

PARTE 1

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

No.	Time	Source	Destination	Protocol	Info	Length
62	4.525701	192.168.100.193	193.136.19.20	HTTP	GET / HTTP/1.1	686
76	4.847876	192.168.100.193	193.136.19.20	HTTP	GET /favicon.ico HTTP/1.1	585
64	4.526745	193.136.19.20	192.168.100.193	HTTP	HTTP/1.1 304 Not Modified	185
78	4.849459	193.136.19.20	192.168.100.193	HTTP	HTTP/1.1 200 OK (text/plain)	1448

```
> Frame 62: 686 bytes on wire (5488 bits), 686 bytes captured (5488 bits) on interface 0
> Ethernet II, Src: AsustekC_47:1f:ab (74:d0:2b:47:1f:ab), Dst: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
> Internet Protocol Version 4, Src: 192.168.100.193, Dst: 193.136.19.20
> Transmission Control Protocol, Src Port: 49826, Dst Port: 80, Seq: 1, Ack: 1, Len: 632
> Hypertext Transfer Protocol

Ethernet II, Src: AsustekC_47:1f:ab (74:d0:2b:47:1f:ab), Dst: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
  Destination: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
  Source: AsustekC_47:1f:ab (74:d0:2b:47:1f:ab)
  Type: IPv4 (0x0800)
```

R: O nosso endereço destino é 00:0c:29:d2:19:f0 e o Origem é 74:d0:2b:47:1f:ab

2. Identifique a que sistemas se referem. Justifique.

```
Ethernet II, Src: AsustekC_47:1f:ab (74:d0:2b:47:1f:ab), Dst: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
  Destination: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
  Source: AsustekC_47:1f:ab (74:d0:2b:47:1f:ab)
  Type: IPv4 (0x0800)
```

R: A trama é destinada ao router que vai estar ligado à rede do departamento, acedendo ao http server e enviando a resposta.

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

```
> Destination: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
> Source: AsustekC_47:1f:ab (74:d0:2b:47:1f:ab)
  Type: IPv4 (0x0800)
```

R: O valor hexadecimal do campo Type da trama Ethernet é 0x0800, o que corresponde ao tipo de dados do IP que vai ser encapsulado.

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.

No.	Time	Source	Destination	Protocol	Info
62	4.525701	192.168.100.193	193.136.19.20	HTTP	GET / HTTP/1.1
76	4.847876	192.168.100.193	193.136.19.20	HTTP	GET /favicon.ico HTTP/1.1
64	4.526745	193.136.19.20	192.168.100.193	HTTP	HTTP/1.1 304 Not Modified
78	4.849459	193.136.19.20	192.168.100.193	HTTP	HTTP/1.1 200 OK (text/plain)

> Transmission Control Protocol, Src Port: 49826, Dst Port: 80, Seq: 1, Ack: 1, Len: 632
 ▾ Hypertext Transfer Protocol
 ▾ GET / HTTP/1.1\r\n
 > [Expert Info (Chat/Sequence): GET / HTTP/1.1\r\n]
 Request Method: GET
 Request URI: /
 Request Version: HTTP/1.1
 Host: miei.di.uminho.pt\r\n
 Connection: keep-alive\r\n
 Cache-Control: max-age=0\r\n
 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/62.0.3202.94 Safari/537.36\r\n
 Upgrade-Insecure-Requests: 1\r\n
 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8\r\n
 Accept-Encoding: gzip, deflate\r\n
 Accept-Language: pt-PT;q=0.9,pt;q=0.8,en-US;q=0.7,en;q=0.6

0030 01 00 fc 98 00 00 47 45 54 20 2f 20 48 54 54 50GET / HTTP

R: Desde o início da trama até ao carácter ASCII “G” do método HTTP GET, são usados 62 bytes. O tamanho da trama é 632, logo o overhead será dado por $62/632 = 0,098 = 9,8 \%$.

5. Em ligações com fios pouco susceptíveis a erros, nem sempre as NICs geram o código de detecção de erros. Através de uma visualização direta de uma trama capturada verifique se o campo FCS está visível, isto é, se está a ser utilizado.

Aceda à opção Edit/Preferences/Protocols/Ethernet e indique que é assumido o uso do campo FCS. Verifique qual o valor hexadecimal desse campo na trama capturada. Que conclui? Reponha a configuração a configuração inicial.

```
> Frame check sequence: 0x0d0a0d0a [incorrect, should be 0x153c06fc]
```

R: Não está a utilizar FCS, pois caso estivesse a utilizar a informação que corresponde a este iria aparecer a seguir à informação do tipo de nível 2. O valor que está na imagem corresponde aos 4 últimos bytes da trama. Concluímos que ao fazer a mudança vai dar como incorreto pois, como não foi utilizado FCS, não tem os bytes de verificação de erro como era suposto ter.

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

The screenshot shows a Wireshark capture on an Ethernet interface. The packet list displays four packets, with packet 62 selected. The packet details pane shows the structure of the selected packet: Ethernet II, Internet Protocol Version 4, Transmission Control Protocol, and Hypertext Transfer Protocol. The packet bytes pane shows the raw data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Info	Length
62	3.051722	192.168.100.193	193.136.19.20	HTTP	GET / HTTP/1.1	678
64	3.053262	193.136.19.20	192.168.100.193	HTTP	HTTP/1.1 304 Not Modified	185
92	3.328311	192.168.100.193	193.136.19.20	HTTP	GET /favicon.ico HTTP/1.1	585
94	3.330326	193.136.19.20	192.168.100.193	HTTP	HTTP/1.1 200 OK (text/plain)	1440

Frame 62: 678 bytes on wire (5424 bits), 678 bytes captured (5424 bits) on interface 0
Ethernet II, Src: AsustekC_47:1f:ab (74:d0:2b:47:1f:ab), 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)
Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.100.193, Dst: 193.136.19.20
Transmission Control Protocol, Src Port: 50650, Dst Port: 80, Seq: 1, Ack: 1, Len: 624
Hypertext Transfer Protocol
GET / HTTP/1.1
Host: miei.di.uminho.pt
Connection: keep-alive

R: O endereço Ethernet da fonte é 74:d0:2b:47:1f:ab. Este pertence à máquina virtual responsável pelo endereçamento da página, pois ao abrirmos a página <http://mei.di.uminho.pt>, o pedido vai ter de ser enviado para a rede que suporta o site, e a resposta provém deste.

