International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 11 – No. 3, August 2016 – www.ijais.org

An Empirical Study of IPV6 Multicast Routing over a Virtual Local Area Network

A. A. Obiniyi
Department of Mathematics,
Ahmadu Bello University Zaria, Nigeria

Ahiaba Solomon
Department of Mathematics,
Ahmadu Bello University Zaria, Nigeria

ABSTRACT
The Internet Protocol Version 6 was designed to efficiently improve on the existing functionalities of Internet Protocol Version 4 and to introduce new constructs that it lacks. Though IPv6 is not an extension of IPv4, as the two protocols have different specifications. Both the new and the formal protocol use multicast routing for many of their operations; this implies multicast routing is core in the protocols. This experiment became imperative especially in this era everyone is looking forward to using IPv6 as the default network. This paper tested the performance of IPv6 multicast routing over a 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 loss and the network over heads were examined. The experiment has shown that, IPv6 multicast routing over a virtual network has 100% throughput, the jitter (variations in latency) varies among the hosts in all the running scenarios, but low and stabled jitters were noticed as the running duration increases and the number of streaming increase from one multicast stream to running two multicast streams simultaneously. There was no data loss.

Keywords
IPv4, IPv6, Unicast routing, Multicast routing, virtual machines, Graphical Network Simulator 3(GNS3), Protocol Independent Multicast-Sparse Mode (PIM-SM),Multicast Listener Discovery (MLD), Microsoft Hyper-V.

1. INTRODUCTION
The major changes from IPv4 to IPv6 fall primarily into the following categories; expanded addressing capabilities, header format simplification, improved support for extensions and options, flow labeling capability [2]. The challenges of IPv6 are directly connected to its protocol stack specification (multicast and header specification), [10]. This means multicast routing contributed to challenges facing IPv6 as a result of new it assume which replaces broadcast. Therefore, testing IPv6 multicast routing became very necessary in this era of IPv4 to IPv6 transition.

1.1 Summary of IPv6
Internet Protocol Version 6 (IPv6) which is a new version of the Internet Protocol, was specified in [8]. It was first introduced in 1998 by the Internet Engineering Task Force (IETF) in order to replace IPv4.

The new protocol has 2128 addresses compare to 232 addresses of IPv4, it is fast, efficient, more secured and support mobile applications [3].

1.2 The IPv6 Header
The standard specification for IPv6 header, according to [8] is displayed in figure 1.1

Version	Traffic class	Flow Label
Playload Length	Next Header	Hop Limit
Source Address		
Destination Address		

Figure 1.1 The IPv6 Header [11]

IPv6 does not support broadcast address as it is in IPv4, this functionality was replaced by some IPv6 multicast addresses. The specific use of IPv6 addresses based on RFC 3513 [6] is shown in table 1.1. The IPv6 addressing architecture which was initially explained in RFC 3513, now obsolete, is now specified in [9], [4].

Table 1.1: Specific Use of IPv6 [8]

Address type	Binary prefix	IPv6 notation
Unspecified	00…0 (128 bits)	::/128
Loopback	00…1 (128 bits)	::1/128
Multicast	11111111	FF00::/8
Link-local unicast	1111111010	FE80::/10
Site-local unicast	1111111011	FEC0::/10
Global unicast	Everything else	Everything else

1.3 Succinct Discussion of IPv6 Multicast
IPv6 multicast addresses which was defined in IP Version 6 Addressing Architecture [9]. Multicast is triggered by the receivers' interest. Multicast group contain arbitrary group of receivers that shows interest in receiving a particular multicast datagram. The members can be located anywhere on the Internet or in any private network without any geographical constrain. Receivers that wish to receive multicast data streamed to a particular group have to join the group by sending a Multicast Listeners Discovery (MLD) message to the router they are connected to. Routers use the MLD protocol to learn whether members of a group are present on their directly attached subnets. Hosts join multicast groups by sending MLD report messages [4].

A multicast group address is selected for the members in a multicast group. This group address is use as the destination address by the sender of a multicast datagram to reach all members who have joined the group or shown interest in receiving the datagram.

