

**glserver-glclient/monsiaj間のクライアント認証
を利用した接続**

第6版

日本医師会総合政策研究機構

平成19年10月31日

目次

1 はじめに	4
1.1 構成	4
1.2 作業概要	5
2 証明書の発行	6
2.1 プライベートCA構築ツールのインストール	6
2.2 CA証明書の作成	6
2.3 サーバ証明書の発行	9
2.3.1 証明書のメディアへの格納	11
2.4 クライアント証明書の発行	14
2.4.1 証明書のメディアへの格納	15
2.5 クライアント証明書のセキュリティデバイスへの格納	19
2.6 ユーザDBファイルの作成	22
3 サーバ設定	23
3.1 証明書とユーザDBファイルの設置	23
3.2 jma-receiptパッケージの再設定	24
4 glclient 設定	26
4.1 パスワード入力ダイアログの設定	26
4.2 クライアント証明書を機材に直接保存する場合	26
4.2.1 証明書の設置	26
4.2.2 コマンドラインからのクライアントの起動	27
4.2.3 ダイアログ画面からのクライアントの起動	28
4.3 セキュリティデバイスを利用する場合	31
4.3.1 CA 証明書の設置	31
4.3.2 セキュリティデバイス利用の準備	31
4.3.3 コマンドラインからのクライアントの起動	31
4.3.4 ダイアログ画面からのクライアントの起動	33
5 monsiyaj 設定	35
5.1 Linux上での証明書の設置と接続	35
5.1.1 証明書の設置	35
5.1.2 クライアントの起動	36
5.2 Windows上での証明書の設置と接続	38
5.2.1 証明書の設置	38
5.2.2 クライアントの起動	38
5.3 MacOS X 10.4上での証明書の設置と接続	39
5.3.1 証明書の設置	39
5.3.2 クライアントの起動	40

変更履歴

第1版

平成18年7月13日
PKCS#12形式に対応

第2版

平成18年12月10日
jma-receipt パッケージの設定画面でのSSL設定に対応
jma-certtool に対応し「プライベートCA構築ツールの利用」と2つの文書に分割

第3版

平成19年1月26日
全体の誤字、誤ファイル名修正
「4.2.1」 証明書ファイル名が固定となることの追記
「4.3」 主サーバ以外で作成された userdb ファイルの登録方法の追記
「4.4.1」「4.5.1」 \$HOME の説明場所変更と PKCS#12 対応漏れの対応
「4.4.2」 <glserver ホスト名>の指定方法を追記

第4版

平成19年2月8日
「プライベートCA構築ツールの利用方法」と用語を統一
「1」 処理概要の内容を変更
「4.3」 ユーザDB更新時の作業を追記

第5版

平成19年5月8日
「5」 詳細設定を設定対象毎に3つの節に分離
「6.2.2」 セキュリティデバイスの設定方法を追加

第6版

平成19年10月31日
「1」「処理概要」を「はじめに」に変更
「2」 証明書の発行手順を詳細に記載
「3」「4」 証明書の設置手順を変更
「1」 構成図を追加

1 はじめに

医院中に構築されている LAN(院内 LAN)で日医標準レセプトソフト（以下日レセ）を使用するにあたり、第三者への情報漏洩を防ぐために通信を暗号化することが考えられます。

このドキュメントでは glserver-glclient/monsiaj 間の通信を SSL で暗号化し、かつ証明書を利用した認証(以下 SSL クライアント認証)を行うための、glserver、glclient および monsiaj に必要な作業を記載します。

このドキュメントの前提条件および範囲は以下の通りです。

- glserver-glclient/monsiaj 間の SSL クライアント認証を対象
- 証明書形式は X.509v3(RFC3280)で定義される仕様に従ったものを対象
- 日レセのバージョンは Ver.4.0.0
- glserver/glclient の動作環境は Debian GNU/Linux 3.1 または Debian GNU/Linux 4.0
- monsiaj の動作環境は J2SE Java Runtime Environment (JRE) 5.0 以降

1.1 構成

SSL クライアント認証では、glserver だけでなく glclient や monsiaj で利用するユーザーアカウント毎に個別の X.509 形式の証明書を持つ必要があります。X.509 形式の証明書は商用サービス(ベリサインなど)や openssl コマンドで作成したものなど様々なものが利用できますが、本文書では組織内の簡易認証局を構築し、そこから発行した証明書を利用します。

以下に構成図(図 1:構成図)を示します。

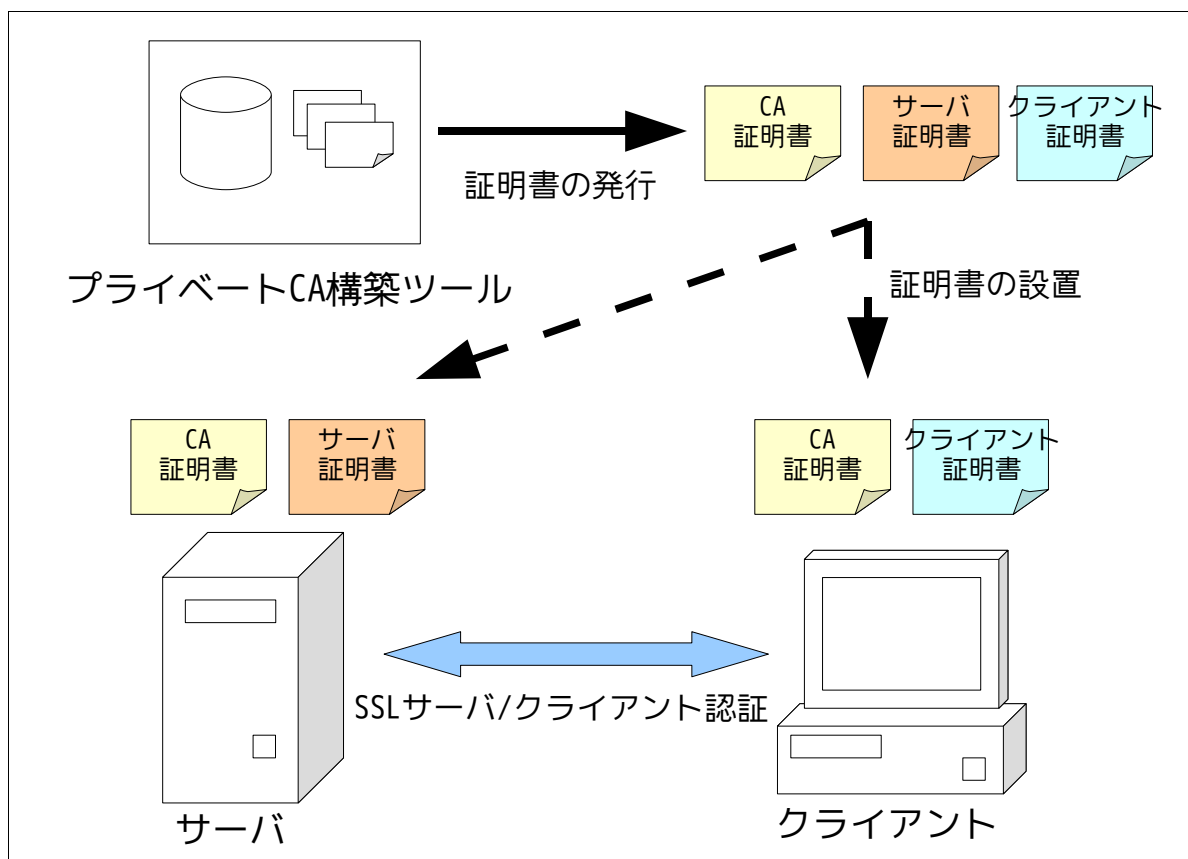


図 1:構成図

1.2 作業概要

本文書での作業の流れは以下のとおりです。

1. 証明書の発行

組織内の簡易認証局を作成し、サーバ証明書とクライアント証明書を発行します。

2. サーバ設定

サーバ証明書を発行し、glserver 用機材の証明書ストアに、CA 証明書とサーバ証明書をインストールします。

またクライアント証明書の DN とログインユーザ名を結びつけるユーザデータベースを設定します。

3. クライアント設定

glclient 及び monsiaj を利用する機材に、CA 証明書とクライアント証明書をインストールします。セキュリティデバイスを利用する場合はCA 証明書のインストールと環境設定を行います。

2 証明書の発行

組織内で証明書を発行、管理する「プライベート CA 構築ツール」を利用し、サーバ証明書、クライアント証明書を発行します。

プライベート CA 構築ツールについての詳細は文書「プライベート CA 構築ツールの利用」に記載されています。

2.1 プライベート CA 構築ツールのインストール

コンソールを開いて/etc/apt/sources.list ファイルに下記の内容を追記します。

- 使用している機材の Debian バージョンにより追記する内容が異なります。
- 日レセ導入済みの機材では既に同様の記述がある可能性があります。その場合 /etc/apt/sources.list の編集は必要ありません。

Debian GNU/Linux 4.0 etch の場合

```
deb http://ftp.orca.med.or.jp/pub/debian/ etch jma
deb-src http://ftp.orca.med.or.jp/pub/debian/ etch jma
```

Debian GNU/Linux 3.1 Sarge の場合

```
deb http://ftp.orca.med.or.jp/pub/debian/ sarge jma
deb-src http://ftp.orca.med.or.jp/pub/debian/ sarge jma
```

/etc/apt/sources.list 編集後、以下のコマンドを実行しプライベート CA 構築ツールをインストールします。

```
$ sudo aptitude update
$ sudo aptitude install jma-certtool
```

