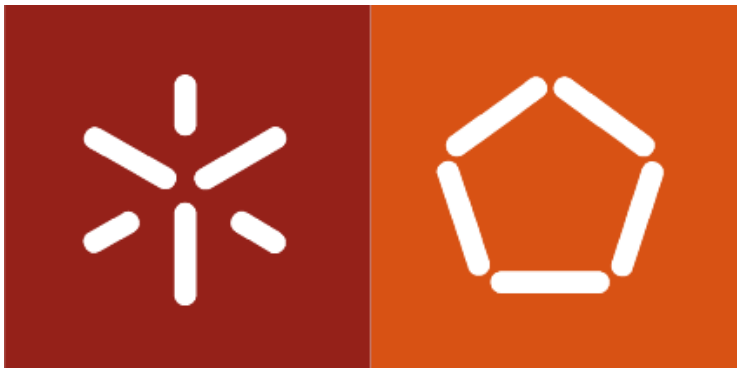


Universidade do Minho

Comunicação por Computador

TP1 – Protocolos da Camada de Transporte

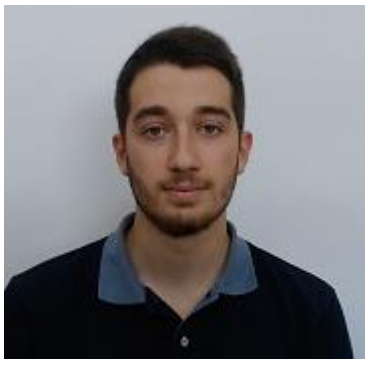
MIEI-3º Ano- 2º semestre

Grupo 3

PL1

Catarina Gil
A85266

Margarida Campos
A85166



Filipe Oliveira
A80330

Questão 1

Inclua no relatório uma tabela em que identifique, para cada comando executado, qual o protocolo de aplicação, o protocolo de transporte, porta de atendimento e overhead de transporte, como ilustrado no exemplo seguinte:

Resposta:

Comando usado (aplicação)	Protocolo de Aplicação (se aplicável)	Protocolo de transporte (se aplicável)	Porta de atendimento (se aplicável)	Overhead de transporte em bytes (se aplicável)
Ping	--------	---------	--------	--------
traceroute	--------	UDP	33434-33450	8 (33.3%)
telnet	telnet	TCP	23	20 (50%)
ftp	ftp	TCP	21	20 (40.8%)
Tftp	Tftp	UDP	69	8 (27.8%)
Browser/http	http	TCP	80	20 (50%)
nslookup	DNS	UDP	53	8 (33.9%)
ssh	ssh	TCP	22	20 (50%)

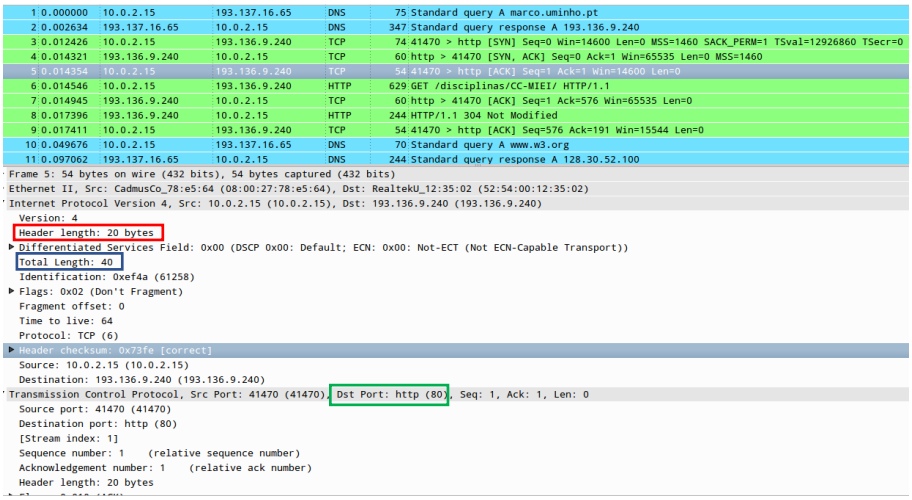


Figura 1- Captura do Wireshark usando o comando HTTP

A figura 1 mostra o protocolo de aplicação http. Podemos verificar que o protocolo de transporte usado é o TCP. A porta de atendimento é a 80 (analisado na figura pelo quadrado verde). Dado que o *Header Length* (quadrado vermelho) é 20 bytes e o *Total Length* é 40 (quadrado azul) , a percentagem de overhead é 20/40 = 0.5.

