

Experiment 4:

Implement simple ESS and with transmitting nodes in wire-less LAN by simulation and determine the performance with respect to transmission of packets.

Step1: Open text editor, type the below program and save with extension .tcl (**prog4.tcl**)

```
set ns [new Simulator]
set tf [open prog4.tr w]
$ns trace-all $tf
set topo [new Topography]
$topo load_flatgrid 1000 1000
set nf [open prog4.nam w]
$ns namtrace-all-wireless $nf 1000 1000
set val(chan) Channel/WirelessChannel ;
set val(prop) Propagation/TwoRayGround ;
```

```
set val(netif) Phy/WirelessPhy ;
set val(mac) Mac/802_11 ;
set val(ifq) Queue/DropTail/PriQueue ;
set val(ll) LL ;
set val(ant) Antenna/OmniAntenna ;
set val(ifqlen) 50 ;
set val(nn) 2 ;
set val(rp) AODV ;
set val(x) 500 ;
set val(y) 400 ;
set val(stop) 10.0 ;
```

```
$ns node-config -adhocRouting $val(rp) \
-llType $val(ll) \
-macType $val(mac) \
-ifqType $val(ifq) \
-ifqLen $val(ifqlen) \
-antType $val(ant) \
-propType $val(prop) \
-phyType $val(netif) \
-channelType $val(chan) \
-topoInstance $topo \
-agentTrace ON \
-routerTrace ON \
```

```
-macTrace OFF \  
-movementTrace ON
```

```
create-god 3  
set n0 [$ns node]  
set n1 [$ns node]  
set n2 [$ns node]
```

```
$n0 label "tcp0"  
$n1 label "sink1/tcp1"  
$n2 label "sink2"
```

```
$n0 set X_ 50  
$n0 set Y_ 50  
$n0 set Z_ 0  
$n1 set X_ 100  
$n1 set Y_ 100  
$n1 set Z_ 0  
$n2 set X_ 600  
$n2 set Y_ 600  
$n2 set Z_ 0  
$ns at 0.1 "$n0 setdest 50 50 15"  
$ns at 0.1 "$n1 setdest 100 100 25"  
$ns at 0.1 "$n2 setdest 600 600 25"  
set tcp0 [new Agent/TCP]  
$ns attach-agent $n0 $tcp0  
set ftp0 [new Application/FTP]  
$ftp0 attach-agent $tcp0  
set sink1 [new Agent/TCPSink]  
$ns attach-agent $n1 $sink1  
$ns connect $tcp0 $sink1  
set tcp1 [new Agent/TCP]  
$ns attach-agent $n1 $tcp1  
set ftp1 [new Application/FTP]  
$ftp1 attach-agent $tcp1  
set sink2 [new Agent/TCPSink]  
$ns attach-agent $n2 $sink2  
$ns connect $tcp1 $sink2  
$ns at 5 "$ftp0 start"  
$ns at 5 "$ftp1 start"
```

```
$ns at 100 "$n1 setdest 550 550 15"
$ns at 190 "$n1 setdest 70 70 15"
proc finish { } {
    global ns nf tf
    $ns flush-trace
    exec nam prog6.nam &
    close $tf
    exit 0
}
$ns at 250 "finish"
$ns run
```

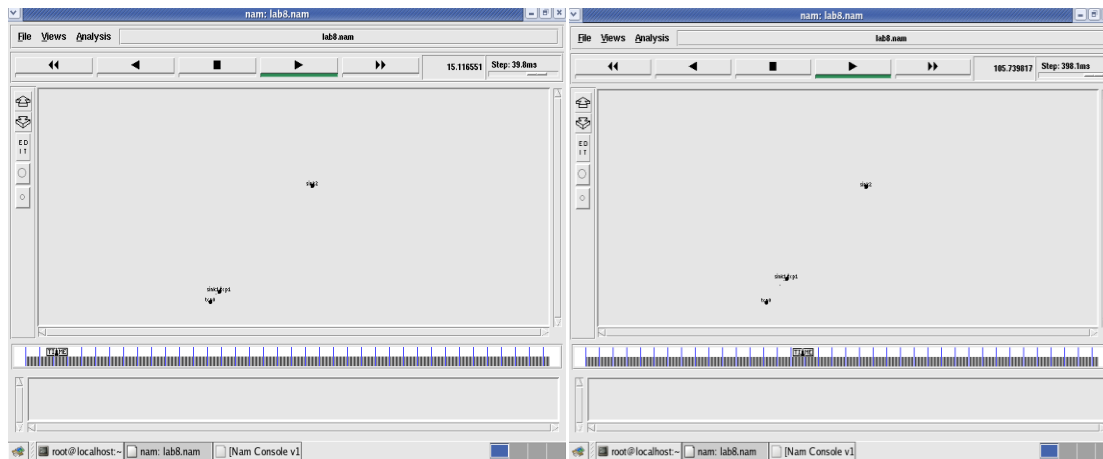
Step2: Open text editor, type the below program and save with extension .awk (**prog4.awk**)

```
BEGIN{
    count1=0
    count2=0
    pack1=0
    pack2=0
    time1=0
    time2=0
}
{
    if($1=="r" && $3=="_1_" && $4=="AGT")
    {
        count1++
        pack1=pack1+$8
        time1=$2
    }
    if($1=="r" && $3=="_2_" && $4=="AGT")
    {
        count2++
        pack2=pack2+$8
        time2=$2
    }
}
END{
printf("The Throughput from n0 to n1: %f Mbps \n", ((count1*pack1*8)/(time1*1000000)));
printf("The Throughput from n1 to n2: %f Mbps", ((count2*pack2*8)/(time2*1000000)));
}
```