2.2 CA 証明書の作成

プライベート CA 構築ツールを起動し、CA 証明書を作成します。

コンソールを開いて以下のコマンドを実行します。

```
$ jma-certtool
```

root ユーザのパスワードが要求されるので入力します（図 2:パスワード入力画面）。

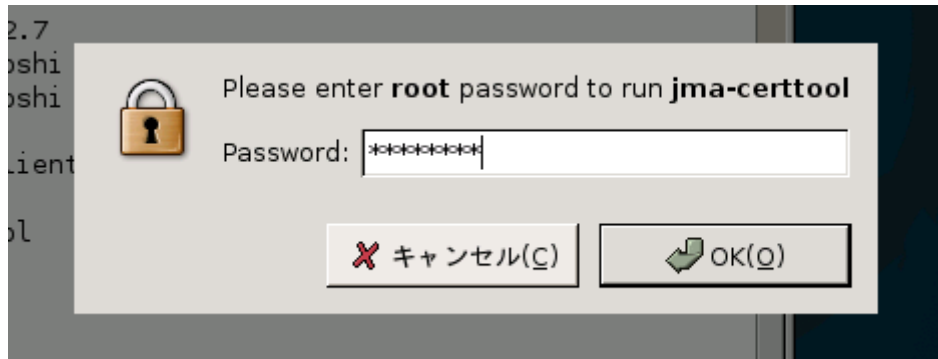


図 2:パスワード入力画面

パスワード入力に成功すると、「CA 証明書の作成画面」（図 3:CA 証明書作成画面）が表示されます。

既にプライベート CA 構築ツールを使用し、認証局を設定している場合は「メインウィンドウ」が表示されます。その場合はメニューの「ファイル」→「新規CAの作成」を選択し、「CA 証明書の作成画面」を起動します。

図 3:CA 証明書作成画面

CA 証明書の以下の設定項目を入力します。

新しいCAの名前

CA の名称を入力します。

組織名(必須)

CA を利用する組織名(医院名など)をローマ字で入力します。

コモンネーム(必須)

CA 証明書の名前をローマ字で入力します。

CA 鍵のパスワード

CA の私有鍵にパスワードを設定する場合は値を入力します。数字とアルファベットが利用可能です。

CA 鍵のパスワード (確認)

入力確認のため、「CA 鍵のパスワード」と同じ値を入力します。

入力後、「OK」をクリックすると CA 証明書が作成され、メインウィンドウ(図 4:メインウィンドウ)が表示されます。以降の操作はメインウィンドウで行います。

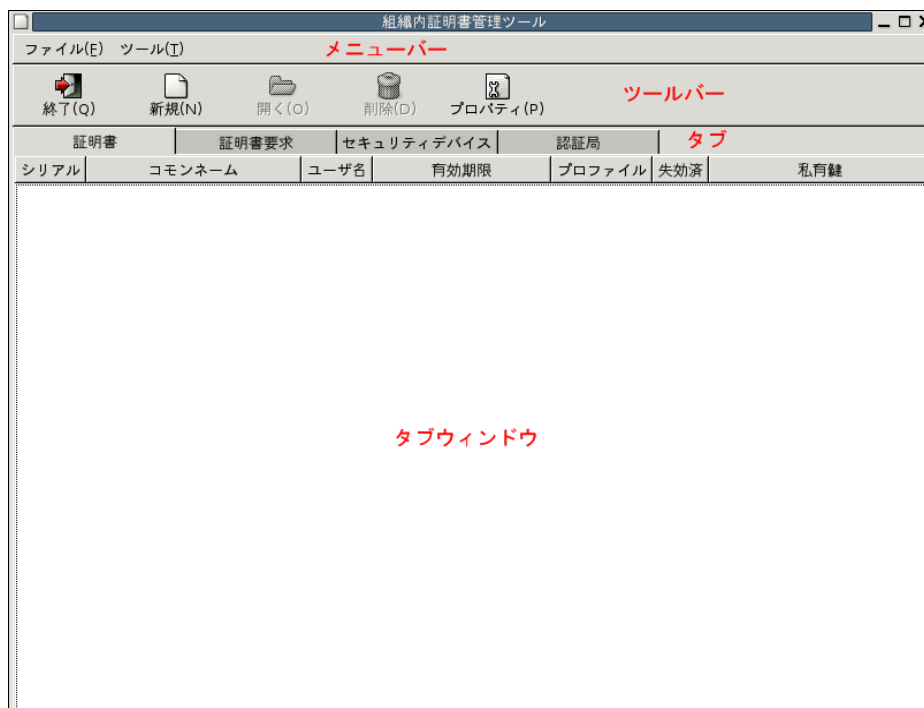


図 4:メインウィンドウ

2.3 サーバ証明書の発行

glserver が使用する証明書を発行します。

「証明書タブウィンドウ」を開き、「新規」ボタンをクリックします(図 5:証明書発行新規ボタン)。

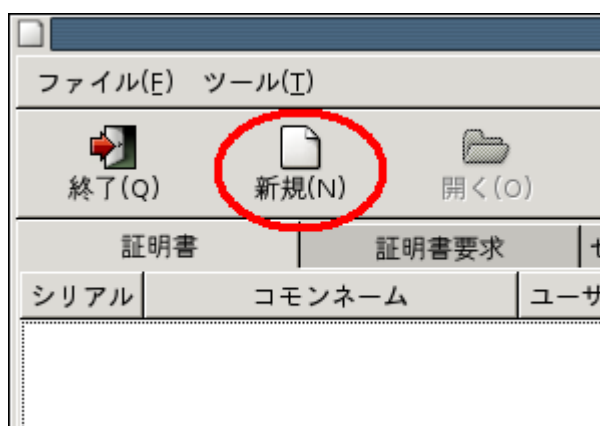


図 5:証明書発行新規ボタン

「証明書要求の編集」画面(図 7:CAによる署名画面)が表示されたら、コモンネームのみ入力します。コモンネームにはglclient、あるいはmonsiajが接続する際に指定するサーバ名、あるいはIPアドレスを入力します。

コモンネームが正しく設定されていない場合、glclient、monsiajが接続できないため注意してください。

国名(必須)	JP
都道府県名	
市町村名	
組織名(必須)	Nichii Clinic
部署名	
コモンネーム(必須)	main.example.or.jp
Eメールアドレス	
シリアル番号	
▶ サブジェクト別名	
鍵アルゴリズム	<input checked="" type="radio"/> RSA <input type="radio"/> DSA
鍵長(ビット数)	<input type="radio"/> 1024 <input checked="" type="radio"/> 2048 <input type="radio"/> 4096
ダイジェスト	<input type="radio"/> MD5 <input checked="" type="radio"/> SHA1 <input type="radio"/> SHA256
<input type="button" value="X キャンセル(C)"/> <input type="button" value="OK(O)"/>	

