

Encryption Plug-In

This manual describes the use of the *EncryptPlugIn* for thinkDB 2.5. This Plug-In will work for DataViz SmartList To Go but some of the terminology used in this document may have changed for SmartList To Go. For more information on SmartList To Go, please visit <http://www.dataviz.com/products/smartlisttogo> . If you want additional information on thinkDB Plug-Ins, please visit <http://ca.geocities.com/sltgplugins> .

1. **thinkDB Plug-In Overview**
2. ***EncryptPlugIn* Overview**
 - a. Purpose
 - b. Requirements
3. **Installation**
 - a. Upgrading from a previous version
4. **UnInstalling**
5. **Usage Overview**
 - a. Configuring Plug-In fields
 - b. Running the Plug-In
6. **Known Problems**

=====

1. thinkDB Plug-In Overview

A thinkDB Plug-In is simply an add-on program that fills a specific purpose in thinkDB. As such, they extend the functionality of thinkDB. Plug-Ins are usually created by third-party developers to "plug into" thinkDB on a Palm OS based handheld.

2. *EncryptPlugIn* Overview

a. Purpose:

This Plug-In encrypts and decrypts up to five Text/Memo fields and up to two float fields per record in a tinyByte using the Blowfish encryption method. Blowfish was designed in 1993 by Bruce Schneier and is a Feistel algorithm using variable length keys up to 56 characters (448 bits). No one has come close to developing an attack that breaks Blowfish. This *EncryptPlugIn* has been 'designed' to look like the Demo/Beta thinkDB Encryption Plug-In created by ThinkingBytes Technologies Inc in 2001.

b. Requirements:

This Plug-In requires no special libraries or other applications to run.

3. Installation

IMPORTANT (READ THIS): If you are currently using a previous version of *EncryptPlugIn*, you must decrypt ALL your data before installing this version and use the Update.prc Plug-In described next.

a. Upgrading from a previous version:

If you are upgrading from a previous version of the *EncryptPlugIn*, the following is some special information to help you avoid problems:

1. It is necessary to decrypt all your previous data **before** installing this new version.
2. Now install the new version as explained below.
3. "Un-configure" the old *EncryptPlugIn* by using the special Plug-In named Update.prc that is included in the zip. Install Update.prc if you do not have it already installed and then do the following steps:
 - From the Field Editor, select the small box to the right of the field you have previously selected as the *EncryptPlugIn* field to get to the Field Properties screen.
 - From the Plug-In list, select *Upgrade* and press the OK button.
 - Select the small box to the right of the field you have selected as the *Upgrade* Plug-In field and re-select *EncryptPlugIn* and configure the Plug-In as described in Section 5.

Installation (continued)

To install the *EncryptPlugIn*, double-click the *tpiEncrD.prc* (demo) file or the *tpiEncrP.prc* (purchased) file depending on what you have. You can also select Install from with the Palm Desktop and choose the .prc file you have. There are two demo tinyBytes included in the zip named *encrDemo.pdb* and *myVault.pdb* which show working examples of what the *EncryptPlugIn* can do. The names on the handheld will be TD2_EncryptDemo and TD2_myVault.

The Palm Install Tool will then present a dialog to continue installation of the selected files to install. After doing a HotSync, the Plug-In will be installed into thinkDB and is ready for use. To verify that it was installed correctly, launch thinkDB on your handheld, choose Tools/Plug-in Manager from the thinkDB application menu. Tap the arrow beside "Plug-In" and you should see the *EncryptPlugIn* listed. If not, repeat install instructions. If you do see it listed, it has been installed.

4. Un-Installing

To remove a Plug-In from thinkDB, first remove the Plug-In from all fields in all your tinyBytes. If you do not, you will encounter problems and could potentially corrupt data. Once you have removed all references of it from your tinyBytes, choose Tools/Plug-In Manger from the thinkDB application menu. Tap the arrow beside "Plug-In", tap on the Plug-In you wish to remove, then tap the "Delete ..." button.

5. Usage Overview

Before configuring any Plug-In, you must first create all the fields that the Plug-In requires for both input and output.

a. Configuring Plug-In Fields:

1. From the Field Editor, select Field Type thinkDB Plug-In from the Field Type list. Selecting the small box to the right will then open the Field Properties dialog.
2. From the Plug-In list, select the *EncryptPlugIn* and then 12 input fields will be shown plus one "Results in:" field shown. Actually there are 8 "Results in" fields which can be accessed by selecting the small box to the right.

