Escola Politécnica - Pontifícia Universidade Católica do Rio Grande do Sul - Curso de Engenharia de Software - Laboratório de Redes - Professora Cristina Nunes
Arthur Henrique Henz (henzarthur@gmail.com) e Pablo Miguel Chong Alles (p.chong@edu.pucrs.br)

Relatório da atividade de Laboratório de Redes usando sockets

1. INTRODUÇÃO

Este relatório descreve a elaboração de atividades de programar aplicações em rede usando sockets. Neste caso, foi utilizada uma aplicação cliente/servidor entre dois computadores, em que posteriormente foi representado nas imagens de captura de pacotes o cliente será a máquina com o endereço IP 10.32.143.132 e o servidor a máquina com o endereço IP 10.32.143.221. O cliente diante dessa situação deve ser capaz, por meio da aplicação e da rede, de enviar um arquivo texto para o servidor. Dessa forma, foi utilizado o programa de captura de pacotes em rede Wireshark para que se pudesse visualizar as principais diferenças de comportamento dos protocolos de transporte TCP e UDP.

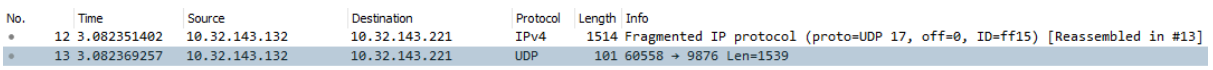
Após a conclusão de quatro algoritmos, sendo eles dois servidores e dois clientes TCP e UDP, realizamos a entrega de dois arquivos: um contendo 1.5Kb, e outro, 10Kb. Com o auxílio do *netem* (Network Emulation para Linux), pudemos manipular a latência com variação de milissegundos, além de determinar uma porcentagem de perda de todos os pacotes que transitam na rede (neste caso, 20%).

A seguir a primeira seção deste relatório é reservada a explicar as diferenças entre os sockets TCP e UDP no quesito de tráfego de rede e envio de pacotes, seguida da segunda seção que apresenta o comportamento desses sockets diante de situações adversas como perda de pacotes e latência (atraso) variável.

2. COMPARANDO COMPORTAMENTO TCP E UDP

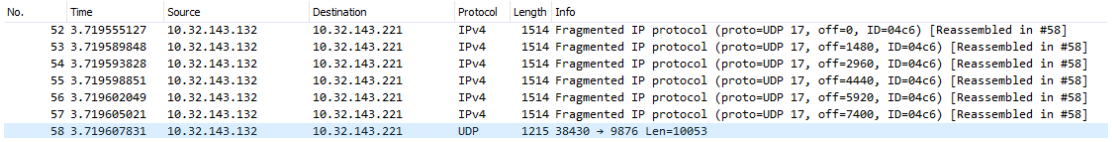
Depois de termos desenvolvido as aplicações para envio dos arquivos de texto com as mensagens que queríamos passar do cliente para o servidor, um com 1.5Kb e outro com 10Kb, usamos ambas aplicações tanto UDP quanto TCP com a captura de pacotes do Wireshark ligada para realizar o monitoramento de tráfego na rede. Dessa maneira, foi possível visualizar o comportamento de empacotamento de dados de ambos os protocolos para que pudéssemos realizar comparações.

A respeito do UDP, a característica que se destacou foi o recebimento único de pacote para cada envio. Como não há necessidade de estabelecer nenhum tipo de conexão antes do envio por conta do próprio protocolo não ser orientado à conexão, as instruções são relativamente mais simples e didáticas do ponto de vista lógico. Como o número de Bytes do protocolo IP é grande demais para passar pelo tunelamento, principalmente, no caso do arquivo de 10Kb, ocorre a sua fragmentação em múltiplos pacotes IP fragmentados como é possível comparar pela figura 1 e a figura 2 que, respectivamente, referem-se ao envio da mensagem de 1.5Kb e 10Kb. Abaixo, é possível visualizar as tais figuras do protocolo na captura desses pacotes feita pelo Wireshark, com os arquivos de 1.5Kb e 10Kb.



No.	Time	Source	Destination	Protocol	Length	Info
12	3.082351402	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=ff15) [Reassembled in #13]
13	3.082369257	10.32.143.132	10.32.143.221	UDP	101	60558 → 9876 Len=1539

Figura 1. Captura de pacote do envio de mensagem de 1.5Kb feita com socket UDP

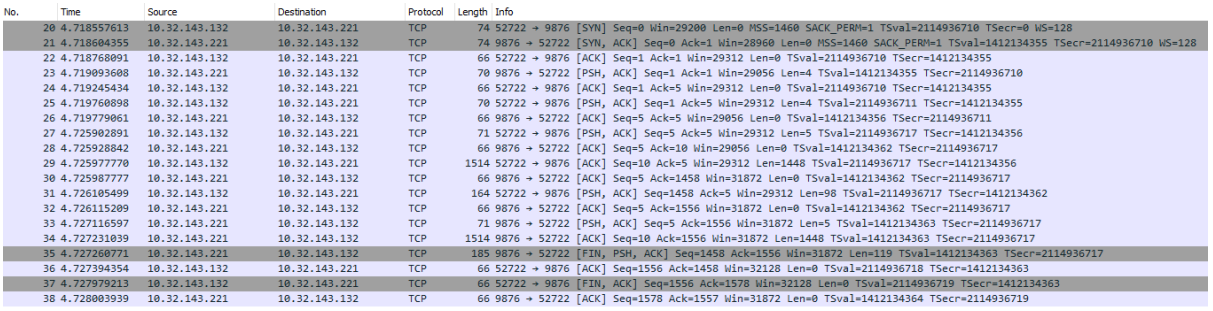


No.	Time	Source	Destination	Protocol	Length	Info
52	3.719555127	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=04c6) [Reassembled in #58]
53	3.719589848	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=1480, ID=04c6) [Reassembled in #58]
54	3.719593828	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=2960, ID=04c6) [Reassembled in #58]
55	3.719598851	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=4440, ID=04c6) [Reassembled in #58]
56	3.719602049	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=5920, ID=04c6) [Reassembled in #58]
57	3.719605021	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=7400, ID=04c6) [Reassembled in #58]
58	3.719607831	10.32.143.132	10.32.143.221	UDP	1215	38430 → 9876 Len=10053

Figura 2. Captura de pacote do envio de mensagem de 10Kb feita com socket UDP

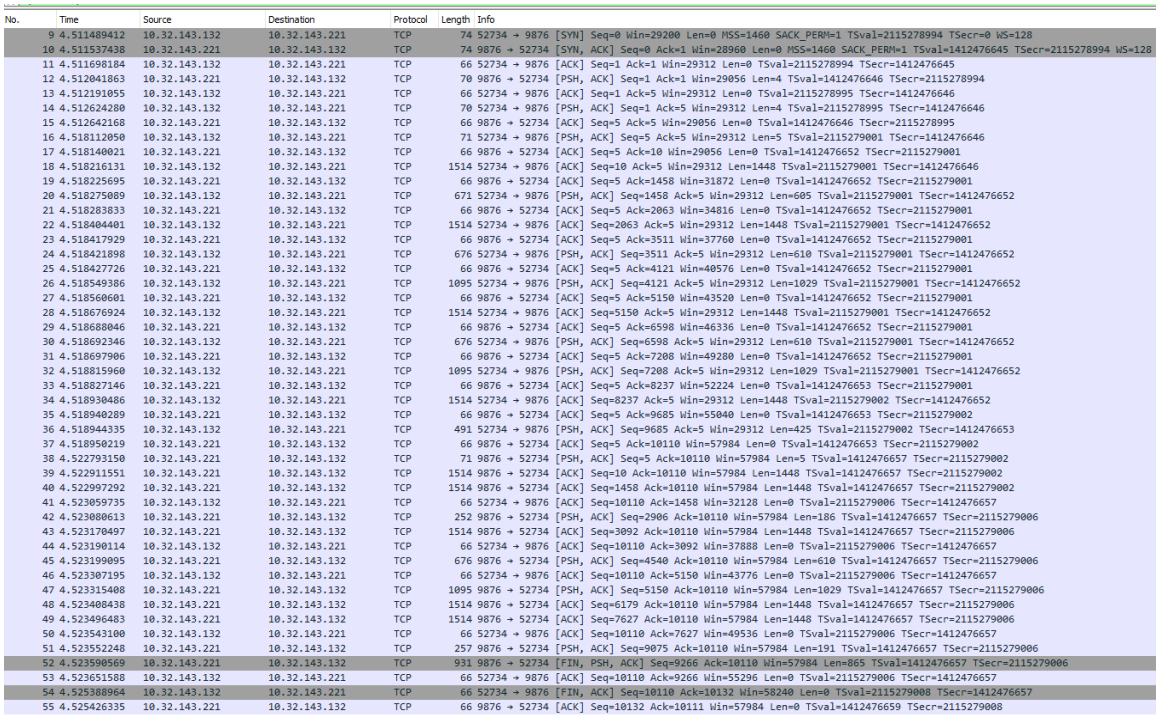
Após a análise do UDP, executamos os algoritmos cliente e servidor em máquinas distintas para realizar envios dos mesmos arquivos com o protocolo TCP. Como este protocolo é orientado a conexão, as áreas em cinza representam o que é chamado de Three-way Handshake, o que consiste em comandos em ACK (reconhecer que a outra máquina recebeu o dado), SYN (estabelece uma conexão TCP), FIN (utilizado para encerrar a sessão) e PSH (usado para dizer ao cliente/servidor para prosseguir no envio dos dados mesmo se o buffer não tenha sido preenchido).

