

デジタルコンテンツにおける不正コピー防止方式の実装と評価

稻村 勝樹[†] 田中 俊昭[†]

†株式会社 KDDI 研究所 〒356-8502 埼玉県上福岡市大原 2-1-15

E-mail: †{minamura, tl-tanaka}@kddilabs.jp

あらまし コピー回数の制限機能を有しつつ、コピーを使用する端末を限定せず、特殊なハードウェアに依存しない不正コピー防止機能を提案する。本稿では、提案方式を実装し、安全性および性能の観点からその実現性を評価したので報告する。

キーワード 著作権保護、不正コピー、端末 ID、私的コピー

Implementation and Evaluation of Illegal Copy Protection for Digital Contents

Masaki INAMURA[†] and Toshiaki TANAKA[†]

† KDDI R&D Laboratories Inc. 2-1-15 Ohara, Kamifukuoka-shi, Saitama, 356-8502 Japan

E-mail: †{minamura, tl-tanaka}@kddilabs.jp

Abstract We propose a new method of illegal copy protection, which allows private copies to arbitrary terminals with in the limited times, and requires no secure hardware. In this paper, we implement the proposed method and evaluate whether our method is feasible from the viewpoint of security and performance.

Keyword Copyright Protection, Illegal Copy, Terminal ID, Private Copy

1.はじめに

コンピュータやネットワーク技術の発達により、アプリケーションソフトの流通や音楽・動画コンテンツのデジタル化が発展してきた。しかし、デジタル情報は劣化せず、簡単にコピーが作れてしまうことから、コンテンツの不正コピーといった著作権侵害の問題が深刻化してきている。そのため、不正コピーを防止し、デジタルコンテンツの著作権が保護される方式が必要となる。

これまでにも、不正コピーを防止する様々な著作権保護方式が提案・実現されてきた。その中でもっとも多い方式は、一切のコピー行為を認めない方式である。現在でも、ネットワーク経由でインストール時に使用できる端末のハードウェア情報を著作権管理者に登録し、常にソフトウェアを使用する端末の確認を行うプロダクトアクティベーションや、CCCDで採用され、コピーを作成しようとしてもコピー作成機器が有効に機能しないようにメディアに記録する CDS などの方式が実現されている[1,2]。今後、ユーザが複数の端末を所有し、例えばモバイル端末上で一時的に使用するような利用形態が想定される。このような利用環境において、著作権保持者の利益を損なうことなく、ユーザの私的用途を含めてコピーを限定的に許容する仕組みが必要になると考えられる。

そこで、昨今では私的範囲内のコピーが作れるよ

うな著作権保護方式が提案・実現されている。例えば SD カードや DVD-R/RW などに採用されている CPRM や OpenMG/MagicGate メモリスティックのように、耐タンパメモリ内に暗号化鍵を用意し、暗号化コンテンツを復号化する鍵をこの鍵で暗号化して記録し、耐タンパ領域に鍵を保管し、通常の方法でアクセスできないようにすることでコンテンツを保護すると同時に、コピー作成時は端末と記録メディアとで相互認証することで、この耐タンパメモリにのみ私的コピーを作れるようになっている方式がある[3,4,5]。これらの方では耐タンパハードウェアが必要となり、コピーが使用できるのもこのハードウェア上ののみとなり、ユーザの利用環境を制限することで私的コピーを実現している。

それに対し、藤井らは証明書発行局によりコンテンツと再生プログラムと所有者情報の 3 つにそれぞれ個人情報をあらかじめ埋め込み、コピーする端末の数だけ、その端末の固有情報を埋め込んだ再生プログラムと所有者証明情報を発行することにより、指定した端末でコピーしたコンテンツが使用可能な方式を提案している[6]。この方式は、すべてのコピー先の端末をあらかじめ証明書発行局に通知する必要がある。

筆者らはこれまで、不正コピーはオリジナルのコンテンツがコピーされる(これを一次コピーと定義する)ことよりも、コピーされたものからさらにコピーされ

る（同二次コピー）ことが大きな問題とされる場合が多いことに着目し、

- コンテンツの一次コピーは許容し、二次コピーを防止する。
- 特殊なハードウェアの拡張を必要としない。
- 一次コピーの数のみ制限し、コピー先の端末情報など不要なプライバシ情報は露呈しないようとする。
- オフラインでコンテンツが利用できる（コンテンツ提供側との通信をコンテンツ利用時ごとに実施する必要がない）。

の特徴を有することで、コンテンツを使用するユーザの利便性も確保した著作権保護方式の提案を行った[7]。この方式により、ユーザは使用する端末の環境に合わせてコンテンツを使用できるといった利便性を確保できると同時に、二次コピー作成を防止することでき、コピーの作成元を把握でき、さらにコピーできる台数を制限できる機能を有することでコンテンツ提供者側も著作権の保護を行いつつ柔軟なコンテンツ配信を実現できる。

