マルチメディア通信と分散処理　97－17
コンピュータセキュリティ　8－17
（2000．3．22）

セキュアコンテンツ制御ライブラリの開発

中嶋 春光，宮崎 一哉

三菱電機株式会社 情報技術総合研究所

概要

セキュアコンテンツ制御ライブラリを利用することで、ビューアプログラム開発者は、既存のビューアプログラムに容易に、コンテンツの不正使用を防止するセキュリティ機能を組み込むことが出来るようになる。本稿では、セキュアコンテンツ制御ライブラリが備える、コンテンツの使用制御機能、暗号処理機能の実現方式について報告し、次に、地図ビューアにセキュアコンテンツ制御ライブラリを適用し、その有効性を検証した結果について報告する。

Development of Secure Content Control Library

Harumitsu Nakajima, Kazuya Miyazaki

Information Technology R&D Center, Mitsubishi Electric Corporation

Abstract

Our Secure Content Control Library makes it easy to develop a viewer program that can securely handle contents according to the usage rules attached to the contents. In this paper, we describe the functions and the mechanisms of Secure Content Control Library, and a result to verify the effect of this library by applying to a map viewer.

1. はじめに

著者らは、インターネットなどのオープンなネットワーク上で、電子文書、デジタル素材等のデジタルコンテンツを安全に配布/販売/利用することを目的としたデジタルコンテンツ配布システムーDIGICAPSULE(旧称 DigiGuard)－[1][2][3]を研究開発している。

一般に、デジタルコンテンツ配布システムにおいては、デジタルコンテンツが持つ「完全な複製や改変が極めて容易である」という性質により、コンテンツの不正使用防止、安全な著作権管理や課金管理といった課題を抱え持っている。これらの課題を解決するために、本システムでは、コンテンツを使用する権利を持たない第三者が使用できないように安全な形でコンテンツを内部に格納したカプセルを利用者に配布している。

しかしながら、本技術においても、コンテンツを閲覧(または編集)するビューアプログラムにおいて、コンテンツの不正使用を防止するセキュリティ機能が用意されていなければ、コンテンツの権利者が満足する安全なシステムを構築することは難しい。

従来の DIGICAPSULE で、デジタル文書の標準フォーマットである PDF ファイルについては、本システムで提供するプログラムが、そのビューアプログラムが提供する API を利用することで、PDF ファイルの使用制御(印刷や編集の禁止)、及び、履歴情報に基づく使用制御を実現していたが、他のコンテンツについては、多くの場合、そのビューアプログラムに外部からのメニュー制御等、コンテンツの使用を制御する API やコマンドが用意されていないために、PDF ファイルの場合と同様の効果を挙げることが出来ずにいた。

今回、著者らは、他のコンテンツにおいても PDF ファイルの場合と同様の効果が得られるよう、ビューアプログラム開発者が、ビューアプログラムに容易に、コンテンツの不正使用を防止するセキュリティ機能を組み込むことが出来るセキュアコンテンツ制御ライブラリを開発した。

本稿では、セキュアコンテンツ制御ライブラリが備えるコンテンツの使用制御機能、暗号処理機能の実現方式について報告し、次に、地図ビューアにセキュアコンテンツ制御ライブラリを適用し、その

有効性を検証した結果について報告する。

2. 開発の目的

2.1. 従来方式の課題

DIGICAPSULE において、利用者に配布するコンテンツは、そのデータをコンテンツ鍵で暗号化した暗号化データ、コンテンツ鍵をコンテンツの正規利用者の公開鍵で暗号化した暗号化データ、コンテンツの使用規則を記述したスクリプト及びコンテンツの著作権情報から構成される一つまたは複数のコンテンツオブジェクトを格納したカプセルの形で配布される。

従来の DIGICAPSULE では、カプセル実行プログラムが、カプセルに格納する暗号化コンテンツを復号し、出力した平文コンテンツをビューアプログラムに入力することで、コンテンツを利用者に提供していた。また、このとき、コンテンツの使用規則を記述したスクリプトを、カプセル実行プログラムが内部に持つインタープリタで解釈・実行した結果に従って、ビューアプログラムを制御し、コンテンツの使用制御、及び、履歴情報に基づく使用制御を行う枠組みを提供していた(図 1)。