Por ser um protocolo relativamente mais complexo, há a abertura de algumas possibilidades de controle interessante, em troca de velocidade. Algumas dessas características são: reenvio de pacotes, a possibilidade da confirmação da chegada dos pacotes e o envio de múltiplos pacotes fragmentando a mensagem se isso for necessário. Abaixo, é possível visualizar o envio dos dois mesmos pacotes anteriormente enviados no UDP, mas dessa vez enviados com o socket TCP.



No.	Time	Source	Destination	Protocol	Length	Info
20	4.718557613	10.32.143.132	10.32.143.221	TCP	74	52722 → 9876 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=2114936710 TSecr=0 WS=128
21	4.718604355	10.32.143.221	10.32.143.132	TCP	74	9876 → 52722 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=1412134355 TSecr=2114936710 WS=128
22	4.718768091	10.32.143.132	10.32.143.221	TCP	66	52722 → 9876 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=2114936710 TSecr=1412134355
23	4.719093608	10.32.143.221	10.32.143.132	TCP	70	9876 → 52722 [PSH, ACK] Seq=1 Ack=1 Win=29056 Len=4 TSval=1412134355 TSecr=2114936710
24	4.719245434	10.32.143.132	10.32.143.221	TCP	66	52722 → 9876 [ACK] Seq=1 Ack=5 Win=29312 Len=0 TSval=2114936710 TSecr=1412134355
25	4.719760898	10.32.143.132	10.32.143.221	TCP	70	52722 → 9876 [PSH, ACK] Seq=1 Ack=5 Win=29312 Len=4 TSval=2114936711 TSecr=1412134355
26	4.719779061	10.32.143.221	10.32.143.132	TCP	66	9876 → 52722 [ACK] Seq=5 Ack=5 Win=29056 Len=0 TSval=1412134356 TSecr=2114936711
27	4.725902891	10.32.143.132	10.32.143.221	TCP	71	52722 → 9876 [PSH, ACK] Seq=5 Ack=5 Win=29312 Len=5 TSval=2114936717 TSecr=1412134356
28	4.725928842	10.32.143.221	10.32.143.132	TCP	66	9876 → 52722 [ACK] Seq=5 Ack=10 Win=29056 Len=0 TSval=1412134362 TSecr=2114936717
29	4.725977770	10.32.143.132	10.32.143.221	TCP	1514	52722 → 9876 [ACK] Seq=10 Ack=5 Win=29312 Len=1448 TSval=2114936717 TSecr=1412134356
30	4.725987777	10.32.143.221	10.32.143.132	TCP	66	9876 → 52722 [ACK] Seq=5 Ack=1458 Win=31872 Len=0 TSval=1412134362 TSecr=2114936717
31	4.726105499	10.32.143.132	10.32.143.221	TCP	164	52722 → 9876 [PSH, ACK] Seq=1458 Ack=5 Win=29312 Len=98 TSval=2114936717 TSecr=1412134362
32	4.726115209	10.32.143.221	10.32.143.132	TCP	66	9876 → 52722 [ACK] Seq=5 Ack=1556 Win=31872 Len=0 TSval=1412134362 TSecr=2114936717
33	4.727116597	10.32.143.221	10.32.143.132	TCP	71	9876 → 52722 [PSH, ACK] Seq=5 Ack=1556 Win=31872 Len=5 TSval=1412134363 TSecr=2114936717
34	4.727231039	10.32.143.221	10.32.143.132	TCP	1514	9876 → 52722 [ACK] Seq=10 Ack=1556 Win=31872 Len=1448 TSval=1412134363 TSecr=2114936717
35	4.727260771	10.32.143.221	10.32.143.132	TCP	185	9876 → 52722 [FIN, PSH, ACK] Seq=1458 Ack=1556 Win=31872 Len=119 TSval=1412134363 TSecr=2114936717
36	4.727394354	10.32.143.132	10.32.143.221	TCP	66	52722 → 9876 [ACK] Seq=1556 Ack=1458 Win=32128 Len=0 TSval=2114936718 TSecr=1412134363
37	4.727979213	10.32.143.132	10.32.143.221	TCP	66	52722 → 9876 [FIN, ACK] Seq=1556 Ack=1578 Win=32128 Len=0 TSval=2114936719 TSecr=1412134363
38	4.728003939	10.32.143.221	10.32.143.132	TCP	66	9876 → 52722 [ACK] Seq=1578 Ack=1557 Win=31872 Len=0 TSval=1412134364 TSecr=2114936719

