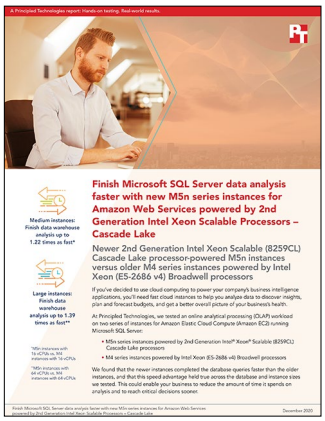

The science behind the report:

Finish Microsoft SQL Server data analysis faster with new M5n series instances for Amazon Web Services powered by 2nd Generation Intel Xeon Scalable Processors – Cascade Lake

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 Finish Microsoft SQL Server data analysis faster with new M5n series instances for Amazon Web Services powered by 2nd Generation Intel Xeon Scalable Processors – Cascade Lake.

We concluded our hands-on testing on October 26, 2020. 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 October 11, 2020 or earlier. Unavoidably, these configurations may not represent the latest versions available when this report appears.

Our results

Table 1: Medium instance results. Source: Principled Technologies.

M-series 16 m4 vs m5n (30GB database)			
Time to complete in seconds	m4	m5n	m5n times as fast
1 stream	27	24	1.13
2 streams	43	37	1.16
3 streams	57	48	1.19
4 streams	73	60	1.22

Table 2: Large instance results. Source: Principled Technologies.

M-series 64 m4 vs m5n (100GB database)			
Time to complete in seconds	m4	m5n	m5n times as fast
1 stream	36	27	1.33
2 streams	55	41	1.34
3 streams	70	55	1.27
4 streams	89	64	1.39
5 streams	106	76	1.39

System configuration information

Table 3: Detailed information on the systems we tested.

System configuration information	m4.4xlarge	m4.16xlarge	m5n.4xlarge	m5n.16xlarge
Tested by	Principled Technologies	Principled Technologies	Principled Technologies	Principled Technologies
Test date	10/21/2020	10/21/2020	10/21/2020	10/21/2020
CSP/region	us-east1-a	us-east1-a	us-east1-a	us-east1-a
Workload & version	HammerDB v3.3 TPC-H-like	HammerDB v3.3 TPC-H-like	HammerDB v3.3 TPC-H-like	HammerDB v3.3 TPC-H-like
Workload-specific parameters	CCI, MAXDOP 16, Lock Pages in Memory, 90% Reserved SQL Memory	CCI, MAXDOP 64, Lock Pages in Memory, 90% Reserved SQL Memory	CCI, MAXDOP 16, Lock Pages in Memory, 90% Reserved SQL Memory	CCI, MAXDOP 64, Lock Pages in Memory, 90% Reserved SQL Memory
Iterations and result choice	3 runs, median	3 runs, median	3 runs, median	3 runs, median
Server platform	m4.4xlarge	m4.16xlarge	m5n.4xlarge	m5n.16xlarge
BIOS name and version	Xen 4.2.amazon, 8/24/2006	Xen 4.2.amazon, 8/24/2006	Amazon EC2 1.0, 10/16/2017	Amazon EC2 1.0, 10/16/2017
Operating system name and version/build number	Microsoft Windows Server 2019 Datacenter 10.0.17763 / Build 17763	Microsoft Windows Server 2019 Datacenter 10.0.17763 / Build 17763	Microsoft Windows Server 2019 Datacenter 10.0.17763 / Build 17763	Microsoft Windows Server 2019 Datacenter 10.0.17763 / Build 17763
Date of last OS updates/ patches applied	10/20/2020	10/20/2020	10/20/2020	10/20/2020
Processor				
Number of processors	1	2	1	2
Vendor and model	Intel® Xeon® CPU E5-2686 v4	Intel Xeon CPU E5-2686 v4	Intel Xeon Platinum 8259CL	Intel Xeon Platinum 8259CL
Core count (per processor)	18	18	24	24
Core frequency (GHz)	2.30	2.30	2.50	2.50
Stepping	1	1	7	7
Hyper-threading	Yes	Yes	Yes	Yes
Turbo	Yes	Yes	Yes	Yes
Number of vCPU per VM	16	64	16	64
Memory module(s)				
Total memory in system (GB)	64	256	63.2	253
NVMe memory present?	No	No	No	No
Total memory (DDR+NVMe RAM)	64	256	63.2	253
General hardware				
Storage: Network or Direct attached	Network	Network	Network	Network
Network bandwidth per instance	High	10 Gb	Up to 25 Gb	75 Gb
Storage bandwidth per instance	1,000 Mbps	10,000 Mbps	Up to 4,750 Mbps	13,600 Mbps

