

Лабораторная работа 2. Исследование протокола TCP и алгоритма управления очередью RED

Ильин Никита Евгеньевич

Содержание

1	Цель работы	5
2	Ход работы	6
3	Выводы	14
4	Библиография	15

List of Figures

2.1	Рис 1. Схема сети	6
2.2	Рис 2. Код программы(1)	7
2.3	Рис 3. Код программы(2)	8
2.4	Рис 4. Код программы(3)	9
2.5	Рис 5. График №1 - Динамика размера окна ТСР	10
2.6	Рис 6. График №2 - Динамика очереди и средней длины очереди .	11
2.7	Рис 7. Графики для каждого протокола	12
2.8	Рис 8. График с измененными характеристиками отображения . .	13

List of Tables

1 Цель работы

Научиться разрабатывать сценарий, реализующий модель сети. Исследовать протокол TCP и алгоритм управления очередью RED.

2 Ход работы

1. Требуется разработать сценарий, реализующий модель согласно рис. 2.4, по- строить в Xgraph график изменения ТСР-окна, график изменения дли- ны очереди и средней длины очереди.

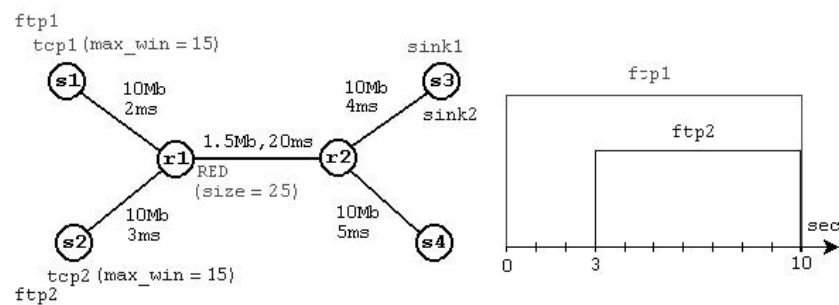


Figure 2.1: Рис 1. Схема сети

2. Копирую содержимое файла shablon.tcl(создан при выполнении lab01) в файл lab02_1.tcl. Добавляю в файл код, предоставленный в задании к лаб. работе.

```

1 # создание объекта Simulator
2 set ns [new Simulator]
3 # открытие на запись файла out.nam для визуализатора nam
4 set nf [open out.nam w]
5 # все результаты моделирования будут записаны в переменную nf
6 $ns namtrace-all $nf
7 # открытие на запись файла трассировки out.tr
8 # для регистрации всех событий
9 set f [open out.tr w]
10 # все регистрируемые события будут записаны в переменную f
11 $ns trace-all $f
12 # процедура finish закрывает файлы трассировки
13 # и запускает визуализатор nam
14 proc finish {} {
15     global ns f nf # описание глобальных переменных
16     $ns flush-trace # прекращение трассировки
17     close $f # закрытие файлов трассировки
18     close $nf # закрытие файлов трассировки nam
19     # запуск nam в фоновом режиме
20     exec nam out.nam &
21     exit 0
22 }
23 # Узлы сети:
24 set N 5
25 for {set i 1} {$i < $N} {incr i} {
26     set node_($i) [$ns node]
27 }
28 set node_(r1) [$ns node]
29 set node_(r2) [$ns node]
30
31 # Соединения:
32 $ns duplex-link $node_(s1) $node_(r1) 10Mb 2ms DropTail
33 $ns duplex-link $node_(s2) $node_(r1) 10Mb 3ms DropTail
34 $ns duplex-link $node_(r1) $node_(r2) 1.5Mb 20ms RED
35 $ns queue-limit $node_(r1) $node_(r2) 25
36 $ns queue-limit $node_(r2) $node_(r1) 25
37 $ns duplex-link $node_(s3) $node_(r2) 10Mb 4ms DropTail
38 $ns duplex-link $node_(s4) $node_(r2) 10Mb 5ms DropTail
39
40 # Агенты и приложения:
41 set tcp1 [$ns create-connection TCP/Reno $node_(s1) TCPSink $node_(s3) 0]
42 $tcp1 set window_ 15
43 set tcp2 [$ns create-connection TCP/Reno $node_(s2) TCPSink $node_(s3) 1]
44 $tcp2 set window_ 15
45 set ftp1 [$tcp1 attach-source FTP]
46 set ftp2 [$tcp2 attach-source FTP]
47
48 # Мониторинг размера окна TCP:
49 set windowVsTime [open WindowVsTimeReno w]
50 set qmon [$ns monitor-queue $node_(r1) $node_(r2) [open qm.out w] 0.1];
51 [$ns link $node_(r1) $node_(r2)] queue-sample-timeout;
52

