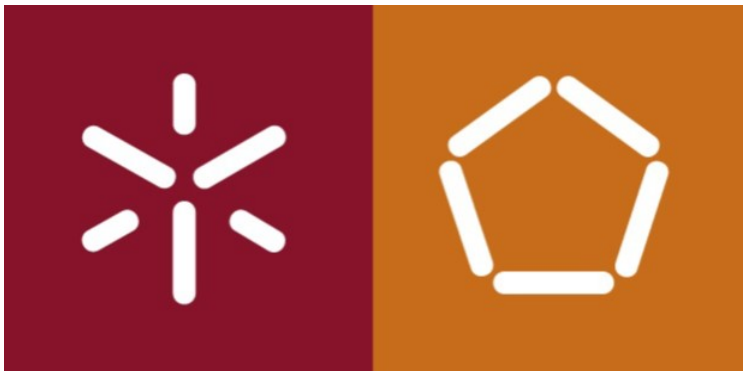

Universidade do Minho
Escola de Engenharia

Mestrado Integrado em Engenharia Informática

Comunicações por Computador

Relatório do Trabalho Prático 1

A78890 Alexandre Costa
A75248 Ana Sofia Gomes Marques
A65277 Flávio Manuel Machado Martins
A79799 Gonçalo Costeira

Grupo 8

3 Março 2020

Conteúdo

1 Questões e Respostas

1.1 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:

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 a 33534	8
telnet	telnet	TCP	23	20
ftp	ftp	TCP	21	20
Tftp	Tftp	UDP	69	8
Browser/http	http	TCP	80	20
nslookup	DNS	UDP	53	8
ssh	sshv2	TCP	22	20

Por forma a justificar os valores de overhead acima referidos para os diferentes comandos, sabemos que o segmento TCP possui 20 bytes de overhead de cabeçalho em cada segmento, enquanto o UDP possui apenas 8 bytes de overhead.

1.1.1 Ping

No caso do ping, este não possui protocolo de aplicação, protocolo de transporte e nem porta de atendimento. Por consequência, não terá nenhum overhead de transporte, uma vez que este opera ao mandar apenas mensagens ICMP para o destino à espera de obter um ICMP echo reply.
Este comportamento poderá ser observado no seguinte printscreen onde foi efetuado um ping para o endereço IP 193.137.16.65.

