MsSQL: Run CzechIdM with MsSQL database

sqlserver, sql, mssql, install, database, supported, drivers

The tutorial describes how to run CzechIdM with a Microsoft SQL Server. When following the tutorials Server preparation and CzechIdM installation please do skip all the settings and setup related to database.

The tutorial doesn't describe how to install an MsSQL database. For development you can install MsSQL in a docker, see the section MsSQL in a docker.

Don't forget to run these queries:

ALTER DATABASE bcv_idm_storage SET READ_COMMITTED_SNAPSHOT ON;

ALTER DATABASE bcv_idm_storage SET ALLOW_SNAPSHOT_ISOLATION ON;

Otherwise you risk deadlock on database.

Setup MsSQL with CzechIdM

Create database

There is a script for setting up a database for CzechIdM. Please follow these instructions. If you omit any of the statements, CzechIdM might not work properly.

```
-- create database
CREATE DATABASE bcv_idm_storage;
GO
-- set read committed snapshot
ALTER DATABASE bcv_idm_storage SET READ_COMMITTED_SNAPSHOT ON;
GO
-- allow snapshot isolation - setting up a lock escalation
ALTER DATABASE bcv_idm_storage SET ALLOW_SNAPSHOT_ISOLATION ON;
GO
-- set active database
USE bcv_idm_storage;
GO
-- create login, login is used for connecting to the server, check policy is
for development purposes (password idmadmin)
CREATE LOGIN idmadmin WITH PASSWORD = 'idmadmin', check_policy = off;
```

```
GO
-- create a user, this user will be used to connect to the database
CREATE USER idmadmin FOR LOGIN idmadmin;
GO
-- create a schema
CREATE SCHEMA bcv_idm_storage AUTHORIZATION idmadmin;
GO
-- set a default schema
ALTER USER idmadmin WITH DEFAULT_SCHEMA = bcv_idm_storage;
GO
-- grant permission for the schema (grant all is deprecated)
-- sometimes this is not needed when the user is owner of the schema - then
you get 'Cannot grant, deny, or revoke permissions to sa, dbo, entity owner,
information_schema, sys, or yourself.' which is OK
GRANT ALTER, CONTROL, CREATE SEQUENCE, DELETE, EXECUTE, INSERT, REFERENCES,
SELECT, TAKE OWNERSHIP, UPDATE, VIEW CHANGE TRACKING, VIEW DEFINITION ON
SCHEMA::bcv_idm_storage TO idmadmin;
GO
-- grant create table to idmadmin
GRANT CREATE TABLE TO idmadmin;
GO
-- grant create view to idmadmin
GRANT CREATE VIEW TO idmadmin;
GO
```

If you use a domain user to connect to the database, use this syntax: `CREATE SCHEMA bcv\_idm\_storage AUTHORIZATION "DOMAIN\idmadmin;"`.

If your domain user has a very long username (e.g. "sql_dbinstance'_czechidm", then you must supply only the shorter sAMAccountName, typically first 20 characters - e.g."sql_dbinstance_czech").

Download a JDBC driver

Please download a JDBC driver for MsSQL. For example: driver from official Microsoft Docs, or http://clojars.org/repo/com/microsoft/sqlserver/sqljdbc4/4.0/sqljdbc4-4.0.jar, or see the section other supported drivers.

The driver must be placed for example into */opt/tomcat/current/lib/* tomcat external classpath, or libraries of tomcat. If CzechIdM runs on Windows, the path is *C:\Program Files\Apache Software Foundation\Tomcat 8.5\lib*.

Choose the type of authentication and set connection properties (application.properties)

The configuration depends on the type of authentication, which you will use for connecting to the

CzechIdM database. There are several options:

- SQL Server Authentication - this can be used with local DB accounts
- Windows Integrated Authentication - this can be used with domain user accounts (who have granted permissions to access the DB) when CzechIdM runs on Windows.

SQL Server Authentication

The correct data source url and password are required. To see an example, visit the github page with developer profile for mssql.

