/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\* The Instructions for the AppleS Artifact Evaluation by Online Proxy \*\*\*/ /\*\*\* created by Ning Li \*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\* ning.li@uta.edu \*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\* 02/13/2022 \*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

Dear Reviewers,

I've established an online proxy with a public IP (107.172.196.122) that can help you easily access our AppleS testbed. You can have root privilege for all the things (e.g., OS and database) by the password "Eurosys2022-AppleS". Please refer to the file Detailed\_Instructions\_AppleS.txt or Detailed\_Instructions\_AppleS.docx that can be downloaded from https://github.com/NingBellWind/AppleS\_Artifact/. You can execute the commands one by one from the document to finish the two experiments (the former has 3 steps while the latter has 5 steps) by using all the resources of the testbed that has the latest source code of AppleS in /root/apples and sets up all the required dependent software and hardware equipments (i.e., the testbed is all set and ready for your testing). In addition, the document includes all the configurations and examples for each step of the two experiments and tips for how to get their results. Please let us know if you have any questions or encounter any problems when you access our testbed.

[ E1]:\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

Experiment (E1): [MySQL 8.0.23+CentOS 8.3+Kernel 5.10.10] [30 human-minutes + 1 compute-hour]:

E1 aims to assess AppleS's capability to improve throughput for MySQL 8.0.23 with significantly enhanced user-level fairness and a low latency variability.

[How to]

E1 consists of 3 steps, i.e., baseline measurement, P optimization, and measurement under the AppleS control. Step 1: conduct 3 runs of TPC-C workload with 256 concurrent connections accessing MySQL 8.0.23 running with disabled AppleS, which only tracks user I/O statistics and records them in log files. Step 2: enable AppleS to conduct P optimization by setting "opt" at 1 and "disable" at 0. The P optimization is only required to run once for a specific database system and lasts about 6 minutes. Step 3: almost the same with Step 1 except for enabling AppleS by setting "disable" at 0. And then, one can compare between the measures obtained under the baseline and the AppleS-controlled case. The expected outcome would be over 20% throughput improvement, over 10 times user I/O fairness enhancement, and over 5 times lower latency variability.

Note: Before each run, please make sure that no mysqld process is still running. This means that you need to manually stop mysqld after the previous run by "pkill mysql".

>>Step1:---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Description:

1) Data directory (MySQL 8.0.23): /sdata1/mysql //Store all the MySQL TPCC database files;

2) Source directory: /root/apples //Store all the latest required source code, compiled libraries, and executable shells.

3) Running directory: /usr/sbin //The directory where mysqld is in and you need to put AppleS configurations (i.e., apples\_configuration.txt).

4) Working directory: /root/mysql\_baseline1 //The directory where AppleS records experimental results and optimization settings for the run.

a) Open a terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /usr/sbin

[root@computing3 ~]# yes | cp /root/apples\_conf/E1\_step1/apples\_configuration.txt /usr/sbin

[root@computing3 sbin]# echo 1 > als\_ctl.txt

[root@computing3 sbin]# LD\_PRELOAD=/root/apples/sys\_xpslo\_mon.so /usr/sbin/mysqld --basedir=/usr --datadir=/sdata1/mysql --plugin-dir=/usr/lib64/mysql/plugin --user=mysql --log-error=/var/log/mysqld.log --pid-file=/var/run/mysqld/mysqld.pid --socket=/sdata1/mysql/mysqld.sock

b) Open another terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/mysql/tpcc-mysql-master

[root@computing3 tpcc-mysql-master]# ./tpcc\_start -h127.0.0.1 -P3306 -dtpcc1000 -uroot -p Eurosys2022-AppleS -w1000 -c256 -r60 -i10 -l300

Finally, please execute "pkill mysql" on the second terminal to stop mysqld after this run;

Tips: 1) You can find a example of experiemental results for E1\_step1 in /root/apples\_conf/E1\_step1 (i.e., One can find CV of request latency and user I/O unfairness of the 300-second run from the last two columns of the 3000th line in bslo\_cmd\_log1.txt (e.g., 00014233 00015330 (i.e., CV of request latency = 14.233 and user-level I/O unfairness = 153.30%), the smaller the better) and the throughput measure at the last line of tpcc.txt (e.g., 3299.200 TpmC)).

