TRUECRYPT

FREE OPEN-SOURCE ON-THE-FLY ENCRYPTION

USER'S GUIDE

www.truecrypt.org

バージョン情報

TrueCrypt User's Guide, version 4.3a. 2007年5月3日発行

ライセンスと特許情報

TrueCrypt をインストールする、または動作させる前に、あなたはバイナリとソースコードの配布パッケージに含まれている Licence.txt に書かれているライセンスに同意することが必要です。

著作権情報

A TrueCrypt Foundation Release

商標情報

TrueCrypt とTrueCryptのロゴはTrueCrypt Foundationの商標です。名称や製品を金銭化することが目的ではありませんが、TrueCryptの評判を守り同名あるいは似た名称の製品の存在が引き起こすサポートやその他の問題発生を防止するためです。TrueCryptは商標ではありますが、TrueCryptは今後もオープンソースのフリーソフトウェアでありつづけるでしょう。

その他の商標は、すべてそれぞれ個々の所有者のものです。

制限

TrueCrypt Foundationは、このドキュメントがあなたの要求に合っているとか、情報に誤りがないとかを保証しません。情報は技術的な不正確さがあったり、タイプミスがあったりするかもしれません。

目次

まえがき

この文書の多くの章(たとえば*技術解説*と*かみせかけの拒否*)はほぼすべてのバージョンのTrueCryptに対応していますが、いくつかの節ではまずWindows版TrueCryptユーザーを対象としていることに注意してください。そのため、それらの節ではいくつかの箇所にLinux版には適切ではない情報があります。Linux特有の情報については http://www.truecrypt.org/downloads.php で入手できるTrueCryptバイナリとソースのディストリビューションパッケージに含まれる TrueCrypt man page に記載しています。

はじめに

TrueCrypt は自動即時暗号化するボリューム(データ保存装置)の、作成と維持についてのソフトウェアです。自動即時暗号化(on-the-fly-encryption)というのは、データが読み出しまたは保存の直前にユーザーの介在なしに自動的に暗号化されるということです。暗号化されたボリュームのデータは、正しいパスワード/キーファイルまたは暗号化キーがなければ、読むことはできません。ファイルシステム全体(ファイル名、でィレクトリ名、空き領域、メタデータ他)が暗号化されます。

ファイルは通常のディスクと同じにマウントされた TrueCrypt ボリュームから、またはそのボリュームへコピーすることができます。(たとえば、単純なドラッグ・アンド・ドロップ操作でも可能) ファイルは暗号化された TrueCrypt ボリュームから読み込まれたりコピーされたりするつど(メモリー中で)即時に自動的に復号されます。同様に、ファイルは TrueCrypt ボリュームに書き込む直前に即時に自動的に RAM で暗号化されます。ただし、このことは暗号化されるまたは復号されるファイル全体が RAM 中に存在しなければならないということではありません。TrueCrypt には特別なメモリー(RAM)の必要はありません。これがどのように実行されるかは、以下を参照してください。

.avi ビデオファイルが TrueCrypt ボリュームに保存されているとします。(つまり、ビデオファイルはまるごと暗号化されているということです) ユーザーは正しいパスワードまたはキーファイルによって TrueCrypt ボリュームをマウント(オープン)します。ユーザーがビデオファイルのアイコンをダブルクリックすると、OS はそのファイルタイプに関連づけられたアプリケーション(通常はメディアプレーヤー)を起動します。メディアプレーヤーは再生するためにビデオファイの最初の一部分を TrueCrypt 暗号化ボリュームから RAM(メモリー)へと読み込み始めます。この一部分が読み込まれるときに TrueCrypt は自動的に(RAM に)それを復号します。復号されたビデオの一部分はメディアプレーヤーで再生されます。この一部分が再生されているときに、メディアプレーヤーは TrueCrypt 暗号化ボリュームからビデオファイルの次の一部分を RAM(メモリー)へと読み込み、この過程がくりかえされます。この過程を即時(オン・ザ・フライ)暗号化/復号と呼び、ビデオファイルだけでなくすべてのファイルタイプについて機能します。

TrueCrypt は絶対に暗号化されたデータをディスクには置きません。臨時に RAM(メモリー)に置くだけです。ボリュームがマウントされていても、そのボリュームに保存されているデータは暗号化されたままです。Windows を再起動したり PC の電源を切ったりすると、ボリュームはアンマウントされそこに保存されたファイルは暗号化された状態でアクセス不能となります。正しいシ

ャットダウン手順なしで電源供給が突然遮断されたとしても、そのボリュームに保存されたファイルは暗号化された状態でアクセス不能となります。ふたたびアクセス可能にするには、正しいパスワードやキーファイルを使ってボリュームをマウントする必要があります。

初心者のためのチュートリアル

TrueCrypt コンテナの作り方と使い方

この章では TrueCrypt.ボリュームの作り方、マウントのしかたと使い方を順を追って説明します。なお、他の章にも重要な情報が記載されているので、それらもぜひお読みください。

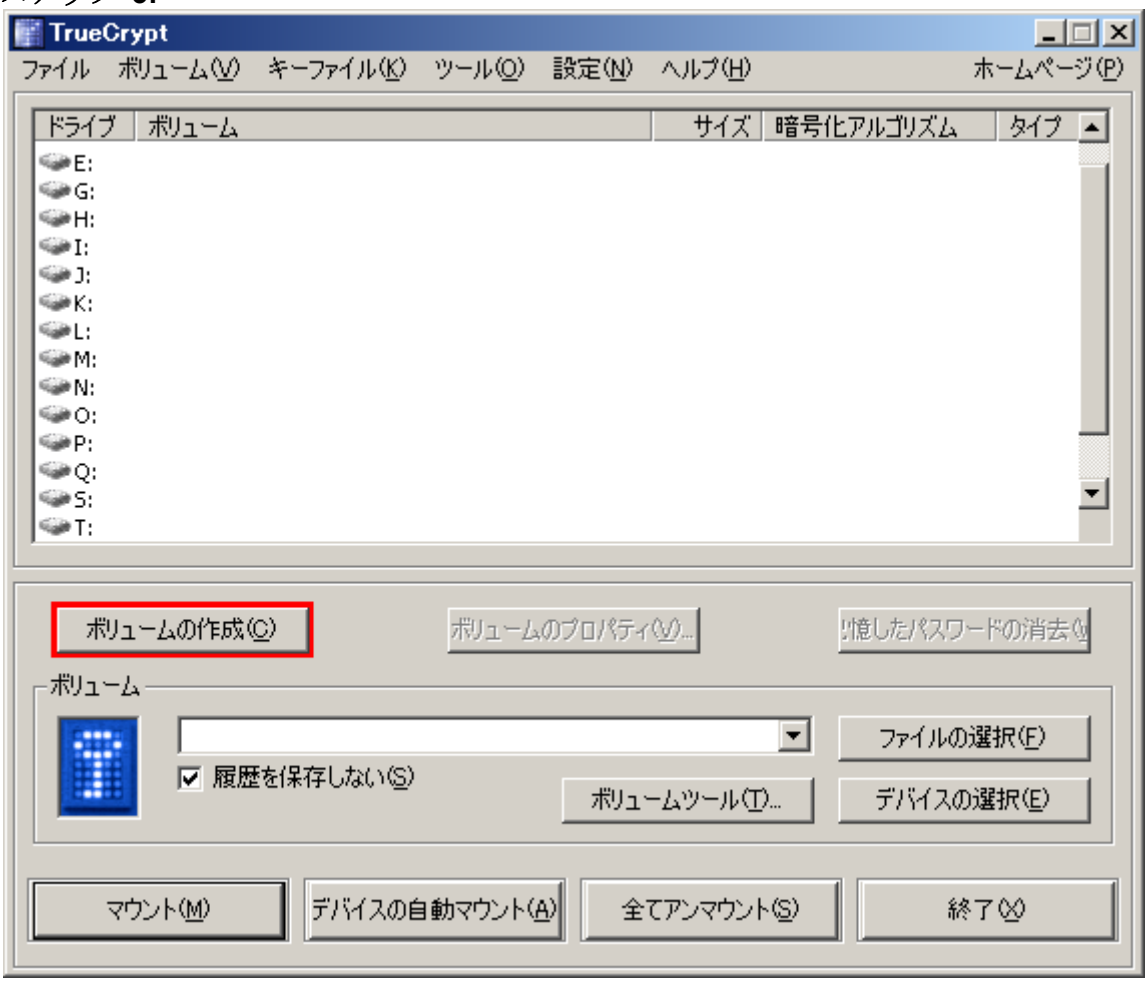
ステップ 1:

まず TrueCrypt をダウンロード、展開しインストールしてください。(インストールは *TrueCryptSetup.exe* をダブルクリックし、**Install** をクリックしてください)

ステップ 2:

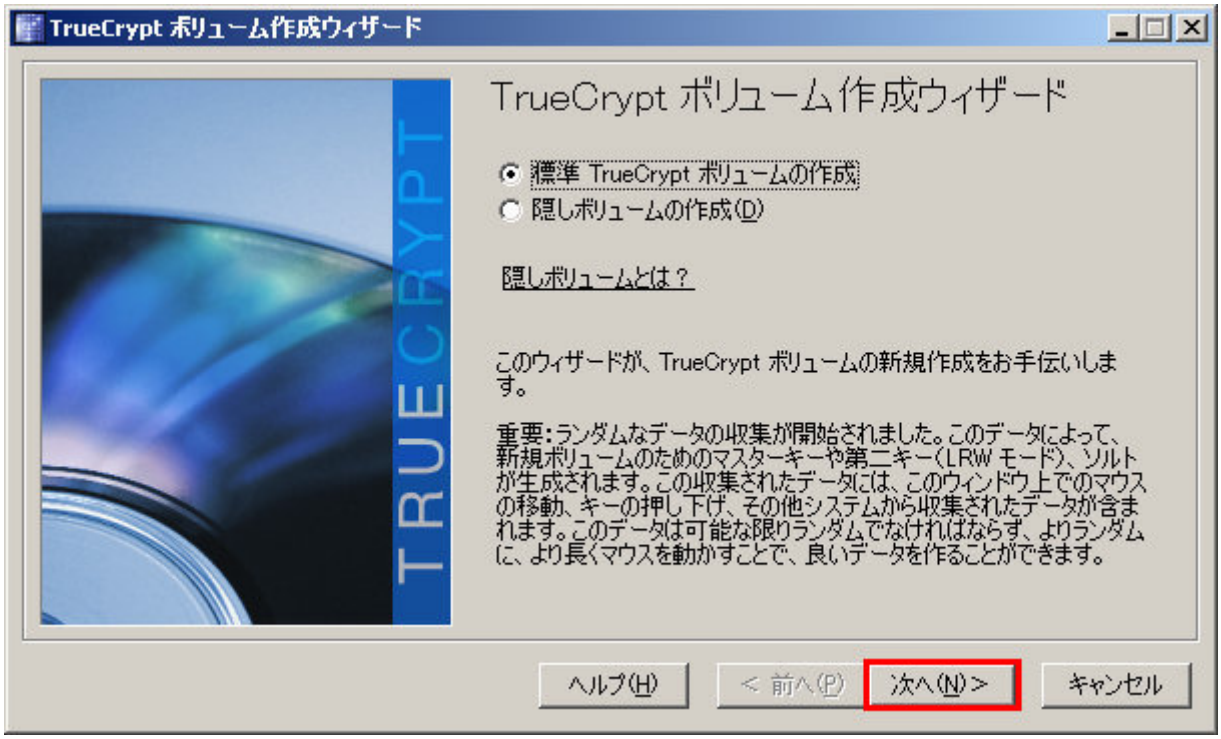
TrueCrypt.exe をダブルクリックするか Windows のスタートメニューの TrueCrypt ショートカットをクリックして TrueCrypt を起動してください。

ステップ 3:



TrueCrypt のメインウィンドウが表示されます。「**ボリュームの作成**」をクリックしてください。(赤で囲われている部分)

ステップ **4:**

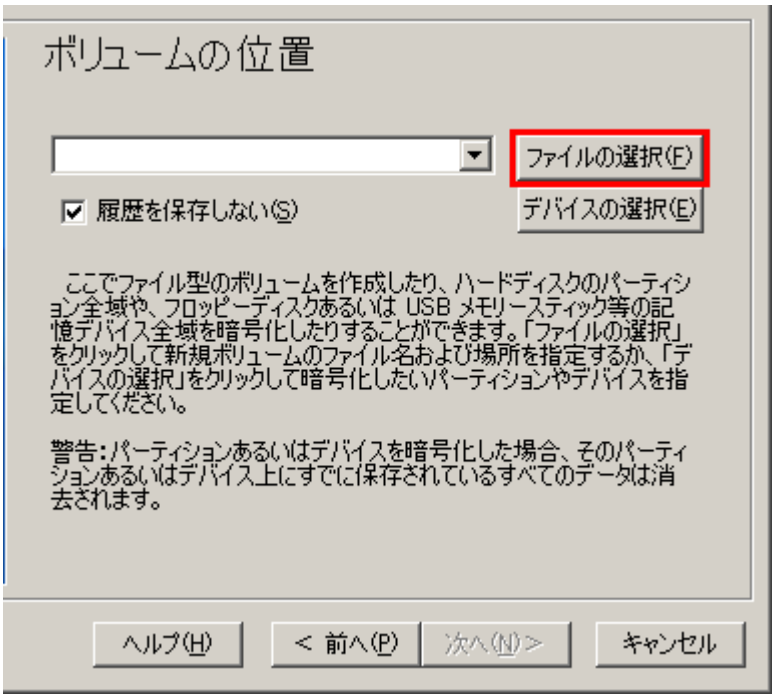


「TrueCrypt ボリューム作成ウィザード」ウィンドウが表示されます。

ウィザードウィンドウの説明を読んで、「**次へ**」をクリックしてください。

注意: 以降のステップではウィザードウィンドウの右側だけを掲載します。

ステップ **5:**



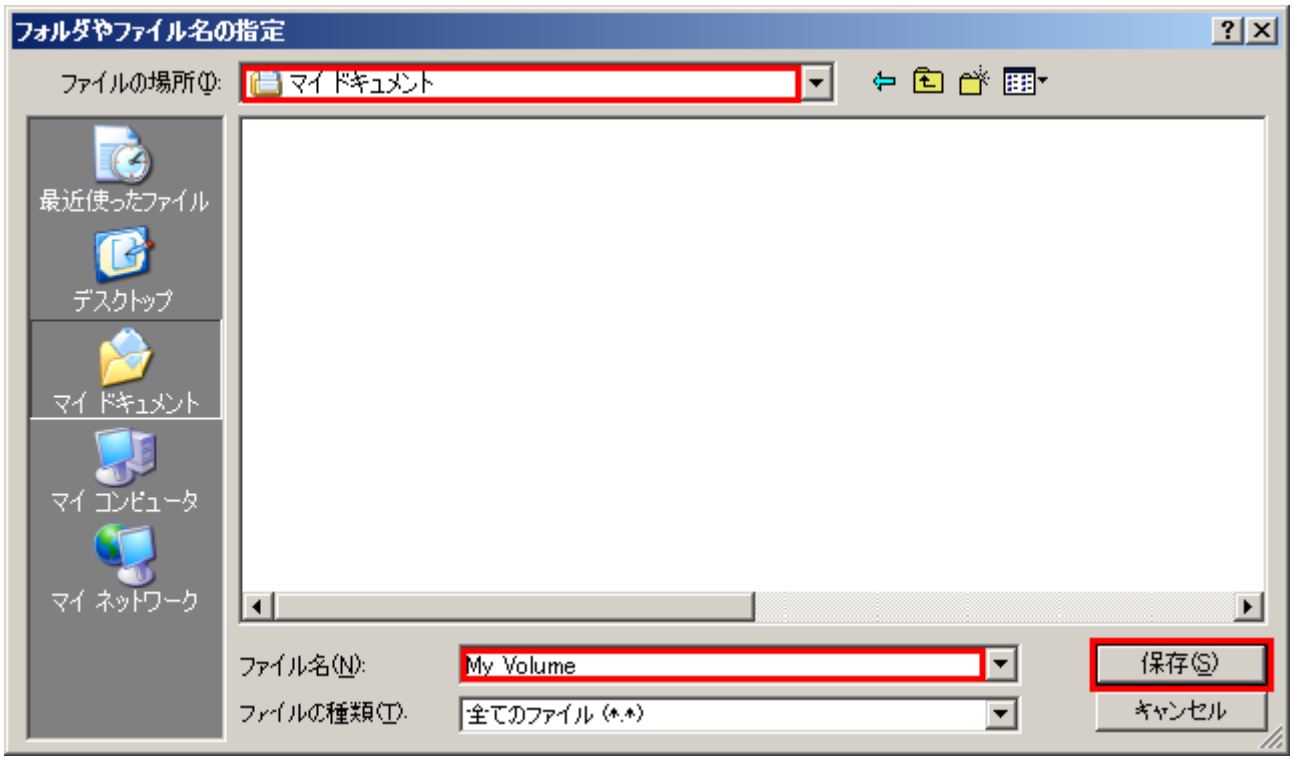
このステップでは、どこにどういう名前で TrueCrypt コンテナを作るのかを決めます。TrueCrypt のボリュームはコンテナとも呼ばれるファイルとしてでも、特定のパーティション(デバイス)にでも作ることができます。

TrueCrypt コンテナは通常ファイルとまったく同じであることに留意してください。したがって、普通のファイルと同様に移動、コピー、削除ができます。また、次のステップで説明するようにファイル名を必要とします。

「**ファイルの選択**」をクリックしてください。

Windows の標準的なファイル選択ダイアログが表示されます。(TrueCrypt ボリューム作成ウィザードは背景に開いたままです)

ステップ **6:**



このチュートリアルでは TrueCrypt ボリュームを上のスクリーンショットのとおり *D:¥My Documents¥* に置くこととし、ボリューム(コンテナ)のファイル名を(上のスクリーンショットのとおりに) *My Volume* とします。もちろん、他のファイル名、他の場所(たとえば USB メモリ)、にすることができます。

この時点ではまだ *My Volume* は存在しません。－TrueCrypt がこれから作成します。

重要 ：TrueCrypt は既存のファイルを*暗号化するのではない*ことに注意してください。既存のファイルを選択すると、それは新しく生成されるボリュームで上書きされます。(つまり、元ファイルは*暗号化されるのではなく、失われる*ことになります) これから作成する TrueCrypt ボリュームに既存ファイルをコピーすることで、暗号化が可能になります。[1]

ファイル選択でコンテナを置きたいパスを選んでください。

コンテナの希望のファイル名を入力して、

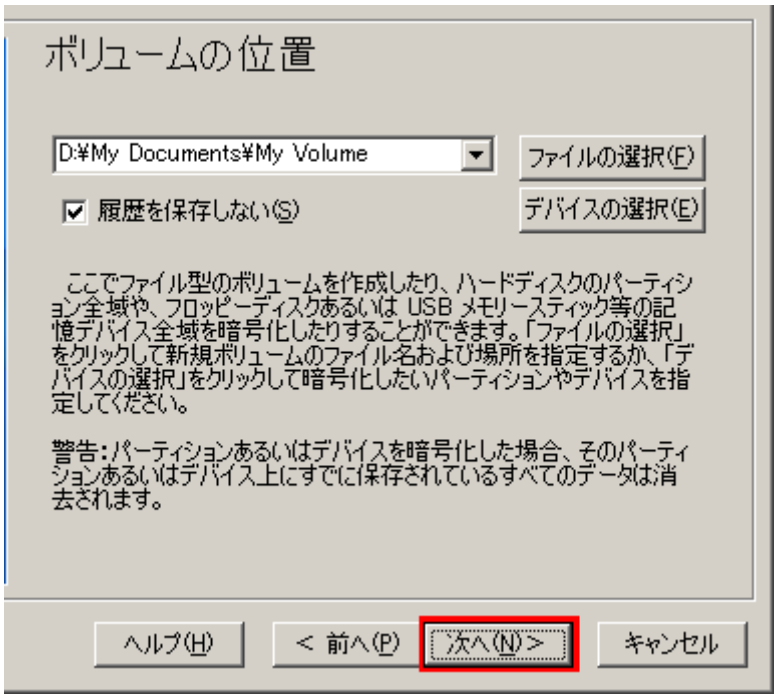
ダイアログの「**保存**」をクリックしてください。

ファイル選択ウィンドウは消えます。

[1]TrueCrypt ボリュームに既存の非暗号化ファイルをコピーしたあと、元の非暗号化ファイルを完全削除するべきです。完全削除のためのツールは(多くはフリーで)存在します。

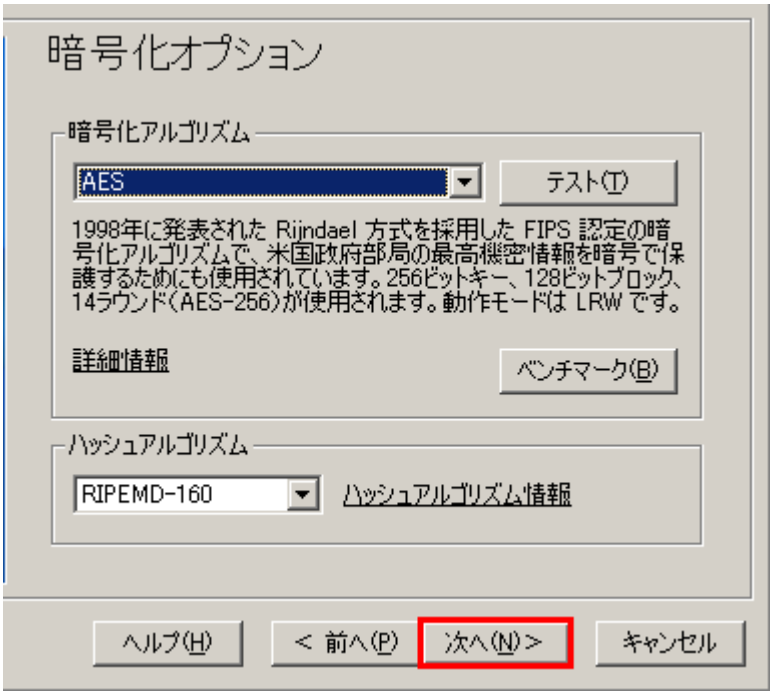
以降のステップでは「TrueCrypt ボリューム作成ウィザード」へ戻ります。

ステップ **7:**



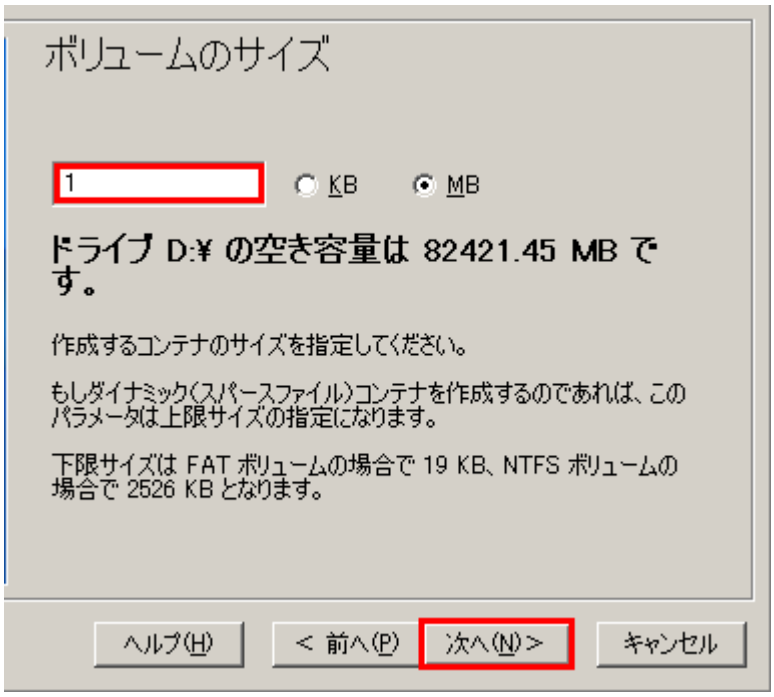
ボリューム作成ウィザードで「**次へ**」をクリックします。

STEP 8:



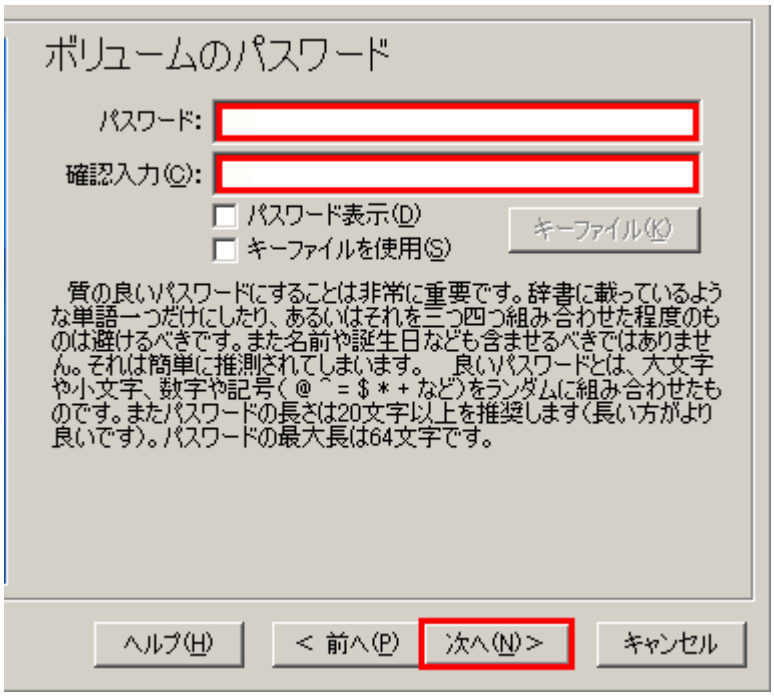
ここで暗号化アルゴリズムとハッシュアルゴリズムを選択します。どれを選べばいいかわからなければ、既定値のままで「**次へ**」をクリックしてください。(詳細は「暗号化アルゴリズム」と「ハッシュアルゴリズム」の章を参照)

STEP 9:



ここでは例として TrueCrypt コンテナのサイズ(容量)を 1MB にします。もちろん、これとは異なるサイズにすることができます。希望するサイズを入力欄(赤でマーク)に記入し「**次へ**」をクリックします。

ステップ **10:**



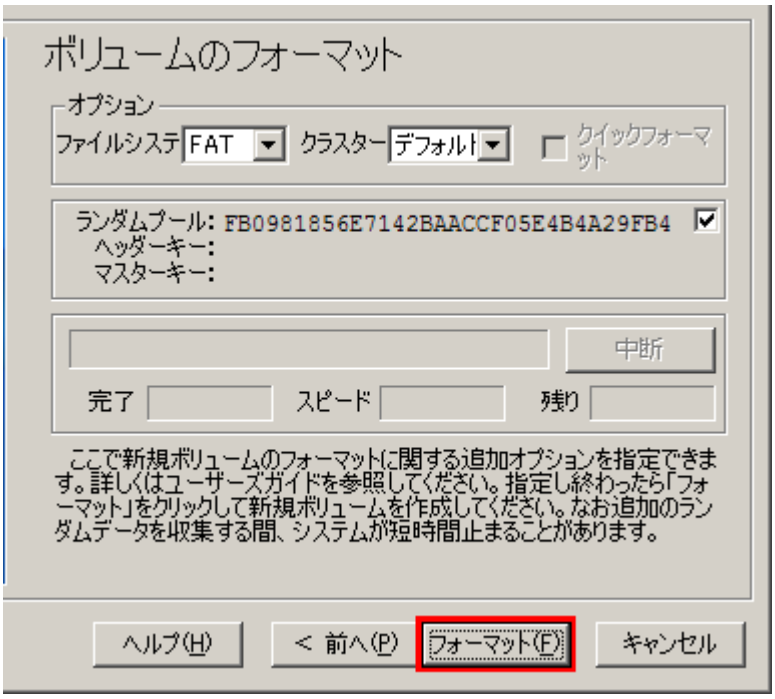
ここは重要なステップです。ここでボリュームの「良い」パスワードを決めなくてはいけません。

どのようなパスワードが「良い」のかウィザードウィンドウの説明を注意深く読んでください。

良いパスワードを決めたら、最初の入力欄に記入し、その直下の入力欄に同じものをもう一度記入して、「**次へ**」をクリックしてください。

注意: 「**次へ**」ボタンは両方の欄に同じパスワードを記入しないと、クリックできるようになりません。

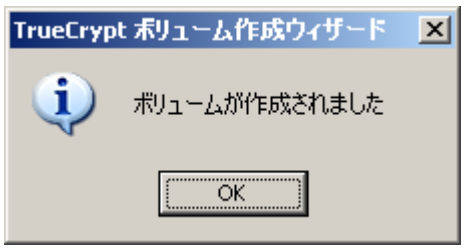
ステップ **11:**



ウィザードウィンドウの中ですくなくとも 30 秒間、マウスをランダムに動かしてください。動かすのが長ければ長いほど、いいのです。これは暗号化キーの質の向上のために重要なことです。

