



UNIVERSIDADE DO MINHO

DEPARTAMENTO DE INFORMÁTICA

Redes de Computadores
TP3:Camada de Ligação Lógica
Grupo N^o 8 PL6

Gonçalo Almeida (A84610)

Emanuel Rodrigues (A84776)

Lázaro Pinheiro (A86788)

27 de Novembro de 2019

Conteúdo

1	Questões e Respostas	3
1.1	Captura e análise de tramas Ethernet	3
1.2	Protocolo ARP	5
1.3	ARP Gratuito	8
1.4	Domínios de colisão	9
2	Conclusão	17

Capítulo 1

Questões e Respostas

1.1 Captura e análise de tramas Ethernet

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

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
http						
No.	Time	Source	Destination	Protocol	Length	Info
171	5.997637440	192.168.100.206	194.210.238.81	HTTP	383	GET /success.txt HTTP/1.1
172	6.005248268	194.210.238.81	192.168.100.206	HTTP	450	HTTP/1.1 200 OK (text/plain)
176	6.010087933	192.168.100.206	194.210.238.81	HTTP	388	GET /success.txt?ipv4 HTTP/1.1
178	6.018662014	194.210.238.81	192.168.100.206	HTTP	450	HTTP/1.1 200 OK (text/plain)
201	6.638630687	192.168.100.206	193.136.19.40	HTTP	486	GET / HTTP/1.1
203	6.640055995	193.136.19.40	192.168.100.206	HTTP	547	HTTP/1.1 301 Moved Permanently (text/html)
220	6.708964543	192.168.100.206	93.184.220.29	OCSP	466	Request
223	6.812315993	93.184.220.29	192.168.100.206	OCSP	831	Response

Frame 201: 486 bytes on wire (3888 bits), 486 bytes captured (3888 bits) on interface 0

Ethernet II, Src: NeostarT_17:35:8a (00:24:32:17:35:8a), Dst: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)

- Destination: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
 - Address: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
 -0. = LG bit: Globally unique address (factory default)
 -0. = IG bit: Individual address (unicast)
- Source: NeostarT_17:35:8a (00:24:32:17:35:8a)
 - Address: NeostarT_17:35:8a (00:24:32:17:35:8a)
 -0. = LG bit: Globally unique address (factory default)
 -0. = IG bit: Individual address (unicast)
 - Type: IPv4 (0x0800)
- Internet Protocol Version 4, Src: 192.168.100.206, Dst: 193.136.19.40
- Transmission Control Protocol, Src Port: 44728, Dst Port: 80, Seq: 1, Ack: 1, Len: 420
- Hypertext Transfer Protocol

0000	00 0c 29 d2 19 f0 00 24	32 17 35 8a 08 00 45 00	..).\$ 2.5..E-
0010	01 d8 e1 16 40 00 40 06	5d e2 c0 a8 64 ce c1 88	...@.@]...d...
0020	13 28 ae b8 00 50 4b 7d	22 21 83 37 a8 96 80 18	...PK}!..7...
0030	00 e5 fb f1 00 00 01 01	08 0a e5 77 44 9d fd 53wD..S
0040	5f bd 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	6d 69 65 69 2e 64 69 2e	...Host: miei.di.
0060	75 6d 69 6e 68 6f 2e 70	74 0d 0a 55 73 65 72 2d	uminho.p t..User-
0070	41 67 65 6e 74 3a 20 4d	6f 7a 69 6c 6c 61 2f 35	Agent: Mozilla/5
0080	2e 30 20 28 58 31 31 3b	20 55 62 75 6e 74 75 3b	.0 (X11; Ubuntu;
0090	20 4c 69 6e 75 78 20 78	38 36 5f 36 34 3b 20 72	Linux x 86_64; r
00a0	76 3a 37 30 2e 30 29 20	47 65 63 6b 6f 2f 32 30	v:70.0) Gecko/20
00b0	31 30 30 31 30 31 20 46	69 72 65 66 6f 78 2f 37	100101 Firefox/7
00c0	30 2e 30 0d 0a 41 63 63	65 70 74 3a 20 74 65 78	0.0..Accept: tex
00d0	74 2f 68 74 6d 6c 2c 61	70 70 6c 69 63 61 74 69	t/html,a pplicati
00e0	6f 6e 2f 78 68 74 6d 6c	2b 78 6d 6c 2c 61 70 70	on/xhtmll +xml,app
00f0	6c 69 63 61 74 69 6f 6e	2f 78 6d 6c 3b 71 3d 30	lication /xml;q=0
0100	2e 39 2c 2a 2f 2a 3b 71	3d 30 2e 38 0d 0a 41 63	.9,*/";q=0.8..Ac
0110	63 65 70 74 2d 4c 61 6e	67 75 61 67 65 3a 20 70	cept-Lan guage: p

Figura 1.1

Os endereços MAC de origem e destino da trama são 00:24:32:17:35:8a e 00:0c:29:d2:19:f0 respetivamente.

