



The science behind the report:

Maximize your upgrade by minimizing hypervisor costs with a new AMD EPYC 7551P processor-powered single-socket Dell EMC PowerEdge R7415

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 [Maximize your upgrade by minimizing hypervisor costs with a new AMD EPYC 7551P processor-powered single-socket Dell EMC PowerEdge R7415](#).

We concluded our hands-on testing on July 9th, 2019. 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 2nd, 2019 or earlier. Unavoidably, these configurations may not represent the latest versions available when this report appears.

Our results

The table below presents our findings in detail.

| | AMD EPYC™ 7551P processor-powered single-socket Dell EMC™ PowerEdge™ R7415 | | Intel® Xeon® Gold 6130 processor-powered dual-socket Supermicro SuperServer® 2029U | | Intel Xeon E5-2680 v3 processor-powered dual-socket Supermicro SuperServer 2028U | |
|---|--|----------------|--|----------------|--|--------------|
| | TPM | NOPM | TPM | NOPM | TPM | NOPM |
| VM01 | 32,929 | 10,887 | 32,378 | 10,695 | 4,421 | 1,463 |
| VM02 | 32,528 | 10,730 | 37,848 | 12,477 | 4,428 | 1,448 |
| VM03 | 31,521 | 10,410 | 29,294 | 9,688 | 4,437 | 1,465 |
| VM04 | 33,269 | 11,000 | 30,647 | 10,124 | 4,391 | 1,444 |
| VM05 | 31,387 | 10,370 | 34,204 | 11,297 | | |
| VM06 | 29,120 | 9,590 | 30,685 | 10,122 | | |
| VM07 | 30,510 | 10,079 | 33,443 | 11,006 | | |
| VM08 | 33,001 | 10,934 | 29,351 | 9,701 | | |
| VM09 | 32,621 | 10,757 | 31,113 | 10,268 | | |
| VM10 | 31,034 | 10,232 | 29,392 | 9,681 | | |
| VM11 | 30,252 | 9,968 | 29,570 | 9,744 | | |
| VM12 | 29,765 | 9,865 | 29,665 | 9,792 | | |
| VM13 | 30,464 | 10,044 | 34,174 | 11,278 | | |
| VM14 | 35,862 | 11,824 | 29,422 | 9,720 | | |
| VM15 | 30,133 | 9,926 | 32,233 | 10,596 | | |
| Sum | 474,396 | 156,616 | 473,419 | 156,189 | 17,677 | 5,820 |
| Average | 31,626 | 10,441 | 31,561 | 10,413 | 4,419 | 1,455 |
| Power average under test (W) | 299.07 | | 391.64 | | 349.73 | |
| Idle power average (W) | 171.54 | | 248.88 | | 262.35 | |
| AMD EPYC 7551P solution reduced power under load by* | 23.64% | | | | | |
| AMD EPYC 7551P solution reduced idle power by* | 31.08% | | | | | |

*Compared to Intel Xeon Gold 6130 solution

System configuration information

The table below presents detailed information on the systems we tested.

