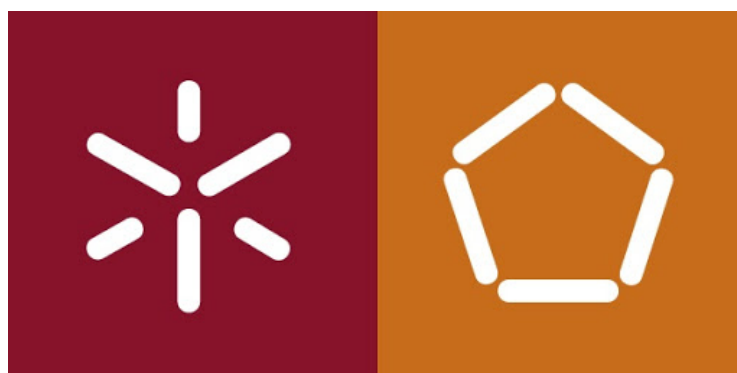


# Tarefa Prática 2

## Modelação e Caracterização de Tráfego

**PG39254 - Igor Araújo**  
**PG39255 - Matheus Gonçalves**  
**PG41017 - I-Ping**



Departamento de Informática  
Universidade do Minho  
Braga - Portugal  
13 de março de 2020

# Sumário

<b>Sumário</b> .....	2
Objetivo .....	3
Parte I - Captura e análise de tráfego .....	3
Parte II - Filtragem de tráfego .....	4
Conclusão .....	10

## Objetivo

Realizar a captura, visualização, análise e filtragem de tráfego de rede, onde no final desse relatório o grupo vai estar mais familiarizado com as ferramentas e os conceitos de captura e análise de tráfego.

## Parte I - Captura e análise de tráfego

- a) Inicie a captura de tráfego na interface de rede disponível. Faça uma primeira análise comparativa dos cabeçalhos e formatos dos PDUs do protocolos TCP, UDP e IP. Identifique para cada um deles os campos geralmente utilizados na classificação de tráfego:

O protocolo TCP possui header que contém diversos campos, mas os campos que são utilizados geralmente para identificação e classificação de um tráfego são as portas de origem e destino, e da mesma maneira para o UDP. Além destas, para a melhor classificação do tráfego é utilizado também os campos da PDU da camada de redes IP, que utilizam os endereços de origem e destino IP o número de protocolo, assim é formada a 5 tupla. Em posse desses parâmetros é possível em muitos casos classificar o tráfego. Porém a cada dia novas aplicações com diversos tipos de tráfegos são enviadas através de tráfegos encriptados o que torna ainda mais difícil sua identificação e classificação. Pode-se observar tais campos mencionados na figura 1.

```

  Internet Protocol Version 4, Src: 172.26.63.165, Dst: 193.137.16.65
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      Total Length: 87
      Identification: 0xe71e (59166)
    > Flags: 0x0000
      Fragment offset: 0
      Time to live: 128
      Protocol: UDP (17)
      Header checksum: 0x95ed [validation disabled]
      [Header checksum status: Unverified]
      Source: 172.26.63.165
      Destination: 193.137.16.65
  User Datagram Protocol, Src Port: 49941, Dst Port: 53
    Source Port: 49941
    Destination Port: 53
    Length: 67
    Checksum: 0x28e1 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 9]
    > [Timestamps]
  > Domain Name System (query)
```

Figura 1. Exemplificação de PDU.

