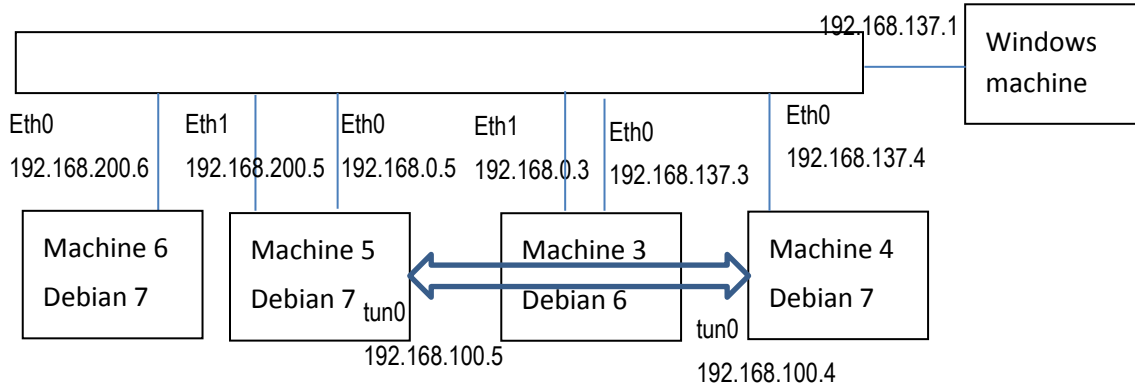


Simplemux tests for VoIP and a First Person Shooter game

Scenario: 4 VMWare machines running Debian7, in a Windows7



Simplemux runs between Machine 5 and Machine 4, through Machine 3.

Creating ip rules in Machine 5, in order to make the traffic go through the Simplemux tunnel:

Add a rule that makes the kernel route packets marked with 2 through table 3:

```
# ip rule add fwmark 2 table 3
```

Result:

```
# ip rule show
0:      from all lookup local
32765:  from all fwmark 0x2 lookup 3
32766:  from all lookup main
32767:  from all lookup default
```

Add a new route for table 3:

```
# ip route add default via 192.168.100.5 table 3
# ip route flush cache
```

Use iptables for sending certain ports to a destination

```
# iptables -t mangle -A PREROUTING -p udp --dport 9000 -j MARK --set-mark 2
# iptables -t mangle -A PREROUTING -p udp --dport 9001 -j MARK --set-mark 2
# iptables -t mangle -A PREROUTING -p udp --dport 9002 -j MARK --set-mark 2
# iptables -t mangle -A PREROUTING -p udp --dport 9003 -j MARK --set-mark 2
# iptables -t mangle -A PREROUTING -p udp --dport 9004 -j MARK --set-mark 2
# iptables -t mangle -A PREROUTING -p udp --dport 9005 -j MARK --set-mark 2
# iptables -t mangle -A PREROUTING -p udp --dport 9006 -j MARK --set-mark 2
# iptables -t mangle -A PREROUTING -p udp --dport 9007 -j MARK --set-mark 2
# iptables -t mangle -A PREROUTING -p udp --dport 9008 -j MARK --set-mark 2
# iptables -t mangle -A PREROUTING -p udp --dport 9009 -j MARK --set-mark 2
```

Set simplemux in Machine5 (ingress):

```
# ./simplemux -i tun0 -e eth0 -c 192.168.137.4 -P 10000 -r -l
log_ingress
```

Set simplemux in Machine4 (egress):

```
# ./simplemux -i tun0 -e eth0 -c 192.168.0.5 -r
```

Traffic generation

Traffic is generated from Machine 6 to Machine4, using [D-ITG](#).