***EncryptPlugIn* Inputs:**

Signature: To use the *EncryptPlugIn*, you must first generate a 'signature'. Selecting "Signature", you will be asked for a secret passphrase that will be used to encrypt/decrypt the desired fields (text/memo, float). Tapping the Info Icon at the top right, will show some brief instructions on how to use the *EncryptPlugIn*. **IMPORTANT:** Before changing your secret passphrase or ANY of the fields defined to this Plug-In, make sure all data in the tinyByte has been decrypted. Any data that is still encrypted will forever be encrypted!. Choose a unique phrase that you will be sure to remember. This phrase is case sensitive. Memorize your secret passphrase. If you forget this phrase, your data cannot be decrypted. A 'signature' will be generated by a one-way hash of this passphrase and will be stored in the tinyByte under "Signature". This 'signature' will be used to verify that the secret passphrase entered at a later date is correct.

Mode Select: The *EncryptPlugIn* has three modes: Encrypt Only, Decrypt Only, and Automatic. If the *EncryptPlugIn* is set to Encrypt Only, it cannot decrypt fields. If it encounters an encrypted field, it will leave it untouched. The idea is the same for Decrypt Only; it will decrypt encrypted fields, but if it encounters a plain-text field, it will leave it untouched. The Automatic setting toggles between encryption and decryption. That is, it will decrypt any encrypted field, and encrypt any decrypted field, that it encounters. In general, it is often useful to leave *EncryptPlugIn* on Automatic, and use Encrypt Only or Decrypt Only when applying *EncryptPlugIn* to multiple records using the "Filtered Operations" feature.

Confirm Encr: You may choose to be prompted for confirmation before each encryption. The valid selections are "Always" and "Never".

Confirm Decr: You may choose to be prompted for confirmation before each decryption. The valid selections are "Always" and "Never".

Encrypt Field x: There are five of these fields to hold up to five text/memo fields you wish to encrypt/decrypt.

Encrypt Float x: There are two of these fields to hold up to two float fields you wish to encrypt/decrypt. If you wish to encrypt floats you must also supply a text field for **FloatString** to hold the encrypted data.

EncryptPlugIn “Results-in”:

These eight fields are used to hold the “Results of the encryption” and must be the exact same fields you selected for Inputs (Encrypt Field x, Encrypt Float x, FloatString). In order to ensure proper operation, it is **CRITICAL** that the Encrypt and Result fields correspond exactly. Data loss will result if this is not the case!

b. Running the Plug-In:

There are two methods for having any Plug-In perform its duties:

1. Place the “plugged-in” field onto a form and tap it.
2. “Apply” the Plug-In from the “Filtered Operations” screen (which is one of the menu options that appears by tapping on the “tool” icon).

6. Known Problems

Blowfish is used in conjunction with the Cipher Block Chaining Mode (CBC). This method results in the encrypted data being longer in length than the original plain text. Text Fields can be at most 255 characters, and Memo Fields can be at most 4095 characters. If the encrypted data is longer than these limits, a “**Text Too Long**” message will occur and the text will not be encrypted. There is no set max length the plain text should be before this action takes place, but Text Fields less than 245 characters should get encrypted, but there is no guarantee. One shortcoming I noticed in the thinkDB/SLTG Plug-In API, is that one cannot differentiate within the Plug-In whether the incoming “text” is a Text Field or a Memo Field. Therefore, if a Memo Field has say 253 characters to be encrypted, the encrypted data would not exceed 4095 characters and would be ok, but the Plug-In must indicate “**Text Too Long**” just in case it is a Text Field. If you encounter this problem, just put extra characters, such as spaces, in the Memo Field to make it longer than 255 characters. Doing so will allow the Plug-In to know the incoming text is a Memo Field since it already exceeds 255 characters.

Editing encrypted text/memo fields will result in data corruption and the inability of recovering the plain text. thinkDB v2.0 will allow you to edit these fields, thinkDB v2.5 & SLTG displays a padlock for these fields and you cannot edit the data. Float fields are still editable but the actual encrypted data is stored in the text field you assigned to **FloatString**. The thinkDB/SLTG Desktop allows editing all fields (DO NOT TRY TO EDIT!).