Computer networks Laboratory

Ns2 - Simulator

Overview of the Document

1. Ns2 - Introduction
2. Installation of ns2
3. TCL commands
4. Creating nodes and Links (wired Network)
5. UDP Data traffic (wired Network)
6. TCP data traffic (wired Network)
7. Error Model
8. NS Commands
9. Post Processing

Dr.G.Rajesh
Assistant Professor(Sr.G)
Department of Information technology
MIT campus, Anna university
gr@annauniv.edu
raajiimegce@gmail.com

1. ns2- Network Simulator

- One of the most popular simulators among networking researchers **<u>Open source, free</u>**
- Discrete event, Packet level simulator
- Network protocol stack written in C++
- Tcl (Tool Command Language) used for specifying scenarios and events.
- **Ns ->>>the simulator itself**
- **Nam-- > the network animator (helps to visualize the output)**

Preprocessing	**Post Processing**
Topology creation Traffic generation	Trace analysis -Perl or TCL or Grep

2. Installation of NS2

In our laboratory both Linux and ns2 are already installed. You want to try at your own follow the manual in the below link.

https://drive.google.com/file/d/1zABwJ3OLL6Qnucb3pupX1t2Zf_pGClaW/view?usp=sharing

3. TCL Commands

variables:

```
set a 10

set b 15

puts $a

puts "the value of a = $a"

puts "the value of b = $b
```

functions and expressions

```
set c [expr $a + $b]

puts "the value of c = $c"

set d [expr [expr $b - $a] * $c]

puts "the value of d = $d"

----------------------------------------
```

procedures

```
proc display {} {

puts "this is the testing message"

 }
display
----------------------------------------

proc add {x y} {

set z [expr $x + $y]

puts "the value of $x + $y = $z"

 }

add 10 20

----------------------------------------
```

```
proc print { k } {
for {set i 0} {$i < $k} {incr i} {
 puts "node_($i)"
 puts "n$i"
 }
 }
print 10
```

control flow

```
if {$x > 0} { return $x }
else { return [expr -$x] }
while { $x > 0 } { puts $x incr x -1 }
```

Arrays

```
set matrix(1,1) 140
```

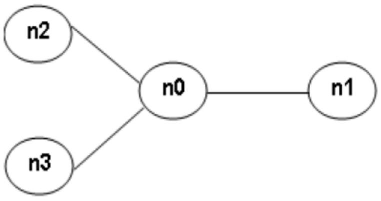
Files

```
set test [open file1.txt w]
puts $test "Testing message"
```

4. CREATE NODES AND LINK

```
set ns [new Simulator]  #Create a simulator object # (Create event scheduler)
#create file for analysis mode
set tr [open out.tr w]     #Open trace file
$ns trace-all $tr
#create file for Animation Mode

set namtr [open out.nam w]
$ns namtrace-all $namtr
#Create Node
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
#Create Link     // wired network
$ns duplex-link $n0 $n1 10Mb 5ms DropTail
$ns duplex-link $n2 $n0 10Mb 5ms DropTail
$ns duplex-link $n3 $n0 10mb 5ms DropTail
#Create Orientation
$ns duplex-link-op $n0 $n1 orient right
$ns duplex-link-op $n0 $n2 orient left-up
$ns duplex-link-op $n0 $n3 orient left-down
$ns at 10.0 "$ns halt"
$ns run
```



5. UDP DATA TRAFFIC

```
#create UDP Source
set udp0 [new Agent/UDP]
$ns attach-agent $n3 $udp0
#create UDP Destination
set null0 [new Agent/Null]
$ns attach-agent $n1 $null0
#connecting UDP Source & Destination
$ns connect $udp0 $null0
#create application traffic
set cbr0 [new Application/Traffic/CBR]
$cbr0 attach-agent $udp0
#Application start time
$ns at 1.0 "$cbr0 start"
#Application Stop time
$ns at 5.0 "$cbr0 stop"
```

6. TCP DATA TRAFFIC

```
#create TCP Source
set tcp0 [new Agent/TCP]
$ns attach-agent $n2 $tcp0
#create TCP Destination
set sink0 [new Agent/TCPSink]
$ns attach-agent $n1 $sink0
$ns connect $tcp0 $sink0
#create application traffic
set ftp0 [new Application/FTP]
$ftp0 attach-agent $tcp0
#Application start & stop time
$ns at 2.0 "$ftp0 start"
$ns at 5.0 "$ftp0 stop
```

7. ERROR MODEL

```
#Link Error Module
Set loss_mod [new ErrorModel]
$loss_mod set rate_ 0.2
$loss_mod ranvar [new RandomVariable/Uniform]
$loss_mod drop-target [new Agent/Null]
$ns lossmodel $loss_mod $n0 $n1
```

8. NS Commands

#Application Parameters

```
$cbr0 set packetSize_ 512

$cbr0 set rate_ 1Mb

#$cbr0 set interval_ 0.1
```

#To change Color of Nodes

```
$n0 color red
```

#To set flow ID

```
$udp0 set fid_ 2

$tcp0 set fid_ 1
```

#To set Flow ID color

```
$ns color 0 Brown

$ns color 1 Blue

$ns color 2 Red
```

9. Post Processing

GREP

#"Total Number of packets send from node 3 to node 0 "

```
grep “^+” out.tr| grep “ 3 0 “|grep “cbr”|wc -l
```

#"Total Number of packets received from node 0 to node 1 "

```
grep “^r” out.tr| grep “ 0 1 “|grep “cbr”|wc -l
```