

目次

メイン (2023.12)	2
シェアウェア (2010.03)	3
概要 (2010.05)	4
関数一覧 (2010.02)	7
KeyMart (2010.02)	8
補足 (2023.12)	9
動作環境 (2023.12)	12
変更履歴 (2023.12)	13

m.magic F DLL with KeyMart は、プロダクトキーの発生および正常性を確認するプログラムです。キーの発生に暗号理論を応用することで1つのIDに対し200000人分のキーを持つことができます。200000人に異なるキーを渡すことが可能です。瞬時に正常性を判断できます。認証コードシステムにより、キーを記録しておく必要がないため盗難防止ができます。また、改ざんが難しい時間形式による経過日数カウントが可能です。キー発生処理スピードアップ。使い方は簡単です。x64使用可能です。サンプルも付属しています。Visual Basic から利用できます。本 DLL は、.Net FrameWork がなくても動作します。KeyMart を動作させるためには .Net FrameWork が必要です。

使用方法

1. Visual C++ 2008 SP1 再頒布可能パッケージをインストールする (注)
2. .NET Framework 3.0 をインストールする
3. ダウンロードしたファイルを伸張する
4. サンプル中のDLLをロードするクラス等をコピーして使用する
5. KeyMart は伸張した状態で使用する

必要ソフト

Visual C++/Visual Basic.NET/Visual Basic 6.0
Visual C++ 2008 SP1 再頒布可能パッケージ (注)
.NET Framework 3.0 (KeyMart とサンプル動作のため)

サポート方法

e-mail のみのサポートです。回答には一週間程度かかることがあります。

(注) Ver.3.2 以降必要なし

メールアドレス m.magic@dream.nifty.jp

シェアウェア (2010.03)

ダウンロードありがとうございます。m.magic F DLL with KeyMart は、シェアウェアです。試用後、継続して使用する場合は、下記の方法で 999 円 を送金してください。E-MAIL の回答には一週間程度かかることがあります。

利用条件

- 1.m.magic F DLL with KeyMart に関するプログラム及びドキュメント等の著作権は、作者が保有します。
- 2.mmagic3f.dll を継続使用する、または、評価の目的以外で使用する場合、必ずシェアウェア料の支払いが必要になります。
- 3.試用期間中は、“4.再配布可能ファイル”に記載されるファイルの再配布はできません。
- 4.再配布可能ファイルは mmagic3f.dll, Trunk1f.dll のみであり、その他添付されているファイルは含みません。
- 5.試用期間中 mmagic3f.dll を使用したプログラムの配布を行うことはできません。
- 6.このプログラムの使用によるいかなる損害に対して作者は責任を負いません。

ライセンス取得

A.ベクターシェアレジ利用

作者名 : 丸山一利

ベクターシェアレジ利用 ライブラリ作者番号 : PA031043

作品名 : m.magic F DLL with KeyMart レジ作品番号 : SR034035

B.銀行口座

みずほ銀行 雷門前支店 支店番号 601 (普)1498221 マルヤマカストシ

概要 (2010.05)

m.magic F DLL with KeyMart は、プロダクトキーの発生および正常性を確認するプログラムです。1つのアプリケーションIDに対し 200000人分のキーを設定し識別できます。アプリケーションID、プロダクトキーを発生するのが、KeyMart そのチェックを行うのが m.magic F DLL です。アプリケーションID等は、KeyMart のみで発生することができます。認証コードが利用可能です。また、改ざんが難しい時間形式による経過日数カウントも可能です。

- 1.KeyMart で アプリケーションID(260バイト) および プロダクトキーを発生する。
- 2.アプリケーションID(260バイト)をアプリケーションに添付する。
- 3.ユーザーにプロダクトキーを入力させ m.magic F DLL で正常性を確認する。
- 4.正常なら認証コードを生成し記録する。(任意)
- 5.正常でないなら時間値を取得し記録する。(任意)

