MXLOGON2SC 説明書

USB トークンを利用した Windows ログイン

https://ribig.co.jp/MxLogon2sc

2018 年 12 月 1 日

有限会社リビッグ
横浜市港南区上大岡西 1-12-2

内容

1. MxLogon2sc について

スマートカード USB トークン認証とユーザ名/パスワード認証の 2 重認証によって Windows へログインを可能とするプログラムです。スマートカード USB トークン認証に成功しなければユーザ名/パスワード認証画面は表示されません。2 段階目のユーザ・パスワード認証は自動化できます。

2. USB スマートカードトークン/ドライバ

MxLogon2sc をインストールする前に USB トークンをコンピュータに接続してください。ドライバーが自動的インストールされます。配布ファイル内の x86 フォルダ内の changepin.exe を実行してみてください。トークンにシリアル番号が左リストに表示されていれば利用できる状態になっています。

USB トークンはドライバ/ユーティリティプログラムを含んだ CD-ROM 領域を搭載しています。Windows は その領域を CD-ROM（CryptoIDE_Setup）として認識し、CD-ROM 内のセットアッププログラムを実行するかどうか尋ねてきます。MxLogon2sc には CD-ROM のプログラムは不要です。もし、ドライバが自動でインストールされなければ CD-ROM の cryptoIDE_setup.exe を実行してください。

USB トークンはスマートカードリーダ内蔵の汎用的なスマートカードとして利用できます。CD-ROM 内のユーティリティをインストールすると、トークン設定ユーティリティや証明書ユーティリティが利用できるようになります。MxLogon2sc 納品時に設定されたデータを削除しなければ、スマートカードとして利用しても MxLogon2sc の動作には影響しません。

3. USB トークンのリダイレクトの確認

MxLogon2sc がインストールされたリモートコンピュータに USB トークンをつかってサインインすることができます。Windows のリモートデスクトップには、クライアント接続のスマートカード USB トークンをリモートコンピュータにリダイレクトする機能がありま

す。この機能が有効になっていると、クライアント側に接続している USB トークンがリモートコンピュータに接続している状態になります。

USB トークンでリモートコンピュータにログインする予定があれば、事前に、トークンがリダイレクトされるようにしてください。Windows バージョンによって設定は異なります。

リダイレクトされているかどうかを確認する方法

1．配布ファイルの x86 フォルダ内の changepin.exe と cryptoide_pkcs11.dll をリモートコンピュータの任意のフォルダにコピー
2．クライアントからリモートにログイン
3．リモートコンピュータで changepin.exe を実行。USB トークンが左リストに表示されればリダイレクトされています。トークンの抜き差しも試してください。

4. MxLogon2sc のインストール

配布パッケージ内の auto-setup.exe を実行してください。

MxLogon2sc セットアップが起動したら [インストール]ボタンのクリックでインストールが開始します。通常、数秒で完了します。

インストールが完了したら、Windows を再起動してください。

5. サインイン・ログイン

MxLogon2sc がインストールされていれば Windows 8/10 ではサインインオプションで MxLogon2sc を選択できます。Windows Vista/7 では MxLogon2sc のアイコンが表示されます。MxLogon2sc を選択後、正当なトークンが接続されていれば PIN を 入力できるようになります。

接続されていなければトークンを接続してください。 PIN フィールドが表示されます。

複数の正当なトークンを接続されているとリストからログインで使うトークンを選択できます。

初期設定 PIN は”12345678” です。PIN を入力後、リターンするとユーザ/パスワード認証画面に切り替わります。正しいユーザ/パスワードを入力後、リターンでサインインします。

PIN ロック

PIN を 10 回連続で間違えるとトークンはロックします。ロック解除/PIN 再設定はトークン設定ツール(cert.exe)で行います。

6. ロック

サインイン・ログイン中にトークンを抜き取ると Windows は ロックします。

ロックはログインと同じようにトークンを使って解除できます。

7. リモートデスクトップ経由サインイン・ロック

MxLogon2sc のトークンは Windows によってスマートカードとして認識されます。 RDP の USB リダイレクトを有効であれば、クライアント側に接続された USB トークンはリモート側で認識されます。リモートデスクトップでも USB トークンでのログイン、ロックが可能です。

ネットワークレベル認証が有効な場合、クライアント側で入力したユーザ/パスワードは MxLogon2sc が受け取ります。MxLogon2sc で PIN 認証が成功すると受け取ったユーザ/パスワードで自動ログインします（　トークンにユーザ名/パスワードが保存されていれば、保存資格情報が優先的に利用されます　）

8. トークン設定

スマートカードＵＳＢトークンはユーザ毎に異なる設定がされています。本ソフトウェアーはユーザ毎に異なるＵＳＢトークンと動作するように、ユーザ毎に作成されています。他ユーザの USB トークンが認識することはありません

