

# Презентация по лабораторной работе №1

Эттеев Сулейман

НКНбд-01-20

# Постановка задачи

Для 1 части:

- Создание шаблона сценария для NS-2.

Для 2 части:

- Требуется смоделировать сеть передачи данных, состоящую из двух узлов, соединенных дуплексной линией связи с полосой пропускания 2 Мб/с и задержкой 10 мс, очередью с обслуживанием типа DropTail. От одного узла у другому по протоколу UDP осуществляя передачу пакетов, размером 500 байт, с постоянной скоростью 200 пакетов в секунду.

Для 3 части:

- Описание моделируемой сети:

- сеть состоит из 4 узлов (n0, n1, n2, n3);
- между узлами n0 и n2, n1 и n2 установлено дуплексное соединение с пропускной способностью 2 Мбит/с и задержкой 10 мс;
- между узлами n2 и n3 установлено дуплексное соединение с пропускной способностью 1,7 Мбит/с и задержкой 20 мс;
- каждый узел использует очередь с дисциплиной DropTail для накопления пакетов, максимальный размер которой составляет 10;
- TCP-источник на узле n0 подключается к TCP-приёмнику на узле n3 (по-умолчанию, максимальный размер пакета, который TCP-агент может генерировать, равняется 1KByte) – TCP-приёмник генерирует и отправляет ACK пакеты отправителю и откидывает полученные пакеты;
- UDP-агент, который подсоединён к узлу n1, подключён к null-агенту на узле n3 (null-агент просто откидывает пакеты);
- генераторы трафика ftp и cbr прикреплены к TCP и UDP агентам соответственно;
- генератор cbr генерирует пакеты размером 1 Кбайт со скоростью 1 Мбит/с;
- работа cbr начинается в 0,1 секунду и прекращается в 4,5 секунды, а ftp начинает работать в 1,0 секунду и прекращает в 4,0 секунды.

Для 4 части:

- Требуется построить модель передачи данных по сети с кольцевой топологией и динамической маршрутизацией пакетов:

- сеть состоит из 7 узлов, соединённых в кольцо;
- данные передаются от узла n(0) к узлу n(3) по кратчайшему пути;
- с 1 по 2 секунду модельного времени происходит разрыв соединения между узлами n(1) и n(2);
- при разрыве соединения маршрут передачи данных должен измениться на резервный.

