

## Writeup

## Assignment No. B2

- Title : AODV - ns2
- Problem Statement: Use network simulation ns2 to implement
  - a) Monitoring traffic for given topology
  - b) Analysis of CSMA and Ethernet protocols
  - c) Networking Routing : Shortest path routing (AODV)
  - d) Analysis of congestion control (TCP, UDP)
- Objective : To study network simulation tool and monitor traffic for given topology.
- Outcome : Students will be able to successfully monitor a network with the help of ns2 tool.
- Software and Hardware Requirements:
  - ns2 (Network simulator tool)
  - latest version of 64-bit Ubuntu 20.04 LTS, etc.
- Theory :
  - Networking Traffic Monitoring : is the process of reviewing, analyzing and managing network traffic for any abnormality or process that can affect network performance, Availability and/or security. It is a network management process that uses various tools and techniques to study computer based communication data/packet traffic.

→ Carrier-Sense Multiple Access (CSMA):

It is a media access control (MAC) protocol in which a node verifies the absence of other traffic before transmitting on a shared transmission medium such as an electrical bus or a band of magnetic spectrum. A transmitter attempts to determine whether another transmission using a carrier-sense mechanism. It tries to detect the presence of carrier signal from another node before attempting to transmit.

If a carrier is sensed, the node waits for the transmission in progress to end before initiating its own transmission.

→ CSMA/CD:

It is a modification of pure CSMA. It is used to improve CSMA performance by transmitting transmission as soon as collision is detected, thus shortening the time required before a retry can be attempted.

→ Network Routing:

Routing is the process of selecting a path for traffic in a network or between or across multiple networks.

→ Congestion in computer network:

A state occurring in network layer when the message traffic is so heavy that it slows down network response time.



→ Effects of congestion:

As delay increases performance decreases. If delays increase, transmission occurs making situation worse.

→ Network Simulator:

A network simulator is a software that predicts the behaviour of a computer network.

→ TCL:

- It is a tool command language.
- It is a powerful scripting language with programming features

• Conclusion:

We have learned the concept of monitoring network traffic, CSMA protocol, network routing, AODV routing protocol, and demonstrated using ns2 (network simulator).

### Sample Code

```
-----b2.tcl-----
```

```
set ns [new Simulator]
```

```
$ns color 1 Blue
```

```
$ns color 2 Red
```

```
set nf [open b2.nam w]
```

```
$ns namtrace-all $nf
```

```
set nt [open b2.tr w]
```

```
$ns trace-all $nt
```

```
$ns use-newtrace
```

```
proc finish {} {
```

```
global ns nf nt
```

```
$ns flush-trace
```

```
close $nf
```

```
close $nt
```

```
exec nam b2.nam &
```

```
exec awk -f cnlb2.awk b2.tr &
```

```
exit 0
```

```
}
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
set n2 [$ns node]
```

```
set n3 [$ns node]
```

```
set n4 [$ns node]
```

```
set n5 [$ns node]
```

```
$ns duplex-link $n0 $n2 10Mbps 10ms DropTail
```

```
$ns duplex-link $n1 $n2 10Mbps 10ms DropTail
```

```
$ns duplex-link $n2 $n3 10Mbps 10ms DropTail
```

```
$ns duplex-link $n3 $n4 10Mbps 10ms DropTail
```

```
$ns duplex-link $n3 $n5 10Mbps 10ms DropTail
```

```
$ns duplex-link-op $n0 $n2 orient down-right
```

```
$ns duplex-link-op $n1 $n2 orient up-right
```

```
$ns duplex-link-op $n2 $n3 orient right
```

\$ns duplex-link-op \$n3 \$n4 orient right-up  
\$ns duplex-link-op \$n3 \$n5 orient right-down

set tcp [new Agent/TCP]  
\$tcp set class\_ 2  
\$ns attach-agent \$n0 \$tcp  
set sink [new Agent/TCPSink]  
\$ns attach-agent \$n4 \$sink  
\$ns connect \$tcp \$sink  
\$tcp set fid\_ 1

set ftp [new Application/FTP]  
\$ftp attach-agent \$tcp  
\$ftp set type\_ FTP  
\$ftp set packet\_size\_ 1000  
\$ftp set rate\_ 1mb

set udp [new Agent/UDP]  
\$ns attach-agent \$n1 \$udp  
set null [new Agent/Null]  
\$ns attach-agent \$n5 \$null  
\$ns connect \$udp \$null  
\$udp set fid\_ 2

set cbr [new Application/Traffic/CBR]  
\$cbr attach-agent \$udp  
\$cbr set type\_ CBR  
\$cbr set packet\_size\_ 1000  
\$cbr set rate\_ 1mb

\$ns at 1.0 "\$ftp start"

\$ns at 3.0 "\$ftp stop"

\$ns at 1.5 "\$cbr start"

\$ns at 3.5 "\$cbr stop"

\$ns at 5.0 "finish"