| Server configuration information | Dell EMC PowerEdge R7415 | Supermicro SYS-2029U-E1CR4 | Supermicro SYS-2028U-E1CNR4T+ |
|--|---|--|---|
| BIOS name and version | 1.8.7 | 3.1 | 3.80 |
| Non-default BIOS settings | System profile: Performance | System profile: Performance | System profile: Performance |
| Operating system name and version/build number | VMware® ESXi, 6.7.0, 13006603 | VMware ESXi 6.7.0 13006603 | VMware ESXi 6.7.0 133006603 |
| Power management policy | Performance | Performance | Performance |
| Processor | | | |
| Number of processors | 1 | 2 | 2 |
| Vendor and model | AMD EPYC 7551P | Intel Xeon Gold 6130 | Intel Xeon E5-2680 v3 |
| Core count (per processor) | 32 | 16 | 12 |
| Core frequency (GHz) | 2.00 | 2.10 | 2.50 |
| Stepping | 2 | H0 | M0 |
| Memory module(s) | | | |
| Total memory in system (GB) | 128 | 128 | 128 |
| Number of memory modules | 16 | 16 | 16 |
| Vendor and model | Hynix® HMA81GR7AFR8N-VK | Hynix HMA81GR7CJR8N-VK | Micron Technologies 9ASF1G72PZ-2G6D1 |
| Size (GB) | 8 | 8 | 8 |
| Type | 1Rx8 | 1Rx8 | 1Rx8 |
| Speed (MHz) | 2666 | 2666 | 2666 |
| Speed running in the server (MHz) | 2133 | 2133 | 2133 |
| Storage controller | | | |
| Vendor and model | Dell PERC H740P Adapter | Avago 3108 MegaRAID | Avago 3108 MegaRAID |
| Cache size (MB) | 8192 | 2048 | 2048 |
| Firmware version | 50.5.0-1750 | 4.680.00-8290 | 4.680.00.8290 |
| Local storage | | | |
| Number of drives | 24 | 24 | 24 |
| Drive vendor and model | Intel SSDSC2KB480G8 | Intel SSDSC2KB480G8 | Seagate ST600MM0009 |
| Drive size (GB) | 480 | 480 | 600 |
| Drive information (speed, interface, type) | 6 Gb/s SATA SSD | 6 Gb/s SATA SSD | 12 Gb/s SAS 10k HDD |
| Network adapter | | | |
| Vendor and model | Broadcom® Adv. Dual 10GBASE-T Ethernet | Intel I350 Gigabit Network Connection | Intel 10 Gigabit X540-AT2 |
| Number and type of ports | 2 x 10G | 4 x 10G | 4 x 10G |

How we tested

We installed VMware ESXi 6.7.0 build 13006603 on each server under test. We then created three RAID 10 volumes. Our OS volume used four drives and housed our backup of our database. Our log volume used four drives. Our database volume used the last eight drives and housed our target MySQL database. On all our tests we ran a 5-minute warmup, followed by a 30-minute test run. Between tests we restored the database, restarted the MySQL VM, and waited 20 minutes before running an additional test. Our results are the median result of three runs based on the total TPM across all VMs.

We targeted the database using HammerDB clients equal to the number of VMs we used in each test. For the AMD EPYC 7551P and Intel Xeon Gold 6130 solutions, we determined the maximum VM count by dividing the 64 total CPU threads in each solution evenly across the VMs while retaining 4 threads for overhead. At four vCPUs per VM, we were able to fit 15 VMs on each solution. For the legacy Intel Xeon E5-2680 v3 solution, we added VMs until the HDDs breached 10ms latency.

- AMD EPYC 7551P processor-powered single-socket Dell EMC PowerEdge R7415 server – 15 VMs
- Intel Xeon Gold 6130 processor-powered dual-socket Supermicro SuperServer 2029U server – 15 VMs
- Intel Xeon E5-2680 v3 processor-powered dual-socket Supermicro SuperServer 2028U server – 4 VMs

Each VM had four vCPUs, 8 GB of memory, and four hard disks—one each for the OS, the database, the logs, and to store the backup file. We placed each hard disk on the corresponding host volume.

We recorded power by attaching each server under test to an Extech power meter, which recorded the power consumption of each server while idle and while under test. We report the average power over the course of the test for the under-load consumption, and during a 10-minute idle period.

Setting up

1. Install ESXi 6.7.0 build 13006603 on each server under test, and your client server.
2. Install VMware vCenter on your client server, and add your servers under test using the vCenter console.
3. In the VMware vCenter console, make the following changes on the Intel servers to mitigate processor vulnerabilities.
 - a. In the vCenter console, select the target server.
 - b. Select the Configure tab.
 - c. Navigate to System, then Advanced System Settings.
 - d. Change the following values:
 - ♦ VMkernel.Boot.hyperthreadingMitigation : true
 - ♦ VMkernel.Boot.hyperthreadingMitigationIntraVM : false
 - e. Click OK.
 - f. Reboot the ESXi host.
4. Create your VMs on your systems under test, and install a minimal CentOS 7 on your MySQL VMs.
5. Create client VMs on your client server, and install RHEL 7.6.

