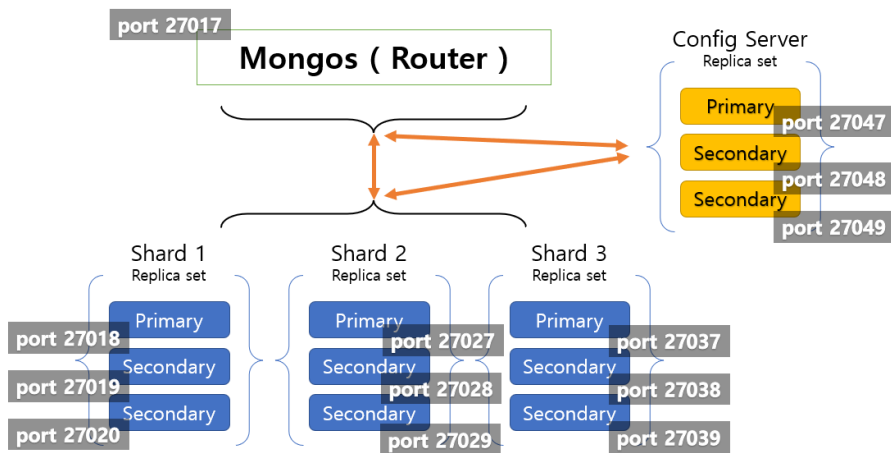


# 3 Shard and Replication set

10/04 시작



개요도

3 개 shard 로 구성

Config Server 와 각 Shard 들은 모두 replica set 으로 구성

Application 접근용 포트를 27017 로 가정하고 mongos 는 한개의 노드로 구성

먼저 각 shard 로 사용될 replica set 을 구성해보자

Dummy data 로 사용할 JSON 파일은

[한국수자원공사\\_도서정보](#) 이고

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
{
  '출판년도': '2008'
},
{
  id: ObjectId("633bc0d9d2af58eed85b9f26"),
  '순번': 8,
  '서명': "'대동아성 전대비'와 7인의 한국인 특공대",
  '저자': '일제강점하강제동원 피해진상규명위원회 편;',
  '출판사': '일제강점하강제동원 피해진상규명위원회',
  '출판년도': '2006'
},
{
  id: ObjectId("633bc0d9d2af58eed85b9f27"),
  '순번': 9,
  '서명': "'도시형 생활주택' 계획방향 및 설계기준 설정에 관한 연구",
  '저자': '건축도시공간연구소 편;김진욱 저;김영현 저;',
  '출판사': '건축도시공간연구소',
  '출판년도': '2009'
},
{
  id: ObjectId("633bc0d9d2af58eed85b9f28"),
  '순번': 10,
  '서명': "'지역재생잠재력지수'의의의와 시사점",
  '저자': '송미령 저;성주인 저;심재현 저',
  '출판사': '한국농촌경제연구원',
  '출판년도': '2021'
}
]
test_db> db.col.find().count()
39646
```

약 4 만개의 데이터를 담고 있다. ( 해당 사진은 자체 개발한 Script 를 사용하여 DB 에  
삽입한후 수를 세본 모습이다. )

10/05

전날 시도했던 실습이 실패하면서 다시 시도하기 위해 모든 과정을 적으려고 한다.

	port 번호	저장 위치
rs0	27027	data/rs0-0
	27028	data/rs0-1
	27029	data/rs0-2
이름	port 번호	저장 위치
rs1	27037	data/rs1-0
	27038	data/rs1-1
	27039	data/rs1-2

이름	port 번호	저장 위치	
rs2	27047	data/rs2-0	
	27048	data/rs2-1	
	27049	data/rs2-2	
이름	port 번호	저장 위치	비고
conf	27057	data/conf0-0	config server
	27058	data/conf0-1	
	27059	data/conf0-2	
이름	port 번호	저장 위치	비고
-	27017	data/mongos	mongos

먼저 데이터베이스가 저장될 디렉토리를 만들어 보겠습니다.

```
[shkim@localhost data]$ mkdir rs0-0 rs0-1 rs0-2
[shkim@localhost data]$ mkdir rs1-0 rs1-1 rs1-2
[shkim@localhost data]$ mkdir rs2-0 rs2-1 rs2-2
[shkim@localhost data]$ mkdir conf0-0 conf0-1 conf0-2
[shkim@localhost data]$ mkdir mongos
[shkim@localhost data]$ ls
conf0-0  conf0-2  rs0-0  rs0-2  rs1-1  rs2-0  rs2-2
conf0-1  mongos   rs0-1  rs1-0  rs1-2  rs2-1
[shkim@localhost data]$
```

다음은 sharding cluster 를 구성해보겠습니다

## 1. Config Server 설정 및 가동

