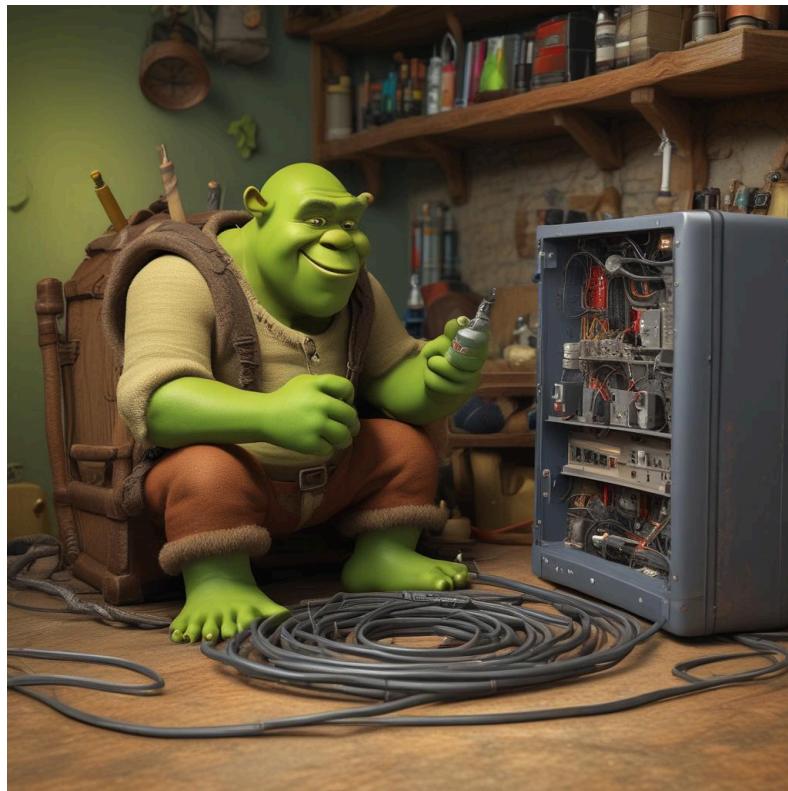


Relatório TP3 | Redes de Computadores

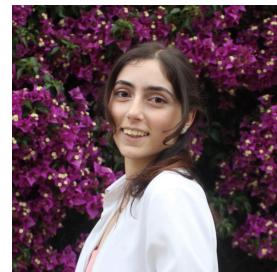
Grupo 31 | 2023/2024



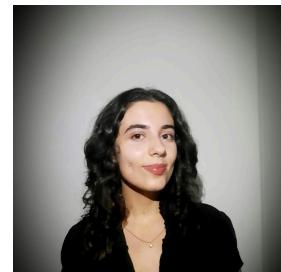
João Lobo
(A104356)



Mariana Rocha
(A90817)



Rita Camacho
(A104439)



1^a Parte

1. Captura e análise de Tramas Ethernet

Com o aumento do preço da habitação em Braga, o **Shrek** e o **Burro** tomam a decisão economicamente sensata e decidem voltar à sua casa no Pântano. A sua rede local é constituída por um *switch* (**n2**), um *router* para acesso à rede (**n1**), assim como os portáteis do **Shrek** e do **Burro**, ligados por *Ethernet* a **n2**. O *router* **n1** está ainda ligado a um *hub* (**n3**), que se conecta ao portátil da **Fiona** e ao servidor do conhecido *site* de notícias pantanews.com.

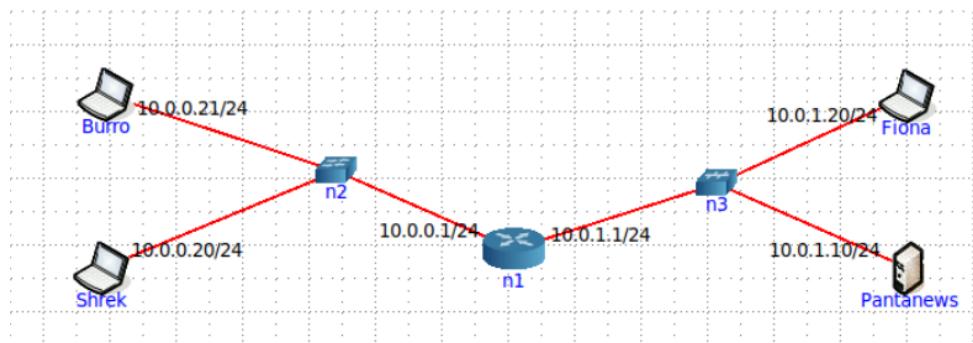


Figura 1: Topologia

A caminho, o **Shrek** fica a saber que houve um ataque aos servidores do seu *site* de notícias favorito, o **Pantanews**, e que todos os seus dados terão sido apagados. Assim que chegam a casa, o **Shrek** aproveita para verificar se realmente há algum problema com o *site* (servidor - 10.0.1.10). Utilize o comando `curl` para o efeito (poderá consultar o manual do comando com `man curl`), apontando diretamente para o endereço do servidor. Pare a captura do *Wireshark*, e analise a trama que contém os primeiros dados HTTP referentes à página alojada no servidor.

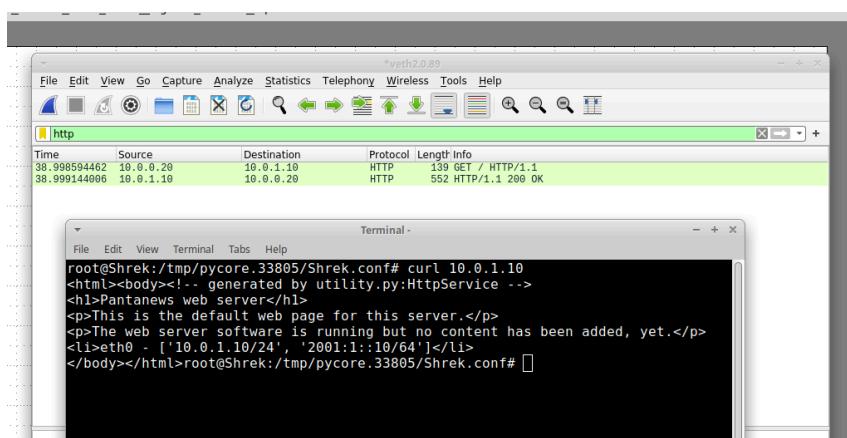


Figura 2: Execução do comando `curl 10.0.1.10`

1.1.

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

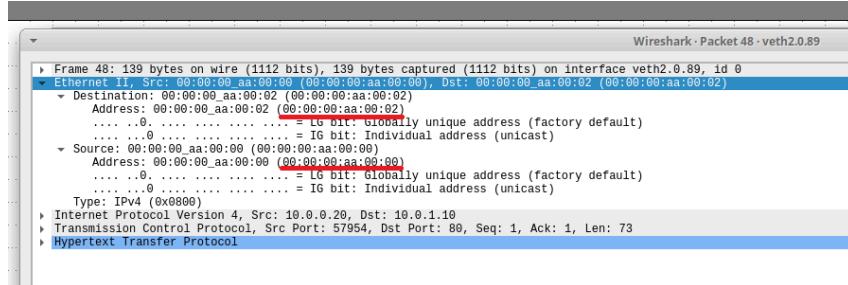


Figura 3: Endereços MAC de origem e de destino da trama captura (GET)

Endereço MAC de Origem (**00:00:00:aa:00:00**): O endereço MAC de origem pertence ao **Shrek**, pois é o dispositivo que está a enviar a solicitação HTTP para o servidor do **Pantanews**. Como o **Shrek** está a solicitar, o endereço MAC de origem é o seu próprio endereço.

Endereço MAC de Destino (**00:00:00:aa:00:02**): O endereço MAC de destino pertence ao *router n1* que está na rede local. Quando o **Shrek** envia uma solicitação HTTP para o servidor do **Pantanews**, o pacote é primeiro enviado para o *router n1*, de seguida este encaminha o pacote para o servidor (**10.0.1.10**). De modo que, o endereço MAC de destino é o do **n1**, porque é o próximo salto na rede para chegar ao destino final.

Pode-se verificar que o endereço MAC de origem é do **Shrek** recorrendo ao comando `ifconfig`.

```
root@Shrek:/tmp/pycore.33805/Shrek.conf# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 10.0.0.20 netmask 255.255.255.0 broadcast 0.0.0.0
        inet6 fe80::200:ff:feaa:0 prefixlen 64 scopeid 0x20<link>
        inet6 2001::20 prefixlen 64 scopeid 0x0<global>
        ether 00:00:00:aa:00:00 txqueuelen 1000 (Ethernet)
        RX packets 442 bytes 38210 (38.2 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 22 bytes 1733 (1.7 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figura 4: Verificação do endereço MAC de origem

Pode-se verificar que o endereço MAC de destino é do *router n1* recorrendo ao comando `ifconfig`.

```
root@n1:/tmp/pycore.33805/n1.conf# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 10.0.0.1 netmask 255.255.255.0 broadcast 0.0.0.0
        inet6 fe80::200:ff:feaa:2 prefixlen 64 scopeid 0x20<link>
        inet6 2001::1 prefixlen 64 scopeid 0x0<global>
        ether 00:00:00:aa:00:02 txqueuelen 1000 (Ethernet)
        RX packets 97 bytes 9835 (9.8 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 417 bytes 34040 (34.0 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figura 5: Verificação do endereço MAC de destino

1.2.

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

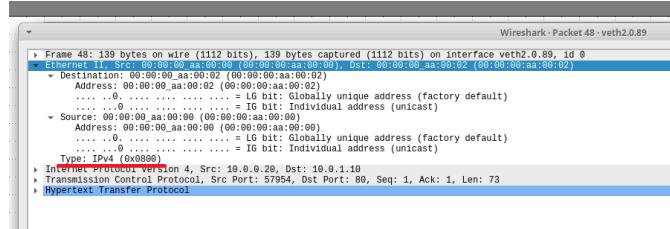


Figura 6: Valor hexadecimal do campo **Type** da trama *Ethernet*

O valor hexadecimal do campo **Type** da trama *Ethernet* é **0x0800**. Neste contexto, **0x0800** é associado ao protocolo IPv4. Quando um dispositivo de rede vê o valor **0x0800** no campo **Type** de uma trama *Ethernet*, ele percebe que o conteúdo da trama é um pacote IPv4.