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

The screenshot shows a Wireshark capture on an Ethernet interface. The packet list displays four packets, with packet 62 selected. The packet details pane shows the structure of the selected packet: Ethernet II, Internet Protocol Version 4, Transmission Control Protocol, and Hypertext Transfer Protocol. The packet bytes pane shows the raw data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Info	Length
62	3.051722	192.168.100.193	193.136.19.20	HTTP	GET / HTTP/1.1	678
64	3.053262	193.136.19.20	192.168.100.193	HTTP	HTTP/1.1 304 Not Modified	185
92	3.328311	192.168.100.193	193.136.19.20	HTTP	GET /favicon.ico HTTP/1.1	585
94	3.330326	193.136.19.20	192.168.100.193	HTTP	HTTP/1.1 200 OK (text/plain)	1440

Frame 62: 678 bytes on wire (5424 bits), 678 bytes captured (5424 bits) on interface 0
Ethernet II, Src: AsustekC_47:1f:ab (74:d0:2b:47:1f:ab), 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)
Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.100.193, Dst: 193.136.19.20
Transmission Control Protocol, Src Port: 50650, Dst Port: 80, Seq: 1, Ack: 1, Len: 624
Hypertext Transfer Protocol
GET / HTTP/1.1
Host: miei.di.uminho.pt
Connection: keep-alive

R: O endereço MAC do destino é 00:0c:29:d2:19:f0, que vai corresponder ao endereço MAC do computador que efetuou o pedido anteriormente.

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

The image shows a Wireshark packet capture analysis of an HTTP GET request. The top pane displays a list of captured packets, with packet 62 selected. The middle pane shows the details of the selected packet, and the bottom pane shows the raw packet data in hexadecimal and ASCII.

Packet List:

No.	Time	Source	Destination	Protocol	Info	Length
62	3.051722	192.168.100.193	193.136.19.20	HTTP	GET / HTTP/1.1	678
64	3.053262	193.136.19.20	192.168.100.193	HTTP	HTTP/1.1 304 Not Modified	185
92	3.328311	192.168.100.193	193.136.19.20	HTTP	GET /favicon.ico HTTP/1.1	585
94	3.330326	193.136.19.20	192.168.100.193	HTTP	HTTP/1.1 200 OK (text/plain)	1440

Packet Details (Frame 62):

- Frame 62: 678 bytes on wire (5424 bits), 678 bytes captured (5424 bits) on interface 0
- Ethernet II, Src: AsustekC_47:1f:ab (74:d0:2b:47:1f:ab), 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: AsustekC_47:1f:ab (74:d0:2b:47:1f:ab)
 - Type: IPv4 (0x0800)
 - Internet Protocol Version 4, Src: 192.168.100.193, Dst: 193.136.19.20
 - Transmission Control Protocol, Src Port: 50650, Dst Port: 80, Seq: 1, Ack: 1, Len: 624
 - Hypertext Transfer Protocol
 - GET / HTTP/1.1\r\n
 - Host: mie1.di.uminho.pt\r\n
 - Connection: keep-alive\r\n

Raw Data (Hex/ASCII):

```
0000  00 0c 29 d2 19 f0 74 d0 2b 47 1f ab 08 00 45 00  ..)...t. +G...E.
0010  02 98 25 e6 40 00 00 00 00 00 c0 a8 54 c1 c1 88  ..X.@...d...
0020  13 14 c5 da 00 50 24 ac 67 b3 50 00 e9 7e 50 18  ....P$.g.P.-P.
0030  01 00 fc 90 00 00 47 45 54 20 2f 20 48 54 54 50  ....GE T / HTTP
0040  2f 31 2e 31 0d 0a 48 6f 73 74 3a 20 6d 69 65 69  /1.1..Ho st: mie1
0050  2e 64 69 2e 75 6d 69 6e 68 6f 2e 70 74 0d 0a 43  .di.umin ho.pt..C
```

R: Os vários protocolos contidos nesta trama são o IPV4, HTTP E o TCP e Ethernet e o Frame.

