Connection Pool How to Guide

Connection Pool How to guide for PINsafe

Overview

This guide covers the use of the Connection Pool.

By default, PINsafe creates a new database connection every time one is required. The connection is then closed when it is no longer required, but the underlying connection manager holds the connection for several minutes (2 minutes by default) before releasing it for re-use. When database connection pooling is enabled in PINsafe, rather than releasing the connection, PINsafe retains it for re-use, and when a new connection is required, it re-uses an open one if there is one available, in preference to creating a new one. This avoids the overhead involved in opening a new connection. It also limits the maximum number of connections in use, so there should be no problem with running out of connections.

Swivel recommends using database connection pooling for larger customers. It can be configured on the PINsafe Administration Console under Database/Connection Pool.

Connection Pool Options

Enable Connection Pool: Yes/No, default No

Maximum Pool size:

Connection test statement: This is a SQL statement which is sent to the database server at intervals to keep the connection alive. It therefore needs to be a statement that takes little processing time. "SELECT 1" is used by SQL Servers. When a test statement is set either Test on borrow or Test while idle should be set.

Connection test interval (s): How often to run test, default 300 seconds

Test on borrow: Yes/No, default No. If set to Yes, then the test statement will only be sent when a connection is requested. If there has not been a connection in a while, then it may time out and produce an error log message. This is not required if Test on idle is set to Yes.

Test on idle is more efficient, but you then have the possibility that the connection fails between the last time it was tested and the time it's actually used.

Test while idle: Yes/No, default No. If set to Yes, then the statement will be sent at regular intervals (as set by the Connection test interval) for all connections that are not currently active. This should mean that connections will never time out (unless you set the test interval too long), but will result in more traffic. This is not required if Test on borrow is set to Yes.

Test on borrow has more overhead, because you execute an extra SQL command every time, rather than at the set time.