漏洩者の特定と配信停止が可能なマルチキャスト配信方式

豊島 鑑[1]　井上 武[1]　上松 仁[1]　高橋 宏和[1]
仁科 五月[2]　高木 剛[2]　湊 真一[3]

NTT未来ねっと研究所[1]
公立はこだて未来大学 システム情報科学部[2]
北海道大学 大学院 情報科学研究科[3]

背景

高価なディジタルコンテンツ（例、ディジタルシネマ映像）のネットワーク配信が始まっている
（参考文献　[1]NTT技術ジャーナル）

現状：　各受信者（映画館など）向けに個別に暗号化後、
　　各受信者に向けてユニキャスト配信
今後：　配信先増加＆マルチキャスト配信一般化
　　↓
高価なディジタルコンテンツもマルチキャスト配信したい
但し、高いセキュリティも当然必要（暗号化＋α）

課題

高価なディジタルコンテンツをマルチキャスト配信する場合の課題

（1）現在行われているマルチキャスト配信：
一つのコンテンツをネットワーク内装置（ルータ等）でコピーするため、どの受信者も同一コンテンツを受信

↓

コンテンツ漏洩時に、漏洩者の特定が困難

（2）漏洩者を特定できたとしても、全受信者の復号鍵が同じなため、新しい復号鍵の再配布が必要

↓

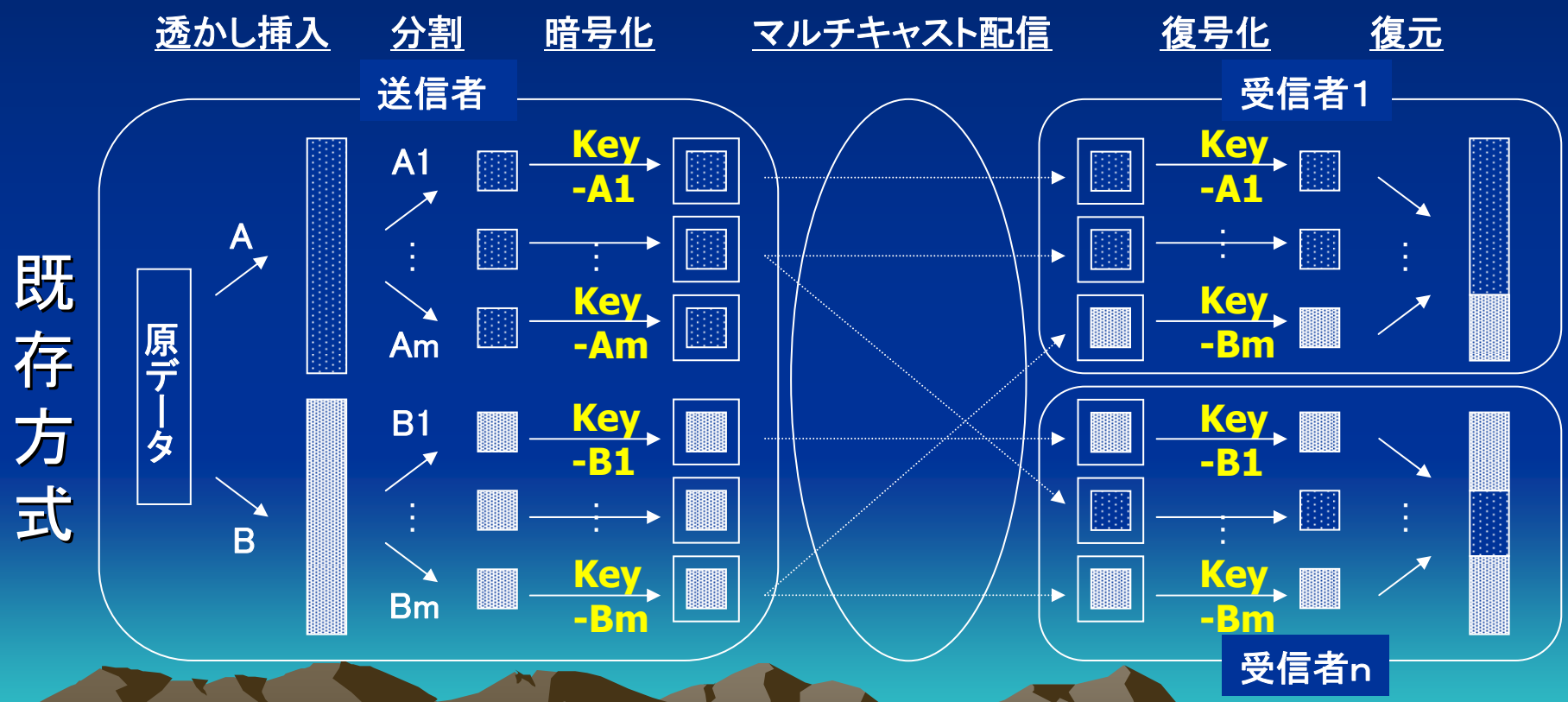
漏洩者のみへの配信停止を直ちに簡単に行えない

発表概要

- 本発表では、漏洩者の特定と、漏洩者のみへの配信停止を直ちに簡単に実施可能なマルチキャスト配信方式を提案
  - 漏洩者の特定方法
  - 漏洩者への配信停止方法
  - 提案方式
  - BGW方式
  - 各演算時間の評価
  - まとめ

漏洩者の特定方法(1)

- マルチキャスト配信で、受信情報漏洩者を特定する方法
  - Parnes 等が考案； 以下、“マルチキャスト透かし方式”と呼ぶ
    （参考文献: [3] IFIP CMS2001, 信学総大2007 B-7-4の図1）



漏洩者の特定方法(2)

マルチキャスト透かし方式の概要

（例、信学総大2007 B-7-4の図1の場合）

・配信元：

[事前]

- 各受信者の電子透かしパターンを決定（例、A,B,A,A,B・・・）
- 電子透かしパターンを受信者に通知
- 各分割コンテンツを復号する鍵の組を受信者に配布

