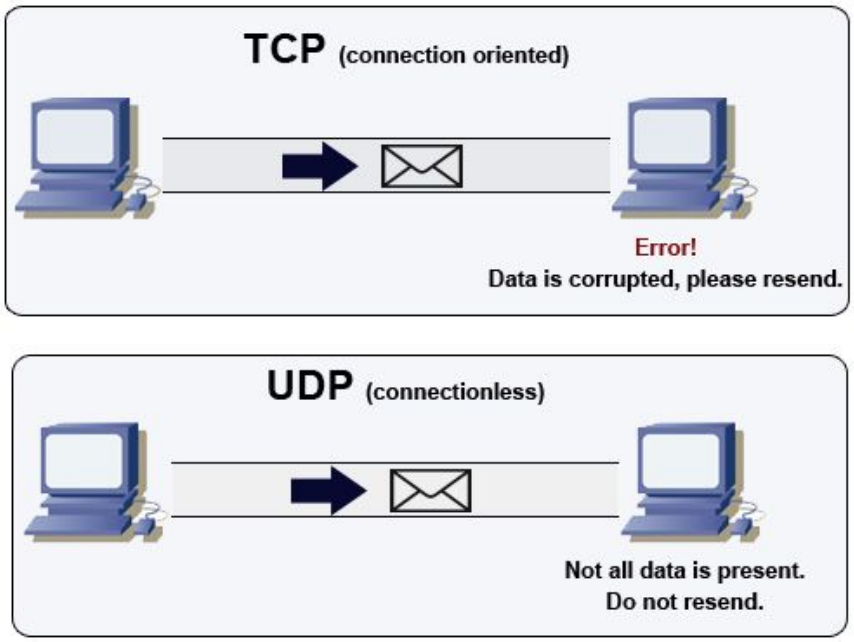
Name: Christine Polly **Roll no: 17**

Experiment No 9

Title:
UDP and TCP implementation using NS2

Theory:

In a wireless network, nodes communicate using the communication model that consists of UDP agent, Null agent, and CBR traffic. The sender node is attached to the UDP agent while the receiver node is attached to the Null agent. The connection between UDP agent and a Null agent is established using the keyword "connect". Transport agent (UDP) and application (CBR) are connected using the keyword "attach-agent". The CBR traffic object generates the data packet (traffic) based on a deterministic rate. The generated data packets are constant in size. The coding in sample4.tcl illustrates the data transmission between two nodes.



With the perception that the reader is having prior knowledge of NS2 I am posting the .tcl code for implementation directly. Use any linux platform istalled with NS2 to run these codes

Procedure/ Algorithm :

UDP	TCP
set ns [new Simulator] set nf [open udp.nam w]	set ns [new Simulator] set nf [open mi.nam w]

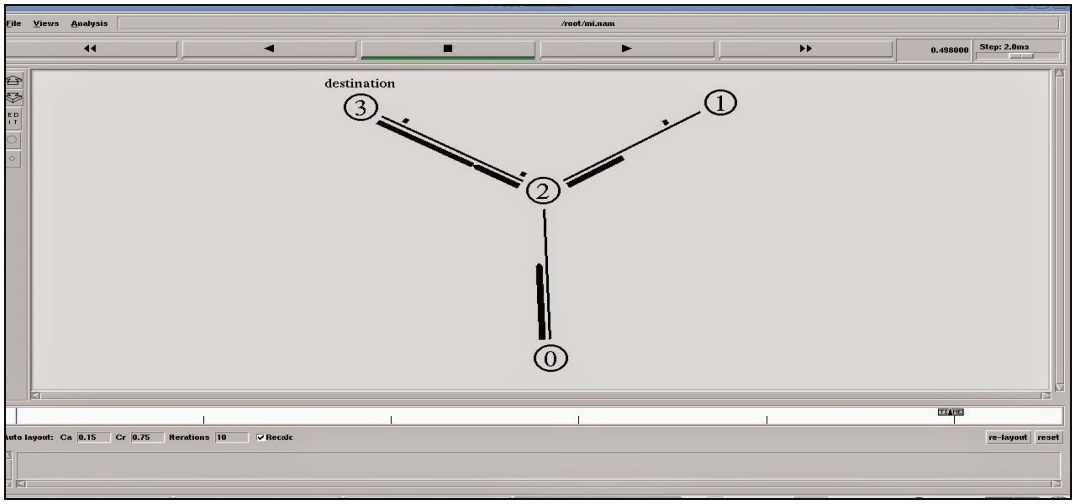
```
$ns namtrace-all $nf

set tf [open mo.tr w]
$ns trace-all $tf

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

set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
$n3 label “destination”

$ns duplex-link $n0 $n2 10Mb 1ms DropTail
$ns duplex-link $n1 $n2 10Mb 1ms DropTail
$ns duplex-link $n2 $n3 10Mb 1ms DropTail

$ns queue-limit $n0 $n2 10
$ns queue-limit $n1 $n2 10

set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0

set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0

set udp1 [new Agent/UDP]
$ns attach-agent $n1 $udp1
```

```
$ns namtrace-all $nf

set tf [open tcp.tr w]
$ns trace-all $tf

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

set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
$n3 label “destination”

$ns duplex-link $n0 $n2 10Mb 1ms DropTail
$ns duplex-link $n1 $n2 10Mb 1ms DropTail
$ns duplex-link $n2 $n3 10Mb 1ms DropTail


set tcp0 [new Agent/TCP]
$ns attach-agent $n0 $tcp0

set ftp0 [new Application/FTP]
$ftp0 set packet_Size_ 500
$ftp0 set interval_ 0.005
$ftp0 attach-agent $tcp0

set tcp1 [new Agent/TCP]
$ns attach-agent $n1 $tcp1

set ftp1 [new Application/FTP]
```

```
set cbr1 [new Application/Traffic/CBR]
$cbr1 attach-agent $udp1

set udp2 [new Agent/UDP]
$ns attach-agent $n2 $udp2

set cbr2 [new Application/Traffic/CBR]
$cbr2 attach-agent $udp2

set null0 [new Agent/Null]
$ns attach-agent $n3 $null0

$ns connect $udp0 $null0
$ns connect $udp1 $null0
$ns connect $udp2 $null0

$ns at 0.1 "$cbr1 start"
$ns at 0.2 "$cbr0 start"
$ns at 5 "finish"
$ns run
```

```
$ftp1 set packet_Size_ 500
$ftp1 set interval_ 0.005
$ftp1 attach-agent $tcp1

set sink0 [new Agent/TCPSink]
$ns attach-agent $n3 $sink0

set sink1 [new Agent/TCPSink]
$ns attach-agent $n3 $sink1

$ns connect $tcp0 $sink0
$ns connect $tcp1 $sink1

$ns at 0.1 "$ftp1 start"
$ns at 0.1 "$ftp0 start"
$ns at 5 "finish"
$ns run
```

The network animator will look something like this.



Refernce:

http://cs-pages.blogspot.com/2011/10/compare-and-contrast-advantages-and.html