```

Figure 2.2: Рис 2. Код программы(1)

```

52
53 # Мониторинг очереди:
54 set redq [[ $ns link $node_(r1) $node_(r2)] queue]
55 set tchan_ [open all.q w]
56 $redq trace curq_
57 $redq trace ave_
58 $redq attach $tchan_
59
60 # Добавление at-событий:
61 $ns at 0.0 "$ftp1 start"
62 $ns at 1.1 "plotWindow $tcp1 $windowVsTime"
63 $ns at 3.0 "$ftp2 start"
64 $ns at 10 "finish"
65
66 # Формирование файла с данными о размере окна TCP:
67 proc plotWindow {tcpSource file} {
68     global ns
69     set time 0.01
70     set now [$ns now]
71     set cwnd [$tcpSource set cwnd_]
72     puts $file "$now $cwnd"
73     $ns at [expr $now+$time] "plotWindow $tcpSource $file"
74 }
75
76 # Процедура finish:
77 proc finish {} {
78     global tchan_
79
80 # подключение кода AWK:
81 set awkCode {
82 {
83 if ($1 == "Q" && NF>2) {
84 print $2, $3 >> "temp.q";
85 set end $2
86 }
87 else if ($1 == "a" && NF>2)
88 print $2, $3 >> "temp.a";
89 }
90 }
91 set f [open temp.queue w]
92 puts $f "TitleText: Red"
93 puts $f "Device: Postscript"
94 if { [info exists tchan_] } {
95 close $tchan_
96 }
97 exec rm -f temp.q temp.a
98 exec touch temp.a temp.q
99

```

Figure 2.3: Рис 3. Код программы(2)


```

84 print $2, $3 >> "temp.q";
85 set end $2
86 }
87 else if ($1 == "a" && NF>2)
88 print $2, $3 >> "temp.a";
89 }
90 }
91 set f [open temp.queue w]
92 puts $f "TitleText: Red"
93 puts $f "Device: Postscript"
94 if { [info exists tchan_] } {
95 close $tchan_
96 }
97 exec rm -f temp.q temp.a
98 exec touch temp.a temp.q
99
100 # выполнение кода AWK
101 exec awk $awkCode all.q
102 puts $f "\"queue_2
103 exec cat temp.q >@ $f
104 puts $f "\\n\\\"ave_queue_2
105 exec cat temp.a >@ $f
106 close $f
107
108 # Запуск xgraph с графиками окна TCP и очереди:
109 exec xgraph -bb -tk -x time -t "TCPRenoCWND" WindowVsTimeReno &
110 exec xgraph -bb -tk -x time -y queue temp.queue &
111 exit 0
112 }
113
114 # запуск модели
115 $ns run

```

Figure 2.4: Рис 4. Код программы(3)

3. Запускаю программу с помощью команды

```
ns lab02_1.tcl
```

Результат выполнения программы:



Figure 2.5: Рис 5. График №1 - Динамика размера окна TCP

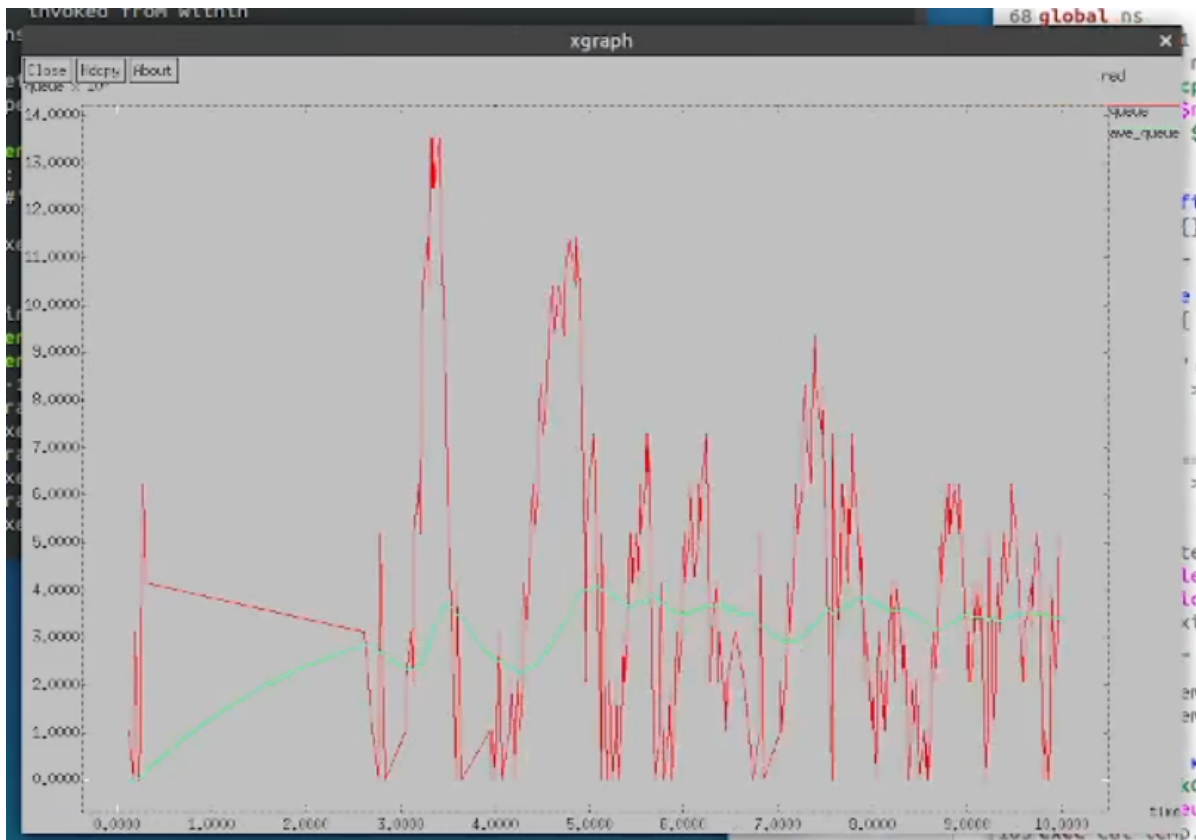


Figure 2.6: Рис 6. График №2 - Динамика очереди и средней длины очереди

4. В упражнении требуется изменить тип протоколов на Newreno и Vegas. На скриншоте ниже представлены графики для трех разных типов протоколов.