Figura 3. Captura de pacote do envio de mensagem de 1.5Kb feita com socket TCP



No.	Time	Source	Destination	Protocol	Length	Info
9	4.511489412	10.32.143.132	10.32.143.221	TCP	74	52734 → 9876 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=2115278994 TSecr=0 WS=128
10	4.511537438	10.32.143.221	10.32.143.132	TCP	74	9876 → 52734 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=1412476645 TSecr=2115278994 WS=128
11	4.511698184	10.32.143.132	10.32.143.221	TCP	66	52734 → 9876 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=2115278994 TSecr=1412476645
12	4.512041863	10.32.143.221	10.32.143.132	TCP	70	9876 → 52734 [PSH, ACK] Seq=1 Ack=1 Win=29056 Len=4 TSval=1412476646 TSecr=2115278994
13	4.512191055	10.32.143.132	10.32.143.221	TCP	66	52734 → 9876 [ACK] Seq=1 Ack=5 Win=29312 Len=0 TSval=2115278995 TSecr=1412476646
14	4.512624280	10.32.143.132	10.32.143.221	TCP	70	52734 → 9876 [PSH, ACK] Seq=1 Ack=5 Win=29312 Len=4 TSval=2115278995 TSecr=1412476646
15	4.512642168	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=5 Ack=5 Win=29056 Len=0 TSval=1412476646 TSecr=2115278995
16	4.518112050	10.32.143.132	10.32.143.221	TCP	71	52734 → 9876 [PSH, ACK] Seq=5 Ack=5 Win=29312 Len=5 TSval=2115279001 TSecr=1412476646
17	4.518140021	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=5 Ack=10 Win=29056 Len=0 TSval=1412476652 TSecr=2115279001
18	4.518216131	10.32.143.132	10.32.143.221	TCP	1514	52734 → 9876 [ACK] Seq=10 Ack=5 Win=29312 Len=1448 TSval=2115279001 TSecr=1412476646
19	4.518225695	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=5 Ack=1458 Win=31872 Len=0 TSval=1412476652 TSecr=2115279001
20	4.518275089	10.32.143.132	10.32.143.221	TCP	671	52734 → 9876 [PSH, ACK] Seq=1458 Ack=5 Win=29312 Len=605 TSval=2115279001 TSecr=1412476652
21	4.518283833	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=5 Ack=2063 Win=34816 Len=0 TSval=1412476652 TSecr=2115279001
22	4.518404401	10.32.143.132	10.32.143.221	TCP	1514	52734 → 9876 [ACK] Seq=2063 Ack=5 Win=29312 Len=1448 TSval=2115279001 TSecr=1412476652
23	4.518417929	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=5 Ack=3511 Win=37760 Len=0 TSval=1412476652 TSecr=2115279001
24	4.518421898	10.32.143.132	10.32.143.221	TCP	676	52734 → 9876 [PSH, ACK] Seq=3511 Ack=5 Win=29312 Len=610 TSval=2115279001 TSecr=1412476652
25	4.518427726	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=5 Ack=4121 Win=40576 Len=0 TSval=1412476652 TSecr=2115279001
26	4.518549386	10.32.143.132	10.32.143.221	TCP	1095	52734 → 9876 [PSH, ACK] Seq=4121 Ack=5 Win=29312 Len=1029 TSval=2115279001 TSecr=1412476652
27	4.518560601	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=5 Ack=5150 Win=43520 Len=0 TSval=1412476652 TSecr=2115279001
28	4.518676924	10.32.143.132	10.32.143.221	TCP	1514	52734 → 9876 [ACK] Seq=5150 Ack=5 Win=29312 Len=1448 TSval=2115279001 TSecr=1412476652
29	4.518688046	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=5 Ack=6598 Win=46336 Len=0 TSval=1412476652 TSecr=2115279001
30	4.518692346	10.32.143.132	10.32.143.221	TCP	676	52734 → 9876 [PSH, ACK] Seq=6598 Ack=5 Win=29312 Len=610 TSval=2115279001 TSecr=1412476652
31	4.518697906	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=5 Ack=7208 Win=49280 Len=0 TSval=1412476652 TSecr=2115279001
32	4.518815960	10.32.143.132	10.32.143.221	TCP	1095	52734 → 9876 [PSH, ACK] Seq=7208 Ack=5 Win=29312 Len=1029 TSval=2115279001 TSecr=1412476652
33	4.518827146	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=5 Ack=8237 Win=52224 Len=0 TSval=1412476653 TSecr=2115279001
34	4.518930486	10.32.143.132	10.32.143.221	TCP	1514	52734 → 9876 [ACK] Seq=8237 Ack=5 Win=29312 Len=1448 TSval=2115279002 TSecr=1412476652
35	4.518940289	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=5 Ack=9685 Win=55040 Len=0 TSval=1412476653 TSecr=2115279002
36	4.518944335	10.32.143.132	10.32.143.221	TCP	491	52734 → 9876 [PSH, ACK] Seq=9685 Ack=5 Win=29312 Len=425 TSval=2115279002 TSecr=1412476653
37	4.518950219	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=5 Ack=10110 Win=57984 Len=0 TSval=1412476653 TSecr=2115279002
38	4.522793150	10.32.143.221	10.32.143.132	TCP	71	9876 → 52734 [PSH, ACK] Seq=5 Ack=10110 Win=57984 Len=5 TSval=1412476657 TSecr=2115279002
39	4.522911551	10.32.143.221	10.32.143.132	TCP	1514	9876 → 52734 [ACK] Seq=10 Ack=10110 Win=57984 Len=1448 TSval=1412476657 TSecr=2115279002
40	4.522997292	10.32.143.221	10.32.143.132	TCP	1514	9876 → 52734 [ACK] Seq=1458 Ack=10110 Win=57984 Len=1448 TSval=1412476657 TSecr=2115279002
41	4.523059735	10.32.143.132	10.32.143.221	TCP	66	52734 → 9876 [ACK] Seq=10110 Ack=1458 Win=32128 Len=0 TSval=2115279006 TSecr=1412476657
42	4.523080613	10.32.143.221	10.32.143.132	TCP	252	9876 → 52734 [PSH, ACK] Seq=2906 Ack=10110 Win=57984 Len=186 TSval=1412476657 TSecr=2115279006
43	4.523170497	10.32.143.221	10.32.143.132	TCP	1514	9876 → 52734 [ACK] Seq=3092 Ack=10110 Win=57984 Len=1448 TSval=1412476657 TSecr=2115279006
44	4.523190114	10.32.143.132	10.32.143.221	TCP	66	52734 → 9876 [ACK] Seq=10110 Ack=3092 Win=37888 Len=0 TSval=2115279006 TSecr=1412476657
45	4.523199095	10.32.143.221	10.32.143.132	TCP	676	9876 → 52734 [PSH, ACK] Seq=4540 Ack=10110 Win=57984 Len=610 TSval=1412476657 TSecr=2115279006
46	4.523307195	10.32.143.132	10.32.143.221	TCP	66	52734 → 9876 [ACK] Seq=10110 Ack=5150 Win=43776 Len=0 TSval=2115279006 TSecr=1412476657
47	4.523315408	10.32.143.221	10.32.143.132	TCP	1095	9876 → 52734 [PSH, ACK] Seq=5150 Ack=10110 Win=57984 Len=1029 TSval=1412476657 TSecr=2115279006
48	4.523408438	10.32.143.221	10.32.143.132	TCP	1514	9876 → 52734 [ACK] Seq=6179 Ack=10110 Win=57984 Len=1448 TSval=1412476657 TSecr=2115279006
49	4.523496483	10.32.143.221	10.32.143.132	TCP	1514	9876 → 52734 [ACK] Seq=7627 Ack=10110 Win=57984 Len=1448 TSval=1412476657 TSecr=2115279006
50	4.523543100	10.32.143.132	10.32.143.221	TCP	66	52734 → 9876 [ACK] Seq=10110 Ack=7627 Win=49536 Len=0 TSval=2115279006 TSecr=1412476657
51	4.523552248	10.32.143.221	10.32.143.132	TCP	257	9876 → 52734 [PSH, ACK] Seq=9075 Ack=10110 Win=57984 Len=191 TSval=1412476657 TSecr=2115279006
52	4.523590569	10.32.143.221	10.32.143.132	TCP	931	9876 → 52734 [FIN, PSH, ACK] Seq=9266 Ack=10110 Win=57984 Len=865 TSval=1412476657 TSecr=2115279006
53	4.523651588	10.32.143.132	10.32.143.221	TCP	66	52734 → 9876 [ACK] Seq=10110 Ack=9266 Win=55296 Len=0 TSval=2115279006 TSecr=1412476657
54	4.525388964	10.32.143.132	10.32.143.221	TCP	66	52734 → 9876 [FIN, ACK] Seq=10110 Ack=10132 Win=58240 Len=0 TSval=2115279008 TSecr=1412476657
55	4.525426335	10.32.143.221	10.32.143.132	TCP	66	9876 → 52734 [ACK] Seq=10132 Ack=10111 Win=57984 Len=0 TSval=1412476659 TSecr=2115279008

Figura 4. Captura de pacote do envio de mensagem de 10Kb feita com socket TCP

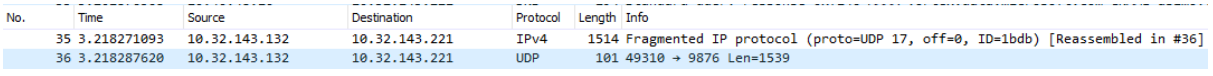
Por fim, como é possível observar, embora ambos protocolos trabalhem com dois sockets por meio de uma comunicação de cliente e servidor, eles têm se comportam de forma distinta. Como um é orientado a conexões, é possível extrair recursos interessantes para a administração do envio de pacotes, com um custo de desempenho. O outro, não sendo orientado a conexões, trabalha de forma mais assíncrona e proporciona uma independência de vinculações para a transmissão de pacotes, perdendo algumas funcionalidades de controle do envio de seus pacotes.

3. COMPORTAMENTO DOS SOCKETS COM ALTERAÇÕES NA REDE

Utilizando a ferramenta *netem*, nativa em máquinas Linux, foi possível simular um ambiente com interferências na rede durante o envio dos pacotes. Primeiramente, configuramos uma chance de 20% de um pacote ser perdido em seu envio e testamos os dois protocolos, enviando dois arquivos diferentes. Após isso, simulamos um ambiente com latência que variava de 200ms até 2000ms novamente com os dois protocolos e com os mesmos arquivos. Dessa forma, podendo visualizar através do Wireshark o comportamento do envio de pacotes de ambas aplicações diante dessas duas interferências na rede.

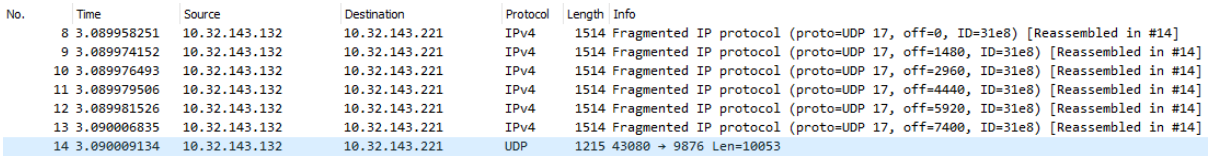
3.1. PERDA DE PACOTES

Assim, após estabelecemos na ferramenta *netem* a perda de pacotes de 20% e termos começado a captura dos pacotes pelo Wireshark novamente e rodamos a aplicação com UDP e pudemos visualizar o comportamento registrado nas seguintes figuras 5 e 6:



No.	Time	Source	Destination	Protocol	Length	Info
35	3.218271093	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=1bdb) [Reassembled in #36]
36	3.218287620	10.32.143.132	10.32.143.221	UDP	101	49310 → 9876 Len=1539

Figura 5. Captura de pacote do envio de mensagem de 1.5Kb feita com socket UDP e perda de pacotes de 20%

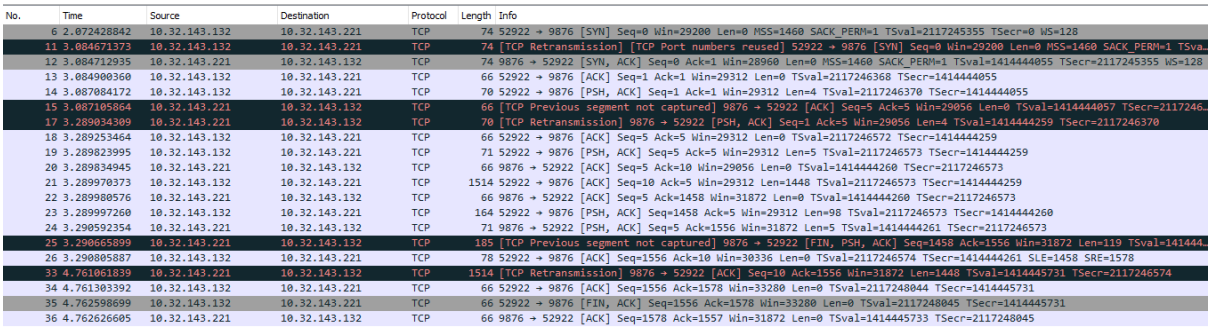


No.	Time	Source	Destination	Protocol	Length	Info
8	3.089958251	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=31e8) [Reassembled in #14]
9	3.089974152	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=1480, ID=31e8) [Reassembled in #14]
10	3.089976493	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=2960, ID=31e8) [Reassembled in #14]
11	3.089979506	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=4440, ID=31e8) [Reassembled in #14]
12	3.089981526	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=5920, ID=31e8) [Reassembled in #14]
13	3.090006835	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=7400, ID=31e8) [Reassembled in #14]
14	3.090009134	10.32.143.132	10.32.143.221	UDP	1215	43080 → 9876 Len=10053