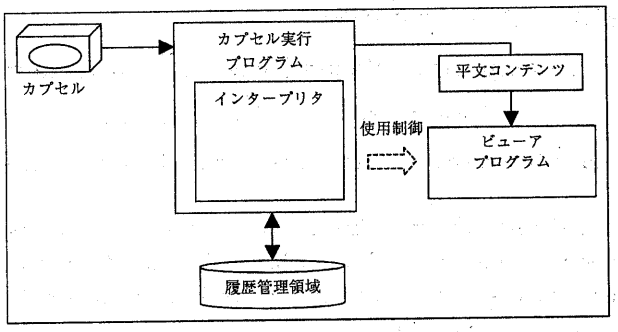

図 1

しかしながら、ここで、コンテンツを閲覧するビューアプログラムに、外部(カプセル実行プログラム)からのメニュー制御等、コンテンツの使用を制御する API やコマンドが用意されていなければ、その十分な効果を期待することが出来なかった。

また、本システムでは、カプセル実行プログラムとビューアプログラムが独立したプログラムであるため、両者の間で操作に連続性が無く、それにより、利用者にとってはコンテンツを閲覧する際の操作が煩雑であるという問題点があった。

また、従来のコンテンツ配布システム[4][5]は、多くの場合、コンテンツの不正使用を防止する目的から、コンテンツの利用者に、システムに特化したビューアプログラムの使用を義務付けていた。その結果、利用者は、限られたビューアプログラムしか使用できず、使用上の制約を受けることが多かった。

2.2. 提案方式とその特徴

著者らは、上記の課題を解決するために、次の方式を提案し、その開発を行った。

- ビューアプログラムで閲覧しているコンテンツと、そのときに発生したイベントに対応する使用規則スクリプトを解釈・実行した結果に従って、履歴情報を管理し、ビューアプログラムを制御するセキュアコンテンツ制御ライブラリを提供する。
- セキュアコンテンツ制御ライブラリがビューアプログラムに提供する、カプセル及びコンテンツを処理する API で、復号機能、利用者検証機能、改竄検証機能といった暗号処理機能を隠蔽する。

上記提案方式に従って開発したセキュアコンテンツ制御ライブラリの特徴を以下に示す。

- 使用規則スクリプトの記述に従った、コンテンツの使用制御が可能になる。
- ビューアプログラム開発者が、復号機能、利用者検証機能、改竄検証機能といった暗号処理機能

を意識せずに、ビューアプログラムに、コンテンツの不正使用を防止するセキュリティ機能を実装することが出来る。

3. 実現方式

以下、上記提案方式に基づき、セキュアコンテンツ制御ライブラリに実装した、次の機能の実現方式について説明する。

(1) コンテンツの使用制御機能

(2) 暗号処理機能

3.1. コンテンツの使用制御機能

ビューアプログラム開発者が、コンテンツの使用制御の対象となる操作に対して、処理の実行を制御する(処理を実行または中止する)コマンドや、ビューアプログラムの機能の使用を制限する(例えば、あるメニューの選択を禁止する)コマンドをビューアプログラムに用意することで、使用規則スクリプトの記述に従ったコンテンツの使用制御、及び、履歴情報に基づく使用制御が可能になるよう、セキュアコンテンツ制御ライブラリでは、次の処理手順で示されるように、コンテンツの使用制御を実現している(図 2)。