図 6:証明書要求の編集

コモンネーム入力後「OK」ボタンを押すと、「CAによる署名」画面(図 7:CAによる署名画面)が表示されるので、以下の項目を設定します。

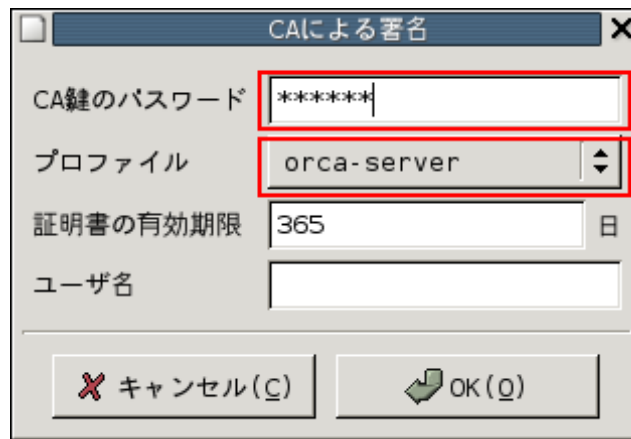


図 7:CA による署名画面

CA 鍵のパスワード

「CA 証明書の作成」画面でパスワードを設定した場合、パスワードを入力します。

プロファイル

プルダウンメニューから「orca-server」を選択します。

「OK」をクリックするとサーバ証明書が発行されます。

発行後、証明書タブに発行したサーバ証明書のエントリがあることを確認します。

2.3.1 証明書のメディアへの格納

発行したサーバ証明書と CA 証明書をサーバ機に設置するため、メディア(ここではフロッピーディスク)に格納します。

フロッピーディスクをマウントします。

```
$ mount /media/floppy
```

プライベート CA 構築ツールの「証明書タブウィンドウ」を開き、サーバ証明書のエントリの上で画面を右クリックします。

ポップアップウィンドウ(図 8:証明書の操作ポップアップウィンドウ)が表示されるので「エクスポート(PKCS#12)」を選択します。

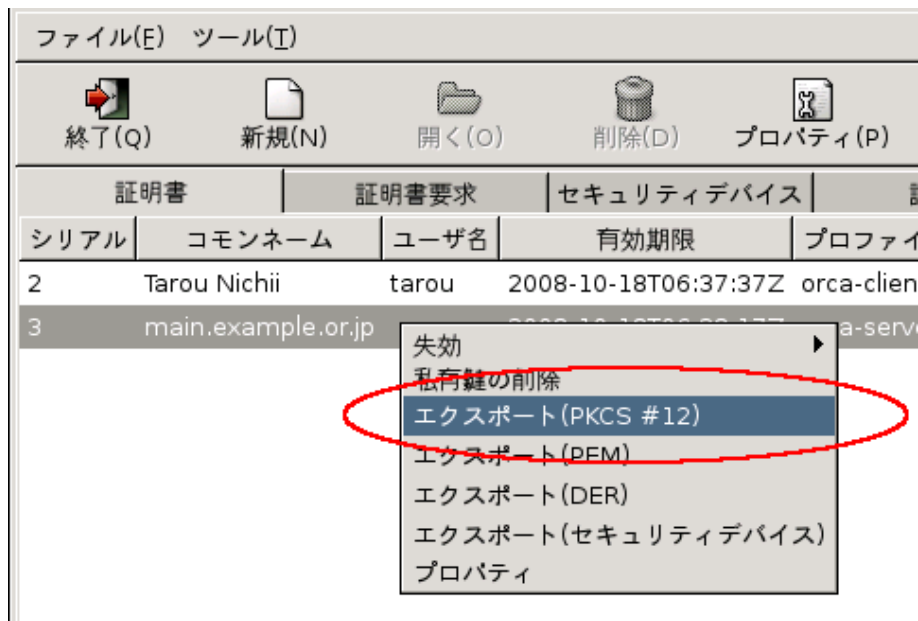


図 8: 証明書の操作ポップアップウィンドウ

パスワード入力画面（図 9: PKCS#12 パスワード入力画面）が表示されますが、パスワード欄は空のまま「OK」をクリックします。

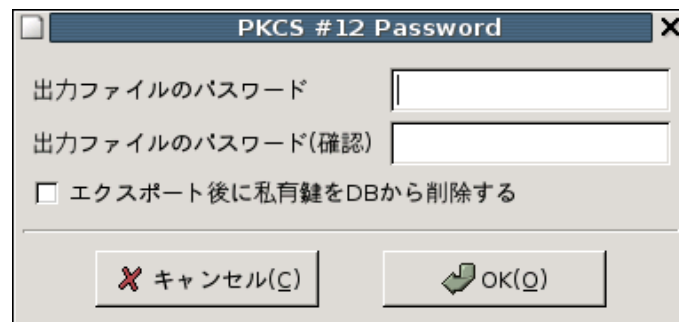


図 9: PKCS#12 パスワード入力画面

「空のパスワードを設定しますか？」と警告画面が表示されるので、「OK」をクリックします。（図 10: 空のパスワード警告画面）

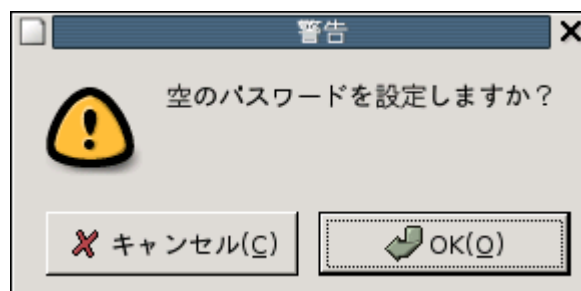


図 10: 空のパスワード警告画面

「出力ファイル保存」画面（図 11: 出力ファイル保存）が表示されたら、`/media/floppy/glserv.p12` に保存します。

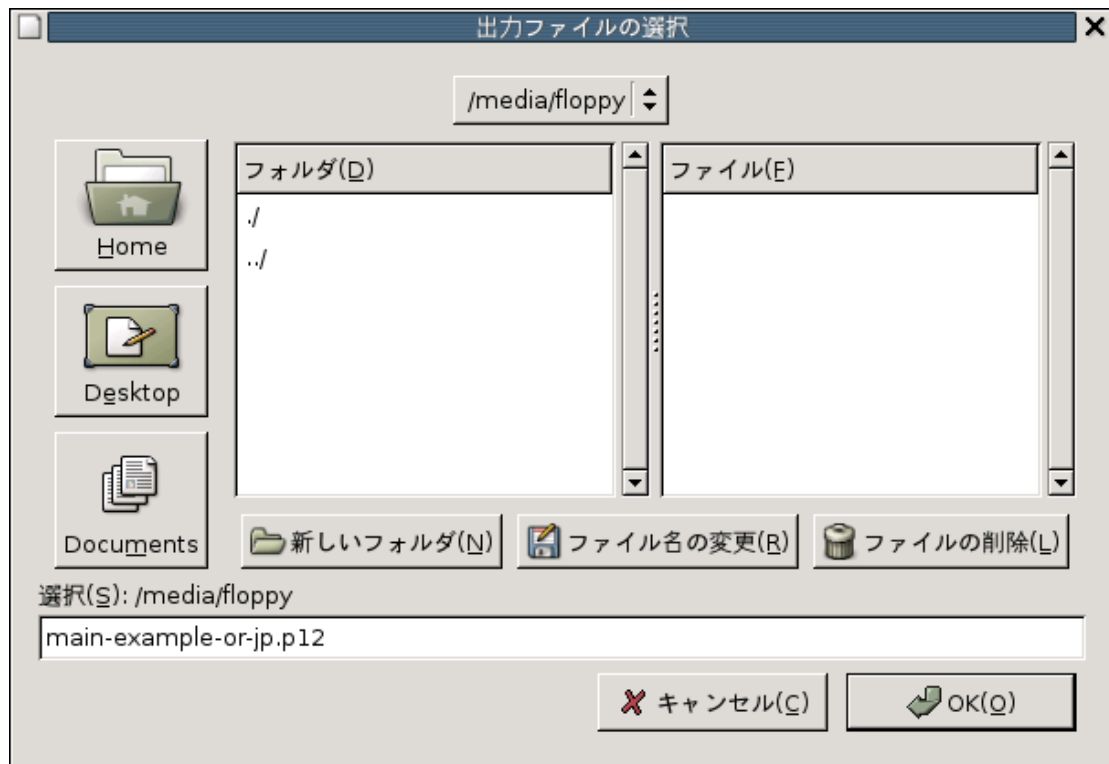


図 11:出力ファイル保存

次に CA 証明書をメディアに格納します。

「認証局」タブをクリックして「認証局タブウィンドウ」に移動します。

「CA 証明書のエクスポート (PEM)」ボタン(図 12:CA 証明書のエクスポート)をクリックすると「出力ファイル保存」画面が表示されるので `/media/floppy/gl-cacert.pem` として保存します。

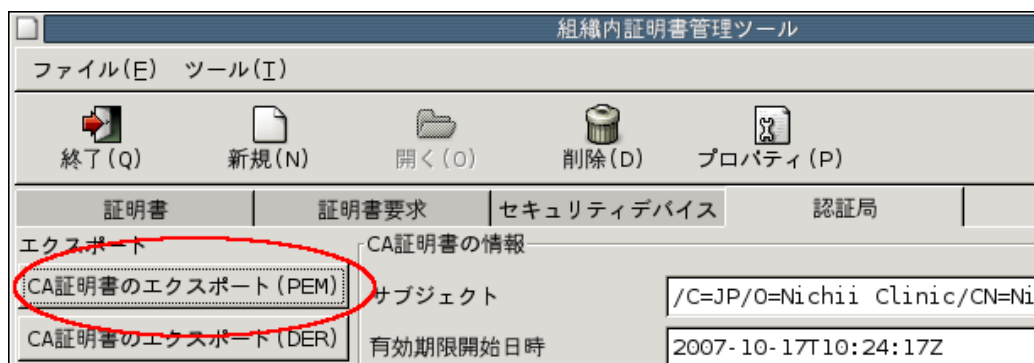


図 12:CA 証明書のエクスポート

保存後、フロッピーディスクをアンマウントします。

```
$ umount /media/floppy
```

2.4 クライアント証明書の発行

glclient、monsiaj で利用する証明書を発行します。

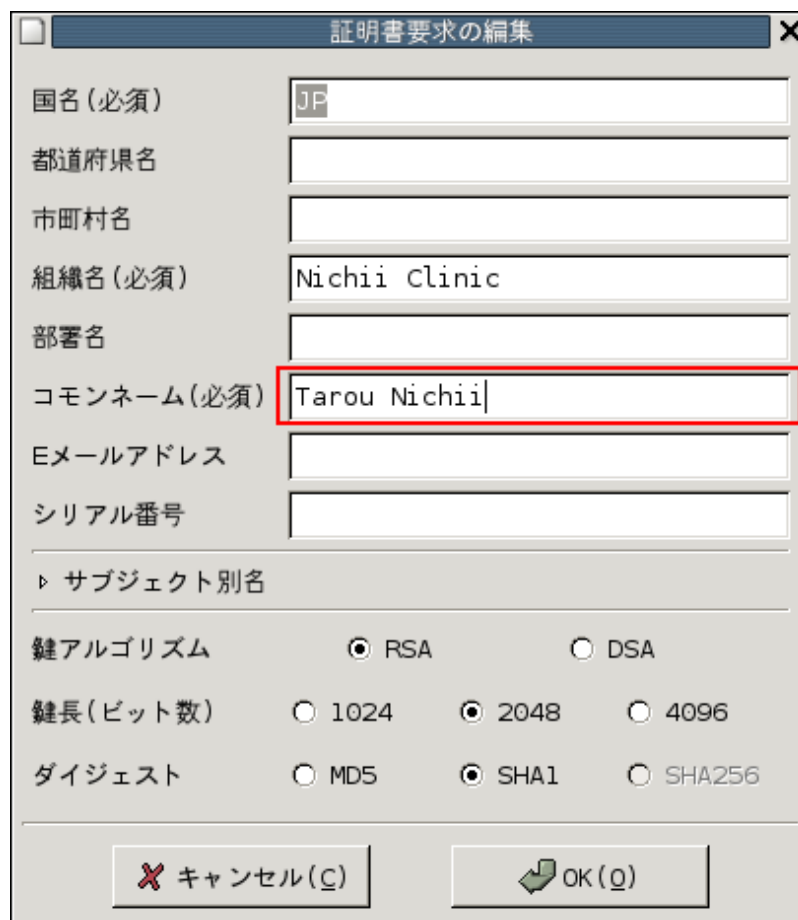
「証明書タブウィンドウ」を開き、「新規」ボタンをクリックします(図 5:証明書発行新規ボタン)。

「証明書要求の編集」画面(図 13:証明書要求の編集)が表示されたら、コモンネームを入力します。その他の項目は変更しません(セキュリティデバイス Aladdin eToken PRO を使用する場合を除く)。

Aladdin eToken PRO 32K を使用する場合

「鍵長 (ビット数)」を 1024 に変更します

コモンネームには証明書を使用する職員のフルネーム等、クライアントを特定する名前を設定します。



国名(必須)	JP
都道府県名	
市町村名	
組織名(必須)	Nichii Clinic
部署名	
コモンネーム(必須)	Tarou Nichii
Eメールアドレス	
シリアル番号	
▶ サブジェクト別名	
鍵アルゴリズム	<input checked="" type="radio"/> RSA <input type="radio"/> DSA
鍵長(ビット数)	<input type="radio"/> 1024 <input checked="" type="radio"/> 2048 <input type="radio"/> 4096
ダイジェスト	<input type="radio"/> MD5 <input checked="" type="radio"/> SHA1 <input type="radio"/> SHA256
✕ キャンセル(C)	
OK(O)	

