



ACTIVIDAD 5 - CONFIGURACIÓN MONGODB

Cristóbal Suárez Abad

ADMINISTRACIÓN DE SISTEMAS GESTORES DE BASES DE DATOS - 2º ASIR



Contenido

1) Configura el servicio para que se inicie automáticamente al arrancar el sistema.	2
2) Investiga donde se cambia el motor que usa MongoDB, di algunas de las posibilidades explica que diferencias hay.	2
3) Crear un nuevo usuario con permisos de administración y habilita la autenticación. Prueba que esto funciona.	4
4) Configura las siguientes características de red:.....	8
5) Configura un tamaño máximo para una colección. Comprueba que ocurre cuando se supera.....	9

1) Configura el servicio para que se inicie automáticamente al arrancar el sistema.

systemctl enable mongod

```
[root@mongodb01 usuario]$systemctl enable mongod
Tu Nombre  lunes 10 noviembre 2025 15:40
[root@mongodb01 usuario]$
```

Tendría que salir un mensaje de confirmación, pero como ya se hizo en la primera actividad de MongoDB, pues ahora no sale.

2) Investiga donde se cambia el motor que usa MongoDB, di algunas de las posibilidades explica que diferencias hay.

En el archivo de configuración de MongoDB¹: **/etc/mongod.conf**

Debemos descomentar la línea de “**engine**” y especificar el motor que queremos usar.

```
# where and how to store data.
storage:
  dbPath: /var/lib/mongodb
# engine:
# wiredTiger:
```

Hay que apagar el servicio y luego reiniciarlo.

También puedes usar la variable “**--storageEngine**” cuando inicias el servicio².

Ejemplo: **mongod --storageEngine inMemory --dbpath /var/lib/mongodb**

De igual manera que el anterior. El servicio debe estar apagado.

¿Qué opciones de motores tenemos?³

- **WiredTiger Storage Engine:** Es el que viene por defecto. Tiene persistencia (checkpoints cada 60 segundos y journal).
- **In-Memory Storage Engine:** Solo disponible en MongoDB Enterprise. No tiene persistencia (todo se almacena en la RAM, lo que le da una gran velocidad de

¹ <https://stackoverflow.com/questions/52594128/how-to-convert-mongodb-storage-engine-from-wiredtiger-to-in-memory-in-replica-se>

² <https://docs.percona.com/percona-server-for-mongodb/4.0/switch-storage-engines.html>

³ <https://www.mongodb.com/docs/manual/core/storage-engines/>

operación). Su tamaño debe ser especificado con “--inMemorySizeGB”, para evitar errores.

Para comprobar que motor se está usando podemos usar:

db.serverStatus().storageEngine

```
rs0 [direct: primary] admin> db.serverStatus().storageEngine
{
  name: 'wiredTiger',
  supportsCommittedReads: true,
  oldestRequiredTimestampForCrashRecovery: Timestamp({ t: 1762793022, i: 1 }),
  supportsPendingDrops: true,
  dropPendingIdents: Long('0'),
  supportsSnapshotReadConcern: true,
  readOnly: false,
  persistent: true,
  backupCursorOpen: false
}
rs0 [direct: primary] admin> _
```

3) Crear un nuevo usuario con permisos de administración y habilita la autenticación. Prueba que esto funciona.

Entramos en el Shell de mongodb. Importante, debe hacerse en el primario.

mongosh

Y ahora ejecutamos lo siguiente:

Nos posicionamos en la tabla:

use admin

```
rs0 [direct: secondary] test> use admin
```

Y ahora debemos introducir el siguiente código:

```
db.createUser({  
  user: "usuario",  
  pwd: "Usuario.25",  
  roles: [ { role: "root", db: "admin" } ]  
})
```

```
switched to db admin
rs0 [direct: primary] admin> db.createUser({
... user: "usuario",
... pwd: "Usuario.25",
... roles: [ { role: "root", db: "admin" } ]
... })
{
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1762854556, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1762854556, i: 1 })
}
rs0 [direct: primary] admin> |
```

Le hemos dicho que en la tabla “**admin**” (“use admin”) cree un usuario, llamado “**usuario**” con los privilegios de “**root**”.

Ahora podremos autenticarnos con el usuario “usuario”.

