Arthi Kundadka
W1606761

COEN 241 HW 1 System vs OS Virtualization

Detailed configurations of the experimental setup:

QEMU commands and VM configurations:

I installed QEMU using HomeBrew. The command was:

```
brew install qemu
```

After installing QEMU using Homebrew method, a QEMU image was created and the VM was installed using the following command:

```
$sudo qemu-img create ubuntu.img 10G
```

```
$sudo qemu-system-x86_64 -hda ubuntu.img -boot
d -m 1536
```

Enabling a docker container:

```
(base) arthikundadka@Arthis-MacBook-Air ~ % cd ubuntu
cd: no such file or directory: ubuntu
(base) arthikundadka@Arthis-MacBook-Air ~ % cd desktop
(base) arthikundadka@Arthis-MacBook-Air desktop % cd ubuntu
(base) arthikundadka@Arthis-MacBook-Air ubuntu % $sudo qemu-system-x86_64 -hda ubuntu.img -boot d -m 1536
WARNING: Image format was not specified for 'ubuntu.img' and probing guessed raw.
         Automatically detecting the format is dangerous for raw images, write operations on block 0 will be restricted.
         Specify the 'raw' format explicitly to remove the restrictions.
```

I installed the Docker Desktop on my Mac using the link provided on the Homework Document. Then I installed the docker image called csminpp/ubuntu-sysbench.

```
arthikundadka — root@9cdb2dd8b97c: / — com.docker.cli ‹ docker run -it csminpp/ubuntu-sysbench — 153×47
Last login: Fri Oct 15 08:05:13 on ttys000
(base) arthikundadka@Arthis-MacBook-Air ~ % docker pull csminpp/ubuntu-sysbench
Using default tag: latest
latest: Pulling from csminpp/ubuntu-sysbench
Image docker.io/csminpp/ubuntu-sysbench:latest uses outdated schema1 manifest format. Please upgrade to a schema2 image for better future compatibility.
More information at https://docs.docker.com/registry/spec/deprecated-schema-v1/
d89e1bee20d9: Pull complete
9e0bc8a71bde: Pull complete
27aa681c95e5: Pull complete
a3ed95caeb02: Pull complete
55734f896640: Pull complete
Digest: sha256:90fd06985472eec3aa99b665618c23f074deb326fcc87a5fb59d2be1f9d97435
Status: Downloaded newer image for csminpp/ubuntu-sysbench:latest
docker.io/csminpp/ubuntu-sysbench:latest
```

```
arthikundadka — -zsh — 115×30
Last login: Fri Oct 15 08:26:03 on ttys001
(base) arthikundadka@Arthis-MacBook-Air ~ % docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED              STATUS            PORTS      NAMES
c4d8e854bbad   ubuntu    "bash"    About a minute ago   Up About a minute            compassionate_wu
(base) arthikundadka@Arthis-MacBook-Air ~ % docker images
REPOSITORY                 TAG       IMAGE ID       CREATED        SIZE
ubuntu                     latest    597ce1600cf4   2 weeks ago    72.8MB
csminpp/ubuntu-sysbench    latest    2787c5e16909   5 years ago    336MB
(base) arthikundadka@Arthis-MacBook-Air ~ %
```