今回、この提案方式に基づいたソフトウェアを作成し、これをコンピュータ上へ実装して、その安全性と性能について評価を行った。本稿ではその評価結果を報告する。以下、第2章では提案方式を説明し、第3章では実装方法とその評価結果について述べ、第4章では考察を行い、第5章でまとめと今後の課題について述べる。

2. 提案方式

2.1. 提案方式のモデル

前提案方式のモデルは次の通りである。

プレイヤは以下の3つから構成される。

Service Provider (SP) コンテンツ提供、ユーザ（正規コンテンツ使用者）登録、ユーザのアカウント管理、ユーザのセキュリティ情報（公開鍵など）管理、利用コンテンツリスト管理を実施するサーバ。

コンテンツ配信先端末 正規コンテンツ使用者がSPからコンテンツを受け取る端末。コンテンツの正当な使用と正当なコンテンツコピー作成を行うことができる。

一次コピー先端末 コンテンツ配信先端末の作成する正規なコピー（一次コピー）を受け取る端末。コンテンツの正当な使用はできるが、同内容のコピー（二次コピーなど）はできない。

使用手順は以下の通りである。

- (1) SP からコンテンツ配信先端末へ、ネットワークを介してコンテンツ配信を行う。
- (2) コンテンツ配信先端末から一次コピー先端

末へのコンテンツのコピーはネットワークあるいはローカル通信を介して行われる。ここで、コンテンツ配信先端末からコピーを行う一次コピー先端末の台数は制限される。

- (3) コンテンツ配信端末および一次コピー先端末において、コンテンツを利用する。

また、このモデルは以下の条件を前提とする。

- SP は不正を行わないものとし、それ以外のプレイヤは不正を行う可能性があるとする。
- 特殊なハードウェアを用いない。端末上で動作する各種プログラムは解析できないものとする。
- 端末固有情報は、端末固有のハードウェア情報（製造番号など）に基づくもので、端末利用者の勝手な変更、改ざんは不可であることを前提に置く。
- コンテンツ配信先端末および一次コピー先端末のメモリ上のデータは解析できないものとする。
- PKC（公開鍵証明書）や認証局は整備されており、以下の認証などの手順において、公開鍵・秘密鍵、署名鍵・検証鍵が正当に発行され、利用されるものとする。

2.2. 提案方式のプロトコル

2.2.1. 記号の説明

使用する記号は以下の通りである。

Account: コンテンツ使用者のアカウント名

DC: デジタルコンテンツ

CID: DCの識別子

p, g: Diffie-Hellman 鍵交換で使用する素数及び生成元

IDa, IDb: コンテンツ配信先端末 (*a*)、一次コピー先端末 (*b*) の端末固有情報

H(·): 端末固有情報の一方向性関数 (*IDa*または*IDb*を入力)

Ksp^a, Ka^b: SP - コンテンツ配信先端末間及びコンテンツ配信先端末 - 一次コピー先端末間の共通鍵

E(α, β): 共通鍵暗号方式により情報 α を鍵 β で暗号化したもの

SIGsp(·), SIGa(·), SIGb(·): SP、コンテンツ配信先端末、一次コピー先端末の署名

CRYsp(·), CRYa(·): 公開鍵暗号方式により SP、コンテンツ配信先端末の公開鍵で暗号化したもの

H(·): 一方向性ハッシュ関数

||: 前後の値の結合

また、上記以外の記号についてはそのつど説明する。

2.2.2. コンテンツ配信先端末への配信手順

コンテンツ配信先端末へのコンテンツ配信手順を以下に説明する。

1. 準備手続き（図1参照）

SPからは鍵生成用数値 $g||p||SIGsp(g||p)$ とアカウント $Account||SIGsp(Account)$ を、コンテンツ配信先端末からはSPから受信した鍵生成用数値と自分の IDa から鍵生成プログラムを用いて作成した公開鍵 $g^{t(IDa)} \text{ mod } p$ を作成し、それにコンテンツ配信先端末の署名をつけたものと連接しSPの公開鍵を使って暗号化した $CRYsp(g^{t(IDa)} \text{ mod } p || SIGa(g^{t(IDa)} \text{ mod } p))$ を送信する。以後、コンテンツ配信を行う時の情報として、SPはアカウント・公開鍵リストとして $Account||g^{t(IDa)} \text{ mod } p$ を、コンテンツ配信先端末ではコンテンツ使用権情報として鍵生成用数値 $g||p||SIGsp(g||p)$ とアカウント $Account||SIGsp(Account)$ をそれぞれ保存しておく。なお、この作業は初めて一回だけ行っておけばよい。