認証関数

- 1.mfReadCodeEx TRUNKDLLから認証コード、経過時間を読取る
- 2.mfMakeProductKeyEx 認証コードをプロダクトキーに変換する
- 3.mfCheckKeyEx プロダクトキーの正常性を確認する
- 4.mfMakeValidationCodeEx プロダクトキーから認証コードを生成する
- 5.mfWriteCodeEx TRUNKDLLに認証コード、経過時間を書き込む
- 6.mfCheckDaysEx 必要に応じ経過時間を取得する

アプリケーションID

アプリケーションを識別するための固有値です。この値が変わるとプロダクトキーも異なる。キーバージョンによりフォーマットが異なる。MF_IDENTITY_KV17 は、キーバージョン1. 7以降、それ以外は、MF_IDENTITY_KV13 を使用する。

プロダクトキー

ユーザーに入力させる値です。ひとつのアプリケーションIDに対し複数のキーが設定できる。キーバージョン3. 0より大文字と小文字を区別しています。

認証コード

認証コードは、ユーザーに入力させたプロダクトキーから生成される。プロダクトキーとは異なる数値となる。プロダクトキーを記録しておく必要はない。認証コードを記録する。この値からプロダクトキーを生成できる。許可範囲は、ハードディスクのボリュームナンバーです。保存場所は、レジストリ、TRUNKDLL等選択可能。

時間値

時間のカウントができる。mfCheckDaysEx 関数で可能する。MF_GET_DAYS でカウント開始時間を取得し、MF_CHECK_DAYS で経過日数を取得できます。アプリケーションIDごとに時間値のパターンは異なります。戻り値は、経過日数です。正常な時間にならない場合 -1 を返します。

保存すべきデータ

○が保存すべきデータ。

	プロダクト キー	認証コード	時間値
プロダクトキー、認証コードを使用する	×	○	×
プロダクトキー、認証コード、時間値使用する	×	○	○
プロダクトキーのみ使用する	○	×	×
プロダクトキー、時間値を使用する	○	×	○

キーバージョンと桁数

○が使用可能。

	18桁	24桁	32桁
キーバージョン1.4以前	×	○	○
キーバージョン1.7	○	○	○
キーバージョン2.0	○	○	○
キーバージョン3.0	○	×	×

不正なプロダクトキー

プロダクトキーが正しくないとき、ウェイトがかかる。DLLがアンロードされるときもウェイトがかかる。

TRUNKDLL

Ver2.0 より 付属のDLL内に認証コードと時間値を記入できる。初期ファイル名は、Trunk1f.dll ですがファイル名を変更することができる。

32ビットと64ビットの区別はありません。

関数戻り値

すべての関数成功0、障害負の数。その他正の数。

mfInitialize 関数

Trunk DLL のファイル名をセットする必要がある。ファイル名はフルパスです。使用しない場合 NULL をセットする。

使用ファイル

配布先の環境に、.NET Framework は、必要はない。

二重起動

本DLLは二重起動できません。DLLがアンロードされるときウェイトがかかる場合がある。

KeyMart

KeyMart は、プロダクトキーを発生するためのプログラムです。.Net Framework を必要とします。

サンプルプログラム

サンプルプログラムにDLLをロードするクラス等がある。コピーして使うと便利です。サンプルを動作させるためには、本DLL等必要なファイルをカレントディレクトリにコピーしてください。

