06	7a	c3	22	df	7c	2d	ac	1a
> Destination: IntelCor_5f:e5:42 (28:d0:ea:5f:e5:42)	0020	47	11	00	50	d0	5b	07	dc	cb	bd	1d	dd	79	be	50	18
> Source: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)	0030	00	dc	ca	2f	00	00	04	b1	3e	7a	90	65	d0	12	01	5b
Type: IPv4 (0x0800)	0040	a1	06	86	35	3a	8e	5a	63	ac	1a	92	4c	d4	f8	40	f4
	0050	5e	aa	d9	71	hc	90	a3	66	f1	d5	68	0b	16	df	e3	77

Tipo (IPv4):

> Frame 17: 350 bytes on wire (2800 bits), 350 bytes captured (2800 bits) on interface \Device\NPF_{58B78EF0-7BDE-4DE}	0000	28	d0	ea	5f	e5	42	00	d0	03	ff	94	00	08	00	45	00
> Ethernet II, Src: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00), Dst: IntelCor_5f:e5:42 (28:d0:ea:5f:e5:42)	0010	01	50	9d	ac	40	00	cf	06	7a	c3	22	df	7c	2d	ac	1a
> Destination: IntelCor_5f:e5:42 (28:d0:ea:5f:e5:42)	0020	47	11	00	50	d0	5b	07	dc	cb	bd	1d	dd	79	be	50	18
> Source: ComdaEnt_ff:94:00 (00:d0:03:ff:94:00)	0030	00	dc	ca	2f	00	00	04	b1	3e	7a	90	65	d0	12	01	5b
Type: IPv4 (0x0800)	0040	a1	06	86	35	3a	8e	5a	63	ac	1a	92	4c	d4	f8	40	f4
	0050	5e	aa	d9	71	hc	90	a3	66	f1	d5	68	0b	16	df	e3	77

## Identificadores do Protocolo IPv4:

Podemos observar a versão do IP (IPv4), o tamanho do header que é 20 bytes, os IPs da Fonte e Destino.

01	50	9d	ac	40	00	c	f	06	7a	c3	22	df	7c	2d	ac	1a	28	d0	ea	5f	e5	42	00	d0	03	ff	94	00	08	00	45	00		
47	11	00	50	d0	5b	07	dc	cb	bd	1d	dd	79	be	50	18	47	11	01	50	9d	ac	40	00	c	f	06	7a	c3	22	df	7c	2d	ac	1a
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

> Frame 17: 350 bytes on wire (2800 bits), 350 bytes captured (2800 bits) on interface \Device\NPF\_{58B78FE0-7BDE-4DE} Ethernet II, Src: ComdaEnt\_ff:94:00 (00:d0:03:ff:94:00), Dst: IntelCor\_5f:e5:42 (28:d0:ea:5f:e5:42)

> Destination: IntelCor\_5f:e5:42 (28:d0:ea:5f:e5:42)

> Source: ComdaEnt\_ff:94:00 (00:d0:03:ff:94:00)

