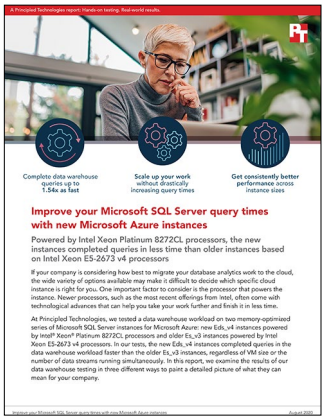

The science behind the report:

Improve your Microsoft SQL Server query times with new Microsoft Azure instances

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 Improve your Microsoft SQL Server query times with new Microsoft Azure instances.

We concluded our hands-on testing on June 30, 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 July 22, 2020 or earlier. Unavoidably, these configurations may not represent the latest versions available when this report appears.

Our results

Performance comparison results

We tested a warehouse workload from the HammerDB benchmark suite on two series of Microsoft SQL Server instances for Microsoft Azure: new Eds_v4 instances powered by Intel® Xeon® Platinum 8272CL processors, and older Es_v3 instances powered by Intel Xeon E5-2673 v4 processors. The HammerDB developers derived this particular workload from the TPC-H specifications.

Table 1: Time (in seconds) to complete data warehouse query set streams from the HammerDB benchmark suite. Less time is better. Source: Principled Technologies.

	Es_v3	Eds_v4	Eds_v4 % faster
Small VMs (4 vCPUs)			
1 stream	74	56	32.14%
2 streams	140	103	35.92%
3 streams	206	151	36.40%
4 streams	280	201	39.30%
Medium VMs (16 vCPUs)			
1 stream	80	54	48.15%
2 streams	139	92	51.09%

	Es_v3	Eds_v4	Eds_v4 % faster
3 streams	203	131	54.96%
4 streams	270	181	49.17%
5 streams	337	224	50.45%
Large VMs (64 vCPUs)			
1 stream	78	57	36.84%
2 streams	136	94	44.68%
3 streams	189	127	48.82%
4 streams	238	160	48.75%
5 streams	298	194	53.61%
6 streams	354	231	53.25%

E4ds_v4 scaling results

Table 2: Time (in seconds) to complete data warehouse query set streams from the HammerDB benchmark suite. Source: Principled Technologies.

	Time to complete (s)	Database scale (GB)
1 stream		
E4ds_v4	56	30
E16ds_v4	54	100
E64ds_v4	57	300
2 streams		
E4ds_v4	103	30
E16ds_v4	92	100
E64ds_v4	127	300
3 streams		
E4ds_v4	151	30
E16ds_v4	131	100
E64ds_v4	160	300
4 streams		
E4ds_v4	201	30
E16ds_v4	181	100

	Time to complete (s)	Database scale (GB)
E64ds_v4	160	300

Results analysis: Eds_v4 advantage over Es_v3

Table 3: Percentage by which Eds_v4 instances outperformed the Es_v3 instances in TPC-H-like data warehouse query tests. Source: Principled Technologies.

	Percentage advantage	Database scale (GB)
1 stream		
Small VMs	32.14%	30
Medium VMs	48.15%	100
Large VMs	36.84%	300
2 stream		
Small VMs	35.92%	30
Medium VMs	51.09%	100
Large VMs	44.68%	300
3 stream		
Small VMs	36.42%	30
Medium VMs	54.96%	100
Large VMs	48.82%	300
4 stream		
Small VMs	39.30%	30
Medium VMs	49.17%	100
Large VMs	48.75%	300

System configuration information

Table 4: Detailed configuration information for the Es_v3 instances we tested. Source: Principled Technologies.

