学術情報処理研究 No.21
2017年9月

暗号化ラッパプログラムの開発

Development of Wrapper Program for Encryption

松澤 英之
Hideyuki Matsuzawa

matuzawa@cc.miyazaki-u.ac.jp

宮崎大学 情報基盤センター

Information Technology Center, University of Miyazaki

概要

昨今情報漏えい対策は重要である。宮崎大学でも個人情報の漏えい対策として、個人情報が書き込まれているファイル・フォルダをパスワードで保護する事を求めている。具体的にはファイル・フォルダをパスワード付 ZIP で暗号化して保存する事を推奨している。本研究では、ユーザが常に暗号化されているパスワード付 ZIP ファイルを簡単に参照・編集できるラッパプログラムを開発した。

キーワード

暗号化・復号化, ラッパプログラム,開発

1. はじめに

独立行政法人 情報処理推進機構(IPA)が作成した情報セキュリティ啓発のための対策のしおり「初めての情報セキュリティ対策のしおり(第 1 版)」[1]では、新入社員に対する最初のセキュリティ対策として個人情報及び企業情報の漏洩対策を挙げている。また NPO 日本ネットワークセキュリティ協会セキュリティ被害調査ワーキンググループが作成した 2015 年情報セキュリティインシデントに関する調査報告書【速報版】Ver. 1.0[2]から、2006 年度の個人情報漏えいの原因の約半数は紛失・置き忘れ、盗難であることがわかる。IPA が作成した「暗号化による<情報漏えい>対策のしおり(第 1 版)」[3]では、紛失・置き忘れ、盗難の対策の例として情報の暗号化を挙げている。

宮崎大学では「教員の個人情報ファイルの取り扱い方針」で個人情報を含むファイル・フォルダに対してパスワードで保護する事を求めている。しかし Microsoft Office ファイルなどを除いてファイル・フォルダ自体にパスワードを付ける機能はない。そこで宮崎大学ではファイル・フォルダをパスワード付 ZIP で圧縮する方法を推奨している。パスワード付 ZIP ファイル作成時には暗号化も行われるので、宮崎大学の方針によれば個人情報を含むファイル・フォルダは”常に”暗号化して保存する事が推奨されていることになる。暗号化されているファイル・フォルダをユーザが参照・編集する時には、どのような工程を経る必要があるだろうか。

1. 暗号化されたファイル・フォルダを暗号化ソフトで復号化する。
2. 復号化されたファイルを参照・編集する。
3. 復号化したファイル・フォルダを暗号化ソフトで再度暗号化する。再暗号化の際に暗号化ソフトが

暗号化前のファイル・フォルダを削除しないものもある。この場合秘密保持のためには、ファイル・フォルダを暗号化したのち暗号化前のファイル・フォルダを手動で削除する必要がある。

暗号化ソフトには自動でこれら 3 つの工程を行うものもある[4][5]。

ではパソコンユーザは宮崎大学の方針を遵守するためにどうすればよいのだろうか？現在宮崎大学ではパスワード付 ZIP ファイルを自動的に復号・暗号化する便利な暗号化ソフトは導入されていない。更に Mac OS X ではデフォルトでパスワード付 ZIP ファイルの暗号化・復号化を行えるが、Windows クライアント OS ではデフォルトでパスワード付 ZIP ファイルを作成する機能はない。つまり、Windows ユーザはまずパスワード付 ZIP ファイルを作成するソフトウェアをインストールする必要がある。

次に自動的に復号化して参照・編集できる暗号化ソフトを導入している人以外は、ファイルを参照・編集するたびに上記 3 工程を手作業で行わなければならない。暗号化したファイルを参照・編集するたびに行われるこれらの作業は、パソコンユーザにとっては非常に面倒な作業である。2015 年に発生した年金機構における不正アクセスによる情報流出事案の検証報告書[6]では、原則として厳格なルールを定めていてもルールが遵守されなければ情報が漏洩する事に言及している。同様にこのファイル・フォルダの暗号・復号化、参照・編集工程が面倒な作業であると思えばそれを回避する方向、例えばファイルを暗号化しないで保存するようになることが考えられる。