Figura 6. Captura de pacote do envio de mensagem de 10Kb feita com socket UDP e perda de pacotes de 20%

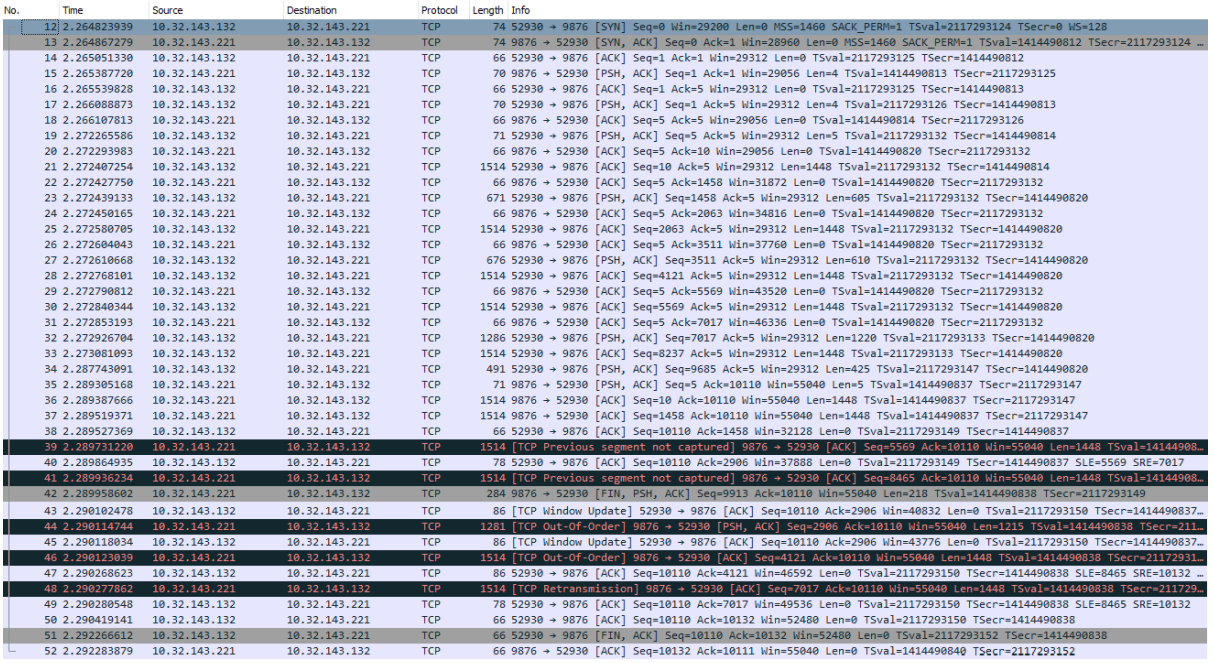
Como é possível observar, ambas entregas de pacote tiveram êxito no envio pelo cliente e recebimento pelo servidor. Embora houvesse 20% de chance de perda de algum pacote (que, neste caso, causaria perda total), a simulação de perda de pacotes não afetou negativamente a comunicação entre as duas máquinas. Todavia, vale ressaltar, que caso o pacote fosse perdido, isso significaria que nenhuma mensagem seria transmitida ao destinatário. Isso se pelo UDP não ser orientado a conexão e algumas características de controle não se fazerem presentes no envio da mensagem.

Em relação às imagens apresentadas acima, não houve uma diferença perceptível a respeito da perda de pacotes. Isso se dá pelo fato de que as chances de ocorrência de perda em somente um pacote são consideravelmente reduzidas se comparada ao outro protocolo. O protocolo TCP, diferentemente do UDP, como já foi visto, acaba utilizando múltiplos pacotes para o envio das mensagens. Logo, a possibilidade de perda desses pacotes acaba sendo maior. Entretanto, como o TCP possui um controle de seus pacotes que o UDP não tem, em caso de o pacote se perder ou duplicar, o protocolo tem como reenviar a informação perdida por meio de uma retransmissão, tornando a troca de dados íntegra. Abaixo nas figuras 7 e 8, é possível observar a perda de pacotes em ambos arquivos 1.5Kb e 10Kb, nas linhas em preto, visto que significam retransmissão, captura de segmentos anteriores e segmentos fora de ordem. O estabelecimento de conexão (Three-way handshake) está em cinza, e o envio de dados (juntamente com o ACK de confirmação), em lilás.



No.	Time	Source	Destination	Protocol	Length	Info
6	2.072428842	10.32.143.132	10.32.143.221	TCP	74	52922 → 9876 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=2117245355 TSecr=0 WS=128
11	3.084671373	10.32.143.132	10.32.143.221	TCP	74	[TCP Retransmission] [TCP Port numbers reused] 52922 → 9876 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSva…
12	3.084712935	10.32.143.221	10.32.143.132	TCP	74	9876 → 52922 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=1414444055 TSecr=2117245355 WS=128
13	3.084900360	10.32.143.132	10.32.143.221	TCP	66	52922 → 9876 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=2117246368 TSecr=1414444055
14	3.087084172	10.32.143.132	10.32.143.221	TCP	70	52922 → 9876 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=4 TSval=2117246370 TSecr=1414444055
15	3.087105864	10.32.143.221	10.32.143.132	TCP	66	[TCP Previous segment not captured] 9876 → 52922 [ACK] Seq=5 Ack=5 Win=29056 Len=0 TSval=1414444057 TSecr=2117246…
17	3.289034309	10.32.143.221	10.32.143.132	TCP	70	[TCP Retransmission] 9876 → 52922 [PSH, ACK] Seq=1 Ack=5 Win=29056 Len=4 TSval=1414444259 TSecr=2117246370
18	3.289253464	10.32.143.132	10.32.143.221	TCP	66	52922 → 9876 [ACK] Seq=5 Ack=5 Win=29312 Len=0 TSval=2117246572 TSecr=1414444259
19	3.289823995	10.32.143.132	10.32.143.221	TCP	71	52922 → 9876 [PSH, ACK] Seq=5 Ack=5 Win=29312 Len=5 TSval=2117246573 TSecr=1414444259
20	3.289834945	10.32.143.221	10.32.143.132	TCP	66	9876 → 52922 [ACK] Seq=5 Ack=10 Win=29056 Len=0 TSval=1414444260 TSecr=2117246573
21	3.289970373	10.32.143.132	10.32.143.221	TCP	1514	52922 → 9876 [ACK] Seq=10 Ack=5 Win=29312 Len=1448 TSval=2117246573 TSecr=1414444259
22	3.289980576	10.32.143.221	10.32.143.132	TCP	66	9876 → 52922 [ACK] Seq=5 Ack=1458 Win=31872 Len=0 TSval=1414444260 TSecr=2117246573
23	3.289997260	10.32.143.132	10.32.143.221	TCP	164	52922 → 9876 [PSH, ACK] Seq=1458 Ack=5 Win=29312 Len=98 TSval=2117246573 TSecr=1414444260
24	3.290592354	10.32.143.221	10.32.143.132	TCP	71	9876 → 52922 [PSH, ACK] Seq=5 Ack=1556 Win=31872 Len=5 TSval=1414444261 TSecr=2117246573
25	3.290665899	10.32.143.221	10.32.143.132	TCP	185	[TCP Previous segment not captured] 9876 → 52922 [FIN, PSH, ACK] Seq=1458 Ack=1556 Win=31872 Len=119 TSval=141444…
26	3.290805887	10.32.143.132	10.32.143.221	TCP	78	52922 → 9876 [ACK] Seq=1556 Ack=10 Win=30336 Len=0 TSval=2117246574 TSecr=1414444261 SLE=1458 SRE=1578
33	4.761061839	10.32.143.221	10.32.143.132	TCP	1514	[TCP Retransmission] 9876 → 52922 [ACK] Seq=10 Ack=1556 Win=31872 Len=1448 TSval=1414445731 TSecr=2117246574
34	4.761303392	10.32.143.132	10.32.143.221	TCP	66	52922 → 9876 [ACK] Seq=1556 Ack=1578 Win=33280 Len=0 TSval=2117248044 TSecr=1414445731
35	4.762598699	10.32.143.132	10.32.143.221	TCP	66	52922 → 9876 [FIN, ACK] Seq=1556 Ack=1578 Win=33280 Len=0 TSval=2117248045 TSecr=1414445731
36	4.762626605	10.32.143.221	10.32.143.132	TCP	66	9876 → 52922 [ACK] Seq=1578 Ack=1557 Win=31872 Len=0 TSval=1414445733 TSecr=2117248045

