Information Security

s

Name: Aayush Gupta

Roll No: 11813031

Section: IT-3 (IT-A)

Class: 3rd year, B.Tech

ASSIGNMENT

Ques1. **Program in NS2 to implement a bus topology.**

Code:

```
set node1 [$ns node]

setnode2 [$ns node]

set node3 [$ns node]

setnode4 [$ns node]

set node5 [$ns node]


set lan0 [$ns newLan "$node1 $node2$node3 $node4  $node5" 0.7Mb 20ms
LL Queue/FQ MAC/Csma/Cd Channel]


set tcp0 [new Agent/TCP]

$tcp0 set class_ 1

$ns attach-agent $node1 $tcp0


set sink0 [new Agent/TCPSink]

$ns attach-agent $node3 $sink0


$ns connect $tcp0 $sink0
```

set cbr0 [new Application/Traffic/CBR]

$cbr0 set packetSize_ 500

$ cbr0 set interval_ 0.05

$cbr0 attach-agent $tcp0

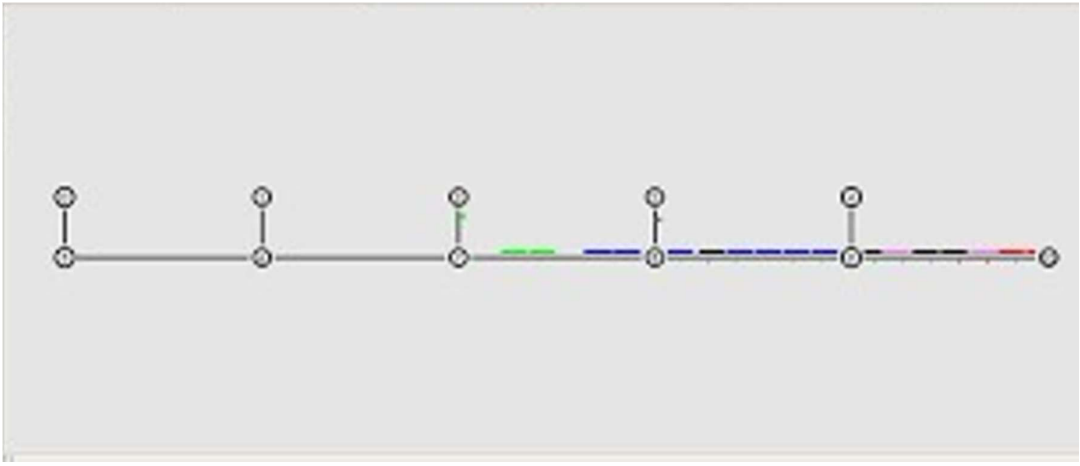
$ns at 0.5 "$cbr0 start time"

$ ns at 5.5 "$cbr0 stop time"

$ns at 10.0 "End"

$ns run

Output:



Ques2. Program in NS2 for connecting multiple routers and nodes and building a hybrid topology.

