System and OS Virtualization

Santa Clara University Department of Computer Science and Engineering

Cloud Computing CSEN – 241

Homework – 1

Student Name – Avani Sanjay Vaidya
Student Id - 007700005517

Student Name – Avani Sanjay Vaidya
Student Id - 007700005517

INDEX

Sr. No	Topic	Page Number
1.	Environment Setup	3
2.	System Virtualization Setup	4
	Disk Format – qcow2	5
	Disk Format – raw	7
	LUKS Encryption	8
3.	Operating System Virtualization Setup	11
4.	Sysbench Setup	14
5.	Sysbench Experiment	16
6.	Experiment Results and Analysis	69
7.	Sysbench Findings and Conclusion	74
8.	Shell Script	75
9.	Automation Scripts	78

1. ENVIRONMENT SETUP

Host System Configuration

The configuration of the host system used to perform the experiment is as follows –

Chip	Apple M2
Total Number of Cores	8 (4 performance and 4 efficiency)
Memory	16 GB
Disk Space	245.11 GB
Available Disk Space	54.33 GB
Operating System	macOS Ventura Version 13.5

2. SYSTEM VIRTUALIZATION SETUP

To perform System Virtualization, we have used the QEMU hypervisor. Further, an Ubuntu Virtual Machine is created with Qemu. Following are the detailed steps to install Qemu and create an Ubuntu Virtual Machine with specific Disk space, CPU and RAM.

a. Install Qemu –

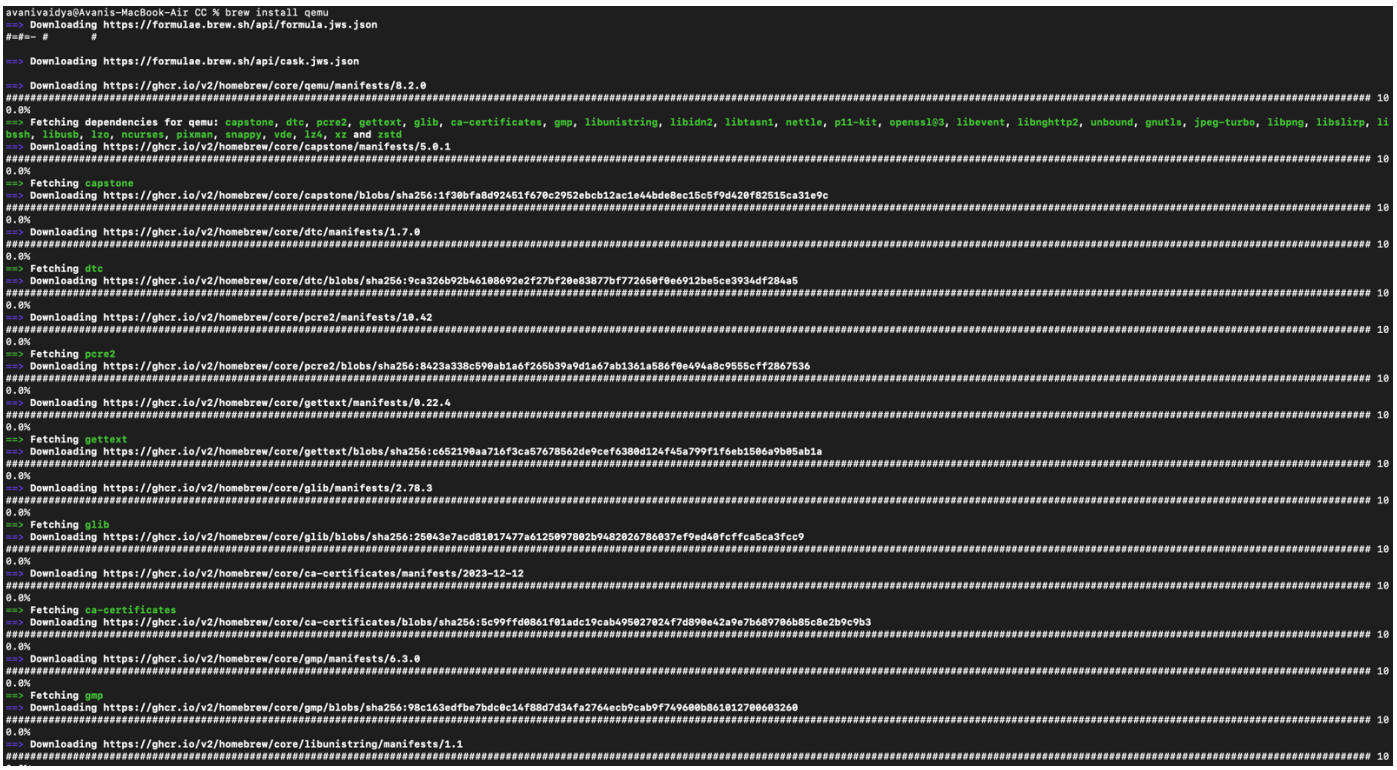
To install Qemu on macOS, we use the package manager Homebrew. We need to install Homebrew first. Open terminal on your host machine and run the following is the command to install Homebrew.

```
avanivaidya@Avanis-MacBook-Air ~ % /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Further, we would need to add Homebrew to the PATH variable as follows.

```
avanivaidya@Avanis-MacBook-Air ~ % (echo; echo 'eval "$(/opt/homebrew/bin/brew shellenv)"') >> /Users/avanivaidya/.zprofile
avanivaidya@Avanis-MacBook-Air ~ % eval "$(/opt/homebrew/bin/brew shellenv)"
```

Now we will install Qemu using Homebrew and following is the command used with the output.



b. Install Ubuntu server ISO image –

In order to install the Ubuntu Guest Virtual Machine, we will need to install the Ubuntu 20.04 server ISO image from the following link

https://releases.ubuntu.com/focal/

c. Create Qemu Image and Install Ubuntu VM –

Disk Image Format – qcow2

Now we will create a Qemu image of Ubuntu in qcow2 disk format with 10G of disk space with the following command

```
sudo qemu-img create ubuntu.img 10G -f qcow2
```

where,

-"ubuntu.img" is the disk image filename

-"10g" of disk space

-"-f" option is used to mention the disk format, here it is qcow2.

Further we will run following command to install the Ubuntu VM using the Ubuntu iso server image from step (b) in the disk image created by above command.

```
sudo qemu-system-x86_64 -hda ubuntu.img -boot d -cdrom
./[UBUNTU_SERVER_ISO_FILE_NAME] -m 2046 -boot strict=on
```

In the above command the flags/option used have following meanings –

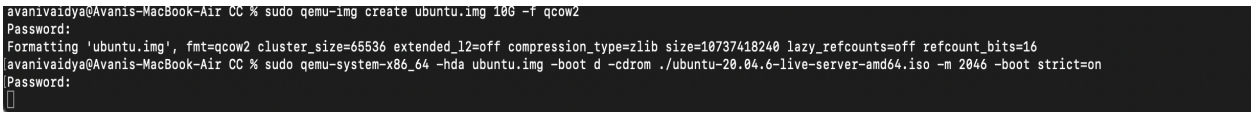
-hda: use file as a hard disk

-boot: this option specifies the boot order, for x86 architecture these drive letters are - a, b (for floppy drives), c (first hard-disk), d (first CD-ROM), n-p (Etherboot from network adapter 1-4). Hard-disk boot is the default option.

-cdrom: this option specifies the .iso file to be used as a base for image we are creating

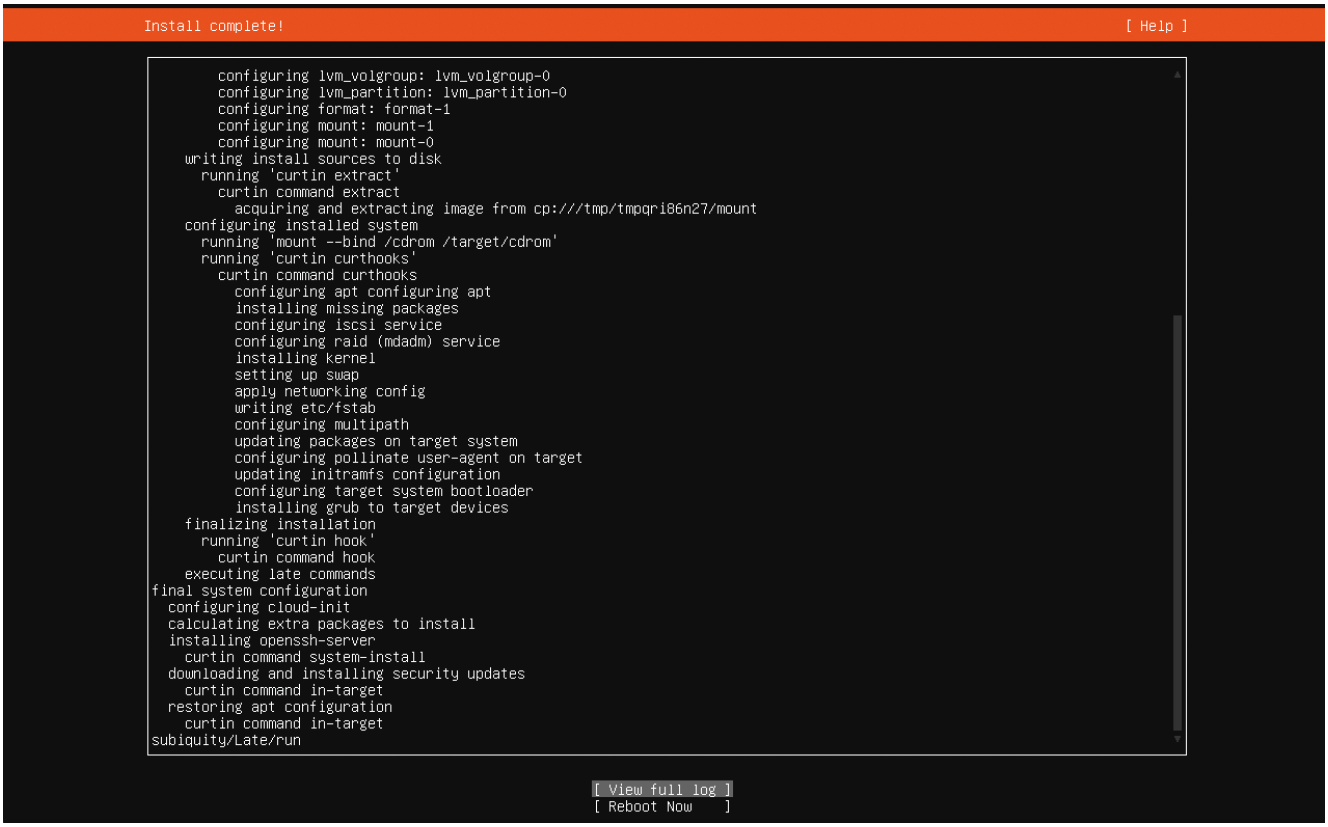
-m: this option sets guest OS's startup RAM to specified value i.e 2048 MB

Following the screenshot of the above commands



After the above command, a QEMU command prompt will open. There will be multiple steps follow and choose the appropriate configurations for our VM like Language, Packages to install etc. to install the VM successfully.

Following screen appears once installation is successful.



Thus, we have successfully installed the Ubuntu Guest VM on the Qemu image. We need to reboot the system now.

Once the system is rebooted, we can start the server using the following command.