**b) Utilizando o sniffer em modo de captura, proceda à invocação de várias aplicações conhecidas, nomeadamente:**

- Acesso via browser ao URL: `http://marco.uminho.pt`
- Acesso ftp (anonymous): `ftp.di.uminho.pt`
- Acesso em tftp para router-ext (193.136.9.33)
- Acesso via telnet para router-ext (193.136.9.33) ou para router-lab (192.168.90.254)
- Acesso ssh para qualquer host da sala de aula
- Resolução de nomes usando `nslookup www.uminho.pt`
- `traceroute cisco.uminho.pt`

**e construa uma tabela onde, para cada aplicação, conste o protocolo de transporte e a porta de atendimento do servidor (quando aplicável).**

Protocolo de Transporte	Porta de Origem	Porta de Destino
HTTP	6	87837
FTP	7	78
TFTP		69
TELNET	545	23
SSH	88	22
DNS	88	53
ICMP	88	53

**Tabela 1.** Tabela de aplicações

## Parte II - Filtragem de tráfego

**a) Explore e descreva:**

- i A utilidade dos filtros de captura e visualização;
- ii A sintaxe e semântica dos filtros.

**Dê alguns exemplos simples de utilização dos mesmos.**

Os filtros, tal como sugere, são opções que nos permitem seleccionar um conjunto de informação a ser apresentadas no visualizador. Esse filtro pode ser utilizado por protocolo, endereço de rede, por porta, por endereço MAC. Eles possuem uma forma bem simples de se utilizar, basta aplicarmos no campo de pesquisa o que queremos mostrar. No exemplo a seguir iremos filtrar pelo protocolo TELNET.

No.	Time	Source	Destination	Protocol	Length	Info
59	2.769576	193.136.9.33	193.136.9.171	TELNET	66	Telnet Data ...
61	2.774089	193.136.9.33	193.136.9.171	TELNET	471	Telnet Data ...
63	2.776083	193.136.9.33	193.136.9.171	TELNET	96	Telnet Data ...
64	2.778872	193.136.9.171	193.136.9.33	TELNET	57	Telnet Data ...
66	2.780613	193.136.9.171	193.136.9.33	TELNET	72	Telnet Data ...
81	2.980339	193.136.9.33	193.136.9.171	TELNET	68	Telnet Data ...
82	2.981111	193.136.9.171	193.136.9.33	TELNET	64	Telnet Data ...
93	3.620222	193.136.9.171	193.136.9.33	TELNET	55	Telnet Data ...
97	4.818585	193.136.9.171	193.136.9.33	TELNET	55	Telnet Data ...
103	4.218594	193.136.9.171	193.136.9.33	TELNET	55	Telnet Data ...
108	4.461908	193.136.9.171	193.136.9.33	TELNET	55	Telnet Data ...

**Figura 2.** Exemplo aplicação do filtro TELNET.

Como exemplo utilizamos o protocolo TELNET, e sua semântica de pesquisa, nesse caso, é bem simples, bastando apenas escrever o protocolo, mas caso queremos procurar o ip, a semântica permite adicionar outras informações, como o ip.dst, que seria o IP destino.

No.	Time	Source	Destination	Protocol	Length	Info
10	0.445856	194.25.7.28	193.136.9.171	TLSv1.2	93	Application Data
47	2.646627	52.209.135.164	193.136.9.171	TCP	60	443 → 55783 [ACK] Seq=1 Ack=425 Win=4628 Len=0
49	2.694961	52.209.135.164	193.136.9.171	TCP	60	443 → 55783 [ACK] Seq=1 Ack=1361 Win=48312 Len=0
50	2.697886	52.209.135.164	193.136.9.171	TLSv1.2	278	Application Data
53	2.758106	52.209.135.164	193.136.9.171	TCP	60	443 → 55782 [ACK] Seq=1 Ack=417 Win=138 Len=0
57	2.765517	193.136.9.33	193.136.9.171	TCP	60	25 → 55787 [SYN, ACK] Seq=8 Ack=1 Win=4128 Len=0 MSS=1460
59	2.769576	193.136.9.33	193.136.9.171	TELNET	66	Telnet Data ...
61	2.774089	193.136.9.33	193.136.9.171	TELNET	471	Telnet Data ...
63	2.776083	193.136.9.33	193.136.9.171	TELNET	96	Telnet Data ...
65	2.794674	52.209.135.164	193.136.9.171	TCP	60	443 → 55783 [ACK] Seq=225 Ack=1796 Win=50996 Len=0
67	2.797915	52.209.135.164	193.136.9.171	TCP	60	443 → 55782 [ACK] Seq=1 Ack=596 Win=143 Len=0

