同期集約署名と計算仮定

東京工業大学大学院　情報理工学院　数理・計算科学系

手塚 真徹　　田中 圭介

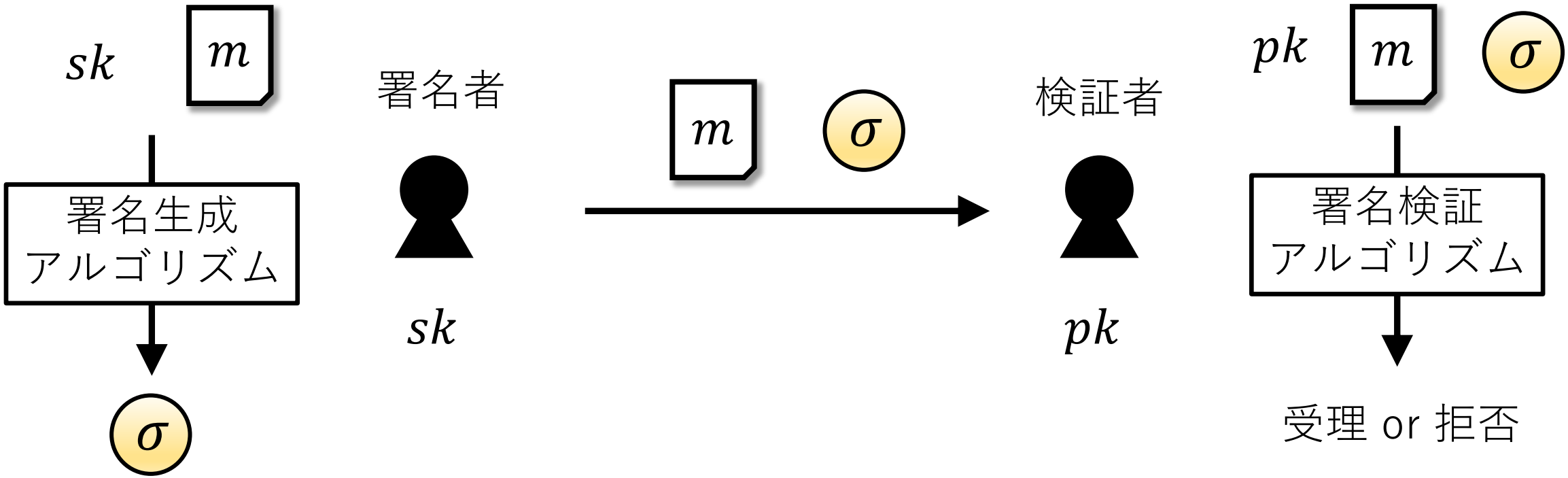
2022年3月14日
CREST 暗号数理最終ワークショップ

電子署名と同期集約署名の背景

電子署名

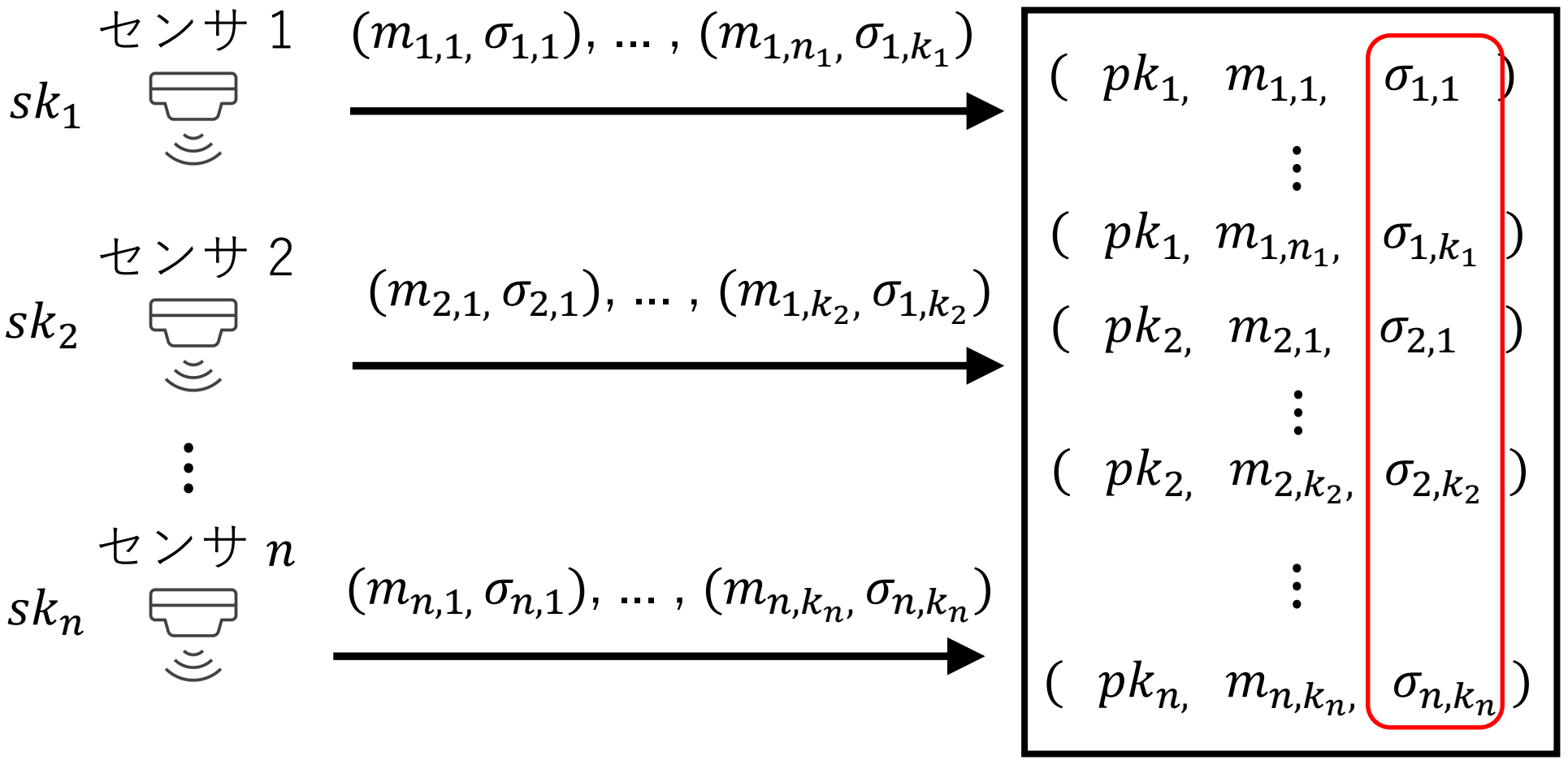
電子署名

メッセージが作成者（署名者）本人により作成され，メッセージが改竄されていないことを保証する技術.