Membership in a multicast group is dynamic; hosts can join and

International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 11 – No. 3, August 2016 – www.ijais.org

leave at will. In IPv6, multicast address is an identifier for a set of interfaces that typically belong to different nodes and prefixed by ff00::/8 (1111 1111). A packet destined to a multicast address is delivered to all the interfaces identified by this address using best-effort reliability (there is no datagram safety guarantee) [1]. Figure 1.2 shows the format of the IPv6 multicast address.

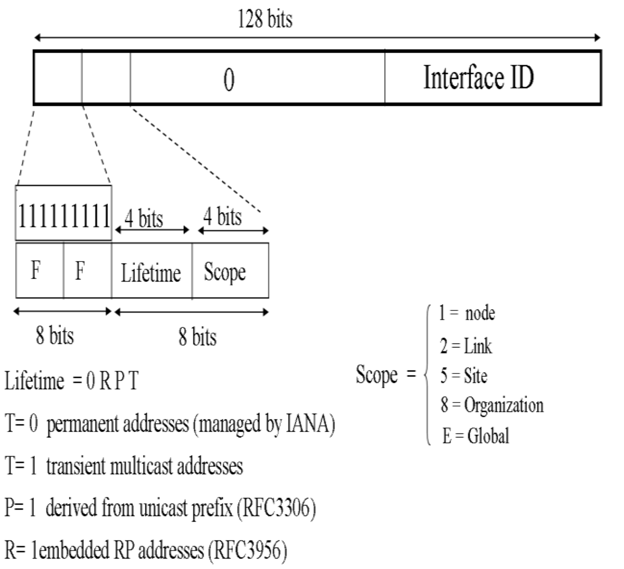

Figure 1.2: IPv6 Multicast Address Format [9]

1.4 Concept of Virtualization

In computing, virtualization means creating a virtual (rather than actual) version of something. These include virtual network resources, virtual storage devices, virtual operating system and virtual hardware platform.

Normally, every physical computer has one instance of the operating system which supports one or more application programs; a single physical computer runs software that abstracts the physical computer's resources so that they may be shared between multiple "virtual computers" in a virtualization environment [7]. Each virtual computer may be running a different operating system from all of the other virtual machines on the physical machine. A crash or other program error on any of the virtual machines leaves all of the other virtual machines unaffected [5].

In hardware virtualization, virtualization takes place on the host (physical computer), and the guest which is the machine sitting on the host, is the virtual machine. Host and guest are used to differentiate the program set that runs on the physical machine from that which runs on the virtual machine. With the help of a Hypervisor/Virtual Machine Manager virtual machines can be created (Figure 1.3).

Virtualization can be full, partial, or Para virtualization, these depend on the hardware environment is simulated or whether the guest OS is modified or not.

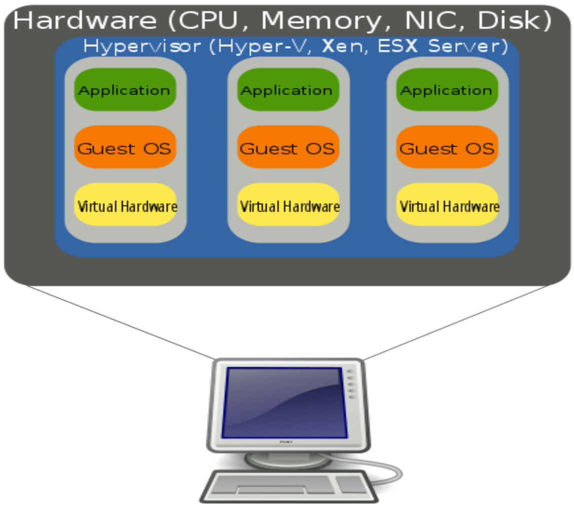

Figure 1.3 Logical Diagram of Full Virtualization [12]

2. EXPERIMENTAL SETUP

Graphical Network Simulator 3 was used to configure the network. Microsoft Hyper-V was used as the hypervisor on which the six virtual machines (hosts) reside and a host system running Windows 10 Professional (figure 2.1).

