

## Actividad 5 - Configuración MongoDB

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

Habilitamos e iniciamos los 3 nodos.

```
sudo systemctl enable mongod
```

```
sudo systemctl start mongod
```

```
sudo systemctl status mongod
```

```
Armando viernes 7 noviembre 2025 09:03
[usuario@server2asir ~]$ sudo systemctl status mongod
[sudo] password for usuario:
● mongod.service - MongoDB Database Server
  Loaded: loaded (/lib/systemd/system/mongod.service; enabled; vendor preset: enabled)
  Active: active (running) since Fri 2025-11-07 09:01:52 UTC; 2min 28s ago
    Docs: https://docs.mongodb.org/manual
 Main PID: 670 (mongod)
   Memory: 250.3M
      CPU: 3.694s
     CGroup: /system.slice/mongod.service
             └─670 /usr/bin/mongod --config /etc/mongod.conf

nov 07 09:01:52 server2asir systemd[1]: Started MongoDB Database Server.
nov 07 09:01:52 server2asir mongod[670]: {"t":{"$date":"2025-11-07T09:01:52.930Z"}, "s": "I", "c": "CONTROL", "id": 74845}
[lines 1-12/12 (END)]
Armando viernes 7 noviembre 2025 09:04
[usuario@server2asir ~]$
```

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

Editamos el archivo mongod.conf para definir el motor de almacenamiento en cada nodo.

```
sudo nano /etc/mongod.conf
```

```
GNU nano 6.2                               /etc/mongod.conf

# 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: wiredTiger
```

Reiniciamos el servicio.

```
sudo systemctl restart mongod
```

```
sudo systemctl status mongod
```

```
Armando viernes 7 noviembre 2025 09:12
[usuario@server2asir ~]$ sudo systemctl restart mongod
Armando viernes 7 noviembre 2025 09:12
[usuario@server2asir ~]$ sudo systemctl status mongod
● mongod.service - MongoDB Database Server
  Loaded: loaded (/lib/systemd/system/mongod.service; enabled; vendor preset: enabled)
  Active: active (running) since Fri 2025-11-07 09:12:35 UTC; 7s ago
    Docs: https://docs.mongodb.org/manual
 Main PID: 1239 (mongod)
   Memory: 172.8M
      CPU: 1.996s
     CGroup: /system.slice/mongod.service
             └─1239 /usr/bin/mongod --config /etc/mongod.conf

nov 07 09:12:35 server2asir systemd[1]: Started MongoDB Database Server.
nov 07 09:12:35 server2asir mongod[1239]: {"t":{"$date":"2025-11-07T09:12:35.648Z"}, "s": "I", "c": "CONTROL", "id": 74845}
[lines 1-12/12 (END)]
Armando viernes 7 noviembre 2025 09:12
[usuario@server2asir ~]$
```

Y comprobamos que el motor está en uso.

```
mongosh
db.serverStatus().storageEngine
```

```
Current Mongosh Log ID: 690db8483ccc96b10a9dc29c
Connecting to: mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2
.5.9
Using MongoDB: 6.0.26
Using Mongosh: 2.5.9

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

-----
The server generated these startup warnings when booting
2025-11-07T09:12:35.676+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
2025-11-07T09:12:37.409+00:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
2025-11-07T09:12:37.409+00:00: vm.max_map_count is too low
-----

rs0 [direct: secondary] test> db.serverStatus().storageEngine
{
  name: 'wiredTiger',
  supportsCommittedReads: true,
  oldestRequiredTimestampForCrashRecovery: Timestamp({ t: 1762506811, i: 1 }),
  supportsPendingDrops: true,
  dropPendingIdentifiers: Long('0'),
  supportsSnapshotReadConcern: true,
  readOnly: false,
  persistent: true,
  backupCursorOpen: false
}
rs0 [direct: secondary] test> |
```

- **wiredTiger**: Motor por defecto desde MongoDB 3.2.

Ventajas: Compresión, concurrencia, alto rendimiento.

Inconvenientes: Requiere más memoria.

- **inMemory**: Mantiene los datos sólo en RAM.

Ventajas: Es muy rápido.

Inconveniente: No guarda datos en disco.

- **mmapv1 (obsoleto)**: Antiguo motor basado en mapeo de archivos.

Ventajas: Simplicidad.

Inconvenientes: Bajo rendimiento y sin compresión.

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

```
mongosh
```

Ponemos el comando **rs.status()** para saber cual es el nodo primario.