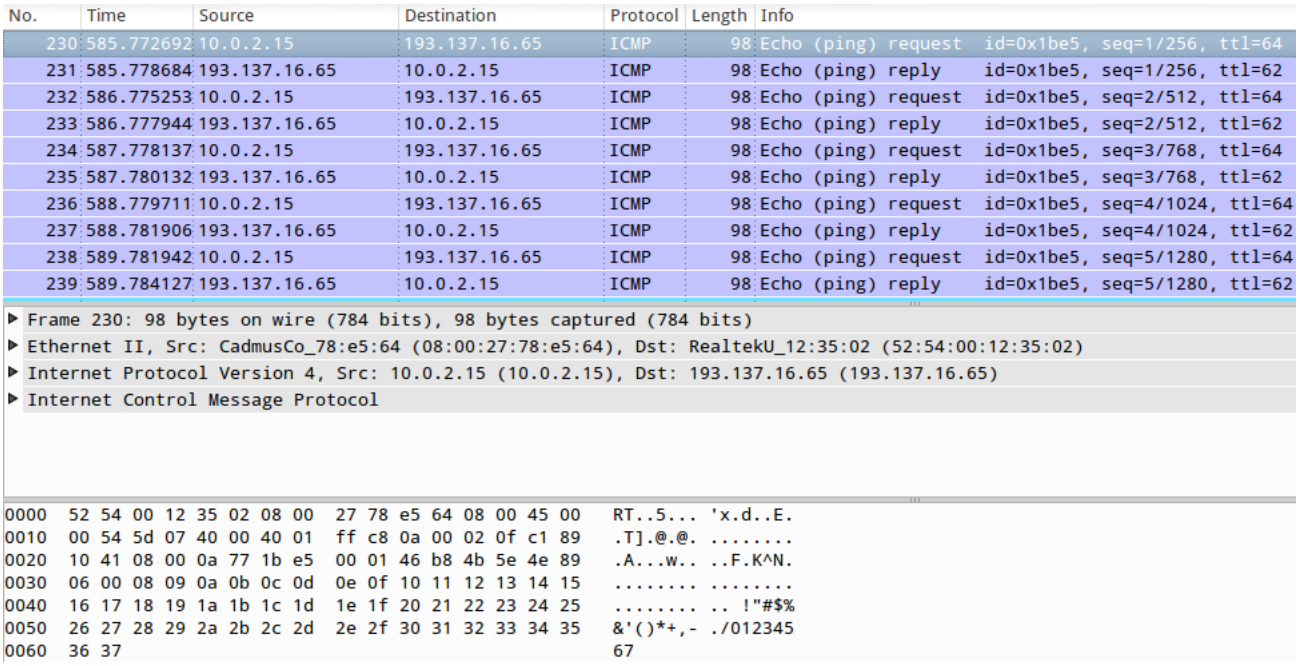

Figura 1: Tramas capturadas usando o comando ping.

1.1.2 Traceroute

No traceroute não existe protocolo de aplicação, mas possui protocolo de transporte UDP (User Datagram Protocol).
As portas de atendimento variam entre o 33434 e o 33534, como se pode verificar na próxima figura:

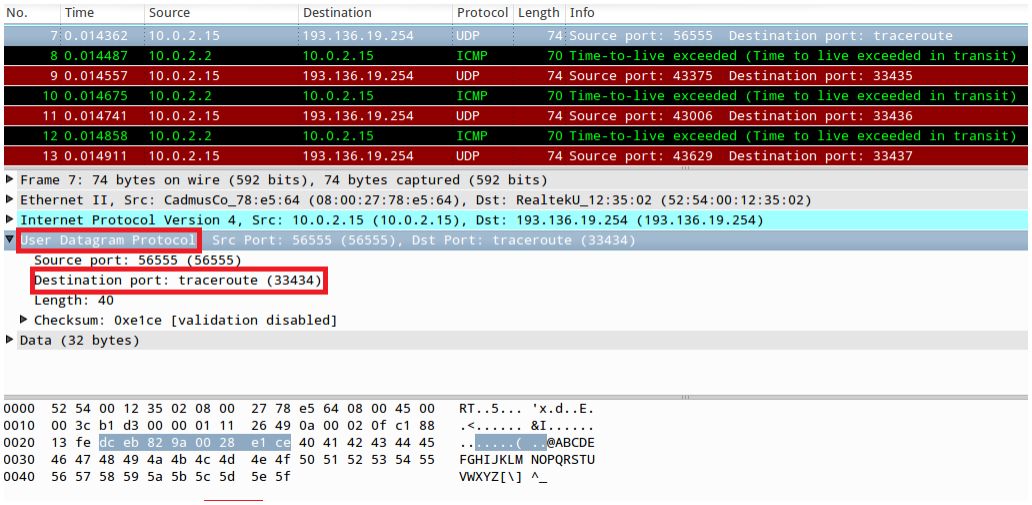

Figura 2: Tramas capturadas usando o comando traceroute.

1.1.3 Telnet

No comando telnet é utilizado o protocolo de aplicação telnet, este é utilizado para providenciar uma comunicação, utilizando uma conecção terminal virtual.
O protocolo de transporte é o TCP (Transmission Control Protocol) e possui uma porta de atendimento 23, como se pode verificar pela figura abaixo apresentada.