System configuration information	m4.4xlarge	m4.16xlarge	m5n.4xlarge	m5n.16xlarge
Local storage (OS)				
Number of drives	1	1	1	1
Drive size (GB)	50	100	50	100
Drive information (speed, interface, type)	gp2	gp2	gp2	gp2
Local storage (data drive)				
Number of drives	1	1	1	1
Drive size (GB)	50	120	50	120
Drive information (speed, interface, type)	io1	io1	io1	io1
Network adapter				
Vendor and model	Intel 82599 Virtual Function	Amazon Elastic Network Adapter	Amazon Elastic Network Adapter	Amazon Elastic Network Adapter
Number and type of ports	1x 10Gb	1x 25Gb	1x 25Gb	1x 25Gb

How we tested

Testing overview

For this project, we tested AWS instances featuring older Intel processors versus instances updated with 2nd Generation Intel Xeon Scalable processors. We ran a TPC-H-like workload on Microsoft SQL Server on the AWS instances to show the improvement in analysis completion times that customers could expect to see by choosing the updated M5n instances.

Using our methodology to aid your own deployments

While the methodology below describes in great detail how we accomplished our testing, it is not a deployment guide. However, because we include many basic installation steps for operating systems and testing tools, reading our testing methodology may help with your own installation.

Creating the Windows Server 2019 baseline image

1. Log into AWS and navigate to the AWS Management Console.
2. Click on EC2
3. To open the Launch Instance wizard, click Launch instance, and select Launch instance from the drop-down menu.
4. In the search window, enter `Windows Server` and press Enter.
5. On the Quick Start tab, next to Microsoft Windows Server 2019 Base, click the Select button.
6. On the Choose Instance Type tab, select t2.micro, and click Next: Configure Instance Details.
7. On the Configure Instance tab, 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 the Add Storage tab, 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 the Add Tags tab, add any tags you wish to use. Click Next: Configure Security Group.
10. On the Configure Security Group tab, leave defaults, and click Review and Launch.
11. On the Review Tab, click Launch.
12. Choose the appropriate option for the key pair, and click Launch Instances.

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 reboot.
5. Open Server Manager again, and click Local Server.
6. Click Updates.
7. Run updates, rebooting when prompted, until the server shows no new updates to install.

Installing SQL Server 2019 Enterprise

1. Download or copy the ISO to the server and unzip it.
2. Double-click the Setup application.
3. Click Installation→New SQL Server Standalone installation or add features to an existing installation.
4. Choose the trial version, and click Next.
5. Check the I accept the license terms and Privacy Statement box, and click Next.
6. Check the Use Microsoft Update to check for updates (recommended) box, and click Next.
7. On the Install Rules page, click Next.
8. Check the boxes for the following features, and click Next:
    a. Database Engine Services
    b. Full-Test and Semantic Extractions for Search
    c. Client Tools Connectivity
    d. Client Tools Backwards Compatibility
9. Leave the Default instance, and click Next.
10. Leave the default Service Accounts, and click Next.
11. On the Server Configuration tab, choose Mixed Mode and enter and confirm a Password for the SQL Server system administrator (sa) account.
12. Click Add Current User to Specify the SQL Server administrators.
13. Click Next.
14. Once you've passed the rule check, click Next.
15. Click Install.
16. When the install is finished, go back to the SQL Server Installation Center, and click Install SQL Server Management Tools.
17. Download the SSMS file, and install with defaults.
18. Reboot the server when prompted.
19. Run Windows Update one more time to ensure there aren't any new updates for SQL (make sure Windows Updates are set to get updates for other Microsoft products).
20. Once you've installed all available updates, disable Windows Update service. Click Start, type `services` to open the Services list, and disable the Windows Update service.