Docker and QEMU runtime environment:


```
[root@9cdb2dd8b97c:/# sysbench --num-threads=16 --test=fileio --file-total-size=2G --file-test-mode=rndrw prepare
sysbench 0.4.12:  multi-threaded system evaluation benchmark

128 files, 16384Kb each, 2048Mb total
Creating files for the test...
[root@9cdb2dd8b97c:/# sysbench --num-threads=16 --test=fileio --file-total-size=2G --file-test-mode=rndrw run
sysbench 0.4.12:  multi-threaded system evaluation benchmark

Running the test with following options:
Number of threads: 16

Extra file open flags: 0
128 files, 16Mb each
2Gb total file size
Block size 16Kb
Number of random requests for random IO: 10000
Read/Write ratio for combined random IO test: 1.50
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing random r/w test
Threads started!
Done.

Operations performed:  6027 Read, 4010 Write, 12801 Other = 22838 Total
Read 94.172Mb  Written 62.656Mb  Total transferred 156.83Mb  (61.384Mb/sec)
 3928.58 Requests/sec executed

Test execution summary:
    total time:                          2.5549s
    total number of events:              10037
    total time taken by event execution: 17.0535
    per-request statistics:
         min:                            0.02ms
         avg:                            1.70ms
         max:                            12.71ms
         approx.  95 percentile:         6.43ms

Threads fairness:
    events (avg/stddev):          627.3125/19.45
    execution time (avg/stddev):  1.0658/0.02

[root@9cdb2dd8b97c:/# sysbench --num-threads=16 --test=fileio --file-total-size=2G --file-test-mode=rndrw cleanup
sysbench 0.4.12:  multi-threaded system evaluation benchmark

Removing test files...
root@9cdb2dd8b97c:/#
```

```
                max:                          118.23
                95th percentile:               44.17
                sum:                       156830.07

threads fairness:
    events (avg/stddev):        657.4375/30.11
    execution time (avg/stddev):   9.8019/0.06

arthi_k@ccserver:~$ sysbench --num-threads=16 --test=fileio --file-total-size=2G --file-test-mode=rndrw cleanup
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
WARNING: --num-threads is deprecated, use --threads instead
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Removing test files...
arthi_k@ccserver:~$ sysbench --test=cpu --cpu-max-prime=20000 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time


Prime numbers limit: 20000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:    91.46

General statistics:
    total time:                          10.0184s
    total number of events:              917

Latency (ms):
         min:                             4.22
         avg:                            10.80
         max:                            31.39
         95th percentile:                16.41
         sum:                          9900.50

threads fairness:
    events (avg/stddev):         917.0000/0.00
    execution time (avg/stddev):    9.9005/0.00

arthi_k@ccserver:~$
```

Three different tests for each virtualization technology: CPU and FileIO tests

CPU Test1 for Docker:


```
🔴 🟡 🟢          📄 arthikundadka — root@9cdb2dd8b97c: / — com.docker.cli ‹ docker run -it csminpp/ubuntu-sysbench — 132×29
root@9cdb2dd8b97c:/# sysbench --test=cpu --cpu-max-prime=20000 run
sysbench 0.4.12:  multi-threaded system evaluation benchmark

Running the test with following options:
Number of threads: 1

Doing CPU performance benchmark
```

CPU Test1 for QEMU:


```
rthi_k@ccserver:~$ sysbench --test=cpu --cpu-max-prime=20000 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time


Prime numbers limit: 20000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:     91.46

General statistics:
    total time:                          10.0184s
    total number of events:              917

Latency (ms):
         min:                                4.22
         avg:                               10.80
         max:                               31.39
         95th percentile:                   16.41
         sum:                             9900.50

Threads fairness:
    events (avg/stddev):           917.0000/0.00
    execution time (avg/stddev):   9.9005/0.00

rthi_k@ccserver:~$
```

CPU Test2 for Docker:
Command used - sysbench --test=cpu --cpu-max-prime=40000 run

CPU Test2 for QEMU:
Command used - sysbench --test=cpu --cpu-max-prime=40000 run

CPU Test3 for Docker:
Command used - sysbench --test=cpu --cpu-max-prime=80000 run

CPU Test3 for QEMU:
Command used - sysbench --test=cpu --cpu-max-prime=80000 run

FileIO Test1 for Docker:

FileIO Test1 for QEMU:

FileIO Test2 for Docker:
sysbench --num-threads=16 --test=fileio --file-total-size=2G --file-test-mode=rndwr prepare
sysbench --num-threads=16 --test=fileio --file-total-size=2G --file-test-mode=rndwr run
sysbench --num-threads=16 --test=fileio --file-total-size=2G --file-test-mode=rndwr cleanup

