

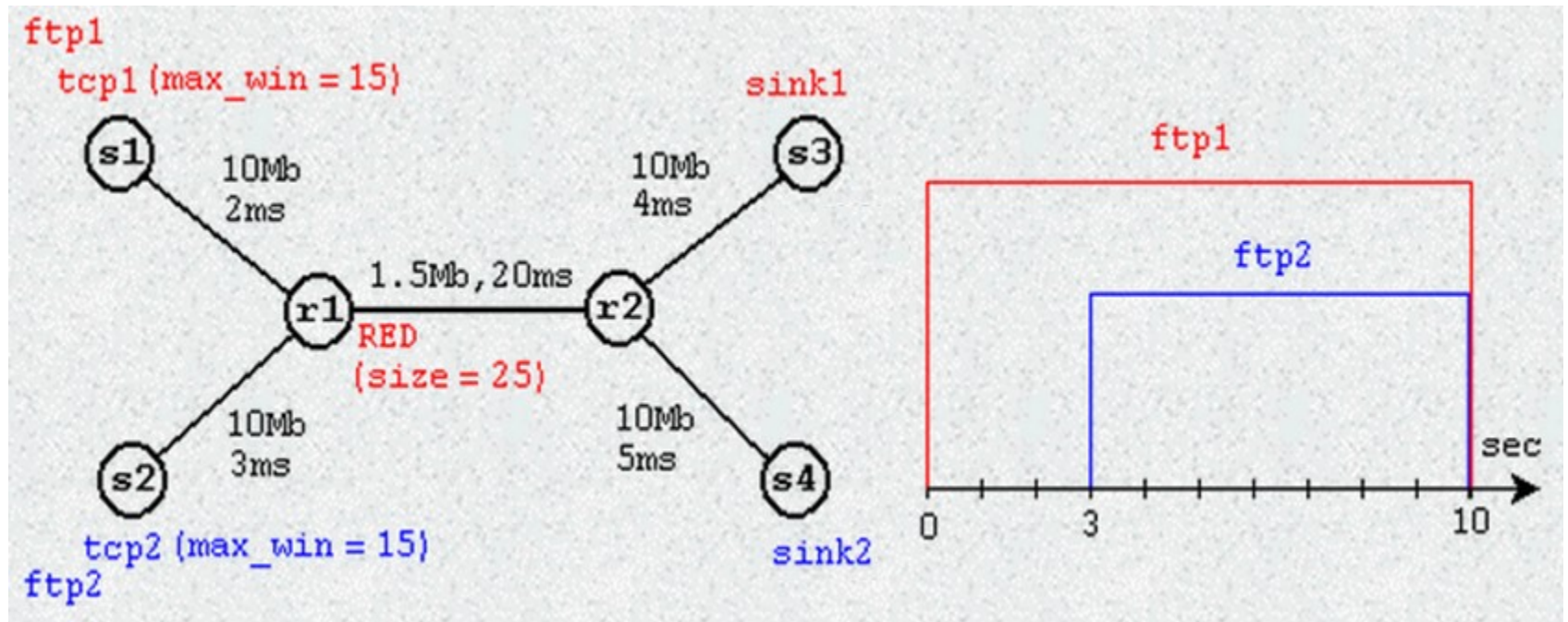
Προσομοίωση Δικτύων

5η Άσκηση

Study of TCP versions over FTP application

Δημιουργία multihop τοπολογίας

- Based on already existing example create the following network



Δημιουργία τοπολογίας

- #Create nodes
- set node_(s1) [\$ns node]
- set node_(s2) [\$ns node]
- set node_(r1) [\$ns node]
- set node_(r2) [\$ns node]
- set node_(s3) [\$ns node]
- set node_(s4) [\$ns node]

Δημιουργία τοπολογίας

#Create a duplex link between the nodes

```
$ns duplex-link $node_(s1) $node_(r1) 10Mb 2ms DropTail
```

```
$ns duplex-link $node_(s2) $node_(r1) 10Mb 3ms DropTail
```

```
$ns duplex-link $node_(r1) $node_(r2) 1.5Mb 20ms RED
```

```
$ns queue-limit $node_(r1) $node_(r2) 25
```

```
$ns queue-limit $node_(r2) $node_(r1) 25
```

```
$ns duplex-link $node_(s3) $node_(r2) 10Mb 4ms DropTail
```

```
$ns duplex-link $node_(s4) $node_(r2) 10Mb 5ms DropTail
```

Δημιουργία τοπολογίας

#better presentation for NAM

```
$ns duplex-link-op $node_(s1) $node_(r1) orient right-down
```

```
$ns duplex-link-op $node_(s2) $node_(r1) orient right-up
```

```
$ns duplex-link-op $node_(r1) $node_(r2) orient right
```

```
$ns duplex-link-op $node_(s3) $node_(r2) orient left-down
```

```
$ns duplex-link-op $node_(s4) $node_(r2) orient left-up
```

Προσθήκη για καλύτερη απεικόνιση

```
#Create a simulator object
```

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

```
#Define different colors for data flows (for NAM)
```

```
$ns color 1 Blue
```

```
$ns color 2 Red
```