**Figura 3.** Exemplo aplicação do filtro IP.DST.

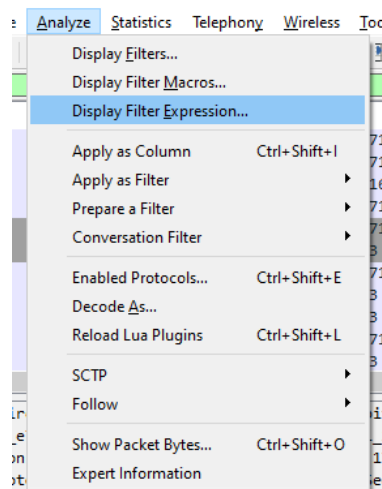
Também permite adicionar operadores lógicos, por exemplo o símbolo || que permite pesquisar dois protocolos ao mesmo tempo.

No.	Time	Source	Destination	Protocol	Length	Info
51	2.782142	193.136.9.171	52.209.135.164	TLSv1.2	489	Application Data
52	2.782208	193.136.9.171	52.209.135.164	TLSv1.2	478	Application Data
53	2.758106	52.209.135.164	193.136.9.171	TCP	60	443 → 55782 [ACK] Seq=1 Ack=417 Win=138 Len=0
54	2.758103	193.136.9.171	52.209.135.164	TLSv1.2	233	Application Data
56	2.765518	193.136.9.33	193.136.9.171	TCP	60	59780 → 23 [RST] Seq=8 Win=64248 Len=0 MSS=1460 Win=256 SACK_Flags=1
57	2.765517	193.136.9.33	193.136.9.171	TCP	60	25 → 55787 [SYN, ACK] Seq=8 Ack=1 Win=4128 Len=0 MSS=1460
58	2.765518	193.136.9.171	193.136.9.33	TCP	54	55787 → 23 [ACK] Seq=1 Ack=1 Win=64248 Len=0
59	2.769576	193.136.9.33	193.136.9.171	TELNET	66	Telnet Data ...
61	2.774089	193.136.9.33	193.136.9.171	TELNET	471	Telnet Data ...
62	2.774758	193.136.9.171	193.136.9.33	TCP	54	55787 → 23 [ACK] Seq=1 Ack=438 Win=63811 Len=0
63	2.776083	193.136.9.171	193.136.9.171	TELNET	96	Telnet Data ...

**Figura 4.** Exemplo aplicação do filtro com duplo parâmetros.

E por fim, o Wireshark possui uma opção no menu onde é possível consultar todos os filtros que existem para ser aplicado, e montar qual o filtro melhor se encaixa na pesquisa que deseja realizar.

- b) Baseando-se nas tramas capturadas acima (1.b), e em outros exemplos que achar conveniente, explore a utilidade e utilização dos filtros de captura e visualização, nomeadamente na captura/visualização de:



**Figura 5.** Aba para acesso ao menu avançado com todos os filtros.

- protocolos aplicativos;
- protocolos de transporte;
- endereços IP;
- pacotes com valores específicos nos campos principais dos cabeçalhos de transporte e rede (ver opção "+Expression");
- pacotes com flags de iniciação e término de conexões TCP;

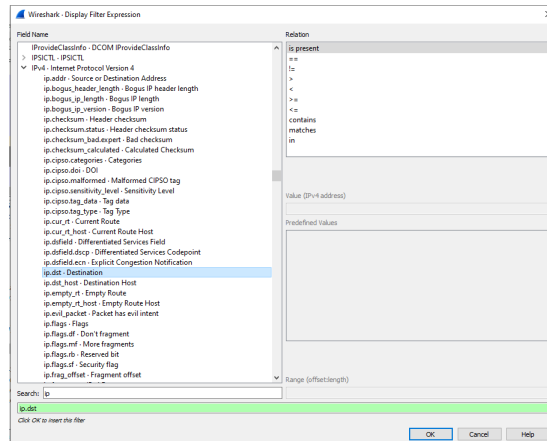
**Exemplifique a exploração que realizou, indicando a sintaxe utilizada nos filtros e, muito sucintamente os resultados obtidos.**

