

Part-A

Experiment No: 1

Aim: *Simulate a three node point to point network with duplex links between them. Set queue size and vary the bandwidth and find number of packets dropped.*

Program:

```
set ns [new Simulator]                # Letter S is capital
set nf [open PA1.nam w]                # open a nam trace file in write mode
$ns namtrace-all $nf                  # nf  nam filename
set tf [open PA1.tr w]                 # tf  trace filename
$ns trace-all $tf

proc finish { } {
    global ns nf tf
    $ns flush-trace                     # clears trace file contents
    close $nf
    close $tf
    exec nam PA1.nam &
    exit 0
}
set n0 [$ns node]                      # creates 3 nodes
set n2 [$ns node]
set n3 [$ns node]

$ns duplex-link $n0 $n2 200Mb 10ms DropTail # establishing links
$ns duplex-link $n2 $n3 1Mb 1000ms DropTail
$ns queue-limit $n0 $n2 10

set udp0 [new Agent/UDP]                # attaching transport layer protocols
$ns attach-agent $n0 $udp0
set cbr0 [new Application/Traffic/CBR]   # attaching application layer protocols
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0

set null0 [new Agent/Null]               # creating sink(destination) node
$ns attach-agent $n3 $null0
$ns connect $udp0 $null0

$ns at 0.1 "$cbr0 start"
$ns at 1.0 "finish"
$ns run
```

AWK file: (Open a new editor using “gedit or vi command” and write awk file and save with “.awk” extension)

#immediately after BEGIN should open braces ‘{‘

```
BEGIN{ c=0;}
{
    if($1=="d")
    {
        c++;
        printf("%s\t%s\n",$5,$11);
    }
}
END{ printf("The number of packets dropped =%d\n",c); }
```

Steps for execution (When using gedit as editor):

- Open gedit editor and type program. Program name should have the extension “.tcl ”
[root@localhost ~]# gedit lab1.tcl
- Open gedit editor and type awk program. Program name should have the extension “.awk ”
[root@localhost ~]# gedit lab1.awk
- Run the simulation program
[root@localhost~]# ns lab1.tcl
- Here “ns” indicates network simulator. We get the topology shown in the snapshot.
- Now press the play button in the simulation window and the simulation will begins.
- After simulation is completed run awk file to see the output ,
[root@localhost~]# awk -f lab1.awk PA1.tr
- To see the trace file contents open the file as ,
[root@localhost~]# gedit PA1.tr

Steps for execution (When using vi as editor):

- Open vi editor and type program. Program name should have the extension “.tcl ”
[root@localhost ~]# vi lab1.tcl
- Save the program by pressing “ESC key” first, followed by “Shift and :” keys simultaneously and type “wq” and press Enter key.
- Open vi editor and type awk program. Program name should have the extension “.awk ”
[root@localhost ~]# vi lab1.awk
- Save the program by pressing “ESC key” first, followed by “Shift and :” keys simultaneously and type “wq” and press Enter key.
- Run the simulation program
[root@localhost~]# ns lab1.tcl
- Here “ns” indicates network simulator. We get the topology shown in the snapshot.
- Now press the play button in the simulation window and the simulation will begins.
- After simulation is completed run awk file to see the output ,
[root@localhost~]# awk -f lab1.awk PA1.tr
- To see the trace file contents open the file as ,
[root@localhost~]# vi PA1.tr

Trace file contains 12 columns:

Event type, Event time, From Node, To Node, Packet Type, Packet Size, Flags (indicated by -----), Flow ID, Source address, Destination address, Sequence ID, Packet ID

The left screenshot shows a terminal window with the following output:

```

root@localhost:~# cat lab01.tr
0.1 0 2 cbr 500 ----- 0 0.0 3.0 0 0
- 0.1 0 2 cbr 500 ----- 0 0.0 3.0 0 0
r 0.10108 0 2 cbr 500 ----- 0 0.0 3.0 0 0
+ 0.10108 2 3 cbr 500 ----- 0 0.0 3.0 0 0
- 0.10108 2 3 cbr 500 ----- 0 0.0 3.0 0 0
+ 0.105 0 2 cbr 500 ----- 0 0.0 3.0 1 1
- 0.105 0 2 cbr 500 ----- 0 0.0 3.0 1 1
r 0.10608 0 2 cbr 500 ----- 0 0.0 3.0 1 1
+ 0.10608 2 3 cbr 500 ----- 0 0.0 3.0 1 1
- 0.10608 2 3 cbr 500 ----- 0 0.0 3.0 1 1
+ 0.11 0 2 cbr 500 ----- 0 0.0 3.0 2 2
- 0.11 0 2 cbr 500 ----- 0 0.0 3.0 2 2
r 0.11108 0 2 cbr 500 ----- 0 0.0 3.0 2 2
+ 0.11108 2 3 cbr 500 ----- 0 0.0 3.0 2 2
- 0.11108 2 3 cbr 500 ----- 0 0.0 3.0 2 2
+ 0.115 0 2 cbr 500 ----- 0 0.0 3.0 3 3
- 0.115 0 2 cbr 500 ----- 0 0.0 3.0 3 3
r 0.11608 0 2 cbr 500 ----- 0 0.0 3.0 3 3
+ 0.11608 2 3 cbr 500 ----- 0 0.0 3.0 3 3
- 0.11608 2 3 cbr 500 ----- 0 0.0 3.0 3 3
+ 0.12 0 2 cbr 500 ----- 0 0.0 3.0 4 4
- 0.12 0 2 cbr 500 ----- 0 0.0 3.0 4 4
r 0.12108 0 2 cbr 500 ----- 0 0.0 3.0 4 4
+ 0.12108 2 3 cbr 500 ----- 0 0.0 3.0 4 4

```

The right screenshot shows a network topology diagram with three nodes: source2 (1), destination (3), and router1 (0). Arrows indicate traffic flow from source2 to router1 and then to destination.

The bottom screenshot shows a terminal window with the following output:

```

root@localhost:~# vi lab01.tcl
root@localhost:~# awk -f PA1.awk lab01.tr
cbr 139
cbr 143
cbr 130
cbr 149
cbr 151
cbr 154
cbr 139
cbr 159
cbr 163
cbr 145
cbr 169
cbr 171
cbr 174
cbr 177
cbr 179
cbr 182
The number of packets dropped =16
root@localhost:~#

```

Contents of Trace File

Topology

Output