Example settings:

```
spring.datasource.url=jdbc:sqlserver://localhost;databaseName=bcv_idm_storag
e
spring.datasource.username=idmadmin
spring.datasource.password=idmadmin
spring.datasource.driver-class-
name=com.microsoft.sqlserver.jdbc.SQLServerDriver
spring.datasource.test-on-borrow=true
spring.datasource.validationQuery=SELECT 1
```

An example is valid for JDBC driver from official Microsoft Docs or Sqljdbc4 4.0, if you want to use another driver please setup the correct url.

If you need to use a domain user, set the property username like this: `DOMAIN\\idmamin`. Java will translate the double backslash to a single backslash. If you use only single backslash, it will not be sent to the SQL Server at all and the SQL Server will not find the user. Also, if you have too long login, use its shorter version (sAMAccountName). Also you should probably set `authentication=ActiveDirectoryIntegrated`, but it hasn't been truly tested yet. Fix Me!

Windows Integrated Authentication

This type of authentication can be used for a domain user account. It requires a bit more configuration, but the main advantage is that the password of the domain user is not directly written in the application properties, so it's more secure and recommended by Microsoft. Connecting with integrated authentication On Windows.

This authentication is not supported by Microsoft JDBC drivers prior to the version 6.

Steps to setup:

1. Set the **Apache Tomcat8 service** to **Log on as** the domain user which you will use to connect to the database (instead of "Local Service" recommended by the Server preparation tutorial).

2. Grant write access to the Tomcat directory (C:\Program Files\Apache Software Foundation\Tomcat 8.5) to the domain user (or simply Full control).

3. Extract the file `sqljdbc\_8.2\enu/auth/x86/mssql-jdbc_auth-8.2.2.x86.dll` from the downloaded Microsoft driver. (Use the same package which you already downloaded when

downloading JDBC driver in the previous part.)

4. Put `mssql-jdbc\_auth-8.2.2.x86.dll` to `C:\CzechIdM\lib`.

5. Add the option `-Djava.library.path=C:\CzechIdM\lib` to the **Tomcat Java Options**. To do it, run the Monitor Tomcat application from the Start menu (or run Tomcat8w.exe from the Tomcat bin directory - C:\Program Files\Apache Software Foundation\Tomcat 8.5\bin) → Java → Java Options.

6. Set the `C:\CzechIdM\etc\application-production.properties` to use Integration Authentication (the property integratedSecurity).

```
spring.datasource.url=jdbc:sqlserver://xsqlserver123;databaseName=bcv_idm_st
orage;instanceName=CZECHIDM;integratedSecurity=true
spring.datasource.driver-class-
name=com.microsoft.sqlserver.jdbc.SQLServerDriver
spring.datasource.test-on-borrow=true
spring.datasource.validationQuery=SELECT 1
```

The example is valid for SQL server running on the server "xsqlserver123", instance "CZECHIDM", database "bcv_idm_storage". Note that the parameters username and password are omitted; this is required for this type of authentication.

7. Add SQL server certificate to Java truststore (Fix Me! how?). Workaround: add property `trustServerCertificate=true` to the JDBC URL above. </note>

8. Finally, restart the Apache Tomcat8 service so all changes take place.

Windows Authentication with NTLM

If you need to use Windows Authentication but can't use the integrated authentication as above (e.g. you are not running IdM on Windows), it's possible to use NTLM authentication. You will explicitly set username and password.

Example properties:

```
spring.datasource.url=jdbc:sqlserver://xsqlserver123\CZECHIDM:1433;databaseN
ame=bcv_idm_storage;integratedSecurity=true;authenticationScheme=NTLM;domain
=yourdomain.tld
spring.datasource.username=someserviceuser
spring.datasource.password=somepassword
spring.datasource.driver-class-
name=com.microsoft.sqlserver.jdbc.SQLServerDriver
spring.datasource.test-on-borrow=true
spring.datasource.validationQuery=SELECT 1
```

The example is valid for SQL server running on the server "xsqlserver123", instance "CZECHIDM", database "bcv_idm_storage", domain "yourdomain.tld". Note that you don't specify the domain in the username.

Scheduler setup (quartz.properties)

For full example please visit the github page with developer profile.