DLL設定

関数名	機能説明
mfInitialize	DLL を初期化します

認証機能

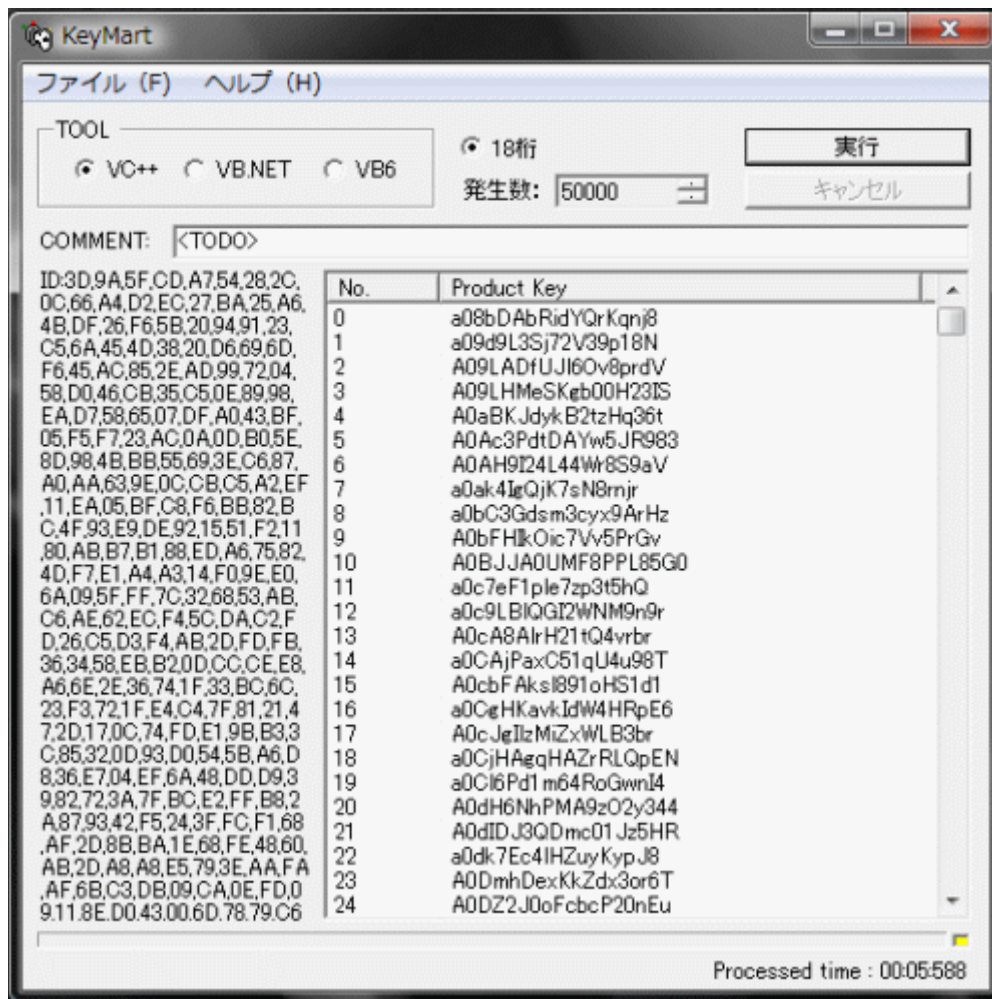
関数名	機能説明
mfCheckDaysEx	経過日数のを計算します。時間値は独自形式です
mfCheckKeyEx	プロダクトキーの正常性を確かめます
mfMakeProductKeyEx	認証コードからプロダクトキーを生成します
mfMakeValidationCodeEx	プロダクトキーから認証コードを生成します
mfReadCodeEx	認証コードをTRUNKDLLから読む
mfWriteCodeEx	認証コードをTRUNKDLLに書く

その他

関数名	機能説明
mfShowInputDialog	プロダクトキー入力ダイアログを表示します
mfGetFileVersion	ファイルバージョンを取得
mfSetRegValue	レジストリに値を書く
mfGetRegValue	レジストリの値を読む
mfGetRegValueSize	レジストリの値のサイズを取得

KeyMart (2010.02)

KeyMart は、アプリケーションID および プロダクトキーを発生させます。アプリケーションIDは、アプリケーション固有の値です。アプリケーションに添付してください。プロダクトキーは一度に200000個まで発生できます。同じアプリケーションIDのキーを再発生することはできません。アプリケーションID等は、テキストファイル形式で保存することができます。使用しているプログラミングツールに合わせた形式で出力されます。そのままコピーして利用できます。KeyMartを動作させるためには、.Net Framework が必要です。32ビット、64ビットどちらでも使えます。ただし、64ビットで使用する場合、64ビット用のDLLを KeyMart と同じディレクトリに配置すること。



補足 (2023.12)