2. Identifique a que sistemas se referem. Justifique.

O endereço MAC de origem da trama corresponde à interface ativa do nosso computador e o endereço de destino corresponde à interface de um router da rede local à qual estamos ligados.

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

O valor hexadecimal do campo Type da trama Ethernet é 0x0800 e significa que os dados da trama correspondem a um pacote IPv4.

4. Quantos 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.

O número de bytes utilizados desde o início da trama até ao caractere “G” é 47, sendo 486 o número de bytes total. A percentagem da sobrecarga (overhead) introduzida pela pilha protocolar no envio do HTTP GET é calculada da seguinte forma:

$$(47/486) \times 100 = 9.67078 \approx 9,7\%$$

5. Através de visualização direta de uma trama capturada, verifique que, possivelmente, o campo FCS (Frame Check Sequence) usado para deteção de erros não está a ser usado. Em sua opinião, porque será?

Como podemos observar na figura 1.1, o campo FCS não está a ser usado porque não foram detetados erros.

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

O endereço Ethernet da fonte é 00:0c:29:d2:19:f0 e corresponde à interface do router da rede local à qual estamos ligados.

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

O endereço MAC do destino é 00:24:32:17:35:8a e corresponde à interface ativa do nosso computador.

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

Como podemos observar na figura 1.1, os vários protocolos contidos na trama recebida são Ethernet II, IPv4 (Internet Protocol version 4), TCP (Transmission Control Protocol) e HTTP (Hypertext Transfer Protocol).

1.2 Protocolo ARP

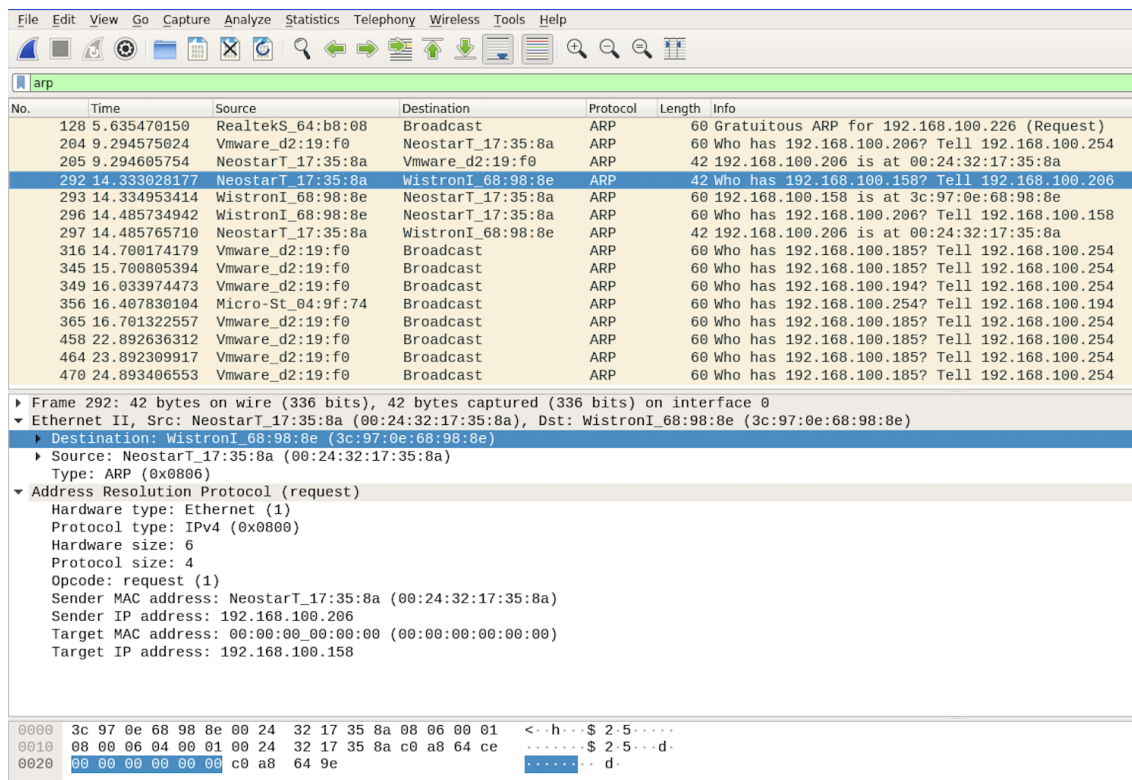
9. Observe o conteúdo da tabela ARP. Explique o significado de cada uma das colunas.

```
Ficheiro Editar Ver Procurar Terminal Ajuda
~ arp -a
brom158.sa.di.uminho.pt (192.168.100.158) em 3c:97:0e:68:98:8e [ether] em enx00243217358a
gw.sa.di.uminho.pt (192.168.100.254) em 00:0c:29:d2:19:f0 [ether] em enx00243217358a
? (192.168.100.214) em 88:d7:f6:1b:2d:80 [ether] em enx00243217358a
? (192.168.100.195) em 68:f7:28:81:40:a0 [ether] em enx00243217358a
? (192.168.100.190) em 54:a0:50:0f:38:da [ether] em enx00243217358a
```

Figura 1.2: Tabela Arp

Na tabela ARP temos os endereços IPs guardados na cache ARP do computador associados aos endereços MAC guardados e a coluna Type que contém o tipo de endereçamento utilizado.