[コンテンツ配信毎]

- オリジナルコンテンツをコピー
- 異なる2種類の電子透かしを入れたコンテンツを作成
- それぞれをm個に分割（例えば、m=100）
- 各分割コンテンツ（2m個）を個別の鍵で暗号化し配信

漏洩者の特定方法(3)

- 各受信者

[事前]

- 電子透かしパターンを受信
- 各電子透かし入り分割コンテンツを復号する鍵（m個）の組を受信

[コンテンツ配信毎]

- 指定された受信パターンに従って受信要求をマルチキャストネットワークに送信
- 分割コンテンツを受信
- 復号鍵で復号（指定された分割コンテンツのみが、配信元から渡された鍵で正しく復号される）
- コンテンツを復元（電子透かしパターン入り）

漏洩者の特定方法(4)

ネットワーク透かし方式の特徴

長所1： 受信者毎にユニークな電子透かしのパターンにより、マルチキャストでありながら、漏洩者を特定可能

長所2： 2種類の電子透かしを入れることによるネットワーク帯域の増加は、最大でも高々2倍（ユニキャストでは最大N倍）

短所 ： マルチキャスト配信なので、漏洩者への配信を直ちに止められない（全受信者への鍵の組の再配布が必要）

漏洩者への配信停止方法(1)

- マルチキャスト配信で、コンテンツ漏洩者への配信を停止する方法:

  → ブロードキャスト暗号を適用することを考案

漏洩者への配信停止方法(2)

- ブロードキャスト暗号
  - 1人の送信者と複数の受信者間で使用
  - 送信者は同じ暗号文を受信者全体に送信可
  - 正しく復号できるのは、暗号化時に指定された受信者グループのみ
  - 具体的には、公開鍵からメッセージ暗号化鍵を作成して暗号化
    → 許可されたグループの受信者のみが各々の秘密鍵を用いて復号化可能
  - この画期的な暗号の概念は1993年に提案された[4]が、実用可能なBGW方式は2005年

漏洩者への配信停止方法(3)

ブロードキャスト暗号(BGW方式)による配信の例

② 受信者全体に送信

① 送信者は、復号化を許可する受信者を指定(ここではA,B)してコンテンツを暗号化

③ 受信者A, Bはコンテンツを復号化可能

③' 受信者Cはコンテンツを復号化不可

漏洩者への配信停止方法(4)