Первый столбец - Протокол Vegas

Второй столбец - Протокол Newreno

Третий столбец - Протокол Reno

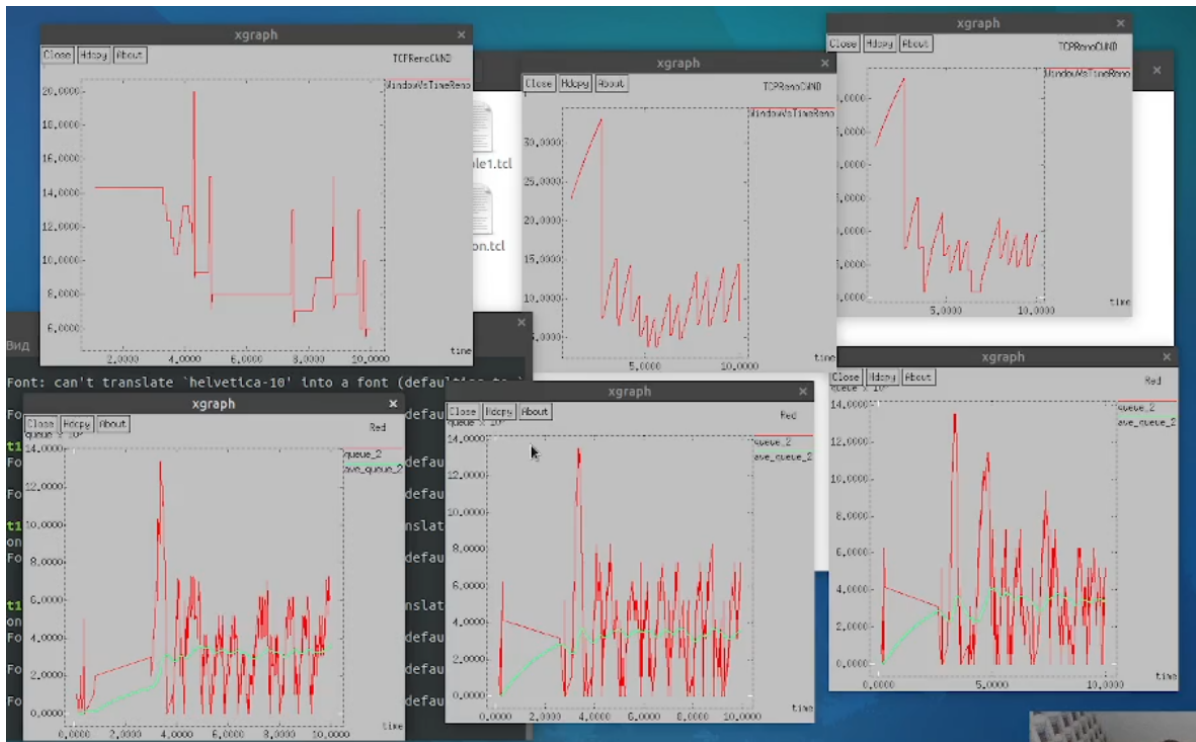


Figure 2.7: Рис 7. Графики для каждого протокола

5. Также в упражнении было необходимо внести изменения при отображении окон. Через Xgraph, установленный в образе, предоставленном нам, это сделать невозможно, т.к. там очень старая версия. Я попробовал установить Xgraph последней версии в своих системах Ubuntu 22 и Windows 11, но так и не удалось заставить это работать через команду ns. Если открыть файл через Xgraph самостоятельно, то в опциях возможно изменить все необходимые параметры, кроме цвета траекторий. Для изменения цвета траекторий я нашел способ, но у меня не получилось запустить его.

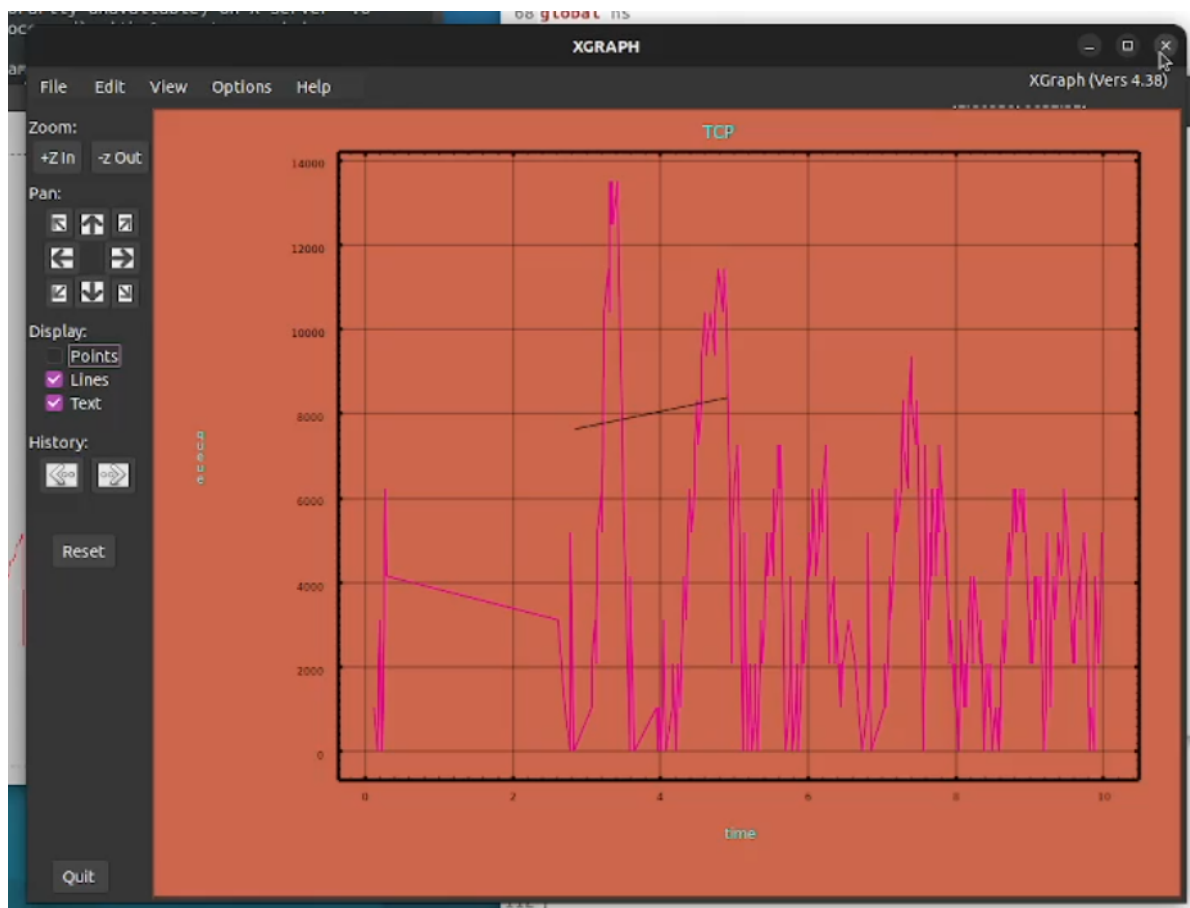


Figure 2.8: Рис 8. График с измененными характеристиками отображения

3 Выводы

Получены навыки работы с ns2 в связке с Xgraph

4 Библиография

1. Методические материалы курса