



The science behind the report:

Performance benchmark results: Amazon Web Services (AWS) SAN in the Cloud vs. comparable on-premises all-flash SAN solution

This document describes what we tested, how we tested, and what we found. To learn how these facts translate into real-world benefits, read the report [Performance benchmark results: Amazon Web Services \(AWS\) SAN in the Cloud vs. comparable on-premises all-flash SAN solution](#).

We concluded our hands-on testing on August 30, 2021. During testing, we determined the appropriate hardware and software configurations and applied updates as they became available. The results in this report reflect configurations that we finalized on August 27, 2021 or earlier. Unavoidably, these configurations may not represent the latest versions available when this report appears.

Our results

To learn more about how we have calculated the wins in this report, go to <http://facts.pt/calculating-and-highlighting-wins>. Unless we state otherwise, we have followed the rules and principles we outline in that document.

Table 1 shows the results of the median runs from our HammerDB testing of both solutions, and Table 2 shows the results of the median runs from our CrystalDiskMark testing of both solutions in multiple profiles.

Table 1: The results of our HammerDB testing.

Solution	New orders per minute (NOPM)	Total input/output operations per second (IOPS)	Average throughput in MB per second
On-premises SAN	762,132	178,172	1,594.59
EBS io2 Block Express storage and EC2 R5b.24xlarge instance	787,901	182,479	1,637.62

Table 2: The results of our CrystalDiskMark testing.

I/O profile	Workers/threads	I/O queue depth	File size	Throughput (GB/s)	IOPS	Latency (ms)
On-premises SAN						
8k all read random	8	32	One 64GB	3.388	413,691	0.59
8k all write random	8	32	One 64GB	1.797	219,406	1.12
64k all read sequential	8	32	One 64GB	6.187	94,415	2.64
64k all write sequential	8	32	One 64GB	4.263	65,062	3.86
EBS io2 Block Express storage and EC2 R5b.24xlarge instance						
8k all read random	8	32	One 64GB	2.148	262,304	0.97
8k all write random	8	32	One 64GB	2.148	262,235	0.97
64k all read sequential	8	32	One 64GB	7.852	119,820	2.13
64k all write sequential	8	32	One 64GB	6.226	95,005	2.68

System configuration information

Table 3: Detailed information on the system we tested.

Configuration information	HP ProLiant DL380 Gen10	R5b.24xlarge instance
Tested by	Principled Technologies	Principled Technologies
Test date	07/09/2021	07/15/2021
Cloud service provider/region	N/A	us-east-1c
Workload and version	HammerDB v4.2 TPROC-C	HammerDB v4.2 TPROC-C
Workload specific parameters	30,000 Warehouses 144 virtual users	30,000 Warehouses 144 virtual users
Iterations and result choice	Three runs, median	Three runs, median
Server platform	HPE ProLiant DL380 Gen10	R5b.24xlarge instance
BIOS name and version	HP U30 v2.42	Amazon EC2 1.0, 10/16/2017
Operating system name and version/build number	Microsoft Windows Server 2019 Datacenter Edition Version 10.0.17763 (Build 17763)	Microsoft Windows Server 2019 Datacenter Edition Version 10.0.17763 (Build 17763)
Date of last OS updates/patches applied	07/02/2021	07/12/2021
Processor		
Number of processors	2	1
Vendor and model	Intel® Xeon® Gold 6240R	Intel Xeon Platinum 8259CL
Core count (per processor)	24	24
Core frequency (GHz)	2.40	2.50
Stepping	7	7
Hyper-Threading	Yes	Yes
Turbo	Yes	Yes
Number of vCPU per VM	N/A	96
Memory module(s)		
Total memory in system (GB)	768	768
NVMe memory present?	No	No
Total memory (DDR + NVMe RAM)	768	768
General hardware		
Storage	Network attached	Network attached
Network bandwidth/instance (Gbps)	16	25
Storage bandwidth/instance (Mbps)	4,750	60,000

Configuration information	HP ProLiant DL380 Gen10	R5b.24xlarge instance
Storage		
OS		
Number of drives	2	1
Drive size (GB)	400	40
Drive information (speed, interface, type)	6Gbps SATA SSD (SSDSC2BA400G3)	gp2, EBS, 100/3000 IOPS
Data drive		
Number of drives	48	2
Drive size (TB)	1.92	2
Drive information (speed, interface, type)	SAS SSD (DOPE1920S5xnNMR1)	io2 BX, EBS, 130000 IOPS
Temporary drive		
Number of drives	0	0
Drive size (GB)	N/A	N/A
Network adapter		
Vendor and model	2 x HPE SN1100Q	Amazon Elastic Network Adapter
Number and type of ports	2 x 16Gb	1 x 25Gb

