



Technical Report

Microsoft Hyper-V R2 Performance Comparison: Fibre Channel and iSCSI

Bhavik Desai, NetApp
April 2010 | TR-3846

ABSTRACT

This document compares the performance of 4G FC, 1GbE, and 10GbE iSCSI protocols using Microsoft® Hyper-V™ R2 on NetApp® storage systems. This document compares the individual protocol throughput performance, latency, and CPU utilization at varying workloads.

TABLE OF CONTENTS

1	EXECUTIVE SUMMARY	3
2	INTRODUCTION	3
2.1	PURPOSE	3
2.2	METHODOLOGY.....	3
2.3	PERFORMANCE SUMMARY	4
3	HYPER-V R2 PERFORMANCE TEST RESULTS	5
3.1	HYPER-V R2 PERFORMANCE COMPARISON OF ISCSI WITH JUMBO FRAMES AND FC (4KB WORKLOAD).....	5
3.2	HYPER-V R2 PERFORMANCE COMPARISON OF ISCSI WITH JUMBO FRAMES AND FC (8KB WORKLOAD).....	8
3.3	HYPER-V R2 PERFORMANCE COMPARISON OF ISCSI WITH AND WITHOUT JUMBO FRAMES (4KB WORKLOAD)	11
3.4	HYPER-V R2 PERFORMANCE COMPARISON OF ISCSI WITH AND WITHOUT JUMBO FRAMES (8KB WORKLOAD)	14
4	TEST SETUP AND CONFIGURATION	17
4.1	HARDWARE AND SOFTWARE SPECIFICATIONS.....	17
4.2	CONNECTIVITY	18
4.3	STORAGE-TO-HOST PROVISIONING	19
4.4	VIRTUAL MACHINES.....	21
4.5	IOMETER CONFIGURATION	22
4.6	TEST RUN DETAILS	22
5	REFERENCES	23
6	ACKNOWLEDGEMENTS	23

1 EXECUTIVE SUMMARY

The performance comparison testing documented by this technical report shows that Hyper-V R2 on NetApp storage is capable of delivering the performance required for any type of production environment irrespective of whether the Fibre Channel (FC) protocol or the iSCSI protocol is used.

Some key items that illustrate this point are as follows:

- The test results show that, in terms of throughput performance, iSCSI and FC delivered comparable performance irrespective of the load on the system.
- The latency observed to achieve the performance is comparable for the same load point irrespective of the protocol used.
- iSCSI consumed approximately three to five percentage points more Hyper-V R2 CPU resources than FC to achieve comparable performance.
- Performance using iSCSI with jumbo frames enabled was comparable to iSCSI performance without jumbo frames while consuming approximately one to four percentage points fewer Hyper-V R2 server CPU resources.

2 INTRODUCTION

Server virtualization in the data center is a key initiative that organizations are undertaking today. Microsoft is an important player in this field with its Hyper-V R2 server virtualization solution. Since storage is a major piece in effective virtualization solutions, NetApp storage arrays are an ideal complement to Hyper-V R2 server virtualization solutions by providing efficient, flexible, cost-effective, and integrated storage solutions.

NetApp and Microsoft Hyper-V R2 natively support data access using FC and iSCSI protocols. Because each protocol has different management and deployment needs, determining which of these protocols to implement is a key step in designing and deploying a virtualized data center. Knowing how each protocol performs in terms of throughput and CPU utilization can help in making this important consideration.

2.1 PURPOSE

This technical report shows the results of tests conducted to compare the performance of the FC and software-initiated iSCSI protocols using Gigabit Ethernet and 10-Gigabit Ethernet technology in a Microsoft Hyper-V R2 environment using NetApp storage. The goal of these tests is to help customers decide how to build their virtual infrastructures.

These tests were not designed to show the maximum throughput achieved for each protocol but to simulate a real-world environment running real-world loads. The test and validation environment used is similar to what can be found in a typical Hyper-V R2 environment using FC and iSCSI protocols having multiple Hyper-V R2 hosts running multiple virtual machines accessing datastores. The tests used realistic I/O patterns, I/O block sizes, read/write mixes, and I/O loads common to various operating systems and enterprise applications.

The tests were designed so that no piece of the configuration used for testing would become a bottleneck; the NetApp FAS3160 storage was configured to provide adequate performance for these tests but was not driven to the limits of its performance capabilities. As a result, the actual IOPS values shown in this report do not reflect the maximum values that the FAS3160 can generate using these specific loads.

2.2 METHODOLOGY

The tests we conducted simulated low, medium, and high workloads as experienced by Hyper-V R2 environments using both FC and iSCSI protocols. The test environment included the following:

- Four Windows[®] Server 2008 R2 host systems on IBM x3550 servers
- The Hyper-V R2 role enabled for each of the host systems
- Each host system with 8 VMs running 64-bit Windows Server 2008 Enterprise Edition SP2
- Two NetApp FAS3160 controllers in an active-active controller configuration providing data accessed by the VMs
- Identical test infrastructure and loads while running tests for both FC and iSCSI protocols

After the environment was set up, we used the industry-standard IOMeter benchmark to generate load and measure performance for each protocol using workloads ranging from light to heavy amounts of I/O going from Hyper-V R2 hosts to NetApp FAS3160 storage. The tests used varying workloads and were run with an increasing number of outstanding I/Os per each of the 32 VMs.

The following tests are presented in this document:

- Performance comparison of FC, iSCSI using 1-Gigabit Ethernet (1GbE) with jumbo frames enabled, and iSCSI using 10GbE with jumbo frames enabled
- Performance comparison of iSCSI using 1GbE and 10GbE, with and without jumbo frames

For both tests, we observed the following:

- The total throughput in IOPS reported at the IOMeter controller running load on all VMs
- The average latency reported in milliseconds at the IOMeter controller running load on all VMs
- The average % total CPU time for the Hyper-V hypervisor Logical Processor across all four Hyper-V R2 hosts

2.3 PERFORMANCE SUMMARY

These tests validate that FC and iSCSI using 1GbE or 10GbE storage protocols are production ready and provide end users with enhanced flexibility when designing and deploying their storage environments. The following sections highlight the test results that led to this conclusion.

For the complete test results, refer to Section 3, “Hyper-V R2 Performance Test Results,” on page 5.

HYPER-V R2 PERFORMANCE COMPARISON OF FC AND ISCSI

- The observed throughput is comparable for both FC and iSCSI using either 10GbE or 1GbE across all load points.
- The latency observed is comparable for both FC and iSCSI using either 10GbE or 1GbE across various load points.
- iSCSI using 10GbE had slightly better performance compared to 1GbE in terms of IOPS and latency.
- The iSCSI CPU utilization was three to five percentage points higher than FC using either 1GbE or 10GbE.
- The percentage difference in CPU utilization between iSCSI and FC decreased as the load increased.

HYPER-V R2 PERFORMANCE COMPARISON OF ISCSI WITH AND WITHOUT JUMBO FRAMES

- The throughput for iSCSI using 1GbE and jumbo frames was comparable to performance using 1GbE without jumbo frames for both 4K and 8K request sizes.
- The throughput for iSCSI using 10GbE and jumbo frames was comparable to performance using 10GbE without jumbo frames for both 4K and 8K request sizes.
- The CPU utilization observed using iSCSI with jumbo frames enabled was approximately 1.5–4 percentage points lower compared to results without jumbo frames for loads using 4K request sizes, regardless of whether 1GbE or 10GbE was used.
- The CPU utilization observed using iSCSI with jumbo frames enabled was approximately 1–3.5 percentage points lower compared to results without jumbo frames for loads using 8K request sizes, regardless of whether 1GbE or 10GbE was used.

These tests showed that using jumbo frames with iSCSI used fewer Hyper-V R2 CPU resources while producing comparable performance, regardless of wire speed.

3 HYPER-V R2 PERFORMANCE TEST RESULTS

3.1 HYPER-V R2 PERFORMANCE COMPARISON OF ISCSI WITH JUMBO FRAMES AND FC (4KB WORKLOAD)

This test uses a 4KB workload to compare the Hyper-V R2 performance in IOPS, latency in milliseconds, and CPU utilization of the following:

- 4Gb FC
- 1GbE iSCSI with jumbo frames
- 10GbE iSCSI with jumbo frames

THROUGHPUT COMPARISON

Figure 1 shows the throughput comparison results:

- The throughput performance is comparable irrespective of the protocol used.
- iSCSI using 10GbE with jumbo frames enabled delivered slightly better performance compared to using 1GbE with jumbo frames.