図 13:証明書要求の編集

コモンネーム入力後「OK」ボタンを押すと、「CAによる署名」ウィンドウ(図 14:CAによる署名画面)が表示されるので以下の項目を入力します。

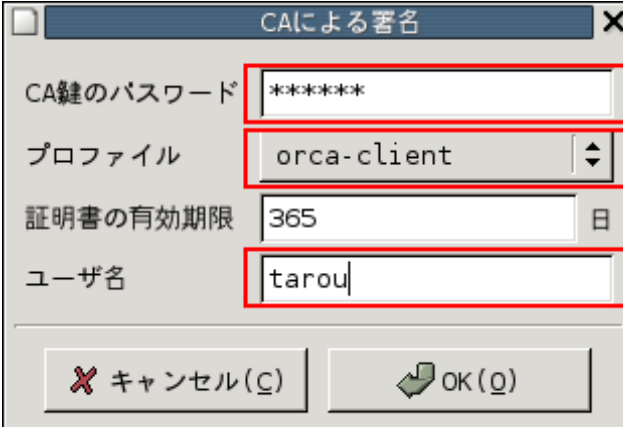


図 14:CAによる署名画面

CA 鍵のパスワード

「CA 証明書の作成」画面でパスワードを設定した場合、パスワードを入力します。

プロファイル

プルダウンメニューから「orca-client」を選択します。

ユーザ名

glserver にログインするユーザ名を指定します。このユーザ名はあらかじめ日レセに登録されている必要があります。

「OK」をクリックするとクライアント証明書が発行されます。

証明書タブに発行したクライアント証明書のエントリがあることを確認します。

2.4.1 証明書のメディアへの格納

発行したクライアント証明書と CA 証明書をクライアント機に設置するため、メディア(ここではフロッピーディスク)に格納します。作業はクライアント証明書の数だけ繰り返します。

フロッピーディスクをマウントします。

```
$ mount /media/floppy
```

プライベート CA 構築ツールの「証明書タブウィンドウ」を開き、対象のクライアント証明書のエントリの上で画面を右クリックします。

ポップアップウィンドウ(図 15:証明書の操作ポップアップウィンドウ)が表示されるので「エクスポート(PKCS#12)」を選択します。

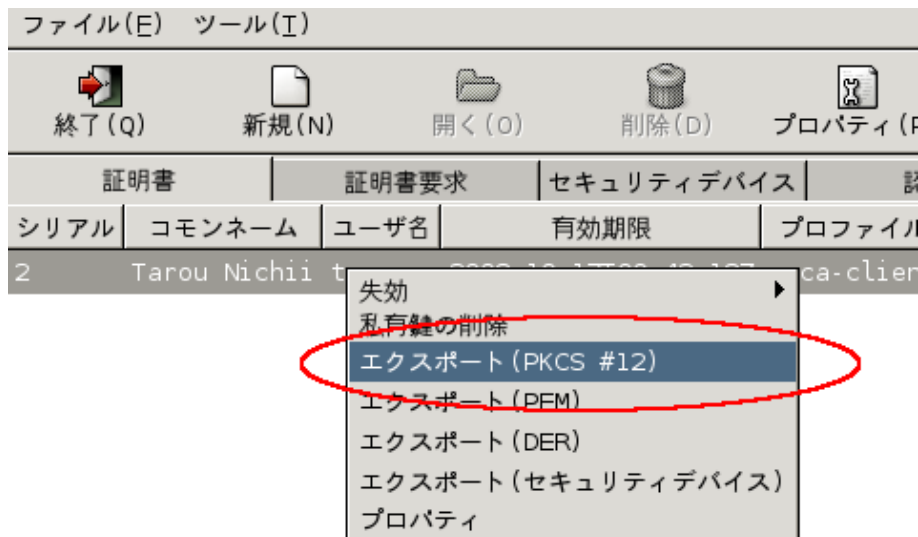


図 15: 証明書の操作ポップアップウィンドウ

パスワード入力画面（図 16: PKCS#12 パスワード入力画面）が表示されます。

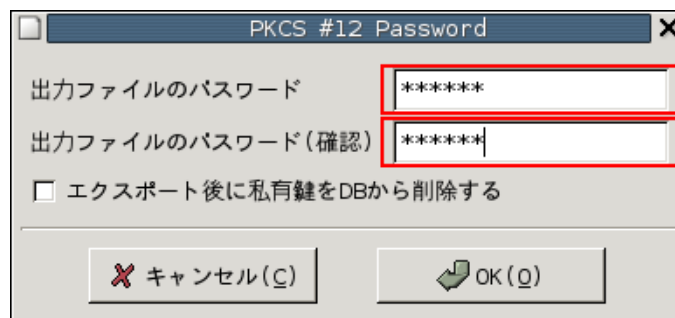


図 16: PKCS#12パスワード入力画面

パスワード入力画面に適切なパスワードを設定します。

monsiaj では、空のパスワードで作成された証明書を使用することができないためパスワードの設定が必要です。

パスワードに空を設定した場合、「空のパスワードを設定しますか？」と警告画面が表示されるので、「OK」をクリックします。（図 17: 空のパスワード警告画面）

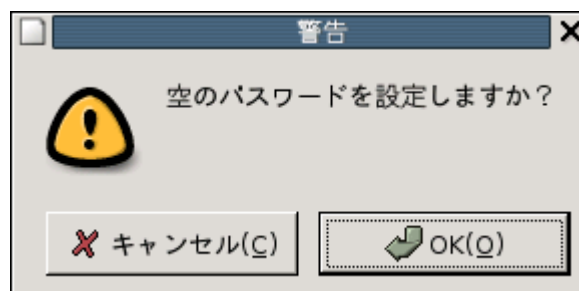


図 17: 空のパスワード警告画面

「出力ファイル保存」画面（図 18:出力ファイル保存）が表示されたら、/media/floppyの下に保存します。デフォルトでは保存されるファイル名は〈コモンネーム〉.p12 となります。

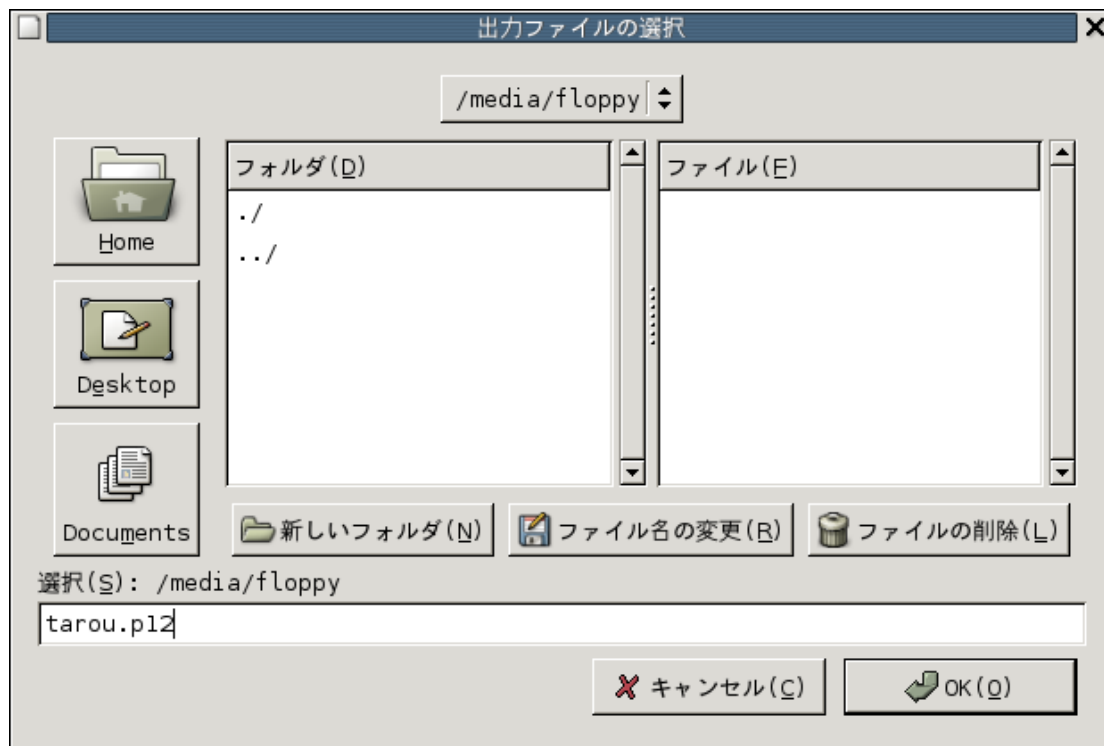


図 18:出力ファイル保存

次に CA 証明書をメディアに格納します。

「認証局」タブをクリックして「認証局タブウィンドウ」に移動します。

「CA 証明書のエクスポート(PEM)」ボタン(図 19:CA 証明書のエクスポート)をクリックすると「出力ファイル保存」画面が表示されるので /media/floppy/gl-cacert.pem として保存します。

