# Enable MongoDB Authentication

By default, FileCloud installs the Mongo database server on the same machine as the web server without any authentication settings.

However, you may need to enable authentication for the following reasons:

- Added security
- Hosting the database server on a different machine than the web server.

Follow the steps here to enable authentication for MongoDB.

## Set Up a Database User

A DB user has to be first created in MongoDB and this user can be later used in FileCloud for secure database access.
Assuming we will add a user with following details:

| User Name | Password |
|---|---|
| dbuser | passw0rd1 |

Use a command line mongo client and execute the following commands to create the required DB user.

The following command lists all the databases in the system (depending on the configuration one or more dbs may not exist (or new ones may be present). So it is important to set authentication for each of the DB in the system. (Ignore the "local" database that shows up when you type "show databases")

**For MongoClient v3.0 and above**

```
use admin
db.createUser( { user:"dbuser", pwd:"passw0rd1", roles:[ "root" ] })
```

For Mongo Client v 2.4

**Mongo Client**

```
> show databases
admin             0.078GB
tonidoauditdb     0.078GB
tonidoclouddb     0.078GB
tonidos3storage   0.078GB
tonidosettings    0.078GB
tonidostoragedb   0.078GB
tonidosyncdb      0.078GB

> use admin;
> db.addUser('dbuser','passw0rd1')
> use tonidoauditdb;
> db.addUser('dbuser','passw0rd1')
> use tonidoclouddb;
> db.addUser('dbuser','passw0rd1')
> use tonidostoragedb;
> db.addUser('dbuser','passw0rd1')
> use tonidosyncdb;
> db.addUser('dbuser','passw0rd1')
> use tonidosettings;
> db.addUser('dbuser','passw0rd1')
```

Upon executing all the above commands, 'dbuser' will be added as valid database user.

# Configure Settings DB URL

FileCloud's settings database is where all the information is bootstrapped from. The default implicit URL for this database is "mongodb://127.0.0.1". Set this URL explicitly to reflect the fact that a database user needs to be used and the database server resides on different server. To do this, edit the configuration file WWWROOT/config/cloudconfig.php and add the following line:

**Override Settings DB URL**

```
define("TONIDOCLOUD_SETTINGS_DBSERVER", "mongodb://dbuser:passw0rd1@192.168.1.102:27017");
```

In the above example, we assumed the database server is installed on a different machine (i.e., 192.168.1.102) than the webserver. In collocated scenarios, 127.0.0.1 can be used as well.

Note: If you use special characters in the password, make sure to URI encode them. For example: using 'password@2090' as the password, you will need to specify it like

> ⓘ mongodb://dbuser:password%402090@localhost:27017

## Configure Other DB URLs In Config File

If you have never updated the database URLs in the admin UI, follow this sub-section. If not, skip to the next sub-section.

Other database URLs required for FileCloud needs to be changed to reflect the database user as well.
To do this, edit the configuration file WWWROOT/config/cloudconfig.php and update the following lines:

**Update DB URLs in cloudconfig.php**

```
// ... Cloud Database
define("TONIDOCLOUD_DBSERVER", "mongodb://dbuser:passw0rd1@192.168.1.102:27017");
// ... Audit Database
define("TONIDOCLOUD_AUDIT_DBSERVER", "mongodb://dbuser:passw0rd1@192.168.1.102:27017");
// ... Settings Database
define("TONIDOCLOUD_SETTINGS_DBSERVER", "mongodb://dbuser:passw0rd1@192.168.1.102:27017");
```

and configuration file WWWROOT/config/localstorageconfig.php and update the following line:

**Update DB URLs in localstorageconfig.php**

```
// ... Cloud Database
define("TONIDO_LOCALSTORAGE_DBSERVER", "mongodb://dbuser:passw0rd1@192.168.1.102:27017");
```

## Configure Other DB URLs In Settings DB

If you have updated the database URLs in the admin UI, then changing the values in the config files as described above will not work.

In this case use a mongodb client and update the URLs with the following information.

**Update settings database with a mongo client**

```
Database: tonidosettings
Collection: sites
Records:  {
        "name" : "TONIDOCLOUD_DBSERVER",
        "value" : "mongodb://dbuser:passw0rd1@192.168.1.102:27017"
        }, {
        "name" : "TONIDOCLOUD_AUDIT_DBSERVER",
        "value" : "mongodb://dbuser:passw0rd1@192.168.1.102:27017"
        }, {
        "name" : "TONIDO_LOCALSTORAGE_DBSERVER",
        "value" : "mongodb://dbuser:passw0rd1@192.168.1.102:27017"
}
```

## Enable MongoDB Security

Now that FileCloud is updated with the security info, enable security in MongoDB. To do this open the file mongodb.conf that can be typically found in the following location:

| | |
|---|---|
| **Windows** | C:\xampp\mongodb\bin\mongodb.conf |
| **Linux** | /etc/mongodb.conf |

Edit this file and add/update with the following line. If the line is already there, ensure it is not commented.

**Enable mongodb security in Windows and mongodb v2.x on Linux**

```
# Turn on/off security.  Off is currently the default
#noauth = true
auth = true
```

If you are using a version of MongoDB that creates a YAML conf file, you might need to enable authentication using the following format.

**Enable mongodb v3.x on Linux**

```
security:
  authorization: enabled
```

# Restart Services

Finally, it is necessary to restart both MongoDB and Apache to get the security in-place.

⚠ **Note**

- In case of any issues, disable security in mongodb and fix the problems.
- To disable security, mongodb auth has to be disabled and the database URLs has to be reverted back.