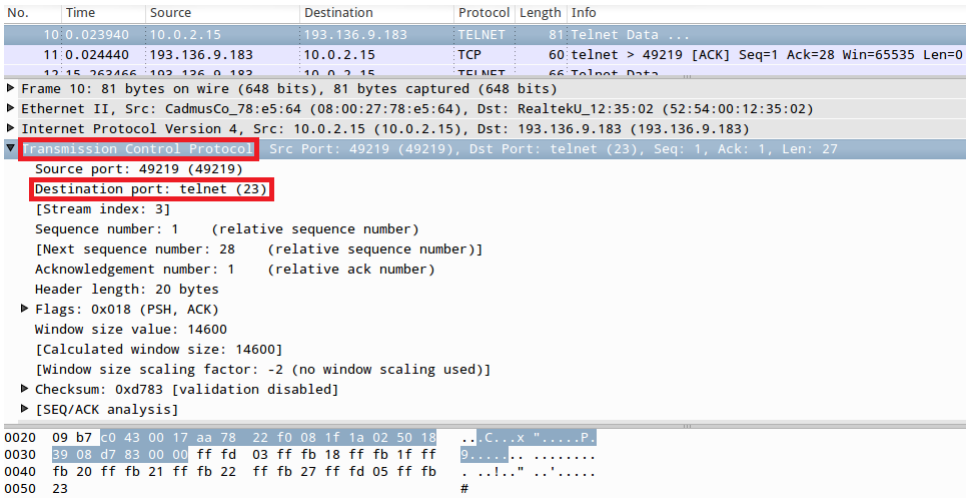

Figura 3: Tramas capturadas usando o comando telnet.

1.1.4 ftp

O ftp utiliza o protocolo, com o mesmo nome, sendo este usado para a transferênciade dados. O protocolo de aplicação é o ftp sendo que o de transporte é o TCP (Transmission Control Protocol), a porta de atendimento é a 21, como se pode verificar na seguinte figura:

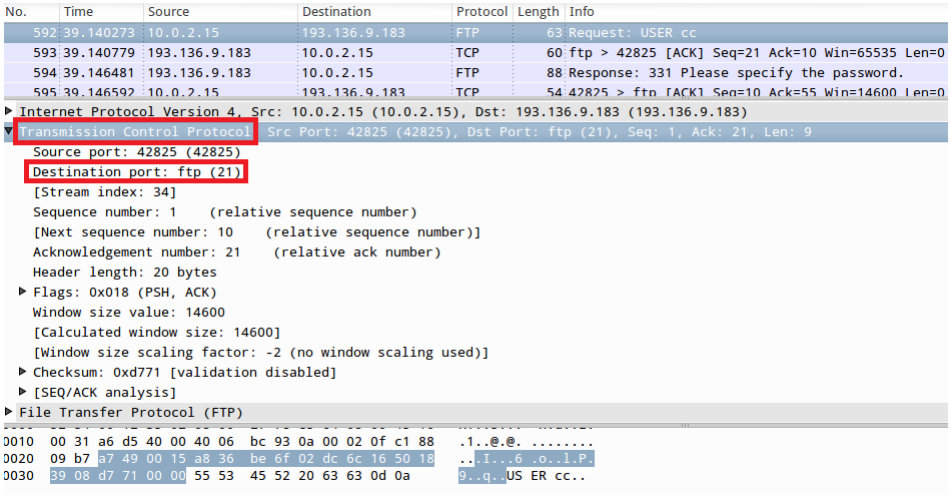

Figura 4: Tramas capturadas usando o comando ftp.

1.1.5 Tftp

Neste caso foi utilizado um comando curl que funciona como um protocolo de transferência de dados simples, semelhante ao ftp.
Deste modp, o protocolo de aplicação é o tftp, o protocolo de transporte é o UDP (User Datagram Protocol) com a porta de atendimento 69, assim como se pode verificar no printscreen abaixo apresentado.

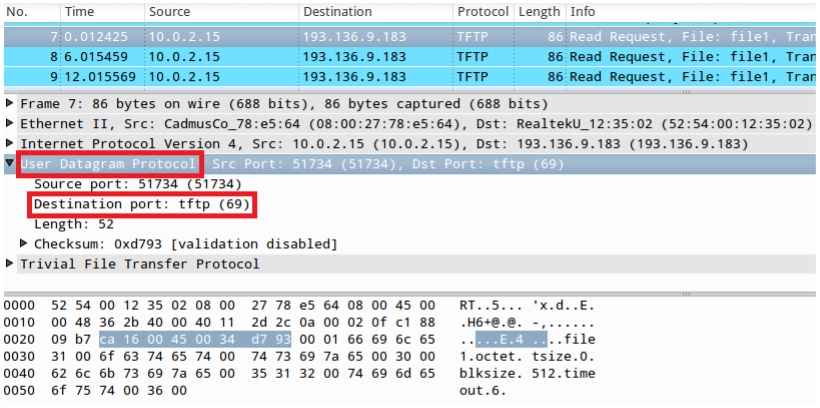

Figura 5: Tramas capturadas usando o comando Tftp.