2.1 System Architecture

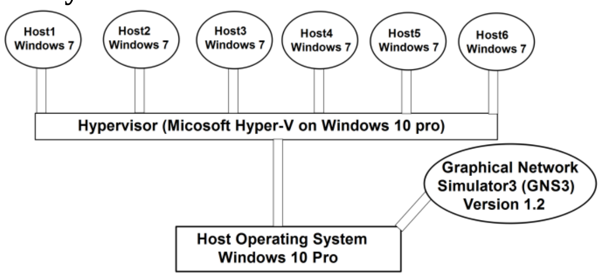

Figure 2.1: System Architecture

2.2 IPv6 Interface Design

Figure 2.2 is a virtual network design in GNS3 (Graphical Network Simulator 3). There are three 7200 series router and six hosts that participated in the multicast session. IPv6 unicast-routing, IPv6 multicast-routing, IPv6 Open Shortest Path First (OSPFv3) were all enable on each of the router. Each of the interfaces in the router is configured with IPv6 global unicast address and Protocol Independent Multicast Sparse Mode (PIMSM) enabled. Router 2 loopback0 interface was configured as the Rendezvous point (RP) address. Each of the guest operating system (Arbitrarily named Host1 to Host6) is connected to the Microsoft Hyper-V adapters on host operating system (figure 2.1) via the cloud and runs JPERF (A Java front-end of Iperf, an application for generating multicast traffics).

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

International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 11 – No. 3, August 2016 – www.ijais.org

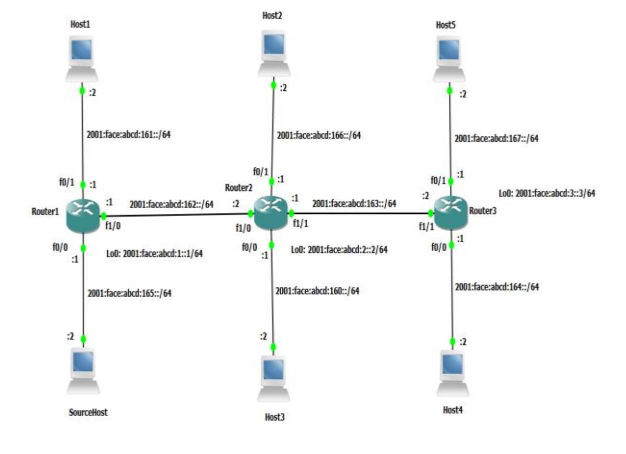

Figure 2.2 IPv6 network interface design

2.3 Setting Up IPv6 Multicast Routing

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

In the first setup, all the five hosts joined one multicast group.

a. SourceHost was chosen arbitrary (not configured) as the source of the multicast traffic.

b. Host1, Host2, Host3, Host4 and Host5 joined the multicast group ff7e:240:2001:face:abcd:2:0:2

c. SourceHost send packets to this multicast address ff7e:240:2001:face:abcd:2:0:2.

d. Figure 2.3 shows one of the listeners waiting for packets

e. Figure 2.4 shows the responses from the five listeners.

f. All other settings for JPERF take the default values.

g. Table 3.1 shows the average jitters of the five hosts that received a multicast datagram for ten minutes and one hour running and figure 3.1 shows the graph.

In the second scenario,

a. SourceHost and Host1 were chosen as the sources of the multicast traffics.

b. Host2 and Host4 joined the multicast group ff7e:240:2001:face:abcd:2:0:3

c. Host3 and Host5 joined the multicast group ff7e:240:2001:face:abcd:2:0:5

d. SourceHost send packets to this multicast address ff7e:240:2001:face:abcd:2:0:3

Host1 send packets to this multicast address ff7e:240:2001:face:abcd:2:0:5 simultaneously.

e. Table 3.2 shows the jitters for the groups ten minutes and 1 hour streaming.

f. Figure 3.2 shows the jitters' graph for the groups ten minutes and 1 hour streaming.

g. All other settings for JPERF take the default values.