2) You can find the wroking directory for this case in /root/mysql\_baseline.

>>Step2:---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

1) Data directory (MySQL 8.0.23): /sdata1/mysql //Store all the MySQL TPCC database files;

2) Source directory: /root/apples //Store all the latest required source code, compiled libraries, and executable shells.

3) Running directory: /usr/sbin //The directory where mysqld is in and you need to put AppleS configurations (i.e., apples\_configuration.txt).

4) Working directory: /root/mysql\_opt //The directory where AppleS records experimental results and optimization settings for the run.

a) Open a terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /usr/sbin

[root@computing3 ~]# yes | cp /root/apples\_conf/E1\_step2/apples\_configuration.txt /usr/sbin

[root@computing3 sbin]# echo 1 > als\_ctl.txt

[root@computing3 sbin]# LD\_PRELOAD=/root/apples/sys\_xpslo\_mon.so /usr/sbin/mysqld --basedir=/usr --datadir=/sdata1/mysql --plugin-dir=/usr/lib64/mysql/plugin --user=mysql --log-error=/var/log/mysqld.log --pid-file=/var/run/mysqld/mysqld.pid --socket=/sdata1/mysql/mysqld.sock

b) Open another terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/mysql/tpcc-mysql-master

[root@computing3 tpcc-mysql-master]# ./tpcc\_start -h127.0.0.1 -P3306 -dtpcc1000 -uroot -p Eurosys2022-AppleS -w1000 -c256 -r60 -i10 -l600

Finally, please execute "pkill mysql" on the second terminal to stop mysqld after this run;

Tips: 1) You can find a example of optimization results for E1\_step2 in /root/apples\_conf/E1\_step2 (i.e., t the last line of /root/mysql\_opt/bslo\_opt\_log1.txt (e.g., ,8,1,8,128,32, 8797.91, 5090.62, 6095.79)).

2) You can find the wroking directory for this case in /root/mysql\_opt.

>>Step3:---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

1) Data directory (MySQL 8.0.23): /sdata1/mysql //Store all the MySQL TPCC database files;

2) Source directory: /root/apples //Store all the latest required source code, compiled libraries, and executable shells.

3) Running directory: /usr/sbin //The directory where mysqld is in and you need to put AppleS configurations (i.e., apples\_configuration.txt).

4) Working directory: /root/mysql\_opt //The directory where AppleS records experimental results and optimization settings for the run.

a) Open a terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /usr/sbin

[root@computing3 ~]# yes | cp /root/apples\_conf/E1\_step3/apples\_configuration.txt /usr/sbin

[root@computing3 sbin]# echo 1 > als\_ctl.txt

[root@computing3 sbin]# LD\_PRELOAD=/root/apples/sys\_xpslo\_mon.so /usr/sbin/mysqld --basedir=/usr --datadir=/sdata1/mysql --plugin-dir=/usr/lib64/mysql/plugin --user=mysql --log-error=/var/log/mysqld.log --pid-file=/var/run/mysqld/mysqld.pid --socket=/sdata1/mysql/mysqld.sock

b) Open another terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/mysql/tpcc-mysql-master

[root@computing3 tpcc-mysql-master]# ./tpcc\_start -h127.0.0.1 -P3306 -dtpcc1000 -uroot -p Eurosys2022-AppleS -w1000 -c256 -r60 -i10 -l300

Finally, please execute "pkill mysql" on the second terminal to stop mysqld after this run;

Tips: 1) You can find a example of experiemental results for E1\_step3 in /root/apples\_conf/E1\_step3 (i.e., One can find CV of request latency and user I/O unfairness of the 300-second run from the last two columns of the 3000th line in bslo\_cmd\_log1.txt (e.g., 00001135 00000266 (i.e., CV of request latency = 1.135 and user-level I/O unfairness = 2.66%), the smaller the better) and the throughput measure at the last line of tpcc.txt (e.g., 5106.800 TpmC)).

