社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2007－DPS－130 (23)
2007－CSEC－ 36 (23)
2007／3／1

プロトコル変更可能なマルチアプリケーションICカードシステムの検討

内山 宏樹† 梅澤 克之† 小林 賢†

† 株式会社 日立製作所 システム開発研究所 〒212-8567 神奈川県川崎市幸区鹿島田 890

E-mail: †{hiroki.uchiyama.tm, katsuyuki.umezawa.ue, ken.kobayashi.kj}@hitachi.com

概要. 近年，個人が利用する電子機器が様々な形で外部のネットワークと接続される社会がますます進展しつつある。このようなユビキタスネットワークでは利用環境に応じた「安全」の提供が必要となる。また，セキュリティシステムにおいては，実サービス開始後に方式の脆弱性が発見される場合があり，このような脆弱性に対する対策を迅速に行う仕組みを提供する必要がある。こうした状況を踏まえ，多種多様なアプリケーションに柔軟に対応可能なセキュリティプロトコルの動的生成技術の開発を（株）KDDI 研究所，および，(株) 日立製作所の2社により実施している。本稿では，特に制約が厳しい IC カード環境においてセキュリティプロトコルの動的生成システムを実証するための技術について報告する。

キーワード. IC カード，プロトコル，ユビキタス

A study on the system of multi-application smart card that has protocol change function

Hiroki Uchiyama† Katsuyuki Umezawa† Ken Kobayashi†

†Systems Development Laboratory, Hitachi, Ltd. 890 Kashimada, Saiwai-ku, Kawasaki-shi, Kanagawa, 212-8567 Japan

E-mail: †{hiroki.uchiyama.tm, katsuyuki.umezawa.ue, ken.kobayashi.kj}@hitachi.com

Summary. Recently, the ubiquitous network is widespreading. In the ubiquitous network, it is important to provide "Safety" corresponding to the using environment. Moreover, when the vulnerability of the method is discovered for the security system after it begins to operate, it is also important to take countermeasures to meet this vulnerability promptly. Against this situation, the dynamic generation technology of the security protocol that can flexibly adapt to various applications is researched and developed by KDDI R&D Laboratories and Hitachi Ltd. In this paper, we report on the technology to prove the dynamic generation system of the security protocol in smart card environment where the restriction is especially severe.

Keywords. Smart Card, Protocol, Ubiquitous

1 はじめに

近年，携帯電話，情報家電，ネット接続が可能なゲーム機，携帯電話や非接触 IC カード等の普及にともない，個人が利用する電子機器が様々な形で外部のネットワークと接続される社会がますます進展しつつある。ユビキタスネットワークでは，多種多様な端末，ネットワークが利用され，利用環境に応じた「安全」の提供が必要となる。例えば，電子決済のような高度なセキュリティが要求されるアプリケーションや，公共施設の利用のように資格を確認できればよいサービスなど，提供されるアプリケーションやサービスによって様々なレベルのセキュリティが必要となる。このような多種多様なアプリケーションに柔軟に対応することは，従来の機器では現実的ではなかった。特に IC カードにおいては，計算能力や利用可能メモリ量の制約から適用範囲が限定的にならざるを得ない状況にある。

また，セキュリティシステムにおいては，実サービス開始後に，方式の脆弱性が発見される場合があり，このような脆弱性への対策を迅速に行う仕組みを提供する必要がある。例えば，SSL のセキュリティプロトコルが用いる暗号アルゴリズム（RSA 暗号モード）においても，脆弱性を指摘され，仕様を変更するといったことが過去に発生しており，迅速な対策を実現可能な技術の開発が急務となっている。特に IC カードのようなデバイスでは，暗号アルゴリズムを LSI などのハードウェアに実装して用