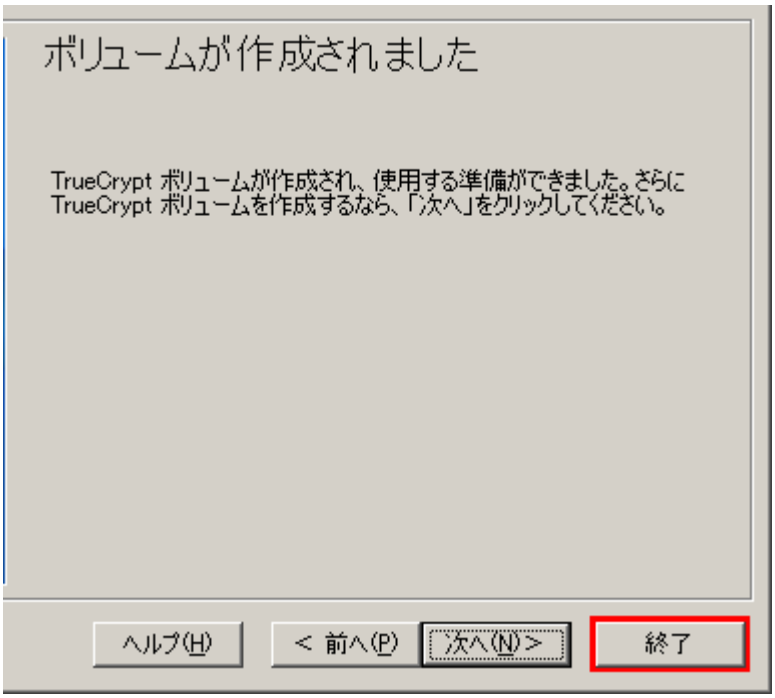
「**フォーマット**」をクリックしてください。

ボリューム作成が始まります。TrueCrypt は(Step 6 で指定したように)My Documents フォルダーに My Volume という名前のボリュームを作ります。このファイルは TrueCrypt コンテナであり、暗号化された TrueCrypt ボリュームを含みます。ボリュームの大きさによってはボリューム生成に時間がかかるかもしれません。完了すると、次のダイアログが表示されます。



「**OK**」をクリックしてダイアログを閉じてください。

ステップ **12:**



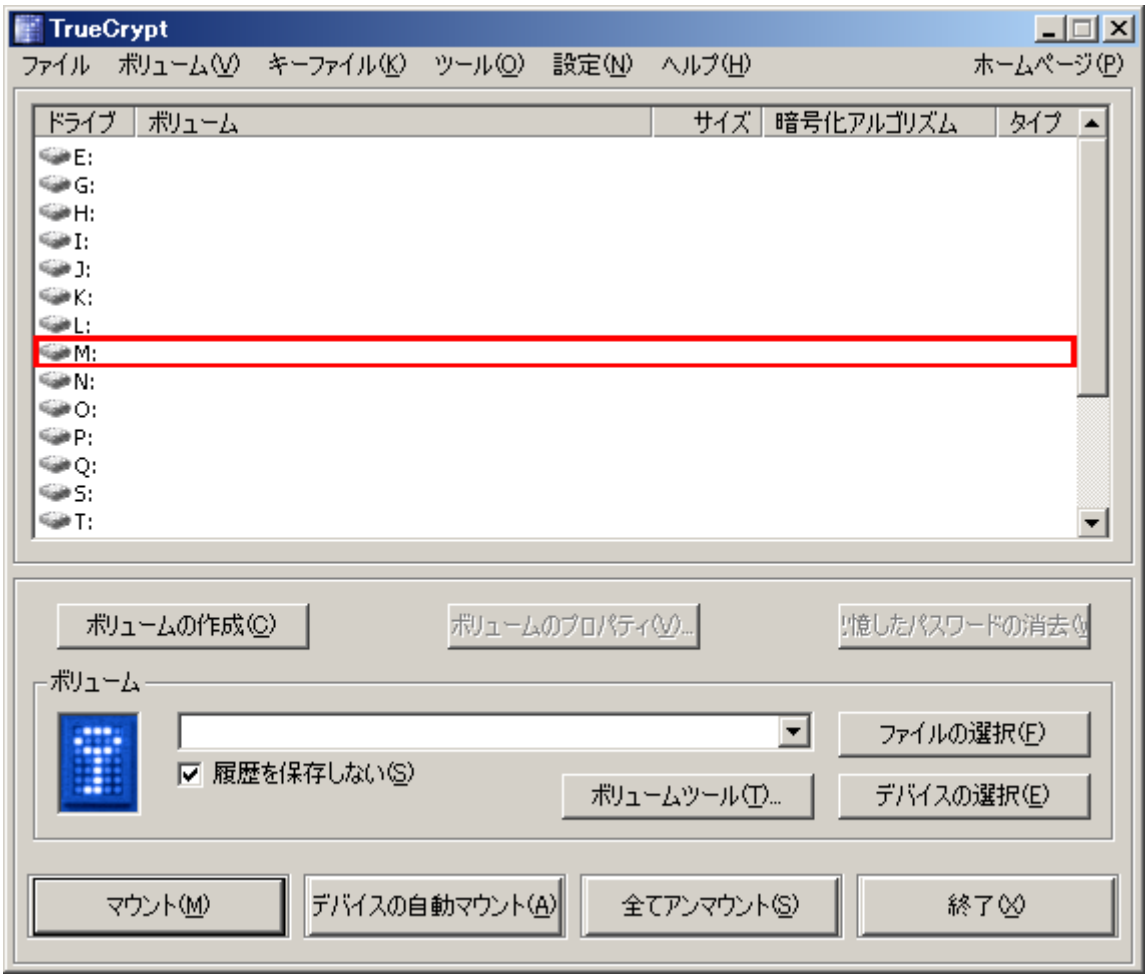
これで TrueCrypt ボリューム(ファイルコンテナ)の作成ができました。

TrueCrypt ボリューム作成ウィザードの「**終了**」をクリックしてください。

ウィザードウィンドウが消えます。

残りのステップでは、作ったばかりのボリュームをマウントします。TrueCrypt ウィンドウに戻りますが、これは表示されたままのはずです。もし、そうでなければステップ 2 に戻って TrueCrypt を起動し、ステップ 13 から続けてください。

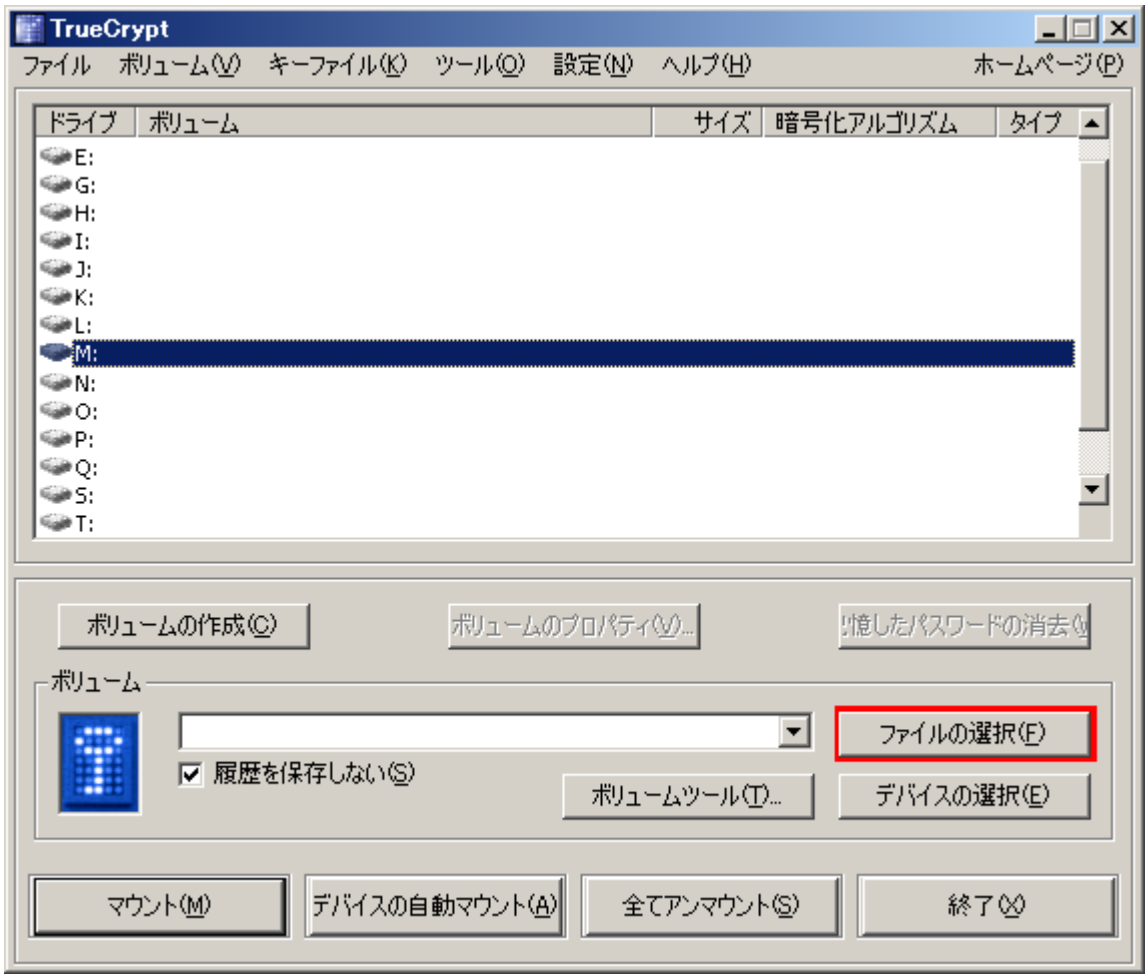
ステップ **13:**



リストから(赤で囲ってある)ドライブレターを選んでください。これが TrueCrypt コンテナがマウントされるドライブレターになります。

注意: このチュートリアルではドライブ M を選びます。しかし、もちろんどの空きドライブレターでも選ぶことができます。

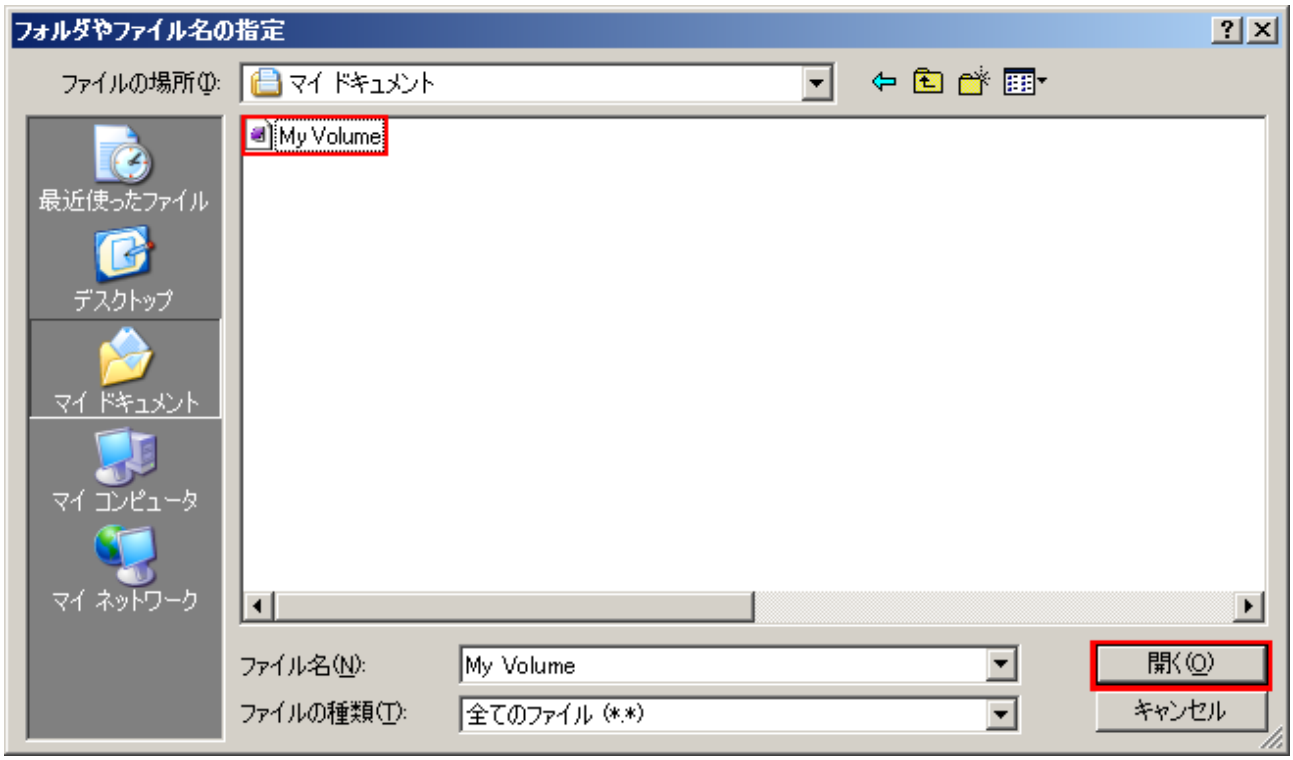
ステップ **14:**



「**ファイルの選択**」をクリックしてください。

標準ファイル選択ウィンドウが表示されます。

ステップ **15:**



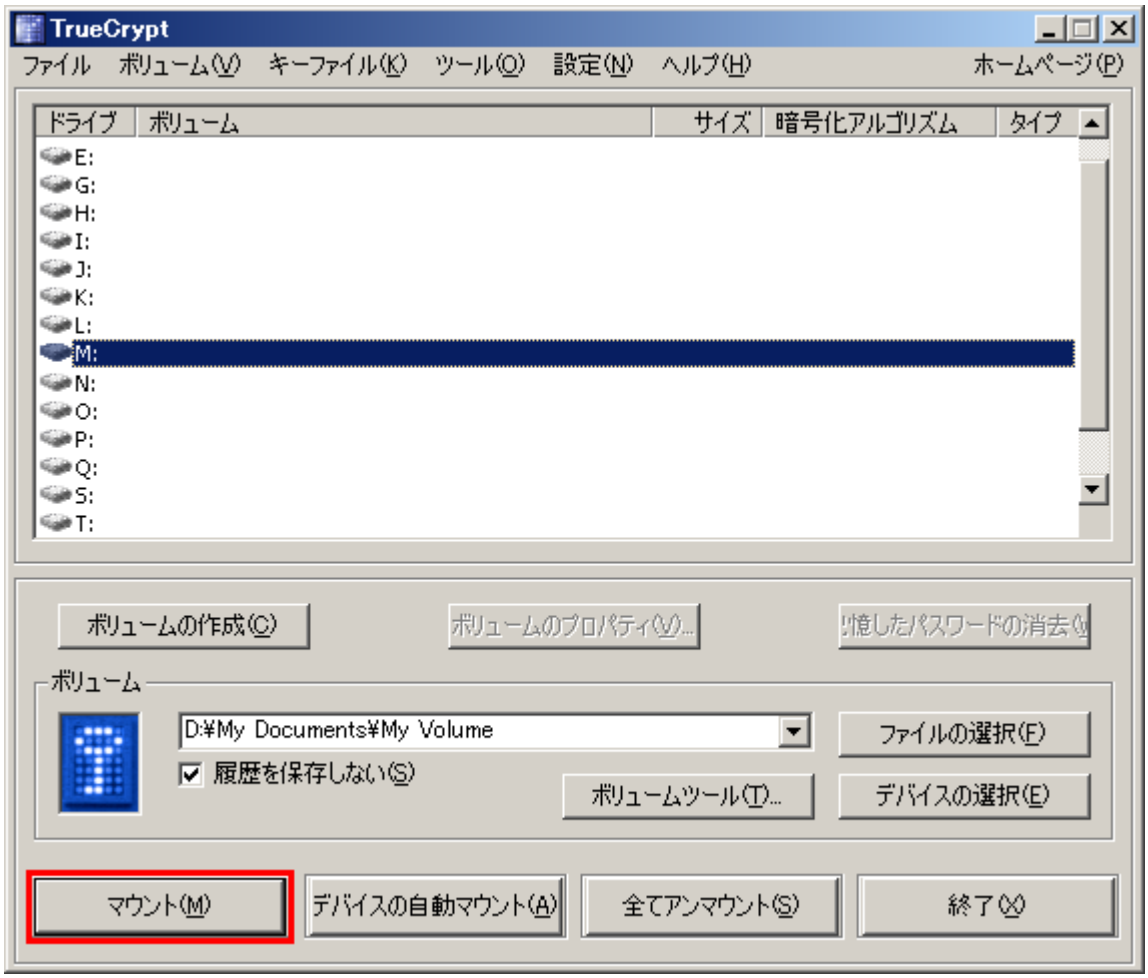
ファイル選択でコンテナファイル(ステップ 6-11 で作成したもの)を探し、それを選択してください。

ファイル選択ウィンドウの「**開く**」をクリックしてください。

ファイル選択ウィンドウが消えます。

以降のステップでは、TrueCrypt のメインウィンドウに戻ります。

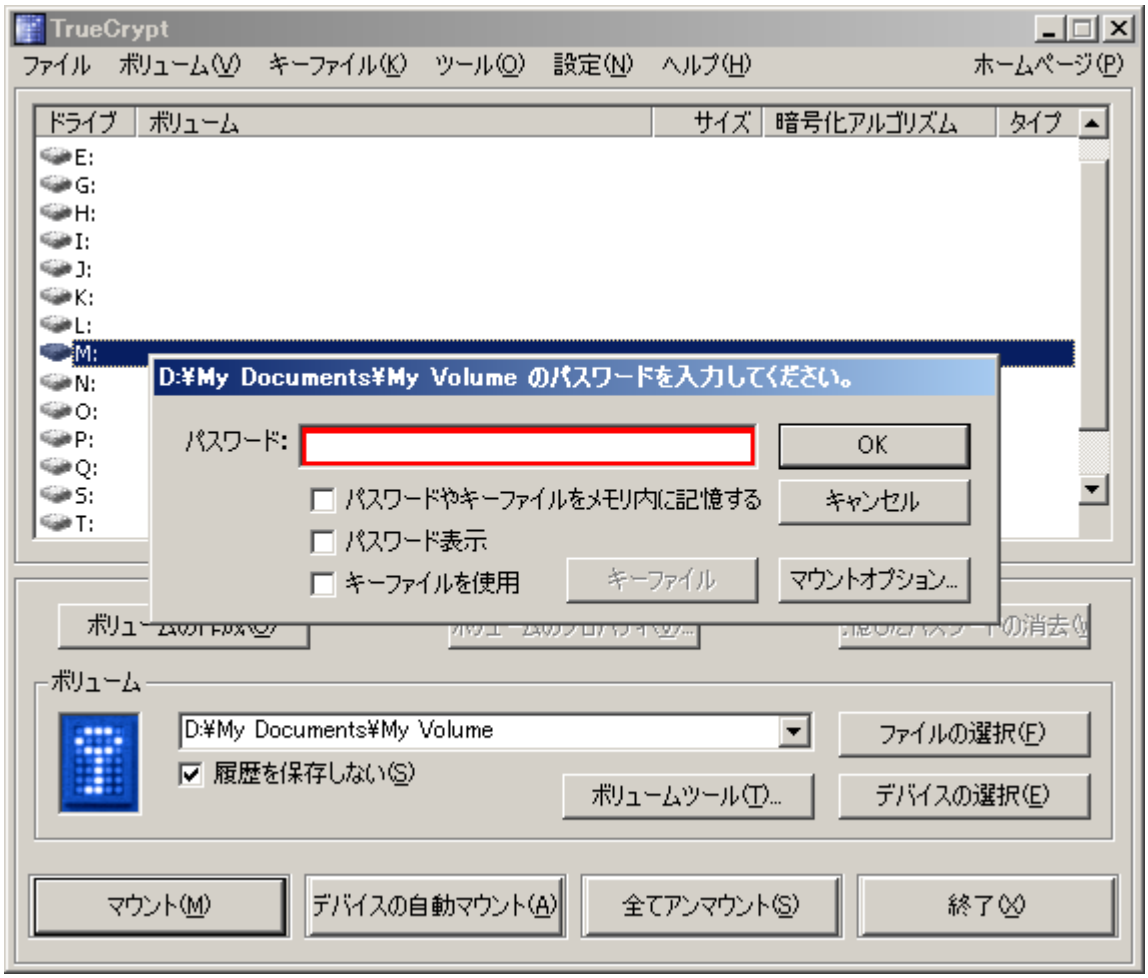
ステップ **16:**



TrueCrypt メインウィンドウで、「**マウント**」をクリックしてください。

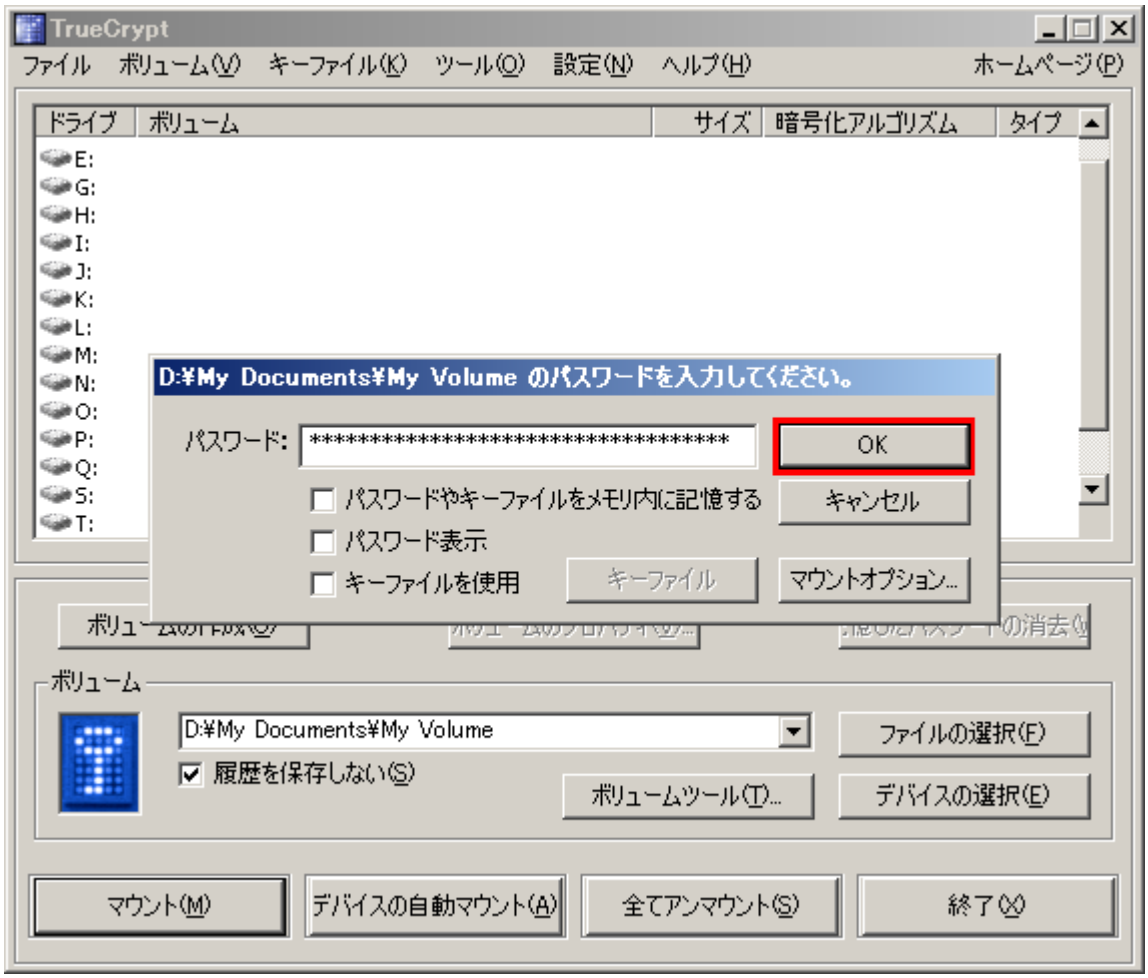
パスワード入力を求めるダイアログが表示されます。

ステップ **17:**



ステップ 10 で設定したパスワードをパスワード入力欄(赤で囲んである)に記入してください。

ステップ **18:**

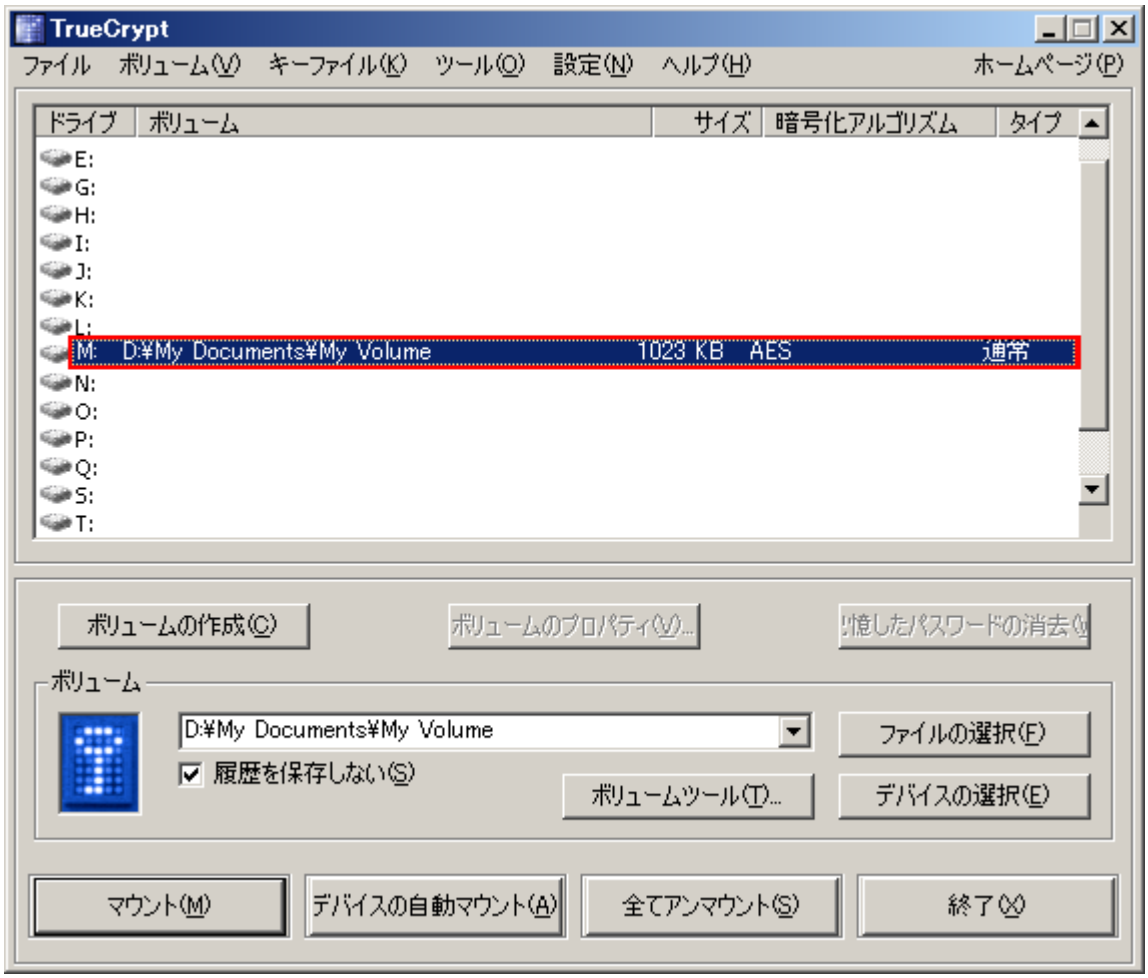


パスワード入力ウィンドウの「**OK**」をクリックしてください。

TrueCrypt はボリュームをマウントしようとします。もしパスワードが一致しなければ(たとえばパスワード入力を間違えたとか)、その旨が報告され、前のステップに戻って、パスワードを再入力し OK をクリックすることになります。パスワードが一致すれば、ボリュームはマウントされます。

(次のページに続く)

最終ステップ:



これでコンテナを仮想ディスク M:としてマウントできました。

仮想ディスクは全体(ファイル名、アロケーションテーブル、空き領域など)が暗号化されており、実際のディスクと同じに扱えます。ファイルをそこに保存(またはコピー、移動)すれば、書込時に即時に暗号化されます。

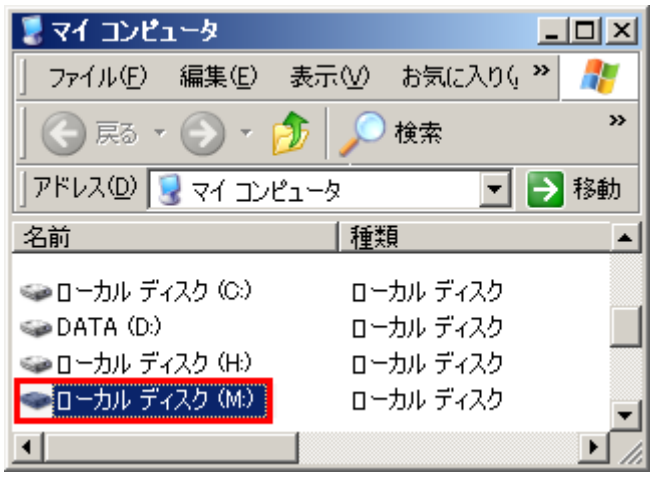
たとえばメディアプレーヤーで TrueCrypt ボリュームにあるファイルを開くと、ファイルは読み出し時に即時に RAM(メモリ)に復号されます。