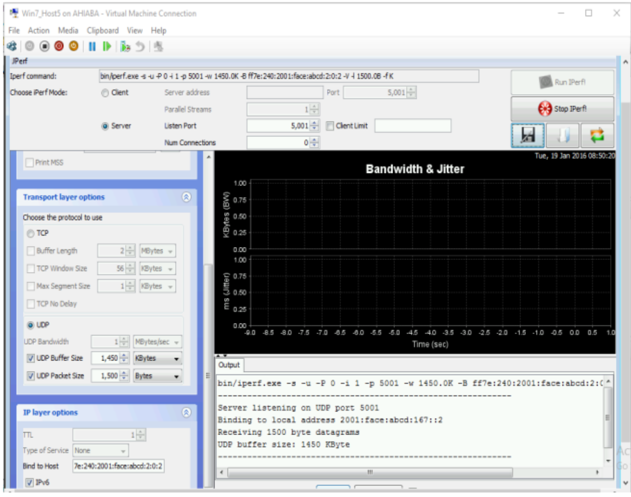

Figure 2.3: One of the five hosts waiting to receive a multicast streaming

Figure 2.4: All the five hosts receiving a multicast streaming

3. RESULT ANALYSIS

3.1 Throughput and Latency

Table 3.1 shows stable jitters among the five hosts that participated in a single multicast stream for both the ten minutes and one hour running. The same stable jitters was noticed when streaming two different multicast datagram simultaneously with two hosts each, participating in the streams for ten minutes and the one run (Table 3.2). Figure 3.1 and figure 3.2 are the graphs for table 3.1 and table 3.2 respectively. Therefore, there was no significant increase in latencies of the receiving hosts as multicast group increase from one to two.

From the result obtained from all the average 10 minutes and average 1 hour runs, the throughput was 100% and the jitters range from 0 to 5.734ms

Table 3.1: Test result for five listeners received ten minutes and1hour streams

	Jitters of Receiving hosts				
Sender (Source-Host)	**Host1**	**Host2**	**Host3**	**Host4**	**Host5**
10 min Run	3.404	1.400	1.257	1.625	1.860
10 min 2nd Run	1.980	2.532	3.448	2.563	3.269
10 min 3rd Run	1.434	2.950	1.814	1.348	1.425
Average of the three 10 min Run	**2.272**	**2.294**	**2.173**	**1.845**	**2.184**
1hr Run	1.063	1.117	1.487	1.254	0.005
1hr 2nd Run	5.734	4.601	2.754	2.911	1.766
1hr 3rd Run	0.638	1.022	1.069	0.931	0.329
Average of the three 1hr Run	**2.478**	**2.246**	**1.77**	**1.698**	**0.7**

