

## **EXPERIMENT 9**

### **Aim :**

Effect of VLAN on Network Performance – Multiple VLAN and Single Router.

### **Theory :**

A virtual LAN (VLAN) is any broadcast domain that is partitioned and isolated in a computer network at the data link layer (OSI layer 2). LAN is the abbreviation for local area network and in this context virtual refers to a physical object recreated and altered by additional logic. VLANs work by applying tags to network frames and handling these tags in networking systems - creating the appearance and functionality of network traffic that is physically on a single network but acts as if it is split between separate networks. In this way, VLANs can keep network applications separate despite being connected to the same physical network, and without requiring multiple sets of cabling and networking devices to be deployed.

VLANs allow network administrators to group hosts together even if the hosts are not directly connected to the same network switch. Because VLAN membership can be configured through software, this can greatly simplify network design and deployment. Without VLANs, grouping hosts according to their resource needs the labour of relocating nodes or rewiring data links. VLANs allow networks and devices that must be kept separate to share the same physical cabling without interacting, improving simplicity, security, traffic management, or economy. For example, a VLAN could be used to separate traffic within a business due to users, and due to network administrators, or between types of traffic, so that users or low priority traffic cannot directly affect the rest of the network's functioning. Many Internet hosting services use VLANs to separate their customers' private zones from each other, allowing each customer's servers to be grouped together in a single network segment while being located anywhere in their data centre. Some precautions are needed to prevent traffic "escaping" from a given VLAN, an exploit known as VLAN hopping.

To subdivide a network into VLANs, one configures network equipment. Simpler equipment can partition only per physical port (if at all), in which case each VLAN is connected with a dedicated network cable. More sophisticated devices can mark frames through VLAN tagging, so that a single interconnect (trunk) may be used to transport data for multiple VLANs. Since VLANs share bandwidth, a VLAN trunk can use link aggregation, quality-of-service prioritization, or both to route data efficiently.

**Code :**

```
set ns [new Simulator]
$ns color 1 Blue
$ns color 2 Red
set tracefile1 [open out.tr w]
set winfile [open winfile w]
$ns trace-all $tracefile1
set namfile [open out.nam w]
$ns namtrace-all $namfile
proc finish {} {
    global ns tracefile1 namfile
    $ns flush-trace
    close $tracefile1
    close $namfile
    exec nam out.nam &
    exit 0
}
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
set n4 [$ns node]
set n5 [$ns node]
$n1 color Red
$n1 shape box
$ns duplex-link $n0 $n2 2Mb 10ms DropTail
$ns duplex-link $n1 $n2 2Mb 10ms DropTail
$ns simplex-link $n2 $n3 0.3Mb 100ms DropTail
$ns simplex-link $n3 $n2 0.3Mb 100ms DropTail
```

```
set lan [$ns newLan "$n3 $n4 $n5" 0.5Mb 40ms LL Queue/DropTail MAC/Csma/Cd
Channel]

$ns duplex-link-op $n0 $n2 orient right-down
$ns duplex-link-op $n1 $n2 orient right-up
$ns simplex-link-op $n2 $n3 orient right
$ns simplex-link-op $n3 $n2 orient left
$ns queue-limit $n2 $n3 20

set tcp [new Agent/TCP/Newreno]
$ns attach-agent $n0 $tcp
set sink [new Agent/TCPSink/DelAck]
$ns attach-agent $n4 $sink
$ns connect $tcp $sink
$tcp set fid_ 1
$tcp set packet_size_ 552
set ftp [new Application/FTP]
$ftp attach-agent $tcp
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_ 0.01Mb
$cbr set random_ false
$ns at 0.1 "$cbr start"
$ns at 1.0 "$ftp start"
$ns at 124.0 "$ftp stop"
```

**Mohd. Aftab Alam**  
**02013302717**

```
$ns at 125.5 "$cbr stop"
proc plotWindow {tcpSource file} {
global ns
set time 0.1
set now [$ns now]
set cwnd [$tcpSource set cwnd_]
puts $file "$now $cwnd"
$ns at [expr $now+$time] "plotWindow $tcpSource $file"
}
$ns at 0.1 "plotWindow $tcp $winfile"
$ns at 125.0 "finish"
puts "running nam"
$ns run
```

### **Screen Shots :**



The screenshot shows a terminal window with a menu bar (File, Edit, Selection, View, Go, Run, Terminal, Help) and a sidebar with icons for Explorer, Search, Source Control, Run and Debug, and Extensions. The terminal output is as follows:

```
aftab@Dell16:~$ code lab9.tcl
aftab@Dell16:~$ ns lab9.tcl
warning: no class variable LanRouter::debug_
        see tcl-object.tcl in tclcl for info about this warning.
running nam
aftab@Dell16:~$ |
```

