高品質インターネット　1－25
（2001. 11. 21）

IPマルチキャストを用いた放送システムにおけるPay Per Viewの実現

川北 良一† 辻 義一‡ 上原 哲太郎‡ 佐藤 敬§ 山岡 克式¶
泉 裕# 齋藤 彰一‡ 國枝 義敏‡ 結城 皖曠¶

†和歌山大学大学院 システム工学研究科 ‡和歌山大学システム工学部 #和歌山大学システム情報学センター
〒640-8510 和歌山県和歌山市栄谷 930
§北九州市立大学 国際環境工学部 〒808-0135 福岡県北九州市若松区ひびきの 1-1
¶文部科学省メディア情報教育開発センター 研究開発部 〒261-0014 千葉県千葉市美浜区若葉 2-12

E-mail: †‡ {s011013,s031036,tetsu,shoichi,kunieda}@sys.wakayama-u.ac.jp
§tsatoh@env.kitakyu-u.ac.jp ¶ {yamaoka,yuki}@nime.ac.jp # yutaka@center.wakayama-u.ac.jp

あらまし インターネットの広帯域化に伴い、映像や音楽のストリーム放送も可能になってきた。しかし、現存のほとんどの放送システムは通信にユニキャストを用いており、多数のユーザに同時にサービスを行うのが困難である。この問題を解決するには IP マルチキャストの利用が有効であるが、これを用いて有料サービスを行おうとすると、システムの安全性を保つために考慮しなければならない点は多い。本論文では、IP マルチキャストを用いた Pay-per-view システムのプロトタイプとして製作した音楽放送システムの実装について述べる。

キーワード インターネット、IP マルチキャスト、Pay-per-view システム、ストリーム通信、セキュリティ、暗号化

Implementation of a Pay-per-view Broadcasting System via IP Multicast

Ryoichi Kawakita† Yoshikazu Tsuji‡ Tetsutaro Uehara‡ Takashi Satoh§
Katsunori Yamaoka¶ Yutaka Izumi# Shoichi Saito‡ Yoshitoshi Kunieda‡ Kiyohiro Yuki¶

†Graduate School of Systems Engineering ‡Faculty of Systems Engineering #Center for Information Science
Wakayama University 930 Sakaedani, Wakayama-city, 640-8510 Japan
§ The University of Kitakyushu 1-1 Hibikino, Wakayamtsu-ku, Kitakyushu-city, 808-0135 Japan
¶ National Institute of Multimedia Education 2-12 Wakaba, Mihama-ku, Chiba-city, 261-0014 Japan

Abstract Recently the Internet has become broadbanded enough to enable streaming services of audio-visual contents. But most of the existing services are realized via unicasting and it is difficult to serve huge number of users simultaneously. IP multicasting can be utilized to solve this problem, but if we want to serve the contents only for authorized users, there are a number of considerations to keep the system secure. This paper describes the implementation of a music broadcasting system as a prototype of pay-per-view broadcasting systems via IP multicast.

Keywords Internet, IP multicast, Pay-per-view system, Streaming, Security, Encryption

1. はじめに

近年、インターネットの広帯域化が進むにつれて、インターネット上での動画像や音声をリアルタイムで伝送する、いわゆるストリーム通信が実現可能になってきた。現在のインターネット放送システムでは、コンテンツの送信に数十kbpsから数百kbps程度の帯域しか利用できず、品質の高い動画像の送信には十分とはいえない。しかし今後Fiber To The Homeが実現してゆくにつれ、近い将来各家庭へのアクセスラインも数Mbpsから数十Mbpsへ広帯域化され、現在のテレビ画像と同等の品質での配信が可能になると考えられる。

インターネットでの放送サービスとしては、現在でもReal Networks, Windows Media, QuickTimeなどを用いた比較的狭帯域のシステムがいくつか実用化されている。これらのシステムの多くは通信にユニキャストを用いており、ビデオ・オン・デマンドやミュージック・オン・デマンドを単に同報的にしたシステムとなっている。このようなシステムでは、サーバから送出されるデータ転送量がユーザ数に比例して増大するため、コンテンツを非常に多数のユーザに同時に配信するのは現実的ではない。この問題は、今後コンテンツを高帯域化させるに従って次第に深刻になると考えられる。よって、ユーザ数に対してスケーラブルに対応できる本格的インターネット放送システムの実現には、IPマルチキャスト[1]の利用が必要不可欠である。IPマルチキャストはIPv4では実験段階にあるが、IPv6では標準機能として実装されているため、今後ますます普及が進み、一般に利用が可能になると予想できる。