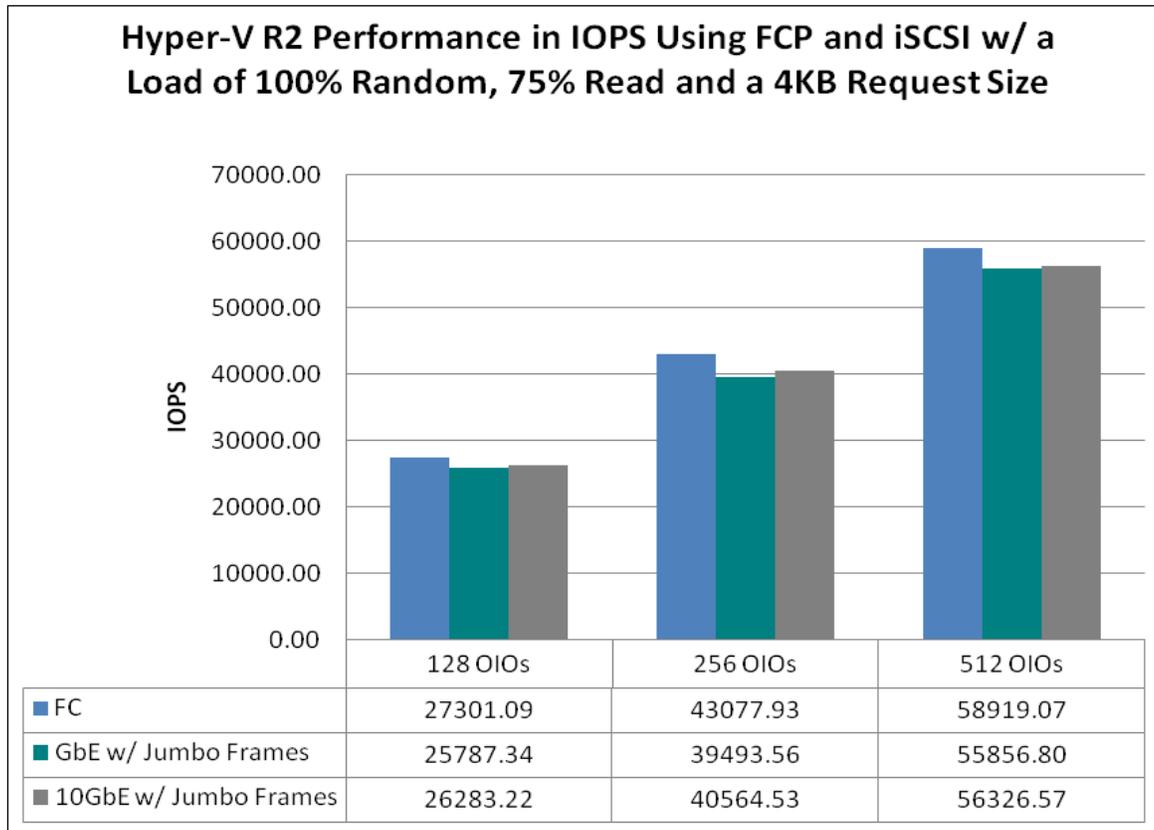


Figure 1) FC and iSCSI performance in IOPS using a 4K request size.

LATENCY COMPARISON

Figure 3 shows the latency comparison results:

- The average latency observed is comparable irrespective of protocol used.
- iSCSI using 10GbE with jumbo frames enabled delivered slightly better latency compared to using 1GbE with jumbo frames.

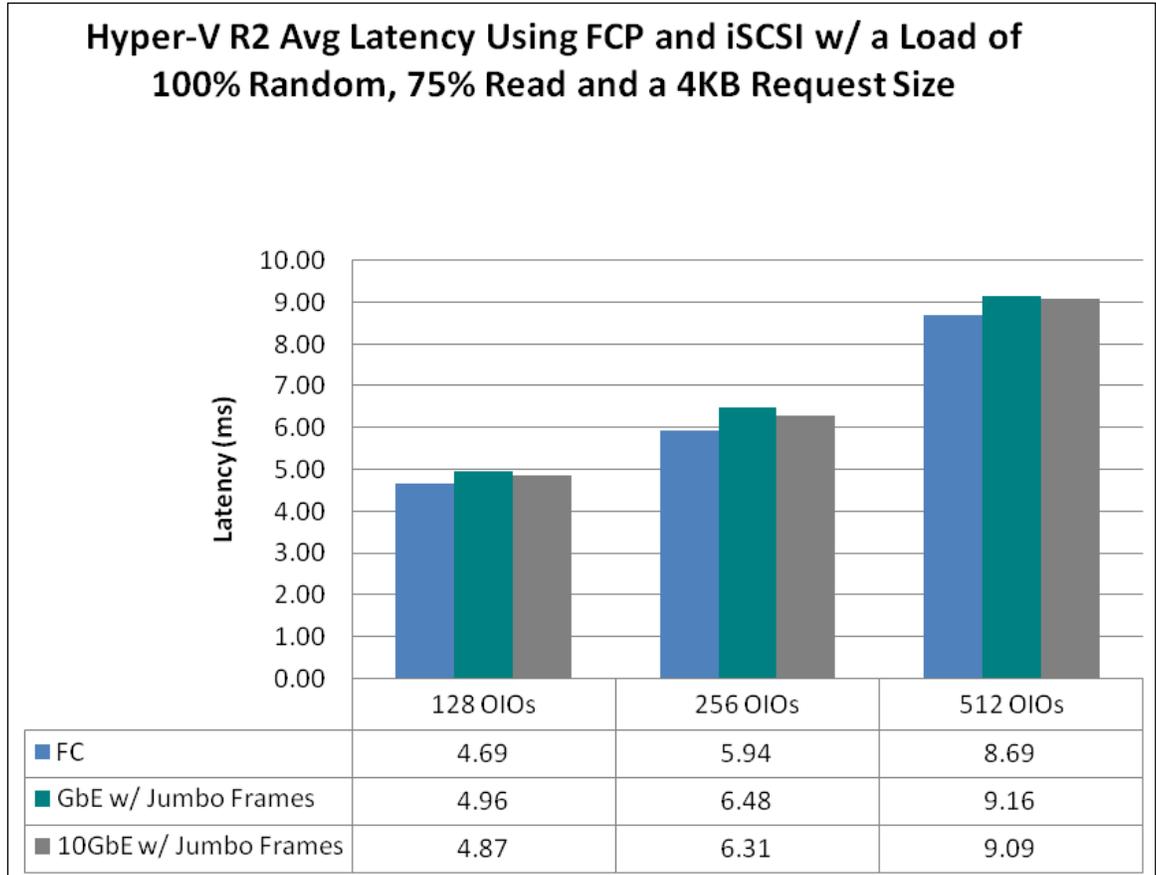


Figure 2) FC and iSCSI average latency using a 4K request size.

CPU UTILIZATION COMPARISON

Figure 5 shows the CPU utilization comparison results:

- iSCSI used approximately three to four percentage points more Hyper-V R2 CPU resources than FC for 4K request sizes at all load points.
- iSCSI using 1GbE with jumbo frames and 10GbE with jumbo frames consumed comparable amounts of Hyper-V R2 CPU resources.

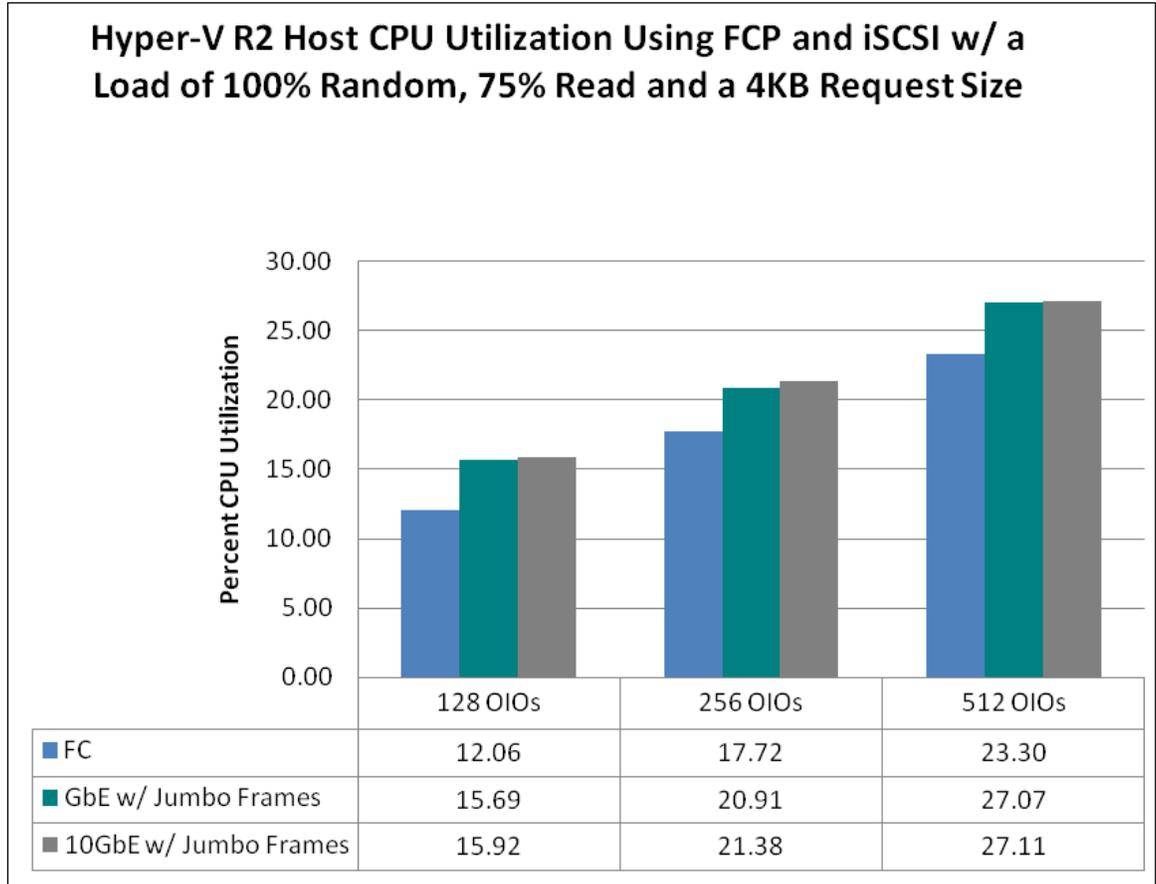


Figure 3) FC and iSCSI CPU utilization using a 4K request size.