・ブロードキャスト暗号を用いた配信停止方法

漏洩者が分かった時点で、送信側の暗号化鍵を作り替える(グループから漏洩者を除いた受信者の秘密鍵から暗号化鍵を作成して暗号化)

↓

漏洩者は、コンテンツを受信できても復号化できない

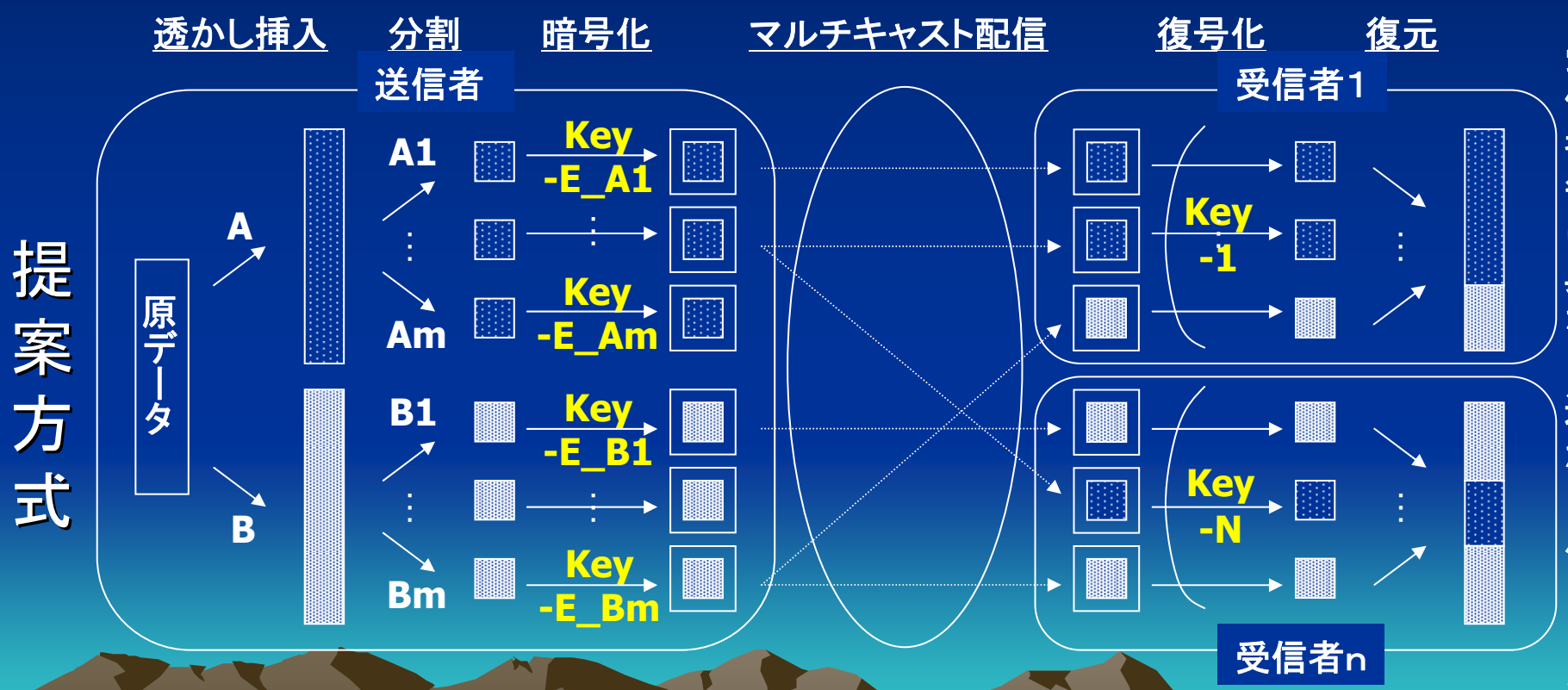
↓

即ち、漏洩者への配信を実質的に停止(直ちに簡単)

・但し、ブロードキャスト暗号で配信される情報は全て同じなため、漏洩者の特定は困難

提案方式(1)

前提　　：　マルチキャスト配信
提案方式：　マルチキャスト透かしとブロードキャスト暗号を組み合わせる



提案方式(2)

