ハイブリッド暗号について

黒澤　馨（茨城大学）
kurosawa@mx.ibaraki.ac.jp

ハイブリッド暗号とは (1)

- **公開鍵暗号**
  鍵配送が不要。しかし、処理が遅い。
- **共通鍵暗号**
  処理が高速。しかし、鍵配送が必要。
- **ハイブリッド暗号**
  両者の欠点を補い合う方式

ハイブリッド暗号とは (2)

- 暗号文＝ 公開鍵暗号(K)
  ＋ 共通鍵暗号(K, 長い平文M)
- 公開鍵を利用するので、公開鍵暗号の1つ。
- メール用暗号で有名なPGP、
  Webサイトで使われるSSL

講演の具体的内容

- 公開鍵暗号の安全性
- ランダム･オラクル･モデルにおいて安全な
  ハイブリッド暗号方式
- KEM-DEM フレームワーク
- 標準モデルにおいて安全なKEM
- Kurosawa-Desmedtのハイブリッド暗号

講演の具体的内容

- **公開鍵暗号の安全性とは**
- ランダム･オラクル･モデルにおいて安全な
  ハイブリッド暗号方式
- KEM-DEM フレームワーク
- 標準モデルにおいて安全なKEM
- Kurosawa-Desmedtのハイブリッド暗号
- 安全性証明

RSA暗号

- 公開鍵: N(=pq), e
  ただし、p,qは大きな素数。
- 秘密鍵: d ただし、ed=1 mod (p-1)(q-1)
- 平文: M
- 暗号文: $C = M^e \bmod N$
- 復号: $M = C^d \bmod N$

受動的な敵に対する安全性

- **鍵完全解読**
  公開鍵 (N,e) → 敵 → 秘密鍵 d
- **一方向性**
  公開鍵、暗号文 → 敵 → 平文全体 M
  (N, e, C)
- Semantic Security
  公開鍵、暗号文 → 敵 → 平文Mの何らかの部分情報

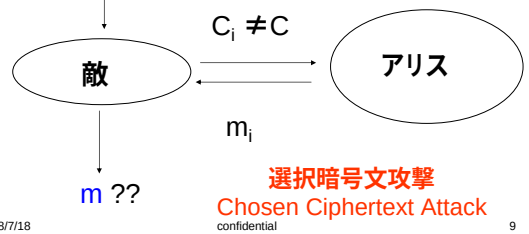
ＲＳＡ暗号の安全性（1）

敵	鍵完全解読 (N,e) → d	一方向性 (N,e,C) → M全体	Semantic Security (N,e,C) → Mの部情報
受動的	素因数分解	ＲＳＡ仮定	ある部分情報がもれる（＊）

（＊）現代暗号の基礎数理、150ページ
黒澤・尾形著 （電子情報通信学会編）

能動的な敵

公開鍵、及び
暗号文 C=E(m)



選択暗号文攻撃
Chosen Ciphertext Attack

ＲＳＡ暗号の安全性(2)

敵	鍵完全解読	一方向性	Semantic Security
受動的	素因数分解	ＲＳＡ仮定	ある部分情報がもれる
能動的（ＣＣＡ）		完全に破れる	完全に破れる

公開鍵暗号の安全性

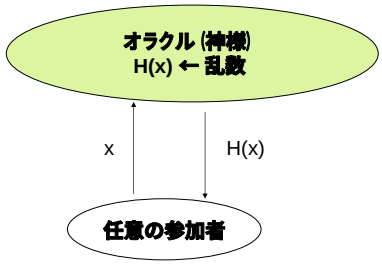
敵	鍵完全解読	一方向性	Semantic Security
受動的	最も弱い暗号		
能動的 (CCA)			最も安全な暗号（ＣＣＡ安全）

本講演の内容

- 公開鍵暗号の安全性とは
- ランダム・オラクル・モデルにおいて安全なハイブリッド暗号方式
- KEM-DEM フレームワーク
- 標準モデルにおいて安全なKEM
- Kurosawa-Desmedtのハイブリッド暗号
- 安全性証明