3.2 HYPER-V R2 PERFORMANCE COMPARISON OF ISCSI WITH JUMBO FRAMES AND FC (8KB WORKLOAD)

This test uses an 8KB workload to compare the Hyper-V R2 performance in IOPS, latency in milliseconds, and CPU utilization of the following:

- 4Gb FC
- 1GbE iSCSI with jumbo frames
- 10GbE iSCSI with jumbo frames

THROUGHPUT COMPARISON

Figure 7 shows the throughput comparison results:

- The throughput performance is comparable irrespective of protocol used.
- iSCSI using 10GbE with jumbo frames enabled delivered slightly better performance compared to using 1GbE with jumbo frames.

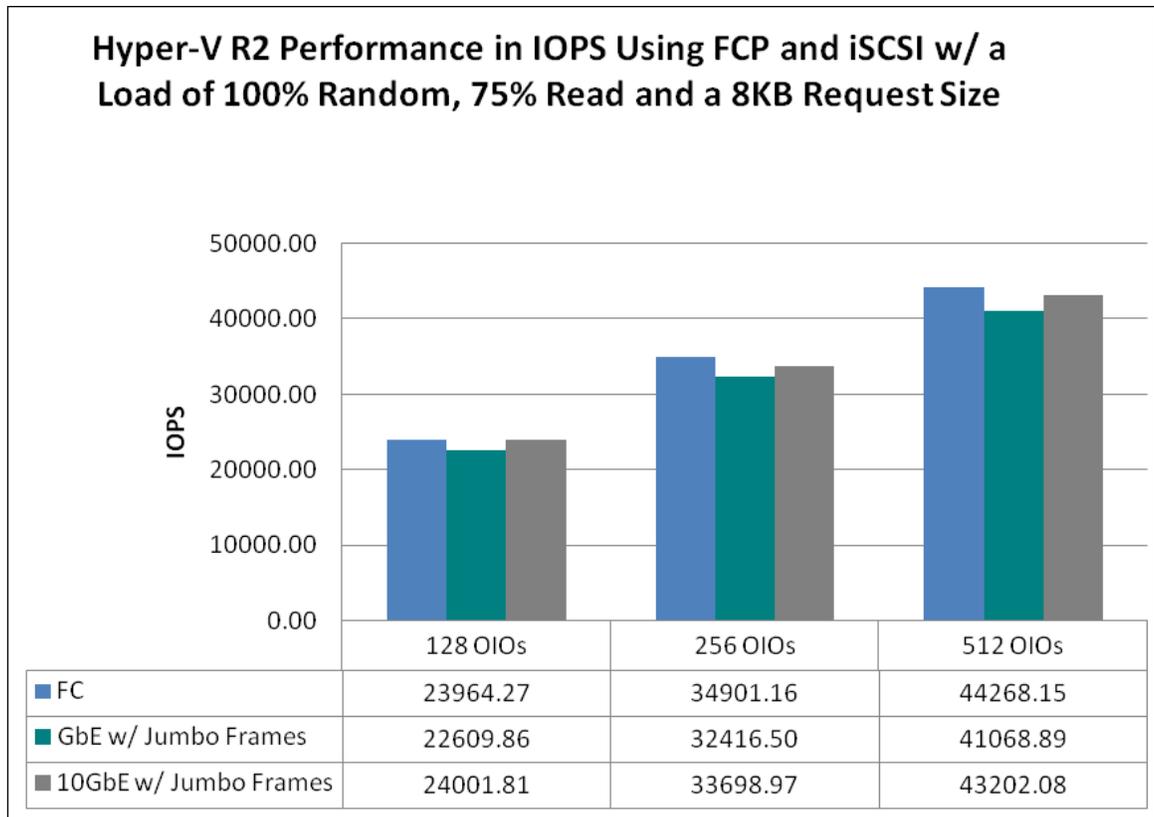


Figure 4) FC and iSCSI performance in IOPS using an 8K request size.

LATENCY COMPARISON

Figure 9 shows the latency comparison results:

- The average latency observed is comparable irrespective of protocol used.
- iSCSI using 10GbE with jumbo frames enabled delivered slightly better latency compared to using 1GbE with jumbo frames.

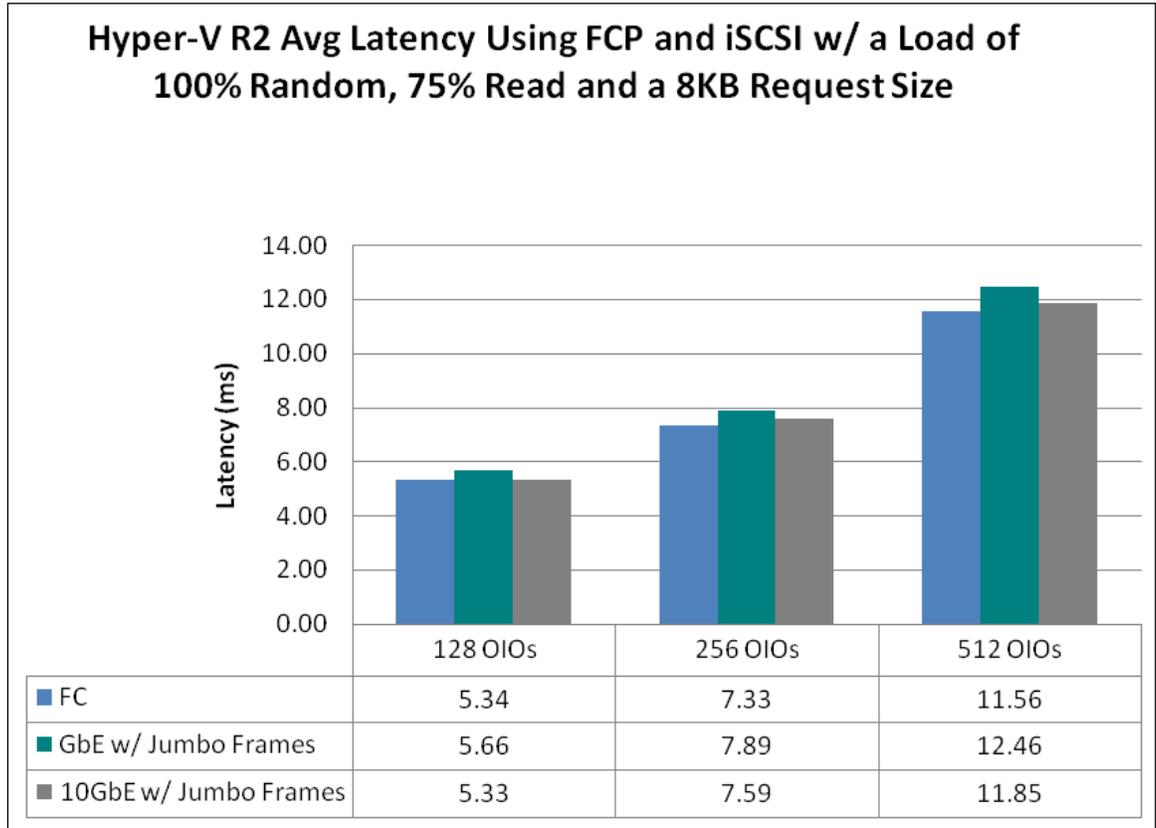


Figure 5) FC and iSCSI average latency using an 8K request size.

CPU UTILIZATION COMPARISON

Figure 11 shows the CPU utilization comparison results:

- iSCSI used approximately five percentage points more Hyper-V R2 CPU resources than FC for 8K request sizes.
- iSCSI using 1GbE with jumbo frames and 10GbE with jumbo frames consumed comparable amounts of Hyper-V R2 CPU resources.