En el nodo primario entramos en la base de datos admin y creamos un usuario con rol de administrador.

```
use admin
```

```
db.createUser({
  user: "admin",
  pwd: "admin123",
  roles: [ { role: "root", db: "admin" } ]
})
```

```

rs0 [direct: primary] test> use admin
switched to db admin
rs0 [direct: primary] admin> db.createUser({
...   user: "admin",
...   pwd: "admin123",
...   roles: [ { role: "root", db: "admin" } ]
... })
...
{
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1762507928, i: 4 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAA='),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1762507928, i: 4 })
}
rs0 [direct: primary] admin> |

```

En los 3 nodos habilitamos la autenticación, pero después al comprobar el estado nos pedirá una key.

```

GNU nano 6.2                               /etc/mongod.conf
# network interfaces
net:
port: 27017
bindIp: 127.0.0.1,10.2.4.10

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

security:
  authorization: enabled

#operationProfiling:

replication:
  replSetName: "rs0"

#sharding:

```

En el nodo primario generamos esa key y después la copiamos a los secundarios.

```

openssl rand -base64 756 > mongod.key
sudo mv mongod.key /etc/mongod.key
sudo chown mongodb:mongodb /etc/mongod.key
sudo chmod 400 /etc/mongod.key

```

```

Armando viernes 7 noviembre 2025 09:42
[usuario@server2asir ~]$openssl rand -base64 756 > mongod.key
Armando viernes 7 noviembre 2025 09:43
[usuario@server2asir ~]$sudo mv mongod.key /etc/mongod.key
Armando viernes 7 noviembre 2025 09:43
[usuario@server2asir ~]$sudo chown mongodb:mongodb /etc/mongod.key
Armando viernes 7 noviembre 2025 09:43
[usuario@server2asir ~]$sudo chmod 400 /etc/mongod.key
Armando viernes 7 noviembre 2025 09:43
[usuario@server2asir ~]$

```

```
Armando viernes 7 noviembre 2025 10:17
[usuario@server2asir ~]$sudo chmod 700 /etc/mongod.key
Armando viernes 7 noviembre 2025 10:17
[usuario@server2asir ~]$ls -l /etc/mongod.key
-rwx----- 1 mongodb mongodb 1024 nov  7 09:43 /etc/mongod.key
Armando viernes 7 noviembre 2025 10:17
[usuario@server2asir ~]$
```

Volvemos a modificar el mongod.conf ahora añadiendo la fila keyFile.

```
GNU nano 6.2                                     /etc/mongod.conf

net:
  port: 27017
  bindIp: 127.0.0.1,10.2.4.10

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

security:
  keyFile: /etc/mongod.key
  authorization: enabled

#operationProfiling:

replication:
  replSetName: "rs0"
```

En los 3 nodos habilitamos el RootLogin y establecemos la contraseña.

```
GNU nano 6.2                                     /etc/ssh/sshd_config

# Authentication:
#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

```
sudo systemctl restart ssh.service
sudo systemctl status ssh.service
```

```
sudo passwd root
```

```
Armando viernes 7 noviembre 2025 10:02
[usuario@server2asir ~]$sudo passwd root
New password:
Retype new password:
passwd: password updated successfully
Armando viernes 7 noviembre 2025 10:02
[usuario@server2asir ~]$
```

Ahora en el nodo primario copiamos el archivo con el siguiente comando:

```
sudo scp /etc/mongod.key usuario@10.2.4.20:/tmp/  
sudo scp /etc/mongod.key usuario@10.2.4.30:/tmp/
```

```
Armando viernes 7 noviembre 2025 10:17  
[usuario@server2asir ~]$sudo scp /etc/mongod.key usuario@10.2.4.20:/tmp/  
The authenticity of host '10.2.4.20 (10.2.4.20)' can't be established.  
ED25519 key fingerprint is SHA256:SZwXLmx+wUdcSP8wh7BqQ/a59GCadpfJN5Fuh/1UA/Q.  
This key is not known by any other names  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '10.2.4.20' (ED25519) to the list of known hosts.  
usuario@10.2.4.20's password:  
mongod.key 100% 1024  
Armando viernes 7 noviembre 2025 10:18  
[usuario@server2asir ~]$
```

Y en los nodos secundarios ponemos los siguientes comandos:

```
sudo mv /tmp/mongod.key /etc/mongod.key  
sudo chown mongodb:mongodb /etc/mongod.key  
sudo chmod 400 /etc/mongod.key
```

```
Armando viernes 7 noviembre 2025 10:19  
[usuario@server2asir ~]$sudo mv /tmp/mongod.key /etc/mongod.key  
[sudo] password for usuario:  
Armando viernes 7 noviembre 2025 10:19  
[usuario@server2asir ~]$sudo chown mongodb:mongodb /etc/mongod.key  
Armando viernes 7 noviembre 2025 10:19  
[usuario@server2asir ~]$sudo chmod 400 /etc/mongod.key
```

Y editamos el archivo mongod.conf añadiendo la fila keyFile

```
GNU nano 6.2 /etc/mongod.conf  
net:  
port: 27017  
bindIp: 127.0.0.1,10.2.4.20  
  
# how the process runs  
processManagement:  
timeZoneInfo: /usr/share/zoneinfo  
  
security:  
keyFile: /etc/mongod.key  
authorization: enabled  
  
#operationProfiling:  
  
replication:  
replSetName: "rs0"
```

```
sudo systemctl restart mongod  
sudo systemctl status mongod
```

Por último verificamos la autenticación.