# Выполнение работы

```

202 LPU
% запуск модуля
203 9f 2'0 "tutur"
% uhopeiylba tutur nebez p c posle naziva modulinovannya
% 9f-corning tux iznannovannya corningu' kotoboe zaulyksei
}
exit 0
ex6c uaw onf' uaw 0
% запуск uaw в фоновом режиме
c7026 2uf % zakrytie fayzov ibessnovovki uaw
c7026 2f % zakrytie fayzov ibessnovovki
204 fupr-fl9c6 % ubezbezhenie ibessnovovki
0709f uz f u1 % oshchene luozryrnyx ubezbezheniy
bloc tutur {} {
% n zaulyksei vnzayuzetob uaw
% uhopeiylba tutur zakryvaei fayz ibessnovovki

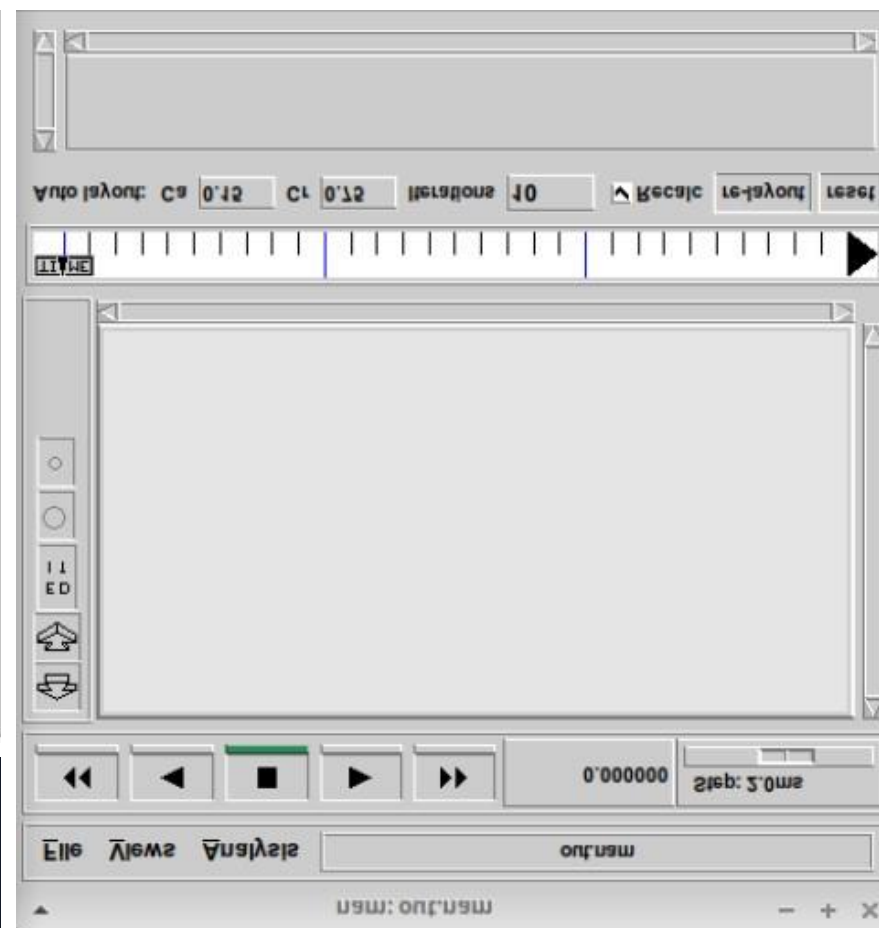
205 fl9c6-9ff 2f
% vse belncibvnyie corninga oluly zayusceni v ubezbezheniyu f
26f f [obeu onf' fl M]
% tux belncibvnyie vsech corningu
% otkrytie na zayusc fayza ibessnovovki onf' fl
206 uawfl9c6-9ff 2uf
% vse bezlyuzetib modulinovannya oluly zayusceni v ubezbezheniyu f
26f u1 [obeu onf' uaw M]
% otkrytie na zayusc fayza onf' uaw tux vnzayuzetobu uaw
26f uz [uaw 2fupf9f0f]
% sozryanie oprekiv 2fupf9f0f
\rome\obeshuofet9c9\wtb\gpr-uz\gpr9ou'9c9 [----] 2 f: { 14

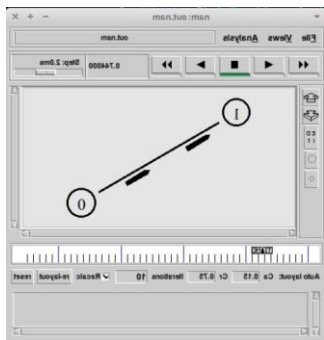
```

```

0beuwoqefjtc9@0beuwoqefjtc9-ATLcn9fbox:~\wtb\j9p-u2$ wc
0beuwoqefjtc9@0beuwoqefjtc9-ATLcn9fbox:~\wtb\j9p-u2$ f0ncp zmp9rou'fcj
0beuwoqefjtc9@0beuwoqefjtc9-ATLcn9fbox:~$ cq wtb\j9p-u2
0beuwoqefjtc9@0beuwoqefjtc9-ATLcn9fbox:~$ wqqtL -b wtb\j9p-u2

```





```
graph LR
    0((0)) --- 1((1))
```

Простой пример  
описания топологии  
сети, состоящей из  
двух узлов и одного  
соединения.

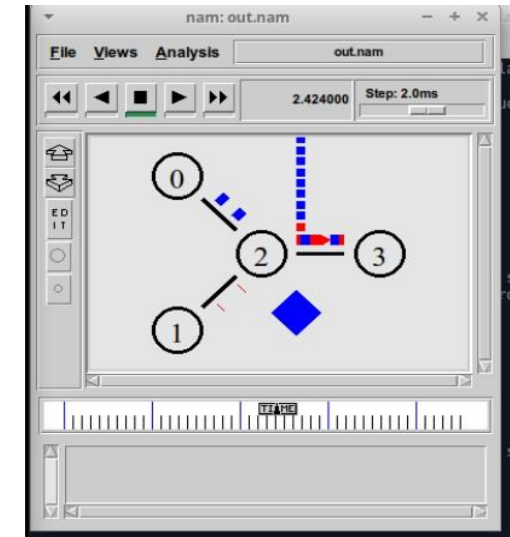
```
обменщогтс9бобешщогтс9-л7Л7н9гвох:-\штб\гэр-и2$ 2  
обменщогтс9бобешщогтс9-л7Л7н9гвох:-\штб\гэр-и2$ cb гурлгои'тсј 6х9шбгет'тсј
```