Ans.

```
set ns [new Simulator]
set nf [open out.nam w]
$ns namtrace-all $nf

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

$ns rtproto DV

for {set i 1} {$i<5} {incr i} {
        set r($i) [$ns node]
        set h($i) [$ns node]
}

for {set i 1} {$i<15} {incr i} {
        set p($i) [$ns node]
}
```

```
for {set i 1} {$i<4} {incr i}
{
        $ns duplex-link $h(1) $p($i) 1.5Mb 10ms SFQ
        $ns duplex-link $h(3) $p([expr ($i+7)]) 1.5Mb 10ms SFQ
}

for {set i 4} {$i<8} {incr i}
{
        $ns duplex-link $h(2) $p($i) 1.5Mb 10ms SFQ
        $ns duplex-link $h(4) $p([expr ($i+7)]) 1.5Mb 10ms SFQ
}



for {set i 1} {$i<4} {incr i}
 {
        $ns duplex-link $r($i) $r([expr ($i+1)]) 1.5Mb 10ms SFQ
        $ns duplex-link $r($i) $h($i) 1.5Mb 10ms SFQ
}

$ns duplex-link $r(4) $r(1) 1.5Mb 10ms SFQ
$ns duplex-link $r(4) $h(4) 1.5Mb 10ms SFQ

    $ns duplex-link-op $r(1) $h(1) orient up
$ns duplex-link-op $r(2) $h(2) orient right $ns duplex-link-op $r(3)
$h(3) orient down
```

```
$ns duplex-link-op $r(4) $h(4) orient left



set tcp3 [new Agent/TCP]
set tcp9 [new Agent/TCPSink]

$ns attach-agent $p(3) $tcp3
$ns attach-agent $p(9) $tcp9

$ns connect $tcp3 $tcp9

set tcp5 [new Agent/TCP]
set tcp12 [new Agent/TCPSink]

$ns attach-agent $p(5) $tcp5
$ns attach-agent $p(12) $tcp12

$ns connect $tcp5 $tcp12

set ftp3 [new Application/FTP]
set ftp5 [new Application/FTP]

$ftp3 attach-agent $tcp3
$ftp5 attach-agent $tcp5

set udp13 [new Agent/UDP]
```

```
set udp6 [new Agent/Null]

$ns attach-agent $p(13) $udp13
$ns attach-agent $p(6) $udp6

$ns connect $udp13 $udp6

set udp1 [new Agent/UDP]
set udp8 [new Agent/Null]

$ns attach-agent $p(1) $udp1
$ns attach-agent $p(8) $udp8

$ns connect $udp1 $udp8



#creating CBR applications for udp
set cbr13 [new Application/Traffic/CBR]

# send 50 packets in 1 second i.e 1 packet every 1/50 second
$cbr13 set packetSize_ 1536
$cbr13 set interval_ 0.02
$cbr13 attach-agent $udp13

set cbr1 [new Application/Traffic/CBR]
```

```
# send 400 packets in 1 second i.e 1 packet every
1/400 second
$cbr1 set packetSize_ 5632
$cbr1 set interval_ 0.0025
$cbr1 attach-agent $udp1

# ------------------------------------------------

# setting events

$ns rtmodel-at 0.7 down $r(1) $r(2)
$ns rtmodel-at 1.0 up $r(1) $r(2)

$ns rtmodel-at 0.9 down $r(4) $r(3)
$ns rtmodel-at 1.3 up $r(4) $r(3)



$ns at 0.2 "$ftp3 start"
$ns at 1.8 "$ftp3 stop"

$ns at 0.3 "$ftp5 start"
$ns at 1.4 "$ftp5 stop"



$ns at 0.4 "$cbr13 start"
$ns at 1.6 "$cbr13 stop"
```

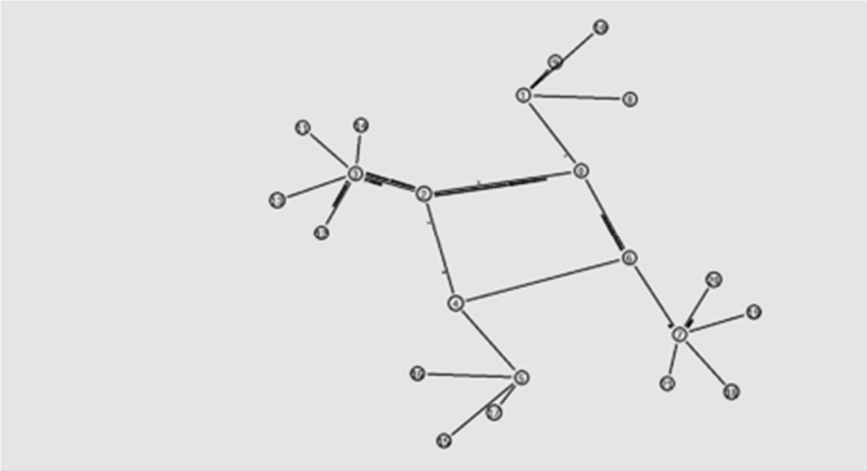
$ns at 0.7 "$cbr1 start"

$ns at 1.7 "$cbr1 stop"

$ns at 2.0 "finish"

$ns run

OUTPUT:



Ques 3. Program in NS2 to implement FTP using TCP bulk transfer.

Code:

Set ns [new Simulator]

Set n0 [$ns node]

Set n1 [$ns node]

$ns duplex-link-op $n0 $n1 2MB 10ms Droptail

Set tcp0[new Agent/TCP]

```
$ns attach-agent $n0 $tcp0

Set ftp0 [new Application/FTP]

$ftp0 attach-agent $tcp0
$tcp0 set packet_size_512
```

Ques 4. Program in NS2 for connecting multiple routers and nodes and building a hybrid topology and then calculating network perform