Server configuration information	E4s_v3	E16s_v3	E64s_v3
Tested by	Principled Technologies	Principled Technologies	Principled Technologies
Test date	7/8/2020	7/22/2020	7/22/2020
Workload & version	HammerDB v3.3 TPC-H-Like	HammerDB v3.3 TPC-H-Like	HammerDB v3.3 TPC-H-Like
Server platform	E4s_v3	E16s_v3	E64s_v3
BIOS name and version	Microsoft Corporation Hyper-V UEFI Release v2.0, 8/26/2016	Microsoft Corporation Hyper-V UEFI Release v2.0, 8/26/2016	Microsoft Corporation Hyper-V UEFI Release v2.0, 8/26/2016
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
Date of last OS updates/patches applied	06/29/2020	06/29/2020	06/29/2020
Processor			
Number of processors	1	1	2
Vendor and model	Intel Xeon CPU E5-2673 v4	Intel Xeon CPU E5-2673 v4	Intel Xeon CPU E5-2673 v4
Core count (per processor)	20	20	20
Core frequency (GHz)	2.30	2.30	2.30
Stepping	1	1	1
Hyper-Threading	Yes	Yes	Yes
Turbo	Yes	Yes	Yes
Memory module(s)			
Total memory in system (GB)	32	128	432
Number of memory modules	1	1	1
Vendor and model	Not listed	Not listed	Not listed
Size (GB)	Not listed	Not listed	16
Type	Not listed	Not listed	Not listed
Speed (MHz)	Not listed	Not listed	Not listed
Speed running in the server (MHz)	Not listed	Not listed	Not listed
NVMe memory present?	No	No	No
Total memory (DDR+NVMe RAM)	32	128	432
Local storage (OS)			
Number of drives	1	1	1
Drive vendor and model	Not listed	Not listed	Not listed
Drive size (GB)	127	127	127
Drive information (speed, interface, type)	Standard HDD	Standard HDD	Premium SSD

Server configuration information	E4s_v3	E16s_v3	E64s_v3
Data drive			
Number of drives	1	1	1
Drive vendor and model	Not Listed	Not Listed	Not Listed
Drive size (GB)	128	256	512
Drive information (speed, interface, type)	Premium SSD	Premium SSD	Ultra Disk
Temporary drive			
Number of drives	1	1	1
Drive vendor and model	Not Listed	Not Listed	Not Listed
Drive size (GB)	64	256	864?
Drive information (speed, interface, type)	Not Listed	Not Listed	Not Listed
Network adapter			
Vendor and model	Microsoft Hyper-V Network Adapter	Microsoft Hyper-V Network Adapter	Microsoft Hyper-V Network Adapter
Number and type of ports	1x 40Gb	1x 50Gb	1x 50Gb

Table 5: Detailed configuration information for the Eds_v4 instances we tested. Source: Principled Technologies.

Server configuration information	E4ds_v4	E16ds_v4	E64ds_v4
Tested by	Principled Technologies	Principled Technologies	Principled Technologies
Test date	7/8/2020	7/22/2020	7/22/2020
CSP / Region	Microsoft Azure East US (Zone 2)	Microsoft Azure East US (Zone 2)	Microsoft Azure East US (Zone 2)
Workload & version	HammerDB v3.3 TPC-H-Like	HammerDB v3.3 TPC-H-Like	HammerDB v3.3 TPC-H-Like
WL specific parameters	CCI, MAXDOP 4, 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
Server platform	E4ds_v4	E16ds_v4	E64ds_v4
BIOS name and version	Microsoft Corporation Hyper-V UEFI Release v4.0, 3/12/2019	Microsoft Corporation Hyper-V UEFI Release v4.0, 3/12/2019	Microsoft Corporation Hyper-V UEFI Release v4.0, 3/12/2019
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
Date of last OS updates/ patches applied	06/29/2020	06/29/2020	06/29/2020
Processor			
Number of processors	1	1	2
Vendor and model	Intel Xeon Platinum 8272CL	Intel Xeon Platinum 8272CL	Intel Xeon Platinum 8272CL
Core count (per processor)	26	26	26

Server configuration information	E4ds_v4	E16ds_v4	E64ds_v4
Core frequency (GHz)	2.60	2.60	2.60
Stepping	7	7	7
Hyper-Threading	Yes	Yes	Yes
Turbo	Yes	Yes	Yes
Number of vCPU per VM	4	16	64
Memory module(s)			
Total memory in system (GB)	32	128	504
NVMe memory present?	No	No	No
Total memory (DDR+NVMe RAM)	32	128	504
Local storage (OS)			
Number of drives	1	1	1
Drive size (GB)	127	127	127
Drive information (speed, interface, type)	Standard HDD	Standard HDD	Premium SSD
Data drive			
Storage: NW or Direct Att / Instance	Direct Att	Direct Att	Direct Att
Network BW / Instance	N/A	N/A	N/A
Storage BW / Instance	N/A	N/A	N/A
Data drive			
Number of drives	1	1	1
Drive size (GB)	128	256	512
Drive information (speed, interface, type)	Premium SSD	Premium SSD	Ultra Disk
Temporary drive			
Number of drives	1	1	1
Drive size (GB)	149	599	2,340
Drive information (speed, interface, type)	Not listed	Not listed	Not listed
Network adapter			
Vendor and model	Microsoft Hyper-V Network Adapter	Microsoft Hyper-V Network Adapter	Microsoft Hyper-V Network Adapter
Number and type of ports	1x 50Gb	1x 50Gb	1x 50Gb

