モバイルFeliCaを使用した認証システムの応用

日大生産工(学部)　○川田　大
日大生産工　　栃窪　孝也

1　はじめに

近年、非接触ICであるFeliCaの普及が著しい。FeliCaとは、東日本旅客鉄道株式会社の提供する電子乗車券Suica(スイカ)、ビットワレット株式会社の提供する電子マネーEdy(エディ)等で利用されており、他にもさまざまな種類の電子マネーが発行されている。更に数年前からは、携帯電話にも搭載され始めてきており、ポイントカードや電子クレジットなど新たな使い道も出てきており利便性が向上している。

一方、従来の動画コンテンツのレンタルサービスは、レンタル店で借りてレンタル店に返却する方法やネットで借りてポストに返却する方法などが一般的である。また、ブロードバンドの普及により専用の機器を使用してテレビで動画を視聴できるような動画レンタル配信サービスが利用できるようになった。しかし、パスワードの入力の手間やオンラインでなければ視聴できないなどの問題がある。

そこで本稿では、FeliCaを使うことで安全性と利便性とを両立するコンテンツレンタル配信システムを提案する。提案システムは、ユーザ認証をFeliCaで行うことで、オフラインでも再生することができる。また、日数だけではなく、回数に応じたコンテンツ管理ができるのが特徴である。

2　FeliCaの特徴及びコンテンツ配信

2.1　FeliCaの特徴と動作原理

FeliCaとは、ソニーが1994年に開発した非接触ICカードの技術方式である。国内では、1997年に初めて導入されている。リーダ/ライタのカード間の通信は、リーダ/ライタから発信される電磁波によって行われる。電源の供給も電磁波によって行われる。通信の周波数は13.56MHzで、通信速度は約212kbps～424kbpsで行われる。FeliCaには以下のような特徴がある。

・高速処理
独自の効率的な相互認証方式と、非接触の利用形態に適した通信方式によって、リーダ/ライタとカードの間の処理は、暗号処理を含めて約0.1秒で終了する。

・マルチアプリケーション
FeliCaは、1枚のカードの中で多目的のデータを管理することができる。それぞれのデータには個別のアクセス権を設定することが可能で、これによってアプリケーション間の安全な相互運用が実現される。

・アンチ・ブロークン・トランザクション
FeliCaは、1ブロックを16バイトとして8ブロックまでの同時アクセスをサポートしている。非接触の利用形態においては、カード内で処理が完了しないうちにICカードがリーダ/ライタの電力供給範囲の外に出てしまった場合、処理未了が発生する恐れがある。このようなときにも、カード内で自動的に元の状態に戻し、データの不整合を防ぐ。

・通信時の高いセキュリティ
相互認証にはTripleDES、通信データの暗号化においては、DES又はTripleDESを採用し、信頼性の高いセキュリティを実現している。また、通信データの暗号鍵は相互

Application of Authentication System using Mobile FeliCa

Masaru Kawata and Kouya Tochikubo

認証ごとに新しく生成され、成りすましを防いでいる。

2.2　モバイルFeliCaの特徴

モバイルFeiCaとは携帯電話端末にFeliCaチップを搭載させたものを指す。それぞれの事業者が提供するアプリをダウンロードすることで複数のサービスを1つの端末で利用することができる。モバイルFeliCaの場合は端末の電池パックより電源が供給される。更に、アプリとの連携によりFeliCaではできないような演算や複雑な処理を行わせることで利便性が向上している。