FileIO Test2 for QEMU:
sysbench --num-threads=16 --test=fileio --file-total-size=1G --file-test-mode=rndwr prepare
sysbench --num-threads=16 --test=fileio --file-total-size=1G --file-test-mode=rndwr run
sysbench --num-threads=16 --test=fileio --file-total-size=1G --file-test-mode=rndwr cleanup

FileIO Test3 for Docker:
sysbench --num-threads=16 --test=fileio --file-total-size=1G —file-test-mode=seqwr prepare
sysbench --num-threads=16 --test=fileio --file-total-size=1G —file-test-mode=seqwr run
sysbench --num-threads=16 --test=fileio --file-total-size=1G —file-test-mode=seqwr cleanup

FileIO Test3 for QEMU:
sysbench --num-threads=16 --test=fileio --file-total-size=1G —file-test-mode=seqwr prepare
sysbench --num-threads=16 --test=fileio --file-total-size=1G —file-test-mode=seqwr run
sysbench --num-threads=16 --test=fileio --file-total-size=1G —file-test-mode=seqwr cleanup

Shell scripts for running the experiment:
The automated shell scripts have been uploaded to the Git under the following file names:
 • Docker_CPU_Test.sh
 • Docker_FileIO_Test.sh
 • QEMU_CPU_test.sh
 • QEMU_FileIO_test.sh

User level vs kernel level comparison:
I have used the command "top" to make this comparison. Before running sys bench tests:

While running sys bench tests:

```
● ● ●  arthikundadka — root@b11d8fa0e7fa: / — com.docker.cli ‹ docker run -it csminpp/ubuntu-sysbench — 115×30

top — 02:25:08 up 15:23,  0 users,  load average: 1.37, 0.32, 0.10
Tasks:   2 total,   1 running,   1 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.7 us, 20.3 sy,  0.0 ni, 52.7 id, 26.1 wa,  0.0 hi,  0.2 si,  0.0 st
KiB Mem:   2033396 total,  1961292 used,    72104 free,    49520 buffers
KiB Swap:  1048572 total,    24720 used,  1023852 free.  1561352 cached Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
    1 root      20   0   18184   3256   2764 S   0.0  0.2   0:00.06 bash
   22 root      20   0   19868   2344   2016 R   0.0  0.1   0:00.00 top
```

Presentation and analysis of the performance data:

CPU Test1 for Docker: sysbench --test=cpu --cpu-max-prime=20000 run

	Total Time	Total number of events	Total time taken by event execution	Total Number of events per second
Test 1	10.3349s	10000	10.2538	967.6
Test 2	12.3368s	10000	12.2357	810.58
Test 3	10.6740s	10000	10.5872	936.86
Test 4	10.3842s	10000	10.3014	963
Test 5	10.3978s	10000	10.3161	961.74

CPU Test1 for QEMU: sysbench --test=cpu --cpu-max-prime=20000 run

	Total Time	Total number of events	Total Number of events per second
Test 1	10.0120s	942	94.09
Test 2	10.0184s	917	91.53
Test 3	10.0141s	803	80.18
Test 4	10.0083s	948	94.72

	Total Time	Total number of events	Total Number of events per second
Test 5	10.0070s	848	84.74

CPU Test2 for Docker: sysbench --test=cpu --cpu-max-prime=40000 run

	Total Time	Total number of events	Total time taken by event execution	Total Number of events per second
Test 1	39.8249s	10000	39.6952	251
Test 2	45.5699s	10000	45.4223	219.44
Test 3	40.1914s	10000	40.0624	248.8
Test 4	39.4065s	10000	39.2812	253.77
Test 5	37.7730s	10000	37.6495	264.74

CPU Test2 for QEMU: sysbench --test=cpu --cpu-max-prime=40000 run

	Total Time	Total number of events	Total Number of events per second
Test 1	10.0170s	324	32.35
Test 2	10.0155s	312	31.15
Test 3	10.0087s	330	32.97
Test 4	10.0450s	312	31.06
Test 5	10.0141s	326	32.55