Figura 7. Captura de pacote do envio de mensagem de 1.5Kb feita com socket TCP e perda de pacotes de 20%

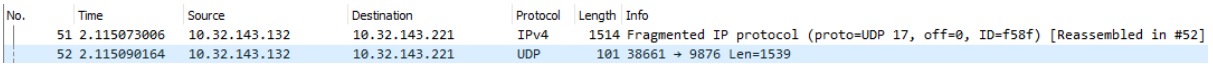


No.	Time	Source	Destination	Protocol	Length	Info
12	2.264823939	10.32.143.132	10.32.143.221	TCP	74	52930 → 9876 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=2117293124 TSecr=0 WS=128
13	2.264867279	10.32.143.221	10.32.143.132	TCP	74	9876 → 52930 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=1414490812 TSecr=2117293124 …
14	2.265051330	10.32.143.132	10.32.143.221	TCP	66	52930 → 9876 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=2117293125 TSecr=1414490812
15	2.265387720	10.32.143.221	10.32.143.132	TCP	70	9876 → 52930 [PSH, ACK] Seq=1 Ack=1 Win=29056 Len=4 TSval=1414490813 TSecr=2117293125
16	2.265539828	10.32.143.132	10.32.143.221	TCP	66	52930 → 9876 [ACK] Seq=1 Ack=5 Win=29312 Len=0 TSval=2117293125 TSecr=1414490813
17	2.266088873	10.32.143.132	10.32.143.221	TCP	70	52930 → 9876 [PSH, ACK] Seq=1 Ack=5 Win=29312 Len=4 TSval=2117293126 TSecr=1414490813
18	2.266107813	10.32.143.221	10.32.143.132	TCP	66	9876 → 52930 [ACK] Seq=5 Ack=5 Win=29056 Len=0 TSval=1414490814 TSecr=2117293126
19	2.272265586	10.32.143.132	10.32.143.221	TCP	71	52930 → 9876 [PSH, ACK] Seq=5 Ack=5 Win=29312 Len=5 TSval=2117293132 TSecr=1414490814
20	2.272293983	10.32.143.221	10.32.143.132	TCP	66	9876 → 52930 [ACK] Seq=5 Ack=10 Win=29056 Len=0 TSval=1414490820 TSecr=2117293132
21	2.272407254	10.32.143.132	10.32.143.221	TCP	1514	52930 → 9876 [ACK] Seq=10 Ack=5 Win=29312 Len=1448 TSval=2117293132 TSecr=1414490814
22	2.272427750	10.32.143.221	10.32.143.132	TCP	66	9876 → 52930 [ACK] Seq=5 Ack=1458 Win=31872 Len=0 TSval=1414490820 TSecr=2117293132
23	2.272439133	10.32.143.132	10.32.143.221	TCP	671	52930 → 9876 [PSH, ACK] Seq=1458 Ack=5 Win=29312 Len=605 TSval=2117293132 TSecr=1414490820
24	2.272450165	10.32.143.221	10.32.143.132	TCP	66	9876 → 52930 [ACK] Seq=5 Ack=2063 Win=34816 Len=0 TSval=1414490820 TSecr=2117293132
25	2.272580705	10.32.143.132	10.32.143.221	TCP	1514	52930 → 9876 [ACK] Seq=2063 Ack=5 Win=29312 Len=1448 TSval=2117293132 TSecr=1414490820
26	2.272604043	10.32.143.221	10.32.143.132	TCP	66	9876 → 52930 [ACK] Seq=5 Ack=3511 Win=37760 Len=0 TSval=1414490820 TSecr=2117293132
27	2.272610668	10.32.143.132	10.32.143.221	TCP	676	52930 → 9876 [PSH, ACK] Seq=3511 Ack=5 Win=29312 Len=610 TSval=2117293132 TSecr=1414490820
28	2.272768101	10.32.143.132	10.32.143.221	TCP	1514	52930 → 9876 [ACK] Seq=4121 Ack=5 Win=29312 Len=1448 TSval=2117293132 TSecr=1414490820
29	2.272790812	10.32.143.221	10.32.143.132	TCP	66	9876 → 52930 [ACK] Seq=5 Ack=5569 Win=43520 Len=0 TSval=1414490820 TSecr=2117293132
30	2.272840344	10.32.143.132	10.32.143.221	TCP	1514	52930 → 9876 [ACK] Seq=5569 Ack=5 Win=29312 Len=1448 TSval=2117293132 TSecr=1414490820
31	2.272853193	10.32.143.221	10.32.143.132	TCP	66	9876 → 52930 [ACK] Seq=5 Ack=7017 Win=46336 Len=0 TSval=1414490820 TSecr=2117293132
32	2.272926704	10.32.143.132	10.32.143.221	TCP	1286	52930 → 9876 [PSH, ACK] Seq=7017 Ack=5 Win=29312 Len=1220 TSval=2117293133 TSecr=1414490820
33	2.273081093	10.32.143.132	10.32.143.221	TCP	1514	52930 → 9876 [ACK] Seq=8237 Ack=5 Win=29312 Len=1448 TSval=2117293133 TSecr=1414490820
34	2.287743091	10.32.143.132	10.32.143.221	TCP	491	52930 → 9876 [PSH, ACK] Seq=9685 Ack=5 Win=29312 Len=425 TSval=2117293147 TSecr=1414490820
35	2.289305168	10.32.143.221	10.32.143.132	TCP	71	9876 → 52930 [PSH, ACK] Seq=5 Ack=10110 Win=55040 Len=5 TSval=1414490837 TSecr=2117293147
36	2.289387666	10.32.143.221	10.32.143.132	TCP	1514	9876 → 52930 [ACK] Seq=10 Ack=10110 Win=55040 Len=1448 TSval=1414490837 TSecr=2117293147
37	2.289519371	10.32.143.221	10.32.143.132	TCP	1514	9876 → 52930 [ACK] Seq=1458 Ack=10110 Win=55040 Len=1448 TSval=1414490837 TSecr=2117293147
38	2.289527369	10.32.143.132	10.32.143.221	TCP	66	52930 → 9876 [ACK] Seq=10110 Ack=1458 Win=32128 Len=0 TSval=2117293149 TSecr=1414490837
39	2.289731220	10.32.143.221	10.32.143.132	TCP	1514	[TCP Previous segment not captured] 9876 → 52930 [ACK] Seq=5569 Ack=10110 Win=55040 Len=1448 TSval=14144908…
40	2.289864935	10.32.143.132	10.32.143.221	TCP	78	52930 → 9876 [ACK] Seq=10110 Ack=2906 Win=37888 Len=0 TSval=2117293149 TSecr=1414490837 SLE=5569 SRE=7017
41	2.289936234	10.32.143.221	10.32.143.132	TCP	1514	[TCP Previous segment not captured] 9876 → 52930 [ACK] Seq=8465 Ack=10110 Win=55040 Len=1448 TSval=14144908…
42	2.289958602	10.32.143.221	10.32.143.132	TCP	284	9876 → 52930 [FIN, PSH, ACK] Seq=9913 Ack=10110 Win=55040 Len=218 TSval=1414490838 TSecr=2117293149
43	2.290102478	10.32.143.132	10.32.143.221	TCP	86	[TCP Window Update] 52930 → 9876 [ACK] Seq=10110 Ack=2906 Win=40832 Len=0 TSval=2117293150 TSecr=1414490837…
44	2.290114744	10.32.143.221	10.32.143.132	TCP	1281	[TCP Out-Of-Order] 9876 → 52930 [PSH, ACK] Seq=2906 Ack=10110 Win=55040 Len=1215 TSval=1414490838 TSecr=211…
45	2.290118034	10.32.143.132	10.32.143.221	TCP	86	[TCP Window Update] 52930 → 9876 [ACK] Seq=10110 Ack=2906 Win=43776 Len=0 TSval=2117293150 TSecr=1414490837…
46	2.290123039	10.32.143.221	10.32.143.132	TCP	1514	[TCP Out-Of-Order] 9876 → 52930 [ACK] Seq=4121 Ack=10110 Win=55040 Len=1448 TSval=1414490838 TSecr=21172931…
47	2.290268623	10.32.143.132	10.32.143.221	TCP	86	52930 → 9876 [ACK] Seq=10110 Ack=4121 Win=46592 Len=0 TSval=2117293150 TSecr=1414490838 SLE=8465 SRE=10132 …
48	2.290277862	10.32.143.221	10.32.143.132	TCP	1514	[TCP Retransmission] 9876 → 52930 [ACK] Seq=7017 Ack=10110 Win=55040 Len=1448 TSval=1414490838 TSecr=211729…
49	2.290280548	10.32.143.132	10.32.143.221	TCP	78	52930 → 9876 [ACK] Seq=10110 Ack=7017 Win=49536 Len=0 TSval=2117293150 TSecr=1414490838 SLE=8465 SRE=10132
50	2.290419141	10.32.143.132	10.32.143.221	TCP	66	52930 → 9876 [ACK] Seq=10110 Ack=10132 Win=52480 Len=0 TSval=2117293150 TSecr=1414490838
51	2.292266612	10.32.143.132	10.32.143.221	TCP	66	52930 → 9876 [FIN, ACK] Seq=10110 Ack=10132 Win=52480 Len=0 TSval=2117293152 TSecr=1414490838
52	2.292283879	10.32.143.221	10.32.143.132	TCP	66	9876 → 52930 [ACK] Seq=10132 Ack=10111 Win=55040 Len=0 TSval=1414490840 TSecr=2117293152