Aqui podemos mostrar a análise do arquivo gerado anteriormente. Em nosso exemplo vamos procurar a utilização de um protocolo para transferência de um ficheiro. Começamos por filtrar a captura, utilizando a sintaxe mais simples possível: TFTP. Podemos ver o resultado obtidos na imagem 7

Indo para a análise de um pacote, visualizando suas informações, podemos verificar que ele utiliza como protocolo de transporte o UDP (figura 8), em seguida podemos verificar os endereços MAC dos dispositivos e os Ips utilizados (origem/Destino – figura 9).

E por último, nesse exemplo utilizado, podemos conferir as portas da aplicação que são utilizadas e como se trata de um protocolo com mensagem junto, ele nós informa qual o código enviado pela origem (figura 10).

E claro, ao montar o stream desse filtro(figura 11), podemos acompanhar qual foi a solicitação e a mensagem retornada pelo destino, conforme a figura 12.



**Figura 6.** Display com todos os filtros.

No.	Time	Source	Destination	Protocol	Length	Info
25	1.217376	193.136.9.171	193.136.9.33	TFTP	61	Read Request, File: index.html, Transfer type: octet
45	2.289331	193.136.9.171	193.136.9.33	TFTP	61	Read Request, File: index.html, Transfer type: octet
98	4.289734	193.136.9.171	193.136.9.33	TFTP	61	Read Request, File: index.html, Transfer type: octet
193	8.210433	193.136.9.171	193.136.9.33	TFTP	61	Read Request, File: index.html, Transfer type: octet
373	16.217899	193.136.9.171	193.136.9.33	TFTP	61	Read Request, File: index.html, Transfer type: octet
496	24.217982	193.136.9.171	193.136.9.33	TFTP	61	Read Request, File: index.html, Transfer type: octet
598	32.224083	193.136.9.171	193.136.9.33	TFTP	61	Read Request, File: index.html, Transfer type: octet
777	48.230413	193.136.9.171	193.136.9.33	TFTP	61	Read Request, File: index.html, Transfer type: octet
977	48.237231	193.136.9.171	193.136.9.33	TFTP	65	Error Code, Code: Not defined, Message: timeout on receive

**Figura 7.** Aplicando o filtro para o protocolo TFTP.

- c) Para uma das aplicações que usam o protocolo TCP (e.g. Telnet router-ext), explore a opção "Analyse - Follow TCP Stream". Indique os filtros automaticamente aplicados por essa opção. Discuta eventuais fragilidades de segurança e confidencialidade dos dados.

Com a opção de filtragem via menu Analyse > Follow > TCP Stream, é possível selecionar um pacote entre vários capturados e reunir todos os pacotes que pertencem ao mesmo stream de dados. Neste caso foi feito inicialmente uma filtragem pelo protocolo Telnet, conforme visto na figura 13 abaixo:

Em seguida foi utilizado o menu Analyse > Follow > TCP Stream, mencionado anteriormente e com isso foi possível agrupar todos os pacotes pertencentes ao stream de pacotes pertencentes ao stream do pacote selecionado, inclusive aqueles que não são exclusivamente de protocolo Telnet, conforme visto a seguir na figura 14.