\$ns run

-----cnlb2.awk-----

BEGIN{

sentPkts = 0

recvPkts = 0

forwardedPkts = 0

stime = 0

ftime = 0

flag = 0

fsize = 0

throughput = 0

latency = 0

sentPkts1 = 0

recvPkts1 = 0

forwardedPkts1 = 0

stime1 = 0

ftime1 = 0

flag1 = 0

fsize1 = 0

throughput1 = 0

latency1 = 0

}

```
{

if($1=="r" && $4==4)
{
fsize+=$6

if(flag==0)
{
stime = $2
flag = 1
}

ftime = $2
}

if($1=="r" && $4==5)
{
fsize1+=$6

if(flag1==0)
{
stime1 = $2
flag1 = 1
}

ftime1 = $2
}

if($1=="s")
{
sentPkts++;
```

```

}

if($1=="r")
{
recvPkts++;
}

if($1=="s")
{
forwardedPkts++;
printf("\nF P = %f", forwardedPkts);
}

}

END{
latency = ftime-stime
throughput = (fsize*8)/latency

latency1 = ftime1-stime1
throughput1 = (fsize1*8)/latency1

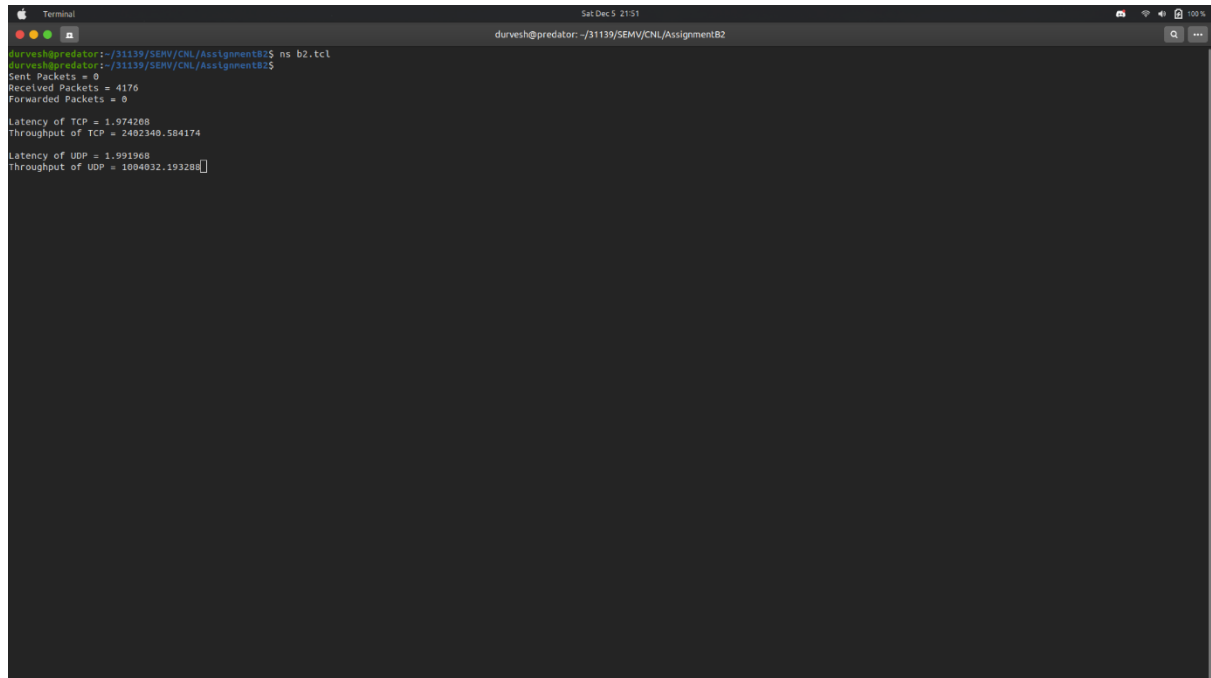
printf("\nSent Packets = %d", sentPkts)
printf("\nReceived Packets = %d", recvPkts)
printf("\nForwarded Packets = %d", forwardedPkts)

printf("\n\nLatency of TCP = %f", latency)
printf("\nThroughput of TCP = %f", throughput)
printf("\n\nLatency of UDP = %f", latency1)
printf("\nThroughput of UDP = %f", throughput1)
}

```



## Output



```
Terminal
Sat Dec 5 21:51
durvesh@predator: ~/31139/SEHV/CNL/AssignmentB2
durvesh@predator:~/31139/SEHV/CNL/AssignmentB2$ ns b2.tcl
durvesh@predator:~/31139/SEHV/CNL/AssignmentB2$
Sent Packets = 0
Received Packets = 4176
Forwarded Packets = 0

Latency of TCP = 1.974208
Throughput of TCP = 2402340.584174

Latency of UDP = 1.991868
Throughput of UDP = 1004032.193288
```