Step3: Run the simulation program

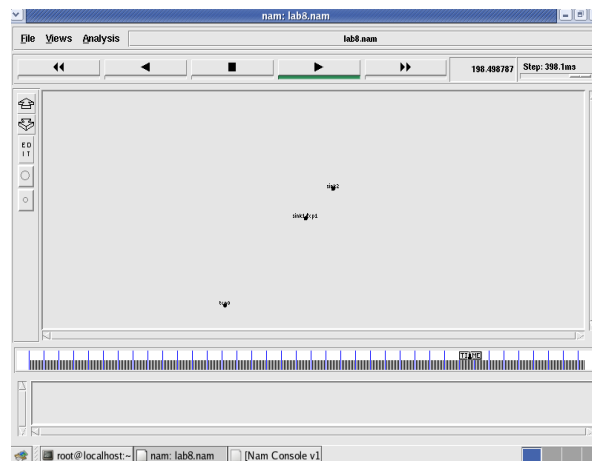
[root@localhost~]# ns prog4.tcl

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



Node 1 and 2 are communicating

Node 2 is moving towards node 3

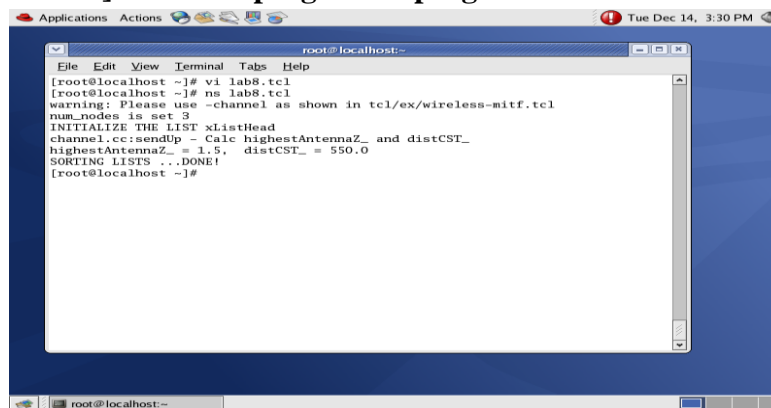


Node 2 is coming back from node 3 towards node 1

Step 4: Now press the play button in the simulation window and the simulation will begin.

Step 5: After simulation is completed run **awk** file to see the output ,

[root@localhost~]# awk -f prog4.awk prog4.tr



```

root@localhost:~
File Edit View Terminal Tabs Help
[root@localhost ~]# awk -f lab8.awk lab8.tr
The Throughput from n0 to n1: 5863.442245Mbps
The Throughput from n1 to n2: 1307.611834 Mbps[root@localhost ~]#

```

Step 6: To see the trace file contents open the file as ,

[root@localhost~]# vi prog4.tr

```

root@localhost:~
File Edit View Terminal Tabs Help
0.036400876 _0_ RTR --- 0 message 32 [0 0 0 0] ----- [0:255 -1:255 32 0]
r 0.037421112 _1_ RTR --- 0 message 32 [0 ffffffff 0 800] ----- [0:255 -1:255
32 0]
M 0.10000 0 (50.00, 50.00, 0.00), (50.00, 50.00), 15.00
M 0.10000 1 (100.00, 100.00, 0.00), (100.00, 100.00), 25.00
M 0.10000 2 (600.00, 600.00, 0.00), (600.00, 600.00), 25.00
s 0.182633994 _1_ RTR --- 1 message 32 [0 0 0 0] ----- [1:255 -1:255 32 0]
r 0.183694230 _0_ RTR --- 1 message 32 [0 ffffffff 1 800] ----- [1:255 -1:255
32 0]
s 0.882774710 _2_ RTR --- 2 message 32 [0 0 0 0] ----- [2:255 -1:255 32 0]
s 5.000000000 _0_ AGT --- 3 tcp 40 [0 0 0 0] ----- [0:0 1:0 32 0] [0 0] 0 0
r 5.000000000 _0_ RTR --- 3 tcp 40 [0 0 0 0] ----- [0:0 1:0 32 0] [0 0] 0 0
s 5.000000000 _0_ RTR --- 3 tcp 60 [0 0 0 0] ----- [0:0 1:0 32 1] [0 0] 0 0
s 5.000000000 _1_ AGT --- 4 tcp 40 [0 0 0 0] ----- [1:1 2:0 32 0] [0 0] 0 0
r 5.000000000 _1_ RTR --- 4 tcp 40 [0 0 0 0] ----- [1:1 2:0 32 0] [0 0] 0 0
r 5.004812650 _1_ AGT --- 3 tcp 60 [13a 1 0 800] ----- [0:0 1:0 32 1] [0 0] 1
0
s 5.004812650 _1_ AGT --- 5 ack 40 [0 0 0 0] ----- [1:0 0:0 32 0] [0 0] 0 0
r 5.004812650 _1_ RTR --- 5 ack 40 [0 0 0 0] ----- [1:0 0:0 32 0] [0 0] 0 0
s 5.004812650 _1_ RTR --- 5 ack 60 [0 0 0 0] ----- [1:0 0:0 32 0] [0 0] 0 0
r 5.006977357 _0_ AGT --- 5 ack 60 [13a 0 1 800] ----- [1:0 0:0 32 0] [0 0] 1
0
s 5.006977357 _0_ AGT --- 6 tcp 1040 [0 0 0 0] ----- [0:0 1:0 32 0] [1 0] 0 0
"lab8.tr" 128664L, 11456314C
1,1 Top

```

Here “M” indicates mobile nodes, “AGT” indicates Agent Trace, “RTR” indicates Router