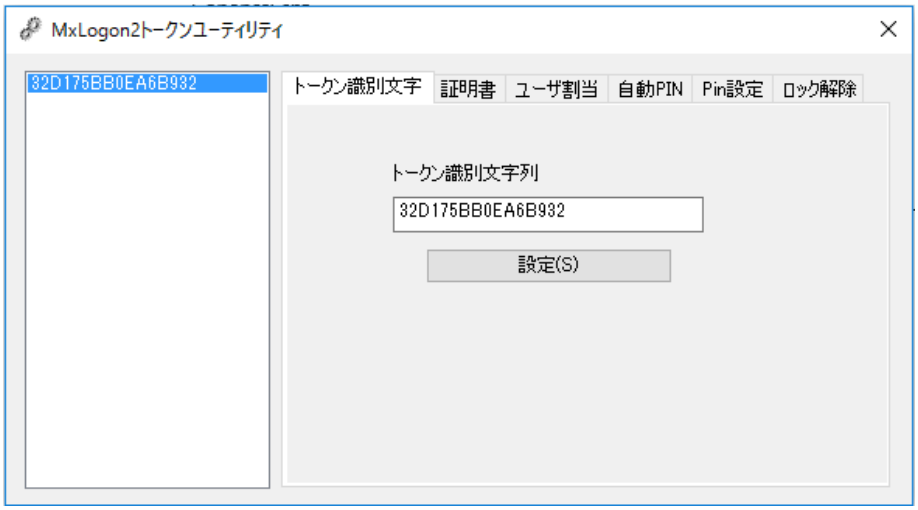
しかし、初期設定 PIN は全ユーザ同じものが設定されています。利用前に変更しなければ安全ではありません。トークンには自社作成の CA で署名したクライアント証明書を保存できます。そのような証明書が保存されていないトークンと動作しないようにすることが可能です。トークンを設定することで、よりセキュアにご利用いただけます。

トークンは管理者専用プログラム Cert.exe で設定します。配布ファイルに“トークン設定”フォルダが含まれます。このフォルダ内のすべてのファイルとフォルダを任意のフォルダ内に移動してからご利用ください。フォルダ内には重要なファイルが作成されます。移動先のフォルダには管理者以外のアクセス不可設定をしてください。

「トークン設定」フォルダ
Cert.exe
cryptoide_pkcs11.dll
[openssl] フォルダ
+ openssl.cnf
+ openssl.exe
+ libcryptoMD.dll
+ libsslMD.dll

8.1 識別文字列(オプション)

トークンを識別するために分かりやすい名前を設定します。設定した識別文字列でトークンを見分けることができるようになります。



左リストボックスで識別文字列を設定するトークンを選択後、任意の文字列を設定してから[設定]ボタンをクリックしてください

設定しなければトークンのシリアル番号が識別文字列になります。

8.2 証明書（オプション）

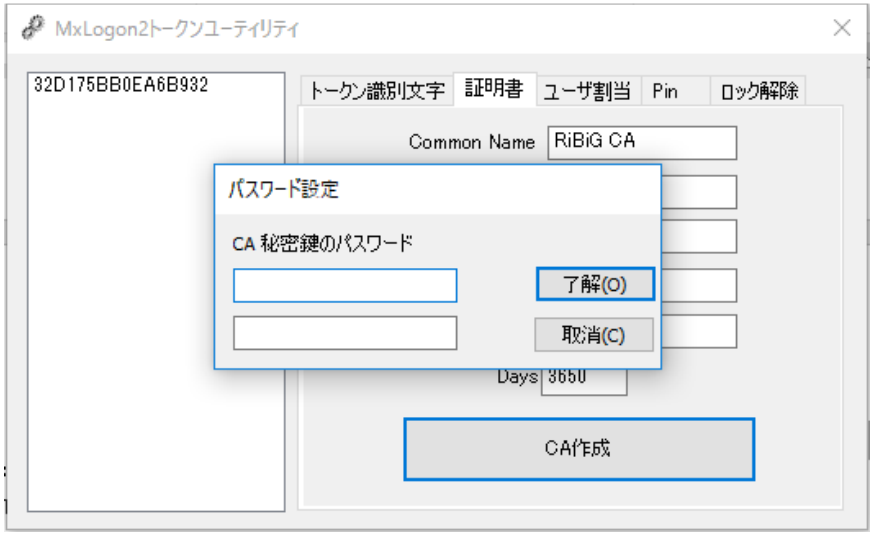
トークンにはクライアント証明書を保存できます。作成した CA によって署名されたクライアント証明書をトークンに保存することで、証明書をもっていないトークンを認識させないようにできます。

この証明書ＴＡＢでは、OpenSSL コマンドラインプログラムを使って CA を生成したり、クライアント証明書を作成したりしています。このＴＡＢを使わずに OpenSSL コマンドで CA 生成やクライアント証明書作成してもかまいません。.pfx 形式のクライアント証明書はＵＳＢトークンのユーティリティでトークンにインポート可能です。