Configuring lock pages In memory

1. Click Start, and type `Local Security Policy`. Open the program when it pops up in the search.
2. Expand Local Policies, and click on User Rights Assignment.
3. In the right-hand pane, scroll down, and double-click Lock pages in memory.
4. Click Add User or Group, type `NT Service\MSSQLSERVER`, and click OK.
5. To close the Properties window, click OK, and close the Local Security Policy window.

Installing HammerDB 3.3

1. Download HammerDB from here: https://hammerdb.com/download.html.
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.

Creating an AMI of the baseline instance

1. Log into AWS, and navigate to the AWS Management Console.
2. Click EC2.
3. Click Running instances.
4. Place a checkmark next to the instance you wish to create an image from.
5. Click the Action drop-down, and select Image→Create Image.
6. Enter the Image name, and click Create Image.
7. To see the new image, navigate to Images→AMIs in the menu on the left-hand side of the page.

Creating an instance with the baseline image

1. Log into AWS, and navigate to the AWS Management Console.
2. Click EC2.
3. Click Images→AMIs.
4. Check the box next to the image you created in the previous step, and click Launch.
5. On the Choose Instance Type tab, select your VM size, and click Next: Configure Instance Details.
6. On the Configure Instance tab, 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
7. Click Next→Add Storage.
8. On the Add Storage tab, set the following:
    a. Size: <Size>
    b. Volume Type: Set your volume type. We chose io1.
    c. Delete on Termination: Unchecked
    d. Encryption: Not Encrypted
9. Click Next→Add Tags.
10. On the Add Tags tab, add any tags you wish to use. Click Next→Configure Security Group.
11. On the Configure Security Group tab, leave the defaults, and click Review and Launch.
12. On the Review Tab, click Launch.
13. Choose the appropriate option for the key pair, and click Launch Instances.

Configuring SQL on the instances under test

In this section, we list the various SQL settings that we changed and the steps to do so.

Setting the SQL memory reserve and max degree of parallelism (MAXDOP)

1. Open the SQL Server Management Studio.
2. Right-click on the SQL Instance, and click Properties.
3. Click Advanced node, scroll down to the Max Degree of Parallelism, and change the value. We set our MAXDOP on each VM instance to match the instance's number of vCPUs. Click OK.
4. Right-click the SQL Instance again, and go to Memory.
5. Set the Max Memory to 90% of the total memory in the system. Click OK, and close the Properties window.
6. Right-click the SQL Instance, and restart the service. Click Yes.

Configuring the tempdb database

1. Open the SQL Server Management Studio.
2. Expand Databases and System databases, and right-click tempdb.
3. Add files, and change the starting size as necessary.
4. Right-click the SQL instance, and restart the service. Click Yes.
5. To move the tempdb to the database drive, open a new Query, and run the following modified for the number of tempdb files your system has:

```
USE [master]
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = tempdev , FILENAME = 'E:\TempDB\tempdb.mdf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp2 , FILENAME = 'E:\TempDB\tempdb_mssql_2.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp3 , FILENAME = 'E:\TempDB\tempdb_mssql_3.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp4 , FILENAME = 'E:\TempDB\tempdb_mssql_4.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp5 , FILENAME = 'E:\TempDB\tempdb_mssql_5.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp6 , FILENAME = 'E:\TempDB\tempdb_mssql_6.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp7 , FILENAME = 'E:\TempDB\tempdb_mssql_7.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = temp8 , FILENAME = 'E:\TempDB\tempdb_mssql_8.ndf' )
GO
ALTER DATABASE tempdb MODIFY FILE ( NAME = templog , FILENAME = 'E:\TempDB\templog.ldf' )
GO
```

6. Right-click the SQL instance, and restart the service. Click Yes.

Running the tests

In this section, we list the steps to run the HammerDB TPC-H-like test on the instances under test. As each instance had different hardware and database sizes, please refer to Table XX to see the number of users to run on each instance. For the maximum number of users we ran, we followed HammerDB TPC-H recommendations for the database size we were testing. Additionally, to show the scaling of each instance pair, we ran with fewer users. Note that for each test we ran a single-stream test first to cache the database into memory before running the second test (normally multi-stream, the exception being the 1-stream test).