How we tested

For the on-premises solution, we configured an HPE ProLiant DL380 Gen10 server and an HPE 3PAR StoreServ 8450. For the AWS solution, we configured an EC2 R5b.24xlarge instance and EBS io2 Block Express (BX) volumes. The EBS data storage was comprised of two 2TB io2 BX volumes striped together in the OS to reach the R5b.24xlarge instance's maximum IOPS and throughput limits. The on-premises data storage was comprised of a two 2TB RAID 1 volume created on the 3PAR storage using its 48 available disks for the OS via Multipath I/O (MPIO). We used Microsoft Windows Server 2019's default settings (Round Robin) to configure the storage for the OS and striped disks together in the same manner as the EBS volumes. We compared the OLTP performance from the TPROC-C workload from HammerDB 4.2. We used a 30,000-warehouse database size with 144 virtual users driving the tests.

Configuring the HPE ProLiant DL380 Gen10 server

Installing Microsoft Windows Server 2019

1. Insert the Microsoft Windows Server 2019 installation media.
2. Power on the server, and boot into the Windows installer.
3. Once the installer has copied the installation files into the memory, select the appropriate language, time and currency format, and keyboard layout. Click Next.
4. Click Install Now.
5. Enter the product key, and click Next.
6. Choose Windows Server 2019 Datacenter (Server with a GUI), and click Next.
7. Check I accept the license terms, and click Next.
8. Choose Custom: Install Windows only (advanced).
9. From the list, select the OS volume, and click Next. This will begin the installation.

Configuring Windows Update on the server

1. In the left pane of the Server Manager window, click Local Server.
2. In the main frame, next to Windows Update, click Not configured.
3. In the Windows Update window, in the main pane, click Let me choose my settings.
4. Under Important updates, select Never check for updates (not recommended), and click OK.
5. In the left pane, click Check for updates, and install all available updates.
6. Close the Windows Update window.

Creating the HPE 3PAR StoreServ 8450 volumes

1. In the 3PAR StoreServ portal, navigate to 3PAR StoreServ→Virtual Volumes.
2. Click Create virtual volume.
3. Choose the appropriate Name and System for the volume.
4. In the Provisioning drop down menu, choose Full Provisioned.
5. Choose the appropriate CPG and Size for the volume you are creating. We used a RAID 1 configuration.
6. In the Export section, click Add and choose the host you would like to export the volume to.
7. Click Create.
8. Complete steps 1 through 7 until you have created two 2TB data volumes and one 2TB backup volume for the SUT.

Configuring MPIO on Microsoft Windows Server 2019 hosts for the on-premises solution

1. Open Server Manager, and click Manage→Add Roles and Features.
2. In the Add Roles and Features Wizard, click Next.
3. Choose Role-based or Feature-based installation, and click Next.
4. Choose Select a server from the server pool, choose the local server, and click Next.
5. On the Select server roles screen, click Next.
6. On the Select features screen, choose Multipath I/O, and click Next.
7. On the Confirmation screen, click Install.
8. When the installation is complete, reboot the server.
9. Take the following actions:
 - a. When the server finishes rebooting, open the Start menu, and type MPIO.
 - b. In the MPIO app, click the Discover Multi-Paths tab.
 - c. Select the desired multipath device, and click Add.
 - d. When the app is finished, close the window.

Adding the drives in Windows Server 2019

1. Log into the Windows Server 2019 host.
2. Right-click the Start menu, and click Disk Management.
3. Right-click the new disks, and click Online.
4. Right-click the new disks, and click Initialize. At the pop-up, ensure GPT is selected as the initialization type, and click OK.
5. Right-click the log disk, and click New Simple Volume.
6. To create a new drive with the chosen drive letter and NTFS formatting, follow the wizard.
7. In Windows Explorer, right-click the drive, and choose Format.
8. In the Allocation unit size field, choose 64K, and click Start.

You will strip together the data drives later according to the steps in "Partitioning and formatting the data drives."