2) You can find the wroking directory for this case in /root/mysql\_opt.

[ E2]:\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

Experiment (E2): [MongoDB 4.4.3+Cgroup (V2)] [30 human-minutes + 1 compute-hour]:

E2 aims to verify AppleS's capability to cooperate with Cgroup (V2) for MongoDB 4.4.3 to gain further improvement on user-level fairness and latency variability.

[How to]

E2 consists of 5 steps, i.e., baseline measurement, P optimization, measure under the AppleS control, measure under the control of Cgroup (V2), and measure under the control of both AppleS and Cgroup. Step 1: conduct 3 runs of YCSB workload with 512 concurrent connections accessing MongoDB 4.4.3 running with disabled AppleS. Step 2: the same with the Step 2 of E1. Step 3: almost the same with Step 1 except for enabling AppleS by setting "disable" at 0. Step 4: almost the same with Step 1 except for setting /cgroup2/cg1/cgroup.procs as the PID of MongoDB to enable the Cgroup (V2) control. Step 5: almost the same with Step 4 except for enabling AppleS. And then, one can compare among the measures obtained under the baseline, the AppleS-controlled case, the Cgroup-controlled case, and the dual-controlled case. The expected outcome is that AppleS can work with Cgroup to achieve a high user-level fairness (over 30 times improvement than the baseline) and low latency variability (over 2 times improvement than the baseline).

Note: Before each run, please make sure that no mongod or mongodx process is still running. This means that you need to manually stop mongod or mongodx after the previous run by "CTRL+C".

>>Step1:---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

1) Data directory (MongoDB 4.4.3): /sdata4 //Store all the MongoDB YCSB database files;

2) Source directory: /root/apples //Store all the latest required source code, compiled libraries, and executable shells.

3) Running directory: /root/mongo-master //The directory where mongod is in and you need to put AppleS configurations (i.e., apples\_configuration.txt).

4) Working directory: /root/mongo\_baseline1 //The directory where AppleS records experimental results and optimization settings for the run.

a) Open a terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/mongo-master

[root@computing3 ~]# yes | cp /root/apples\_conf/E2\_step1/apples\_configuration.txt /root/mongo-master

[root@computing3 mongo-master]# echo 1 > als\_ctl.txt

[root@computing3 mongo-master]# LD\_PRELOAD=/root/apples/sys\_xpslo\_mon.so ./mongodx --dbpath /sdata4 --bind\_ip\_all

b) Open another terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/YCSB-master

[root@computing3 YCSB-master]# ./bin/ycsb run mongodb -s -threads 512 -p "mongodb.maxconnections=1026" -p maxexecutiontime=360 -P workloads/workloadax > runSync.txt

Finally, please type CTRL+C on the first terminal to stop MongoDB after this run;

Tips: 1) You can find a example of optimization results for E2\_step1 in /root/apples\_conf/E2\_step1 ((i.e., One can find CV of request latency and user I/O unfairness of the 300-second run from the last two columns of the 3000th line in bslo\_cmd\_log1.txt (e.g., 00004627 00001300 (i.e., CV of request latency = 4.627 and user-level I/O unfairness = 13.00%), the smaller the better) and the throughput measure at the 3rd line of runSync.txt (e.g., [OVERALL], Throughput(ops/sec), 19724.22704685085)).

2) You can find the wroking directory for this case in /root/mongodb\_baseline.