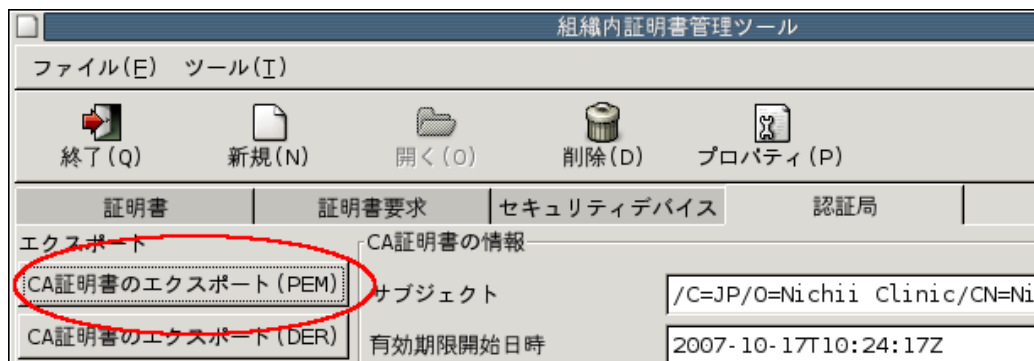


図 19:CA 証明書のエクスポート

保存後、フロッピーディスクをアンマウントします。

```
$ umount /media/floppy
```

2.5 クライアント証明書のセキュリティデバイスへの格納

glclient でセキュリティデバイスを使用する場合、クライアント証明書をセキュリティデバイスに格納します。また CA 証明書をメディア(ここではフロッピーディスク)に格納します。

この作業を行う前に、文書「プライベート CA 構築ツールの利用」の「2.1 セキュリティデバイスの環境設定」を実施しておく必要があります。

セキュリティデバイスを機材に接続します。プライベート CA 構築ツールのセキュリティデバイスタブウィンドウを表示します。

セキュリティデバイスが正しく認識されている場合「利用できるデバイス」ウィンドウにエントリが追加されます。エントリを選択して「デバイスの初期化」ボタンをクリック、またはエントリを右クリックして表示されるメニューから「デバイスの初期化」を選択して初期化を行います(図 20: デバイスの初期化)。

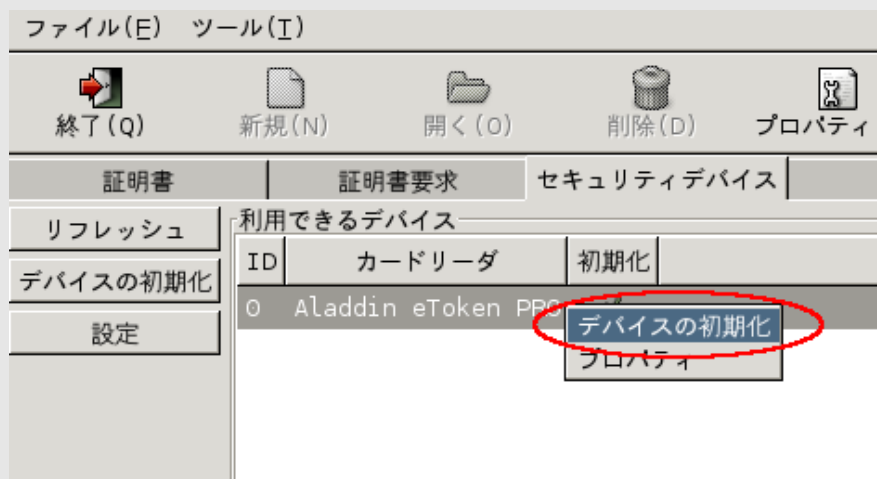


図 20: デバイスの初期化

既にデバイスに証明書が格納されている場合、初期化確認ダイアログ(図 21: 初期化確認ダイアログ)が表示されるので「はい」を押下し、PIN入力ダイアログ(図 22: PIN入力ダイアログ)に以前設定したPINを入力します。

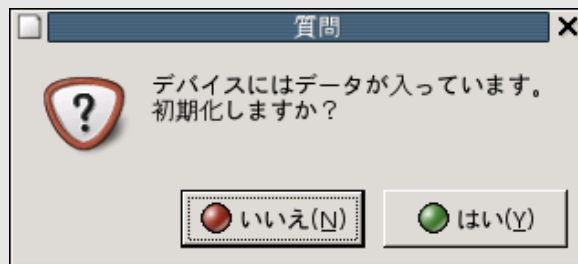


図 21:初期化確認ダイアログ

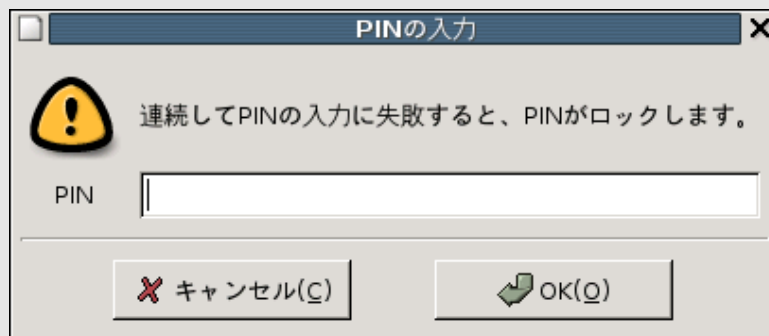


図 22:PIN入力ダイアログ

セキュリティデバイスのPINを設定する画面が表示されますので、任意の4～8文字の文字列を入力します（図 23:PINの設定）。

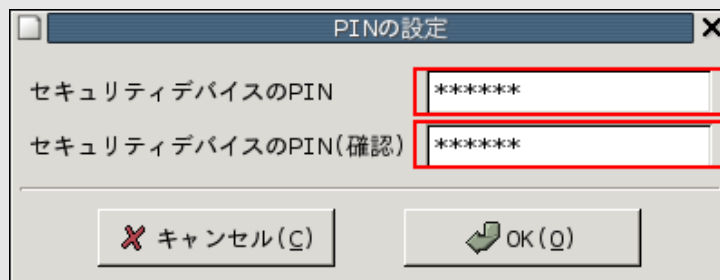


図 23:PINの設定

初期化に成功すると初期化完了ダイアログ(図 24:初期化完了ダイアログ)が表示されます。

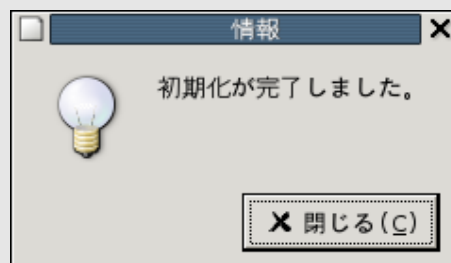


図 24:初期化完了ダイアログ

つづいてクライアント証明書のエクスポートを行います。

「証明書タブウィンドウ」に移動し、セキュリティデバイスに格納したいクライアント証明書のエントリの上で右クリックします。（図 25:セキュリティデバイスへの証明書イン

ポート)

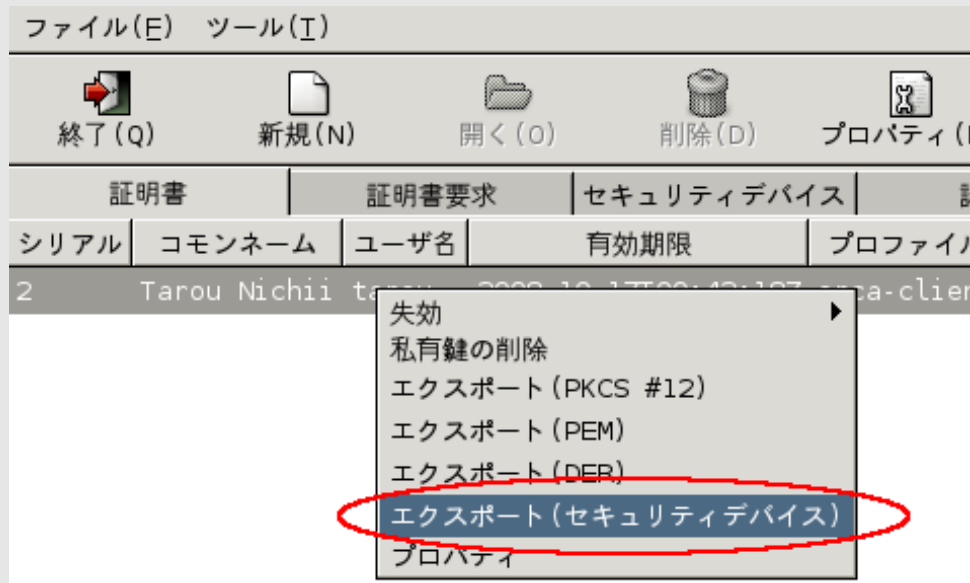


図 25:セキュリティデバイスへの証明書インポート

ポップアップウィンドウの「エクスポート (セキュリティデバイス)」を選択するとセキュリティデバイスのPINを入力する画面が表示されます。セキュリティデバイスの初期化時に設定したPINを入力して証明書をエクスポートします。

エクスポートに成功するとエクスポート成功ダイアログが表示されます(図 26:エクスポート成功ダイアログ)。

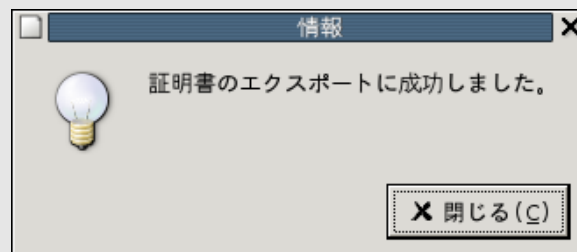


図 26:エクスポート成功ダイアログ

PINについて

セキュリティデバイスの使用時や初期化時にPINの入力が必要となります。PINを忘れた場合、セキュリティデバイスの使用ができなくなるので注意が必要です。

またPIN入力を連続して失敗するとPINがブロックし、以降正しいPINを入力してもデバイスの使用ができなくなる場合があります。詳しくは、文書「プライベートCA構築ツール」の「5.2.4 PINのブロック」を参照してください。