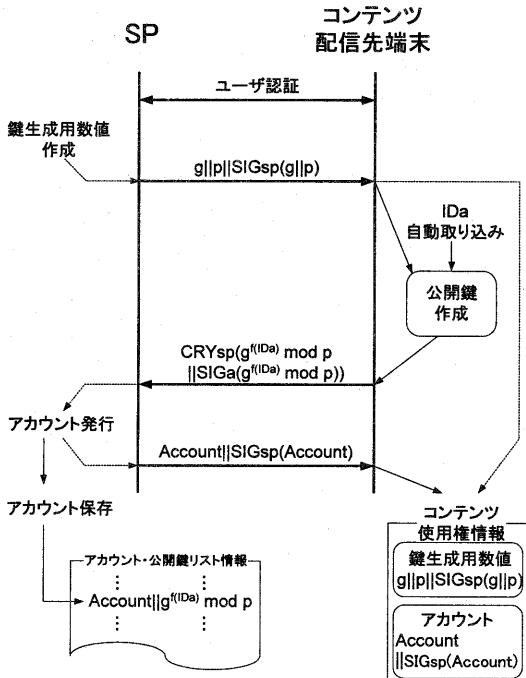


図 1. コンテンツ提供のための準備手続き

2. コンテンツ配信先端末へのコンテンツ配信（図2参照）

なお、配信時にはユーザが受信するコンテンツのリストはあらかじめコンテンツ配信先端末へ送信されており、そのコンテンツを指示すCIDは取得されているものとする。

(a) コンテンツ配信先端末からSPへアカウント

$Account||SIGsp(Account)$ を送信する。

- (b) SP はアカウント確認後乱数 t を生成し、アカウントに対応する公開鍵から暗号化鍵 $Ksp-a = g^{t(IDa)} \text{ mod } p$ を作成する。SPの公開鍵 $g^t \text{ mod } p$ と端末認証子 $E(Account, Ksp-a)$ を作成し、コンテンツ配信先端末へ送信する。
- (c) コンテンツ配信先端末は配信してもらうCIDを選択し、受信した情報と鍵生成用数値から $E(CID, Ksp-a)$ を作成する。これと保存しているアカウントを SPへ送信する。
- (d) SP は $E(CID, Ksp-a)$ を確認後、暗号化コンテンツ $E(DC, Ksp-a)$ と発行証明書 $SIGsp(H(DC))$ を作成しコンテンツ配信先端末へ送信する。
- (e) コンテンツ配信先端末は蓄積情報として暗号化コンテンツ $E(DC, Ksp-a)$ 、SPの公開鍵 $g^t \text{ mod } p$ 、発行証明書 $SIGsp(H(DC))$ 、端末認証子 $E(Account, Ksp-a)$ を保存する。

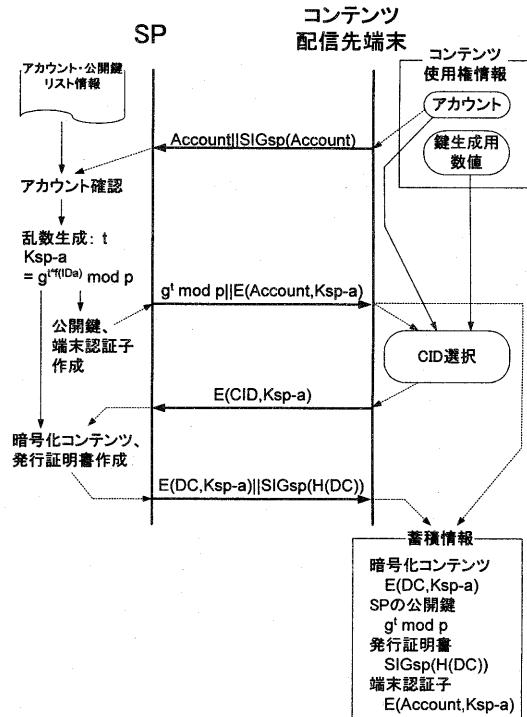


図 2. コンテンツ配信先端末へのコンテンツ配信情報 (セキュリティ情報のみ)

2.2.3. コンテンツ使用手順

コンテンツ使用手順を以下示す（図3参照）。

1. コンテンツ配信時に蓄積された情報のうち、暗号化コンテンツ $E(DC, Ksp-a)$ 、SPの公開鍵 $g^t \text{ mod } p$ 、発行証明書 $SIGsp(H(DC))$ の3つと、自動的に取り

込まれるコンテンツ配信先端末の端末固有情報 IDa をコンテンツ閲覧時の入力値とする。

2. IDa から一方向性関数を通して $f(IDa)$ を求め、この値と $g^t \bmod p$ から、暗号化コンテンツの復号化鍵 $Ksp-a = g^{t*f(IDa)} \bmod p$ を生成する。
3. 2 で作成した鍵で暗号化コンテンツ $E(DC, Ksp-a)$ を復号し DC を得る。
4. 3 で得られた DC とプログラムに埋め込まれてある検証鍵で発行証明書 $SIGsp(H(DC))$ の署名を検証する。
5. 4 の検証に成功したらコンテンツの使用を開始する。ただし DC はファイルとして残らない。