Internet Protocol Version 4, Src: 34.223.124.45, Dst: 172.26.71.17

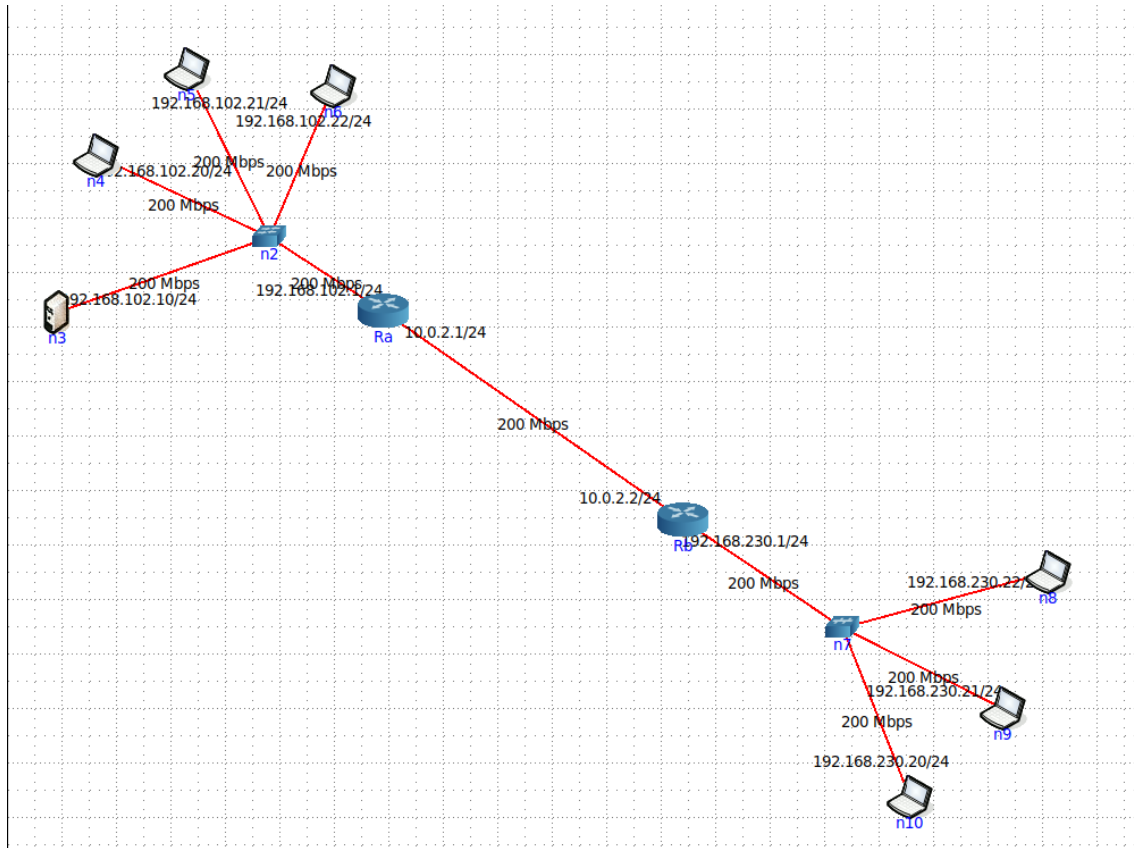
0000	28	d0	ea	5f	e5	42	00	d0	03	ff	94	00	08	00	45	00	(...B...E	
0010	01	50	9d	ac	40	00	c	f	06	7a	c3	22	df	7c	2d	ac	1a	(...P...P
0020	00	dc	ca	2f	00	00	00	00	00	00	00	00	00	00	00	00	00	(.../...>E
0030	00	dc	ca	2f	00	00	00	00	00	00	00	00	00	00	00	00	00	(.../...>E
0040	a1	06	86	35	3a	8e	5a	63	63	1a	92	4d	d4	f8	04	f4	(...5;Zc...L@	
0050	5e	aa	49	71	bc	03	a	66	f1	5d	68	0b	16	d	e3	77	(...q...f...7	
0060	24	95	38	4c	99	4f	d2	75	81	8e	39	27	b1	3f	77	78	(...\$L0...9...7	

### Identificadores do TPC:

## Podemos observar as Portas da fonte e do destino

```
> Frame 17: 350 bytes on wire (2800 bits), 350 bytes captured (2800 bits) on interface \Device\NPF_{58B78EF0-7BDE-4DE9-8000-000000000000}
> Ethernet II, Src: ComdaEnt_ff:94:00 (00:0d:03:ff:94:00), Dst: IntelCon_Sf:e5:42 (28:d0:ea:5f:e5:42)
> Internet Protocol Version 4, Src: 34.223.124.45, Dst: 172.26.71.13
> Transmission Control Protocol, Src Port: 80, Dst Port: 51339, Seq: 1251, Ack: 625, Len: 296
> [2 Reassembled TCP Segments (1546 bytes): #16(1250), #17(296)]
> Hypertext Transfer Protocol
> Line-based text data: text/html (79 lines)
```

## 4. Protocolo ARP



.1

a) Podemos observar na imagem o IP address e o seu MAC address correspondente e de seguida o tipo de protocolo usado e interface de rede.

```
root@n5:/tmp/pycore.43403/n5.conf# arp -a  
? (192.168.102.1) at 00:00:00:aa:00:04 [ether] on eth0
```

b) Seria o router do departamento A pois este poderia criar uma tabela onde pertenceriam todos os dispositivos do departamento, logo criaria a maior tabela ARP em termos de números de entradas.

.2

a) Endereço MAC origem: 00:00:00:aa:00:01

Endereço MAC destino: ff:ff:ff:ff:ff:ff

```
▼ 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)
```

O endereço destino é identificado como Broadcast, pois a máquina que pede o ARP request precisa de saber o endereço MAC de destino, para isso envia uma mensagem para todas as estações da network usando o endereço de Broadcast esperando uma resposta.