そこで、この面倒な 3 工程をなるべく簡単にし、パソコンユーザに個人情報を含んだファイル・フォルダをパスワード付 ZIP で保存・利用してもらうためにパスワード付 ZIP ファイルに対する Windows と Mac OS X 向け暗号化ラッパプログラム(CryptoWrapper)を開発した。

2. データ転送時の暗号化と暗号化ラッパプログラム

一般ユーザが”暗号化”というキーワードで思い出す、あるいは一般ユーザに暗号化を推奨している場面は、メールに重要なファイルを添付する場合にパスワード付 ZIP で暗号化して添付することだと思われる[7]。このパスワード付 ZIP ファイルを添付する場面と今回開発した暗号化ラッパプログラムの違いについて考察する。メールにパスワード付 ZIP で暗号化したファイルが添付されている場合、添付ファイルを一度復号化した後に再度暗号化することはまずありえない。添付ファイルの暗号化はメールが送信者から受信者に送られる間だけ暗号化で添付ファイルを守るためである。一方暗号化ラッパプログラムは常時暗号化で守られているファイル、フォルダを如何に簡単に閲覧・編集し、閲覧・編集後に再暗号化をするかを目指している。つまり暗号化ラッパプログラムは暗号化で守られているファイル、フォルダを簡単に活用することを目指している。私見であるが普通のパソコンユーザに暗号化が利用されない理由は暗号化に関する用語、解説が難しいことに加え、暗号化されたファイルを利用する際の工程が多くて面倒くさいことであると考える。この暗号化ラッパプログラムは暗号化の利用を簡便にする。更に有料暗号化ソフトで用いられる独自の暗号化と比べ暗号化方式が汎用的な暗号化を用いることで他の OS 間でもデータ交換を容易にできる。この暗号化ラッパファイルを利用する想定としては学生の実習先に個人情報を含んだファイルを持ち込んで活用する場合などが想定される。

3. 暗号化方法

宮崎大学の方針を遵守するためにラッパプログラムの暗号化方法としてパスワード付 ZIP を用いた。パスワード付 ZIP ファイルを暗号化方法として採用する利点は、

1. Windows クライアント OS ではデフォルトでパスワード付 ZIP の復号化が行える
2. Mac OS X ではデフォルトでパスワード付 ZIP ファイルの暗号化・復号化が行える
3. IPA「電子メール利用時の危険対策のしおり(第 4 版)」[8]で、電子メールにファイルを添付する際にはファイルをパスワード付 ZIP で暗号化してから添付する事を紹介しているなどパスワード付 ZIP ファイルの作成はパソコンユーザに極端な負荷を与えるものではないと考える。

一方、パスワード付 ZIP ファイルに対しては暗号化の解読のためのフリーソフトが出回っているなどセキュリティ強度の点で問題があるが、今回はあくまで宮崎大学の方針を遵守する為セキュリティ強度の問題は目をつぶってパスワード付 ZIP を用いた。

以上の理由で本研究では暗号化方式にパスワード付 ZIP を用いるが他の暗号方式、有料の暗号化ソフトについても比較しておく。

パスワード付 ZIP のようにファイル、フォルダを暗号化するのではなく、Windows EFS、Linux の暗号化ファイルシステムなど OS レベルでファイル、フォルダを暗号化する方法がある。ファイル・フォルダは自動的に暗号化されるのでユーザは一切暗号化・復号化を行う必要がない。OS レベルの暗号化は初心者にも非常にやさしい

暗号化方式になっている。しかしユーザ権限を乗っ取られた場合 OS レベルではどのユーザが乗っ取られたか判別できないので情報漏洩を防げない。

Windows RMS などDRM メカニズムは著作権保護などにもちられている。DRM は主にコンテンツの再生を制御するためのもので、最終成果物へのアクセス制御には有効である。しかしパスワード付 ZIP のように、暗号化・復号化を簡単に行い、共同でコンテンツを作成する場合などには向いていない。また DRM メカニズムの多くは OS、再生ソフトウェアに依存している。異なる OS 間でのファイル交換には向かない。