重要: TrueCrypt ボリュームにファイルを保存したりコピーしたするときには、パスワード入力を求められません。パスワードはボリュームをマウントするときに必要なだけです。

上のスクリーンショットでいえば、赤で囲まれた項目をダブルクリックすることで、マウントされたボリュームを開くこともできます。

(次のページに続く)

また、通常のボリュームを参照するのと同じ方法でマウントされたボリュームを参照することもできます。たとえば、*コンピュータ*(または*マイ コンピュータ*)を開いて該当のドライブ文字(この場合は M:)をダブルクリックするということです。

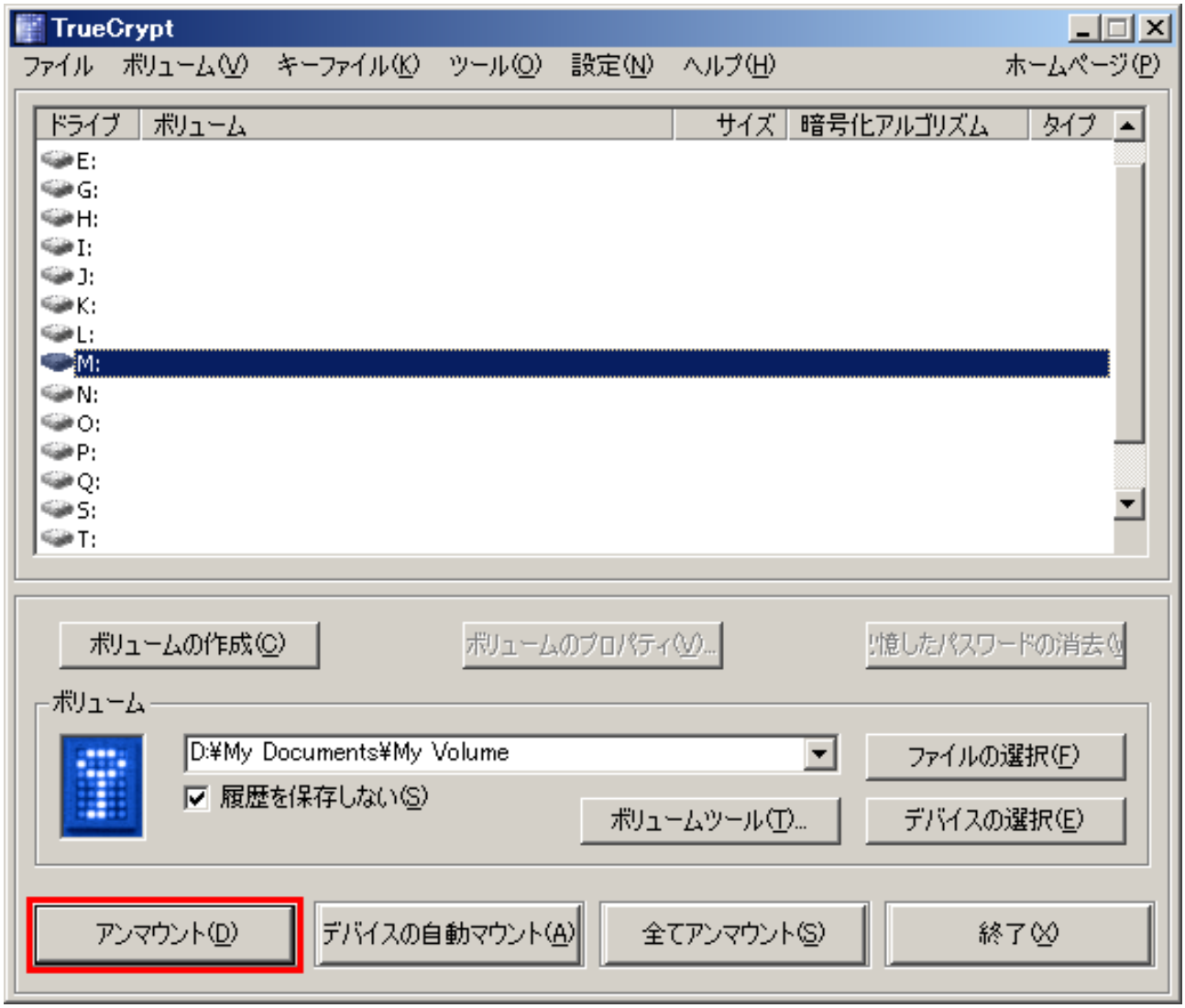


ファイルを TrueCrypt.ボリュームから、あるいは TrueCrypt ボリュームへコピーするのははかの通常のディスクに対するのと同じに実行できます。(たとえば、単純にドラッグ・アンド・ドロップもできます) 暗号化された TrueCrypt ボリュームから読み出したりコピーしたりされるファイルは、自動的に即時に(メモリ/RAM に)復号されます。
同様に、暗号化された TrueCrypt ボリュームに書き込まれるファイルは、自動的に(ディスクに書き込まれる直前に)RAM に暗号化されます。

TrueCrypt は絶対に復号されたデータをディスクに書き込みません。臨時に RAM(メモリ)に置くだけです。ボリュームがマウントされていても、そのボリュームにあるデータは暗号化されたままです。Windows を再起動したり電源を切ったりしたときにはボリュームはアンマウントされそこに保存されたファイルは(暗号化されて)アクセスできなくなります。電源供給が(正しい手順ではなく)突然停止してもボリュームのファイルはアクセスできなくなります。もう一度アクセスするには、そのボリュームをマウントする必要があります。この手順はステップ 13-18 です。

(次のページに続く)

ボリュームを閉じてそこに保存されたファイルにアクセスできないようにするには、OS を再起動するかボリュームをアンマウントしてください。それは以下の手順で実行します。



主 TrueCrypt ウィンドウのマウントされたボリュームのリスト(上のスクリーンショットで赤く囲まれた部分)を選択し、**アンマウント**(同様に赤で囲まれています)をクリックしてください。そこに保存されたファイルに再度アクセス可能にするには、ステップ 13-18 を再度実行してください。

TrueCrypt パーティション/デバイスの作り方と使い方

ファイルコンテナのかわりに、物理的パーティションやデバイスを暗号化する(TrueCrypt デバイス型ボリューム)ことができます。 これを実行するには、このチュートリアルのステップ 1-18 を実行してください。ただし、すべての関連ステップで「**ファイルの選択**」ではなく「**デバイスの選択**」をクリックしてください。

重要: このマニュアルの他の章にはチュートリアルを簡単にするため省略した重要な情報が含まれています。それらの章もぜひ読んでください。

みせかけの拒否

敵対者があなたにパスワードを明かすことを強制するような場合、TrueCrypt は 2 レベルのみせかけの拒否法をあなたに提供します。

1. 隠しボリューム(詳細は後記の「隠しボリューム」の節を参照)

2. TrueCrypt ボリュームであるかどうかを特定するのは不可能です。復号されるまでは TrueCrypt ボリュームはランダムなデータにしか見えません。（TrueCrypt ボリュームであるという「署名」のようなものはありません） ですから、あるファイル、パーティション、デバイスが TrueCrypt ボリュームであるとか暗号化されているということを証明することはできません。

TrueCrypt コンテナ(ファイル型ボリューム)は、どんな拡張子をつけてもかまいません。(たとえば .raw, .iso, .bin, .img, dat, .rnd, .tc) または拡張子がなくてもまったくかまいません。 TrueCrypt は拡張子を無視します。もし「みせかけの拒否」をする必要があれば、 TrueCrypt ボリュームに .tc という拡張子をつけてはいけません。(この拡張子は'公式に'TrueCrypt に関連づけられているからです)

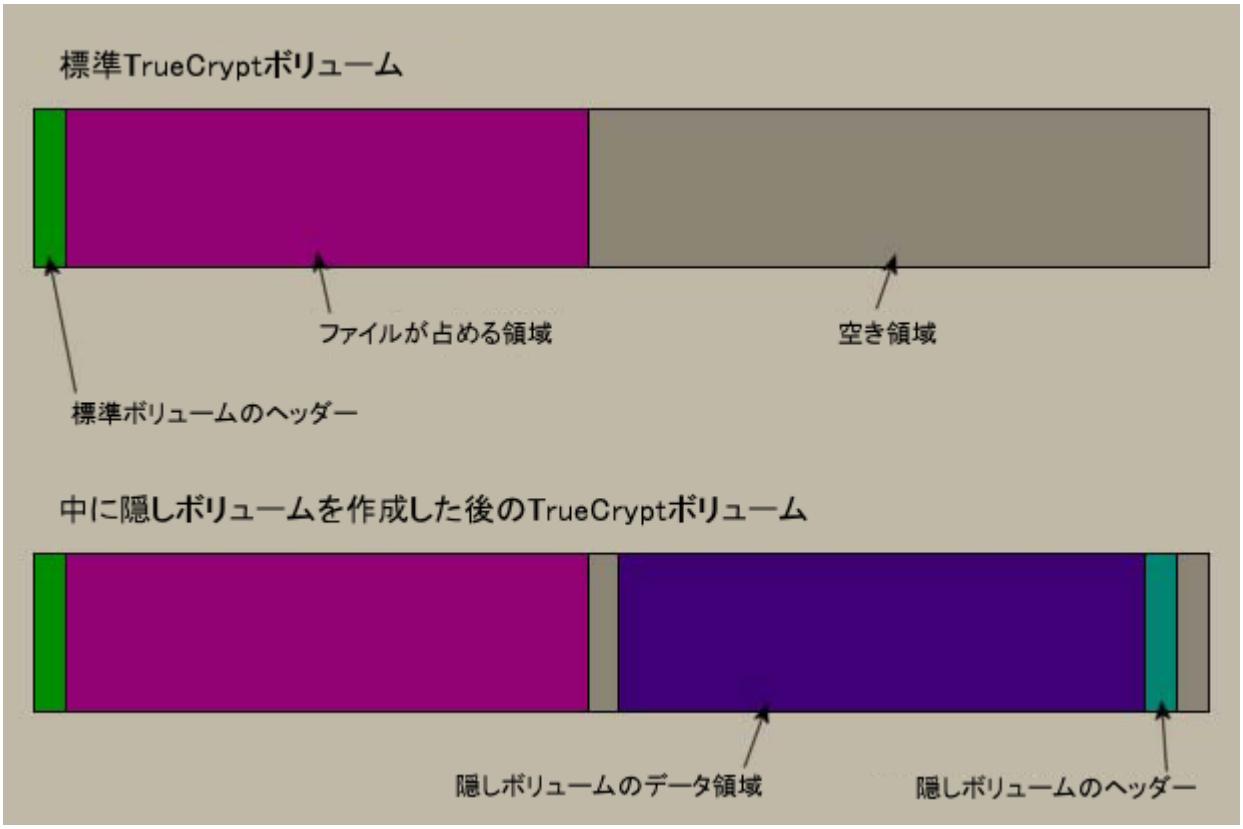
ハードディスクのパーティションを TrueCrypt ボリュームとしてフォーマットする場合でも、パーティションテーブル(パーティションタイプを含む)は変更されません。(TrueCrypt の署名や ID のようなものはパーテションテーブルには書き込まれません)

TrueCrypt がファイル型ボリュームにアクセス(アンマウント、マウント試行、パスワードの作成と変更、中に隠しボリュームを作成、など)したりキーファイルにアクセスしたりするどのような場合でも、コンテナやキーファイルのタイムスタンプ(コンテナやキーファイルの最終アクセス[1]、変更日時)は変更されません。ただし、設定でこの機能を無効にしていなければ、です。

[1]Windows でコンテナやキーファイルのタイムスタンプを見るために「プロパティ」を参照(コンテナやキーファイルを右クリックして「プロパティ」を選択)すると、コンテナやキーファイルの*アクセス日時*を変更することになります。また、Windows のファイル選択でサムネール表示にする(縮小表示モードでコンテナやキーファイルを選択する場合)には、Windows が*アクセス日時*を変更することがあります。

隠しボリューム

誰かが暗号化ボリュームのパスワードを明かすよう強要するかもしれません。それを拒否できない状況、たとえば脅迫などもあり得ます。そこで、いわゆる「隠しボリューム」を使うことで、ボリュームのパスワードを明かさずに策略で解決する方法があります。



隠しボリューム作成前後の標準 TrueCrypt ボリュームの状態

他の TrueCrypt ボリュームの空き領域に TrueCrypt ボリュームを作るというのが、ポイントです。外殻ボリュームがマウントされた状態でも、それが隠しボリュームを含むかどうかを判断することはできません。なぜなら、どの TrueCrypt ボリュームの空き領域も作成時[1]にランダム値で埋められているからです。そして、マウントされていない隠しボリュームのどの部分もランダムデータと区別できません。また、TrueCrypt は外殻ボリュームのファイルシステム(空き領域情報など)を変更することもありません。

[1]クイックフォーマットとダイナミックのオプションは使用不可になっています。空き領域をランダムデータで満たす方法については、「技術解説」の章、「TrueCrypt ボリュームフォーマット仕様」を参照してください。

隠しボリュームのパスワードは、外殻ボリュームのパスワードとは異なったものでなくてはいけません。隠しボリュームを作成する前に、外殻ボリュームには本当には隠そうとは思っていない何か秘密情報らしいファイルをいくつかコピーしておいてください。これらのファイルは、あなたにパスワードを明かすことを強要する人に見せるためのものです。隠しボリュームのパスワードは守り、外殻ボリュームのものだけを明かせばいいのです。本当に秘密にしたいファイルは隠しボリュームに入れてください。

隠しボリュームは通常の TrueCrypt ボリュームと同じ手順でマウントできます。「*ファイルの選択*」または「*デバイスの選択*」をクリックし外殻ボリュームを選択してください。(重要: それらがすでにマウントされて*いない*ことを確認してください)　「マウント」をクリックし、隠しボリュームのパスワードを入力してください。隠しボリュームがマウントされるか、外殻ボリュームがマウントされるかは、入力されたパスワードで決定されます。(つまり、外殻ボリュームのパスワードを入力すれば外殻ボリュームが、隠しボリュームのパスワードを入力すれば隠しボリュームがマウントされます)

TrueCrypt は最初に、入力されたパスワードを使って標準ボリュームヘッダーを復号しようとします。もし失敗すると隠しボリュームのヘッダーが存在する可能性があるセクター(ボリュームの終端からの第 3 セクター)を RAM に読み込み入力されたパスワードで復号しようとします。
隠しボリュームのヘッダーはそれとわかるようにはなっていないことに留意してください。それはまったくランダムなデータとしか見えません。ヘッダーがうまく復号できたら(TrueCrypt がどうやってうまく復号できたかを判断するかについては、「暗号化の仕組み」の節を参照)、復号されたヘッダー(RAM に保持)から隠しファイルのサイズについての情報が得られ、隠しボリュームがマウントされます。(そのサイズはオフセットを決定することにもなります)

隠しボリュームはどのようなタイプの TrueCrypt ボリュームにでも作成することができます。ファイル型にでもパーティション/デバイス型(管理者権限が必要)にでも、です。TrueCrypt の隠しボリュームを作成するには、メインウィンドウで「*ボリュームの作成*」をクリックし「***TrueCrypt 隠しボリュームを作成する***」を選択してください。ウィザードは TrueCrypt 隠しボリュームを作成するためのヘルプと必要な情報を表示します。
隠しボリューム作成時に、隠しボリュームが外殻ボリュームのデータを上書きしてしまわないように隠しボリュームの容量を決めるのは、経験のないユーザーには難しい、あるいはほとんど不可能です。ですから、ボリューム作成ウィザードは隠しボリュームが生成される前に外殻ボリュームのクラスタ配置を調べて、隠しボリュームを作成可能な最大容量を決めます。[1]

隠しボリュームは FAT の TrueCrypt ボリュームにだけ作成できます。(外殻ボリュームのファイルシステムは FAT12, FAT16, または FAT32 に制限されます) NTFS ファイルシステムは(FAT とは対照的に)いろいろなデータをボリューム全体に散らばって格納するので、隠しボリュームを作る余地をほとんど残してくれません。したがって、ボリューム作成ウィザードは外殻ボリュームのファイルシステムとして NTFS を選択することを防止します。隠しボリューム自体はお好みのどのようなファイルシステムでもかわまず、(ファイル形式の)外殻ボリュームはどんなファイルシステムにでも格納できます。

[1]この機能は Windows 版のみに実装されています。ウィザードは外殻ボリュームの終端に一致する連続した空き領域のサイズを得るように、クラスタ配置を調査します。それが得られれば、この領域が隠しボリュームとなり、隠しボリュームの可能な最大容量となります。

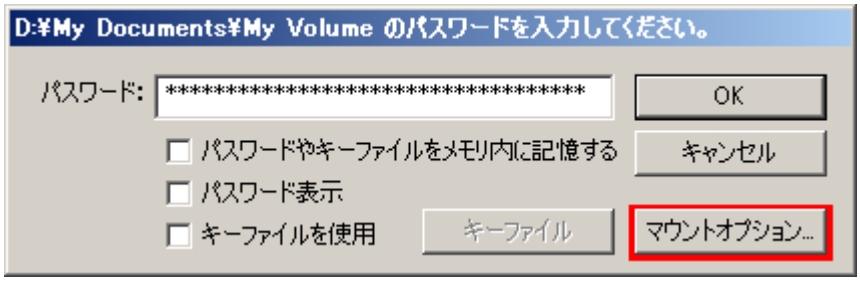
補足: 外殻ボリュームがなぜ FAT なのかと聞かれたら、すべての設定を初期値(デフォルト)のままにしたから、と答えてください。(TrueCrypt ではどのボリュームも初期値は FAT です) NTFS ではなく FAT を使うほかの理由もあります。(たとえば、FAT のほうが早く、断片化しにくいとかです)

隠しボリュームを作成するのに何か問題があれば、「問題が起こったら」の章で解決策を探してください。

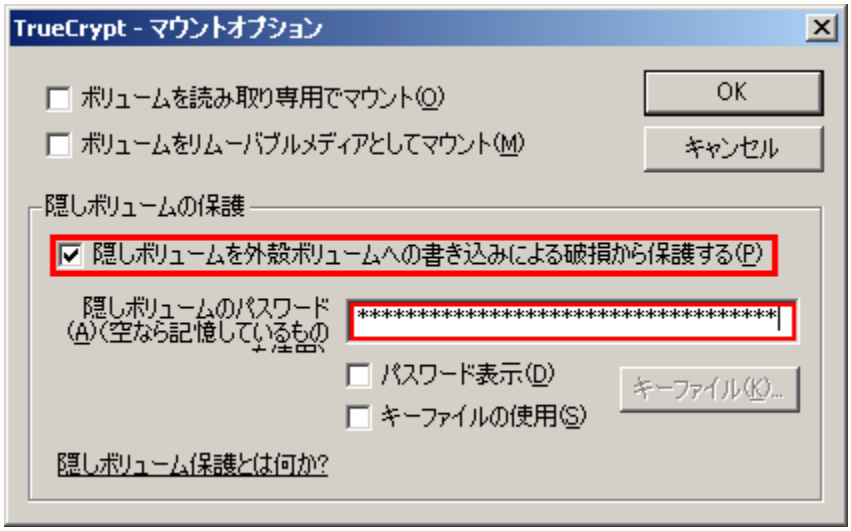
隠しボリュームを破損から守る

隠しボリュームを含む TrueCrypt ボリュームをマウントすると、何の危険もなしに外殻ボリュームのデータを読むことができます。しかし、外殻ボリュームにデータを保存しようとすると、隠しボリュームの一部が上書きされ破損する危険があります。これを防ぐため、ここで記載する方法で保護してください。

外殻ボリュームをマウントするときに、パスワードを入力し、「*OK*」をクリックする前に「*マウントオプション*」をクリックしてください。



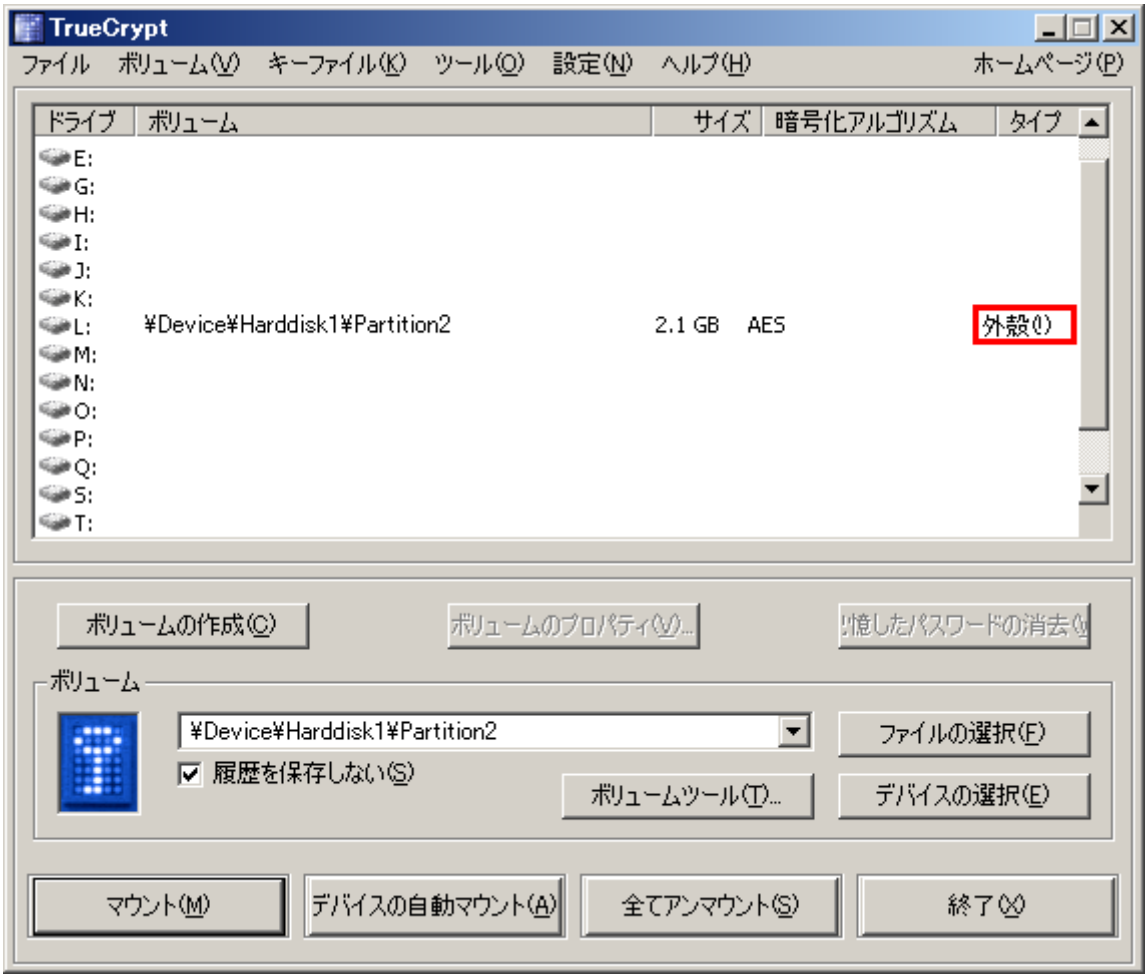
「*マウントオプション*」ダイアログで「*隠しボリュームを外殻ボリュームへの書き込みによる破損から保護する*」を有効にしてください。つぎに「*隠しボリュームパスワード*」の入力欄に隠しボリュームのパスワードを記入してください。そして「*OK*」をクリックし、メインパスワード入力ダイアログの「*OK*」をクリックしてください。



両方のパスワードが正しいものでなくてはいけません。そうでなければ、外殻ボリュームはマウントされません。隠しボリューム保護が有効な場合、TrueCrypt は隠しボリュームをマウントするのではなく(RAM にある)ヘッダーを復号し、隠しボリュームのサイズを(復号されたヘッダーから)得るだけです。そして、外殻ボリュームがマウントされ、(外殻ボリュームがアンマウントされるまで)隠しボリューム領域へのどんなデータ保存も拒否されます。**TrueCrypt は外殻ボリュームのファイルシステム(クラスタ割り当て情報、空き領域情報など)をいっさい変更しません。ボリュー**

ムがアンマウントされると、ただちに保護は機能しなくなります。そのボリュームが再マウントされても、そのボリュームが隠しボリューム保護に使われているかどうかの判別はできません。隠しボリューム保護機能はユーザーが隠しボリューム用の正しいパスワード(またはキーファイル)を入力/提供した場合のみ、有効となります。

隠しボリューム領域への書き込み動作が(隠しボリューム保護のため)拒否/防止されるとただちにホストボリューム(外殻ボリュームと隠しボリュームの両方)はアンマウントされるまで書き込み不可に設定(TrueCrypt ドライバーがそのボリュームへの書き込みに対して「不正なパラメータ」エラーを返す)されます。これが「みせかけの拒否」を守ります。(そうでなければ、ある種のファイルシステムの矛盾がそのボリュームが隠しボリューム保護を使っているということを示してしまうかもしれません) 隠しボリューム破損が防止されると、警告が表示されます。(TrueCrypt が常駐している場合のみ表示 -「TrueCrypt の常駐」を参照)
さらに、メインウィンドウで表示されるマウントされている外殻ボリュームのタイプは「外殻(!)」に変わります。



さらに、「ボリュームプロパティ」の「隠しボリューム保護」フィールドでは「はい(破損は防止されました!)」と表示されます。

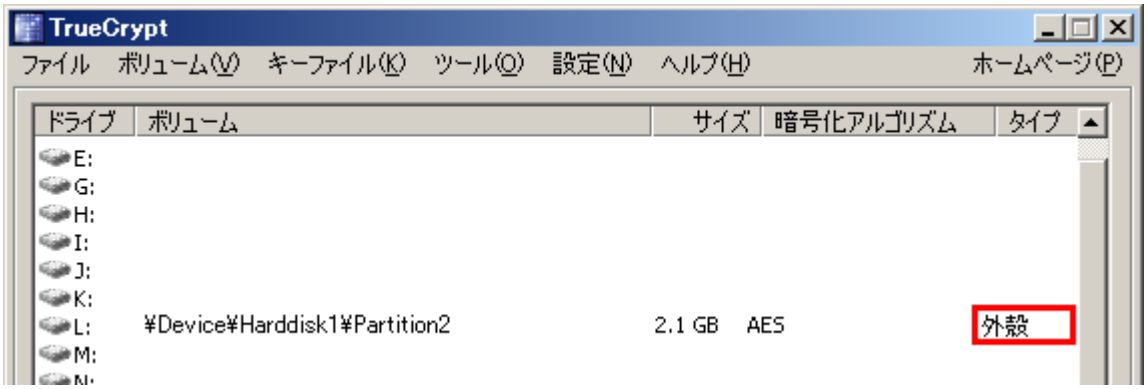
隠しボリュームの破損が防止されても、そのことについての情報はボリュームには書き込まれません。外殻ボリュームをアンマウントして、ふたたびマウントしてもボリュームプロパティには「破損は防止されました」というメッセージは表示されません。

隠しボリュームが破損から保護されているかどうかを調べるには、いくつかの方法があります。

1. 外殻ボリュームがマウントされたあと、確認メッセージボックスに隠しボリュームは保護されているという表示がでます。(これが表示されなければ、隠しボリュームは保護されていません)

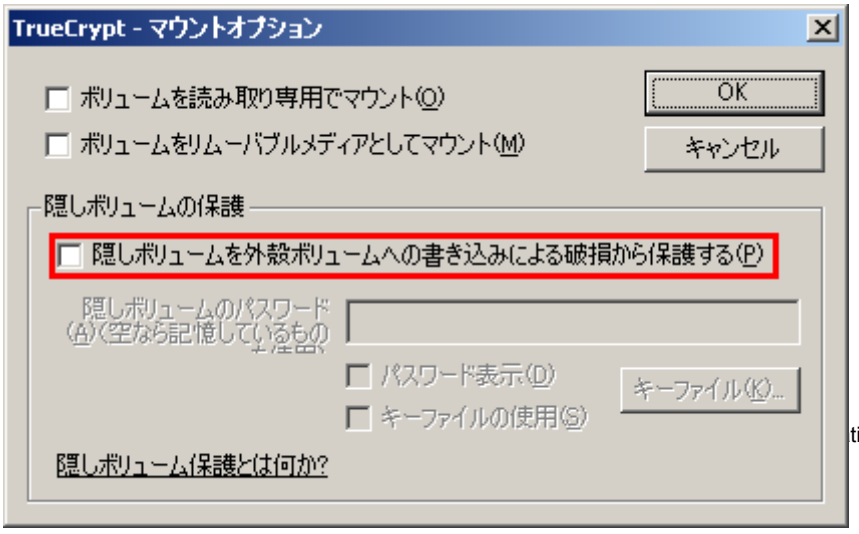
2. 「ボリュームプロパティ」ダイアログで、「隠しボリューム保護」は「はい」になります。

3. マウントされた外殻ボリュームは「外殻」 と表示されます。



重要: 敵対者が外殻ボリュームをマウンとするよう求めてきた場合には、当然のことながら隠しボリューム保護を有効にして外殻ボリュームをマウントしてはいけません。隠しボリューム保護を有効にして外殻ボリュームをマウントしていると、敵対者は(ボリュームがアンマウントされるまでは)外殻ボリュームに隠しボリュームが存在することを発見することができてしまうことに注意してください。