```
[avanivaidya@Avanis-MacBook-Air CC % sudo qemu-system-x86_64 -smp 2 -m 4G -hda ubuntu.img
[Password:
```

where,

-smp is the flag used to set the number of CPU's

-m is the flag used to set the amount of RAM

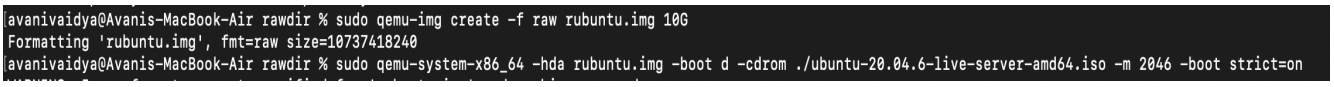
-had is used to specify file to be used as hard disk, in this case it is the Ubuntu image.

We can check the details of the image created above using following command.

```
avanivaidya@Avanis-MacBook-Air CC % qemu-img info ubuntu.img
image: ubuntu.img
file format: qcow2
virtual size: 10 GiB (10737418240 bytes)
disk size: 6.55 GiB
cluster_size: 65536
Format specific information:
    compat: 1.1
    compression type: zlib
    lazy refcounts: false
    refcount bits: 16
    corrupt: false
    extended l2: false
Child node '/file':
    filename: ubuntu.img
    protocol type: file
    file length: 6.53 GiB (7014645760 bytes)
    disk size: 6.55 GiB
avanivaidya@Avanis-MacBook-Air CC %
```

Disk Image format – raw

To create a Qemu image of Ubuntu in raw disk format with 10G of disk space we will follow the same steps as seen in 2.c except the file format will be changed to “raw”. Following is the command to create Qemu image of Ubuntu with raw disk format and output of the command.



Here,

-rubuntu.img” is the disk image file name

-option “-f” is used to specify the disk image format, in this case it is “raw”

-with “10G” of disk space.

The flag/options for the second command seen in the screenshot is similar to mentioned in section 2.c

Thus, after running the above commands a Qemu command prompt will open, where we will need to choose the appropriate configurations for our VM to install it successfully. Once, installed we will need to reboot the system as mentioned in 2.c.

Once the system is rebooted, we can start the server using the same command as qcow2 disk format, as follows –

```
avanivaidya@Avanis-MacBook-Air rawdir % sudo qemu-system-x86_64 -smp 2 -m 4G -hda rubuntu.img
Password:
```

We can check the details of the image created with the following command.

```
avanivaidya@Avanis-MacBook-Air rawdir % qemu-img info rubuntu.img
image: rubuntu.img
file format: raw
virtual size: 10 GiB (10737418240 bytes)
disk size: 4.39 GiB
Child node '/file':
    filename: rubuntu.img
    protocol type: file
    file length: 10 GiB (10737418240 bytes)
    disk size: 4.39 GiB
avanivaidya@Avanis-MacBook-Air rawdir %
```

LUKS Encryption

To generate an encrypted QEMU image of Ubuntu in the qcow2 file format we use the following command with LUKS encryption format.

```
avanivaidya@Avanis-MacBook-Air lukendir % sudo qemu-img create --object secret,id=sec0,data=123456 -f qcow2 -o encrypt.format=luks,encrypt.key-secret=sec0 eubuntu.qcow2 10G
Formatting 'eubuntu.qcow2', fmt=qcow2 encrypt.format=luks encrypt.key-secret=sec0 cluster_size=65536 extended_l2=off compression_type=zlib size=10737418240 lazy_refcounts=off refcount_bits=1
6
```

Description of the flags used in the above command.

--object secret,id=sec0,data=123456: This part creates a QEMU secret object named sec0 with a data value of 123456. Secrets in QEMU can be used for various purposes, including encryption keys.

-o encrypt.format=luks,encrypt.key-secret=sec0: It indicates that the encryption format is LUKS (Linux Unified Key Setup), and it associates the secret object sec0 as the key for encryption.

Now further, we will create the Ubuntu Virtual Machine using the following command

```
avanivaidya@Avanis-MacBook-Air lukendir % sudo qemu-system-x86_64 --object secret,id=sec0,data=123456 -drive driver=qcow2,file.filename=eubuntu.qcow2,encrypt.key-secret=sec0 -boot d -cdrom u
buntu-20.04.6-live-server-amd64.iso -m 2048 -boot strict=on
```

Description of the flags used in the above command.

-had: Specifies the file to be used as a hard disk.

-boot: Defines the boot order for x86 architecture, where drive letters represent different boot options. For example, a, b (for floppy drives), c (first hard disk), d (first CD-ROM), n-p (Etherboot from network adapter 1-4). The default option is the hard disk boot.

-cdrom: Indicates the .iso file to be utilized as the base for the image being created.

-m: Sets the startup RAM for the guest OS to the specified value, in this case, 2048 MB.

--object secret,id=sec0,data=123456: This flag passes a QEMU secret object named sec0 with a data value of 123456. This secret can be used for various purposes, including encryption keys.

-drive driver=qcow2,file.filename=eubuntu.qcow2,encrypt.key-secret=sec0: This flag configures a virtual drive for the virtual machine. It specifies that the driver is qcow2, sets the filename to eubuntu.qcow2, and uses the secret object sec0 as the key for encryption. This indicates that the virtual machine will use an encrypted qcow2 image file.

The information of the created image can be seen using following command.

```
avanivaidya@Avanis-MacBook-Air lukendir % sudo qemu-img info eubuntu.qcow2
[Password:
[image: eubuntu.qcow2
file format: qcow2
virtual size: 10 GiB (10737418240 bytes)
disk size: 4.67 GiB
encrypted: yes
cluster_size: 65536
Format specific information:
    compat: 1.1
    compression type: zlib
    lazy refcounts: false
    refcount bits: 16
    encrypt:
        ivgen alg: plain64
        hash alg: sha256
        cipher alg: aes-256
        uuid: fd95eed3-a378-46cc-879c-f60f75d75004
        format: luks
        cipher mode: xts
        slots:
            [0]:
                active: true
                iters: 1510510
                key offset: 4096
                stripes: 4000
            [1]:
                active: false
                key offset: 262144
            [2]:
                active: false
                key offset: 520192
            [3]:
                active: false
                key offset: 778240
            [4]:
                active: false
                key offset: 1036288
            [5]:
                active: false
                key offset: 1294336
            [6]:
                active: false
                key offset: 1552384
            [7]:
                active: false
                key offset: 1810432
        payload offset: 2068480
        master key iters: 377889
    corrupt: false
    extended l2: false
Child node '/file':
    filename: eubuntu.qcow2
    protocol type: file
    file length: 4.66 GiB (5004066816 bytes)
    disk size: 4.67 GiB
```

3. OPERATING SYSTEM VIRTUALIZATION SETUP

To perform OS Virtualization we will use Docker, which is a container management platform.

We will install the Ubuntu Operating System on our Docker image.

a. Install Docker Desktop -

To install Docker Desktop on macOS use the following link.

https://docs.docker.com/desktop/mac/install/

Follow the installation steps to successfully install Docker Desktop on your machine.

Once Docker is installed you can open terminal and check if installation is successful with following command and the output is seen as -

```
avanivaidya@Avanis-MacBook-Air CC % docker

Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Common Commands:
  run         Create and run a new container from an image
  exec        Execute a command in a running container
  ps          List containers
  build       Build an image from a Dockerfile
  pull        Download an image from a registry
  push        Upload an image to a registry
  images      List images
  login       Log in to a registry
  logout      Log out from a registry
  search      Search Docker Hub for images
  version     Show the Docker version information
  info        Display system-wide information

Management Commands:
  builder     Manage builds
  buildx*     Docker Buildx (Docker Inc., v0.12.0-desktop.2)
  compose*    Docker Compose (Docker Inc., v2.23.3-desktop.2)
  container   Manage containers
  context     Manage contexts
  dev*        Docker Dev Environments (Docker Inc., v0.1.0)
  extension*  Manages Docker extensions (Docker Inc., v0.2.21)
  feedback*   Provide feedback, right in your terminal! (Docker Inc., 0.1)
  image       Manage images
  init*       Creates Docker-related starter files for your project (Docker Inc., v0.1.0-beta.10)
  manifest    Manage Docker image manifests and manifest lists
  network     Manage networks
  plugin      Manage plugins
  sbom*       View the packaged-based Software Bill Of Materials (SBOM) for an image (Anchore Inc., 0.6.0)
  scan*       Docker Scan (Docker Inc., v0.26.0)
  scout*      Docker Scout (Docker Inc., v1.2.0)
  system      Manage Docker
  trust       Manage trust on Docker images
  volume      Manage volumes

Swarm Commands:
  swarm       Manage Swarm

Commands:
  attach      Attach local standard input, output, and error streams to a running container
  commit      Create a new image from a container's changes
  cp          Copy files/folders between a container and the local filesystem
  create      Create a new container
  diff        Inspect changes to files or directories on a container's filesystem
  events      Get real time events from the server
  export      Export a container's filesystem as a tar archive
  history     Show the history of an image
  import      Import the contents from a tarball to create a filesystem image
```

b. Download latest Ubuntu image –

To install latest Ubuntu image run the following command from terminal and the output is seen as follows.

```
avanivaidya@Avanis-MacBook-Air CC % docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
ce9ebea987c2: Pull complete
Digest: sha256:e6173d4dc55e76b87c4af8db8821b1feae4146dd47341e4d431118c7dd060a74
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview ubuntu
```

c. Create Docker container –

Now we will create a Docker container with the above Ubuntu image with the following command. The command will start the container and we will be redirected to the bash shell of the Ubuntu container as seen in following screenshot.

```
avanivaidya@Avanis-MacBook-Air CC % docker run -i -t ubuntu /bin/bash
root@90b645ea06bf:/# sudo apt update
```

We can also run the docker container by mentioning the number of CPU cores and amount of RAM as follows, which we need in sysbench experiment testing.