Como pode também ser visto na figura 14 o filtro que é gerado pelo menu executado basicamente é filtrar na captura pelo stream TCP de número 6 que sintaticamente possui a expressão (tcp.stream eq 6), o trecho tcp.stream

```
Frame 25: 61 bytes on wire (488 bits), 61 bytes captured (488 bits) on Interface 'DeviceVPP_185FE12A-2E78-491A-9290-2C1D283629AE', id 0
> Interface id: 0 ('DeviceVPP_185FE12A-2E78-491A-9290-2C1D283629AE')
Encapsulation type: Ethernet (1)
Arrival Time: Nov  6, 2020 16:25:06.783608000 (UTC Standard Time)
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 158512088.783608000 seconds
[Time delta from previous captured frame: 0.049632000 seconds]
[Time delta from previous displayed frame: 0.000000000 seconds]
[Time since reference or first frame: 1.217376000 seconds]
Frame Number: 25
Frame Length: 61 bytes (488 bits)
Capture Length: 61 bytes (488 bits)
[Frame is marked: False]
[Frame is ignored: False]
[Protocols in frame: ethertypeipudpipftp]
[Coloring Rule Name: UDP]
[Coloring Rule String: udp]
```

Figura 8. Protocolo de transporte UDP.

```
[Coloring Rule String: udp]
Ethernet II, Src: Dell_2b:dd:e5 (f0:1f:af:2b:dd:e5), Dst: Cisco_e7:cc:20 (00:e0:f7:e7:cc:20)
  Destination: Cisco_e7:cc:20 (00:e0:f7:e7:cc:20)
  Source: Dell_2b:dd:e5 (f0:1f:af:2b:dd:e5)
  Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 193.136.9.171, Dst: 193.136.9.33
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 47
  Identification: 0x0341 (833)
  > Flags: 0x0000
  Fragment offset: 0
  Time to live: 128
  Protocol: UDP (17)
  Header checksum: 0x0000 [validation disabled]
  [Header checksum status: Unverified]
  Source: 193.136.9.171
  Destination: 193.136.9.33
  [Source GeoIP: Braga, PT, ASN 1930, Fundacao para a Ciencia e a Tecnologia, I.P.]
  [Destination GeoIP: Braga, PT, ASN 1930, Fundacao para a Ciencia e a Tecnologia, I.P.]
```

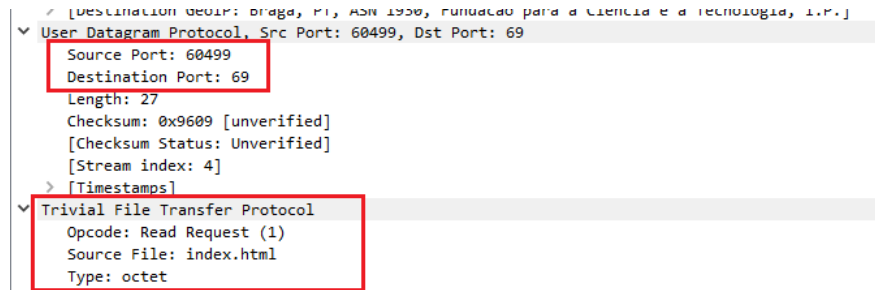
Figura 9. Exibindo o MAC e IP dos hosts origem e destinos.

indica intuitivamente que quer se filtrar por streams TCP e a parte (eq 6) indica que o stream específico que se deseja é o de número igual (eq) a 6. E o mais interessante do resultado da ação executado pelo menu selecionado é a reconstrução e apresentação das trocas de mensagens trocadas entre origem e destino, de tal forma que seja possível capturar e entender uma troca de mensagens por completo, se forem enviados em texto claro, que é o caso do protocolo Telnet. Tal resultado é visualizado na figura 15