表 1: IC カードカスタマイズ方式の比較

方式	外部カスタマイズ方式	内部カスタマイズ方式
対象	セキュリティプロトコルや暗号アルゴリズムが危殆化するなどの異常事態への対応	環境の変化（ユビキタス環境）への対応
利点	カスタマイズに比較的大きな時間がかかってしまうが，セキュリティプロトコルの大規模な変換が必要な場合に有効。	環境パラメータに合わせて，動的に，比較的短い時間でプロトコルの変更が必要な場合に有効
適用例	特に危殆化への対応のための大規模なセキュリティプロトコル変更が必要となる場合。従来はハードウェア実装されている暗号アルゴリズムのソフトウェア実装による追加が必要な場合。内部カスタマイズ方式のコンパイラそのものを追加，変更したい場合。	IC カードに実装されている暗号アルゴリズムの呼び出し順序の変更や，暗号アルゴリズムに与えるデータの変更など，カスタマイズの程度が軽微であり，その変更に速さが要求される場合。モバイル環境，ETC 環境，ネットカフェ環境等のユビキタス環境において迅速なプロトコル変換が必要な場合。セキュリティプロトコルの危殆化に対する小規模な変更が必要な場合。

いる場合が一般的であるため，脆弱性が指摘された暗号アルゴリズムを用いない新たなセキュリティプロトコルを動的に設計する仕組みが必要となる。こうした状況を踏まえ，リソースの限られた環境で，多種多様なアプリケーションに柔軟に対応可能なセキュリティプロトコルの動的生成技術の開発を（株）KDDI 研究所，および，(株) 日立製作所の 2 社により実施している。

本稿では，特に制約が厳しい IC カード環境においてセキュリティプロトコルの動的生成システムを実証するための技術について報告する。以下では，まず，2 章で IC カード向けセキュリティプロトコルカスタマイズ方式の概要について記述する。3 章でプロトコルカスタマイズ方式の一方式について記述し，4 章でプロトコルカスタマイズを実施する上で必要となるアクセス制御方式について記述する。そして最後に 5 章でまとめと今後の課題を示す。

2 IC カード向けセキュリティプロトコルカスタマイズ方式の検討

本節では，特に制約が厳しいと考えられる IC カードシステムに対してセキュリティプロトコルをカスタマイズする方式の検討方針の概要を示す。

2.1 方式検討

本研究では，セキュリティプロトコルをカスタマイズする方式として以下に示す 2 種類の方式の検討を行う。第 1 の方式は，IC カード外部でカスタマイズしたセキュリティプロトコルモジュールを安全に IC カードに入れ替える方式によるセキュリティプロトコルのカスタマイズ技術であり，第 2 の方式は，IC カード内部でのセキュリティプロトコルカスタマイズ技術である。

ここで第 1 の方式，つまり，IC カード外部でカスタマイズしたセキュリティプロトコルモジュールを入れ替える方式 (以降，「外部カスタマイズ方式」と呼ぶ) と，第 2 の方式，つまり，IC カード内部でセキュリティプロトコルをカスタマイズする方式 (以降，「内部カスタマイズ方式」と呼ぶ) を比較する。表 1 に両方式の比較を示す。両方式を組み合わせることにより，両者の利点を生かしたアプリケーション（サービス）への適用が可能であると考えられる。

2.2 開発実装方針

前節で示した検討結果に従い，マルチアプリケーション IC カードに対応したセキュリティプロトコルカスタマイズシステムの実装方針を検討する。具体的には，セキュリティプロトコルモジュールを，メモリ容量面，計算能力面の制約が大きい IC カード内に安全に実装するための方式を開発するとともに，IC カード内にセキュリティプロトコルモジュールが実装されても，上位のアプリケーション (以下，AP と記述する) が変更なく動作できるよう，セキュリティプロトコルモジュールとそれを利用する AP との間の適切な連携方式について検討する。