1.3.

Quantos *bytes* são usados no encapsulamento protocolar, i.e. desde o início da trama até ao início dos dados do nível aplicacional? Calcule e indique, em percentagem, a sobrecarga (*overhead*) introduzida pela pilha protocolar.

Sendo no total **139 bytes**, apenas **16** destes são usados no encapsulamento protocolar. Este valor total de **139 bytes** divide-se da seguinte forma:

- **73 bytes** correspondem a **http**
- **16 bytes** correspondem a **get**, sendo utilizados no encapsulamento protocolar
- **32 bytes** correspondem a **tcp**
- **20 bytes** correspondem a **ipv4**
- **14 bytes** correspondem a **etherent**

$$\text{Sobrecarga} = \frac{16}{139} \times 100 \simeq 11.5\%$$

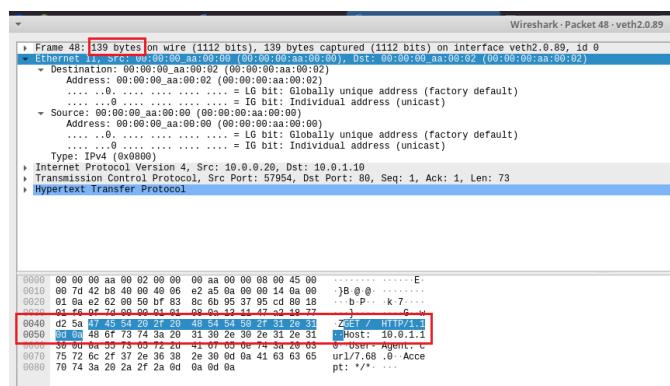
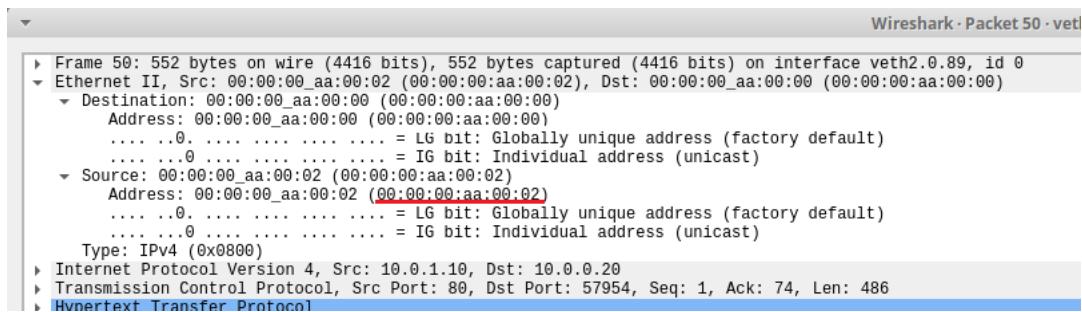


Figura 7: Número de *bytes* utilizados no encapsulamento protocolar

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

1.4.

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



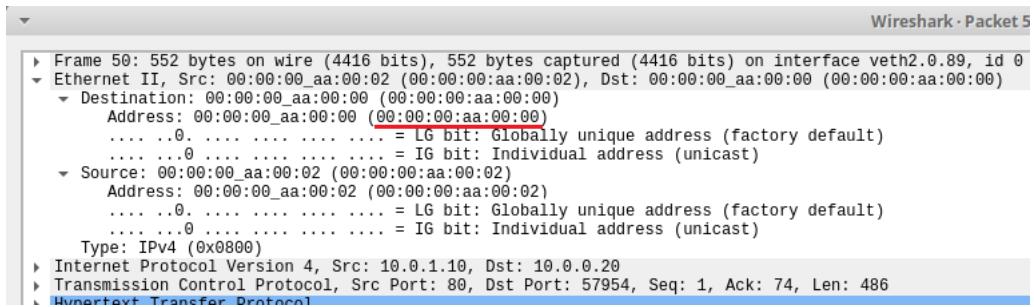
```
Frame 50: 552 bytes on wire (4416 bits), 552 bytes captured (4416 bits) on interface veth2.0.89, id 0
  Ethernet II, Src: 00:00:00_aa:00:02 (00:00:00:aa:00:02), Dst: 00:00:00_aa:00:00 (00:00:00:aa:00:00)
    Destination: 00:00:00_aa:00:00 (00:00:00:aa:00:00)
      Address: 00:00:00_aa:00:00 (00:00:00:aa:00:00)
        ....0. .... .... .... = LG bit: Globally unique address (factory default)
        ....0. .... .... .... = IG bit: Individual address (unicast)
    Source: 00:00:00_aa:00:02 (00:00:00:aa:00:02)
      Address: 00:00:00_aa:00:02 (00:00:00:aa:00:02)
        ....0. .... .... .... = LG bit: Globally unique address (factory default)
        ....0. .... .... .... = IG bit: Individual address (unicast)
    Type: IPv4 (0x0800)
  Internet Protocol Version 4, Src: 10.0.1.10, Dst: 10.0.0.20
  Transmission Control Protocol, Src Port: 80, Dst Port: 57954, Seq: 1, Ack: 74, Len: 486
  Hypertext Transfer Protocol
```

Figura 8: Endereço MAC de origem da trama capturada

Através da análise da captura, verifica-se que o endereço *Ethernet* da fonte (MAC *Address*) é **00:00:00:aa:00:02**, correspondendo ao *router n1* que está na rede local, responsável por reencaminhar o último salto que o pacote vindo do servidor terá de efetuar para chegar ao destino final.

1.5.

Qual é o endereço MAC do destino? A que interface corresponde?



```
Frame 50: 552 bytes on wire (4416 bits), 552 bytes captured (4416 bits) on interface veth2.0.89, id 0
  Ethernet II, Src: 00:00:00_aa:00:02 (00:00:00:aa:00:02), Dst: 00:00:00_aa:00:00 (00:00:00:aa:00:00)
    Destination: 00:00:00_aa:00:00 (00:00:00:aa:00:00)
      Address: 00:00:00_aa:00:00 (00:00:00:aa:00:00)
        ....0. .... .... .... = LG bit: Globally unique address (factory default)
        ....0. .... .... .... = IG bit: Individual address (unicast)
    Source: 00:00:00_aa:00:02 (00:00:00:aa:00:02)
      Address: 00:00:00_aa:00:02 (00:00:00:aa:00:02)
        ....0. .... .... .... = LG bit: Globally unique address (factory default)
        ....0. .... .... .... = IG bit: Individual address (unicast)
    Type: IPv4 (0x0800)
  Internet Protocol Version 4, Src: 10.0.1.10, Dst: 10.0.0.20
  Transmission Control Protocol, Src Port: 80, Dst Port: 57954, Seq: 1, Ack: 74, Len: 486
  Hypertext Transfer Protocol
```

Figura 9: Endereço MAC de destino da trama capturada

Na captura da trama em análise, o endereço MAC de destino é **00:00:00:aa:00:00** e corresponde ao **Shrek**, destinatário do pacote enviado pelo servidor.

Através da execução do comando **ifconfig** na interface **Shrek**, confirma-se este endereço.

2. Protocolo ARP e Domínios de Colisão

Deverá ter a cache ARP completamente vazia antes de iniciar esta secção: reinicie a topologia, ou utilize o comando `arp -d`. Um pouco mais preocupado com a segurança dos seus dados, o **Shrek** repara que a **Fiona** sabe sempre por onde andou a navegar. Para averiguar esta situação, o **Shrek** experimenta de novo aceder ao *site do pantanews.com* (`10.0.1.10`) através do comando `curl`. Certifique-se que está a capturar tráfego com o *Wireshark* na interface do **Shrek** e na do **Burro**.

2.1.

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

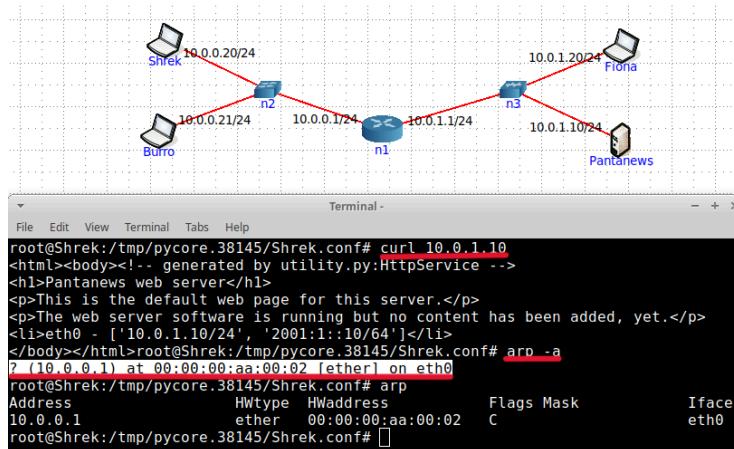


Figura 10: Tabela obtida na execução do comando `arp -a`

Significado das colunas da tabela resultante:

- `10.0.0.1` → IP de **n1**;
- `00:00:00:aa:00:02` → MAC *Address* de **n1**;
- `[ether]` → *HWTyoe* de MAC *Address*;
- `eth0` → Interface do **n1**.

2.2.

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

2.2. a)

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

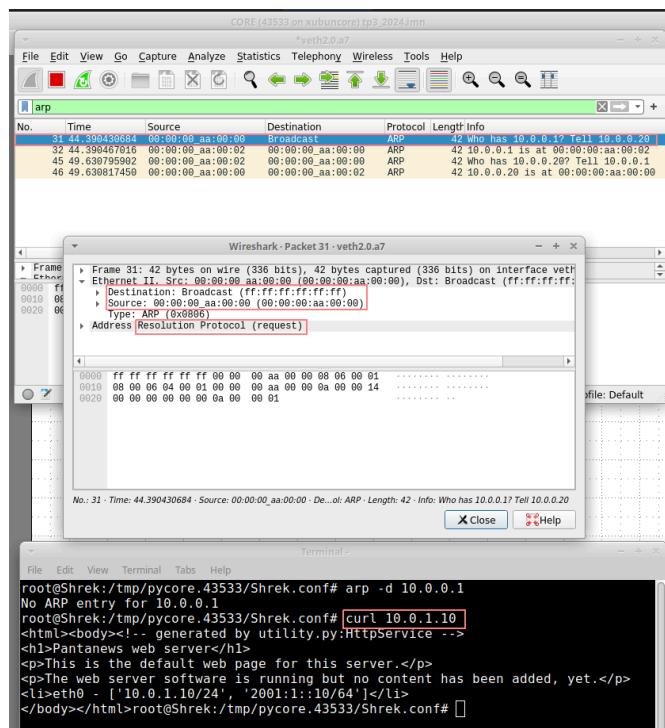


Figura 11: Execução do comando `curl 10.0.1.10`

Endereço MAC de Origem (`00:00:00:aa:00:00`): O endereço MAC de origem pertence ao **Shrek**, pois é o dispositivo que está a enviar a solicitação HTTP para o servidor do **Pantanews**. Como o **Shrek** está a solicitar, o endereço MAC de origem é o seu próprio endereço.