Creating the Windows Server 2019 client and SQL instances on AWS

Creating the baseline instance

1. Log into AWS, and navigate to the AWS Management Console.
2. Click EC2
3. Click Launch instance, and to open the Launch Instance in the drop down wizard, click Launch instance.
4. In the search window, type `Windows Server`, and press enter.
5. Next to Microsoft Windows Server 2019 Base on Quick Start, click Select.
6. On Choose Instance Type, select R5b.24xlarge, and click "Next: Configure Instance Details."
7. On Configure Instance, set the following:
 - a. Number of instances: 1
 - b. Purchasing option: Leave unchecked
 - c. Network: Default VPC
 - d. Subnet: Choose the region you are working in
 - e. Auto-assign Public IP: Enable
 - f. Placement Group: Leave unchecked
 - g. Capacity Reservation: Open
 - h. Domain join directory: No Directory
 - i. IAM role: None
 - j. Shutdown behavior: Stop
 - k. Click Next: Add Storage
8. On Add Storage, set the following:
 - a. Size: 30GB
 - b. Volume Type: gp2
 - c. Delete on Termination: Checked
 - d. Encryption: Not Encrypted
 - e. Click Next: Add Tags
9. On Add Tags, add any tags you wish to use. Click Next: Configure Security Group.
10. On Configure Security Group, leave defaults, and click Review and Launch.
11. On Review, click Launch.
12. Choose the appropriate option for the key pair, and click Launch Instances.
13. Repeat these steps to create a second instance, changing the instance type in step 6 to m5.8xlarge, to be used as a client instance.

Configuring Windows Server 2019

1. Open Server Manager, and click Local Server.
2. Disable IE Enhanced Security Configuration.
3. Change the time zone to your local time zone.
4. Change the name of your server, and when prompted, reboot.
5. Open Server Manager again, and click Local Server.
6. Click to run updates.
7. Run updates, rebooting when prompted, until the server shows no new updates to install.

Adding networking rules to security group

1. Log into AWS, and navigate to the AWS Management Console.
2. Click EC2.
3. Click Security Groups.
4. Choose your newly created Security Group.
5. Under Inbound rules, click Edit inbound rules.
6. Click Add Rule.
7. Under Type, select All traffic.
8. Under Source, select the Security Group.
9. Click Add Rule.
10. Under Type, select RDP.
11. Under source, select My IP.
12. Click Save Rules.

Creating and attaching data disks

1. Log into AWS, and navigate to the AWS Management Console.
2. Click EC2.
3. In the navigation pane, choose Elastic Block Store, Volumes.
4. Click Create Volume.
5. Under Volume Type, select io2 Block Express.
6. Set size to 2,048GB.
7. Choose your desired Availability Zone.
8. Add any Tags you wish to use.
9. Click Create Volume.
10. On the Elastic Block Store, Volumes page, select your newly created drive.
11. Click Actions, and select Attach Volume.
12. From the list, choose your instance, and click Attach.
13. Complete steps 4 through 12 to create a second attached data disk of the same size.

Partitioning and formatting the data drives (AWS and on-premises solutions)

1. Log into the VM or server under test.
2. Open a PowerShell instance with Administrator privileges.
3. To pool, partition, and format the disks as a single striped volume, run the following command:

```
$PhysicalDisks = Get-PhysicalDisk | Where-Object {$_.CanPool -eq $true}
New-StoragePool -FriendlyName "DataFiles" -StorageSubsystemFriendlyName "Windows Storage*" `
-PhysicalDisks $PhysicalDisks | New-VirtualDisk -FriendlyName "Datafiles" `
-Interleave 65536 -NumberOfColumns $PhysicalDisks.Count -ResiliencySettingName simple `
-UseMaximumSize | Initialize-Disk -PartitionStyle GPT -PassThru | New-Partition -AssignDriveLetter `
-UseMaximumSize | Format-Volume -FileSystem NTFS -NewFileSystemLabel "DataDisks" `
-AllocationUnitSize 65536 -Confirm:$false
```

Installing and running CrystalDiskMark 8.0.4 (AWS and on-premises solutions)

Installing CrystalDiskMark 8.0.4

1. In a browser, navigate to crystalmark.info/en/download/#CrystalDiskMark
2. Download the relevant zip file.
3. Unzip the folder and double-click DiskMark64.