```
# Conectar con usuario admin
mongosh -u "admin" -p "admin123" --authenticationDatabase "admin"
```

```
db.runCommand({ connectionStatus: 1 })
```

```
Armando domingo 9 noviembre 2025 15:38
[usuario@server2asir ~]$mongosh -u "admin" -p "admin123" --authenticationDatabase "admin"
Current Mongosh Log ID: 6910b5805853f787439dc29c
Connecting to: mongodb://<credentials>@127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&authSource=admin&appName=mongosh+2.5.9
Using MongoDB: 6.0.26
Using Mongosh: 2.5.9

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

-----
The server generated these startup warnings when booting
2025-11-09T15:28:48.733+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
2025-11-09T15:28:51.011+00:00: vm.max_map_count is too low
```

```
rs0 [direct: primary] test> db.runCommand({ connectionStatus: 1 })
{
  authInfo: {
    authenticatedUsers: [ { user: 'admin', db: 'admin' } ],
    authenticatedUserRoles: [ { role: 'root', db: 'admin' } ]
  },
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1762702653, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('r369LFN1m8AGT55hQcgrvYlAtFU='), 0,
      keyId: Long('7567329206971924485')
    }
  },
  operationTime: Timestamp({ t: 1762702653, i: 1 })
}
rs0 [direct: primary] test>
```

#### 4.- Configura las siguientes características de red:

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

Limitamos el número de conexiones simultáneas en cada nodo. MongoDB no permite valores inferiores a 5.

```
GNU nano 6.2                                     /etc/mongod.conf *
# network interfaces
net:
  port: 27017
  bindIp: 127.0.0.1,10.2.4.10
  maxIncomingConnections: 5
```

```
sudo systemctl restart mongod
```

- 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.

Volvemos a editar el mismo archivo y quitamos la línea que pusimos antes.

```
GNU nano 6.2                                         /etc/mongod.conf *
# network interfaces
net:
  port: 27017
  bindIp: 127.0.0.1,10.2.4.10
```

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

**Añade un documento con toda la configuración, los cambios, pantallazos y explicaciones.**

En el nodo primario entramos a MongoDB, creamos la base de datos y colección “capped” e insertamos documentos.

```
mongosh -u "admin" -p "admin123" --authenticationDatabase "admin"
```

```
use pruebas
```

```
db.createCollection("logs", { capped: true, size: 10240, max: 5 })
```

```
Armando domingo 9 noviembre 2025 15:57
[usuario@server2asir ~]$mongosh -u "admin" -p "admin123" --authenticationDatabase "admin"
Current Mongosh Log ID: 6910ba019f932e65ff9dc29c
Connecting to:          mongdb://<credentials>@127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&authSource=admin&appName=mongosh+2.5.9
Using MongoDB:          6.0.26
Using Mongosh:          2.5.9
For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/
-----
The server generated these startup warnings when booting
2025-11-09T15:54:47.042+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
2025-11-09T15:54:48.518+00:00: vm.max_map_count is too low
-----
rs0 [direct: primary] test> use pruebas
switched to db pruebas
rs0 [direct: primary] pruebas> db.createCollection("logs", { capped: true, size: 10240, max: 5 })
{ ok: 1 }
rs0 [direct: primary] pruebas>
```

```
for (let i = 1; i <= 10; i++) {
  db.logs.insertOne({ mensaje: "Documento " + i, fecha: new Date() })
}
rs0 [direct: primary] pruebas> for (let i = 1; i <= 10; i++) {
...   db.logs.insertOne({ mensaje: "Documento " + i, fecha: new Date() })
...
{
  acknowledged: true,
  insertedId: ObjectId('6910ba499f932e65ff9dc2a6')
}
```

```
db.logs.find()
rs0 [direct: primary] pruebas> db.logs.find()
[
  {
    _id: ObjectId('6910ba499f932e65ff9dc2a2'),
    mensaje: 'Documento 6',
    fecha: ISODate('2025-11-09T15:59:05.495Z')
  },
  {
    _id: ObjectId('6910ba499f932e65ff9dc2a3'),
    mensaje: 'Documento 7',
    fecha: ISODate('2025-11-09T15:59:05.509Z')
  },
  {
    _id: ObjectId('6910ba499f932e65ff9dc2a4'),
    mensaje: 'Documento 8',
    fecha: ISODate('2025-11-09T15:59:05.521Z')
  },
  {
    _id: ObjectId('6910ba499f932e65ff9dc2a5'),
    mensaje: 'Documento 9',
    fecha: ISODate('2025-11-09T15:59:05.532Z')
  },
  {
    _id: ObjectId('6910ba499f932e65ff9dc2a6'),
    mensaje: 'Documento 10',
    fecha: ISODate('2025-11-09T15:59:05.543Z')
  }
]
rs0 [direct: primary] pruebas>
```