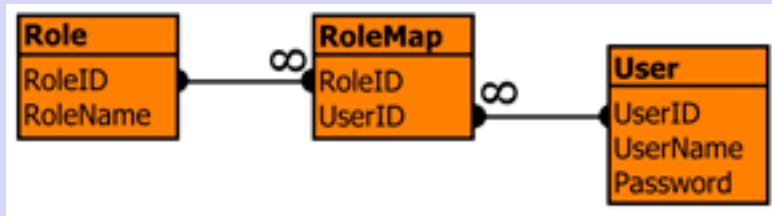


AfterthoughtSoft-Secure Pro Edition

If you haven't read it already, please read the introduction that is part of the description of the [Community Edition](#).

The Pro Edition picks up where the Community Edition leaves off and adds support for authenticating against existing Windows NT/XP, Unix and ANY JDBC compliant database domains. The default setting for JDBC in the Pro edition is set to authenticate against a MySQL database by default but any valid JDBC database will work (the JDBC settings have been tested against Oracle for example).

The Entity Relationship Diagram (ERD) that describes the tables you need and their relationships is shown below:



Ok, what if you have an EXISTING set of tables that contain user / role information? No problem! As long as you can project some read-only virtual tables from your existing information in the form shown above, you should be good to go. Some sample SQL to get started putting users into a mysql database is shown below given the above ERD is:

```
drop table Role;
drop table User;
drop table RoleMap;
create table Role (RoleID INT(10) auto_increment not null primary key, RoleName varchar(50));

create table User (UserID INT(10) auto_increment not null primary key, UserName varchar(50),
Password varchar(150));

create table RoleMap (RoleID INT(10), UserID INT(10), INDEX role_ind (RoleID), INDEX user_ind
(UserID), FOREIGN KEY (RoleID) REFERENCES Role(RoleID), FOREIGN KEY (UserID)
REFERENCES User(UserID));

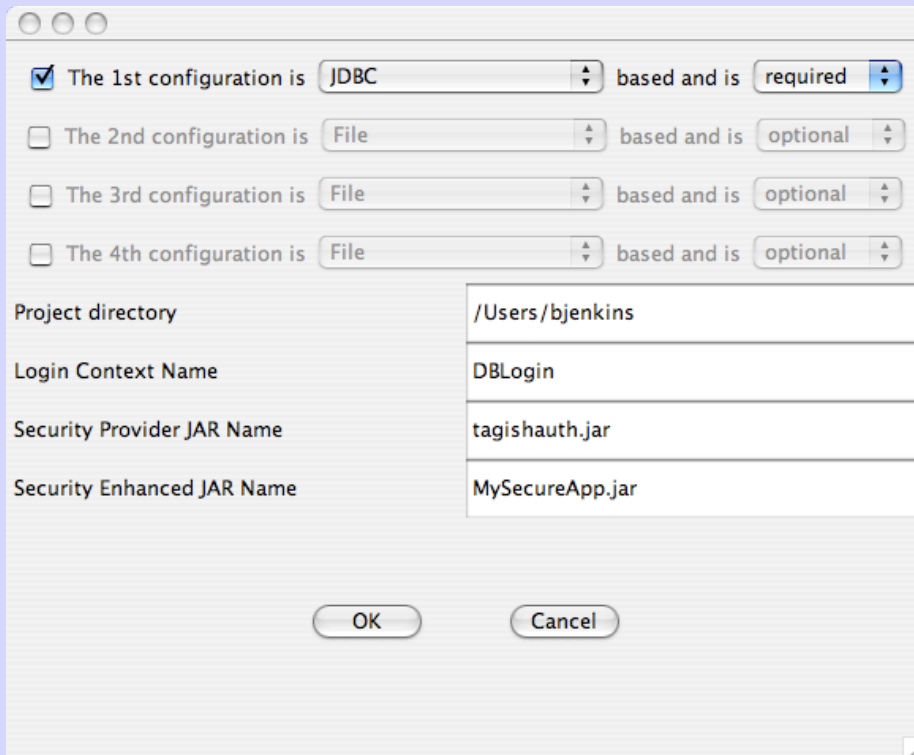
insert into User (UserName, Password) values ("user1","24c9e15e52afc47c225b757e7bee1f9d");
insert into Role (RoleName) values ("root");
insert into Role (RoleName) values ("administrator");
insert into User (UserName, Password) values ("user2","7e58d63b60197ceb55a1c487989a3720");
insert into Role (RoleName) values ("group2");
insert into RoleMap (RoleID, UserID) values (1,1);
insert into RoleMap (RoleID, UserID) values (2,1);
insert into RoleMap (RoleID, UserID) values (3,2);
insert into User (UserName, Password) values ("bjenkins","e34a61aadb70ef57235af7588e3222c");
insert into RoleMap (RoleID, UserID) values (3,3);
```