How we tested

Table 6: Reference CPU, memory, and database information for the instances we used during testing. Source: Principled Technologies.

Instance Type	CPU Model	# vCPUs	Memory (GB)	Database drive	Database size
E4ds_v4	Platinum 8272CL	4	32	Premium SSD P10	30-scale
E4s_v3	Xeon E5-2673 v4	4	32	Premium SSD P10	30-scale
E16ds_v4	Platinum 8272CL	16	128	Premium SSD P15	100-scale
E16s_v3	Xeon E5-2673 v4	16	128	Premium SSD P15	100-scale
E64ds_v4	Platinum 8272CL	64	504	Ultra Disk (512GB)	300-scale
E64s_v3	Xeon E5-2673 v4	64	432	Ultra Disk (512GB)	300-scale

Table 7: Reference SQL information for the instances we used during testing. Source: Principled Technologies.

Test pair	Memory reserved for SQL (MB)	# of tempdb files	Location of tempdb files	SQL MAXDOP	Max # of users in simultaneous stream test
E4	29,491	4	Data drive	4	4
E16	117,964	8	Data drive	16	5
E64	398,131 (Eds_v4) 464,486 (Es_v3)	8	Local drive	64	6

Testing overview

We created a baseline SQL VM in our Microsoft Azure account with Windows Server 2019. On this base VM, we ran Windows updates, installed and configured SQL Server 2019 Enterprise Edition, and installed HammerDB 3.3. We created a snapshot of this base VM and used it to create a specialized image. We then created all of our test instances using this image to ensure that our base settings were consistent across all testing. For testing, we used the data warehouse workload from the HammerDB suite, which the HammerDB developers derived from TPC-H specifications. Because HammerDB does not meet the full requirements for the TPC-H benchmark specification, our results are not directly comparable to published TPC-H results. We sized the data warehouse database on each VM pair to fit into the RAM. The more robust VMs had larger databases to make sure we could stress the systems adequately. Due to the different hardware amounts and database sizes, some tuning settings differ among the three pairs. However, within each pair, we kept everything as identical as possible outside of the instance types themselves. See below for the steps we followed as well as tables to show the VM configurations and settings. Also note that the Premium SSDs that we use for the 4 and 16 vCPU pairs have a bursting option that gives them a higher IOPS and throughput cap for 30 minutes per day. Since our databases fit in RAM and we run a single-stream test to cache the database before each run, we were able to keep our results consistent despite the fluctuations in disk performance introduced by this bursting.

Creating the Windows Server 2019 baseline image

Creating the baseline image VM

1. Log into the Azure Portal and navigate to the Virtual Machines service.
2. Click Add to open the Add VM wizard.
3. On the Basics tab, set the following:
    a. Choose your Subscription from the dropdown menu.
    b. Choose your Resource group from the dropdown menu.
    c. Name the Virtual Machine.
    d. Choose your Region from the dropdown menu.
    e. Leave the Availability options set to No infrastructure redundancy required.
    f. Choose Windows Server 2019 Datacenter from the Image dropdown menu.
    g. Leave Azure Spot instance set to No.
    h. Select the instance size you wish to use, we used Standard B4ms.
    i. Leave the Authentication type set to SSH public key.

j. Either choose a new Username or leave the default.
k. Choose Generate new key pair for the SSH public key source.
l. Enter a name for the Key pair name.
m. Leave Public inbound ports set to Allow selected ports.
n. For Select inbound ports, choose SSH (22).

4. On the Disks tab, set the following:
   a. For the OS disk type, choose Standard HDD from the dropdown menu.
   b. Leave the default Encryption type.
5. On the Networking tab, set the following:
   a. Choose your Virtual network from the dropdown menu.
   b. Choose Create new to create a new Public IP.
   c. Leave the rest of the settings at defaults.
6. On the Management tab, set the following:
   a. Choose your Diagnostics storage account from the dropdown menu.
   b. Leave the rest set to defaults.
7. On the Advanced tab, leave all defaults.
8. On the Tags tab, add any tags you wish to use.
9. On the Review + create tab, review your settings, and click Create.