2.6 ユーザDBファイルの作成

ユーザDBファイルを作成し、「2.4.1 証明書のメディアへの格納」で作成したサーバ証明書が格納されたフロッピーディスクに保存します。

サーバ証明書が格納されたフロッピーディスクをマウントします。

```
$ mount /media/floppy
```

プライベート CA 構築ツールの「ツール」メニューを開き、「日レセ用のユーザDBを生成する」を選択します（図 27:日レセ用ユーザDBの作成）。

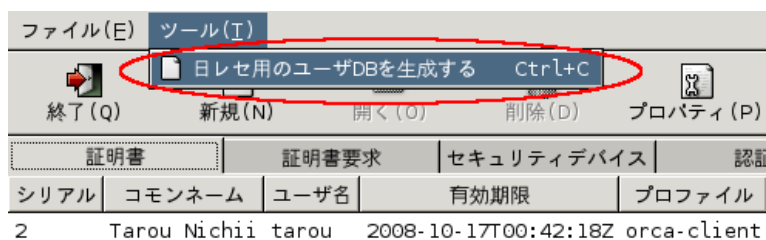


図 27:日レセ用ユーザDBの作成

「ファイルの保存」画面が表示されるので、/media/floppy/userdb として保存します。

保存後、フロッピーディスクをアンマウントします。

```
$ umount /media/floppy
```

3 サーバ設定

3.1 証明書とユーザDBファイルの設置

サーバ証明書、CA 証明書およびユーザDB ファイルをサーバ機に設置します。それぞれの設置先は以下です。

サーバ証明書

/etc/jma-receipt/glserver.p12

CA 証明書

/etc/ssl/certs/gl-cacert.pem

ユーザDB ファイル

/etc/jma-receipt/userdb

設置先は `jma-receipt` パッケージによってあらかじめ決められていますので、作業時には十分注意してください。

「2.3サーバ証明書の発行」で作成したフロッピーディスクをセットし、以下のコマンドを実行します。

```
$ mount /media/floppy
$ cd /media/floppy/
$ sudo cp gl-cacert.pem /etc/ssl/certs/gl-cacert.pem
$ sudo cp glserver.p12 \
  /etc/jma-receipt/glserver.p12
$ sudo chown -R orca:orca /etc/jma-receipt/glserver.p12
$ sudo chmod 400 /etc/jma-receipt/glserver.p12
$ sudo cp userdb /etc/jma-receipt/userdb
$ cd
$ umount /media/floppy
```

ユーザDB ファイルについて

日レセのクライアント認証では、従来の `glauth` サーバを利用せず、ユーザDB ファイルを利用します。ユーザDB ファイルには証明書のDNとユーザの組が記載されており、接続に使用された証明書と記載されたDNが一致した場合に対応するユーザとして認証される仕組みになっています。

ユーザの追加、削除、変更を行う場合は、ユーザ DB ファイルの再生成と置き換えが必要となります。

3.2 jma-receipt パッケージの再設定

jma-receipt パッケージを再設定します。以下のコマンドを実行します。

```
# sudo /usr/sbin/dpkg-reconfigure jma-receipt
```

入院版の場合は以下のコマンドを実行してください。

```
# sudo /usr/sbin/dpkg-reconfigure jma-receipt-hosp
```

SSL を有効にするか問い合わせる画面(図 28:SSL 設定画面)が表示されるまで設定画面のデフォルトの値を変更せずに作業を進めます。

「SSL を有効にしますか」には「はい」を選択し、その後の画面では表示されるデフォルトの設定を選択して設定画面を終了します。

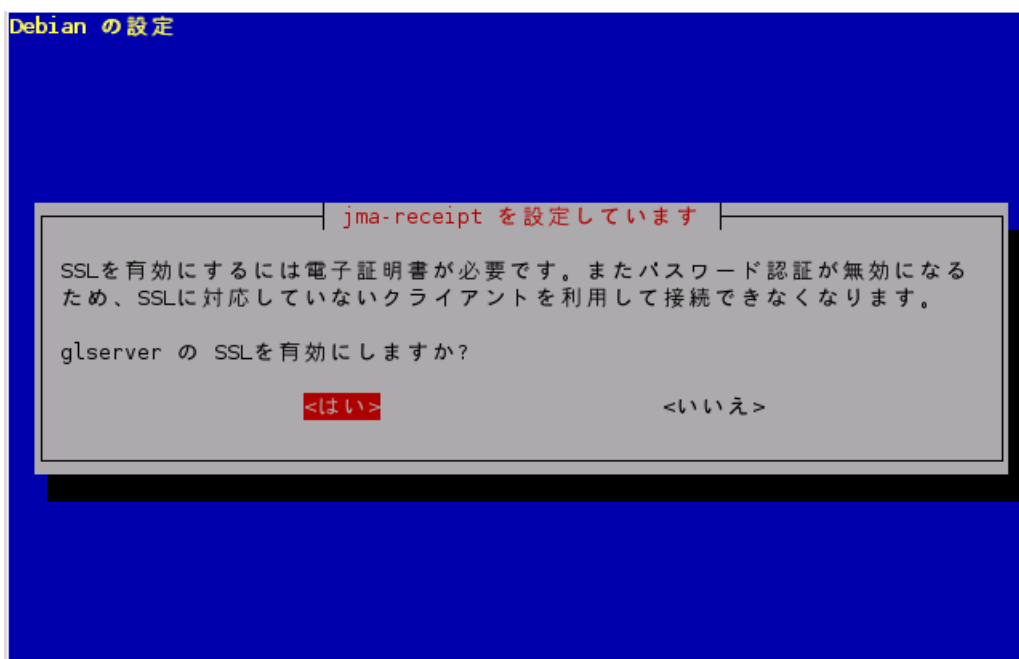


図 28:SSL 設定画面

設定画面で「デーモンを起動しない」と設定した場合は、日レセが停止しているため以下のコマンドで日レセを起動します。

```
$ sudo /etc/init.d/jma-receipt start
```

4 glclient 設定

4.1 パスワード入力ダイアログの設定

glclient 起動時にクライアント証明書のパスワードをダイアログから入力できるようにするため、ssh-askpass-gnome パッケージをインストールします。

```
$ sudo aptitude update
$ sudo aptitude install ssh-askpass-gnome
```

4.2 クライアント証明書を機材に直接保存する場合

4.2.1 証明書の設置

「2.4 クライアント証明書の発行」で作成したフロッピーディスクからクライアント証明書と CA 証明書を取り出し適切な場所にコピーします。

フロッピーディスクをマウントします。

```
$ mount /media/floppy/
```

CA 証明書のインストール

CA 証明書を/etc/ssl/certs ディレクトリにコピーします。

サーバと同一機材を使用する場合は、この作業は不要です。

```
$ sudo cp /media/floppy/gl-cacert.pem /etc/ssl/certs/
```

クライアント証明書のインストール

クライアント証明書を日レセクライアント利用者の UNIX アカウントのホームディレクトリにコピーします。

ここでは例として、UNIX アカウントが **tarou** で、クライアント証明書が **tarou.p12** というファイル名でフロッピーディスクに格納されているものとします。適宜読みかえて作

業を行ってください。

```
$ sudo mkdir -p ~tarou/.glclient
$ sudo cp /media/floppy/tarou.p12 \
  ~tarou/.glclient/gl-client.p12
```

他のユーザがアクセスできないようクライアント証明書の権限を変更します。

```
$ sudo chown -R tarou:tarou ~tarou/.glclient/
$ sudo chmod 500 ~tarou/.glclient/
$ sudo chmod 400 ~tarou/.glclient/gl-client.p12
```

フロッピーディスクをアンマウントします。

```
$ umount /media/floppy/
```

4.2.2 コマンドラインからのクライアントの起動

日レセクライアント利用者のUNIXアカウントでシステムにログインし、glserverに接続できることを確認します。

コンソールから以下のコマンドを実行します。

-port オプションにはサーバ証明書の共通ネームに設定したホスト名を指定します。

ここでは例として `main.example.or.jp` を指定しています。適宜読みかえてコマンドを実行してください。

```
$ glclient -port main.example.or.jp:8000 \
  -ssl \
  -cert ~/.glclient/gl-client.p12 \
  -CAfile /etc/ssl/certs/gl-cacert.pem \
  panda:orca00
```

クライアント証明書にパスワードが設定されている場合、パスワード入力ダイアログ(☒

29:パスワード入力ダイアログが表示されるのでパスワードを入力します。

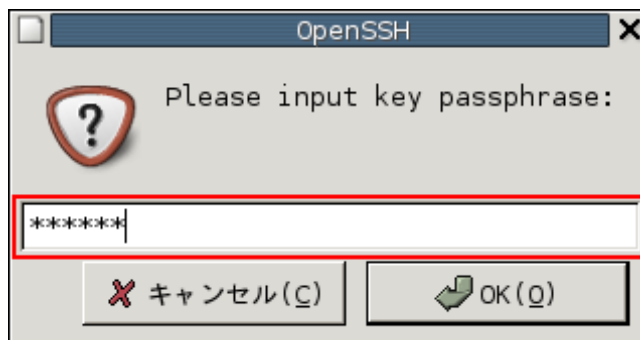


図 29:パスワード入力ダイアログ

日レセの画面が表示されたらクライアント認証に成功しています。

失敗する場合は、コンソールに表示されるエラーメッセージを確認してください。

4.2.3 ダイアログ画面からのクライアントの起動

glclient コマンドの引数に-diallog をつけて起動します。

```
$ glclient -diallog
```

glclient ランチャーが起動したら「基本」タブの「ホスト(ポート)」にサーバ証明書の
コモンネームに設定したホスト名を指定します。また「SSLを使う」にチェックを入れます。