※下へ行くほど情報は古い

MFCをスタティックとした(2023.12)

バージョン 3.2 より MFC をスタティックとした。すなわちランタイムをインストールする必要はない。

TRUNKDLL x64 (Ver3.1)

32ビット、64ビットどちらでも使える。

KeyMart x64 (Ver3.1)

32ビット、64ビットどちらでも使える。ただし、64ビットで使用する場合、64ビット用のDLLを KeyMart.exe と同じディレクトリに配置すること。

x64 (Ver3.1)

AMD64版のDLLを作った。

ランタイムDLL (Ver3.1)

外部ランタイム方式で動作している。Visual C++ 2008 SP1 再頒布可能パッケージをインストールしてください。

サンプル

Visual Studio 2003 で作られています。新しいバージョンで読み込むと変換後、コンパイル時にワーニングがでる場合があります。修正してください。

プロダクトキー (Ver3.0)

大文字と小文字は区別されるようになりました。

プロダクトキーの発生時間 (Ver3.0)

200000キーを発生させるのに約20秒かかります。(AthlonXP 2600+/WindowsVista の参考値)

キーの互換性 (Ver 3.0)

プロダクトキーバージョンは3. 0となりました。キーバージョン3. 0のみ生成します。

互換性 (Ver 3.0)

Version 3.x と 2.x の間に互換性はありません。全面的に変更しました。関数の引数、削除された関数等大幅に変わっています。呼び出す側のプログラムの修正は必要だが、古いプロダクトキーは診断できる。

mfReadCodeEx & mfWriteCodeEx 関数

書き込むDLLのファイル名は、フルパスです。

mfGetShortFileName を VISTA で使用する場合(Ver2.2)

APIの動作が変わったようです。以前は、受け取るファイル名を受け取るバッファのサイズを指定しなくても動作していたのですが指定しなければならなくなっています。

KeyMart Version

下線の番号は、.NET Framework のバージョンを表している。

2.1.3000.1234

↙ .NET Version3.0

二重ロード

本DLLは**二重ロードできません**。本DLLを利用したアプリケーションを二重起動したい場合は、各関数を使用した後にアンロードして下さい。

ランタイムDLL

外部ランタイム方式で動作しています。mfcr71.dll と msucr71.dll が必ず必要です。ない場合動作しません。必要なすべてのDLLを配布する際は、システムディレクトリに配置せず、実行ファイルと同じディレクトリに配置することを推奨します。

認証コード保存(Ver2.0)

Ver2.0 以前は、認証コードをレジストリに書き込んでいたが、保存場所を自由に選択できるようになった。mfMakeValidationCodeEx 関数を利用すると認証コードを取得できる。

アプリケーションID(Ver2.0)

Ver2.0 で発生させたプロダクトキーを使う場合は必ず mfInitialize 関数に MF_VERSION_20 をセットした後、チェック関数等を利用する。初期設定は MF_VERSION_20 です。また、Ver2.0 形式以外のプロダクトキーを取得することはできない。mfCheckDaysEx/mfCheckKeyEx/mfWriteCodeEx/mfReadCodeEx 関数 は、Ver2.0 用のチェック関数です。**MF_VERSION_17 以前のキーは、古い関数を使用する。**

関数の機能変更(Ver2.0)

mfReadCodeEx と mfWriteCodeEx 関数が機能変更された。TRUNKDLLに認証コード等を書き込みます。レジストリに書き込む場合は、mfMakeValidationCode と mfMakeProductKey を利用して mfSetRegValue & mfGetRegValue 関数で読み書きする。

互換性(Ver2.0)

Ver1.* と Ver2.0 との間での互換性はない。アプリケーション側を多少修正する必要がある。

アプリケーションID

アプリケーションIDは、**260バイト固定です。すべてを添付し、各チェック関数に渡さなければなりません。**

プロダクトキーの再発生

アプリケーションIDを指定して、**プロダクトキーを再発生させることはできません**。できない構造になっています。使っているIDがわかってしまった場合、キーを発生させることを防ぐためです。