Configuring Windows Server 2019

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

Installing SQL Server Enterprise 2019

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 Server (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 by clicking the Start button, typing `services` to open the

Services list, and disabling the Windows Update service.

Locking 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 on Lock pages in memory.
4. Click Add User or Group, type `NT Service\MSSQLSERVER`, and click OK.
5. Click OK to close the Properties window, 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. Chose a destination location, and click Next.
6. Click Next.
7. Click Finish.

Creating a snapshot of the baseline VM

1. In your Azure portal, navigate to the Snapshots service.
2. Click Add to open the Snapshot wizard.
3. On the Basics tab, set the following:
    a. Choose your Subscription from the dropdown menu.
    b. Choose your Resource group from the dropdown menu.
    c. Enter a name for your snapshot.
    d. Choose your Region from the dropdown menu.
    e. Select Full - make a complete read-only copy of the selected disk for the Snapshot type.
    f. Choose the OS disk from your baseline VM.
    g. Choose Standard HDD for the Storage type.
4. On the Encryption tab, leave all defaults.
5. On the Tags tab, add any tags you wish to use.
6. On the Review + create tab, review your settings, and click Create.

Creating an image with the baseline snapshot

To create an image, you must first have a Shared Image Gallery. The steps below will walk you through the creation of the gallery as well as the image creation steps. Once you have created your gallery, you will not need to do so again to add new images.

1. In your Azure portal, navigate to the Shared image galleries service.
2. Click Add to open the Add gallery wizard.
3. On the Basics tab, set the following:
    a. Choose your Subscription from the dropdown menu.
    b. Choose your Resource from the dropdown menu.
    c. Name your gallery.
    d. Choose your Region from the dropdown menu.
    e. Enter a Description if you like.
4. On the Tags tab, add any tags you wish to use.
5. On the Review + create tab, review your settings, and click Create.
6. Click on your new image gallery, and click Add new image definition to open the wizard.
7. On the Basics tab, set the following:
    a. Set the Operating System to Windows.
    b. Set the VM generation to Gen 2.
    c. Set the Operation system state to Specialized.
    d. Enter whatever you wish for the Publisher, Offer, and SKU entries.
8. Skip the Version tab.
9. Skip the Publishing options tab.

10. On the Tags tab, add any tags you wish to use.
11. On the Review + create tab, review your settings, and click Create.
12. Click on the image definition you've created, and click Add version to open the wizard.
13. On the Basics tab, set the following:
    a. Enter a version number such as 1.0.0.
    b. Choose the OS disk snapshot of the baseline VM you created from the dropdown menu.
    c. Leave the rest as defaults.
14. On the Encryption tab, leave defaults.
15. On the Tags tab, add any tags you wish to use.
16. On the Review + create tab, review your settings, and click Create.

Creating the VMs under test

In this section we list the steps required to create a VM from the image we created previously. See Table 6 for the list of VMs and SSDs we used and follow the steps six times using the proper specifications for the VM you wish to create. The Ultra Disks we use on the largest 64 vCPU VMs are only available in specific regions and availability zones, so be sure that your choices for those areas are compatible. For our testing, we used the East US Region and Availability Zone 2.

Creating the VMs from the specialized image

1. Open the Azure Portal and navigate to the Share image galleries service.
2. Click on the Shared image gallery you created.
3. Navigate to the image version you created (should be something like 1.0.0), and click Create VM.
4. On the Basics tab, set the following:
    a. Choose your Subscription from the dropdown menu.
    b. Choose your Resource group from the dropdown menu.
    c. Enter a Virtual machine name.
    d. Choose Availability Zone and set the Zone you desire.
    e. Select the instance size you want.
    f. Leave the rest as defaults.
5. On the Disks tab, set the following:
    a. Change the OS disk type to Standard HDD for the 4 vCPU VMs or Premium SSD for the other two.
    b. Click Create and attach a new disk.
        i. In the Create a new disk wizard, click Change size and pick the size of Premium SSD that matches your instance type.
        ii. Leave the rest as defaults, and click OK.
6. Skip the Networking, Management, and Advanced tabs.
7. On the Tags tab, assign any tags you wish to use.
8. On the Review + create tab, review your settings, and click Create.
9. Once the VM creation is finished, click Go to resource (or navigate to the virtual machine service and click on the new VM).
10. Click Connect→RDP and download the RDP file.
11. Double-click the RDP file and log in with the user and password you set.
12. Right click the Windows Start button and click Disk Management.
13. Click OK on the Popup window about GPT partition.
14. Right-click the Premium SSD you added, and follow the prompts to create a new NTFS volume for the database.