ここでは例として、ホスト名に main.example.or.jp を指定しています。適宜読みかえて
コマンドを実行してください。

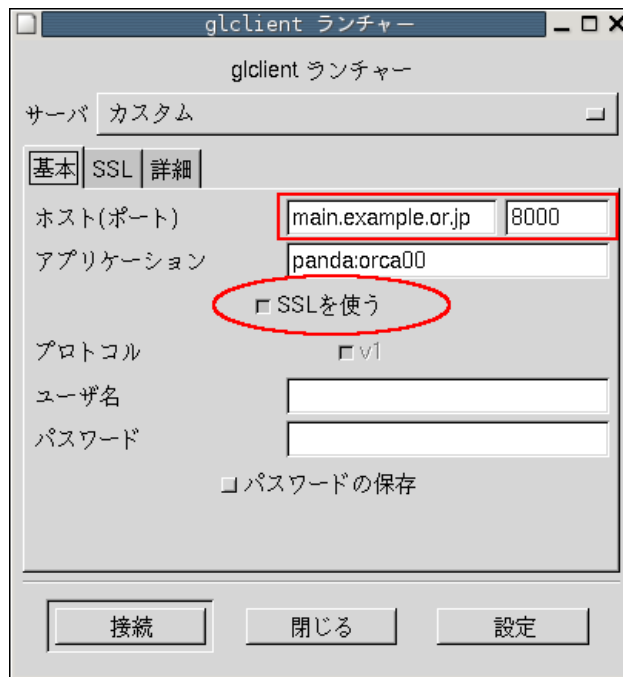


図 30:基本タブ(glclient ランチャー)

「SSL」タブを開き、参照ボタンを利用してCA証明書とクライアント証明書のパスを追加します(図 31:SSLタブ(glclient ランチャー))。

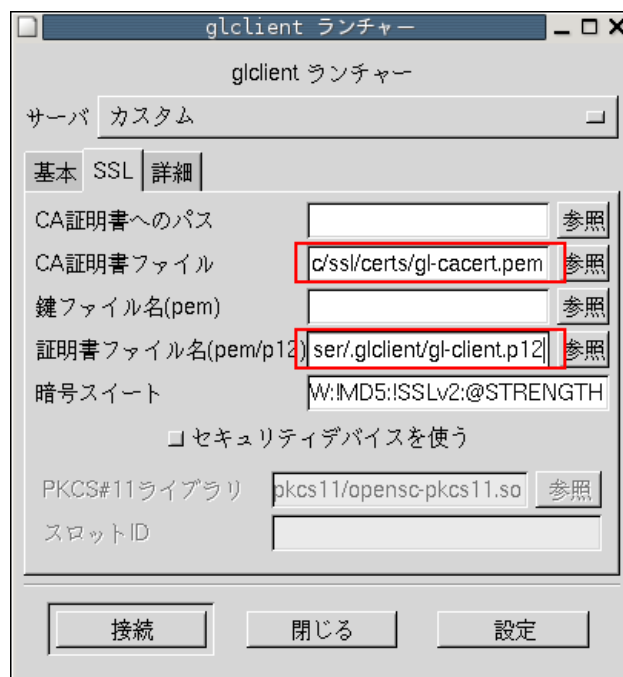


図 31:SSLタブ(glclient ランチャー)

設定が完了したら、「接続」ボタンを押下します。クライアント証明書にパスワードが設定されている場合、パスワード入力ダイアログ(図 29:パスワード入力ダイアログ)が表示されるのでパスワードを入力します。

日レセの画面が表示されたらクライアント認証に成功しています。
失敗する場合は、コンソールに表示されるエラーメッセージを確認してください。

4.3 セキュリティデバイスを利用する場合

4.3.1 CA 証明書の設置

CA 証明書を格納したフロッピーディスクを使用し、CA 証明書を/etc/ssl/certs ディレクトリにコピーします。

サーバと同一機材を使用する場合は、この作業は不要です。

```
$ mount /media/floppy/  
$ sudo cp /media/floppy/gl-cacert.pem /etc/ssl/certs/  
$ umount /media/floppy/
```

4.3.2 セキュリティデバイス利用の準備

セキュリティデバイスを利用するために機材の設定を行います。まずセキュリティデバイスの使用に必要なパッケージをインストールします。

使用している機材の Debian バージョンによりインストールするパッケージが異なるので注意が必要です。

Debian GNU/Linux 4.0 etch の場合

```
$ sudo aptitude update  
$ sudo aptitude install openssl openct pscsd libccid \  
libengine-pkcs11-openssl
```

Debian GNU/Linux 3.1 Sarge の場合

```
$ sudo aptitude update  
$ sudo aptitude install openssl openct pscsd libccid \  
libopenssl-openssl
```

必要なパッケージをインストール後、セキュリティデバイスの設定を行います。セキュリティデバイスの設定については文書「プライベート CA 構築ツールの利用」の「2.1 セキュリティデバイスの環境設定」を参照してください。

4.3.3 コマンドラインからのクライアントの起動

日レセクライアント利用者の UNIX アカウントでシステムにログインし、glserver に接続

できることを確認します。

コンソールから以下のコマンドを実行します。

- `-port` オプションにはサーバ証明書の共通ネームに設定したホスト名を指定します。ここでは例として `main.example.or.jp` を指定しています。適宜読みかえてコマンドを実行してください。
- Debian のバージョンによって `-pkcs11_lib` オプションで指定する PKCS#11 ライブラリのパスが異なります。

Debian GNU/Linux 4.0 etch の場合

```
$ glclient -port main.example.or.jp:8000 \  
-ssl \  
-CAfile /etc/ssl/certs/gl-cacert.pem \  
-pkcs11 \  
-pkcs11_lib /usr/lib/opensc/opensc-pkcs11.so \  
panda:orca00
```

Debian GNU/Linux 3.1 Sarge の場合

```
$ glclient -port main.example.or.jp:8000 \  
-ssl \  
-CAfile /etc/ssl/certs/gl-cacert.pem \  
-pkcs11 \  
-pkcs11_lib /usr/lib/pkcs11/opensc-pkcs11.so \  
panda:orca00
```

コマンド実行後、PIN 入力ダイアログ(図 32:PIN 入力ダイアログ)が表示されるので PIN を入力します。



図 32:PIN 入力ダイアログ

日レセの画面が表示されたらクライアント認証に成功しています。

失敗する場合は、コンソールに表示されるエラーメッセージを確認してください。

4.3.4 ダイアログ画面からのクライアントの起動

glclient コマンドの引数に-dialoagをつけて起動します。

```
$ glclient -dialoag
```

glclient ランチャーが起動したら「基本」タブの「ホスト(ポート)」にサーバ証明書の
コモンネームに設定したホスト名を指定します。また「SSLを使う」にチェックを入れます。

ここでは例として、ホスト名に **main.example.or.jp** を指定しています。適宜読みかえて
コマンドを実行してください。

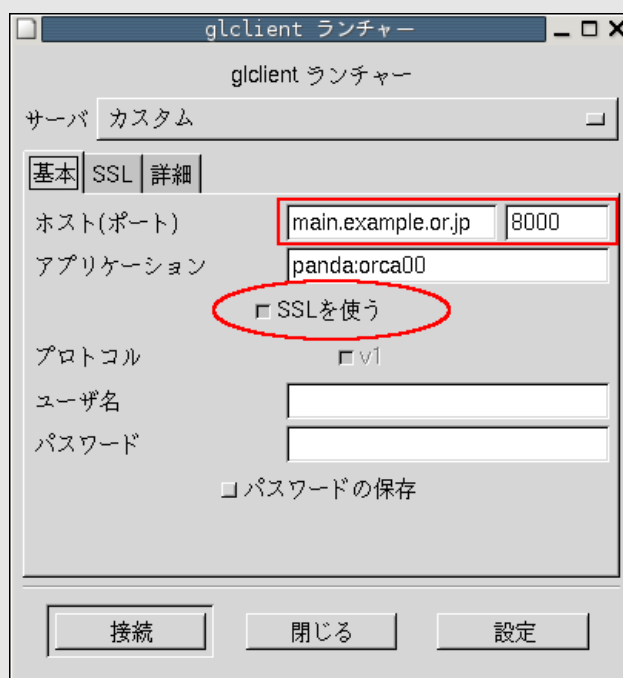


図 33:基本タブ(glclient ランチャー)

「SSL」タブを開き、参照ボタンを利用してCA証明書のパスを追加します。また「セキュ
リティデバイスを使う」にチェックを入れます(図 34:SSLタブ(glclient ランチャー))。

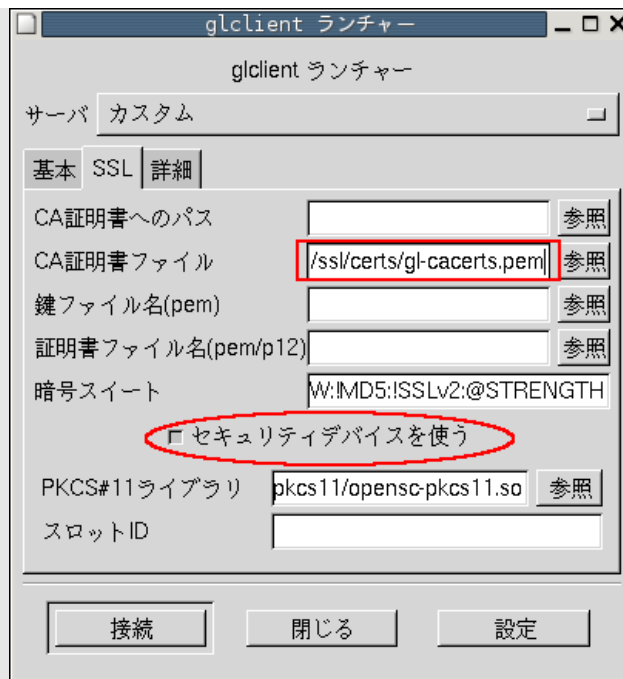


図 34:SSL タブ(glclient ランチャー)

設定が完了したら、「接続」ボタンを押下するとPINの入力画面(図 32:PIN入力ダイアログ)が表示されるのでPINを入力します。

日レセの画面が表示されたらクライアント認証に成功しています。

失敗する場合は、コンソールに表示されるエラーメッセージを確認してください。

5 monsiaj 設定

5.1 Linux 上での証明書の設定と接続

5.1.1 証明書の設置

クライアント証明書が格納されたフロッピーディスクを機材にセットし、マウントします。

```
$ mount /media/floppy/
```

CA 証明書のインストール

1. 「Java Control Panel」を起動します。

Java を etch の sun-java5-jre パッケージからインストールした場合は以下のコマンドを実行してください。

```
$ /usr/bin/ControlPanel
```