警告: 「*マウントオプション*」ダイアログウィンドウの「*隠しボリュームを外殻ボリュームへの書き込みによる破損から保護する*」というオプションは、マウント試行が完了すると、マウントが成功したか否かによらず自動的に不可になります。(すでに保護されているすべての隠しボリュームは、もちろん保護されたままです) したがって、(隠しボリュームを保護したいなら)外殻ボリュームをマウントしようとするときには*毎回*オプションをチェックする必要があります。



キャッシュされたパスワードを使って、隠しボリュームを保護しながら外殻ボリュームをマウントしたいなら、次のステップで実行してください。: コントロール(Ctrl)キーを押しながらマウントをクリック(または、ボリュームメニューの「*オプション付でマウント*」をクリック。「*マウントオプション*」ダイアログが開きます。「*隠しボリュームを外殻ボリュームへの書き込みによる破損から保護する*」を有効にしてください。そして、パスワードを空欄のままにし、「*OK*」をクリックしてください。

外殻ボリュームをマウントする必要があるが、どのようなデータもそこに保存しないということがわかっている場合には、隠しボリュームを破損から保護する最も簡単な方法は読み出し専用で外殻ボリュームをマウントすることです。(「マウントオプション」参照)

隠しボリューム区画づくりの前の安全策

- もし敵対者が、アンマウントされた TrueCrypt ボリュームの特定の場所を何回もアクセスすると、ボリュームのどのセクターに変更があったかをつきとめることができます。あなたがファイルをつくったり、コピーしたり、ファイルの更新、削除、リネーム、移動などで隠しボリュームの内容に変更を加えると、隠しボリュームにあるセクターの暗号化された内容は変更されることになります。外殻ボリュームのパスワードを教えたにもかかわらず、なぜこれらのセクターの内容に変更が生じているのかについて追求されるかもしれません。あなたの回答如何によっては、相手はボリュームに隠されたボリュームがあると疑うかもしれません。

  上記の問題は以下のような場合でも起きる可能性があります。

  - ファイル型 TrueCrypt コンテナをデフラグして、ホストボリューム(デフラグされたファイルシステム)の空き領域にコンテナや断片のコピーが残っている場合。これを防ぐには以下のどれかを実行してください。
    - ファイル型のかわりに、パーティション/デバイス型 TrueCrypt ボリュームを使う
    - ホストボリューム(デフラグされたファイルシステム)の空き領域に完全消去をかける
    - TrueCrypt ボリュームを格納するファイルシステムではデフラグをしない

  - ファイル型 TrueCrypt コンテナが(NTFS のような)ジャーナリングファイルシステムに格納されている場合。TrueCrypt コンテナあるいはその断片のコピーがホストボリュームに残る可能性があります。これを防ぐには以下のどれかを実行してください。
    - ファイル型のかわりに、パーティション/デバイス型 TrueCrypt ボリュームを使う
    - FAT32 のようなジャーナリング機能を持たないファイルシステムにコンテナを格納する

  - ファイル型 TrueCrypt ボリュームがウェアレベリング機構を持つデバイス(たとえば、いくつかの USB フラッシュメモリ)に格納されている場合。TrueCrypt ボリュームの断片のコピーがそのデバイスに残る可能性があります。ウェアレベリングについての詳細は「安全のための予防策」の「ウェアレベリング」を参照してください。

- 隠しボリュームを作ろうとするパーティション/デバイスを暗号化するときには、*クイックフォーマット*はしてはいけません。

- 隠しボリュームをつくろうとするボリュームのどのファイルも削除していないことを確認してください。(クラスタ配置調査ツールでは、削除されたファイルを検出できません)

- Linux ではファイル型 TrueCrypt ボリュームの中に隠しボリュームを作る場合には、スパース(sparse)ファイルシステムのボリュームであってはいけません。(Windows 版では TrueCrypt はスパースファイルシステムを区別し、その中に隠しボリュームを作ることはできないようになっています)

TrueCrypt ボリューム

二つのタイプの TrueCrypt ボリュームがあります:

- ファイル型 (コンテナ)
- パーティション/デバイス型

TrueCrypt ファイル型ボリュームは、どんな記憶装置にでも存在することができる通常のファイルです。これは内部に、暗号化され完全に独立した仮想ディスク・デバイスを含みます。

TrueCrypt パーティションは TrueCrypt で暗号化されたハードディスクのパーティションです。ハードディスク、USB ハードディスク、フロッピーディスク、USB メモリスティック、および他の形式の記憶装置の全体をを暗号化することもできます。

新規 TrueCrypt ボリュームの作成

新しく TrueCrypt のファイル形式ボリュームを作ったりパーティションを暗号化(管理者権限が必要)するには、メインウィンドーの「ボリュームの作成」をクリックしてください。TrueCrypt ボリューム作成ウィザードが現れます。ウィザードは現れたらすぐに、新規ボリュームのためのマスターキー、第二キー(LRW モード)、ソルトを生成するためのデータを集めはじめます。収集されたデータは可能なかぎりランダムであるべきで、マウスの動き、マウスボタンのクリック、キーストロークなどを含み、システムから集められます。(詳細は、「乱数発生機構」を参照) ウィザードは、新規 TrueCrypt ボリュームを確実に作るために必要な情報とヘルプを提供します。しかしながら、いくつかの項目ではさらに詳細な説明が必要です。

ハッシュアルゴリズム

TrueCrypt がどのハッシュ・アルゴリズムを使うかを選択することができます。選択されたハッシュ・アルゴリズムは、マスターキー、第二キー(LRW モード)、ソルトを生成する乱数発生機構(疑似乱数混合関数)で使われます。(詳細は「乱数発生機構」を参照) また、これはボリュームの新規ヘッダーキー、第二ヘッダーキーを導出することにも使われます。(詳細は「ヘッダーキーの導出、ソルト、および反復回数」を参照)

実装されているハッシュアルゴリズムについては、「ハッシュアルゴリズム」を参照してください。

ハッシュ関数の出力はけっして直接には暗号化キーとして使われないことを知っておいてください。詳細は「技術解説」を参照してください。

暗号化アルゴリズム

新規ボリュームを暗号化する暗号化アルゴリズムを選択することができます。暗号化アルゴリズムはボリューム作成後には変更できないことに注意してください。詳細は「暗号化アルゴリズム」を参照してください。

クイックフォーマット

ここにチェックが入っていない場合、新規ボリュームの各セクターはフォーマットされます。このことは、新規ボリュームはランダムなデータで完全に満たされるということを意味します。クイックフォーマットははるかに速く実行されますが、安全性は劣ります。なぜなら、ボリューム全体がファイルで満たされるまでは、(空き領域がランダムデータで前もって満たされなかった場合には)どれだけのデータがそのボリュームに存在するかがわかってしまうかもしれないからです。クイックフォーマットをしてもよいかどうか判断がつかない場合には、このオプションにチェックをいれないことを勧めます。パーティション/デバイスを暗号化する場合のみ、クイックフォーマットが可能になることに注意してください。
重要:　隠しボリュームを後で作成するつもりのパーティション/デバイスを暗号化する場合は、このオプションにチェックをいれないでください。

ダイナミック

ダイナミックな(動的な)TrueCrypt コンテナは、データの増加にともなって物理的容量(実際のディスク上のサイズ)が増加する NTFS スパースファイルに割り当てられます。TrueCrypt ボリューム上でファイルを削除してもコンテナの物理的容量(実際にディスク上でコンテナが占めるサイズ)は減少しないことに留意してください。コンテナの物理的容量はボリューム生成過程でユーザーがきめた最大値まで増加するだけです。きめられた最大値に達すると、コンテナの物理的容量はそこで一定することになります。

スパースファイルは NTFS ファイルシステムにのみ作成することができます。FAT ファイルシステムにコンテナを作るときには「ダイナミック」オプションは選択不可になります。

Windows や TrueCrypt から返されるダイナミックな(スパースファイルの)TrueCrypt ボリュームのサイズは常にボリューム作成時に指定した最大容量になります。コンテナの現在の物理的容量(ディスク上の実際のサイズ)を知るには、、Windows のエクスプローラウィンドウでコンテナファイルを右クリックして、プロパティを選び「ディスク上のサイズ」を見てください。(TrueCrypt のウィンドウ上では、このようになりません)

警告: ダイナミックな(スパースファイルの)TrueCrypt ボリュームでの速度は通常のボリュームより大きく悪化します。また、ダイナミックな(スパースファイルの)TrueCrypt ボリュームはどのセクターが未使用かを知ることができるので、セキュリティも劣ります。さらに、ホストファイルシステムに充分な空き領域がない場合にダイナミックなボリュームに書き込みをすると、暗号化したファイルシステムが破損する可能性があります。

クラスタのサイズ

クラスタはファイル配置の単位です。 例えば、1 バイトのファイルのために FAT ファイルシステムで少なくとも 1 個のクラスタを割り当てられます。ファイルがクラスタ境界を越えて大きくなると、別のクラスタが割り当てられます。理論的に、クラスタサイズが大きくなるほど、(性能はあがりますが)ディスクにより多く無駄な部分が増えます。クラスタサイズにどのような値をセットすればいいかわからなければ、初期値のままにしておいてください。

CD や DVD にある TrueCrypt ボリューム

TrueCrypt ボリュームを CD や DVD に置きたい場合には、まずファイル形式のボリュームをハードディスクに作成してください。それから、CD/DVD 書き込みソフト(Windows XP/Vista ならば、OS 標準の CD 書き込みツールでも可)でそれを CD/DVD に書き込んでください。

Windows2000 で読み出し専用メディア(CD/DVD 他)にある TrueCrypt ボリュームをマウントする場合には、TrueCrypt ボリュームを FAT でフォーマットしなくてはならないことを憶えておいてください。なぜなら Windows2000 では読み取り専用メディアの NTFS ファイルシステムはマウントできないからです。(Windows XP/Vista なら可能です)

ハードウェア/ソフトウェア・レイドと Windows ダイナミックボリューム

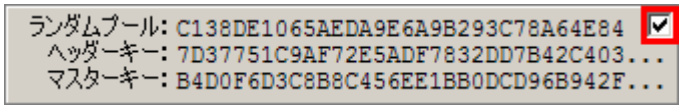
TrueCrypt はハードウェア/ソフトウェア・レイドと同様に Windows のダイナミックボリュームをサポートします。Windows のダイナミックボリュームを TrueCrypt ボリュームとしてフォーマットする場合には(Windows のディスク管理ツールを使って)ダイナミックボリュームを作成したあと、システムの再起動が必要です。そうすれば TrueCrypt ボリューム作成ウィザードの「デバイス選択」に目的のボリュームが表示され、選択できるようになります。

「デバイス選択」ウィンドーで、ダイナミックボリュームは単一のデバイス(項目)としては表示されません。そのかわり、ダイナミックボリュームを構成するすべてのボリュームが表示されるので、ダイナミックディスク全体をフォーマットするために、そのうちのどれか一つを選択してください。

ボリューム作成に関する追加情報

ボリューム作成ウィザードの最終段階で「フォーマット」ボタンをクリックしたあと、システムが追加のランダムデータを得るのに少し間があきます。その後、新規ボリュームのためのマスターキー、ヘッダーキー、第二キー(LRW モード)、ソルトなどが生成され、マスターキーとヘッダーキーの内容が表示されます。

セキュリティを強化するために、該当のフィールドの右上のチェックボックスにチェックを入れないことで、ランダムプール、マスターキー、ヘッダーキーの内容を表示しないようにできます。



プール/キーの最初の 128 ビットだけが表示されます(全体の内容ではありません)

警告: *ハードドライブパーティション/デバイス全体を暗号化するとき(**TrueCrypt** ボリュームとしてフォーマットするとき)には、パーティション/デバイスのすべてのデータは失われます！*

重要: 数人のユーザーから、TrueCrypt ボリュームのデータが破損すると報告がありました。その後、これらのユーザーは原因が TrueCrypt ではなくハードウェア(チップセット、USB ハードドライブ ケーブル、USB PCI カード 他)であることを発見しました。ですから、TrueCrypt ボリュームを作ろうとするデバイスに書かれたデータが破損しないか確認することをおすすめします。たとえば、大量のファイル(少なくとも合計 数 GB)をコピーし、その内容を Windows 標準のコマンドラインツール fc でオリジナルと比較するというようなことです。

TrueCrypt では FAT(FAT12、FAT16、FAT32 のいずれかかはクラスタの数で自動的に決定される)か NTFS のボリュームを作成することができます。(しかし、NTFS ボリュームを作成するには、管理者権限が必要です) マウントされた TrueCrypt ボリュームは、いつでも FAT(FAT12、FAT16、FAT32)や NTFS にフォーマットしなおすことができます。これらは通常のディスク・デバイスと同じに扱うことができるので、マウントされた TrueCrypt ボリュームのドライブレターを(たとえば、「コンピュータ」または「マイ コンピュータ」の中で)右クリックしてフォーマットを選択してください。

TrueCrypt ボリュームに関する詳細については、「隠しボリューム」も参照してください。

メインプログラムウィンドウ

ファイルの選択

ファイル形式の TrueCrypt ボリュームを選びます。選択したあとで、いろいろな操作(たとえば「マウント」をクリックすることでマウント)ができます。ボリュームのアイコンを TrueCrypt.exe のアイコンまたはメインプログラムウィンドウにドラッグ&ドロップして、TrueCrypt を起動させることもできます。

デバイスの選択

TrueCrypt パーティションか記憶デバイス(たとえばフロッピーディスクや USB メモリスティック)を選びます。選択したあとで、いろいろな操作(たとえば「マウント」をクリックすることでマウント)ができます。

補足: TrueCrypt パーティション/デバイスをマウントするもっと簡単な方法があります。「デバイスの自動マウント」を参照してください。

マウント

「マウント」をクリックすると、TrueCrypt はキャッシュにパスワードがあればそれを使ってマウントしようとします。キャッシュになければ、ユーザーにパスワード入力を要求します。正しいパスワードを入力すれば、マウントされることになります。正しいパスワードを入力するか(あるいは正しいキーファイルを指定すれば)、ボリュームはマウントされます。
重要： TrueCrypt アプリケーションを終了しても TrueCrypt ドライバーが機能しており、どの TrueCrypt ボリュームもアンマウントされません。

デバイスの自動マウント

この機能を使うと(「デバイスの選択」を使って)手動で目的のパーティション/デバイスを選択しなくても TrueCrypt パーティション/デバイスをマウントすることができます。TrueCrypt はあなたのシステムの有効なパーティション/デバイスのヘッダーを調べて、それぞれを TrueCrypt ボリュームとしてマウントしようとします。TrueCrypt パーティション/デバイスであるかどうかは特定できず、使われている暗号の種類も特定できないことに注意してください。ですから、プログラムは目的の TrueCrypt パーティションを直接には見つけることはできません。そのかわり、TrueCrypt は暗号化されていてもいなくても、すべての暗号化アルゴリズムと(存在するなら)キャッシュにあるパスワードを使って、パーティション/デバイスを一つずつ試します。このため遅いマシンでは、このプロセスに長時間かかることは了承してください。

入力したパスワードが不正であれば、キャッシュのパスワードを(存在すれば)使ってマウントを試行します。デバイスの自動マウントでは、空のパスワードを入力し「*キーファイルの使用*」にチェックが入っていなければ、パーティション/デバイスのマウント試行にはキャッシュされたパスワードのみが使われます。マウントオプションを設定する必要がなければ、「*デバイスの自動マ*

ウント」でシフトキーを押しながらクリックすることで、パスワード入力要求をとばしてしまうこともできます。(この場合、存在すればキャッシュされたパスワードのみが使われます)

ドライブレターはメインウィンドウのドライブリストで選択された最初のものに割り当てられます。

アンマウント

TrueCrypt ボリュームをアンマウントすると、そのボリュームは読み書き不可になります。

すべてアンマウント

TrueCrypt ボリュームをアンマウントすると、そのボリュームは読み書き不可になります。この機能は、現在マウントされているすべての TrueCrypt ボリュームをアンマウントします。

記憶したパスワードの消去

ドライバのメモリーに記憶(キャッシュ)されたすべてのパスワード(処理されたキーファイルの内容を含む)を消去します。キャッシュにパスワードが存在しなければ、このボタンは押せないようになっています。(詳細は「パスワードをドライバのメモリーに記憶する」を参照)

履歴を保存しない

これが無効になっていると、マウントしたボリュームの直近 20 件のファイル名やパスは履歴ファイルに保存されます。(履歴はメインウィンドウのボリュームのコンボボックスをクリックすると表示されます) このオプションが有効になると、TrueCrypt はコンテナやキーファイルが Windows のファイル選択でどこから選択されていようとも Windows のファイル選択機能が TrueCrypt について作成したレジストリエントリをクリアし、現在のディレクトリをユーザーのホームディレクトリとして設定します。(トラベラーモードの場合は、TrueCrypt が起動されたディレクトリに設定します) ですから、Windows のファイル選択機能は最後にマウントされたコンテナ(または最後に選択されたキーファイル)のパスを記憶しません。さらに、このオプションが有効になっていれば、TrueCrypt をどこに隠したとしても TrueCrypt の主ウィンドウのボリュームパス入力欄はクリアされます。

補足: 「ツール –> ボリューム履歴の消去」を選んで、ボリューム履歴を消去することができます。

終了

TrueCrypt アプリケーションを終了します。ドライバーは継続して動作し、TrueCrypt ボリュームはアンマウントされません。トラベラーモードのときには、ドライバは必要がなくなれば(つまり、主アプリケーションとボリューム作成ウィザードが閉じられ、マウントされた TrueCrypt ボリュームがない状態になったとき)、メモリーから除去されます。 しかし、TrueCrypt がトラベラーモ

ードで動いているときに、TrueCryptボリュームが強制的にアンマウントされると「終了」してもTrueCryptドライバーは除去されません。(システムを停止するかリスタートする場合のみ、除去されます) これはWindowsのバグによって引き起こされるいろいろな問題(たとえば、アンマウントされたボリュームを使っているアプリケーションがあるとTrueCryptを再起動できない)を防止します。

ボリュームツール

ボリュームパスワードの変更

「ボリューム -> ボリュームのパスワードを変更する」を参照

ヘッダーキー導出アルゴリズムの設定

「ボリューム -> ヘッダーキー導出アルゴリズムの設定」を参照

ボリュームヘッダーのバックアップ

「ツール -> ボリュームヘッダーのバックアップ」を参照

ボリュームヘッダーのリストア

「ツール -> ボリュームヘッダーのリストア」を参照

プログラムメニュー

注意: 自明のメニュー項目は、このドキュメントでは説明しません。

ファイル -> 終了

TrueCrypt アプリケーションを終了します。ドライバーは継続して動作し、どの TrueCrypt ボリュームもアンマウントされません。トラベラーモードのときには、ドライバは必要がなくなれば(つまり、主アプリケーションとボリューム作成ウィザードが閉じられ、マウントされた TrueCrypt ボリュームがない状態になったとき)、メモリーから除去されます。 しかし、TrueCrypt がトラベラーモードで動いているときに、TrueCrypt ボリュームが強制的にアンマウントされると「終了」しても TrueCrypt ドライバーは除去されません。(システムを停止するかリスタートする場合のみ、除去されます) これは Windows のバグによって引き起こされるいろいろな問題(たとえば、アンマウントされたボリュームを使っているアプリケーションがあると TrueCrypt を再起動できない)を防止します。

ボリューム -> デバイスのボリュームをすべて自動でマウント

「デバイスの自動マウントデバイスの自動マウント」の項を参照。

ボリューム -> 現在マウントされているボリュームをお気に入りに保存

この機能はひんぱんに一つあるいは複数の TrueCrypt ボリュームを同時に開いて仕事をし、それらがつねに特定のドライブレターにマウントされている必要がある場合に役にたちます。

すべての現在マウントされているボリューム(およびマウントされているドライブレター)がアプリケーションのデータを保存するフォルダー(たとえば *C:¥Documents and Settings¥YourUserName¥Application Data¥TrueCrypt*)に *Favorite Volumes.xml* という名前のファイルに保存されます。 トラベラーモードでは、ファイルは *TrueCrypt.exe* を起動したフォルダー(*TrueCrypt.exe* が存在するフォルダー)に保存されます。

この機能を使うと、お気に入りに以前に保存したすべてのアンマウントされたボリュームはお気に入りリストから削除されます。

「お気に入り」として保存されたボリュームをマウントするには、*ボリューム -> お気に入りボリュームをマウント*を選択してください。

お気に入りボリュームリストを削除するには、TrueCrypt ボリュームをすべてアンマウントし、「ボリューム -> 現在マウントされているボリュームをお気に入りとして登録」を選択してください。

ボリューム -> お気に入りボリュームをマウント

この機能は、以前に「お気に入り」として保存したボリュームをマウントします。「ボリューム -> 現在マウントされているボリュームをお気に入りに保存」を参照してください。

ボリューム -> ヘッダーキー導出アルゴリズムの設定

この機能は、異なる PRF 関数で導出されたヘッダーキーでボリュームヘッダーの再暗号化を可能にします。(たとえば、HMAC-SHA-1 のかわりに HMAC-Whirlpool を使うということが可能です) ボリュームヘッダーはボリュームを暗号化するマスターキーを含んでいることに留意してください。このため、この機能を使ってもボリュームに保存されたデータはいっさい失われることはありません。詳細は「技術解説」の章、「ヘッダーキーの導出、ソルト、および反復回数」を参照してください。

注意: TrueCrypt がボリュームヘッダーを再暗号化する場合、敵対者が微視的残留磁気[17]から上書きされたヘッダーを復元できないようにするため、最初に元のボリュームヘッダーをランダムデータで 35 回の上書きをします。(「安全のための予防策」も参照)

ボリューム -> ボリュームのパスワードを変更する

現在選ばれている TrueCrypt ボリュームのパスワードを変更することができます。(通常ボリュームか隠しボリュームかを問いません) ヘッダーキーと第二ヘッダーキー(LRW モード)のみが変更され、マスターキーは変更されません。この機能は、新しいパスワードから導出されるヘッダー暗号化キーを使ってボリュームヘッダーを再暗号化します。ボリュームヘッダーはボリュームを暗号化するマスターキーを格納していることに留意してください。ですから、この機能を使ってもボリュームに保存されたデータが失われることはありません。(パスワード変更は、ほんの数秒で完了します)

TrueCrypt ボリュームのパスワードを変更するには、「*ファイルの選択*」か「*デバイスの選択*」をクリックし、ボリュームを選択し、「*ボリュームツール*」メニューで「*ボリュームパスワードの変更*」を選んでください。

「安全のための予防策」も参照してください。

導出アルゴリズム: この入力欄では、新しいボリュームヘッダーキー(詳細は「ヘッダーキーの導出、ソルト、および反復回数」を参照)の導出と新しいソルト(詳細は「乱数発生機構」を参照)を生成するアルゴリズムを選択することができます。

注意: TrueCrypt がボリュームヘッダーを再暗号化する場合、敵対者が微視的残留磁気[17]から上書きされたヘッダーを復元できないようにするため、最初に元のボリュームヘッダーをランダムデータで 35 回の上書きをします。(「安全のための予防策」も参照)

ツール -> ボリューム履歴を消去

直近 20 件の正常にマウントされたボリュームのファイル名(ファイル型の場合)とパスのリストを消去します。

ツール -> トラベラーディスクセットアップ

「トラベラーモード」の章を参照してください。

ツール -> キーファイル生成

「キーファイル」の章、「キーファイル -> ランダムキーファイルの生成」を参照してください。

ツール -> ボリュームヘッダーのバックアップ

TrueCrypt ボリュームにあるすべてのファイルをバックアップするだけの空き領域がない場合、この機能を使って、少なくともボリュームヘッダーだけでもバックアップをとっておくことを強くおすすめします。ここにはマスターキーが記録されています。(バックアップしたファイルのサイズは 1024 バイトになるはずです) ボリュームヘッダーが破損すると、ほとんどの場合はボリュームはマウントできなくなります。
ボリュームヘッダーをバックアップするには、「デバイスの選択」か「ファイルの選択」をクリックし、ボリュームを選択してください。それから「*ツール -> ボリュームヘッダのバックアップ*」をクリックしてください。ヘッダーをリストア(復旧)するには、同じ手順で最後に「*ボリュームヘッダのリストア*」を選択してください。

TrueCrypt のボリュームヘッダーのバックアップは暗号化されたボリュームヘッダーの正確なコピーです。バックアップコピーにはいっさいの追加情報は含まれません。TrueCrypt ボリュームヘッダーバックアップは正しいパスワードを知っているか正しいキーファイルを用意しなければ、復号することはできません。

注意: 標準のボリュームヘッダーと隠しボリュームのボリュームヘッダーが格納される領域とがバックアップ(バックアップファイルにコピー)されます。そのボリュームに隠しボリュームがなかったとしてもです。(隠しボリュームのみせかけの拒否を確実にするため) しかし、ボリュームヘッダーをリストアするときには、隠しボリュームのヘッダーか標準ボリュームのヘッダーかを選択することになります。一度に一つのヘッダーのみをリストアすることができます。両方のヘッダーをリストアする場合は、この機能を二回実行する必要があります。(「ツール」 -> 「ボリュームヘッダのリストア」)

警告: ボリュームヘッダーをリストアすると、ボリュームのパスワードはバックアップ作成時に有効だったものに置き替えられます。さらに、バックアップ作成時にボリュームをマウントするのにキーファイルが必要だった場合には、ボリュームヘッダーをリストア後にボリュームを再マウントするのに同じキーファイルが必要になります。

ボリュームヘッダーバックアップ作成後にボリュームパスワードやキーファイルを変更したために、新しいバックアップを作る必要があるかもしれません。しかしながら、ボリュームヘッダー

は変更されないので、ボリュームヘッダーバックアップは最新の状態のままということになります。

補足: この仕組みは企業などで、ユーザーがパスワードを忘れた(あるいは、キーファイルを失った)場合の対策として使うこともできます。ボリュームを作ったあと、管理者権限を持たないユーザーにそのボリュームの使用を認める前に、(*ツール -> ボリュームヘッダーのバックアップ*を選択して)そのヘッダーのバックアップをとります。パスワード/キーファイルから導出された暗号化されたヘッダーキーで暗号化されているボリュームヘッダーは、ボリュームを暗号化したマスターキーを持っています。そこで、ユーザーにパスワードを選んでもらいその人のためにパスワードを設定します。(*ボリューム -> ボリュームのパスワード変更*) そうすれば、ユーザーにそのボリュームの使用許可を与えるとともに、いつでも管理者の許可や助力なしで任意のパスワードに変更させることができます。ユーザーが自分が決めたパスワードを忘れた場合でも、ボリュームヘッダーのリストアを実行(*ツール -> ボリュームヘッダーのリストア*)をすることで、ボリュームのパスワードをオリジナルの管理者パスワード/キーファイルに戻すことができます。

ツール -> ボリュームヘッダーのリストア

TrueCrypt ボリュームがマウントできなくなった場合にはヘッダーが破損している可能性があります。ボリュームヘッダーをバックアップしておけば、この機能で復旧できます。

ボリュームヘッダーをリストアするときには、隠しボリュームのヘッダーか標準ボリュームのヘッダーかを選択することになります。一度に一つのヘッダーのみをリストアすることができます。両方のヘッダーをリストアする場合は、この機能を二回実行する必要があります。(「ツール」 -> 「ボリュームヘッダのリストア」)

警告: ボリュームヘッダーをリストアすると、ボリュームのパスワードはバックアップ作成時に有効だったものに置き替えられます。さらに、バックアップ作成時にボリュームをマウントするのにキーファイルが必要だった場合には、ボリュームヘッダーをリストア後にボリュームを再マウントするのに同じキーファイルが必要になります。

設定 -> 各種設定

終了時に記憶していたパスワードを消去

有効にされていれば、ドライバのメモリーに記憶されているパスワード(処理されたキーファイルを含む)を、TrueCrypt 終了時に消去します。

パスワードをドライバのメモリーに記憶する

チェックされていると、直近の正常にマウントされた TrueCrypt ボリュームのパスワードやキーファイルの内容を最大 4 件まで記憶します。これはボリュームをマウントするときに、繰り返し同じパスワードを入力したりキーファイルを選択したりしなくてもよくします。TrueCrypt は絶対にいかなるパスワードもディスクには保存しません。(しかし、「安全のための予防策」も参照してください) パスワードの記憶は設定(*設定 -> 各種設定*)とパスワード入力ウィンドウで有効にも無効にもできます。

マウント成功時にそのボリュームのウィンドウを開く

このオプションがチェックされていると、TrueCrypt ボリュームが正常にマウントされたあと、エクスプローラのウィンドーが自動的に開きそのボリュームのルートディレクトリ （たとえば T:¥） を表示します。

ボリュームがアンマウントされたときウィンドウを閉じる

TrueCrypt ボリュームをアンマウントしたいときに、そのボリュームにある何かのファイルかフォルダーが使用中でロックされているためにアンマウントできないということがあります。エクスプローラウィンドーが TrueCrypt ボリュームにあるディレクトリを表示しているときも同様です。このオプションがチェックされていると、そのようなウィンドーはアンマウント前にすべて自動的にクローズされ、ユーザーが手動でクローズする必要がありません。

***TrueCrypt* の常駐 - 常駐する**

「TrueCrypt の常駐」を参照してください。

***TrueCrypt* の常駐 - マウントされたボリュームがなくなれば常駐終了**

このオプションがチェックされていると、TrueCrypt はマウントされたボリュームがなくなったら、自動的に何もメッセージは出さずに常駐終了します。詳細は「TrueCrypt の常駐」を参照してください。このオプションは TrueCrypt がトラベラーモードで稼働しているときには、不可にはできないことに注意してください。