ランダム・オラクル・モデルとは

- ハッシュ関数 H を極度に理想化したモデル



RSA暗号に基づく単純なハイブリッド暗号

RSA暗号により、共通鍵暗号の鍵Kを暗号化。

K ← ランダム

暗号文

$$\begin{cases} \varphi = K^e \bmod N \\ \chi = \text{共通鍵暗号}(K, \text{平文M}) \end{cases}$$

これは、CCA安全とはいえない。

CCA安全にするには

共通鍵 K の生成をちょっと工夫

r ← ランダム

K＝H (r)

暗号文

$$\begin{cases} \varphi = r^e \bmod N \\ \chi = \text{共通鍵暗号}(K, \text{平文M}) \end{cases}$$

共通鍵暗号の部分

- K → 擬似乱数生成器 → K_1, K_2
- 暗号文 χ

$$\begin{cases} E = M \oplus K_1 \\ Tag = One - Time - MAC_{K_2}(E) \end{cases}$$

CCA安全なハイブリッド暗号

RSA仮定の下で、CCA安全 in the RO model

r ← ランダム

K＝H (r)

暗号文

$$\begin{cases} \varphi = r^e \bmod N \\ \chi = (E, Tag) \end{cases}$$

CCA安全なハイブリッド暗号

RSA仮定の下で、CCA安全 in the RO model

r ← ランダム

K＝H (r)

暗号文

$$\begin{cases} \varphi = r^e \bmod N \\ \chi = (E, Tag) \end{cases}$$

RSA-KEM

ハイブリッド暗号の効能

- CCA安全な方式を簡単に作れる。
- RSA-KEMは、ISO18033-2のドラフト。

- RSA暗号に基づく他のハイブリッド暗号：
  RSA-OAEP, OAEP+等
- RSA-KEMの方が簡単で、
  安全性の帰着効率も、よりbetter。

共通鍵暗号の部分

- K → 擬似乱数生成器 → K_1, K_2

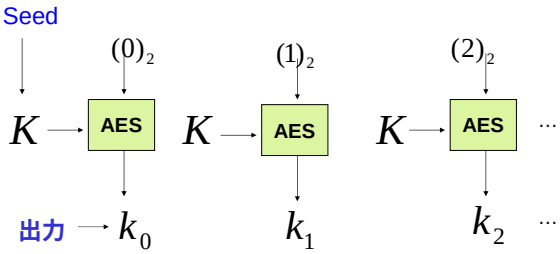
- 暗号文 χ

$$\begin{cases} E = M \oplus K_1 \\ Tag = One - Time - MAC_{K_2}(E) \end{cases}$$

擬似乱数生成器

- ランダム･オラクル･モデルにおいては、
  ランダム･オラクルと考えればよい。
- 理論的には、
  一方向性関数から構成可能。
- 実用的には、
  AES暗号を利用して、以下のように構成。

AES暗号を利用した擬似乱数生成器



AES暗号が擬似ランダム置換と仮定すると、
これは、標準モデルで擬似乱数生成器

共通鍵暗号の部分

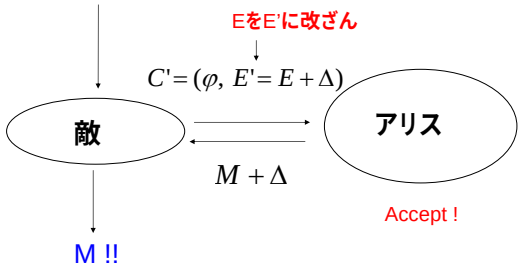
- K → 擬似乱数生成器 → K_1, K_2

- 暗号文 χ

$$\begin{cases} E = M \oplus K_1 \\ Tag = One - Time - MAC_{K_2}(E) \end{cases}$$

Tagが無いと

$C = (\varphi = r^e,\ E = M + K_1, \sout{Tag})$



Tagの作り方

- **理論的には、**One-Time-Macでよい。
  情報理論的に安全な方式が存在。

$$E=(e_1,e_2,\cdots,e_t),\ 鍵 K=(a,b)$$

$$Tag=a+be_1+b^2e_2+\cdots+b^te_t \quad \text{over GF(2^n)}$$

- **実用的には、**
  計算量的に安全なMACでもよい。
  例：CMAC (米国標準：岩田、黒澤)

