

Лабораторная работа №2

Исследование протокола TCP и алгоритма управления очередью RED

Оширова Юлия Николаевна, НФИбд-01-22

22 февраля 2025

Российский университет дружбы народов, Москва, Россия

Информация

- Оширова Юлия Николаевна
- студентка группы НФИбд-01-22
- Российский университет дружбы народов

Цель работы

Исследовать протокол TCP и алгоритм управления очередью RED.

Задание

- 1) Выполнить пример с дисциплиной RED;
- 2) Изменить в модели на узле s1 тип протокола TCP с Reno на NewReno, затем на Vegas.
- 3) Сравнить и пояснить результаты;
- 4) Внести изменения при отображении окон с графиками (изменить цвет фона, цвет траекторий, подписи к осям, подпись траектории в легенде).

Выполнение лабораторной работы

Выполним построение сети в соответствии с описанием:

сеть состоит из 6 узлов; между всеми узлами установлено дуплексное соединение с различными пропускной способностью и задержкой 10 мс; узел r1 использует очередь с дисциплиной RED для накопления пакетов, максимальный размер которой составляет 25; TCP-источники на узлах s1 и s2 подключаются к TCP-приёмнику на узле s3; генераторы трафика FTP прикреплены к TCP-агентам. Теперь разработаем сценарий, реализующий модель согласно описанию, чтобы построить в Xgraph график изменения TCP-окна, график изменения длины очереди и средней длины очереди. (рис. (fig:001?)) После запуска кода получаем график изменения TCP-окна (рис. (fig:002?)), а также график изменения длины очереди и средней длины очереди (рис. (fig:003?)).

Код задания

```
/home/openmodelica/mip/lab-ns/lab2.tcl - Mousepad
Файл  Правка  Поиск  Вид  Документ  Справка

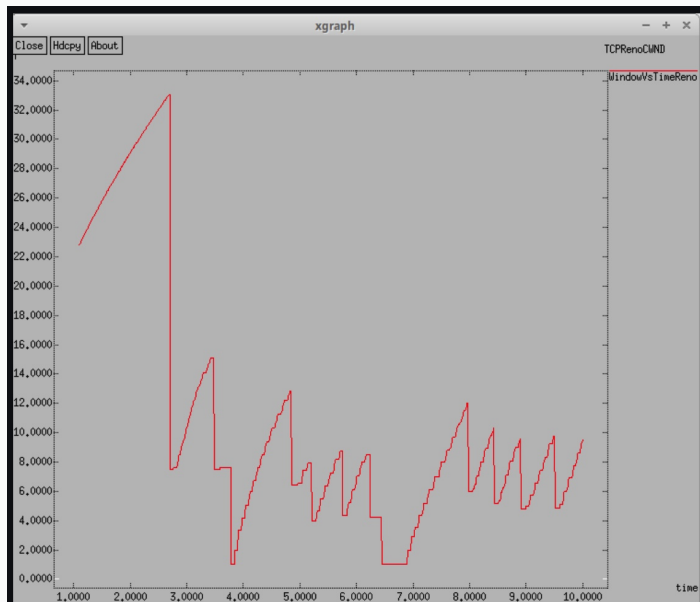
set nf [open out.nam w]

# все результаты моделирования будут записаны в переменную nf
$ns namtrace-all $nf

# открытие на запись файла трассировки out.tr
# для регистрации всех событий
set f [open out.tr w]
# все регистрируемые события будут записаны в переменную f
$ns trace-all $f

# Процедура finish:
proc finish {} {
    global tchan_
    # подключение кода AWK:
    set awkCode {
        {
            if ($1 == "Q" && NF>2) {
                print $2, $3 >> "temp.q";
                set end $2
            }
            else if ($1 == "a" && NF>2)
                print $2, $3 >> "temp.a";
        }
    }
    set f [open temp.queue w]
    puts $f "TitleText: red"
    puts $f "Device: Postscript"
    if { [info exists tchan_] } {
        close $tchan_
    }
    exec rm -f temp.q temp.a
    exec touch temp.a temp.q
    exec awk $awkCode all.q
    puts $f "Device: Postscript"
}
```

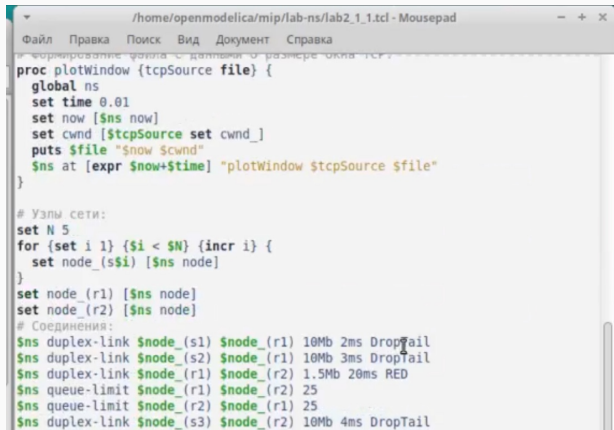
Что получилось



Изменение протокола TCP

Сначала требуется изменить тип Reno на NewReno. Для этого изменим код (рис. (fig:004?))

