TP1: Protocolos da Camada de Transporte

Catarina Machado - a81047 Gonçalo Faria - a86264
João Vilaça - a82339

26 de Fevereiro de 2019

Resumo

Análise do uso da camada de transporte por parte de aplicações
Keywords: Camada de transporte, XubunCORE, PING, SFTP, FTP, TFTP, HTTP.

Questões e Respostas

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	Não	Não	Não	Não
traceroute	MDNS	UDP	33450	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	ssh	TCP	22	20

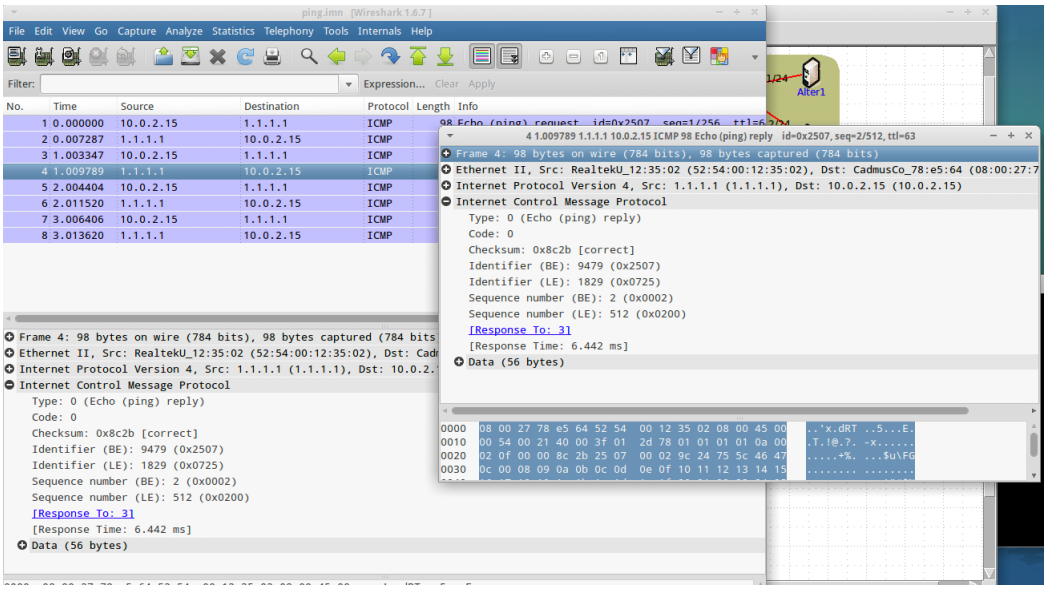

Figura 1: Ping.

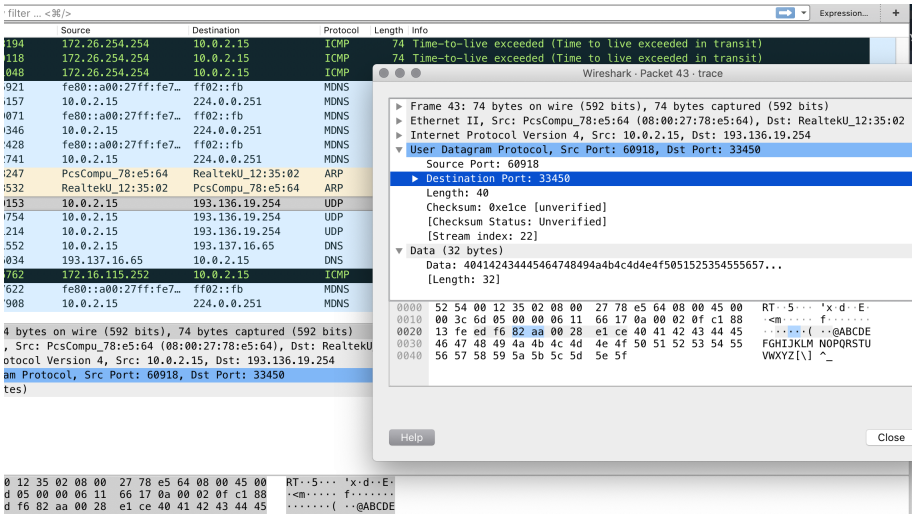

Figura 2: Traceroute.