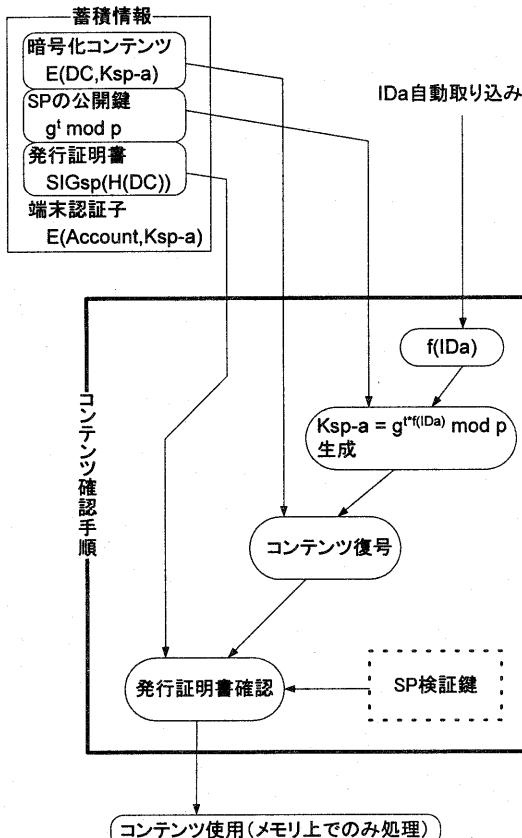


図 3. コンテンツ使用手順

上記の手順は、次に示すように各パラメータを置き換えれば、一時コピー先端末でのコンテンツ使用の手順となる。

$$IDa \Rightarrow IDb$$

$$g^t \bmod p \Rightarrow g^{f(IDa)} \bmod p$$

$$Ksp-a = g^{t*f(IDa)} \bmod p \Rightarrow Ka-b = g^{(f(IDa)*f(IDb))} \bmod p$$

$$E(DC, Ksp-a) \Rightarrow E(DC, Ka-b)$$

2.2.4. 一次コピー先端末へのコピー送信手順

コンテンツのコピー手順を以下に示す(図 4 参照)。

1. コンテンツ配信先端末は、コンテンツ使用権情報として保存してあるセキュリティパラメータのうち、鍵生成用数値から $g||p||SIGsp(g||p)$ を作成し、一次コピー先端末へ送信する。
 2. 一次コピー先端末は g, p の値の検証に成功すると端末固有情報 IDb を取り込み、鍵作成プログラムにより端末の公開鍵 $g^{f(IDb)} \bmod p$ を作成し、これに署名をつけたものとを連接し、コンテンツ配信先端末の公開鍵で暗号化した $CRYa(g^{f(IDb)} \bmod p || SIGb(g^{f(IDb)} \bmod p))$ をコンテンツ配信先端末へ送信する。
 3. コンテンツ配信先端末は、2 で受信したものから $g^{f(IDb)} \bmod p$ を得て、これと発行証明書を用いて $H(g^{f(IDb)} \bmod p) || SIGa[SIGsp(H(DC)) || H(g^{f(IDb)} \bmod p)]$ を作成し、これをコピー作成申請として SP に送信する。
 4. SP は 3 で受信したものから $H(DC)$ を確認し、コピー記録と $H(g^{f(IDb)} \bmod p)$ を比較し、同じものがあるか、コピー台数の制限内である場合は $SIGsp(H(g^{f(IDb)} \bmod p))$ を作成し、コピー許可証としてコンテンツ配信先端末に送信する。そのとき、 $H(g^{f(IDb)} \bmod p)$ がコピー記録に無い場合は $SIGa[SIGsp(H(DC)) || H(g^{f(IDb)} \bmod p)]$ を保存することでコピー登録を行い、残りのコピー台数の数を 1 減らす。
 5. コンテンツ配信先端末は、2 で一次コピー先端末から受信したものから公開鍵、3 で SP から受信したコピー許可証、コンテンツ使用権情報として保存してある鍵生成用数値、蓄積情報として保存してある暗号化コンテンツ、SP の公開鍵、発行証明書、端末認証子、自動で取り込まれる IDa をコピー作成時の入力値とし、コピー許可処理を起動する。
 6. コンテンツ配信先端末のコピー許可処理では、上記 5 の入力値からコンテンツ配信先端末の正当性などを確認し、再暗号化コンテンツ $E(DC, Ka-b)$ 、コンテンツ配信先端末の公開鍵 $g^{f(IDa)} \bmod p$ 、発行証明書を出力する。なお、コピー許可処理の詳細については後述する。
 7. コンテンツ配信先端末は 6 で作成したものを $E(DC, Ka-b) || g^{f(IDa)} \bmod p || SIGsp(H(DC))$ として一次コピー先端末へ送信する。
- コピー作成はコピー配信先端末でしか行えない。そこで上記手順 6 で端末認証子を用い、コピー作成に使用するその他の情報と検証することでコピー配信先端末であることを確認する。