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?



No.	Time	Source	Destination	Protocol	Length	Info
128	5.635470150	RealtekS_64:b8:08	Broadcast	ARP	60	Gratuitous ARP for 192.168.100.226 (Request)
204	9.29457024	Vmware_d2:19:f0	Neostart_17:35:8a	ARP	60	Who has 192.168.100.206? Tell 192.168.100.254
205	9.294605754	Neostart_17:35:8a	Vmware_d2:19:f0	ARP	42	192.168.100.206 is at 00:24:32:17:35:8a
292	14.333028177	Neostart_17:35:8a	WistronI_68:98:8e	ARP	42	Who has 192.168.100.158? Tell 192.168.100.206
293	14.334953414	WistronI_68:98:8e	Neostart_17:35:8a	ARP	60	192.168.100.158 is at 3c:97:0e:68:98:8e
296	14.485734942	WistronI_68:98:8e	Neostart_17:35:8a	ARP	60	Who has 192.168.100.206? Tell 192.168.100.158
297	14.485765710	Neostart_17:35:8a	WistronI_68:98:8e	ARP	42	192.168.100.206 is at 00:24:32:17:35:8a
316	14.700174179	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
345	15.700805394	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
349	16.033974473	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.194? Tell 192.168.100.254
356	16.407830104	Micro-St_04:9f:74	Broadcast	ARP	60	Who has 192.168.100.254? Tell 192.168.100.194
365	16.701322557	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
458	22.892636312	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
464	23.892309917	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
470	24.893406553	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254

Frame 292: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0

Ethernet II, Src: Neostart_17:35:8a (00:24:32:17:35:8a), Dst: WistronI_68:98:8e (3c:97:0e:68:98:8e)

Destination: WistronI_68:98:8e (3c:97:0e:68:98:8e)

Source: Neostart_17:35:8a (00:24:32:17:35:8a)

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: Neostart_17:35:8a (00:24:32:17:35:8a)

Sender IP address: 192.168.100.206

Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)

Target IP address: 192.168.100.158

0000 3c 97 0e 68 98 8e 00 24 32 17 35 8a 08 06 00 01 <..h...\$ 2.5....

0010 08 00 06 04 00 01 00 24 32 17 35 8a c0 a8 64 ce\$ 2.5...d.

0020 00 00 00 00 00 00 c0 a8 64 9ed.

Figura 1.3: Trama Ethernet do ARP Request

Os valores hexadecimais dos endereços origem e destino na trama Ethernet são 00:24:32:17:35:8a e ff:ff:ff:ff:ff:ff respetivamente. O valor do endereço destino significa que todos os nós recebem e processam a trama, mas apenas um deles responde.

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

O valor hexadecimal do campo Type da trama Ethernet é 0x0806 e significa que os dados da trama correspondem a um pacote ARP.

12. Qual o valor do campo ARP opcode? O que especifica? Se necessário, consulte a RFC do protocolo ARP (<http://tools.ietf.org/html/rfc826.html>).

O valor do campo ARP opcode é 1 e significa que corresponde a um request.

13. Identifique que tipo de endereços está contido na mensagem ARP? Que conclui?

Na mensagem ARP estão contidos os endereços IP de origem e destino, mas apenas conhecemos o endereço MAC da origem porque a mensagem é um ARP request.

14. Explícite que tipo de pedido ou pergunta é feito pelo host de origem?

A interface do nosso computador pergunta quem tem o endereço de IP indicado e o destino responde com o seu endereço MAC.