>>Step2:---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

1) Data directory (MongoDB 4.4.3): /sdata4 //Store all the MongoDB YCSB database files;

2) Source directory: /root/apples //Store all the latest required source code, compiled libraries, and executable shells.

3) Running directory: /root/mongo-master //The directory where mongod is in and you need to put AppleS configurations (i.e., apples\_configuration.txt).

4) Working directory: /root/mongo\_opt //The directory where AppleS records experimental results and optimization settings for the run.

a) Open a terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/mongo-master

[root@computing3 ~]# yes | cp /root/apples\_conf/E2\_step2/apples\_configuration.txt /root/mongo-master

[root@computing3 mongo-master]# echo 1 > als\_ctl.txt

[root@computing3 mongo-master]# LD\_PRELOAD=/root/apples/sys\_xpslo\_mon.so ./mongodx --dbpath /sdata4 --bind\_ip\_all

b) Open another terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/YCSB-master

[root@computing3 YCSB-master]# ./bin/ycsb run mongodb -s -threads 512 -p "mongodb.maxconnections=1026" -p maxexecutiontime=400 -P workloads/workloadax > runSync.txt

Finally, please type CTRL+C on the first terminal to stop MongoDB after this run;

Tips: 1) You can find a example of optimization results for E2\_step2 in /root/apples\_conf/E2\_step2 (i.e., t the last line of /root/mongodb\_opt/bslo\_opt\_log1.txt (e.g., ,22,1,8,128,32, 8797.91, 5090.62, 6095.79)).

2) You can find the wroking directory for this case in /root/mongodb\_opt.

>>Step3:---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

1) Data directory (MongoDB 4.4.3): /sdata4 //Store all the MongoDB YCSB database files;

2) Source directory: /root/apples //Store all the latest required source code, compiled libraries, and executable shells.

3) Running directory: /root/mongo-master //The directory where mongod is in and you need to put AppleS configurations (i.e., apples\_configuration.txt).

4) Working directory: /root/mongo\_opt //The directory where AppleS records experimental results and optimization settings for the run.

a) Open a terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/mongo-master

[root@computing3 ~]# yes | cp /root/apples\_conf/E2\_step3/apples\_configuration.txt /root/mongo-master

[root@computing3 mongo-master]# echo 1 > als\_ctl.txt

[root@computing3 mongo-master]# LD\_PRELOAD=/root/apples/sys\_xpslo\_mon.so ./mongodx --dbpath /sdata4 --bind\_ip\_all

b) Open another terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/YCSB-master

[root@computing3 YCSB-master]# ./bin/ycsb run mongodb -s -threads 512 -p "mongodb.maxconnections=1026" -p maxexecutiontime=360 -P workloads/workloadax > runSync.txt

Finally, please type CTRL+C on the first terminal to stop MongoDB after this run;

Tips: 1) You can find a example of optimization results for E2\_step3 in /root/apples\_conf/E2\_step3 ((i.e., One can find CV of request latency and user I/O unfairness of the 300-second run from the last two columns of the 3000th line in bslo\_cmd\_log1.txt (e.g., 00001295 00000009 (i.e., CV of request latency = 1.295 and user-level I/O unfairness = 0.09%) and the throughput measure at the 3rd line of runSync.txt (e.g., [OVERALL], Throughput(ops/sec), 19919.9865965849)).

>>Step4:---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

1) Data directory (MongoDB 4.4.3): /sdata4 //Store all the MongoDB YCSB database files;

2) Source directory: /root/apples //Store all the latest required source code, compiled libraries, and executable shells.

3) Running directory: /root/mongo-master //The directory where mongod is in and you need to put AppleS configurations (i.e., apples\_configuration.txt).