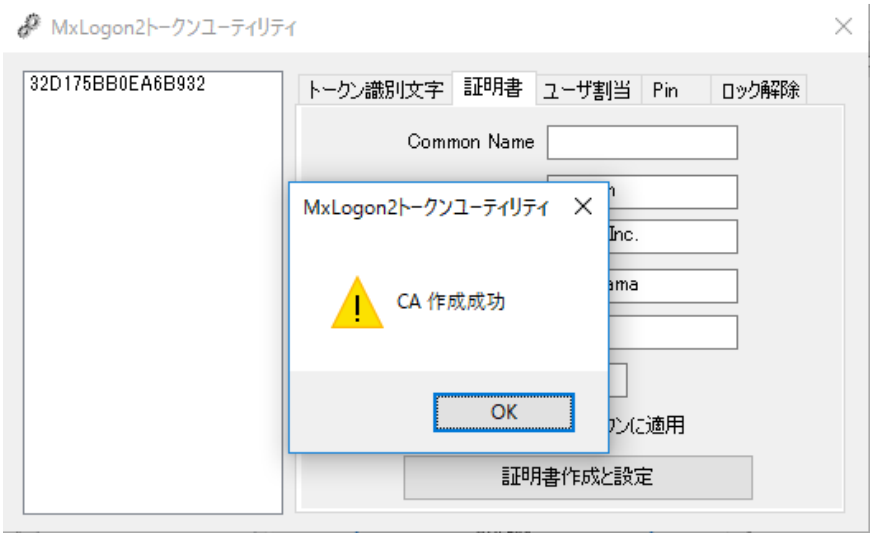
この TAB で証明書をトークンに保存するには、最初にＣＡを作成します。初期状態では CA は作成されていません。証明書タブには [CA 作成]ボタンが表示されます。

Common Name, Organization Unit, Organization, Locality, Country(2 文字), Days(有効期限)を設定後、[CA 作成]ボタンをクリックしてください。

CA の秘密鍵を保護するパスワードを求めてきます。任意のパスワードを入力後、[了解]をクリックしてください。



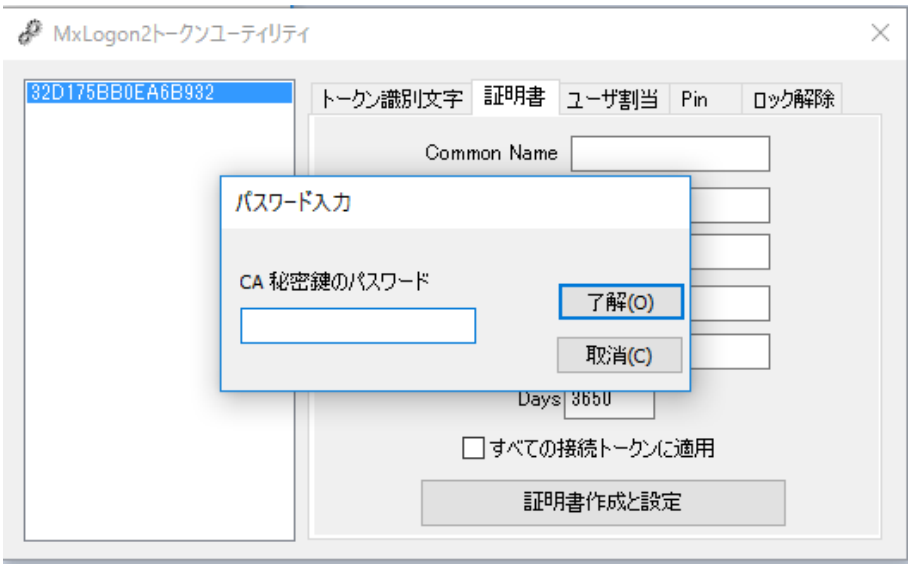
CA 関連ファイルが作成されると “CA 作成成功“と表示され、ボタンの表示は証明書発行用に切り替わります。



[証明書作成と設定]は鍵ペアを生成後、公開鍵を CA によって署名した証明書を作成、その証明書と秘密鍵をトークンに設定します。

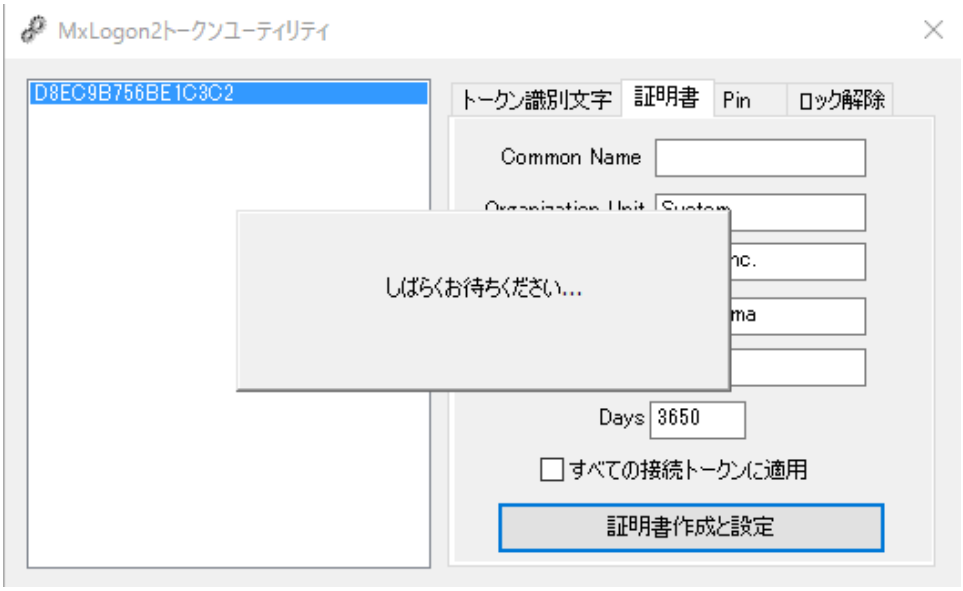
Organization Unit, Organization, Locality, Country, Days には CA 作成時にセットしたものが表示されます。適当に変更してください。

