

Презентация по лабораторной работе №2

Протокол TCP и алгоритм управления очередью RED

Нзита Диатезилуа Катенди

Информация

- Нзита Диатезилуа Катенди
- студент группы НКНбд-01-21
- Российский университет дружбы народов
- <https://github.com/NzitaKatendi/Modeling-of-information-processes>

Цели и задачи работы

Цель лабораторной работы

Целью работы является изучение протокола TCP (Transmission Control Protocol) и алгоритма управления очередью RED (Random Early Detection). Оба эти элемента играют важную роль в современных компьютерных сетях.

Протокол TCP является одним из основных протоколов передачи данных в сетях TCP/IP. Его основное назначение - обеспечение надежной передачи данных путем управления потоком информации, обнаружения ошибок и повторной передачи потерянных пакетов. #
Выполнение лабораторной работы

Задача 1

Рассмотрел пример моделирования сети со следующими характеристиками:

- сеть состоит из 6 узлов;
- между всеми узлами установлено дуплексное соединение с различными пропускной способностью и задержкой 10 мс (см. рис. 2.4);
- 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.

На базе файла shablon.tcl, создал файл lab2.tcl и отредактировал его.

```
|
# создание объекта Simulator
set ns [new Simulator]

# 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
$ns duplex-link $node_(s4) $node_(r2) 10Mb 5ms DropTail

# Агенты и приложения:
set tcp1 [$ns create-connection TCP/Reno $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 WindowVsTimeNewReno 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 ат-событий:
$ns at 0.0 "$ftp1 start"
$ns at 1.1 "plotWindow $tcp1 $windowVsTime"
$ns at 3.0 "$ftp2 start"
$ns at 10 "finish"
```

Figure 1: Редактирование файла lab2.tcl

Запустил его командой *ns*.

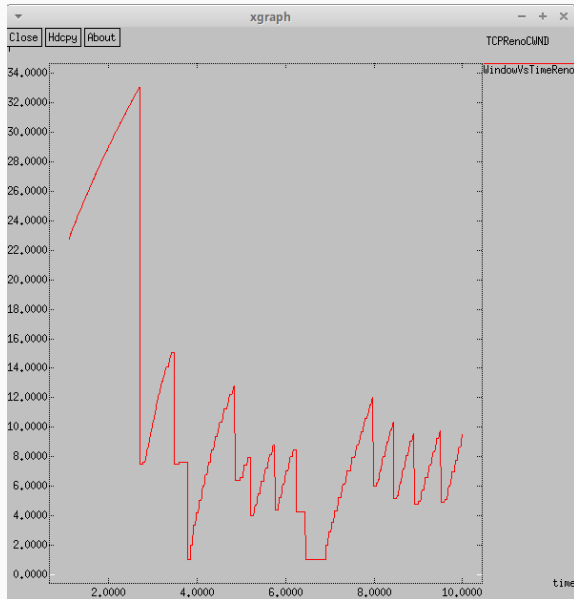
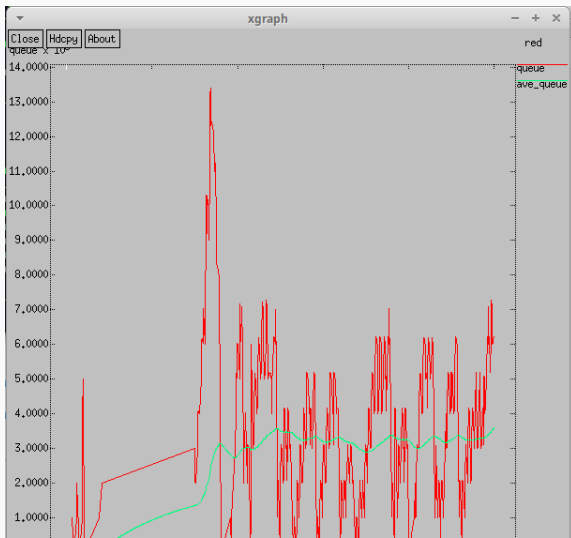