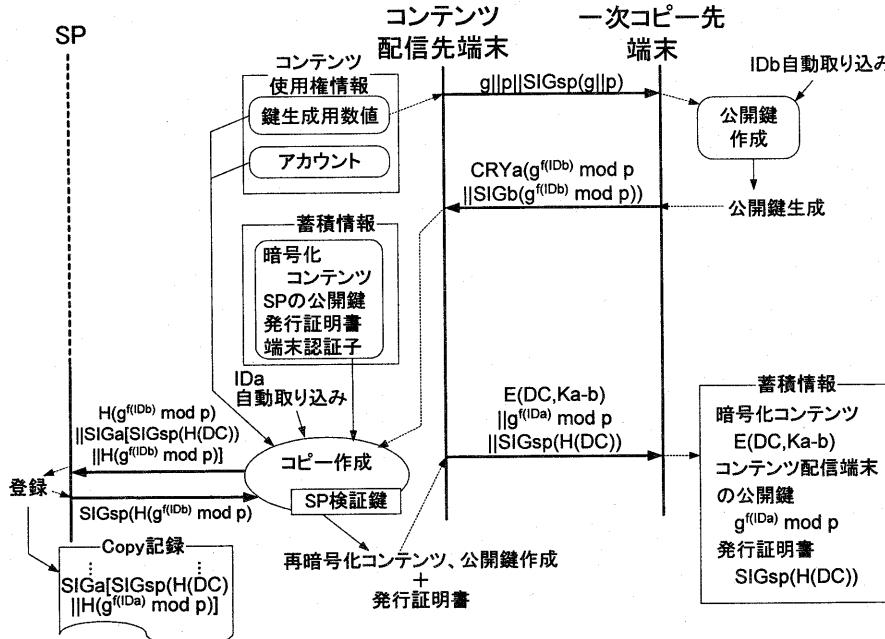


図 4. コピー送信手順（セキュリティ情報のみ）

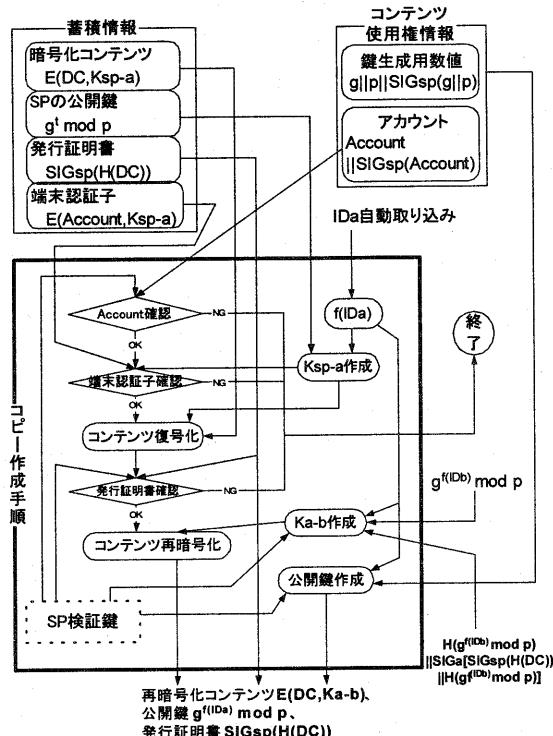


図 5. コピー許可処理の概要

ここでコピー許可処理は、動作している端末がコンテンツ配信先端末であることを認証し、コンテンツの再暗号化を行う処理である。以下にコピー許可処理の手順を示す（図 5）。

1. 入力した SP の公開鍵 $g^t \bmod p$ とコンテンツ配信先端末の固有情報 IDa から、復号化鍵 $Ksp-a = g^{f(IDa)} \bmod p$ を生成する。
2. 入力したアカウント $Account||SIGsp(Account)$ を検証する。
3. 入力した端末認証子 $E(Account, Ksp-a)$ を 1 で作成した $Ksp-a$ で復号化し、2.で検証に成功した $Account$ と比較することで、正当なコンテンツ配信先端末であることを確認する。
4. 3 の確認に成功したら、入力した暗号化コンテンツ $E(DC, Ksp-a)$ を $Ksp-a$ で復号化し、 DC を得る。
5. 入力した発行証明書 $SIGsp(H(DC))$ を 4 で得た DC で検証する。
6. 入力したコピー許可証 $SIGsp(H(g^{f(IDb)} \bmod p))$ と一次コピー先端末の公開鍵 $g^{f(IDb)} \bmod p$ を使って、コピー許可証を検証する。検証が成功したら $f(IDa)$ とで暗号化鍵 $Ka-b = g^{f(IDa)*f(IDb)} \bmod p$ を作成する。
7. 5 の検証が成功したら、6 で作成した $Ka-b$ でコンテンツを再暗号化し、 $E(DC, Ka-b)$ を得る。
8. 鍵生成用数値 $g, p, SIGsp(g|p)$ を入力し、署名の検証に成功したら、コンテンツ配信先端末の公開鍵