```
avanivaidya@Avanis-MacBook-Air CC % docker run -i -t --cpus=3 --memory=2g ubuntu /bin/bash
root@b4d47e1d0fc8:/# sysbench --test=cpu --cpu-max-prime=2000 --time=30 run
```

where

--cpus is the flag used to mention the number of CPU cores

--memory is used to mention the amount of RAM

d. Check image id and image history –

Image id of the docker container created above can be checked with the following command.

```
avanivaidya@Avanis-MacBook-Air CC % docker image ls
REPOSITORY                  TAG       IMAGE ID       CREATED         SIZE
ubuntu                      latest    e2e172ecd069   2 weeks ago     69.3MB
docker/welcome-to-docker    latest    648f93a1ba7d   2 months ago    19MB
avanivaidya@Avanis-MacBook-Air CC %
```

History of the image can be seen with the following command.

```
avanivaidya@Avanis-MacBook-Air CC % docker history ubuntu
IMAGE          CREATED       CREATED BY                                      SIZE      COMMENT
e2e172ecd069   2 weeks ago   /bin/sh -c #(nop)  CMD ["/bin/bash"]            0B
<missing>      2 weeks ago   /bin/sh -c #(nop) ADD file:5703a6689620ec495…   69.3MB
<missing>      2 weeks ago   /bin/sh -c #(nop)  LABEL org.opencontainers.…   0B
<missing>      2 weeks ago   /bin/sh -c #(nop)  LABEL org.opencontainers.…   0B
<missing>      2 weeks ago   /bin/sh -c #(nop)  ARG LAUNCHPAD_BUILD_ARCH     0B
<missing>      2 weeks ago   /bin/sh -c #(nop)  ARG RELEASE                  0B
avanivaidya@Avanis-MacBook-Air CC %
```

4. SYSBENCH SETUP

We will use sysbench to understand the performance characteristics of each of the virtualization technologies.

a. Sysbench for Qemu Ubuntu Guest Virtual Machine -

To install sysbench on Ubuntu run the following commands on the Qemu command prompt.

$ sudo apt update

$ sudo apt install sysbench

Following screenshot shows the output of the above command for Qemu.

```
avaidya2@avaidya2:~$ sudo apt install sysbench
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libluajit-5.1-2 libluajit-5.1-common libmysqlclient21 libpq5 mysql-common
The following NEW packages will be installed:
  libluajit-5.1-2 libluajit-5.1-common libmysqlclient21 libpq5 mysql-common sysbench
0 upgraded, 6 newly installed, 0 to remove and 60 not upgraded.
Need to get 1804 kB of archives.
After this operation, 9082 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 libluajit-5.1-common all 2.1.0~beta3+dfsg-5.1build1 [44.3 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 libluajit-5.1-2 amd64 2.1.0~beta3+dfsg-5.1build1 [228 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu focal/main amd64 mysql-common all 5.8+1.0.5ubuntu2 [7496 B]
Get:4 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libmysqlclient21 amd64 8.0.36-0ubuntu0.20.04.1 [1302 kB]
Get:5 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libpq5 amd64 12.17-0ubuntu0.20.04.1 [116 kB]
Get:6 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 sysbench amd64 1.0.18+ds-1 [107 kB]
Fetched 1804 kB in 1s (1300 kB/s)
Selecting previously unselected package libluajit-5.1-common.
(Reading database ... 72290 files and directories currently installed.)
Preparing to unpack .../0-libluajit-5.1-common_2.1.0~beta3+dfsg-5.1build1_all.deb ...
Unpacking libluajit-5.1-common (2.1.0~beta3+dfsg-5.1build1) ...
Selecting previously unselected package libluajit-5.1-2:amd64.
Preparing to unpack .../1-libluajit-5.1-2_2.1.0~beta3+dfsg-5.1build1_amd64.deb ...
Unpacking libluajit-5.1-2:amd64 (2.1.0~beta3+dfsg-5.1build1) ...
Selecting previously unselected package mysql-common.
Preparing to unpack .../2-mysql-common_5.8+1.0.5ubuntu2_all.deb ...
Unpacking mysql-common (5.8+1.0.5ubuntu2) ...
Selecting previously unselected package libmysqlclient21:amd64.
Preparing to unpack .../3-libmysqlclient21_8.0.36-0ubuntu0.20.04.1_amd64.deb ...
Unpacking libmysqlclient21:amd64 (8.0.36-0ubuntu0.20.04.1) ...
Selecting previously unselected package libpq5:amd64.
Preparing to unpack .../4-libpq5_12.17-0ubuntu0.20.04.1_amd64.deb ...
Unpacking libpq5:amd64 (12.17-0ubuntu0.20.04.1) ...
Selecting previously unselected package sysbench.
Preparing to unpack .../5-sysbench_1.0.18+ds-1_amd64.deb ...
Unpacking sysbench (1.0.18+ds-1) ...
Setting up mysql-common (5.8+1.0.5ubuntu2) ...
update-alternatives: using /etc/mysql/my.cnf.fallback to provide /etc/mysql/my.cnf (my.cnf) in auto mode
Setting up libmysqlclient21:amd64 (8.0.36-0ubuntu0.20.04.1) ...
Setting up libpq5:amd64 (12.17-0ubuntu0.20.04.1) ...
Setting up libluajit-5.1-common (2.1.0~beta3+dfsg-5.1build1) ...
Setting up libluajit-5.1-2:amd64 (2.1.0~beta3+dfsg-5.1build1) ...
Setting up sysbench (1.0.18+ds-1) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.14) ...
```

b. Sysbench for Docker Ubuntu Container –

To install sysbench on Docker, you will need to run the above commands. However, docker images are lightweight and thus, we need to explicitly install sudo package. Following screenshots shows the commands to install sudo package and add the current user to sudo group users.

```
root@90b645ea06bf:/# apt-get update
Hit:1 http://ports.ubuntu.com/ubuntu-ports jammy InRelease
Hit:2 http://ports.ubuntu.com/ubuntu-ports jammy-updates InRelease
Hit:3 http://ports.ubuntu.com/ubuntu-ports jammy-backports InRelease
Hit:4 http://ports.ubuntu.com/ubuntu-ports jammy-security InRelease
Reading package lists... Done
root@90b645ea06bf:/# apt-get -y install sudo
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  sudo
0 upgraded, 1 newly installed, 0 to remove and 9 not upgraded.
Need to get 807 kB of archives.
After this operation, 2392 kB of additional disk space will be used.
Get:1 http://ports.ubuntu.com/ubuntu-ports jammy-updates/main arm64 sudo arm64 1.9.9-1ubuntu2.4 [807 kB]
Fetched 807 kB in 2s (490 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package sudo.
(Reading database ... 4387 files and directories currently installed.)
Preparing to unpack .../sudo_1.9.9-1ubuntu2.4_arm64.deb ...
Unpacking sudo (1.9.9-1ubuntu2.4) ...
Setting up sudo (1.9.9-1ubuntu2.4) ...
Processing triggers for libc-bin (2.35-0ubuntu3.6) ...
root@90b645ea06bf:/# useradd -m docker && echo "docker:docker" | chpasswd && adduser docker sudo
Adding user `docker' to group `sudo' ...
Adding user docker to group sudo
Done.
```

Now you can run the following commands (mentioned in section 4.a) to install sysbench.

Following screenshot contains the output after running the commands.

```
Done.
root@a44dee4906a2:/# sudo apt update
Hit:1 http://ports.ubuntu.com/ubuntu-ports jammy InRelease
Hit:2 http://ports.ubuntu.com/ubuntu-ports jammy-updates InRelease
Hit:3 http://ports.ubuntu.com/ubuntu-ports jammy-backports InRelease
Hit:4 http://ports.ubuntu.com/ubuntu-ports jammy-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
9 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@a44dee4906a2:/# sudo apt install sysbench
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libaio1 libldap-2.5-0 libldap-common libluajit-5.1-2 libluajit-5.1-common libmysqlclient21 libpq5 libsasl2-2 libsasl2-modules libsasl2-modules-db mysql-common
Suggested packages:
```

5. SYSBENCH EXPERIMENT

We have performed the Sysbench test cases for –

a. qcow Qemu image format
b. raw Qemu image format
c. Docker

In each case, we have built a Virtual Machine with two arguments for CPU and RAM each.

Further, we have run Sysbench test cases for cpu, fileio and memory test modes, where for each test mode we have chosen two arguments.

Thus, the total test cases performed are –

○ QEMU: 2 disk drives x 2 QEMU CPU x 2 QEMU Memory x 6 sysbench = 48

○ Docker: 2 Docker CPU setting x 2 Docker Memory setting x 6 sysbench = 24

This section includes the experimental results for qcow vs raw vs docker, with the different sysbench test cases mentioned as above.

CPU TEST

1) CPU TEST WITH CPU – 2 CORES AND RAM – 4G

Test 1 – cpu-max-prime = 2000

A. Qemu qcow2 format –

Following screenshot shows the sysbench command with cpu test mode and time set to 30 seconds.

```
avaidya1@avaidya1:~$ sysbench --test=cpu --cpu-max-prime=2000 --time=30 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: 2000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  4499.57

General statistics:
    total time:                          30.0016s
    total number of events:              135016

Latency (ms):
         min:                                    0.21
         avg:                                    0.22
         max:                                    3.48
         95th percentile:                        0.25
         sum:                                29524.09

Threads fairness:
    events (avg/stddev):           135016.0000/0.00
    execution time (avg/stddev):   29.5241/0.00
```

B. Qemu raw format –

```
General statistics:
    total time:                          30.0016s
    total number of events:              134975

Latency (ms):
         min:                                    0.21
         avg:                                    0.22
         max:                                   13.28
         95th percentile:                        0.25
         sum:                                29520.19

Threads fairness:
    events (avg/stddev):           134975.0000/0.00
    execution time (avg/stddev):   29.5202/0.00

avaidya2@avaidya2:~$ sysbench --test=cpu --cpu-max-prime=2000 --time=30 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: 2000

Initializing worker threads...

Threads started!

^[4CPU speed:
    events per second:  4493.69

General statistics:
    total time:                          30.0008s
    total number of events:              134833

Latency (ms):
         min:                                    0.21
         avg:                                    0.22
         max:                                    5.47
```

C. Docker –

```
root@a44dee4906a2:/# sysbench --test=cpu --cpu-max-prime=2000 --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (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: 2000

Initializing worker threads...

Threads started!