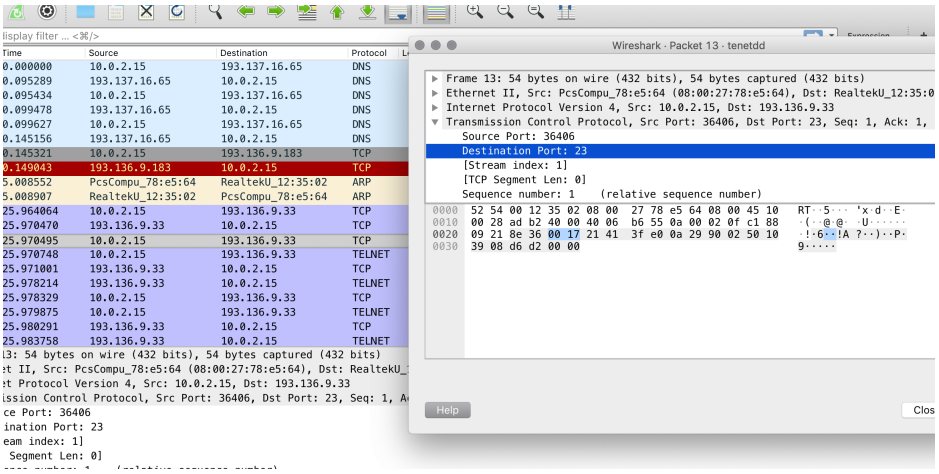

Figura 3: Telnet.

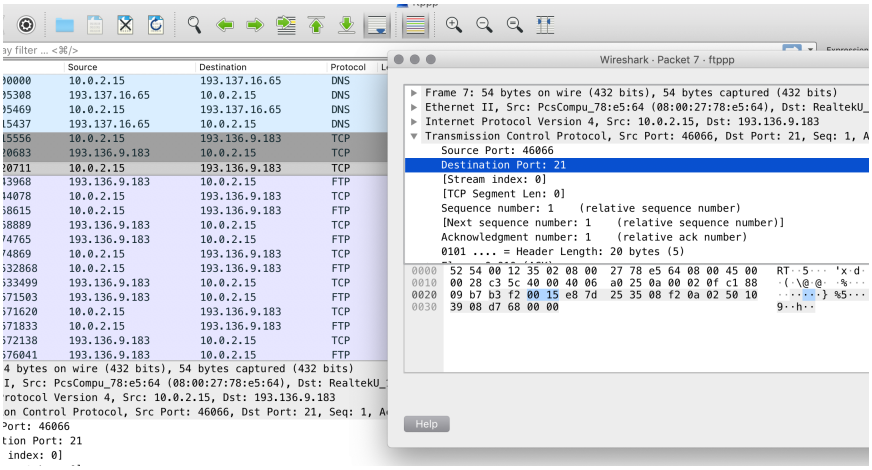

Figura 4: Ftp.

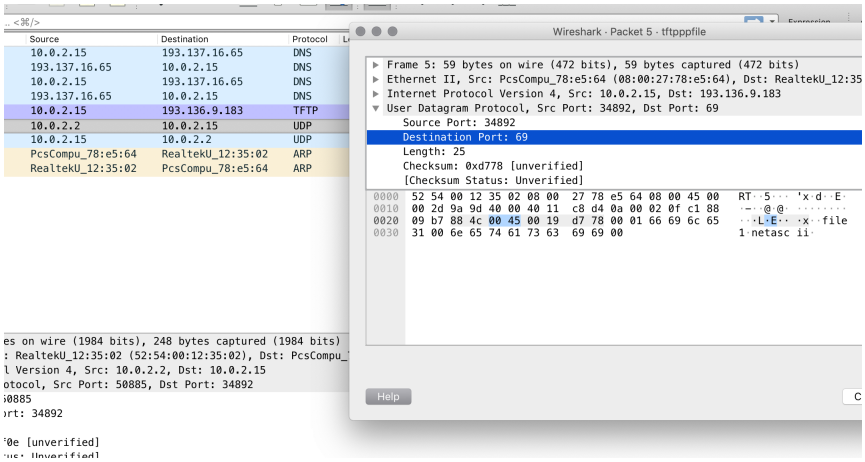

Figura 5: Tftp.