Microsoft Office でのパスワード保護、PDF に対するパスワード保護など個々のソフトウェアに依存したパスワードによる保護機能がある。現在保護が必要なファイルのほとんどは Office で作成されるが、すべての個人情報を含んだファイルを保護するという観点から考えると不十分である。例えば写真データにパスワードを付与できる、かつ OS に依存せず広く普及したソフト、フォーマットは存在しない。

最後に有料の暗号化ソフトについて考える。資金に余裕がある場合有料暗号化ソフト導入が一番簡単なように思われる。しかし、ファイル暗号化の必要性を感じていないユーザがわざわざお金を払って暗号化ソフトを導入するとは思わない。また、有料暗号化ソフトは独自の暗号化を行うので、暗号化したファイルを他人に渡す場合、あらかじめ暗号化を解いてファイルを渡すか、だれでも利用できる暗号化方式に変更してファイルを渡す、あるいは暗号化を解除するためのソフトと方法を譲渡先に紹介する必要がある。暗号化していない状態でファイルを渡すことは暗号化を用いてファイルを保護している趣旨に反する。別方式の暗号化を行うことは手間である。また組織的にマニュアルが存在する場合を除いて、一般のパソコンユーザには解除ソフトと解除方法を他人に説明することは非常に高い敷居だと考える。

4. 暗号化ソフト

Windows クライアント OS にはデフォルトでパスワード ZIP ファイルを作成する機能はない。そこでラッパプログラムでの復号・暗号化、或いはファイルの参照・編集は、ラッパプログラムから復号・暗号化用外部ソフトとファイル参照・編集用の外部ソフトをそれぞれ起動して行う(ここでいう外部ソフトとはラッパプログラム以外にパソコンにインストールされているソフトを指す)。これは以下の利点がある。

1. 車輪の再発明を防ぐ。
2. ユーザがファイルを参照・編集する時、暗号化されていないファイルを参照・編集する時と同様に操作できる。
3. ラッパプログラムからファイルの参照・編集を行う外部ソフトを起動する場合、ファイルの拡張子とソフトウェアの関連付けを利用して起動するので、パソコンで利用できる様々な形式のファイルを参照・編集する事が出来る。
4. ラッパプログラムで手動或いは自動的に暗号化ソフトを選択できるようにすれば、ラッパプログラムが将来的に様々な暗号化形式に対応できる。

今回は、Windows クライアント OS では GUI、CUI で利用可能でパスワード付 ZIP ファイルを作成できる 7-Zip[9]を暗号化ソフトとして利用した。

一方、Mac OS X ではデフォルトでパスワード付 ZIP ファイルを暗号化・復号化できるコマンド(zip,unzip)がある。しかし、Windows で作成され、日本語ファイル名が付いたファイルを含んだパスワード付 ZIP ファイルを Mac OS X のデフォルトのコマンドを用いて復号化するとファイル名が文字化けする。そこで、Mac OS X でパスワード付 ZIP ファイルを復号化するために unar[10]を用いた。デフォルトのコマンドで暗号化した場合、日本語ファイル名が付いたファイルを含んだパスワード付 ZIP ファイルでも Windows で正常に復号化できるので、暗号化にはデフォルトコマンドを用いる。

5. ラッパプログラムのプロセス

今回開発したラッパプログラムは Windows クライアント OS と Mac OS X で利用できることを目指している。Windows と Mac で別々に開発を行うのは非効率なので、開発用のプログラムとして Window と Mac で共通して使える Python を選んだ。

ラッパプログラムの基本コンセプトは

1. 暗号化・復号化、参照・編集する前述の 3 工程は外部ソフトに任せる。
2. ラッパプログラムは外部ソフト同士が円滑に働くように補助する。

とする。

ラッパプログラムの具体的なプロセスは以下の通りである(図-1)。ユーザの操作、あるいはユーザへの表示についてはかっこ( )内に表示する。

1. 復号化したいパスワード付 ZIP ファイル名を引数としてラッパプログラムを起動する。(ユーザ操作=復号化したいファイルをラッパプログラムのアイコンにドロップする)
2. ラッパプログラムは外部暗号化ソフトにパスワード付 ZIP ファイル名を引数として渡し、戻り値としてパスワード付 ZIP ファイルに含まれるファイル・フォルダのリストを取得する。

