# ODBC authentication

**Last updated:** Mon, 26 Mar 2018 20:50:53 GMT

## Overview

EZproxy for Windows supports the ability to authenticate users through an ODBC connection to a database server. Microsoft Windows provides ODBC as a standard method for programs to communicate. The Windows version of EZproxy can authenticate and authorize users for access using SQL commands.

## Configuration

The following is a sample user.txt configuration that incorporates all possible directives.

```plaintext
::ODBC
Debug
DSN SomeSystemDSN
DBUser SomeUser
DBPassword SomePassword
SQL USE SomeDatabase
| Parameter User
| Parameter Password
SQL
SELECT 'Allow'
FROM auth
WHERE
user = ? AND
pass = ?
/ODBC
```

Debug is an optional directive which directs EZproxy to record additional diagnostic messages to messages.txt. Debug is normally used only while developing or diagnosing SQL configurations.

DSN is the ODBC System Data Source Name (DSN) to use. The DSN must be set up separately. If EZproxy is running on a 64-bit version of Windows, you must have the 32-bit database driver installed and set up the DSN as 32-bit using either the "ODBC Data Sources (32-bit)" control panel tool, or if that tool does not appear, by running the 32-bit control panel manually from C:\Windows\SysWow64\odbcad32.exe. Consult your database administrator for additional guidance on how to set up a system DSN.

DBUser and DBPassword are optional. If included, they provide the username and password that EZproxy should use to access the database.

SQL is discussed in more detail further down this page. In this example, SQL appears twice: first, to establish the default database that is being used, and second, to perform the actual user verification.
Parameter may be followed by User, Password, or IP and indicate values that should be supplied for each ? that appears in the SQL statement. The first Parameter value goes to the first ? in the SQL statement, the second Parameter to the second ?, and so forth. Parameter should appear just prior to the SQL directive it is intended to affect.

SQL is followed by an SQL statement. Since SQL statements may become quite long, you may continue SQL statements across multiple lines by ending each line with a \ character. The SQL statement should be constructed to return the literal Allow if the user is to be allowed access, Deny if the user should be denied all access to EZproxy. If the first value returned is neither Allow or Deny, EZproxy moves on to the next authentication check in user.txt.

For Allow, the SQL statement may also return a second column that indicates one or more EZproxy groups to which the user should have access. To use the group feature, the query should return several rows with one group per row, such as:

<table>
<thead>
<tr>
<th>Allow</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow</td>
<td>Medical</td>
</tr>
<tr>
<td>Allow</td>
<td>Legal</td>
</tr>
</tbody>
</table>

For Deny, the SQL statement may also return a second column that indicates the name of the file from the docs directory that should be sent to the user who is being denied access. To use this feature, the query should return a something like this:

| Deny  | alumni.html |

In addition to the directives shown, the ODBC block may also incorporate any of the common conditions and actions.