Endereço MAC de Destino (`ff:ff:ff:ff:ff:ff`): Como a cache ARP está completamente vazia, o dispositivo precisa descobrir o endereço MAC correspondente ao endereço IP **10.0.1.10** antes de poder enviar os pacotes para o servidor. Como não tem essa informação disponível (porque a cache está vazia), envia um ARP *request* para toda a rede (*broadcast*), aguardando que o dispositivo com o endereço IP **10.0.1.10** responda com seu endereço MAC. Assim que recebe a resposta, o dispositivo atualiza a cache ARP com o endereço MAC do servidor, permitindo que os pacotes sejam enviados diretamente para o servidor na próxima vez.

2.2. b)

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

O valor hexadecimal do campo **Type** da trama *Ethernet* é **0x0806**.

Neste contexto, **0x0806** é associado ao protocolo ARP.

Quando um dispositivo de rede vê o valor **0x0806** no campo **Type** de uma trama *Ethernet*, ele percebe que o conteúdo da trama é um pacote ARP.

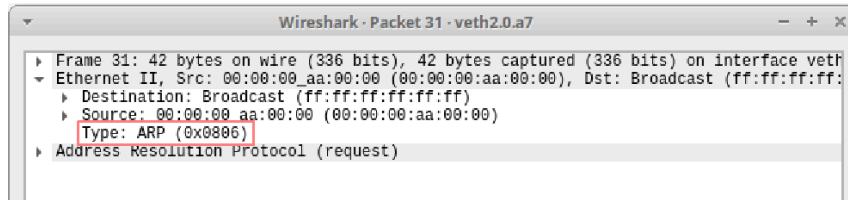


Figura 12: Type da trama *Ethernet*

2.2. c)

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

Existem duas formas distintas de saber que se trata efetivamente de um pedido ARP:

- Ao observar os campos da mensagem ARP, é possível ver que contém os endereços IP origem e destino e o endereço MAC origem. Como a origem ainda não sabe o endereço MAC do destino, devido à cache ARP estar limpa, a mensagem ARP é enviada para todos os dispositivos na rede, onde espera por uma resposta do dispositivo solicitado.
- Dentro da mensagem ARP, existe um campo chamado **opcode**, que indica se é uma solicitação ARP ou uma resposta ARP. No caso de um pedido ARP, o valor desse campo será **request (1)**.

2.3.

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

2.3. a)

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

O valor encontrado no campo ARP opcode foi 2, especificando que o mesmo se trata de uma **reply**.

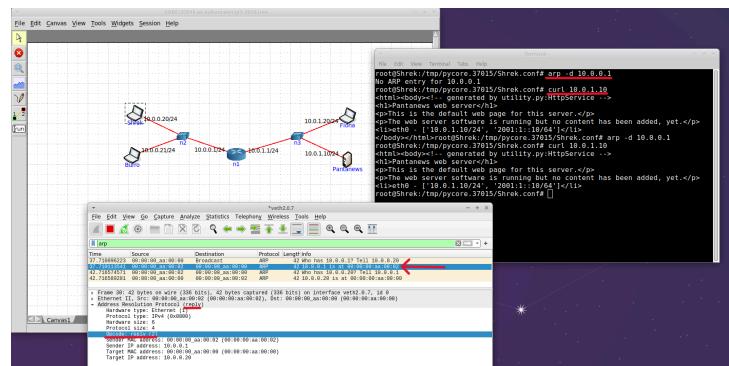


Figura 13: Campo ARP opcode

2.3. b)

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

Encontra-se na **Target MAC Address**. Este campo ocupa 6 bytes, especificamente localizados do 33º ao 38º bytes da trama.

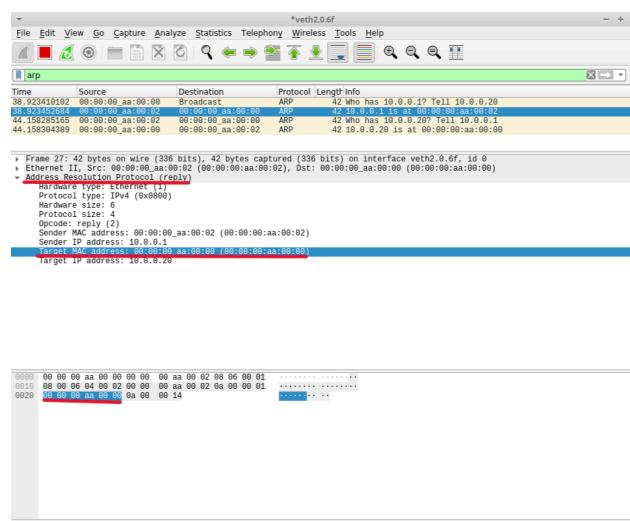


Figura 14: Campo Target MAC Address

2.3. 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.

Os endereços MAC de origem e de destino pertencentes aos sistemas que pretendemos descobrir, através dos comandos indicados, são os seguintes:

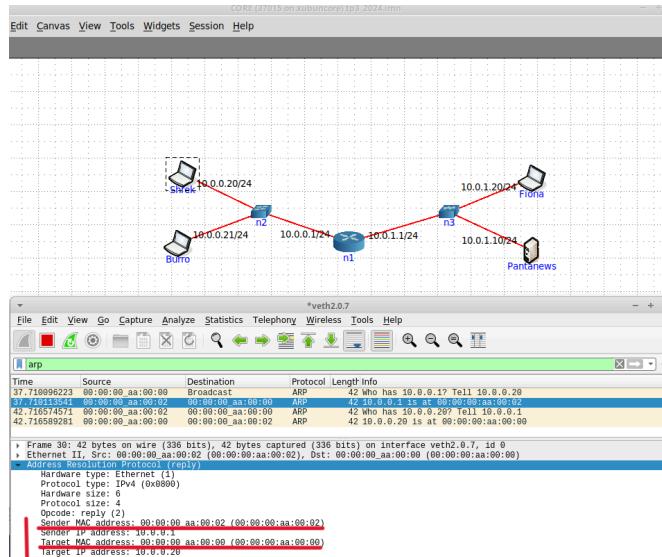


Figura 15: Endereços MAC de origem e de destino da trama

O *output* resultante da execução dos 3 comandos está presente nas seguintes imagens:

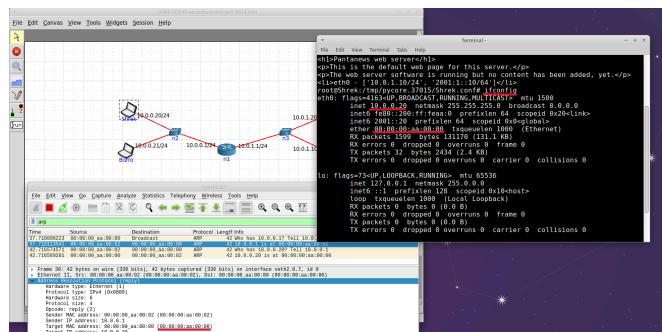


Figura 16: Execução do comando **ifconfig**

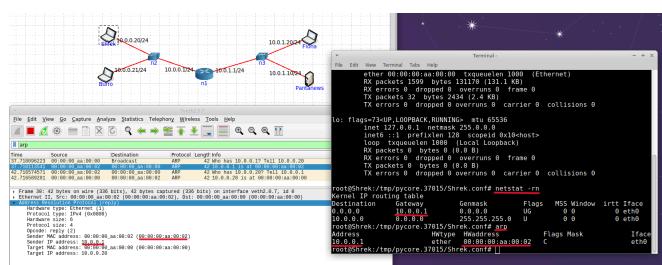


Figura 17: Execução dos comandos **netstat -rn** e **arp**

Assim, os resultados obtidos permitiram concluir que:

- O endereço MAC de origem pertence ao sistema **n1**;
- O endereço MAC de destino pertence ao sistema **Shrek**.

2.3. d)

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

O modo de comunicação poderá ser realizado de duas formas:

- *Unicast* (a mensagem é apenas enviada para um endereço, recebendo um único sistema a mesma);
- *Broadcast* (a mensagem é enviada para todos, recebendo todos os sistemas ao alcance do remetente a mesma).

Como observado nas seguintes imagens, recorre-se ao modo *broadcast* quando é realizado um *request*, de forma a alcançar uma maior probabilidade de resposta; já o modo *unicast* é utilizado quando é realizada uma *reply*, evitando enviar a mesma a quem não efetuou *request* (ou seja, o endereço escolhido é o do remetente do *request*).

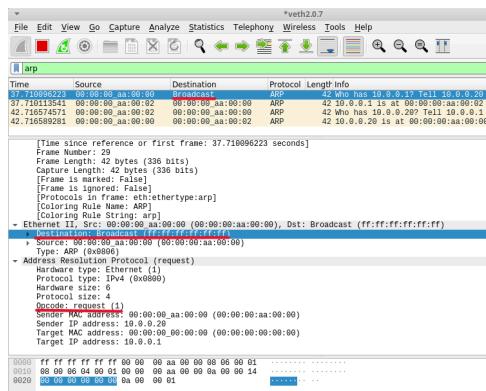


Figura 18: Broadcast - ARP Request

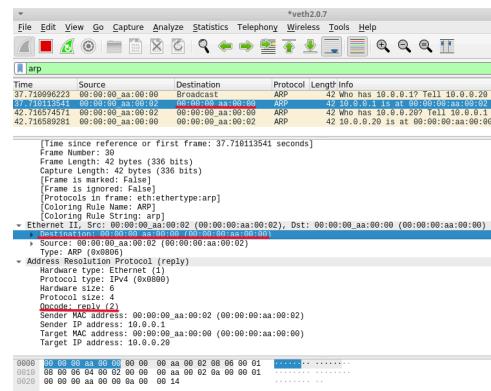


Figura 19: Unicast - ARP Reply

Então, no envio da resposta ARP (ARP *reply*), o modo de comunicação utilizado foi *unicast*, enviando-se apenas a *reply* a quem realizou o *request*.