15. Localize a mensagem ARP que é a resposta ao pedido ARP efectuado.

No.	Time	Source	Destination	Protocol	Length	Info
128	5.635470150	RealtekS_64:b8:08	Broadcast	ARP	60	Gratuitous ARP for 192.168.100.226 (Request)
204	9.294575024	Vmware_d2:19:f0	NeostarT_17:35:8a	ARP	60	Who has 192.168.100.206? Tell 192.168.100.254
205	9.294605754	NeostarT_17:35:8a	Vmware_d2:19:f0	ARP	42	192.168.100.206 is at 00:24:32:17:35:8a
292	14.333028177	NeostarT_17:35:8a	WistronI_68:98:8e	ARP	42	Who has 192.168.100.158? Tell 192.168.100.206
293	14.334953414	WistronI_68:98:8e	NeostarT_17:35:8a	ARP	60	192.168.100.158 is at 3c:97:0e:68:98:8e
296	14.485734942	WistronI_68:98:8e	NeostarT_17:35:8a	ARP	60	Who has 192.168.100.206? Tell 192.168.100.158
297	14.485765710	NeostarT_17:35:8a	WistronI_68:98:8e	ARP	42	192.168.100.206 is at 00:24:32:17:35:8a
316	14.700174179	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
345	15.700805394	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
349	16.033974473	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.194? Tell 192.168.100.254
356	16.407830104	Micro-St_04:9f:74	Broadcast	ARP	60	Who has 192.168.100.254? Tell 192.168.100.194
365	16.701322557	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
458	22.892636312	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
464	23.892309917	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
470	24.893406553	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254

Figura 1.4: Tráfego ARP

(a) Qual o valor do campo ARP opcode? O que especifica?

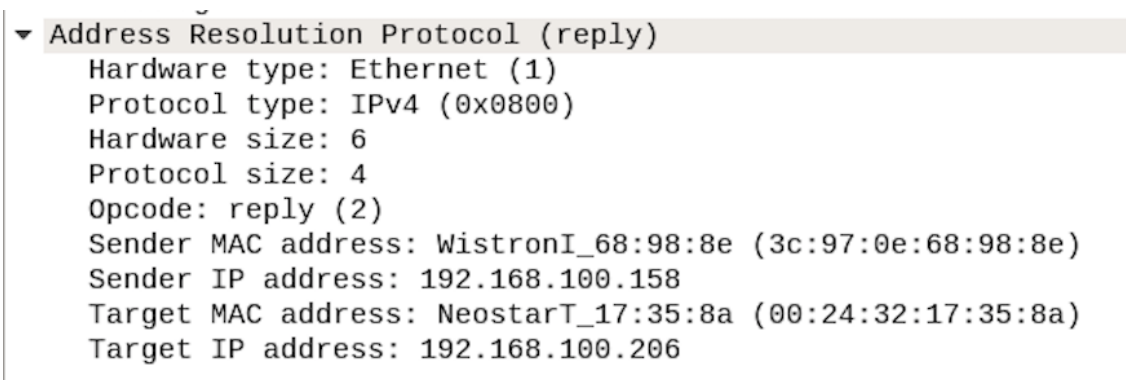


Figura 1.5: ARP reply

O valor do campo ARP opcode é 2 e corresponde a uma reply.

(b) Em que posição da mensagem ARP está a resposta ao pedido ARP?

0000	00	24	32	17	35	8a	3c	97	0e	68	98	8e	08	06	00	01	·\$2·5·<··h···
0010	08	00	06	04	00	02	3c	97	0e	68	98	8e	c0	a8	64	9e	· · · · · < · · h · · · · d ·
0020	00	24	32	17	35	8a	c0	a8	64	ce	00	00	00	00	00	00	·\$2·5· · · d · · · · ·
0030	00	00	00	00	00	00	00	00	00	00	00	00					· · · · · · · · · ·

Figura 1.6

A resposta ao pedido ARP está entre os bytes 7 e 12 inclusive da mensagem ARP.

1.3 ARP Gratuito

16. 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?

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

arp

No.	Time	Source	Destination	Protocol	Length	Info
2	0.458193981	CompalIn_7a:03:0a	Broadcast	ARP	60	Gratuitous ARP for 192.168.100.191 (Request)
5	1.458840995	CompalIn_7a:03:0a	Broadcast	ARP	60	Gratuitous ARP for 192.168.100.191 (Request)
28	5.459258907	CompalIn_7a:03:0a	Broadcast	ARP	60	Gratuitous ARP for 192.168.100.191 (Request)
38	5.810352341	Vmware_d2:19:f0	NeostarT_17:35:8a	ARP	60	Who has 192.168.100.206? Tell 192.168.100.254
39	5.810376322	NeostarT_17:35:8a	Vmware_d2:19:f0	ARP	42	192.168.100.206 is at 00:24:32:17:35:8a
52	6.459039539	CompalIn_7a:03:0a	Broadcast	ARP	60	Gratuitous ARP for 192.168.100.191 (Request)
68	7.458507825	CompalIn_7a:03:0a	Broadcast	ARP	60	Gratuitous ARP for 192.168.100.191 (Request)
72	8.458743541	CompalIn_7a:03:0a	Broadcast	ARP	60	Gratuitous ARP for 192.168.100.191 (Request)
73	9.459529907	CompalIn_7a:03:0a	Broadcast	ARP	60	Gratuitous ARP for 192.168.100.191 (Request)