Configuring and running CrystalDiskMark 8.0.4

1. In the CrystalDiskMark window, click Settings→Settings
2. Under Profile: Default, set the following test parameters:
3. Type: SEQ, Block Size: 8KiB, Queues: 32, Threads: 8
4. Type: RND, Block Size 8KiB, Queues: 32, Threads: 8
5. Type: SEQ, Block Size: 64KiB, Queues 32, Threads 8

6. Type: RND, Block Size 64KiB, Queues 32, Threads 8
7. Click OK.
8. To begin the tests, click "All" in the top left corner.
9. Once the tests have finished running, click File→Save (text) to save your output as a text file.

Configuring Microsoft SQL Server 2019 (AWS and on-premises solutions)

Installing Microsoft SQL Server 2019

1. Attach the SQL Server 2019 installation media .ISO to the VM.
2. Click Run SETUP.EXE.
3. In the left pane, click Installation.
4. Click New SQL Server stand-alone installation or add features to an existing installation.
5. To accept the license terms, click the check box, and click Next.
6. Click Use Microsoft Update to check for updates, and click Next.
7. To install the setup support files, click Install.
8. If the installer doesn't display any failures, click Next.
9. At the Setup Role screen, choose SQL Server Feature Installation, and click Next.
10. At the Feature Selection screen, select Database Engine Services, Full-Text and Semantic Extractions for Search, Client Tools Connectivity, and Client Tools Backwards Compatibility.
11. Click Next.
12. At the Instance Configuration screen, leave the instance default selection, and click Next.
13. At the Database Engine Configuration screen, select the authentication method you prefer. For our testing purposes, we selected Mixed Mode.
14. For the system administrator account, enter and confirm a password.
15. Click Add Current user. This may take several seconds.
16. Click the Data Directories tab.
17. Change the database directory, the database log directory, and the backup directory storage locations to the data, log, and backup volumes.
18. Click TempDB.
19. Change the number of files to eight.
20. Change the initial file size to 1,024 MB.
21. Change the initial size of the log to 1,024 MB.
22. Click Next.
23. At the Error and usage reporting screen, click Next.
24. At the Installation Configuration Rules screen, check that there are no failures or relevant warnings, and click Next.
25. At the Ready to Install screen, click Install.
26. Once the installation finishes, navigate to the Installation tab in the Installation Center, and click Install SQL Server Management Tools.
27. In the browser that opens, click Download SQL Server Management Studio 18.9.1.
28. To open the installer, click the download.
29. Click Run.
30. Click Install.
31. Once the installation completes, close the installation window.

Enabling the Windows option for locking pages in memory

1. In the Start menu, click Run. In the Open box, type `gpedit.msc`
2. On the Local Group Policy Editor console, expand Computer Configuration, and expand Windows Settings.
3. Expand Security Settings, and expand Local Policies.
4. Select the User Rights Assignment folder.
5. In the details pane, double-click Lock pages in memory.
6. In the dialog box for Local Security Setting - Lock pages in memory, click Add User or Group.
7. In the dialog box for Select Users, Service Accounts, or Groups, select the SQL Server Service account.
8. For this setting to take effect, restart the SQL Server Service.

Setting the maximum memory in SQL Server Management Studio (AWS and on-premises solutions)

1. In SQL Server Management Studio, connect to the desired SQL Server database engine, right-click the desired instance, and click Properties.
2. In the Server Properties dialog box, from the left-side list, select the Memory item.
3. In the Maximum server memory (in MB) option, type 707788 .
4. Click OK.

Configuring the test database

Creating the database

1. Open SQL Server Management Studio.
2. Right-click Databases→New Database.
3. Name the database. We named ours tpcc.
4. Navigate to the Filegroups tab, and create a new primary filegroup.
5. Navigate to the Files tab, and to add a data file, click Add.
6. Grow the data file to 3TB by inputting the size in MB in the Size column.
7. Grow the log file to 96GB in steps of 8GB, by inputting the size of the step in MB in the Size column, clicking OK, and repeating until the log has grown to 96GB.
8. Name the database files, and click OK.

Installing HammerDB on the client system

1. Download the latest version of HammerDB from www.hammerdb.com/download.html. We used 4.2.
2. Double-click the .EXE file, choose English, and click OK.
3. Click Yes.
4. Click Next.
5. Choose a destination location, and click Next.
6. Click Next.
7. Click Finish.