Common Name にはトークンのシリアル番号を設定しなければなりません。Common Name フィールドを空のまま（　空のままにするとトークンのシリアル番号が CN としてセットされます　）[証明書作成と設定]ボタンをクリックしてください。

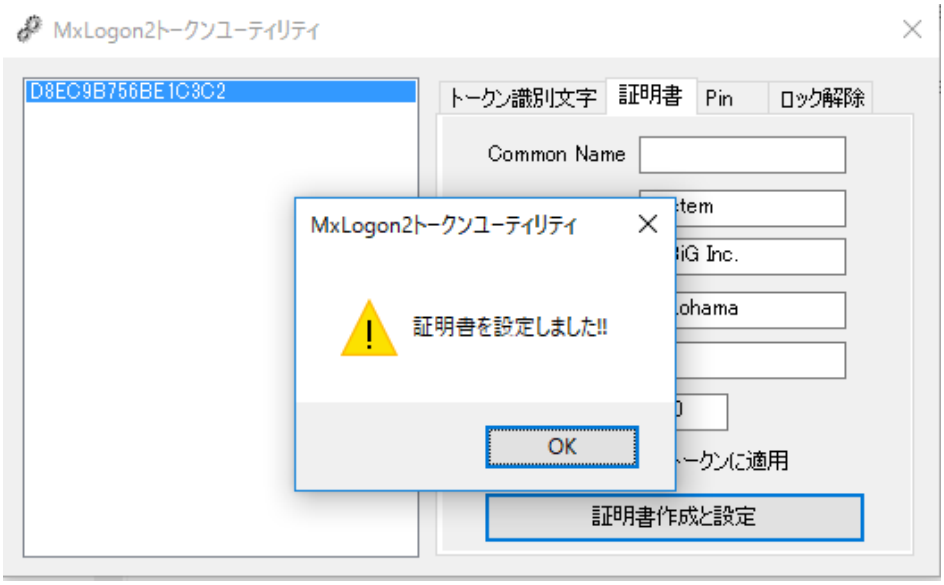


Common Name にトークンのシリアル番号以外を設定しても証明書は作成されトークンに保存されますが、MxLogon2sc のプログラムは正しい証明書と見なしません。証明書エラーとなりトークンは認識されません。

CA の秘密鍵入力後、[了解]ボタンで証明書の作成とトークンへの設定が開始します。



完了すると以下メッセージが表示されます。



複数トークン同時設定

1 度の操作で複数トークンに同時に証明書作成と設定を行うこともできます。“すべての接続トーク人に適用”にチェック後、Common Name を空のまま [証明書作成と設定]をクリックしてください。トークンのシリアル番号を Common Name とした証明書を作成してトークンにセットする処理が、すべてのトークンに対して繰り返されます。

複数 CA の作成

Cert.exe の置かれているフォルダ内に openssl フォルダがあります。Cert.exe はそのフォルダ内の openssl.exe コマンドを使って CA、証明書を作成しています。作成された CA, 証明書はそのフォルダ内に作成されます。

トークン設定フォルダを複数のフォルダにコピーして、それぞれのコピー先のフォルダで cert.exe を実行すると、各 openssl フォルダで CA, 証明書が作成されます。

USB トークン認証で特定 CA によって発行されたクライアント証明書の必須化

特定 CA によって発行されたクライアント証明書がトークンに保存されていなければトークンが認証されないようにできます。そのためには MxLogon2sc フォルダに CA 証明書ファイルを配置します。

CA を作成した cert.exe と同じフォルダ内の openssl フォルダに cacert.pem ファイルがあります。このファイルを　MxLogon2sc フォルダ（既定では C:¥Program Files¥RiBiG¥MxLogon2sc ） にコピーしてください。 MxLogon2sc は同じフォルダに cacert.pem を見つけると、この CA によって署名された証明書がトークン内に存在することを確認します。また、署名された公開鍵に対応する秘密鍵が存在することも確認します。確認できなければトークン認証は不可能になります。