$g^{f(ID_a)} \bmod p$ を生成する。

9. 発行証明書 $SIG_{sp}(H(DC))$ と、7 で得た再暗号化コンテンツ $E(DC, Ka \cdot b)$ 、8 で得たコンテンツ配信先端末の公開鍵 $g^{f(ID_a)} \bmod p$ を出力する。

2.2.5. ニセコピーアンチソリューションについて

2.2.4 節のコピー許可処理 1~3 において SP が許可した正当な端末の場合のみ端末認証子が確認されるので、不正な端末ではこの当該処理を実行できない、すなわち、ニセコピーが行えないようになっている。また、ニセコピーを行おうとしている端末が過去にコンテンツを配信されていて、その時に取得したアカウントと端末認証子を用いてコンテンツをコピーしようとしても、手順 4 で暗号化コンテンツを復号化する鍵とは異なるので 5 で発行証明書の検証に失敗することになり、同様にコピーが行えないようになっている。

3. 提案方式の実装と評価

3.1. システム実装

本提案方式に基づいて、ユーザやコンテンツの管理およびコンテンツ配信を行うサーバソフトウェアと、コンテンツのダウンロード・使用・コピー作成を行うクライアントソフトウェアを実装した。開発環境として Visual C++ Ver.6.0 を用いて Windows 2000 上で実装され、MPEG および MP3 のコンテンツを対象とする。また、提案方式であるプロトコルは TCP/IP 上で動作させる。実装に際しては、利便性・安全性の観点から以下の方針に従った。

1. メモリ上のデータは解析できないという前提に従い、クライアントソフトウェアの処理は機密情報や復号化されたコンテンツをメモリ上でのみ保持することとし、安全性の向上を図った。
2. ダウンロード時のコンテンツ配信先端末、およびコピー時の一次コピー先端末でのユーザビリティの向上を目的として、コンテンツのダウンロードおよび一次コピー先処理を逐次的に行うこととした。すなわち、コンテンツを一定サイズごとのブロックに分割し、逐次的に暗号化・発行証明書作成の処理を行いながらダウンロードする。発行証明書はブロックごとに全部作成する。
3. 様々な端末で使用可能とするため、端末固有情報は汎用的に ID を取得できる HDD および NIC の MAC アドレスを用いることとした。
4. 使用しているこれらの方針の安全性が保てなくなった場合でも、他の安全な方式に変更できるよう、暗号アルゴリズムの処理部をモジュールとして独立化した。

サーバソフトウェアは SP に実装され、以下の機能から構成される。

●セキュリティ機能

SP から受信した情報の復号や署名検証を行い、コンテンツの暗号化およびコンテンツ暗号化用の鍵、発行証明書、端末認証子、コピー許可証を作成時に使用するセキュリティ機能を提供する。

●通信機能

コンテンツ配信先端末からの接続要求があるまで待機し、接続したらプロトコル手順に従ってアカウント発行、コンテンツ配信、コピー許可証の通信を行う。

●ユーザ管理機能

ユーザの氏名や電話番号などのプライバシ情報、発行したアカウント、配信したコンテンツとそのコピー情報の登録、変更、一覧機能を提供する。また、ユーザに対しコンテンツの許容コピー数の設定できることとした。