(Note: The passwords are in MD5 hashed format.)

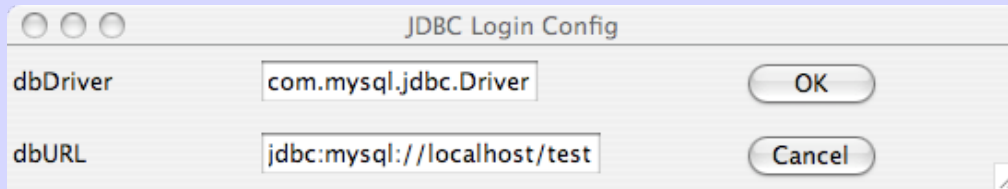
To use JDBC based RBAC security for your application, launch the AfterthoughtSoft-Secure application...



choose the preferences screen, and select a configuration of "JDBC"



This will pull up the JDBC preferences dialog where you can enter the driver class information for you JDBC implementation. The default is a MySQL implementation (I use MySql 5.0) with a database instance called "test" on my same host. So the driver information would be like the following:



(Please refer to the respective database vendor's site for details on the settings for "driver" and "URL")

This will give you a `.java.login.config` file with the above driver and url information...

```
DBLogin
{
  com.tagish.auth.DBLogin required dbDriver="com.mysql.jdbc.Driver"
  dbURL="jdbc:mysql://localhost/test";
};
```

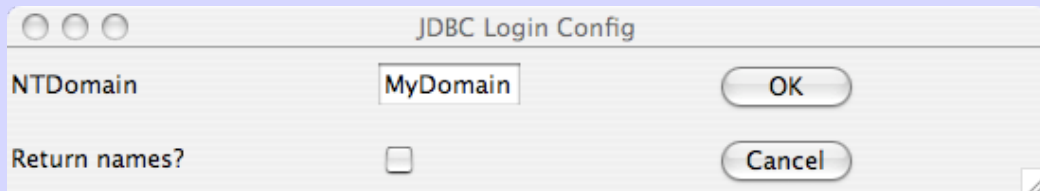
and a corresponding `.java.policy` file where the only material difference from what was produced with the community edition is the type of principal...

```
principal com.tagish.auth.TypedPrincipal "bjenkins"
```

and that is it!

Windows NT/XP domain

The preferences dialog for working in a Windows NT / XP domain is dirt simple. All you need to know is the NT domain of which you are a member and enter that into the NTDomain text box as shown below:



This will place the following entry in the JAAS security policy file:

```
principal com.sun.security.auth.NTUserPrincipal "bjenkins"
```

(note: where "bjenkins" will be replaced with the name of the individual or group you chose earlier)

and the `.java.login.config` file entry will look like the following:

```
com.sun.security.auth.module.NTLoginModule required returnNames=false returnSIDs=false
```

```
defaultDomain="MyDomain";
```

UNIX Domain

Currently nothing more needs to be done to support Unix! Simply choose the Unix option and you will get the following entry in your .java.policy security file:

```
principal com.sun.security.auth.UnixPrincipal "bjenkins"
```

(note: where “bjenkins” will contain the name of the user you entered in the tool)

For even more advanced RBAC environments, you might wish to try the Enterprise Edition. The Enterprise Edition has everything found in the Community and Pro Editions and adds support for LDAP and Kerberos V.