このように、インターネットの広帯域化とマルチキャストの普及によって、インターネットによる動画像などの放送システムは近い将来実用段階を迎え、ビジネスとして成立するようになると考えられる。そこで筆者らは、インターネットにおける有料放送サービスの実現可能性に着目し、予想される技術的な問題について考察を重ねてきた。特に、現在のIPマルチキャストを利用してシステムを構築した場合に、正規のユーザのみに視聴可能な暗号化システムについて検討し、安全性とトラフィックの双方から評価を行ってきた[2]。

本研究では、IPマルチキャストを用いてPay-per-viewシステム、すなわちコンテンツごとに課金するシステムの実現手法について述べるとともに、さまざまな有料放送システムを実装、評価するプラットフォームとして試作したIPマルチキャスト向け音楽放送システムMusicCast/ASについて述べる。

2. インターネット有料放送システム

ここでは、本研究で提案する有料放送システムが目標とするサービスの形態と、その実現の前提にしたインフラストラクチャや技術について述べる。

2.1 放送システムのモデル

本研究で考えるインターネット放送システムは、以下のようなサービスを行うものとする。

①本研究で考えるシステムでは、現在のインターネット放送システムよりもはるかに多い、たとえば10万人規模のユーザが同時に動画像を視聴できるシステムである。よって、コンテンツは、マルチキャストを用いて全クライアントが全く同一のものを同時に受信するものとする。よって、ユーザがコンテンツを受信するには、放送時間中にクライアントをシステムに参加させなければならない。また、サーバが送出するデータの量はユーザ数に比例して増加することはなく、十分なスケーラビリティが確保されていなければならない。

②このシステムの対象となるコンテンツは動画像や音声といったストリーム通信向けのデータであり、コンテンツの全データが全クライアントに届くことを必ずしも保証しなくてよいものとする。各クライアントはデータの欠落が発生した場合も、サーバに欠落データの再送を要求できない。ユーザにはデータの欠落はコンテンツの品質劣化として観測されるが、欠落部分以外のデータは正しく受信、復号し再生できるものとする(以下コンテンツを正しく受信、復号、再生することを単に「視聴する」と言う)。

③ユーザに対する課金方式は、2通り考えられる。一つはいわゆるPay-per-view方式や月単位契約方式のように、料金と引き替えに、ある特定のコンテンツの視聴、あるいはある一定の期間の視聴のための権利とな

