



Last login: Mon Jun 16 20:29:37 on ttys000

rilu@Anshus-MacBook-Air ~ % mkdir -p ~/mongo-replSet/rs1

mkdir -p ~/mongo-replSet/rs2

mkdir -p ~/mongo-replSet/rs3

rilu@Anshus-MacBook-Air ~ % █



Last login: Wed Jun 18 18:51:07 on ttys007

```
rilu@Anshus-MacBook-Air ~ % mongod --dbpath ~/mongo-replSet/rs1 --port 27017 --r  
eplSet rsTest --bind_ip localhost
```

```
{"t":{"$date":"2025-06-18T18:51:34.855+05:30"},"s":"I", "c":"CONTROL", "id":23  
285, "ctx":"thread1","msg":"Automatically disabling TLS 1.0, to force-enable T  
LS 1.0 specify --sslDisabledProtocols 'none'"}  
{"t":{"$date":"2025-06-18T18:51:34.856+05:30"},"s":"I", "c":"CONTROL", "id":59  
45603, "ctx":"thread1","msg":"Multi threading initialized"}  
{"t":{"$date":"2025-06-18T18:51:34.858+05:30"},"s":"I", "c":"NETWORK", "id":46  
48601, "ctx":"thread1","msg":"Implicit TCP FastOpen unavailable. If TCP FastOpen  
is required, set at least one of the related parameters","attr":{"relatedParam  
eters":["tcpFastOpenServer","tcpFastOpenClient","tcpFastOpenQueueSize"]}}
```

```
{"t":{"$date":"2025-06-18T18:51:34.859+05:30"},"s":"I", "c":"NETWORK", "id":49  
15701, "ctx":"thread1","msg":"Initialized wire specification","attr":{"spec":{"i  
ncomingExternalClient":{"minWireVersion":0,"maxWireVersion":25},"incomingInterna  
lClient":{"minWireVersion":0,"maxWireVersion":25},"outgoing":{"minWireVersion":6  
,"maxWireVersion":25},"isInternalClient":true}}}
```

```
{"t":{"$date":"2025-06-18T18:51:34.862+05:30"},"s":"I", "c":"TENANT_M", "id":70  
91600, "ctx":"thread1","msg":"Starting TenantMigrationAccessBlockerRegistry"}  
{"t":{"$date":"2025-06-18T18:51:34.862+05:30"},"s":"I", "c":"CONTROL", "id":46  
15611, "ctx":"initandlisten","msg":"MongoDB starting","attr":{"pid":81557,"port"  
:27017,"dbPath":"/Users/rilu/mongo-replSet/rs1","architecture":"64-bit","host":  
"Anshus-MacBook-Air.local"}}
```