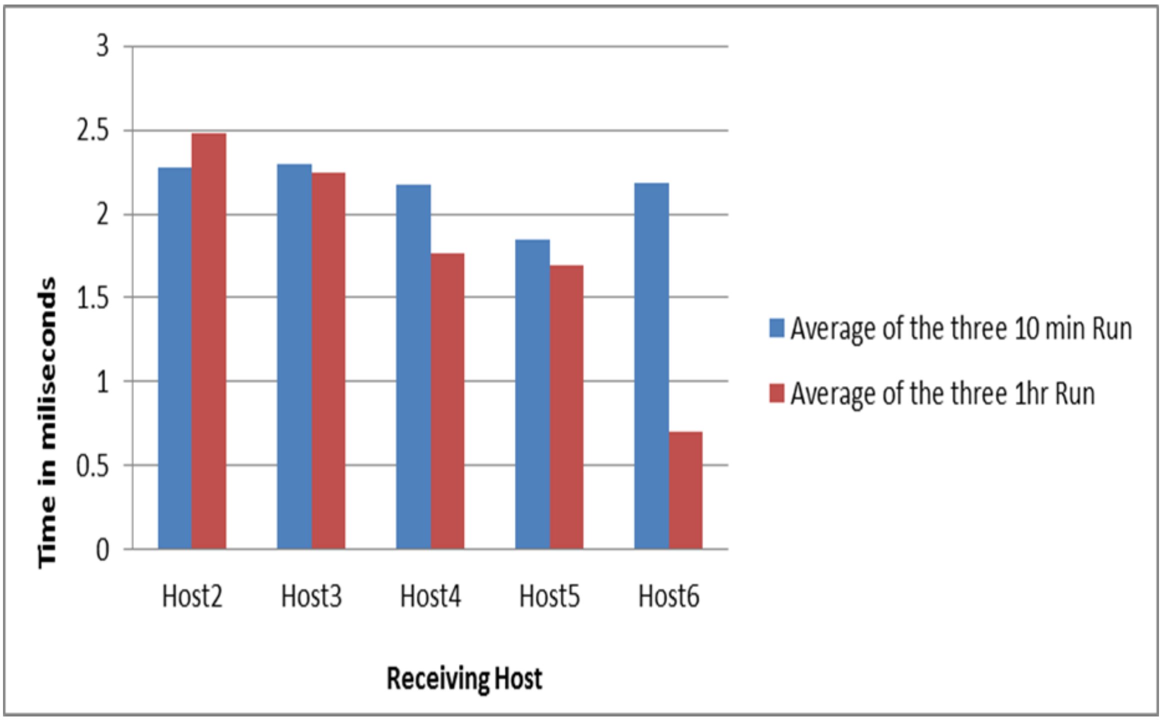

Figure 3.1: Jitter graph for five listeners received ten minutes and1hour streams

Table 3.2: Test result for running two IPv6 groups simultaneously (ten minutes and1hour streams)

	Jitters of Receiving hosts			
	Source-Host multicast to ff7e:240:2001:face:abcd:2:0:3		Host1 multicast ff7e:240:2001:face:abcd:2:0:5	
Durations	**Host2**	**Host4**	**Host3**	**Host5**
10 min Run	0.429	0.694	1.853	2.107
10 min 2nd Run	1.061	2.621	1.875	1.491
10 min 3rd Run	0.539	1.594	2.004	2.027
Average of the three 10 min Run	**0.676**	**1.636**	**1.91**	**1.875**
1hr Run	1.979	1.694	3.727	1.971
1hr 2nd Run	3.747	1.982	3.278	4.751
1hr 3rd Run	0.229	1.594	3.003	2.107
Average of the three 1hr Run	**1.985**	**1.756**	**3.336**	**2.943**

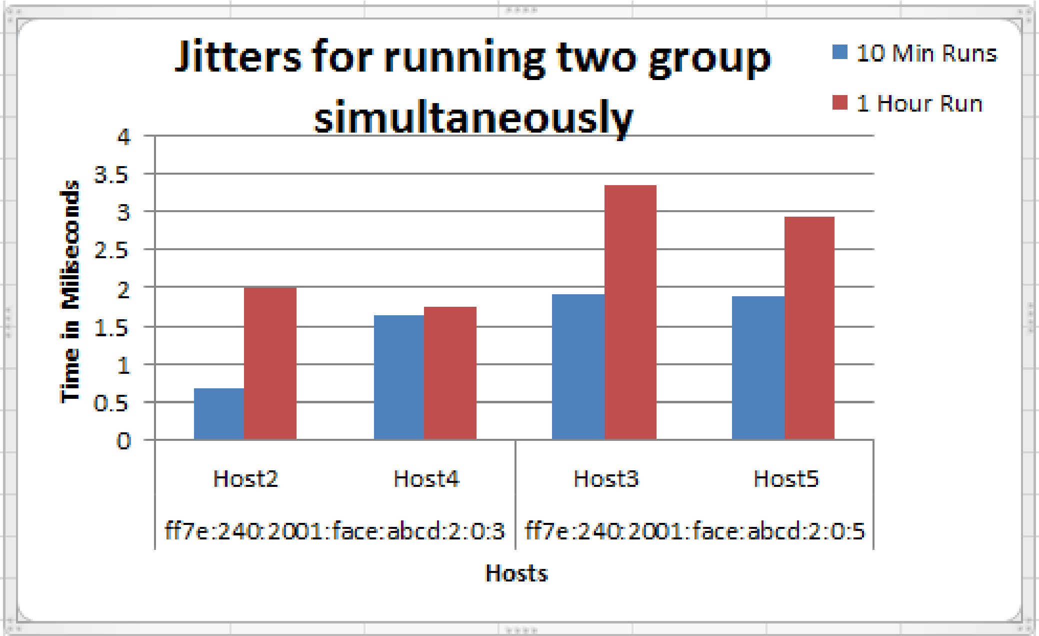

Figure 3.2: Jitter graph for running two IPv6 multicast groups simultaneously (ten minutes and1hour streams)