**b)** Valor hexadecimal: 0x0806, que representa o tipo de protocolo ARP utilizado.

```
-----
Type: ARP (0x0806)
```

**c)** Em opcode podemos observar o campo "request" com valor 1 significando que estamos perante um ARP request, para além disto os IP's contidos em ARP são os de origem e destino, bem como o endereço MAC origem, convém ressaltar que a origem, no acto do pedido, ainda não conhece o endereço MAC destino, uma vez que este foi apagado previamente.

```
▼ 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: 192.168.102.20
  Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
  Target IP address: 192.168.102.1
```

**d)** A máquina de origem pergunta a todos os hosts qual deles possui o endereço IP 192.168.102.1, posto isto, o host que o possuir envia a resposta, incluindo o seu endereço MAC, para o IP 192.168.102.20 .

*veth4.0.22						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
arp						
No.	Time	Source	Destination	Protocol	Length	Info
15	21.419110083	00:00:00_aa:00:01	Broadcast	ARP	42	Who has 192.168.102.1? Tell 192.168.102.20
16	21.419594167	00:00:00_aa:00:04	00:00:00_aa:00:01	ARP	42	192.168.102.1 is at 00:00:00:aa:00:04
31	26.640124304	00:00:00_aa:00:04	00:00:00_aa:00:01	ARP	42	Who has 192.168.102.20? Tell 192.168.102.1
32	26.640139577	00:00:00_aa:00:01	00:00:00_aa:00:04	ARP	42	192.168.102.20 is at 00:00:00:aa:00:01

.3

a) O valor do campo ARP opcode é 2 ou seja trata-se de uma mensagem ARP reply.

```
▼ 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:00:04 (00:00:00:aa:00:04)
  Sender IP address: 192.168.102.1
  Target MAC address: 00:00:00_aa:00:01 (00:00:00:aa:00:01)
  Target IP address: 192.168.102.20
```

b) A resposta está localizada no campo Sender MAC address

c) Endereço MAC de origem: corresponde ao endereço IP que foi pedido pela máquina que enviou o pedido ARP, podemos observar o endereço pelo comando executado abaixo “arp-a”.

Endereço MAC de destino: corresponde á máquina que enviou o pedido ARP, podemos observar o endereço na execução do comando “ifconfig”.

```
root@n4:/tmp/pycore.43403/n4.conf# netstat -nr
Kernel IP routing table
Destination    Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0        192.168.102.1  0.0.0.0         UG      0 0        0 eth0
192.168.102.0  0.0.0.0        255.255.255.0   U        0 0        0 eth0
root@n4:/tmp/pycore.43403/n4.conf# arp -a
? (192.168.102.1) at 00:00:00:aa:00:04 [ether] on eth0
root@n4:/tmp/pycore.43403/n4.conf# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.102.20 netmask 255.255.255.0 broadcast 0.0.0.0
    inet6 2001::20 prefixlen 64 scopeid 0x0<global>
    inet6 fe80::200:ff:feaa:1 prefixlen 64 scopeid 0x20<link>
    ether 00:00:00:aa:00:01 txqueuelen 1000 (Ethernet)
    RX packets 2267 bytes 183910 (183,9 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 31 bytes 2558 (2,5 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 16 bytes 1376 (1,3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 16 bytes 1376 (1,3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

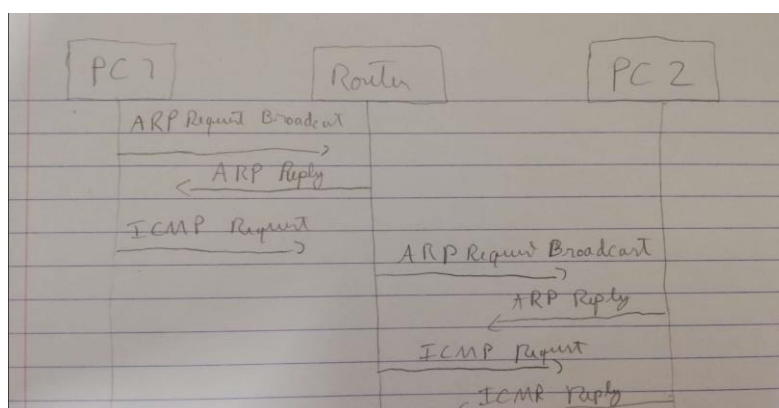
d) O modo de comunicação usado é unicast pois quando um dispositivo envia um pacote ARP para obter um endereço MAC este envia um pacote usando o endereço de Broadcast como destino, isto acontece devido ao dispositivo de origem não conhecer o endereço MAC do dispositivo de destino, então ele envia o pacote a todos os dispositivos da rede para que este chegue ao endereço de destino e este responda com o seu endereço MAC. Ao responder este envia um pacote unicast diretamente para o dispositivo de origem pois ele agora sabe o endereço MAC do dispositivo de origem.

.4 Não originou pacotes ARP pois se o endereço MAC do dispositivo de destino já estiver armazenado na tabela ARP do dispositivo de origem o dispositivo de origem pode enviar o pacote de ping diretamente usando o endereço MAC conhecido, sem precisar enviar uma solicitação ARP para a rede local.

.5 O valor 1 no Hardware type mostra que se trata de um tipo de endereço Ethernet, o campo Hardware size indica o tamanho do endereço da camada de ligação logica tendo este o valor 6, específico para endereços Ethernet, o valor hexadecimal de 0x0800 campo Protocol type mostra que se trate de um protocolo IPv4 e o campo Protocol size indica o tamanho do endereço da camada de rede, sendo este o valor específico para IPv4.

