BÁO CÁO LAB3 LƯU TRỮ XỬ LÝ DỮ LIỆU LỚN

Nhóm: Squad Game

Thành viên: Lại Ngọc Thăng Long 20183581

Nguyễn Đình Dũng 20183506

Nguyễn Thành Long 20183586

Nguyễn Khương Duy 20183513

BÀI LÀM:

I. Trên 3 máy master, slave1, slave2

1st Step: Cài đặt MongoDB trên 3 máy

Step 1 — Installing MongoDB

```
hadoopuser@hadoop-master: ~
Preparing to unpack .../5-mongodb-org-tools_4.4.10_amd64.deb ...
Unpacking mongodb-org-tools (4.4.10) ...
Selecting previously unselected package mongodb-org.
Preparing to unpack .../6-mongodb-org_4.4.10_amd64.deb ...
Unpacking mongodb-org (4.4.10) ...
Setting up mongodb-org-server (4.4.10) ...
Adding system user `mongodb' (UID 128) ...
Adding new user `mongodb' (UID 128) with group `nogroup' ...
Not creating home directory `/home/mongodb'.
Adding group `mongodb' (GID 134) ...
Done.
Adding user `mongodb' to group `mongodb' ...
Adding user mongodb to group mongodb
Done.
Setting up mongodb-org-shell (4.4.10) ...
Setting up mongodb-database-tools (100.5.1) ...
Setting up mongodb-org-mongos (4.4.10) ...
Setting up mongodb-org-database-tools-extra (4.4.10) ...
Setting up mongodb-org-tools (4.4.10) ...
Setting up mongodb-org (4.4.10) ...
Processing triggers for man-db (2.9.1-1) ...
hadoopuser@hadoop-master:~$
```

Step 2 — Starting the MongoDB Service and Testing the Database

```
hadoopuser@hadoop-master: ~
Done.
Adding user `mongodb' to group `mongodb' ...
Adding user mongodb to group mongodb
Done.
Setting up mongodb-org-shell (4.4.10) ...
Setting up mongodb-database-tools (100.5.1) ...
Setting up mongodb-org-mongos (4.4.10) ...
Setting up mongodb-org-database-tools-extra (4.4.10) ...
Setting up mongodb-org-tools (4.4.10) ...
Setting up mongodb-org (4.4.10) ...
Processing triggers for man-db (2.9.1-1) ...
hadoopuser@hadoop-master:~$ sudo systemctl start mongod.service
hadoopuser@hadoop-master:~$ sudo systemctl status mongod
● mongod.service - MongoDB Database Server
     Loaded: loaded (/lib/systemd/system/mongod.service; disabled; vendor prese>
     Active: active (running) since Wed 2021-10-27 16:25:47 +07; 22s ago
       Docs: https://docs.mongodb.org/manual
   Main PID: 12873 (mongod)
     Memory: 59.9M
     CGroup: /system.slice/mongod.service
             └─12873 /usr/bin/mongod --config /etc/mongod.conf

Thg 10 27 16:25:47 hadoop-master systemd[1]: Started MongoDB Database Server.
lines 1-10/10 (END)
```

Check status

```
hadoopuser@hadoop-master: ~
     CGroup: /system.slice/mongod.service
             └─12873 /usr/bin/mongod --config /etc/mongod.conf

Thg 10 27 16:25:47 hadoop-master systemd[1]: Started MongoDB Database Server.

hadoopuser@hadoop-master:~$ sudo systemctl enable mongod
Created symlink /etc/systemd/system/multi-user.target.wants/mongod.service → /li
b/systemd/system/mongod.service.
hadoopuser@hadoop-master:~$ mongo --eval 'db.runCommand({ connectionStatus: 1 })
'
MongoDB shell version v4.4.10
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName
=mongodb
Implicit session: session { "id" : UUID("cd204419-d861-4792-822e-12770b0f5e1f")
}
MongoDB server version: 4.4.10
{
        "authInfo" : {
                "authenticatedUsers" : [ ],
                "authenticatedUserRoles" : [ ]
        },
        "ok" : 1
}
hadoopuser@hadoop-master:~$
```