Protocolo ARP

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

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

Interface: 172.26.36.216 --- 0xd
Internet Address    Physical Address    Type
172.26.254.254      00-d0-03-ff-94-00  dynamic
172.26.255.255      ff-ff-ff-ff-ff-ff  static
224.0.0.22          01-00-5e-00-00-16  static
224.0.0.251         01-00-5e-00-00-fb  static
224.0.0.252         01-00-5e-00-00-fc  static
239.255.255.250     01-00-5e-7f-ff-fa  static
255.255.255.255     ff-ff-ff-ff-ff-ff  static
```

R: A primeira coluna (Internet Adress) corresponde ao endereço IP, a segunda coluna (Physical Adress) corresponde ao MAC Address e a última coluna (Type) refere-se ao tipo, que pode ser estático ou dinâmico.

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?

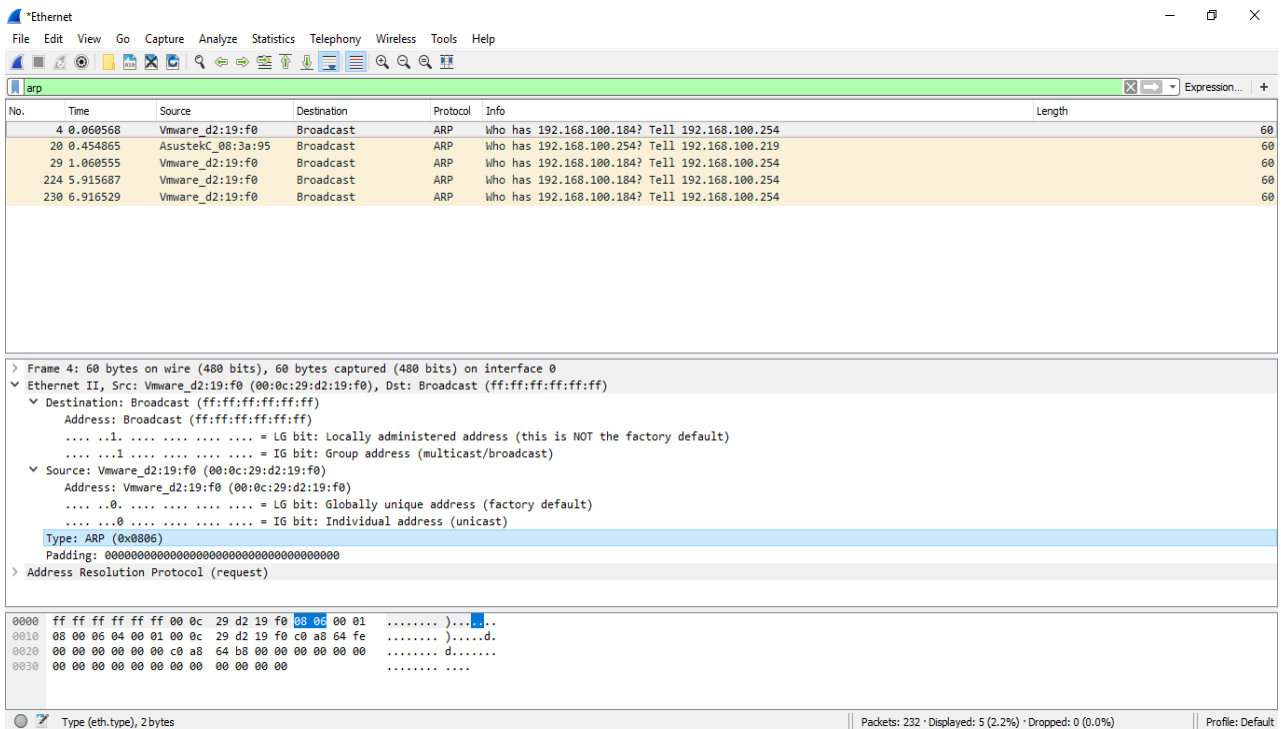
The screenshot shows a Wireshark capture of an ARP request packet. The packet list at the top shows an ARP request from VMware_d2:19:f0 to Broadcast. The packet details pane shows the Ethernet II header with Source: VMware_d2:19:f0 and Destination: Broadcast (ff:ff:ff:ff:ff:ff). The ARP section shows the request type and the target IP address 192.168.100.184.

Frame 4: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
Ethernet II, Src: VMware_d2:19:f0 (00:0c:29:d2:19:f0), 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: 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)
Type: ARP (0x0006)
Padding: 00000000000000000000000000000000
> Address Resolution Protocol (request)

0000 ff ff ff ff ff 00 0c 29 d2 19 f0 00 01
0010 00 00 00 00 00 00 00 00 00 00 00 00
0020 00 00 00 00 00 00 00 00 00 00 00 00
0030 00 00 00 00 00 00 00 00 00 00 00 00

R: O valor hexadecimal do endereço de origem é 00:0c:29:d2:19:f0 e o de destino é ff:ff:ff:ff:ff:ff. Isto indica que estamos a fazer um broadcast para a camada 2.

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



Wireshark packet capture showing an ARP request. The packet list shows five ARP requests from a VMware source to a broadcast destination. The packet details pane shows the Ethernet II header with a type of 0x0806 (ARP). The packet bytes pane shows the raw data with the type field highlighted in blue.

No.	Time	Source	Destination	Protocol	Info	Length
4	0.060568	Vmware_d2:19:f0	Broadcast	ARP	who has 192.168.100.184? Tell 192.168.100.254	60
20	0.454865	AsustekC_08:3a:95	Broadcast	ARP	who has 192.168.100.254? Tell 192.168.100.219	60
29	1.060555	Vmware_d2:19:f0	Broadcast	ARP	who has 192.168.100.184? Tell 192.168.100.254	60
224	5.915687	Vmware_d2:19:f0	Broadcast	ARP	who has 192.168.100.184? Tell 192.168.100.254	60
230	6.916529	Vmware_d2:19:f0	Broadcast	ARP	who has 192.168.100.184? Tell 192.168.100.254	60

> Frame 4: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
Ethernet II, Src: Vmware_d2:19:f0 (00:0c:29:d2:19:f0), 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: 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)
Type: ARP (0x0806)
Padding: 00000000000000000000000000000000
> Address Resolution Protocol (request)

0000 ff ff ff ff ff 00 0c 29 d2 19 f0 08 06 00 01)...
0010 08 00 06 04 00 01 00 0c 29 d2 19 f0 c0 a0 64 fe)....d.
0020 00 00 00 00 00 00 c0 a0 64 b5 00 00 00 00 00 d.....
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00

R: O valor hexadecimal do campo do tipo Ethernet é 0x0806, que significa que o ARP do tipo de Ethernet está associado a esse valor.

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>).

Wireshark interface showing ARP traffic. The packet list shows five ARP requests from VMware_d2:19:f0 to Broadcast. The packet details for the first packet (No. 4) show the ARP opcode as request (1). The packet bytes show the ARP opcode as 00 01.

No.	Time	Source	Destination	Protocol	Info	Length
4	0.060568	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
20	0.454865	AsustekC_08:3a:95	Broadcast	ARP	Who has 192.168.100.254? Tell 192.168.100.219	60
29	1.060555	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
224	5.915687	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
230	6.916529	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60

Frame 4: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
 Ethernet II, Src: Vmware_d2:19:f0 (00:0c:29:d2:19:f0), 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: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
 Sender IP address: 192.168.100.254
 Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
 Target IP address: 192.168.100.184

0000 ff ff ff ff ff ff 00 0c 29 d2 19 f0 00 06 00 01).....
 0010 00 00 06 04 00 01 00 0c 29 d2 19 f0 c0 a8 64 fed.....
 0020 00 00 00 00 00 00 c0 a8 64 b8 00 00 00 00 00d.....
 0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00d.....

Opcode (arp.opcode), 2 bytes

R: O valor do campo ARP opcode é 00 01, que nos diz que isto é relativo a um request ou a uma reply. Neste caso, vai ser um request como visto em cima [request (1)].

13. Identifique que tipo de endereços estão contidos na mensagem ARP? Que conclui?

No.	Time	Source	Destination	Protocol	Info	Length
4	0.060568	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
20	0.454865	AsustekC_08:3a:95	Broadcast	ARP	Who has 192.168.100.254? Tell 192.168.100.219	60
29	1.060555	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
224	5.915687	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
230	6.916529	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60

R: Há 2 IP's contido na mensagem ARP, como podemos observar na imagem acima, estando 1 endereço a seguir a "Tell" que é 192.168.100.254 sendo o nosso IP e o outro a seguir a "Who has" 192.168.100.219. Concluimos então que conseguimos obter um endereço para conectarmos ao nosso.

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

No.	Time	Source	Destination	Protocol	Info	Length
4	0.060568	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
20	0.454865	AsustekC_08:3a:95	Broadcast	ARP	Who has 192.168.100.254? Tell 192.168.100.219	60
29	1.060555	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
224	5.915687	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
230	6.916529	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60

R: Nós conectámos a nossa máquina a uma máquina dos nossos colegas. Para fazermos isso, enviamos um request (broadcast) a todas as máquinas que estão ligadas à nossa rede Ethernet a perguntar quem tem o IP 192.168.100.254 ("Who

has 192.168.100.254?”), obtendo assim o seu endereço MAC para o conectarmos ao nosso.

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

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

Wireshark capture of ARP traffic. The packet list shows five ARP requests from 192.168.100.254 to the broadcast address. The packet details for the first request (No. 4) are expanded, showing the ARP opcode as 'request (1)'. The packet bytes section shows the raw data with the opcode '00 01' highlighted in blue.

No.	Time	Source	Destination	Protocol	Info	Length
4	0.060568	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
20	0.454865	AsustekC_08:3a:95	Broadcast	ARP	Who has 192.168.100.254? Tell 192.168.100.219	60
29	1.060555	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
224	5.915687	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
230	6.916529	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60

Frame 230: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
Ethernet II, Src: Vmware_d2:19:f0 (00:0c:29:d2:19:f0), 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: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
Sender IP address: 192.168.100.254
Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
Target IP address: 192.168.100.184

0000 ff ff ff ff ff ff 0c 29 d2 19 f0 08 06 00 01).....
0010 08 00 06 04 00 01 00 0c 29 d2 19 f0 c0 a8 64 fed.....
0020 00 00 00 00 00 00 c0 a8 64 b8 00 00 00 00 00 d.....
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00

R: A resposta que nós procuramos é o endereço MAC relativo ao endereço IP procurado (“00:0c:29:d2:19:f0”), como é possível observar na primeira imagem. Relativamente ao valor do campo ARP opcode, vai ser 00 01. Ao estarmos a especificar o reply (1), estamos a ir buscar o valor do broadcast que fizemos.

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

*Ethernet

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

arp

No.	Time	Source	Destination	Protocol	Info	Length
4	0.060568	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
20	0.454865	AsustekC_08:3a:95	Broadcast	ARP	Who has 192.168.100.254? Tell 192.168.100.219	60
29	1.060555	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
224	5.915687	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60
230	6.916529	Vmware_d2:19:f0	Broadcast	ARP	Who has 192.168.100.184? Tell 192.168.100.254	60

> Frame 230: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
 > Ethernet II, Src: Vmware_d2:19:f0 (00:0c:29:d2:19:f0), 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: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
 Sender IP address: 192.168.100.254
 Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
 Target IP address: 192.168.100.184