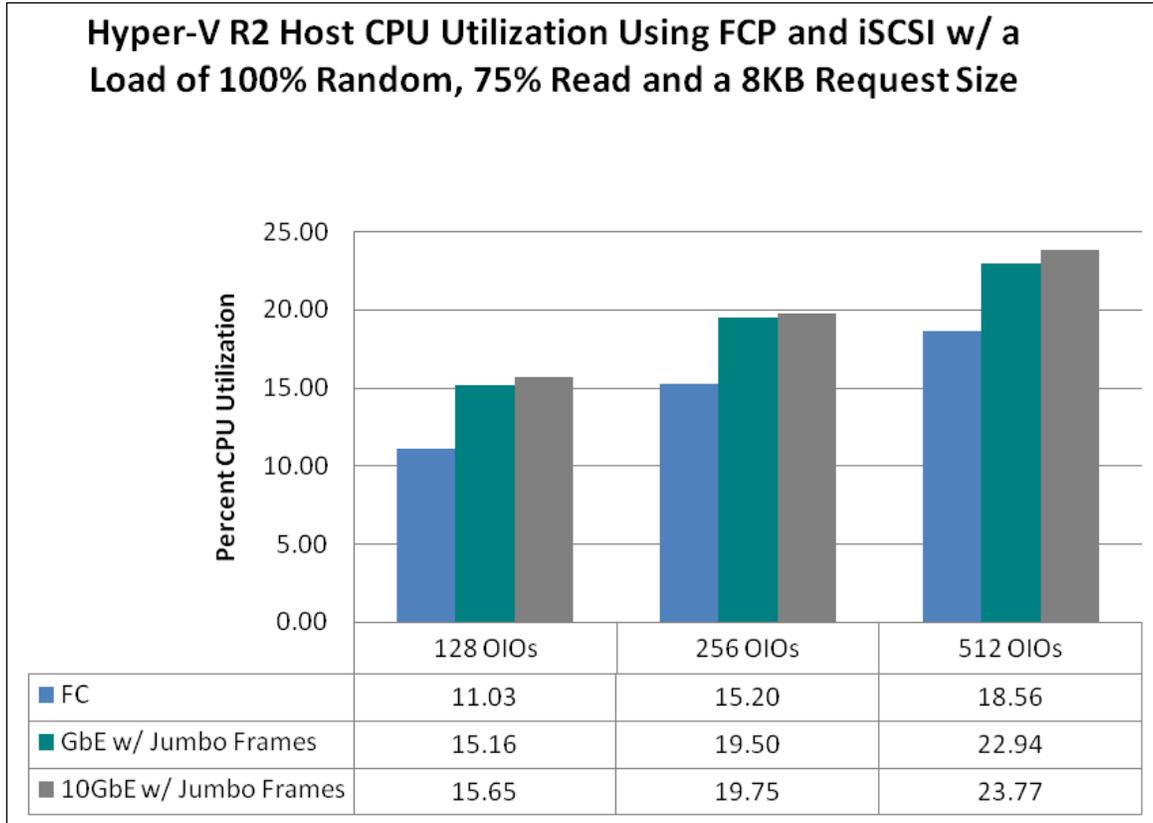


Figure 6) FC and iSCSI CPU utilization using an 8K request size.

3.3 HYPER-V R2 PERFORMANCE COMPARISON OF ISCSI WITH AND WITHOUT JUMBO FRAMES (4KB WORKLOAD)

This test uses a 4KB workload to compare the Hyper-V R2 performance in IOPS, latency in milliseconds, and CPU utilization of the following:

- 1GbE iSCSI with jumbo frames
- 1GbE iSCSI without jumbo frames
- 10GbE iSCSI with jumbo frames
- 10GbE iSCSI without jumbo frames

THROUGHPUT COMPARISON

Figure 13 shows the throughput comparison results:

- Throughput using jumbo frames is comparable to throughput without jumbo frames.
- As the load increased, the percentage differences in throughput generated using iSCSI with and without jumbo frames decreased. This is particularly true when using the 10GbE configurations.

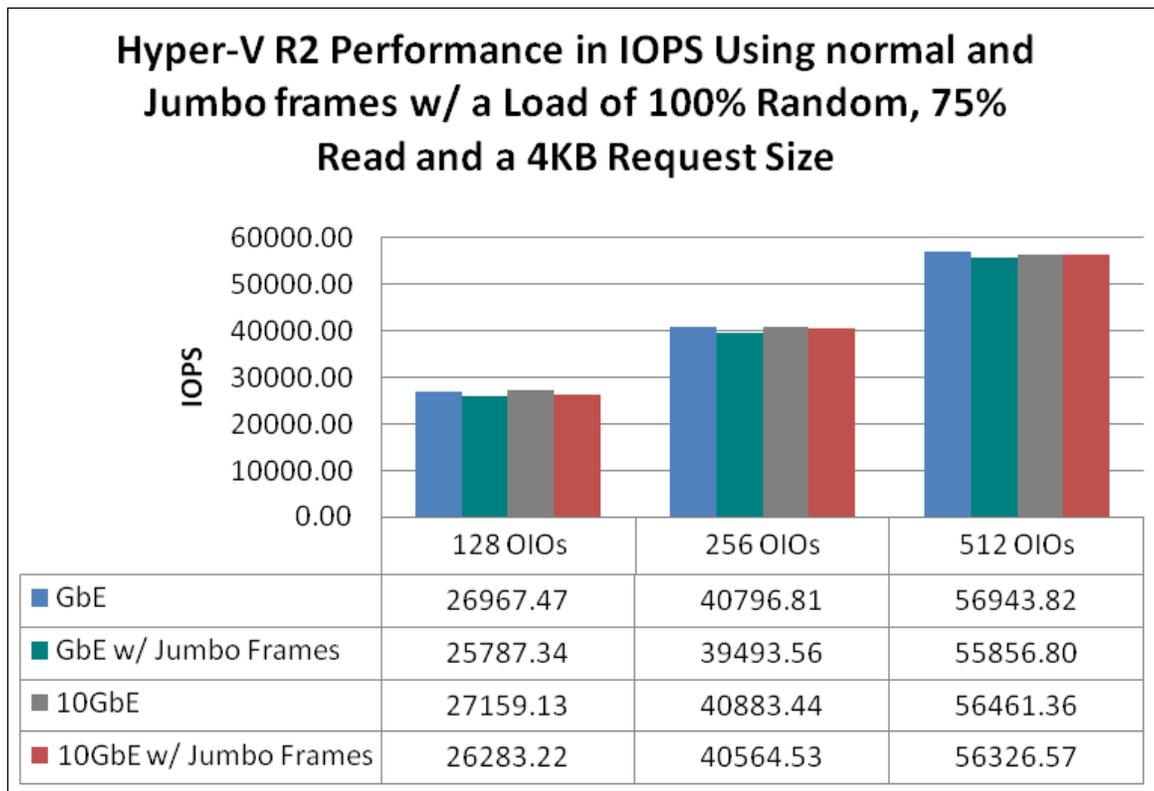


Figure 7) iSCSI performance with and without jumbo frames using a 4K request size.

LATENCY COMPARISON

Figure 15 shows the latency comparison results:

- Average latency using jumbo frames is comparable to average latency without jumbo frames.
- As the load increased, the percentage differences in latency generated using iSCSI with and without jumbo frames decreased.

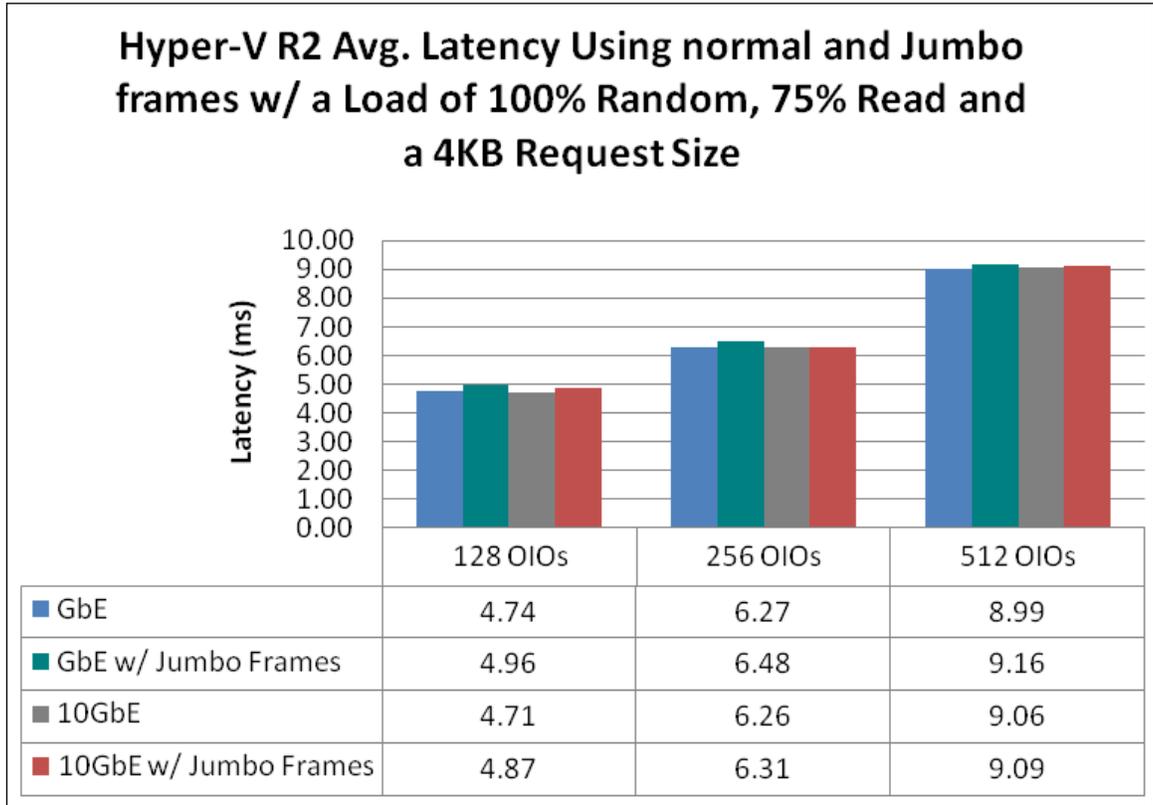


Figure 8) iSCSI latency with and without jumbo frames using a 4K request size.