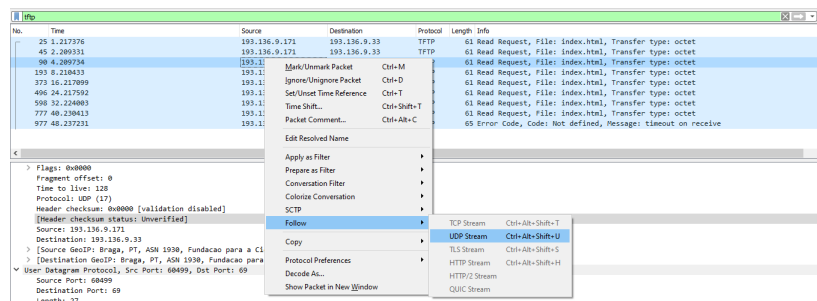
Os trechos apresentados marcados em azul foram recebidos pelo destinatário e em vermelho pela a origem, que está a tentar aceder ao equipamento via protocolo Telnet. Assim vemos de forma clara o password que foi digitado pelo utilizador. Desta forma mostra a fragilidade do protocolo Telnet, bem como outros protocolos que transmitem suas mensagens via texto claro, caso sejam transmitidos conteúdos sensíveis como senhas, informações bancárias e outros, tais dados estarão expostos e a comprometer a confidencialidade das informações caso haja um utilizador malicioso a sniffar os pacotes que são transmitidos pela rede.

d) Analise e identifique dados estatísticos da sua captura de pacotes.





**Figura 10.** Exibindo a porta de origem e destino para identificar a aplicação.



**Figura 11.** Menu para seguir o stream.

Dentre as capturas realizadas selecionou-se a referente ainda ao Telnet. Na tela principal já é possível verificar a quantidade de pacotes capturados no total e quantos estão sendo exibidos, quando há um filtro aplicado.

CRIAR TABELA AQUI:

Quantidade de pacotes capturados: 352

Total de pacotes exibidos: 50(14.2%)

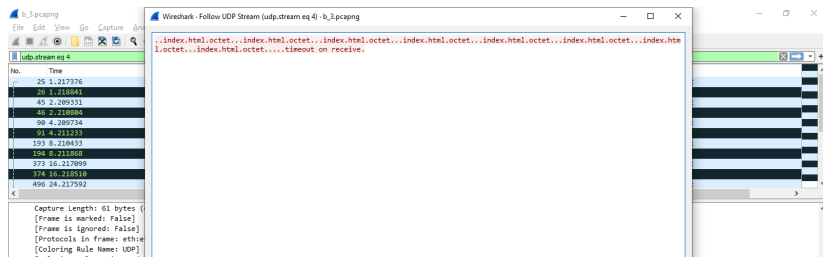
Outra opção para se obter mais estatísticas é utilizar o menu Statistics, nele há uma lista de opções. Uma delas que é interessante é o Conversation, nesta são compiladas todas as conversas entre origem X e destino Y para os protocolos Ethernet, Ipv4, Ipv6, TCP e UDP, sendo essas opções distribuídas em abas, conforme visto abaixo na figura 16.

Outra estatística interessante é listagem hierárquica dos protocolos, nela pode-se ver a representatividade de cada protocolo e subprotocolo no total da captura. Essa estatística pode ser visualizada na figura 17.

E outra forma de visualizar estatísticas é na opção File Properties do menu Statistics, que pode ser visualizado na figura 18

Quantidade de pacotes capturados: 352

Total de pacotes exibidos: 50(14.2%)



**Figura 12.** Ao montar o stream, é apresentado as informações.

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
Telnet						
No.	Time	Source	Destination	Protocol	Length	Info
59	2.769576	193.136.9.33	193.136.9.171	TELNET	66	Telnet Data ...
61	2.774603	193.136.9.33	193.136.9.171	TELNET	471	Telnet Data ...
63	2.776003	193.136.9.33	193.136.9.171	TELNET	96	Telnet Data ...

**Figura 13.** Exemplo lista de stream.

## Conclusão

Ao longo desse trabalho, tivemos a oportunidade de desenvolver nossas habilidades na análise do tráfego gerado. Com fácil aprendizado, a ferramenta Wireshark se mostra extremamente poderosa e eficaz, nos mostrando os detalhes das capturas geradas, com isso conseguimos caracterizar o tráfego, identificar as portas utilizadas e com isso qual o serviço utilizado, por exemplo. No relatório mostramos como podemos verificar tais informações, filtra os dados da captura, mostrando sua sintaxe e complexidade na geração dos filtro. Claro, estamos longe de sermos especialistas na caracterização e análise, mas podemos afirmar que estamos caminhando na direção correta.