証明書の期限もチェックされます。トークンの使用可能期限を証明書で設定することができます。証明書の Common Name はトークンのシリアル番号と一致しなければなりません。

クライアント証明書の Web サイトでの利用（オプション）

トークンに保存したクライアント証明書は、証明書で保護した Web サイトへのアクセスに利用できます。

Apache 設定

1．トークン付属のユーティリティのインストール
2．MxLogon2sc フォルダに配置した CA 証明書ファイル(cacert.pem) を Web サーバにコピー後、そのパスを　SSLCACertificateFile　にセット
3．証明書で保護したいディレクトリの設定

```
<Location /protected>
    SSLOptions +StdEnvVars
    SSLVerifyClient require
</Location>
```

Windows へのログインと Web へのセキュアなアクセスを同時に実現できます。

8.3 ユーザ割当（オプション）

USB トークンにユーザ資格情報を保存すると、ユーザ・パスワード認証は自動化できます。既定では トークン認証が完了すると、ユーザ名、パスワード認証の画面に切り替わり、そこで手入力でユーザ名/パスワードを設定して認証します。トークンにユーザ資格情報を保存しておくとトークン認証完了後、自動で保存した資格情報でユーザ/パスワード認証されてログインします。

ユーザ名、パスワードを設定して[設定]ボタンをクリックしてください。

特定のコンピュータ名のローカルユーザのドメインは、そのコンピュータ名です。任意のコンピュータのローカルユーザを指定するには “.”(ﾄﾞｯﾄ)を指定します。.¥user で任意のコンピュータの ユーザ user でログインできます。

ユーザ名だけを設定してパスワードを指定しなければ、PIN 認証後、自動ログオンしません。代わりにユーザ名/パスワード認証画面に切り替わり、設定ユーザ名が自動でユーザ名フィールドに入力されます。

割当ユーザ削除

割当ユーザを削除するにはユーザ名を空にして[設定]ボタンをクリックしてください。

自動パスワード管理

トークンにユーザ名とパスワードを割り当てると Windows にログイン中のパスワード変更やログイン時のパスワード変更が自動で行われるようになります。

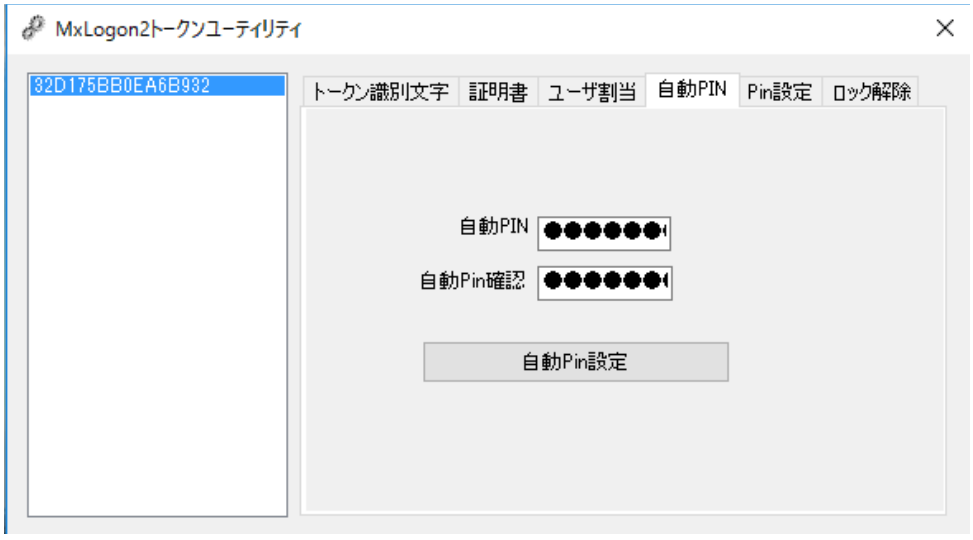
ログイン中に CTRL+ALT+DEL で表示されるメニューで “パスワード変更”を選択後、ログインしたトークンの PIN を入力すると自動でパスワードが変更されます。MxLogon2sc

が自動生成するパスワードが設定されますので、どのようなパスワードが設定されたのかは分かりません。

パスワード変更前に SHIFT+CTRL キーを押し続けているとパスワード自動生成/設定はされずに、新規パスワードは手入力設定が可能になります。

8.4 自動 PIN（オプション）

トークン認証には PIN 入力は必須です。トークンユーザが毎回 PIN を手入力するのが基本です。運用上 PIN 入力を省きたいケースに対応するため MxLogon2sc が自動的に PIN 入力を行う機能があります。トークンの PIN を保存しておくと、MxLogon2sc が PIN 入力を自動で入力してユーザ・パスワード認証に進むことができます。