rilu — mongod --dbpath ~/mongo-replSet/rs2 --port 27018 --replSet rsTe...

```
{"t":{"$date":"2025-06-18T18:59:06.629+05:30"},"s":"I", "c":"REPL", "id":3873106, "ctx":"BackgroundSync","msg":"Cannot select sync source because it is not up","attr":{"syncSourceCandidate":"localhost:27019"}}
{"t":{"$date":"2025-06-18T18:59:06.629+05:30"},"s":"I", "c":"REPL", "id":8423402, "ctx":"BackgroundSync","msg":"Failed to select a sync source on the first attempt. Starting second attempt"}
{"t":{"$date":"2025-06-18T18:59:06.629+05:30"},"s":"I", "c":"REPL", "id":3873113, "ctx":"BackgroundSync","msg":"Cannot select sync source which is not ahead of me","attr":{"syncSourceCandidate":"localhost:27017","syncSourceCandidateLastAppliedOpTime":{"ts":{"$timestamp":{"t":1750081226,"i":1}},"t":3},"lastOpTimeFetched":{"ts":{"$timestamp":{"t":1750081226,"i":1}},"t":3}}}
{"t":{"$date":"2025-06-18T18:59:06.629+05:30"},"s":"I", "c":"REPL", "id":3873106, "ctx":"BackgroundSync","msg":"Cannot select sync source because it is not up","attr":{"syncSourceCandidate":"localhost:27019"}}
{"t":{"$date":"2025-06-18T18:59:06.634+05:30"},"s":"I", "c":"REPL_HB", "id":23974, "ctx":"ReplCoord-1","msg":"Heartbeat failed after max retries","attr":{"target":"localhost:27019","maxHeartbeatRetries":2,"error":{"code":6,"codeName":"HostUnreachable","errmsg":"Error connecting to localhost:27019 (127.0.0.1:27019) :: caused by :: onInvoke :: caused by :: Connection refused"}}}
{"t":{"$date":"2025-06-18T18:59:06.717+05:30"},"s":"I", "c":"-", "id":4939300, "ctx":"monitoring-keys-for-HMAC","msg":"Failed to refresh key cache","attr":{"error":"ReadConcernMajorityNotAvailableYet: Read concern majority reads are currently not possible.","nextWakeupMillis":1600}}
```

```
{ "t": {  
88700,
```

```
bash
```

```
rilu — mongod --dbpath ~/mongo-replSet/rs3 --port 27019 --replSet rsTe...
```

```
{ "t": { "$date": "2025-06-18T18:59:20.354+05:30", "s": "I", "c": "REPL", "id": 21088, "ctx": "BackgroundSync", "msg": "Changed sync source", "attr": { "oldSyncSource": "empty", "newSyncSource": "localhost:27018" } }  
{ "t": { "$date": "2025-06-18T18:59:20.356+05:30", "s": "I", "c": "REPL", "id": 6795400, "ctx": "ReplCoordExtern-0", "msg": "Advancing committed opTime to a new term", "attr": { "newCommittedOpTime": { "ts": { "$timestamp": { "t": 1750081226, "i": 1 } }, "t": 3 }, "newCommittedWallime": { "$date": "2025-06-16T13:40:26.643Z", "oldTerm": -1 } } }  
{ "t": { "$date": "2025-06-18T18:59:20.356+05:30", "s": "I", "c": "REPL", "id": 6795400, "ctx": "ReplCoordExtern-0", "msg": "Advancing committed opTime to a new term", "attr": { "newCommittedOpTime": { "ts": { "$timestamp": { "t": 1750253360, "i": 1 } }, "t": 4 }, "newCommittedWallime": { "$date": "2025-06-18T13:29:20.262Z", "oldTerm": 3 } } }  
{ "t": { "$date": "2025-06-18T18:59:21.846+05:30", "s": "I", "c": "NETWORK", "id": 22943, "ctx": "listener", "msg": "Connection accepted", "attr": { "remote": "127.0.0.1:56452", "uuid": { "uuid": { "$uuid": "e1948c02-6f62-4802-942b-57b0507a5f18" } }, "connectionId": 15, "connectionCount": 5 } } }  
{ "t": { "$date": "2025-06-18T18:59:21.847+05:30", "s": "I", "c": "NETWORK", "id": 51800, "ctx": "conn15", "msg": "client metadata", "attr": { "remote": "127.0.0.1:56452", "client": "conn15", "negotiatedCompressors": ["snappy", "zstd", "zlib"], "doc": { "driver": { "name": "NetworkInterfaceTL-ReplNetwork", "version": "8.0.9", "os": { "type": "Darwin", "name": "Mac OS X", "architecture": "arm64", "version": "24.5.0" } } } } }  
{ "t": { "$date": "2025-06-18T18:59:21.849+05:30", "s": "I", "c": "NETWORK", "id": 6788700, "ctx": "conn15", "msg": "Received first command on ingress connection since session start or auth handshake", "attr": { "elapsedMillis": 2 } }
```



Last login: Wed Jun 18 18:59:32 on ttys023  
rilu@Anshus-MacBook-Air ~ % mongosh --port 27018

Current Mongosh Log ID: 6852c0b6cbafcf804b5baecc  
Connecting to: **mongodb://127.0.0.1:27018/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.5.1**  
Using MongoDB: 8.0.9  
**Using Mongosh:** 2.5.1  
mongosh 2.5.2 is available for download: **<https://www.mongodb.com/try/download/shell>**

For mongosh info see: **<https://www.mongodb.com/docs/mongodb-shell/>**

-----  
**The server generated these startup warnings when booting**  
2025-06-18T18:59:01.082+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted  
2025-06-18T18:59:01.082+05:30: Soft rlimits for open file descriptors too low  
-----

```
rsTest [direct: secondary] test> rs.slaveOk()
TypeError: rs.slaveOk is not a function
rsTest [direct: secondary] test> use demoDB
[switched to db demoDB]
rsTest [direct: secondary] demoDB> db.users.find()
[[
  {
    _id: ObjectId('685012ef6ec003f814732cc9'),
    name: 'Alice',
    role: 'admin',
    time: ISODate('2025-06-16T12:49:51.699Z')
  },
  {
    _id: ObjectId('6852bf89db183de0165d4d31'),
    name: 'Alice',
    role: 'admin'
  }
]
rsTest [direct: secondary] demoDB> █
```

Last login: Wed Jun 18 18:59:11 on ttys019  
rilu@Anshus-MacBook-Air ~ % mongosh --port 27017

Current Mongosh Log ID: 6852bf43db183de0165d4d30  
Connecting to: **mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.5.1**  
Using MongoDB: 8.0.9  
**Using Mongosh:** 2.5.1  
mongosh 2.5.2 is available for download: <https://www.mongodb.com/try/download/shell>