2.4.

O **Burro** recebeu toda a informação trocada na interação anterior? Qual será a razão para tal?

O **Burro** não recebe toda a informação trocada na interação referida: apenas recebe o pedido do **Shrek** a questionar quem tem o IP **10.0.0.1**, não recebendo a resposta associada ao mesmo, pois não é o **Burro** que possui o IP mencionado. Assim, a comunicação realizada pelo **Shrek** passa a ser unicamente estabelecida entre ele mesmo e o *router n1*, já que este permite comunicar com o sistema identificado pelo IP pedido.



Figura 20: Captura na interface do PC da **Fiona**

2.5.

Repita a experiência com uma captura na interface do PC da **Fiona**. Documente as suas observações e conclusões com base no tráfego observado/capturado.

Foram realizados novamente os comandos `arp -d 10.0.0.1` e `curl 10.0.1.10` no **Shrek**.

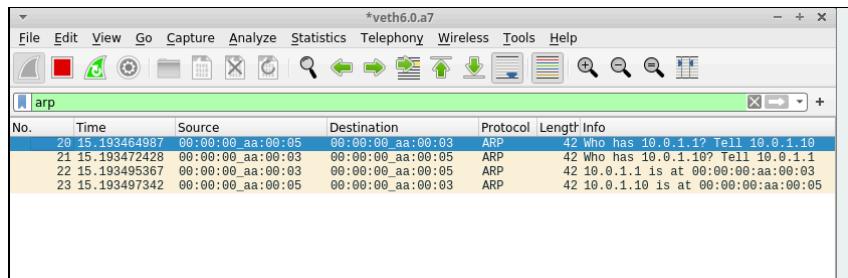


Figura 21: Captura na interface do PC da **Fiona**

Ao analisar a captura na interface do PC da **Fiona** no *Wireshark*, observa-se o seguinte:

1º A primeira mensagem registada é um ARP *request* do servidor **Pantanews** (**10.0.1.1**), este pergunta a todos os dispositivos na rede quem tem o endereço **10.0.1.10**.

2º O *router n1* (**10.0.1.10**) envia um ARP *request* para todos os dispositivos na rede onde pergunta quem tem o endereço IP **10.0.1.10**, que corresponde ao servidor **Pantanews**.

3º O *router n1* (**10.0.1.10**) então responde ao **Pantanews** que o seu endereço MAC (**00:00:00:aa:00:03**).

4º Por fim, o servidor **Pantanews** (10.0.1.1) responde ao *router n1* que o seu endereço MAC (00:00:00:aa:00:05).

Com base nestas observações, podemos concluir que houve uma tentativa de acesso ao servidor **Pantanews** a partir de um dispositivo na rede, porém, não é possível determinar qual dispositivo fez essa solicitação apenas com a captura da **Fiona**. Isto porque as mensagens ARP observadas indicam apenas a comunicação entre o servidor **Pantanews** e o *router n1*, sem revelar diretamente qual dispositivo solicitou o acesso ao servidor.

2.6.

Esboce um diagrama em que indique claramente, e de forma cronológica, todas as mensagens trocadas entre o **Shrek** e os sistemas com os quais comunica, até à recepção do primeiro pacote que contém dados HTTP. Assuma que todas as tabelas ARP se encontram inicialmente vazias.

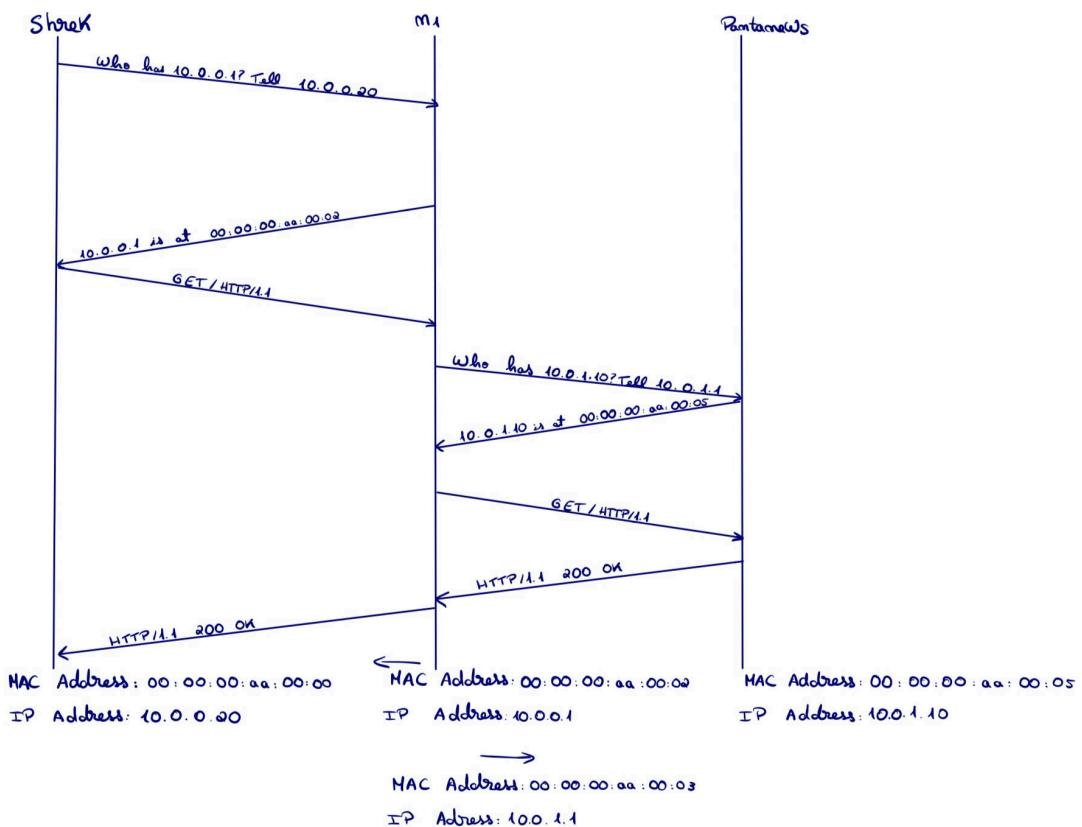


Figura 22: Captura na interface do PC da **Fiona**

Inicialmente, o **Shrek** envia um pedido ARP para descobrir o endereço MAC do *router n1* na rede onde se encontra. O *router n1* responde o seu endereço MAC.

De seguida, o **Shrek** envia um pedido HTTP para o *router n1* que tem como destino o **Pantanews**. O **n1** envia um pedido ARP para descobrir o endereço MAC do servidor **Pantanews**. O **Pantanews** responde o seu endereço MAC.

Tendo conhecimento do endereço MAC do **Pantanews**, o *router n1* envia uma solicitação HTTP para o servidor. O **Pantanews** responde com uma mensagem HTTP onde confirma o sucesso do pedido.

Por último, o **n1** passa a resposta do **Pantanews** para o **Shrek**, de modo que este receba a resposta HTTP do **Pantanews**.

2.7.

Construa manualmente a tabela de comutação do *switch* da casa do **Shrek**, atribuindo números de porta à sua escolha.

MAC	IF	TTL
00:00:00:aa:00:00 (Shrek)	1	30
00:00:00:aa:00:01 (Burro)	2	30
00:00:00:aa:00:02 (n1)	3	30

2^a Parte

A **Fiona** decide ir morar com o **Shrek** e o **Burro**, mas com a condição de deixarem de ter os cabos *Ethernet* espalhados pela casa. O **Shrek** decide então comprar equipamento *Wireless* e faz uma captura de tráfego para perceber melhor o seu funcionamento.

1. Acesso Rádio

Como pode ser observado, a sequência de *bytes* capturada inclui meta-information do nível físico (**radiotap header**, **radio information**) obtida do *firmware* da interface *Wi-Fi*, para além dos *bytes* correspondentes a tramas **802.11**. Selecione a trama de ordem XX correspondente ao seu identificador de grupo (TurnoGrupo, e.g., 11).

1.1.

Identifique em que frequência do espectro está a operar a rede sem fios, e o canal que corresponde a essa frequência.

A rede sem fios está a operar na frequência **2412 MHz**, correspondendo esta ao canal **1**.

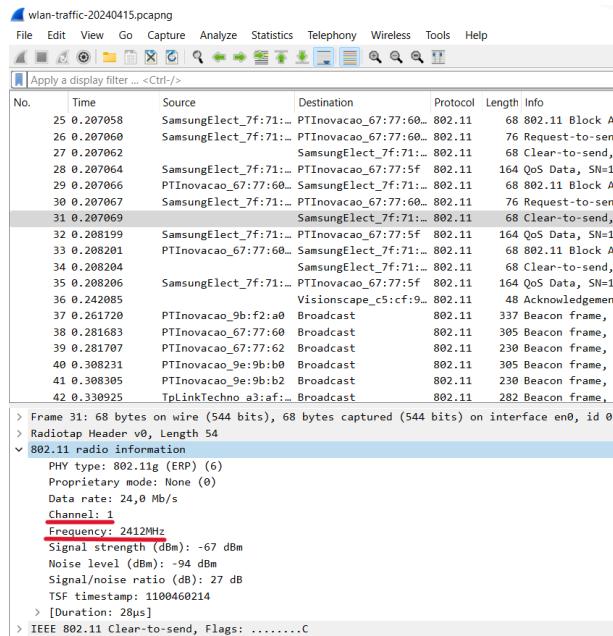


Figura 23: Canal e frequência correspondente onde opera a rede sem fios

1.2.

Identifique a versão da norma IEEE 802.11 que está a ser usada.