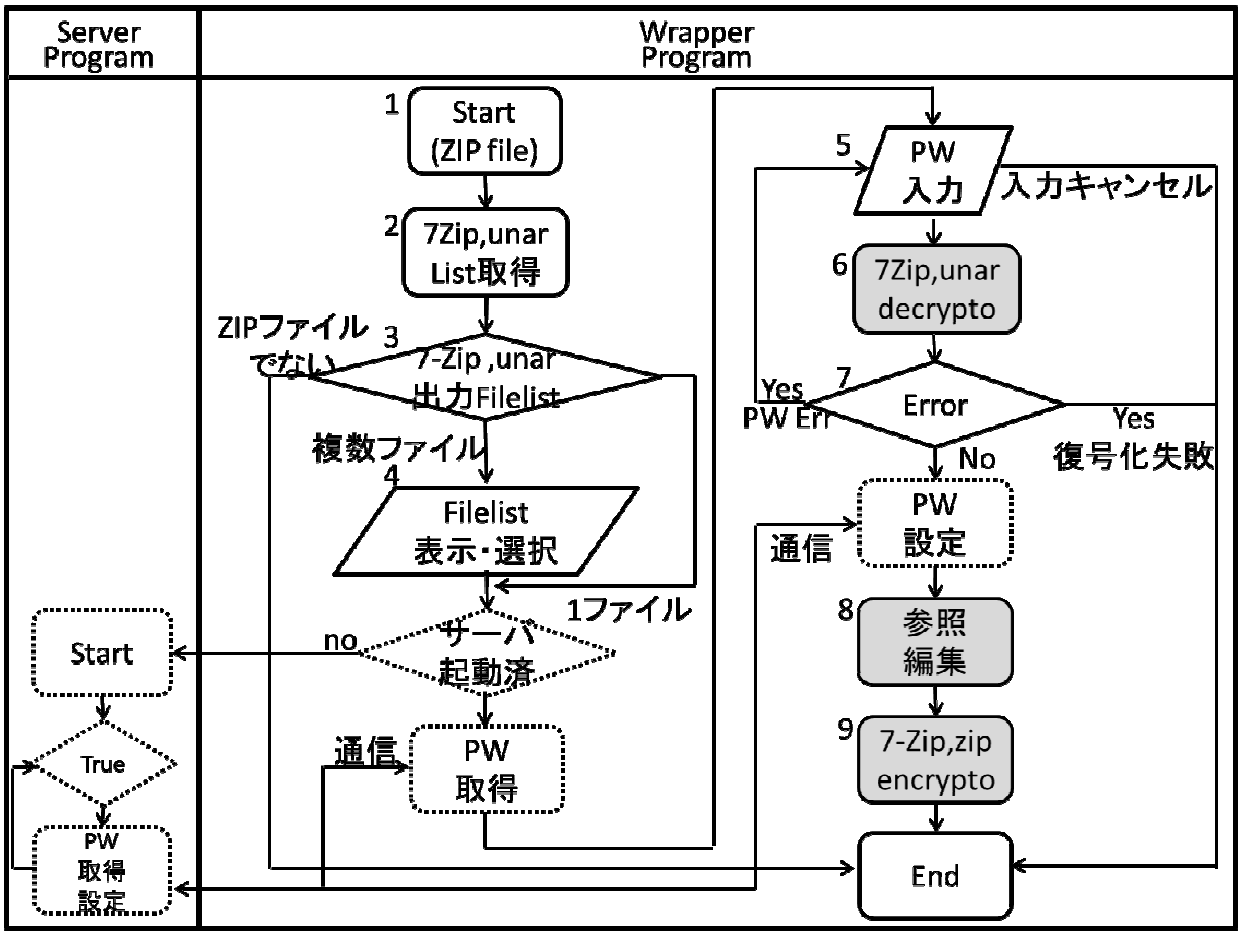

図-1

3. ラッパプログラムは外部暗号化ソフトからの戻り値であるファイル・フォルダのリストについて判断する。ラッパプログラム起動時に引数で指定したファイルが ZIP で暗号化されていない場合、ラッパプログラムは終了する。パスワード付 ZIP ファイルに含まれるファイルが一つの場合は、パスワード入力画面(プロセス 5)に進む。それ以外はプロセス 4 に進む。
4. ラッパプログラムは取得したファイルリスト(フォルダは除く)を表示し、ユーザに参照・編集するファイルを選択させる。(ユーザ画面=ユーザにパスワード付き ZIP ファイル内にあるファイルリストが表示される。ユーザ操作=ユーザは参照・編集したいファイルを選択する)
5. パスワード付 ZIP ファイルを復号化するためのパスワード入力画面を表示する。入力されたパスワードは伏字として表示される。パスワードの入力がキャンセルされた場合は、ラッパプログラムを終了する。ユーザが入力する以外の方法でパスワードを取得する方法については改めて言及する。(ユーザ画面=パスワード入力を促す画面が表示される。ユーザ操作=パスワードを入力する)
6. ラッパプログラムはユーザが入力したパスワード、パスワード付 ZIP ファイル名を引数として外部暗号化ソフトに渡し、外部暗号化ソフトで復号化を行う。復号化したファイル・フォルダはパスワード付 ZIP ファイルと同じフォルダに展開される。ラッパプログラムは復号化した際に外部暗号化ソフトが表示するメッセージを取得する。
7. 外部暗号化ソフトから取得したメッセージで外部暗号化ソフトが正常に終了したか判断する。パスワードが間違っていた場合は、パスワード入力画面(プロセス 5)に戻る。復号化に失敗した場合は、ラッパプログラムを終了する。