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

-----  
**The server generated these startup warnings when booting**  
2025-06-18T18:58:27.661+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted  
2025-06-18T18:58:27.661+05:30: Soft rlimits for open file descriptors too low  
-----

```
rsTest [direct: primary] test> rs.initiate({
...   _id: "rsTest",
...   members: [
...     { _id: 0, host: "localhost:27017" },
...     { _id: 1, host: "localhost:27018" },
...     { _id: 2, host: "localhost:27019" }
...   ]
... })
...
MongoServerError[AlreadyInitialized]: already initialized
rsTest [direct: primary] test> rs.status()
[...
{
  set: 'rsTest',
  date: ISODate('2025-06-18T13:30:07.027Z'),
  myState: 1,
  term: Long('4'),
  syncSourceHost: '',
  syncSourceId: -1,
  heartbeatIntervalMillis: Long('2000'),
  majorityVoteCount: 2,
  writeMajorityCount: 2,
  votingMembersCount: 3,
  writableVotingMembersCount: 3,
  optimes: {
    lastCommittedOpTime: { ts: Timestamp({ t: 1750253400, i: 1 }), t: Long('4') },
    lastCommittedWallTime: ISODate('2025-06-18T13:30:00.272Z'),
    readConcernMajorityOpTime: { ts: Timestamp({ t: 1750253400, i: 1 }), t: Long('4') },
    appliedOpTime: { ts: Timestamp({ t: 1750253400, i: 1 }), t: Long('4') },
    durableOpTime: { ts: Timestamp({ t: 1750253400, i: 1 }), t: Long('4') },
    writtenOpTime: { ts: Timestamp({ t: 1750253400, i: 1 }), t: Long('4') },
    lastAppliedWallTime: ISODate('2025-06-18T13:30:00.272Z'),
    lastDurableWallTime: ISODate('2025-06-18T13:30:00.272Z'),
    lastWrittenWallTime: ISODate('2025-06-18T13:30:00.272Z')
  },
  lastStableRecoveryTimestamp: Timestamp({ t: 1750253360, i: 1 }),
  electionCandidateMetrics: {
    lastElectionReason: 'electionTimeout',
    lastElectionDate: ISODate('2025-06-18T13:29:10.233Z'),
    electionTerm: Long('4'),
    lastCommittedOpTimeAtElection: { ts: Timestamp({ t: 0, i: 0 }), t: Long('-1') },
    lastSeenWrittenOpTimeAtElection: { ts: Timestamp({ t: 1750081226, i: 1 }), t: Long('3') },
    lastSeenOpTimeAtElection: { ts: Timestamp({ t: 1750081226, i: 1 }), t: Long('3') },
    numVotesNeeded: 2,
    priorityAtElection: 1,
    electionTimeoutMillis: Long('10000'),
```



```

    lastHeartbeat: ISODate('2025-06-18T13:30:06.293Z'),
    lastHeartbeatRecv: ISODate('2025-06-18T13:30:06.886Z'),
    pingMs: Long('0'),
    lastHeartbeatMessage: '',
    syncSourceHost: 'localhost:27018',
    syncSourceId: 1,
    infoMessage: '',
    configVersion: 1,
    configTerm: 4
  }
],
ok: 1,
'$clusterTime': {
  clusterTime: Timestamp({ t: 1750253400, i: 1 }),
  signature: {
    hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
    keyId: Long('0')
  }
},
operationTime: Timestamp({ t: 1750253400, i: 1 })
}
rsTest [direct: primary] test> use demoDb
switched to db demoDb
rsTest [direct: primary] demoDb> use demoDB
... db.users.insertOne({ name: "Alice", role: "admin" })
... db.users.find()
...
switched to db demoDB
rsTest [direct: primary] demoDB> use demoDB
[... db.users.insertOne({ name: "Alice", role: "admin" })
... db.users.find()
... db.users.insertOne({ name: "Alice", role: "admin"
already on db demoDB
rsTest [direct: primary] demoDB> db.users.insertOne({ name: "Alice", role: "admin" })
[
  {
    acknowledged: true,
    insertedId: ObjectId('6852bf89db183de0165d4d31')
  }
]
rsTest [direct: primary] demoDB> db.users.find()
[[
  {
    _id: ObjectId('685012ef6ec003f814732cc9'),
    name: 'Alice',
    role: 'admin',
    time: ISODate('2025-06-16T12:49:51.699Z')
  },
  {
    _id: ObjectId('6852bf89db183de0165d4d31'),
    name: 'Alice',
    role: 'admin'
  }
]]

```