Figura 8. Captura de pacote do envio de mensagem de 10Kb feita com socket TCP e perda de pacotes de 20%

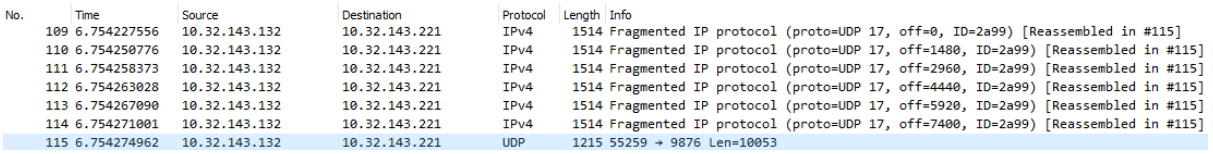
3.2. LATÊNCIA (ATRASO) VARIÁVEL

Para o segundo caso de análise de comportamento dos sockets com interferências na rede, foi decidido variar a latência entre 200ms e 2000ms. Com isso, obtivemos os seguintes resultados para a aplicação usando socket UDP que estão apresentados nas figuras 9 e 10:



No.	Time	Source	Destination	Protocol	Length	Info
51	2.115073006	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=f58f) [Reassembled in #52]
52	2.115090164	10.32.143.132	10.32.143.221	UDP	101	38661 → 9876 Len=1539

Figura 9. Captura de pacote do envio de mensagem de 1.5Kb feita com socket UDP e latência (atraso) variável de 200ms a 2000ms



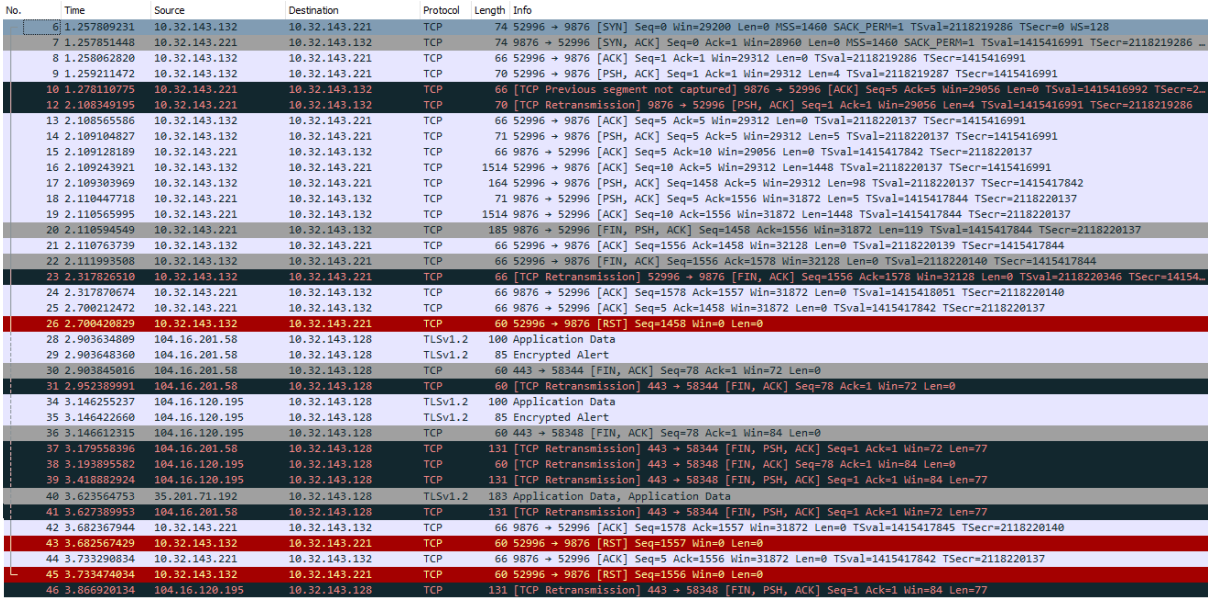
No.	Time	Source	Destination	Protocol	Length	Info
109	6.754227556	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=2a99) [Reassembled in #115]
110	6.754250776	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=1480, ID=2a99) [Reassembled in #115]
111	6.754258373	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=2960, ID=2a99) [Reassembled in #115]
112	6.754263028	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=4440, ID=2a99) [Reassembled in #115]
113	6.754267090	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=5920, ID=2a99) [Reassembled in #115]
114	6.754271001	10.32.143.132	10.32.143.221	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=7400, ID=2a99) [Reassembled in #115]
115	6.754274962	10.32.143.132	10.32.143.221	UDP	1215	55259 → 9876 Len=10053

Figura 10. Captura de pacote do envio de mensagem de 10Kb feita com socket UDP e latência (atraso) variável de 200ms a 2000ms

No caso do primeiro arquivo, se percebeu muito pouca demora no recebimento pelo destino e até mesmo que em casos das seções anteriores, porém o que foi mais perceptível essa demora foi no segundo em que o tempo de envio chegou a quase dobrar. Logo, pelo fato de que a aplicação é UDP, podemos ressaltar o fato de não ter ocorrido nenhuma retransmissão por conta do atraso e nem problemas na ordem de entrega de pacotes, visto que só temos o envio de um pacote, porém sempre lembrando que se o mesmo se perdesse o protocolo não tomaria medidas como o TCP de retransmitir-lo como vimos na seção onde comparamos os dois.

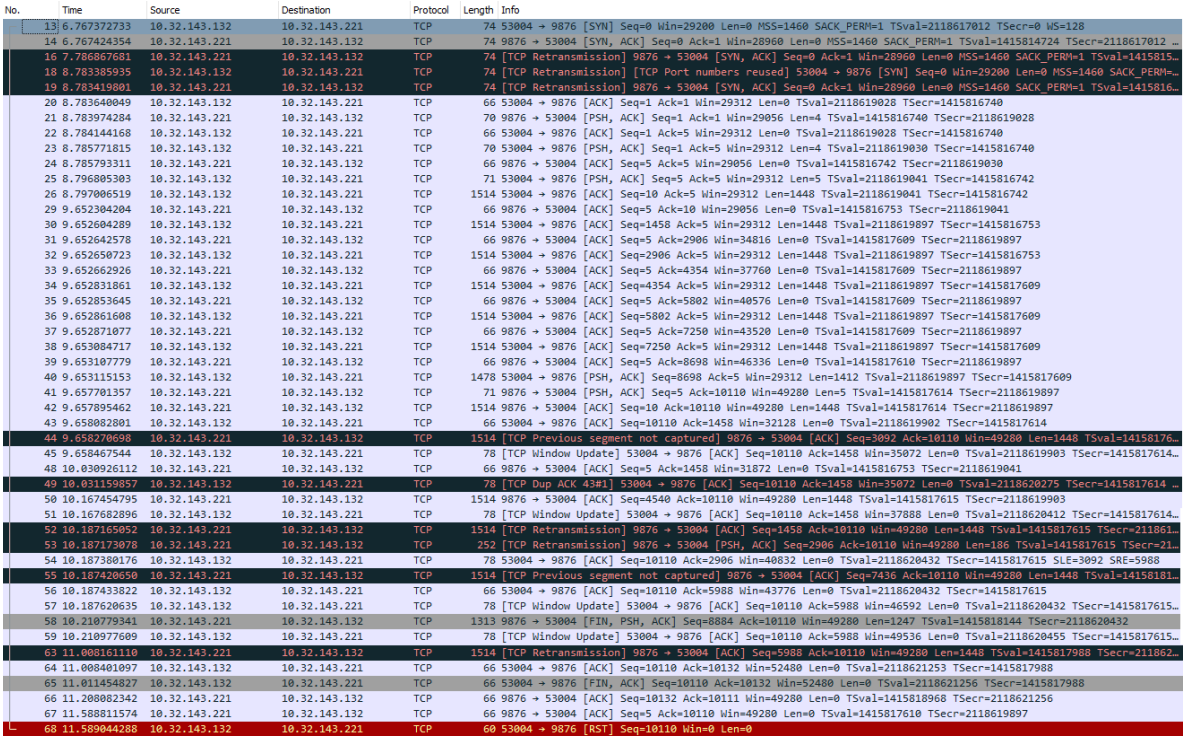
Já no protocolo TCP, devido à latência variável, o Wireshark mostrou tratamentos de erro pontuais, onde houve o tratamento de pacotes. Embora a integralidade dos arquivos tenha sido respeitada, houveram dois casos a serem pontuados. O primeiro, e mais frequente foi a retransmissão de pacotes, que impediu qualquer perda no processo de envio enquanto o Three-way handshake teve-se válido. Outro caso é a ocorrência da flag RST (Reset de TCP) representado por uma linha vermelha na captura do Wireshark. Ele pode ocorrer em dois momentos: quando há o envio de SYN e não há nenhum processo esperando o estabelecimento de conexão na porta do servidor, ou, neste caso, quando é enviado um pacote após o término de uma conexão em que o socket já foi fechado.