●コンテンツ管理機能

コンテンツそのものの他に、そのコンテンツを示す識別子やライセンス情報の登録・変更・一覧機能を提供する。

ユーザおよびコンテンツの管理はウェブブラウザインターフェースを用いた。また、SP 側のサーバソフトウェアの画面例を図 6 に示す。

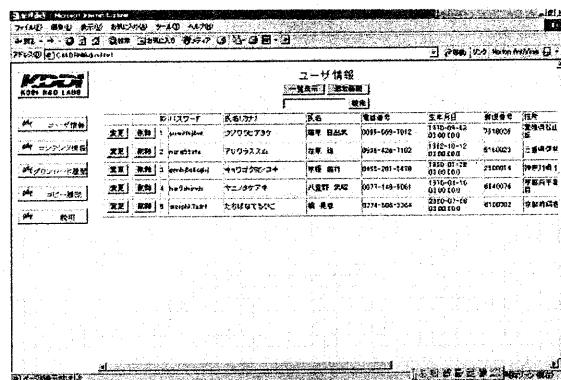


図 6. サーバソフトウェア起動画面

クライアントソフトウェアはコンテンツ配信先端末および一次コピー先端末に実装され、以下の機能から構成される。

●セキュリティ機能

暗号化コンテンツを復号化し、コンテンツ暗号化用の鍵、CID 選択、コピー作成申請、コピー作成時に使用するセキュリティ機能を提供する。

●通信機能

コンテンツ配信先端末および一次コピー先端末

で実行される手順 2.2.4 節に従ってコピー送受信を行う機能と、コンテンツ配信先端末で実行される、SP と接続しダウンロード可能なコンテンツの一覧と各コンテンツのサムネイルを表示・選択する機能および選択されたコンテンツを 2.2.2 節の手順に従いダウンロードする機能を提供する。

● コンテンツプラウジング機能

端末にあるコンテンツを一覧表示する機能で、2.2.3 節の手順に従い、選択されたコンテンツを復号化し表示する機能を提供する。

● コンテンツコピー機能

コンテンツ配信先端末でのみ行える機能で、コンテンツと一次コピー先端末の一覧を表示する機能を有し、この二つを選択することにより 2.2.3 節の手順に従い、その端末用の再暗号化コンテンツを作成する機能を提供する。

クライアントソフトウェアの画面例を図 6 に示す

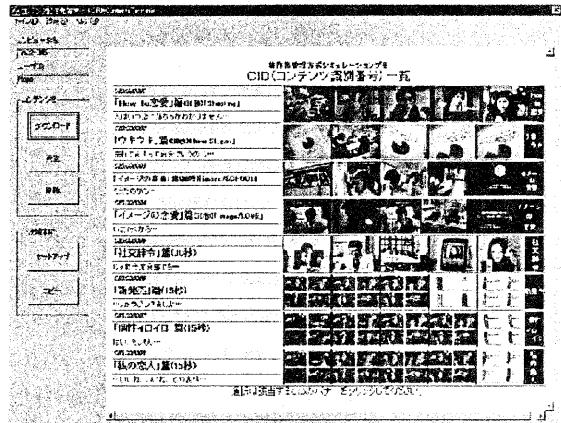


図 7. クライアントソフトウェア起動画面

今回作成したソフトウェアのファイルサイズは、サーバソフトウェアが 248KB、クライアントソフトウェアが 280KB となっている。また、各機能で使用するモジュールおよびサイズを表 1 に示す。

表 1. 機能モジュールのサイズ

ソフトウェア種別	機能	サイズ
サーバ	セキュリティ	116KB
	通信	68KB
	ユーザ管理	92KB
	コンテンツ管理	(共通)
クライアント	セキュリティ	116KB
	通信	68KB
	コンテンツプラウジング	64KB
	コンテンツコピー	76KB

3.2. 提案方式の評価

3.2.1. テスト環境

テスト環境は以下の通りである。

- SP 側マシン

Pentium4 2.4GHz

1GB DDR266 SD-RAM

OS : Windows 2000

- ユーザ側端末

Pentium3 850MHz

512MB SD-RAM

OS : Windows 2000

- ネットワーク環境

100BASE-TX 対応のスイッチングハブを用い、SP 側マシンとユーザ側端末はそのハブに接続。スイッチングハブの実行スループットはおよそ 50Mbps。

- コンテンツファイル

MPEG ファイルを使用

- 使用アルゴリズム

暗号化方式 : DES

署名方式 : RSA

一方向性ハッシュ関数 : SHA-1

3.2.2. 性能評価

性能について、以下の観点から 2 種類のスループットを計測した。

1. 端末でのコンテンツ再生

閲覧において、コンテンツの復号化や発行証明書のチェックの処理を行っても、動画の再生に問題ないようにしなければならない。そこで暗号化されたコンテンツから、コンテンツを復号化し、そのダイジェストを求めて発行証明書とのチェックを行い、トータルのスループットを求めた。150MB の MPEG ファイルを使用して計測を行った結果、50Mbps のスループットが確認できた。

ここでダウンロードしながら逐次再生する場合において、ダウンロード開始からコンテンツが表示されるまでの時間は 5 秒程度であり、コンテンツ全体を受信して復号化するよりもはるかに短い時間で再生を開始することが可能となり、逐次処理の効果が確認できた。

2. サーバ側マシンの配信処理

コンテンツを固定ブロックに分割して逐次的に暗号化・発行証明書作成を行うサーバ側マシンの配信処理のスループットを計測した。図 8 はその結果を示したものである。

ブロックサイズが 128KB になるまでは、ブロック

クサイズが大きくなればなるほどスループットが上がっていく。ブロックサイズが128KB以上はほぼ一定のスループットになり、10MBのMPEGで16Mbps以上、150MBのMPEGで14Mbpsになった。