# .Пример с усложненной топологией сети

```
openmodelica@openmodelica-VirtualBox:~/mip/lab-ns$ ns example1.tcl
openmodelica@openmodelica-VirtualBox:~/mip/lab-ns$ cp shablon.tcl example2.tcl
openmodelica@openmodelica-VirtualBox:~/mip/lab-ns$ mc
```

```
openmodelica@openmodelica-VirtualBox:~/mip/lab-ns$ ns example2.tcl
openmodelica@openmodelica-VirtualBox:~/mip/lab-ns$
```

```
/home/openmodelica/mip/lab-ns/example2.tcl [----] 18 L: [ 31441 72/ 86] * (2488/29256) 0010 8x00A
set n 4
for (set i 0) ($i <= $n) {incr i} {
  set n($i) [new node]
}
ns duplex-link $n(0) $n(2) 200 10ms DropTail
ns duplex-link $n(1) $n(2) 200 10ms DropTail
ns duplex-link $n(3) $n(2) 200 10ms DropTail
ns duplex-link-op $n(0) $n(2) orient right-down
ns duplex-link-op $n(1) $n(2) orient right-up
ns duplex-link-op $n(2) $n(3) orient right
}
# создание агента UDP в пространстве ero x yany n(0)
set udp0 [new Agent/UDP]
ns attach-agent $n(0) $udp0
# создание источника CBR-трафика
# в пространстве ero x yany udp0
set cbr0 [new Application/Traffic/EBR]
cbr0 set packetSize 500
cbr0 set interval 0.005
cbr0 attach-agent $udp0
# создание агента TCP в пространстве ero x yany n(1)
set tcp1 [new Agent/TCP]
ns attach-agent $n(1) $tcp1
# создание приложения FTP
# в пространстве ero x yany tcp1
set ftp [new Application/FTP]
ftp attach-agent $tcp1
# создание агента nonuniform для udp0
set null0 [new Agent/null]
ns attach-agent $n(3) $null0
# создание агента nonuniform для tcp1
set sink1 [new Agent/TCPsink]
ns attach-agent $n(3) $sink1
ns connect $udp0 $null0
ns connect $tcp1 $sink1
ns color 1 Blue
ns color 2 Red
$udp0 set class 1
$tcp1 set class 2
ns duplex-link-op $n(2) $n(3) queuePos 0.5
ns queue-limit $n(2) $n(3) 20
ns at 0.5 '$cbr0 start'
ns at 1.0 '$ftp start'
ns at 4.0 '$ftp stop'
ns at 4.5 '$cbr0 stop'
# и-события для планирования событий, которые запускают
# планировщик планировщика через 5 с после начала моделирования
ns at 5.0 'finish'
```



# Пример с кольцевой топологией сети

```
openmodelica@openmodelica-VirtualBox:~/mip/lab-ns$ cp shablon.tcl example3.tcl
openmodelica@openmodelica-VirtualBox:~/mip/lab-ns$ mc
```

