SQL Authentication Protocol

This document describes the **ViziApps Authentication Process for Mobile App Access to SQL Databases**.

The authentication process for mobile app access to SQL databases relies on access tokens that are reset with each mobile app session. A mobile app session starts when the app is opened on the device and ends when there is an activity timeout. A special login request is made using credentials that users enter in the app, which returns the session access token used for all subsequent app SQL queries during the app session. The special login request is protected against a SQL injection attack on the server side and the use of the one-time access token per app session required for standard SQL queries prevents various other attacks.

Prerequisites

1. ViziApps SQL .NET agent service software is installed behind the customer's firewall.

Design Time in Studio Designer

1. SQL data source is registered in the ViziApps Studio .
2. When the data management design of queries is setup, besides the options of select, insert, update and delete there will be another query type called "login from" with a mapping between a server variable "IsLoginValid" and a mobile app field with a response value of "true" or "false". The designer chooses the table and 2 fields in the WHERE clause for the username and password mapping from the mobile app to the database fields in the chosen database table.

Run Time in Mobile App

1. If the query type is a "login from", then special processing is done on the server side:
   - Login queries are only allowed from the ViziApps Studio or ViziApps mobile apps running on a mobile device.
   - No session access token is needed
   - The username and password values in the request are filtered against single and double quotes at the customer server to avoid an SQL injection attack.
   - For a correct login, where the username and password fields match those in the database, the session access token will be created on the server and be returned to the mobile app and stored in the device keychain as well as stored in the customer server cache with a datetime stamp. The session access tokens have a 90 minute session timeout configurable in the web.config file, so that old session tokens will be automatically removed on the server.

- If a session access token is returned, then the mobile app field mapped to “IsLoginValid” in the login query is set to “true”. Other it is set to “false”. This allows the app to respond with message to the user or other remedial action.

2. Other than the query type “login from”, all other SQL queries require a session access token to execute. If the access token does not exist in the device keychain, a warning message is displayed as an alert saying “Session Access Token does not exist”.
3. If the session access token exists in the mobile app, its value will automatically be used for any SQL query except for the “login from” query described in step 3. When the customer server gets the SQL query request, the session access token is compared to existing session access tokens in the customer server cache. If the session token does not exist in the server cache, including timed out session access tokens, the query will not be executed and instead an error message will be returned: “Bad session token”. If the session token does exist, the query will be executed and the session token datetime stamp will be updated.
4. The SQL statements are all in clear text sent from the mobile app to the customer server via SSL. Since the session access token changes with every session and is required to be sent along with the SQL statements, various attacks are prevented.