IoTシステムでの電子署名

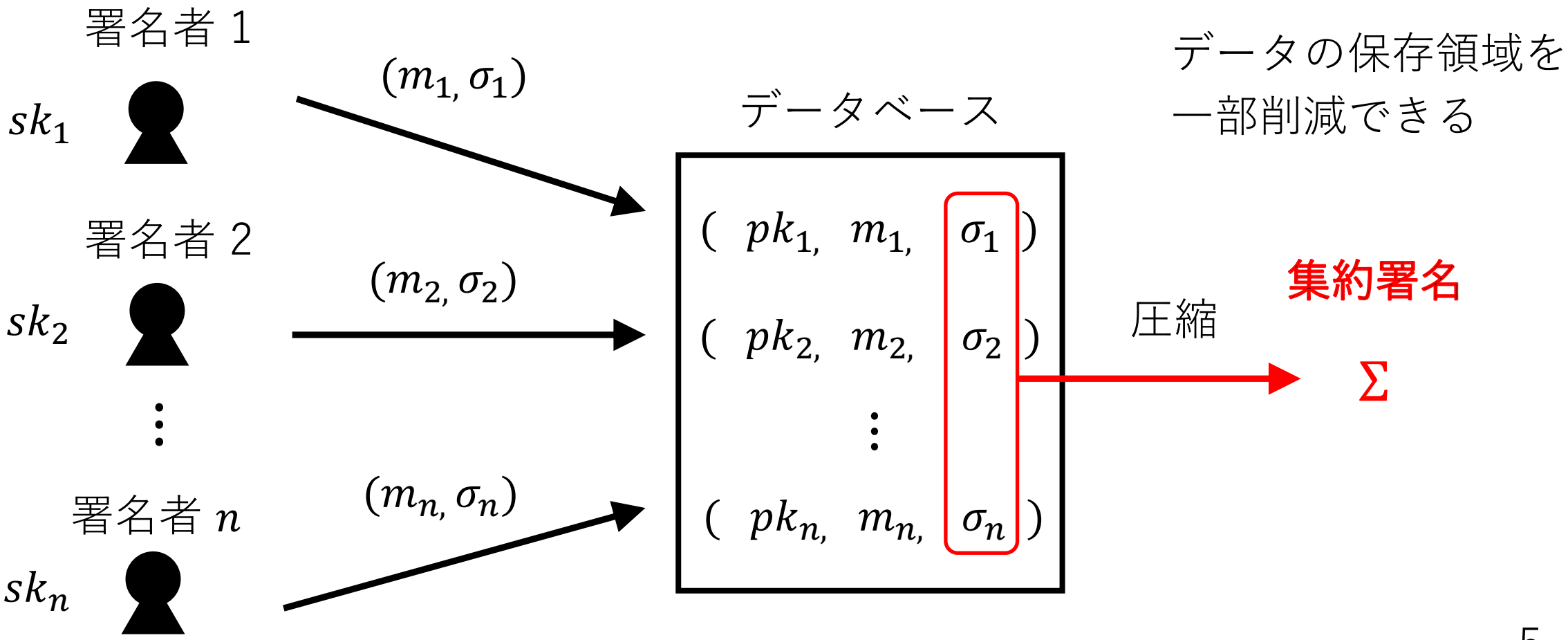
膨大なデータの
保存領域が必要



集約署名[BGLS03]

集約署名

複数個の署名を一つの集約署名へ圧縮できる



集約署名構成の難しさ

集約署名の構成

- 識別不可能難読化を用いた構成 [HKW15]
- 多重線形写像を用いた構成 [HSW13]

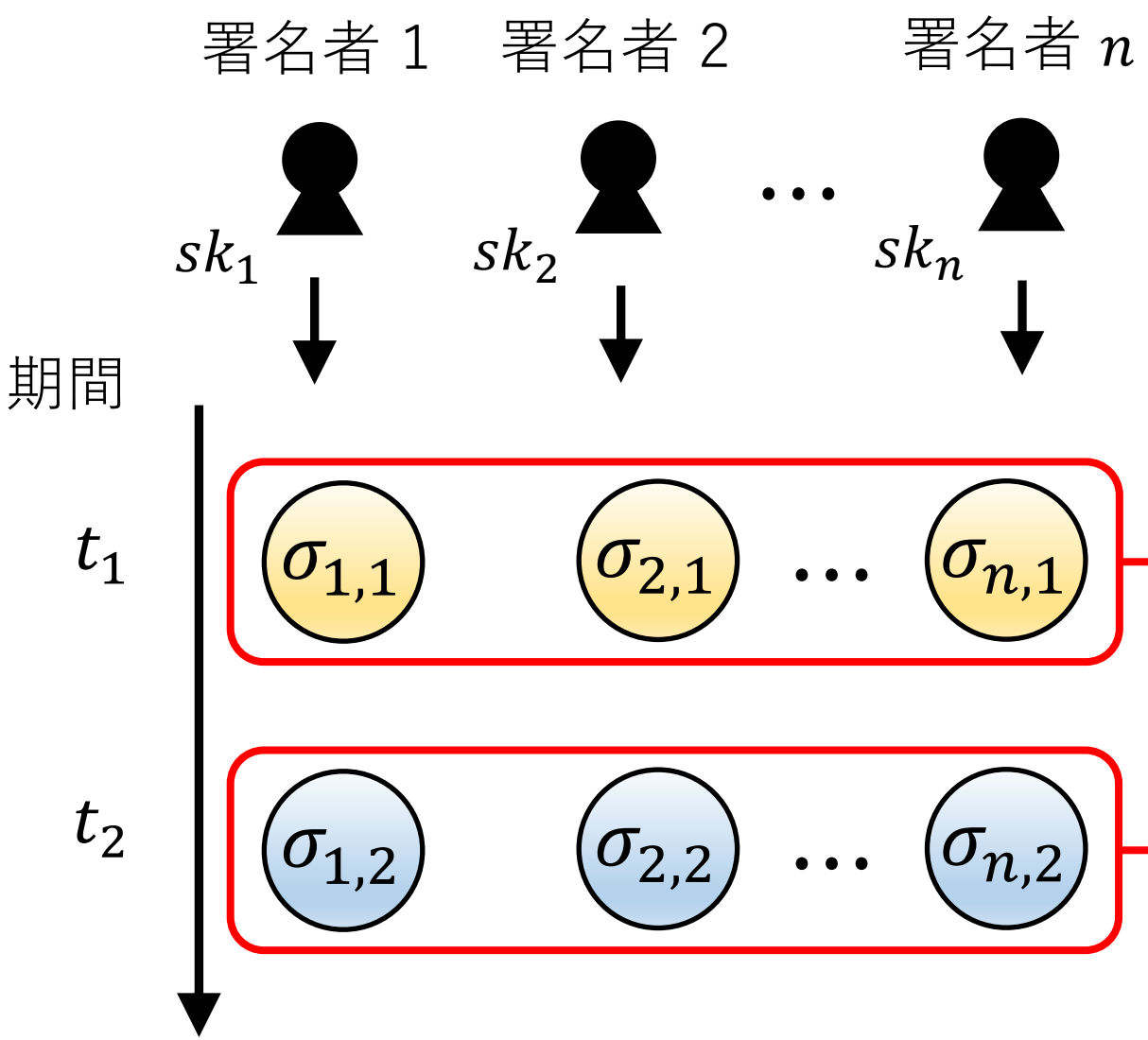
  😕 実用上現実的でない.

- 双線型写像を用いた構成 [BGLS03] （上の二つの方式より効率は現実的だが.）

  😕 ・集約署名の検証効率が悪い（ペアリング演算回数）
  ・ランダムオラクルモデルを用いて安全性が証明されている

集約署名の集約に制限をつければ，上記の欠点を緩和できるのでは？
→集約の仕方に制限をつけた集約署名の変種が考案される.