В результате получим следующие график изменения TCP-окна (рис. (fig:005?)), а также график изменения длины очереди и средней длины очереди (рис. (fig:006?)).

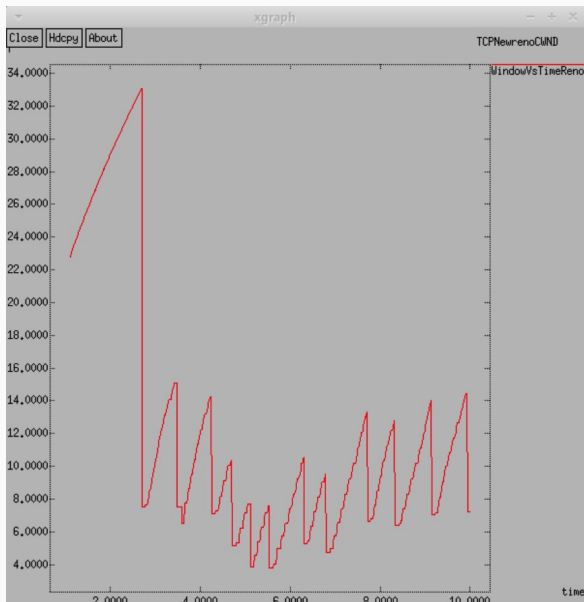


```
/home/openmodelica/mip/lab-ns/lab2_1_1.tcl - Mousepad
Файл  Правка  Поиск  Вид  Документ  Справка
# Формирование файла с данными о размере окна TCP
proc plotWindow {tcpSource file} {
    global ns
    set time 0.01
    set now [$ns now]
    set cwnd [$tcpSource set cwnd_]
    puts $file "$now $cwnd"
    $ns at [expr $now+$time] "plotWindow $tcpSource $file"
}

# Узлы сети:
set N 5
for {set i 1} {$i < $N} {incr i} {
    set node_($i) [$ns node]
}
set node_(r1) [$ns node]
set node_(r2) [$ns node]

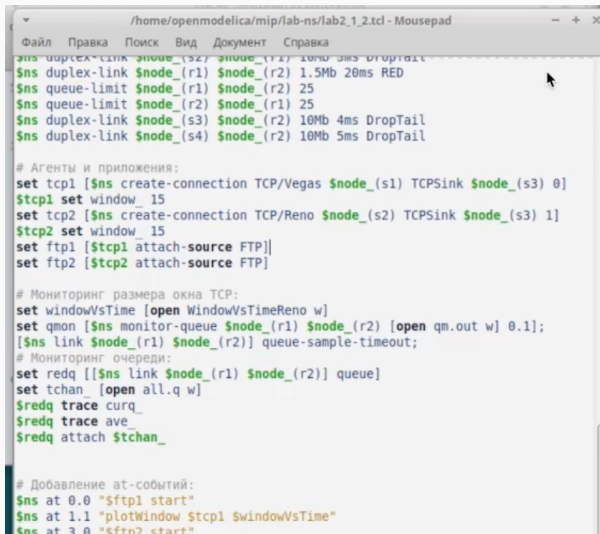
# Соединения:
$ns duplex-link $node_(s1) $node_(r1) 10Mb 2ms DropTail
$ns duplex-link $node_(s2) $node_(r1) 10Mb 3ms DropTail
$ns duplex-link $node_(r1) $node_(r2) 1.5Mb 20ms RED
$ns queue-limit $node_(r1) $node_(r2) 25
$ns queue-limit $node_(r2) $node_(r1) 25
$ns duplex-link $node_(s3) $node_(r2) 10Mb 4ms DropTail
```

