**Exp 3: ETHERNET LAN USING N-NODES**

**Aim:** Simulate an Ethernet LAN using n nodes, change error rate and data rate and compare

throughput.

set ns [new Simulator]

set tf [open lab5.tr w]

$ns trace-all $tf

set nf [open lab5.nam w]

$ns namtrace-all $nf

$ns color 0 blue

set n0 [$ns node]

$n0 color "red"

set n1 [$ns node]

$n1 color "red"

set n2 [$ns node]

$n2 color "red"

set n3 [$ns node]

$n3 color "red"

set n4 [$ns node]

$n4 color "magenta"

set n5 [$ns node]

$n5 color "magenta"

set n6 [$ns node]

$n6 color "magenta"

set n7 [$ns node]

$n7 color "magenta"

$ns make-lan "$n0 $n1 $n2 $n3" 100Mb 300ms LL Queue/ DropTail Mac/802\_3

$ns make-lan "$n4 $n5 $n6 $n7" 100Mb 300ms LL Queue/ DropTail Mac/802\_3

$ns duplex-link $n3 $n4 100Mb 300ms DropTail

$ns duplex-link-op $n3 $n4 color "green"

# set error rate. Here ErrorModel is a class and it is single word and space should not be given between Error and Model

# lossmodel is a command and it is single word. Space should not be given between loss and model

set err [new ErrorModel]

$ns lossmodel $err $n3 $n4

$err set rate\_ 0.1

# error rate should be changed for each output like 0.1,0.3,0.5…. \*/

set udp [new Agent/UDP]

$ns attach-agent $n1 $udp

set cbr [new Application/Traffic/CBR]

$cbr attach-agent $udp

$cbr set fid\_ 0

$cbr set packetSize\_ 1000

$cbr set interval\_ 0.0001

set null [new Agent/Null]

$ns attach-agent $n7 $null

$ns connect $udp $null

proc finish { } {

global ns nf tf

$ns flush-trace

close $nf

close $tf

exec nam lab5.nam &

exit 0

}

$ns at 0.1 "$cbr start"

$ns at 3.0 "finish"

$ns run

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

BEGIN{

pkt=0;

time=0;

}

{

if($1= ="r" && $3= ="9" && $4= ="7"){

pkt = pkt + $6;

time =$2;

}

}

END {

printf("throughput:%fMbps",(( pkt / time) \* (8 / 1000000)));

}

**Steps for execution**

 Open vi editor and type program. Program name should have the extension “ .tcl ”

[root@localhost ~]# vi lab5.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 lab5.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 lab5.tcl

o Here “ns” indicates network simulator. We get the topology shown in the snapshot.

o 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 lab5.awk lab5.tr

 To see the trace file contents open the file as ,

[root@localhost~]# vi lab5.tr

Here “h” indicates host.





This above output is for error rate 0.1. During next execution of simulation change error rate to 0.3, 0.5….., and check its effect on throughput.

**Exp 4: ETHERNET LAN USING N-NODES WITH MULTIPLE TRAFFIC**

**Aim:** Simulate an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination

set ns [new Simulator]

set tf [open pgm7.tr w]

$ns trace-all $tf

set nf [open pgm7.nam w]

$ns namtrace-all $nf

set n0 [$ns node]

$n0 color "magenta"

$n0 label "src1"

set n1 [$ns node]

set n2 [$ns node]

$n2 color "magenta"

$n2 label "src2"

set n3 [$ns node]

$n3 color "blue"

$n3 label "dest2"

set n4 [$ns node]

set n5 [$ns node]

$n5 color "blue"

$n5 label "dest1"

$ns make-lan "$n0 $n1 $n2 $n3 $n4" 100Mb 100ms LL Queue/ DropTail Mac/802\_3

#should come in single line

$ns duplex-link $n4 $n5 1Mb 1ms DropTail

set tcp0 [new Agent/TCP]

$ns attach-agent $n0 $tcp0

set ftp0 [new Application/FTP]

$ftp0 attach-agent $tcp0

$ftp0 set packetSize\_ 500

$ftp0 set interval\_ 0.0001

set sink5 [new Agent/TCPSink] # Listens for a client connection

$ns attach-agent $n5 $sink5

$ns connect $tcp0 $sink5

set tcp2 [new Agent/TCP]

$ns attach-agent $n2 $tcp2

set ftp2 [new Application/FTP]

$ftp2 attach-agent $tcp2

$ftp2 set packetSize\_ 600

$ftp2 set interval\_ 0.001

set sink3 [new Agent/TCPSink]

$ns attach-agent $n3 $sink3

$ns connect $tcp2 $sink3

set file1 [open file1.tr w]

$tcp0 attach $file1

set file2 [open file2.tr w]

$tcp2 attach $file2

$tcp0 trace cwnd\_ # must put underscore after cwnd and no space between them $tcp2 trace cwnd\_

# Congestion Window

proc finish { } {

global ns nf tf

$ns flush-trace

close $tf

close $nf

exec nam pgm7.nam &

exit 0

}

$ns at 0.1 "$ftp0 start"

$ns at 5 "$ftp0 stop"

$ns at 7 "$ftp0 start"

$ns at 0.2 "$ftp2 start"

$ns at 8 "$ftp2 stop"

$ns at 14 "$ftp0 stop"

$ns at 10 "$ftp2 start"

$ns at 15 "$ftp2 stop"

$ns at 16 "finish"

$ns run

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

cwnd:- means congestion window

BEGIN {

}

{

if($6= ="cwnd\_") # don’t leave space after writing cwnd\_

printf("%f\t%f\t\n",$1,$7); # you must put \n in printf

}

END {

}

**Steps for execution**

 Open vi editor and type program. Program name should have the extension “.tcl ”

[root@localhost ~]# vi lab7.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 lab7.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 lab7.tcl

 After simulation is completed run awk file to see the output , [root@localhost~]# awk –f lab7.awk file1.tr > a1 [root@localhost~]# awk –f lab7.awk file2.tr > a2 [root@localhost~]# xgraph a1 a2

 Here we are using the congestion window trace files i.e. file1.tr and file2.tr and we are redirecting the contents of those files to new files say a1 and a2 using output redirection operator (>).

 To see the trace file contents open the file as ,

[root@localhost~]# vi lab7.tr

**Topology:**



Output:

