

# A Performance Study of IPV6 Multicast Routing over a Dual Stack Virtual Local Area Network

Ahiaba Solomon

Department of Mathematics,  
Ahmadu Bello University Zaria,  
Nigeria

A. A. Obiniyi

Department of Mathematics,  
Ahmadu Bello University Zaria,  
Nigeria

ThankGod Sani Adeyi

Department of Mathematics,  
Ahmadu Bello University Zaria,  
Nigeria

## ABSTRACT

The shortage address space of the Internet Protocol Version 4 (IPv4) was the major factor that gave birth to the development of Internet Protocol Version 6 (IPv6). Initially, when the shortage alarmed, network experts used mechanisms like Classless Inter Domain Routing and Address Translation to remedies and slow down the exhaustion rate, yet the necessity for IPv6 could not be avoided. Finally, IPv6 emerged in 1998. Another issue however emerged as well, the migration to the new protocol. Since the arrival date up till now IPv6 is not the default Internet routing protocol. As result of the slow migration process, IPv6 has been running alongside with IPv4 on infrastructures partly meant for IPv4 in a dual stack network. The later and the former protocols employed multicast routing as integral of part their operations; this implies multicast routing is an integral part of the two protocols. This paper tests the performance IPv6 multicast routing over a dual stack virtual local area network. Graphical Network Simulator 3 was used to configure the network and Microsoft Hyper-V was used as the hypervisor on which the six virtual machines (hosts) reside. Parameters such as throughput, latency variations, data lost and the network over heads were measured. The experiment has shown that IPv6 multicast routing did not performed well running it alongside with IPv4, therefore, after more examination, running IPv6 multicast routing alongside IPv4 multicast in dual Stack network should be discontinued.

## Keywords

IPv4, IPv6, Unicast routing, Dual Stack network, Multicast routing, virtual machines, Graphical Network Simulator3 (GNS3), Protocol Independent Multicast-Sparse Mode (PIM-SM)

## 1. INTRODUCTION

### 1.1 Transition Mechanisms

The three basic transition mechanisms are the dual Stack, tunneling and the network Address Translation [6].

Dual Stack and tunneling are defined in RFC 4213 [10]. It provides a complete implementation of both versions of the Internet Protocol (IPv4 and IPv6). Tunneling on the other hand, carries IPv6 packets over unmodified IPv4 routing infrastructures. While Network address translation (NAT) is a method of remapping one IP address space into another by modifying network address information in Internet Protocol (IP) datagram packet headers even though they are in transit across a traffic routing devices. This method was initially used for ease of rerouting traffic in IP networks without renumbering every host. It has become a popular and essential tool in conserving global address space allocations in face of IPv4 address exhaustion [2].

### 1.1.1 Dual Stack Network

The most straightforward way for IPv6 nodes to remain compatible with IPv4-only nodes is by providing a complete IPv4 implementation. IPv6 nodes that provide complete IPv4 and IPv6 implementations are called IPv6/IPv4 nodes. IPv6/IPv4 nodes have the ability to send and receive both IPv4 and IPv6 packets [3]. They can directly interoperate with IPv4 nodes using IPv4 packets, and also directly interoperate with IPv6 nodes using IPv6 packets. Even though a node may be equipped to support both protocols, one or the other stack may be disabled for operational reasons. A stack being enabled has IP addresses assigned, but whether or not any particular application is available on the stacks is explicitly not defined [10]. Thus, IPv6/IPv4 nodes may be operated in either;

- a. With their IPv4 stack enabled and their IPv6 stack disabled.
- b. With their IPv6 stack enabled and their IPv4 stack disabled.
- c. With both stacks enabled.

IPv6/IPv4 nodes with their IPv6 stack disabled will operate like IPv4-only nodes. Similarly, IPv6/IPv4 nodes with their IPv4 stacks disabled will operate like IPv6-only nodes. IPv6/IPv4 nodes may provide a configuration switch to disable either their IPv4 or IPv6 stack [10].

### 1.2 Internet Protocol Version 4 (IPv4)

IPv4 is the fourth version of the Internet Protocol (IP). The specification was stated in RFC 791. IPv4 still dominate the network despite ongoing deployment of IPv6 [8].

IPv4 is a connectionless oriented protocol for use on packet-switch networks. It operates on a best effort delivery model, where there is no assurance for effective packet delivery, nor does it assure proper delivery sequence or avoid duplicate delivery [1]. It has proven to be robust, easy implementable and inter-operable.

#### 1.2.1 The Defects of IPv4

The major challenges of internet Protocol version 4 are the address exhaustion, less security, congestion, data loss and data priority [5]. These also include lack of its support for mobile applications.

The exhaustion of 232 IPv4 addresses was as a result of the explosion in the number of Internet users and Internet enabled electronic devices in the two decades [7]. IPv4 security issues rose as a result of packets not authenticated or encrypted when they are transmitted, network Congestion due to the broadcast feature, packet loss when the Time to Live (TTL) field in the header of an IP expired and the data was unable to get to the destination on time. The inability of IPv4 to prioritize data

with high transmitting priority constituted to the draw back.

The Internet Protocol version six tends to overcome these challenges.

### 1.3 The Synopsis of IPv6

Internet Protocol version six (IPv6) was created as the new version of the Internet Protocol to replace IPv4. It is fast, efficient, more secured, easy to configured, has support for mobile applications and can be configured automatically [4]. The header is simple and has a fixed length compare to the dynamic and bulky IPv4 header. IPv6 does not support broadcast address like that found in IPv4. IPv6 was specified in RFC 2460 [9].

## 2. MULTICAST EXPERIMENTAL SETUP IN DUAL STACK VIRTUAL LOCAL AREA NETWORK

Graphical Network Simulator 3 was used to configure the network logic (figure 2.1). and Microsoft hypervisor (Hyper-V) was used to setup the six virtual machines (hosts) which served as the network nodes and the host (physical computer) system runs Windows 10 Pro.

PIM sparse-mode was used to configure multicast-routing for both IPv4 and IPv6.

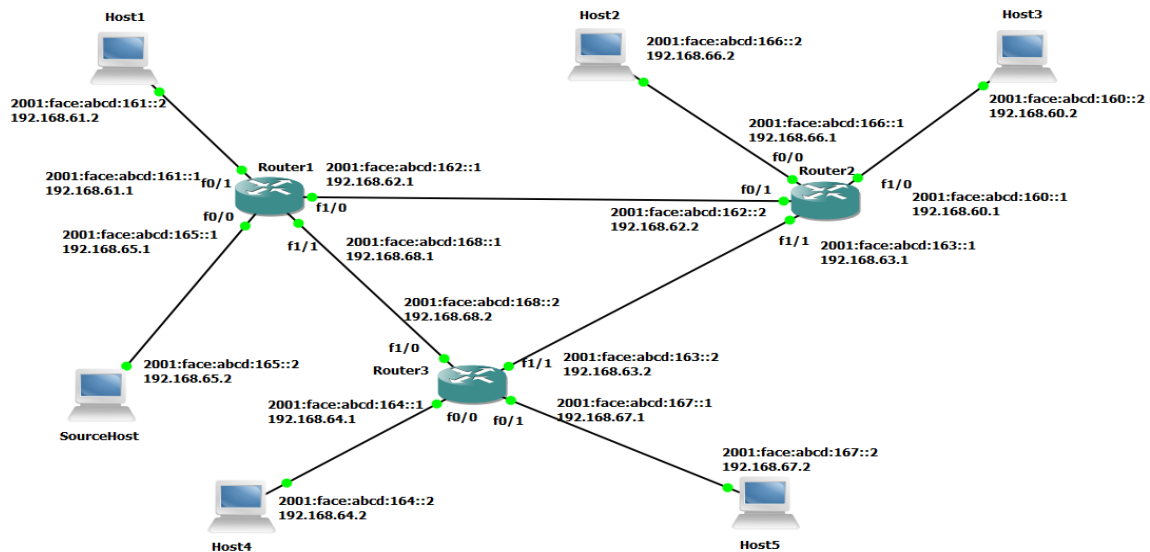


Figure 2.1. Dual Stack Network Interface Design

Figure 2.1 is a virtual local area network configured in GNS3. There are three 7200 series router and six hosts that will be participating in the multicast session. Unicast-routing, Multicast-routing, Open Shortest Path First version 2 (OSPFv2) were enabled for IPv4 unicast and multicast routings. IPv6 unicast-routing, IPv6 multicast-routing, IPv6 Open Shortest Path First version 3 (OSPFv3) were enabled on each of the router, since IPv6 does not support OSPFv2. Each of the interfaces in the router is configured with IPv4 unicast address, IPv6 global unicast address and Protocol Independent Multicast Sparse Mode (PIM-SM) enabled. Router 2 was configured as the Rendezvous point (RP) for both protocols. Each of the guest operating system (Arbitrarily named Host1 to 6) is connected to the Microsoft Hyper-V adapters on host operating system via the cloud in GNS3 and runs JPERF an Internet Performance working group (IPERF) front-end application for generating multicast traffics. JPERF is a graphical interface of IPERF developed in Java to assist those who are not conversant with the command line interpreter.

The network at this point was routable, that is, from Host1 you can reach all other Hosts and vice versa.

### 2.1 Setting up Multicast Routing in Dual Stack Network

Two testing durations were conducted; three ten minutes runs and three one hour runs.

Three test scenarios were carried out, sending;

- IPv4 multicast traffics only on the dual network,

where Host2, Host3, Host4 and Host5 joined the multicast group 239.5.5.5 and Source Host was chosen arbitrarily (not configured) as the source of the multicast traffic (Table 3.1).

- IPv6 multicast traffics only on the dual network where Host1, Host2, Host3, Host4 and Host5 joined the multicast group ff7e:240:2001:face:abcd:2:0:2 and Source Host was chosen arbitrarily (not configured) as the source of the multicast traffic (Table 3.2).
- Both Ipv4 and IPv6 multicast datagram simultaneously on the dual network. Where Host2 and Host4 joined IPv4 multicast group 239.10.10.10 to receive datagram from SourceHost, and Host3 and Host5 Joined IPv6 multicast group ff7e:240:2001:face:abcd:2:0:3 to receive datagram sent by Host1 (Table 3.3).

## 3. RESULT ANALYSIS

Table 3.1 shows the average jitters of running IPv4 only on a Dual Stack network ten minutes and one hour runs, while table 3.2 display average jitters of running IPv6 only on a Dual Stack network ten minutes and one hour runs. Figure 3.1 compares the average ten minutes runs IPv4 and IPv6 on dual stack virtual network. Figure 3.2 represent the one hour running duration. IPv6 has poor latencies across all the hosts on Dual Stack network ten minutes and one hour multicast streaming.

**Table 3.1: Test result for IPv4 only listeners, listening to 239.5.5.5, 10 min and 1hr streams**

	Jitters of Receiving hosts				
Sender (Source-Host)	Host1	Host2	Host3	Host4	Host5
10 min Run	2.043	1.224	1.489	1.454	1.422
10 min 2 <sup>nd</sup> Run	3.837	4.470	3.864	2.300	1.786
10 min 3 <sup>rd</sup> Run	0.190	2.734	1.541	3.423	2.241
<b>Average of the three 10 min Run</b>	<b>2.023</b>	<b>2.809</b>	<b>2.298</b>	<b>2.392</b>	<b>1.816</b>
1hr Run	3.999	3.144	2.732	3.116	4.028
1hr 2 <sup>nd</sup> Run	3.098	1.154	1.223	0.539	4.154
1hr 3 <sup>rd</sup> Run	0.871	1.665	2.355	0.947	0.316
<b>Average of the three 1hr Run</b>	<b>2.656</b>	<b>1.987</b>	<b>2.103</b>	<b>1.534</b>	<b>2.832</b>

**Table 3.2: Test result for IPv6 only listeners receiving from ff7e:240:2001:face:abcd:2:0:2 10 min and 1hr stream**

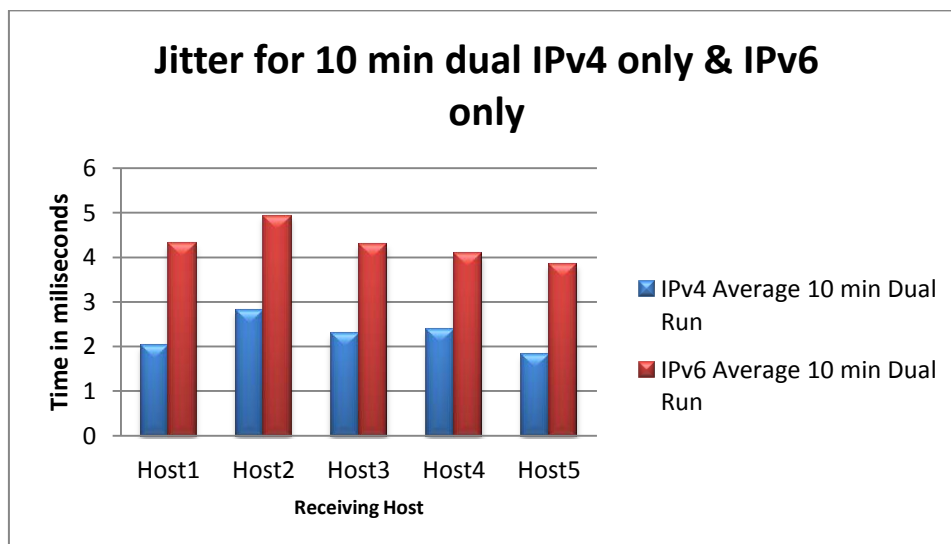
	Jitters of Receiving hosts				
Sender (Source-Host)	Host1	Host2	Host3	Host4	Host5
10 min Run	4.681	4.918	5.483	6.176	4.681
10 min 2 <sup>nd</sup> Run	3.101	4.978	3.567	2.159	3.358
10 min 3 <sup>rd</sup> Run	5.166	4.904	3.864	3.925	3.532
<b>Average of the three 10 min Run</b>	<b>4.316</b>	<b>4.933</b>	<b>4.304</b>	<b>4.086</b>	<b>3.857</b>
1hr Run	3.215	3.223	1.951	4.942	3.948
1hr 2 <sup>nd</sup> Run	5.130	4.331	4.439	7.384	3.748
1hr 3 <sup>rd</sup> Run	4.523	6.783	4.965	3.959	4.952
<b>Average of the three 1hr Run</b>	<b>4.289</b>	<b>4.779</b>	<b>3.785</b>	<b>5.428</b>	<b>4.216</b>

**Table 3.3: Test result for two groups with two listeners each for IPv4 and IPv6, 10 min and 1hour stream**

Durations	Jitters of Receiving hosts			
	Source-Host multicast to 239.10.10.10		Host1 multicast to ff7e:240:2001:face:abcd:2:0:3	
	Host2	Host4	Host3	Host5
10 min Run	2.868	5.738	5.048	4.246
10 min 2 <sup>nd</sup> Run	3.58	3.473	3.016	3.640
10 min 3 <sup>rd</sup> Run	4.858	3.295	6.069	3.620
<b>Average of the three 10 min Run</b>	<b>3.768</b>	<b>4.168</b>	<b>4.708</b>	<b>3.835</b>
1hr Run	3.542	3.35	4.622	4.582
1hr 2 <sup>nd</sup> Run	5.492	3.675	4.913	5.310
1hr 3 <sup>rd</sup> Run	5.090	3.990	3.753	2.757
<b>Average of the three 1hr Run</b>	<b>4.708</b>	<b>3.671</b>	<b>4.429</b>	<b>4.216</b>

Table 3.3 compare the average jitters of running IPv4 multicast simultaneously with IPv6 multicast on a Dual Stack network for the ten minutes and one hour duration of the same run. Figure 4.12 and figure 3.3 represent the graph of table

3.3. Both IPv6 and IPv4 have higher jitters in Dual stack network compare to running two multicast groups simultaneously in IPv4 only network and IPv6 only network.



**Figure 3.1 Comparing Average Jitter for IPv4 only & IPv6 only dual networks 10 min run**

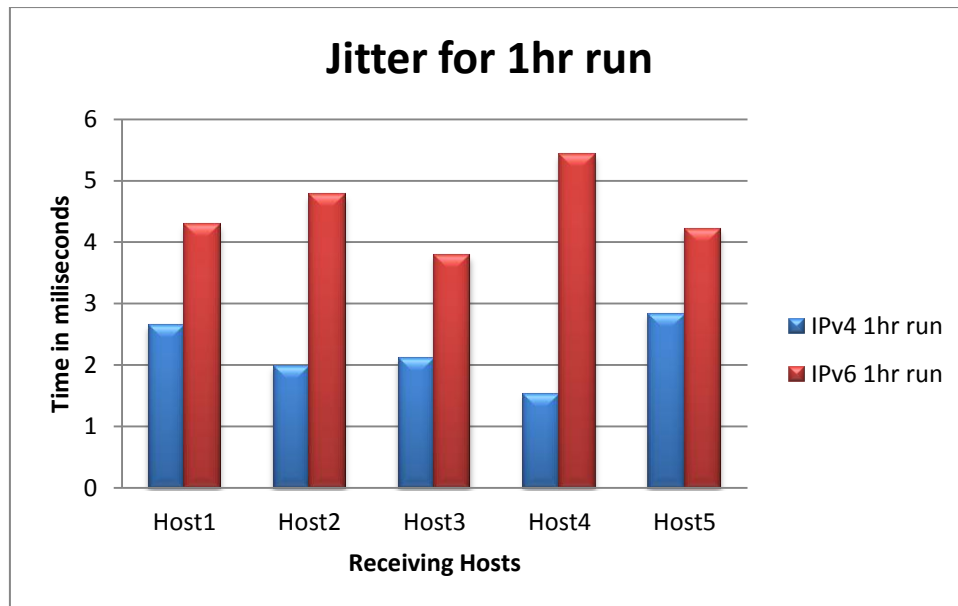


Figure 3.2 Comparing Average Jitter for IPv4 only & IPv6 only dual networks 1hr run

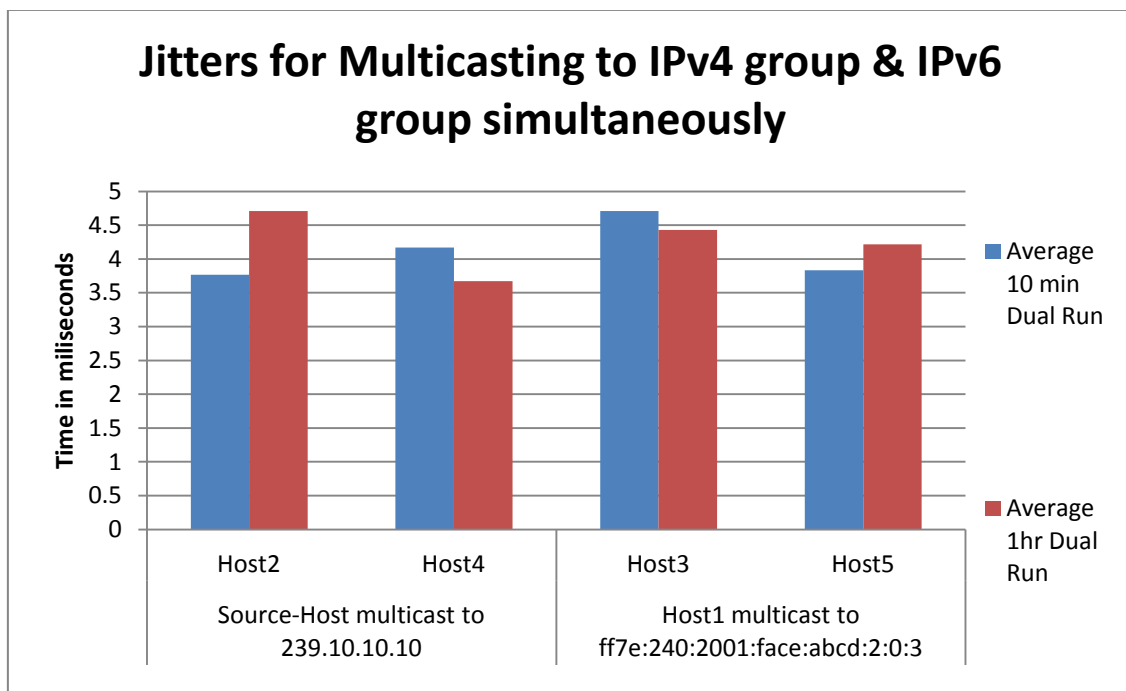


Figure 3.3 Jitters for Multicasting to IPv4 group & IPv6 group simultaneously

### 3.1 Data Loss, Throughput and Latency

From the result obtained from all the average 10 minutes and average 1 hour runs, there was no datagram lost. The throughput was 100% and the jitter range from 0 to 7.384ms.

### 3.2 Protocol Overheads

The following are the protocols running at the background, no other protocol overhead noticed. (Figure 3.4 deduced from the Wire shark captures).

a. PIM-SM was used as the multicast routing protocol. The protocol did not produce much of an overhead. The PIMv2 Hello messages were sent out at irregular intervals.

b. Open Shortest Path First (OSPF) was used as the unicast routing protocol. It sends updates and acknowledgement messages to all routers multicast address ff02::5 at a regular interval 30ms.

c. Internet Control Message Protocol version six (ICMPv6) neighbour solicitation and neighbour advertisement messages were also noticed.

IPv6 fragmented packet offsets messages were sent to the embedded rendezvous point address ff7e:240:2001:face:abcd:2:0:2 about every 0.020767ms or less.

Time	Source	Destination	Protocol	Length	Info
60 0.217653000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	IPv6	1510	IPv6 fragment (nxt=UDP (17) off=0 id=0x2003f2)
61 0.217653000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	UDP	92	Source port: 54840 Destination port: 5001
62 0.217653000	192.168.65.2	239.10.10.10	UDP	1512	Source port: 63803 Destination port: 5001
63 0.217653000	192.168.65.2	239.10.10.10	UDP	1512	Source port: 63803 Destination port: 5001
64 0.233293000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	IPv6	1510	IPv6 fragment (nxt=UDP (17) off=0 id=0x2003f4)
65 0.233293000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	UDP	92	Source port: 54840 Destination port: 5001
66 0.233293000	fe80::c802:1cff:fe64:6	ff02::d	PIMv2	104	Assert
67 0.248903000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	IPv6	1510	IPv6 fragment (nxt=UDP (17) off=0 id=0x2003f6)
68 0.248903000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	UDP	92	Source port: 54840 Destination port: 5001
69 0.248903000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	IPv6	1510	IPv6 fragment (nxt=UDP (17) off=0 id=0x2003f8)
70 0.248903000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	UDP	92	Source port: 54840 Destination port: 5001
71 0.248903000	192.168.65.2	239.10.10.10	UDP	1512	Source port: 63803 Destination port: 5001
72 0.268789000	192.168.65.2	239.10.10.10	UDP	1512	Source port: 63803 Destination port: 5001
73 0.270802000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	IPv6	1510	IPv6 fragment (nxt=UDP (17) off=0 id=0x2003fa)
74 0.270802000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	UDP	92	Source port: 54840 Destination port: 5001
75 0.270802000	192.168.65.2	239.10.10.10	UDP	1512	Source port: 63803 Destination port: 5001
76 0.286442000	192.168.65.2	239.10.10.10	UDP	1512	Source port: 63803 Destination port: 5001
77 0.286442000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	IPv6	1510	IPv6 fragment (nxt=UDP (17) off=0 id=0x2003fc)
78 0.286442000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	UDP	92	Source port: 54840 Destination port: 5001
79 0.286442000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	IPv6	1510	IPv6 fragment (nxt=UDP (17) off=0 id=0x2003fe)
80 0.286442000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	UDP	92	Source port: 54840 Destination port: 5001
81 0.302063000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	IPv6	1510	IPv6 fragment (nxt=UDP (17) off=0 id=0x200400)
82 0.302063000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	UDP	92	Source port: 54840 Destination port: 5001
83 0.302063000	192.168.65.2	239.10.10.10	UDP	1512	Source port: 63803 Destination port: 5001
84 0.302063000	192.168.65.2	239.10.10.10	UDP	1512	Source port: 63803 Destination port: 5001
85 0.317688000	192.168.65.2	239.10.10.10	UDP	1512	Source port: 63803 Destination port: 5001
86 0.317688000	192.168.65.2	239.10.10.10	UDP	1512	Source port: 63803 Destination port: 5001
87 0.317688000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	IPv6	1510	IPv6 fragment (nxt=UDP (17) off=0 id=0x200402)
88 0.317688000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	UDP	92	Source port: 54840 Destination port: 5001
89 0.317688000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	IPv6	1510	IPv6 fragment (nxt=UDP (17) off=0 id=0x200404)
90 0.317688000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	UDP	92	Source port: 54840 Destination port: 5001
91 0.348991000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	IPv6	1510	IPv6 fragment (nxt=UDP (17) off=0 id=0x200406)
92 0.348991000	2001:face:abcd:160:897:636f:c051:ca82	ff7e:240:2001:face:abcd:2:0:3	UDP	92	Source port: 54840 Destination port: 5001
93 0.348991000	fe80::c802:1cff:fe64:8	ff02::5	OSPF	186	LS Update
94 0.348991000	192.168.65.2	239.10.10.10	UDP	1512	Source port: 63803 Destination port: 5001

Figure 3.4: Wire Shark Data Capture to Shows Protocols Over Heads

#### 4. TOOLS USED

- The virtual machines run Windows 7 professional edition
- Microsoft Hyper-V (Hypervisor)
- Graphical Network Simulator 3 (Configured the logic of the virtual network with 3 routers 7200 series)
- Wire Shark (A network protocol analyzer).
- JPERF (Traffic Generator)
- Host system (Intel Core i5, Windows 10 Pro, RAM 6GB, CPU 2.4GHz, Hard Disk 750 GB)

#### 5. CONCLUSION

From the experiment, IPV4 and IPV6 multicast routing over a dual virtual network has 100% throughput, since no significant data lost was noticed in all the experimented duration and scenarios. In terms of jitters (variations in latency), running IPV4 multicast routing only in dual network performed better than running IPV6 multicast routing in a dual setup, but, the jitters when multicasting to IPV4 group and IPV6 group simultaneously in a dual virtual network has no clear difference as it fluctuates from hosts across the two protocols. The fast and efficient attributes of running IPV6 multicast in IPV6 only network was not noticed in the demonstration among the six hosts that participated in multicast routing on a dual stake virtual local area network. From the experiment it is clear that IPV6 multicast routing did not out played IPV4 multicast routing both on a dual stake network. Therefore, after more scrutiny, we recommend that running IPV6 multicast routing alongside IPV4 multicast in dual Stack network should be discontinued. Hence, the need for IPV6 alone network is inevitable. This experiment has the following limitations;

- The experiment was carried out in a virtual lab not a real lab.
- The datagram tested are not real application data.
- The resources available for the virtual machines depend on resource of the host system. Using more resourceful host may better performance.

#### 6. REFERENCES

- Abhilash, B., Narjis, H., Mehra, P. (2014) Communicating between IPv4 and IPv6. International Journal of Advanced Research in Computer Science and Software Engineering, 4(3), 1064 – 1071
- Frankel, S., Grave, R., Pearce, J., Rooks, M. (2010). Guidelines for Secured Deployment of IPv6. . Retrieved on March 17, 2014, from <http://csrc.nist.gov/publications/nistpubs/800-119/sp800-119.pdf>
- Galligan, R. (1996). Transition Mechanisms for IPv6 Hosts and Routers. Retrieved on May 27, 2014, from <https://tools.ietf.org/html/rfc1933>
- IBM (2012). IPv6 Introduction and Configurations. Retrieved on May11, 2014, from <http://www.redbooks.ibm.com/redpapers/pdfs/redp4776.pdf>
- Olubenjo, B., Al-Debagy, U. (2014): A Comparative Review Of Internet Protocol Version 4 (IPv4) and Internet Protocol Version 6 (IPv6). International Journal of Computer Trends and Technology (IJCTT) – volume 13 number 1 – July 2014 ISSN: 2231-5381
- Sellers, C. (2009). IPv6 Transition Mechanisms. . Retrieved 17 March, from <http://www.rmv6tf.org/wp-content/uploads/2012/11/Chuck-Sellers-090421-IPv6->

Transition-Mechanisms-Sellers1.pdf

- [7] Sriraman, A., Butler, K., McDaniel, P., Raghavan, P. (2013). Analysis of the IPv4 Address Space Delegation Structure. Retrieved on June 16, 2016, from <http://ix.cs.uoregon.edu/~butler/pubs/iscc07.pdf>
- [8] RFC 791, (1981). Internet Protocol Specification. Retrieved March 17, 2014 from <https://tools.ietf.org/html/rfc791>
- [9] RFC 2460 (2006): IP Version 6 Addressing Architecture. Retrieved 17 March, 2014 from <http://tools.ietf.org/html/rfc4291>
- [10] RFC 4213 (2005): Basic IPv6 Transition Mechanisms. Retrieved on the 18 January, 2016 from <https://tools.ietf.org/html/rfc4213>