Ahora debemos habilitar la autenticación en cada uno de los nodos. En el archivo “mongod.conf”. En los tres nodos.

security:

keyFile: /etc/mongod.key

authorization: enabled

```
#security:
security:
  keyFile: /etc/mongod.keyfile
  authorization: enabled|
#operationProfiling:
```

Si realizamos ahora un reinicio del servicio, este sufrirá un error y no funcionará. Eso es porque requiere de un “**.keyfile**”. Para ello en el nodo primario

openssl rand -base64 756 > /etc/mongod.keyfile

sudo chown mongodb:mongodb /etc/mongod.keyfile

sudo chmod 400 /etc/mongod.keyfile

```
[root@mongodb03 usuario]$openssl rand -base64 756 > /etc/mongod.keyfile
Tu Nombre martes 11 noviembre 2025 09:52
[root@mongodb03 usuario]$sudo chown mongodb:mongodb /etc/mongod.keyfile
Tu Nombre martes 11 noviembre 2025 09:52
[root@mongodb03 usuario]$sudo chmod 400 /etc/mongod.keyfile
Tu Nombre martes 11 noviembre 2025 09:52
[root@mongodb03 usuario]$ls -l /etc/mongod.keyfile
-r----- 1 mongodb mongodb 1024 nov 11 09:52 /etc/mongod.keyfile
```

Ahora pasamos el archivo “**.keyfile**” a los otros nodos.

sudo scp /etc/mongod.key usuario@10.2.17.80:/tmp/

```
Tu Nombre  lunes 10 noviembre 2025 17:08
[root@mongodb03 usuario]$sudo scp /etc/mongod.key usuario@10.2.17.80:/tmp/
The authenticity of host '10.2.17.80 (10.2.17.80)' can't be established.
ED25519 key fingerprint is SHA256:SZwXLMx+wUdcSP8wh7BqQ/a59GCadpfJN5Fuh/1UA/Q.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.2.17.80' (ED25519) to the list of known hosts.
usuario@10.2.17.80's password:
mongod.key                                100% 1024    791.1KB/s   00:00
Tu Nombre  lunes 10 noviembre 2025 17:10
[root@mongodb03 usuario]$_
```

sudo scp /etc/mongod.key usuario@10.2.17.85:/tmp/

```
Tu Nombre  lunes 10 noviembre 2025 17:10
[root@mongodb03 usuario]$sudo scp /etc/mongod.key usuario@10.2.17.85:/tmp/
The authenticity of host '10.2.17.85 (10.2.17.85)' can't be established.
ED25519 key fingerprint is SHA256:SZwXLMx+wUdcSP8wh7BqQ/a59GCadpfJN5Fuh/1UA/Q.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:1: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.2.17.85' (ED25519) to the list of known hosts.
usuario@10.2.17.85's password:
mongod.key                                100% 1024    273.0KB/s   00:00
Tu Nombre  lunes 10 noviembre 2025 17:10
```

Ahora, en los otros dos nodos, movemos los “**.keyfile**” a el directorio donde hemos dicho que deben estar en el archivo de configuración.

sudo mv /tmp/mongod.key /etc/mongod.key

sudo chown mongodb:mongodb /etc/mongod.key

sudo chmod 400 /etc/mongod.key

```
[root@mongodb02 usuario]$
sudo mv /tmp/mongod.key /etc/mongod.key
sudo chown mongodb:mongodb /etc/mongod.key
sudo chmod 400 /etc/mongod.key
Tu Nombre  lunes 10 noviembre 2025 17:15
[root@mongodb02 usuario]$ls -l /etc/mongod.key
-r----- 1 mongodb mongodb 1024 nov 10 17:10 /etc/mongod.key
Tu Nombre  lunes 10 noviembre 2025 17:15
[root@mongodb02 usuario]$
```

Ahora hacemos en cada uno de los nodos:

sudo systemctl restart mongod

sudo systemctl status mongod

```
[root@mongodb03 usuario]$sudo systemctl restart mongod
sudo systemctl status mongod
● mongod.service - MongoDB Database Server
   Loaded: loaded (/lib/systemd/system/mongod.service; enabled; vendor preset: >
   Active: active (running) since Mon 2025-11-10 17:16:56 UTC; 61ms ago
     Docs: https://docs.mongodb.org/manual
   Main PID: 1643 (mongod)
    Memory: 7.6M
       CPU: 48ms
   CGroup: /system.slice/mongod.service
           └─1643 /usr/bin/mongod --config /etc/mongod.conf

nov 10 17:16:56 mongodb03 systemd[1]: Started MongoDB Database Server.

Tu Nombre  lunes 10 noviembre 2025 17:17
[root@mongodb03 usuario]$
```

Ahora verificamos que podemos conectarnos con el nuevo usuario:

mongosh -u "usuario" -p "Usuario.25" --authenticationDatabase "admin"

```
Tu Nombre  martes 11 noviembre 2025 09:59
[root@mongodb03 usuario]$mongosh -u "usuario" -p "Usuario.25" --authenticationDatabase "admin"
Current Mongosh Log ID: 691308f68a61c198049dc29c
Connecting to:      mongodb://<credentials>@127.0.0.1:27017/?directConnection=true&serverSelection
TimeoutMS=2000&authSource=admin&appName=mongosh+2.5.9
Using MongoDB:      7.0.25
Using Mongosh:      2.5.9

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

-----
The server generated these startup warnings when booting
2025-11-11T09:57:26.227+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger
storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
2025-11-11T09:57:28.070+00:00: vm.max_map_count is too low
-----

rs0 [direct: primary] test> |
```