2nd Step: **Step 1 — Configuring DNS Resolution**

```
hadoopuser@hadoop-master: ~
  GNU nano 4.8                    /etc/hosts
127.0.0.1       localhost
127.0.1.1       hadoop-VirtualBox

192.168.56.101 hadoop-master
192.168.56.102 hadoop-slave1
192.168.56.103 hadoop-slave2
# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Step 2 — Updating Each Server's Firewall Configurations with UFW

Trên máy hadoop-master:

```
hadoopuser@hadoop-master: ~
hadoopuser@hadoop-master:~$ sudo ufw allow from mongo1_server_ip to any port 270
17
ERROR: Bad source address
hadoopuser@hadoop-master:~$ sudo ufw allow from 192.168.56.102 to any port 27017
Rules updated
hadoopuser@hadoop-master:~$ sudo ufw allow from 192.168.56.103 to any port 27017
Rules updated
hadoopuser@hadoop-master:~$
```

Trên máy slave1:

```
hadoopuser@hadoop-slave1: ~
hadoopuser@hadoop-slave1:~$ sudo ufw allow from 192.168.56.101 to any port 2701
7
[sudo] password for hadoopuser:
Rules updated
hadoopuser@hadoop-slave1:~$ sudo ufw allow from 192.168.56.103 to any port 2701
7
Rules updated
hadoopuser@hadoop-slave1:~$
```

Trên máy slave2:

```
hadoopuser@hadoop-slave2: ~
hadoopuser@hadoop-slave2:~$ sudo ufw allow from 192.168.58.101 to any port 2701
7
[sudo] password for hadoopuser:
Rules updated
hadoopuser@hadoop-slave2:~$ sudo ufw allow from 192.168.58.102 to any port 2701
7
Rules updated
hadoopuser@hadoop-slave2:~$
```

Step 3 — Enabling Replication in Each Server's MongoDB Configuration File

Trên máy hadoop-master:

```
  GNU nano 4.8                    /etc/mongod.conf                    Modified
    enabled: true
#  engine:
#  mmapv1:
#  wiredTiger:

# where to write logging data.
systemLog:
  destination: file
  logAppend: true
  path: /var/log/mongodb/mongod.log

# network interfaces
net:
  port: 27017
  bindIp: 127.0.0.1, 192.168.56.101
```

```
hadoopuser@hadoop-master: ~
  GNU nano 4.8                    /etc/mongod.conf                    Modified


# how the process runs
processManagement:
  timeZoneInfo: /usr/share/zoneinfo

#security:

#operationProfiling:

replication:
  replSetName: "rs0"
```

Trên máy slave 1:

```
hadoopuser@hadoop-slave1: ~
  GNU nano 4.8                    /etc/mongod.conf                    Modified
#  mmapv1:
#  wiredTiger:

# where to write logging data.
systemLog:
  destination: file
  logAppend: true
  path: /var/log/mongodb/mongod.log

# network interfaces
net:
  port: 27017
  bindIp: 127.0.0.1, 192.168.56.102
```

```
hadoopuser@hadoop-slave1: ~
  GNU nano 4.8                    /etc/mongod.conf                    Modified

# network interfaces
net:
  port: 27017
  bindIp: 127.0.0.1, 192.168.56.102


# how the process runs
processManagement:
  timeZoneInfo: /usr/share/zoneinfo

#security:

#operationProfiling:

replication:
  replSetName: "rs0"
```

Trên máy slave 2:

```
hadoopuser@hadoop-slave2: ~
  GNU nano 4.8                    /etc/mongod.conf                    Modified
# for documentation of all options, see:
#   http://docs.mongodb.org/manual/reference/configuration-options/