▶ Frame 2: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0

▼ Ethernet II, Src: CompalIn_7a:03:0a (f0:76:1c:7a:03:0a), 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: CompalIn_7a:03:0a (f0:76:1c:7a:03:0a)
 - Address: CompalIn_7a:03:0a (f0:76:1c:7a:03:0a)
 -0..... = LG bit: Globally unique address (factory default)
 -0..... = IG bit: Individual address (unicast)
 - Type: ARP (0x0806)
 - Padding: 00000000000000000000000000000000
- ▼ Address Resolution Protocol (request/gratuitous ARP)
 - Hardware type: Ethernet (1)
 - Protocol type: IPv4 (0x0800)
 - Hardware size: 6
 - Protocol size: 4
 - Opcode: request (1)
 - [Is gratuitous: True]
 - Sender MAC address: CompalIn_7a:03:0a (f0:76:1c:7a:03:0a)
 - Sender IP address: 192.168.100.191
 - Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
 - Target IP address: 192.168.100.191


```

0000 ff ff ff ff ff ff f0 76 1c 7a 03 0a 08 06 00 01 .....v.Z.....
0010 08 00 06 04 00 01 f0 76 1c 7a 03 0a c0 a8 64 bf .....v.Z...d-
0020 ff ff ff ff ff ff c0 a8 64 bf 00 00 00 00 00 00 .....-.d.....
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
  
```

Figura 1.7: Pacote de pedido ARP gratuito

O que distingue um pacote de pedido ARP gratuito dos restantes pacotes de pedido ARP é o valor da flag `Is gratuitous` ser `true`. Face a este pedido, não se espera que exista reply pois, caso existisse, significaria que existe outra interface com o mesmo endereço IP que o nosso.