Что получилось



Смена Reno на Vegas

Теперь изменим тип Reno на Vegas. Для этого изменим код (рис. (fig:007?))



```
/home/openmodelica/mip/lab-ns/lab2_1_2.tcl - Mousepad
Файл  Правка  Поиск  Вид  Документ  Справка

$ns duplex-link $node_(s2) $node_(r1) 10Mb 5ms DropTail
$ns duplex-link $node_(r1) $node_(r2) 1.5Mb 20ms RED
$ns queue-limit $node_(r1) $node_(r2) 25
$ns queue-limit $node_(r2) $node_(r1) 25
$ns duplex-link $node_(s3) $node_(r2) 10Mb 4ms DropTail
$ns duplex-link $node_(s4) $node_(r2) 10Mb 5ms DropTail

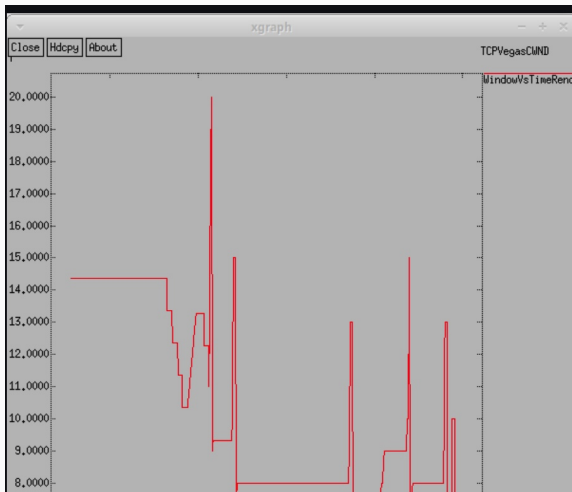
# Агенты и приложения:
set tcp1 [$ns create-connection TCP/Vegas $node_(s1) TCPSink $node_(s3) 0]
$tcp1 set window_ 15
set tcp2 [$ns create-connection TCP/Reno $node_(s2) TCPSink $node_(s3) 1]
$tcp2 set window_ 15
set ftp1 [$tcp1 attach-source FTP]
set ftp2 [$tcp2 attach-source FTP]

# Мониторинг размера окна TCP:
set windowVsTime [open WindowVsTimeReno w]
set qmon [$ns monitor-queue $node_(r1) $node_(r2) [open qm.out w] 0.1];
[$ns link $node_(r1) $node_(r2)] queue-sample-timeout;
# Мониторинг очереди:
set redq [[$ns link $node_(r1) $node_(r2)] queue]
set tchan_ [open all.q w]
$redq trace curq_
$redq trace ave_
$redq attach $tchan_

# Добавление at-событий:
$ns at 0.0 "$ftp1 start"
$ns at 1.1 "plotWindow $tcp1 $windowVsTime"
$ns at 3.0 "$ftp2 start"
```

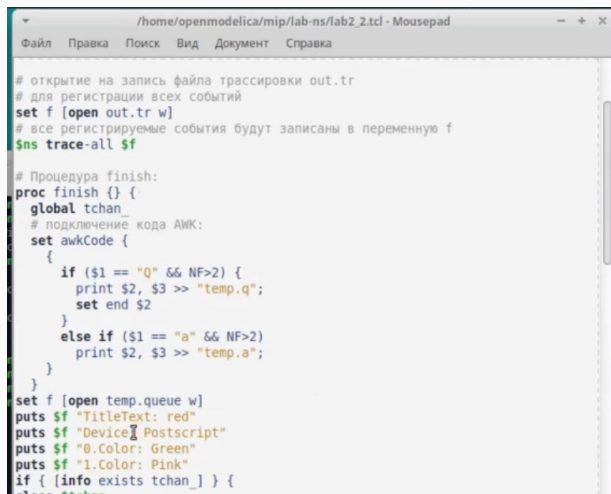
Что получилось

В результате получим следующие график изменения TCP-окна (рис. (fig:008?)), а также график изменения длины очереди и средней длины очереди (рис. (fig:009?)).



Изменение отображения окон с графиками

Внесем изменения при отображении окон с графиками, изменим цвет фона, цвет траекторий, подписи к осям и подпись траектории в легенде. Для этого изменим наш код:

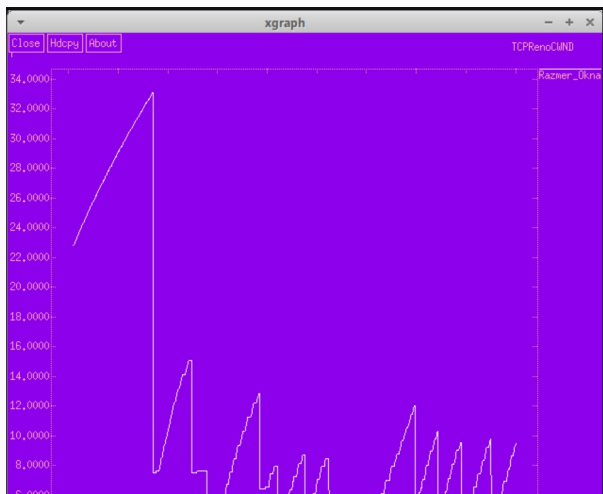


```
# открытие на запись файла трассировки out.tr
# для регистрации всех событий
set f [open out.tr w]
# все регистрируемые события будут записаны в переменную f
$ns trace-all $f

# Процедура finish:
proc finish {} {
    global tchan_
    # подключение кода AWK:
    set awkCode {
        {
            if ($1 == "Q" && NF>2) {
                print $2, $3 >> "temp.q";
                set end $2
            }
            else if ($1 == "a" && NF>2)
                print $2, $3 >> "temp.a";
        }
    }
    set f [open temp.queue w]
    puts $f "TitleText: red"
    puts $f "Device Postscript"
    puts $f "0.Color: Green"
    puts $f "1.Color: Pink"
    if { [info exists tchan_] } {
```


Что получилось

В результате получим следующие график изменения TCP-окна (рис. (fig:011?)), а также график изменения длины очереди и средней длины очереди (рис. (fig:012?)).



Выводы

В процессе выполнения данной лабораторной работы я исследовала протокол TCP и алгоритм управления очередью RED.