1.1.6 browser/http

De modo a capturar o tráfego, foi utilizado o comando:
wget http://marco.uminho.pt/disciplinas/CC-MIEI/
Onde foi possível depois da análise da trama capturada que o protcolo de aplicação é o http e o de transporte o TCP(Transmission Control Protocol) e a porta de atendimento é a 80. Tudo isto pode ser verificado na figura seguinte:

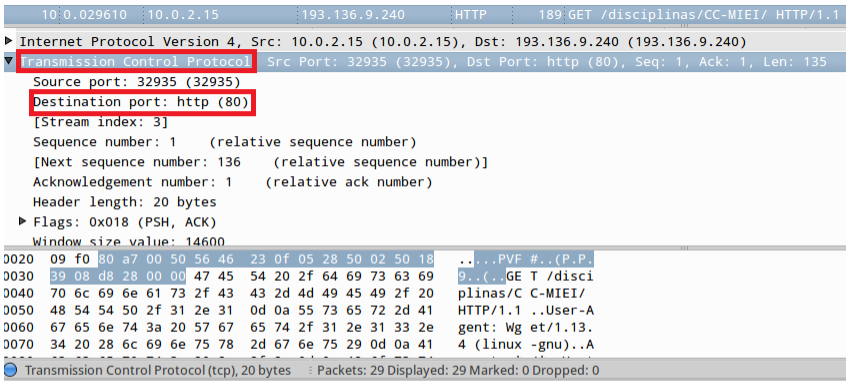

Figura 6: Tramas capturadas usando o comando http.

1.1.7 nslookup

O nslookup funciona como uma ferramenta de resolução de nomes como tal, o protocolo de aplicação é o DNS (Domain Name System), o de transporte é o UDP (User Datagram Protocol)com porta de atendimento 53, como se pode confirmar pela imagem abaixo apresentada.

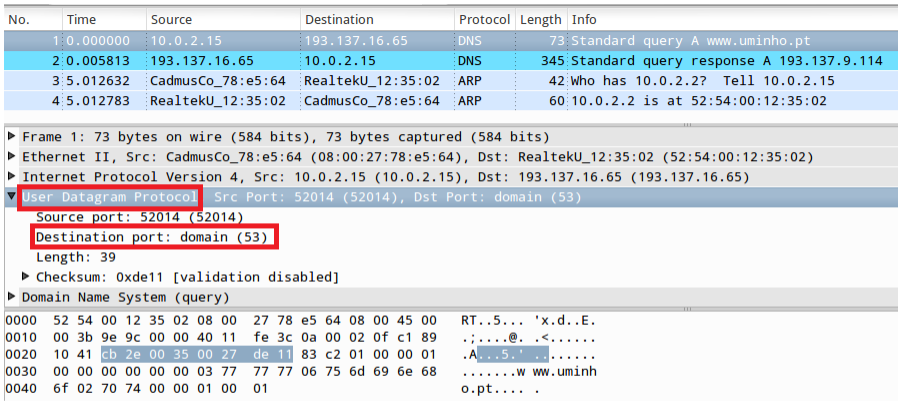

Figura 7: Tramas capturadas usando o comando nslookup.

1.1.8 ssh

O ssh é um protocolo de rede criptográfico cujo intuito é operar em redes de forma segura através de uma rede insegura, tendo sido efetuado um login remoto. O protocolo de aplicação portanto é ssh, o de transporte o TCP e a porta de atendimento a 22.