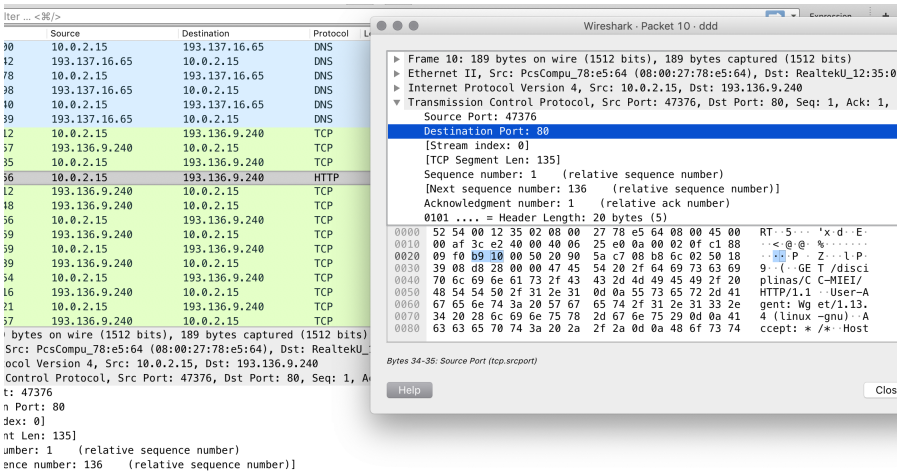

Figura 6: Browser/http.

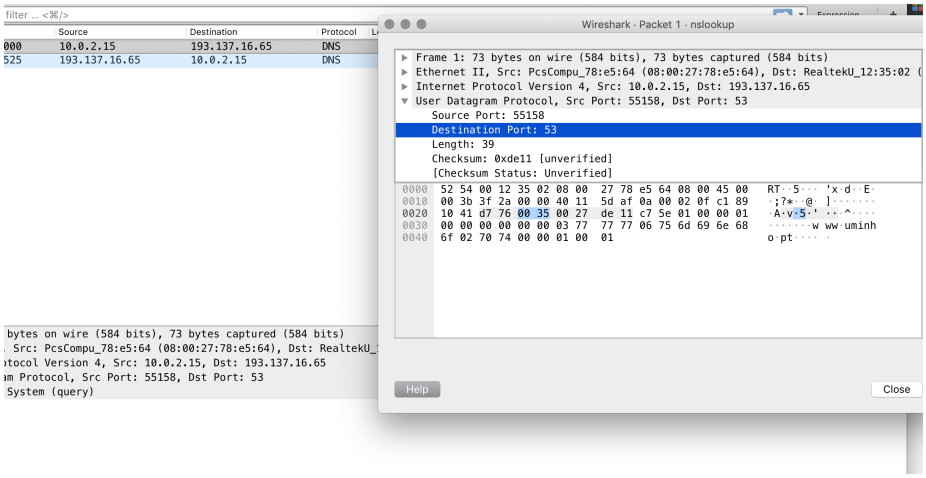

Figura 7: Nslookup.