A versão que está a ser usada da norma IEEE 802.11 é a versão 802.11g, como podemos verificar pela imagem.

```
> Frame 31: 68 bytes on wire (544 bits), 68 bytes captured (544 bits) on interface en0, id 0
  Radiotap Header v0, Length 54
  - 802.11 radio information
    + [PHY type: 802.11g (ERP) (6)]
      Proprietary mode: None (0)
      Data rate: 24,0 Mb/s
      Channel: 1
      Frequency: 2412MHz
      Signal strength (dBm): -67 dBm
      Noise level (dBm): -94 dBm
      Signal/noise ratio (dB): 27 dB
      TSF timestamp: 1100460214
    + [Duration: 28µs]
    IEEE 802.11 frame-to-send, Flags: . .... c
      Type/Subtype: Clear-to-send (0x001c)
      Frame Control Field: 0xc409
      .0000 0000 0111 1000 Duration: 120 microseconds
      Receiver address: Samsung_E_7f:71:a7 (c0:d2:dd:7f:71:a7)
      Frame check sequence: 0xfab34ed0 [unverified]
      [FCS Status: Unverified]
```

Figura 24: Versão da norma IEEE 802.11

1.3.

Qual a taxa de transmissão a que foi enviada a trama escolhida? Será que essa taxa de transmissão corresponde à máxima que a interface *Wi-Fi* pode operar? Justifique.

A taxa de transmissão a que foi enviada a trama é **24 Mb/s**, sendo a taxa de transmissão máxima de operação da interface *Wi-Fi* exatamente o mesmo valor. Assim, conclui-se que a mesma funciona com débito máximo.

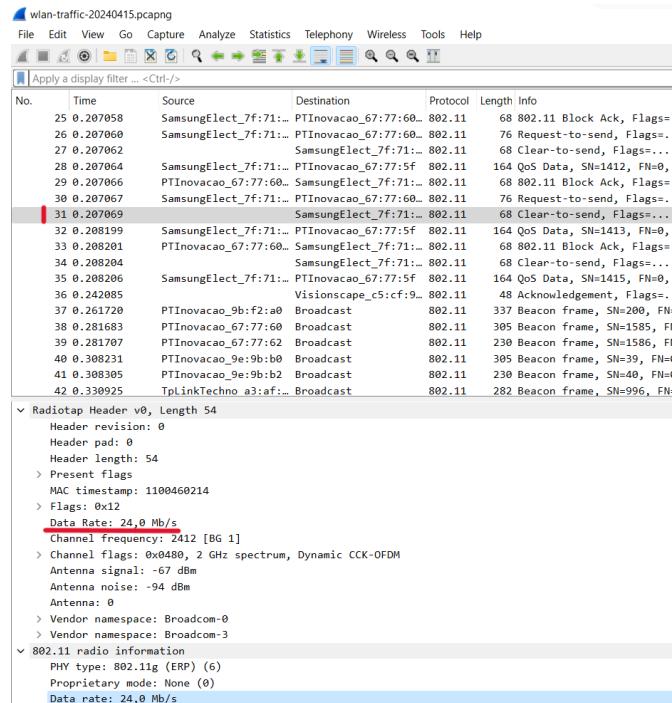


Figura 25: Análise das taxas de transmissão utilizada e máxima

2. Scanning Passivo e Scanning Ativo

Como referido, as tramas *beacon* permitem efetuar *scanning* passivo em redes IEEE 802.11 (*Wi-Fi*). Para a captura de tramas disponibilizada, e considerando XX o seu nº de TurnoGrupo (PLXX), responda às seguintes questões:

2.4.

Selecione uma trama *beacon* cuja ordem (ou terminação) corresponda a XX. Esta trama pertence a que tipo de tramas 802.11? Identifique o valor dos identificadores de tipo e de subtipo da trama. Em que parte concreta do cabeçalho da trama estão especificados (ver Anexo I)?

Utilizou-se o seguinte filtro *Wireshark* para obter a trama *beacon* com a ordem 31:

```
wlan.fc == 0x8000 && wlan.seq == 31
```

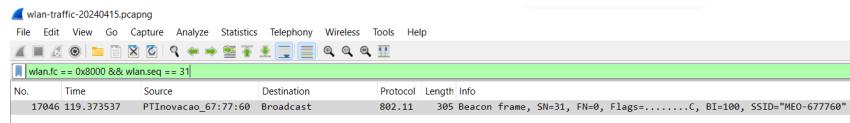


Figura 26: Trama *beacon* selecionada

Analisando a trama obtida, obteve-se as seguintes respostas às perguntas colocadas:

- Tipo de tramas 802.11: *Beacon Frame* (0x0008)
- Valor do identificador de tipo: 0 (*Management Frame*)
- Valor do identificador de subtipo: 8

Esta informação encontra-se do 2º bit ao 7º bit do cabeçalho da trama (inclusive), de acordo com o Anexo I.

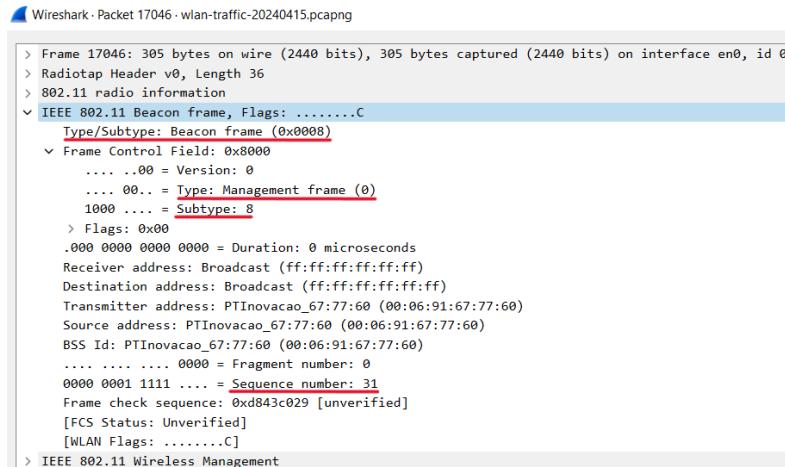


Figura 27: Campos observados para resposta às questões colocadas

2.5.

Verifique se está a ser usado o método de deteção de erros (CRC). Justifique. (Poderá ter de ativar a verificação no *Wireshark*, em **Edit -> Preferences -> Protocols -> IEEE -> “Validate Checksum if Possible”**)

Ao analisar a captura das tramas *Wi-Fi* e aplicar o filtro `wlan.fcs.status == bad`, podemos verificar que não há nenhuma trama com erros (CRC). No entanto, a presença do campo **Frame Check Sequence (fcs)**, temos a confirmação que o método de deteção de erros está a ser usado.

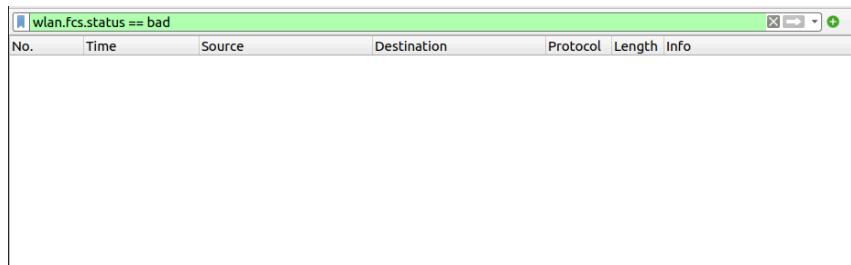


Figura 28: `wlan.fcs.status == bad`

Portanto, ao constatar que a maioria das tramas possui um *status* “bom”, usando o filtro `wlan.fcs.status == good`, podemos ter o reforço que o método de deteção de erros está a ser utilizado no momento da captura.

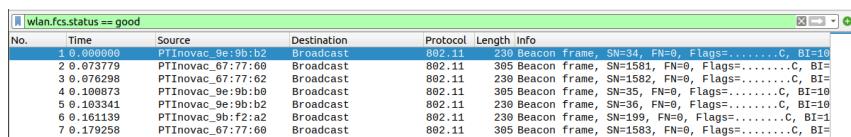


Figura 29: `wlan.fcs.status == good`

2.6.

Justifique o porquê de ser necessário usar deteção de erros em redes sem fios.

As redes sem fios são suscetíveis a erros devido ao ruído, interferência, desfasamento, entre outros fatores que pioram a qualidade dos sinais transmitidos. De forma a garantir que a transmissão de dados é confiável e eficiente, são implementadas várias técnicas de deteção e correção de erros no sistema de comunicação. Assim, é essencial a utilização de deteção de erros para garantir um bom funcionamento das redes sem fios, já que as mesmas são mais sensíveis nos fatores referidos que as redes com fios.

As tramas *beacon* são enviadas periodicamente e permitem especificar parâmetros de funcionamento para apoiar a operação e a gestão das ligações sem fios.

2.7.

Uma trama *beacon* anuncia o intervalo entre *beacons* às várias taxas de transmissão (B) que o AP suporta, assim como várias taxas de transmissão adicionais (*extended supported rates*). Indique qual a periodicidade e as taxas de transmissão suportadas pelo AP da trama *beacon* selecionada.

Na trama *beacon* selecionada, a periodicidade é 0.028271000 segundos, e as taxas de transmissão suportadas são ambas 1.0 Mb/s.

```

# Frame 17046: 305 bytes on wire (2440 bits), 305 bytes captured (2440 bits) on interface en0, id 0
  Section number: 1
  > Interface Id: 0 (eno0)
  > Encapsulation type: IEEE 802.11 plus radiotap radio header (23)
  > Arrival Time: Apr 15, 2024 15:16:37.564100000 GMT Daylight Time
  Epoch Arrival Time: 1713190597.564100000
  [Time shift for this packet: 0.000000000 seconds]
  [Time delta from previous captured frame: 0.028271000 seconds]
  [Time delta from previous displayed frame: 0.000000000 seconds]
  [Time since reference or first frame: 119.375370000 seconds]
  > RadioTap Version: 0, Length: 36
    Header revision: 0
    Header pad: 0
    Header length: 36
  > Present flags
  > Flags: 0x0000
    Channel frequency: 2412 [86 1]
  > Channel flags: 0x0480, 2 GHz spectrum, Dynamic CCK-OFDM
  Antenna signal: -64 dBm
  Antenna noise: -93 dBm
  Antenna: 0
  Radio Type: Wireless: Broadcast
  > IEEE 802.11 radio information
    PHY type: 802.11b (HR/DSSS) (4)
    Short preamble: False
    Data rate: 1.0 Mb/s
  Channel: 1
    Frequency: 2412MHz
    Signal strength (dBm): -64 dBm
    Noise level (dBm): -93 dBm
    Signal/noise ratio (dB): 29 dB
    TSF timestamp: 1219622337
  > Duration: 2344us
  > IEEE 802.11 Beacon frame, Flags: .....
  > IEEE 802.11 Wireless Management

```

Figura 30: Trama *beacon* selecionada com campos essenciais à resposta da questão colocada sublinhados

2.8.