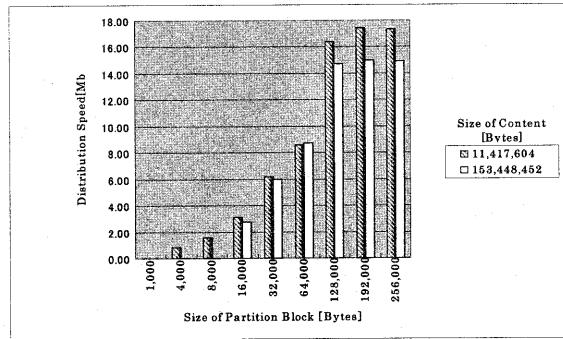


図8. ブロックサイズとスループットの関係

4. 考察

4.1. 安全性の考察

コピー作成プログラムを起動するにはアカウントと端末認証子が必要である。ここで、コンテンツ配信先端末以外の端末が蓄積情報一式を不正に入手した場合、このプログラムが生成する復号化鍵で端末認証子を正しく復号することができない。よって、端末認証子の確認が失敗することになる。また、その不正を行おうとしている端末が、過去にこれをコンテンツ配信先として入手したアカウントと端末認証子を用いようとしても、端末認証子を復号した鍵で暗号化コンテンツを復号化できないため、発行証明書の確認で失敗し、同じくコピーが行えない。

以上のことから、不正に入手したアカウントと端末認証子を用いても、二次コピーの作成は行えないようになっており、二次コピーを防止するという目的が達せられている。

4.2. 性能評価の考察

ダウンロード時のスループットが50Mbps得られており、これはデジタルハイビジョン用コンテンツのスループット20Mbpsと比較しても、MPEGの再生に問題ない速度が得られており、現状ほぼあらゆる種類のコンテンツ再生に適応できると考えられる。

ユーザ側端末でのコンテンツ配信については、128KBごとにブロック処理を行った結果、14Mbps得られている。この速度はDVD-VideoなどのMPEG2のビットレート6Mbpsと比較しても充分な速度である。このことは、ダウンロード用としてだけでなく、放送用コンテンツにも適応できると考えられる。また、ブ

ロックサイズが128KB以上でスループットがほぼ一定になっている。これは、ブロックサイズが小さいほど、暗号化や発行証明書の処理に負荷がかかるためと考えられ、実験環境では128KBのブロックサイズを用いることが望ましい。

暗号アルゴリズムについて今回は評価目的でDESを用いて性能計測を行ったが、AESなどのより安全かつ高速なブロック暗号での評価を進めている。

また、今回はすべてのブロックごとに発行証明書作成を行ったが、これはスループットの制限につながると同時に、ユーザの端末に蓄積される発行証明書のサイズが膨大なものになる。これを回避するために、より効率的なメッセージ認証方式を採用するなどの検討が必要である[8]。

5.まとめ

筆者らが提案した方式に基づいて、PC上で動作するソフトウェアを作成し、安全性の確認と性能の評価を行った。その結果、機密データや復号化されたコンテンツをメモリ上で管理することにより実装の観点からも安全なシステムが構築できた。また、性能については再生・配信の処理を行っても通常のコンテンツ使用に影響がないであろうと予測できるスループットが得られていることが確認できた。

今後は、SPでの負荷などスケーラビリティの観点から検討・評価を行うとともに、更なるスループット向上のためにソフトウェア実装の効率化を行う予定である。

文 献

- [1] マイクロソフト プロダクトアクティベーション, <http://www.microsoft.com/japan/windowsxp/pro/techinfo/productactivation.asp>
- [2] Macrovision The CDS System, <http://www.macrovision.com/solutions/audio/system.php3>
- [3] 4C Entity CPRM, <http://www.4centity.com/>
- [4] Open MG, <http://www.openmg.com/>
- [5] “メモリスティック著作権保護技術 MagicGate,” <http://www.sony.co.jp/Products/SC-HP/CXPAL/CXPAL-44/PDF/tw.pdf>
- [6] 藤井治彦, 武井英明, 中里加奈, 庵祥子, 三宅延久, “自由度の高い私的コピーを実現する著作権保護方式,” 情報処理電子化知的財産・社会基盤, No.11-9, pp.39-46, Feb.2001.
- [7] 稲村勝樹, 田中俊昭, 中尾康二, “デジタルコンテンツにおける不正コピー防止方式の提案,” SCIS2003, Vol. II, pp.1065-1070, 2003.
- [8] 田中俊昭, 中尾康二, 清本晋作, “ストリーミング転送における効率的なメッセージ認証方式の検討,” ISEC, Jul. 2001.