FAQ: Cloud Volumes Service for AWS and SMB Performance Best Practices

Cloud Volumes Service for AWS was built for simplicity – delivering scalable storage to both cloud native as well as classic enterprise application. Whether accessed over NFS (v3 or v4.1) or SMB (2.1, 3.0, 3.1), Cloud Volumes Service for AWS delivers applications the performance they need when they need it. Much has been written regarding NFS performance, this FAQ will focus exclusively on SMB3/3.1.

Question 1: Is SMB Multichannel enabled by default in Cloud Volumes Service for AWS SMB Shares, what can be done if no?

SMB Multichannel is enabled by default

SMB Multichannel has been enabled as of Jan. 14, 2020. All pre-existing SMB volumes have had the feature enabled and all newly created volumes will have the feature enabled as well.

What do I need to do?

Any SMB connection established prior to feature enablement will need to be reset to take advantage of SMB Multichannel. Simply disconnect and reconnect the SMB share.

Does CVS Support Receive-side-scaling?

Once multichannel is enabled, an SMB3 client establishes multiple TCP connections to the CVS SMB server over a single receive-side scaling (RSS)-capable network interface card (NIC).

Which Windows versions support Multichannel?:

Windows has supported SMB Multichannel since Windows 2012, for best performance

Does my AWS Virtual Machine support RSS?

To see that your AWS Virtual Machine network interface cards support RSS, run the command. By default, the official Amazon Windows AMIs are configured with RSS enabled. Please see Amazon's enhanced networking guide for more information.

'Get-SmbClientNetworkInterface' as follows and pay attention to the field **'RSS Capable'**

```
PS 2:\> Get-SmbClientNetworkInterface

Interface Index    RSS Capable    RDMA Capable    Speed    IpAddresses                Friendly Name
---------------    -----------    ------------    -----    -----------                -------------
15                 True           False           40 Gbps  {fe80::244d:6d5:...        Ethernet 3
12                 True           False           40 Gbps  {fe80::b967:257b...        Ethernet
```

Question 2: What is the benefit of SMB Multichannel and what do I need to know when using it along with Cloud Volume Service for AWS?

The Multichannel feature enables an SMB3 client to establish a pool of connections over a single network interface card (NIC) or multiple NICs and use them to send requests for a single SMB session. In contrast, SMB1 and SMB2, by design, require the client to establish one connection and send all the SMB traffic for a given session over that connection. This single connection limits the overall protocol performance that can be achieved from a single client.

Is there value in configuring multiple NICS on my client – for SMB?

Short answer – No.

Long answer -The SMB client will match the nic count returned by the SMB server. Each storage volume is accessible from one and only one storage endpoint, which means that only one nic will be used for any given SMB relationship.

As the output of **'Get-SmbClientNetworkInterace'** below shows, the virtual machine has two network interface "**15**" and "**12.**" As shown below under the command '**Get-SMbMultiChannelConnections**', even though there are two RSS capable nics, only interface 12 is used in connection with the smb share which proves interface "**15**" is not in use.

```
PS Z:\> Get-SmbClientNetworkInterface

Interface Index     RSS Capable         RDMA Capable        Speed               IpAddresses         Friendly Name
---------------     -----------         ------------        -----               -----------         -------------
15                  True                False               40 Gbps             {fe80::244d:6d5:... Ethernet 3
12                  True                False               40 Gbps             {fe80::b967:257b... Ethernet
14                  False               False               0  bps              {fe80::5efe:10.9... isatap.uejkbpbp0...


PS Z:\> Get-SmbMultichannelConnection

Server Name     Selected        Client IP       Server IP       Client Interface Index  Server Interface Index  Client RSS Capable  Client RDMA Capable
-----------     --------        ---------       ---------       ----------------------  ----------------------  ------------------  -------------------
anf-b94c.no...  True            10.9.22.26      10.9.23.246     12                      12                      True                False
```

What about NIC Teaming, is it supported in AWS?

NIC Teaming is supported by AWS. Although there are many reasons for the use of multiple NICs on a VM, adding extra NICs for teaming will have no impact on the machine bandwidth. However, AWS implemented enhanced networking feature that supports network bandwidth up to 100Gpbs for supported EC2 instances.