VoIP:

```
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9000 VoIP -x G.729.2 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9001 VoIP -x G.729.2 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9002 VoIP -x G.729.2 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9003 VoIP -x G.729.2 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9004 VoIP -x G.729.2 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9005 VoIP -x G.729.2 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9006 VoIP -x G.729.2 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9007 VoIP -x G.729.2 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9008 VoIP -x G.729.2 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9009 VoIP -x G.729.2 &
```

Quake 3 (A First Person Shooter game):

```
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9000 Quake3 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9001 Quake3 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9002 Quake3 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9003 Quake3 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9004 Quake3 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9005 Quake3 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9006 Quake3 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9007 Quake3 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9008 Quake3 &
/home/proyecto/D-ITG-2.8.1-r1023/bin/ITGSend -a 192.168.137.4 -rp 9009 Quake3 &
```

The receiver is in Machine4:

```
/home/proyecto/D-ITG-2.8.1-r1023/bin# ./ITGRecv
```

Generating statistics in Machine5

```
# perl simplemux_throughput_pps.pl log_ingress 100000 rec native all all
# perl simplemux_multiplexing_delay.pl log_ingress output
```

Results

VoIP. 10 seconds of traffic of 10 G729 flows with 2 samples per packet

Bandwidth and PPS reductions

Average BW reduction: 36.50%

Average PPS reduction: 78.83%

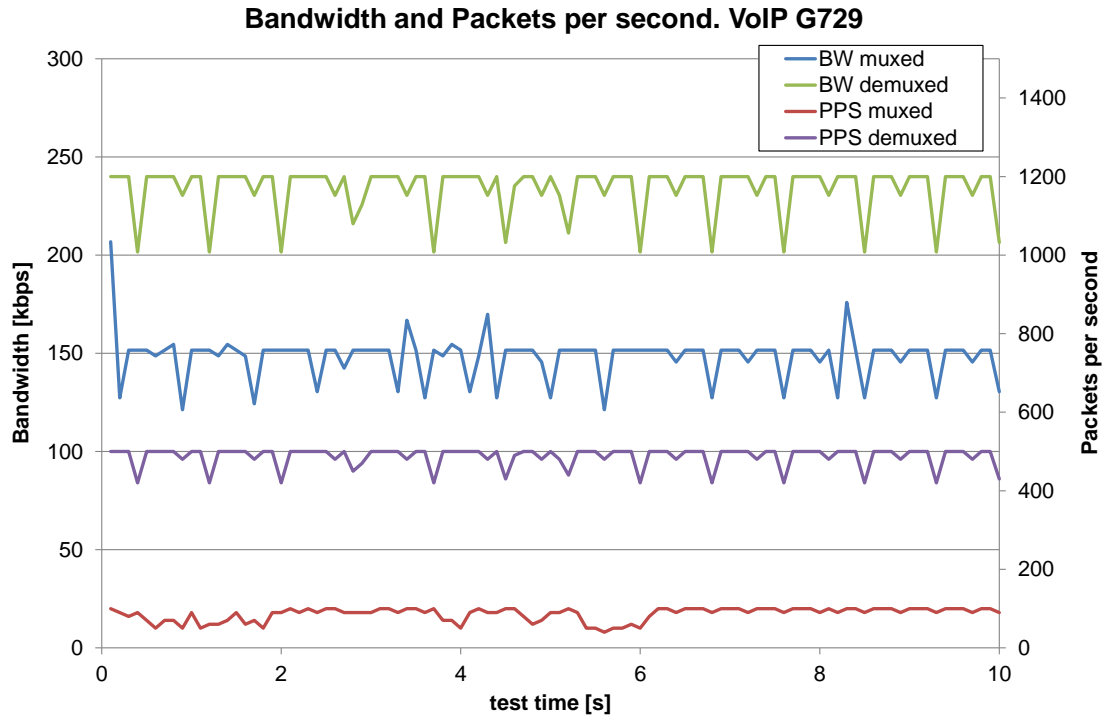


Fig. 1. BW and PPS for VoIP using a tick of 0.1 seconds

Multiplexing delay

total native packets sent: 4872

Average multiplexing delay: 4123.33 us

stdev of the multiplexing delay: 3845.13 us

Quake3. 10 seconds of traffic of 10 flows

Bandwidth and PPS reductions

Average BW reduction: 22.23%

Average PPS reduction: 92.99%

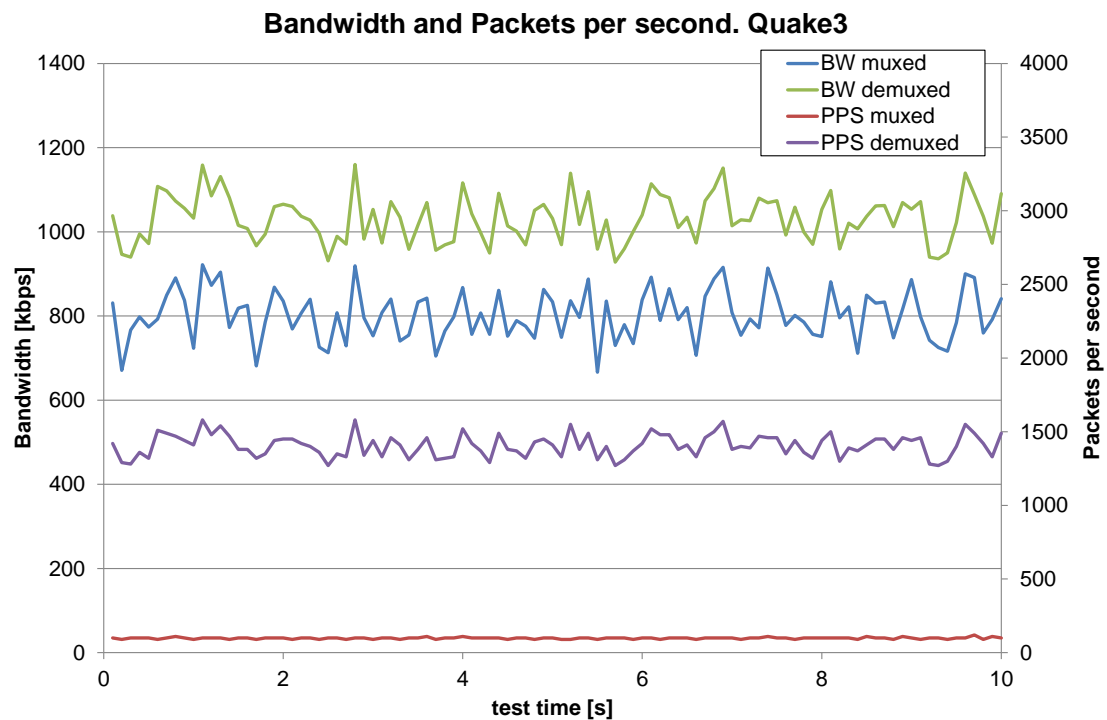


Fig. 2. BW and PPS for Quake 3 using a tick of 0.1 seconds

Multiplexing delay

total native packets: 14060

Average multiplexing delay: 5148.72 us

stdev of the multiplexing delay: 2991.43 us

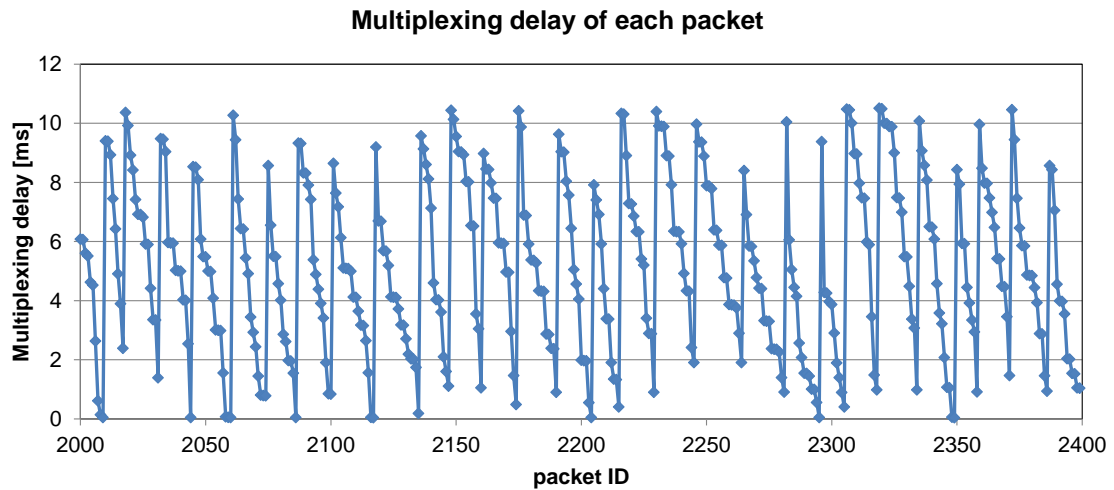


Fig. 3. Multiplexing delay of each packet for Quake 3

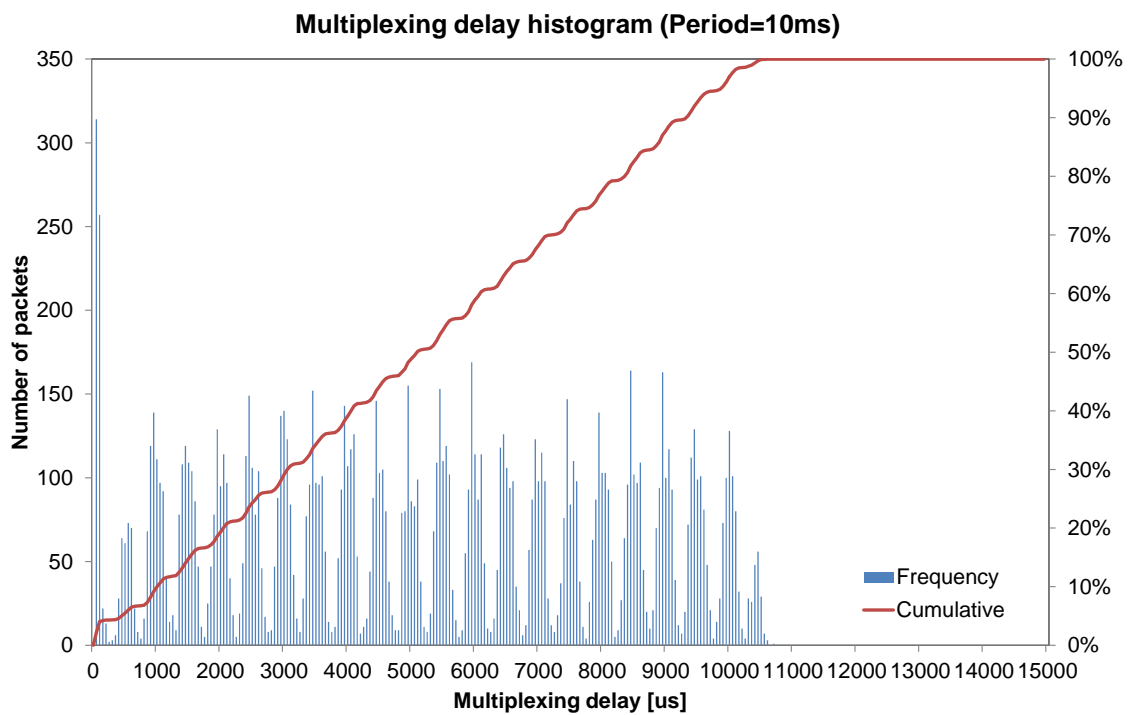


Fig. 4. Multiplexing delay histogram for Quake 3