同期集約署名[AGH10]



同期集約署名

各署名者はそれぞれの期間ごとに一回だけ署名が生成できる.
同じ期間に生成された署名を集約できる.

同期集約署名の応用

同期集約署名の応用例

期間の区切りごとにデータをまとめたいシステムに用いることができる.

- 定期的にログデータが報告されるシステム
- 定期的にセンサーデータが送られるIoTシステム

など

先行研究の同期集約署名方式 (in the ROM)

Scheme	Assumption	pk size	Agg sig size	Agg Ver (parinig op)
[BGLS03] 集約署名	co-CDH + ROM	1	2 (t を署名に追加)	$n+1$ (n:集約した署名数)
[AGH10]	CDH + ROM	1	2	4
[LLY13]	LRSW + ROM	1	2	3

安全性の信頼性 Down

効率 Up

😕 対話型計算仮定！
安全上，対話型計算仮定の利用はできれば避けたい！

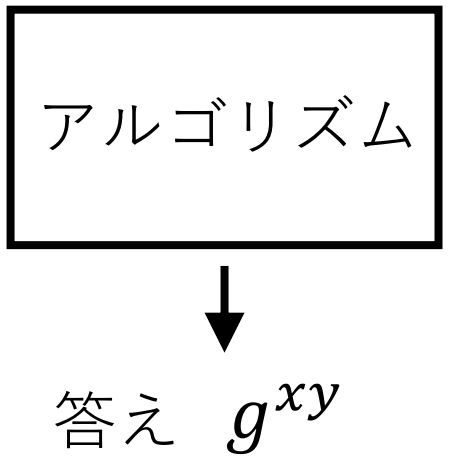
非対話型と対話型の計算仮定

非対話型と対話型の計算問題（計算仮定）

CDH 問題 (p, g, g^x, g^y) から g^{xy} を求めよ.

CDH 問題 (非対話型の計算問題)

問題 (p, g, g^x, g^y)

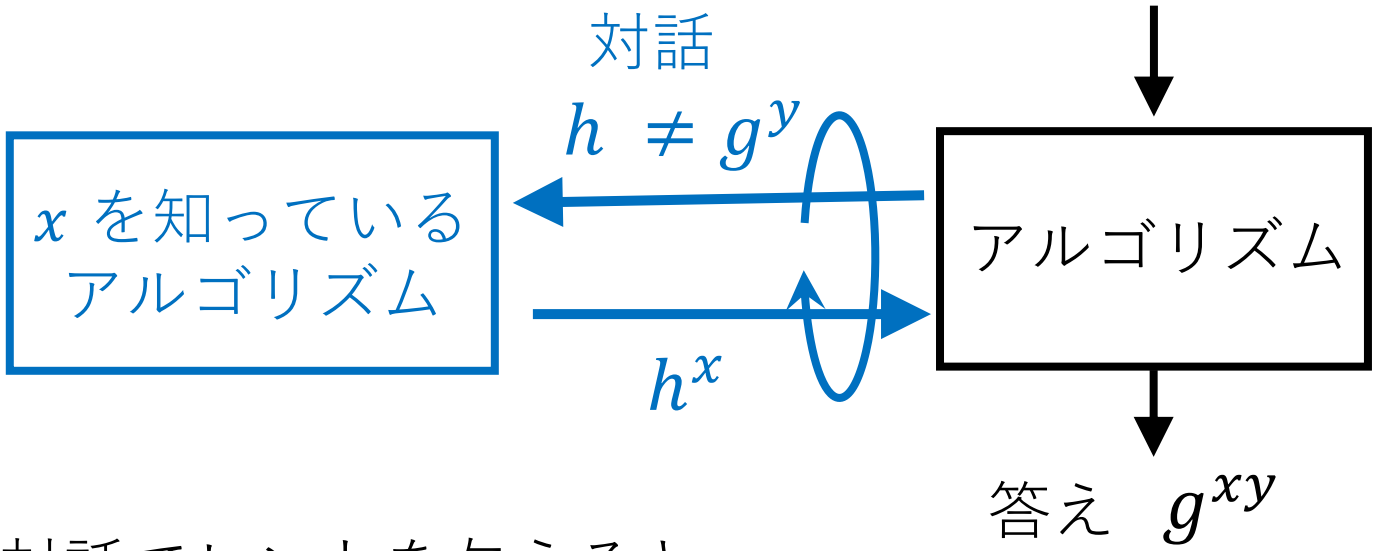


答え g^{xy}

CDH 問題の対話型計算問題版を考えると・・・

例え話なのでマネしないで. . .

問題 (p, g, g^x, g^y)



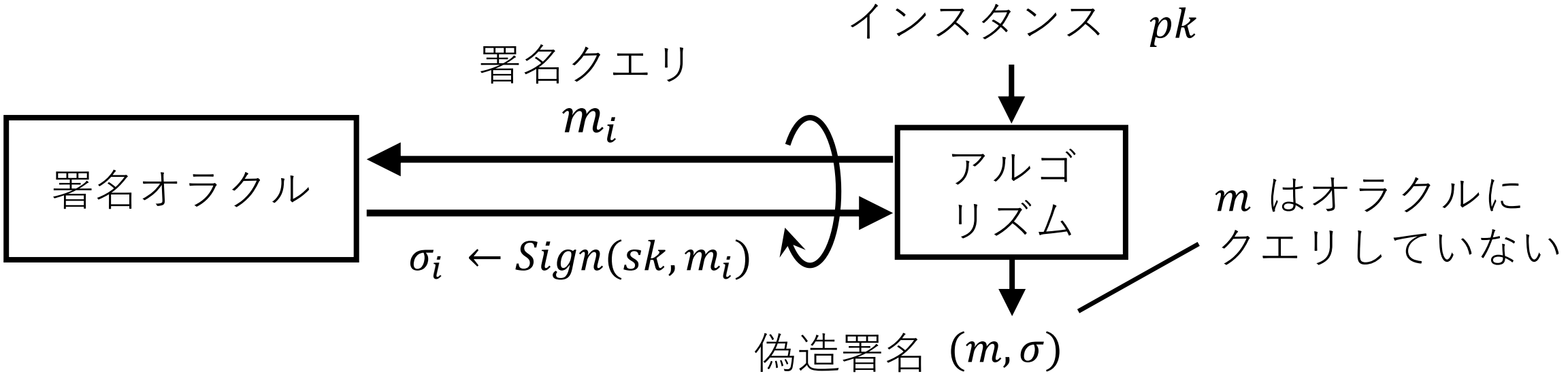
答え g^{xy}

対話でヒントを与えると,
計算問題が容易に解けることがある！

対話型計算仮定が生じる理由

署名方式の安全性が計算仮定に帰着できない場合,
その方式の署名が偽造できないこと自体をそのまま仮定にしてしまう.