Identifique e liste os SSIDs dos APs que estão a operar na vizinhança da STA de captura. Explicite o modo como obteve essa informação (por exemplo, se usou algum filtro para o efeito).

No *Wireshark*, podemos usar o filtro `wlan.ssid` e obtemos as seguintes tramas.

#	Time	Source	Destination	Protocol	Length	Info
1	8:00:00.000	PRTInnowe_80:01:32	Broadcast	802.11	230	Beacon frame, SN<1>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
2	8:07:29.000	PRTInnowe_80:01:32	Broadcast	802.11	230	Beacon frame, SN<102>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
3	8:07:29.000	PRTInnowe_80:01:32	Broadcast	802.11	230	Beacon frame, SN<102>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
4	8:07:29.000	PRTInnowe_80:01:32	Broadcast	802.11	230	Beacon frame, SN<102>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
5	8:07:29.000	PRTInnowe_80:01:32	Broadcast	802.11	230	Beacon frame, SN<102>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
6	8:16:13.000	PRTInnowe_80:01:32	Broadcast	802.11	230	Beacon frame, SN<102>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
7	8:16:13.000	PRTInnowe_80:01:32	Broadcast	802.11	230	Beacon frame, SN<102>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
21	8:20:57.777	PRTInnowe_80:01:32	Broadcast	802.11	395	Beacon frame, SN<102>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
22	8:20:57.777	PRTInnowe_80:01:32	Broadcast	802.11	395	Beacon frame, SN<102>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
37	8:26:17.200	PRTInnowe_80:01:32	Broadcast	802.11	337	Beacon frame, SN<204>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
38	8:26:17.200	PRTInnowe_80:01:32	Broadcast	802.11	337	Beacon frame, SN<204>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
39	8:26:17.200	PRTInnowe_80:01:32	Broadcast	802.11	337	Beacon frame, SN<204>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
39	8:26:17.200	PRTInnowe_80:01:32	Broadcast	802.11	337	Beacon frame, SN<204>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
39	8:26:17.200	PRTInnowe_80:01:32	Broadcast	802.11	337	Beacon frame, SN<204>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
41	8:30:39.000	PRTInnowe_80:01:32	Broadcast	802.11	230	Beacon frame, SN<40>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
42	8:30:39.000	PRTInnowe_80:01:32	Broadcast	802.11	230	Beacon frame, SN<40>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
43	8:36:48.877	PRTInnowe_80:01:32	Broadcast	802.11	337	Beacon frame, SN<202>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
44	8:36:48.877	PRTInnowe_80:01:32	Broadcast	802.11	337	Beacon frame, SN<202>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
45	8:36:50.747	PRTInnowe_80:01:32	Broadcast	802.11	395	Beacon frame, SN<105>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
46	8:36:50.747	PRTInnowe_80:01:32	Broadcast	802.11	395	Beacon frame, SN<105>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
49	8:41:05.050	PRTInnowe_80:01:32	Broadcast	802.11	395	Beacon frame, SN<41>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
50	8:41:05.050	PRTInnowe_80:01:32	Broadcast	802.11	395	Beacon frame, SN<41>, Fw<0>, Flags:....., C, B1>100, SSID:MEOW-WIFI
52	8:43:23.4	Tp-Link_A31-af9f-00	Broadcast	802.11	292	Beacon frame, SN<91>, Fw<0>, Flags:....., C, B1>100, SSID:TP-LINK_AP_AP#00A8
53	8:43:23.4	Tp-Link_A31-af9f-00	Broadcast	802.11	292	Beacon frame, SN<91>, Fw<0>, Flags:....., C, B1>100, SSID:TP-LINK_AP_AP#00A8
54	8:44:50.000	HuaweiE_52:87:00	Broadcast	802.11	398	Beacon frame, SN<382>, Fw<0>, Flags:....., C, B1>100, SSID:HuaweiE-52B77F

Figura 31: Filtro `wlan.ssid`

Onde podemos ver os diversos SSIDs dos APs que estão a operar na vizinhança da STA de captura, como, por exemplo, o MEO-WiFi e MEO-677760. No entanto, como obtivemos uma grande quantidade de tramas, não conseguimos saber com certeza quais são todos os SSIDs dos APs, então fizemos um filtro no terminal, de modo a obtermos todos os SSIDs únicos:

```
tshark -r wlan-traffic-20240415.pcapng -Y "wlan.ssid" -T fields -e wlan.ssid | sort | uniq, onde:
```

- tshark chama o *tshark*;
- -r wlan-traffic-20240415.pcapng indica o ficheiro que tem a captura que pretendemos analisar;
- -Y "wlan.ssid" aplica um filtro para selecionar as tramas que contenham informação sobre o SSID;
- -T fields especifica o formato de saída dos resultados que irão ser mostrados;
- -e wlan.ssid especifica que apenas queremos o campo SSID das tramas já filtradas;
- | sort | uniq ordena os campos e elimina as entradas duplicadas.

```
marianna@marianna-HP:Downloads$ tshark -r wlan-traffic-20240415.pcapng -Y "wlan.ss
id" -T fields -e wlan.ssid | sort | uniq
FlyingNet
GV_Casa
MEO-005DA0
MEO-1FA270
MEO-677760
MEO-828830
MEO-9BF2A0
MEO-9E9BB0
MEO-F17570
MEO-WiFi
NOS-1CA6
NOS-93F3
phi_F41927C3C600
shellyswitch25-C8C9A37A032A
Sky
TP-LINK_AP_AF08
Vodafone-528777
Vodafone-B56E07
```

Figura 32: Execução do filtro indicado no terminal

Onde obtivemos os seguintes SSIDs:

- FlyingNet
- GV Casa
- MEO-005DA0
- MEO-1FA270
- MEO-677760
- MEO-828830
- MEO-9BF2A0
- MEO-9E9BB0
- MEO-F17570
- MEO-WiFi
- NOS-1CA6
- NOS-93F3
- phi_F41927C3C600
- shellyswitch25-C8C9A37A032A
- Sky
- TP-LINK_AP_AF08
- Vodafone-528777
- Vodafone-B56E07

No trace disponibilizado foi também registrado *scanning* ativo (envolvendo tramas *probe request* e *probe response*), comum nas redes *Wi-Fi* como alternativa ao *scanning* passivo.

2.9.

Estabeleça um filtro *Wireshark* apropriado que lhe permita visualizar todas as tramas *probing request* e *probing response*, simultaneamente.

O filtro *Wireshark* estabelecido de forma a visualizar todas as tramas *probing request* e *probing response* foi o seguinte: `wlan.fc == 0x4000 || wlan.fc == 0x5000`.

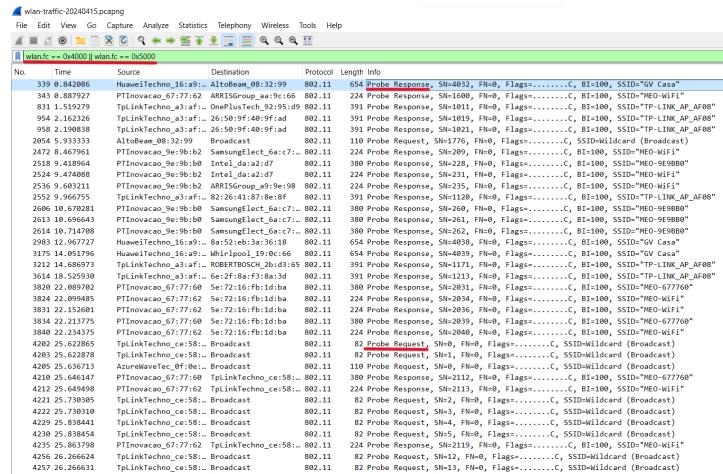


Figura 33: Filtro aplicado no *Wireshark*

Este filtro foi obtido através do seguinte raciocínio, faltando apenas a disjunção final (`||`) para apresentar os 2 tipos de trama simultaneamente:

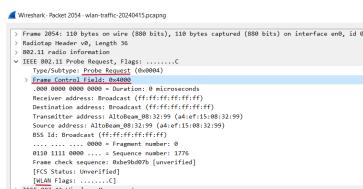


Figura 34: Campos de uma trama *probe request* utilizados na construção do filtro

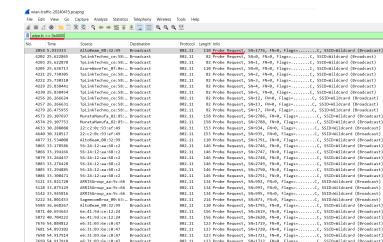


Figura 35: Filtro para obter apenas tramas *probe request*

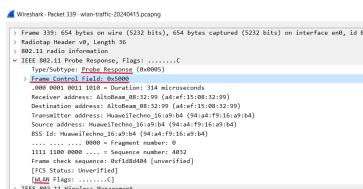


Figura 36: Campos de uma trama *probe response* utilizados na construção do filtro

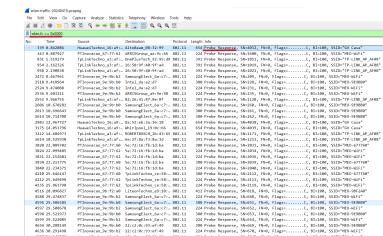


Figura 37: Filtro para obter apenas tramas *probe response*

2.10.

Assuma que a STA de captura consegue se associar a qualquer AP na vizinhança. Dadas as tramas recebidas através do *scanning* ativo e passivo, observe os valores da força do sinal (*Signal Strength*) nas meta-informações de nível físico e aponte qual AP a STA de captura deve se associar para obter a melhor qualidade de ligação possível. Indique como chegou a esta resposta.