CPU UTILIZATION COMPARISON

Figure 17 shows the CPU utilization comparison results:

- 1GbE iSCSI Hyper-V R2 CPU utilization with jumbo frames enabled is approximately three percentage points lower than 1GbE iSCSI CPU utilization without jumbo frames enabled.
- 10GbE iSCSI Hyper-V R2 CPU utilization with jumbo frames enabled is approximately one to two percentage points lower than 10GbE iSCSI CPU utilization without jumbo frames enabled.

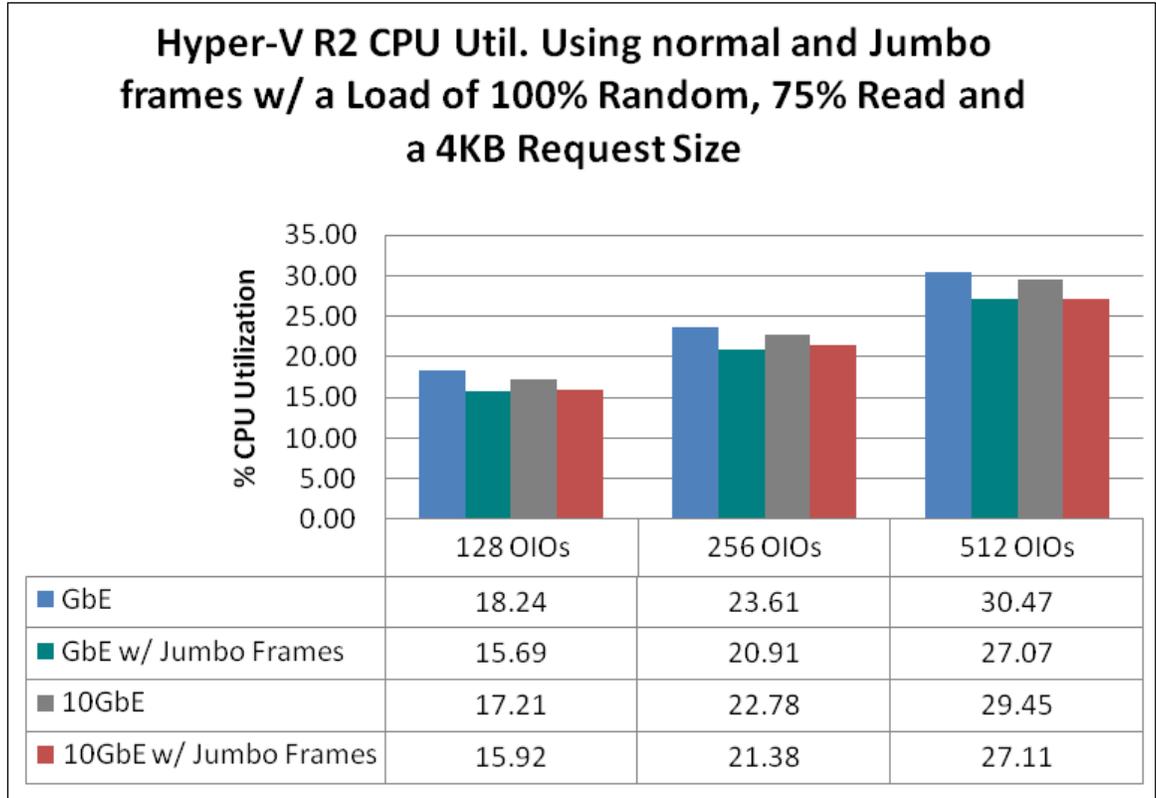


Figure 9) iSCSI CPU throughput with and without jumbo frames using a 4K request size.

TEST CONCLUSION

Overall, using the workload with 4K requests, we found that iSCSI with jumbo frames using 1GbE and 10GbE delivers similar performance to iSCSI without jumbo frames using 1GbE and 10GbE while consuming fewer Hyper-V R2 CPU resources.

3.4 HYPER-V R2 PERFORMANCE COMPARISON OF ISCSI WITH AND WITHOUT JUMBO FRAMES (8KB WORKLOAD)

This test uses an 8KB workload to compare the Hyper-V R2 performance in IOPS, latency in milliseconds, and CPU utilization of the following:

- 1GbE iSCSI with jumbo frames
- 1GbE iSCSI without jumbo frames
- 10GbE iSCSI with jumbo frames
- 10GbE iSCSI without jumbo frames

THROUGHPUT COMPARISON

Figure 19 shows the throughput comparison results:

- Throughput using jumbo frames is comparable to throughput without jumbo frames.
- As the load increases, the percentage differences in throughput generated using iSCSI with and without jumbo frames decreases. This is particularly true when using the 10GbE configurations.

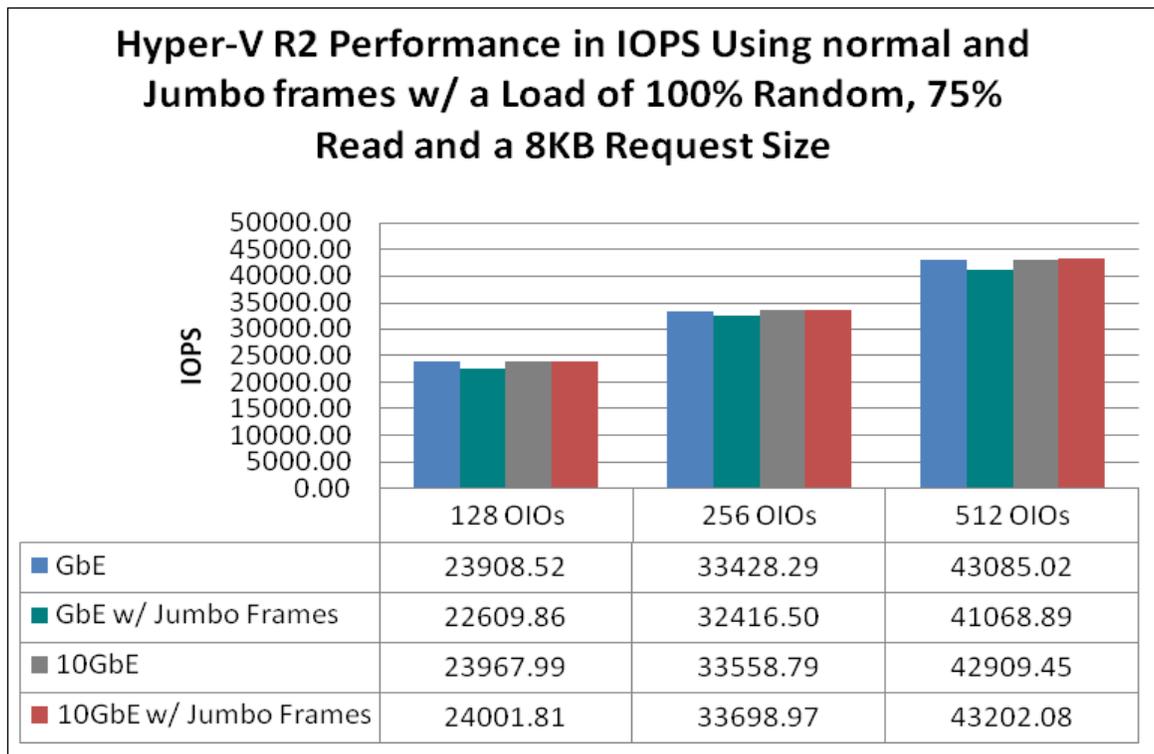


Figure 10) iSCSI performance with and without jumbo frames using an 8K request size.

LATENCY COMPARISON

Figure 21 shows the latency comparison results:

- Average latency using jumbo frames is comparable to average latency without jumbo frames.
- As the load increased, the percentage differences in latency generated using iSCSI with and without jumbo frames decreased.