Elgamal暗号

- **公開鍵**：$p,g,y(=g^x \bmod p)$
- **秘密鍵**：x
- **平文**：　M
- **暗号化**：$r \leftarrow$ランダム

$$C=(g^r,\ M\times y^r)$$

離散対数問題

- Gを位数が素数qの乗法群とする。
  このとき、g, y ∈Gから、

$$y = g^x$$

  となるxを求めよ、という問題。

DH問題とDDH問題

- DH問題

(g,g^x,g^y)からg^{xy}を求めよ。

- DDH問題

(g,g^x,g^y,g^{xy})と$(g,g^x,g^y,$乱数$)$

を区別せよ。

ElGamal暗号の安全性（1）

	鍵完全解読	一方向性	Semantic Security
受動的な敵	離散対数問題	DH問題	DDH問題と等価（＊）

（＊）現代暗号の基礎数理、157ページ
黒澤・尾形著　（電子情報通信学会編）

ElGamal暗号の安全性(2)

	鍵完全解読	一方向性	Semantic Security
受動的な敵	離散対数問題	DH問題	DDH問題
能動的な敵（CCA）		完全に破れる	完全に破れる

ElGamal暗号に基づくハイブリッド暗号

$r \leftarrow$乱数,

$K = H(g^r, y^r)$

暗号文

$\varphi = g^r$

χ = (E, Tag)

この方式も、ISOのドラフトに含まれている。

ElGamal暗号に基づくハイブリッド暗号

Gap-DH仮定の下で、CCA安全 in the RO model

$r \leftarrow$乱数,

$K = H(g^r, y^r)$ } KEM

暗号文

$\varphi = g^r$

χ = (E, Tag)

この方式も、ISOのドラフトに含まれている。

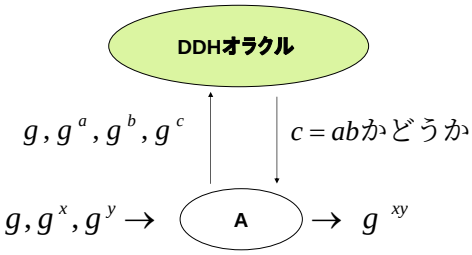
DH問題

- **以下の問題を解け。**

$g, g^x, g^y \rightarrow$ A $\rightarrow g^{xy}$

Gap-DH問題

- DDH問題が解けると仮定して、DH問題を解け。



本講演の内容

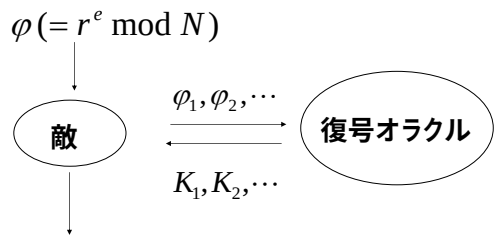
- **公開鍵暗号の安全性とは**
- **ランダム・オラクル・モデルにおいて安全なハイブリッド暗号方式**
- KEM-DEM **フレームワーク**
- **標準モデルにおいて安全な**KEM
- Kurosawa-Desmedt**のハイブリッド暗号**
- **安全性証明**

KEM-DEMフレームワーク

- **ハイブリッド暗号の一般的構成法**
- Shoup**が定式化し、**ISO18033-2**にて記述。**

- KEM: Key Encapsulation Mechanism
  公開鍵暗号を利用して、φ, K を生成する部分
- DEM: Data Encapsulation Mechanism
  K鍵として共通鍵の暗号文 χ を生成する部分

KEMのＣＣＡ安全性



K =H(r) ??

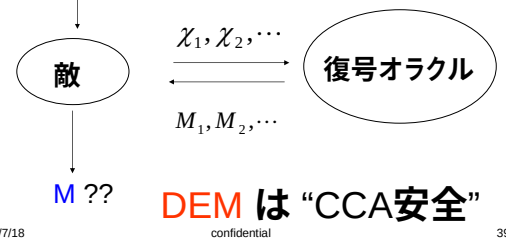
KEM は “CCA安全”

CCA安全なＫＥＭの例

- RSA-KEM は、ＣＣＡ安全なＫＥＭ
  under RSA 仮定 in the RO model.
- 前述のElGamal暗号に基づくKEMは、CCA安全
  under gap-DDH 仮定 in the RO model.