8. プロセス 4 で選択したファイルに対応した参照・編集するソフトウェアを別スレッドとして起動する。ラッパプログラムは参照・編集するソフトウェアが完全に終了するのを監視する。Windows ではコマンドプロンプトでファイル名を入力するだけで、GUI でファイルアイコンをダブルクリックした場合と同じようにファイル形式に対応したソフトウェアを起動する事ができるので、ラッパプログラムは Windows のシェル(コマンドプロンプト)に'start "画面表示名" 選択されたファイル名'を渡す。"start"コマンドは現在利用しているコマンドプロンプトとは別のコマンドプロンプトを起動するためのコマンドである。一方、Mac OS X の場合、’open’コマンドを

使う。このコマンドは Windows と同じように拡張子を参照して対応したプログラムを起動する。参照・編集するソフトウェアの起動に失敗した場合でも、既に復号化されているファイル・フォルダを再度暗号化する必要があるのでラッパプログラムのプロセスは続行する。(ユーザ画面=参照・編集するファイルに対応したソフトウェアが起動する)

9. ラッパプログラムはすべての参照・編集するソフトウェアが終了したことを確認後、外部暗号化ソフトに引数としてパスワード、ZIP ファイル名、暗号化するファイル・フォルダ一覧を渡してファイル・フォルダの暗号化を行う。暗号化ソフトとして使用した Windows 用 7-Zip、Mac OS X 用デフォルト zip コマンドは暗号化時に、暗号化前のファイル・フォルダを削除する機能があるので、この削除機能を利用して復号化したファイル・フォルダを削除する。(ユーザ操作=参照・編集するファイルに対応した全てのソフトウェアを終了させる)

6. パスワード一時保存サーバ

6.1. パスワード一時保存サーバ

上で記述したプロセスでは、ラッパプログラムを起動するたびに、ユーザはファイルを復号・暗号化するためのパスワードを入力する必要がある。ファイル・フォルダを参照・編集するたびにユーザがパスワードを入力するのは大変なので、パスワードを一時保存するサーバを作成した。ラッパプログラムがこのサーバと通信する事でパソコン起動後最初にラッパプログラムを利用する時または、前に入力したパスワードと異なる場合を除いてパスワードの入力を省略できる。