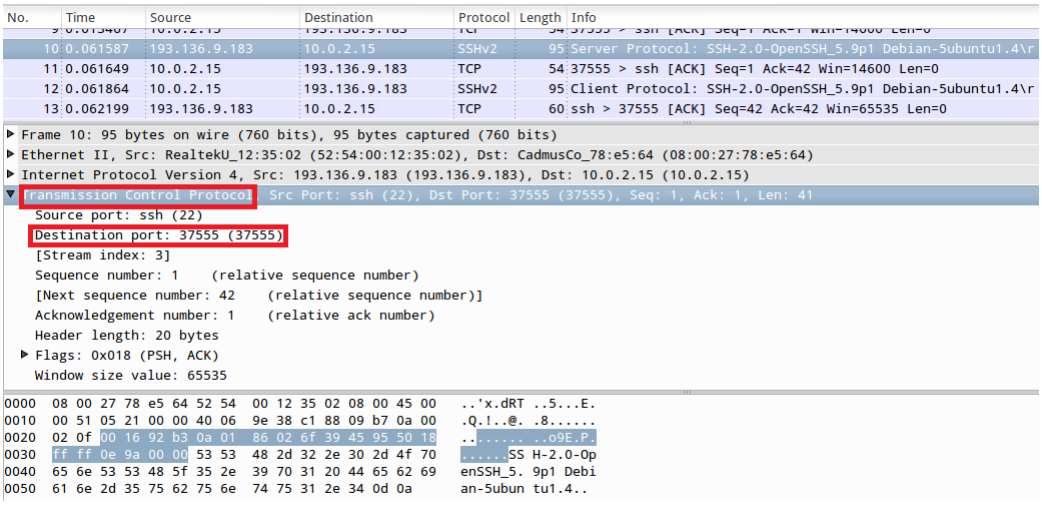

Figura 8: Tramas capturadas usando o comando ssh.

1.2 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.

1.2.1 FTP

146	422.095457	10.1.1.1	10.3.3.1	FTP	78 Request: RETR file1
147	422.095586	10.3.3.1	10.1.1.1	TCP	74 ftp-data > 59234 [SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2512272 TSecr=0 WS=16
148	422.095756	10.1.1.1	10.3.3.1	TCP	74 59234 > ftp-data [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460 SACK_PERM=1 TSval=2512272 TSecr=2
149	422.095904	10.3.3.1	10.1.1.1	TCP	66 ftp-data > 59234 [ACK] Seq=1 Ack=1 Win=14608 Len=0 TSval=2512272 TSecr=2512272
150	422.095965	10.3.3.1	10.1.1.1	FTP	130 Response: 150 Opening BINARY mode data connection for file1 (193 bytes).
151	422.095969	10.3.3.1	10.1.1.1	FTP-DAT	259 FTP Data: 193 bytes
152	422.096024	10.3.3.1	10.1.1.1	TCP	66 ftp-data > 59234 [FIN, ACK] Seq=194 Ack=1 Win=14608 Len=0 TSval=2512272 TSecr=2512272
153	422.096264	10.1.1.1	10.3.3.1	TCP	66 59234 > ftp-data [ACK] Seq=1 Ack=194 Win=15552 Len=0 TSval=2512272 TSecr=2512272
154	422.097146	10.1.1.1	10.3.3.1	TCP	66 59234 > ftp-data [FIN, ACK] Seq=1 Ack=195 Win=15552 Len=0 TSval=2512272 TSecr=2512272
155	422.097533	10.3.3.1	10.1.1.1	TCP	66 ftp-data > 59234 [ACK] Seq=195 Ack=2 Win=14608 Len=0 TSval=2512273 TSecr=2512272
156	422.097688	10.3.3.1	10.1.1.1	FTP	90 Response: 226 Transfer complete.

Figura 9: Tramas da transferência do file1 por FTP.

O protocolo de transferência de arquivos(FTP) utiliza como protocolo de transporte o TCP. Este pode ser dividido em três fases distintas:

- Estabelecimento de conexão;
- Transferência de dados;
- Fim da conexão.

Neste caso, da transferência do file1 via FTP, podemos ver que na fase de estabelecimento de conexão o Cliente envia um Request do ficheiro ao Servidor, e o Servidor entao envia um SYN(Synchronize) ao Cliente, sincronizando e iniciando assim uma conexão entre ambos. Durante esta conexão o Servidor envia exactamente um ficheiro para o cliente e no final termina a conexão. Se durante a sessão o cliente iniciar outra transferência é criada uma nova conexão.