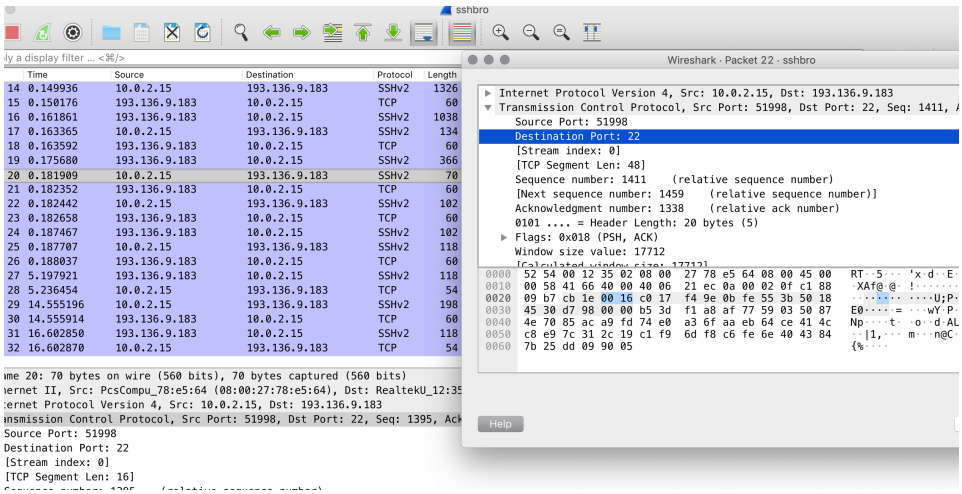

Figura 8: SSH.

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.

(Nota: a transferência por FTP envolve mais que uma conexão FTP, nomeadamente uma de controlo [ftp] e outra de dados [ftp-data]. Faça o diagrama apenas para a conexão de transferência de dados do ficheiro mais pequeno)