```

0000 ff ff ff ff ff 00 0c 29 d2 19 f0 08 06 00 01 .....).
0010 08 00 06 04 00 01 00 0c 29 d2 19 f0 c0 a8 64 fe .....).d.
0020 00 00 00 00 00 00 c0 a8 64 b8 00 00 00 00 00 .....d.....
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
  
```

Target MAC address (arp.dst.hw_mac), 6 bytes

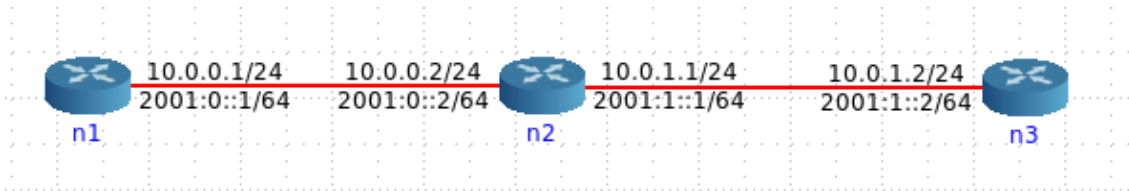
Packets: 232 · Displayed: 5 (2.2%) · Dropped: 0 (0.0%)

Profile: Default

R: A resposta ao pedido ARP vai estar no “Target MAC address” (00:00:00:00:00:00).

ARP numa topologia CORE

No emulador CORE prepare uma topologia com 3 routers em que n1 liga a n2 e este a n3.



16. Com auxílio do comando `ifconfig` obtenha os endereços Ethernet das interfaces dos diversos routers.

```
eth0    Link encap:Ethernet  HWaddr 00:00:00:aa:00:00  
        inet addr:10.0.0.1  Bcast:0.0.0.0  Mask:255.255.255.0
```

Endereço do Ethernet do router n1;

```
eth0    Link encap:Ethernet  HWaddr 00:00:00:aa:00:01  
        inet addr:10.0.0.2  Bcast:0.0.0.0  Mask:255.255.255.0
```

Endereço do Ethernet do router n2(eth0);

```
eth1    Link encap:Ethernet  HWaddr 00:00:00:aa:00:02  
        inet addr:10.0.1.1  Bcast:0.0.0.0  Mask:255.255.255.0
```

Endereço do Ethernet do router n2(eth1);

```
eth0    Link encap:Ethernet  HWaddr 00:00:00:aa:00:03  
        inet addr:10.0.1.2  Bcast:0.0.0.0  Mask:255.255.255.0
```

Endereço do Ethernet do router n3;

17. Usando o comando `arp` obtenha as caches arp dos diversos sistemas.

As caches ARP estão vazias.

18. Faça ping de n1 para n2. Que modificações observa nas caches ARP desses sistemas? Faça ping de n1 para n3. Consulte as caches ARP. Que conclui?