提案方式の長所

(1)漏洩者を特定でき、そこへの配信停止を直ちに簡単に可能

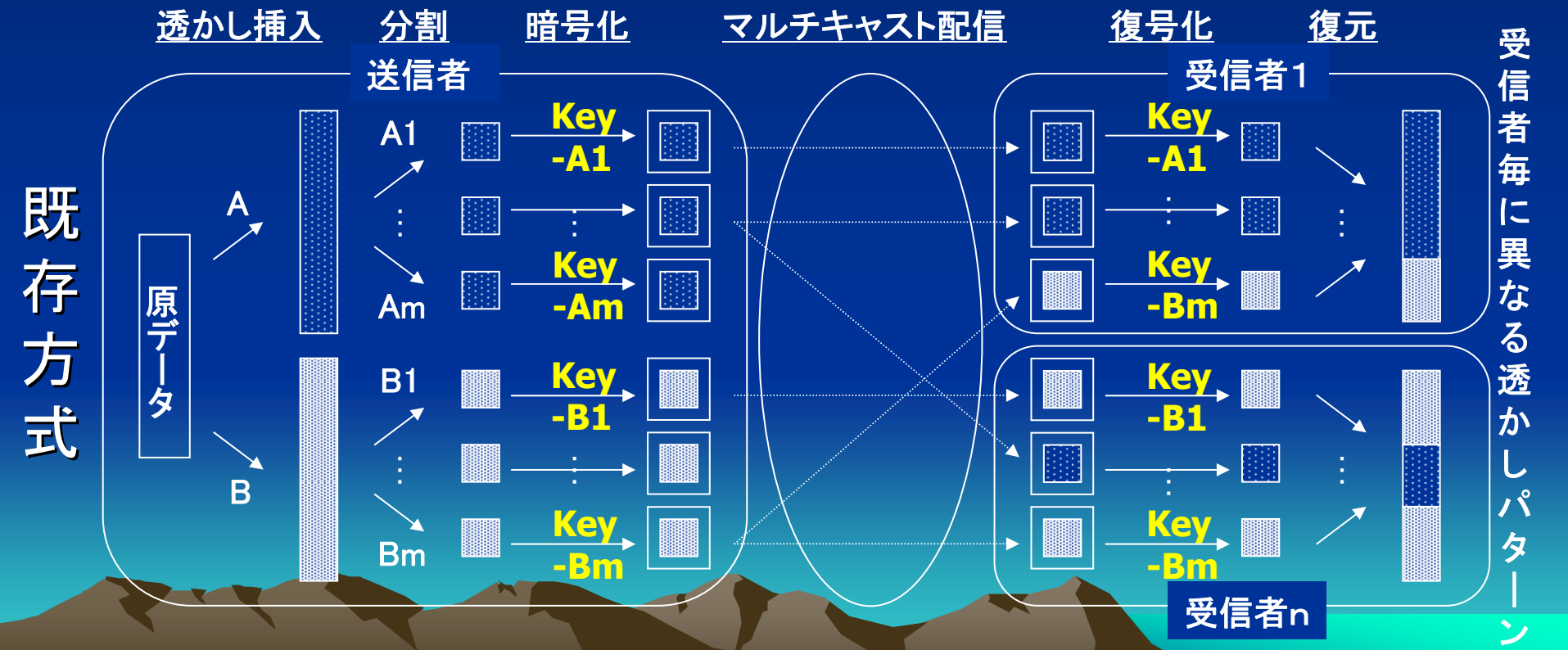
(2)漏洩後の各受信者の秘密鍵の再配布が不要

- 既存方式(＝秘密鍵暗号化＋マルチキャスト透かし)では、
  - ・分割コンテンツの透かしパターンに合わせた復号鍵m個の組を受信者毎に個別に作成、
  - ・漏洩後の各受信者の秘密鍵の再配布が必要

提案方式(3)

(3)各受信者の秘密鍵は1個(分割数mに非依存)

－ 既存方式では、電子透かしによる漏洩者特定能力を向上させるためにコンテンツの分割数mを増やすと、各受信者が復号に使用する鍵の数mも増加 （例えば、m＝100, 1000, ・・・）



BGW方式(1)