# Where and how to store data.
storage:
  dbPath: /var/lib/mongodb
  journal:
    enabled: true
#  engine:
#  mmapv1:
#  wiredTiger:

# where to write logging data.
systemLog:
  destination: file
  logAppend: true
  path: /var/log/mongodb/mongod.log

# network interfaces
net:
  port: 27017
  bindIp: 127.0.0.1, 192.168.56.103
```

```
hadoopuser@hadoop-slave2: ~
  GNU nano 4.8                    /etc/mongod.conf                    Modified

# network interfaces
net:
  port: 27017
  bindIp: 127.0.0.1, 192.168.56.103


# how the process runs
processManagement:
  timeZoneInfo: /usr/share/zoneinfo

#security:

#operationProfiling:

replication:
  replSetName: "rs0"
#sharding:
```

Sau khi config xong, restart lại mỗi máy với lệnh:

$ sudo systemctl restart mongod

Test kết nối:

```
hadoopuser@hadoop-master:~$ nc -zv 192.168.56.102 27017
Connection to 192.168.56.102 27017 port [tcp/*] succeeded!
hadoopuser@hadoop-master:~$ nc -zv 192.168.56.103 27017
Connection to 192.168.56.103 27017 port [tcp/*] succeeded!
hadoopuser@hadoop-master:~$ nc -zv 192.168.56.101 27017
Connection to 192.168.56.101 27017 port [tcp/*] succeeded!
hadoopuser@hadoop-master:~$
```

Step 4 — Starting the Replica Set and Adding Members

Trên máy master (có thể thực hiện trên bất kỳ máy khác trong cụm):

Open Mongo shell:

$ mongo

```
---
> rs.initiate(
... {
... _id: "rs0",
... members: [
... { _id: 0, host: "hadoop-master" },
... { _id: 1, host: "hadoop-slave1" },
... { _id: 2, host: "hadoop-slave2" },
... ]
... })
```

```
hadoopuser@hadoop-master: ~
        To permanently disable this reminder, run the following command: db.
bleFreeMonitoring()
---
> rs.initiate(
... {
... _id: "rs0",
... members: [
... { _id: 0, host: "hadoop-master" },
... { _id: 1, host: "hadoop-slave1" },
... { _id: 2, host: "hadoop-slave2" },
... ]
... })
{
        "ok" : 1,
        "$clusterTime" : {
                "clusterTime" : Timestamp(1635393922, 1),
                "signature" : {
                        "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAA="),
                        "keyId" : NumberLong(0)
                }
        },
        "operationTime" : Timestamp(1635393922, 1)
}
rs0:SECONDARY>
```

Minh chứng cài đặt thành công: Kiểm tra thông tin cụm trên 1 máy bất kỳ bằng lệnh rs.status():

hadoopuser@hadoop-master: ~

```
---
rs0:PRIMARY> rs.status()
{
        "set" : "rs0",
        "date" : ISODate("2021-10-28T13:17:03.449Z"),
        "myState" : 1,
        "term" : NumberLong(2),
        "syncSourceHost" : "",
        "syncSourceId" : -1,
        "heartbeatIntervalMillis" : NumberLong(2000),
        "majorityVoteCount" : 2,
        "writeMajorityCount" : 2,
        "votingMembersCount" : 3,
        "writableVotingMembersCount" : 3,
        "optimes" : {
                "lastCommittedOpTime" : {
                        "ts" : Timestamp(1635427019, 1),
                        "t" : NumberLong(2)
                },
                "lastCommittedWallTime" : ISODate("2021-10-28T13:16:59.251Z"),
                "readConcernMajorityOpTime" : {
                        "ts" : Timestamp(1635427019, 1),
                        "t" : NumberLong(2)
                },
                "readConcernMajorityWallTime" : ISODate("2021-10-28T13:16:59.251Z"),
                "appliedOpTime" : {
                        "ts" : Timestamp(1635427019, 1),
                        "t" : NumberLong(2)
                },
                "durableOpTime" : {
                        "ts" : Timestamp(1635427019, 1),
                        "t" : NumberLong(2)
                },
                "lastAppliedWallTime" : ISODate("2021-10-28T13:16:59.251Z"),
                "lastDurableWallTime" : ISODate("2021-10-28T13:16:59.251Z")
        },
        "lastStableRecoveryTimestamp" : Timestamp(1635427009, 1),
        "electionCandidateMetrics" : {
                "lastElectionReason" : "electionTimeout",
                "lastElectionDate" : ISODate("2021-10-28T13:09:07.943Z"),
                "electionTerm" : NumberLong(2),
                "lastCommittedOpTimeAtElection" : {
                        "ts" : Timestamp(0, 0),
                        "t" : NumberLong(-1)
                },
                "lastSeenOpTimeAtElection" : {
                        "ts" : Timestamp(1635394153, 1),
                        "t" : NumberLong(1)
                },
```

hadoopuser@hadoop-master: ~