以上の方針を踏まえ，本研究では，以下の技術を確立する。

1. セキュリティプロトコルモジュールの外部カスタマイズ方式の検討と実装
2. AP からセキュリティプロトコルモジュールへのアクセス制御方式の検討

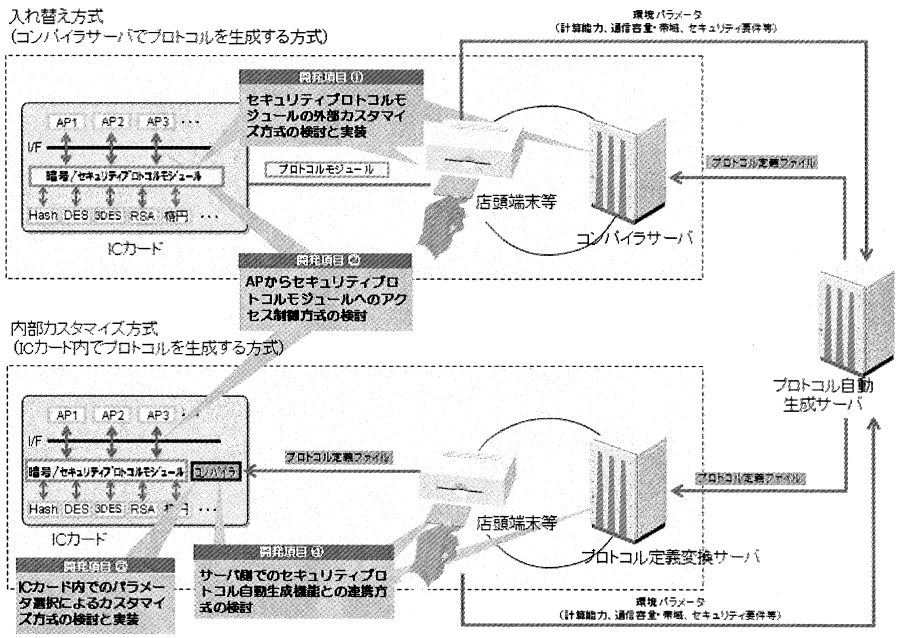

図 1: 開発する技術の関連図

3. IC カード内でのパラメータ選択によるカスタマイズ方式の検討と実装
4. サーバ側でのセキュリティプロトコル自動生成機能との連携方式の検討

また，上記技術の関連図を図 1 に示す。

本稿では，これらの技術のうち，「1. セキュリティプロトコルモジュールの外部カスタマイズ方式の検討と実装」および，「2.AP からのセキュリティプロトコルモジュールへのアクセス制御方式の検討」に関して報告する。

3 外部カスタマイズ方式の検討と実装

3.1 従来の課題および検討方針

マルチアプリケーション対応の IC カードのための IC カード管理スキームとして，GlobalPlatform[1][2] や MULTOS[3] などが代表として挙げられる。2 つのスキームでは AP 搭載許可証の発行主体や，AP 搭載時のコマンド体系など様々な点で違いがあり，外部システムはそれぞれのスキーム用のものを用いなければならない。IC カードの発行スキームが異なると，IC カードへの AP の搭載方法が異なるため，結果的にスキームの異なる IC カードの管理は単一のシステムでは実行不可能になる。もし複数のスキームの IC カードに対応しようとした場合には，IC カードのスキームごとに外部システムを用意し，対象とする IC カードがどのようなスキームに従っているかを判断した上で，適切な外部システムを選択する必要がある。このため，従来 IC カードは，管理スキームの違いにより外部システムを多重化しなければならないという課題が存在していた。そこで，本研究では，GlobalPlaform スキームと，MULTOS スキームの両スキームに対応したセキュリティプロトコルモジュールの安全な置換方式の検討を行い，その方式に基づくセキュリティプロトコルモジュールの置換システムの構築を行う。