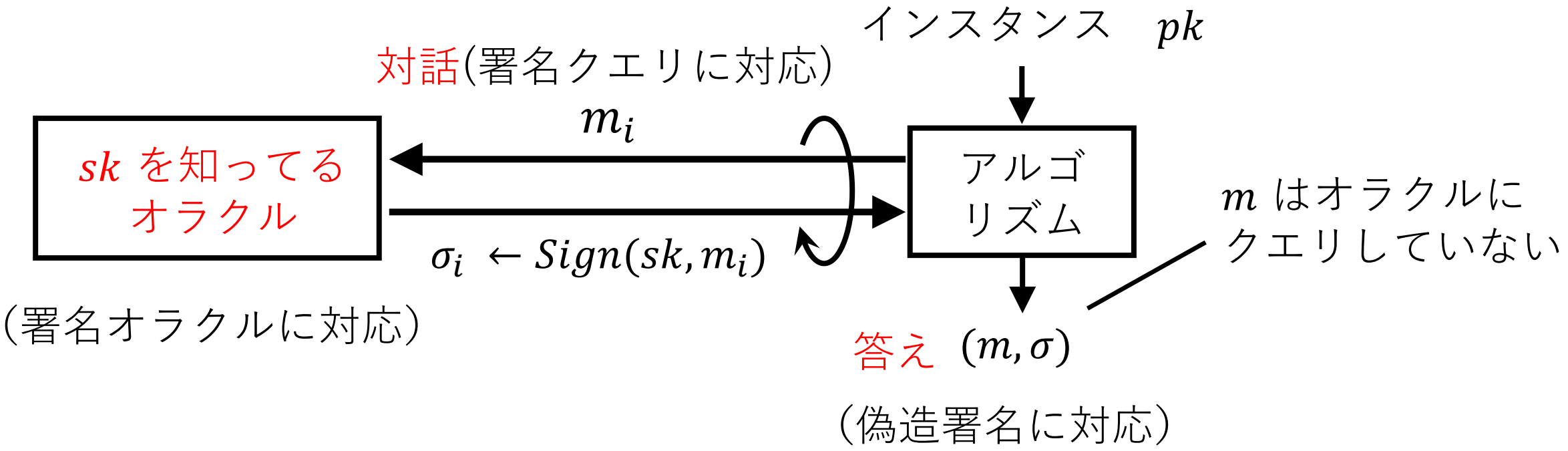
署名方式に対する EUF-CMA 安全性ゲーム



対話型計算仮定が生じる理由

署名方式の安全性が計算仮定に帰着できない場合,
その方式の署名が偽造できないこと自体をそのまま仮定にしてしまう.

対話型計算問題



CL署名[CL04]とLRSW仮定[LRSW99]

CL署名方式 [CL04]

$pp = (p, G, G_T, e)$

$(pk, sk) \leftarrow \mathrm{KeyGen}(pp)$

$sk = (x, y) \leftarrow Z_p, g \leftarrow G^*,$
$X = g^x, Y = g^y, pk = (g, X, Y)$

$\sigma \leftarrow \mathrm{Sign}\,(sk, m)$

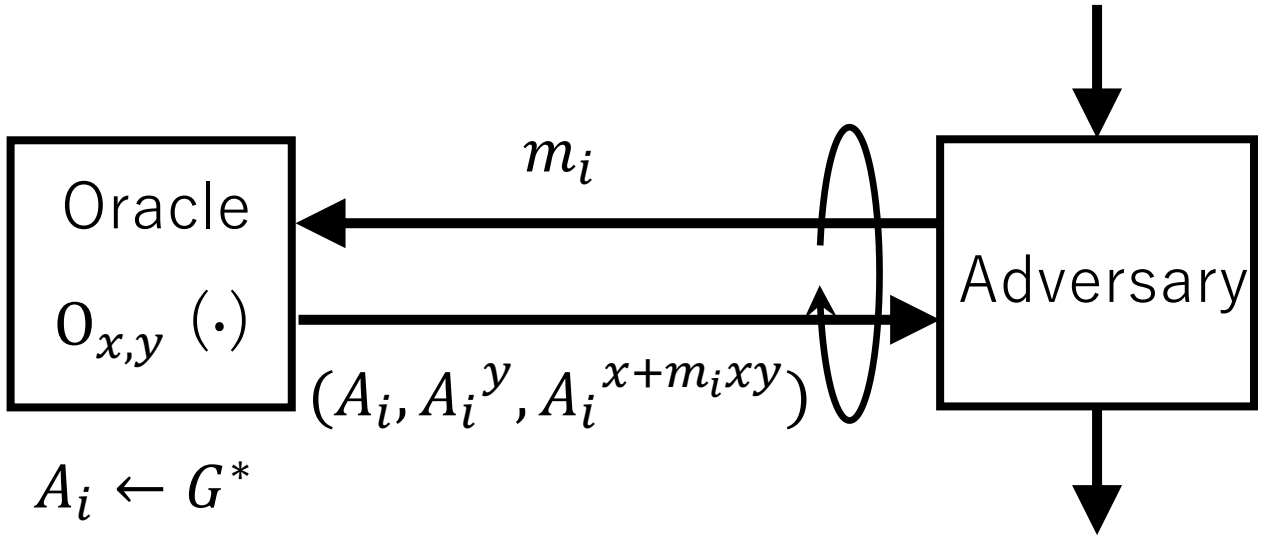
$A \leftarrow G^*, \sigma \leftarrow (A, A^y, A^{x+mxy})$

$1 \text{ or } 0 \leftarrow \mathrm{Verify}(pk, m, \sigma)$

$e(A, Y) = e(B, g)$?
$e(AB^m, X) = e(E, g)$?

LRSW仮定 [LRSW99]

Instance
$(p, G, G_T, e, g, X = g^x, Y = g^y)$



Solution (m, A, B, E)

1. $m \notin \{m_i\}$,
2. $m \in Z_p^*$,
3. $A \in G^*, B = A^y, E = A^{x+mxy}$

研究の成果

研究成果

Scheme	Assumption	pk size	Agg sig size	Agg Ver (parinig op)
[BGLS03] 集約署名	co-CDH + ROM	1	2 (t を署名に追加)	$n+1$ (n:集約した署名数)
[AGH10]	CDH + ROM	1	2	4
[LLY13]	LRSW + ROM	1	2	3
[LLY13] Our result	1-MSDH-2 +ROM	1	2	3

安全性の信頼性 Down

安全性の信頼性 Up

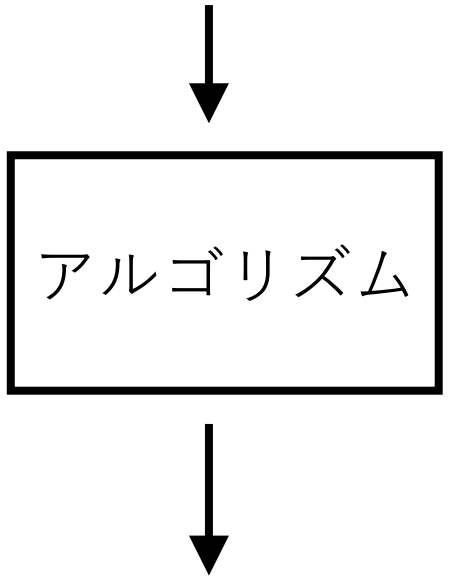
効率 Up

😀 非対話型計算仮定

1-MSDH-2 仮定 [PS18]

1-MSDH-2 問題（非対話型計算問題）

問題 $\left(p, G, G_T, e, g, \left\{g^{x^i}, g^{b \cdot x^i}\right\}_{i=1}^{2}, g^a, g^{abx}\right)$



答え (w, P, s, t)

問題の答えは次の4条件を全て
満たさなければならない.