1.4 Domínios de colisão

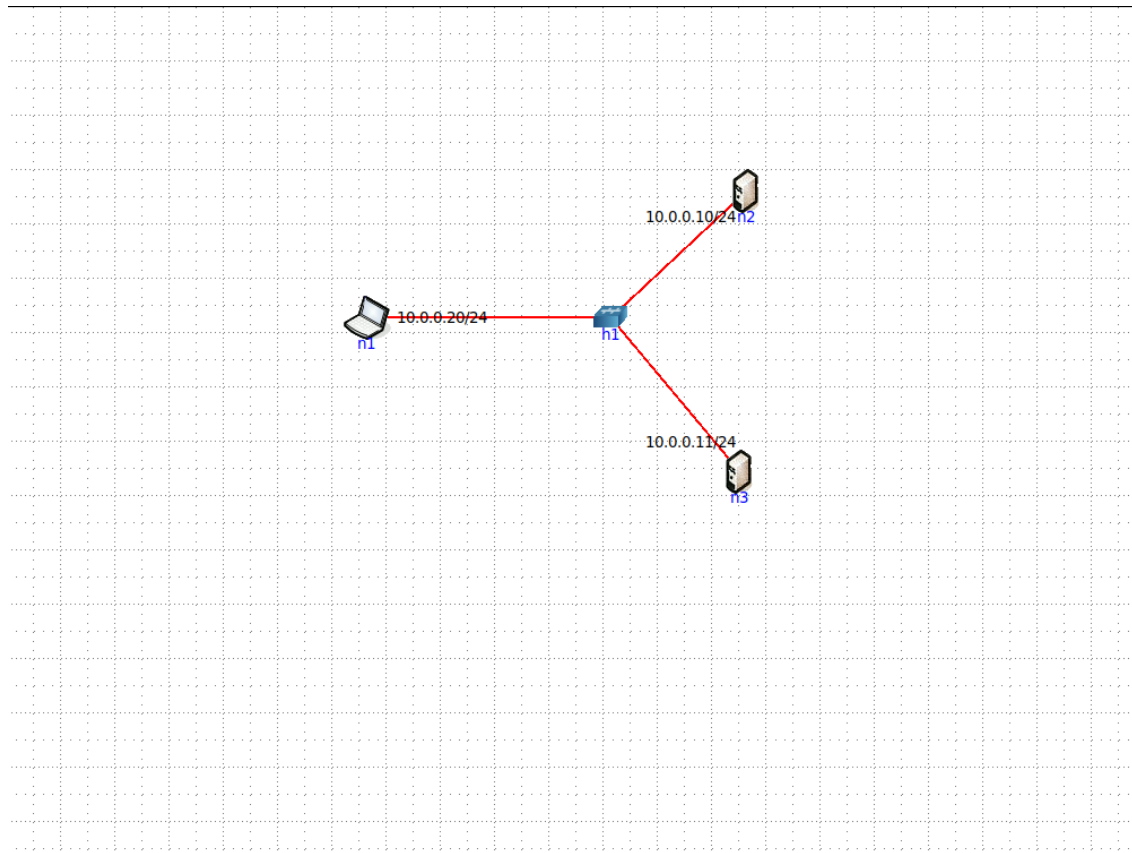


Figura 1.8: Topologia Core

17. Faça ping de n1 para n2. Verifique com a opção `tcpdump` como flui o tráfego nas diversas interfaces dos vários dispositivos. Que conclui?

```
root@n1: /tmp/pycore.43391/n1.conf
root@n1: /tmp/pycore.43391/n1.conf# ping 10.0.0.10
PING 10.0.0.10 (10.0.0.10) 56(84) bytes of data.
64 bytes from 10.0.0.10: icmp_seq=1 ttl=64 time=0.036 ms
64 bytes from 10.0.0.10: icmp_seq=2 ttl=64 time=0.076 ms
64 bytes from 10.0.0.10: icmp_seq=3 ttl=64 time=0.083 ms
64 bytes from 10.0.0.10: icmp_seq=4 ttl=64 time=0.064 ms
64 bytes from 10.0.0.10: icmp_seq=5 ttl=64 time=0.053 ms
64 bytes from 10.0.0.10: icmp_seq=6 ttl=64 time=0.083 ms
^C
--- 10.0.0.10 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5132ms
rtt min/avg/max/mdev = 0.036/0.065/0.083/0.020 ms
root@n1: /tmp/pycore.43391/n1.conf#
```

Figura 1.9: Ping no laptop n1 para o servidor n2

```
root@n2: /tmp/pycore.43391/n2.conf
root@n2: /tmp/pycore.43391/n2.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
^C14:47:18.937870 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 49, seq 1, length 64
14:47:18.937886 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 49, seq 1, length 64
14:47:19.966467 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 49, seq 2, length 64
14:47:19.966487 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 49, seq 2, length 64
14:47:20.996405 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 49, seq 3, length 64
14:47:20.996428 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 49, seq 3, length 64
14:47:22.024764 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 49, seq 4, length 64
14:47:22.024794 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 49, seq 4, length 64
14:47:23.052341 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 49, seq 5, length 64
14:47:23.052364 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 49, seq 5, length 64
14:47:24.070729 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 49, seq 6, length 64
14:47:24.070781 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 49, seq 6, length 64

12 packets captured
12 packets received by filter
0 packets dropped by kernel
root@n2: /tmp/pycore.43391/n2.conf#
```

Figura 1.10: tcpdump no servidor n2

```
root@n3: /tmp/pycore.43391/n3.conf
root@n3:/tmp/pycore.43391/n3.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
^C14:47:13.892513 ARP, Request who-has 10.0.0.20 tell 10.0.0.10, length 28
14:47:13.892515 ARP, Request who-has 10.0.0.10 tell 10.0.0.20, length 28
14:47:13.892534 ARP, Reply 10.0.0.20 is-at 00:00:00:aa:00:00 (oui Ethernet), length 28
14:47:13.892535 ARP, Reply 10.0.0.10 is-at 00:00:00:aa:00:01 (oui Ethernet), length 28
14:47:18.937869 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 49, seq 1, length 64
14:47:18.937888 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 49, seq 1, length 64
14:47:19.966466 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 49, seq 2, length 64
14:47:19.966490 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 49, seq 2, length 64
14:47:20.996403 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 49, seq 3, length 64
14:47:20.996430 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 49, seq 3, length 64
14:47:22.024763 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 49, seq 4, length 64
14:47:22.024797 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 49, seq 4, length 64
14:47:23.052339 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 49, seq 5, length 64
14:47:23.052367 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 49, seq 5, length 64
14:47:24.070728 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 49, seq 6, length 64
14:47:24.070785 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 49, seq 6, length 64

16 packets captured
16 packets received by filter
0 packets dropped by kernel
root@n3:/tmp/pycore.43391/n3.conf#
```

Figura 1.11: tcpdump no servidor n3

Visto que é usado um hub, os pacotes enviados através do ping do laptop n1 para o servidor n2 foram encaminhados para todas as outras interfaces, neste caso o servidor n3.

18. Na topologia de rede substitua o hub por um switch. Repita os procedimentos que realizou na pergunta anterior. 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.

