Java Card のセキュリティ
Security Technologies of Java Card

酒井　高彦
SAKAI Takahiko

IC カード上に Java(注1)動作環境を構築し，Java 言語で書かれたアプリケーション プログラム(以下，アプリケーションと略記)をこの上で動作させる Java Card と呼ばれる新しい考えかたに基づいたカードが市場に登場し始めており，この Java Card は，次世代の IC カードと目されている。当社では，Java Card 2.1 仕様に基づくカードを開発中で，共有データに安全面でどのような配慮をするかがポイントとなる。搭載したアプリケーションを変更できない従来のカードとは異なり，このカードは，搭載したアプリケーションを変更することが可能である。

The Java Card is now emerging in the market. The Java Card is implemented based on the concept that applications described in Java language operate in a Java runtime environment established in an IC card. Unlike existing cards, the Java Card can modify loaded applications.

This paper provides an overview of the Java Card, which is considered to be the next-generation IC card, and describes some of the security technologies of this card.

1 まえがき

キャッシュカードあるいはクレジットカードと同じ大きさのカードに IC チップを埋め込んだカードが IC カードである。近年，電子マネー実験などに使われ始めて注目を浴びている。従来の IC カードは，IC チップ内のマスク ROM にアプリケーションが搭載されているため，チップ製造後にアプリケーションを変更することができなかった。近年，製造後でもアプリケーションの追加や削除を可能とする方式が提案されてきている。当社でもこのような機能を実現する技術である Java Card 仕様を採用し，試作を実施した。これが図1に示すサンプルカード“JMAGIC™”である。従来から IC カードに搭載されているセキュリティ機能に加えて，製造後にアプリケーションを追加・削除する際の安全性を確保するためのセキュリティ機能が検討されている。ここでは，Java Card の概要を紹介するとともに，Java Card の試作に搭載したセキュリティ技術と検討された内容について述べる。

図1．当社 Java Card サンプル“JMAGIC™”　当社 Java Card サンプルの外観と名称を示す。
Sample of Toshiba JMAGIC™ prototype Java Card

2 Java Card の概要

IC カードには通常一つの IC チップが埋め込まれている。図2に示すように，チップ内部は主に CPU と3種類のメモリから構成されている。CPU は，現在8ビットのものが主流であるが，16ビットのものが出始め将来は32ビットのものも検討されている。メモリは，作業領域である RAM，制御コードを搭載する ROM，データの書換えが可能な不揮発性メモリの3種類である。不揮発性メモリ技術は，現在は EEPROM(Electrically Erasable and Programable ROM) 技術が主流であるが，フラッシュメモリ・強誘電体素子技術の利用も検討されている。

図3には，従来の IC カードと Java Card の相違をメモリの使いかたの違いで示している。両者とも COS と称されるカード オペレーティングシステムを ROM に搭載している。COS は，カード起動時の処理，端末とカード間の通信の管理，メモリ管理などを実施している。従来のカードは，ROM にアプリケーションを搭載し，EEPROM