1. On the test instance, restore the database under test to so that the database and log files reside on the io1 SSD.
2. Configure SQL settings and tempdb according to the instructions above.
3. Open HammerDB.
4. Select Options→Benchmark.
5. Choose MSSQL Server and TPC-H.
6. Expand SQl Server→TPC-H→Schema Build.
7. Double-click Options, change the driver to ODBC Driver 17 for SQL Server, set the scale to match your database, set MAXDOP to match SQL's, and check the box for Clustered Columnstore. Click OK.
8. Expand Driver Script, and double-click Options, and click OK to load.
9. Expand Virtual User, and double-click Options.
10. Choose 1 user.
11. Check the boxes for Show Output, Log Output to Temp, and Use Unique Log Name.
12. Click OK.
13. To load the Driver Script, double-click Load.
14. Double-click Create users.
15. To capture performance metrics on the system, start Performance monitor set to record CPU, Memory, and drive usage information.
16. To begin the run, click Start.
17. When the run finishes, stop Perfmon, and save the HammerDB results file and Perfmon output.
18. Stop the HammerDB user.
19. Double-click User options again, and set the number of users to the appropriate count for the multi-stream test.
20. Double-click Create users.
21. To capture performance metrics on the system, set the Performance monitor to record CPU, Memory, and drive usage information, and start it.

22. To begin the run, click Start.
23. When the run finishes, stop Perfmon, and save the HammerDB results file and Perfmon output.
24. Reboot the instance.
25. Repeat the test two more times for a total of three runs at each user count, and record the median run.

Determining CPU vulnerability mitigation

The following figures show the Intel processor mitigation settings on the AWS instances.

```
PS C:\Users\Administrator> Get-SpeculationControlSettings
For more information about the output below, please refer to https://support.microsoft.com/help/4074629

Speculation control settings for CVE-2017-5715 [branch target injection]

Hardware support for branch target injection mitigation is present: False
Windows OS support for branch target injection mitigation is present: True
Windows OS support for branch target injection mitigation is enabled: False
Windows OS support for branch target injection mitigation is disabled by system policy: False
Windows OS support for branch target injection mitigation is disabled by absence of hardware support: True

Speculation control settings for CVE-2017-5754 [rogue data cache load]

Hardware requires kernel VA shadowing: True
Windows OS support for kernel VA shadow is present: True
Windows OS support for kernel VA shadow is enabled: True
Windows OS support for PCID performance optimization is enabled: True [not required for security]

Speculation control settings for CVE-2018-3639 [speculative store bypass]

Hardware is vulnerable to speculative store bypass: True
Hardware support for speculative store bypass disable is present: False
Windows OS support for speculative store bypass disable is present: True
Windows OS support for speculative store bypass disable is enabled system-wide: False

Speculation control settings for CVE-2018-3620 [L1 terminal fault]

Hardware is vulnerable to L1 terminal fault: True
Windows OS support for L1 terminal fault mitigation is present: True
Windows OS support for L1 terminal fault mitigation is enabled: True

Speculation control settings for MDS [microarchitectural data sampling]

Windows OS support for MDS mitigation is present: True
Hardware is vulnerable to MDS: True
Windows OS support for MDS mitigation is enabled: False

Suggested actions

 * Install BIOS/firmware update provided by your device OEM that enables hardware support for the branch target injection mitigation.
 * Follow the guidance for enabling Windows Server support for speculation control mitigations described in https://support.microsoft.com/help/4072698


BTIHardwarePresent                 : False
BTIWindowsSupportPresent           : True
BTIWindowsSupportEnabled           : False
BTIDisabledBySystemPolicy          : False
BTIDisabledByNoHardwareSupport     : True
BTIKernelRetpolineEnabled          : False
BTIKernelImportOptimizationEnabled : False
KVAShadowRequired                  : True
KVAShadowWindowsSupportPresent     : True
KVAShadowWindowsSupportEnabled     : True
KVAShadowPcidEnabled               : True
SSBDWindowsSupportPresent          : True
SSBDHardwareVulnerable             : True
SSBDHardwarePresent                : False
SSBDWindowsSupportEnabledSystemWide : False
L1TFHardwareVulnerable             : True
L1TFWindowsSupportPresent          : True
L1TFWindowsSupportEnabled          : True
L1TFInvalidPteBit                  : 45
L1DFlushSupported                  : False
MDSWindowsSupportPresent           : True
MDSHardwareVulnerable              : True
MDSWindowsSupportEnabled           : False


PS C:\Users\Administrator>
```