Reference:

Enhanced Networking on Windows

https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/enhanced-networking.html

How performant is SMB Multichannel?:

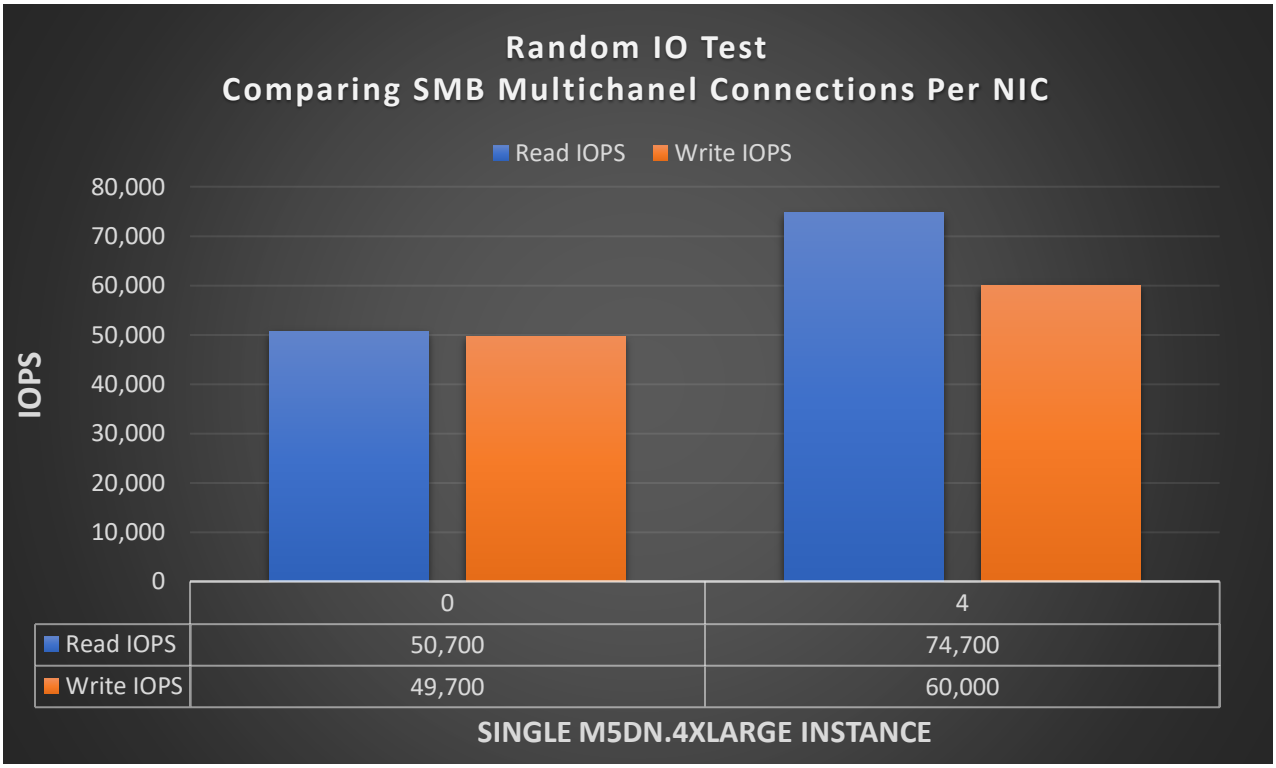
The following graph demonstrates the power of SMB Multichannel on single instance workloads. The graphs are divided into two or more sections:

- Left Section (0): this section demonstrates the storage performance before enabling SMB Multichannel

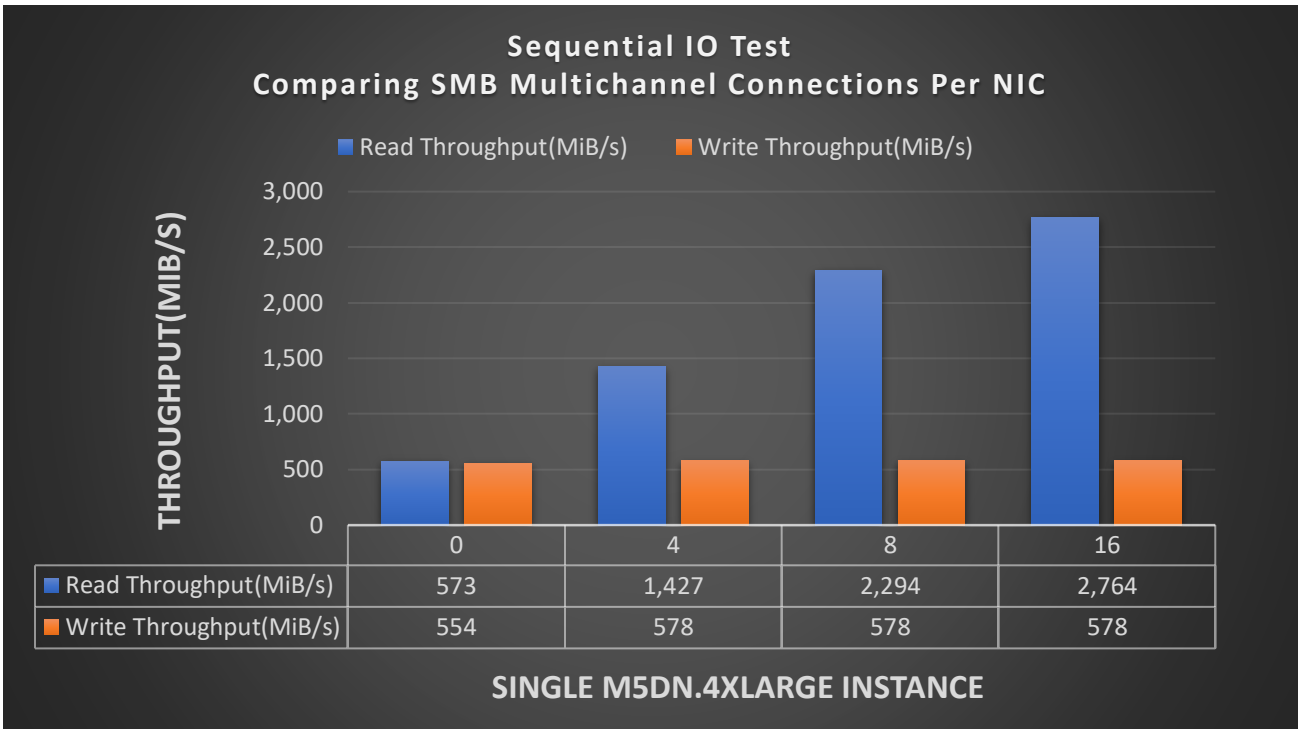
- Right Section (#): this section demonstrates the storage performance after enabling SMB Multichannel. The number ‘4’,’8’,’16’ shown in the X-Axis represents the number of connections per RSS Network Interface.