CPU speed:
    events per second: 42797.66

General statistics:
    total time:                          30.0008s
    total number of events:              1284078

Latency (ms):
         min:                                    0.02
         avg:                                    0.02
         max:                                   21.57
         95th percentile:                        0.02
         sum:                                29829.46

Threads fairness:
    events (avg/stddev):           1284078.0000/0.00
    execution time (avg/stddev):   29.8295/0.00
```

Test 2 – cpu-max-prime=3000

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=cpu --cpu-max-prime=3000 --time=30 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: 3000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  2726.02

General statistics:
    total time:                          30.0011s
    total number of events:              81794

Latency (ms):
         min:                                    0.35
         avg:                                    0.36
         max:                                    3.02
         95th percentile:                        0.39
         sum:                                29688.84

Threads fairness:
    events (avg/stddev):           81794.0000/0.00
    execution time (avg/stddev):   29.6888/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=cpu --cpu-max-prime=3000 --time=30 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: 3000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  2727.96

General statistics:
    total time:                          30.0009s
    total number of events:              81853

Latency (ms):
         min:                                    0.35
         avg:                                    0.36
         max:                                    7.79
         95th percentile:                        0.39
         sum:                                29682.39

Threads fairness:
    events (avg/stddev):           81853.0000/0.00
    execution time (avg/stddev):   29.6824/0.00
```

C. Docker –

```
[root@a44dee4906a2:/# sysbench --test=cpu --cpu-max-prime=3000 --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (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: 3000

Initializing worker threads...

Threads started!

CPU speed:
    events per second: 24077.25

General statistics:
    total time:                          30.0009s
    total number of events:              722407

Latency (ms):
         min:                                    0.04
         avg:                                    0.04
         max:                                   18.15
         95th percentile:                        0.05
         sum:                                29846.29

Threads fairness:
    events (avg/stddev):           722407.0000/0.00
    execution time (avg/stddev):   29.8463/0.00
```

2) CPU TEST WITH CPU – 3 CORES AND RAM – 2G

Test 1 – cpu-max-prime=2000

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=cpu --cpu-max-prime=2000 --time=30 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: 2000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  4463.26

General statistics:
    total time:                          30.0017s
    total number of events:              133930

Latency (ms):
         min:                                    0.21
         avg:                                    0.22
         max:                                   20.85
         95th percentile:                        0.25
         sum:                                29507.74

Threads fairness:
    events (avg/stddev):           133930.0000/0.00
    execution time (avg/stddev):   29.5077/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=cpu --cpu-max-prime=2000 --time=30 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: 2000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  4416.55

General statistics:
    total time:                          30.0016s
    total number of events:              132525

Latency (ms):
         min:                                    0.21
         avg:                                    0.22
         max:                                    3.29
         95th percentile:                        0.25
         sum:                                29501.19

Threads fairness:
    events (avg/stddev):           132525.0000/0.00
    execution time (avg/stddev):   29.5012/0.00
```

C. Docker –

```
root@b4d47e1d0fc8:/# sysbench --test=cpu --cpu-max-prime=2000 --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (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: 2000

Initializing worker threads...

Threads started!

CPU speed:
    events per second: 42981.29

General statistics:
    total time:                          30.0002s
    total number of events:              1289511

Latency (ms):
         min:                                    0.02
         avg:                                    0.02
         max:                                   12.08
         95th percentile:                        0.02
         sum:                                29839.29

Threads fairness:
    events (avg/stddev):           1289511.0000/0.00
    execution time (avg/stddev):   29.8393/0.00
```

Test 2 – cpu-max-prime=3000

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=cpu --cpu-max-prime=3000 --time=30 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: 3000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  2698.77

General statistics:
    total time:                          30.0012s
    total number of events:              80976

Latency (ms):
         min:                                    0.35
         avg:                                    0.37
         max:                                    5.50
         95th percentile:                        0.39
         sum:                                29698.56

Threads fairness:
    events (avg/stddev):           80976.0000/0.00
    execution time (avg/stddev):   29.6986/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=cpu --cpu-max-prime=3000 --time=30 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: 3000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  2713.28

General statistics:
    total time:                          30.0014s
    total number of events:              81415

Latency (ms):
         min:                                    0.35
         avg:                                    0.36
         max:                                   14.59
         95th percentile:                        0.39
         sum:                                29675.76

Threads fairness:
    events (avg/stddev):           81415.0000/0.00
    execution time (avg/stddev):   29.6758/0.00
```

C. Docker –

```
root@b4d47e1d0fc8:/# sysbench --test=cpu --cpu-max-prime=3000 --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (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: 3000

Initializing worker threads...

Threads started!

CPU speed:
    events per second: 24952.04

General statistics:
    total time:                          30.0002s
    total number of events:              748592

Latency (ms):
         min:                                    0.04
         avg:                                    0.04
         max:                                    5.35
         95th percentile:                        0.05
         sum:                                29902.30

Threads fairness:
    events (avg/stddev):           748592.0000/0.00
    execution time (avg/stddev):   29.9023/0.00
```

3) CPU TEST WITH CPU – 2 CORES AND RAM – 2G

Test 1 – cpu-max-prime=2000

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=cpu --cpu-max-prime=2000 --time=30 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: 2000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  4475.95

General statistics:
    total time:                          30.0015s
    total number of events:              134306

Latency (ms):
         min:                                    0.21
         avg:                                    0.22
         max:                                   74.22
         95th percentile:                        0.24
         sum:                                29504.05

Threads fairness:
    events (avg/stddev):           134306.0000/0.00
    execution time (avg/stddev):   29.5041/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=cpu --cpu-max-prime=2000 --time=30 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: 2000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  4470.07

General statistics:
    total time:                          30.0016s
    total number of events:              134128

Latency (ms):
         min:                                    0.21
         avg:                                    0.22
         max:                                   56.56
         95th percentile:                        0.25
         sum:                                29494.38

Threads fairness:
    events (avg/stddev):           134128.0000/0.00
    execution time (avg/stddev):   29.4944/0.00
```

C. Docker –

```
[root@8dbeafbeff3a:/# sysbench --test=cpu --cpu-max-prime=2000 --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (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: 2000

Initializing worker threads...

Threads started!

CPU speed:
    events per second: 42934.35

General statistics:
    total time:                          30.0002s
    total number of events:              1288141

Latency (ms):
         min:                                    0.02
         avg:                                    0.02
         max:                                    9.54
         95th percentile:                        0.02
         sum:                                29837.07

Threads fairness:
    events (avg/stddev):           1288141.0000/0.00
    execution time (avg/stddev):   29.8371/0.00
```

Test 2 – cpu-max-prime=3000

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=cpu --cpu-max-prime=3000 --time=30 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: 3000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  2735.61

General statistics:
    total time:                          30.0009s
    total number of events:              82081

Latency (ms):
         min:                                    0.35
         avg:                                    0.36
         max:                                    2.43
         95th percentile:                        0.39
         sum:                                29692.74

Threads fairness:
    events (avg/stddev):           82081.0000/0.00
    execution time (avg/stddev):   29.6927/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=cpu --cpu-max-prime=3000 --time=30 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: 3000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  2722.04

General statistics:
    total time:                          30.0009s
    total number of events:              81673

Latency (ms):
         min:                                    0.35
         avg:                                    0.36
         max:                                    8.19
         95th percentile:                        0.39
         sum:                                29687.10

Threads fairness:
    events (avg/stddev):           81673.0000/0.00
    execution time (avg/stddev):   29.6871/0.00
```

C. Docker –

```
[root@8dbeafbeff3a:/# sysbench --test=cpu --cpu-max-prime=3000 --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (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: 3000

Initializing worker threads...

Threads started!

CPU speed:
    events per second: 24430.78

General statistics:
    total time:                          30.0012s
    total number of events:              733021

Latency (ms):
         min:                                    0.04
         avg:                                    0.04
         max:                                  127.95
         95th percentile:                        0.05
         sum:                                29867.79

Threads fairness:
    events (avg/stddev):           733021.0000/0.00
    execution time (avg/stddev):   29.8678/0.00
```

4) CPU TEST WITH CPU – 3 CORES AND RAM – 4G

Test 1 – cpu-max-prime=2000

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=cpu --cpu-max-prime=2000 --time=30 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: 2000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  4543.40

General statistics:
    total time:                          30.0012s
    total number of events:              136327

Latency (ms):
         min:                                    0.21
         avg:                                    0.22
         max:                                    1.01
         95th percentile:                        0.24
         sum:                                29512.36

Threads fairness:
    events (avg/stddev):           136327.0000/0.00
    execution time (avg/stddev):   29.5124/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=cpu --cpu-max-prime=2000 --time=30 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: 2000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  4389.99

General statistics:
    total time:                          30.0017s
    total number of events:              131728

Latency (ms):
         min:                                    0.21
         avg:                                    0.22
         max:                                   27.58
         95th percentile:                        0.25
         sum:                                29476.30

Threads fairness:
    events (avg/stddev):           131728.0000/0.00
    execution time (avg/stddev):   29.4763/0.00
```

C. Docker –

```
[root@cf474904ed48:/# sysbench --test=cpu --cpu-max-prime=2000 --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (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: 2000

Initializing worker threads...

Threads started!

CPU speed:
    events per second: 41947.15

General statistics:
    total time:                          30.0010s
    total number of events:              1258509

Latency (ms):
         min:                                    0.02
         avg:                                    0.02
         max:                                    6.40
         95th percentile:                        0.03
         sum:                                29826.40

Threads fairness:
    events (avg/stddev):           1258509.0000/0.00
    execution time (avg/stddev):   29.8264/0.00
```

Test 2 – cpu-max-prime=3000

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=cpu --cpu-max-prime=3000 --time=30 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: 3000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  2733.80

General statistics:
    total time:                          30.0009s
    total number of events:              82028

Latency (ms):
         min:                                    0.35
         avg:                                    0.36
         max:                                   64.43
         95th percentile:                        0.39
         sum:                                29684.59

Threads fairness:
    events (avg/stddev):           82028.0000/0.00
    execution time (avg/stddev):   29.6846/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=cpu --cpu-max-prime=3000 --time=30 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: 3000

Initializing worker threads...

Threads started!

CPU speed:
    events per second:  2713.04

General statistics:
    total time:                          30.0011s
    total number of events:              81405

Latency (ms):
         min:                                    0.35
         avg:                                    0.36
         max:                                   13.13
         95th percentile:                        0.39
         sum:                                29649.12

Threads fairness:
    events (avg/stddev):           81405.0000/0.00
    execution time (avg/stddev):   29.6491/0.00
```

C. Docker –

```
root@cf474904ed48:/# sysbench --test=cpu --cpu-max-prime=3000 --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (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: 3000

Initializing worker threads...

Threads started!

CPU speed:
    events per second: 25135.63

General statistics:
    total time:                          30.0001s
    total number of events:              754110

Latency (ms):
         min:                                    0.04
         avg:                                    0.04
         max:                                    0.83
         95th percentile:                        0.04
         sum:                                29904.57

Threads fairness:
    events (avg/stddev):           754110.0000/0.00
    execution time (avg/stddev):   29.9046/0.00
```