```
                "t" : NumberLong(1)
            },
            "numVotesNeeded" : 2,
            "priorityAtElection" : 1,
            "electionTimeoutMillis" : NumberLong(10000),
            "numCatchUpOps" : NumberLong(0),
            "newTermStartDate" : ISODate("2021-10-28T13:09:07.952Z"),
            "wMajorityWriteAvailabilityDate" : ISODate("2021-10-28T13:09:08.205Z")
    },
    "members" : [
            {
                    "_id" : 0,
                    "name" : "hadoop-master:27017",
                    "health" : 1,
                    "state" : 1,
                    "stateStr" : "PRIMARY",
                    "uptime" : 495,
                    "optime" : {
                            "ts" : Timestamp(1635427019, 1),
                            "t" : NumberLong(2)
                    },
                    "optimeDate" : ISODate("2021-10-28T13:16:59Z"),
                    "syncSourceHost" : "",
                    "syncSourceId" : -1,
                    "infoMessage" : "",
                    "electionTime" : Timestamp(1635426547, 1),
                    "electionDate" : ISODate("2021-10-28T13:09:07Z"),
                    "configVersion" : 1,
                    "configTerm" : 2,
                    "self" : true,
                    "lastHeartbeatMessage" : ""
            },
            {
                    "_id" : 1,
                    "name" : "hadoop-slave1:27017",
                    "health" : 1,
                    "state" : 2,
                    "stateStr" : "SECONDARY",
                    "uptime" : 481,
                    "optime" : {
                            "ts" : Timestamp(1635427019, 1),
                            "t" : NumberLong(2)
                    },
                    "optimeDurable" : {
                            "ts" : Timestamp(1635427019, 1),
                            "t" : NumberLong(2)
                    },
                    "optimeDate" : ISODate("2021-10-28T13:16:59Z"),
                    "optimeDurableDate" : ISODate("2021-10-28T13:16:59Z"),
```

