Cryptfiler Functional Specification

I. Introduction

Cryptfiler is a file encryptor which uses the Microsoft Cryptographic API (CAPI) and the Win32 SDK. Its GUI is an adaptation of Filer, a Win32 file management sample application which is written in C. Cryptfiler demonstrates a large part of the capabilities of CAPI, including encryption, digital signature, and password-based key generation.

Cryptfiler's GUI contains two listboxes: the directory listbox and the file listbox. Users navigate about their directory tree by double-clicking on the directory listbox, which is located on the left of Cryptfiler's client area. Double-clicking on an entry in the directory listbox once causes the directory to be expanded, with all its subdirectories enumerated and displayed indented below the entry. Double-clicking on the entry a second time causes the entry to be collapsed. Users can perform cryptographic operations on files and directories by selecting them in the file listbox and selecting operations from the menu. When a user double-clicks on an item in the file listbox, Cryptfiler encrypts the file and displays its filename in gray text. Note: The plaintext file is deleted. A second double-click causes Cryptfiler to decrypt the file and display its filename in its original color. Users sign files by first selecting them in the file listbox and then selecting "Sign" from the File menu. Signed files appear in red text. Users can verify the signature of a file by selecting the file in the file listbox and choosing "Verify" from the file menu. An encrypted file that has been signed appears in light red text. When users encrypt and then sign files, they know that adversaries have not encrypted the files. Users can select the algorithms they wish to use for encryption and digital signature from the Options menu.

The first time a user runs cryptfiler, the application creates a key called fnKey and stores the key in the registry. From then on, when the user starts cryptfiler, the application access fnKey from the registry. Cryptfiler uses fnKey to encrypt file and directory names. When a user chooses to encrypt a file, Cryptfiler generates a session key with which to encrypt the file, exports the key, generates a random filename with a .CRP extension, creates a hidden file with this name, encrypts the name of the plaintext file with fnKey, stores the encrypted filename to the .CRP file, stores the exported key to the .CRP file, encrypts the data with the session key, stores the encrypted content to the .CRP file, and deletes the plaintext file. When the user chooses to encrypt a directory, Cryptfiler encrypts the name of the directory with fnKey, stores the data in a file called "dirinfo" in the directory, generates a random name with a .CRP extension, renames the directory using this name, hides the directory, encrypts all the files in the directory, and recursively encrypts all subdirectories. If the fnKey is lost or tampered with, Cryptfiler displays the file in the file listbox as "RecoverMe" in gray text. Although the user can recover the original file, the original filename is lost when fnKey is lost.

The "Password" item under the Options menu prompts the user for a password. Cryptfiler generates a session key with this password the next time the user encrypts a file. When the user wishes to decrypt that file, the application again prompts the user for the password.

Cryptfiler never displays the random file and directory names it generates. Cryptfiler only displays the original names. Cryptfiler accomplishes this by maintaining two string tables: one for the directory listbox and one for the file listbox. Each entry in the string table has a hidden and displayed field. The hidden field holds the pathname as it appears on the disk. The displayed field holds the pathname as it appeared on the disk before encryption.

Cryptfiler stores all signatures of files along with the ALG_ID's of the hashing algorithms used to generate the signatures as separate, hidden files in the hidden directory "sig" off the root on the same drive as the system directory. This directory is hidden by default. Users may choose to hide or unhide this directory and its contents. The filenames of these files are SHA hashes of the full pathnames of the files. It is necessary to convert these 20-byte hashes before using them as filenames since there are only 2^6 rather than 2^8 legal characters for filenames. By choosing 64 (2^6) characters for filenames (A-Z, a-z, 0-9, +, and _), it is possible to convert 3 bytes of the hash, containing 24 = 8 * 3 bits of data, to 4 bytes of the filename, also containing 24 = 6 * 4 bits of data. Cryptfiler adds 4 null bytes to the hash in order to have 24 bytes with which to perform the conversion. Cryptfiler knows that a file has a signature if the hash of its full pathname is one of the files in the sig directory. To increase efficiency, Cryptfiler creates in memory a table of the signature files when filling the file listbox.

The default encryption algorithm is RC2, which is not fixed. For password-based key generation, MD4 is the hashing algorithm, and it cannot be changed. The default hashing algorithm for generating signature files is MD4, which the user can change. The algorithm used to generate fnKey is RC4, which is fixed. The algorithm used to hash pathnames of files in order to generate signature filenames is SHA, which is fixed. It is necessary to #define WIN95 when compiling under Windows 95. Making Cryptfiler a Unicode application simply requires a #define UNICODE.

Encrypted files are structured in the following way:
1. Filename encrypted with fnKey (RC4). - 256 * sizeof(TCHAR) bytes
2. Exported key blob length - sizeof(DWORD) bytes. This is zero if the user selected a password-based key generation algorithm
3. Key blob - (key blob length bytes) (0 bytes if key generated from password)
4. Encrypted content

II. Menus

A. File: Contains items which perform cryptographic operations on files.

B. View: Contains items which change the view of the file and directory listboxes.

C. Drives: Contains a list of available drives.

D. Options: Contains items which set options for the cryptographic operations

E. Help: Contains information for the user to obtain help and version information.

III. File Menu

A. Encrypt/Decrypt: If the selected file in the file listbox is plaintext, this menu item causes the application to encrypt the file. If the selected item is encrypted, this menu item causes the application to decrypt the file. During encryption, the plaintext file is deleted. During decryption, the encrypted file is deleted.

B. Sign: Creates a separate digital signature of the selected file in the file listbox. This file is stored in the directory sig off the root on the same drive as the system directory. If a signature file already exists, the application asks users if they want to delete the signature file and replace it with a new signature file.

C. Verify: Verifies the signature of the file selected in the file listbox.

D. Exit: Terminates the application.

IV. View Menu

A. Refresh: Refreshes the view of the current drive to that of the root.

B. Expand Tree: Recursively expands the directory tree in the directory listbox.

C. Swap Sides: Swaps the file and directory listboxes.

V. Drives Menu: Lists available drives. When a user selects one of the items in this menu, the application switches to that drive.

VI. Options Menu

A. Choose Encryption Algorithm...: Prompts the user with a dialog box with radio buttons for each algorithm for generating session keys.

B. Enter Password...: Prompts the user with a dialog box with a field for a password with which to generate session keys.

C. Choose Signature Algorithm...: Prompts the user with a dialog box containing radio buttons for various algorithms with which to create digital signatures.

D. Hide Signatures (either checked or unchecked): Toggles the hidden or shown status of the directory containing digital signatures as well as the hidden or shown status of the signatures themselves.

VII. Help Menu

A. Cryptfiler Help...: Loads the Cryptfiler help file.

B. About Cryptfiler...: Shows the Cryptfiler version information.