4) Working directory: /root/mongodb\_cgroup //The directory where AppleS records experimental results and optimization settings for the run.

a) Open a terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/mongo-master

[root@computing3 ~]# yes | cp /root/apples\_conf/E2\_step4/apples\_configuration.txt /root/mongo-master

[root@computing3 mongo-master]# echo 1 > als\_ctl.txt

[root@computing3 mongo-master]# LD\_PRELOAD=/root/apples/sys\_xpslo\_mon.so ./mongodx --dbpath /sdata4 --bind\_ip\_all

b) Open another terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/apples

[root@computing apples] ./set\_cgroup\_mongodb.sh

[root@computing3 ~]# cd /root/YCSB-master

[root@computing3 YCSB-master]# ./bin/ycsb run mongodb -s -threads 512 -p "mongodb.maxconnections=1026" -p maxexecutiontime=360 -P workloads/workloadax > runSync.txt

Finally, please type CTRL+C on the first terminal to stop MongoDB after this run;

Tips: 1) You can find a example of optimization results for E2\_step4 in /root/apples\_conf/E2\_step4 ((i.e., One can find CV of request latency and user I/O unfairness of the 300-second run from the last two columns of the 3000th line in bslo\_cmd\_log1.txt (e.g., 00003669 00000986 (i.e., CV of request latency = 3.669 and user-level I/O unfairness = 9.86%) and the throughput measure at the 3rd line of runSync.txt (e.g., [OVERALL], Throughput(ops/sec), 23755.650808682432)).

2) You can find the wroking directory for this case in /root/mongodb\_cgroup.

>>Step5:---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

1) Data directory (MongoDB 4.4.3): /sdata4 //Store all the MongoDB YCSB database files;

2) Source directory: /root/apples //Store all the latest required source code, compiled libraries, and executable shells.

3) Running directory: /root/mongo-master //The directory where mongod is in and you need to put AppleS configurations (i.e., apples\_configuration.txt).

4) Working directory: /root/mongodb\_opt //The directory where AppleS records experimental results and optimization settings for the run.

//Note: we share the same working directory of Step 3 with Step 5 to use the optimization settings (i.e., /root/mongodb\_opt/bslo\_opt\_log1.txt), if one specifies another directory (e.g., /root/mongodb\_cgroup\_opt) for step5, please copy /root/mongodb\_opt/bslo\_opt\_log1.txt to the directory. Otherwise, you need to rerun Step 2 with the new directory as its working directory to generate a bslo\_opt\_log1.txt with optimization settings there.)

a) Open a terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/mongo-master

[root@computing3 ~]# yes | cp /root/apples\_conf/E2\_step5/apples\_configuration.txt /root/mongo-master

[root@computing3 mongo-master]# echo 1 > als\_ctl.txt

[root@computing3 mongo-master]# LD\_PRELOAD=/root/apples/sys\_xpslo\_mon.so ./mongodx --dbpath /sdata4 --bind\_ip\_all

b) Open another terminal: ssh root@107.172.196.122 and then run the command "ssh -oPort=6000 root@107.172.196.122" (User: root Password: Eurosys2022-AppleS)

[root@computing3 ~]# cd /root/apples

[root@computing apples] ./set\_cgroup\_mongodb.sh

[root@computing3 ~]# cd /root/YCSB-master

[root@computing3 YCSB-master]# ./bin/ycsb run mongodb -s -threads 512 -p "mongodb.maxconnections=1026" -p maxexecutiontime=360 -P workloads/workloadax > runSync.txt

Finally, please type CTRL+C on the first terminal to stop MongoDB after this run;

Tips: 1) You can find a example of optimization results for E2\_step5 in /root/apples\_conf/E2\_step5 ((i.e., One can find CV of request latency and user I/O unfairness of the 300-second run from the last two columns of the 3000th line in bslo\_cmd\_log1.txt (e.g., 00001030 00000004 (i.e., CV of request latency = 1.030 and user-level I/O unfairness = 0.04%) and the throughput measure at the 3rd line of runSync.txt (e.g., [OVERALL], Throughput(ops/sec), 23208.182475463378)).

2) You can find the wroking directory for this case in /root/mongodb\_opt.