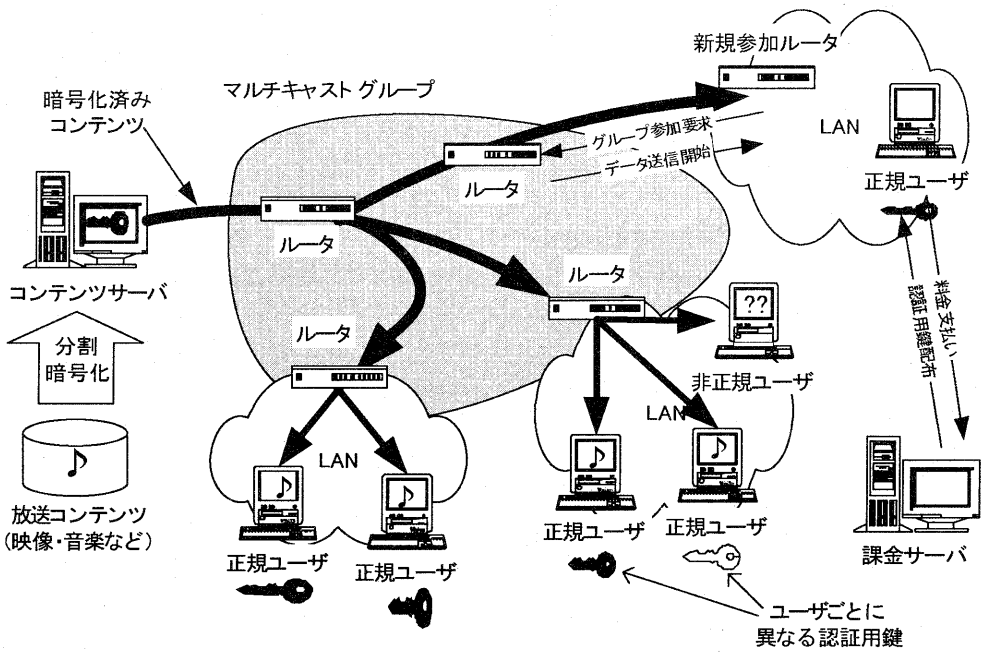

図 1　インターネット有料放送システムの概念図

Fig 1: The big picture of a pay-per-view broadcasting system on the Internet

る鍵を受け取る方式である。この場合、一度ユーザに与えられた権利(鍵)は、サーバ側から無効にすることができず、またユーザにとってもない。その代わり、ユーザはコンテンツの送信の途中であっても、料金を払えば視聴が可能である。たとえば月契約において、月初めからでなくとも参加が可能であるとする。この方式をここでは単にPay-per-view 方式と呼ぶ。

もう一つの方式は、ユーザが任意の時点でシステムに参加したり脱退したりできる方式である。サーバ側からみると、任意の時点で特定のユーザの視聴を即座に許可したり禁止したりできる方式である。これはたとえば秒単位のようなごく短い時間単位で課金するシステムや、月単位であっても契約途中でのユーザのキャンセル(課金の返金)に応じるシステムが考えられる。この方式をここではJoin-leave 方式と呼ぶことにする。

いずれの方式においても、各ユーザには鍵が配布されるが、この鍵が全ユーザで同じものが配布された場合、正規ユーザから鍵を再配布することによって、容易に不正ユーザもコンテンツの視聴が可能になってしまう。そこでここではユーザ毎に内容が異なる鍵を配布するものとする。これをここでは**認証用鍵**と呼ぶことにする。

このようなシステムの概念図を図1 に示す。以後、このようなサービスを行うために考慮すべき問題を議論する。

2. 2　前提とするインフラストラクチャ

次に、このようなシステムの構築のために利用する技術について述べる。

ここで考えるシステムでは、前述の通り、インターネットでの放送にスケーラビリティを持たせるために、データ送信には IP マルチキャストを利用している。現状の IP マルチキャストは、データの配送はマルチキャストグループと呼ばれる単位で管理されている。このマルチキャストグループへは、クライアントからその LAN 内のルータに対して参加要求が発行され、ルータ間で配送経路が確立される。この際に用いられるのが IGMP[3]と呼ばれるプロトコルである。

しかしこの IGMP には、現在ユーザ認証機構は存在せず、マルチキャストアドレスさえわかっていれば、どのクライアントでも無条件でもマルチキャストグループに参加できる。IGMP にユーザ認証を加える拡張は提案されている[4]が、標準として採用されるには至っておらず、実際には利用できない。また、仮に IGMP にユーザ認証機構があったとしても、LAN 内ではマルチキャストデータはブロードキャストされるため、正規ユーザが視聴に使用している LAN に接続している他のクライアントは、容易に配送データを傍受できる。

本研究では、利用できるインフラストラクチャは、通常の IGMP によって管理された現在のインターネット環境をそのまま利用することとする。よって、サーバからのデータ配送は、非正規ユーザの傍受に耐えるよう、適切に暗号化されている必要がある。一対一通信の場合の暗号化には共通鍵方式と公開鍵方式が挙げられるが、このシステムでは一対多の通信となるため公開鍵方式は利用できない。そこで、コンテンツに共通鍵を用いて暗号化することとする。このコンテンツを復号するための共通鍵を、以下では**セッション鍵**と呼ぶ。このセッション鍵を、何らかの方法で正規ユーザだけに安全に配布できれば、この有料放送システムを実現できる。つまり、本研究におけるユーザ認証の問題は、このセッション鍵の配布をいかに行うかという問題に帰着できる。

また、本システムでは、クライアントは通常の PC や PDA などのプログラム可能な機器にソフトウェアを組み込んで実現することを想定している。つまり、ユーザ認証のための特殊なハードウェアなどの存在を仮定していない。