```
> n1 > arp:
Address          HWtype  HWaddress      Flags Mask    Iface
10.0.0.2          ether    00:00:00:aa:00:01  C             eth0

> n2 > arp:
Address          HWtype  HWaddress      Flags Mask    Iface
10.0.0.1          ether    00:00:00:aa:00:00  C             eth0

> n3 > arp:
```

R: n1 para n2

Pela análise da cache de n1, é possível reparar que o endereço 10.0.0.2 está ligado a 00:00:00:aa:00:01.

```
> n1 > arp:
Address          HWtype  HWaddress      Flags Mask    Iface
10.0.0.2          ether    00:00:00:aa:00:01  C             eth0

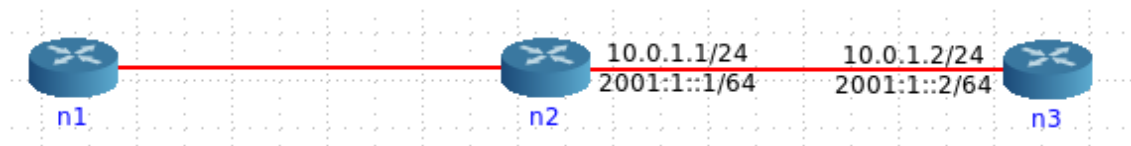
> n2 > arp:
Address          HWtype  HWaddress      Flags Mask    Iface
10.0.0.1          ether    00:00:00:aa:00:00  C             eth0
10.0.1.2          ether    00:00:00:aa:00:03  C             eth1

> n3 > arp:
Address          HWtype  HWaddress      Flags Mask    Iface
10.0.1.1          ether    00:00:00:aa:00:02  C             eth0
```

R: n1 para n3

Pela análise da cache de n1, é possível reparar que o endereço 10.0.0.2 está ligado a 00:00:00:aa:00:01.

19. Em n1 remova a entrada correspondente a n2. Coloque uma nova entrada para n2 com endereço Ethernet inexistente. O que acontece?



```
> n1 > arp:
```

```
> n2 > arp:
```

Address	HWtype	HWaddress	Flags	Mask	Iface
10.0.1.2	ether	00:00:00:aa:00:03	C		eth1

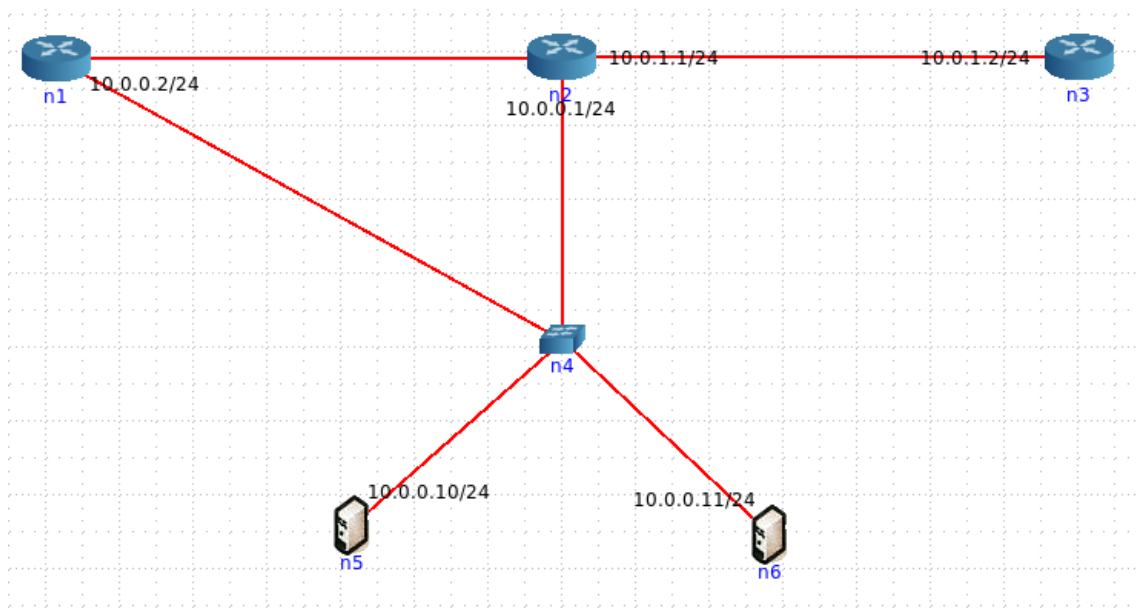
```
> n3 > arp:
```

Address	HWtype	HWaddress	Flags	Mask	Iface
10.0.1.1	ether	00:00:00:aa:00:02	C		eth0

R: Pela imagem, podemos ver que já não existe relação ao contrário da alínea anterior e isto deve-se a remoção da ligação entre Ethernet entre n1 e n2.

Adicione agora um switch (n4) à rede e ligue o router n1, e os hosts n5 e n6 a esse switch.

20 . Faça ping de n6 para n5. Sem consultar a tabela ARP anote a entrada que, em sua opinião, é criada na tabela ARP de n6. Verifique, justificando, se a sua interpretação sobre a operação da rede Ethernet e protocolo ARP estava correto.



A nossa interpretação é devido a um switch encarregar-se a dados entre o destino e a origem no ARP de bash de n5 deverá aparecer o IP de n6 e á frente o endereço de n6.

```

> n2 > arp:
Address          HWtype  HWaddress     Flags Mask    Iface
10.0.1.2         ether   00:00:00:aa:00:03  C           eth1
10.0.0.2         ether   00:00:00:aa:00:05  C           eth2

> n3 > arp:
Address          HWtype  HWaddress     Flags Mask    Iface
10.0.1.1         ether   00:00:00:aa:00:02  C           eth0

> arp
> n5 > arp:
Address          HWtype  HWaddress     Flags Mask    Iface
10.0.0.11        ether   00:00:00:aa:00:07  C           eth0

> n6 > arp:
Address          HWtype  HWaddress     Flags Mask    Iface
10.0.0.10        ether   00:00:00:aa:00:06  C           eth0