CPU Test3 for Docker: sysbench --test=cpu --cpu-max-prime=80000 run

	Total Time	Total number of events	Total time taken by event execution	Total Number of events per second
Test 1	100.7416s	10000	100.6119	99.26
Test 2	102.7413s	10000	102.6437	97.33
Test 3	97.1342s	10000	97.0073	102.95

	Total Time	Total number of events	Total time taken by event execution	Total Number of events per second
Test 4	93.2129s	10000	93.0921	107.28
Test 5	99.5051s	10000	99.3780	100.50

CPU Test3 for QEMU: sysbench --test=cpu --cpu-max-prime=80000 run

	Total Time	Total number of events	Total Number of events per second
Test 1	10.0148s	114	11.38
Test 2	10.0726s	126	12.51
Test 3	10.0108s	126	12.59
Test 4	10.0240s	126	12.57
Test 5	10.0044s	131	13.09

FileIO Test1 for Docker: sysbench --num-threads=16 --test=fileio --file-total-size=2G --file-test-mode=rndrw run

	Total Time	Total number of events	Total time taken by event execution	Total Number of events per second
Test 1	2.5549s	10037	17.0535	3928.53
Test 2	1.3347s	10065	2.8989	7541.02
Test 3	1.5571s	10060	6.3632	6460.73
Test 4	1.9501s	10032	11.0963	5144.35
Test 5	2.0206s	10029	12.0168	4963.38

FileIO Test1 for QEMU: sysbench --num-threads=16 --test=fileio --file-total-size=2G --file-test-mode=rndrw run

	Total Time	Total number of events	Total Number of events per second
Test 1	11.5276s	10519	912.51
Test 2	11.3368s	10844	956.53
Test 3	11.3486s	10848	955.89
Test 4	11.3603s	9683	852.35
Test 5	11.2861s	10231	906.51

FileIO Test2 for Docker: sysbench --num-threads=16 --test=fileio --file-total-size=2G --file-test-mode=rndwr run

	Total Time	Total number of events	Total time taken by event execution	Total Number of events per second
Test 1	1.4380s	10200	0.7180	7093.18
Test 2	1.4213s	10244	0.6592	7207.49
Test 3	1.5310s	10297	0.7894	6725.67
Test 4	1.5042s	10235	0.6511	6804.28
Test 5	1.4808s	10320	0.6343	6969.21

FileIO Test2 for QEMU: sysbench --num-threads=16 --test=fileio --file-total-size=1G --file-test-mode=rndwr run

	Total Time	Total number of events	Total Number of events per second
Test 1	11.3138s	9766	863.19
Test 2	11.4392s	9636	842.37
Test 3	11.6143s	9995	860.58
Test 4	11.1035s	9804	882.96
Test 5	11.4044s	10620	931.22

FileIO Test3 for Docker: sysbench --num-threads=16 --test=fileio --file-total-size=1G —file-test-mode=seqwr run

Arthi Kundadka
W1606761

	Total Time	Total number of events	Total time taken by event execution	Total Number of events per second
Test 1	3.6235s	65536	38.1923	18086.38
Test 2	3.7728s	65536	37.6042	17370.65
Test 3	3.2722s	65536	32.9982	20028.12
Test 4	3.6915s	65536	37.1749	17753.22
Test 5	3.8927s	65536	38.0737	16835.62

FileIO Test3 for QEMU: sysbench --num-threads=16 --test=fileio --file-total-size=1G — file-test-mode=seqwr run

	Total Time	Total number of events	Total Number of events per second
Test 1	11.191s	17457	1559.91
Test 2	11.1993s	17283	1543.22
Test 3	11.1642s	18199	1630.12
Test 4	11.1784s	18103	1619.46
Test 5	11.1880s	17682	1580.44

Observation and analysis:

In all the test cases, the total number of events per second is better for the Docker (OS virtualization) is more than that of QEMU (System virtualization). This implies that OS virtualization performs better than system virtualization. When the Cpu max prime values are increased, the Sysbench tests for Docker still produce higher number of events per second implying OS virtualization is still better in such cases than system virtualization. Also I tried running the tests for different file test modes, and the result remained the same proving OS virtualization performs better.