1. $w \neq 0, s, t \in G$
2. 多項式 $P(X)$ は 1 次多項式.
3. 二つの多項式 $X + w$ と $P(X)$ は互いに素.
4. $s^{a(x+w)} = t^{xP(x)}$ を満たす.

同期集約署名方式と安全性の定義

同期集約署名の定義

$\mathrm{Setup}(1^{\lambda}, 1^{T}) \rightarrow pp$

$\mathrm{KeyGen}(pp) \rightarrow (pk, sk)$

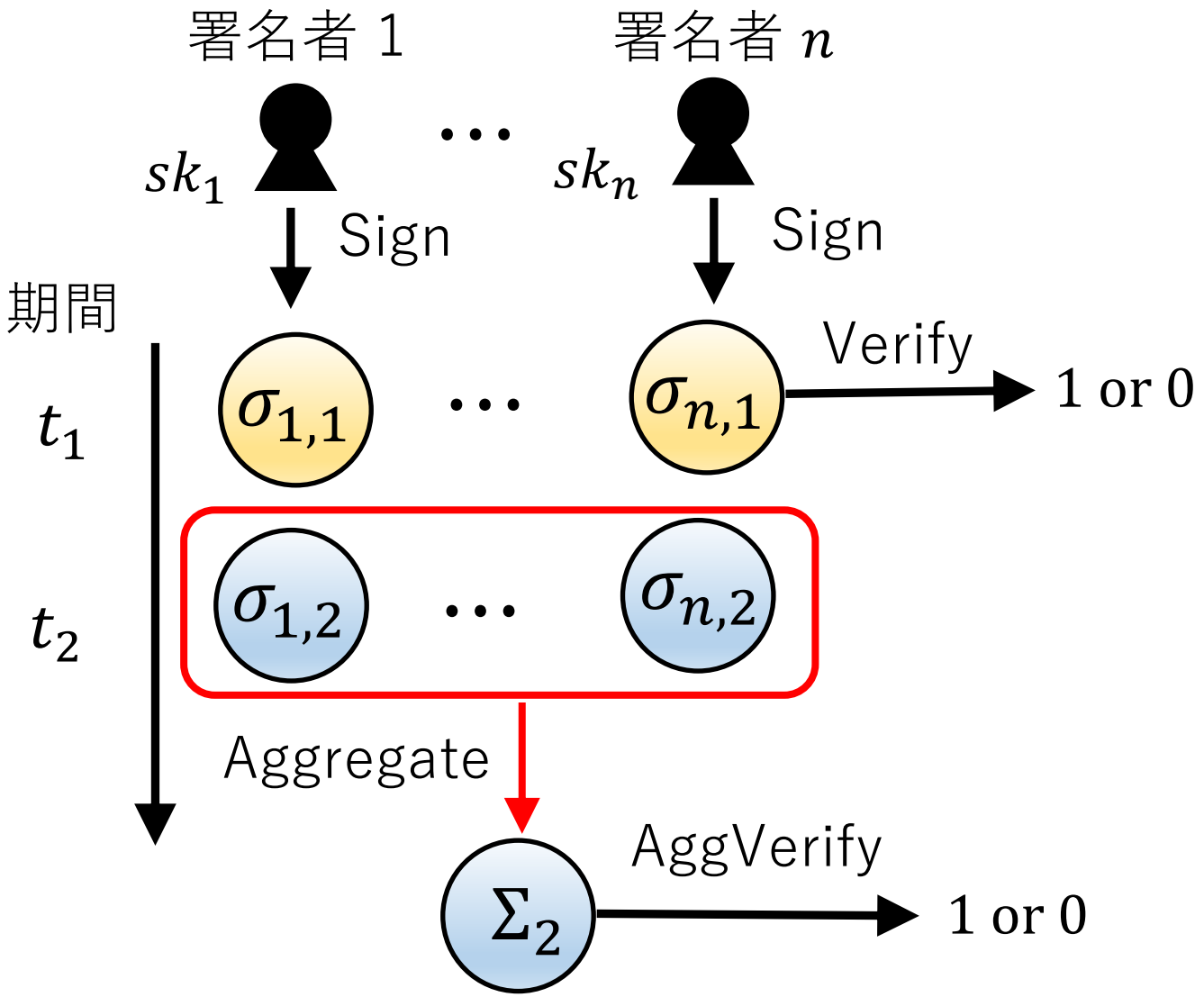
$\mathrm{Sign}(sk, t, m) \rightarrow \sigma$

$\mathrm{Verify}(pk, m, \sigma) \rightarrow 0 \text{ or } 1$

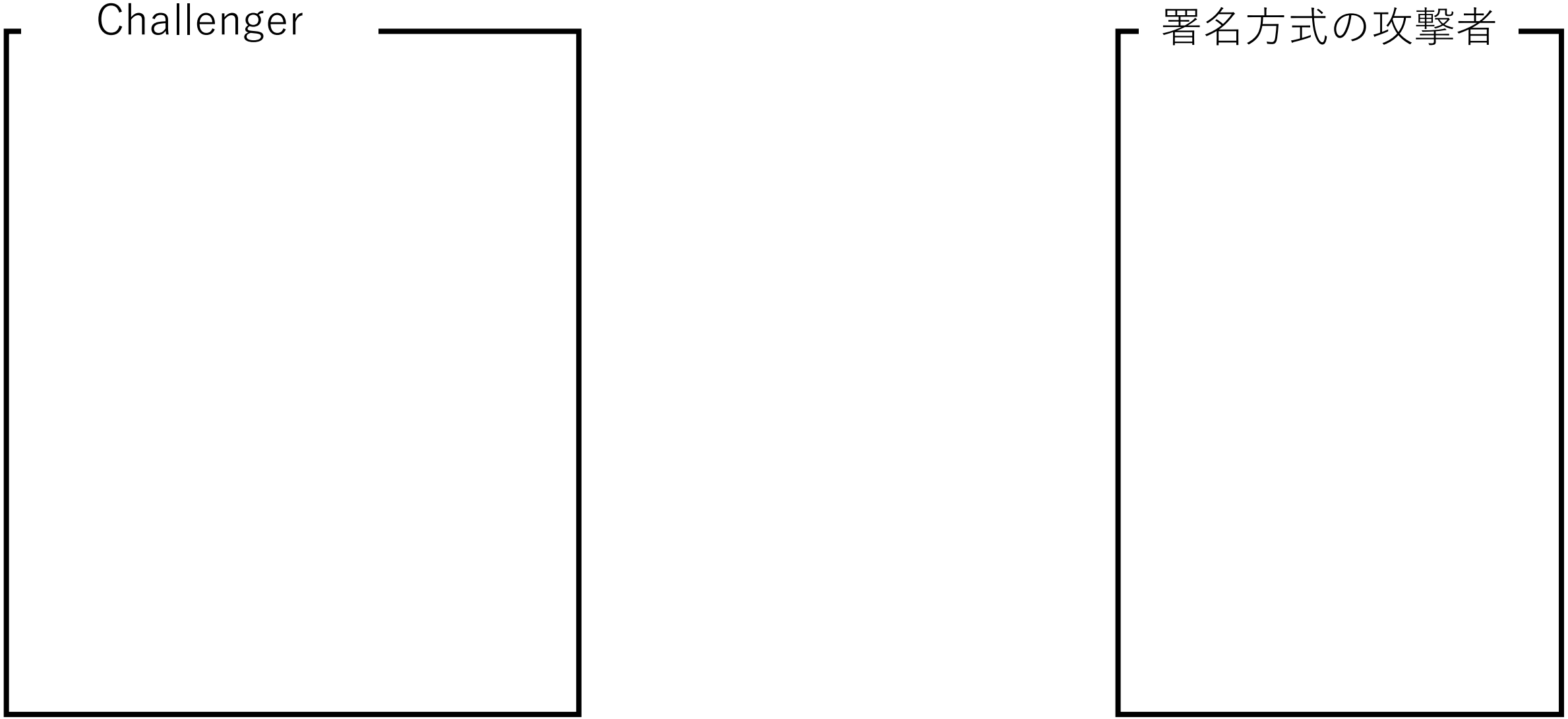
$\mathrm{Aggregate}((pk_i, m_i, \sigma_i)_{i=1}^{n}) \rightarrow \Sigma$