MEMORY TEST

1) MEMORY TEST WITH CPU – 2 CORES AND RAM – 4 G

Test 1 – memory-block-size=1G

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=memory --memory-block-size=1G --time=30 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


Running memory speed test with the following options:
  block size: 1048576KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 80 (    2.66 per second)

81920.00 MiB transferred (2722.14 MiB/sec)


General statistics:
    total time:                          30.0887s
    total number of events:              80

Latency (ms):
         min:                                  327.49
         avg:                                  375.86
         max:                                  608.81
         95th percentile:                      383.33
         sum:                                30068.80

Threads fairness:
    events (avg/stddev):           80.0000/0.00
    execution time (avg/stddev):   30.0688/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=memory --memory-block-size=1G --time=30 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


Running memory speed test with the following options:
  block size: 1048576KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 80 (    2.65 per second)

81920.00 MiB transferred (2716.92 MiB/sec)


General statistics:
    total time:                          30.1477s
    total number of events:              80

Latency (ms):
         min:                                  328.61
         avg:                                  376.65
         max:                                  538.44
         95th percentile:                      390.30
         sum:                                30131.63

Threads fairness:
    events (avg/stddev):           80.0000/0.00
    execution time (avg/stddev):   30.1316/0.00
```

C. Docker –

```
[root@a44dee4906a2:/# sysbench --test=memory --memory-block-size=1G --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Running memory speed test with the following options:
  block size: 1048576KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 100 (   13.57 per second)

102400.00 MiB transferred (13895.87 MiB/sec)


General statistics:
    total time:                          7.3562s
    total number of events:              100

Latency (ms):
         min:                                   44.78
         avg:                                   73.50
         max:                                  570.43
         95th percentile:                      104.84
         sum:                                 7349.59

Threads fairness:
    events (avg/stddev):           100.0000/0.00
    execution time (avg/stddev):   7.3496/0.00
```

Test 2 – memory-block-size=2G

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=memory --memory-block-size=2G --time=30 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


Running memory speed test with the following options:
  block size: 2097152KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 36 (    1.19 per second)

73728.00 MiB transferred (2443.26 MiB/sec)


General statistics:
    total time:                          30.1718s
    total number of events:              36

Latency (ms):
         min:                                  748.39
         avg:                                  837.66
         max:                                 1541.93
         95th percentile:                     1032.01
         sum:                                30155.82

Threads fairness:
    events (avg/stddev):           36.0000/0.00
    execution time (avg/stddev):   30.1558/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=memory --memory-block-size=2G --time=30 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


Running memory speed test with the following options:
  block size: 2097152KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 34 (    1.12 per second)

69632.00 MiB transferred (2297.49 MiB/sec)


General statistics:
    total time:                          30.3034s
    total number of events:              34

Latency (ms):
         min:                                  748.97
         avg:                                  890.86
         max:                                 1738.55
         95th percentile:                     1401.61
         sum:                                30289.09

Threads fairness:
    events (avg/stddev):           34.0000/0.00
    execution time (avg/stddev):   30.2891/0.00
```

C. Docker –

```
root@a44dee4906a2:/# sysbench --test=memory --memory-block-size=2G --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Running memory speed test with the following options:
  block size: 2097152KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 50 (    6.73 per second)

102400.00 MiB transferred (13789.54 MiB/sec)


General statistics:
    total time:                          7.4206s
    total number of events:              50

Latency (ms):
         min:                                    89.86
         avg:                                   148.34
         max:                                   502.25
         95th percentile:                       248.83
         sum:                                  7416.85

Threads fairness:
    events (avg/stddev):           50.0000/0.00
    execution time (avg/stddev):   7.4169/0.00
```

2) MEMORY TEST WITH CPU – 3 CORES AND RAM – 2G

Test 1 – memory-block-size=1G

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=memory --memory-block-size=1G --time=30 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


Running memory speed test with the following options:
  block size: 1048576KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 79 (    2.60 per second)

80896.00 MiB transferred (2667.28 MiB/sec)


General statistics:
    total time:                          30.3247s
    total number of events:              79

Latency (ms):
         min:                                  372.68
         avg:                                  383.64
         max:                                  678.11
         95th percentile:                      411.96
         sum:                                30307.37

Threads fairness:
    events (avg/stddev):           79.0000/0.00
    execution time (avg/stddev):   30.3074/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=memory --memory-block-size=1G --time=30 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


Running memory speed test with the following options:
  block size: 1048576KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 81 (    2.67 per second)

82944.00 MiB transferred (2734.27 MiB/sec)


General statistics:
    total time:                          30.3291s
    total number of events:              81

Latency (ms):
         min:                                  327.21
         avg:                                  374.25
         max:                                  693.05
         95th percentile:                      383.33
         sum:                                30313.92

Threads fairness:
    events (avg/stddev):           81.0000/0.00
    execution time (avg/stddev):   30.3139/0.00
```

C. Docker –

```
root@b4d47e1d0fc8:/# sysbench --test=memory --memory-block-size=1G --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Running memory speed test with the following options:
  block size: 1048576KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 100 (   12.79 per second)

102400.00 MiB transferred (13093.40 MiB/sec)


General statistics:
    total time:                          7.8183s
    total number of events:              100

Latency (ms):
         min:                                   44.99
         avg:                                   78.15
         max:                                  580.48
         95th percentile:                       90.78
         sum:                                 7815.16

Threads fairness:
    events (avg/stddev):           100.0000/0.00
    execution time (avg/stddev):   7.8152/0.00
```

Test 2 – memory-block-size=2G

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=memory --memory-block-size=2G --time=30 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


Running memory speed test with the following options:
  block size: 2097152KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 1 (    0.02 per second)

2048.00 MiB transferred (43.56 MiB/sec)


General statistics:
    total time:                          47.0065s
    total number of events:              1

Latency (ms):
         min:                                    46984.85
         avg:                                    46984.85
         max:                                    46984.85
         95th percentile:                        46941.21
         sum:                                    46984.85

Threads fairness:
    events (avg/stddev):           1.0000/0.00
    execution time (avg/stddev):   46.9849/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=memory --memory-block-size=2G --time=30 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


Running memory speed test with the following options:
  block size: 2097152KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 1 (    0.02 per second)

2048.00 MiB transferred (43.41 MiB/sec)


General statistics:
    total time:                          47.1655s
    total number of events:              1

Latency (ms):
         min:                                    47101.73
         avg:                                    47101.73
         max:                                    47101.73
         95th percentile:                        46941.21
         sum:                                    47101.73

Threads fairness:
    events (avg/stddev):           1.0000/0.00
    execution time (avg/stddev):   47.1017/0.00
```

C. Docker –

```
root@b4d47e1d0fc8:/# sysbench --test=memory --memory-block-size=2G --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Running memory speed test with the following options:
  block size: 2097152KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 7 (    0.22 per second)

14336.00 MiB transferred (445.05 MiB/sec)


General statistics:
    total time:                          32.2093s
    total number of events:              7

Latency (ms):
         min:                                  3349.63
         avg:                                  4600.46
         max:                                  5977.57
         95th percentile:                      6026.41
         sum:                                 32203.20

Threads fairness:
    events (avg/stddev):           7.0000/0.00
    execution time (avg/stddev):   32.2032/0.00
```

3) MEMORY TEST WITH CPU – 2 CORES AND RAM – 2G

Test 1 – memory-block-size=1G

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=memory --memory-block-size=1G --time=30 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


Running memory speed test with the following options:
  block size: 1048576KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 77 (    2.56 per second)

78848.00 MiB transferred (2624.33 MiB/sec)


General statistics:
    total time:                          30.0404s
    total number of events:              77

Latency (ms):
         min:                                  373.56
         avg:                                  389.92
         max:                                  675.19
         95th percentile:                      434.83
         sum:                                30023.56

Threads fairness:
    events (avg/stddev):           77.0000/0.00
    execution time (avg/stddev):   30.0236/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=memory --memory-block-size=1G --time=30 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


Running memory speed test with the following options:
  block size: 1048576KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 80 (    2.64 per second)

81920.00 MiB transferred (2700.52 MiB/sec)


General statistics:
    total time:                          30.3306s
    total number of events:              80

Latency (ms):
         min:                                  327.30
         avg:                                  378.92
         max:                                  657.51
         95th percentile:                      397.39
         sum:                                30313.92

Threads fairness:
    events (avg/stddev):           80.0000/0.00
    execution time (avg/stddev):   30.3139/0.00
```

C. Docker –

```
[root@8dbeafbeff3a:/# sysbench --test=memory --memory-block-size=1G --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Running memory speed test with the following options:
  block size: 1048576KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 100 (   14.85 per second)

102400.00 MiB transferred (15207.76 MiB/sec)


General statistics:
    total time:                          6.7305s
    total number of events:              100

Latency (ms):
         min:                                   45.22
         avg:                                   67.27
         max:                                  165.36
         95th percentile:                       78.60
         sum:                                 6726.79

Threads fairness:
    events (avg/stddev):           100.0000/0.00
    execution time (avg/stddev):   6.7268/0.00
```

Test 2 – memory-block-size=2G

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=memory --memory-block-size=2G --time=30 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


Running memory speed test with the following options:
  block size: 2097152KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 1 (    0.02 per second)

2048.00 MiB transferred (42.49 MiB/sec)


General statistics:
    total time:                          48.1927s
    total number of events:              1

Latency (ms):
         min:                                    48145.99
         avg:                                    48145.99
         max:                                    48145.99
         95th percentile:                        47794.11
         sum:                                    48145.99

Threads fairness:
    events (avg/stddev):           1.0000/0.00
    execution time (avg/stddev):   48.1460/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=memory --memory-block-size=2G --time=30 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


Running memory speed test with the following options:
  block size: 2097152KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 1 (    0.02 per second)

2048.00 MiB transferred (42.04 MiB/sec)


General statistics:
    total time:                          48.6985s
    total number of events:              1

Latency (ms):
         min:                                    48619.63
         avg:                                    48619.63
         max:                                    48619.63
         95th percentile:                        48662.51
         sum:                                    48619.63

Threads fairness:
    events (avg/stddev):           1.0000/0.00
    execution time (avg/stddev):   48.6196/0.00
```

C. Docker –

```
root@8dbeafbeff3a:/# sysbench --test=memory --memory-block-size=2G --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Running memory speed test with the following options:
  block size: 2097152KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 7 (    0.21 per second)

14336.00 MiB transferred (439.83 MiB/sec)


General statistics:
    total time:                          32.5902s
    total number of events:              7

Latency (ms):
         min:                                  2916.04
         avg:                                  4654.77
         max:                                  5888.88
         95th percentile:                      5918.87
         sum:                                 32583.40

Threads fairness:
    events (avg/stddev):           7.0000/0.00
    execution time (avg/stddev):   32.5834/0.00
```