Nas restantes aplicações foi aplicado o mesmo raciocínio sendo posteriormente apresentadas as respetivas capturas do wireshark.

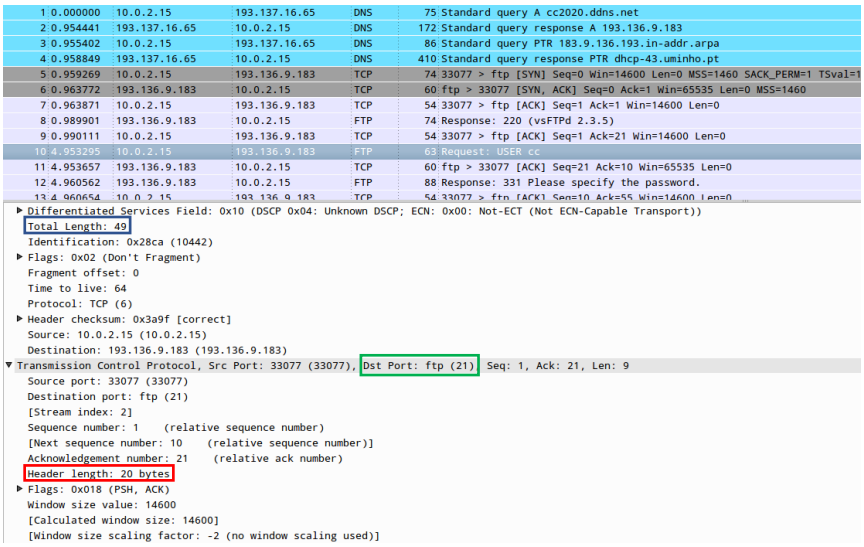


Figura 2- Captura no Wireshark utilizando o comando FTP

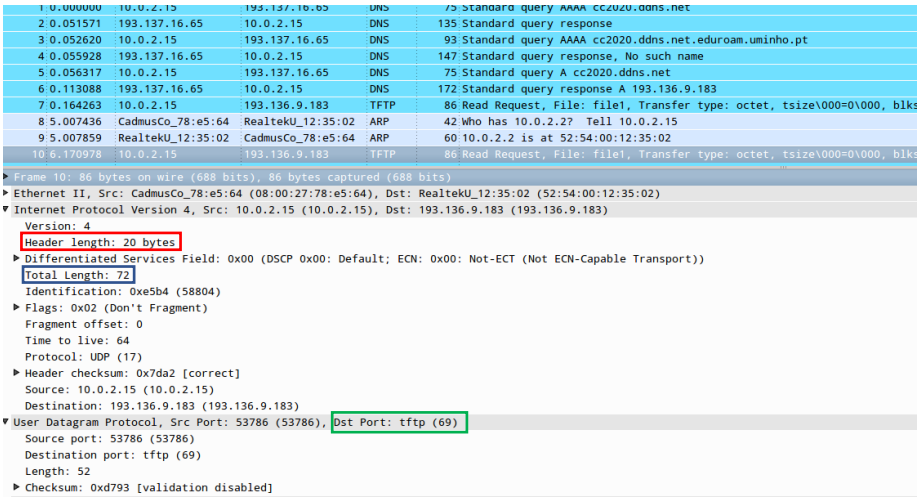


Figura 3- Captura Wireshark utilizando o comando TFTP

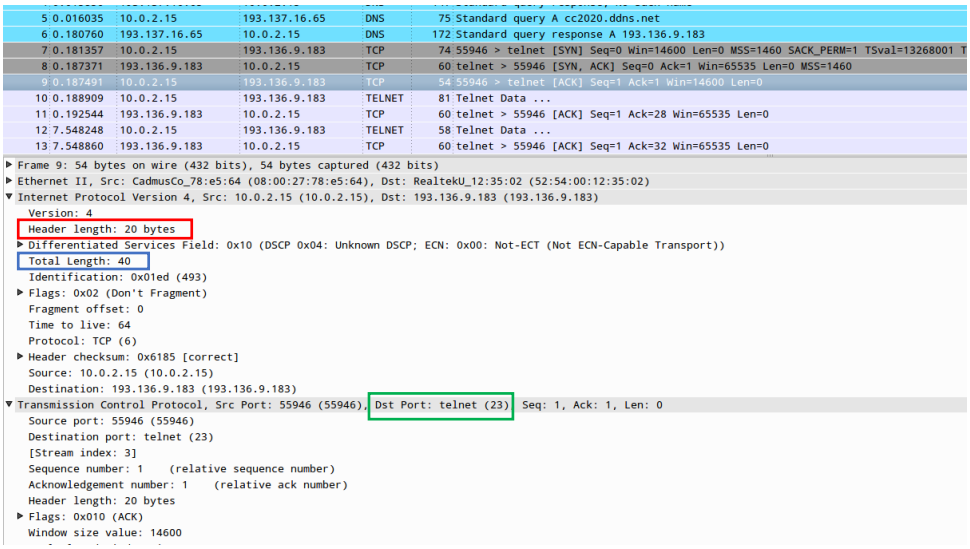


Figura 4- Captura Wireshark utilizando o comando TELNET

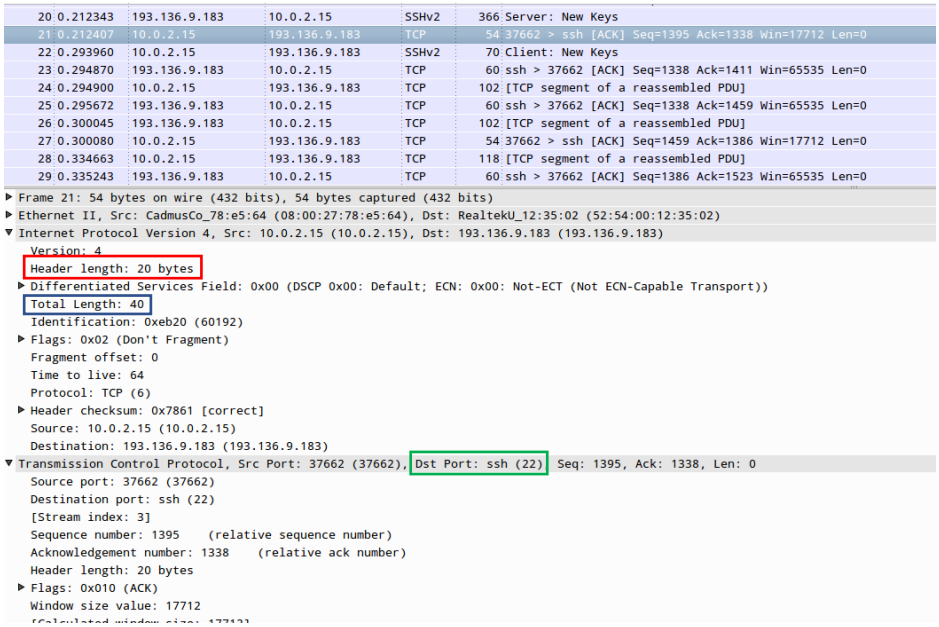


Figura 5- Captura Wireshark utilizando o comando SSH

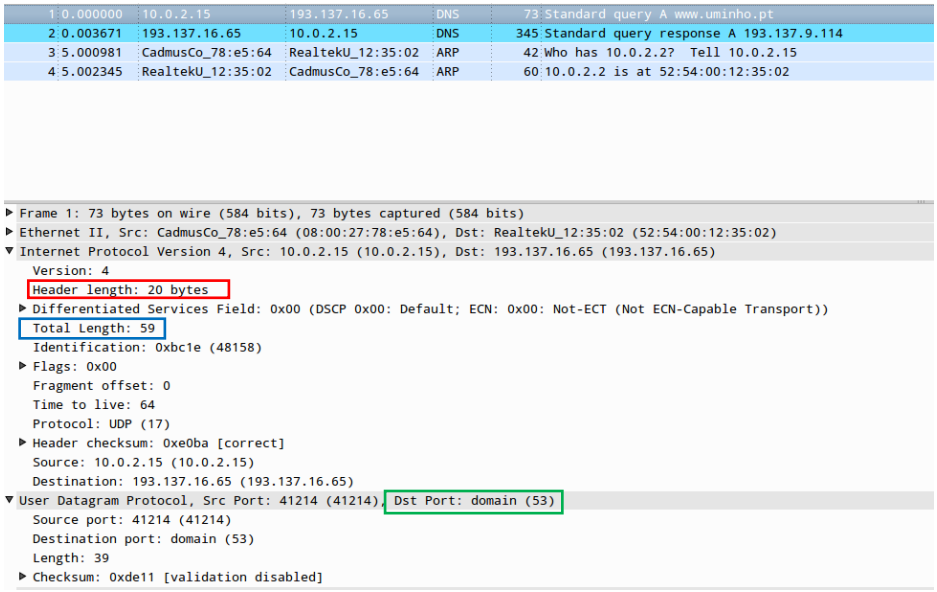


Figura 6- Captura Wireshark utilizando o comando nslookup

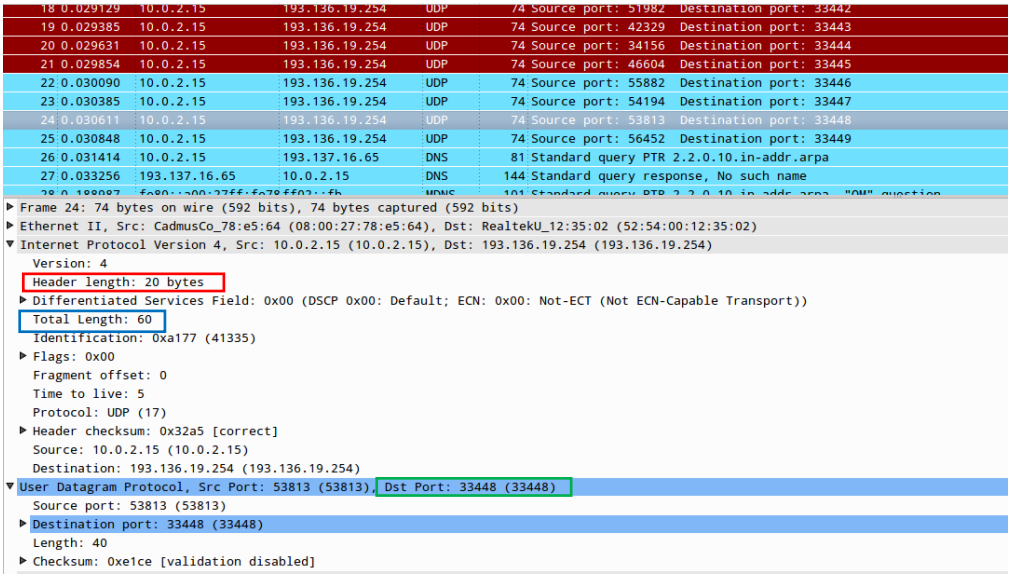


Figura 7- Captura wireshark utilizando o comando traceroute

Questão 2