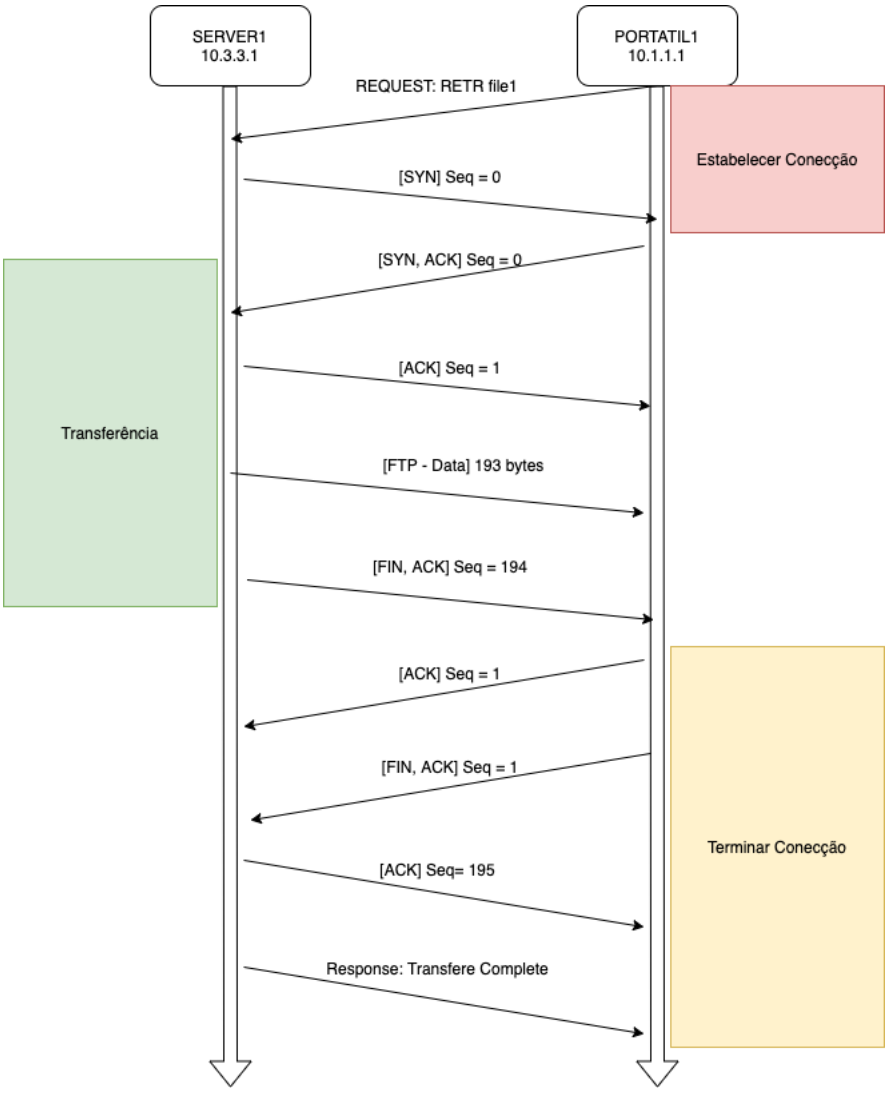

Figura 10: Diagrama Temporal da transferência por FTP

1.2.2 TFTP

43	194.306922	10.1.1.1	10.3.3.1	TFTP	56	Read Request, File: file1, Transfer type: octet
44	194.307381	10.3.3.1	10.1.1.1	TFTP	239	Data Packet, Block: 1 (last)
45	194.307875	10.1.1.1	10.3.3.1	TFTP	46	Acknowledgement, Block: 1

Figura 11: Tramas da transferencia do file1 por TFTP.

Na transferencia por TFTP é utilizado como protocolo de transporte o UDP. Como pode ser observado na figura 11, é inicialmente feito um Read Request pelo cliente ao servidor, de seguida o pacote é transferido do servidor para o cliente, e por fim o cliente envia um acknowledgement ao servidor.

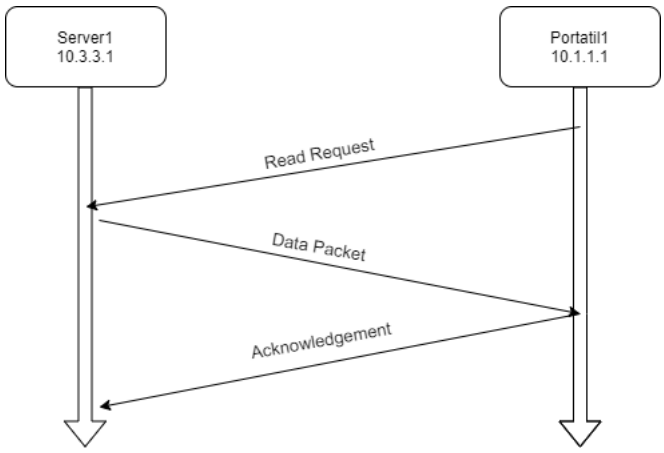

Figura 12: Diagrama temporal da transferência por TFTP.