2.3 システムの評価項目

ここでは、本研究で提案するシステムを設計する上でトレードオフとなる評価項目を挙げる。

(1) トラフィック

システムをユーザ数に対してスケーラブルにするためには、ユーザ数にトラフィックが大きく影響を受けないことが望ましい。ここでは、コンテンツは共通のセッション鍵で暗号化されるため、コンテンツ自体が配送される際のトラフィックはユーザ数によらず一定となる。しかし、セッション鍵を暗号化してマルチキャストデータとして同時に配布する場合には、ユーザ数の増加につれてトラフィックも増加する場合がある。本研究では、ユーザ数 n に対して暗号化後のセッション鍵のサイズが O(log n)以下に抑えられる暗号化方式のみを扱うことにする。

また、各ユーザに配布する認証用鍵の配布にかかるトラフィックも考慮する必要がある。最初にシステムを動作させる際には、全ユーザに一度は認証用鍵を配布しなくてはならないため、そのトラフィックは認証用鍵のサイズとユーザ数の積となる。それ以降は、一度配布した認証用鍵を課金の機会ごとに破棄しなおして更新させるか、あるいは一度ユーザに配布した認証用鍵を複数の課金に渡って再利用させるかによってトラフィックが変わるが、いずれにせよこれを低く抑えることが重要である。

(2)プロトコル

コンテンツは、スケーラビリティの確保と、配送に信頼性を要求していないことから、マルチキャスト上で UDP によって配送する。セッション鍵や認証用鍵の配送はプロトコルによってシステムへの負荷やトラフィックが変化するが、これらをできるだけ低く抑えることも評価項目として重要である。

(3)認証用鍵のサイズ

認証用鍵は各クライアントが保持しなくてはならないが、クライアントのストレージは PDA などの場合には十分確保されるとは言いがたい。さまざまなタイプのクライアントをサポートするためには、この認証用鍵のサイズが極端に大きくならないようにシステムを設計する必要がある。

(4)不正に対する耐性

安全性を確保するためには、非正規ユーザが何の情報も持たずにセッション鍵を入手できる可能性は限りなく小さくなければならない。また、認証用鍵は各ユーザによって異なるが、正規ユーザが何人か結託して非正規ユーザのための認証用鍵を新たに生成できる場合がある。このようなことはないほうが望ましいが、避けられない場合は結託するために必要なユーザ数ができるだけ多いほうがよい。

システム構築の際には、このような評価項目に基づいて、認証用鍵、セッション鍵の配布方法を考える必要がある。

3．セッション鍵配布方式の比較

前節で述べたような評価基準に従って本研究では、いくつかの鍵管理・配布方式について検討してきた。文献[2]で発表したとおり、既に GKMP[5][6]、木構造による管理[7][8]、ISAKMP[9]および放送型暗号[10]を比較した結果、Join-leave 方式を採用する限り、放送型暗号がもっとも本システムの実現に適していると判断した。放送型暗号では、セッション鍵の配布にかかるトラフィックと認証用鍵のサイズがトレードオフになるため、システムの状況に応じてどちらを優先するか設定が可能である（トラフィック優先符号化・ストレージ優先符号化）。

さらに今回、Pay-per-view 方式、すなわち課金単位ごとに認証用鍵を破棄し再配布する方式において、Tracing-Traitors 方式[11]のうち Open One-Level Scheme（以下 TT-OOLS と呼ぶ）を使用した方式を実装し、良好な結果を得た。その詳細については後述する。

3．1 TT-OOLS 法の概要

n 人までのユーザをサポートする TT-OOLS 法の概要は、簡単な例で示すと以下のとおりである。

①n 人までの各ユーザに配布する認証用鍵を作るため、まずサーバで $2\log n$ 個の鍵 $a_1^0, a_1^1, a_2^0, a_2^1, \cdots, a_{\log n}^0, a_{\log n}^1$ を作成する。これを図2 のような行列で表す。

a_1^0	a_1^1
a_2^0	a_2^1
⋮	⋮
$a_{\log n}^0$	$a_{\log n}^1$

図 2　TT-OOLS 法における鍵の行列

Fig 2: A Key Matrix for TT-OOLS

②各ユーザに ID 番号を付与し、それに基づいてこの $2\log n$ 個の鍵のうち $\log n$ 個を各ユーザの認証用鍵として配分する。ユーザID が i であった場合、i は $\log n$ 桁の 2 進数で表せる。この 2 進数の下位から j 番目の桁が 0 なら a_j^0 を、1 なら a_j^1 を付与する。たとえば $n=16$ のときにユーザID が 6 なら、そのユーザは $a_1^0\ a_2^1\ a_3^1\ a_4^0$ の 4 つの鍵を認証用鍵として得る。結果として、ユーザ ID が異なれば保持している認証用鍵は異なる。ここまでの処理をセッション開始までに行っておく。