自動 PIN を削除するには”自動 PIN”フィールドを空にして、[自動 PIN 設定]ボタンをクリックしてください。

7.5 PIN の設定（必須）

トークン識別文字列、証明書、ユーザ割当処理ではトークンの PIN を事前に設定しませんでした。これは cert.exe の 3 つのタブ（トークン識別文字列、証明書、ユーザ割当）ではトークン納品時の既定ユーザ PIN（12345678）がセットされていることを前提としているためです。逆に言うと、トークン識別文字列、証明書、ユーザ割当を行うにはユーザ PIN は 12345678 でなければなりません。トークンを再設定する前に、ユーザ PIN は既定 PIN(12345678) に戻すようにしてください。

識別文字列、証明書、ユーザ割当が済んだら、再度、利用時のＰＩＮに変更します。すべてのトークンは、運用前までにユーザ PIN と SO（管理者）PIN を変更してください。

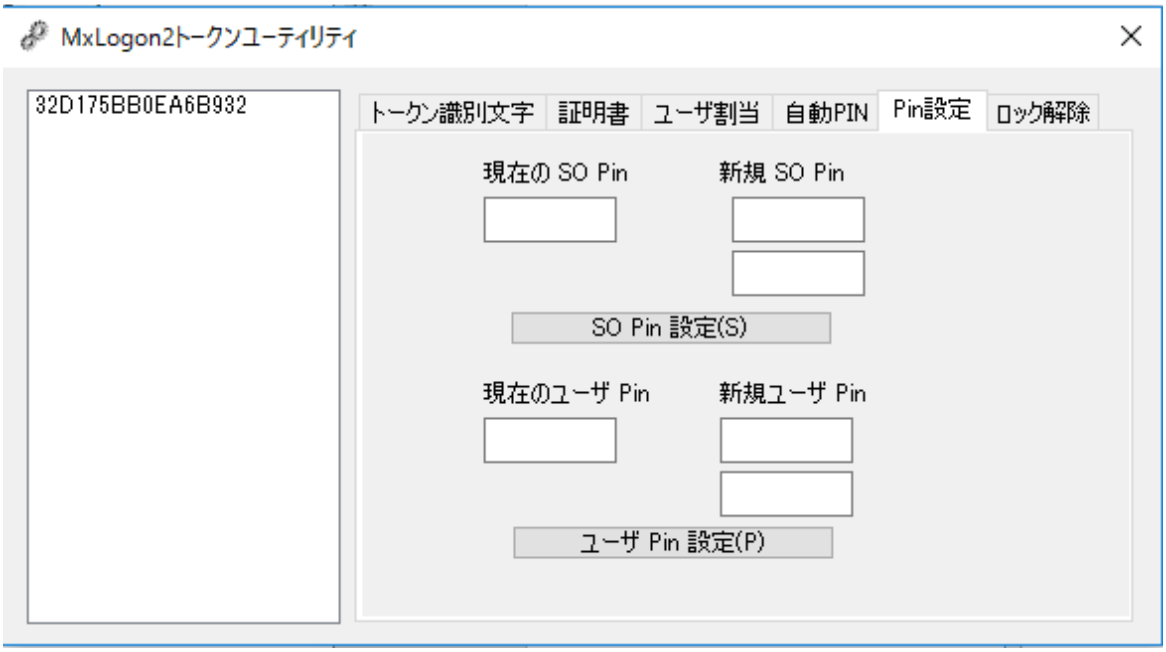
既定 SO （管理）PIN　：　“admin123”

既定 ユーザ PIN　　：　“12345678”

SO PIN は実運用前に必ず変更しなければなりません。変更しないままにしておくと、ツールで勝手にトークン設定を変更されてしまう可能性があります。

Pin 設定タブで接続トークンすべてを対象に SO Pin, ユーザ Pin は変更できます。

ユーザ PIN は既定のままユーザに渡して、ユーザに ChangePin.exe ユーティリティで任意の PIN に変更するよう依頼しても構わないかもしれません。



SO Pin 変更

現在の SO PIN（" admin123" ）と新規 SO PIN をセット後、 [SO Pin 設定]ボタンをクリックすると、すべての接続トークンを新規 SO PIN に変更します。既定 SO Pin が自動入力されますが、実際に設定されている SO PIN をトークンから取得する手段はありません。必ず、現在設定されている SO PIN を現在の SO Pin に設定するようにしてください。

ユーザ Pin 変更

現在のユーザ PIN（" 12345678" ）が自動設定されますが、実際に設定されている ユーザ PIN をトークンから取得することはできません。必ずトークンに設定されているユーザ Pin を現在のユーザ Pin に設定してください。新規ユーザ PIN をセット後、 [ユーザ Pin 設定]ボタンをクリックすると、すべての接続トークンを新規ユーザ PIN に変更します。