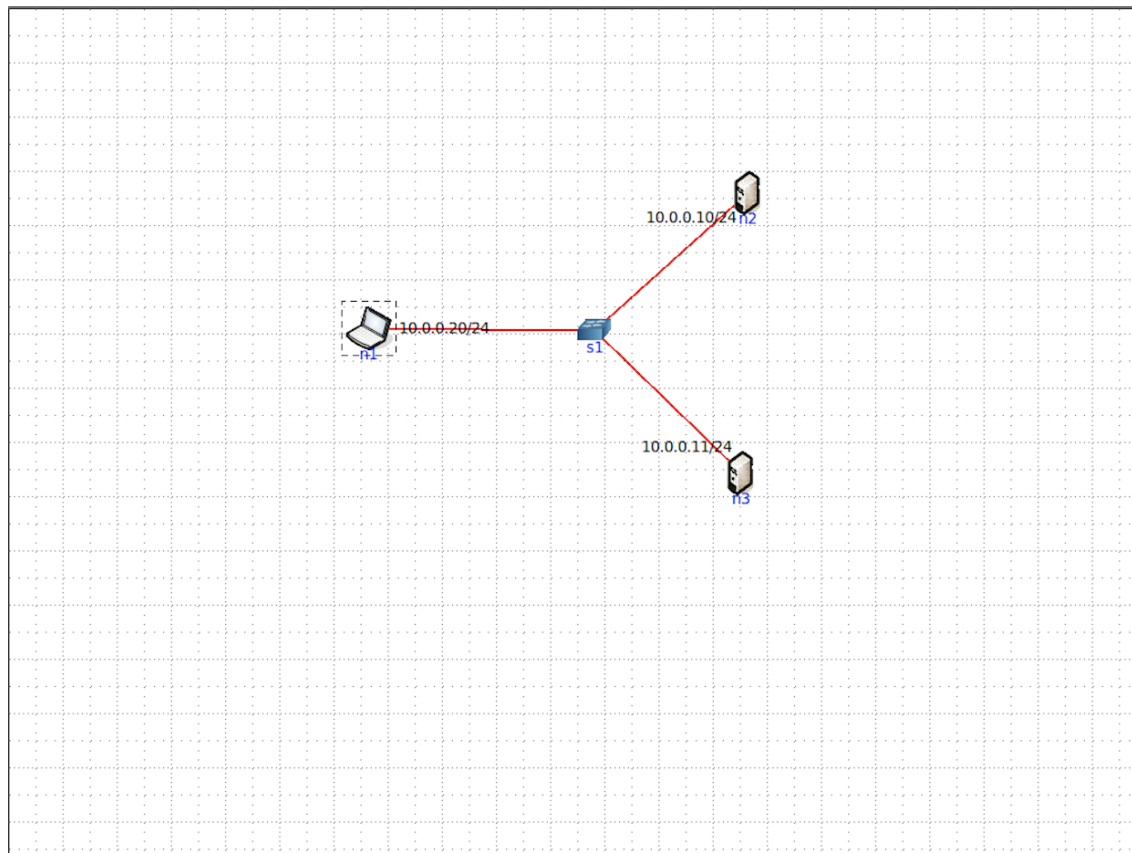


Figura 1.12: Topologia Core

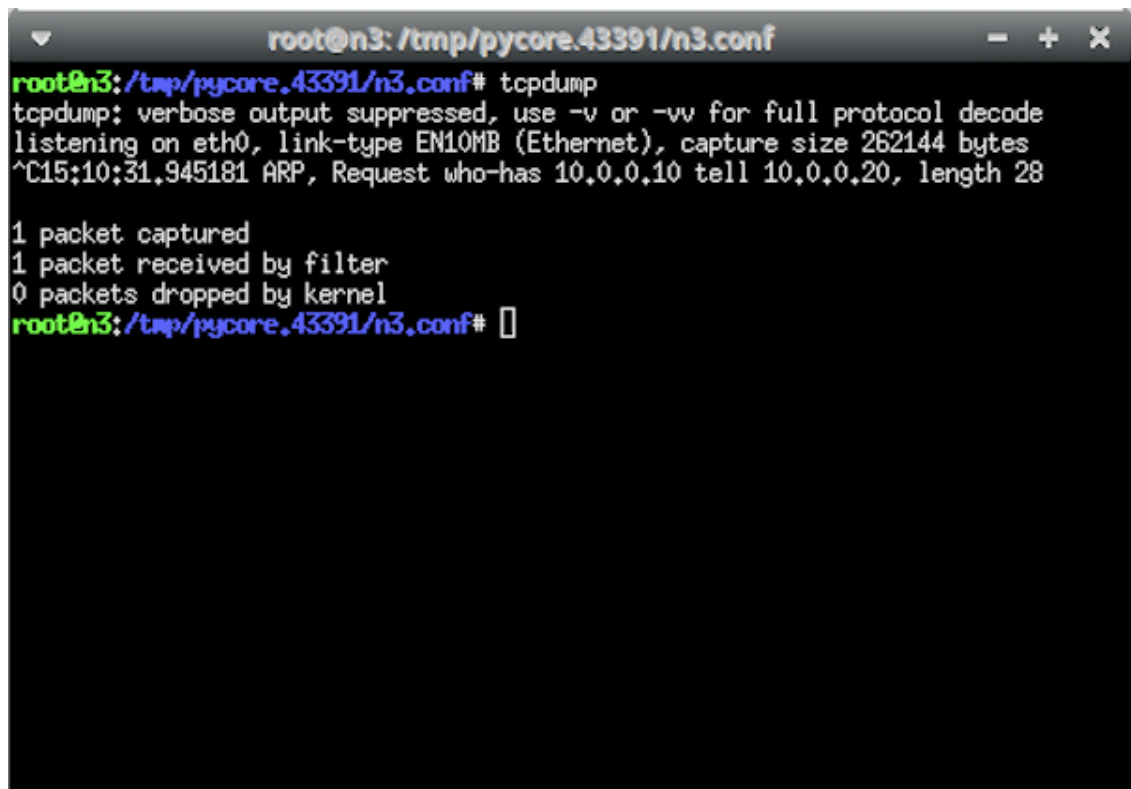
```
root@n1: /tmp/pycore.43391/n1.conf
root@n1: /tmp/pycore.43391/n1.conf# ping 10.0.0.10
PING 10.0.0.10 (10.0.0.10) 56(84) bytes of data.
64 bytes from 10.0.0.10: icmp_seq=1 ttl=64 time=0.066 ms
64 bytes from 10.0.0.10: icmp_seq=2 ttl=64 time=0.041 ms
64 bytes from 10.0.0.10: icmp_seq=3 ttl=64 time=0.081 ms
64 bytes from 10.0.0.10: icmp_seq=4 ttl=64 time=0.082 ms
64 bytes from 10.0.0.10: icmp_seq=5 ttl=64 time=0.082 ms
^C
--- 10.0.0.10 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4091ms
rtt min/avg/max/mdev = 0.041/0.070/0.082/0.017 ms
root@n1: /tmp/pycore.43391/n1.conf#
```