2.3　FeliCaとモバイルFeliCaの違い

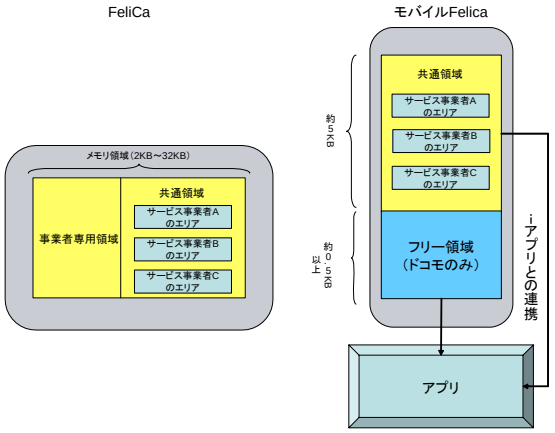

図1　FeliCaとモバイルFeliCaの比較

FeliCaとモバイルFeliCaの大きな違いは内部の構造にある。共通領域は双方とも存在し、サービスエリアごとに割り当てられる。適切なアクセス権があれば異なる事業者エリアへのアクセスも可能である。FeliCaには事業者専用領域があり、それはFeliCaチップ内に設けられたサービス事業者専用のメモリ領域である。基本的に当該事業者がアクセスすることはできない。一方、モバイルFeliCaは基本的に共通領域のみである。しかし、NTTドコモが提供するモバイルFeliCa対応端末についてはフリー領域が存在する。フリー領域により、セキュリティレベルと引き換えにサービスを比較的簡単に提供できる。

3　コンテンツレンタル配信の仕組み

3.1　システム構成

提案するシステムの構成は以下の通りである。

①PCには、コンテンツのレンタル手続きや管理をするアプリケーションをインストールする。また、コンテンツや鍵はHDDに暗号化して保存する。

②サーバでは、①のアプリケーションと連携してレンタル手続きの処理や暗号化されているコンテンツで配信の処理を行う。

③PCに外付けで接続されているリーダ/ライタにより、モバイルFeliCaへのデータの読み書きを行う。

④NTTドコモのモバイルFeliCa搭載端末のiアプリは、モバイルFeliCa内のフリー領域によって動作する。iアプリは、FeliCaチップだけでは実現ができないような計算や探索を行うのに使用する。フリー領域は、認証に必要な情報が保存されている。

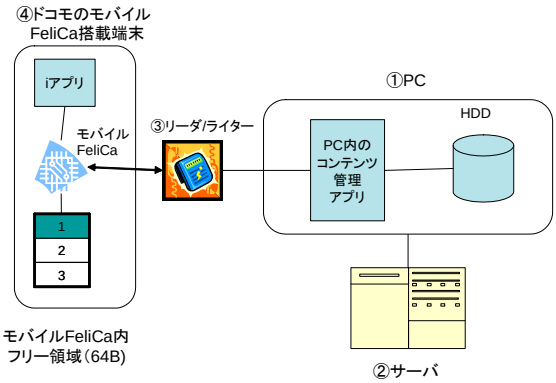

図2　システム構成

3.2　日数制限・回数制限アルゴリズム

3.2.1　日数制限の特徴及び定義と記法

日数制限は、ユーザ側が何日間利用するかを設定する。例えば、ある日にちの14時にレンタル手続きを行い、期間を3日間とした場合は3日後の14時まで利用することができる。3.2.2と3.2.3章で説明する流れの定義と記法について以下に記述する。

<定義と記法>

- XORはビットごとに排他的論理和を行うことを意味し、||は結合を意味する。
- Tをyyyymmddhhnn形式の利用開始日・時刻とする。
- Bを利用日数とする。
- Mは、現在の日付・時刻とする。
- Eは、共通鍵による暗号化を意味する。例えば、$E(F_{ID},C)$とした場合F_{ID}を鍵としてCを暗号化したことを意味する。
- Cをコンテンツとし、C_Eは暗号化されたことを意味する。
- C_{ID}はコンテンツIDとし11バイトとする。
- R1はコンテンツごとに割り当てられ、R2は利用者個別に割り当てられる。どちらも5バイトの乱数である。

・Dは、共通鍵による復号化を意味する。例えば、F_{ID}で暗号化したCを復号したものを$D(F_{ID},C)$のように記述する。
・GをT||Bとしコンテンツ情報とする。G_Eは暗号化されたことを意味する。
・F_{ID}はFeliCaごとに発行されている16バイトの固有のIDである。
・インストールされたアプリケーションには固有の16バイトのID A_{ID}が割り当てられている。
・K_Cをコンテンツ鍵とする。コンテンツ鍵はC_{ID}||R1で構成される。K_{CE}は、暗号化されたコンテンツ鍵を意味する。

<安全性>
提案システムでは、コンテンツをFeliCaごとに発行されている16バイトの固有のIDであるF_{ID}とインストールされたアプリケーションの固有のIDであるA_{ID}により暗号化されているので、対応するFelicaがなければ復号して利用することはできない仕組みになっている。また、日付け・時刻の情報は時刻情報が信頼できないPCからではなく、携帯電話から取得するのが特徴である。このため、ネットワークに接続していないオフライン時でもコンテンツを利用することができる。