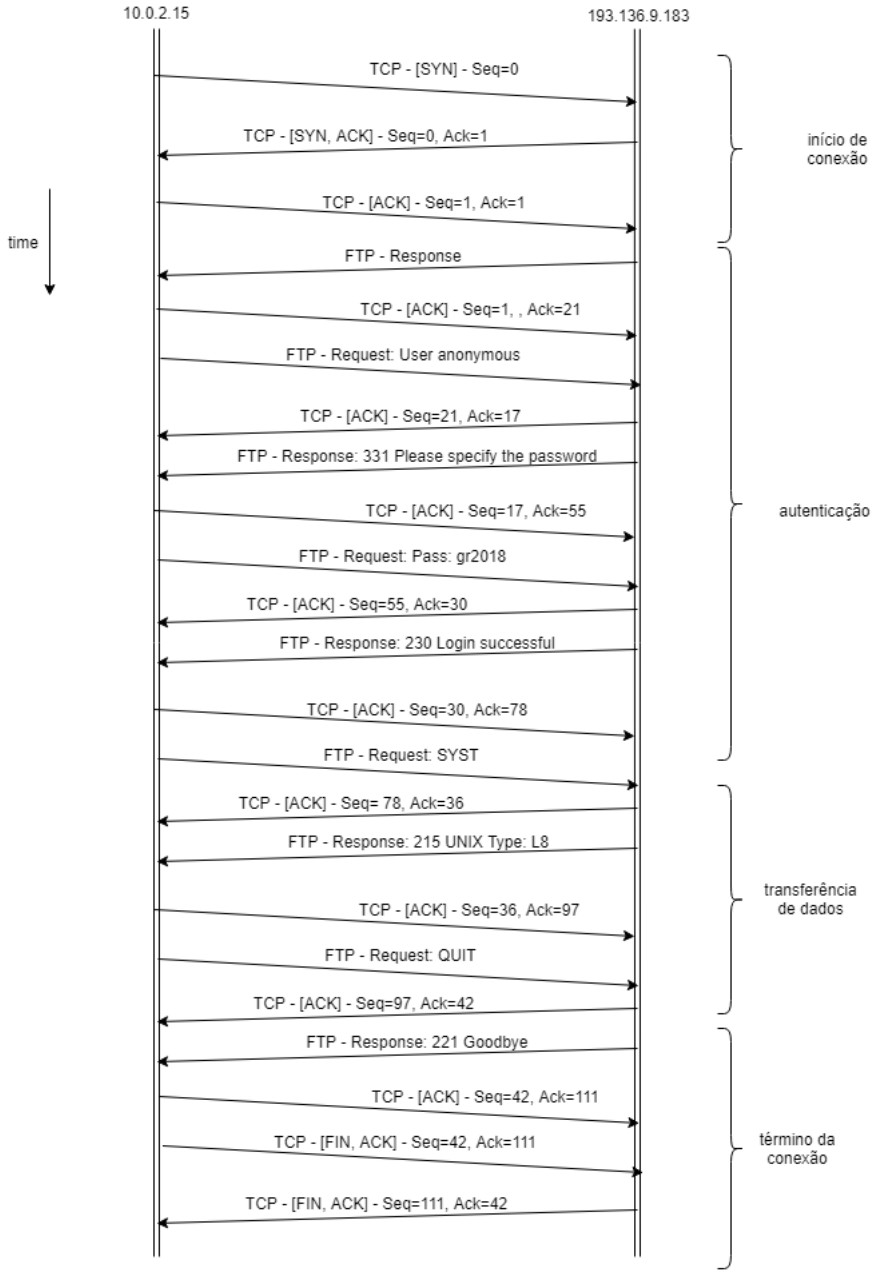

Figura 9: Diagrama Temporal das Transferências do file1 por FTP.

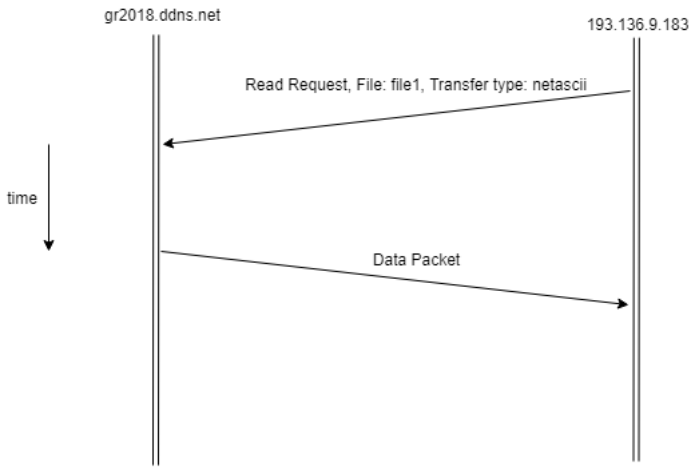

Figura 10: Diagrama Temporal das Transferências do file1 por TFTP.

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;

Pelo que podemos observar, através da análise dos pacotes no wireshark no router 1 da topologia do CORE, o protocolo HTTP utiliza o protocolo TCP como protocolo da camada de transporte e possui vários esquemas de autenticação. No entanto, este é inerentemente inseguro pois para qualquer pessoa na rede é possível ver o conteúdo dos ficheiros antes de chegar ao receptor.

O FTP é um serviço básico de transferência fiável de ficheiros. O FTP usa o protocolo TCP mas não aparenta implementar nenhuma segurança adicional (até pudemos verificar que no processo de autenticação, que este permite, as palavras-chave são transmitidas na rede em texto claro), sendo por isso bastante simples apesar de permitir uma grande variedade de opções e comandos. Tem alguns problemos em termos de eficiência por causa do elevado overhead.

O SFTP é um protocolo muito seguro que, tal commo o FTP, requer autenticação cliente servidor. Como camada de transporte utiliza TCP. Comparativamente ao FTP, o SFTP é mais seguro uma vez que usa ligações SSH. O uso de SSH causa overhead que diminui a eficiência da transmissão.

O TFTP é um serviço básico de transferência não fiável de ficheiros, dado que usa UDP. Este protocolo não implementa nenhuma segurança adicional, tal como o FTP, e não tem mecanismos de autenticação, sendo então também bastante simples. Este protocolo é muito eficiente na transmissão por ter baixo overhead.

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).