1.3 Questão 3 - Com base nas experiências realizadas, distinga e compare sucintamente as quatro aplicações de transferência de ficheiros que usou nos seguintes pontos

1.3.1 FTP - File Transfer Protocol

(i) Uso da camada de transporte: O servidor remoto aceita uma conexão de controlo do cliente a nível local. O cliente envia comandos para o servidor e a conexão persiste ao longo de toda a sessão (tratando-se assim de um protocolo TCP);

(ii) Eficiência na transferência: Não muito eficiente uma vez que envia toda a informação numa só Stream de bits guar- dada como uma string limitando a capacidade de controlo de erros e ate mesmo velocidade de transmissão.

(iii) Complexidade: Envia uma Stream de bits guardada como uma única string que contem a informação e toda o overhead necessário para transmissão (ex : nome do ficheiro, tamanho, timestamp).

(iv) Segurança: FTP é considerada uma das opções menos seguras no que toca a transmissão de ficheiros em rede uma vez que não encripta os dados e para a transferência destes não considera a segurança desta, havendo diversos tipos de formas de intercetar transferências em FTP como por exemplo : FTP Bounce Attack , FTP Brute Force Attack , Packet Capture (or Sniffing), Spoof Attack ou Port Stealing.

1.3.2 SFTP -Secure File Transfer Protocol

(i) Uso da camada de transporte: A transferência dos dados e realizado através de uma conexão previamente assegurada, que utiliza o protocolo SSH (Secure Shell protocol).

(ii) Eficiência na transferência: STFP e um protocolo baseado em transferência de pacotes ao invés de baseados em texto. Assim ao enviar pequenos pacotes com a informação devidamente dividida e preparada entre si consegue ser um método mais rápido e a capaz de produzir um menor numero de erros e de perda de informação.

(iii) complexidade; Transferências em SFTP são realizadas através de uma conexão SSH e ocorre a prepa- ração da informação em data packages. Alem disso, esta transferência é realizada sobre o controlo de conexão principal , eliminando a necessidade de abrir uma nova conexão de dados somente para este evento.

(iv) segurança: Uma vez que utiliza o protocolo SSH é considerada intrinsecamente segura.

1.3.3 TFTP - Trivial File Transfer Protocol

(i) Uso da camada de transporte: Utiliza o protocolo de transporte UDP.

(ii) Eficiência na transferência: Ao utilizar os protocolo UDP o TFTP peca por não possuir tanto controle de erros, ha- vendo assim alguma perda na sua eficiência. Para além disso tem também de suportar a sessão criada para o envio da informação.

(iii) Complexidade: Transferência é iniciada pelo cliente pedindo para ler ou escrever um ficheiro num ser- vidor. Se este aceita o pedido o ficheiro e enviado em blocos de tamanho fixo , cada um geralmente possuindo apenas um único pacote IP de modo a evitar a sua fragmentação e deve esperar pelo acknowledgment de modo a poder enviar o pacote seguinte.

(iv) Segurança: O TFTP não possui nenhum mecanismo de controlo de acesso. Assim devemos ter cui- dado ao enviar ficheiros de carácter privado e ter em atenção aos direitos garantidos ao servidor TFTP de modo a não violar a segurança do servidor onde o ficheiro se encontra guardado.

1.3.4 HTTP - Hypertext Control Protocol

(i) Uso da camada de transporte: Aplicação de camada de transporte desenhada com a framework do protocolo de Internet em mente, assumindo assim capacidade de transporte segura e eficaz.

(ii) Eficiência na transferência: - Eficiente no sentido em que a sua arquitetura é projetada para permitir intermediação entre elementos de rede de modo a melhorar e permitir comunicações entre servidores e clientes. Web sites com muita procura e tráfego normalmente apresentam servers cache que permi- tem melhorar o tempo de resposta.

(iii) Complexidade: Devido a utilização de servers cache e outros métodos que permitem melhorar a eficiência da transmissão, verificamos que existe um aumento a nível de recursos utilizados e também na complexidade da sua gestão que permite este melhoramento.