3.2 Data Loss

There was no datagram lost in all the running scenarios. The zero datagram loss resulted in the 100% throughput achieved (figure 3.3).

```
[ ID] Interval        Transfer     Bandwidth        Jitter    Lost/Total Datagrams
[132] 3580.0-3581.0 sec   122 KBytes   122 KBytes/sec  3.734 ms     0/   85 (0%)
[132] 3581.0-3582.0 sec   122 KBytes   122 KBytes/sec  3.780 ms     0/   85 (0%)
[132] 3582.0-3583.0 sec   121 KBytes   121 KBytes/sec  2.116 ms     0/   84 (0%)
[132] 3583.0-3584.0 sec   119 KBytes   119 KBytes/sec  1.282 ms     0/   83 (0%)
[132] 3584.0-3585.0 sec   122 KBytes   122 KBytes/sec  0.544 ms     0/   85 (0%)
[132] 3585.0-3586.0 sec   121 KBytes   121 KBytes/sec  2.459 ms     0/   84 (0%)
[132] 3586.0-3587.0 sec   121 KBytes   121 KBytes/sec  0.547 ms     0/   84 (0%)
[132] 3587.0-3588.0 sec   121 KBytes   121 KBytes/sec  4.606 ms     0/   84 (0%)
[132] 3588.0-3589.0 sec   122 KBytes   122 KBytes/sec  2.296 ms     0/   85 (0%)
[132] 3589.0-3590.0 sec   119 KBytes   119 KBytes/sec  6.038 ms     0/   83 (0%)
[132] 3590.0-3591.0 sec   119 KBytes   119 KBytes/sec  1.150 ms     0/   83 (0%)
[132] 3591.0-3592.0 sec   121 KBytes   121 KBytes/sec  1.871 ms     0/   84 (0%)
[132] 3592.0-3593.0 sec   121 KBytes   121 KBytes/sec  1.245 ms     0/   84 (0%)
[132] 3593.0-3594.0 sec   122 KBytes   122 KBytes/sec  0.641 ms     0/   85 (0%)
[132] 3594.0-3595.0 sec   121 KBytes   121 KBytes/sec  0.972 ms     0/   84 (0%)
[132] 3595.0-3596.0 sec   121 KBytes   121 KBytes/sec  5.047 ms     0/   84 (0%)
[132] 3596.0-3597.0 sec   122 KBytes   122 KBytes/sec  1.280 ms     0/   85 (0%)
[132] 3597.0-3598.0 sec   119 KBytes   119 KBytes/sec  3.134 ms     0/   83 (0%)
[132] 3598.0-3599.0 sec   122 KBytes   122 KBytes/sec  6.967 ms     0/   85 (0%)
[132] 3599.0-3600.0 sec   121 KBytes   121 KBytes/sec  0.588 ms     0/   84 (0%)
[ ID] Interval        Transfer     Bandwidth        Jitter    Lost/Total Datagrams
[132]  0.0-3600.0 sec  434129 KBytes   121 KBytes/sec  0.517 ms     0/302414  (0%)
```

Zero datagram lost

Figure 3.3: Screen Shot of a Host with No Data Loss

3.3 Protocol Overheads

The following are the protocols over heads 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. 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. ICMPv6 neighbour solicitation and neighbour advertisement messages were also noticed.