$\mathrm{AggVerify}((pk_i, m_i)_{i=1}^{n}, \Sigma) \rightarrow 0 \text{ or } 1$

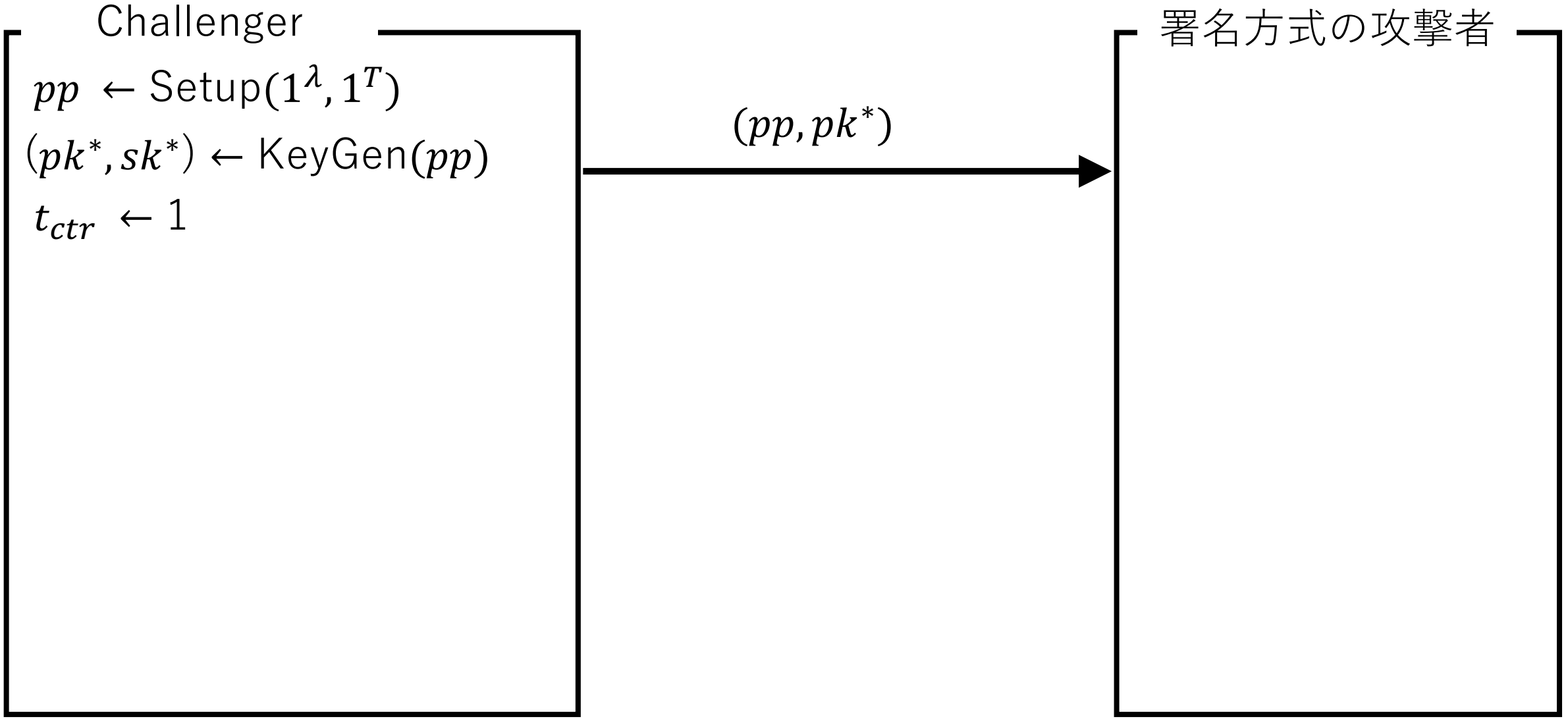
※ 今回は t の情報を σ, Σ に含める.