4) MEMORY TEST WITH CPU – 3 CORES AND RAM – 4G

Test 1 – memory-block-size=1G

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=memory --memory-block-size=1G --time=30 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


Running memory speed test with the following options:
  block size: 1048576KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 79 (    2.60 per second)

80896.00 MiB transferred (2665.25 MiB/sec)


General statistics:
    total time:                          30.3473s
    total number of events:              79

Latency (ms):
         min:                                  327.52
         avg:                                  383.83
         max:                                  589.20
         95th percentile:                      442.73
         sum:                                30322.82

Threads fairness:
    events (avg/stddev):           79.0000/0.00
    execution time (avg/stddev):   30.3228/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=memory --memory-block-size=1G --time=30 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


Running memory speed test with the following options:
  block size: 1048576KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 78 (    2.60 per second)

79872.00 MiB transferred (2658.83 MiB/sec)


General statistics:
    total time:                          30.0359s
    total number of events:              78

Latency (ms):
         min:                                  372.77
         avg:                                  384.87
         max:                                  626.82
         95th percentile:                      397.39
         sum:                                30019.50

Threads fairness:
    events (avg/stddev):           78.0000/0.00
    execution time (avg/stddev):   30.0195/0.00
```

C. Docker –

```
root@cf474904ed48:/# sysbench --test=memory --memory-block-size=1G --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Running memory speed test with the following options:
  block size: 1048576KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 100 (   17.14 per second)

102400.00 MiB transferred (17554.24 MiB/sec)


General statistics:
    total time:                          5.8287s
    total number of events:              100

Latency (ms):
         min:                                   47.61
         avg:                                   58.27
         max:                                   72.02
         95th percentile:                       63.32
         sum:                                 5826.71

Threads fairness:
    events (avg/stddev):           100.0000/0.00
    execution time (avg/stddev):   5.8267/0.00
```

Test 2 – memory-block-size=2G

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=memory --memory-block-size=2G --time=30 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


Running memory speed test with the following options:
  block size: 2097152KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 37 (    1.21 per second)

75776.00 MiB transferred (2475.17 MiB/sec)


General statistics:
    total time:                          30.6100s
    total number of events:              37

Latency (ms):
         min:                                  748.54
         avg:                                  826.87
         max:                                 1416.08
         95th percentile:                     1089.30
         sum:                                30594.05

Threads fairness:
    events (avg/stddev):           37.0000/0.00
    execution time (avg/stddev):   30.5941/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=memory --memory-block-size=2G --time=30 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


Running memory speed test with the following options:
  block size: 2097152KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 37 (    1.21 per second)

75776.00 MiB transferred (2481.32 MiB/sec)


General statistics:
    total time:                          30.5346s
    total number of events:              37

Latency (ms):
         min:                                  752.93
         avg:                                  824.91
         max:                                 1286.76
         95th percentile:                     1032.01
         sum:                                30521.63

Threads fairness:
    events (avg/stddev):           37.0000/0.00
    execution time (avg/stddev):   30.5216/0.00
```

C. Docker –

```
root@cf474904ed48:/# sysbench --test=memory --memory-block-size=2G --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Running memory speed test with the following options:
  block size: 2097152KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 50 (    8.59 per second)

102400.00 MiB transferred (17584.41 MiB/sec)


General statistics:
    total time:                          5.8205s
    total number of events:              50

Latency (ms):
         min:                                    90.22
         avg:                                   116.37
         max:                                   351.69
         95th percentile:                       196.89
         sum:                                  5818.43

Threads fairness:
    events (avg/stddev):           50.0000/0.00
    execution time (avg/stddev):   5.8184/0.00
```

FILEIO TEST

1) FILEIO TEST WITH CPU – 2 CORES AND RAM – 4G

   Test 1 – file-test-mode=seqwr

   A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=fileio --file-test-mode=seqwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing sequential write (creation) test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     2961.38
    fsyncs/s:                     3793.83

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               46.27

General statistics:
    total time:                          30.0164s
    total number of events:              202662

Latency (ms):
         min:                                    0.08
         avg:                                    0.14
         max:                                   19.16
         95th percentile:                        0.17
         sum:                                28942.13

Threads fairness:
    events (avg/stddev):           202662.0000/0.00
    execution time (avg/stddev):   28.9421/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=fileio --file-test-mode=seqwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing sequential write (creation) test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     3157.59
    fsyncs/s:                     4045.64

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               49.34

General statistics:
    total time:                          30.0196s
    total number of events:              216134

Latency (ms):
         min:                                    0.08
         avg:                                    0.13
         max:                                   24.61
         95th percentile:                        0.16
         sum:                                28920.71

Threads fairness:
    events (avg/stddev):           216134.0000/0.00
    execution time (avg/stddev):   28.9207/0.00
```

C. Docker –

```
root@a44dee4906a2:/# sysbench --test=fileio --file-test-mode=seqwr --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing sequential write (creation) test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     11386.63
    fsyncs/s:                     14576.75

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               177.92

General statistics:
    total time:                          30.0079s
    total number of events:              779004

Latency (ms):
         min:                                    0.00
         avg:                                    0.04
         max:                                   32.45
         95th percentile:                        0.05
         sum:                                29736.13

Threads fairness:
    events (avg/stddev):           779004.0000/0.00
    execution time (avg/stddev):   29.7361/0.00
```

Test 2 – file-test-mode=rndwr

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=fileio --file-test-mode=rndwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
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 write test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     2224.25
    fsyncs/s:                     2848.67

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               34.75

General statistics:
    total time:                          30.0295s
    total number of events:              152225

Latency (ms):
         min:                                    0.02
         avg:                                    0.19
         max:                                   20.32
         95th percentile:                        0.54
         sum:                                29219.86

Threads fairness:
    events (avg/stddev):           152225.0000/0.00
    execution time (avg/stddev):   29.2199/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=fileio --file-test-mode=rndwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
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 write test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     2220.98
    fsyncs/s:                     2844.76

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               34.70

General statistics:
    total time:                          30.0287s
    total number of events:              152005

Latency (ms):
         min:                                    0.02
         avg:                                    0.19
         max:                                   32.09
         95th percentile:                        0.54
         sum:                                29243.42

Threads fairness:
    events (avg/stddev):           152005.0000/0.00
    execution time (avg/stddev):   29.2434/0.00
```

C. Docker –

```
root@b4d47e1d0fc8:/# sysbench --test=fileio --file-test-mode=rndwr --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
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 write test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     6827.36
    fsyncs/s:                     8740.65

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               106.68

General statistics:
    total time:                          30.0106s
    total number of events:              467093

Latency (ms):
         min:                                    0.00
         avg:                                    0.06
         max:                                    9.44
         95th percentile:                        0.16
         sum:                                29808.25

Threads fairness:
    events (avg/stddev):           467093.0000/0.00
    execution time (avg/stddev):   29.8082/0.00
```

2) MEMORY TEST WITH CPU – 3 CORES AND RAM – 2G

Test 1 – file-test-mode=seqwr

A. QEMU qcow2 format –

```
root@b4d47e1d0fc8:/# sysbench --test=fileio --file-test-mode=seqwr --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing sequential write (creation) test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     10960.02
    fsyncs/s:                     14031.02

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               171.25

General statistics:
    total time:                          30.0080s
    total number of events:              749830

Latency (ms):
         min:                                    0.00
         avg:                                    0.04
         max:                                   23.91
         95th percentile:                        0.05
         sum:                                29719.98

Threads fairness:
    events (avg/stddev):           749830.0000/0.00
    execution time (avg/stddev):   29.7200/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=fileio --file-test-mode=seqwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing sequential write (creation) test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     2927.38
    fsyncs/s:                     3747.84

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               45.74

General statistics:
    total time:                          30.0236s
    total number of events:              200308

Latency (ms):
         min:                                    0.08
         avg:                                    0.14
         max:                                   34.33
         95th percentile:                        0.17
         sum:                                28997.66

Threads fairness:
    events (avg/stddev):           200308.0000/0.00
    execution time (avg/stddev):   28.9977/0.00
```

C. Docker –

```
root@b4d47e1d0fc8:/# sysbench --test=fileio --file-test-mode=seqwr --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing sequential write (creation) test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     10960.02
    fsyncs/s:                     14031.02

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               171.25

General statistics:
    total time:                          30.0080s
    total number of events:              749830

Latency (ms):
         min:                                    0.00
         avg:                                    0.04
         max:                                   23.91
         95th percentile:                        0.05
         sum:                                29719.98

Threads fairness:
    events (avg/stddev):           749830.0000/0.00
    execution time (avg/stddev):   29.7200/0.00
```

Test 2 – file-test-mode=rndwr

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=fileio --file-test-mode=rndwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
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 write test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     2224.25
    fsyncs/s:                     2848.67

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               34.75

General statistics:
    total time:                          30.0295s
    total number of events:              152225

Latency (ms):
         min:                                    0.02
         avg:                                    0.19
         max:                                   20.32
         95th percentile:                        0.54
         sum:                                29219.86

Threads fairness:
    events (avg/stddev):           152225.0000/0.00
    execution time (avg/stddev):   29.2199/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=fileio --file-test-mode=rndwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
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 write test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     2220.98
    fsyncs/s:                     2844.76

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               34.70

General statistics:
    total time:                          30.0287s
    total number of events:              152005

Latency (ms):
         min:                                    0.02
         avg:                                    0.19
         max:                                   32.09
         95th percentile:                        0.54
         sum:                                29243.42

Threads fairness:
    events (avg/stddev):           152005.0000/0.00
    execution time (avg/stddev):   29.2434/0.00
```

C. Docker –

```
root@b4d47e1d0fc8:/# sysbench --test=fileio --file-test-mode=rndwr --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
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 write test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     6827.36
    fsyncs/s:                     8740.65

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               106.68

General statistics:
    total time:                          30.0106s
    total number of events:              467093

Latency (ms):
         min:                                    0.00
         avg:                                    0.06
         max:                                    9.44
         95th percentile:                        0.16
         sum:                                29808.25

Threads fairness:
    events (avg/stddev):           467093.0000/0.00
    execution time (avg/stddev):   29.8082/0.00
```

3) **MEMORY TEST WITH CPU – 2 CORES AND RAM – 2G**

Test 1 – file-test-mode=seqwr

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=fileio --file-test-mode=seqwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing sequential write (creation) test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     3034.23
    fsyncs/s:                     3886.95

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               47.41

General statistics:
    total time:                          30.0203s
    total number of events:              207674

Latency (ms):
         min:                                    0.08
         avg:                                    0.14
         max:                                   23.40
         95th percentile:                        0.17
         sum:                                28949.30

Threads fairness:
    events (avg/stddev):           207674.0000/0.00
    execution time (avg/stddev):   28.9493/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=fileio --file-test-mode=seqwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing sequential write (creation) test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     3027.91
    fsyncs/s:                     3878.66

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               47.31

General statistics:
    total time:                          30.0249s
    total number of events:              207283

Latency (ms):
         min:                                    0.08
         avg:                                    0.14
         max:                                   39.54
         95th percentile:                        0.17
         sum:                                28935.18

Threads fairness:
    events (avg/stddev):           207283.0000/0.00
    execution time (avg/stddev):   28.9352/0.00
```