3.2 提案システムの全体概要

図 2 に外部カスタマイズ方式の提案システムの全体概要を示す。

まず，プロトコル生成装置でプロトコルフロー定

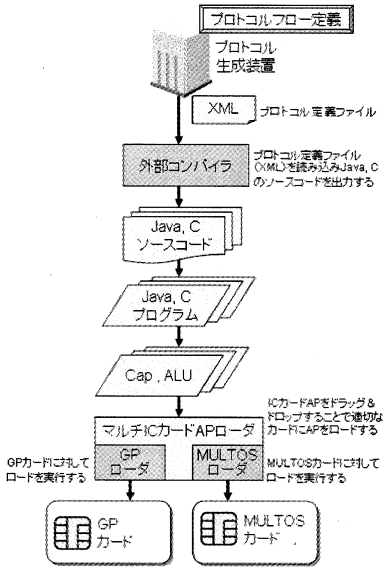

図 2: 外部カスタマイズ方式の提案システムの全体概要

義を生成する。その結果，XML で記述されたプロトコル定義ファイルが出力される。次に，プロトコル定義ファイルを外部コンパイラに入力する。外部コンパイラはプロトコル定義ファイルを Java や C のソースコードに変換する。変換されたソースコードからプログラムモジュールを作成し，マルチ IC カード AP ローダを用いて GP カードや MULTOS カードに書き込む。このようなフローを実装することにより，IC カードの管理スキームの差異を気にすることなくプロトコルの置換が可能になる。

以下では，プロトコルフロー定義，外部コンパイラ，マルチ IC カード AP ローダについて詳細に記載する。

3.3 プロトコルフロー定義

図 3 にチャレンジレスポンス方式のプロトコルフロー定義の例を示す。この例ではエンティティ 2 がエンティティ 1 をチャレンジレスポンス方式によって認証している。

このように定義されたフローの場合，XML で記述されたプロトコル定義ファイルに変換することができる。

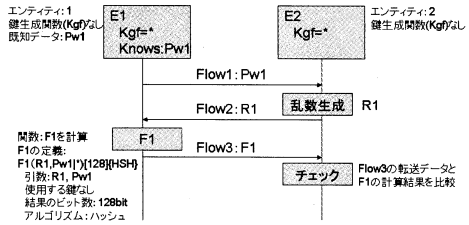

図 3: プロトコルフロー定義の例

3.4 外部コンパイラ

外部コンパイラは，Parse 部，変換部，ソースコード生成部で構成し，前述の XML で記述されたプロトコル定義ファイルを読み込み，JavaCard[4] 用のセキュリティプロトコルを実現するソースファイルを出力する。出力されるソースの一部を以下に示す。

```
// フロー 1，フロー 2 のフロー実行メソッド
void flow1flow2(APDU apdu) {
 byte[] apduBuffer = apdu.getBuffer();
 short bytes_read = apdu.setIncomingAndReceive();
 // APDU コマンドからデータパスワード 1 を取得する
 Util.arrayCopy(apduBuffer,
   (short)(ISO7816.OFFSET_CDATA), dataPassword1,
   (short)0,LENGTH_DATAPASSWORD1);
 dataPassword1Size = LENGTH_DATAPASSWORD1;
 // データランダム 1 を生成する
 RandomData randomData;
 randomData = RandomData.getInstance(
                RandomData.ALG_SECURE_RANDOM);
 randomData.generateData(dataRandom1, (short) 0,
   (short) LENGTH_DATARANDOM1);
 dataRandom1Size = LENGTH_DATARANDOM1;
 // 送信するデータを連結する
 Util.arrayCopy(dataRandom1, (short) 0,
   ResponseTmpData, (short)0, dataRandom1Size);
 // データを送信する
 apdu.setOutgoing();
 apdu.setOutgoingLength((short) (LENGTH_DATARANDOM1));
 apdu.sendBytesLong(ResponseTmpData, (short) 0,
     (short) (LENGTH_DATARANDOM1));
}
```