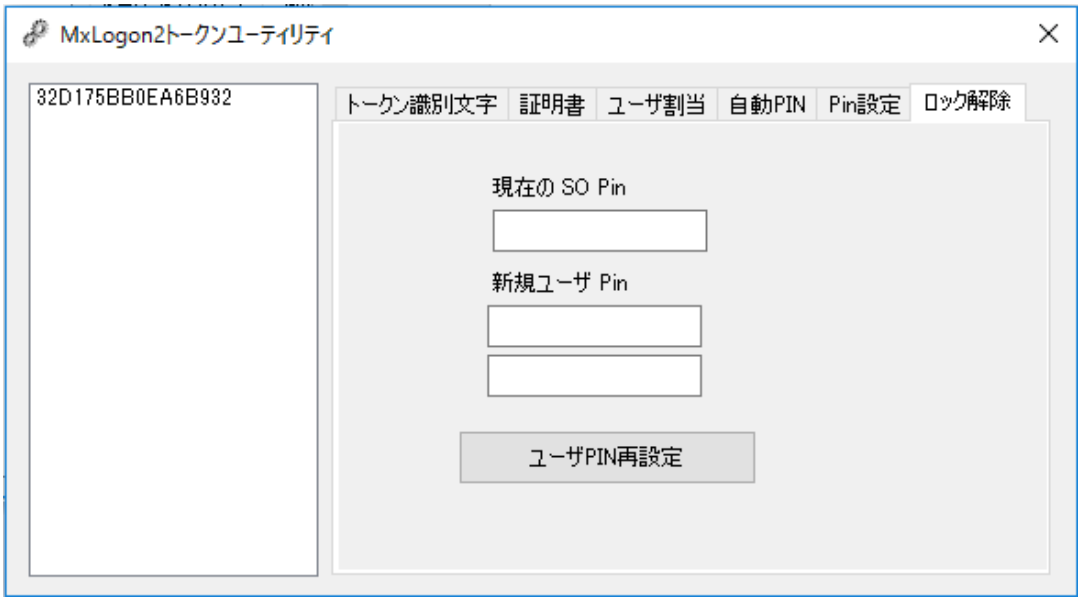
変更ログ

PIN 変更操作は cert.exe と同じフォルダ内の keypin.log というログファイルに書き込まれます。トークンに設定された PIN はログに残ります。

7.6 ロック解除（必要時）

このタブはハードウェアロックしてしまったトークンの ユーザ PIN を解除するときに利用します。

ユーザ PIN がロックしたトークンを接続後、SO Pin とユーザ Pin をセット後にボタンをクリックすると設定したユーザ PIN にリセットされます。



9. MxLogon2sc の設定

管理者は「スタート」－「MxLogon2sc」－「設定」で MxLogon2sc の設定を行えます。

（c:¥Program Files¥RiBiG¥MxLogon2sc¥config.exe）

6.1 設定タブ

トークン取り外し時処理

ログイン中にトークンを抜き取ったときの処理を設定できます。既定ではロックしますが、サインアウト、または、何もしないように設定可能です。

登録キーのみ利用可能

MxLogon2sc がコンピュータに登録したトークンだけを認識するようにできます。チェックすると登録済みトークン一覧を表示したり、接続トークンを登録できます。

注意：

このオプションを有効にしたら必ずトークンを登録してください。トークンを登録しないままにしているとトークンによるログインが不可能になります。

この問題を防止するため以下機能が組み込まれています。

- 登録されているキーがなければオプションは有効になりません。
- 識別文字列の末尾を “-master”とするとそのトークンは登録していなくても利用可能になります。

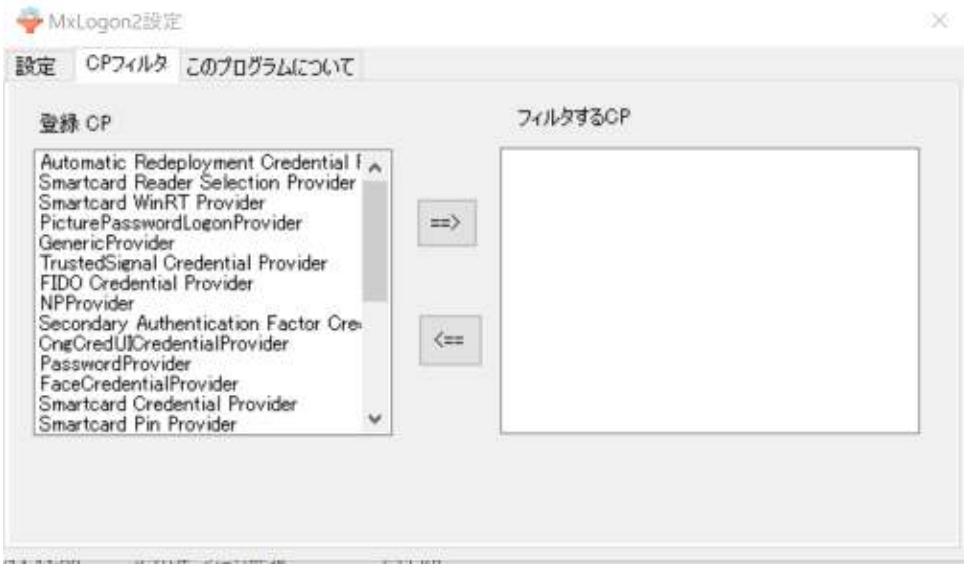
登録キー一覧

MxLogon2sc フォルダ内の tokens フォルダ内に登録トークンの情報が書き込まれています。そのフォルダを表示します。トークン登録を削除するには対応するファイルを削除してください。