プロダクトキーの発生時間

50000キーを発生させるのに約60秒かかります。(Celeron 533Mhz/WindowsXP の参考値)

mfInitial 関数

mfInitial 関数は、Ver1.7 より mfInitialize 関数に名称変更になりました。DLL の中には、旧名称が残っているので以前にコンパイルしたアプリケーションでもそのまま動作します。

アプリケーションID(Ver1.7)

Ver1.7 より アプリケーションID がバイト列に変わりサブIDが廃止されました。Ver1.7(Ver1.5/Ver1.4/Ver1.3) で発生させたプロダクトキーを使う場合は必ず mfInitialize 関数に MF_VERSION_17(MF_VERSION_15/MF_VERSION_14/MF_VERSION_13)をセットした後、チェック関数等を利用してください。初期設定は MF_VERSION_13 です。また、Ver1.7 形式以外のプロダクトキーを取得することはできません。mfCheckDaysEx/mfCheckKeyEx/mfWriteCodeEx/mfReadCodeEx 関数 は、Ver1.7 用のチェック関数です。**MF_VERSION_17 のときは ***Ex 関数を使用しなければなりません。**

アプリケーションID(Ver1.6)

Ver1.5 からの変更はありません。必ず mfInitialize 関数に MF_VERSION_15 をセットした後、チェック関数等を利用してください。

アプリケーションID(Ver1.5)

Ver1.5 より プロダクトキー文字が英数字に変わりました。Ver1.5(Ver1.4/Ver1.3) で発生させたプロダクトキーを使う場合は必ず `mfInitialize` 関数に `MF_VERSION_15`(`MF_VERSION_14`/`MF_VERSION_13`) をセットした後、チェック関数等を利用してください。初期設定は `MF_VERSION_13` です。また、Ver1.5 形式以外のプロダクトキーを取得することはできません。

アプリケーションID(Ver1.4)

安全性向上のため Ver1.4 より アプリケーションID の形式を変更しました。Ver1.4 で発生させた アプリケーションID を使う場合は必ず `mfInitialize` 関数に `MF_VERSION_14` をセットした後、チェック関数等を利用してください。初期設定は `MF_VERSION_13` です。また、新規に Ver1.3 形式でアプリケーションIDを取得することはできません。

再配布

アプリケーションに添付し再配布するためには、3個のファイル(`mmgic2f.dll` & `mfc71.dll` & `msvcr71.dll`)が必要です。配布する実行ファイルと同じディレクトリにこのDLLを配布してください。また、配布先の環境に、.NET Framework は、必要ありません。

mfShowInputBox 関数

プロダクトキーを入力させるためのダイアログですが、必ず使う必要はありません。ただし文字は大文字にしてください。

mfGetTempFileName 関数

ウィンドウズテンポラリファイル名を取得します。引数に3文字をセットした後、実行してください。

動作環境 (2023.12)

DLL仕様

- MFCをスタティックライブラリとしてとして使用

KeyMart 仕様

- .Net Framework 3.0 を使用

動作可能OS

○ 可 — 未確認 × 不可

OS	x86	x64(WOW64)	x64
Windows10	—	○	○
Windows8.1	—	○	○
Windows 7	—	○	○
Windows Vista	○	—	—
Windows XP	○	—	—

動作確認環境

PC	CPU	RAM	OS
ASRock X300	Ryzen 3200G	8.0G	Windows 10
ASUSTek M4A785TD-M EVO	AthlonII 615e	2.0G	Windows 8/8.1
ASUSTek M4A785TD-M EVO	AthlonII 240e	2.0G	Windows 7/Vista/8.1
ASUSTek A7S8X-MX	Sempron 3000+	1.0G	Windows XP

開発環境

Windows10
Windows8.1
Windows8
Windows7
Windows Vista(SP2)
Windows XP(SP3)
Visual Studio 2008(SP1)
Visual Studio.NET 2003(SP1)
Visual Basic 6.0(SP6)

変更履歴 (2023.12)