Através da execução do seguinte comando, obtiveram-se todos os valores da força do sinal (*Signal Strength*), ordenados por ordem decrescente, encontrados nas tramas da captura *Wireshark* fornecida:

```
tshark -r wlan-traffic-20240415.pcapng -Y "wlan_radio.signal_dbm" -T fields -e wlan_radio.signal_dbm | sort | uniq
```

Quanto mais elevado for o valor da força do sinal, melhor será a qualidade de ligação possível. Assim, observando os resultados obtidos, concluiu-se que o melhor valor possível de encontrar seria **0 dBm**.

```
marielana@marielana-HP:Downloads$ tshark -r wlan-traffic-20240415.pcapng -Y "wlan_radio.signal_dbm" -T fields -e wlan_radio.signal_dbm | sort | uniq
0
-20
-21
-22
-23
-24
-25
-26
-27
-28
-29
-30
-31
-32
-33
-34
-35
-36
-37
-38
-40
-41
-42
-44
-45
-48
-50
-55
-59
-60
-61
-62
-63
-64
```

Figura 38: Valores da força do sinal presentes nas tramas da captura *Wireshark*

Obteve-se todas as tramas com **Signal Strength == 0** aplicando-se o seguinte filtro *Wireshark*:

```
wlan.fc == 0x8841 && wlan_radio.signal_dbm == 0
```

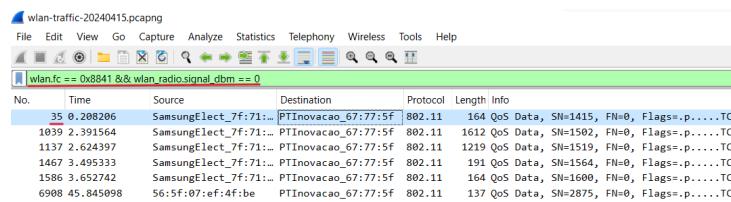


Figura 39: Tramas obtidas após aplicação do filtro *Wireshark* indicado

Depois, selecionou-se a primeira trama listada.

Inspecionando o seu conteúdo, concluiu-se que a STA de captura deveria associar-se ao seguinte AP (**00:06:91:67:77:60**) para alcançar a melhor qualidade de ligação:

- *Receiver Address: PTInovacao_67:77:60 (00:06:91:67:77:60)*

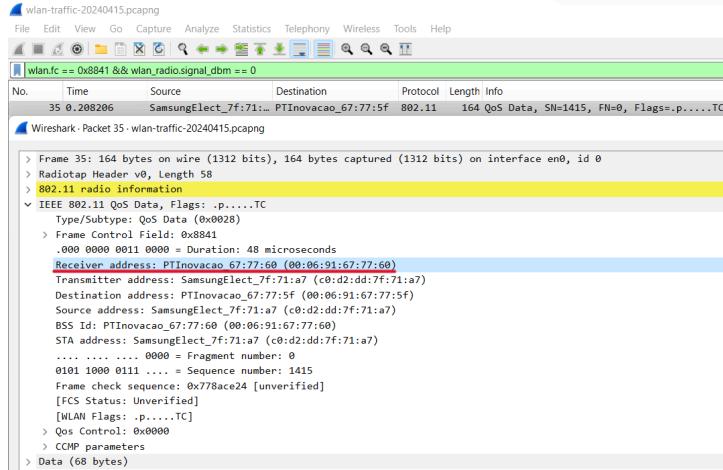


Figura 40: Melhor AP para associação

2.11.

Os valores de taxa de transmissão do *Wi-Fi* estão diretamente associados à qualidade da receção do sinal, utilizando-se dos valores de sensibilidade mínima (*Minimum Sensivity*) e taxa de transmissão (*Data Rate*) das tabelas referência do Anexo II, da força do sinal recebido nas tramas do AP indicado da resposta anterior, estime o débito que a STA obterá nessa ligação.

Para estimar o débito que a STA obterá nessa ligação, podemos utilizar os valores de sensibilidade mínima (*Minimum Sensivity*) e taxa de transmissão (*Data Rate*) da trama capturada acima (trama 35). Nesta trama, observamos que a sensibilidade mínima é de 0 dB (*Signal strength*). A taxa de transmissão é de 72.2222 Mb/s (*Data rate*). Os dois valores enquadram-se na modulação 64-QAM segundo o Anexo II.

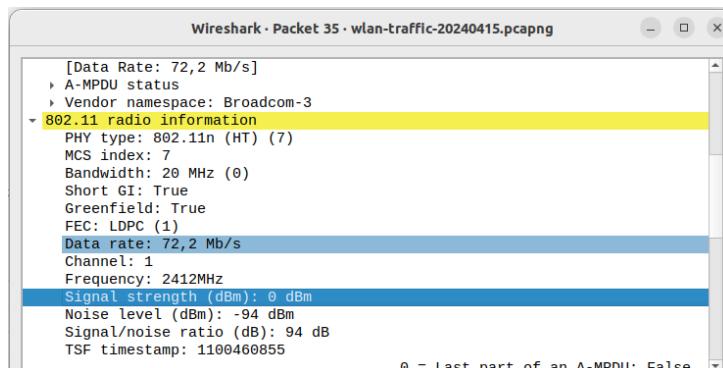


Figura 41: Trama 35

3. Processo de Associação

Numa rede *Wi-Fi* estruturada, um nodo ou STA deve associar-se a um ponto de acesso antes de enviar dados. O processo de associação nas redes IEEE 802.11 é executada enviando a trama *association request* da STA para o AP e a trama *association response* enviada pelo AP para a STA, em resposta ao pedido de associação recebido. Este processo é antecedido por uma fase de autenticação. Para a sequência de tramas capturada:

3.12.

Identifique uma sequência de tramas que corresponda a um processo de associação realizado com sucesso entre a STA e o AP, incluindo a fase de autenticação.

Filtro *Wireshark* utilizado:

```
wlan.fc == 0x0000 || wlan.fc == 0x1000 || wlan.fc == 0xb000
```

Sendo:

- 0xb000 correspondente à autenticação;
- 0x0000 correspondente ao *association request*;
- 0x1000 correspondente à *association response*.

wlan-traffic-20240415.pcapng						
No.	Time	Source	Destination	Protocol	Length	Info
<input type="checkbox"/> wlan.fc == 0x0000 wlan.fc == 0x1000 wlan.fc == 0xb000						
12855	98.374622	92:97:e1:69:c3:d5	PTInovacao_67:77:62	802.11	105	Authentication, SN=674, FM=0, Flags=.....C
12857	98.374728	PTInovacao_67:77:62	92:97:e1:69:c3:d5	802.11	81	Authentication, SN=3667, FM=0, Flags=.....C
12861	98.387225	92:97:e1:69:c3:d5	PTInovacao_67:77:62	802.11	213	Association Request, SN=675, FM=0, Flags=.....C, SSID="HEO-WiFi"
12863	98.387244	PTInovacao_67:77:62	92:97:e1:69:c3:d5	802.11	192	Association Response, SN=3670, FM=0, Flags=.....C

Figura 42: Sequência de tramas correspondente a uma associação entre STA e AP bem-sucedida

3.13.

Efetue um diagrama que ilustre a sequência de todas as tramas trocadas no processo.

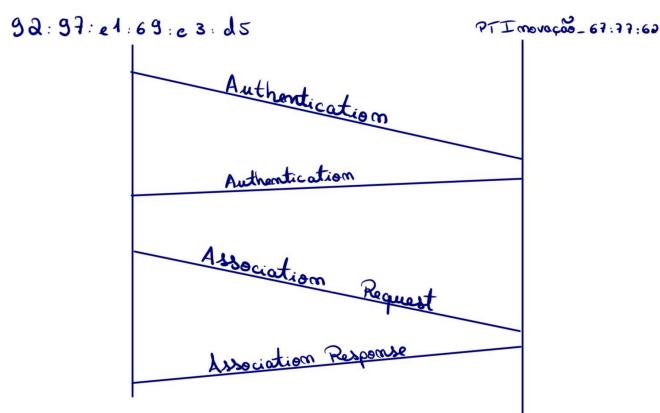


Figura 43: Sequência de tramas trocadas

4. Transferência de Dados

O *trace* disponibilizado, para além de tramas de gestão da ligação de dados, inclui tramas de dados e tramas de controlo da transferência desses mesmos dados.

4.14.

Estabeleça um filtro apropriado e selecione uma trama de dados (*Data ou QoS Data*), cujo número de ordem inclua o seu identificador de grupo (terminação XX, ou X caso não exista XX). Sabendo que o campo **Frame Control** contido no cabeçalho das tramas **802.11** permite especificar a direccionalidade das tramas, o que pode concluir face à direccionalidade dessa trama, será local à WLAN?

O filtro *Wireshark* utilizado foi o seguinte:

```
wlan.fc == 0x8841 && wlan.seq == 1431,
```

obtendo-se uma trama de dados, *QoS Data*, (0x8841) com a terminação igual a XX = 31 (1431).

A direccionalidade da trama selecionada (**Frame from STA to DS via an AP (To DS: 1 From DS: 0)** (0x1)) revela que a mesma não será local à WLAN (**To DS: 1**).

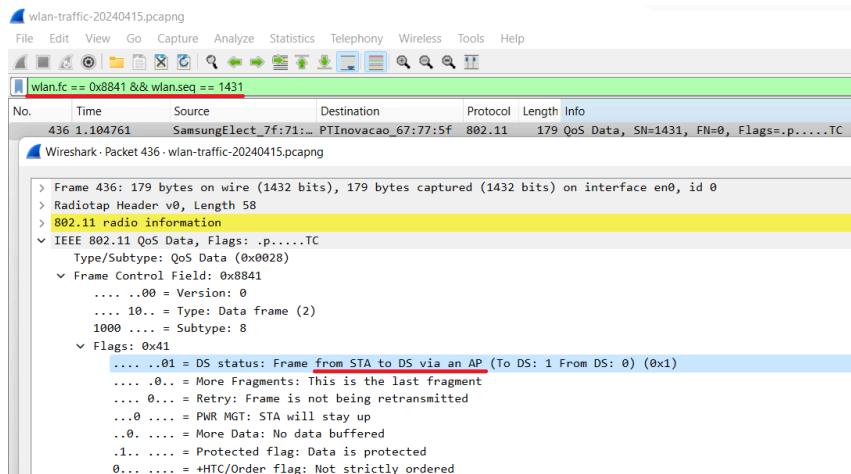


Figura 44: Direccionalidade da trama

4.15.