4) Configura las siguientes características de red:

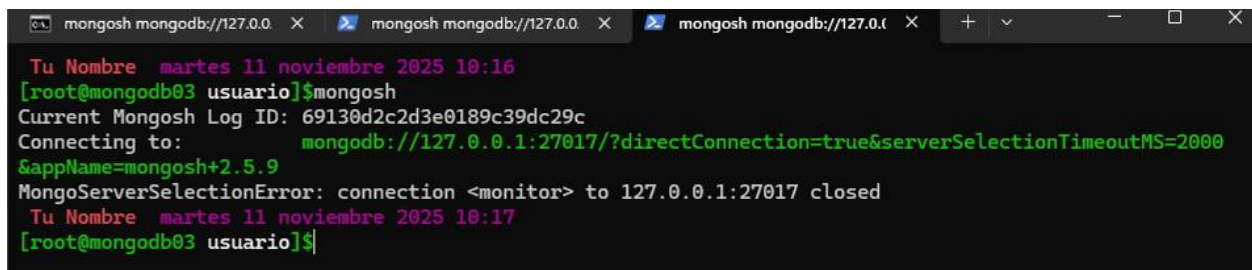
- Solo puede haber 8 conexiones de forma concurrente. Comprueba usando el equipo de tu compañero que esta configuración funciona.

En el archivo “**/etc/mongod.conf**” debes añadir la siguiente línea:

maxIncomingConnections: 8

```
net:
  port: 27017
  bindIp: 127.0.0.1, 10.2.17.90
  maxIncomingConnections: 8
# how the process runs
```

Si intentamos entrar nos dará seguramente un problema parecido a este:



```
Tu Nombre martes 11 noviembre 2025 10:16
[root@mongodb03 usuario]$mongo
Current Mongosh Log ID: 69130d2c2d3e0189c39dc29c
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
&appName=mongosh+2.5.9
MongoServerSelectionError: connection <monitor> to 127.0.0.1:27017 closed
Tu Nombre martes 11 noviembre 2025 10:17
[root@mongodb03 usuario]$
```

Si inspeccionamos el log: “**cat /var/log/mongodb/mongod.log**”

Veremos que el problema es que rechaza nuestra conexión porque ya se ha llegado al límite de conexiones. Creo que es debido a las réplicas.

- Inhabilita para que solo te puedas conectar desde la ip de tu equipo, prueba desde el equipo de un compañero que no te puedes conectar.

Habilitamos el firewall: **ufw enable**

Bloqueamos todo: **ufw default deny**

Permitimos solo nuestra ip por el puerto predeterminado de mongodb:

ufw allow from 172.16.40.108 to any port 27017

5) Configura un tamaño máximo para una colección. Comprueba que ocurre cuando se supera.

Entramos en mongodb: mongosh

He introducidos el siguiente código:

```
db.createCollection("cni", {  
  capped: true,  
  size: 512 * 512, // 0.5 MB  
  max: 5  
})
```

Lo que hace es crear una colección nueva en la base de datos donde estamos.

“**capped : true**” establece que las limitaciones que se van a definir se cumplan.

“**size: 512 * 512**” Limita a archivos de 256KB (512*512= 262144 Bytes).

“**max: 5**” El número máximo de archivos.

La manera de probarlo es que si vamos a la colección y añadimos documentos, no se podrá tener más de 5. Los antiguos los irá eliminando.

connections Edit View Collection Help

The screenshot shows the MongoDB Compass interface. On the left, the 'CONNECTIONS (3)' sidebar lists several databases and collections. The 'cni' collection under the 'test' database is selected. The main panel displays the 'Documents' tab for the 'cni' collection, showing 5 documents. Each document contains a single field: '_id' with an ObjectId value. The interface includes a search bar, a query editor, and buttons for 'ADD DATA', 'EXPORT DATA', 'UPDATE', and 'DELETE'.

Compass

La del 80 > test > cni

Documents 5 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#)

ADD DATA EXPORT DATA UPDATE DELETE

_id: ObjectId('6913192b1096f5d1d45b9175')
_id: ObjectId('6913192f1096f5d1d45b9176')
_id: ObjectId('691319341096f5d1d45b9177')
_id: ObjectId('6913193c1096f5d1d45b9178')
_id: ObjectId('691319421096f5d1d45b9179')