Figura 1.13: Ping no laptop n1 para o servidor n2

```
root@n2: /tmp/pycore.43391/n2.conf
root@n2:/tmp/pycore.43391/n2.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
^C15:10:31.945183 ARP, Request who-has 10.0.0.10 tell 10.0.0.20, length 28
15:10:31.945207 ARP, Reply 10.0.0.10 is-at 00:00:00:aa:00:01 (oui Ethernet), length 28
15:10:31.945213 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 27, seq 1, length 64
15:10:31.945221 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 27, seq 1, length 64
15:10:32.968161 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 27, seq 2, length 64
15:10:32.968175 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 27, seq 2, length 64
15:10:33.988282 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 27, seq 3, length 64
15:10:33.988311 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 27, seq 3, length 64
15:10:35.016568 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 27, seq 4, length 64
15:10:35.016597 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 27, seq 4, length 64
15:10:36.037164 IP 10.0.0.20 > 10.0.0.10: ICMP echo request, id 27, seq 5, length 64
15:10:36.037194 IP 10.0.0.10 > 10.0.0.20: ICMP echo reply, id 27, seq 5, length 64
15:10:37.028114 ARP, Request who-has 10.0.0.20 tell 10.0.0.10, length 28
15:10:37.028187 ARP, Reply 10.0.0.20 is-at 00:00:00:aa:00:00 (oui Ethernet), length 28

14 packets captured
14 packets received by filter
0 packets dropped by kernel
root@n2:/tmp/pycore.43391/n2.conf#
```

Figura 1.14: tcpdump no servidor n2

A terminal window titled 'root@n3: /tmp/pycore.43391/n3.conf' with standard window controls. The terminal shows the execution of 'tcpdump' and its output. The output indicates that verbose output is suppressed, it is listening on eth0, and a packet was captured. The captured packet is an ARP request from 10.0.0.10 asking for 10.0.0.20. Summary statistics show 1 packet captured, 1 received by filter, and 0 dropped by kernel.

```
root@n3: /tmp/pycore.43391/n3.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
^C15:10;31.945181 ARP, Request who-has 10.0.0.10 tell 10.0.0.20, length 28

1 packet captured
1 packet received by filter
0 packets dropped by kernel
root@n3: /tmp/pycore.43391/n3.conf#
```

Figura 1.15: tcpdump no servidor n3

Como podemos observar nas figuras anteriores, os pacotes enviados através do ping do laptop n1 para o servidor n2, são encaminhados apenas para o servidor n2, não havendo colisões.

Capítulo 2

Conclusão

Com a realização deste trabalho prático, através da análise de tramas Ethernet, aprofundamos o nosso conhecimento sobre os protocolos ARP e Ethernet II, endereços MAC e interligação de redes. Isto foi possível com a ajuda da ferramenta Wireshark que permitiu analisar as tramas relativamente aos protocolos utilizados, os pacotes pedido/resposta e a deteção de erros. Com a ferramenta Core ficamos com uma ideia mais clara sobre o uso de dispositivos como os hubs e switches e sobre colisões em redes reais de modo a controlar e dividir domínios de colisão.