ICカードファイルフォーマット設定ソフトウェアの自動生成

5T－2

太田 英憲、榊原 裕之、齋藤 和美、辻 宏郷
三菱電機（株）情報技術総合研究所

1　はじめに

我々はPKI (Public Key Infrastructure)で必要になる各種機能を実装したPKI暗号ライブラリを開発しており[2]、鍵管理デバイスとしてISO 7816及びJICSAP準拠[1]のICカード、ICカードを用いた暗号ライブラリのAPIにPKCS #11[3]を採用している。ICカードを使用するためには、カード内部のEEPROMに適切な初期情報を書き込んでおく必要があるが、その内容や構造は使用するアプリケーション毎に異なる。従来、情報を書き込むためにはICカードのドライバを直接呼び出す必要があったが、ドライバに依存しない設定ファイルを元に、容易にICカードファイルフォーマット設定ソフトウェアを自動生成することを可能にした。本稿では、自動生成ツールの内容と利点について報告する。

2　ファイルフォーマット設定ソフトウェア

2.1 構成

我々はWindows上のDLLとして、ICカードファイルフォーマット設定ソフトウェア(以下、フォーマッタ)の開発を行なった。フォーマッタは、以下の要素から構成される。

- 共通コンポーネント
- ユーザ定義コンポーネント

共通コンポーネントは、ICカードの活性化、ドライバのロードやICカードの有無のチェック等、ICカードに書き込まれる情報には依存しない機能を持っている。また、ICカードのフォーマットを行なうための関数のAPIは、このコンポーネント中に含まれている。

ユーザ定義コンポーネントは、実際にフォーマットを行なう機能を含んでおり、ここに定義されている情報に基づいて、ICカードのファイル構造を決定する。共通コンポーネント中のフォーマット関数は、呼び出されると、カードの活性化等、必要な前処理を行なった後に、ユーザ定義コンポーネント中のフォーマット機能を実行する。

2.2 問題点及び解決策

ICカードに格納される情報は、アプリケーション毎に異なるため、内部のファイル構造を変更する必要がある。ICカードを複数のアプリケーションから使用する場合には、それぞれに対応したファイル構造になっている必要がある。また、ICカードを同一のアプリケーションから使用する場合でも、パスワードの入力を何回間違えたときに閉塞するのか、閉塞したカードを解除する権限は誰に与えるか等、セキュリティ要件に応じて、同一のファイル構造に対しても、それぞれのファイルの属性を変更する必要がある。

以上のように、ユーザ定義コンポーネントは用途に応じて修正する必要がある。しかし、ICカードドライバを直接呼び出すプログラムを記述するためには、ドライバ毎の詳細な知識が必要であり、コーディングミスによっては、セキュリティホールを生じる危険がある。したがって、アプリケーション毎に、ユーザコンポーネントのプログラムを記述する代わりに、より簡単にユーザ定義コンポーネントを記述する方法が必要であった。そこで、ドライバ依存の知識なしに記述可能な設定ファイルの仕様を考案し、ユーザ定義コンポーネントに自動的に変換するツールを開発した。

Auto-Generation of the File Format Configuration Software for IC Card
Hidenori OHTA, Hiroyuki SAKAKIBARA, Kazumi SAITO, Hirosato TSUJI
Information Technology R&D Center, Mitsubishi Electric Corporation

3　DLLの作成

3.1 基本機能

設定ファイルには、ICカードに書き込みを行なうためのコマンドの記述を行なう。コマンドの種類として、通常のファイルシステムでのディレクトリにあたるDF、オブジェクトを格納するためのWEF、認証キーを格納するためのIEFの作成や、IEFの認証、DFの移動、ICカードの初期化等、フォーマットを行なう上で必要な機能を定義している。また、設定ファイルには、使用するICカードドライバの種類を指定可能とし、変換ツールでは、指定されたドライバの種類に応じて、そのドライバ用のソースファイルを生成することにより、複数のドライバへの対応を図っている。

3.2 補助機能

フォーマッタは、設定ファイルが異なっていても同一のAPIにより構成されているため、そのままでは、DLLを作成した後では設定ファイルの違いを区別することができない。そこで、設定ファイルにユーザが定義するバージョン情報を記述可能とし、実行時に、その情報を取得することで、違いを区別する。また、生成されるソースファイル中にコメントとして、設定ファイル名と変換された時刻情報を記述し、設定ファイルとソースファイルとの対応付けを可能としている。さらに、作成されるDLLには影響しないコメントを設定ファイル中に記述可能である。

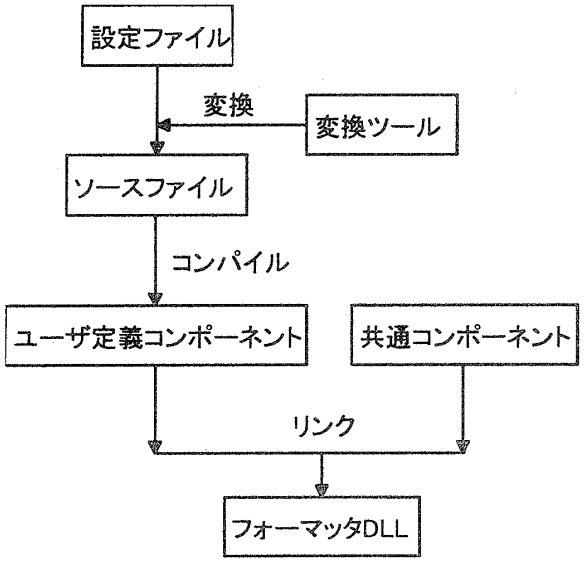

図 1: フォーマッタ作成手順

3.3 変換

変換ツールは、このようにして記述された設定ファイルを元にソースファイルを生成し、このソースファイルを共通コンポーネントと共に、コンパイル及びリンクを行なうことで、DLLを作成する。(図 1)

4　適用例

本変換ツールを使って、PKCS #11用ICカードのフォーマッタの生成を行なった。各種オブジェクトを格納するためのDFや、SOとUSERの認証を行なうためのIEF等含めて10個以上のファイルをICカード中に作成するように、設定ファイルの記述を行なう必要があり、コメントを含めて、200行程度になっている。この設定ファイルをソースファイルに変換すると、600行になった。また、アプリケーション毎に設定ファイルの変更が必要な部分は、200行中1、2行であったのに対して、ソースファイルの変更箇所は10行以上に及んだ。従来、ソースファイルを直接変更していたことに比べ、アプリケーション毎に対応するフォーマッタの作成が効率良く行なえるようになった。

5　まとめ

PKIでICカードを使うためのフォーマッタを、ドライバに依存しない設定ファイルから生成するための変換ツールを開発した。今後は、業界標準のドライバAPIに変換することにより、より多くのICカードをサポートしていく予定である。

参考文献

[1] IC カードシステム利用促進協議会, "JICSAP 外部端子付きICカード仕様," 1998

[2] 辻・榊原・齋藤・太田, "PKI暗号ライブラリにおけるICカードの利用(1)—概要—," 情報処理学会第58回全国大会 2L-04, 1999

[3] RSA Laboratories, "PKCS #11 Cryptographic Token Interface Standard," 1997