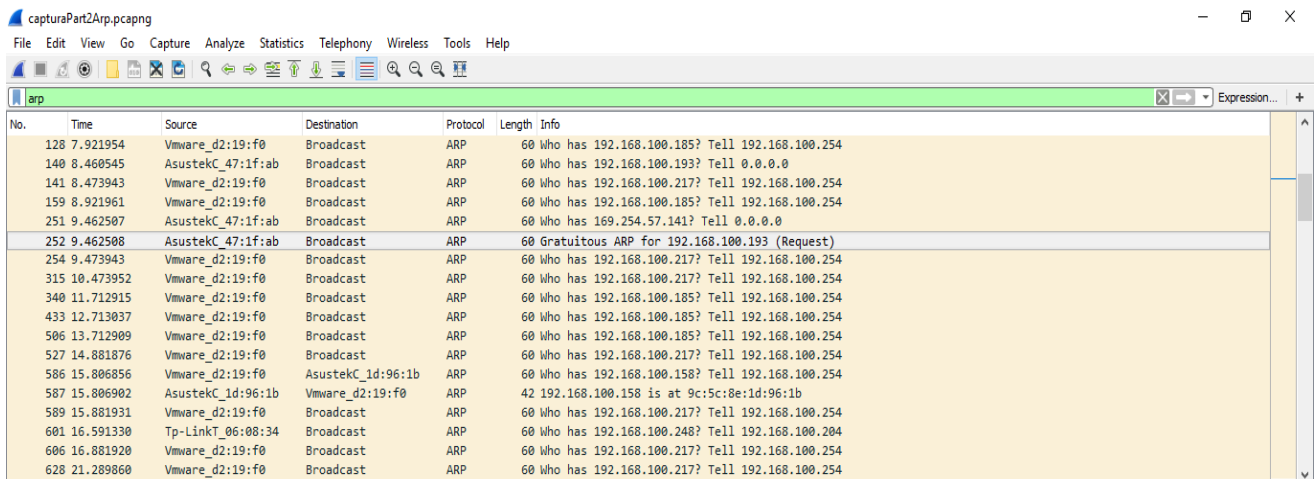
```

R: Pela imagem apresentada acima, é possível verificar que a nossa interpretação feita estava correta. Em n6, aparece também uma nova entrada com o IP de n5.

PARTE 2

ARP Gratuito

1. Identifique um pacote de pedido ARP gratuito originado pelo seu sistema. Verifique quantos pacotes ARP gratuito foram enviados e com que intervalo temporal?

A screenshot of the Wireshark network protocol analyzer interface. The title bar shows 'caputraPart2Arp.pcapng'. The menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, and Help. The toolbar contains various icons for file operations, capture control, and analysis. The main display area is titled 'arp' and shows a list of captured packets. The packet list table has columns for No., Time, Source, Destination, Protocol, Length, and Info. The selected packet is number 252, a gratuitous ARP request from AsustekC_47:1f:ab to the broadcast address. The packet details pane on the right shows the structure of the ARP request.

No.	Time	Source	Destination	Protocol	Length	Info
128	7.921954	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
140	8.460545	AsustekC_47:1f:ab	Broadcast	ARP	60	Who has 192.168.100.193? Tell 0.0.0.0
141	8.473943	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.217? Tell 192.168.100.254
159	8.921961	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
251	9.462507	AsustekC_47:1f:ab	Broadcast	ARP	60	Who has 169.254.57.141? Tell 0.0.0.0
252	9.462508	AsustekC_47:1f:ab	Broadcast	ARP	60	Gratuitous ARP for 192.168.100.193 (Request)
254	9.473943	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.217? Tell 192.168.100.254
315	10.473952	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.217? Tell 192.168.100.254
340	11.712915	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
433	12.713037	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
506	13.712909	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.185? Tell 192.168.100.254
527	14.881876	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.217? Tell 192.168.100.254
586	15.806856	Vmware_d2:19:f0	AsustekC_id:96:1b	ARP	60	Who has 192.168.100.158? Tell 192.168.100.254
587	15.806902	AsustekC_id:96:1b	Vmware_d2:19:f0	ARP	42	192.168.100.158 is at 9c:5c:8e:1d:96:1b
589	15.881931	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.217? Tell 192.168.100.254
601	16.591330	Tp-LinkT_06:08:34	Broadcast	ARP	60	Who has 192.168.100.248? Tell 192.168.100.204
606	16.881920	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.217? Tell 192.168.100.254
628	21.289860	Vmware_d2:19:f0	Broadcast	ARP	60	Who has 192.168.100.217? Tell 192.168.100.254

R: Foi enviado um pacote ARP gratuito que demorou 9.46s.

2. 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?

ARP Gratuito

```
Hardware size: 6
Protocol size: 4
Opcode: request (1)
[Is gratuitous: True]
Sender MAC address: AsustekC_47:1f:ab (74:d0:2b:47:1f:ab)
Sender IP address: 192.168.100.193
Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
Target IP address: 192.168.100.193
```

ARP Normal