Ροή Δεδομένων FTP over TCP

```
#Create data flows
```

```
set tcp1 [new Agent/TCP]
```

```
$tcp1 set window_ 10
```

```
$tcp1 set packetSize_ 700
```

```
$tcp1 set fid_ 1
```

```
$ns attach-agent $node_(s1) $tcp1
```

```
set sink1 [new Agent/TCPSink]
```

```
$ns attach-agent $node_(s3) $sink1
```

```
$ns connect $tcp1 $sink1
```

```
set ftp1 [new Application/FTP]
```

```
$ftp1 attach-agent $tcp1
```

Supported TCP Versions

- **Agent/TCP** - a “tahoe” TCP sender
- **Agent/TCP/Reno** - a “Reno” TCP sender
- **Agent/TCP/Newreno** - Reno with a modification
- **Agent/TCP/Sack1** - TCP with selective repeat (follows RFC2018)
- **Agent/TCP/Vegas** - TCP Vegas
- **Agent/TCP/Fack** - Reno TCP with “forward acknowledgment”
- **Agent/TCP/Linux** - a TCP sender with SACK support that runs TCP congestion control modules from Linux kernel

Supported TCP Versions

- **Agent/TCP** - a “tahoe” TCP sender
- **Agent/TCP/Reno** - a “Reno” TCP sender
- **Agent/TCP/Newreno** - Reno with a modification
- **Agent/TCP/Sack1** - TCP with selective repeat (follows RFC2018)
- **Agent/TCP/Vegas** - TCP Vegas
- **Agent/TCP/Fack** - Reno TCP with “forward acknowledgment”
- **Agent/TCP/Linux** - a TCP sender with SACK support that runs TCP congestion control modules from Linux kernel

Supported TCP Versions

```
Agent/TCP set window_ 20 ;# max bound on window size
Agent/TCP set windowInit_ 1 ;# initial/reset value of cwnd
Agent/TCP set windowOption_ 1 ;# cong avoid algorithm (1: standard)
Agent/TCP set windowConstant_ 4 ;# used only when windowOption != 1
Agent/TCP set windowThresh_ 0.002 ;# used in computing averaged window
Agent/TCP set overhead_ 0 ;# !=0 adds random time between sends
Agent/TCP set ecn_ 0 ;# TCP should react to ecn bit
Agent/TCP set packetSize_ 1000 ;# packet size used by sender (bytes)
Agent/TCP set bugFix_ true ;# see explanation
Agent/TCP set slow_start_restart_ true ;# see explanation
Agent/TCP set tcpTick_ 0.1 ;# timer granulatiry in sec (.1 is NONSTANDARD)
Agent/TCP set maxrto_ 64 ;# bound on RTO (seconds)
Agent/TCP set dupacks_ 0 ;# duplicate ACK counter
Agent/TCP set ack_ 0 ;# highest ACK received
Agent/TCP set cwnd_ 0 ;# congestion window (packets)
Agent/TCP set awnd_ 0 ;# averaged cwnd (experimental)
Agent/TCP set ssthresh_ 0 ;# slow-stat threshold (packets)
Agent/TCP set rtt_ 0 ;# rtt sample
Agent/TCP set srtt_ 0 ;# smoothed (averaged) rtt
Agent/TCP set rttvar_ 0 ;# mean deviation of rtt samples
Agent/TCP set backoff_ 0 ;# current RTO backoff factor
Agent/TCP set maxseq_ 0 ;# max (packet) seq number sent
```

Σενάριο **ΑΝΑΦΟΡΑΣ**

- Με βάση τις προηγούμενες διαφάνειες και το κώδικα που σας δίνεται στο eclass υλοποιήστε το δίκτυο που φαίνεται στην διαφάνεια 2
- Με βάση το αρχείο `delay_ftp.txt` που σας δίνεται στο eclass κάντε τις παρακάτω μετρήσεις

Agent/TCP

ΑΝΑΦΟΡΑ

Mean Delay

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Successfully Transmitted Packets

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Dropped Packets 0→4

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Dropped Packets 1→5

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Agent/TCP/Reno

ΑΝΑΦΟΡΑ

Mean Delay

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Successfully Transmitted Packets

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Dropped Packets 0→4

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Dropped Packets 1→5

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Agent/TCP/Vegas

ΑΝΑΦΟΡΑ

Mean Delay

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Successfully Transmitted Packets

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Dropped Packets 0→4

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Dropped Packets 1→5

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Agent/TCP/Linux

ΑΝΑΦΟΡΑ

Mean Delay

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Successfully Transmitted Packets

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Dropped Packets 0→4

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Dropped Packets 1→5

		Packet Size			
		40	200	500	900
Window	5				
	10				
	20				
	30				

Discussion

ΑΝΑΦΟΡΑ

- Σχολιάστε την επίδραση των παραμέτρων που χρησιμοποιήσατε
- Σχολιάστε την απόδοση των διαφορετικών εκδόσεων TCP στο συγκεκριμένο σενάριο