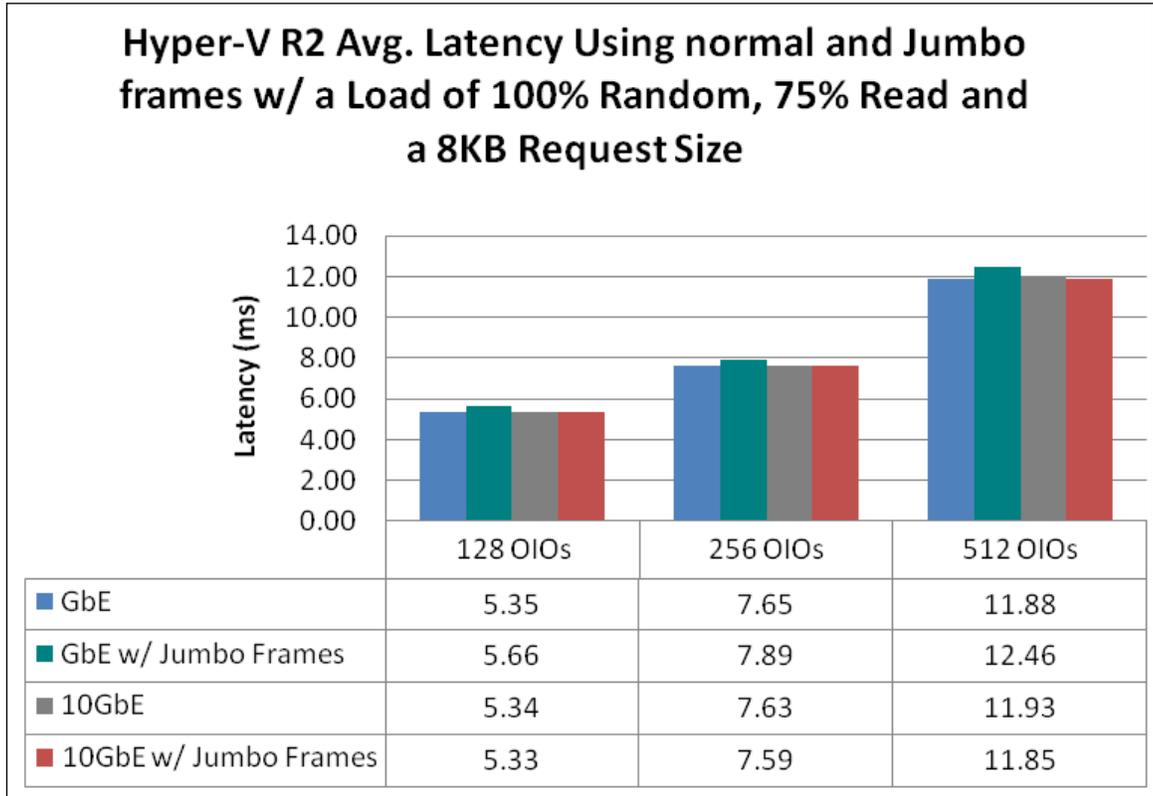


Figure 11) iSCSI latency with and without jumbo frames using an 8K request size.

CPU UTILIZATION COMPARISON

Figure 23 shows the CPU utilization results:

- 1GbE iSCSI Hyper-V R2 CPU utilization with jumbo frames enabled is approximately two to four percentage points lower than 1GbE iSCSI CPU utilization without jumbo frames enabled.
- 10GbE iSCSI Hyper-V R2 CPU utilization with jumbo frames enabled is one to two percentage points lower than 10GbE iSCSI CPU utilization without jumbo frames enabled.

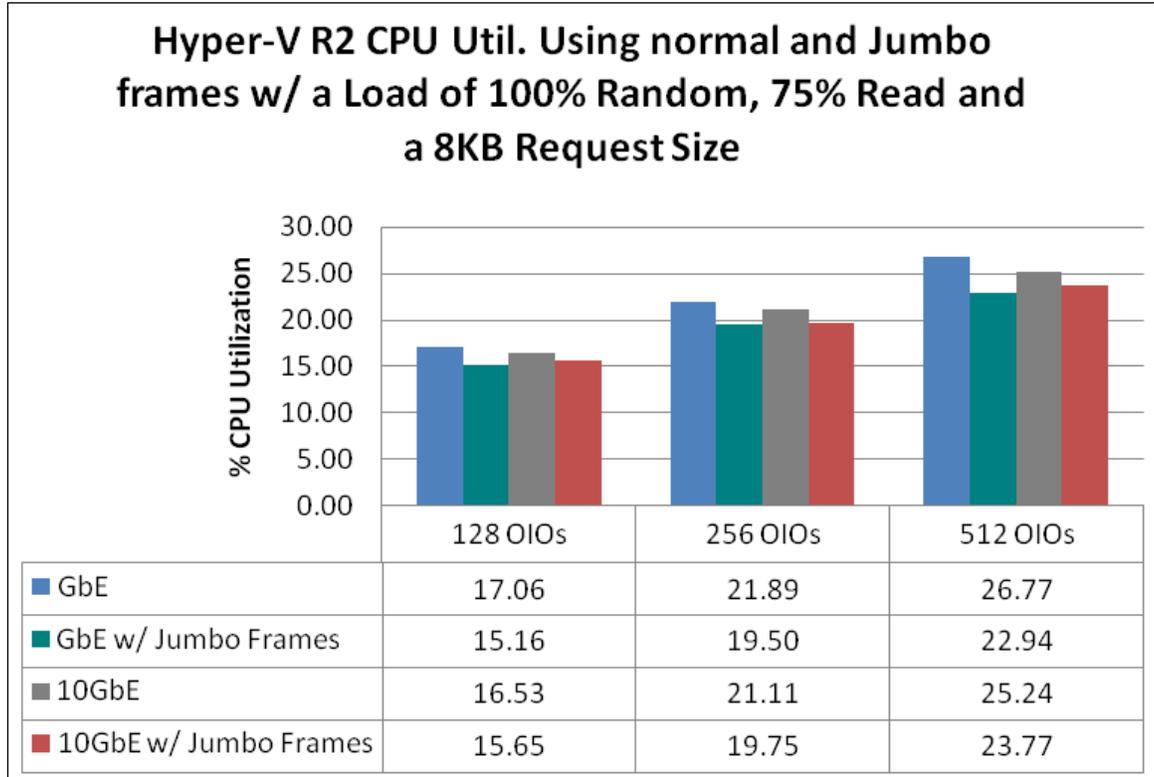


Figure 12) iSCSI CPU throughput with and without jumbo frames using an 8K request size.

TEST CONCLUSION

Overall, using the workload with 8K requests, we found that iSCSI with jumbo frames using 1GbE and 10GbE delivers similar performance to iSCSI without jumbo frames using 1GbE and 10GbE while consuming fewer Hyper-V R2 CPU resources.

4 TEST SETUP AND CONFIGURATION

The physical test-bed remained the same for tests run with FC and iSCSI. The only difference was that the physical connections connecting the Hyper-V R2 hosts to NetApp storage and the switching infrastructure changed between different protocols.

In the first phase for tests run with FC, we used 4Gb/s connections from the Hyper-V R2 hosts to NetApp storage. For tests run with iSCSI using 1GbE, we used 1GbE connections and switching infrastructure from Hyper-V R2 hosts to NetApp storage. In the second phase, we upgraded the test-bed to use a 10GbE infrastructure and executed the same set of tests.

4.1 HARDWARE AND SOFTWARE SPECIFICATIONS

The setup consisted of four Hyper-V R2 host servers connected to NetApp storage using FC, 1GbE iSCSI, and 10GbE iSCSI. The four Hyper-V R2 hosts were managed using Hyper-V manager.

Table 1) Hardware and software for the Hyper-V R2 test-bed.

Hardware/Software Component	Details
Server	IBM x3550 servers
Processors	1 Quad-Core Intel® Xeon™ E5420 @ 2.5GHz
Memory	14GB
OS	Windows Server 2008 R2
Fibre Channel Network	4Gbps FC
Fibre Channel HBA	QLogic QLE2462 FC HBA
Dedicated NIC for Hyper-V Host Management	Broadcom BCM5708C NetXtreme II
Dedicated NIC for Virtual Machine Network Configuration (Virtual Switch)	Broadcom BCM5708C NetXtreme II
Dedicated NIC for 1GbE iSCSI Connections	Intel PRO/1000 PT Dual-Port Server Adapter
Dedicated NIC for 10GbE iSCSI Connections	Intel Ethernet Server Adapter X520
4Gbp Fibre Channel Switch	Brocade 200E
Gigabit Ethernet Switch	Cisco 4948 Ethernet Switch
10-Gigabit Ethernet Switch	Fujitsu XG1200

Table 2) Hardware and software for NetApp storage.