No.	Time	Source	Destination	Protocol	Length	Info
56	2.763150	193.136.9.171	193.136.9.33	TCP	66	55707 → 23 [SYN] Seq=
57	2.765317	193.136.9.33	193.136.9.171	TCP	60	23 → 55707 [SYN, ACK]
58	2.765510	193.136.9.171	193.136.9.33	TCP	54	55707 → 23 [ACK] Seq=
59	2.769576	193.136.9.33	193.136.9.171	TELNET	66	Telnet Data ...
61	2.774603	193.136.9.33	193.136.9.171	TELNET	471	Telnet Data ...
62	2.774758	193.136.9.171	193.136.9.33	TCP	54	55707 → 23 [ACK] Seq=
63	2.776003	193.136.9.33	193.136.9.171	TELNET	96	Telnet Data ...
64	2.778872	193.136.9.171	193.136.9.33	TELNET	57	Telnet Data ...

Figura 14. Exemplo lista de stream.

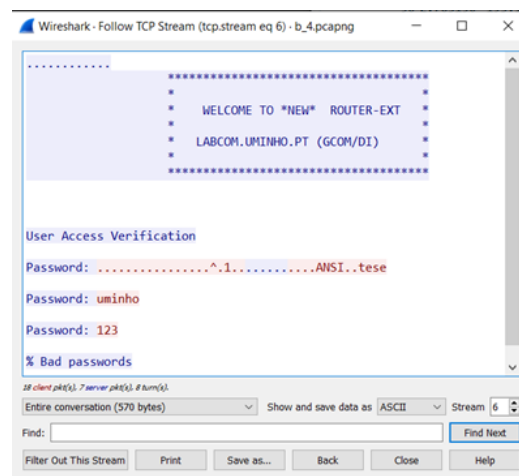


Figura 15. Montando o stream.

Wireshark - Conversations - b\_4.pcapng

Ethernet · 20IPv4 · 23IPv6 · 5TCP · 14UDP · 8

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A		
10.0.0.70	239.255.255.250	10	4830	10	4830	0	0	0.542857	16.0179	2412	0		
13.231.243.251	193.136.9.165	1	98	1	98	0	0	6.764173	0.0000	—	—		
13.231.243.251	193.136.9.163	1	98	1	98	0	0	6.768843	0.0000	—	—		
18.176.59.1	193.136.9.165	1	98	1	98	0	0	0.450132	0.0000	—	—		
18.176.59.1	193.136.9.163	1	98	1	98	0	0	0.450133	0.0000	—	—		
18.234.244.172	193.136.9.163	1	98	1	98	0	0	0.5458797	0.0000	—	—		
40.122.78.225	193.136.9.171	1	60	1	60	0	0	2.950045	0.0000	—	—		
40.122.78.225	193.136.9.159	1	60	1	60	0	0	0.15035282	0.0000	—	—		
49.50.100.160	193.136.9.171	1	60	1	60	0	0	0.11824447	0.0000	—	—		
52.108.56.8	193.136.9.171	24	14 k	15	6536	9	7518	4.624107	10.9331	4782	5501		
52.114.77.164	193.136.9.171	1	60	1	60	0	0	0.10436370	0.0000	—	—		
52.209.135.164	193.136.9.171	23	5359	12	1751	11	3608	2.554625	12.2626	1142	2353		
54.175.15.59	193.136.9.163	1	98	1	98	0	0	0.12023928	0.0000	—	—		
81.161.59.89	193.136.9.171	2	114	1	60	1	54	7.293388	0.0001	—	—		
87.9.171.164	193.136.9.159	1	60	1	60	0	0	0.0457774	0.0000	—	—		
92.119.160.52	193.136.9.161	1	60	1	60	0	0	0.248704	0.0000	—	—		
104.26.7.28	193.136.9.171	3	240	1	93	2	147	0.439493	0.0519	14 k	22 k		
122.228.19.80	193.136.9.162	1	60	1	60	0	0	0.4509092	0.0000	—	—		
185.156.73.60	193.136.9.187	1	60	1	60	0	0	0.2486212	0.0000	—	—		
193.136.9.33	193.136.9.171	50	3384	24	1918	26	1466	2.763150	7.5180	2040	1559		
193.136.9.154	239.255.255.250	4	1581	4	1581	0	0	0.13353360	0.3007	42 k	0		
193.136.9.155	239.255.255.250	4	1581	4	1581	0	0	0.8396456	0.3008	42 k	0		
193.136.9.207	224.0.0.251	6	519	6	519	0	0	0.1899045	10.5781	392	0		