Configuring SQL Server on the VMs under test

In this section, we list the various SQL settings that we changed and the steps to do so. To see the exact settings for each VM, please refer to table 7.

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

1. Open the SQL Server Management Studio.
2. Right-click the SQL Instance, and click Properties.
3. Click Advanced node, and scroll down to the Max Degree of Parallelism and change the value. 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 when prompted.

Configuring the tempdb database

1. Open the SQL Server Management Studio
2. Expand Databases and System databases, and right click on tempdb.
3. Add files and change the starting size as necessary.
4. Right click the SQL Instance, and restart the service. Click Yes when prompted.
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 when prompted.

Running the tests

In this section, we list the steps to run the HammerDB TPC-H-like test on the VMs under test. As each VM had different hardware and database sizes, please refer to Table 7 to see the number of users to run on each VM. For the maximum number of users we ran, we followed HammerDB TPC-H recommendations for the size database we were testing. Additionally, to show the scaling of each VM 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 VM you're testing, restore the database under test to so that the database and log files reside on the Premium SSD.
2. Make sure your SQL settings and tempdb are configured properly according to the instructions above and the instance you're running on.
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, then 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. Double-click Load to load the Driver Script.
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. Click Start to begin the run.
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, start Performance monitor set to record CPU, Memory, and drive usage information.
22. Click Start on HammerDB to begin the run.

23. When the run finishes, stop Perfmon and save the HammerDB results file and Perfmon output.
24. Reboot the VM.
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

We ran the following command in PowerShell on each VM to determine the Intel processor mitigation settings that Azure employs:

```
PS C:\Users\AzureUser> 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: True
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: True

Suggested actions

 * Install BIOS/firmware update provided by your device OEM that enables hardware support for the branch target injection mitigation.


BTIHardwarePresent                 : False
BTIWindowsSupportPresent           : True
BTIWindowsSupportEnabled           : False
BTIDisabledBySystemPolicy          : True
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           : True


PS C:\Users\AzureUser>
```

Figure 1: Output for the Es_v3 migration source. Source: Principled Technologies.

Figures 1 and 2 display the output for the Es_v3 VMs and the Eds_v4 VM.

```
PS C:\Users\AzureUser> 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: True
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: False

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: False

Speculation control settings for MDS [microarchitectural data sampling]

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

Suggested actions

 * Install BIOS/firmware update provided by your device OEM that enables hardware support for the branch target injection mitigation.


BTIHardwarePresent                 : False
BTIWindowsSupportPresent           : True
BTIWindowsSupportEnabled           : False
BTIDisabledBySystemPolicy          : True
BTIDisabledByNoHardwareSupport     : True
BTIKernelRetpolineEnabled          : False
BTIKernelImportOptimizationEnabled : False
KVAShadowRequired                  : False
KVAShadowWindowsSupportPresent     : True
KVAShadowWindowsSupportEnabled     : False
KVAShadowPcidEnabled               : False
SSBDWindowsSupportPresent          : True
SSBDHardwareVulnerable             : True
SSBDHardwarePresent                : False
SSBDWindowsSupportEnabledSystemWide : False
L1TFHardwareVulnerable             : False
L1TFWindowsSupportPresent          : True
L1TFWindowsSupportEnabled          : False
L1TFInvalidPteBit                  : 0
L1DFlushSupported                  : False
MDSWindowsSupportPresent           : True
MDSHardwareVulnerable              : False
MDSWindowsSupportEnabled           : False



PS C:\Users\AzureUser>
```

Figure 2: Output of the Eds_v4 migration. Source: Principled Technologies.

Read the report at http://facts.pt/ikpqek2 ▶

This project was commissioned by Intel.

Principled Technologies®

Facts matter.®