(注1)　Java ならびにその他の Java を含む商標は，米国 SunMicrosystems 社の商標。

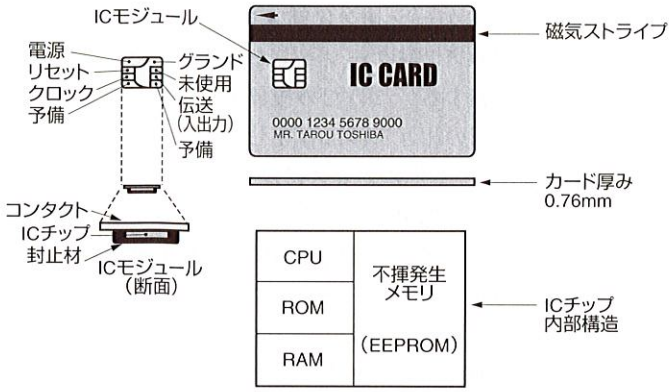

図2．ICカードの構造　ICカードの構造と埋め込まれている半導体チップ内部の主要構成要素であるCPUおよび各種メモリを示す。

Components of IC card chip

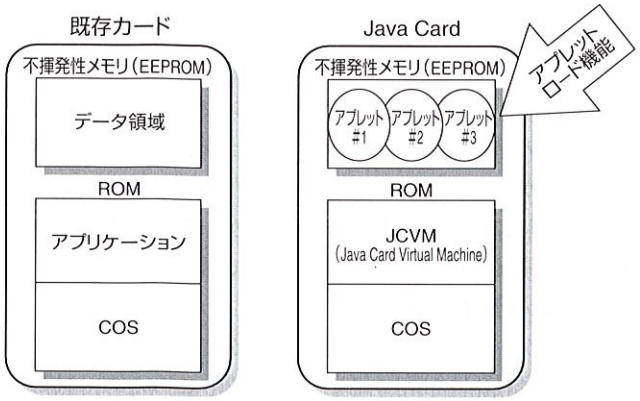

図3．既存カードとJava Cardの違い　ICカード用半導体チップのメモリの使いかたから，既存カードとJava Cardの違いを示す。

Comparison of existing card and Java Card

をデータ領域として利用している。これに対しJava Cardは，ROMにはアプリケーションを搭載せずにJavaの実行環境を搭載してEEPROMに搭載されるアプリケーションをこの実行環境で動作させている。Javaはオブジェクト指向言語であるため，データとこれを処理する手続きを一体化させたものがアプリケーションである。Java Cardの場合，カードに搭載するデータと手続きが一体化されたアプリケーションをアプレットと称している。Java Cardでは，書換え可能な不揮発性メモリであるEEPROMにアプレットを搭載するため，アプリケーションの追加や削除が可能となっている。

図4は，アプレットの開発とカードへの搭載までの手順を示している。標準のJava環境ではアプリケーションを標準のコンパイラ(注2)で処理した結果を端末上で動作させている。ICカードのハードウェアリソースは前述のように限られているため，標準のコンパイル処理の後カード上

(注2)　プログラミング言語で作成されたプログラムを，コンピュータが理解できる機械語に一括変換するするソフトウェア。

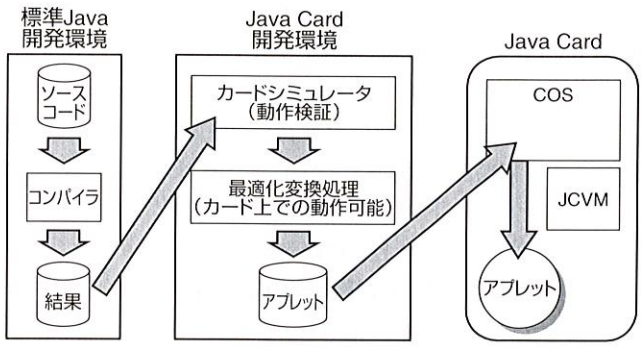

図4．アプリケーション開発から搭載まで　標準Java開発環境を使ってのアプリケーション開発から，これをカードに搭載するまでの開発過程を示す。

From application development to downloading

で動作可能な形に最適化するための変換処理を実施し，この結果をカードにロードして動作させている。

3 Java Cardのセキュリティ

Java Cardも当然ながら従来のICカードが採用していたセキュリティ機能を搭載している。ここでは，Java Card特有のセキュリティ機能について述べる。Java Cardのセキュリティ上の課題は，アプリケーションの追加や削除を安全に実施させることと，標準Javaのセキュリティ機能をどのように前述のような限られたハードウェアリソース上に実現させるかの二点である。また，限定されたハードウェアリソースのためセキュリティ機能の搭載に際しては，カード単体ではなく図4に示した開発環境と協調しJava Cardシステムとして実現している。

3.1 アプリケーション開発環境での検査

Javaは仮想マシン上でプログラムを逐次解釈実行させる方式を採用している。Java Cardも同様であり，図3に示したJava実行環境にはJava Card用の仮想マシンが搭載されている。標準Javaでは，プログラムを逐次解釈する際に強力な各種の検証を実施している。例えば，取り扱うデータの型検証などである。検証内容は多岐にわたるため，この検証に要するコード量はきわめて大きくカードには搭載できない。このためカード外部の開発環境でこの検証を実施させ，カード内部では，簡単な項目だけの検証にとどめている。

開発環境には，Java言語で書かれたプログラムが動作するカードシミュレータが組み込まれている。前述の検証は，このシミュレータ動作時に実施される。シミュレータを利用した動作検証の十分性を計る手段として，用意したテストデータが分岐を含めたプログラムの処理経路を，どの程度通ったかを示すツールも用意されている。