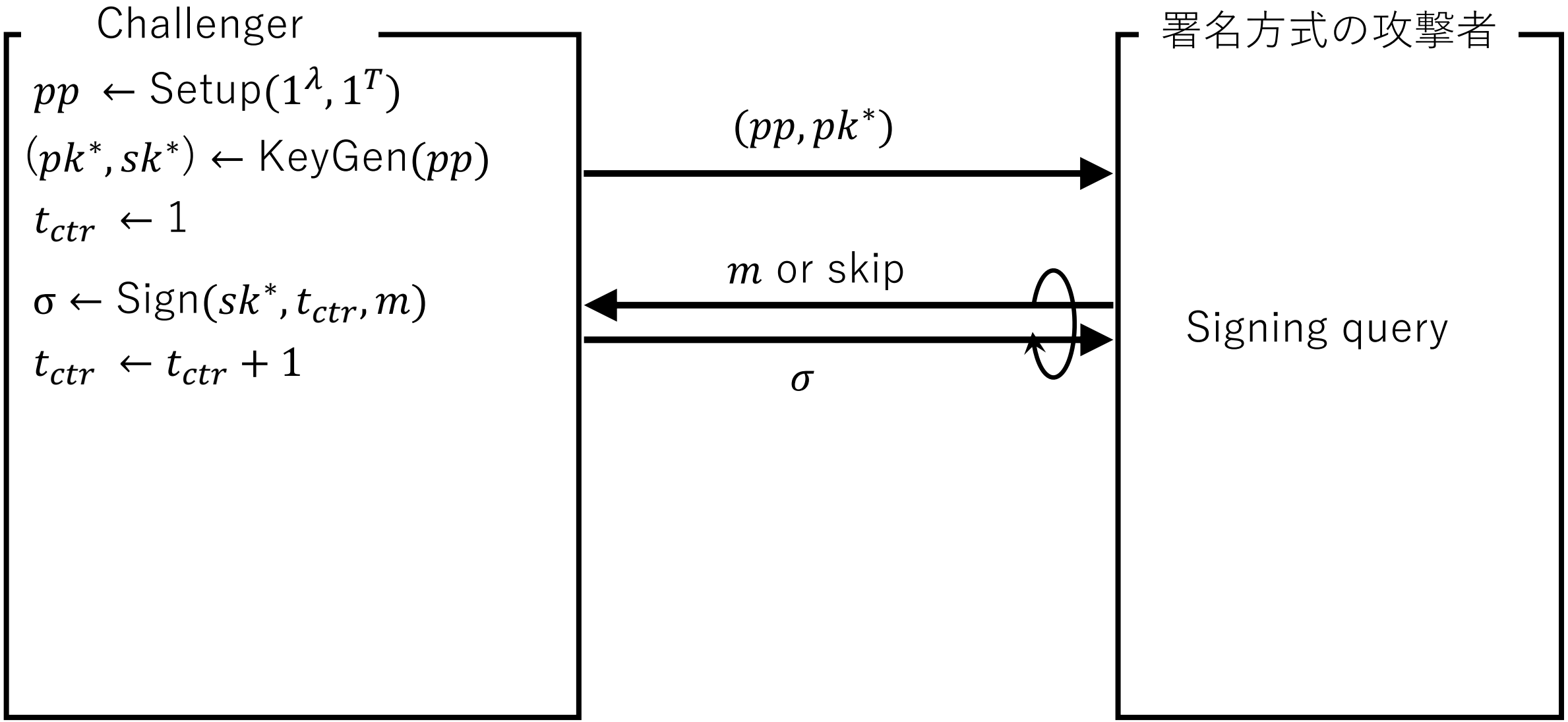
EUF-CMA 安全性（Certified-Key Model）



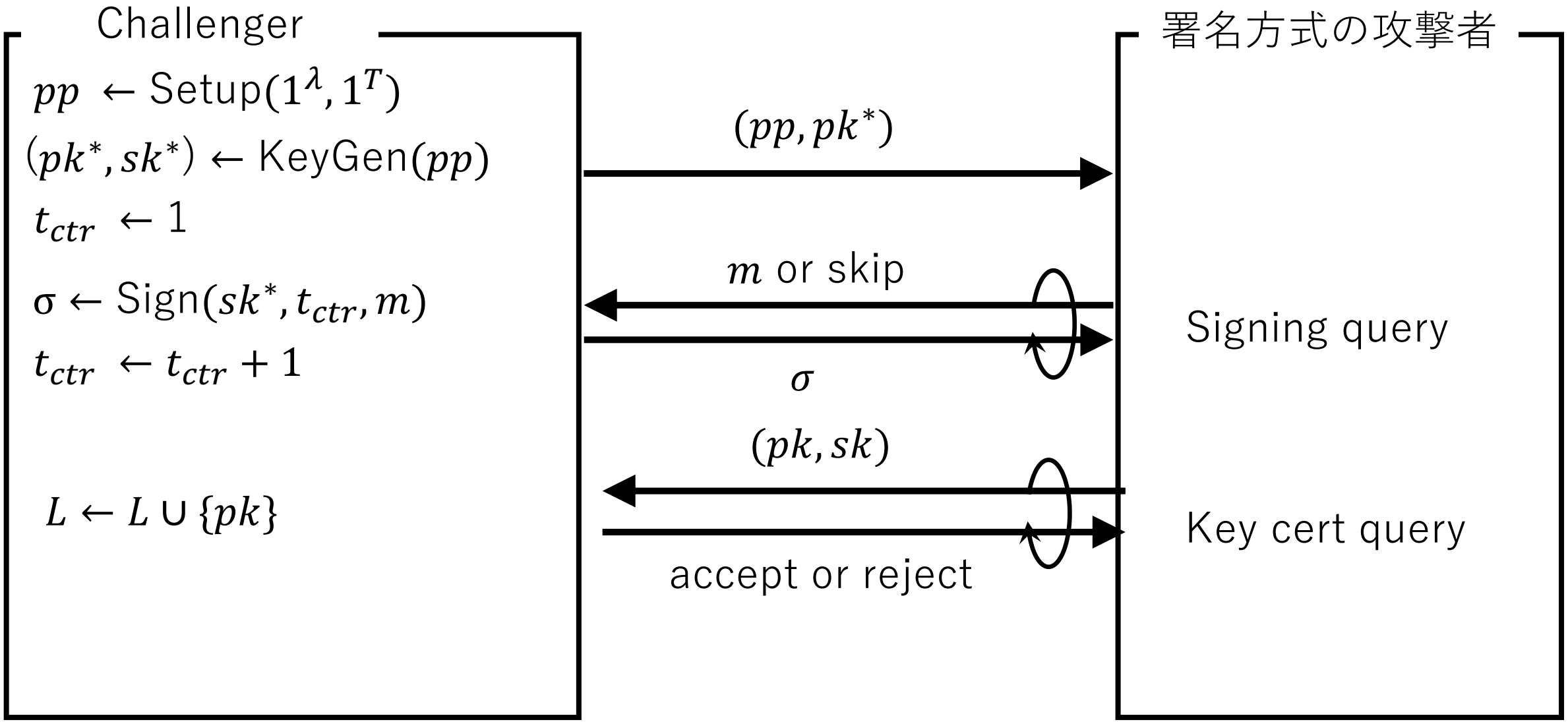
EUF-CMA 安全性（Certified-Key Model）



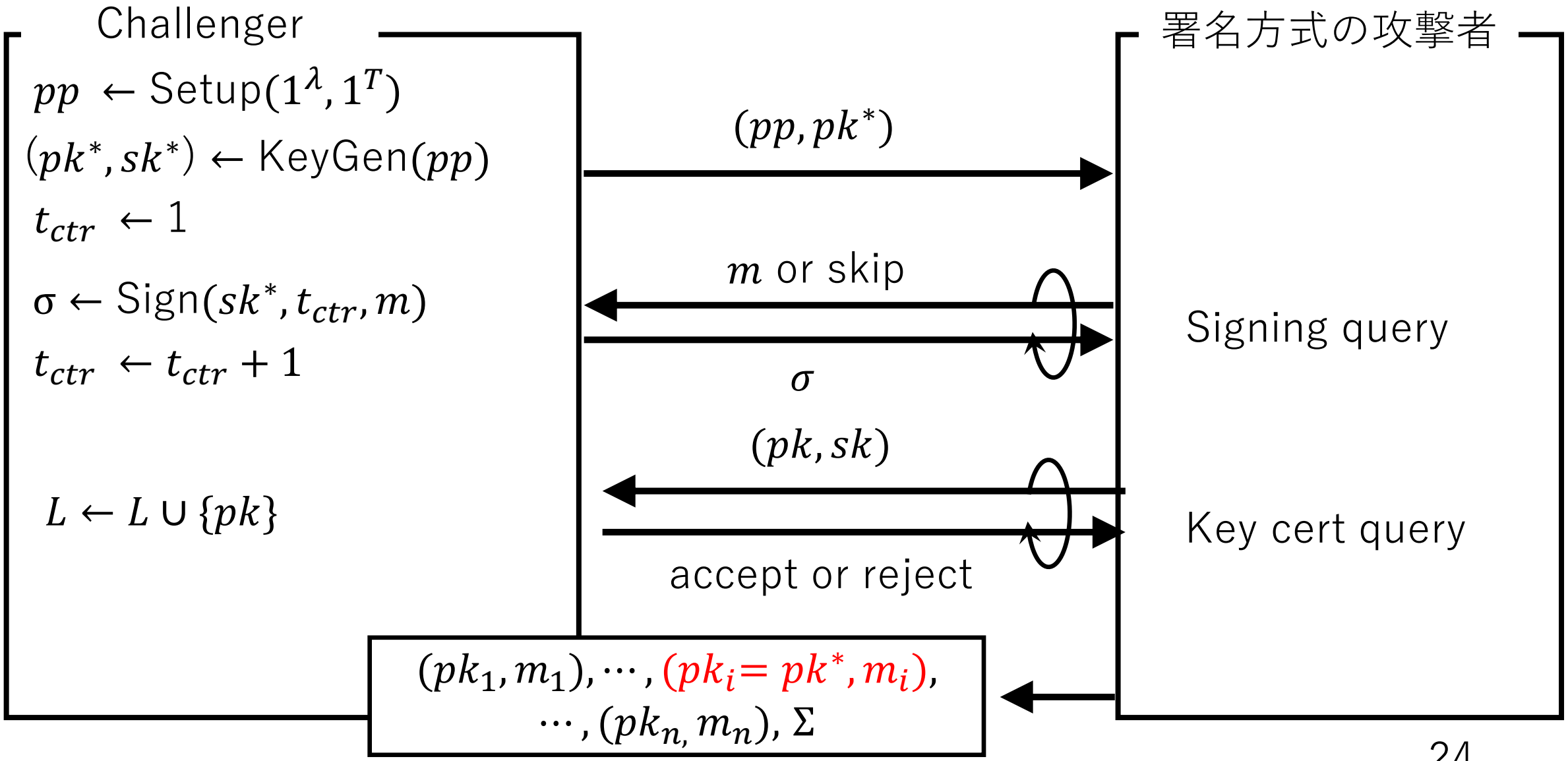
EUF-CMA 安全性（Certified-Key Model）



EUF-CMA 安全性（Certified-Key Model）



EUF-CMA 安全性（Certified-Key Model）



EUF-CMA 安全性（Certified-Key Model）

署名方式の攻撃者が偽造成功するとは

攻撃者の最終出力 $(pk_1, m_1), \cdots, (pk_i = pk^*, m_i), \cdots, (pk_n, m_n), \Sigma$

が次の 3 条件のを全て満たすこと.

1. AggVerify$((pk_i, m_i)_{i=1}^n, \Sigma) = 1$ であること.

2. Signing query で m_i をクエリしていない.

3. 全ての $(pk_1, \cdots, pk_n)$ が Key cert query で Accept されている.

LLY 同期集約署名の構成について

Bilinear Groups (Type 1)

Bilinear Group (p, G, G_T, e)

p：素数,　　G, G_T: 位数 p の巡回群

ペアリング $e : G \times G \rightarrow G_T$ 非退化な双線型写像

双線型性： $u, v \in G$, $a, b \in Z_p$ に対して

