

Лабораторная Работа №1

Простые модели компьютерной сети

Ощепков Дмитрий Владимирович

Российский университет дружбы народов им. Патриса Лумумбы, Москва, Россия

- Ощепков Дмитрий Владимирович
- НФИбд-01-22
- Российский университет дружбы народов
- [1132226442@pfur.ru]

Выполнение лабораторной работы

Приобретение навыков моделирования сетей передачи данных с помощью средства имитационного моделирования NS-2, а также анализ полученных результатов моделирования.

1.1. Шаблон сценария для NS-2 1.2. Простой пример описания топологии сети, состоящей из двух узлов и одного соединения 1.3. Пример с усложнённой топологией сети 1.4. Пример с кольцевой топологией сети

Подготовил папки для лабораторных работ (рис. 1)

```
openmodelica@doshchepkov:~/Desktop$  
openmodelica@doshchepkov:~/Desktop$ mkdir -p mip/lab-ns  
openmodelica@doshchepkov:~/Desktop$ cd mip/lab-ns  
openmodelica@doshchepkov:~/Desktop/mip/lab-ns$ touch shablon.tcl  
openmodelica@doshchepkov:~/Desktop/mip/lab-ns$ ls  
shablon.tcl  
openmodelica@doshchepkov:~/Desktop/mip/lab-ns$ nano shablon.tcl
```

Figure 1: Базовые команды линукс

Скопировал код шаблона (рис. 2)

```
GNU nano 2.9.3

set ns [new Simulator]

set nf [open out.nam w]

$ns namtrace-all $nf
set f [open out.tr w]
# 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

    # запуск nam в фоновом режиме
    exec nam out.nam &
    exit 0
}

# at-событие для планировщика событий, которое запускает
# процедуру finish через 5 с после начала моделирования
$ns at 5.0 "finish"
# запуск модели
$ns run
```

Расширил программу после шаблона (рис. 3)

```
GNU nano 2.9.3

# создание объекта Simulator
set ns [new Simulator]

# открытие на запись файла out.nam для визуализатора nam
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

proc finish {} {
    global ns f nf
    $ns flush-trace
    close $f
    close $nf
    exec nam out.nam &
    exit 0
}

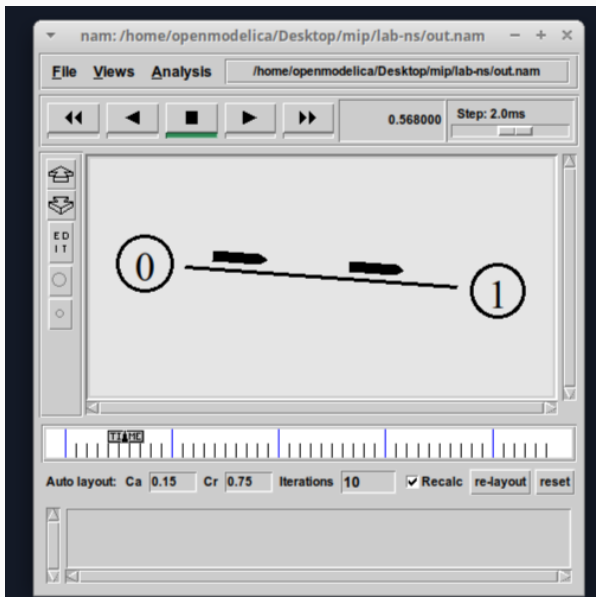
set N 2
for {set i 0} {$i < $N} {incr i} {
    set n($i) [$ns node]
}

$ns duplex-link $n(0) $n(1) 2Mb 10ms DropTail
set udp0 [new Agent/UDP]
$ns attach-agent $n(0) $udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize 500
$cbr0 set interval 0.005
$cbr0 attach-agent $udp0

set null0 [new Agent/Null]
$ns attach-agent $n(1) $null0
$ns connect $udp0 $null0
$ns at 0.5 "$cbr0 start"
$ns at 4.5 "$cbr0 stop"

$ns at 5.0 "finish"
# запуск модели
$ns run
```


Запустил файл через ns (рис. 4)