C. Docker –

```
root@8dbeafbeff3a:/# sysbench --test=fileio --file-test-mode=seqwr --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing sequential write (creation) test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     10426.82
    fsyncs/s:                     13347.76

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               162.92

General statistics:
    total time:                          30.0080s
    total number of events:              713327

Latency (ms):
         min:                                    0.00
         avg:                                    0.04
         max:                                   32.82
         95th percentile:                        0.05
         sum:                                29711.33

Threads fairness:
    events (avg/stddev):           713327.0000/0.00
    execution time (avg/stddev):   29.7113/0.00
```

Test 2 – file-test-mode=rndwr

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=fileio --file-test-mode=rndwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
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 write test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     2193.20
    fsyncs/s:                     2807.36

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               34.27

General statistics:
    total time:                          30.0446s
    total number of events:              150126

Latency (ms):
         min:                                    0.02
         avg:                                    0.19
         max:                                   12.79
         95th percentile:                        0.56
         sum:                                29225.15

Threads fairness:
    events (avg/stddev):           150126.0000/0.00
    execution time (avg/stddev):   29.2251/0.00
```

B. QEMU raw format –

```
avaidya1@avaidya1:~$ sysbench --test=fileio --file-test-mode=rndwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
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 write test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     2154.95
    fsyncs/s:                     2759.56

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               33.67

General statistics:
    total time:                          30.0670s
    total number of events:              147653

Latency (ms):
         min:                                    0.02
         avg:                                    0.20
         max:                                   24.87
         95th percentile:                        0.56
         sum:                                29274.97

Threads fairness:
    events (avg/stddev):           147653.0000/0.00
    execution time (avg/stddev):   29.2750/0.00
```

C. Docker –

```
root@8dbeafbeff3a:/# sysbench --test=fileio --file-test-mode=rndwr --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
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 write test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     6436.98
    fsyncs/s:                     8239.37

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               100.58

General statistics:
    total time:                          30.0129s
    total number of events:              440369

Latency (ms):
         min:                                    0.00
         avg:                                    0.07
         max:                                   21.47
         95th percentile:                        0.16
         sum:                                29790.75

Threads fairness:
    events (avg/stddev):           440369.0000/0.00
    execution time (avg/stddev):   29.7908/0.00
```

4) **MEMORY TEST WITH CPU – 3 CORES AND RAM – 4G**

Test 1 – file-test-mode=seqwr

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=fileio --file-test-mode=seqwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing sequential write (creation) test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     2780.64
    fsyncs/s:                     3561.95

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               43.45

General statistics:
    total time:                          30.0232s
    total number of events:              190334

Latency (ms):
         min:                                    0.08
         avg:                                    0.15
         max:                                   16.08
         95th percentile:                        0.18
         sum:                                29042.27

Threads fairness:
    events (avg/stddev):           190334.0000/0.00
    execution time (avg/stddev):   29.0423/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=fileio --file-test-mode=seqwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing sequential write (creation) test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     2761.82
    fsyncs/s:                     3538.29

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               43.15

General statistics:
    total time:                          30.0223s
    total number of events:              189038

Latency (ms):
         min:                                    0.08
         avg:                                    0.15
         max:                                   12.29
         95th percentile:                        0.18
         sum:                                29002.37

Threads fairness:
    events (avg/stddev):           189038.0000/0.00
    execution time (avg/stddev):   29.0024/0.00
```

C. Docker –

```
root@cf474904ed48:/# sysbench --test=fileio --file-test-mode=seqwr --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing sequential write (creation) test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     9824.24
    fsyncs/s:                     12577.73

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               153.50

General statistics:
    total time:                          30.0063s
    total number of events:              672097

Latency (ms):
         min:                                    0.00
         avg:                                    0.04
         max:                                   25.07
         95th percentile:                        0.05
         sum:                                29734.13

Threads fairness:
    events (avg/stddev):           672097.0000/0.00
    execution time (avg/stddev):   29.7341/0.00
```

Test 2 – file-test-mode=rndwr

A. QEMU qcow2 format –

```
avaidya1@avaidya1:~$ sysbench --test=fileio --file-test-mode=rndwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
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 write test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     2108.16
    fsyncs/s:                     2700.17

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               32.94

General statistics:
    total time:                          30.0670s
    total number of events:              144476

Latency (ms):
         min:                                    0.02
         avg:                                    0.20
         max:                                   39.50
         95th percentile:                        0.59
         sum:                                29237.96

Threads fairness:
    events (avg/stddev):           144476.0000/0.00
    execution time (avg/stddev):   29.2380/0.00
```

B. QEMU raw format –

```
avaidya2@avaidya2:~$ sysbench --test=fileio --file-test-mode=rndwr --time=30 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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
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 write test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     2073.40
    fsyncs/s:                     2655.35

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               32.40

General statistics:
    total time:                          30.0435s
    total number of events:              141958

Latency (ms):
         min:                                    0.02
         avg:                                    0.21
         max:                                   26.39
         95th percentile:                        0.59
         sum:                                29248.53

Threads fairness:
    events (avg/stddev):           141958.0000/0.00
    execution time (avg/stddev):   29.2485/0.00
```

C. Docker –

```
root@cf474904ed48:/# sysbench --test=fileio --file-test-mode=rndwr --time=30 run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

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


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
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 write test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      0.00
    writes/s:                     6367.03
    fsyncs/s:                     8150.00

Throughput:
    read, MiB/s:                  0.00
    written, MiB/s:               99.48

General statistics:
    total time:                          30.0128s
    total number of events:              435586

Latency (ms):
         min:                                    0.00
         avg:                                    0.07
         max:                                   50.20
         95th percentile:                        0.16
         sum:                                29800.71

Threads fairness:
    events (avg/stddev):           435586.0000/0.00
    execution time (avg/stddev):   29.8007/0.00
```

6. EXPERIMENT RESULTS AND ANALYSIS

CPU Test - QEMU VM qcow2 vs Qemu VM raw vs. Docker Container

To conduct the CPU Test, I've considered the following two primary parameters:

1. --cpu-max-prime Determines the maximum number up to which the tested numbers are checked for primality.

2. --time Sets the maximum allowable time for the process to complete.

Test 1 – cpu-max-prime=2000

```
root@b4d47e1d0fc8:/# sysbench --test=cpu --cpu-max-prime=2000 --time=30 run
```

Technology	CPU	RAM	min	max	avg	Std(events/sec)	No of events
Qcow2	**2**	**4**	**0.21**	**23.31**	**0.22**	**4472.55**	**134196**
Raw	**2**	**4**	**0.21**	**15.63**	**0.22**	**4495.27**	**134880**
docker	**2**	**4**	**0.2**	**21.57**	**0.2**	**42759.31**	**1282917**
Qcow2	**3**	**2**	**0.21**	**18.32**	**0.22**	**4463.26**	**133517**
Raw	**3**	**2**	**0.21**	**31.19**	**0.22**	**4467.20**	**134044**
Docker	**3**	**2**	**0.02**	**5.268**	**0.02**	**42981.45**	**1289511**
Qcow2	**2**	**2**	**0.21**	**32.91**	**0.22**	**4472.61**	**134201**
Raw	**2**	**2**	**0.21**	**35.47**	**0.22**	**4469.76**	**134113**
Docker	**2**	**2**	**0.2**	**17.59**	**0.2**	**42928.10**	**1288141**
Qcow2	**3**	**4**	**0.21**	**14.89**	**0.22**	**4530.87**	**135459**
Raw	**3**	**4**	**0.21**	**18.73**	**0.22**	**4455.50**	**133709**
Docker	**3**	**4**	**0.02**	**18.96**	**0.02**	**42744.20**	**1276016**

Test 2 – cpu-max-prime=3000

```
root@b4d47e1d0fc8:/# sysbench --test=cpu --cpu-max-prime=3000 --time=30 run
```

Technology	CPU	RAM	min	max	avg	Std(events/sec)	No of events
Qcow2	**2**	**4**	**0.35**	**28.45**	**0.36**	**2725.47**	**80273**
Raw	**2**	**4**	**0.35**	**9.02**	**0.36**	**2727.96**	**81853**
docker	**2**	**4**	**0.04**	**17.4**	**0.04**	**24671.45**	**740218**
Qcow2	**3**	**2**	**0.35**	**17.92**	**0.37**	**2705.03**	**133517**
Raw	**3**	**2**	**0.35**	**18.2**	**0.36**	**2720.03**	**81623**
Docker	**3**	**2**	**0.04**	**7.12**	**0.04**	**25042.10**	**751305**
Qcow2	**2**	**2**	**0.35**	**30.02**	**0.36**	**2723.40**	**81284**
Raw	**2**	**2**	**0.35**	**9.4**	**0.36**	**2721.80**	**81666**
Docker	**2**	**2**	**0.04**	**46.2**	**0.04**	**24819.51**	**744703**
Qcow2	**3**	**4**	**0.35**	**25.95**	**0.36**	**2741.94**	**82273**
Raw	**3**	**4**	**0.35**	**18.4**	**0.36**	**2721.89**	**81462**
Docker	**3**	**4**	**0.04**	**18.16**	**0.04**	**25066.33**	**752019**

MEMORY Test - QEMU VM qcow2 vs Qemu VM raw vs. Docker Container

To conduct the Memory Test, I've considered the following three primary parameters:

1. --memory-block-size the size of memory blocks utilized for the benchmarking process.

2. --time Sets the maximum allowable time for the process to complete.