パスワードを共有する方法として、サーバではなく一時ファイルにパスワードを保存・共有する方法も検討した。しかしファイルにパスワードを保存した場合、パソコンが異常終了した時でもパスワードを保存したファイルを完全に削除する方法を思いつけなかった。つまりパソコンが異常終了した場合は、パソコン内にパスワードを保存したファイルが残されてしまう恐れがある。しかしサーバプログラムの場合は、メモリー上にパスワードが保存されるので、パソコンが異常終了した場合でもパスワードがパソコン上に残されることはないと判断した。

サーバプログラムはユーザが起動する複数のラッパプログラムからアクセスされるので、ラッパプログラムとサーバ間の通信には TCP/IP を用いた。このサーバは重要なファイルを復号化するパスワードを保存しているので、特にセキュリティには気を付ける必要がある。そこでパスワードを保持しているサーバが起動している時間を短くするように努めた。そのため、Windows ではパソコンが起動している間は常にプログラムが働くサービスとしては登録せずに、ラッパプログラムと同じように起動しているときはコマンドプロンプトが表示されるコンソールプログラムとした。Windows のコンソールプログラムの場合、サーバプログラムが動いているコマンドプロンプト画面を閉じることで簡単にサーバプログラムを終了できる。また Mac の場合は、パソコン起動時に自動的に立ち上がるデーモンプロセスとして登録していない。

6.2. プロセス

サーバを起動する場合は、ラッパプログラムのプロセス 4 終了後に行う。まずパソコンでサーバプログラムが起動しているかどうかラッパプログラムが確認する。サーバが起動していない場合は、ラッパプログラムからサーバプログラムを起動する。サーバプログラムが既に起動していた場合は、ラッパプログラムはサーバと通信して保存されているパスワードを取得、プロセス 5 でパスワード入力画面に伏字としてパスワードを表示する。ラッパプログラムはプロセス 7 でパスワードが正しいと確定した後にサーバへパスワードの保存を行う。

サーバを利用するかどうかは設定ファイルで指定できるようになっている。デフォルトはサーバを利用できない。この設定については後程議論する。

6.3. プロトコル

サーバの送受信プロトコルは HTTP プロトコルを参照した。サーバのプロトコルは HTTP プロトコルと同じステートレスである。サーバはクライアント(ラッパプログラム)からのアクセス要求を受け、パスワードが保存されている場合は保存してあるパスワードを、パスワードが保存されていない場合は空文字をクライアントに送信してクライアントの接続を終了する。

クライアントからの送信データは"GET:”と"POST:パスワード"の 2 種類。送信データが"POST:パスワード"の場合は、まずサーバに保存されているパスワードをクライアントから送られたパスワードで更新・保存する。そのあとどちらの場合もサーバに保存されているパスワードをラッパプログラムに送信する。

6.4. セキュリティ

ラッパプログラムからサーバプログラムへのアクセスは同一パソコン内に限定している。TCP/IP で通信しているので、サーバはパソコンに登録されている全ユーザか

らアクセス可能である。一般にアクセス制御等を行うサーバの設定はサーバが起動しているコンピュータに保存されている。サーバとクライアントが別のパソコンで動いている場合は、アクセス制御のためのパスワード等サーバの設定をクライアントから見る或いは変更することはできない。しかしサーバとクライアントが同一のパソコンで動いている場合、管理者権限等を用いてサーバの設定を閲覧・変更する事ができる。例えば公開鍵暗号方式を用いてパスワードを保護した場合、秘密鍵を盗まれないためにサーバとは別の機器である USB メモリーなどに秘密鍵を保存し、秘密鍵が必要な時だけ USB メモリーをサーバが起動するパソコンに接続させる必要がある。今回はなるべく簡単な操作を目指しているので、この方式は採用しなかった。そこでパスワードを保護するために管理者権限を所有しているユーザが一人だけの場合にサーバプログラムを起動できるように、ラッパプログラムの設定でサーバプログラムの起動を選択できるようにした。デフォルトの設定ではサーバプログラムは起動しない。

