Table of Contents

If you use the DBSecurityManager for authentication, or if you use a database in your application, the configuration is in your applications config.xml.

The configuration is plain xml, and it's possible to use encryption. It's also possible to read the configuration via JNDI.

Another option is to use JNDI to create a database connection. In this case, you should set the JDBC URL to something like this:

config.xml

```
<application>
  ...
  <securitymanager>
    <class>com.sibvisions.rad.server.security.DBSecurityManager</class>
    <database>
      <url>java:/comp/env/jvx/myapp/config/db</url>
    </database>
  </securitymanager>
  ...
</application>
```

The JNDI resource can be an instance of `DBAccess`, `IConnectionPool`, `java.sql.Connection` or `javax.sql.DataSource`.

The application code, e.g., in your Session LCO:

```
protected DBAccess getDBAccess() throws Exception
{
    DBAccess dba = (DBAccess)get("dBAccess");

    if (dba == null)
    {
        DBCredentials dbcred =
DBSecurityManager.getCredentials(SessionContext.getCurrentSessionConfig());

        //no credentials -> no database access possible (should not happen,
only if configuration is not finished)
        if (dbcred != null)
        {
            dba = DBAccess.getDBAccess(dbcred);
            dba.open();

            put("dBAccess", dba);
        }
    }

    return dba;
}
```

will work without changes.

From:
https://doc.sibvisions.com/ - **Documentation**

Permanent link:
https://doc.sibvisions.com/jvx/server/security/db_jndi

Last update: **2024/11/18 10:33**