Code:

```
set ns [new Simulator]
set nf [open out.nam w]
$ns namtrace-all $nf

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

$ns rtproto DV
```

```
for {set i 1} {$i<5} {incr i} {
        set r($i) [$ns node]
        set h($i) [$ns node]
}

for {set i 1} {$i<15} {incr i} {
        set p($i) [$ns node]
}

for {set i 1} {$i<4} {incr i}
{
        $ns duplex-link $h(1) $p($i) 1.5Mb 10ms SFQ
        $ns duplex-link $h(3) $p([expr ($i+7)]) 1.5Mb 10ms SFQ
}

for {set i 4} {$i<8} {incr i}
{
        $ns duplex-link $h(2) $p($i) 1.5Mb 10ms SFQ
        $ns duplex-link $h(4) $p([expr ($i+7)]) 1.5Mb 10ms SFQ
}


#Creating Links between r nodes and connecting r to h
for {set i 1} {$i<4} {incr i} {
$ns duplex-link $r($i) $r([expr ($i+1)]) 1.5Mb 10ms
SFQ
```

```
$ns duplex-link $r($i) $h($i) 1.5Mb 10ms SFQ
}

$ns duplex-link $r(4) $r(1) 1.5Mb 10ms SFQ
$ns duplex-link $r(4) $h(4) 1.5Mb 10ms SFQ

$ns duplex-link-op $r(1) $h(1) orient up
$ns duplex-link-op $r(2) $h(2) orient right $ns duplex-link-op $r(3)
$h(3) orient down
$ns duplex-link-op $r(4) $h(4) orient left

set tcp3 [new Agent/TCP]
set tcp9 [new Agent/TCPSink]

$ns attach-agent $p(3) $tcp3
$ns attach-agent $p(9) $tcp9

$ns connect $tcp3 $tcp9

set tcp5 [new Agent/TCP]
set tcp12 [new Agent/TCPSink]

$ns attach-agent $p(5) $tcp5
$ns attach-agent $p(12) $tcp12

$ns connect $tcp5 $tcp12
```

```
set ftp3 [new Application/FTP]
set ftp5 [new Application/FTP]

$ftp3 attach-agent $tcp3
$ftp5 attach-agent $tcp5


set udp13 [new Agent/UDP]
set udp6 [new Agent/Null]

$ns attach-agent $p(13) $udp13
$ns attach-agent $p(6) $udp6

$ns connect $udp13 $udp6

set udp1 [new Agent/UDP]
set udp8 [new Agent/Null]

$ns attach-agent $p(1) $udp1
$ns attach-agent $p(8) $udp8

$ns connect $udp1 $udp8


set cbr13 [new Application/Traffic/CBR]
```

```
$cbr13 set packetSize_ 1536
$cbr13 set interval_ 0.02
$cbr13 attach-agent $udp13

set cbr1 [new Application/Traffic/CBR]

# send 400 packets in 1 second i.e 1 packet every
1/400 second
$cbr1 set packetSize_ 5632
$cbr1 set interval_ 0.0025
$cbr1 attach-agent $udp1

$ns rtmodel-at 0.7 down $r(1) $r(2)
$ns rtmodel-at 1.0 up $r(1) $r(2)

$ns rtmodel-at 0.9 down $r(4) $r(3)
$ns rtmodel-at 1.3 up $r(4) $r(3)


$ns at 0.2 "$ftp3 start"
$ns at 1.8 "$ftp3 stop"

$ns at 0.3 "$ftp5 start"
$ns at 1.4 "$ftp5 stop"
```

$ns at 0.4 "$cbr13 start"

$ns at 1.6 "$cbr13 stop"

$ns at 0.7 "$cbr1 start"

$ns at 1.7 "$cbr1 stop"

$ns at 2.0 "finish"

$ns run

Ques 5. To analyze network traces using Wireshark software.

Ans.

Program Wireshark is used for tracing the communication between devices interconnected by LAN. The devices (e.g. PBX connected with a GSM gateway via LAN) send to each other packets which are captured by the above mentioned program. Wireshark is distributed under the Open source licence.

Before starting network capture

Before Wireshark (or in general, any packet capture tool) is used, careful consideration should be given to where in the network packets are to be captured.

In order to capture all the packets which are sent via particular LAN, the devices have to be part of the same network segment (they have to be connected via HUB). If you do not have a hub you can use a switch which supports so called port mirroring

Download Wireshark

Start Wireshark

On a Linux or Unix environment, select the Wireshark or Ethereal entry in the desktop environment's menu, or run "wireshark" (or "ethereal") from a root shell in a terminal emulator.

In a Microsoft Windows environment, launch wireshark.exe from program installation folder.

Note that on Unix systems, a non-GUI version of Wireshark called "tshark" (or "tethereal") may be available as well, but its use is beyond the scope of this article.

Use Wireshark (capture network communication)

After starting Wireshark, do the following:

1. Select **Capture** | **Interfaces**
2. Select the interface on which packets need to be captured.
3. Now click the **Start** button to start the capture.
4. Recreate the issue. The capture dialog should show the number of packets increasing. If packets are not being captured, try removing any filters that have been defined.
5. **Important:** In order to provide us with detailed information about the possible problem with the device, please do not apply any filter and simply capture all the communication during the test (call/smpp/email2sms,etc...).
6. Once the issue which is to be analysed has been reproduced, click on **Stop**. It might take a few seconds for Wireshark to display the packets captured.

7. Save the packet trace in following format: **„File>Save As->Wireshark/tcpdump/… libpcap(*.pcap;*.cap)** .