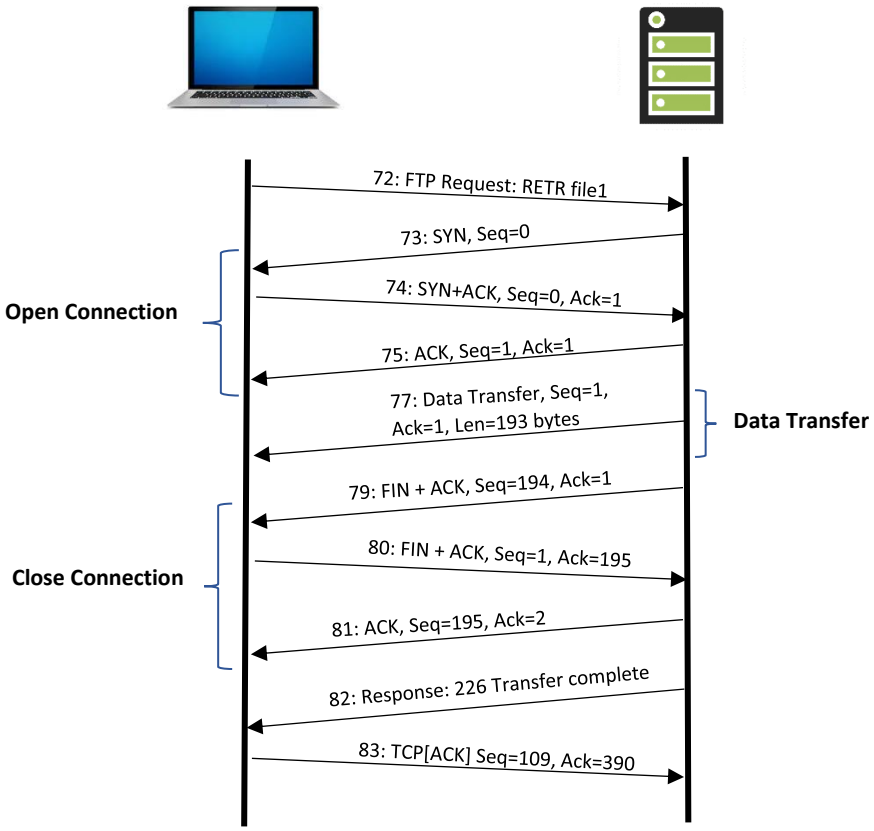
Uma representação num diagrama temporal das transferências da file1 por FTP e TFTP respetivamente. Se for caso disso, identifique as fases de estabelecimento de conexão, transferência de dados e fim de conexão. Identifica também claramente os tipos de segmentos trocados e os números de sequência usados quer nos dados como nas confirmações.

Resposta:

FTP

No.	Time	Source	Destination	Protocol	Length	Info
69	128.084783	10.3.3.1	10.1.1.1	FTP	97	Response: 200 Switching to Binary mode.
70	128.084823	10.1.1.1	10.3.3.1	FTP	89	Request: PORT 10,1,1,1,218,225
71	128.085146	10.3.3.1	10.1.1.1	FTP	117	Response: 200 PORT command successful. Consider using PASV.
72	128.085174	10.1.1.1	10.3.3.1	FTP	78	Request: RETR file1