```
Hardware size: 6
Protocol size: 4
Opcode: request (1)
Sender MAC address: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
Sender IP address: 192.168.100.254
Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
Target IP address: 192.168.100.217
```

R: Num ARP Normal, é procurado o MAC correspondente a um certo IP. No “Gratuito”, a máquina questiona-se a si própria sobre qual o MAC que corresponde ao próprio IP, para descobrir se tem mais alguma máquina a usar o nosso IP.

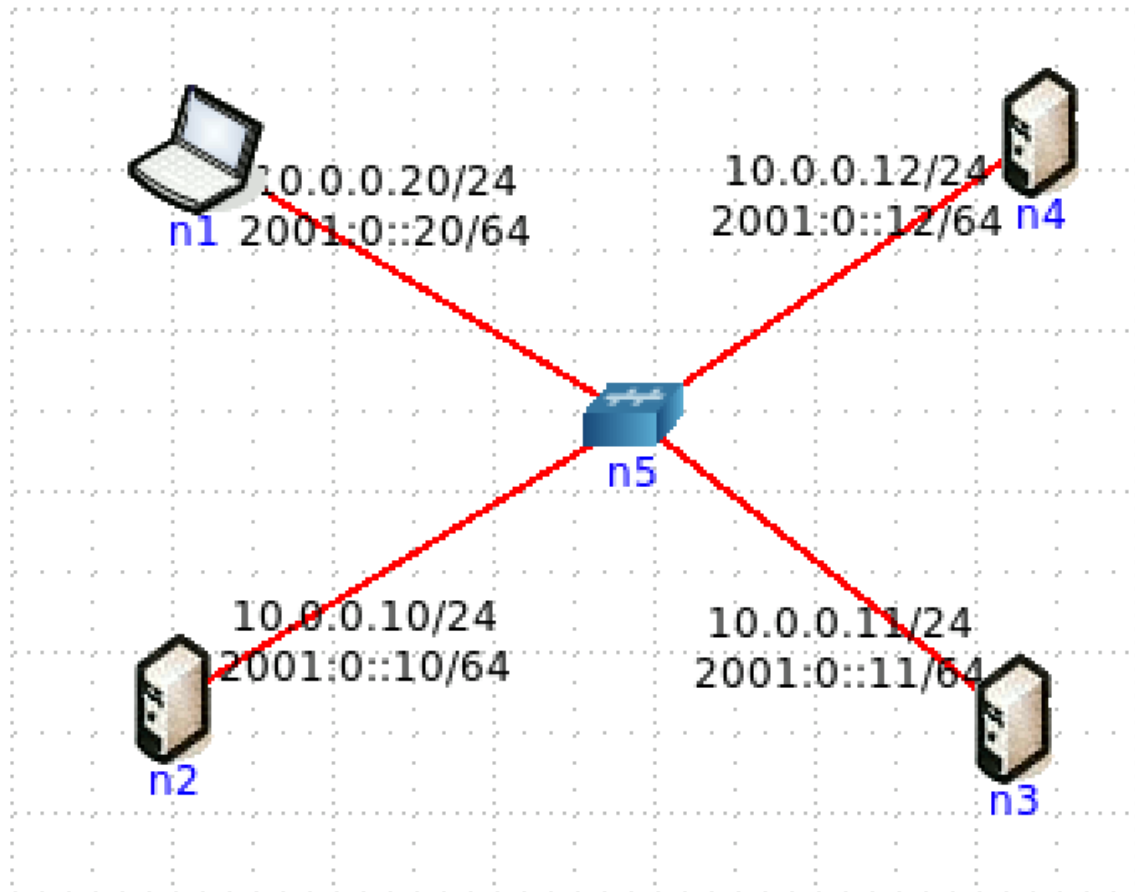
A trama Ethernet que corresponde ao ARP Gratuito é a seguinte:

0000	ff ff ff ff ff ff 74 d0 2b 47 1f ab 08 06 00 01t. +G.....
0010	08 00 06 04 00 01 74 d0 2b 47 1f ab c0 a8 64 c1t. +G....d.
0020	00 00 00 00 00 00 c0 a8 64 c1 00 00 00 00 00 d.....
0030	00 00 00 00 00 00 00 00 00 00 00 00

O resultado esperado face ao pedido ARP Gratuito enviado será “verdadeiro” se a nossa máquina for a única a utilizar este endereço IP.

Domínios de Colisão

1. Faça ping de n1 para n4. Verifique com a opção tcpdump como flui o tráfego nas diversas interfaces dos vários dispositivos. Que conclui?

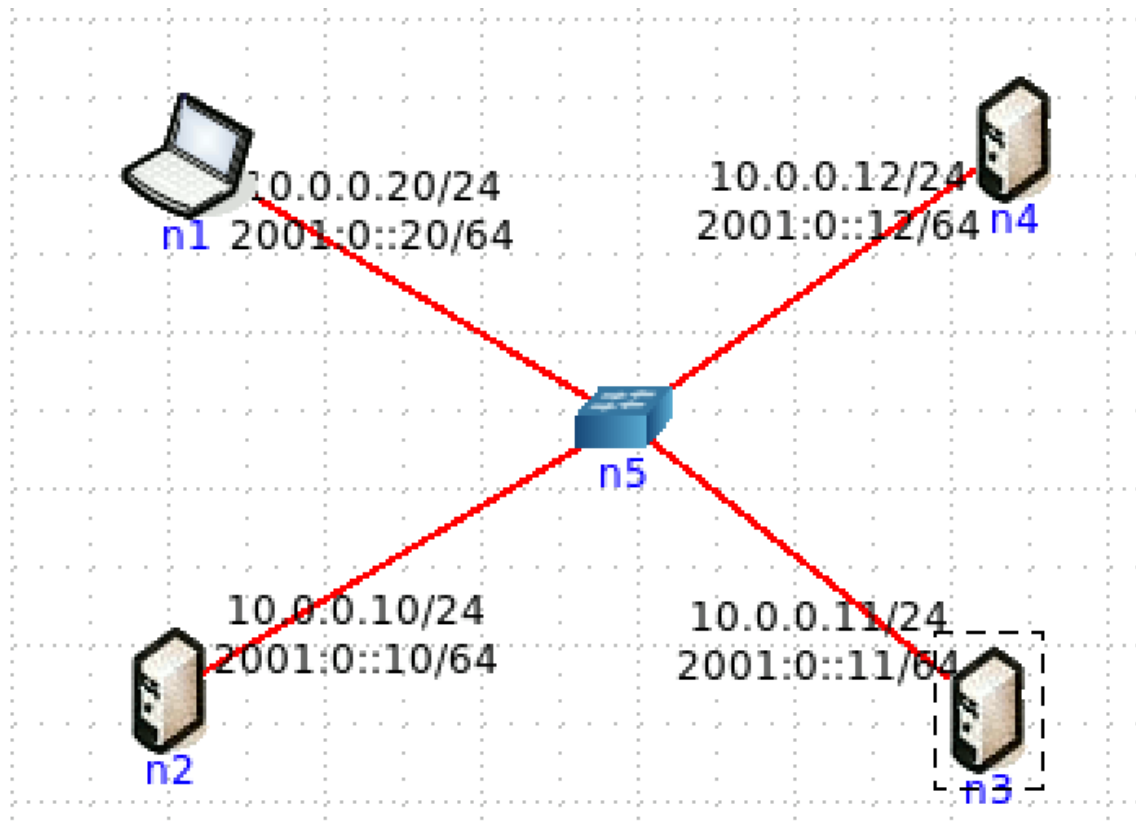


<pre>vcmd root@n1:/tmp/pycore.40680/n1.conf# ping 10.0.0.12 PING 10.0.0.12 (10.0.0.12) 56(84) bytes of data. 64 bytes from 10.0.0.12: icmp_req=1 ttl=64 time=0.061 ms ^C --- 10.0.0.12 ping statistics --- 1 packets transmitted, 1 received, 0% packet loss, time 0ms rtt min/avg/max/mdev = 0.061/0.061/0.061/0.000 ms root@n1:/tmp/pycore.40680/n1.conf#</pre>	<pre>vcmd root@n2:/tmp/pycore.40680/n2.conf# tcpdump -i eth0 tcpdump: verbose output suppressed, use -v or -vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes ^C07:54:59.079705 IP 10.0.0.20 > 10.0.0.12: ICMP echo request, id 44, seq 1, length 64 07:54:59.079728 IP 10.0.0.12 > 10.0.0.20: ICMP echo reply, id 44, seq 1, length 64 2 packets captured 2 packets received by filter 0 packets dropped by kernel root@n2:/tmp/pycore.40680/n2.conf#</pre>
<pre>vcmd root@n3:/tmp/pycore.40680/n3.conf# tcpdump -i eth0 tcpdump: verbose output suppressed, use -v or -vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes ^C07:54:59.079702 IP 10.0.0.20 > 10.0.0.12: ICMP echo request, id 44, seq 1, length 64 07:54:59.079727 IP 10.0.0.12 > 10.0.0.20: ICMP echo reply, id 44, seq 1, length 64 2 packets captured 2 packets received by filter 0 packets dropped by kernel root@n3:/tmp/pycore.40680/n3.conf#</pre>	<pre>vcmd root@n4:/tmp/pycore.40680/n4.conf# tcpdump -i eth0 tcpdump: verbose output suppressed, use -v or -vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes ^C07:54:59.079699 IP 10.0.0.20 > 10.0.0.12: ICMP echo request, id 44, seq 1, length 64 07:54:59.079717 IP 10.0.0.12 > 10.0.0.20: ICMP echo reply, id 44, seq 1, length 64 2 packets captured 2 packets received by filter 0 packets dropped by kernel root@n4:/tmp/pycore.40680/n4.conf#</pre>

R: No terminal do canto superior esquerdo, está o ping do computador n1 para o host n4, no superior direito está o tcpdump de n2 e em baixo encontra-se o tcpdump de n3 e, por fim, no canto inferior direito encontra-se o tcpdump de n4.

Como temos um hub, há partilha de meio e o pedido vai fluir por todas as máquinas conectadas ao hub.

2. 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.