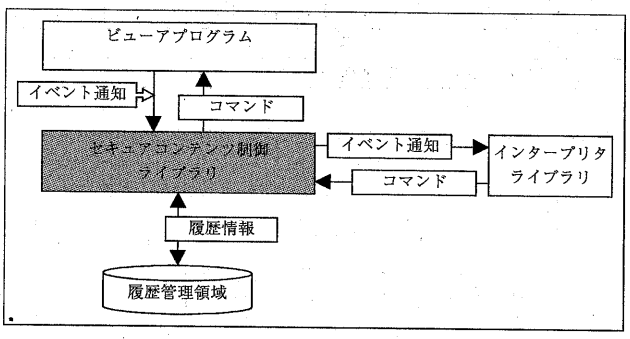

図 2

(1) ビューアプログラムは、操作対象のコンテンツとその操作に対応するイベントをセキュアコンテンツ制御ライブラリに通知する。

(2) セキュアコンテンツ制御ライブラリは、イベントの発生を、使用規則スクリプトを解釈・実行するインタープリタライブラリに中継、通知する。

(3) インタープリタライブラリは、使用規則スクリプトの記述に従って、ビューアプログラムにおける処理の実行制御や機能の使用制限、または、コンテンツの履歴情報の読み取りや更新を、コマンドの形で、セキュアコンテンツ制御ライブラリに命令する。

[ビューアプログラムにおける処理の実行制御または機能の使用制限が命令された場合]

(4) セキュアコンテンツ制御ライブラリは、コマンドを、ビューアプログラムに中継、送信する。

(5) ビューアプログラムは、イベントに対応する処理を実行または中止したり、あるいは、ビューアプログラムにおける機能の使用を制限する。

[履歴情報の読み取りや更新が命令された場合]

(4) セキュアコンテンツ制御ライブラリは、履歴管理領域で管理する履歴情報の読み取りや更新を実行する。(さらに、実行規則スクリプトの記述に従って、履歴情報に基づくコンテンツの使用制御が行われる場合もある。)

また、本システムでは、ビューアプログラム開発者が、コンテンツの種別、及び、ビューアプログラムに適した形で、使用規則スクリプトの記述フォーマットを変更または拡張することが出来るよう、セキュアコンテンツ制御ライブラリーインタープリタライブラリ間のインターフェイスを公開している。

ビューアプログラム開発者は、変更または拡張した使用規則スクリプトを解釈・実行するインター

プリタライブラリを独立に開発し、開発したインタープリタライブラリをビューアプログラム(セキュアコンテンツ制御ライブラリ)に組み込むことが出来る。

3.2. 暗号処理機能

ビューアプログラム開発者が、既存のビューアプログラムに容易に、セキュリティ機能を組み込めるようにするためには、従来の DIGICAPSULE でカプセル実行プログラムが提供していた復号機能、利用者検証機能、改竄検証機能といった暗号処理機能を、ビューアプログラムに対して隠蔽する必要がある。

そのため、セキュアコンテンツ制御ライブラリでは、カプセルやコンテンツのオープンの処理を、ビューアプログラムに対する API として提供し、これらの処理に伴う暗号処理は、セキュアコンテンツ制御ライブラリが内部で実行するようにしている(表 1)。

表 1

API	説明
OpenCapsule	カプセルをオープンする。
CloseCapsule	カプセルをクローズする。
OpenContent	コンテンツをオープンする。(利用者検証、暗号化コンテンツの復号を行う。)
CloseContent	コンテンツをクローズする。
VerifyContent	コンテンツの改竄検証を行う。

ここで、セキュアコンテンツ制御ライブラリは、次の処理手順で、カプセルのオープンを実現している(図 3)。

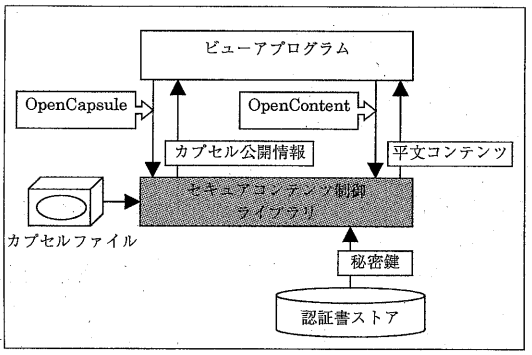

図 3

(1) ビューアプログラムは、セキュアコンテンツ制御ライブラリにカプセルのオープンを要求する(OpenCapsule の API を呼ぶ)。

(2) セキュアコンテンツ制御ライブラリは、カプセルファイルをロードし、カプセルに格納されているコンテンツファイルの名前や属性、著作権情報等、ビューアプログラムまたは利用者にとって必要な公開情報のみを、ビューアプログラムに渡す。