73	128.085731	10.3.3.1	10.1.1.1	TCP	74	ftp-data > 56033 [SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=11596533 TSecr=0 WS=
74	128.085731	10.1.1.1	10.3.3.1	TCP	74	56033 > ftp-data [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460 SACK_PERM=1 TSval=11596533
75	128.086020	10.3.3.1	10.1.1.1	TCP	66	ftp-data > 56033 [ACK] Seq=1 Ack=1 Win=14608 Len=0 TSval=11596533 TSecr=11596533
76	128.086073	10.3.3.1	10.1.1.1	FTP	130	Response: 150 Opening BINARY mode data connection for file1 (193 bytes).
77	128.086081	10.3.3.1	10.1.1.1	FTP-DAT	259	FTP Data: 193 bytes
78	128.086091	10.1.1.1	10.3.3.1	TCP	66	56033 > ftp-data [ACK] Seq=1 Ack=194 Win=15552 Len=0 TSval=11596534 TSecr=11596533
79	128.086113	10.3.3.1	10.1.1.1	TCP	66	ftp-data > 56033 [FIN, ACK] Seq=194 Ack=1 Win=14608 Len=0 TSval=11596533 TSecr=11596533
80	128.086300	10.1.1.1	10.3.3.1	TCP	66	56033 > ftp-data [FIN, ACK] Seq=1 Ack=195 Win=15552 Len=0 TSval=11596534 TSecr=11596533
81	128.086871	10.3.3.1	10.1.1.1	TCP	66	ftp-data > 56033 [ACK] Seq=195 Ack=2 Win=14608 Len=0 TSval=11596534 TSecr=11596534
82	128.086878	10.3.3.1	10.1.1.1	FTP	90	Response: 226 Transfer complete.
83	128.086891	10.1.1.1	10.3.3.1	TCP	66	59256 > ftp [ACK] Seq=108 Ack=390 Win=14608 Len=0 TSval=11596534 TSecr=11596533
84	130.007670	10.1.1.254	224.0.0.5	OSPF	78	Hello Packet
85	130.378628	fe80::200:ff:feaa:1	ff02::5	OSPF	90	Hello Packet
86	132.264207	10.1.1.1	10.3.3.1	FTP	72	Request: QUIT
87	132.266283	10.3.3.1	10.1.1.1	FTP	80	Response: 221 Goodbye.