右に示す時間内に読み書きがなければ自動的にアンマウント

TrueCrypt ボリュームに n 分間書き込みも読み出しもなければ、そのボリュームは自動的にアンマウントされます。

ボリュームに開かれたファイルやフォルダーがあっても強制的にアンマウント

このオプションは、自動アンマウントのみに適用されます。(通常のアンマウントには適用されません) これは、ボリュームのファイルやフォルダー(ディレクトリ)が開いている場合でもメッセージを出さずに強制的に自動アンマウントをします。(システムやアプリケーションで使われているファイルやディレクトリがあった場合です)

TrueCrypt ボリュームのマウント

まだ実行したことがなければ、「*メインプログラムウィンドウ*」の章の「マウント」と「デバイスの自動マウント」を読んでください。

パスワードをドライバのメモリーに記憶する

このオプションは特定のマウント試行にのみ適用されるように、パスワード入力ダイアログで設定することができます。また、「設定」で既定値として設定することもできます。詳細は「設定 -> 各種設定」の節、「*パスワードをドライバのメモリーに記憶する*」を参照してください。

マウントオプション

マウントオプションはボリュームのマウントのされかたに影響します。*マウントオプション*ダイアログはパスワード入力ダイアログの*マウントオプション*ボタンをクリックすることで開きます。正しいパスワードが記憶されていると、*マウント*をクリックするだけでボリュームは自動的にマウントされます。記憶されたパスワードを使ってマウントされているボリュームのマウントオプションを変更したい場合には、*コントロール(Ctrl)*を押しながら*マウント*をクリックするか、*ボリュームメニューのオプションを指定してボリュームをマウント*を選択してください。

マウントオプションの既定値は、メインプログラム設定(*設定* -> *各種設定*)で設定しなおすことができます。

ボリュームを読み取り専用でマウント

チェックが入っていると、マウントされたボリュームにはいっさい書き込みができません。なお、Windows 2000 では NTFS ボリュームを読み取り専用ではマウントできません。

ボリュームをリムーバブルメディアとしてマウント

Windows が勝手に *Recycler* や *System Volume Information* といったフォルダー(これらはごみ箱やシステム復元機能のために使われます)を作ることを防止したいなら、このオプションにチェックを入れてください。

隠しボリュームの保護

「隠しボリュームを破損から守る」を参照してください。

ホットキー

システム全般にわたる TrueCrypt ホットキーを設定するには、「*設定* -> *ホットキー*」をクリックしてください。ホットキーは TrueCrypt が起動中か TrueCrypt が常駐している場合にのみ動作することに留意してください。

キーファイル

キーファイルはパスワードと結合される内容を持つファイルです。(どのようにキーファイルとパスワードを結合させるかについての詳細は「技術解説」の章、「キーファイル」の項を参照) 正しいキーファイルが与えられるまで、キーファイルを使うボリュームはマウントされません。

かならずしもキーファイルを使う必要はありません。しかし、キーファイルを使う便利な理由があります。:

- キーロガーへの対策になる(敵対者がキーロガーでパスワードをキャプチャしても、キーファイルなしではボリュームをマウントできません。
- 総当たり攻撃からの保護を強化します。(特にパスワードが脆弱な場合)
- 複数のユーザーでの共有アクセスを可能にします(すべてのキーファイル所有者は、ボリュームがマウントされる前にキーファイルを提示しなければなりません)

どんな種類のファイル(たとえば .txt, .exe, mp3, .avi) でも TrueCrypt キーファイルとして使うことができます。(しかし、.mp3, .jpg, .zip のような圧縮形式のファイルをおすすめします) TrueCrypt はキーファイル自体に改変を加えることはないことに注意してください。ですから、たとえば巨大な mp3 コレクションの中から 5 個のファイルを TrueCrypt キーファイルとして使うことができるわけです。(そしてファイルを調べても、それらがキーファイルとして使われているということはわかりません)

複数のキーファイルを選択することができます。順番はどうでもかまいません。また、TrueCrypt にランダムな内容のファイルを生成させ、それをキーファイルとして使うこともできます。そうするためには、「*キーファイル -> ランダムキーファイルを生成*」を選んでください。

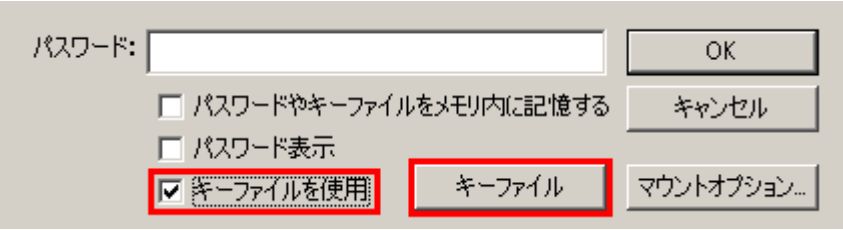
重要: 総当たり攻撃に対抗するため、ひとつのボリュームに対するキーファイルの大きさは少なくとも 30 バイトが必要です。ボリュームが複数のキーファイルを使うなら、そのうちのひとつは 30 バイト以上が必要です。30 バイトという制限は、キーファイルの平均情報量を増大させます。ファイルの先頭 1024 キロバイトの平均情報量が少ないと、(ファイルサイズに関わりなく)キーファイルとしては使われません。平均情報量という意味がよくわからなければ、TrueCrypt にランダムな内容のファイルを生成させ(*キーファイル->ランダムキーファイルの生成*を選択)、それをキーファイルとして使うことをすすめます。

*警告: キーファイルを紛失したり、キーファイルの先頭 **1024** キロバイトが **1** ビットでも破損したりすると、キーファイルを使ったボリュームをマウンとするのは不可能になります！*

警告: パスワードの記憶が有効になっていると、パスワード記憶にはボリュームを正常にマウントしたキーファイルの処理された内容も含まれます。このため、その後にキーファイルがなくなっても再マウントが可能になります。*これを防ぐには「記憶したパスワードの消去」をクリックするかパスワードの記憶を無効にしてください。**(詳細は「設定 -> 各種設定」の「パスワードをドライバのメモリーに記憶する」を参照してください)*

キーファイルダイアログウィンドウ

ボリュームを作成したりマウントしたり、パスワードを変更したりするときに、キーファイルを使いたい(適用したい)ならば、下図のパスワード入力フィールドの「キーファイルを使う」と「キーファイル」ボタンを探してください。



これらの要素はいろいろなダイアログに出現し、常に同じ機能を意味します。「*キーファイルを使う*」オプションをチェックし、「*キーファイル*」をクリックしてください。キーファイルダイアログウィンドウが表示され、使うキーファイルを指定(「*ファイルの追加*」をクリック)するか、キーファイル検索パス(「*フォルダの追加*」をクリック)を指定できます。キーファイルとキーファイル検索パスでは、該当のファイル/フォルダーをキーファイルダイアログウィンドウにドラッグすることでも選択できます。

キーファイル検索パス

ファイルのかわりにキーファイルダイアログウィンドウで(「*フォルダの追加*」をクリックして)フォルダーを追加することで、キーファイル検索パスを指定できます。そのフォルダーで見つかるファイルすべてが[1]キーファイルとして使われます。

重要: キーファイルフォルダーの中のフォルダー(と、その中のファイル)は無視されます。

キーファイル検索パスは、たとえば、持ち歩く USB メモリースティックにキーファイルを保存するときなど、特に有用です。USB メモリースティックのドライブレターをキーファイルの既定の設定に追加することもできます。このためには「キーファイル 」-> 「*デフォルトキーファイル/フォルダの設定*」を選んでください。そして、「*フォルダの追加*」をクリックし USB メモリースティックに割りあてるドライブレターを決め、「*OK*」をクリックしてください。これでボリュームをマウントするたびに(パスワードダイアログの「*キーファイルを使う*」がチェックされていれば)TrueCrypt はフォルダーを調べてそこにあるファイルすべてをキーファイルとして使います。

警告: 既定のキーファイルリストに(ファイルではなく)フォルダーを追加すると、パス(フォルダー)だけが記憶されファイル名は記憶されません！ということは、そのフォルダーに新規にファイルを作成したり追加したりすると、そのフォルダーに依存しているキーファイルを使うボリュームはすべてマウント不可になります。(新しく追加されたファイルをフォルダーから除去すれば復旧します)

[1]ボリュームをマウントする、パスワードを変更する、その他ボリュームヘッダーを再暗号化するときに見つかったすべて

空のパスワードとキーファイル

キーファイルを使うときに、パスワードは空かもしれません。そうすると、キーファイルのみがボリュームをマウントする唯一のアイテムになります。(これは推薦されません) 既定のキーファイルが設定されボリュームをマウントするときに使える状態なら、パスワード入力画面の前にTrueCryptはまず空のパスワードと既定のキーファイルを使ってマウントしようとします。もしこの方法でマウントするボリュームにマウントオプション(読取専用でマウントとか隠しボリュームを保護するとか)を設定する必要があるなら、コントロール(Ctrl)キーを押しながら「マウント」をクリック(または「*ボリューム*」メニューの「*ボリュームをオプションを指定しながらマウント*」を選択 してください。「*マウントオプション*」ダイアログが開きます。

キーファイル -> ボリュームへのキーファイルの追加/削除

この機能はいくつかのキーファイル(パスワードなし、またはあり)またはキーファイルがなしで生成されたヘッダー暗号化キーでボリュームヘッダーを再暗号化します。パスワードのみでマウント可能なボリュームを、(パスワードに加えて)キーファイルが必要なボリュームに変換します。ボリュームヘッダーはそのボリュームを暗号化しているマスター暗号化キーを含むことに注意してください。そのボリュームに保存されたデータはこの機能を使ってもまったく失われたりはしません。

また、この機能はボリュームのキーファイルを変更/設定することにも使われます。(いくつか、あるいは全部のキーファイルを除外し新しいものを適用する)

補足: この機能は内部的にはパスワード変更機能と同じです。

注意: TrueCryptがボリュームヘッダーを再暗号化する場合、敵対者が微視的残留磁気[17]から上書きされたヘッダーを復元できないようにするため、最初に元のボリュームヘッダーをランダムデータで35回の上書きをします。(「安全のための予防策」も参照)

キーファイル -> ボリュームから全てのキーファイルを除去

この機能は、キーファイルではなくパスワードから導出されたヘッダー暗号化キーでボリュームヘッダーを再暗号化します。(キーファイルをまったく使わずに、パスワードのみでマウントされるようになります) ボリュームヘッダーはそのボリュームを暗号化しているマスター暗号化キーを含むことに注意してください。そのボリュームに保存されたデータはこの機能を使ってもまったく失われたりはしません。

補足: この機能は内部的にはパスワード変更機能と同じです。

注意: TrueCryptがボリュームヘッダーを再暗号化する場合、敵対者が微視的残留磁気[17]から上書きされたヘッダーを復元できないようにするため、最初に元のボリュームヘッダーをランダムデータで35回の上書きをします。(「安全のための予防策」も参照)

キーファイル　-> ランダムキーファイルの生成

この機能を使って、キーファイルとして使えるランダムな内容のファイル(推奨)を生成できます。この機能は TrueCrypt の乱数発生機構を使います。結果として生成されるファイルのサイズは常に 64 バイト(512 ビット)であり、これは TrueCrypt のパスワードの最大長でもあります。

キーファイル -> デフォルトキーファイル/フォルダの設定

既定のキーファイルまたはいっしょにキーファイル検索パスを設定するには、この機能を使ってください。これは、たとえば、持ち歩く USB メモリースティックにキーファイルを保存するときなど、特に有用です。ドライブレターをキーファイルの既定の設定に追加することもできます。このためには「*キーファイル -> 既定キーファイルパス*」を選んでください。そして、「*パスの追加*」をクリックし USB メモリースティックに割りあてるドライブレターを決め、「*OK*」をクリックしてください。これでボリュームをマウントするたびに(パスワードダイアログの「キーファイルを使う」がチェックされていれば)TrueCrypt はパスを調べてそこにあるファイルすべてをキーファイルとして使います。
警告: 既定のキーファイルリストに(ファイルではなく)フォルダーを追加すると、パスだけが記憶されファイル名は記憶されません！ということは、そのフォルダーに新規にファイルを作成したり追加したりすると、そのフォルダーに依存しているキーファイルを使うボリュームはすべてマウント不可になります。(新しく追加されたファイルをフォルダーから除去すれば復旧します)

重要: デフォルトキーファイルやデフォルトキーファイルフォルダを設定すると、ファイル名やパスは暗号化されずに Default Keyfiles.xml に保存されることに注意してください。詳細は「TrueCrypt システムファイルとアプリケーションデータ」を参照してください。

トラベラーモード

TrueCrypt はいわゆるトラベラー(旅行者)モードで動作させることができます。これは、TrueCrypt を稼働する OS に対してインストールしなくていいということです。しかし、次の 2 項目は憶えておいてください。

1) TrueCrypt をトラベラーモードで動かすには管理者権限が必要
2) トラベラーモードで起動したとしても、レジストリファイルを検査すれば、Windows で TrueCrypt を使った(そてけ、TrueCrypt ボリュームをマウントした)ということがわかってしまうかもしれません。

この問題に対処する必要があるなら、BartPE を使うことをすすめます。また「よくある質問(FAQ)と答え」の「*Windows で痕跡を残さずに TrueCrypt を使うことはできますか?*」を参照してください。

TrueCrypt トラベラーモードを使うには、二つの方法があります。

1) バイナリ配布パッケージを展開し、(インストールせずに)直接 TrueCrypt.exe を走らせる。
2) 「トラベラーディスク作成」を利用して、特別なトラベラーディスクを作りそこから TrueCrypt を起動する。

2 番目のほうがいくつか有利な点があり、この章の以下の節でそれらについて説明します。

注意: トラベラーモードのときには、ドライバは必要がなくなれば(つまり、主アプリケーションとボリューム作成ウィザードが閉じられ、マウントされた TrueCrypt ボリュームがない状態になったとき)、メモリーから除去されます。 しかし、TrueCrypt がトラベラーモードで動いているときに、TrueCrypt ボリュームが強制的にアンマウントされると「終了」しても TrueCrypt ドライバーは除去されません。(システムを停止するかリスタートする場合のみ、除去されます) これは Windows のバグによって引き起こされるいろいろな問題(たとえば、アンマウントされたボリュームを使っているアプリケーションがあると TrueCrypt を再起動できない)を防止します。

ツール -> トラベラーディスクのセットアップ

特別なトラベラーディスクを作りそこから TrueCrypt を起動するために、この機能を利用できます。TrueCrypt *トラベラーディスク*は TrueCrypt ボリュームではなく暗号化されてもいないことに注意してください。トラベラーディスクは TrueCrypt 実行ファイルとオプションとして‘autorun.inf’を含みます。(「*自動実行設定*」を参照) 「*ツール* 」-> 「*トラベラーディスクのセットアップ*」を選択すると、「*トラベラーディスクセットアップ*」ダイアログが表示されます。そこで設定できるいくつかの設定項目については、これから説明します。

TrueCrypt ボリューム作成ウィザードを含める

トラベラーディスクから起動したTrueCryptを使って新しいTrueCryptボリュームを作りたいなら、ここにチェックを入れてください。このオプションをチェックしなければ、トラベラーディスクの容量の節約になります。

自動実行ファイル(autorun.inf)の設定

この項目で、トラベラーディスクが挿入されると自動的にTrueCryptを起動したり、自動的に特定のTrueCryptボリュームをマウントするように設定できます。これは、トラベラーディスクに*autorun.inf*という特別なスクリプトファイルを作ることで可能になります。このファイルはトラベラーディスクが挿入されるつどOSによって自動実行されます。ただし、これはCD/DVDのようなリムーバブルメディアのみで、それらが読取可能な場合のみに動作します。(USBメモリスティックでこの機能を使うには、Windows XP SP2かWindows Vistaが必要です)
また、この機能を有効にするためは、*autorun.inf*ファイルは暗号化されていないディスクのルートディレクトリに置かれなくてはならないことに注意してください。(たとえば、G:¥, X:¥, Y:¥などです)

TrueCrypt を管理者権限なしで使う

Windows では管理者権限がないユーザーでも TrueCrypt を使うことができます。しかし、管理者がシステムに TrueCrypt をインストールしたあと(あるいは、管理者がユーザーに管理者権限を与えたあと)に限ります。その理由は、TrueCrypt 即時自動暗号化/復号のデバイスドライバを必要とし、管理者権限がないと Windows にデバイスドライバをインストールできないからです。

システム管理者が TrueCrypt をインストールしたあとは、管理者権限がないユーザーでも TrueCrypt を起動しどんな種類の TrueCrypt ボリュームでもマウント/アンマウントすることができ、データをそこに保存/読み出しができ、ファイル型 TrueCrypt ボリュームの作成もできます。しかし、管理者権限がないユーザーはパーティションを暗号化/フォーマットしたり NTFS ボリュームをつくることはできませんし、TrueCrypt のインストール/アンインストールもできません。また、デバイス型ボリュームのパスワード/キーファイル変更や TrueCrypt をトラベラーモードで動かすこともできません。

TrueCrypt の常駐

メイン TrueCrypt ウィンドウが閉じても、TrueCrypt は常駐し以下の機能を実行します。

1) ホットキー
2) 自動アンマウント(ログオフ時、不用意なデバイスの取り外し時、タイムアウト時など)
3) 通知メッセージ (隠しボリュームの破損が防止されたとき)
4) タスクトレイアイコン

警告: TrueCrypt が常駐していず TrueCrypt も動いていなければ、上記の機能は無効になります。

TrueCrypt の常駐は実際には TrueCrypt.exe そのものであり、TrueCrypt メインウィンドウを閉じてもバックグラウンドで動きつづけているということです。それが起動中であるかどうかは、タスクトレイで判別できます。TrueCrypt アイコンがあれば、TrueCrypt は常駐しているということです。アイコンをクリックして、TrueCrypt メインウィンドウを開くことができます。アイコンを右クリックすれば、いろいろな TrueCrypt 関連機能のポップアップメニューが開きます。

常駐はタスクトレイの TrueCrypt アイコンを右クリックして、「終了」を選択することで停止できます。TrueCrypt の常駐を完全に永続的に止めたいなら、「*設定* -> 各種設定」を選び、「*各種設定*」ダイアログき「*TrueCrypt の常駐*」の「*常駐する*」のチェックを外してください。

言語パック

言語パックは TrueCrypt ユーザーインターフェースのテキストの第三者の翻訳を含みます。いくつかの言語パックは TrueCrypt ユーザーズガイドの翻訳も含みます。言語パックは、現在のところ TrueCrypt の Windows 版のみでサポートされていることに留意してください。

インストール

言語パックは以下の手順でインストールしてください。

1. 言語パックをダウンロードする: http://www.truecrypt.org/localizations.php
2. TrueCrypt を(稼働中であれば)終了する。
3. 言語パックを TrueCrypt をインストールしたフォルダー(TrueCrypt.exe が存在するフォルダー、たとえば C:¥Program Files¥TrueCrypt とか C:¥Program Files (X86)¥TrueCrypt など)に展開する。
4. TrueCrypt を起動する。
5. 言語パックは自動的に検出され、既定の言語パックとして設定されます。(「*設定* -> *言語*」をクリックしていつでも言語を選択できます)

英語にもどすには、「*設定* -> *言語*」を選んで、*English* を選び、「*OK*」をクリックしてください。

暗号化アルゴリズム

TrueCrypt ボリュームは以下のアルゴリズムで暗号化することができます。

アルゴリズム	設計者	キーサイズ (Bits)	ブロックサイズ (Bits)	動作モード
AES	J. Daemen, V. Rijmen	256	128	LRW
Serpent	R. Anderson, E. Biham, L. Knudsen	256	128	LRW
Twofish	B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, N. Ferguson	256	128	LRW
AES-Twofish		256; 256	128	LRW
AES-Twofish-Serpent		256; 256; 256	128	LRW
Serpent-AES		256; 256	128	LRW
Serpent-Twofish-AES		256; 256; 256	128	LRW
Twofish-Serpent		256; 256	128	LRW

LRW モードについての詳細は「動作モード」を参照してください。

AES

Advanced Encryption Standard は FIPS（連邦情報処理規格）で承認された暗号アルゴリズム (Rijndael, designed by Joan Daemen and Vincent Rijmen, published in 1998)であり、アメリカ政府各部局、各機関で重要(機密扱いでない)情報を暗号化して保護するために[3]使われています。TrueCrypt は AES を LRW モード(「動作モード」を参照)で 14 ラウンド、256-bit キー(AES-256, published in 2001)として使っています。

2003 年 6 月に、NSA (US National Security Agency)が AES を分析、評価し、U.S. CNSS (Committee on National Security Systems)は[2]の中で AES-256(および AES-192)の強度は最高機密にいたるまでの機密扱いの情報を保護するのに充分であると発表しました。これは、Advanced Encryption Standard (AES)を使うか組み込むことで国家安全システムと国家安全情報に関連する Information Assurance の要求を満たすと考えるアメリカ政府各部局、各機関で採用可能ということです。[2]

Serpent

Ross Anderson, Eli Biham, および Lars Knudsen によって設計され、1998 年に発表されました。256-bit キー、128-bit ブロックで LRW モード(「動作モード」を参照)です。Serpent は AES の最終候補の一つです。これは Rijndael [4]より高度な安全性があるように見えるにもかかわらず、AES の推薦には選ばれませんでした。具体的には、Rijndael でも安全確保に充分であるのに対し、Serpent は高度な安全確保ができるように見えます。また、Rijndael はその数学的構造が将来攻撃対象となるかもしれないという、いくつかの批判を受けています。[4]

[5]において、Twofishチームは各AES最終候補の安全係数の表を示しています。安全係数は、完全に暗号化するラウンド数をすでに破られた最大のラウンド数で割ったもので定義されます。だから、破られた暗号は最低の係数1ということになります。SerpentはAES最終候補の中で、(すべてのサポートされたキーサイズで)もっとも高い安全係数3.56を持ちます。Rijndael-256の安全係数は1.56であり、Rijndael-256は安全係数1.56です。
これらの事実にもかかわらず、Rijndaelは安全性、速度、効率、実装のしやすさ[4]、柔軟性などのバランスのよさで、AESの中で適切な選択であると考えられています。最後のAES会議で、Rijndaelは86票、Serpentは59票、Twofishは31票、RC6は23票、MARSは13票でした。[18, 19][1]

Twofish

Bruce Schneier, David Wagner, John Kelsey, Niels Ferguson, Doug Whiting, Chris Hall によって設計され、1998 年に発表されました。256-bit キー、128-bit ブロックで LRW モード(「動作モード」を参照)で動きます。Twofish は AES の最終候補の一つです。この暗号は、キーから独立した S-ボックスを使います。Twofish は、2^{128}(2 の 128 乗)の異なった暗号システムの集まりに見え、256-bit キーから導出される 128bits がその集まりの中からの暗号システムの選択をコントロールします。[4] [13]の中で、Twofish チームは、キーから独立した S ボックスが未知の攻撃に対する安全性を高めると主張しています。[4]

AES-Twofish

2 つの暗号が LRW モード(「動作モード」を参照)でカスケード(多段処理)[15, 16] されます。それぞれの 128-bit ブロックは、まず Twofish (256-bit キー)で暗号化され、つぎに AES (256-bit キー)で暗号化されます。カスケードの暗号のそれぞれは、それ自身のキーを使います。それぞれのキーは相互に独立しています。(補足: ヘッダーキーは一つのパスワードから導出されるものの、きちんと独立しています。「ヘッダーキーの*導出*、*ソルト*、*および反復回数*」を参照) カスケードのそれぞれの暗号については、上記の個別解説を参照してください。

AES-Twofish-Serpent

モードで 3 つの暗号[15,16]が LRW モード(「動作モード」を参照)でカスケード(多段処理)されます。128-bit ブロックは、まず Serpent (256-bit キー, 128-bit ブロック)で暗号化され、次に Twofish (256-bit キー)、最後に AES (256-bit キー, 128-bit ブロック)で暗号化されます。カスケードの暗号

[1] これは肯定的な票です。肯定的な票から否定的な票を引くと、次の結果となります。Rijndael: 76 票, Serpent: 52 票, Twofish: 10 票, RC6: -14 票, MARS: -70 票 [19]

のそれぞれは、それ自身のキーを使います。それぞれのキーは相互に独立しています。(補足: ヘッダーキーは一つのパスワードから導出されるものの、きちんと独立しています。「ヘッダーキーの導出、ソルト、および反復回数」を参照) カスケードのそれぞれの暗号については、上記の個別解説を参照してください。

Serpent-AES

2 つの暗号[15,16]が LRW モード(「動作モード」を参照)でカスケード(多段処理)されます。それぞれの 128-bit ブロックは、まず AES (256-bit key)で暗号化され、つぎに Serpent (256-bit key)で暗号化されます。カスケードの暗号のそれぞれは、それ自身のキーを使います。それぞれのキーは相互に独立しています。(補足: ヘッダーキーは一つのパスワードから導出されるものの、きちんと独立しています。「ヘッダーキーの導出、ソルト、および反復回数」を参照) カスケードのそれぞれの暗号については、上記の個別解説を参照してください。

Serpent-Twofish-AES

3 つの暗号[15,16]が LRW モード(「動作モード」を参照)でカスケード(多段処理)されます。128-bit ブロックは、まず AES (256-bit key)で暗号化され、次に Twofish (256-bit キー)、最後に Serpent (256-bit key)で暗号化されます。カスケードの暗号のそれぞれは、それ自身のキーを使います。それぞれのキーは相互に独立しています。(補足: ヘッダーキーは一つのパスワードから導出されるものの、きちんと独立しています。「ヘッダーキーの導出、ソルト、および反復回数」を参照) カスケードのそれぞれの暗号については、上記の個別解説を参照してください。

Twofish-Serpent

2 つの暗号[15,16]が LRW モード(「動作モード」を参照)でカスケード(多段処理)されます。それぞれの 128-bit ブロックは、まず Serpent (256-bit key)で暗号化され、つぎに Twofish (256-bit key)で暗号化されます。カスケードの暗号のそれぞれは、それ自身のキーを使います。それぞれのキーは相互に独立しています。(補足: ヘッダーキーは一つのパスワードから導出されるものの、きちんと独立しています。「ヘッダーキーの導出、ソルト、および反復回数」を参照) カスケードのそれぞれの暗号については、上記の個別解説を参照してください。

ハッシュアルゴリズム

ボリューム作成ウィザードやパスワード変更ダイアログウィンドウ、キーファイル生成ダイアログウィンドウなどで、ハッシュアルゴリズムを選択できます。ユーザーが選択したハッシュアルゴリズムはTrueCrypt乱数発生機構で疑似乱数混合関数で使われ、ヘッダーキー導出関数(PKCS #5 v2.0で規定されているとおりHMACハッシュアルゴリズムに依存します)で疑似乱数関数として使われます。新しいボリュームを作成するとき、乱数発生機構はマスター暗号化キー、第二キー(LRWモード)、ソルトを生成します。詳細は「乱数発生機構」と「ヘッダーキーの導出、ソルト、および反復回数」の項を参照)

Whirlpool

Whirlpoolハッシュアルゴリズムは Vincent Rijmen (AES encryption algorithmの共作者)と Paulo S. L. M. Barretoによる設計です。このアルゴリズムの出力サイズは512bitsです。 Whirlpool-0と呼ばれるようになったWhirlpoolの最初のバージョンは2000年11月に発表されました。Whirlpool-Tと呼ばれるようになった第二版はNESSIE (*New European Schemes for Signatures, Integrity and Encryption*)の暗号資産(AESコンテストに似て、EUによって組織されたプロジェクト)に選択されました。TrueCryptは、International Organization for Standardization (ISO) やISO/IEC 10118-3:2004 international standard [21] のIECに採択されたWhirlpoolの第三版(最終版)を採用しています。

SHA-1

SHA-1はNSAが設計し1995年に発表されたハッシュアルゴリズムです。このアルゴリズムの出力サイズは160bitsです。2005年に、総当たり攻撃よりは平均的には楽に(2^{80}ステップのかわりに2^{63}ステップ)SHA-1の問題点を探す方法が発表されました。しかし、TrueCryptはSHA-1を疑似乱数関数としてはほとんど使いませんし、近い将来にSHA-1の問題点が発見されて、TrueCryptボリュームの安全性に影響がでるとは思えません。このことは[25]で発表された証明からでも言えることです。しかし、より確実性を求めて WhirlpoolやRIPEMD-160を使うこともできます。

RIPEMD-160

RIPEMD-160は1996年に発表され、Hans Dobbertin, Antoon Bosselaers, と Bart Preneelによってオープンな学術的コミュニティで設計されました。そしてNSAが設計したSHA-1の価値ある代替えでもあります。 RIPEMD-160の出力サイズは160bitsです。 RIPEMD-160は、EUのRIPE(*RACE Integrity Primitives Evaluation*)プロジェクト(1988-1992)で開発され2004年に問題が発見されたRIPEMDハッシュアルゴリズムの強化版です。RIPEMD-160はこれまでのところ問題は発見されておらず、平均的には総当たり攻撃より楽な方法もありません。(ハッシュ関数の問題点をどのように発見するかが、TrueCryptにどう影響するかは、SHA-1を参照) RIPEMD-160は国際標準化機構(ISO)と IEC in the ISO/IEC 10118-3:2004の国際規格[21]に適合しています。

動作対象 OS

TrueCrypt は次の OS で稼働します。

- Windows Vista
- Windows Vista x64 (64-bit) Edition
- Windows XP
- Windows XP x64 (64-bit) Edition
- Windows Server 2003
- Windows 2000
- Linux (kernel 2.6.5 or later)

注意: 次の OS はサポートされていません: Windows 2003/XP IA-64, Windows NT/95/98/ME.