Random I/O: Starting with Multichannel disabled on the client, pure 4KiB read and write tests were performed using FIO and a 40GiB workingset. Having detached the SMB share between each test and incrementing upward the SMB client connection count per rss network interface setting ‘1’,’4’,’8’,’16’ **‘set-SmbClientConfiguration -ConnectionCountPerRSSNetworkInterface <count>’** therein, the tests show that the default setting of ‘4’ sufficient for I/O intensive workloads as incrementing upwards to ‘8’ and ‘16’ had no effect.

The command **‘netstat -na | findstr 445’** proved that additional connections were established when incrementing from ‘1’ to ‘4’ to ‘8’ and to ‘16’. With that said, four CPU cores were fully utilized for SMB during each test, as confirmed by inspecting the perfmon **‘Per Processor Network Activity Cycles’** statistic, which is not included in this report.



Sequential IO: Similar tests to the above were run, except with 64KiB sequential I/O. While the increases in client connection count per RSS network interface beyond ‘4’ had no noticeable effect upon random I/O, the same could not be said for sequential workloads. As the following graph shows, with each increase came a corresponding increase in read throughput. Write throughput remained flat due to network bandwidth restrictions placed by AWS upon each instance type/size.



	0	4	8	16
Read Throughput(MiB/s)	573	1,427	2,294	2,764
Write Throughput(MiB/s)	554	578	578	578

Question 3: What about jumbo frames

Jumbo frames are supported with AWS Virtual Machines, at the time of this writing Jumbo Frames are not supported in connection with the Cloud Volumes Service for AWS.

SMB Signing: What is it? Is it supported by Cloud Volumes Service for AWS? What is the performance impact of its use?

What is it? The Server Message Block (SMB) protocol provides the basis for file and print sharing and many other networking operations, such as remote Windows administration. To prevent man-in-the-middle attacks that modify SMB packets in transit, the SMB protocol supports the digital signing of SMB packets.

Is SMB Signing supported? SMB Signing is supported for all SMB protocol versions supported by Cloud Volumes Service for AWS.

What is the performance impact? SMB Signing has a deleterious effect upon SMB performance. Among potential other causes of the performance degradation, the digital signing of each packet consumes additional client-side CPU as the perfmon output below shows: Core 0 is responsible, in this case, for SMB including—it would seem—SMB signing. Comparisons to the non-multichannel sequential read throughput numbers in the previous section demonstrate that SMB signing reduced overall throughput from 573MiB/s to ~250MiB/s.

FIO Configuration Parameter File

[global]

```
name=fio-test
directory=.\       #This is the directory where files are written
```

```
ioengines=windowsaio # Windows native asynchronous I/O. Default on Windows.
direct=1           #Use directio
numjobs=1          #To match how many users on the system
nrfiles=4          #Num files per job
runtime=300        #If time_based is set, run for this amount of time

time_based         #This setting says run the jobs until this much time has
elapsed
stonewall
bs=64K
rw=rw||randrw      #choose rw if sequential io, choose randrw for
random io
rwmixread=100||0   #<-- Modify to get different i/o distributions
iodepth=TBD        #<-- Modify this to get the i/o they want (latency * target op count)
size=40G           #Aggregate file size per job (if nrfiles = 4, files=2.5GiB)
ramp_time=20       #Warm up
[test]
```