☐ Name resolution☐ Limit to display filter☐ Absolute start time

Conversation Types ▾

Copy ▾Follow Stream...Graph...CloseHelp

Figura 16. Tabela conversation.

Wireshark - Protocol Hierarchy Statistics - b\_4.pcapng

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
▼ Frame	100.0	352	100.0	46003	21 k	0	0	0
▼ Ethernet	100.0	352	10.7	4928	2319	0	0	0
> Logical-Link Control	2.3	8	0.7	304	143	0	0	0
> Internet Protocol Version 6	4.3	15	1.3	600	282	0	0	0
▼ Internet Protocol Version 4	39.8	140	6.1	2800	1317	0	0	0
▼ User Datagram Protocol	6.8	24	0.4	192	90	0	0	0
Simple Service Discovery Protocol	5.1	18	15.7	7236	3406	18	7236	3406
Multicast Domain Name System	1.7	6	0.6	267	125	6	267	125
▼ Transmission Control Protocol	31.3	110	42.7	19645	9247	62	5644	2656
Transport Layer Security	6.5	23	36.6	16851	7931	23	16851	7931
Telnet	7.1	25	1.2	570	268	25	570	268
Internet Control Message Protocol	1.7	6	0.8	384	180	6	384	180
> Configuration Test Protocol (loopback)	0.3	1	0.1	46	21	0	0	0
Address Resolution Protocol	53.4	188	18.8	8648	4070	188	8648	4070

No display filter.
 Close
Copy ▾
Help

Figura 17. Tabela de estatística hierárquica.

Details

**File**

Name: C:\Users\igorv\Universidade do Minho\Matheus dos Santos Gonçalves - MERSTEL\Semestre-02\Modelação e Caracterização de Tráfego\TP2\Capturas\b\_4.pcapng

Length: 57 kB

Hash (SHA256): 23e2ff7863171696e5d92d0c40a13c910ffbc517cda753a95d65a2d8ff826d16

Hash (RIPEMD160): c556162077b351d1cede7a95a028be543d684ac1

Hash (SHA1): b7433c11b38ea42c8b9aa4ad3cd4a41aa7560d6f

Format: Wireshark/... - pcapng

Encapsulation: Ethernet

**Time**

First packet: 2020-03-06 18:26:34

Last packet: 2020-03-06 18:26:51

Elapsed: 00:00:16

**Capture**

Hardware: Intel(R) Core(TM) i5-3340M CPU @ 2.70GHz (with SSE4.2)

OS: 64-bit Windows 10 (1909), build 18363

Application: Dumpcap (Wireshark) 3.2.2 (v3.2.2-0-ga3efece3d640)

**Interfaces**

Interface	Dropped packets	Capture filter	Link type	Packet size limit
Ethernet	0 (0.0%)	none	Ethernet	262144 bytes

**Statistics**

Measurement	Captured	Displayed	Marked
Packets	352	352 (100.0%)	—
Time span, s	16.996	16.996	—
Average pps	20.7	20.7	—
Average packet size, B	131	131	—
Bytes	46003	46003 (100.0%)	0
Average bytes/s	2706	2706	—
Average bits/s	21 k	21 k	—

Figura 18. File Properties.