DEMのＣＣＡ安全性

共通鍵暗号の暗号文 χ



M ??

DEM は “CCA安全”

CCA安全なＤＥＭの例

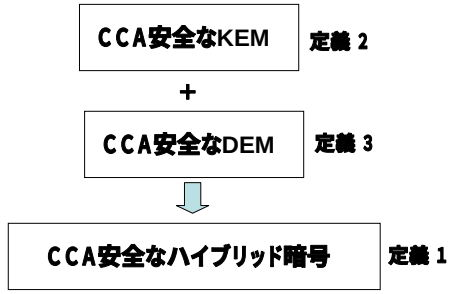
- K → 擬似乱数生成器 → K_1, K_2
- 暗号文 χ

$$\begin{cases} E = M \oplus K_1 \\ Tag = One-Time-MAC_{K_2}(E) \end{cases}$$

ＣＣＡ安全の定義は3つ

	ハイブリッド暗号（公開鍵暗号）	ＫＥＭ	ＤＥＭ
ＣＣＡ安全	定義1	定義2	定義3

KEM-DEMフレームワーク



ROモデルにおける例

- RSA-KEM (CCA安全なKEM)
  ＋ 前述のDEM (CCA安全なDEM)
  ＝RSA仮定の下でCCA安全なハイブリッド暗号
- Elgamalに基づくKEM ＋前述のDEM
  ＝gap-DH仮定の下でCCA安全なハイブリッド暗号

本講演の内容

- 公開鍵暗号の安全性とは
- ランダム・オラクル・モデルにおいて安全なハイブリッド暗号方式
- KEM-DEM フレームワーク
- **標準モデルにおいて安全なKEM**
- Kurosawa-Desmedtのハイブリッド暗号
- 安全性証明

ROモデルと標準モデル

- **ランダム・オラクル・モデル**
  ハッシュ関数Hを極度に理想化したモデル
  RSA-KEM, RSA-OAEP, OAEP+ , SAEP, SAEP+, ECIES, PSEC, HIME(R)
- **標準モデル**
  Hを具体的はハッシュ関数に固定するモデル

ROモデルにおける安全性？

- 以下のような暗号方式がいくつか知られている。
  Hをランダム・オラクルと仮定すると安全。
  しかし、Hを固定した途端に破れる。
- 特に、
  そのようなハイブリッド暗号方式が知られている。

⇓

- ROモデルにおける安全性証明は、
  標準モデルにおける安全証明にはならない。

標準モデルにおいてCCA安全なKEM

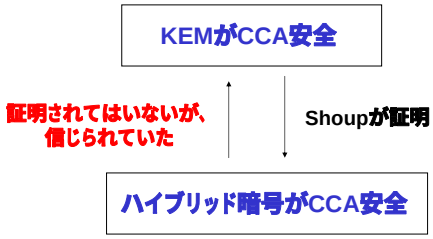
- Cramer-Shoup公開鍵暗号方式に基づくKEM。
- ISO18033-2の提案中で、唯一、
  標準モデルにおいてCCA安全な方式。

- Cramer-ShoupのKEM (CCA安全なKEM)
  ＋ 前述のDEM (CCA安全なDEM)
  ＝DDH仮定の下でCCA安全なハイブリッド暗号
  (KEM-DEMフレームワークより)

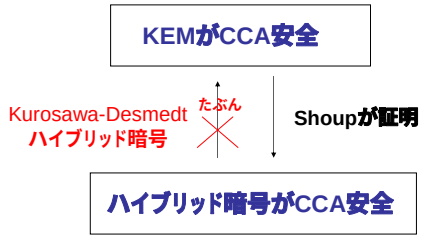
本講演の内容

- 公開鍵暗号の安全性とは
- ランダム・オラクル・モデルにおいて安全なハイブリッド暗号方式
- KEM-DEM フレームワーク
- 標準モデルにおいて安全なKEM
- Kurosawa-Desmedtのハイブリッド暗号
- 安全性証明

従来のパラダイム



Kurosawa-Desmedtの ハイブリッド暗号 (Crypto'04)