7. ラッパプログラムの起動

Windows でファイルを参照・編集する場合、ファイルをダブルクリックすると OS がファイルの種類に関連したプログラムを起動させる。Windows のデフォルト動作では ZIP ファイルをダブルクリックするとエクスプローラーが起動する。Windows ではファイルの種類を特定する拡張子と特定のプログラムを関連付ける機能がある。この機能はコントロールパネルから操作できるが、ZIP ファイルに対して特定のプログラムを関連付ける項目が存在せず、ZIP ファイルをダブルクリックした場合ラッパプログラムを起動するように変更する事はできなかった。そこで、ラッパプログラムに復号化する ZIP ファイルをドロップしてラッパプログラムを起動する事とした。この動きは Windows GUI でプログラムに引数(ファイル名)を引き渡す場合のデフォルトの動作である。

一方、Mac OS X で Python を用いてコンパイルした場合、コマンドタイプのプログラムが作成される。GUI でファイルをコマンドタイプのプログラムにドロップしただけでは、プログラムにファイル名を引数として引き渡して起動できない。そこで、AppleScript を用いて Windows と同様に復号化するファイルをドロップすることでラッパプログラムが起動するようにした。

8. 実証実験

現在、3 名の宮崎大学教員(農学部(理系)1 名、地域資源創成学部(文系)2 名)に半年程度に亘ってラッパプログラムを試用していただいている。今のところ動作不良、不具合等は出ていない。

9. 今後

今回パスワード付 ZIP ファイルを簡単に復号、参照・編集、再暗号化するラッパプログラムを開発した。今後いくつかラッパプログラムの改良を予定している。

ユーザからラッパプログラムが異常終了した場合復号化されたファイル・フォルダがどうなるか質問があった。現在の仕様ではラッパプログラムが異常終了した場合復号化されたファイル・フォルダはそのままパソコンに残る。復号化されたファイル・フォルダがパソコンに残ることはセキュリティ上好ましい動きではない。ラッパプログラムが異常終了した場合でもセキュリティを確保できる方策(復号化したファイルの削除、参照・編集前のパスワード付 ZIP ファイルに戻す等)を検討したい。

また、セキュリティと利便性を高めるためにパスワード一時保存サーバへのユーザ毎のアクセス制限を付加したい。

今回は暗号化方式としてセキュリティ強度の高くないパスワード付 ZIP を選んだ。これ以外にも暗号化強度の高い暗号化方式を活用出来るようにする予定である。

10. 参考文献

[1] 独立行政法人 情報処理推進機構(IPA)、初めての情報セキュリティ対策のしおり(第 1 版)、1-5 ページ、2012 年、http://www.ipa.go.jp/security/antivirus/documents/09_hazimete.pdf

[2] NPO 日本ネットワークセキュリティ協会セキュリティ被害調査ワーキンググループ、2015 年情報セキュリティインシデントに関する調査報告書【速報版】Ver. 1.0 、4 ページ、2016 年、http://www.jnsa.org/result/incident/data/2015incident_survey_sokuhou.pdf

[3] 独立行政法人 情報処理推進機構(IPA)、暗号化による<情報漏えい>対策のしおり(第 1 版)、7-8 ページ、2014 年、https://www.ipa.go.jp/security/antivirus/documents/12_crypt.pdf

[4] Dekart Private Disk 、http://www.dekart.com/products/encryption/private_disk/

[5] セキュリティ・ウェアハウス、http://www.ost-net.com/swh/

[6] 日本年金機構における不正アクセスによる情報流出事案検証委員会、検証報告書、12 ページ、2015 年、http://www.mhlw.go.jp/kinkyu/dl/

houdouhappyou_150821-02. pdf
[7] 独立行政法人 情報処理推進機構(IPA)、暗号化による<情報漏えい>対策のしおり(第 1 版)、5 ページ、2014 年、https://www.ipa.go.jp/security/antivirus/documents/12_crypt.pdf
[8] 独立行政法人 情報処理推進機構(IPA)、電子メール利用時の危険対策のしおり(第 4 版)、11 ページ、2012 年 、 http://www.ipa.go.jp/security/antivirus/documents/07_mail.pdf
[9] 7-Zip、http://www.7-zip.org/
[10] unar,、https://unarchiver.c3.cx/commandline