Figura 8- Captura de Tráfego de transferência de file1 por FTP no portátil1

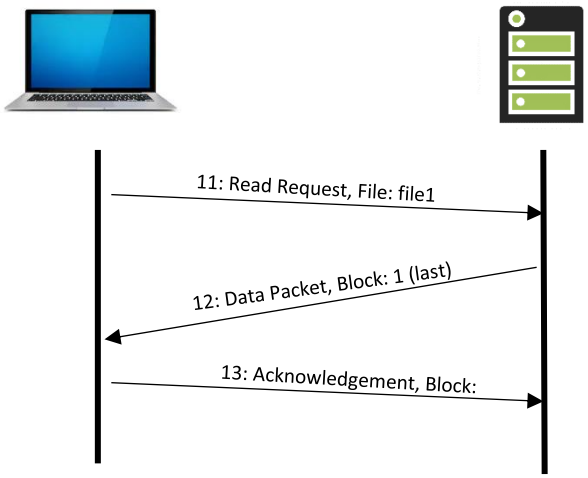


A partir da análise de tráfego (figura 8) e do diagrama temporal desenvolvido a partir dessa captura podemos verificar que: após feito o pedido (FTP request), o servidor estabelece a conexão enviando um pacote SYN com o número de sequência 0. O utilizador responde com um SYN+ACK. Posteriormente, o servidor envia um ACK de volta ao utilizador. Depois de todos os dados pretendidos serem enviados, é iniciada a fase de encerramento de conexão. A flag FIN, presente nesta fase, indica que se pretende terminar a conexão e fica à espera de uma resposta ACK.

TFTP

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.1.1.254	224.0.0.5	OSPF	78	Hello Packet
2	0.684537	fe80::200:ff:feaa:1	ff02::5	OSPF	90	Hello Packet
3	10.000148	10.1.1.254	224.0.0.5	OSPF	78	Hello Packet
4	10.716638	fe80::200:ff:feaa:1	ff02::5	OSPF	90	Hello Packet
5	20.001066	10.1.1.254	224.0.0.5	OSPF	78	Hello Packet
6	20.726775	fe80::200:ff:feaa:1	ff02::5	OSPF	90	Hello Packet
7	30.001166	10.1.1.254	224.0.0.5	OSPF	78	Hello Packet
8	30.732838	fe80::200:ff:feaa:1	ff02::5	OSPF	90	Hello Packet
9	32.365763	00:00:00_aa:00:1b	Broadcast	ARP	42	Who has 10.1.1.254? Tell 10.1.1.1
10	32.365917	00:00:00_aa:00:11	00:00:00_aa:00:1b	ARP	42	10.1.1.254 is at 00:00:00:aa:00:11
11	32.365917	10.1.1.1	10.3.3.1	TFTP	56	Read Request, File: file1, Transfer type: octet
12	32.366453	10.3.3.1	10.1.1.1	TFTP	239	Data Packet, Block: 1 (last)
13	32.366491	10.1.1.1	10.3.3.1	TFTP	46	Acknowledgement, Block: 1
14	37.373021	00:00:00_aa:00:11	00:00:00_aa:00:1b	ARP	42	Who has 10.1.1.1? Tell 10.1.1.254
15	37.373051	00:00:00_aa:00:1b	00:00:00_aa:00:11	ARP	42	10.1.1.1 is at 00:00:00:aa:00:1b
16	40.001312	10.1.1.254	224.0.0.5	OSPF	78	Hello Packet
17	40.737332	fe80::200:ff:feaa:1	ff02::5	OSPF	90	Hello Packet

Figura 9- Captura de Tráfego de transferência de file1 por TFTP no portátil1



Com base na captura de tráfego (figura 9) e no correspondente diagrama temporal podemos verificar que o utilizador envia um Read Request para o servidor contendo variadas informações entre as quais o nome do ficheiro. O servidor responde com um pacote de dados. Quando o utilizador recebe os dados, envia um pacote ACK para o servidor.