ハードウェアリソースが限定されるため，Java Cardは標準Javaに対して種々の制約事項を設定している。図3に示したように開発環境は標準Javaのコンパイル結果を入力としている。開発環境の最初の処理は，制約事項の検証である。同時に正当なコンパイラの出力であることを確認して不正なコードの侵入を阻止している。この後，シミュレータ利用の検証へと処理が進められる。

3.2 アプリケーション搭載時の安全性

従来のカードはマスクROMにアプリケーションが搭載されているため，製造後はアプリケーションの改変が不可能であった。Java Cardは製造後でもアプリケーションの追加や削除ができるため，この際の安全性が強く要望される。安全性の主な要件は，アプリケーションの追加や削除の権限が確認できること，搭載されるアプリケーションに不正なコードが紛れ込んでいないこと，搭載されるまでにアプリケーションが不正な改変がされていないこと，アプリケーションの内容が途中で盗まれないことである。また，用意されたアプリケーションが確実にカード内に搭載されることも求められる。

不正の排除としては，開発環境側でアプリケーションに署名や認証を付加してカード内部でこれを確認することで実現している。盗用の防止のためには，アプリケーションを開発環境で暗号化してカードに送り込み，カード内部で復号化して搭載することで対応している。

ICカードは，ICチップ一つだけで構成される製品であり電源は搭載されていない。このICチップを動作させる電源やクロックは外部から供給される。したがって，例えば端末からICカードが動作中に引き抜かれるなどで電源やクロックの供給が中断される事態が発生する。アプリケーション搭載時にこのような事態が発生するとカード内に中途半端なアプリケーションデータが存在することになる。この状態を解消するために，カードが起動される際に搭載途中のアプリケーションの存在の有無を確かめ，存在する場合にはそのデータをすべて削除する方式をとっている。

3.3 複数アプリケーションの相互不干渉

Java Cardでは複数のアプリケーションを搭載してこれを切り換えて使用することができる。複数のアプリケーションが一枚のカードに搭載された際には，アプリケーション間の相互不干渉が安全面から要求される。この相互不干渉はJava仕様が兼ね備えている機能で実現できる。Java Cardでは，この部分のJava仕様をそのままカード上に搭載することで対応している。

Java Cardでは動作するアプリケーションはつねに一つである。動作させるアプリケーションの選択を実施してから，このアプリケーションに端末側から処理が指示される。その後，別のアプリケーションを実施させたい場合は，再度選択を実施する。このため複数アプリケーションが同時に動作する際の相互干渉は存在しない。

Java言語はデータアクセスに際してポインタを使用せず名前参照方式を採用している。このためポインタの計算誤りによる不正アクセスは言語仕様上不可能である。また，第三者のサブルーチンを使用する際にもその実体を直接使用せず，自分用に新たな実体を作成して使用している。このため他人のデータを直接アクセスすることがないため，不正アクセスが防止できる。

3.4 メモリ管理

Java CardのアプリケーションはEEPROM上にデータ領域も含めて搭載され，削除の要求があるまでカード上に存在し続ける。これとは別にアプリケーションの実行中に一時的にRAM上にデータ領域を設定して作業領域として利用できる手続きをアプリケーションで指定できる機能がある。この領域を他のアプリケーションに不正に利用されないことが安全上求められる。このためアプリケーションの切換え時に前のアプリケーションが一時的にRAM上に確保した領域を確実に取り除くことで対応している。

3.5 暗号ライブラリ

アプリケーションがその処理のなかで安全性を確保するために暗号技術を利用することがある。このため，世の中で標準的に使用されている暗号方式のライブラリが準備されている。このライブラリを使用することによりアプリケーションで容易に暗号技術を扱うことが可能となる。

4 あとがき

当社の試作は，Java Card 2.0仕様に基づいている。この仕様は，改訂作業を終了しJava Card 2.1として発表済みである。現在，こちらの仕様に基づくカードを開発中である。安全面で考慮すべき課題は，共有データ機能の追加である。本文で述べたように，複数アプリケーションの相互不干渉を安全面で確保しているが，複数のアプリケーションでのデータ共有を改訂版では規定している。共有データに安全面でどのような配慮をするかが，これからの大きな課題である。

酒井　高彦　SAKAI Takahiko

デジタルメディア機器社 柳町デジタルメディア工場
カードシステム部設計担当参事。
ICカードシステムの開発に従事。情報処理学会会員。
Yanagicho Operations - Digital Media Equipment