3.2.2　日数制限レンタル手続きの流れ

ユーザが日数制限によりコンテンツをレンタルする場合、以下のようサーバと通信を行う。

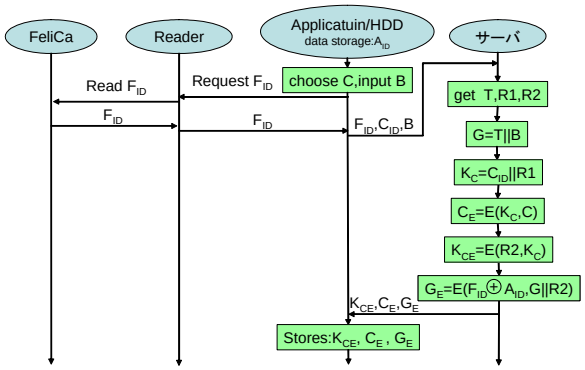

図3　日数制限レンタル手続きアルゴリズム

①Application/HDDでコンテンツCと利用日数Bを選択する。
②ReaderにFeliCaをかざすことでF_{ID}を取得し、Application/HDDはF_{ID}、C_{ID}、利用日数Bをサーバへ送る。
③サーバは利用開始日・時刻T、乱数R1、乱数R2を得る。
④サーバは利用開始日・時刻Tと利用日数Bを結合してコンテンツ情報Gとする。
⑤サーバはC_{ID}と乱数R1を結合してコンテンツ鍵K_Cを得る。
⑥サーバはコンテンツCをコンテンツ鍵K_Cで暗号化してC_Eを得る。
⑦サーバはコンテンツ鍵K_Cを乱数R2で暗号化してK_{CE}を得る。
⑧サーバはコンテンツ情報Gと乱数R2を結合したものを、F_{ID} XOR A_{ID}で暗号化してG_Eを得る。
⑨サーバからコンテンツ鍵K_{CE}、暗号化されたコンテンツC_E、コンテンツ情報を暗号化したG_EをApplication/HDDに送信し保存する。

3.2.3　日数制限での視聴の流れ

ユーザが日数制限によりコンテンツを視聴する場合の処理は以下の通りである。

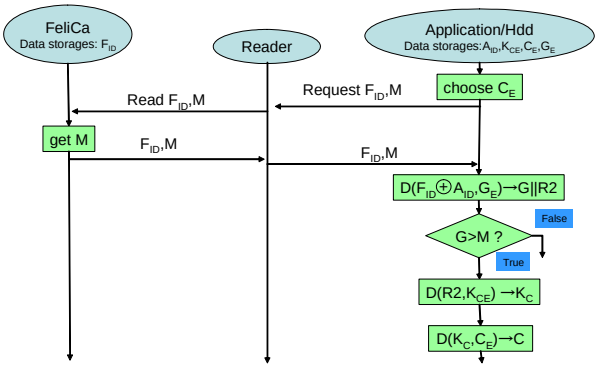

図4　日数制限での視聴アルゴリズム

①Application/HDDで視聴するC_Eを選択し、ReaderにF_{ID}、現在の日付け・時刻Mの読み取り要求を出す。Readerがモバイル FeliCaをかざすように要求する。
②FeliCaをかざしF_{ID}と現在の日付け・時刻MをReaderが取得し、Application/HDDに送る。
③コンテンツ情報を暗号化したG_EをF_{ID} XOR A_{ID}で復号してコンテンツ情報G||乱数R2を抽出し切断する。
④コンテンツ情報G>日付け・時刻Mの場合trueへ、G<=Mの場合はFalseとする。
⑤暗号化されたコンテンツ鍵K_{CE}を乱数R2で復号し、コンテンツ鍵K_Cを抽出する。
⑥暗号化されたコンテンツC_Eをコンテンツ鍵K_Cで復号し、コンテンツCを抽出する。

3.2.4　回数制限の特徴及び定義と記法

回数制限は、ユーザ側が何回利用するかを指定する。日数制限はないが回数によって制限され、利用回数が指定した回数になるまで有効となる。ほとんど日数制限と同じだが、違うところは視聴した回数をFeliCaチップ内保存しなければならないことである。3.2.5章で説明する流れの定義と記法について以下に記述する。