There are two propeties that are different:

```
org.quartz.jobStore.driverDelegateClass=org.quartz.impl.jdbcjobstore.MSSQLDelegate
org.quartz.jobStore.tablePrefix=QRTZ_
```

Supported version of MsSQL

- **14.0** - 2017 SQL Server 2017
  - Tested version: 14.0.3294.2
  - Unsupported version: 14.0.1000.169 - contains a bug, which causes failure of the task DeleteExecutedEventTaskExecutor with the following error: `SQL Error: 21, SQLState: S0001. Warning: Fatal error 605 occurred`, event viewer: `Attempt to fetch logical page (1:604718) in database 5 failed. It belongs to allocation unit 72057594065387520 not to 72057594089439232.`

Supported drivers

- Sqljdbc4 4.0,
- Microsoft JDBC Driver For SQL Server 6.4.0.jre8,
- Microsoft JDBC Driver for SQL Server - official Microsoft Docs - tested version 8.2.2
- *Not fully tested* jTDS - SQL Server and Sybase JDBC driver.

Not supported drivers

- Sqljdbc4 4.0.0 (this is version 4.0.0 not 4.0)

Develop CzechIdM with MsSQL and a docker

Beware this is recommended only for develop.

Run lasted MsSQL database (change <SA-PASSWORD> with your password):

```
 $ docker run --name=test-mssql -e 'ACCEPT_EULA=Y' -e 'SA_PASSWORD=<SA-PASSWORD>' -p 1433:1433 -d microsoft/mssql-server-linux:latest
```

Copy the initial script (init script is described above. Just copy and create file with defined sql queries, in our example the name of the script file is import.sql)

```
 $ docker cp import.sql test-mssql:/import.sql
```

Run the init script in the docker (change SA-PASSWORD):

```
 $ docker exec test-mssql /opt/mssql-tools/bin/sqlcmd -S localhost -U sa -P
<SA-PASSWORD> -d master -i /import.sql
```

Use docker-compose

You can also use the following docker-compose.yml file. The advantage is that it uses persistent volumes and docker-compose cleans after itself better. Copy and edit (if needed) the code below to a file called 'docker-compose.yml':

```
version: "3.2"
services:
  sql-server-db:
    container_name: sql-server-db
    image: microsoft/mssql-server-linux:2017-latest
    ports:
      - "1433:1433"
    environment:
      SA_PASSWORD: "Password123456"
      ACCEPT_EULA: "Y"
      MSSQL_BACKUP_DIR: "/var/opt/sqlserver"
      MSSQL_DATA_DIR: "/var/opt/sqlserver"
      MSSQL_LOG_DIR: "/var/opt/sqlserver"
    volumes:
      - 'systemdbs:/var/opt/mssql'
      - 'userdbs:/var/opt/sqlserver'
volumes:
  systemdbs:
  userdbs:
```

Then, in the same directory, use the command `docker-compose up` to start the database.

Troubleshooting

Increase log level

For debugging the problems when connecting to SQL Server database from CzechIdM, it's useful to turn on the Debug mode of the SQL Server JDBC Driver. Set C:\Program Files\Apache Software Foundation\Tomcat 8.5\conf\logging.properties like this:

```
1.catalina.out.org.apache.juli.AsyncFileHandler.level = FINEST (instead of
the original INFO)
```

```
com.microsoft.sqlserver.jdbc.level=FINEST
```

After this, the file C:\Program Files\Apache Software Foundation\Tomcat 8.5\conf\tomcat.log shows detailed information about processing the different connection parameters and its results.

Make sure to change the settings back after you finish debugging. It generates **really big** log files.

Event Viewer

If you have access to the Event Viewer on the server running the SQL Server, you will see the connection attempts under Windows Logs → Application. The log level is Information even for unsuccessful atempts. You can see e.g.:

- if SQL server knows the account, which is used for connecting
- which type of authentication is used - e.g. if you use a domain account ("NT account") in combination with SQL Server authentication made for local accounts.

Instances and ports

Usually, you don't need to specify the port when connecting to SQL Server. By default the driver calls built-in SQL Server Browser service and obtains the dynamic port, where the DB instance runs. This works also when there is only a default instance.

The default SQL Server port (1433) doesn't need to be accessible through the network, so don't use it in the connection URL if you don't need to.