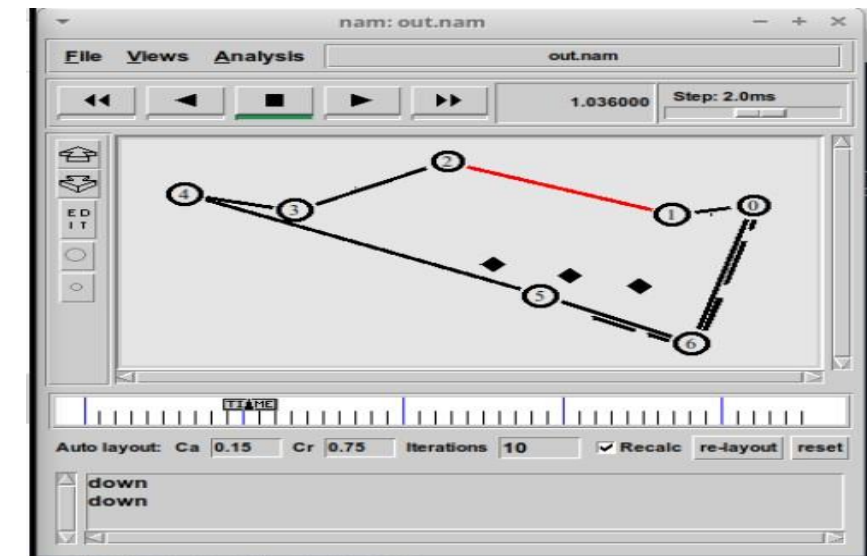
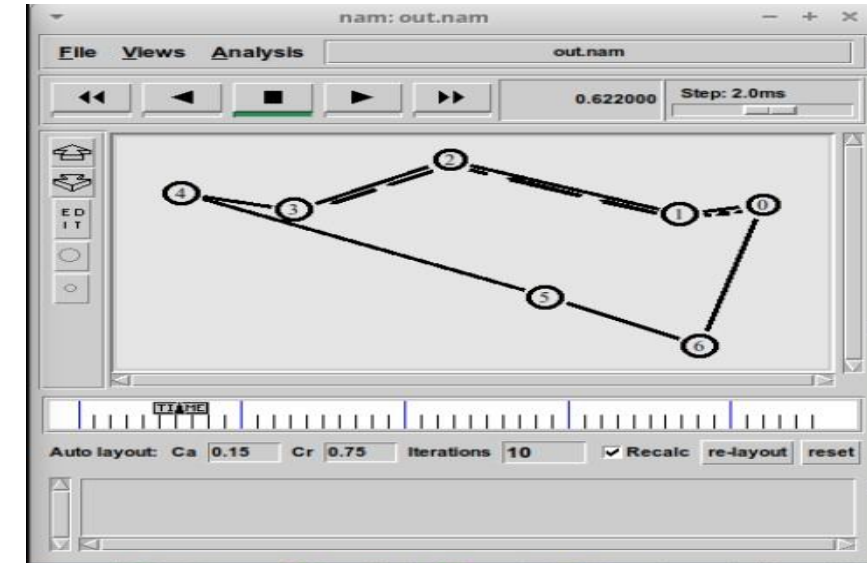
```
/home/openmodelica/mip/lab-ns/example3.tcl [----] 24 L:[ 11+35 46/ 57] *
* все регистрируемые события будут записаны в переменную f
$ns trace-all $f

* процедура finish закрывает файлы трассировки
* и запускает визуализатор nam
proc finish {} {
  global ns f nf
  * описание глобальных переменных
  $ns flush-trace
  * прекращение трассировки
  close $f
  * закрытие файлов трассировки
  close $nf
  * закрытие файлов трассировки nam
  * запуск nam в фоновом режиме
  exec nam out.nam &
  exit 0
}

set N 7
for {set i 0} {$i < $N} {incr i} {
  set n($i) [$ns node]
}

for {set i 0} {$i < $N} {incr i} {
  $ns duplex-link $n($i) $n([expr ($i+1)%$N]) 1Mb 10ms DropTail
}

set udp0 [new Agent/UDP]
$ns attach-agent $n(0) $udp0
set cbr0 [new Agent/CBR]
$ns attach-agent $n(0) $cbr0
$cbr0 set packetSize 500
$cbr0 set interval 0.005
set null0 [new Agent/Null]
$ns attach-agent $n(3) $null0
$ns connect $cbr0 $null0
$ns at 0.5 "$cbr0 start"
$ns rtmodel-at 1.0 down $n(1) $n(2)
$ns rtmodel-at 2.0 up $n(1) $n(2)
$ns at 4.5 "$cbr0 stop"
$ns at 5.0 "finish"
$ns rtproto DV
* at-событие для планировщика событий, которое запускает
* процедуру finish через 5 с после начала моделирования
$ns at 5.0 "finish"
* запуск модели
$ns run
```



# Заключение

В этой лабораторной работе я приобрел навыки моделирования сетей передачи данных с помощью средства имитационного моделирования NS-2.