<pre>vcmd root@n1:/tmp/pycore.40681/n1.conf# ping 10.0.0.12 PING 10.0.0.12 (10.0.0.12) 56(84) bytes of data: 64 bytes from 10.0.0.12: icmp_req=1 ttl=64 time=0.120 ms ^C --- 10.0.0.12 ping statistics --- 1 packets transmitted, 1 received, 0% packet loss, time 0ms rtt min/avg/max/mdev = 0.120/0.120/0.120/0.000 ms root@n1:/tmp/pycore.40681/n1.conf#</pre>	<pre>vcmd root@n4:/tmp/pycore.40681/n4.conf# tcpdump -i eth0 tcpdump: verbose output suppressed, use -v or -vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes ^C08:00:46.595958 ARP, Request who-has 10.0.0.12 tell 10.0.0.20, length 28 08:00:46.595991 ARP, Reply 10.0.0.12 is-at 00:00:00:aa:00:03 (oui Ethernet), len gth 28 08:00:46.596023 IP 10.0.0.20 > 10.0.0.12: ICMP echo request, id 28, seq 1, lengt h 64 08:00:46.596032 IP 10.0.0.12 > 10.0.0.20: ICMP echo reply, id 28, seq 1, length 64 4 packets captured 4 packets received by filter 0 packets dropped by kernel root@n4:/tmp/pycore.40681/n4.conf#</pre>
<pre>vcmd root@n2:/tmp/pycore.40681/n2.conf# tcpdump -i eth0 tcpdump: verbose output suppressed, use -v or -vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes ^C08:00:46.595965 ARP, Request who-has 10.0.0.12 tell 10.0.0.20, length 28 1 packet captured 1 packet received by filter 0 packets dropped by kernel root@n2:/tmp/pycore.40681/n2.conf#</pre>	<pre>vcmd root@n3:/tmp/pycore.40681/n3.conf# tcpdump -i eth0 tcpdump: verbose output suppressed, use -v or -vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes ^C08:00:46.595962 ARP, Request who-has 10.0.0.12 tell 10.0.0.20, length 28 1 packet captured 1 packet received by filter 0 packets dropped by kernel root@n3:/tmp/pycore.40681/n3.conf#</pre>

R: No terminal do canto superior esquerdo, está o ping do computador n1 para o host n4, no superior direito está o tcpdump de n4 e em baixo no lado esquerdo encontra-se o tcpdump de n2. No lado direito em baixo encontra-se o o tcpdump de n3.

Como temos um switch, não há partilha de meio e o pedido vai apenas passar pelas máquinas envolvidas no ping.

Os switches eliminam as colisões, conectando cada dispositivo a uma porta do computador, enquanto que os hubs permitem que estas colisões possam existir. Ao utilizar um hub para comunicar entre um computador e um host, também passava tráfego pelas outras máquinas do esquema, no entanto, ao mudarmos para o switch, observamos que apenas passava tráfego pelo computador e pelo host ao efetuar o ping, eliminando assim, o tráfego pelas outras máquinas e evitando colisões.

Conclusões

Neste nosso trabalho de Redes de Computadores foram abordados vários temas, incluindo a camada de ligação lógica, direcionada para a tecnologia Ethernet, e do protocolo ARP, aprendendo como funcionam os endereços MAC e outros subjects essenciais abordados no nosso relatório.

Mais especificamente em relação à ligação lógica falámos de transferência de dados, deteção e correção de erros, protocolos de acesso de controlo de ligação, endereços MAC (como dito anteriormente), Address Resolution Protocol, Ethernet e interligação de redes locais.

Aprendemos também como funcionam as tramas Ethernet, respondendo às várias questões propostas pelo docente.

Por último, na parte 2 do protocolo ARP abordamos o ARP gratuito e domínios de colisão. E com o CORE fomos funcionando com os vários tipos de ligação, as diferenças entre HUB"s e Switch"s.