Figure 1: This figure shows the CPU mitigation settings on the M4 series instances powered by Intel E5_v4 processors. Source: Principled Technologies.

```
PS C:\Users\Administrator> Get-SpeculationControlSettings
For more information about the output below, please refer to https://support.microsoft.com/help/4074629

Speculation control settings for CVE-2017-5715 [branch target injection]

Hardware support for branch target injection mitigation is present: False
Windows OS support for branch target injection mitigation is present: True
Windows OS support for branch target injection mitigation is enabled: False
Windows OS support for branch target injection mitigation is disabled by system policy: False
Windows OS support for branch target injection mitigation is disabled by absence of hardware support: True

Speculation control settings for CVE-2017-5754 [rogue data cache load]

Hardware requires kernel VA shadowing: True
Windows OS support for kernel VA shadow is present: True
Windows OS support for kernel VA shadow is enabled: True
Windows OS support for PCID performance optimization is enabled: True [not required for security]

Speculation control settings for CVE-2018-3639 [speculative store bypass]

Hardware is vulnerable to speculative store bypass: True
Hardware support for speculative store bypass disable is present: False
Windows OS support for speculative store bypass disable is present: True
Windows OS support for speculative store bypass disable is enabled system-wide: False

Speculation control settings for CVE-2018-3620 [L1 terminal fault]

Hardware is vulnerable to L1 terminal fault: True
Windows OS support for L1 terminal fault mitigation is present: True
Windows OS support for L1 terminal fault mitigation is enabled: True

Speculation control settings for MDS [microarchitectural data sampling]

Windows OS support for MDS mitigation is present: True
Hardware is vulnerable to MDS: True
Windows OS support for MDS mitigation is enabled: False

Suggested actions

 * Install BIOS/firmware update provided by your device OEM that enables hardware support for the branch target injection mitigation.
 * Follow the guidance for enabling Windows Server support for speculation control mitigations described in https://support.microsoft.com/help/4072698


BTIHardwarePresent                 : False
BTIWindowsSupportPresent           : True
BTIWindowsSupportEnabled           : False
BTIDisabledBySystemPolicy          : False
BTIDisabledByNoHardwareSupport     : True
BTIKernelRetpolineEnabled          : False
BTIKernelImportOptimizationEnabled : False
KVAShadowRequired                  : True
KVAShadowWindowsSupportPresent     : True
KVAShadowWindowsSupportEnabled     : True
KVAShadowPcidEnabled               : True
SSBDWindowsSupportPresent          : True
SSBDHardwareVulnerable             : True
SSBDHardwarePresent                : False
SSBDWindowsSupportEnabledSystemWide : False
L1TFHardwareVulnerable             : True
L1TFWindowsSupportPresent          : True
L1TFWindowsSupportEnabled          : True
L1TFInvalidPteBit                  : 45
L1DFlushSupported                  : False
MDSWindowsSupportPresent           : True
MDSHardwareVulnerable              : True
MDSWindowsSupportEnabled           : False


PS C:\Users\Administrator> _
```

Figure 2: This figure shows the CPU mitigation settings on the M5 series instances powered by Intel 2nd Generation Xeon processors. Source: Principled Technologies.

Read the report at http://facts.pt/30SQBuS ▶

This project was commissioned by Intel.

Principled Technologies®

Facts matter.®