コマンドラインの使い方

この節は Windows 版 TrueCrypt を対象とします。(Linux 特有の事項は TrueCrypt バイナリおよびソースコードパッケージに含まれる truecrypt man に記載しています。これはつぎのところで入手できます: http://www.truecrypt.org/download.php

/help or */?*
コマンドラインヘルプを表示します。

/volume or */v*
マウントする TrueCrypt ボリュームのファイルとパスの名前(アンマウント時には使わないこと)。ハードディスクのパーティションをマウントする場合の例は /v ¥Device¥Harddisk1¥Partition3(パーティションのパスを決めるには、TrueCrypt を起動して「*デバイスの選択*」をクリックしてください。デバイスのパスは大文字小文字を区別します。

/letter or */l*
ボリュームをマウントするドライブレター。/l が省略され /a が指定されている場合には最初の空きドライブレターを使います。

/explore or */e*
ボリュームがマウントされると、そのボリュームのウィンドウを開きます。

/beep or */b*
ボリュームが正常にマウントまたはアンマウントされるとビープを鳴らします。

/auto or */a*
パラメータが指定されていなければ、ボリュームを自動マウントします。devices がパラメータとして指定(/a devices)されていれば、すべての使用可能なデバイス/パーティション型 TrueCrypt ボリュームを自動マウントします。 パラメータとして favorites が指定されていれば、お気に入りボリュームを自動マウントします。/quit と /volume が指定されると /auto も暗黙のうちに指定されたことになることに注意してください。

/dismount or */d*
ドライブレターで指定されたボリュームをアンマウントします。(例: /d x) ボリュームが指定されていないと、現在マウントされているすべての TrueCrypt ボリュームをアンマウントします。

/force or */f*
強制的に(そのボリュームのファイルがシステムかアプリケーションに使われていても)アンマウントを実行し、マウントを共有モード(排他制御なし)にします。

/keyfile or */k*

キーファイルかキーファイル検索パスを指定します。複数のキーファイルの指定は　/k c:¥keyfile1.dat /k d:¥KeyfileFolder /k c:¥keyfile2 のようにします。

/cache or */c*

y またはパラメータなしの場合は、パスワード記憶を有効にします。n の場合(/c n)はパスワード記憶を無効にします。パスワード記憶を無効にしても記憶したものを消去するわけではありません。(消去するには /w を使ってください)

/history or /h

y またはパラメータなし : マウントしたボリュームの履歴を保存; n: マウントしたボリュームの履歴を保存しない。(例 /h n)

/wipecache or */w*

ドライバに記憶したパスワードをすべて消去

/password or */p*

ボリュームのパスワード。パスワードに空白を含む場合には引用符で囲むこと (例 /p ”My Password”). 空パスワードを表すには /p ”” としてください。
警告: この方法でボリュームパスワードを入力することは、暗号化されていないコマンドプロンプトの履歴が暗号化されていないディスクに保存される場合に、安全に問題があるかもしれません。代わりに /q を使うことを検討してください。

/quit or */q*

要求された動作を実行し、終了します。(TrueCrypt メインウィンドウは表示されません) preferences が指示されていれば(/q preferences) プログラム設定が読込/保存されます。
/q background は TrueCrypt 常駐 (トレイアイコン) を開始します。
/q はコンテナがローカルユーザー名前空間でしかアクセスできない場合 (ネットワークボリューム) には効果がなく、TrueCrypt はボリュームがアンマウントされた後のみ終了します。

/silent or */s*

/q が指定されていれば、ユーザーへのメッセージ(プロンプト、エラーメッセージ、警告 など)を表示しません。

/mountoption or */m*

ro または readonly: 読取専用でマウント

rm または removable: リムーバブルメディアとしてマウント

ts または timestamp: ボリューム/キーファイルのタイムスタンプを変更

persistent: GUI でボリュームを表示せず、自動アンマウントを抑止。「すべてをアンマウント」でもアンマウントをしない。コマンドラインで個々にアンマウントすることはできる。このオプションは /q と同時に指定されたときのみ機能することに注意

system: persistent と同じ。追加機能として、Windows のページングファイルをボリュームに格納することができる。このオプションは /q と同時に指定されたときのみ機能することに注意

例: /m ro　複数のマウントオプションを指定する場合は、/m rm　/m ts を使う

文法

truecrypt [/a [devices|favorite]] [/b] [/c [y|n]] [/d [*drive letter*]] [/e] [/f] [/h [y|n]] [/k *Keyfile or search path*] [/l *drive letter*] [/m {persistent|rm|ro|system|ts}] [/p *Password*] [/q [background|preferences]] [/s] [/v *Volume*] [/w]

オプションを記述する順番は重要ではありません。

使用例

d:¥myvolume という名前のボリュームを最初の空きドライブレターに割り当ててマウント、パスワードプロンプトを表示(メインプログラムウィンドウは表示しない)

```
truecrypt /q /v d:¥myvolume
```

ドライブ X としてマウントされているボリュームをアンマウントする。

```
truecrypt /q /dx
```

myvolume.tc という名前のボリュームを　MyPassword というパスワードで、ドライブ X にマウント
TrueCrypt はウィンドウを開き、ビープを鳴らし、自動でマウントします。

```
truecrypt /v myvolume.tc /lx /a /p MyPassword /e /b
```

安全のための予防策

この章ではTrueCryptボリュームに保存された機密データの安全性に影響するいくつかの項目について述べます。すべての危険性について網羅することはできないことを、ご了承ください。残念ながら非常に多くの種類の危険があり、すべてを解説しようとするとあまりに膨大になってしまうためです。

ページングファイル

スワップファイルとも呼ばれます。Windowsはこの(通常ハードディスクに置かれる)ファイルを、メモリに入りきらないプログラムやデータファイルを保持するために使います。ということは、メモリ上だけにあると信じている機密データが実際には知らないうちにWindowsによって暗号化もされずにディスクに書かれているということです。

TrueCryptはパスワード、暗号化キー、および他の機密データがあるメモリー領域を、それらのデータがページングファイルへもれないように、つねにロックしようとします。しかし、Windowsではいろいろな(文書化さたものも、されていないものもある)理由で、ロックが拒否されることがあります。さらに、TrueCryptは、RAM上に開かれた機密ファイルが*暗号化されない状態*でスワップに保存されることを防ぐことはできません。(TrueCryptボリュームのファイルをテキストエディターとかなにかで開くと、そのファイルの内容は暗号化されていない状態でRAMに置かれます)
ですから、Windows XP/Vistaユーザーには、スワップファイル機能を無効にすること、少なくとも機密データを扱ったりTrueCryptをマウントするセッションの間だけでも無効にすることを強くおすすめします。これをするにはデスクトップかスタートメニューのコンピュータまたはマイコンピュータ・アイコンの上で右クリックし、*プロパティ->(*Windows Vistaでは*->高度なシステム設定->)詳細設定->パフォーマンス->設定->詳細設定->仮想メモリ>変更->ページングファイルなし->設定->OK*としてください。
知る限りでは、Windows 2000ではこの方法では完全に無効にはできません。Windows 2000ユーザーには、コンピュータをシャットダウンするつどにページングファイルをクリアするようセキュリティの設定を変更することをおすすめします。(詳細はWindowsのマニュアルまたはwww.microsoft.comを参照してください)

ハイバネーションモード

コンピュータがハイバネーションモード(省電力モード)に入るとき、システムメモリの内容はハードディスクに書き出されます。TrueCryptは記憶したパスワード、暗号化キーやRAM上に開かれた機密ファイルが*暗号化されない状態*でハイバネーション・ファイルに保存されることを防ぐことはできません。TrueCryptボリュームのファイルをテキストエディターとかなにかで開くと、そのファイルの内容は暗号化されていない状態でRAMに置かれます。(そして、電源を切るまでそのまま暗号化されない状態でRAMに残るかもしれません) また、TrueCryptボリュームがマウントされていると、そのマスターキーは暗号化されていない状態でRAMに保持されます。ですから、少なくとも機密データを扱ったりTrueCryptをマウントするセッションの間だけでもハイバネーション機能を無効にするか、ハイバネーションの起動を抑止することを強くおすすめします。

メモリダンプファイル

Windows を含むほとんどの OS でデバッグ情報の取得やエラー発生時(システムクラッシュ、ブルースクリーン、バグチェック)のシステムメモリの内容の取得(メモリーダンプ)が可能です。このメモリーダンプファイルには機密データを含んでいるかもしれません。TrueCrypt は記憶したパスワード、暗号化キーや RAM に展開された機密ファイルの内容が暗号化されていない状態でメモリーダンプファイルに書き出されることを防ぐことはできません。TrueCrypt ボリュームのファイルをテキストエディターとかなにかで開くと、そのファイルの内容は暗号化されていない状態で RAM に置かれます。(そして、電源を切るまでそのまま暗号化されない状態で RAM に残るかもしれません) また、TrueCrypt ボリュームがマウントされていると、そのマスターキーは暗号化されていない状態で RAM に保持されます。ですから、少なくとも機密データを扱ったり TrueCrypt をマウントするセッションの間だけでもコンピュータのメモリーダンプファイル生成機能を無効にすることを強くおすすめします。WindowsXP/Vista の場合には、デスクトップかスタートメニューのマイコンピュータ・アイコンの上で右クリックし、*プロパティ*->Windows Vista では->*高度なシステム設定*->*詳細設定*->*起動と回復*->*設定*->*デバッグ情報の書き込みの項目*->*(なし)を選択*>*OK* としてください。

マルチユーザー環境

マウントされた TrueCrypt ボリュームの内容はすべてのログオンしたユーザーには見え、アクセス可能になるということを忘れないでください。(NTFS ではファイルの許可情報の設定で、このようなことを防ぐことは可能です) また、WindowsXP/Vista(簡易ユーザー切替)のユーザー切替やログオフは正常にマウントされた TrueCrypt ボリュームをアンマウントしないことに注意してください。 (システムを再起動する場合には、すべてのマウントされた TrueCrypt ボリュームはアンマウントされます)

RAM にある暗号化されていないデータ

ほとんどのプログラムは TrueCrypt ボリュームから読み込んだファイルの暗号化されていないデータをあるメモリー領域(バッファ)に置き、クリアしないことに気をつけてください。これは、そのようなプログラムを終了しても、そのプログラムが使った暗号化されていないデータは電源を切るまでメモリーに残っているかもしれないということを意味します。また、テキストエディターなどで TrueCrypt ボリュームのファイルを開いて、そのボリュームを強制アンマウントしたとしても、テキストエディターのバッファには暗号化されない状態でファイルは残ります。このことは自動アンマウントについても同じです。

パスワードとキーファイルの変更

ボリュームヘッダー(パスワードやキーファイルから導出されるヘッダーキーで暗号化されている)はボリュームを暗号化しているマスターキーを含んでいることに留意してください。もし、敵対者がパスワードやキーファイルを変更する前のボリュームのコピーを取得可能なら、そのコピーあるいは断片(旧ヘッダー)とともにパスワードやキーファイルの変更前にボリュームをマウントす

るのに必要だったパスワードを推測(たとえばキーロガーで取得するなど)したりキーファイルを推測したりして、TrueCrypt ボリュームをマウントすることができるかもしれません。
パスワードやキーファイルを変更するときに敵対者がパスワードやキーファイルを知っているかどうか、ボリュームのコピーを持っているかどうかに不安があるなら、新しい(異なるマスターキーを持つ)TrueCrypt ボリュームを作成し旧ボリュームから新ボリュームへファイルを移動させることをおすすめします。

また、注意すべきは敵対者がパスワードを知っていたりキーファイルを持っていてボリュームへアクセスできるとすると、敵対者はマスターキーを再取得して保管しておくことができるかもしれないということです。そうだとすると、敵対者はパスワードやキーファイルを変更してもボリュームを復号できることになります。(パスワードやキーファイルを変更しても、マスターキーは変更されないからです) このような場合には、新しい TrueCrypt ボリュームを作成し旧ボリュームから新ボリュームへファイルを移動させてください。

第二キー

TrueCrypt ボリュームが暗号化されるのに使われるマスターキーの一部である LRW 加工キーが同じ TrueCrypt ボリュームに保存されるので、攻撃者は(ボリュームがマウントされていなくても)加工キーを復元できるかもしれません。たとえば、TrueCrypt ボリュームをマウントしそのボリュームの加工キーを保持している RAM をボリューム上のファイルに保存する場合にこのようなことが発生するかもしれません。(メモリーダンプファイル、ページングファイル、ハイバネーションモードも参照してください)

Windows レジストリ

TrueCrypt の「みせかけの拒否」は、ファイルやパーティションが TrueCrypt のボリュームであるかどうか、隠しボリュームが存在するかどうかを確認することが不可能であるということに依存しています。Windows は TrueCrypt がかくじつ安全に消去できないさまざまなデータをレジストリに保存します。レジストリファイルを調査すると、攻撃者は TrueCrypt がそのシステムで実行されたことがあるか、TrueCrypt ボリュームがマウントされたかどうか(そのボリュームの位置/ファイル名/サイズ/タイプ[1]まではわかりません)、どのドライブ文字が TrueCrypt ボリュームに使われたか(そのボリュームの位置/ファイル名/サイズ/タイプまではわかりません)などを知ることができます。

データの破損

ハードウェアやソフトウェアのエラーや誤動作で、TrueCrypt ボリュームのファイルが破損することもありえます。ですから、重要ファイルは定期的にバックアップをとることをすすめます。(もちろん、TrueCrypt ボリュームに記録された暗号化データにかぎらず、すべての重要なデータについて言えることです)

[1]タイプとは、隠しボリュームか通常ボリュームかという意味です。

TrueCrypt ボリュームにあるすべてのファイルをバックアップするだけの空き領域がない場合、少なくともボリュームヘッダーだけでもバックアップをとっておくことを強くおすすめします。ここにはマスターキーが記録されています。(バックアップしたファイルのサイズは 1024 バイトになるはずです) ボリュームヘッダーが破損すると、ほとんどの場合はボリュームはマウントできなくなります。ボリュームヘッダーをバックアップするには、「*デバイスの選択*」か「*ファイルの選択*」をクリックし、ボリュームを選択してください。それから「*ツール -> ボリュームヘッダーのバックアップ*」をクリックしてください。ヘッダーを復旧するには、同じ手順で最後に「*ボリュームヘッダーのリストア*」を選択してください。

重要: 数人のユーザーから、TrueCrypt ボリュームのデータが破損すると報告がありました。その後、これらのユーザーは原因が TrueCrypt ではなくハードウェア(チップセット、USB ハードドライブ ケーブル、USB PCI カード 他)であることを発見しました。ですから、TrueCrypt ボリュームを作ろうとするデバイスに書かれたデータが破損しないか確認することをおすすめします。たとえば、大量のファイル(少なくとも合計で数 GB)をコピーし、Windows 標準のコマンドラインツール fc をを使ってオリジナルとその内容を比較するというようなことです。

ウェアレベリング

いくつかの記憶装置(たとえば、いくつかの USB フラッシュドライブ)やいくつかのファイルシステムでは装置や媒体の寿命を延ばすため、ウェアレベリングという機能を持ちます。この機能は、アプリケーションが同じ論理セクターに繰り返しデータを書き込む場合に、メディア全体に分散して書き込む(論理セクターが違う物理セクターに再配置される)というものです。ですから、あるセクターの複数の版が攻撃者に入手可能になるかもしれません。これはセキュリティに問題を生じます。たとえば、ボリュームパスワードやキーファイルを変更した場合に通常ではヘッダーを再暗号化したもので上書きします。しかし、ボリュームがウェアレベリング機能を持つデバイスにあると、TrueCrypt は古いヘッダーがほんとうに上書きされるとは保証できなくなります。もし敵対者が本来なら上書きされてしまうはずの古いヘッダーをそのデバイス上で見つけたとすると、古い(ヘッダーが再暗号化される前にマウントするのに必要だった)パスワードやキーファイルを使ってボリュームをマウントすることができてしまいます。安全上の理由から、TrueCrypt ボリュームをウェアレベリング機能を持つデバイス(またはファイルシステム)に置かないことをすすめます。デバイスにウェアレベリング機能があるかどうかは、そのデバイスの説明書を参照するかメーカーに問い合わせてください。

デフラグ

ファイル型 TrueCrypt コンテナを格納したファイルシステムをデフラグする場合、TrueCrypt コンテナ(あるいは、その断片)のコピーがホストボリューム(断片化していたファイルシステム)の空き領域に残る可能性があります。このことはいろいろなセキュリティの問題を生じます。たとえば、ボリュームのパスワードやキーファイルをあとから変更しても、敵対者が TrueCrypt ボリュームの古い(ヘッダーが再暗号化される前にマウントするのに必要だった)ヘッダーやその断片を見つけたら、古いパスワードでボリュームをマウントできるかもしれません。これを防ぐには、以下のどれかを実行してください。

- ファイル型のかわりに、パーティション/デバイス型 TrueCrypt ボリュームを使う。

- デフラグのあとで、ホストボリューム(断片化していたファイルシステム)の空き領域に完全消去をかける。
- TrueCrypt ボリュームを格納しているホストファイルシステムではデフラグをしない

ジャーナリングファイルシステム

ファイル型 TrueCrypt コンテナをジャーナリングファイルシステム(NTFS のような)に格納する場合、TrueCrypt コンテナ(あるいは、その断片)のコピーがホストボリュームの空き領域に残る可能性があります。このことはいろいろなセキュリティの問題を生じます。たとえば、ボリュームのパスワードやキーファイルをあとから変更しても、敵対者が TrueCrypt ボリュームの古い(ヘッダーが再暗号化される前にマウントするのに必要だった)ヘッダーやその断片を見つけたら、古いパスワードでボリュームをマウントできるかもしれません。これを防ぐには、以下のどれかを実行してください。

- ファイル型のかわりに、パーティション/デバイス型 TrueCrypt ボリュームを使う。
- コンテナをジャーナリング機能がないファイルシステム(たとえば FAT32)に格納する。

「隠しボリューム区画づくりの前の安全策」も参照してください。

問題が起こったら

ここではTrueCryptを使っていて遭遇するかもしれない一般的な問題への解決策を提示します。ここにない問題であれば、次のところに記載があるかもしれません。

非互換性
既知の問題と制限
FAQ(よくある質問)

問題::

正常にボリュームがマウントされたのに、Windowsから「このデバイスは有効なファイルシステムではありません」というようなメッセージが出る。

想定される原因:

TrueCryptボリュームのファイルシステムが破損している、あるいはボリュームがフォーマットされていない。

対策案:

TrueCryptボリュームを修復するためにOSが用意しているファイルシステム修復ツールを使うことができます。Windowsでは chkdsk です。TrueCryptはこのツールをTrueCryptボリュームで使う簡単な方法を用意しています。(chkdskはファイルシステムを破損する可能性があるため) 最初にTrueCryptボリュームのバックアップコピーをとってから、そのボリュームをマウントしてください。TrueCryptメインウィンドウの(ドライブリストで)マウントされたボリュームを右クリックしてください。そして、表示されるメニューから「ファイルシステムの修復」を選択してください。

問題:

ボリュームは正常にマウントされ、TrueCryptではそのボリュームはマウントされていると表示されているにもかかわらず、WindowsのExplorerからボリュームにアクセスできない。(コンピュータまたはマイコンピュータなどにも表示されない) ときにはWindowsエクスプローラに正しいボリュームラベルが表示されない。

想定される原因:

Windows Explorerの問題です。

対策案:

「*ツール -> ドライブリストの更新*」をクリックしてください。これでだめなら、Windows Explorer を再起動してください。(たとえば、ログオフして再度ログオンするなど) TrueCrypt メインウィンドーでドライブ文字をダブルクリックしてもボリュームを開くことができます。

問題:

ボリュームへの読み書きが非常に遅い。ベンチマークの結果によれば、私が使っている暗号化方式はハードディスクの速度より早いはずなのですが。

想定される原因:

なにかのアプリケーションがじゃまをしている可能性があります。T

対策案:

最初に、TrueCryptコンテナのファイル名になにかのアプリケーションに関連づけられた拡張子(たとえば、.exe, .sys, .dll)がつけられていないことを確認してください。もし、そういった拡張子がついていると、Windowsやアンチウィルスソフトがコンテナを妨害したり、ボリュームのパフォーマンスを低下させることがあります。
つぎに、じゃまをしている可能性があるアプリケーションを停止するかアンインストールしてみてください。これは、アンチウィルスや自動デフラグツールなどによることが多いようです。アンチウィルスが原因の場合は、その設定画面でリアルタイムスキャンを停止することができることがあります。これでも効果がなければ、アンチウィルスソフトを臨時に停止してください。それでもだめなら、完全にアンインストールして、再起動してみてください。

問題:

隠しボリュームを作ろうとしたら、作成可能な最大サイズが予想外に小さい。(外殻ボリュームにはこれよりずっと大きい空き容量があるのですが)

想定される原因:

ファイルの断片化(フラグメンテーション)

または

クラスタサイズが小さすぎるところに、外殻ボリュームのルートディレクトリに置いたフォルダーやファイルが多すぎるということが考えられます。

対策案:

外殻ボリュームにデフラグをかける。(マウントしてコンピュータまたはマイコンピュータのそのドライブレターを右クリック、プロパティをクリック、ツール・タブを選択、「最適化する」をクリック) ボリュームのデフラグが終わったら、もう一度隠しボリューム作成を試してください。

これで効果がなければ、外殻ボリュームのすべてのファイルとフォルダーを Shift+Delete を押すことで削除してください。フォーマットで消してはいけません。(事前に「ごみ箱」と「システムの復元」を無効にすることを忘れないでください) そして、完全に空になった外殻ボリュームに隠しボリュームを作成してみてください。(テスト目的だけです) それでも隠しボリュームの可能な最大サイズが変わらなければ、問題は拡張ルートディレクトリにありそうです。もし(ウィザードの最終ステップで)クラスタサイズを既定値のままにしなかったなら、こんどはクラスタサイズを既定値のままにして外殻ボリュームをフォーマットしなおしてください。

さらにこれでもだめなら、外殻ボリュームを再フォーマットして前回より少ないファイルやフォルダーをルートに置いてください。それでだめなら、再フォーマットしてルートのファイルやフォルダーを減らすことを繰り返してください。やってられないとか、効果なしなら、より大きいクラスタサイズで外殻ボリュームを再フォーマットしてください。それでも解決しなければ、解決するまで外殻ボリュームをクラスタサイズを大きくしながら再フォーマットを繰り返してください。もし、なぜクラスタサイズがそんなに大きいのかと聞かれたら、より高性能(高速)を目指したからと答えてください。(「クラスタのサイズ」を参照)

問題:

パーティション/デバイスを暗号化しようとすると、TrueCrypt ボリューム作成ウィザードから使用中だというメッセージが出て、実行できません。

対策案:

OS のブートパーティションを暗号化しようとはしていませんね？(TrueCrypt は、これはサポートしていません)
そうでなければ、そのパーティション/デバイスを何らかの形で使うプログラム(たとえば、アンチウィルスなど)を停止、アンインストールなどしてください。これでもだめなら、デスクトップのコンピュータ(またはマイコンピュータ)アイコンを右クリックして*管理 -> 記憶域 -> ディスクの管理*を選んでください。そこで暗号化したいパーティションをクリックし、*ドライブレターの変更*をクリックし、*ドライブ文字とパスの変更*をクリック、*削除*をクリックして *OK* としてください。最後にシステムを再起動してください。

問題:

隠しボリュームを作成しようとすると、ウィザードが外殻ボリュームをロックできないと言ってきます。

想定される原因:

外殻ボリュームのファイルを何かのアプリケーションが開いています。

対策案:

外殻ボリュームのファイルを使うアプリケーションをすべて閉じてください。それでもだめなら、アンチウィルスを停止するかアンインストールし、再起動して試してください。

問題:

以下のどれかが発生:

1. *TrueCrypt ボリュームをマウントできない。*
2. *NTFS TrueCrypt ボリュームを作成できない。*

さらに、エラーメッセージが出る: 「*他のプロセスで使用中のため、プロセスはファイルにアクセスできません*」

想定される原因:

他のアプリケーションが干渉している可能性があります。これは TrueCrypt のバグではありません。OS が他のアプリケーションが排他アクセスのためデバイスをロックしていると TrueCrypt へ通知しています。(だから TrueCrypt はデバイスにアクセスできないわけです)

対策案:

干渉するアプリケーションを停止またはアンインストールすることで、通常は解決します。アンチウィルスやディスク管理ツールなどがこの例です。

問題:

ネットワークの先で共有になっているファイル型コンテナをアクセスしようとすると、「メモリー不足」のエラーになります。

想定される原因:

Windows レジストリの IRP スタックサイズの値が小さすぎる。

対策案:

Windows レジストリで *IRP スタックサイズ*キーを探し、その値を大きくし、システムを再起動する。このキーがレジストリに存在しなければ、次のように作成してください。
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters

そして、その値を 16 以上に設定し、システムを再起動してください。詳細については下記を参照
http://support.microsoft.com/default.aspx?scid=kb;en-us;285089

非互換性

現在のところ、特に非互換性について記載することはありません。

既知の問題と制限

- TrueCrypt ボリュームパスワードはプリンタブルな ASCII キャラクターでなくてはいけません。パスワードに ASCII キャラクター以外を使うことはサポートしていませんし、問題を起こすこともあります。(ボリュームをマウントできないなど)

- Windows2000 の制限のため、TrueCrypt は Windows2000 での Windows マウントマネージャをサポートしていません。したがって、Windows2000 のいくつかの組み込みツール(たとえばディスク・デフラグ)は TrueCrypt ボリュームに対しては機能しません。さらに、Windows2000 のマウントマネージャを使うこともできません。たとえばマウントポイントに TrueCrypt ボリュームを割り当てる(TrueCrypt ボリュームをフォルダーとして割り当てる)ということなどです。

- Windows ボリュームシャドーコピーサービスは現在のところサポートしていません。

- TrueCrypt で暗号化されたフロッピーディスク: フロッピーディスクが排出され他のディスクが挿入されると、ゴミが書かれたり読まれたりしてデータが破損するかもしれません。これはフロッピーディスクをまるごとボリュームとして扱う場合で、フロッピーディスク上のファイル形式コンテナの場合ではありません)

よくある質問(FAQ)と答え

TrueCrypt FAQ の最新版は http://www.truecrypt.org/faq.php で入手できます。(英語版)

Q: 「クイックスタートガイド」のような初心者用の説明はありますか?

A: はい。第 1 章の「初心者のためのチュートリアル」が TrueCrypt ボリュームの作成、マウント、使用についてスクリーンショットや段階を追った解説を記載しています。

Q: パスワードを忘れてしまいました。TrueCrypt ボリュームのファイルを復元する方法はありませんか?

A: TrueCrypt は正しいパスワードまたは暗号化に使ったキーなしで、暗号化されたデータを部分的でも完全にでも復元する機能はまったく持っていません。復元するたった一つの方法はパスワードやキーをクラックして暗号を破ることですが、パスワード/キーファイルの質や長さ、キーのサイズ、ソフトやハードの効率性、その他の要素によって、数千年、数百万年かかるかもしれません。

Q: TrueCrypt はパスワードをディスクに保存しますか?

A: いいえ。

Q: TrueCrypt ボリュームに保存されたビデオ(.avi, .mpg, etc.) を直接再生できますか?

A: はい、TrueCrypt の暗号化ボリュームは通常のディスクと同じです。正しいパスワードやキーファイルで TrueCrypt ボリュームをマウント(オープン)してください。ビデオファイルをダブククリックすれば、OS がそのファイルタイプに関連づけられているアプリケーション(通常は再生ソフト)を起動します。再生ソフトはビデオファイルの最初のある部分を TrueCrypt の暗号化ボリュームから RAM に読み込みます。その部分が読み込まれているあいだ、TrueCrypt は RAM にデータを復号します。そして、復号された RAM 中のデータが再生ソフトによって再生されるということになります。それが再生されているあいだに、再生ソフトは次の一定部分を TrueCrypt の暗号化ボリュームから RAM に読み込み、このプロセスがくりかえされることになります。
同じことが録画でもおこなわれます。ビデオファイルの一部でも TrueCrypt ボリュームに書き込まれる前に、TrueCrypt は RAM 中でそれを暗号化しディスクに書き込みます。このプロセスは即時自動暗号化/復号(on-the-fly encryption/decryption)と呼ばれ、ビデオファイルだけではなくすべてのファイルタイプに適用されます。

Q: TrueCrypt はずっとこのままオープンソースでフリーなのですか?

A: はい、そうです。商業版は計画していませんし、そうもならないでしょう。私たちはオープンソースでフリーなセキュリティソフトウェアに信頼をおいています。

Q: TrueCrypt プロジェクトに寄付できますか?