Version	日付	主要内容
3.2	2023.12	<ul style="list-style-type: none"> ● スタティックライブラリとしてコンパイルした
3.1	2010.05	<ul style="list-style-type: none"> ● Visual Studio 2008 でコンパイルした ● x64対応 ● mfReadCodeEx 関数無駄な処理を整理
3.0	2010.02	<ul style="list-style-type: none"> ● TRUNKDLLへの認証コード等書き込み方法変更 ● プロダクトキーは、小文字と大文字を区別することにした ● プロダクトキーの桁数、発生数の変更 ● プロダクトキーの作成方法変更 ● 関数の引数変更 ● 不要な関数の削除 ● 処理中応答なしにならないように変更(KeyMart) ● ファイルの書き込み方法変更(KeyMart)
2.2	2008.05	<ul style="list-style-type: none"> ● コールバック関数の呼び出し回数を減らした ● mfWriteCodeEx 関数の戻り値を増やした ● KeyMart を .NET Framework Ver3.0 で動作するように変更
2.1	2008.01	<ul style="list-style-type: none"> ● アンロード時のウェイト動作変更 ● 出力ファイルの拡張子を .txt に変更(KeyMart)
2.0	2006.10	<ul style="list-style-type: none"> ● mfMakeProductKeyEx & mfMakeValidationCodeEx & mfMakeProductKey & mfMakeValidationCode 関数追加 ● mfSetCreateTime & mfGetCreateTime & mfSetWriteTime & mfGetWriteTime & mfSetRegValue & mfGetRegValue & mfGetRegValueSize & mfSetAttributes & mfGetAttributes 関数追加 ● mfOperateCreateTime & mfOperateWriteTime 関数廃止 ● mfReadCodeEx & mfWriteCodeEx 関数機能変更 ● mfOperateRegData 関数廃止 ● DLLファイル名変更 ● Ver2.0 対応(KeyMart)
1.7b	2005.11	<ul style="list-style-type: none"> ● 処理スピード向上(KeyMart) ● コメントを入力可とした(KeyMart)
1.7	2005.02	<ul style="list-style-type: none"> ● mfInitialize 関数追加 ● 乱数発生方法変更 ● mfCheckDaysEx 関数追加 ● mfCheckKeyEx 関数追加

		<ul style="list-style-type: none"> ● mfWriteCodeEx 関数追加 ● mfReadCodeEx 関数追加 ● 処理時間短縮化(KeyMart) ● キーの最大発生数を 50,000 に変更(KeyMart) ● 処理時間を計測するように変更(KeyMart) ● インジケータ表示の変更(KeyMart)
1.6	2004.11	<ul style="list-style-type: none"> ● キー生成に利用する乱数の発生方法変更 ● キーをソートするように変更 ● キーの最大発生数を 40,000 に変更(KeyMart) ● InpuKey Dialog 18桁入力時 EditBox 入力最大桁数を正常化した ● InpuKey Dialog のクライアントサイズを固定にした ● ダイアログのレイアウト変更 (KeyMart)
1.5	2004.05	<ul style="list-style-type: none"> ● プロダクトキー18桁を追加 ● プロダクトキーを A-Z & 0-9 とした ● mfGetOsVersion & mfGetWinProductID & mfGetPcName & mfGetFileVersion 関数追加 ● プログレスバー追加 (KeyMart) ● キャンセルボタン追加 (KeyMart) ● 出力ファイルフォーマット変更 (KeyMart)
1.4	2003.12	<ul style="list-style-type: none"> ● セキュリティーの向上 ● mfShowInputBox 関数文字診断機構修正 ● mfInitial 関数追加 ● プロダクトキーの発生上限 10000 個にした (KeyMart)
1.3	2003.06	<ul style="list-style-type: none"> ● mfWriteCode & mfReadCode & mfCheckDays & mfGetLoginName & mfOperateRegData & OperateWriteTime & OperateCreateTime 関数追加 ● mfShowInputBox 関数文字診断機構追加と32桁時のボタン位置修正 ● KeyMart ファイル形式をプログラミングツールごとに選択化