Figure 2: График динамики размера окна TCP

Задача 2

Отредактировал файл lab2.tc, изменив в модели на узле s1 тип протокола TCP с Reno на NewReno:



Отредактировал файл lab2.tc, изменив в модели на узле s1 тип протокола TCP с NewReno на Vegas:

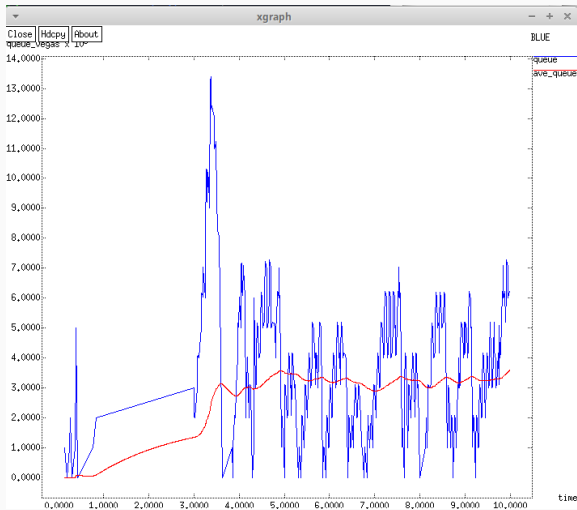
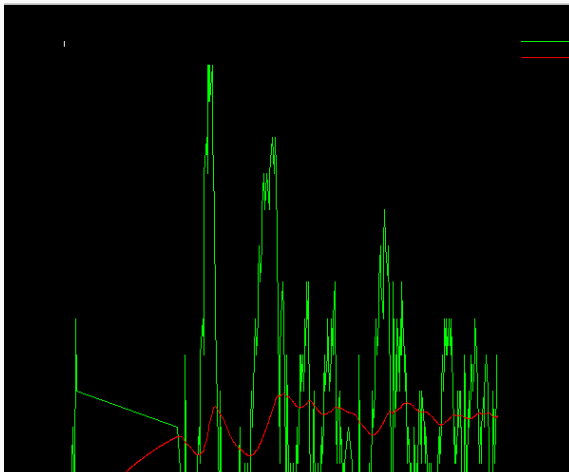


Figure 5: Изменение на узле s1 типа протокола TCP с NewReno на Vegas

Внес следующие изменения при отображении окон с графиками:

- Изменение цвета фона
- Изменение цвета траекторий
- Изменение подписей к осям
- Изменение подписи траектории в легенде



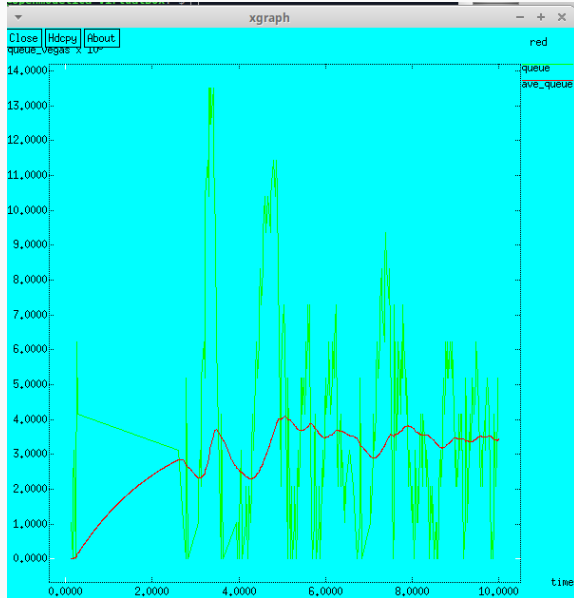


Figure 7: Изменение №2

Выводы

В результате проделанной лабораторной работы мы изучили протокол TCP и алгоритм управления очередью RED. В целом, изучение протокола TCP и алгоритма управления очередью RED подчеркивает их важную роль в обеспечении надежной и эффективной передачи данных в современных компьютерных сетях, а также необходимость балансирования между надежностью, производительностью и адаптивностью с целью оптимизации работы сети.