③セッションが開始されると、この鍵の行列を用いて、サーバがセッション鍵をブロードキャストで送信する。セッション鍵を S とすると、この S を $\log n$ 個の鍵 $s_1, s_2, \ldots, s_{\log n}$ に分割する。これらの鍵をそれぞれ、j 番目のセッション鍵 s_j を a_j^0, a_j^1 それぞれを用いて暗号化し、e_j^0, e_j^1 を得る。この暗号化された $2\log n$ 個のセッション鍵で図 2 と同様の行列を作成し、これを送信する。

④受信したクライアントでは、暗号化されたセッション鍵の行列を受け取り、その中からユーザ ID に相当する要素を取り出して、保持している $\log n$ 個の認証用鍵で復号化する。その結果を結合することにより、セッション鍵 S を得ることが出来る。

この方式によれば、n 人のユーザに対してセッション鍵を安全に配布するためには、必要となる認証用鍵のサイズは $O(\log n)$ でしか増加せず、セッション鍵の放送にかかるトラフィックも定数オーダ（この場合は 2 倍）でしか増加しないため、非常にトラフィックの増加を低く抑えられる。

しかし反面、この方式は正規ユーザ同士の結託があると簡単にサーバが持つ元の鍵の行列が判明してしまう恐れがある。ここで述べた例の場合、ユーザ ID がお互いにビット反転した番号になっている 2 名のユーザが結託するだけで、元の鍵の行列が復元でき、自由に新しいユーザ用の認証用鍵が生成できてしまう。この対策には、ユーザIDと行列との対応付けをする際に一方向性関数を導入したり、鍵の行列の列数を増やしたりすることで結託に対する安全性を上げることができる。

4．MusicCast/AS の概要

これまで述べてきたように、本研究ではマルチキャストを用いたインターネット放送システムにおけるユーザ認証機構、すなわちセッション鍵およびユーザ認証用鍵の配布機構を評価してきた。これらを

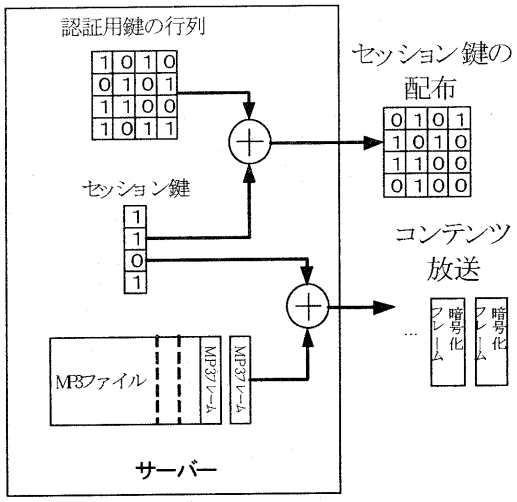

図 3 サーバ内の処理

Fig3: Processed in the broadcasting server

実装によって評価、有用性を検証するため、マルチキャストを用いたユーザ認証付き音楽放送システム MusicCast/AS を開発した。以下、その実装について述べる。

4. 1 システムの全体構成

MusicCast/AS は、図 1 に示した一般的放送システムと同じく、放送用サーバと視聴用クライアントからなるシステムである。現在サーバ、クライアントとも Linux が動作した IBMPC 互換機を用いている。ネットワークには 100BASE-TX で構成された LAN 環境を使用した。

MusicCast/AS では、放送するコンテンツとして MPEG Audio Layer 3(MP3)形式でエンコードされた音楽データを対象としている。サーバは MP3 形式のファイルを分割して暗号化し、マルチキャストを用いてネットワークに送出する。クライアントでは、暗号化・断片化されたコンテンツを復号、結合してリアルタイムで再生する。

