

~~~~~ Generic setup ~~~~~

1. # Create an instance of Simulator (ns)

set myNS [new Simulator]

2. # Create and enable tracing and animation for analysis and visualizations.

#Setting up files for trace & NAM

set traceFile [open out.tr w]

set namFile [open out.nam w]

Enable Event collections for numerical tracing and animated visualization of network

\$myNS trace-all \$traceFile

\$myNS namtrace-all \$namFile

~~~~~ Network specific setup ~~~~~

3. #Creating NODES (L1-L3)

set <nodeName> [\$<my NS instance> node]

set n0 [\$myNS node]

set n1 [\$myNS node]

set n2 [\$myNS node]

set n3 [\$myNS node]

#Label the nodes

\$n0 label "TCP Source"

\$n1 label "UDP Source"

\$n2 label "Router"

\$n3 label "Sink/Null"

#Define different colors for data flow

\$myNS color 1 blue

\$myNS color 2 red

4. Create the Agents (L4) to be attached over the Nodes (add as many agents as necessary for the given application)

set <TRANSPORT_INSTANCE> [new <TYPE of AGENT>]

set udp_0 [new Agent/UDP]

set null [new Agent/Null]

set tcp_0 [new Agent/TCP]

set sink [new Agent/TCPSink]

5. Create the applications (L5) which runs over different nodes / Agents

set <APP_INSTANCE> [new <TYPE of APPLICATION>]

set cbr [new Application/Traffic/CBR]

set ftp [new Application/FTP]

6. Setup the Links and positions of nodes (for visualizations only - optional to have)

#Creating LINKS

\$myNS <Type of link> <SOURCE> <DESTINATION> <Link BW> <DELAY> <Type of Queuing>

\$myNS duplex-link \$n0 \$n2 2Mb 10ms DropTail

\$myNS duplex-link \$n1 \$n2 2Mb 10ms DropTail

\$myNS duplex-link \$n2 \$n3 1.0Mb 20ms DropTail

#Set the Queue-Limit for each link

\$myNS queue-limit <SOURCE> <DESTINATION> <MAX PACKETS QUEUES>

\$myNS queue-limit \$n2 \$n3 40

#Node position for NAM

\$myNS duplex-link-op \$n0 \$n2 orient right-down

\$myNS duplex-link-op \$n1 \$n2 orient right-up

\$myNS duplex-link-op \$n2 \$n3 orient right

7. Attach Agents over nodes (L4)

\$myNS attach-agent \$n1 \$udp

\$myNS attach-agent \$n3 \$null

\$myNS attach-agent \$n0 \$tcp

\$myNS attach-agent \$n3 \$sink

8. Attach Applications over Agents

#CBR over UDP

\$cbr attach-agent \$udp

#FTP over TCP connection

\$ftp attach-agent \$tcp

9. Create the events to occur in the network - when to start the application, when to stop, when to finish the simulation.

\$myNS at 0.1 "\$cbr start"

\$myNS at 1.0 "\$ftp start"

\$myNS at 4.0 "\$ftp stop"

\$myNS at 4.5 "\$cbr stop"

\$myNS at 5.0 "finish"

~~~~~ Generic setup ~~~~~

10. #Closing trace file and NAM file

```
proc finish { } {  
    global myNS traceFile namFile  
    $myNS flush-trace  
    close $traceFile  
    close $namFile
```

Following lines are needed for initiating numerical results / animated results at the end of the simulation.

```
# exec nam out.nam &  
# exec awk -f n1.awk out.tr &  
exit 0  
}
```

11. FINALLY initiate the simulation (\$ns run)

```
$myNS run
```

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

#immediately after BEGIN should open braces ‘{‘

BEGIN{ c=0;} # Performed once at the start of post processing....

{ # This block of operations are repeated for each entry in the TRACE file.... (xxx.tr)

```
if($1=="d")
{
    c++;
    printf("%s\t%s\n",$5,$11);
}
}
```

END{ printf("The number of packets dropped =%d\n",c); } # Performed once at the END of post processing....

TRACE FORMAT depends upon type of network and connectivity....

Event	Time	From Node	To Node	PKT Type	PKT Size	Flags	Fid	Src Addr	Dest Addr	Seq Num	Pkt id
-------	------	--------------	------------	-------------	-------------	-------	-----	-------------	--------------	------------	-----------