Populating the database

1. Open HammerDB, and click Options→Benchmark.
2. Choose MSSQL Server and TPROC-C.
3. Expand SQL Server→TPROC-C→Schema Build.
4. Double-click Options.
5. Input the IP of the SUT in the SQL Server field.
6. Select 30000 warehouses and 16 virtual users.
7. Click OK.
8. Double-click Build. This build could take a few days.

Backing up the database

1. Open SQL Server Management Studio.
2. Right-click the TPROC-C database, and click Tasks→Back up....
3. Choose a location to store the backup, and click OK.
4. Performing the test on Windows Server 2019
5. On the client system, start HammerDB.
6. Set the database server to SQL Server, and set the workload to TPROC-C.
7. Open the Options panel for the Driver Script: SQL Server→TPROC-C→Driver Script→Options.
8. Enter the IP of the system under test in the SQL Server field.
9. Change the ODBC Driver to ODBC Driver 17 for SQL Server.
10. Choose SQL Server Authentication, and change the SQL Server User Password to the password you chose during SQL Server setup.
11. Choose Timed Driver Script.
12. Change the Total Transactions per User to 1000000000.
13. Enter 10 for the Rampup Time, and 20 for the Test Duration.
14. Check the box next to Use All Warehouses.

15. Click OK.
16. Open the Options panel for the Virtual Users: SQL Server→TPROC-C→Virtual User→Options.
17. Use 144 Virtual Users.
18. Select the following: Show Output, Log Output to Temp, and Use Unique Log Name.
19. Click OK.→Start a custom Perfmon data collector set to capture CPU, RAM, and disk performance counters.
20. Click the green arrow.
21. Note the name of the log file. To start the run, click OK.
22. When the run finishes, stop Perfmon.
23. Save the HammerDB results text file and Perfmon output.
24. Drop and restore the database from the system under test, reboot the system.
25. Complete steps 1 through 24 three times, and collect the median run.

Performing the test on Windows Server 2019

1. On the client system, start HammerDB.
2. Set the database server to SQL Server, and set the workload to TPROC-C.
3. Open the Options panel for the Driver Script: SQL Server→TPROC-C→Driver Script→Options.
4. Enter the IP of the system under test in the SQL Server field.
5. Change the ODBC Driver to ODBC Driver 17 for SQL Server.
6. Choose SQL Server Authentication, and change the SQL Server User Password to the password you chose during SQL Server setup.
7. Choose Timed Driver Script.
8. Change the Total Transactions per User to 1000000000.
9. Enter 10 for the Rampup Time, and 20 for the Test Duration.
10. Select Use All Warehouses.
11. Click OK.
12. Open the Options panel for the Virtual Users: SQL Server→TPROC-C→Virtual User→Options.
13. Use 144 Virtual Users.
14. Select the following: Show Output, Log Output to Temp, and Use Unique Log Name.
15. Click OK.
16. To capture CPU, RAM, and disk performance counters, start a custom Perfmon data collector set.
17. Click the green arrow.
18. Note the name of the log file. To start the run, click OK.
19. When the run finishes, stop Perfmon.
20. Save the HammerDB results text file and Perfmon output.
21. Drop and restore the database from the system under test, reboot the system.
22. Complete steps 1 through 21 three times, and collect the median run.

Read the report at <http://facts.pt/JIBjXqh> ▶

This project was commissioned by AWS.



Facts matter.®

Principled Technologies is a registered trademark of Principled Technologies, Inc. All other product names are the trademarks of their respective owners.

DISCLAIMER OF WARRANTIES; LIMITATION OF LIABILITY:

Principled Technologies, Inc. has made reasonable efforts to ensure the accuracy and validity of its testing, however, Principled Technologies, Inc. specifically disclaims any warranty, expressed or implied, relating to the test results and analysis, their accuracy, completeness or quality, including any implied warranty of fitness for any particular purpose. All persons or entities relying on the results of any testing do so at their own risk, and agree that Principled Technologies, Inc., its employees and its subcontractors shall have no liability whatsoever from any claim of loss or damage on account of any alleged error or defect in any testing procedure or result.

In no event shall Principled Technologies, Inc. be liable for indirect, special, incidental, or consequential damages in connection with its testing, even if advised of the possibility of such damages. In no event shall Principled Technologies, Inc.'s liability, including for direct damages, exceed the amounts paid in connection with Principled Technologies, Inc.'s testing. Customer's sole and exclusive remedies are as set forth herein.