これにより、ビューアプログラム開発者は、カプセルファイルのフォーマットを意識せずに、セキュアコンテンツ制御ライブラリを組み込んだビューアプログラムを開発することが出来るようになる。また、公開情報以外の、コンテンツ鍵、署名等のデータについては、セキュアコンテンツ制御ライブラリが内部で管理し、公開しないため、ビューアプログラム開発者は、これらのデータや暗号処理の中身については意識せずに、上記の暗号処理機能をビューアプログラムに実装することが出来る。

また、従来の DIGICAPSULE とは異なり、コンテンツを閲覧するための暗号処理機能が、ビューアプログラム自身に組み込まれるため、ビューアプログラムのメニュー等で、カプセルのオープンやコ

ンテンツのオープンを実行でき、コンテンツを閲覧する利用者に煩雑な操作を要求することがない。

さらに、セキュアコンテンツ制御ライブラリは、次の処理手順で、コンテンツのオープンを実現している(図 3)。

(1) ビューアプログラムは、セキュアコンテンツ制御ライブラリにコンテンツのオープンを要求する(OpenContent の API を呼ぶ)。

(2) セキュアコンテンツ制御ライブラリは、暗号化コンテンツをビューアプログラムが理解できる形に復号した平文コンテンツを、ビューアプログラムに渡す。

ここで、多くの場合、市販ビューアプログラムにおけるコンテンツの入力形式はファイルであるため、従来の DIGICAPSULE では、配布対象のコンテンツが、このような市販ビューアプログラムでの閲覧を前提としていた場合、復号したコンテンツをファイルとして一時的にローカルディスク等に保存せざるを得ず、コンテンツが不正に流通する危険があった。

この不正流通を防止するために、セキュアコンテンツ制御ライブラリは、平文コンテンツを、コンテンツの利用者さえも容易に取得することが出来ないよう、メモリ経由でビューアプログラムに渡すようにしている。

さらに、本システムでは、公開している API を利用した不正ビューアの開発を防止するための仕組みを持っている。悪意を持つビューアプログラム開発者が、セキュアコンテンツ制御ライブラリの API を利用して、コンテンツの使用規則を無視した、コンテンツの使用制御を行わないようなビューアプログラムを開発する危険性が考えられる。本システムでは、全てのカプセルファイルを難読化しておき、カプセルに種別を設け、その種別ごとに用意したある秘密情報をセキュアコンテンツ制御ライブラリに与えなければ、カプセルファイルを解読することができない仕組みになっている。この秘密情報は、その種別のカプセルを処理するための権限を与えられた正規のビューアプログラムのみに開示され、ビューアプログラムの開発時にビューアプログラム固有の変形を施した上でプログラム内部に格納するようにしている。

4. 地図ビューアへの適用

4.1. 要件

セキュアコンテンツ制御ライブラリを地図ビューアに適用することで、その有効性を検証する。地図ビューアで閲覧、編集する地図コンテンツは、複数のコンテンツを合成した複合コンテンツであり、その特徴は以下の通りである[6]。

- 一つの地図コンテンツは制作者が異なる複数のコンテンツを重ね合わせた複合コンテンツ(階層構造)であり得る。
- 各階層に相当する個々のコンテンツ毎に著作権を主張し得る。
- 3 次コンテンツ(既存の地図コンテンツを 2 次利用して作成した地図コンテンツをさらに 2 次利用して作成したコンテンツ)以降のコンテンツ販売は許可されない傾向にある。

ここで、利用者に提供するカプセルは、地図コンテンツの各階層に相当する個々のコンテンツを暗号化し、内部に格納したものとする。地図ビューアに求められる要件を以下にまとめた。

(1) 制作者や配布者の意図に従って、個々のコンテンツ毎に異なる使用制限、使用履歴管理を実現できるようにする。

(2) 地図ビューアの開発者が暗号処理を意識せずに、地図ビューアを開発できるようにする。

(3) 各階層に相当する個々のコンテンツ毎に著作権を管理できるようにする。

4.2. 適用結果

[(1)の要件について]