- 提案方式を実用可能か演算時間のチェックが必要
- ブロードキャスト暗号の具体的方式として、基本となるBGW方式を選択
- BGW方式：以下の3つのアルゴリズムで構成

  鍵生成 → 暗号化 → 復号化

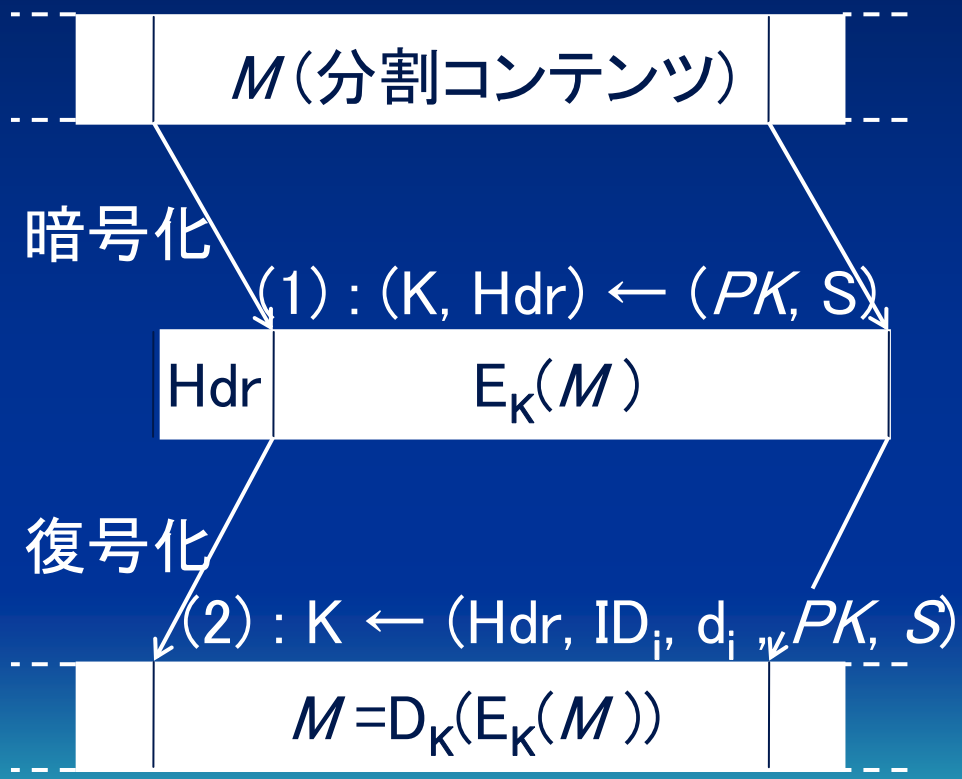
  - 各受信者にIDを割り振る(1～n までの自然数)
  - 各受信者は個人秘密鍵($d_1, d_2, \cdots, d_n$)を所有
  - 公開鍵を*PK*
  - 送信者が復号化を許可する受信者の集合をS
    $S \subset \{1, 2, \cdots, n\}$とする

BGW方式(2)

- 鍵生成
  - 入力： 受信者の総数n
  - 出力： 公開鍵*PK*と個人秘密鍵 d_i (i= 1, 2 , ･･･, n)
- 暗号化
  - 入力： メッセージ*K*、公開鍵*PK*,
    復号化を許可する受信者の集合*S*
  - 出力： 暗号文*C*とヘッダHdr
- 復号化
  - 入力： 暗号文*C*, ヘッダHdr, 受信者ID_i,
    個人秘密鍵 d_i, 公開鍵*PK*,
    復号化を許可された受信者の集合*S*
  - 出力： メッセージ*K*

ハイブリッド型BGW方式

ハイブリッド型
（コンテンツの暗号/復号化鍵
をブロードキャスト暗号化）



K：メッセージ暗号化鍵
HdrはKを暗号化したもの
E：共通鍵暗号の暗号化関数
D：共通鍵暗号の復号化関数
$D_K(E_K(M))=M$ を満たす

(1)：公開鍵*PK*と受信者集合*S*からメッセージ暗号化鍵Kとヘッダ Hdrを生成

(2)：ヘッダHdr，受信者IDi，秘密鍵di，公開鍵PK，受信者集合S からメッセージ暗号化鍵Kを生成

各演算時間の評価(1)