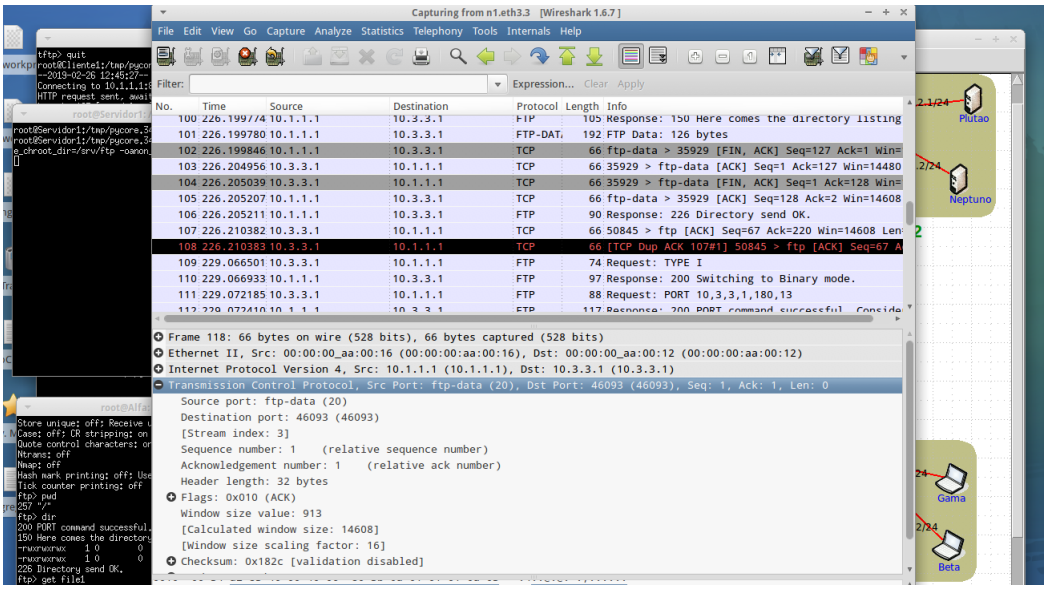

Figura 11: Pedido de retransmissão de um pacote.

No desenvolvimento de uma aplicação fiável é muito importante ter um cuidado especial com a escolha do protocolo de transporte.

Uma das principais opções para esta escolha é o TCP, um protocolo de transporte fiável, porque garante que todos os pacotes são enviados na ordem correta e sem erros. Para cada pacote recebido é sempre enviado pelo menos um Acknowledgment em como esse pacote foi recebido com sucesso. Infelizmente todo este processo é bastante complexo e exige que sejam trocados muitos pacotes de controlo entre ambas as partes. Assim sendo, é indiscutível que é enorme a influência das caraterísticas da rede neste protocolo. Quando estamos numa rede de menor qualidade são perdidos e corrompidos imensos pacotes o que obriga a que sejam transmitidas mensagens de erro e a que sejam reenviados os vários pacotes causando mais sobrecarga na rede e atraso na aplicação.

Por outro lado, as alternativas não fiáveis, como o UDP, também são uma opção no desenvolvimento de uma aplicação fiável uma vez que poderá ser a própria aplicação a garantir que os dados são efetiva e corretamente recebidos. A vantagem do UDP, sendo um protocolo que enfatiza a redução de latência da ligação, como não faz estas verificações, não sobrecarrega a rede nem com Acknowledgment nem com reenvios em condições de redes menos favoráveis.

Conclusões

Este trabalho prático serviu de complemento às aulas teóricas e ajudou a consolidar a matéria lecionada nas mesmas.

Depois de finalizado o trabalho prático número 1, relativo aos protocolos de Transporte, obtivemos mais conhecimentos sobre o funcionamento deste nível de rede.

Conceitos como, por exemplo, TCP e UDP, várias protocolos aplicacionais e a comunicação entre aplicações foram recordados e aplicados.

O aspeto em que investimos mais tempo neste trabalho prático foi na análise do tráfego gerado pelo uso dos diferente protocolos quer de Aplicação quer de Transporte recorrendo ao Wireshark. Para exemplos de cada um destes tipos protocolos foram analisados os detalhes do encapsulamento dos datagramas.