Test 1 – memory-block-size=1G

```
root@b4d47e1d0fc8:/# sysbench --test=memory --memory-block-size=1G --time=30 run
```

Technology	CPU	RAM	min	max	avg	Std(events/sec)	No of events
Qcow2	**2**	**4**	**331.97**	**525.21**	**375.86**	**2.65**	**80**
Raw	**2**	**4**	**327**	**538.44**	**378.63**	**2.64**	**80**

docker	**2**	**4**	**45**	**277.8**	**62.8**	**16.4**	**100**
Qcow2	**3**	**2**	**327.30**	**498.68**	**373.23**	**2.68**	**81**
Raw	**3**	**2**	**331**	**491.2**	**369.4**	**2.7**	**81.8**
Docker	**3**	**2**	**44.62**	**198.2**	**60.2**	**17.31**	**100**
Qcow2	**2**	**2**	**337**	**613.2**	**378.2**	**2.6**	**80**
Raw	**2**	**2**	**346**	**469.8**	**376.6**	**2.65**	**80**
Docker	**2**	**2**	**44.67**	**126.4**	**56.2**	**17.6**	**100**
Qcow2	**3**	**4**	**345**	**464.6**	**375.4**	**2.65**	**80**
Raw	**3**	**4**	**345.2**	**588.2**	**374.2**	**2.6**	**80.6**
Docker	**3**	**4**	**44.6**	**50.2**	**58.6**	**20**	**100**

Test 2 – memory-block-size=2G

```
root@b4d47e1d0fc8:/# sysbench --test=memory --memory-block-size=2G --time=30 run
```

Technology	**CPU**	**RAM**	**min**	**max**	**avg**	**Std(events/sec)**	**No of events**
Qcow2	**2**	**4**	**748.39**	**1676.49**	**836.88**	**1.19**	**36**
Raw	**2**	**4**	**747.07**	**1541.53**	**816.58**	**1.22**	**36**
docker	**2**	**4**	**89.18**	**254.6**	**114.4**	**8.96**	**50**
Qcow2	**3**	**2**	**46896.28**	**46896.28**	**46896.28**	**0.02**	**1**
Raw	**3**	**2**	**46468.28**	**46468.28**	**46468.28**	**0.02**	**1**
Docker	**3**	**2**	**2727.8**	**6274**	**4070.6**	**0.246**	**8**
Qcow2	**2**	**2**	**48430.2**	**48430.2**	**48430.2**	**0.02**	**1**
Raw	**2**	**2**	**48381.2**	**48381.2**	**48381.2**	**0.02**	**1**
Docker	**2**	**2**	**2363**	**6398**	**3623.8**	**0.30**	**9**

Qcow2	**3**	**4**	**744.6**	**1440.4**	**832.6**	**1.2**	**36**
Raw	**3**	**4**	**732.6**	**1290.8**	**809.4**	**1.23**	**38**
Docker	**3**	**4**	**89**	**257.8**	**100.4**	**9.4**	**50**

FILEIO Test - QEMU VM qcow2 vs Qemu VM raw vs. Docker Container

For conducting the FILE-IO Test, I have considered the following three primary parameters:

1. --threads Specifies the number of threads to be utilized for the test.

2. --file-test-mode dictates the mode of the file test, with five available modes of file I/O:

a. rndrd b. rndrw c. rndwr d. seqrd e. seqrewr f. Seqwr

Test 1 – file-test-mode=seqwr

```
root@b4d47e1d0fc8:/# sysbench --test=fileio --file-test-mode=seqwr --time=30 run
```

Technology	**CPU**	**RAM**	**min**	**max**	**avg**	**Std(events/sec)**	**No of events**
Qcow2	**2**	**4**	**0.08**	**17.96**	**0.14**	**2964.70**	**2895276**
Raw	**2**	**4**	**0.08**	**17.53**	**0.13**	**3170.41**	**216134**
docker	**2**	**4**	**0.00**	**26.6**	**0.04**	**11424**	**781604**
Qcow2	**3**	**2**	**0.08**	**25.15**	**0.14**	**2951.44**	**202277**
Raw	**3**	**2**	**0.08**	**25.29**	**0.14**	**2940**	**201226.6**
Docker	**3**	**2**	**0.00**	**23.8**	**0.04**	**10679.6**	**730661.4**
Qcow2	**2**	**2**	**0.08**	**26.8**	**0.14**	**3083.6**	**211052.2**
Raw	**2**	**2**	**0.08**	**18**	**0.14**	**3054.8**	**209102.8**
Docker	**2**	**2**	**0.00**	**42.38**	**0.04**	**10445.2**	**714645**
Qcow2	**3**	**4**	**0.08**	**15.8**	**0.15**	**2794.2**	**191267.8**
Raw	**3**	**4**	**0.08**	**17.8**	**0.15**	**2773.8**	**189979**
Docker	**3**	**4**	**0.00**	**21.6**	**0.04**	**11000**	**752668.6**

Test 2 – file-test-mode=rndwr

```
root@b4d47e1d0fc8:/# sysbench --test=fileio --file-test-mode=rndwr --time=30 run
```

Technology	CPU	RAM	min	max	avg	Std(events/sec)	No of events
Qcow2	**2**	**4**	**0.01**	**9.01**	**0.14**	**2992.44**	**68334**
Raw	**2**	**4**	**0.02**	**19.2**	**0.19**	**2261.47**	**156864**
docker	**2**	**4**	**0.00**	**26.2**	**0.07**	**6481.8**	**443500.8**
Qcow2	**3**	**2**	**0.02**	**16.59**	**0.19**	**2211.09**	**152225**
Raw	**3**	**2**	**0.02**	**15.37**	**0.19**	**2229.2**	**152610.4**
Docker	**3**	**2**	**0.00**	**20.2**	**0.07**	**6591.8**	**451846.6**
Qcow2	**2**	**2**	**0.02**	**22.8**	**0.20**	**2175.2**	**148956.6**
Raw	**2**	**2**	**0.02**	**20.6**	**0.20**	**2172.4**	**148727.2**
Docker	**2**	**2**	**0.00**	**43**	**0.07**	**6445.4**	**440986.6**
Qcow2	**3**	**4**	**0.02**	**21.4**	**0.20**	**2120.6**	**145215.6**
Raw	**3**	**4**	**0.02**	**18**	**0.20**	**2085**	**142760.8**
Docker	**3**	**4**	**0.00**	**31.2**	**0.07**	**6570.2**	**449535**

7. SYSBENCH FINDINGS AND CONCLUSION.

The analysis presented above utilizes two separate test cases for each CPU, File-IO, and Memory test, encompassing diverse CPU and RAM configurations. Additionally, each test is repeated five times to evaluate the reliability and consistency of results across various scenarios. Based on the sysbench test results, several conclusions can be inferred.

CPU –

1. Across different configurations, Docker consistently demonstrates strong CPU performance.
2. Docker outperforms in terms of event processing efficiency, as evidenced by a higher number of events per second.
3. QEMU qcow2 and raw exhibit similar trends, with qcow2 tends to show slightly higher CPU usage.
4. Increased CPU count generally correlates with higher CPU usage, while the impact of memory configurations on performance varies.
5. In terms of CPU usage and event processing efficiency, Docker outshines QEMU with qcow2 or raw vm. Raw vm, in turn, displays greater stability compared to qcow2.

Memory –

1. Observations for memory test mode are like the above ones, where Docker consistently demonstrates strong CPU performance across different configurations.
2. QEMU and qcow2 exhibit similar trends in raw storage.
3. Thus, docker performs well than qcow2 and raw QEMU images with providing an efficient environment for given memory.

FileIO –

1. Docker consistently demonstrates lower CPU usage and tends to exhibit a lower standard deviation, making it a more stable option in terms of performance consistency.
2. Qemu qcow2 and raw show similar trends in all the test scenarios for fileio.

8. SHELL SCRIPT

The Following shell script performs the tests for CPU, Memory, and FileIO.

1. Shell script for CPU

```
#! /bin/bash

echo "CPU Test!"

PRIMES_UPTO=("2000" "3000")
MAX_TIME=30
TEST_RUNS=5
TEST_CASES=2

for ((i=0; i<$TEST_CASES;i++))
do
        echo "$((i+1))st Test Case"
        for (( j=1; j <=$TEST_RUNS; j++ ))
        do
                echo "Running ${j}st run of Test Case $((i+1))"
                sysbench --test=cpu --cpu-max-prime=${PRIMES_UPTO[$i]} --
time=${MAX_TIME} run
                echo "Completed ${j}st run of Test Case $((i+1))"
        done
        echo "Completed $((i+1))st Test Case"
done
```

2. Shell script for Memory –

```
#! /bin/bash

echo "Memory Test!"

MEMORY_BLOCK_SIZE=("1G" "2G")
```

```
MAX_TIME=30
TEST_RUNS=5
TEST_CASES=2

for ((i=0; i<$TEST_CASES;i++))
do
        echo "$((i+1))st Test Case"
        for (( j=1; j <=$TEST_RUNS; j++ ))
        do
                echo "Running ${j}st run of Test Case $((i+1))"
                sysbench --test=memory --memory-block-
size=${MEMORY_BLOCK_SIZE[$i]} --time=${MAX_TIME} run
                echo "Completed ${j}st run of Test Case $((i+1))"
        done
        echo "Completed $((i+1))st Test Case"
done
```

3. Shell script for Fileio –

```
#! /bin/bash

echo "Fileio Test"

TEST_MODE=("rndwr" "seqwr")
MAX_TIME=30
TEST_RUNS=5
TEST_CASES=2

for ((i=0; i<$TEST_CASES; i++))
do
        echo "$((i+1))st Test Case"
        for (( j=1; j <=$TEST_RUNS; j++ ))
```

```
        do
                echo "Running ${j}st run of Test Case $((i+1))"
      sysbench --test=fileio --file-test-mode=${TEST_MODE[i]} --time=${MAX_TIME}
cleanup
                sysbench --test=fileio --file-test-mode=${TEST_MODE[i]} --
time=${MAX_TIME} prepare
                sysbench --test=fileio --file-test-mode=${TEST_MODE[i]} --
time=${MAX_TIME} run
                echo "Completed ${j}st run of Test Case $((i+1))"
        done
        echo "Completed $((i+1))st Test Case"
done
```

9. AUTOMATION SCRIPTS

A. Docker File –

```
FROM avaidya2/ubuntu-sysbench:v1

COPY docker_script.sh /docker_script.sh
COPY cputest.sh /cputest.sh
COPY fileiotest.sh /fileiotest.sh
COPY memorytest.sh /memorytest.sh

RUN chmod +x docker_script.sh
RUN chmod +x cputest.sh
RUN chmod +x fileiotest.sh
RUN chmod +x memorytest.sh

ENTRYPOINT bash docker_script.sh
```

B. Vagrant File –

```
# All Vagrant configuration is done below. The "2" in Vagrant.configure
# configures the configuration version
Vagrant.configure("2") do |config|

  # Every Vagrant development environment requires a box.
  config.vm.box = "bento/ubuntu-20.04"

  # Provider Settings
  config.vm.provider "virtualbox" do |vb|
    vb.memory = 2048
    vb.cpus = 2
  end

  # Folder Settings
  config.vm.synced_folder ".", "/vagrant_data"

  # Provision Settings
  config.vm.provision "shell", path: "vagrant_script.sh"

end
```

RESOURCES –

Docker Image - https://hub.docker.com/repository/docker/avaidya2/ubuntu-sysbench/general

Github Link - https://github.com/AvaniVaidya/cloud-computing.git