```
▼ Address Resolution Protocol (request)
  Hardware type: Ethernet (1)
  Protocol type: IPv4 (0x0800)
  Hardware size: 6
  Protocol size: 4
```

.6

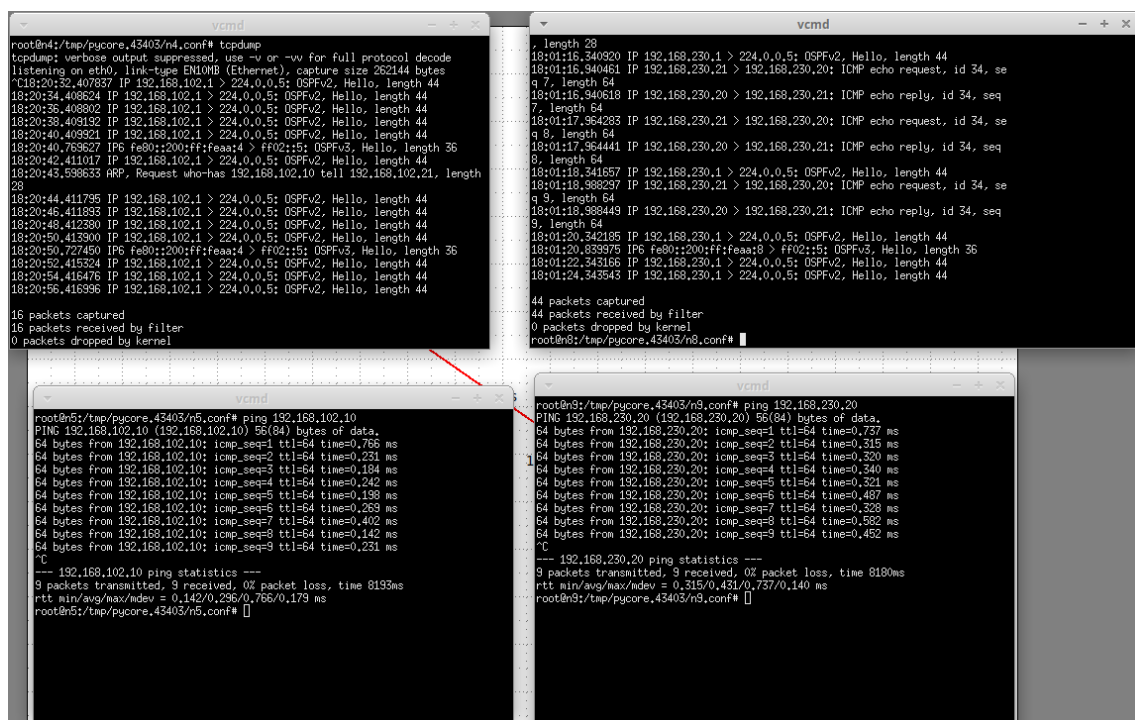




## 5. Domínios de colisão

**.1** Como podemos observar na imagem, no departamento A (n4,n5) como é utilizado um switch o computador n4 não apanha as tramas enviadas pelo computador n5 ao host n3 no entanto são apanhadas outros que não são relacionadas com as tramas enviadas por n5.

No departamento B como é usado um hub o computador n8 consegue apanhar as tramas enviadas pelo computador n9, podemos ver o echo request e echo reply entre n9 e n10 (192.168.230.20) obtido através do comando ping observado em baixo.