接続トークンのコンピュータへの登録

リストに表示されている接続トークンに対応するファイルを tokens フォルダに作成します。

6.2 CP フィルタタブ



コンピュータに登録されているログインプロバイダをフィルタ（利用不可）できます。左側リストでフィルタするプロバイダを選択後、＝＞ボタンで右側リストに移動させてください。

Windows ログイン時にトークンがなければログインできないようにするには、

PasswordProvider,

PinLogonProvider,

PicturePasswordProvider,

WLIDCredentialProvider

Smartcard Credential Provider

NGC Credential Provider

をフィルタします（ Windows Vista/7 には PinLogonProvider, PicturePasswordProvider, WLIDCredentialProvider は存在しません）

セーフモードでの無効化

セーフモードではスマートカードドライバが動作しませんので、USB トークンが利用できません。そのため MxLogon2sc はセーフモードでは無効化されます。

セーフモードでコンピュータを保護するには、市販 USB メモリでログインを制限する弊社 SimplePCLock 利用をご検討ください。

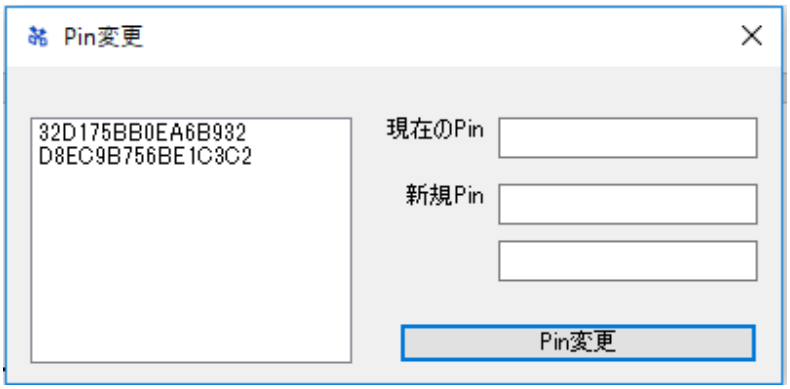
https://ribig.co.jp/simplepclock

10. ユーザ PIN 変更

トークンユーザは「スタート」－「MxLogon2sc」－「PIN 変更」でトークンの PIN を変更できます。

（%ProgramFiles%¥RiBiG¥MxLogon2sc¥changePin.exe）

起動後、左リストで PIN を変更するトークン選択後、現在のユーザ PIN と新しく設定する PIN 入力後、PIN 変更ボタンをクリックしてください。



11. アンインストール

コントールパネルの「プログラムをアンインストール」から MxLogon2sc をアンインストールできます。インストールしたアカウントでアンインストールしてください。

12. ライセンス

OpenSSL License

/*

==========

=====

* Copyright (c) 1998-2017 The OpenSSL Project. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in
*   the documentation and/or other materials provided with the
*   distribution.
*
* 3. All advertising materials mentioning features or use of this
*   software must display the following acknowledgment:
*   "This product includes software developed by the OpenSSL Project
*   for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
*
* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
*   endorse or promote products derived from this software without
*   prior written permission. For written permission, please contact
*   openssl-core@openssl.org.
*
* 5. Products derived from this software may not be called "OpenSSL"
*   nor may "OpenSSL" appear in their names without prior written
*   permission of the OpenSSL Project.
*
* 6. Redistributions of any form whatsoever must retain the following
*   acknowledgment:
*   "This product includes software developed by the OpenSSL Project
*   for use in the OpenSSL Toolkit (http://www.openssl.org/)"
*
* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY
* EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
* PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

```
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 *
=====================================================================
=====
 *
 * This product includes cryptographic software written by Eric Young
 * (eay@cryptsoft.com).  This product includes software written by Tim
 * Hudson (tjh@cryptsoft.com).
 *
 */

 Original SSLeay License
 -----------------------

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are adheared to.  The following conditions
 * apply to all code found in this distribution, be it the RC4, RSA,
 * lhash, DES, etc., code; not just the SSL code.  The SSL documentation
 * included with this distribution is covered by the same copyright terms
 * except that the holder is Tim Hudson (tjh@cryptsoft.com).
 *
 * Copyright remains Eric Young's, and as such any Copyright notices in
 * the code are not to be removed.
 * If this package is used in a product, Eric Young should be given attribution
```

* as the author of the parts of the library used.
* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. All advertising materials mentioning features or use of this software
*   must display the following acknowledgement:
*   "This product includes cryptographic software written by
*   Eric Young (eay@cryptsoft.com)"
*   The word 'cryptographic' can be left out if the rouines from the library
*   being used are not cryptographic related :-).
* 4. If you include any Windows specific code (or a derivative thereof) from
*   the apps directory (application code) you must include an acknowledgement:
*   "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
*
* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be

```
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*/
```