手動でインストールした場合は、インストール先の ControlPanel コマンドを実行してください。

```
$ <Java のインストール先>/bin/ControlPanel
```

2. 「Java Control Panel」が開いたら、「セキュリティ」タブをクリックします。
3. 「証明書」ボタンをクリックし、「証明書」ウィンドウを開きます。
4. 「証明書タイプ」で「セキュアサイト CA」を選択し、「ユーザ」タブを選択します。
5. 「インポート」ボタンをクリックしてファイルの選択画面を開き、「ファイルタイプ」を「AllFiles」に変更します。
6. CA 証明書(g1-cacert.pem)を選択して「開く」ボタンをクリックして画面を閉じま

す。

7. 「ユーザ」タブの「セキュアサイト CA」の中に証明書エントリが増えていることを確認し、画面を閉じます。

クライアント証明書のインストール

クライアント証明書を日レセクライアント利用者のUNIXアカウントのホームディレクトリにコピーします。

ここでは例として、UNIXアカウントが **tarou** で、クライアント証明書が **tarou.p12** というファイル名でフロッピーディスクに格納されているものとします。適宜読みかえて作業を行ってください。

```
$ sudo mkdir -p ~tarou/.glclient
$ sudo cp /media/floppy/tarou.p12 \
  ~tarou/.glclient/gl-client.p12
```

クライアント証明書はログインしたアカウントで利用するため、他のユーザに利用されないよう権限を変更します。

```
$ sudo chown -R tarou:tarou ~tarou/.glclient/
$ sudo chmod 500 ~tarou/.glclient/
$ sudo chmod 400 ~tarou/.glclient/gl-client.p12
```

フロッピーディスクをアンマウントします。

```
$ umount /media/floppy/
```

5.1.2 クライアントの起動

1. monsiaj のフォルダを開き、jmareceipt.jar をダブルクリックします。(あるいは日医標準レセプトソフトを試してみる (<http://www.orca.med.or.jp/receipt/trial/jws.rhtml>)の「glclient / Java JWS Start!」をクリックします)。
2. ログイン画面が表示されたら、「基本設定」タブの「ホスト名」に接続先となる glserver ホスト名を入力します。「glserver ホスト名」はサーバ証明書の共通

ネームで設定した値を指定する必要があります。

3. 「SSL 設定」タブに移動し、「SSL を使用」にチェックを入れて「クライアント証明書」ボックスにクライアント証明書のパスを指定します(図 35:monsi aj の SSL 設定画面(Linux, Windows, MacOS X 共通))。

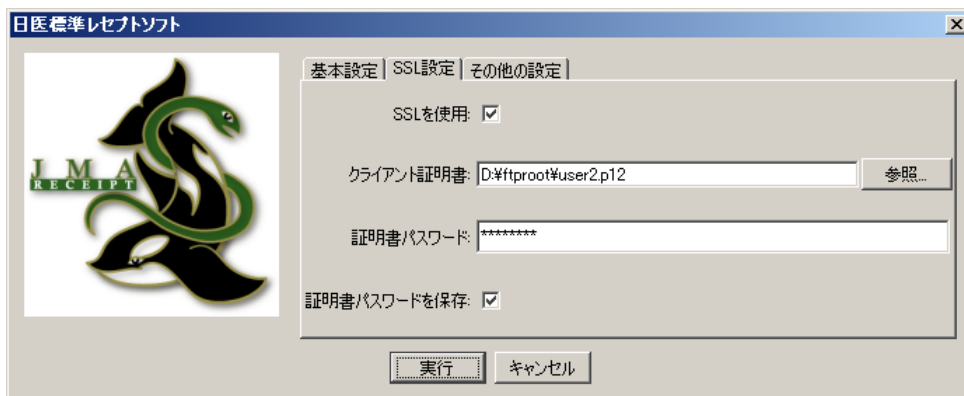


図 35:monsi aj の SSL 設定画面(Linux, Windows, MacOS X 共通)

4. PKCS#12 ファイルのパスワードを「パスワード」ボックスに入力します。
 5. 「実行」ボタンをクリックし、起動することを確認します。
- パスワードを保存する場合は、「基本設定」タブの「パスワードを保存」チェックボックスにチェックします。
 - パスワードが設定されていないPKCS#12 ファイルは利用できません。
 - 5.の「実行」ボタン押下後、「図 36:サーバ証明書検証失敗ダイアログ」が表示された場合は、CA 証明書のインストールが正常に行われていません。「中止する」を選択して monsi aj を終了し、CA 証明書がインストールされているかどうか確認してください。

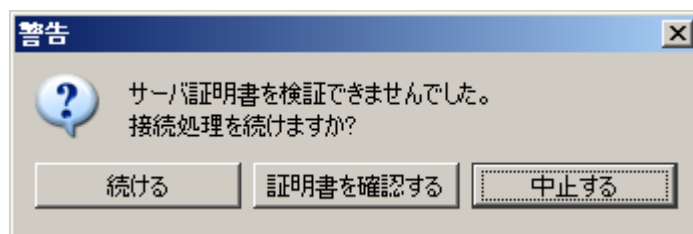


図 36:サーバ証明書検証失敗ダイアログ

5.2 Windows 上での証明書の設定と接続

5.2.1 証明書の設置

フロッピーディスクのクライアント証明書、CA 証明書を「マイドキュメント」にコピーします。

CA 証明書のインストール

1. 「Java」コントロールパネルをダブルクリックして開きます。
2. 「セキュリティ」タブをクリックします。
3. 「証明書」ボタンをクリックし、「証明書」ウィンドウを開きます。
4. 「証明書タイプ」で「セキュアサイト CA」を選択し、「ユーザ」タブを選択します
5. 「インポート」ボタンをクリックしてファイルの選択画面を開き、「ファイルタイプ」を「AllFiles」に変更します。
6. CA 証明書(gl-cacert.pem)を選択して「開く」ボタンをクリックして画面を閉じます。
7. 「ユーザ」タブの中に証明書エントリが増えていることを確認し、画面を閉じます。
8. 「Java」コントロールパネルを終了します。

5.2.2 クライアントの起動

1. monsiaj のフォルダを開き、jmareceipt.jar をダブルクリックします。(あるいは日医標準レセプトソフトを試してみる
(<http://www.orca.med.or.jp/receipt/trial/jws.rhtml>)の「 glclient / Java JWS Start! 」をクリックします)。
2. ログイン画面が表示されたら、「基本設定」タブの「ホスト名」に接続先の glserver ホスト 名を入力します。「glserver ホスト名」はサーバ証明書のCOMMONで設定した値を指定する必要があります。

3. 「SSL 設定」タブに移動し、「SSL を使用」にチェックを入れて「クライアント証明書」ボックスにクライアント証明書のパスを指定します(図 35:monsiaj の SSL 設定画面(Linux, Windows, MacOS X 共通))。
 4. PKCS#12 ファイルのパスワードを「パスワード」ボックスに入力します。
 5. 「実行」ボタンをクリックし、起動することを確認します。
- パスワードを保存する場合は、「基本設定」タブの「パスワードを保存」チェックボックスにチェックします。
 - パスワードが設定されていないPKCS#12 ファイルは利用できません。
 - 5.の「実行」ボタン押下後、「図 36:サーバ証明書検証失敗ダイアログ」が表示された場合は、CA 証明書のインストールが正常に行われていません。「中止する」を選択して monsiaj を終了し、CA 証明書がインストールされているかどうか確認してください。

5.3 MacOS X 10.4 上での証明書の設置と接続

5.3.1 証明書の設置

フロッピーディスクのクライアント証明書、CA 証明書を日レセクライアント利用者アカウントの Desktop フォルダにコピーします。

CA 証明書のインストール

1. 「Finder」を開き、サイドバーの「アプリケーション」をクリックします。
2. ウィンドウで「ユーティリティ」-「Java」の順に開きます。
3. 「java 1.4.2 プラグイン設定」をダブルクリックし、「Java Plug-in コントロールパネル」が開いたら、「証明書」パネルを開きます。
4. 「署名済みアプレット」を選択し、「インポート」ボタンをクリックします。
5. 「開く」ウィンドウが表示されたら「デスクトップ」をダブルクリックして開き、「フォーマット」を「すべてのファイル」に変更します。

6. ウィンドウから CA 証明書ファイル(gl-cacert.pem)を選択して「開く」ボタンをクリックし、「署名済みアプレット」画面に証明書が追加されたことを確認します。
7. 「Java Plug-in コントロールパネル」を閉じます。

5.3.2 クライアントの起動

1. monsiaj のフォルダを開き、jmareceipt.jar をダブルクリックします。(あるいは日医標準レセプトソフトを試してみる
(<http://www.orca.med.or.jp/receipt/trial/jws.rhtml>)の「 glclient / Java JWS Start! 」をクリックします)。
2. ログイン画面が表示されたら、「基本設定」タブの「ホスト名」に接続先の glserver ホスト名を入力します。「glserver ホスト名」はサーバ証明書の共通ネームで設定した値を指定する必要があります。
3. 「SSL 設定」タブに移動し、「SSL を使用」にチェックを入れて「クライアント証明書」ボックスにクライアント証明書のパスを指定します。(図 35:monsiaj の SSL 設定画面(Linux, Windows, MacOS X 共通))
4. PKCS#12 ファイルのパスワードを「パスワード」ボックスに入力します。
5. 「実行」ボタンをクリックし、起動することを確認します。
 - パスワードを保存する場合は、「基本設定」タブの「パスワードを保存」チェックボックスにチェックします。
 - パスワードが設定されていないPKCS#12 ファイルは利用できません。
 - 5.の「実行」ボタン押下後、「図 36:サーバ証明書検証失敗ダイアログ」が表示された場合は、CA 証明書のインストールが正常に行われていません。「中止する」を選択して monsiaj を終了し、CA 証明書がインストールされているかどうか確認してください。