<定義と記法>

・iは、コンテンツを視聴した回数とし、日数制限の現在の日付・時刻Mに対応する。コンテンツを視聴するたびに1増加する。初期値は0である。
・nはユーザが指定した利用回数とし、日数制限の利用日数Bに対応する。n_Eは暗号化されたことを意味する。
・他の表記については3.2.1章と同様である。

3.2.5　回数制限レンタル手続きの流れ

ユーザが回数制限によりコンテンツをレンタルする場合、以下のようにサーバと通信を行う。

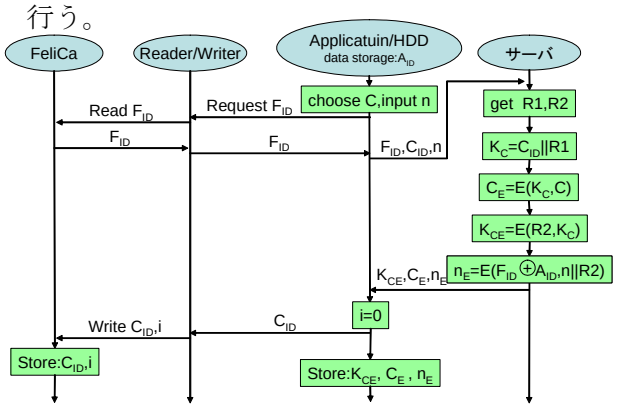

図5　回数制限レンタル手続きアルゴリズム

①Application/HDDでコンテンツCとユーザが指定した回数nを選択する。
②ReaderにFeliCaをかざすことでF_{ID}を取得し、Application/HDDはF_{ID}、C_{ID}、ユーザが指定した利用回数nをサーバへ送る。
③サーバは乱数R1、R2を取得する。
④サーバはC_{ID}と乱数R1を結合してコンテンツ鍵K_Cを得る。
⑤サーバはコンテンツCをコンテンツ鍵K_Cで暗号化しC_Eを得る。
⑥サーバはコンテンツ鍵K_Cを乱数R2で暗号化しK_{CE}を得る。
⑦サーバはユーザが指定した利用回数nと乱数R2を結合したものを、F_{ID} XOR A_{ID}で暗号化してn_Eを得る。
⑧サーバは暗号化されたコンテンツ鍵K_{CE}、暗号化されたコンテンツ鍵C_E、暗号化されたユーザが指定した利用回数n_EをApplication/HDDに送信し保存する。
⑨Application/HDDは視聴した回数iとC_{ID}をWriterに送り、視聴した回数iとC_{ID}をFeliCaに書き込む。

※回数制限での視聴の流れについては、3.2.3章で説明している日数制限での視聴の流れとほぼ同様なので省略する。

4　考察

提案システムの有効性を検証するために、ドコモのモバイルFeliCaに対応した携帯電話端末、リーダ/ライタ、ソニーが無料で配布しているFeliCaブラウザエクステンションを使用してプロトタイプを実装した。なお、FeliCaブラウザエクステンションとは、Felicaの機能をWebブラウザを使って利用するためのものである。

モバイルFeliCaを使用しない場合と比較するとIDやパスワードを覚える必要がなく、かざすだけで高速の認証を行うことができる。更に、コンテンツに対しても十分なセキュリティを確保できることがわかった。

5　おわりに

今回は、モバイルFelicaを利用したコンテンツレンタル配信システムを提案し、その有効性を検証した。提案システムは、ユーザ認証をFeliCaで行うことで、オフラインでも再生することができる。また、日数だけではなく、回数に応じたコンテンツ管理ができるのが特徴である。今後の課題としては、Webブラウザを介してのみしか利用できないFelicaブラウザエクステンションではなく、Felicaを利用するアプリケーションを開発できるSDK for Felicaを使った実装が挙げられる。

参考文献

1)Sony Japan | FeliCa
http://www.sony.co.jp/Products/felica/
2)FeliCa | FeliCaの仕組み
http://www.rdsc.jp/felica/system.html
3)モバイルFeliCaプログラミング
アスキー書籍編集部