A: はい、できます。詳細については *http://www.truecrypt.org/donations/*を参照してください。

Q: ファイル名やフォルダー名も暗号化されるのですか？

A: はい、そうです。TrueCrypt ボリュームの中のファイルシステム全体(ファイル名、フォルダー名、ファイルの内容なども含む)が暗号化されます。これはファイルコンテナ(仮想 TrueCrypt ディスク)と TrueCrypt 暗号化パーティション/デバイスの両方について適用されます。

Q: USB フラッシュドライブでどのようにして TrueCrypt を使うことができますか？

A: 二つの方法があります

1) USB フラッシュドライブ全体を暗号化する。しかし、この方法では TrueCrypt を USB フラッシュドライブから起動することはできません。
   注意: Windows では USB フラッシュドライブの複数パーティションをサポートしていません。
2) USB フラッシュドライブに TrueCrypt ファイルコンテナを作る。(作り方については「初心者のためのチュートリアル」を参照) USB フラッシュドライブに充分な空き領域があれば(そうなるように TrueCrypt コンテナの大きさを決めれば)、TrueCrypt を USB フラッシュドライブの中に(コンテナの中ではなく、コンテナと併存して)格納し、TrueCrypt を USB フラッシュドライブから起動することができるでしょう。(詳細は「トラベラーモード」参照)

Q: TrueCrypt が扱える最大ボリュームサイズはどのくらいですか？

A: TrueCrypt ボリュームは *8589934592* GB までを扱えます。しかしいくつかの制限となる要因を考慮する必要があります。たとえば、ファイルシステムの制限、ハードウェア接続や OS による制約などです。

Q: SHA-1 は破られたと聞きました。TrueCrypt に影響しますか?

A: SHA-1 は TrueCrypt で使われている三つのハッシュアルゴリズムの一つ(ユーザーが選択可能)です。2005 年に、総当たり攻撃よりは平均的には楽に(2^{80} ステップのかわりに 2^{63} ステップで) SHA-1 の問題点を探す方法が発明されました。しかし、TrueCrypt は数値署名を作るのに SHA-1 を使いません(TrueCrypt は SHA-1 を疑似乱数関数としてはほとんど使いません)し、近い将来に SHA-1 の問題点が発見され TrueCrypt ボリュームの安全性に影響がでるとは思えません。この見込みについては M. Bellare の New Proofs for NMAC and HMAC: Security without Collision-

Resistance という論文で証明されています。詳細は「ハッシュアルゴリズム」の章を参照してください。

Q: ホットプラグデバイス(USB フラッシュディスクや USB ハードディスク)にマウントされた TrueCrypt ボリュームがあるときに、そのデバイスを取り外したり電源を切ったりできますか?

A: デバイスを取り外したり電源を切ったりする前に、TrueCrypt で TrueCrypt ボリュームをアンマウントし、可能なら「取り出し」(「コンピュータ」か「マイコンピュータ」の該当デバイスを右クリック)操作をするか、「ハードウェアの安全な取り外し」(タスクバーから操作可能)をしてください。そうしないと、データが失われるかもしれません。

Q: 私の TrueCrypt パーティション/コンテナをどのコンピュータにでもマウントできますか?

A: TrueCrypt ボリュームは OS から独立しています。TrueCrypt を起動できるコンピューターならどれにでもマウントできます。(「管理者権限がなくても Windows で TrueCrypt を使えますか?」も参照)

Q: OS を再インストールしても元からある TrueCrypt パーティション/コンテナをマウントできますか?

A: はい、TrueCrypt ボリュームは OS から独立しています。ただし、OS のインストーラが TrueCrypt ボリュームがあるパーティションをフォーマットしないようにしてください。

Q: 隠しボリュームはどうやってマウントするのですか?

A: 隠しボリュームは通常の TrueCrypt ボリュームと同じ方法でマウントできます。「ファイルの選択」または「デバイスの選択」をクリックして、外殻ボリュームを選択(すでにマウント済でないことを確認)してください。つぎに「マウント」をクリックし、隠しボリューム用のパスワードを入力してください。マウントしようとしているのが隠しボリュームか外殻ボリュームかは入力されたパスワードで決定されます。(つまり、外殻ボリューム用パスワードを入力すれば外殻ボリュームが、隠しボリューム用パスワードを入力すれば隠しボリュームがマウントされます)

注意: TrueCrypt は入力されたパスワードで標準ボリュームヘッダーを復号しようとします。それに失敗すれば、通常なら隠しボリュームのヘッダーがあるはずのセクター(ボリュームの最後からの第 3 セクター)を RAM に読み込み、入力されたパスワードでそれを復号しようとします。隠しボリュームのヘッダーは単なるランダムデータにしか見えないので、それと特定することはできないことに留意してください。ヘッダーの復号に成功(どのように成功したかを判断するかについては「暗号化の仕組み」を参照)すると、まだ RAM にあるヘッダーから隠しボリュームの大きさを得て、隠しボリュームをマウントします。(大きさはオフセットで決定されます)

詳細については「隠しボリューム」に記述しています。

Q: 管理者権限がなくても Windows で TrueCrypt を使うことはできますか？

A: はい。しかし、管理者がシステムに TrueCrypt をインストールしたあと(または管理者があなたに管理者権限を付与したあと)に限ります。その理由は、TrueCrypt 即時自動暗号化/復号のデバイスドライバを必要とし、管理者権限がないと Windows にデバイスドライバをインストールできないからです。システム管理者が TrueCrypt をインストールしたあとは、管理者権限がないユーザーでも TrueCrypt を起動しどんな種類の TrueCrypt ボリュームでもマウント/アンマウント、書き込み、読み出しすることができ、ファイル型 TrueCrypt ボリュームの作成もできます。しかし、管理者権限がないユーザーはパーティションを暗号化/フォーマットしたり NTFS ボリュームをつくることはできませんし、TrueCrypt のインストール/アンインストールもできません。また、デバイス型ボリュームのパスワード/キーファイル変更や TrueCrypt パーティション/デバイスのヘッダーのバックアップとリストアはできません。TrueCrypt をトラベラーモードで動かすこともできません。

Q: パスワードのハッシュはどこかに保存されますか？

A: いいえ。

Q: TrueCrypt ボリュームにアプリケーションをインストールし、動かすことができますか？

A: はい。

Q: TrueCrypt はどのようにして正しいパスワードが入力されたかを判断しているのですか？

「技術解説」の「暗号化の仕組み」を参照してください。

Q: TrueCrypt はハードウェア/ソフトウェア レイドと Windows のダイナミックボリュームをサポートしていますか？

A: はい、サポートしています。Windows のダイナミックボリュームを TrueCrypt ボリュームとしてフォーマットする場合には、(Windows のディスク管理ツールを使って)ダイナミックボリュームを作成したあと、システムを再起動して、TrueCrypt ボリューム作成ウィザードの「デバイス選択」に目的のボリュームが表示され、選択できるようにすることを忘れないようにしてください。「デバイス選択」ウィンドーで、ダイナミックボリュームは単一のデバイスとしては表示されません。そのかわり、ダイナミックボリュームを構成するすべてのボリュームが表示されるので、ダイナミックディスク全体をフォーマットするために、そのうちのどれか一つを選択してください。

Q: CD や DVD に保管された TrueCrypt コンテナをマウントできますか？

A: はい、できます。しかし、Windows2000 で読み出し専用メディア(CD/DVD 他)にある TrueCrypt ボリュームをマウントする場合には、TrueCrypt ボリュームを FAT でフォーマットしな

くてはならないことを憶えておいてください。(Windows2000 では読み取り専用メディアの NTFS ファイルシステムはマウントできません)

Q: TrueCrypt をインストールせずに実行できますか？

A: はい、「トラベラーモード」の章を参照してください。

Q: トラベラーモードで TrueCrypt を実行しようとすると、なぜ Windows Vista は毎回許可を求めてくるのですか？

A: TrueCrypt をトラベラーモードで動かすときには、TrueCrypt は TrueCrypt デバイスドライバを読み込んで起動する必要があります。TrueCrypt は透過的な即時暗号化/復号機能を提供するためにデバイスドライバを必要としますが、管理者権限がないユーザーは Windows でデバイスドライバを起動することができません。だから、Windows Vista は管理者権限で TrueCrypt を起動してもいいうどうかを問い合わせてくるというわけです。

TrueCrypt をトラベラーモードで動かすのではなく、システムにインストールすれば、毎回許可を求められることはありません。

Q: Windows の終了や再起動の前に、TrueCrypt ボリュームをアンマウントする必要がありますか?

A: いいえ。TrueCrypt はシステムの終了や再起動時には、すべてのマウントされた TrueCrypt ボリュームを自動的にアンマウントします。

Q: パーティションとファイルコンテナと、どちらの TrueCrypt ボリュームがいいでしょうか?

A: ファイルコンテナは簡単に移動、リネームができ通常のファイルと同じに扱うことができます。パーティション/デバイスは性能に関しては優れています。コンテナがひどく断片化していると、コンテナへの読み書きがあきらかに遅くなることに注意してください。また、コンテナがひどく断片化していると、コンテナの中の隠しボリュームをマウントするのがあきらかに遅くなります。この理由としては、隠しボリュームのヘッダーが外側のコンテナの終端部分に記録され、コンテナが断片化していると終端までいくのに時間がかかるためです。これを解決するにはコンテナがアンマウントされている状態のときに、デフラグを実行してください。

Q: TrueCrypt パーティションをフォーマットするとどうなるのでしょうか?

この FAQ の「暗号化ボリュームのファイルシステムを変更できますか？」を参照してください。

Q: 暗号化ボリュームのファイルシステムを変更できますか？

A: マウントされていれば、可能です。TrueCrypt ボリュームは FAT12, FAT16, FAT32, NTFS, またはほかのどんなファイルシステムででもフォーマットすることができます。TrueCrypt ボリュームは普通のボリュームと同じように扱うことができるので、コンピュータまたはマイコンピュー

タなどでデバイスのアイコンを右クリックし、フォーマットを選んでください。ボリュームの内容は失われますが、ボリームは暗号化された状態のままになります。もし、パーティション形式の TrueCrypt ボリュームがマウントされていないときにそのパーティションをフォーマットすると、ボリュームは破壊され、パーティションは暗号化された状態ではなくなり、空となります。

Q: Windows 起動時に自動的に TrueCrypt を起動してパスワード要求を表示し、ボリュームをマウントするように設定できますか？

はい、以下の手順で可能です。

1. ボリュームをマウントし、「ボリューム -> 現在マウントされているボリュームをお気に入りに保存」を選択
2. 「設定 -> 各種設定」の Windows の項目、「ログオン時に自動的に実行する内容」で次のオプションを有効にしてください。
   - TrueCrypt を開始
   - お気に入りボリュームをマウント
3. 「各種設定」ウィンドウで OK をクリックしてください。

Q: 隠しボリュームのパスワードを変更できますか？

A: はい。パスワード変更ダイアログは標準ボリュームにも隠しボリュームにも機能します。*ボリュームパスワード変更ダイアログの「現在のパスワード」*に隠しボリュームのパスワードを入力してください。

注: TrueCrypt は最初に標準ボリュームヘッダーを復号しようとします。これに失敗するとその中に隠しボリュームがあると想定し、隠しボリュームのヘッダーがあると想定される位置のデータを復号しようとします。これが成功するとパスワード変更は隠しボリュームに対して適用されることになります。(どちらの試みも「*現在のパスワード*」に入力されたパスワードを使います)

Q: HMAC-RIPEMD-160 か HMAC-SHA-1 を使うとき、キーサイズは 160 ビットに制限されているのですか？

A: いいえ。TrueCrypt は(HMAC アルゴリズムだけではなく)ハッシュ関数の出力を直接暗号化キーとして使うことはありません。詳細は「ヘッダーキーの導出、ソルト、および反復回数」を参照してください。

Q: ボリュームに保存されたデータを失わずに、ヘッダーキー導出アルゴリズムを変更できますか? (たとえば、HMAC-SHA-1 から HMAC-Whirlpool へ)

A: はい。「*ボリューム* 」-> 「*ヘッダーキー導出アルゴリズムの設定*」を選択してください。

Q: CBC モードで暗号化されたボリューム(TrueCrypt4.0 かそれ以前で作られたボリューム)を最新版の TrueCrypt でマウントできますか?

A: はい。しかし、CBC モードより LRW モードのほうがより安全です。したがって、最新版の TrueCrypt で新しくボリュームを作り、古いボリュームの中身をすべて新しいほうに移すことを強くおすすめします。TrueCrypt 4.1 ではボリュームはつねに LRW モードで暗号化されます。いまや CBC モードは使われず、互換性のためにサポートしているだけです)

Q: 2GB 以上の TrueCrypt コンテナをどうやって DVD に焼くのですか？

A: あなたが使っている DVD 作成ソフトで DVD のフォーマットを選択できるはずです。そこで、UDF フォーマットを選んでください。(ISO フォーマットは 2GB を越えるファイルをサポートしていません)

Q: TrueCrypt はどのようなライセンス形態で配布されているのですか？

A: ライセンスは TrueCrypt のバイナリまたはソースコードのパッケージに含まれる License.txt に記載されており、d *http://www.truecrypt.org/license.php* で入手することもできます。

Q: Windows の *ファイルセレクタ*がマウントした最後のコンテナや最後に選択したキーファイルを記憶しています。防止できますか？

A: はい。まだであれば、TrueCrypt4.2a 以降にアップグレードしてください。TrueCrypt を起動してメインウィンドウの「*履歴を保存しない*」を有効にしてください。「*履歴を保存しない*」を有効にしたくなければ、コンテナアイコンを TrueCrypt.exe のアイコンにドラッグ(TrueCrypt は自動的に起動します)するか、TrueCrypt プログラムウィンドウにドラッグすれば、ファイルセレクタを使うことを避けることができます。同様に、キーファイルもキーファイルウィンドウかパスワード入力ウィンドウへドラッグすることができます。

Q: 現在保存しているデータを失わずに、パーティションを暗号化できますか？

A: 残念ですが、TrueCrypt ではこのようなことはできません。

Q: マウントされた TrueCrypt ボリュームの内容に対して、chkdsk や Defrag といったツールを使うことはできますか？

A: はい。TrueCrypt ボリュームは本物の物理的なディスクと同じに扱うことができますから、どんなファイルシステムのチェックや修復、デフラグのツールでもマウントされた TrueCrypt ボリュームに対して使うことができます。

Q: Windows に痕跡を残さずに TrueCrypt を使うことはできますか?

A: はい。これは BarPE のもとで TrueCrypt をトラベラーモードで起動することで実現できます。BartPE とは Bart's Preinstalled Environment (バートのプリインストール環境)を意味します。これは、基本的に用意された WindowsOS そのものを CD/DVD に格納し(レジストリ、臨時ファイル、他は RAM に保持されます - ハードディスクはまったく使いませんし、ハードディスクが存在する

必要もありません)、そこからWindowsを起動するというものです。フリーウェアである Bart's PE Builder はWindowsインストールCDをBartPEに変換することができます。TrueCrypt 3.1以降を使っているなら、BartPEのTrueCryptプラグインは必要ありません。BartPEを起動し、最新のTrueCryptをRAMディスク(BartPEが作成)にダウンロードし、パッケージをRAMディスクに展開、TrueCrypt.exeをRAMディスクのSetup Filesフォルダーから起動するだけです。
(SetupFilesフォルダーはパッケージを展開すれば生成されます)

Q: TrueCryptボリュームの中に格納されているTrueCryptボリュームをマウントすることはできますか?

A: はい、TrueCryptボリュームは無制限に入れ子にできます。

Q: TrueCryptと他の自動即時暗号化ツールを同じシステムで併用できますか?

A: TrueCryptと他の自動即時暗号化ツールを併用することで問題が起きるとも起きないとも聞いていません。

Q: TrueCryptパーティションのサイズを変更できますか?

A: 残念ですが、こういったことはできません。PartitionMagicのようなプログラムでTrueCryptパーティションのサイズを変更すると、多くの場合はデータを壊すことになるでしょう。

Q: TrueCryptはWindows Vista x64 (64-bit) Editionで動きますか?

A: はい、動きます。(バージョン 4.0の場合) 注意: すべてのTrueCryptの.sysと.exeファイルは認証機関GlobalSignによって発行されたTrueCrypt Foundationのデジタル認証によってデジタル署名されています。

Q: TrueCryptはWindows XP x64 (64-bit) Editionで動きますか?

A: はい、動きます。

Q: TrueCryptはWindows98やWindowsMEで動きますか?

A: Windows98/MEで動くTrueCryptの最後のバージョンは1.0です。しかし、われわれがこのバージョンを(Windows98/MEについても)サポートしないことに注意してください。(詳細は「バージョン履歴」を参照)

Q: TrueCryptはLinuxで動きますか?

A: はい。

Q: Windows と Linux の両方で TrueCrypt ボリュームをマウントできますか?

A: はい。TrueCrypt ボリュームは完全にクロスプラットフォーム(OS を問わない)です。

Q: TrueCrypt ボリュームの一部が破損するとどうなりますか?

A: 暗号化データではあるひとつのバイトが破損すると、通常はそれが発生した暗号化ブロック全体が破損したことになります。TrueCrypt では暗号化ブロックのサイズは 16 バイト(128 ビット)です。TrueCrypt で使われる動作モードはあるブロック内でのデータ破損が他のブロックに影響を及ぼさないことを保証します。(詳細は「動作モード」を参照)

ハードウェアやソフトウェアのエラーや誤動作で、TrueCrypt ボリュームのファイルが破損することもありえます。ですから、重要ファイルは定期的にバックアップをとることをすすめます。(もちろん、TrueCrypt ボリュームに記録された暗号化データにかぎらず、すべての重要なデータについて言えることです)TrueCrypt ボリュームにあるすべてのファイルをバックアップするだけの空き領域がない場合、少なくともボリュームヘッダーだけでもバックアップをとっておくことを強くおすすめします。ここにはマスターキーが記録されています。(バックアップしたファイルのサイズは 1024 バイトになるはずです) ボリュームヘッダーが破損すると、ほとんどの場合はボリュームはマウントできなくなります。ボリュームヘッダーをバックアップするには、「*デバイスの選択*」か「*ファイルの選択*」をクリックし、ボリュームを選択してください。それから「*ツール -> ボリュームヘッダーのバックアップ*」をクリックしてください。ヘッダーを復旧するには、同じ手順で最後に「*ボリュームヘッダーのリストア*」を選択してください。

「*TrueCrypt ボリュームの暗号化したファイルシスムが破損した場合、どうすればいですか?*」という質問も参照してください。

Q: TrueCrypt ボリュームの暗号化したファイルシスムが破損した場合、どうすればいですか?

A: TrueCrypt ボリュームのファイルシステムは他の暗号化されていないファイルシステムと同様に破損の可能性があります。こうなったとき、ファイルシステム OS が提供する修復ツールを利用することができます。Windows では chkdsk です。TrueCrypt はこのツールを TrueCrypt ボリュームで使う簡単な方法を用意しています。(chkdsk はファイルシステムを破損する可能性があるため) 最初に TrueCrypt ボリュームのバックアップコピーをとってから、そのボリュームをマウントしてください。TrueCrypt メインウィンドウの(ドライブリストで)マウントされたボリュームを右クリックしてください。そして、表示されるメニューから「*ファイルシステムの修復*」を選択してください。

Q: 企業内で TrueCrypt を使っています。ユーザーがボリュームのパスワードを忘れたとき(またはキーファイルを失ったとき)に管理者がリセットする方法はありますか?

A: TrueCrypt には「裏口」は用意されていません。しかし、TrueCrypt ボリュームのパスワード/キーファイルをリセットする方法はあります。ボリュームを作ったあと管理者権限を持たないユーザーにそのボリュームの使用を認める前に、(*ツール* -> *ボリュームヘッダーのバックアップ*を選択して)そのヘッダーのバックアップをとります。パスワード/キーファイルから導出された暗号化されたヘッダーキーで暗号化されているボリュームヘッダーは、ボリュームを暗号化したマスターキーを持っています。そこで、ユーザーにパスワードを選んでもらいその人のためにパスワードを設定します。(*「ボリューム」* -> *「ボリュームのパスワード変更」*) そうすれば、ユーザーにそのボリュームの使用許可を与えるとともに、いつでも管理者の許可や助力なしで任意のパスワードに変更させることができます。ユーザーが自分が決めたパスワードを忘れた場合でも、ボリュームヘッダーのリストアを実行(*ツール* -> *ボリュームヘッダーのリストア*)をすることで、ボリュームのパスワードをオリジナルの管理者パスワード/キーファイルに戻すことができます。

Q: ある単一の TrueCrypt ボリュームを複数の OS から同時にマウントできますか(ボリュームがネットワークで共有されている場合など)?

A: 可能です。しかし、ボリュームはそれぞれのシステムで読み出し専用でマウントする必要があります。(「マウントオプション」を参照) この制限は非暗号化ボリュームについても同様です。その理由として、たとえば、実際のところある OS の通常のファイルシステムから読み出したデータは他の OS でファイルシステムが変更されたりすると整合性が保てなくなってしまいます。(その結果、データ破損になります)

Q: 必要がなくなった場合、どうすれば暗号を解除できますか?

A: 「暗号化を解除するには」を参照してください。

Q: TrueCrypt コンテナがどれほど断片化していてもマウントできますか ?

A: はい。しかし、コンテナが非常に断片化していると、コンテナへの読み書きがあきらかに遅くなることに注意してください。また、コンテナが非常に断片化していると、コンテナの中の隠しボリュームをマウントするのがあきらかに遅くなります。この理由としては、隠しボリュームのヘッダーが外側のコンテナの終端部分に記録され、コンテナが断片化していると終端までいくのに時間がかかるためです。これを解決するにはアンマウントした状態のときにコンテナ全体をデフラグするか、パーティションかデバイスに隠しボリュームを作るようにしてください。

Q: TrueCrypt コンテナをコピーする前にコンピュータを再起動する必要はありますか?

A: いいえ、必要ありません。

Q: ボリュームをリムーバブルメディアとしてマウントすると、何が変わるのですか?

A: たとえば Windows が自動的に TrueCrypt ボリュームに *Recycled* や *System Volume Information* といったフォルダー(これらはごみ箱やシステムの復元機能のために作られます)を作ることを防止したいなら、このオプションにチェックを入れてください。しかし、これには不利

な点もあります。たとえば、このオプションを有効にすると、コンピュータまたはマイコンピュータのリストでは空き領域を表示しません。(これは TrueCrypt のバグではなく、Windows の制限です)

Q: TrueCrypt の空き領域やファイルなどを完全消去しなくてはなりませんか？

補足: 完全消去=安全な削除; 機密データを上書きして復活不可能にすること

A: 敵対者が(あなたにパスワードを明かさせるなど)ボリュームを復号できると信じるなら、そうしてください。そうでなければ必要ありません。ボリュームはまるごと暗号化されていますから。

Q: TrueCrypt はどのようにして、データを暗号化したアルゴリズムを判別するのですか?

A: TrueCrypt ボリュームを暗号化したアルゴリズムや動作モードは、試行することで判別されます。そのプロセスは正しいパスワードやキーファイルを与えられた場合のみ成功します。詳細は「技術解説」の「暗号化の仕組み」を参照してください。

Q: 既存のコンテナを複製することで、新しいコンテナを作っても安全ですか？

A: 新しい TrueCrypt コンテナを作る場合は、つねにボリューム作成ウィザードを使ってください。もし、コンテナをコピーして両方を使うと、両方に異なったデータが入ることになり暗号解析の手がかりになるかもしれません。なぜなら、両方のボリュームが同じキーセットを持つためです。

Q: TrueCrypt は OS の起動パーティションを暗号化できますか？

A: はい、ただし直接にではありません。Virtual PC , VirtualBox, VMware, QEMU, Bochs, などの仮想マシン(またはエミュレータ)で起動する OS を含むディスクイメージを TrueCrypt は即時自動暗号化することができます。(Buchs, QEMU, VirtualBox OSU などはフリーでオープンソースです。Virtual PC や VMware のいくつかのバージョンはフリーです)

暗号化を解除するには

TrueCrypt はコンテナやデバイスそのものの復号はサポートしていません。もし、暗号化が必要なくなって、暗号化を除去したいなら、下記の手順にしたがってください。

1. TrueCrypt ボリュームをマウントする。
2. TrueCrypt ボリューム内のすべてのファイルを TrueCrypt 外へ移動する。
3. TrueCrypt ボリュームをアンマウントする。
4. **TrueCrypt ボリュームがファイル型の場合**には、他の一般のファイルと同様の操作でそのファイル(コンテナ)を削除する。

   ボリュームがパーティション型(USB フラッシュドライブも含む)の場合には上記 1-3 に続いて、下記の手順による。
   a. デスクトップかスタートメニューの「コンピュータ」か「マイコンピュータ」を右クリックし「管理」を選択する。「コンピュータの管理」ウィンドウが表示される。

   b. 「コンピュータの管理」ウィンドウの左のリストの「記憶域」の下の「ディスク管理」を選択する。
   c. 復号したいパーティションを右クリックして「ドライブ文字とパスの変更」を選択。

   d. 「ドライブ文字とパスの変更」ウィンドウでドライブ文字が表示されなければ「追加」、それ以外は「キャンセル」をクリックする。
   「追加」をクリックした場合は「ドライブ文字またはパスの追加」が表示されるので、割り当てたいドライブ文字を選んで OK をクリックする。
   e. 「コンピュータの管理」ウィンドウで復号したいパーティションを再度クリックする。そして、「フォーマット」を選択すると「フォーマット」ウィンドウが表示される。
   f. 「フォーマット」ウィンドウで OK をクリックする。フォーマットが完了すれば、そのパーティションは読み書きのために TrueCrypt でマウントする必要はない。

   ボリュームがデバイス型(つまり、デバイスが区画にわけられていなくて、デバイスがまるごと暗号化されている)の場合には上記 1-3 に続いて、下記の手順による。

   a. デスクトップかスタートメニューの「コンピュータ」か「マイコンピュータ」を右クリックし「管理」を選択する。「コンピュータの管理」ウィンドウが表示される。

   b. 「コンピュータの管理」ウィンドウの左のリストの「記憶域」の下の「ディスク管理」を選択する。
   c. 暗号化デバイスを示す領域を右クリックし、「新規パーティション」または「新規シンプルボリューム」を選択する。
   d. 警告: 作業を続ける前に、目的のデバイスを選んでいるかどうかを確認してください。そうでないと、そこに保存されたすべてのファイルが失われることになります。
   「新規パーティションウィザード」か「新規シンプルボリュームウィザード」が表示されるので、新規パーティションを作成するためにウィザードの指示にしたがう

こと。パーティションが作成されれば、そのパーティションは読み書きのために TrueCrypt でマウントする必要はない。

TrueCrypt のアンインストール

TrueCrypt をアンインストールするには、Windows Xp では「スタート->コントロールパネル->プログラムの追加と削除」->TrueCrypy->変更と削除」と進んでください。Windows Vista では「スタート->コントロールパネル->プログラム: プログラムの削除->TrueCrypt-> 変更と削除」と進んでください。

TrueCrypt をアンインストールしても TrueCrypt ボリュームは削除されません。TrueCrypt をインストールするかトラベラーモードで起動すれば、その TrueCrypt ボリュームをまたマウントできます。

TrueCrypt システムファイルとアプリケーションデータ

注意: `%windir%` は windows をインストールした主要パス(通常は `C:¥WINDOWS`)のことです。

TrueCrypt ドライバ

`%windir%\SYSTEM32\DRIVERS\truecrypt.sys` (32-bit Windows)

または

`%windir%\SysWOW64\drivers\truecrypt.sys` (64-bit Windows)

注意: TrueCrypt がトラベラーモードで動くなら、このファイルは存在しません。

TrueCrypt 設定 / アプリケーションデータ:

次のファイルがアプリケーションデータが通常保存される場所に保存されます。(たとえば C:¥Documents and Settings¥*UserName*¥Application Data¥TrueCrypt¥, *UserName* はあなたの Windows のユーザー名) トラベラーモードでは、これらのファイルは TrueCrypt.exe を起動するフォルダー(TrueCrypt.exe が存在するフォルダー)に保存されます。**警告:TrueCrypt はこれらのファイルを暗号化しません。**

`Configuration.xml`

`Default Keyfiles.xml`
注意 TrueCrypt の該当する機能を使っていなければ、このファイルは存在しないかもしれません。

`Favorite Volumes.xml`
注意 TrueCrypt の該当する機能を使っていなければ、このファイルは存在しないかもしれません。

History.xml（TrueCrypt ボリュームとして直近のマウント試行があったか TrueCrypt ホストとして使われたファイルやデバイスや直近 20 件のリスト； この機能は無効にすることができます。「履歴を保存しない」の項を参照）

注意 TrueCrypt の該当する機能を使っていなければ、このファイルは存在しないかもしれません。

技術解説

表記法

C	暗号テキストブロック
$D_K()$	暗号化/復号キー K を使う復号アルゴリズム
$E_K()$	暗号化/復号キー K を使う暗号化アルゴリズム
$H()$	ハッシュ関数
i	n-bit ブロックのブロックインデックス; n は状況による
K	暗号キー
P	プレーンテキストブロック
^	排他的論理和 (XOR)
⊕	加算して 2^n で割った余り。n が左のオペランドと結果のビットサイズ。 (左のオペランドが 1-bit 値で、右のオペランドが 2-bit 値の場合: 1 ⊕ 0 = 1; 1 ⊕ 1 = 0; 1 ⊕ 2 = 1; 1 ⊕ 3 = 0; 0 ⊕ 0 = 0; 0 ⊕ 1 = 1; 0 ⊕ 2 = 0; 0 ⊕ 3 = 1)
⊗	有限体 GF(2^{128}): 二つの多項式剰余の乗算 $x^{128}+x^7+x^2+x+1$ (多項式の約分)
||	連結