4. 2 サーバの処理

サーバは、以下の 2 種類の処理を行う。

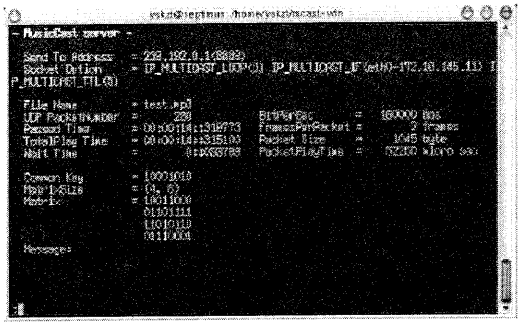

図 4 サーバの実行画面

Fig 4: Screen image of the server program

- セッション鍵の配布
- コンテンツの分割・暗号化・送信

これらの処理の流れを図3に、またサーバが動作している様子を図4に示す。

サーバにはあらかじめ、認証用鍵の行列を保持している。新たなセッションを始める際には、乱数でセッション鍵を生成し、認証用鍵を使用して TT-OOLS 法で暗号化し、セッション鍵を含むパケットとしてフラグをたてて、マルチキャストで配布する。

次にコンテンツを配布する。コンテンツの元となる MP3 ファイルは、MP3 フレームと呼ばれる基本単位に分割される。1 つの MP3 フレームの大きさは通常約 400 バイトである。MusicCast/AS ではこの MP3 フレーム 2 個ごとにまとめてパケットを作成し、セッション鍵によって暗号化した後、シーケンス番号をつけて順に送出する。この送出の際、MP3 ファイルをあらかじめ解釈しておき、MP3 ファイル内に指定された転送レートで送出するようにパケットの送出間隔を調整している。なお、セッションを開始してコンテンツの送信が始まった後でも、100 パケットに 1 つの割合でセッション鍵を再度配布し、途中から参加してくるクライアントに対応できるようにしている。

サーバはこれらの 2 種類の処理の結果を、独自のヘッダをつけたパケットとしてマルチキャストで送出する。ヘッダには、パケットの種類(セッション鍵を含むかコンテンツを含むか)を示すフラグとシーケンス番号、およびデータが格納された単純な構造となっている。

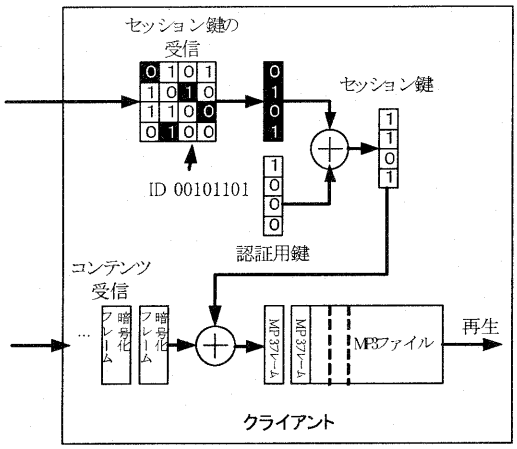

図 5 クライアントの処理

Fig 5: Process in the client program

図 6 クライアントの実行画面

Fig 6: Screen image of the client program

図 7 MP3 プレーヤによる再生

Fig 7: Playback via a MP3 player

また、現状ではセッション鍵の暗号化およびコンテンツの暗号化は、単純な XOR によって行っている。

4. 3 クライアントの処理

クライアントで行われる処理は、以下の2種類である。

・セッション鍵の復号

・コンテンツの複合化・結合・再生

これらの処理の様子を図5に示す。また、実際のプログラムの実行画面を図6に示す。

これらの処理は、受信したパケットの種類によってそれぞれ起動される。

セッション鍵の復号化においては、以下の処理を行う。まず、受信した鍵の行列の中から、自らのユーザ ID に対応した要素を抜き出し、認証用鍵を使ってセッション鍵を得る。

コンテンツの受信処理では、受信したパケットのシーケンス番号に欠落がないのを確認した後、セッション鍵を使ってコンテンツデータを復号し、結合して MP3 ファイルを復元してゆく。もしパケットに欠落があった場合には、ダミーとして無音の MP3 フレームを挿入して補う。