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

No.	Time	Source	Destination	Protocol
6353	39.226133000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6354	39.226133000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6355	39.226133000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6356	39.226133000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6357	39.246900000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6358	39.247768000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6359	39.263406000	fe80::c803:3bff:fe94:8	ff02::d	PIMv2
6360	39.263406000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6361	39.263406000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6362	39.279049000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6363	39.279049000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6364	39.294665000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6365	39.294665000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6366	39.310282000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6367	39.310282000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6368	39.325906000	fe80::c803:3bff:fe94:8	ff02::d	PIMv2
6369	39.325906000	fe80::c801:36ff:fe24:8	fe80::c802:38ff:fef4:6	ICMPv6
6370	39.325906000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6371	39.325906000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6372	39.342043000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6373	39.342043000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6374	39.348041000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6375	39.348041000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6376	39.379337000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6377	39.379337000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6378	39.379337000	fe80::c802:38ff:fef4:6	fe80::c801:36ff:fe24:8	ICMPv6
6379	39.379337000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6380	39.379337000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6381	39.394975000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6
6382	39.394975000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	UDP
6383	39.410554000	fe80::c802:38ff:fef4:8	ff02::5	OSPF
6384	39.410554000	fe80::c802:38ff:fef4:6	ff02::5	OSPF
6385	39.410554000	2001:face:abcd:160:7d35:77a8:c777:c3a6	ff7e:240:2001:face:abcd:2:0:2	IPv6

Figure 3.4: Wire shark data capture to show protocols over heads

3.4 Tools Used

a. The virtual machines run Windows 7 pro edition

b. Microsoft Hyper-V (Hypervisor)

c. Graphical Network Simulator 3 (Configured the logic of the virtual network with 3 routers 7200 series)

d. Wire Shark (Data Captures)

e. JPERF (Traffic Generator)

f. Host system (Intel Core i5, Windows 10 Pro, RAM 6GB, CPU 2.4GHz, Hard Disk 750 GB)

4. CONCLUSION

From the experiment, IPv6 multicast routing over a virtual network has 100% throughput, the jitter (variations in latency) varies among the hosts in all the running scenarios, but low and stabled jitters were noticed as the running duration increases and the number multicast stream increased from one to running two simultaneously. No data loss was noticed in all the run. IPv6 multicast routing was successfully demonstrated among six participating hosts on a virtual local area network. However, this paper has the following limitations:

a. The experiment was carried out in a virtual lab not a real lab.

b. The datagram tested are not real application data.

c. The resources available for the virtual machines depend on resource of the host system. Host with better resources may better performance.

5. REFERENCES

[1] Cisco. (2001). Overview of IP Multicast. Retrieved March 12, 2013, from http://www.cisco.com/en/US/tech/tk828/technologies_white_paper09186a0080092942.shtml

[2] Daniel, M (2001). Voice Over IPv6: Architecture for Next Generation VoIP Networks. Linacre House, Jordan Hill. Oxford Ox2 8DP, USA, pg 251-291.

[3] Deering, S., & Hinden, R. (1998). RFC 2460 Internet Protocol Version 6 (IPv6) Specification. Retrieved on March 17, 2014, from https://www.ietf.org/rfc/rfc2460.txt

[4] Deering, S., & Hinden, R. (2006): IP Version 6 Addressing Architecture. Retrieved 17 March, 2014 from http://tools.ietf.org/html/rfc4291

[5] IBM (2007): Virtualization in Education. White Paper Retrieved 27 May 2014 from http://www-07.ibm.com/solutions/in/education/download/Virtualization%20in%20Education.pdf

[6] Network Working Group. (1998). RFC 2460 Internet Protocol Version 6 (IPv6) Specification. Retrieved on March 17, 2014, from https://www.ietf.org/rfc/rfc2460.txt

[7] Osero, B. O. & Mwathi, D. G. (2014). Implementing Security on virtualized network storage environment. International Journal of Education and Research, 2(4), 1-10

[8] RFC 2460. IPv6 Header. Retrieved on February 12, 2014 from https://en.wikipedia.org/wiki/IPv6_packet

[9] RFC 4291. IP Version 6 Addressing Architecture . Retrieved on 12 February, 2014 from http://tools.ietf.org/html/rfc4291

[10] Webber, J. (2013). IPv6 Test Laboratory. Retrieved on March 17, 2014, from https://www.hpc.mil/images/hpcdocs/ipv6/masterthesis_johannes_weber_ipv6securitytestlaboratory.pdf

[11] Wikipedia: IPv6 Packet Header. Retrieved on 17 March, 2014, from https://en.wikipedia.org/wiki/IPv6_packet

[12] Wikipedia. Logical Diagram of Full Virtualization. Retrieved on 18 January, 2016 from http://isa.unomaha.edu/wp-content/uploads/2012/08/Virtualization.pdf