3.5 マルチ IC カード AP ローダ

図 4 にマルチ IC カード AP ローダの外観を示す。前節の外部コンパイラにより生成された JavaCard ソースファイルをコンパイルし，生成された cap ファイル (実行モジュール) を本マルチ IC カード AP ローダにドラッグ & ドロップすることで，IC カードにロードすることが可能である[1]。

[1]本例は JavaCard の例を示しているが，MULTOS カードに対しても MULTOS 用の ALU ファイル (実行モジュール) をドラッグ & ドロップすることによってロードが可能である。

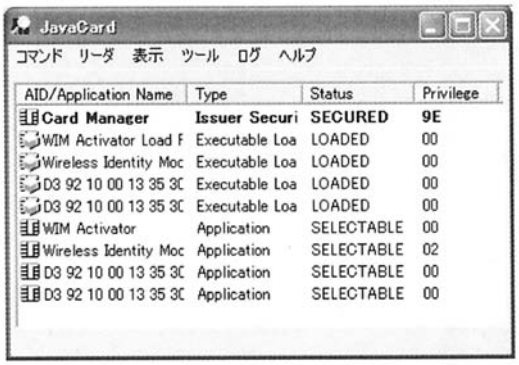

図 4: マルチ IC カード AP ローダの概観

3.6 実行結果

図 3 で示したチャレンジレスポンスのフローを実現するモジュールを外部コンパイラおよびマルチ IC カード AP ローダを用いて IC カードに実装し，動作させた結果を以下に示す。この結果，期待通りの動作を行うことが確認できた。

```
Flow1（外部 password の送信）
APDU Name: Test1 SendChallenge
Command APDU :
00000000  00A5010C101112131415161718191A1B
00000010  1C1D1E1F20

Flow2（カード乱数の受信）
Response APDU :
00000000  ED0D055BCD8B3B7E2A2522FF147AAF2F
00000010  9000

Flow3（認証計算値受信および認証，認証結果受信）
APDU Name: Test2 Authentication
Command APDU :
00000000  00A6020C143B4D633A02A06BEF463BB1
00000010  3A719B614100409CE6
Response APDU :
00000000  9000
```

4 セキュリティプロトコルモジュールのアクセス制御方式の検討

本節では，置換あるいはカスタマイズされたセキュリティプロトコルモジュールとそれを利用する IC カード内の AP 間のアクセス制御方式の検討を行う。

4.1 従来の課題

セキュリティプロトコルモジュールは，IC カード内の OS あるいは AP から呼ばれるミドルウェアを想定している。このようなセキュリティプロトコルモジュールが置換あるいはカスタマイズされた場合，呼び出し側の OS や AP に対しても変更が必要になることが想定される。マルチアプリケーション対応の IC カードの場合，一般に複数の AP が搭載されることが予想されるため，セキュリティプロトコルモジュールのカスタマイズに伴う影響は大きくなる可能性がある。また，それらの AP は異なる事業者がオーナーであることが想定されるため，AP への影響はさらに膨大なものになる可能性がある。このような影響を避けるために，セキュリティプロトコルモジュールが置換あるいはカスタマイズされても，IC カード内の OS や AP は変更することなく動作しなければならないという課題がある。（課題 1）

また，IC カードに複数の AP が搭載される場合，各 AP で要求されるセキュリティレベルは異なることが想定される。このため，セキュリティプロトコルモジュールの置換あるいはカスタマイズにより，プロトコルのセキュリティレベルが変更された場合には，そのプロトコルのセキュリティレベルで利用可能な AP を適切に判断しなければならないという課題もある。（課題 2）

4.2 アクセス制御方式の提案

前述の課題のうち，課題 2 を解決するために，環境に応じてセキュリティプロトコルが変更される場合の IC カード内 AP のセキュリティプロトコルモジュールに対するアクセス制御方式を提案する。