従来の DIGICAPSULE でも、カプセルに格納されている各コンテンツ毎に、異なる使用規則を記述することは出来たが、既存の地図ビューアを利用する場合、カプセル実行プログラムとの連携が十分でないため、細かな制御が出来ないという制限があった。

本システムでは、地図ビューア開発者が、次の地図ビューアを開発することで、この要件を満たす

ことが出来るようになる。

- ある階層に相当するコンテンツに対して操作が実行されたとき、例えば、あるメニューの選択時、操作対象のコンテンツとその操作、例えば、印刷等に対応するイベントをセキュアコンテンツ制御ライブラリに通知する。
- イベントに対応する処理の実行(印刷、コピー&ペースト)を中止したり、あるメニューの選択を禁止したりするコマンドを実装する。
- 地図ビューアが備えるべき使用制御の仕様に合わせて、使用規則スクリプトの記述フォーマットを定義し、そのスクリプトを解釈・実行するインタープリタライブラリを開発する。

[(2)の要件について]

セキュアコンテンツ制御ライブラリが提供するAPIを利用することで、地図ビューア開発者は、暗号化鍵、利用者の秘密鍵、署名等を意識せずに、地図ビューアで次の機能を実装することが出来る。

- 各階層に相当するコンテンツの利用者検証。
- 各階層に相当するコンテンツの平文データの取得。
- 各階層に相当するコンテンツの改竄検証。

また、上記暗号処理機能で必要となる認証書(秘密鍵)にアクセスするために、利用者による認証書ストアへのログイン操作が必要になるが、ログイン用ダイアログも含め、ログイン操作の処理もセキュアコンテンツ制御ライブラリが提供しているため、地図ビューアの開発者がこのログイン操作の処理を実装する必要は無い。

[(3)の要件について]

カプセル内の個々のコンテンツは異なる鍵で暗号化されており、それをコンテンツの権利者毎に分散して管理することにより、各階層に相当する個々のコンテンツ毎に著作権を管理することが出来るようになる。

但し、既存の地図コンテンツの一部階層を入れ替えたり、階層を追加することで新規に作成した、2次または3次の地図コンテンツの配布、販売については、セキュアコンテンツ制御ライブラリでは、サポートできていない。これをサポートするためには、既存のカプセルに格納されているコンテンツを取り出し、新たなカプセルを生成するカプセル複合化機能と、新規に作成した地図コンテンツで2次または3次利用している全てのコンテンツの利用許諾を得るためのライセンス複合化機能をシステムに実装する必要がある[6]。これらの機能については、今後、本システムに実装し、さらに有効性を検証する予定である。

5. まとめ

本稿では、著者らが開発したセキュアコンテンツ制御ライブラリについて報告した。セキュアコンテンツ制御ライブラリにより、ビューアプログラム開発者は、コンテンツの利用者に煩雑な操作を要求すること無く、コンテンツの不正使用を防止するセキュリティ機能を容易にビューアプログラムに組み込むことが出来るようになる。

今後は、コンテンツの2次利用を考慮したシステム構築のため、ビューアプログラムにカプセル生成の機能を実装するためのライブラリを開発する予定である。

[参考文献]

[1] 宮崎ほか,「セキュアデジタルコンテンツ配布方式の検討」情報処理学会第55回全国大会 6Q-1(1997)
[2] 中嶋ほか,「セキュアデジタルコンテンツ配布システム-DIGITEX-の開発」1998年電子情報通信学会総合大会 SD-3-7
[3] 宮崎,「セキュアコンテンツ配布システム DigiGuard」CALS/EC Japan 1998 論文集(1998/11)
[4] 玉井ほか,「情報販売における不正コピー防止方式の実装」情報処理学会第57回全国大会 2K-3(1998)
[5] IBM, [Cryptolope Technology] <http://www-4.ibm.com/software/security/cryptolope/about.html>
[6] 宮崎ほか,「地図コンテンツ流通における分散著作権管理方式の提案」情報処理学会 第59回全国大会 5D-06(1999)