```
root@n4:/tmp/pycore.43403/n4.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
18:20:32.407837 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:20:34.408624 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:20:36.408602 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:20:38.409132 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:20:40.409321 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:20:40.763627 IP6 fe80::200:ff:feaa:4 > ff02::5: OSPFv3, Hello, length 36
18:20:42.411017 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:20:43.598633 ARP, Request who-has 192.168.102.10 tell 192.168.102.21, length 28
18:20:44.411795 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:20:46.411893 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:20:48.412380 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:20:50.413900 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:20:50.727450 IP6 fe80::200:ff:feaa:4 > ff02::5: OSPFv3, Hello, length 36
18:20:52.415324 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:20:54.416476 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:20:56.416996 IP 192.168.102.1 > 224.0.0.5: OSPFv2, Hello, length 44

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

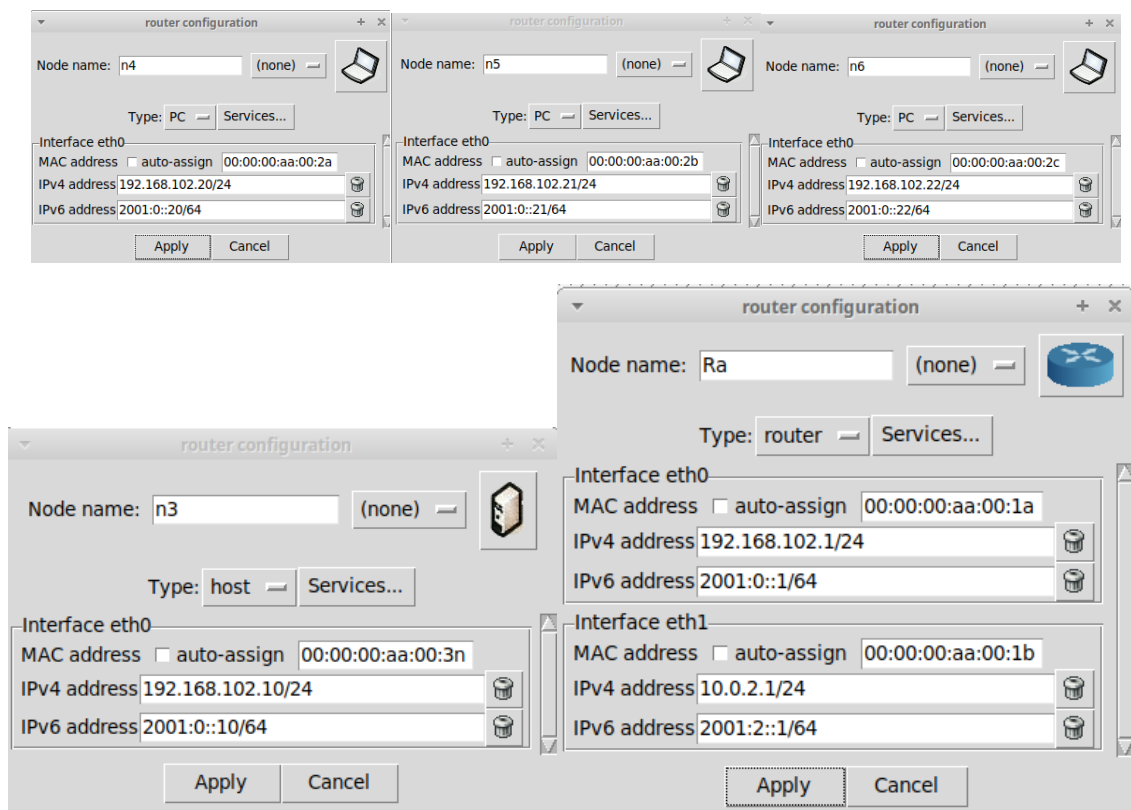
root@n8:/tmp/pycore.43403/n8.conf#
length 28
18:01:16.340820 IP 192.168.230.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:01:16.340461 IP 192.168.230.21 > 192.168.230.20: ICMP echo request, id 34, seq 7, length 64
18:01:16.940618 IP 192.168.230.20 > 192.168.230.21: ICMP echo reply, id 34, seq 7, length 64
18:01:17.934283 IP 192.168.230.21 > 192.168.230.20: ICMP echo request, id 34, seq 8, length 64
18:01:17.964441 IP 192.168.230.20 > 192.168.230.21: ICMP echo reply, id 34, seq 8, length 64
18:01:18.341657 IP 192.168.230.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:01:18.988297 IP 192.168.230.21 > 192.168.230.20: ICMP echo request, id 34, seq 9, length 64
18:01:18.988449 IP 192.168.230.20 > 192.168.230.21: ICMP echo reply, id 34, seq 9, length 64
18:01:20.342185 IP 192.168.230.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:01:20.833975 IP6 fe80::200:ff:feaa:8 > ff02::5: OSPFv3, Hello, length 36
18:01:22.343168 IP 192.168.230.1 > 224.0.0.5: OSPFv2, Hello, length 44
18:01:24.343543 IP 192.168.230.1 > 224.0.0.5: OSPFv2, Hello, length 44