```
storage:
  dbPath: data/conf0-0

systemLog:
  destination: file
  logAppend: true
  path: log/configsvr.log

net:
  port: 27057
  bindIp: 0.0.0.0

sharding:
  clusterRole: configsvr

replication:
  replSetName: conf #Config Server replset name
```

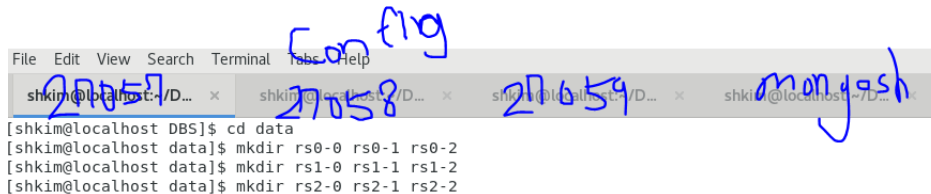
config 서버의 세부설정을 위해 `mongodConfig.conf` 를 생성하고 위와 같이 적는다

들여쓰기는 띄어쓰기 두개 '\_\_\_' 이다. 이를 주의하자..!

해당 config 파일을 이용해 config 서버를 가동시킨다.

```
[shkim@localhost DBS]$ sudo mongod --config conf/mongodConfig.conf
```

## 2.



```
[shkim@localhost DBS]$ cd data
[shkim@localhost data]$ mkdir rs0-0 rs0-1 rs0-2
[shkim@localhost data]$ mkdir rs1-0 rs1-1 rs1-2
[shkim@localhost data]$ mkdir rs2-0 rs2-1 rs2-2
```

sharding 을 구성하기 전 Config 서버의 replica set 의 초기화를 위해 각 서버를 가동시키고 다른 한 쉘로 `mongosh` 명령어로 3 개중 아무것에나 접근한다.

( 27057 포트 쉘의 경우 config 파일을 이용해 가동시킨 서버라는 점을 기억하자 )

```
test> rs.initiate({ _id: "conf", members: [ { _id:0, host: "localhost:27057"}, { _id:1, host: "localhost:27058" }, { _id:2, host: "localhost:27059"} ] })
{ ok: 1, lastCommittedOpTime: Timestamp({ t: 1664933902, i: 1 }) }
conf [direct: other] test>
conf [direct: secondary] test>
```

*에러발생 : 문제 해결을 위해 구글링 해보았지만 정확한 원인을 파악 할 수 없었고, config 파일 먼저가 아닌 shard 들을 먼저 replicaset 으로 구성하는*

방법으로 우회해보려고한다.

```
"t": {
  "$date": "2022-10-04T18:44:08.252-07:00"
},
"s": "I",
"c": "REPL",
"id": 21394,
"ctx": "ReplCoord-0",
"msg": "This node is not a member of the config"
}
```

3.

```
rs0 [direct: secondary] test> rs.hello()
{
  topologyVersion: {
    processId: ObjectId("633e1bdc921e2f370dbc64e0"),
    counter: Long("4")
  },
  hosts: [ 'localhost:27027', 'localhost:27028', 'localhost:27029' ],
  setName: 'rs0',
  setVersion: 1,
  isWritablePrimary: false,
  secondary: true,
  primary: 'localhost:27028',
  me: 'localhost:27027',
  ...
}
```

rs0 replica set 완성 같은 과정으로

rs1 rs2 conf 를 구성한다.

주의 : config 서버에 대한 repl set 을 구성하기 위한 각 conf 파일들은 기존과 다르다

```
storage:
  dbPath: data/conf0-0
  journal:
    enabled: true
    commitIntervalMs: 200

net:
  port: 27057
  bindIp: 0.0.0.0

setParameter:
  enableLocalhostAuthBypass: false

sharding:
  clusterRole: configsvr

replication:
  replSetName: "conf"
```

4. 이제 router 역할을 할 mongos 에 대한 conf 파일을 생성하고 실행해보자

```
[shkim@localhost DBS]$ mongos --config conf/mongoRouter.conf
```

이제 해당 포트 ( 본인은 27017 번 포트로 지정해두었다 ) 에 접속해보자,

mongos 라고 뜨면 성공이다

```
[shkim@localhost DBS]$ mongosh --port 27017
Current Mongosh Log ID: 633e34b171f8bdc73c35309
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&
ppName=mongosh+1.6.0
Using MongoDB:      6.0.1
Using Mongosh:      1.6.0

For mongosh info see: https://docs.mongodb.com/mongosh-shell/

-----
  The server generated these startup warnings when booting
  2022-10-05T18:51:01.433-07:00: Access control is not enabled for the database. Read and write access
  to data and configuration is unrestricted
  -----

[direct: mongos] test> █
```

와우우우+우우우우우우우!!!!!!!!!!!!!!!!!!!!!! 성공이다

5. 이제 Shard 서버를 위한 conf 파일을 생성 가동해보자

```
storage:
  dbPath: data/shardDB/
  journal:
    enabled: true
    commitIntervalMs: 200

systemLog:
  destination: file
  logAppend: true
  path: log/shard.log

net:
  port: 27018
  bindIp: 0.0.0.0

setParameter:
  enableLocalhostAuthBypass: false

sharding:
  clusterRole: shardsvr

replication:
  replSetName: "ShardReplSet"
```