図 5 に IC カード内 AP とセキュリティプロトコルモジュールとの適切なアクセス制御方式のシステム概要を示す。

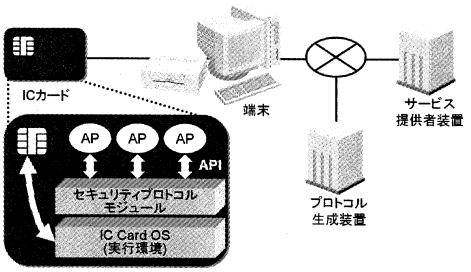

図 5: IC カード内 AP とセキュリティプロトコルモジュール間のアクセス制御方式のシステム概要

図 6 は，図 5 のシステムにおいてアクセス制御を実施する際のフローを示す。まず，プロトコル生成装置がプロトコル定義を生成し，それに基づき IC カードとサービス提供者装置が，両者の間で実行されるセキュリティプロトコルの動作をカスタマイズする。その際，カスタマイズされたプロトコルのセキュリティレベルがセキュリティプロトコルモジュールに設定される。その後，カスタマイズされたセキュリティプロトコルに従って，ユーザはサービスを享受する。

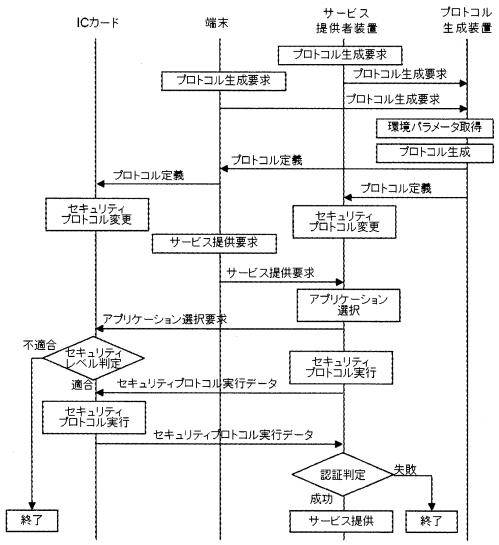

図 6: アクセス制御方式の実行フロー

このようなフローを IC カードやサービス提供者装置に実装することにより，サービスを行う IC カード内の AP は，カスタマイズされたセキュリティプロトコルのセキュリティレベルが，事業者が事前に設定したセキュリティレベルに達していない場合には，動作を終了させることが可能となる。また，サービス提供者装置は，IC カード内の AP に対して適切なセキュリティプロトコルを実行することが可能となる。

5 まとめと今後の課題

本報告では，「セキュリティプロトコルモジュールの外部カスタマイズ方式の検討と実装」および「AP からのセキュリティプロトコルモジュールへのアクセス制御方式の検討」について述べた。前者に関して，複数種類の IC カードに対して外部カスタマイズ方式を実現でき，従来のように外部システムの多重化を行うことなく，セキュリティプロトコルモジュールを置換することに成功した。後者に関して，セキュリティプロトコルモジュールを置換あるいはカスタマイズしても，サービスのセキュリティレベルの低下を防ぐことが可能であることが分かった。

今後は，前者の別の方式として IC カード側で両者の差異を吸収する方式を検討する予定である。また，今回は検討対象外とした「IC カード内でのパラメータ選択によるカスタマイズ方式の検討と実装」および「サーバ側でのセキュリティプロトコル自動生成機能との連携方式の検討」に関しても検討を推進する予定である。

謝辞

本研究は，独立行政法人情報通信研究機構 (NICT) の委託研究「ユビキタスネットワークにおける環境に応じたセキュリティプロトコルの自動生成・カスタマイズ技術に関する研究」の一環として行なわれた。

参考文献

[1] “GlobalPlatform Card Specification Version 2.1.1,” GlobalPlatform Inc., March 2003

[2] “GlobalPlatform Card Specification Version 2.2,” GlobalPlatform Inc., March 2006

[3] “MULTOS カード発行ガイド,” マルトス推進協議会, March 2003

[4] “Java Card 2.2.1 Application Programming Interface,” Sun Microsystems, Inc., October 2003