Para a trama de dados selecionada, transcreva os endereços MAC em uso, identificando quais os endereços correspondentes à estação sem fios (STA), ao AP e ao *router* de acesso ao sistema de distribuição (DS)?

Os endereços MAC em uso são os seguintes:

- **STA** (`SamsungElect_7f:71:a7` - *Transmitter Address*):

`c0:d2:dd:7f:71:a7`

- **AP** (`PTInovacao_67:77:60` - *Receiver Address*):

`00:06:91:67:77:60`

- **Router** (`PTInovacao_67:77:5f` - *Destination Address*):

`c0:d2:dd:7f:71:a7`

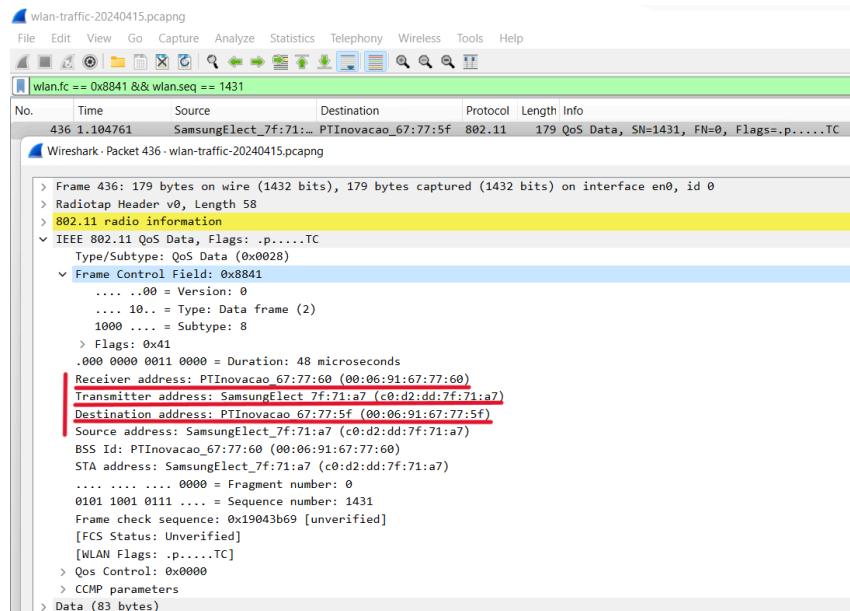


Figura 45: Endereços MAC em uso

4.16.

O uso de tramas *Request To Send* e *Clear To Send*, apesar de opcional, é comum para efetuar “pré-reserva” do acesso ao meio quando se pretende enviar tramas de dados, com o intuito de reduzir o número de colisões resultante maioritariamente de STAs escondidas. Para o envio de dados selecionado acima, verifique se está a ser usada a opção RTS/CTS na troca de dados entre a STA e o AP/*Router* da WLAN, identificando a direccionalidade das tramas e os sistemas envolvidos. Dê um exemplo de uma transferência de dados em que é usada a opção RTC/CTS e um outro em que não é usada.

No envio de dados selecionado acima, está a ser usada a opção RTS/CTS na troca de dados entre a STA e o AP/*Router* da WLAN, como podemos ver na imagem abaixo.

```
434 1.104751    SamsungElect_7f:71... PTInovacao_67:77:60... 802.11      76 Request-to-send, Flags=.....C
435 1.104758    SamsungElect_7f:71... 802.11      68 Clear-to-send, Flags=.....C
436 1.104761    SamsungElect_7f:71... PTInovacao_67:77:5f  802.11      179 QoS Data, SN=1431, FN=0, Flags=.p....TC
```

Figura 46: Transferência de dados onde é usada RTC/CTS

Nesta troca de dados onde é usada a opção RTS/CTS (*Request To Send/Clear To Send*), a direccionalidade das tramas seria a seguinte:

Trama *Request To Send* (RTS):

- From DS: 0 (Do STA)
- To DS: 0 (Para o AP)

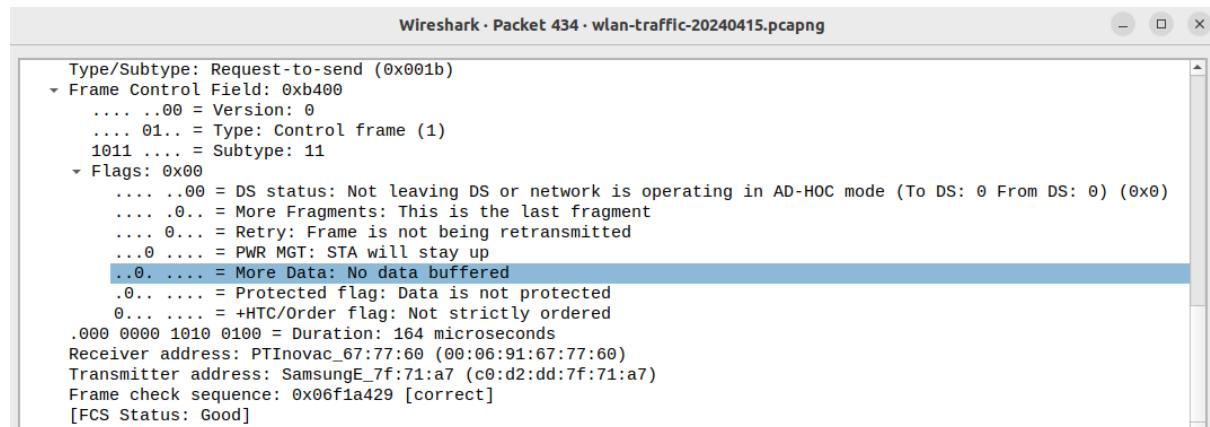


Figura 47: Direccionalidade da trama *Request To Send*, RTC

Trama *Clear To Send* (CTS):

- From DS: 0 (Do AP)
- To DS: 0 (Para o STA)

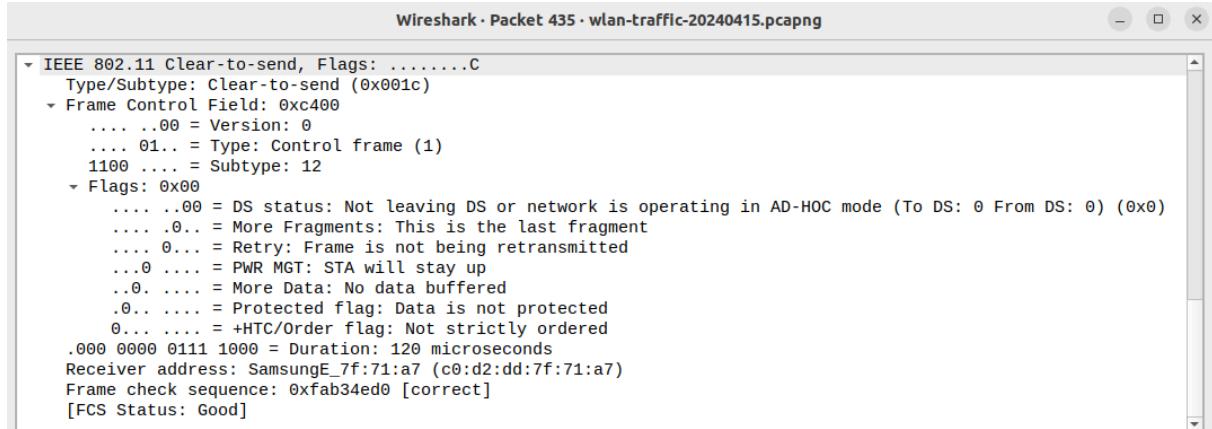


Figura 48: Direccionalidade da trama *Clear To Send*, CTS

Trama *QoS Data* (dados):

- From DS: 1 (Do STA)
- To DS: 0 (Para AP)

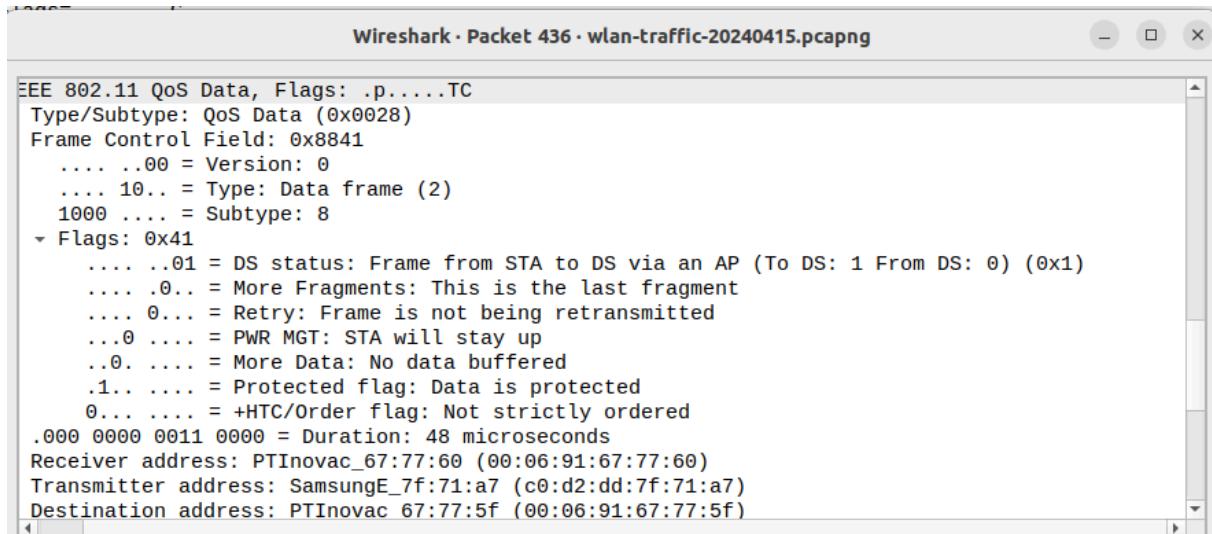


Figura 49: Direccionalidade da trama Clear To Send CTS

Na imagem seguinte, temos um exemplo onde a transferência de dados não usa o método RTC/CTS.

18138 137.099689	SamsungE_7c:79:cb ... 802.11	48 Acknowledgement, Flags=.....C
18139 137.099691	SamsungE_7c:79:cb ... 802.11	64 QoS Null function (No data), SN=409, FN=0, Flags=.....F.C

Figura 50: Transferência de dados onde não é usada RTC/CTS