暗号化の仕組み

TrueCrypt ボリュームをマウントするとき(パスワード/キーファイルが記憶されていないと仮定して)、次のステップが実行されます。

1. ボリュームの最初の 512 バイト(標準ボリュームのヘッダー)が RAM に読み込まれます。その最初の 64 ビットがソルトです。(「TrueCrypt ボリュームフォーマット仕様」を参照)

2. ボリュームの最後から 1536 バイトの位置から 512 バイトが RAM に読み込まれます。(「TrueCrypt ボリュームフォーマット仕様」を参照) もしそのボリュームに隠しファイルがあれば、この時点でそのヘッダーを読み込んだことになります。(隠しボリュームがあるかないかは、このデータを復号できるかどうかで決まります。詳細は「隠しボリューム」の項を参照)

3. TrueCrypt は(1)で読み込んだ標準ボリュームヘッダーを復号しようとします。復号の過程で使われたり生成されたりしたデータは RAM に保持されます。(TrueCrypt はこれらをけっしてディスクに保存しません) 次のパラメータは未知[1]で、試行錯誤で決定していきます。(以下の可能な組み合わせをすべて試します)

    a. ヘッダーキー導出に使われる PRF(PKCS #5 v2.0 に規定。「ヘッダーキーの導出、ソルト、および反復回数」を参照) これは以下のどれかになります:
    HMAC-RIPEMD-160, HMAC-SHA-1, HMAC-Whirlpool.
    ユーザーが入力したパスワード(一つ以上のキーファイルも適用されるかもしれない - 「キーファイル」の節を参照)と(1)で読み込まれたソルトはヘッダーキー導出関数へ渡され、一連の値(「ヘッダーキーの導出、ソルト、および反復回数」を参照)が作られます。そしてそれから、ヘッダー暗号化キーが生成され、第二ヘッダーキー(LRW モード)が形づくられます。

    b. 暗号化アルゴリズム: AES-256, Serpent, Twofish, AES-Serpent, AES-Twofish-Serpent など

    c. 動作モード: LRW, CBC(旧式で使われない), inner-CBC(旧式で使われない), outer-CBC(旧式で使われない)

    d. キーサイズ

4. 復号データの最初の 4 バイトが”TRUE”という ASCII 文字列であり、復号されたデータ(ボリュームヘッダー)の最後の 256 バイトの CRC-32 チェックサムが復号データの 8 番目のバイトの値と一致したなら、復号が成功したと判断します。(この値は暗号化されているので、敵対者にはわかりません。「TrueCrypt ボリュームフォーマット仕様」を参照) この条件が満たされなければ、プロセスは(3)に戻って継続します。

[1] これらのパラメータは、攻撃の困難さを強化するために秘密にされているのではなく、TrueCrypt ボリュームであるかどうかを事前に知ることができないためです。(単なるランダムデータと区別がつかない)　ボリュームヘッダーにこれらのパラメータを格納しておくと、こうはなりません。

しかし、今回は(1)で読んだデータの替わりに(2)で読んだデータ(隠しボリュームのボリュームヘッダーの可能性)を使います。これでも条件に合わなければ、マウント動作は終了します。(間違ったパスワード、ボリュームの破損、またはTrueCryptボリュームではないということになる)

5. これで正しいパスワード、適切な暗号化アルゴリズム、モード、キーサイズ、正しいヘッダーキー導出アルゴリズムがわかった(あるいは非常に高い可能性でわかったと仮定できる)ことになります。また、(2)で読んだデータを復号できたなら、隠しボリュームをマウントしようとしているとういことがわかり、そのサイズは(2)で読み込んで(3)で復号された結果から得ることができます。

6. 暗号化ルーチンは復号されたボリュームヘッダーから得られたマスターキー[1]と第二キーで再初期化されます。(「TrueCryptボリュームフォーマット仕様」を参照) このキーはボリュームヘッダー領域をのぞく、ボリュームのどのセクターでも復号するのに使うことができます。(ボリュームヘッダー領域は、ヘッダーキーで暗号化されます) これでボリュームはマウントされました。

「動作モード」、「ヘッダーキーの導出、ソルト、および反復回数」も参照してください。

動作モード

このバージョンのTrueCryptで 作成されるボリュームはLRWモードのみで暗号化されます。CBCモードは使われません。(しかし、TrueCryptの現バージョンではCBCモードで暗号化されたボリュームもマウントすることはできます) LRWモードはCBCモードより安全で、ディスクの暗号化に適しています。

LRWモードの説明

$C_i = E_{K1}(P_i \wedge (K2 \otimes i)) \wedge (K2 \otimes i)$

ここでは:
$K1$ は暗号化キー
$K2$ は第二キー(「弱い」キーと呼ばれることもある)
i は $K1$ から見た暗号ブロックのインデックス; 最初の暗号ブロック　$i = 1$
$\otimes$ は二つの多項式の乗算 剰余(modulo) $x^{128}+x^7+x^2+x+1$

128ビットブロックの暗号では $K2$ と i は128ビット値。

LRWモードについての詳細は[12]を参照。

[1] マスターキーはボリューム作成のときに生成され、あとで変更することはできません。ボリュームのパスワード変更は、新しいパスワードから導出される新しいヘッダーキーでボリュームヘッダーを再暗号化することで実施されます。

次の表は TrueCrypt に実装されているすべてのアルゴリズムと、それらの動作モードです。

暗号化アルゴリズム	動作モード	動作モード詳細
AES[1]	LRW	$C_i = E_{K1}(P_i \text{ ^ } (K2 \otimes i)) \text{ ^ } (K2 \otimes i)$
AES-Twofish (*E*2) (*E*1)	LRW	$C_i = E2_{K2}(E1_{K1}(P_i \text{ ^ } (K3 \otimes i))) \text{ ^ } (K3 \otimes i)$
AES-Twofish-Serpent (*E*3) (*E*2) (*E*1)	LRW	$C_i = E3_{K3}(E2_{K2}(E1_{K1}(P_i \text{ ^ } (K4 \otimes i)))) \text{ ^ } (K4 \otimes i)$
Serpent	LRW	$C_i = E_{K1}(P_i \text{ ^ } (K2 \otimes i)) \text{ ^ } (K2 \otimes i)$
Serpent-AES (*E*2) (*E*1)	LRW	$C_i = E2_{K2}(E1_{K1}(P_i \text{ ^ } (K3 \otimes i))) \text{ ^ } (K3 \otimes i)$
Serpent-Twofish-AES (*E*3) (*E*2) (*E*1)	LRW	$C_i = E3_{K3}(E2_{K2}(E1_{K1}(P_i \text{ ^ } (K4 \otimes i)))) \text{ ^ } (K4 \otimes i)$
Twofish	LRW	$C_i = E_{K1}(P_i \text{ ^ } (K2 \otimes i)) \text{ ^ } (K2 \otimes i)$
Twofish-Serpent (*E*2) (*E*1)	LRW	$C_i = E2_{K2}(E1_{K1}(P_i \text{ ^ } (K3 \otimes i))) \text{ ^ } (K3 \otimes i)$

カスケードの暗号のそれぞれは、それ自身のキーを使います。それぞれのキーは相互に独立しています。(補足: ヘッダーキーは一つのパスワードから導出されるものの、きちんと独立しています。「ヘッダーキーの導出、ソルト、および反復回数」を参照)

[1] この表では、”AES”は”AES-256”を意味しています。LRW で動く AES は LRW-AES とも呼ばれます。

ヘッダーキーの導出、ソルト、および反復回数

ヘッダーキーはマスターキー他のデータを持つ TrueCrypt ボリュームヘッダーの暗号化領域を暗号化、復号するのに使われます。(「暗号化の仕組み」と「TrueCrypt ボリュームフォーマット仕様」を参照) TrueCrypt ヘッダーキーと第二キー(LRW モード)を生成する技法は PBKDF2 であり、PKCS #5 v2.0 に規定されています。[7]を参照。(PKCS #5 v2.0 文書は RSA 研究所のご厚意で http://www.truecrypt.org/docs/pkcs5v2-0.pdf で入手可能)

512-bit ソルト(ボリューム作成プロセスで組み込みの乱数発生機構で生成されるランダム数)が使われます。ということは、それぞれのパスワードについて 2^{512} (2 の 512 乗)のキーがあるということです。これは、オフライン辞書攻撃に対する脆弱さを非常に大きく減少させます。(ソルトが使われると、事前にすべてのキーをコンピュータで組み合わせてパスワード辞書を作るということは、非常に難しくなります) [7] ソルトは TrueCrypt ボリューム作成過程で乱数発生機構によって生成される乱数値から成ります。ヘッダーキー導出関数は、HMAC-RIPEMD-160、 HMAC-Whirlpool、および HMAC-SHA-1([8, 9, 20, 22]を参照) に基づいており、ユーザーはどれかを選択できます。導出されるキーの長さは、基礎となるハッシュ関数の出力サイズに制限されません。(HMAC-SHA-1 または HMAC-RIPEMD-160 を使ったとしても、AES-256 のヘッダーキーはつねに 256 ビット長です) 詳細は[7]を参照してください。 ヘッダーキーを導出するにはキー導出関数を 2000 回(HMAC-Whirlpool を基礎としている場合は 1000 回)繰り返さなくてはいけません。これは徹底したパスワード探索(総当たり攻撃)に要する時間を非常に増大させます。 [7]

カスケードの個々の暗号が使うヘッダーキーは同じパスワード(キーファイルも適用されるかもしれない)から導出されますが、相互に独立しています。たとえば、AES-Twofish-Serpent では、ヘッダーキー導出関数はパスワードから 768-bit キーを導出するように指示を受けます。その後、このキーは三つの 256-bit キーに分割され、最初のものが Serpent で、二番目のものが Twofish、三番目のものが AES で使われます。キーが導出される元になったパスワードを求める方法は(弱いパスワードへの総当たり攻撃を除いて)ないので、敵対者がキーの一つを知ったとしても、それから他のキーを導出することはできません。

乱数発生機構

TrueCrypt 乱数発生機構(RNG)は RAM(メモリ)にランダム値のプール(集合)を作ります。プールは 320 バイト長で、以下から発生するデータで満たされます。

- マウスの動き
- キーストローク
- Linux のみ: Linux 内蔵 RNG(/dev/random と/dev/urandom の両方)から生成される値
- Windows のみ: MS Windows 暗号 API (500-ms 間隔で定期的に収集される)
- Windows のみ: ネットワークインターフェース統計(NETAPI32)
- Windows のみ: さまざまな Win32 ハンドル、時間変数、カウンタ(500-ms ごとに収集)

上記のソースのどれかから得られた値はプールに書き込まれ、個々のバイトに分割されます。(32-bit 値は 4 バイトに分割されます) これらのバイトは個々に modulo 2^8 addition 演算をしてキーファイルプールの(プールの古い値の上書きではなく)プールカーソルの位置に書き込まれます。バイトが書き込まれたら、プールカーソルは 1 バイト進み、終端までくるとプールの先頭に位置づけられます。プールに 8 バイト書き込むごとに、プール混合関数がプール全体に適用されます。(下記参照)

プール混合関数

この関数の目的は拡散です。拡散することで、個々の「生の」入力ビットの影響をできるだけ広げます。これは統計的関連を隠すことにもなります。プールに 8 バイトを書き込むごとに、プール混合関数がプール全体に適用されます。

プール混合関数の説明は以下のとおり:

1. R を乱数プールとする。
2. H をユーザーが選択したハッシュ関数(RIPEMD-160, SHA-1, または Whirlpool)とする。
3. l = ハッシュ関数 H の出力のバイト長。(H が SHA-1 か RIPEMD-160 なら、$l = 20$; H が Whirlpool なら $l = 64$)
4. z = ランダムプール R のバイト長 (320 バイト)
5. $q = z / l - 1$ (H が Whirlpool なら $q = 4$)
6. R を l-バイトブロック $B_0...B_q$ に分割
   条件 $0 \le i \le q$ (各ブロック B ごとに) であるあいだ、以下のステップを実行:
   a. $M = H(B_0 \| B_1 \| .. \| B_q)$ [ランダムプールはハッシュ M を作るハッシュ関数 H で処理される]
   b. $B_i = B_i \wedge M$
7. $R = B_0 \| B_1 \| .. \| B_q$

たとえば、$q = 1$ ならば、ランダムプールは次のように混合される:
$(B_0 \| B_1) = R$

$B_0 = B_0 \text{ ^ } H(B_0 \parallel B_1)$
$B_1 = B_1 \text{ ^ } H(B_0 \parallel B_1)$
$R = B_0 \parallel B_1$

乱数発生機構の設計と実装は下記の論文に基づく:

- *Software Generation of Practically Strong Random Numbers* by Peter Gutmann [10]
- *Cryptographic Random Numbers* by Carl Ellison [11]

キーファイル

TrueCrypt キーファイルは、その内容がパスワードと結びつけられ混合されるファイルです。キーファイルの内容について、特別の制限はありません。ユーザーは TrueCrypt RNG によってランダムな内容のファイルを生成する組み込みのキーファイル生成機能を使って、キーファイルを生成することもできます。(TrueCrypt RNG についての詳細は「乱数発生機構」を参照) キーファイルの最大サイズに制限はありませんが、先頭の 1,048,576 bytes (1 MB)だけが処理対象となります。(巨大なファイルを処理するのに伴う性能上の問題から、残りの部分は無視されます) ユーザーは複数のキーファイルを使うことができます。(キーファイル数に制限はありません)

キーファイルは以下の方法で処理され、パスワードに適用されます。

1. *P* をユーザーが入力したパスワード(空かもしれません)とする。
2. *KP* をキーファイルプールとする。
3. *kpl* をキーファイルプール *KP* のバイト長(64 つまり 512 ビット)とする。
4. *pl* をパスワード *P* のバイト長(現バージョンでは $0 \le pl \le 64$)とする。
5. $kpl > pl$ ならば($kpl - pl$)の長さのバイト(値はゼロ)をパスワード *P* *に追加する。*
6. キーファイルプール *KP* を *kpl* バイトのゼロで満たす。
7. それぞれのキーファイルについて、以下のステップを実行:
    a. キーファイルプールのカーソル位置をプールの先頭にセットする。
    b. ハッシュ関数 *H* を初期化する。
    c. キーファイルの全バイトを 1 個づつロード、それぞれについて以下のステップを実行する。
        i. 中間ハッシュ(状態) *M* を得るために、ハッシュを初期化せずにハッシュ関数 *H* でロードされたバイトのハッシュを作る。ハッシュの終了処理はしない(次回のために状態を保持する)。
        ii. 状態 *M* を個々のバイトに分割する。例として、ハッシュの出力が 4 バイトなら $(T_0 \| T_1 \| T_2 \| T_3) = M$
        iii. (7.c.ii で得られた)これらのバイトを個々に modulo 2^8 addition 演算をしてキーファイルプールの(プールの古い値の上書きではなく)プールカーソルの位置に書き込む。バイトが書き込まれたらプールカーソルは 1 バイト進む。カーソルがプールの終端までくると、位置はプールの先頭に設定される。
8. キーファイルプールの内容を以下の方法でパスワード *P* に適用する。
    a. パスワード *P* を個々のバイト $B_0 ... B_{pl}$ に分割する。
    b. キーファイルプール *KP* を個々のバイト $G_0 ... G_{kpl}$ に分割する。
    c. For $0 \le i \le kpl$ の条件で順に実行 $B_i = B_i \oplus G_i$
    d. $P = B_0 \| B_1 \| ... \| B_{pl-1} \| B_{pl}$
9. パスワード *P* は(キーファイルプールの内容が適用されたあと)ヘッダーキー導出関数 PBKDF2 (PKCS #5 v2)へ渡され、それがユーザーが選択した安全なハッシュアルゴリズム (RIPEMD-160 か Whirlpool)の暗号を使って(ソルトや他のデータとともに)処理します。詳細は「ヘッダーキーの導出、ソルト、および反復回数」を参照してください。

関数 *H* の役割はたんに拡散が目的です[26]。CRC-32 はハッシュ関数 *H* で使われます。CRC-32 の出力はつづけて安全なハッシュアルゴリズムの暗号で処理されます。キーファイルプールの内容は(CRC-32 でハッシュされたのに加え)、パスワードに適用されます。それがヘッダーキー導出関数 PBKDF2 (PKCS #5 v2)へ渡され、それがユーザーが選択した安全なハッシュアルゴリズム(RIPEMD-160 か Whirlpool)の暗号を使って(ソルトや他のデータとともに)処理します。結果として得られる値がヘッダーキーと第二ヘッダーキー(LRW モード)として使われます。

TrueCrypt ボリュームフォーマット仕様

ファイル型ボリュームのフォーマットはパーティション/デバイス型ボリュームと同じです。TrueCrypt ボリュームには署名や ID 文字列のようなものはありません。復号されるまでは、すべてがランダムなデータにしか見えません。したがって、TrueCrypt コンテナやパーティションであるかどうかを判断することはできません。

それぞれの TrueCrypt ボリュームの空き領域はボリュームが作られるときに(オプションのクイックフォーマットとダイナミックが無効になっていれば) ランダム値で満たされます。 ランダム値は以下のように生成されます: TrueCrypt ボリュームのフォーマットが始まる直前に臨時の暗号化キーと臨時の第二キー(LRW モード)が組み込みの乱数発生機構(「乱数発生機構」参照)で生成されます。ユーザーが選んだ暗号化アルゴリズムは臨時キーで初期化されます。つづいて組み込みの乱数発生機構で生成されたプレーンテキストを暗号化します。暗号化アルゴリズムは LRW モードで動きます。(「動作モード」参照) それが作り出した暗号テキストブロックがボリュームの空き領域を満たす(上書きする)のに使われます。キーは RAM 中に保管され、フォーマットが終了すると安全に廃棄されます。

TrueCrypt ボリュームフォーマット 仕様:

オフセット (bytes)	サイズ[1] (bytes)	暗号化	備考
0	64	非暗号化	ソルト[2]
64	4	暗号化	ASCII 文字列 “TRUE”
68	2	暗号化	ボリュームヘッダーフォーマットバージョン
70	2	暗号化	ボリュームを開く最小プログラムバージョン
72	4	暗号化	(復号された) 256-511 バイトの CRC-32 チェックサム
76	8	暗号化	ボリューム作成日時
84	8	暗号化	ヘッダー作成/変更日時
92	8	暗号化	予約(0 をセット)
100	156	暗号化	未使用
256	*Var.*	暗号化	第二キー(LRW モード)
288	*Var.*	暗号化	マスター暗号化キー[3]
512	*Var.*	暗号化	データ領域(実際のボリュームの内容)

byte #0(ソルト)、byte #256(第二キー) 、byte #288(マスター暗号化キー)のフィールドはボリューム生成過程の間、乱数発生機構(「乱数発生機構」参照)で生成された乱数が入れられます。

TrueCrypt ボリュームの空き領域に隠しボリュームがある場合には、隠しボリュームのヘッダーはホストボリュームの最後から 1536 バイトの位置にあります。(ホスト/外殻ボリュームのヘッダーはボリュームの先頭にあります – 「隠しボリューム」参照)隠しボリュームのヘッダーのフォーマットについては、次の表で説明します。

オフセット (bytes)	サイズ (bytes)	暗号化状態	備考
0	64	非暗号化	ソルト
64	4	暗号化	ASCII 文字列 “TRUE”
68	2	暗号化	ボリュームヘッダーフォーマットバージョン
70	2	暗号化	ボリュームを開く最小プログラムバージョン
72	4	暗号化	(復号された) 256-511 バイトの CRC-32 チェックサム
76	8	暗号化	ボリューム作成日時
84	8	暗号化	ヘッダー作成/変更日時
92	8	暗号化	隠しボリュームのサイズ
100	156	暗号化	現在未使用
256	*Var.*	暗号化	第二キー(LRW モード)
288	*Var.*	暗号化	マスター暗号化キー

TrueCrypt ボリュームヘッダーはつねに 512 バイトです。隠しボリュームのヘッダーも 512 バイトです。

TrueCrypt がサポートする最大ボリュームサイズは 8,589,934,592 GB (2^{63} bytes)です。

[1]ボリュームヘッダーの暗号化領域はヘッダーキー(および LRW モードでの第二キー)で暗号化されます。詳細は「暗号化の仕組み」と「ヘッダーキーの導出、ソルト、および反復回数」を参照してください。
[2] ソルトは暗号化する必要がありません。秘密にする必要がないからです。[7](ソルトは一連のランダム値です)
[3]ボリュームが暗号のカスケードで暗号化されている場合には、マスターキーは複数になります。

準拠規格

TrueCrypt は以下の規格、仕様、勧告に準拠しています：

- PKCS #5 v2.0 [7]
- FIPS 197 [3]
- FIPS 198 [22]
- FIPS 180-2 [14]
- ISO/IEC 10118-3:2004 [21]

実装された暗号化アルゴリズムの正確さは、テストベクターを使う(*ツール->テストベクターをクリック*)か TrueCrypt のソースコードを調べることで検証できます。

ソースコード

TrueCrypt はオープンソースのフリーソフトウェアです。TrueCrypt の完全なソースコード(C および C++で書かれています)はみなさんのレビューのため次のところで自由に入手できます:

http://www.truecrypt.org/downloads.php

今後の開発予定

将来の計画に含まれている機能については以下を参照してください:
http://www.truecrypt.org/future.php

ライセンス

TrueCrypt の公開についてのライセンスは TrueCrypt バイナリあるいはソースコードの配布パッケージに含まれる License.txt に記載されています。また、次のところでも入手できます:
http://www.truecrypt.org/license.php

連絡先

われわれへの連絡方法については、次のところを参照してください:
http://www.truecrypt.org/contact.php

バージョン履歴

4.3a
2007 年 5 月 3 日

機能改善:

- ディスプレーの DPI 設定に対応(Windows の GUI)
- その他の小さい改善(Windows および Linux)

バグ修正:

- ある条件で「ハードウェアの安全な取り外し」が失敗するバグの修正
- Windows Vista で UDF フォーマットのメディアにあるファイル型 TrueCryprt ボリュームが読み出し専用でマウントされた婆場合でもデータを読めるようにした。
- ボリューム作成ウィザードの各項目が画面の DPI 設定に従って正しく表示される。(Windows の GUI)
- その他の小さいバグ修正(Windows と Linux)

セキュリティ改善:

- Linux: 管理者権限がない状態で起動すると、TrueCrypt は必要であれば自動的に sudo コマンドで権限を昇格させる。TrueCrypt の Linux 版では実行 uid(euid)設定による root モードの実行をサポートしないこととした。これは実行 uid(euid)設定による root モード実行に関連する既知の(あるいは既知ではない)問題の修正でもある。これは TrueCrypt の Linux 版のすべての旧バージョンに関係する、管理者権限を持たないユーザーが管理者権限を得たり管理者権限でのサービスを拒否されるという問題であった。

その他:

- TrueCrypt がトラベラーモードで動いているときに、TrueCrypt ボリュームが強制的にアンマウントされると「終了」しても TrueCrypt ドライバーは除去されない。(システムを停止するかリスタートする場合のみ、除去される) これは Windows のバグによって引き起こされるいろいろな問題(たとえば、アンマウントされたボリュームを使っているアプリケーションがあると TrueCrypt を再起動できない)を防止する。

旧バージョンでの変更履歴は http://www.truecrypt.org/docs/?s=version-history を参照してください。

謝辞

私たちは以下のみなさんに感謝します:

Paul Le Roux は彼の E4M ソースコードを入手できるようにしてくれました； TrueCrypt は E4M に基づいています。

Dr. Brian Gladman, 彼はすばらしい AES, Twofish, SHA-1 そして多様な有限体 GF(2^{128}) ルーチンを書いてくれました。

Peter Gutmann, 彼の乱数についての論文と、TrueCrypt の乱数発生機構の一部のソースである cryptlib を作ってくれたことに。

Wei Dai は Serpent ルーチンを書いてくれました。*Dag Arne Osvik* には「*Serpent の高速化*」論文について。

Markus Friedl は RIPEMD-160 ルーチン(OpenBSD より)を書いてくれました。

暗号化とハッシュ・アルゴリズムの設計者のみなさん:
Horst Feistel, Don Coppersmith, Walt Tuchmann, Lars Knudsen, Ross Anderson, Eli Biham, Bruce Schneier, David Wagner, John Kelsey, Niels Ferguson, Doug Whiting, Chris Hall, Joan Daemen, Vincent Rijmen, Carlisle Adams, Stafford Tavares, Hans Dobbertin, Antoon Bosselaers, Bart Preneel, Paulo S. L. M. Barreto.

このプロジェクトを可能にしてくれたみなさん、精神的に支援してくれたみなさん、バグレポートや改善提案を送ってくれたみなさん

ありがとうございました。

参考文献

[1] U.S. Committee on National Security Systems (CNSS), *National Policy on the Use of the Advanced Encryption Standard (AES) to Protect National Security Systems and National Security Information*, CNSS Policy No. 15, Fact Sheet No. 1, June 2003, available at http://www.cnss.gov/Assets/pdf/cnssp_15_fs.pdf and also at http://csrc.nist.gov/cryptval/CNSS15FS.pdf.

[2] C. E. Shannon, *Communication Theory of Secrecy Systems*, Bell System Technical Journal, v. 28, n. 4, 1949

[3] NIST, *Advanced Encryption Standard (AES)*, Federal Information Processing Standards Publication 197, November 26, 2001, available at http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

[4] J. Nechvatal, E. Barker, L. Bassham, W. Burr, M. Dworkin, J. Foti, E. Roback, NIST, *Report on the Development of the Advanced Encryption Standard (AES)*, October 2, 2000, available at http://csrc.nist.gov/CryptoToolkit/aes/round2/r2report.pdf.

[5] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, N. Ferguson, T. Kohno, M. Stay, *The Twofish Team's Final Comments on AES Selection,* May 15, 2000, available at http://csrc.nist.gov/CryptoToolkit/aes/round2/comments/20000515-bschneier.pdf.

[6] M. Bellare, *New Proofs for NMAC and HMAC: Security without Collision-Resistance*, Cryptology ePrint Archive: Report 2006/043, February 6, 2006, available at http://eprint.iacr.org/2006/043

[7] RSA Laboratories, *PKCS #5 v2.0: Password-Based Cryptography Standard*, RSA Data Security, Inc. Public-Key Cryptography Standards (PKCS), March 25, 1999, available at ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-5v2/pkcs5v2-0.pdf
and also courtesy of RSA Laboratories at: http://www.truecrypt.org/docs/pkcs5v2-0.pdf

[8] H. Krawczyk, M. Bellare, R. Canetti, *HMAC: Keyed-Hashing for Message Authentication*, Request for Comments 2104, February 1997, available at http://www.ietf.org/rfc/rfc2104.txt.

[9] P. Cheng, IBM, R. Glenn, NIST, *Test Cases for HMAC-MD5 and HMAC-SHA-1*, Request for Comments 2202, February 1997, available at http://www.ietf.org/rfc/rfc2202.txt.

[10] Peter Gutmann, *Software Generation of Practically Strong Random Numbers*, presented at the 1998 Usenix Security Symposium, available at http://www.cs.auckland.ac.nz/~pgut001/pubs/usenix98.pdf.

[11] Carl Ellison, *Cryptographic Random Numbers*, originally an appendix to the P1363 standard, available at http://world.std.com/~cme/P1363/ranno.html.

[12] M. Liskov, R. Rivest, D. Wagner, *Tweakable Block Ciphers*, Advances in Cryptology – CRYPTO '02, vol. 2442 of Lecture Notes in Computer Science, pp. 31-46. Springer-Verlag, 2002; also available at: http://theory.lcs.mit.edu/~rivest/LiskovRivestWagner-TweakableBlockCiphers.pdf

[13] J. Kelsey, *Twofish Technical Report #7: Key Separation in Twofish*, AES Round 2 public comment, April 7, 2000

[14] NIST, *Secure Hash Standard*, August 1, 2002, available at http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf.

[15] U. Maurer, J. Massey, *Cascade Ciphers: The Importance of Being First*, Journal of Cryptology, v. 6, n. 1, 1993

[16] Bruce Schneier, *Applied Cryptography*, Second Edition, John Wiley & Sons, 1996

[17] Peter Gutmann, *Secure Deletion of Data from Magnetic and Solid-State Memory*, first published in the Sixth USENIX Security Symposium Proceedings, San Jose, California, July 22-25, 1996, available at http://www.cs.auckland.ac.nz/~pgut001/pubs/secure_del.html

[18] Serpent home page: http://www.cl.cam.ac.uk/~rja14/serpent.html.

[19] M. E. Smid, *AES Issues*, AES Round 2 Comments, May 22, 2000, available at http://csrc.nist.gov/CryptoToolkit/aes/round2/comments/20000523-msmid-2.pdf.

[20] A. Menezes, P. van Oorschot, S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, October 1996

[21] International Organization for Standardization (ISO), *Information technology – Security techniques – Hash-functions – Part 3: Dedicated hash-functions*, ISO/IEC 10118-3:2004, February 24, 2004

[22] NIST, *The Keyed-Hash Message Authentication Code (HMAC)*, Federal Information Processing Standards Publication 198, March 6, 2002, available at http://csrc.nist.gov/publications/fips/fips198/fips-198a.pdf.