44 packets captured
44 packets received by filter
0 packets dropped by kernel
root@n8:/tmp/pycore.43403/n8.conf#

root@n5:/tmp/pycore.43403/n5.conf# ping 192.168.102.10
PING 192.168.102.10 (192.168.102.10) 56(84) bytes of data:
64 bytes from 192.168.102.10: icmp_seq=1 ttl=64 time=0.766 ms
64 bytes from 192.168.102.10: icmp_seq=2 ttl=64 time=0.281 ms
64 bytes from 192.168.102.10: icmp_seq=3 ttl=64 time=0.184 ms
64 bytes from 192.168.102.10: icmp_seq=4 ttl=64 time=0.242 ms
64 bytes from 192.168.102.10: icmp_seq=5 ttl=64 time=0.198 ms
64 bytes from 192.168.102.10: icmp_seq=6 ttl=64 time=0.269 ms
64 bytes from 192.168.102.10: icmp_seq=7 ttl=64 time=0.402 ms
64 bytes from 192.168.102.10: icmp_seq=8 ttl=64 time=0.142 ms
64 bytes from 192.168.102.10: icmp_seq=9 ttl=64 time=0.231 ms
^C
--- 192.168.102.10 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8193ms
rtt min/avg/max/mdev = 0.142/0.286/0.766/0.173 ms
root@n5:/tmp/pycore.43403/n5.conf#

root@n9:/tmp/pycore.43403/n9.conf# ping 192.168.230.20
PING 192.168.230.20 (192.168.230.20) 56(84) bytes of data:
64 bytes from 192.168.230.20: icmp_seq=1 ttl=64 time=0.737 ms
64 bytes from 192.168.230.20: icmp_seq=2 ttl=64 time=0.315 ms
64 bytes from 192.168.230.20: icmp_seq=3 ttl=64 time=0.320 ms
64 bytes from 192.168.230.20: icmp_seq=4 ttl=64 time=0.340 ms
64 bytes from 192.168.230.20: icmp_seq=5 ttl=64 time=0.321 ms
64 bytes from 192.168.230.20: icmp_seq=6 ttl=64 time=0.497 ms
64 bytes from 192.168.230.20: icmp_seq=7 ttl=64 time=0.328 ms
64 bytes from 192.168.230.20: icmp_seq=8 ttl=64 time=0.392 ms
64 bytes from 192.168.230.20: icmp_seq=9 ttl=64 time=0.452 ms
^C
--- 192.168.230.20 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8180ms
rtt min/avg/max/mdev = 0.315/0.431/0.757/0.140 ms
root@n9:/tmp/pycore.43403/n9.conf#
```

.2



Logo a tabela de comutação seria:

Nome	MAC Address	Port
n4	00:00:00:aa:00:2a	1
n5	00:00:00:aa:00:2b	2
n6	00:00:00:aa:00:2c	3
n3	00:00:00:aa:00:3n	4
Ra	00:00:00:aa:00:1a	5

## Conclusão

Ao longo da realização deste trabalho, em simbiose com a matéria abordada nas aulas T, pusemos à prova as nossas capacidades de trabalho com a tecnologia Ethernet e protocolo ARP. Encontrámos diversas dificuldades ao longo do trajeto, mas fomos capazes de as ultrapassar através da consulta do material da

UC e com a ajuda da professora. Saímos satisfeitos com o trabalho realizado e convictos da interiorização da matéria abordada.