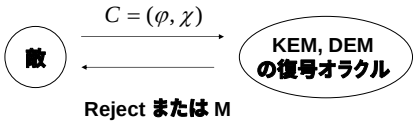


KD ハイブリッド暗号の意義

- 本方式 (Crypto'04、採択率33/212 =0.16)
  (理論的側面)
  従来のパラダイムには従わない方式
  KEMはCCA安全とはいえない。
  しかし、ハイブリッド暗号全体はCCA安全。
  (実用的側面)
  Cramer-Shoupに比べ、暗号文サイズ等、効率の面で優れている。

CCAアタックに対する耐性

- (Cramer-Shoup)
  KEMの復号: 不正な φ をreject.
  DEMの復号: 不正な χ をreject.
- (本方式)
  KEMの復号: reject機能が無い
  DEMの復号: まとめてreject.



Cramer-Shoup ハイブリッド暗号との比較

- 暗号文: 1 group element shorter
- 公開鍵: 1 group element shorter
- 秘密鍵: 2×|q|-bit shorter
- 暗号化 / 復号化における計算量
  べき乗計算が一回少ない。

標準モデルにおける本方式の安全性

- CCA安全なKEMとはいえない。
- しかし、
  本KEM + 前述のDEM
  =DDH仮定の下でCCA安全なハイブリッド暗号
  を証明できる。

Open problem

- CCA安全にかわるKEMの安全性をどう定義？

- 阿部正幸、R.Gennaro、黒澤:
  1つの解答をTag-KEMとして定式化
- 国際会議Eurocrypt 2005にて発表。
  採択率: 33/190=0.17。

本講演の内容

- 公開鍵暗号の安全性とは
- ランダム・オラクル・モデルにおいて安全なハイブリッド暗号方式
- KEM-DEM フレームワーク
- 標準モデルにおいて安全なKEM
- Kurosawa-Desmedtのハイブリッド暗号
- 安全性証明

KD ハイブリッド暗号

- Public-key

$$g_1,\ g_2,\ c = g_1^{x_1} g_2^{x_2},\ d = g_1^{y_1} g_2^{y_2}$$

- Private-key

$$x_1, x_2, y_1, y_2$$

Encryption

- r ← random
  $u_1 = g_1^r$, $u_2 = g_2^r$,
  χ = CCA安全なDEM(K, m)
- where
  $v = c^r \times d^{r\alpha}$ with α=TCR(u_1, u_2)
  K = H(v)
- The ciphertext is (u_1, u_2, χ)

秘密鍵の自由度

- A private-key
  (x_1, x_2, y_1, y_2)
  is randomly chosen in such a way that
- the public-key is g_1, g_2 and

$$c = g_1^{x_1} g_2^{x_2},\ d = g_1^{y_1} g_2^{y_2}$$

- The freedom is 4 – 2 = 2
- We consider the above probability space

Decryption

- For (u_1, u_2) , compute

α = TCR(u_1, u_2)

$$v = u_1^{x_1 + y_1\alpha} u_2^{x_2 + y_2\alpha}$$

(秘密鍵 x_1, x_2, y_1, y_2の自由度は2)
K=H(v)
KでDEMの暗号文χを復号。

(In)Valid KEM

- We say that
  $(u_1, u_2) = (g_1^r, g_2^r)$ is valid
  and
  $(u_1, u_2) = (g_1^r, g_2^s)$ is invalid

復号で得られる v

- If (u_1, u_2) is valid,
  v is decoded uniquely.
- If (u_1, u_2) is invalid,
  v is random.
- If (u_1, u_2) and (u_1', u_2') are both invalid,
  v and v' are independently random.

（**上記の性質**：strongly universal_2）

If (u_1, u_2) and (u_1', u_2') are both invalid,

- **公開鍵**
  Log c=(x_1, x_2, y_1, y_2)**の一次式**
  Log d= 〃 **の** 〃
- **復号で得られる** v
  Log v = 〃 **の** 〃
  Log v' = 〃 **の** 〃
- **これら4つの式は一次独立**