実際の音楽の再生は、クライアントプログラム自身では行わず、一般に普及している HTTP 対応 MP3 プレーヤを利用する。クライアントプログラム自身は HTTP サーバとして動作し、localhost からの接続を受け付けて復元した MP3 ファイルをリアルタイムで送出する。よってユーザは、クライアントプログラムが動作しているマシン内で MP3 プレーヤを起動することによって、放送されている音楽を聴取することができる。その実行の様子を図7に示す。

なお、放送の途中でクライアントプログラムを起動した場合は、MP3 ファイルの途中からの再生となるが、多くの MP3 プレーヤは正常な MP3 フレームが連続している限りファイルの途中からでも再生できることがわかった。また、逆に認証用鍵の入力を間違うなどして正常なセッション鍵が復元できなかった場合は、MP3 ファイルも復元できず、プレーヤによっては再生できないか、雑音として再生される。

5. 評価と今後の課題

今回実装した MusicCast/AS は現在和歌山大学内で稼動状態にあり、研究室内という限られた環境であるが、マルチキャストを用いた認証付きの音楽放送が実現可能であることを示すことが出来た。よって、有料インターネット放送システムのプロトタイプとしては十分利用で

きるものと考えられる。

認証機構として TT-OOLS 法を実装したことにより、認証のためのトラフィックは、各ユーザに認証用鍵を配布する際のものがほとんどである。放送時にセッション鍵の配布に必要なパケット数は、現在の実装ではコンテンツデータのパケット数の 1%である。パケットのサイズもコンテンツデータのパケット(約 1kB)に比べてはるかに小さく、鍵の行列の大きさにもよるが数十バイト以内である。よって、事実上そのトラフィックは無視可能である。

現在の実装では、鍵の行列のサイズを可変としたため、特に列数を増やすことによってユーザの結託に対する耐性を向上させることができ、安全性も確保できる。

今後の課題としては、以下のようなものがあげられる。

①現在暗号化方式として TT-OOLS 法しか実装されていないが、今後ほかの Tracing Traitors 法や、放送型暗号などの方式の実装も行い、実用性を評価する必要がある。また、独自の暗号技法の開発も検討中である。

②現在は実際は課金サーバにあたるものを実装していないが、実用実験のためには認証用鍵のためのサーバの構築と運用も必要になると考えられる。

③現在マルチキャストでのデータ送信には独自の形式のパケットを用いているが、RTP[12]のように、ほかのストリーム放送で実際に使われている手法との互換性を確保することも考えられる。

謝辞

議論に参加くださった和歌山大学システム工学部國枝・上原研究室の諸氏に感謝いたします。

参考文献

[1] Thomas A.Maufer 著(楠本博之訳):IP マルチキャスト入門、共立出版(2000)

[2] 山岡 克式, 佐藤 敬, 上原 哲太郎, 結城 皖曠:マルチキャストを用いたスケーラブルな有料放送の実現、電子情報通信学会技術報告, NS2001-106, IN2001-70, CS2001-67, pp.77-82 (2001)

[3] W.Fennner: "Internet Group Management Protocol, Version 2", RFC2236 (1997)

[4] N. Ishikawa, N.Yamanouchi and O.Takahashi: "IGMP Extension for Authentication of IP Multicast", Internet Draft, 1998.

[5] H. Harney and C. Muckenhirn : "Group Key Management Protocol (GKMP) : Specification", RFC2093(1997)

[6] H. Harney and C. Muckenhirn : "Group Key Management Protocol (GKMP) : Architecture", RFC2094(1997)

[7] T. Ballardie : "Scalable Multicast Key Distribution", RFC1949(1996)

[8] D. Wallner, E. Harder and R. Agee : "Key Management for Multicast : Issues and Architectures", RFC2627(1999)

[9] D. Maughan, M. Schertler, M. Schneider and J.Turner, "Internet Security Association and Key Management Protocol (ISAKMP)", RFC2408 (1998)

[10] A. Fiat and M. Naor, "Broadcast Encryption", Proc. Advances in Cryptology–Crypto '93, pp.480–491 (1994)

[11] B. Chor, A. Fiat, M. Naor and B. Pinkas, "Tracing Traitors", IEEE Transactions on Information Theory , Vol. 46, pp. 893–910 (2000).

[12] Audio-Video Transport Working Group : "RTP: A Transport Protocol for Real-Time Applications", RFC1889 (1996)