- 処理を実装してその演算時間を実測
- 共通鍵暗号による分割コンテンツの暗号化/復号化演算時間を省略
  - 従来方式と同等なため（BGW方式はハイブリッド暗号）

↓

即ち、分割コンテンツ1個に対する
①個人秘密鍵生成、②Hdr暗号化、③Hdr復号化
の各時間を測定

各演算時間の評価(2)

- 双線形写像として、高速動作が知られているη_Tペアリングを使用 （参考文献[9]）
- 標数3、拡大次数97の有限体$F_{3^{97}}$を使用
- Hdrヘッダ長：320bit、メッセージ暗号化鍵長：160bit
- 測定用PCのスペック

  CPU：Athlon X2 3800+　　メモリ：2GB

  OS：Windows Vista (32bit)
- C言語で実装

  コンパイラ：　gcc 3.4.4

  最適化オプション：　-O2 -fomit-frame-pointer

  多倍長ライブラリ：　GMP 4.2.2を使用

各演算時間の評価(3)

表1　BGW方式の各処理時間　[秒]

人数*	個人秘密鍵生成	Hdr暗号化	Hdr復号化
100	0.479	0.010	0.008
500	2.377	0.025	0.023
1000	4.783	0.044	0.042
1500	7.144	0.063	0.061
2000	9.522	0.082	0.080

＊鍵生成:　全受信者数n

暗号化と復号化:　復号化を許可した受信者数s

- 個人秘密鍵生成の演算時間:　全受信者数nに比例
- 暗号/復号化演算時間:復号許可受信者数sにほぼ比例

各演算時間の評価(4)

・結果

- 配信の最初に必要な個人秘密鍵生成時間は十分短い
- Hdr暗号化に必要な時間も十分短い

即ち、漏洩者への配信を停止するためのメッセージ暗号化鍵Kの作り替えに必要な時間も十分短い

- Hdr復号化に必要な時間も十分短い

まとめ

（1）マルチキャスト配信において、コンテンツの漏洩者を特定し、漏洩者への配信を簡単に速やかに停止できる方式を提案した。

（2）ブロードキャスト暗号のベースとなるBGW方式の処理を実装してその演算時間を実測し、実用可能な見通しを得た。

ご清聴ありがとうございました。

補足資料

ブロードキャスト暗号の詳細(1)

・受信者各個人の公開鍵により暗号化を行うことでブロードキャスト暗号を実現した場合（最も単純な方法）、
- 受信者全体の公開鍵のサイズ：O(n)
- 受信者個人の秘密鍵のサイズ：O(1)
- 暗号文のサイズ　　：O(s)

但し、 n：全受信者数、
s：復号化を許可する受信者数、
r：復号化を許可しない受信者数
即ち、s + r = n

・公開鍵サイズと暗号文サイズを小さくすることが、効率的なブロードキャスト暗号を実現する上で重要

ブロードキャスト暗号の詳細(2)

- 2005年　Boneh, Gentry, Waters
  - 暗号文サイズが O(1) のブロードキャスト暗号方式を提案 （参考文献[5]．以下、BGW方式と呼ぶ）
  - 公開鍵サイズ：O(n) これまでと同じ

- 2007年　Delerablee, Paillier, Pointcheval
  - 受信者の動的な追加を可能にした Dynamic Broadcast Encryption を提案 （参考文献[6]．以下、DBE方式と呼ぶ）
  - 公開鍵サイズ：O(n)　これまでと同じ
    暗号文サイズ：O(r)

ブロードキャスト暗号の詳細(3)

- 2007年　Sakai等とDelerablee等は独立に、
  公開鍵を更新することなく動的に受信者を追加できる
  Identity-Based Broadcast Encryption を提案
  （参考文献[7][8]．以下、IBBE方式と呼ぶ）
  - 但し、復号化を許可できる受信者数kに上限あり
  - 公開鍵サイズ：O(k)
    暗号文サイズ：O(1)

- 以上の3方式の中で、基本となるBGW方式について、実現性を評価するために、ソフトウェア実装して演算時間を測定