```
hadoopuser@hadoop-master: ~
                        "lastHeartbeat" : ISODate("2021-10-28T13:17:01.827Z"),
                        "lastHeartbeatRecv" : ISODate("2021-10-28T13:17:02.392Z"),
                        "pingMs" : NumberLong(0),
                        "lastHeartbeatMessage" : "",
                        "syncSourceHost" : "hadoop-master:27017",
                        "syncSourceId" : 0,
                        "infoMessage" : "",
                        "configVersion" : 1,
                        "configTerm" : 2
                },
                {
                        "_id" : 2,
                        "name" : "hadoop-slave2:27017",
                        "health" : 1,
                        "state" : 2,
                        "stateStr" : "SECONDARY",
                        "uptime" : 476,
                        "optime" : {
                                "ts" : Timestamp(1635427019, 1),
                                "t" : NumberLong(2)
                        },
                        "optimeDurable" : {
                                "ts" : Timestamp(1635427019, 1),
                                "t" : NumberLong(2)
                        },
                        "optimeDate" : ISODate("2021-10-28T13:16:59Z"),
                        "optimeDurableDate" : ISODate("2021-10-28T13:16:59Z"),
                        "lastHeartbeat" : ISODate("2021-10-28T13:17:01.827Z"),
                        "lastHeartbeatRecv" : ISODate("2021-10-28T13:17:03.239Z"),
                        "pingMs" : NumberLong(0),
                        "lastHeartbeatMessage" : "",
                        "syncSourceHost" : "hadoop-master:27017",
                        "syncSourceId" : 0,
                        "infoMessage" : "",
                        "configVersion" : 1,
                        "configTerm" : 2
                }
        ],
        "ok" : 1,
        "$clusterTime" : {
                "clusterTime" : Timestamp(1635427019, 1),
                "signature" : {
                        "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAA="),
                        "keyId" : NumberLong(0)
                }
        },
        "operationTime" : Timestamp(1635427019, 1)
}
rs0:PRIMARY>
```

Trên một node SECONDARY bất kì, sử dụng lệnh: rs.isMaster() để kiểm tra node master trong cụm:

hadoopuser@hadoop-slave1: ~

```
toring()
        To permanently disable this reminder, run the following command: db.dis
ableFreeMonitoring()
---
rs0:SECONDARY> rs.isMaster()
{
        "topologyVersion" : {
                "processId" : ObjectId("617aa0e70d5a4287bbbbc3fa"),
                "counter" : NumberLong(4)
        },
        "hosts" : [
                "hadoop-master:27017",
                "hadoop-slave1:27017",
                "hadoop-slave2:27017"
        ],
        "setName" : "rs0",
        "setVersion" : 1,
        "ismaster" : false,
        "secondary" : true,
        "primary" : "hadoop-master:27017",
        "me" : "hadoop-slave1:27017",
        "lastWrite" : {
                "opTime" : {
                        "ts" : Timestamp(1635427469, 1),
                        "t" : NumberLong(2)
                },
                "lastWriteDate" : ISODate("2021-10-28T13:24:29Z"),
                "majorityOpTime" : {
                        "ts" : Timestamp(1635427469, 1),
```

hadoopuser@hadoop-slave2: ~

```
toring()
        To permanently disable this reminder, run the following command: db.dis
ableFreeMonitoring()
---
rs0:SECONDARY> rs.isMaster()
{
        "topologyVersion" : {
                "processId" : ObjectId("617aa0edae98f974066635ae"),
                "counter" : NumberLong(4)
        },
        "hosts" : [
                "hadoop-master:27017",
                "hadoop-slave1:27017",
                "hadoop-slave2:27017"
        ],
        "setName" : "rs0",
        "setVersion" : 1,
        "ismaster" : false,
        "secondary" : true,
        "primary" : "hadoop-master:27017",
        "me" : "hadoop-slave2:27017",
        "lastWrite" : {
                "opTime" : {
                        "ts" : Timestamp(1635427509, 1),
                        "t" : NumberLong(2)
                },
                "lastWriteDate" : ISODate("2021-10-28T13:25:09Z"),
                "majorityOpTime" : {
                        "ts" : Timestamp(1635427509, 1)
```

Step 5 — Import 1GB data