If (u_1, u_2) and (u_1', u_2') are both invalid,

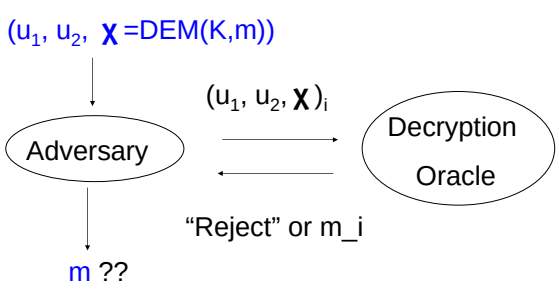
- **公開鍵**
  固定 { Log c=(x_1, x_2, y_1, y_2)**の一次式**; Log d= 〃 **の** 〃 }
- **復号で得られる** v
  任意 { Log v = 〃 **の** 〃; Log v' = 〃 **の** 〃 }
- **これら4つの式は一次独立**
- **任意の**v, v'**に対し、解**(x_1, x_2, y_1, y_2)**が存在。**
  ⇨ v, v'**は独立ランダム**

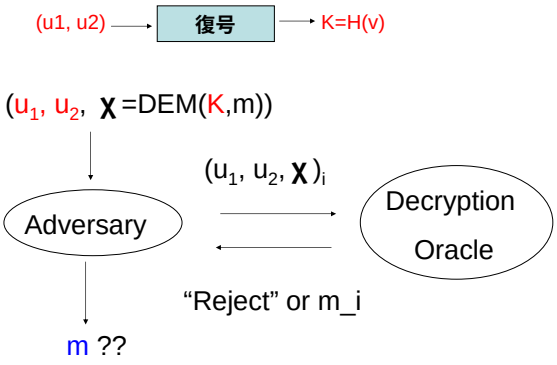
復号で得られる K=H(v)

- If (u_1, u_2) is valid,
  K=H(v) is decoded uniquely.
- If (u_1, u_2) is invalid,
  K=H(v) is random.
- If (u_1, u_2) and (u_1', u_2') are both invalid,
  K=H(v) and K'=H(v') are **独立**random.

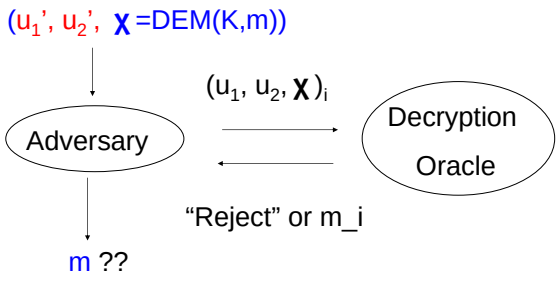
（**これらの性質**：strongly universal_2）

Chosen Ciphertext Attack

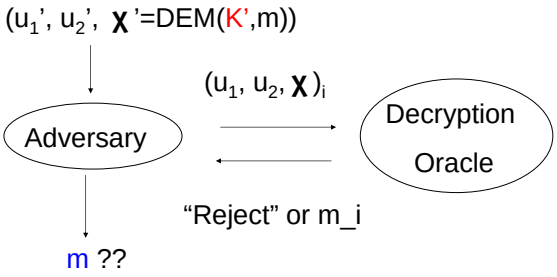




Invalidな(u1', u2')に置き換え（DDH仮定より）



ランダムなK'に置き換え（Strongly Universal_2より）



復号 query $(u_1, u_2, \chi)_i$

- (Type 1) Valid
- (Type 2) Invalid and $(u_1, u_2)_i = (u_1', u_2')$
- (Type 3) Invalid and $(u_1, u_2)_i \neq (u_1', u_2')$

For Type 3 query,

- $K_i = H(v_i)$ is random independently of K' from strongly universal_2
- Hence “rejected” by one-time MAC

(x_1,x_2,y_1,y_2)に関する情報は、何も増えない。

Type 2 query

- $(u_1, u_2)_i = (u_1', u_2')$
- In this case, $K_i = H(v_i) = K'$
- Hence χ_i is decrypted by the same K' that is used in the challenge ciphertext E'