(iv) segurança: O HTTP utiliza vários métodos de autentificação para a ocorrência de transferências, quer seja acessos básicos ou acessos a web sites com informação maus importante os métodos de autentificação e segurança vão melhorando, fornecendo assim um ambiente seguro para o envio de dados.

1.4 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)

Em transmissão de dados é normal ouvir falar em package loss, este ocorre quando um ou mais pacotes que navegam sobre uma determinada rede falham em alcançar o destinatário, é assim lógico que ao aumentar a percentagem de pacotes a serem enviados na ligação, se verifique que mais pacotes são perdidos ou duplicados.
Sabemos que aplicações de transferência fiáveis, sendo um exemplo a SFTP (SSH File Transfer Protocol),tipicamente utilizado com o protocolo de segurança SSH, que utiliza protocolo de transporte TCP (Transmission Control Protocol), este permite a recuperação de pacotes perdidos e a eliminação de pacotes duplicados, utilizam a retransmissão para garantir que todos os dados chegam ao destino e que estes são confiáveis.

A perda de pacotes em uma conexão TCP também é usada para evitar congestionamentos e, portanto, produz uma taxa de transferência intencionalmente reduzida para a conexão.
A perda de pacotes está intimamente associada a considerações de qualidade de serviço. A quantidade de perda de pacotes aceitável depende do tipo de dados que está sendo enviado, mas ronda em torno de valores bastante baixos.
Assim, ao aumentar o package loss, acabamos por ter um maior overhead, causado pelo facto de que quanto maior for a perda, mais tráfego vai circular na rede por forma a ser capaz de corrigir os erros existentes, o que acaba por diminuir o desempenho global da rede.

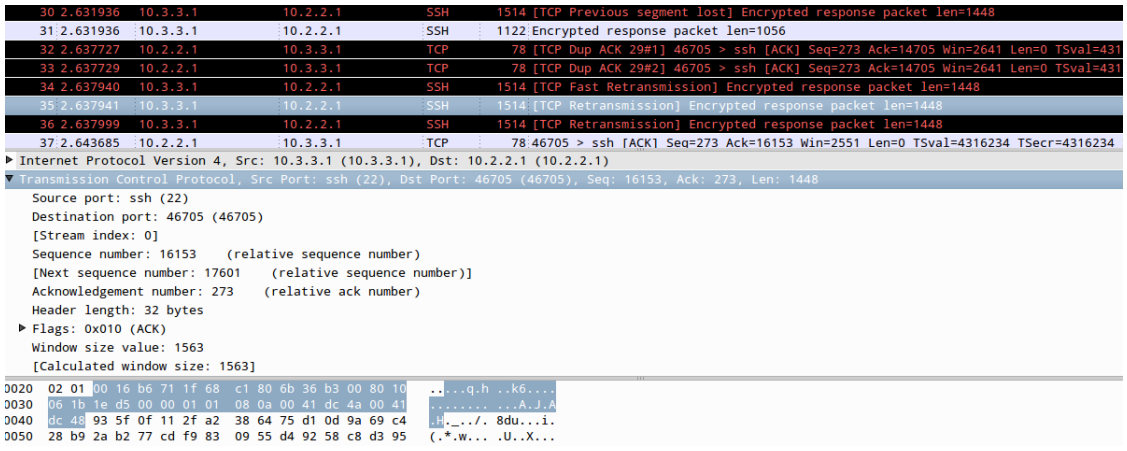

Figura 13: SFTP com perdas e duplicações.

2 Conclusão

Com a realização deste trabalho podemos por em prática alguns dos aspetos que aprendemos nas aulas teóricas, tais como, sermos capazes de classificar os protocolos de transporte, respetivamente UDP e TCP, em diversas aplicações diferentes e o seu respetivo overhead.
De seguida cada uma destas aplicações foi testada utilizando a topologia CORE fornecida. Foi feita uma análise dos resultados entre as várias aplicações, sendo possível estabelecer uma conecção entre a consequência do uso de diferentes protocolos de transporte em diferentes aplicações.
Finalmente, foi feita uma análise numa situação de package loss de forma a perceber como os mecanismos de correção de erros funcionam.
Assim sendo, com a elaboração deste trabalho foram aprofundados os conhecimentos relativos à camada de transporte, cruciais para a melhor compreensão do seu funcionamento e importância.