```
hadoopuser@hadoop-master: ~
3091,   "ctx":"initandlisten","msg":"Fatal assertion","attr":{"msgid":40486,"fi
le":"src/mongo/transport/transport_layer_asio.cpp","line":919}}
{"t":{"$date":"2021-10-28T22:39:31.128+07:00"},"s":"F",  "c":"-",        "id":2
3092,   "ctx":"initandlisten","msg":"\n\n***aborting after fassert() failure\n\
n"}
hadoopuser@hadoop-master:~$
hadoopuser@hadoop-master:~$ mongoimport --type csv -d test -c products --header
line --drop test.csv
2021-10-28T22:40:13.225+0700    connected to: mongodb://localhost/
2021-10-28T22:40:13.225+0700    dropping: test.products
2021-10-28T22:40:16.234+0700    [........................] test.products      6
.19MB/595MB (1.0%)
2021-10-28T22:40:19.227+0700    [........................] test.products      1
2.3MB/595MB (2.1%)
2021-10-28T22:40:22.227+0700    [........................] test.products      1
8.3MB/595MB (3.1%)
2021-10-28T22:40:25.227+0700    [........................] test.products      2
4.5MB/595MB (4.1%)
2021-10-28T22:40:28.226+0700    [#.......................] test.products      3
0.0MB/595MB (5.0%)
2021-10-28T22:40:31.226+0700    [#.......................] test.products      3
5.8MB/595MB (6.0%)
```

1 vài thông số của data

```
hadoopuser@hadoop-master: ~
rs0:PRIMARY> db.products.count()
10000000
rs0:PRIMARY> db.products.findOne()
{
        "_id" : ObjectId("617ac68687901bd4b1a37476"),
        "Region" : "Australia and Oceania",
        "Country" : "Palau",
        "Item Type" : "Office Supplies",
        "Sales Channel" : "Online",
        "Order Priority" : "H",
        "Order Date" : "3/6/2016",
        "Order ID" : 517073523,
        "Ship Date" : "3/26/2016",
        "Units Sold" : 2401,
        "Unit Price" : 651.21,
        "Unit Cost" : 524.96,
        "Total Revenue" : 1563555.21,
        "Total Cost" : 1260428.96,
        "Total Profit" : 303126.25
}
rs0:PRIMARY>
```

Bằng chứng lưu 1gb data và lưu phân tán trên 3 máy:

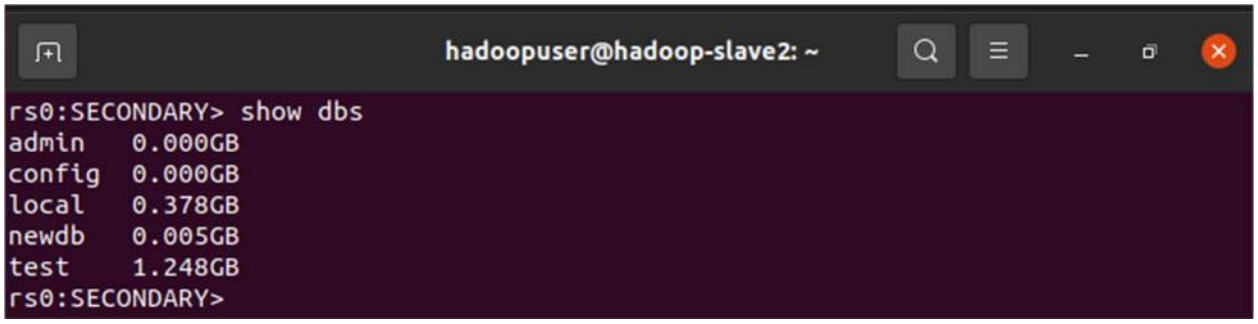
- Trên máy master:

```
hadoopuser@hadoop-master: ~
rs0:PRIMARY> show dbs
admin   0.000GB
config  0.000GB
local   0.382GB
newdb   0.005GB
test    1.245GB
rs0:PRIMARY>
```

- Trên máy slave 1:

```
hadoopuser@hadoop-slave1: ~
rs0:SECONDARY> show dbs
admin   0.000GB
config  0.000GB
local   0.380GB
newdb   0.005GB
test    1.255GB
rs0:SECONDARY>
```

- Trên máy slave 2:



Chú ý vào cơ sở dữ liệu test sẽ thấy được dung lượng lưu trữ là > 1gb và mục local ở 3 máy có dung lượng lưu trữ là khác nhau, và tổng dung lượng cộng lại > 1GB, bằng tổng số dung lượng dữ liệu đã thêm vào cơ sở dữ liệu test.