Type 1 query

- $(u_1, u_2)\_i = (g_1^r, g_2^r)$: valid
- Normal **に復号**

⇩

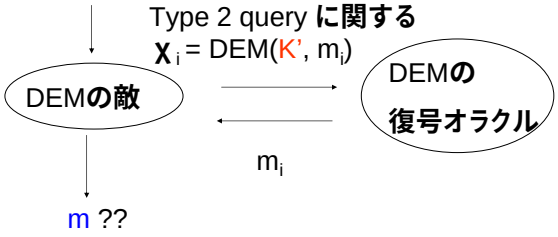
(x_1,x_2,y_1,y_2)**に関する一次独立な式は増えない。**

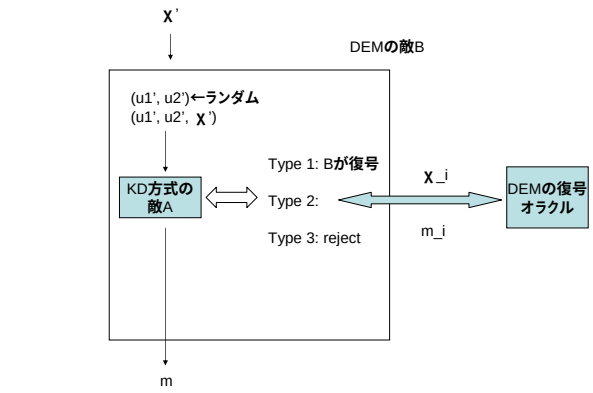
To summarize,

- Type 3 query is rejected
- Type 2 query is decrypted by K'
- Type 1 (valid) query is decrypted in the normal way
- Consequently, CCA-attack is reduced to a CCA-attack on **共通鍵**(DEM) as follows

CCA attack on DEM(**共通鍵**)

χ' = DEM(K', m)





DEM(**共通鍵**)**の敵**B

- KD**ハイブリッド暗号の敵**Aを起動する。
- **ただし、**KD**方式の秘密鍵は**Bが選ぶ。
  ⇨**復号オラクルをシミュレートできる。**
- g_2=g_1^w **となる**w**を自分で選ぶ。**
  ⇨ Type 1, 2 3**の区別をできる。**
- Invalid **な**(u_1', u_2')**をランダムに生成し、**
  (u_1', u_2', χ')**をA**へのチャレンジ暗号文とする
  ただし、χ'**は**Bに与えられた暗号文。

Finally,

- Our DEM is CCA-secure

⇩

- Our hybrid encryption scheme is CCA-secure under DDH **仮定**

Q.E.D.

Encryption

- r ← random
  $u_1 = g_1^r$, $u_2 = g_2^r$, χ = **共通鍵暗号**(K, m)
- where
  $v = c^r \times d^{r\alpha}$ with α=TCR(u_1, u_2)
  K = H(v)
- The ciphertext is (u_1, u_2, χ)

TCR **ハッシュ関数族** {H}

ランダムな x
ランダムな H
→ **敵** → y s.t. H(x)=H(y)

Pr(**敵が成功**) < ε

UOWH**関数の特別な場合**

- **理論的**: **任意の一方向性関数から構成可能**
- **実用的**: SHA-1

Encryption

- r ← random
  $u_1 = g_1^r$, $u_2 = g_2^r$, χ = **共通鍵暗号**(K, m)
- where
  $v = c^r \times d^{r\alpha}$ with α=TCR(u_1, u_2)
  K = H(v)
- The ciphertext is (u_1, u_2, χ)

$K = H(v)$ **のHついて**

- v**が一様分布するとき、**K**も一様分布すると仮定**
- **そのようなH**は、Left-over hash lemma **により、容易に構成可能** (**情報理論的**)

- Gennaro and Shoup**による改良**:
  H(v)**は** (**計算量的**) **擬似ランダムならよい。**

認証子Tag**について**

- **情報理論的**なone-time-MAC**を仮定。**

- Gennaro and Shoup**による改良**:
  計算量的に安全なMAC**を使用してよい。**

Projective Hash Family **へ一般化**

- DDH**仮定のみならず、**
- **平方剰余仮定、**
- N**次剰余仮定、**

それぞれの仮定下で、
同様なハイブリッド暗号を構成可能。

まとめ

- 公開鍵暗号の安全性とは
- ランダム･オラクル･モデルにおいて安全なハイブリッド暗号方式
- KEM-DEM フレームワーク
- 標準モデルにおいて安全なKEM
- Kurosawa-Desmedtのハイブリッド暗号
- 安全性証明