Configuring the VMs

1. Disable SELinux and Firewall:

```
sed -i 's/SELINUX=.*SELINUX=disabled/' /etc/selinux/config
systemctl disable firewalld
```
2. Install updates, and reboot:

```
yum update -y
reboot
```
3. Install the EPEL repository, and enable the optional and extras repositories:

```
yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
subscription-manager repos --enable "rhel-*-optional-rpms" --enable "rhel-*-extras-rpms"
yum update -y
```
4. Install extra packages:

```
yum install -y xfsprogs tuned numactl wget vim openssh-clients nfs-utils man unzip ipmitool OpenIPMI
sysstat nmon lz4 device-mapper-multipath hdparm pciutils yum-utils
```

Tuning OS for MySQL and installing MySQL

1. Add the following to /etc/sysctl.conf:

```
vm.swappiness = 0
kernel.sem = 250 32000 100 128
fs.file-max = 6815744
net.ipv4.ip_local_port_range = 9000 65500
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576
fs.aio-max-nr = 1048576
vm.nr_hugepages = 3072
vm.hugetlb_shm_group=1000
```

2. Configure MySQL permission access to huge pages:

```
sed -i '/vm.hugetlb_shm_group/d' /etc/sysctl.conf
echo vm.hugetlb_shm_group=$(id -g mysql) >> /etc/sysctl.conf
reboot
```

3. Install MySQL (Ver 8.0.13 for Linux on x86_64):

```
yum install -y https://dev.mysql.com/get/mysql80-community-release-el7-1.noarch.rpm
yum install -y mysql-community-server
```

Configuring MySQL

1. Create a backup of your config file:

```
systemctl stop mysqld
systemctl disable mysqld
cp -p /etc/my.cnf{,.orig}
```

2. Edit the /etc/my.cnf file to match the file below:

```
[mysqld]
default-authentication-plugin=mysql_native_password

datadir=/mnt/db_data/mysql_data
socket=/var/lib/mysql/mysql.sock

log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid

skip_log_bin=1

innodb_log_group_home_dir=/mnt/db_log/mysql_innodb_log
validate_password.policy=LOW
validate_password.length=3

large-pages
# default-authentication-plugin=mysql_native_password
port=3306

# general
max_connections=4000
table_open_cache=8000
table_open_cache_instances=16
back_log=1500
default_password_lifetime=0
ssl=0
performance_schema=OFF
max_prepared_stmt_count=128000
skip_log_bin=1
character_set_server=latin1
collation_server=latin1_swedish_ci
```

```

transaction_isolation=REPEATABLE-READ
# files
innodb_file_per_table
innodb_log_file_size=1024M
innodb_log_files_in_group=32
innodb_open_files=4000
# buffers
innodb_buffer_pool_size=6000M
innodb_buffer_pool_instances=16
innodb_log_buffer_size=64M
# tune
innodb_doublewrite=0
innodb_thread_concurrency=0
innodb_flush_log_at_trx_commit=2
innodb_max_dirty_pages_pct=90
innodb_max_dirty_pages_pct_lwm=10
join_buffer_size=32K
sort_buffer_size=32K
innodb_use_native_aio=1
innodb_stats_persistent=1
innodb_spin_wait_delay=6
innodb_max_purge_lag_delay=300000
innodb_max_purge_lag=0
innodb_flush_method=O_DIRECT_NO_FSYNC
innodb_checksum_algorithm=none
innodb_io_capacity=40000
innodb_io_capacity_max=200000
innodb_lru_scan_depth=9000
innodb_change_buffering=none
innodb_read_only=0
innodb_page_cleaners=4
innodb_undo_log_truncate=off
# perf special
innodb_adaptive_flushing=1
innodb_flush_neighbors=0
innodb_read_io_threads=16
innodb_write_io_threads=16
innodb_purge_threads=4
innodb_adaptive_hash_index=0
# monitoring
innodb_monitor_enable='%'

```

3. Clean up, create, and configure data and log directory:

```

rm -rf /mnt/db_data/mysql_data
mkdir /mnt/db_data/mysql_data
chown -R mysql:mysql /mnt/db_data/mysql_data
rm -rf /mnt/db_log/mysql_innodb_log
mkdir /mnt/db_log/mysql_innodb_log
chown -R mysql:mysql /mnt/db_log/mysql_innodb_log

```

4. Initialize MySQL (this will create the log files):

```
systemctl start mysqld
```

5. Set root password:

```
grep password /var/log/mysqld.log
```

- Example output: A temporary password is generated for root@localhost: <uAdom9lyo/s

6. Use the temporary password to log in to the MySQL console:

```
mysql -u root -p
```

7. Run the following commands in the MySQL console to change the root user password:

```

ALTER USER 'root'@'localhost' IDENTIFIED BY 'mysql';
use mysql;
update user set host='%' where user='root' and host='localhost';flush privileges;
grant all on *.* to root@'%';
quit