Questão 3

Com base nas experiências realizadas, distinga e compare sucintamente as quatro aplicações de transferência de ficheiros que usou nos seguintes pontos (i) uso da camada de transporte; (ii) eficiência na transferência; (iii) complexidade; (iv) segurança;

Resposta:

	SFTP	**FTP**	**TFTP**	**HTTP**
Uso da camada de transporte	TCP	TCP	UDP	TCP
Eficiência na transferência	O facto de ser fiável tem um impacto negativo na eficiência.	É fiável pelo que terá um maior overhead.	Possui um overhead mais baixo pois não se encarrega pela entrega dos dados.	Possui grande eficiência.
Complexidade	É bastante complexo uma vez que possui múltiplas funcionalidades.	Contribui para a segurança no processo de transferência de dados. É, portanto, complexo.	Como o protocolo é UDP, existem menos funcionalidades disponíveis. Como tal, a complexidade é reduzida.	Complexidade reduzida.
Segurança	É seguro uma vez que aplica encriptação e autenticação dos dados.	Recorre à autenticação, mas a segurança é reduzida.	Não possui técnicas de autenticação nem de encriptação. Tem por isso um nível de segurança bastante reduzido.	Recorre à autenticação, mas a segurança é reduzida.

Questão 4

As características das ligações de rede têm uma enorme influência nos níveis de Transporte e de Aplicação. Discuta, relacionando a resposta com as experiências realizadas, as influências das situações de perda ou duplicação de pacotes IP no desempenho global de Aplicações fiáveis (se possível, relacionando com alguns dos mecanismos de transporte envolvidos).

Resposta:

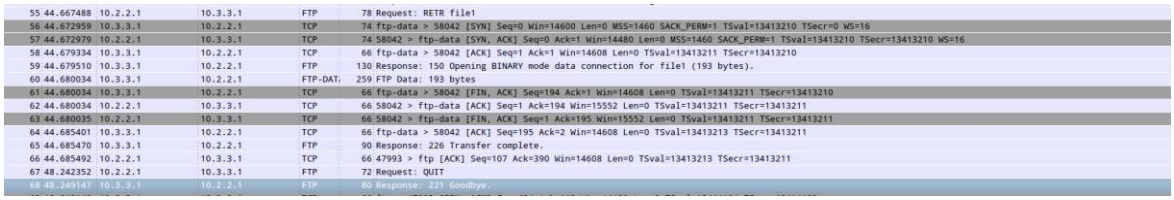


Figura 10- Captura de Tráfego de transferência da file1 por FTP em alpha

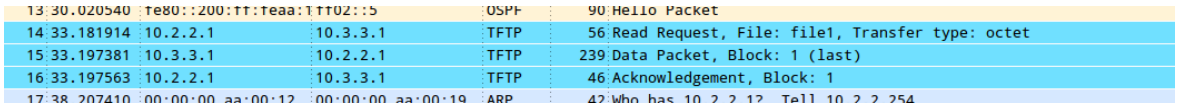


Figura 11- Captura de Tráfego de transferência da file1 por TFTP em alpha

Como já visto anteriormente, a aplicação FTP utiliza o protocolo de transporte TCP e a aplicação TFTP utiliza o protocolo de transporte UDP. Nas figuras apresentadas verificamos que estes dois protocolos tratam a ação de transferência de ficheiros de formas distintas.

O UDP não possui mecanismos de deteção e recuperação de perdas de pacotes, pelo que se forem perdidos, os protocolos que correm em cima do UDP poderão identificar e corrigir estas folhas. Para tal poderá ser atribuído um número de identificação ao pacote de modo a verificar a sua receção.

Por contrapartida, o TPC possui tais mecanismos de deteção e recuperação, assim caso haja uma perda de pacotes estes serão retransmitidos. O TCP garante que todos os pacotes são entregues.

Conclusão

Com a elaboração deste trabalho foi possível observar e estudar os diferentes protocolos de aplicação e os respetivos protocolos de transporte (TCP e UDP), assim como o overhead e as portas de atendimento associadas.

Posteriormente observamos a transferência de um ficheiro recorrendo a serviços diferentes (SFTP, FTP, TFTP, HTTP). Deste modo foi possível identificar as diferentes fases para a sua conceção entre as quais: estabelecimento de conexão, transferência de dados e fim de conexão. Numa fase final distinguimos como o TCP e o UDP lidam com a perda e duplicação de pacotes.