Hardware/Software Component	Details
Storage Controller	FAS3160A
Data ONTAP® Version	7.3.2
Total Number of Drives	128
Size of Drives	300GB FC
Drive Speed	15K
Drive Type	Fibre Channel

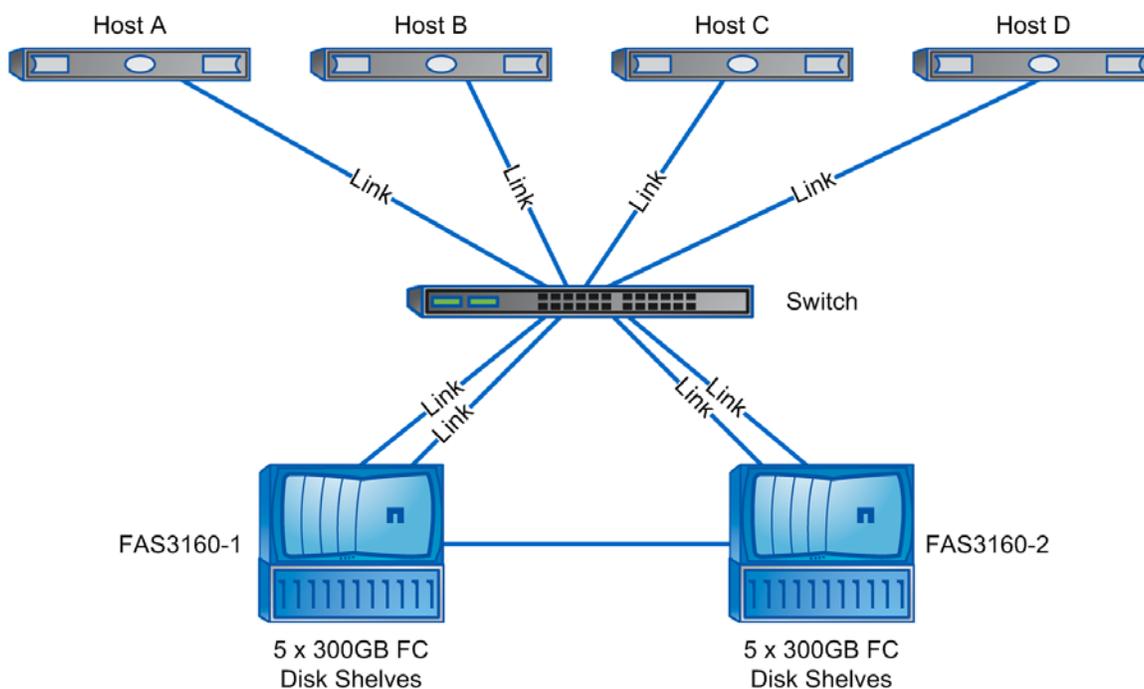
Table 3) Virtual machine components.

Component	Details
OS	64-bit Windows Server 2008 SP2 Enterprise Edition
Virtual Processors	1
Virtual Memory	1GB
OS Virtual Disk Size	15GB
IOMeter Data Virtual Disk Size	10GB

4.2 CONNECTIVITY

The storage controllers were set up in an active-active controller configuration providing access to shared storage. The LUNs were masked using igroups so that:

- VMs in hosts A and B were serviced by FAS3160-1
- VMs in hosts C and D were serviced by FAS3160-2



Test-Bed	Switch	Links
Fibre Channel	Brocade 200E FC	4Gb/s FC
1-Gigabit Ethernet	Cisco 4948 Gigabit Ethernet	1-Gigabit Ethernet
10-Gigabit Ethernet	Fujitsu XG1200 10-Gigabit Ethernet	10-Gigabit Ethernet

Figure 13) Connectivity between Hyper-V R2 hosts and NetApp storage.

4.3 STORAGE-TO-HOST PROVISIONING

After setting up the physical test-bed connections, we provisioned storage on NetApp storage controllers to host the 32 virtual machines used in the tests. Each FAS3160 storage controller was configured with:

- 5 shelves of 300GB 15k FC disks for a total of 140 drives
 - 1 aggregate consisting of 64 drives set up as 4 RAID-DP[®] groups of 16 disks each
- FlexVol[®] volumes and LUNs were carved out from the aggregate on each controller as follows:
- Two 200GB FlexVol volumes, one for each host of vhd files for virtual machines. Each 200GB volume contains eight 20GB LUNs for hosting vhd files for the eight virtual machines on each Hyper-V R2 host.
 - Two 100GB FlexVol volumes, one for each host of IOMeter data files. Each 100GB volume contains eight 10GB LUNs for hosting IOMeter data files for each virtual machine on a Hyper-V R2 host.

Testing the HA and live migration capabilities was beyond the scope of this project. For the purposes of these tests, we provided a one-to-one mapping between vhds and LUNs. Each VM had its vhd file in its own LUN. Each of the 10GB LUNs was presented to each VM as a pass-through disk attached to a SCSI controller for hosting the IOMeter data file.

FC and iSCSI testing was done using the same set of 10GB LUNs presented as pass-through disks. When provisioning storage for accesses using FC or iSCSI, the storage was masked so that the appropriate Hyper-V R2 host or virtual machine could connect to them.

In a NetApp storage controller, LUN masking is handled by the creation of initiator groups, or “igroups.” We created four FC igroups and four iSCSI igroups, one for each Hyper-V R2 host. FC igroups had WWPNs of the FC HBAs associated with that host and iSCSI igroups had IQN identifiers associated with that host to mask access to LUNs.

For more details about provisioning storage and creating igroups for Hyper-V R2, refer to “NetApp Storage Best Practices for Microsoft Virtualization” (TR-3702).

Figure 26 shows the different igroups for one of the FAS3160 storage controllers.

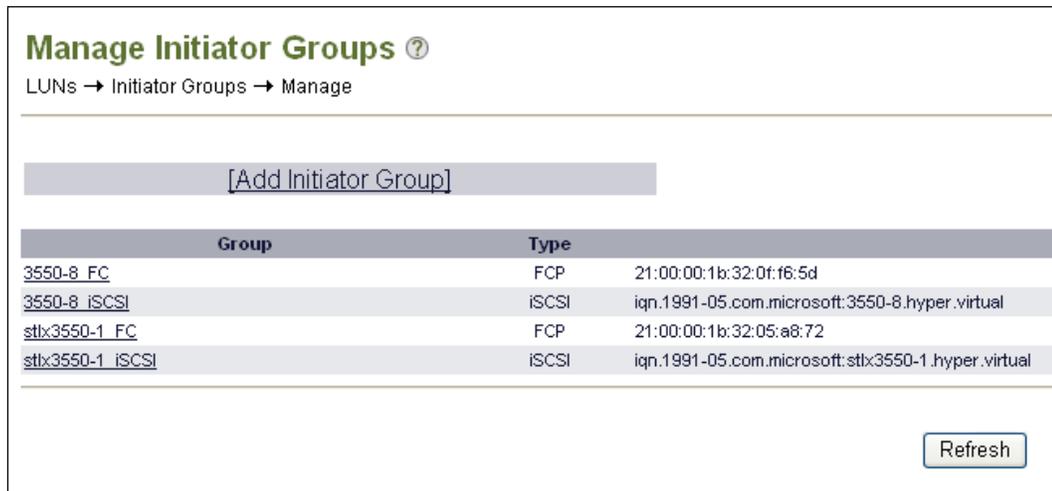


Figure 14) Initiator group configuration for a FAS3160.

After creating the igroups, we used them to create the mappings between each Hyper-V R2 host and the LUNs created previously. Since we used the same set of LUNs for testing with both FC and iSCSI, the LUNs were initially mapped using FC. Figure 28 shows LUNs mapped for use with FC for one of the FAS3160 storage controllers.

Add New LUN			Hide Maps	
LUN	Size	Status	Maps Group : LUN ID	
/vol/host1_ptvol/vm1_pt	10.0G	online	3550-8	FC : 8
/vol/host1_ptvol/vm2_pt	10.0G	online	3550-8	FC : 9
/vol/host1_ptvol/vm3_pt	10.0G	online	3550-8	FC : 10
/vol/host1_ptvol/vm4_pt	10.0G	online	3550-8	FC : 11
/vol/host1_ptvol/vm5_pt	10.0G	online	3550-8	FC : 12
/vol/host1_ptvol/vm6_pt	10.0G	online	3550-8	FC : 13
/vol/host1_ptvol/vm7_pt	10.0G	online	3550-8	FC : 14
/vol/host1_ptvol/vm8_pt	10.0G	online	3550-8	FC : 15
/vol/host1_vhdvol/vm1_vhd	20.0G	online	3550-8	FC : 0
/vol/host1_vhdvol/vm2_vhd	20.0G	online	3550-8	FC : 1
/vol/host1_vhdvol/vm3_vhd	20.0G	online	3550-8	FC : 2
/vol/host1_vhdvol/vm4_vhd	20.0G	online	3550-8	FC : 3
/vol/host1_vhdvol/vm5_vhd	20.0G	online	3550-8	FC : 4
/vol/host1_vhdvol/vm6_vhd	20.0G	online	3550-8	FC : 5
/vol/host1_vhdvol/vm7_vhd	20.0G	online	3550-8	FC : 6
/vol/host1_vhdvol/vm8_vhd	20.0G	online	3550-8	FC : 7
/vol/host2_ptvol/vm10_pt	10.0G	online	stx3550-1	FC : 9
/vol/host2_ptvol/vm11_pt	10.0G	online	stx3550-1	FC : 10
/vol/host2_ptvol/vm12_pt	10.0G	online	stx3550-1	FC : 11
/vol/host2_ptvol/vm13_pt	10.0G	online	stx3550-1	FC : 12
/vol/host2_ptvol/vm14_pt	10.0G	online	stx3550-1	FC : 13
/vol/host2_ptvol/vm15_pt	10.0G	online	stx3550-1	FC : 14
/vol/host2_ptvol/vm16_pt	10.0G	online	stx3550-1	FC : 15
/vol/host2_ptvol/vm9_pt	10.0G	online	stx3550-1	FC : 8
/vol/host2_vhdvol/vm10_vhd	20.0G	online	stx3550-1	FC : 1
/vol/host2_vhdvol/vm11_vhd	20.0G	online	stx3550-1	FC : 2
/vol/host2_vhdvol/vm12_vhd	20.0G	online	stx3550-1	FC : 7
/vol/host2_vhdvol/vm13_vhd	20.0G	online	stx3550-1	FC : 3
/vol/host2_vhdvol/vm14_vhd	20.0G	online	stx3550-1	FC : 4
/vol/host2_vhdvol/vm15_vhd	20.0G	online	stx3550-1	FC : 5
/vol/host2_vhdvol/vm16_vhd	20.0G	online	stx3550-1	FC : 6
/vol/host2_vhdvol/vm9_vhd	20.0G	online	stx3550-1	FC : 0

Figure 15) FAS3160 LUNs configured for tests with FC.

After finishing tests with FC, the LUN mappings using FC were removed and the LUNs were mapped using iSCSI. Figure 30 shows the same LUNs mapped for use with iSCSI.

Add New LUN		Hide Maps	
LUN	Size	Status	Maps Group : LUN ID
/vol/host1_ptvol/vm1_pt	10.0G	online	3550-8 iSCSI : 0
/vol/host1_ptvol/vm2_pt	10.0G	online	3550-8 iSCSI : 2
/vol/host1_ptvol/vm3_pt	10.0G	online	3550-8 iSCSI : 4
/vol/host1_ptvol/vm4_pt	10.0G	online	3550-8 iSCSI : 6
/vol/host1_ptvol/vm5_pt	10.0G	online	3550-8 iSCSI : 8
/vol/host1_ptvol/vm6_pt	10.0G	online	3550-8 iSCSI : 10
/vol/host1_ptvol/vm7_pt	10.0G	online	3550-8 iSCSI : 12
/vol/host1_ptvol/vm8_pt	10.0G	online	3550-8 iSCSI : 14
/vol/host1_vhdvol/vm1_vhd	20.0G	online	3550-8 iSCSI : 1
/vol/host1_vhdvol/vm2_vhd	20.0G	online	3550-8 iSCSI : 3
/vol/host1_vhdvol/vm3_vhd	20.0G	online	3550-8 iSCSI : 5
/vol/host1_vhdvol/vm4_vhd	20.0G	online	3550-8 iSCSI : 7
/vol/host1_vhdvol/vm5_vhd	20.0G	online	3550-8 iSCSI : 9
/vol/host1_vhdvol/vm6_vhd	20.0G	online	3550-8 iSCSI : 11
/vol/host1_vhdvol/vm7_vhd	20.0G	online	3550-8 iSCSI : 13
/vol/host1_vhdvol/vm8_vhd	20.0G	online	3550-8 iSCSI : 15
/vol/host2_ptvol/vm10_pt	10.0G	online	stx3550-1 iSCSI : 0
/vol/host2_ptvol/vm11_pt	10.0G	online	stx3550-1 iSCSI : 2
/vol/host2_ptvol/vm12_pt	10.0G	online	stx3550-1 iSCSI : 4
/vol/host2_ptvol/vm13_pt	10.0G	online	stx3550-1 iSCSI : 6
/vol/host2_ptvol/vm14_pt	10.0G	online	stx3550-1 iSCSI : 8
/vol/host2_ptvol/vm15_pt	10.0G	online	stx3550-1 iSCSI : 10
/vol/host2_ptvol/vm16_pt	10.0G	online	stx3550-1 iSCSI : 12
/vol/host2_ptvol/vm9_pt	10.0G	online	stx3550-1 iSCSI : 14
/vol/host2_vhdvol/vm10_vhd	20.0G	online	stx3550-1 iSCSI : 1
/vol/host2_vhdvol/vm11_vhd	20.0G	online	stx3550-1 iSCSI : 3
/vol/host2_vhdvol/vm12_vhd	20.0G	online	stx3550-1 iSCSI : 5
/vol/host2_vhdvol/vm13_vhd	20.0G	online	stx3550-1 iSCSI : 7
/vol/host2_vhdvol/vm14_vhd	20.0G	online	stx3550-1 iSCSI : 9
/vol/host2_vhdvol/vm15_vhd	20.0G	online	stx3550-1 iSCSI : 11
/vol/host2_vhdvol/vm16_vhd	20.0G	online	stx3550-1 iSCSI : 13
/vol/host2_vhdvol/vm9_vhd	20.0G	online	stx3550-1 iSCSI : 15

Figure 16) FAS3160 LUNs configured for tests with iSCSI.

4.4 VIRTUAL MACHINES

After provisioning storage and presenting it to Hyper-V R2 hosts, a single virtual machine was created on each Hyper-V R2 host following best practices (see NetApp Technical Reports TR-3702 and TR-3733). This golden virtual machine was installed with the IOMeter dynamo application that was used to generate the load for all test cases. This golden image vhd file was then used to provision the remaining virtual machines on all the Hyper-V R2 hosts. Each virtual machine was configured with its own static IP address.

A separate server was set up to work as the Active Directory® domain controller and DNS server for all the Hyper-V R2 hosts and VMs. This server also hosted the IOMeter controller, which drove the load against dynamo applications in each VM.

The following settings were used on each Hyper-V R2 host to prevent host-side bottlenecks that might limit performance:

- FC HBA queue depth: 128
- iSCSI queue depth: 255

4.5 IOMETER CONFIGURATION

We used the publicly available IOMeter software to simulate workloads and run tests. IOMeter is a client server application that works both as a workload generator and a measurement tool. The client portion is called **dynamo**, which was installed on each of the VMs. The server portion is called the **IOMeter controller**, which was installed on the standalone server different from the Hyper-V R2 hosts (mentioned on page 21). This server also provided the AD and DC functionality. The IOMeter controller was used to manage the dynamo load generators running on the VMs and gather test results from each of the dynamos.

IOMeter generates a data file on each VM on the disk under test, then performs read and write operations to this file during testing. We created a 10GB IOMeter data file on each VM. Before running the tests, we had to initialize the disk with the data file that was used by IOMeter:

1. Power on all VMs on all Hyper-V R2 hosts.
2. Run the dynamo application on each VM.
3. Create an IOMeter access specification to generate a 100% random read load using 4K request sizes.
4. Execute this for five minutes on all VMs.

This test was conducted only once to initialize the IOMeter data files and the results of this test were not used. After the IOMeter files were initialized, we shut down all the VMs and Hyper-V R2 hosts and created a Snapshot™ copy of the aggregate containing the FlexVol volumes and LUNs defined above. Using NetApp Snapshot technology allowed us to restore the IOMeter data to its initial state between tests to provide consistent results between test runs.

4.6 TEST RUN DETAILS

The access specifications used were a mixture of random reads and writes using varying request sizes to represent real-world workloads as follows:

- 75% read, 25% write, 100% random using a 4K request size
- 75% read, 25% write, 100% random using an 8K request size

For each workload, we measured the following:

- Throughput in IOPS of 4Gb FC, 1GbE iSCSI, and 10GbE iSCSI protocols
- Latency in milliseconds of 4Gb FC, 1GbE iSCSI, and 10GbE iSCSI protocols
- Average % total CPU time for the Hyper-V hypervisor Logical Processor across all four Hyper-V R2 hosts

Each of the access specifications was run at three different workload levels simulating light, average, and high loads. All 32 of the VMs participated in each test and ran a single IOMeter worker. The load was varied by changing the number of outstanding I/Os generated by IOMeter from each VM (Table 4).

Table 4) Different load points based on outstanding I/Os.

VMs	Outstanding I/Os per VM	Total Outstanding I/Os
32	4	128
32	8	256
32	16	512

Tests ran for a total of five minutes with a five-minute ramp-up time to allow performance to reach a stable state before IOMeter began to record results. We also collected perfstat data on the FAS3160 storage controllers during the five-minute test run to enable a match between results on the storage side and on the host side. Each FAS3160 storage controller serviced 16 VMs, making a total working set of 160GB (10GB per VM). This working set is large enough so that the data served to the hosts was not from the FAS3160A cache and resulted in modest to high levels of disk activity on FAS3160A, depending on the load generated.

In order to start the tests and capture the results at the same point for all the protocols, the aggregate Snapshot copies were restored to the point at which they were initialized after a set of three tests was finished for a particular specification. The Snapshot copies were also restored for test runs between different protocols.

5 REFERENCES

- [TR-3702: NetApp Storage Best Practices for Microsoft Virtualization](#)
- [TR-3733: NetApp Implementation Guide for Microsoft Virtualization](#)

6 ACKNOWLEDGEMENTS

Special thanks to the following people for their contributions:

- Angus Wang, Senior Product Manager, NetApp
- Christopher Lemmons, Senior Manager, Workload Engineering, NetApp
- Keith Griffin, EDCA Lab Support, NetApp
- Lee Gates, Senior Manager, Microsoft Alliance Engineering, NetApp
- Kenny Speer, Member of Technical Staff, NetApp
- Ricky Stout, Lab Manager, NetApp
- Duaine Fioritto, Solutions Technologies Lab Manager, NetApp

NetApp provides no representations or warranties regarding the accuracy, reliability, or serviceability of any information or recommendations provided in this publication, or with respect to any results that may be obtained by the use of the information or observance of any recommendations provided herein. The information in this document is distributed AS IS, and the use of this information or the implementation of any recommendations or techniques herein is a customer's responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. This document and the information contained herein may be used solely in connection with the NetApp products discussed in this document.