Поменял код, тоже скопировал код из ТУИС (рис. 5)

```
GNU nano 2.9.3

[создание объекта Simulator]
ns [new Simulator]

[открытие на запись $out.out.nam для визуализатора nam]
nf [open out.nam w]
[все результаты моделирования будут записаны в переменную n]
ns namtrace-all $nf

[открытие на запись $out.tr.trace для трассировки out.tr]
nf [open out.tr w]
[все пересланные события будут записаны в переменную n]
ns trace-all $f

rec finish () {
    global ns f nf
    $ns flush-trace
    close $f
    close $nf
    exec nam out.nam &
    exit 0
}

set N 4
for {set i 0} {$i < $N} {incr i} {
    set n($i) $ns node

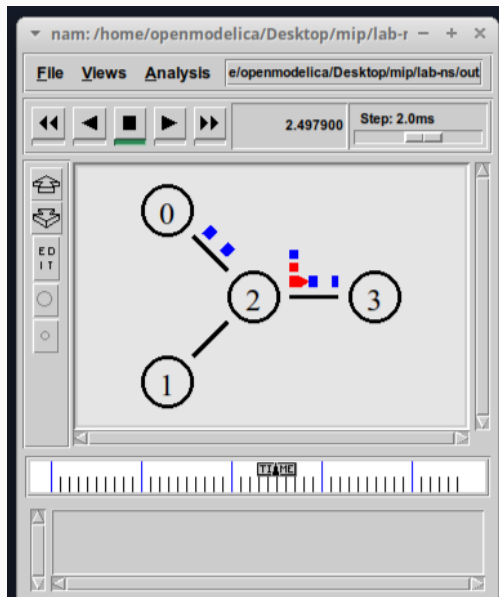
    ns duplex-link $n(0) $n(2) 2Mb 10ms DropTail
    ns duplex-link $n(1) $n(2) 2Mb 10ms DropTail
    ns duplex-link $n(3) $n(2) 2Mb 10ms DropTail
    ns duplex-link-op $n(0) $n(2) orient right-down
    ns duplex-link-op $n(1) $n(2) orient right-up
    ns duplex-link-op $n(2) $n(3) orient right

    [создание агента UDP и присоединение его к узлу n(0)]
    ns udpr0 [new Agent/UDP]
    ns attach-agent $n(0) $udpr0
    [создание источника CBR-трафика]
    [и присоединение его к агенту udpr0]
    ns cbr0 [new Application/Traffic/CBR]
    cbr0 set packetSize 500
    cbr0 set interval 0.005
    cbr0 attach-agent $udpr0

    [создание агента TCP и присоединение его к узлу n(1)]
    ns tcpl [new Agent/TCP]
    ns attach-agent $n(1) $tcpl
    [создание приложения FTP]
    [и присоединение его к агенту tcpl]
    ns ftp [new Application/FTP]
    ftp attach-agent $tcpl

    [создание агента-получателя для udpr0]
    ns null0 [new Agent/Null]
    ns attach-agent $n(3) $null0
}
```

Запустил файл через ns (рис. 6)



Поменял код под кольцевую модель, тоже скопировал код из ТУИС (рис. 7)

```
# создание объекта Simulator
set ns [new Simulator]
$ns rtproto DV

# открытие на запись файла out.nam для визуализатора nam
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

proc finish {} {
    global ns f nf
    $ns flush-trace
    close $f
    close $nf
    exec nam out.nam &
    exit 0
}

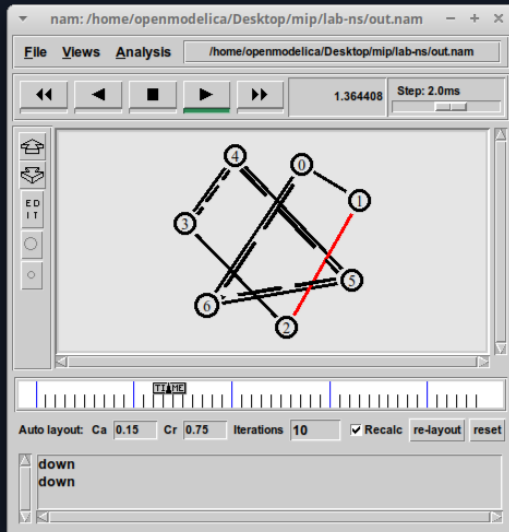
set N 7
for {set i 0} {$i < $N} {incr i} {
    set n($i) [$ns node]
}

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"
# запуск модели
```

Запустил файл через ns (рис. 8)



Приобрел навыки моделирования сетей передачи данных с помощью средства имитационного моделирования NS-2, а также научился анализу полученных результатов моделирования.