Abaixo, é possível conferir os envios, reenvios e tratamento de erros sobre os processos gerais do protocolo TCP com latência variável. Vale salientar que alguns processos externos no momento da captura estavam presentes durante o tempo de verificação dos pacotes das aplicações. Logo, é aconselhável checar os IPs de origem e destinos a serem considerados que foram mencionados na introdução, mas que estão também disponíveis nas figuras abaixo.



No.	Time	Source	Destination	Protocol	Length	Info
6	1.257809231	10.32.143.132	10.32.143.221	TCP	74	52996 → 9876 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=2118219286 TSecr=0 WS=128
7	1.257851448	10.32.143.221	10.32.143.132	TCP	74	9876 → 52996 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=1415416991 TSecr=2118219286 …
8	1.258062820	10.32.143.132	10.32.143.221	TCP	66	52996 → 9876 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=2118219286 TSecr=1415416991
9	1.259211472	10.32.143.132	10.32.143.221	TCP	70	52996 → 9876 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=4 TSval=2118219287 TSecr=1415416991
10	1.278110775	10.32.143.221	10.32.143.132	TCP	66	[TCP Previous segment not captured] 9876 → 52996 [ACK] Seq=5 Ack=5 Win=29056 Len=0 TSval=1415416992 TSecr=2…
12	2.108349195	10.32.143.221	10.32.143.132	TCP	70	[TCP Retransmission] 9876 → 52996 [PSH, ACK] Seq=1 Ack=1 Win=29056 Len=4 TSval=1415416991 TSecr=2118219286
13	2.108565586	10.32.143.132	10.32.143.221	TCP	66	52996 → 9876 [ACK] Seq=5 Ack=5 Win=29312 Len=0 TSval=2118220137 TSecr=1415416991
14	2.109104827	10.32.143.132	10.32.143.221	TCP	71	52996 → 9876 [PSH, ACK] Seq=5 Ack=5 Win=29312 Len=5 TSval=2118220137 TSecr=1415416991
15	2.109128189	10.32.143.221	10.32.143.132	TCP	66	9876 → 52996 [ACK] Seq=5 Ack=10 Win=29056 Len=0 TSval=1415417842 TSecr=2118220137
16	2.109243921	10.32.143.132	10.32.143.221	TCP	1514	52996 → 9876 [ACK] Seq=10 Ack=5 Win=29312 Len=1448 TSval=2118220137 TSecr=1415416991
17	2.109303969	10.32.143.132	10.32.143.221	TCP	164	52996 → 9876 [PSH, ACK] Seq=1458 Ack=5 Win=29312 Len=98 TSval=2118220137 TSecr=1415417842
18	2.110447718	10.32.143.221	10.32.143.132	TCP	71	9876 → 52996 [PSH, ACK] Seq=5 Ack=1556 Win=31872 Len=5 TSval=1415417844 TSecr=2118220137
19	2.110565995	10.32.143.221	10.32.143.132	TCP	1514	9876 → 52996 [ACK] Seq=10 Ack=1556 Win=31872 Len=1448 TSval=1415417844 TSecr=2118220137
20	2.110594549	10.32.143.221	10.32.143.132	TCP	185	9876 → 52996 [FIN, PSH, ACK] Seq=1458 Ack=1556 Win=31872 Len=119 TSval=1415417844 TSecr=2118220137
21	2.110763739	10.32.143.132	10.32.143.221	TCP	66	52996 → 9876 [ACK] Seq=1556 Ack=1458 Win=32128 Len=0 TSval=2118220139 TSecr=1415417844
22	2.111993508	10.32.143.132	10.32.143.221	TCP	66	52996 → 9876 [FIN, ACK] Seq=1556 Ack=1578 Win=32128 Len=0 TSval=2118220140 TSecr=1415417844
23	2.317826510	10.32.143.132	10.32.143.221	TCP	66	[TCP Retransmission] 52996 → 9876 [FIN, ACK] Seq=1556 Ack=1578 Win=32128 Len=0 TSval=2118220346 TSecr=14154…
24	2.317870674	10.32.143.221	10.32.143.132	TCP	66	9876 → 52996 [ACK] Seq=1578 Ack=1557 Win=31872 Len=0 TSval=1415418051 TSecr=2118220140
25	2.700212472	10.32.143.221	10.32.143.132	TCP	66	9876 → 52996 [ACK] Seq=5 Ack=1458 Win=31872 Len=0 TSval=1415417842 TSecr=2118220137
26	2.700420829	10.32.143.132	10.32.143.221	TCP	60	52996 → 9876 [RST] Seq=1458 Win=0 Len=0
28	2.903634809	104.16.201.58	10.32.143.128	TLSv1.2	100	Application Data
29	2.903648360	104.16.201.58	10.32.143.128	TLSv1.2	85	Encrypted Alert
30	2.903845016	104.16.201.58	10.32.143.128	TCP	60	443 → 58344 [FIN, ACK] Seq=78 Ack=1 Win=72 Len=0
31	2.952389991	104.16.201.58	10.32.143.128	TCP	60	[TCP Retransmission] 443 → 58344 [FIN, ACK] Seq=78 Ack=1 Win=72 Len=0
34	3.146255237	104.16.120.195	10.32.143.128	TLSv1.2	100	Application Data
35	3.146422660	104.16.120.195	10.32.143.128	TLSv1.2	85	Encrypted Alert
36	3.146612315	104.16.120.195	10.32.143.128	TCP	60	443 → 58348 [FIN, ACK] Seq=78 Ack=1 Win=84 Len=0
37	3.179558396	104.16.201.58	10.32.143.128	TCP	131	[TCP Retransmission] 443 → 58344 [FIN, PSH, ACK] Seq=1 Ack=1 Win=72 Len=77
38	3.193895582	104.16.120.195	10.32.143.128	TCP	60	[TCP Retransmission] 443 → 58348 [FIN, ACK] Seq=78 Ack=1 Win=84 Len=0
39	3.418882924	104.16.120.195	10.32.143.128	TCP	131	[TCP Retransmission] 443 → 58348 [FIN, PSH, ACK] Seq=1 Ack=1 Win=84 Len=77
40	3.623564753	35.201.71.192	10.32.143.128	TLSv1.2	183	Application Data, Application Data
41	3.627389953	104.16.201.58	10.32.143.128	TCP	131	[TCP Retransmission] 443 → 58344 [FIN, PSH, ACK] Seq=1 Ack=1 Win=72 Len=77
42	3.682367944	10.32.143.221	10.32.143.132	TCP	66	9876 → 52996 [ACK] Seq=1578 Ack=1557 Win=31872 Len=0 TSval=1415417845 TSecr=2118220140
43	3.682567429	10.32.143.132	10.32.143.221	TCP	60	52996 → 9876 [RST] Seq=1557 Win=0 Len=0
44	3.733290834	10.32.143.221	10.32.143.132	TCP	66	9876 → 52996 [ACK] Seq=5 Ack=1556 Win=31872 Len=0 TSval=1415417842 TSecr=2118220137
45	3.733474034	10.32.143.132	10.32.143.221	TCP	60	52996 → 9876 [RST] Seq=1556 Win=0 Len=0
46	3.866920134	104.16.120.195	10.32.143.128	TCP	131	[TCP Retransmission] 443 → 58348 [FIN, PSH, ACK] Seq=1 Ack=1 Win=84 Len=77

Figura 11. Captura de pacote do envio de mensagem de 1.5Kb feita com socket TCP e latência (atraso) variável de 200ms a 2000ms