```

Building the database

1. Create file innoddbuild.tcl:

```
vi innoddbuild.tcl
```
2. Edit the innoddbuild.tcl file to include the following:

```
#!/bin/tclsh
puts "SETTING CONFIGURATION"
dbset db mysql
diset connection mysql_host test01-direct
diset connection mysql_port 3306
diset tpcc mysql_user root
diset tpcc mysql_pass mysql
diset tpcc mysql_count_ware 200
diset tpcc mysql_partition true
diset tpcc mysql_num_vu 100
diset tpcc mysql_storage_engine innodb
print dict
buildschema
```
3. Install HammerDB following the same steps as the section below for the clients.
4. Run the build command:

```
./hammerdbcli < innoddbuild.tcl
```
5. When the build is complete, type `exit` to end the program.

Creating a backup (100K)

1. Run the following commands to create a database backup:

```
systemctl stop mysqld
sync
cd /mnt
umount /mnt/net_backup
mount -v /mnt/net_backup
rm -rf /mnt/net_backup/mysql_tpcc_100k
mkdir -p /mnt/net_backup/mysql_tpcc_100k/
echo 1 > /proc/sys/vm/drop_caches
echo 0 > /proc/sys/vm/nr_hugepages
time cp -av db_log /mnt/net_backup/mysql_tpcc_100k/
time cp -av db_data /mnt/net_backup/mysql_tpcc_100k/
sync
```

Installing the HammerDB client on the client system

We installed HammerDB on each of our RHEL 7.6 HammerDB VMs.

1. Run the following commands to install the HammerDB client:

```
yum install -y tcl
cd ~
wget https://downloads.sourceforge.net/project/hammerdb/HammerDB/HammerDB-3.1/HammerDB-3.1-Linux.tar.gz
tar -xf HammerDB-3.1-Linux.tar.gz
cd HammerDB-3.1/
yumdownloader --enablerepo=mysql57-community mysql-community-libs-5.7.24-1.el7.x86_64
rm -rf ./usr
rpm2cpio mysql-community-libs-5.7.24-1.el7.x86_64.rpm | cpio -idmv ./usr/lib64/mysql/libmysqlclient.so.20*
mv -f ./usr/lib64/mysql/libmysqlclient.so.20*
rm -rf ./usr
```

Running the test

We created the following file in the HammerDB folder of each HammerDB VM and changed the MySQL target IP for each VM.

1. Create the runtest.tcl file:

```
vi runtest.tcl
```

2. Edit the runtest.tcl file to contain the following:

```
#!/bin/tclsh
puts "SETTING CONFIGURATION"
dbset db mysql
dbset bm TPC-C
diset connection mysql_host [target IP for mysql VM]
diset connection mysql_port 3306
diset tpcc mysql_driver timed
diset tpcc mysql_total_iterations 1000000000
diset tpcc mysql_rampup 5
diset tpcc mysql_duration 30
#diset tpcc mysql_duration 5
diset tpcc mysql_allwarehouse true
vuset logtotemp 1
loadscript
puts "TEST RUN STARTED"
foreach z { 6 } {
vuset vu $z
vucreate
vurun
sleep 2200
#sleep 700
vudestroy
after 5000
}
puts "TEST RUN COMPLETE"
exit
```

3. Run test using the following command:

```
./hammerdbcli < runtest.tcl
```

Restoring from the backup

1. Run the following commands to restore from backup before running a new test:

```
systemctl stop mysqld
sync
cd /mnt
rm -rf db_data/mysql_data
rm -rf db_log/mysql_innodb_log/*
umount /mnt/db_backup
mount -v /mnt/db_backup
echo 1 > /proc/sys/vm/drop_caches
echo 0 > /proc/sys/vm/nr_hugepages
time cp -av /mnt/db_backup/mysql_tpcc_100k/db_data/mysql_data db_data/
sync
systemctl start mysqld
systemctl stop mysqld
date
shutdown now
```


Read the report at <http://facts.pt/rvrcdk3> ▶



Facts matter.®

Principled Technologies is a registered trademark of Principled Technologies, Inc.
AMD and EPYC are trademarks of Advanced Micro Devices, 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.