No.	Time	Source	Destination	Protocol	Length	Info
13	6.767372733	10.32.143.132	10.32.143.221	TCP	74	53004 → 9876 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=2118617012 TSecr=0 WS=128
14	6.767424354	10.32.143.221	10.32.143.132	TCP	74	9876 → 53004 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=1415814724 TSecr=2118617012 …
16	7.786867681	10.32.143.221	10.32.143.132	TCP	74	[TCP Retransmission] 9876 → 53004 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=1415815…
18	8.783385935	10.32.143.132	10.32.143.221	TCP	74	[TCP Retransmission] [TCP Port numbers reused] 53004 → 9876 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=…
19	8.783419801	10.32.143.221	10.32.143.132	TCP	74	[TCP Retransmission] 9876 → 53004 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=1415816…
20	8.783640049	10.32.143.132	10.32.143.221	TCP	66	53004 → 9876 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=2118619028 TSecr=1415816740
21	8.783974284	10.32.143.221	10.32.143.132	TCP	70	9876 → 53004 [PSH, ACK] Seq=1 Ack=1 Win=29056 Len=4 TSval=1415816740 TSecr=2118619028
22	8.784144168	10.32.143.132	10.32.143.221	TCP	66	53004 → 9876 [ACK] Seq=1 Ack=5 Win=29312 Len=0 TSval=2118619028 TSecr=1415816740
23	8.785771815	10.32.143.132	10.32.143.221	TCP	70	53004 → 9876 [PSH, ACK] Seq=1 Ack=5 Win=29312 Len=4 TSval=2118619030 TSecr=1415816740
24	8.785793311	10.32.143.221	10.32.143.132	TCP	66	9876 → 53004 [ACK] Seq=5 Ack=5 Win=29056 Len=0 TSval=1415816742 TSecr=2118619030
25	8.796805303	10.32.143.132	10.32.143.221	TCP	71	53004 → 9876 [PSH, ACK] Seq=5 Ack=5 Win=29312 Len=5 TSval=2118619041 TSecr=1415816742
26	8.797006519	10.32.143.132	10.32.143.221	TCP	1514	53004 → 9876 [ACK] Seq=10 Ack=5 Win=29312 Len=1448 TSval=2118619041 TSecr=1415816742
29	9.652304204	10.32.143.221	10.32.143.132	TCP	66	9876 → 53004 [ACK] Seq=5 Ack=10 Win=29056 Len=0 TSval=1415816753 TSecr=2118619041
30	9.652604289	10.32.143.132	10.32.143.221	TCP	1514	53004 → 9876 [ACK] Seq=1458 Ack=5 Win=29312 Len=1448 TSval=2118619897 TSecr=1415816753
31	9.652642578	10.32.143.221	10.32.143.132	TCP	66	9876 → 53004 [ACK] Seq=5 Ack=2906 Win=34816 Len=0 TSval=1415817609 TSecr=2118619897
32	9.652650723	10.32.143.132	10.32.143.221	TCP	1514	53004 → 9876 [ACK] Seq=2906 Ack=5 Win=29312 Len=1448 TSval=2118619897 TSecr=1415816753
33	9.652662926	10.32.143.221	10.32.143.132	TCP	66	9876 → 53004 [ACK] Seq=5 Ack=4354 Win=37760 Len=0 TSval=1415817609 TSecr=2118619897
34	9.652831861	10.32.143.132	10.32.143.221	TCP	1514	53004 → 9876 [ACK] Seq=4354 Ack=5 Win=29312 Len=1448 TSval=2118619897 TSecr=1415817609
35	9.652853645	10.32.143.221	10.32.143.132	TCP	66	9876 → 53004 [ACK] Seq=5 Ack=5802 Win=40576 Len=0 TSval=1415817609 TSecr=2118619897
36	9.652861608	10.32.143.132	10.32.143.221	TCP	1514	53004 → 9876 [ACK] Seq=5802 Ack=5 Win=29312 Len=1448 TSval=2118619897 TSecr=1415817609
37	9.652871077	10.32.143.221	10.32.143.132	TCP	66	9876 → 53004 [ACK] Seq=5 Ack=7250 Win=43520 Len=0 TSval=1415817609 TSecr=2118619897
38	9.653084717	10.32.143.132	10.32.143.221	TCP	1514	53004 → 9876 [ACK] Seq=7250 Ack=5 Win=29312 Len=1448 TSval=2118619897 TSecr=1415817609
39	9.653107779	10.32.143.221	10.32.143.132	TCP	66	9876 → 53004 [ACK] Seq=5 Ack=8698 Win=46336 Len=0 TSval=1415817610 TSecr=2118619897
40	9.653115153	10.32.143.132	10.32.143.221	TCP	1478	53004 → 9876 [PSH, ACK] Seq=8698 Ack=5 Win=29312 Len=1412 TSval=2118619897 TSecr=1415817609
41	9.657701357	10.32.143.221	10.32.143.132	TCP	71	9876 → 53004 [PSH, ACK] Seq=5 Ack=10110 Win=49280 Len=5 TSval=1415817614 TSecr=2118619897
42	9.657895462	10.32.143.221	10.32.143.132	TCP	1514	9876 → 53004 [ACK] Seq=10 Ack=10110 Win=49280 Len=1448 TSval=1415817614 TSecr=2118619897
43	9.658082801	10.32.143.132	10.32.143.221	TCP	66	53004 → 9876 [ACK] Seq=10110 Ack=1458 Win=32128 Len=0 TSval=2118619902 TSecr=1415817614
44	9.658270698	10.32.143.221	10.32.143.132	TCP	1514	[TCP Previous segment not captured] 9876 → 53004 [ACK] Seq=3092 Ack=10110 Win=49280 Len=1448 TSval=14158176…
45	9.658467544	10.32.143.132	10.32.143.221	TCP	78	[TCP Window Update] 53004 → 9876 [ACK] Seq=10110 Ack=1458 Win=35072 Len=0 TSval=2118619903 TSecr=1415817614…
48	10.030926112	10.32.143.221	10.32.143.132	TCP	66	9876 → 53004 [ACK] Seq=5 Ack=1458 Win=31872 Len=0 TSval=1415816753 TSecr=2118619041
49	10.031159857	10.32.143.132	10.32.143.221	TCP	78	[TCP Dup ACK 43#1] 53004 → 9876 [ACK] Seq=10110 Ack=1458 Win=35072 Len=0 TSval=2118620275 TSecr=1415817614 …
50	10.167454795	10.32.143.221	10.32.143.132	TCP	1514	9876 → 53004 [ACK] Seq=4540 Ack=10110 Win=49280 Len=1448 TSval=1415817615 TSecr=2118619903
51	10.167682896	10.32.143.132	10.32.143.221	TCP	78	[TCP Window Update] 53004 → 9876 [ACK] Seq=10110 Ack=1458 Win=37888 Len=0 TSval=2118620412 TSecr=1415817614…
52	10.187165052	10.32.143.221	10.32.143.132	TCP	1514	[TCP Retransmission] 9876 → 53004 [ACK] Seq=1458 Ack=10110 Win=49280 Len=1448 TSval=1415817615 TSecr=211861…
53	10.187173078	10.32.143.221	10.32.143.132	TCP	252	[TCP Retransmission] 9876 → 53004 [PSH, ACK] Seq=2906 Ack=10110 Win=49280 Len=186 TSval=1415817615 TSecr=21…
54	10.187380176	10.32.143.132	10.32.143.221	TCP	78	53004 → 9876 [ACK] Seq=10110 Ack=2906 Win=40832 Len=0 TSval=2118620432 TSecr=1415817615 SLE=3092 SRE=5988
55	10.187420650	10.32.143.221	10.32.143.132	TCP	1514	[TCP Previous segment not captured] 9876 → 53004 [ACK] Seq=7436 Ack=10110 Win=49280 Len=1448 TSval=14158181…
56	10.187433822	10.32.143.132	10.32.143.221	TCP	66	53004 → 9876 [ACK] Seq=10110 Ack=5988 Win=43776 Len=0 TSval=2118620432 TSecr=1415817615
57	10.187620635	10.32.143.132	10.32.143.221	TCP	78	[TCP Window Update] 53004 → 9876 [ACK] Seq=10110 Ack=5988 Win=46592 Len=0 TSval=2118620432 TSecr=1415817615…
58	10.210779341	10.32.143.221	10.32.143.132	TCP	1313	9876 → 53004 [FIN, PSH, ACK] Seq=8884 Ack=10110 Win=49280 Len=1247 TSval=1415818144 TSecr=2118620432
59	10.210977609	10.32.143.132	10.32.143.221	TCP	78	[TCP Window Update] 53004 → 9876 [ACK] Seq=10110 Ack=5988 Win=49536 Len=0 TSval=2118620455 TSecr=1415817615…
63	11.008161110	10.32.143.221	10.32.143.132	TCP	1514	[TCP Retransmission] 9876 → 53004 [ACK] Seq=5988 Ack=10110 Win=49280 Len=1448 TSval=1415817988 TSecr=211862…
64	11.008401097	10.32.143.132	10.32.143.221	TCP	66	53004 → 9876 [ACK] Seq=10110 Ack=10132 Win=52480 Len=0 TSval=2118621253 TSecr=1415817988
65	11.011454827	10.32.143.132	10.32.143.221	TCP	66	53004 → 9876 [FIN, ACK] Seq=10110 Ack=10132 Win=52480 Len=0 TSval=2118621256 TSecr=1415817988
66	11.208082342	10.32.143.221	10.32.143.132	TCP	66	9876 → 53004 [ACK] Seq=10132 Ack=10111 Win=49280 Len=0 TSval=1415818968 TSecr=2118621256
67	11.588811574	10.32.143.221	10.32.143.132	TCP	66	9876 → 53004 [ACK] Seq=5 Ack=10110 Win=49280 Len=0 TSval=1415817610 TSecr=2118619897
68	11.589044288	10.32.143.132	10.32.143.221	TCP	60	53004 → 9876 [RST] Seq=10110 Win=0 Len=0

Figura 12. Captura de pacote do envio de mensagem de 10Kb feita com socket TCP e latência (atraso) variável de 200ms a 2000ms