$$e(u^a, v^b) = e(u, v)^{ab}$$ が成り立つ.

非退化性： $g, \tilde{g} \in G^* = G/\{1_G\}$ に対して

$$e(g, \tilde{g}) \neq 1_{G_T}$$ が成り立つ.

LLY同期集約署名 [LLY13]

$pp = (p, G, G_T, e, g, H_1, H_2, H_3)$　　$g \in G^*$, $H_1, H_2 : [T] \to G$, $H_3 : [T] \times \{0,1\}^* \to G$

$(pk, sk) \leftarrow \mathrm{KeyGen}(pp)$

$sk_i = x_i \leftarrow Z_p$, $pk_i = X_i = g^{x_i}$

$\sigma \leftarrow \mathrm{Sign}\,(sk, m)$

$E_i = H_1(t)^{x_i} \cdot H_2(t)^{H_3(t,m)x_i}$,
$\sigma = (E_i, t)$

$\Sigma \leftarrow \mathrm{Aggregate}(\{pk_i, m_i, \sigma_i\}_{i=1}^n)$

$E = \prod_{i=1}^n E_i$, $\Sigma = (E_i, t)$

$1 \text{ or } 0 \leftarrow \mathrm{Vefiry}(pk, m, \sigma)$

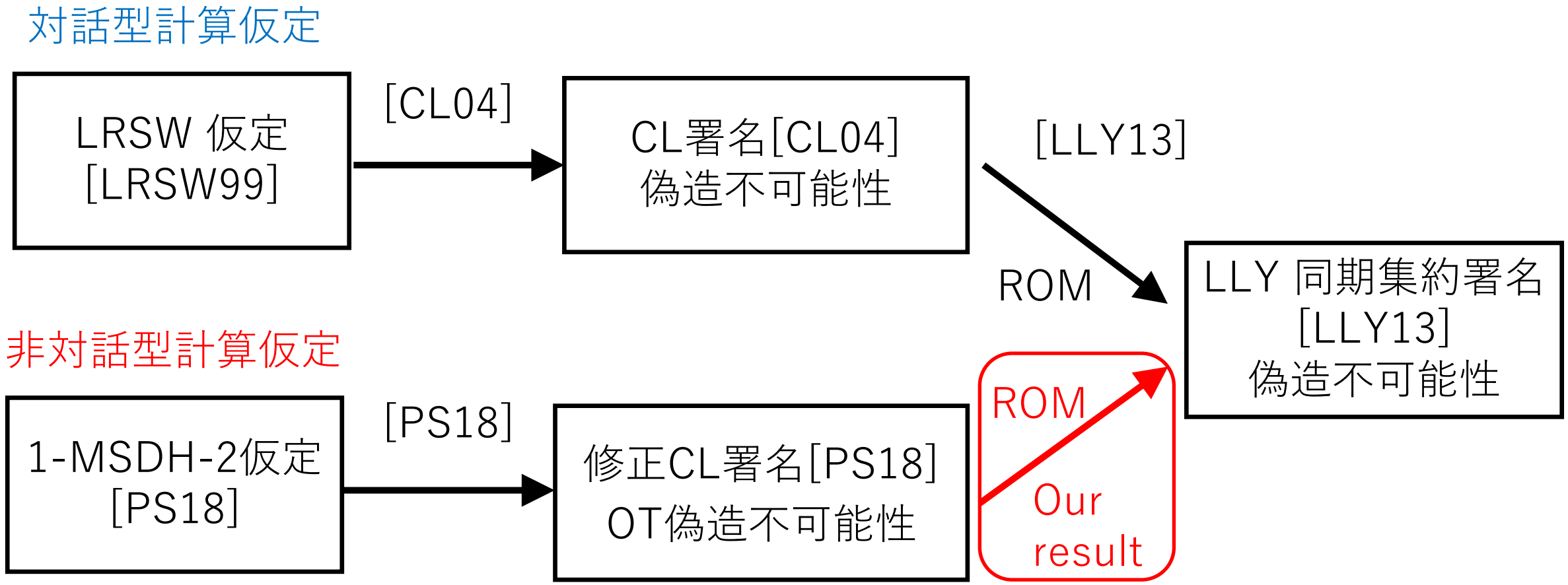
$m' = H_3(t, m)$
$e(E, g) = e\left(H_1(t)\, H_2(t)^{m'}, X\right)$?

$1 \text{ or } 0 \leftarrow \mathrm{AggVefiry}(\{pk_i, m_i\}_{i=1}^n, \Sigma)$

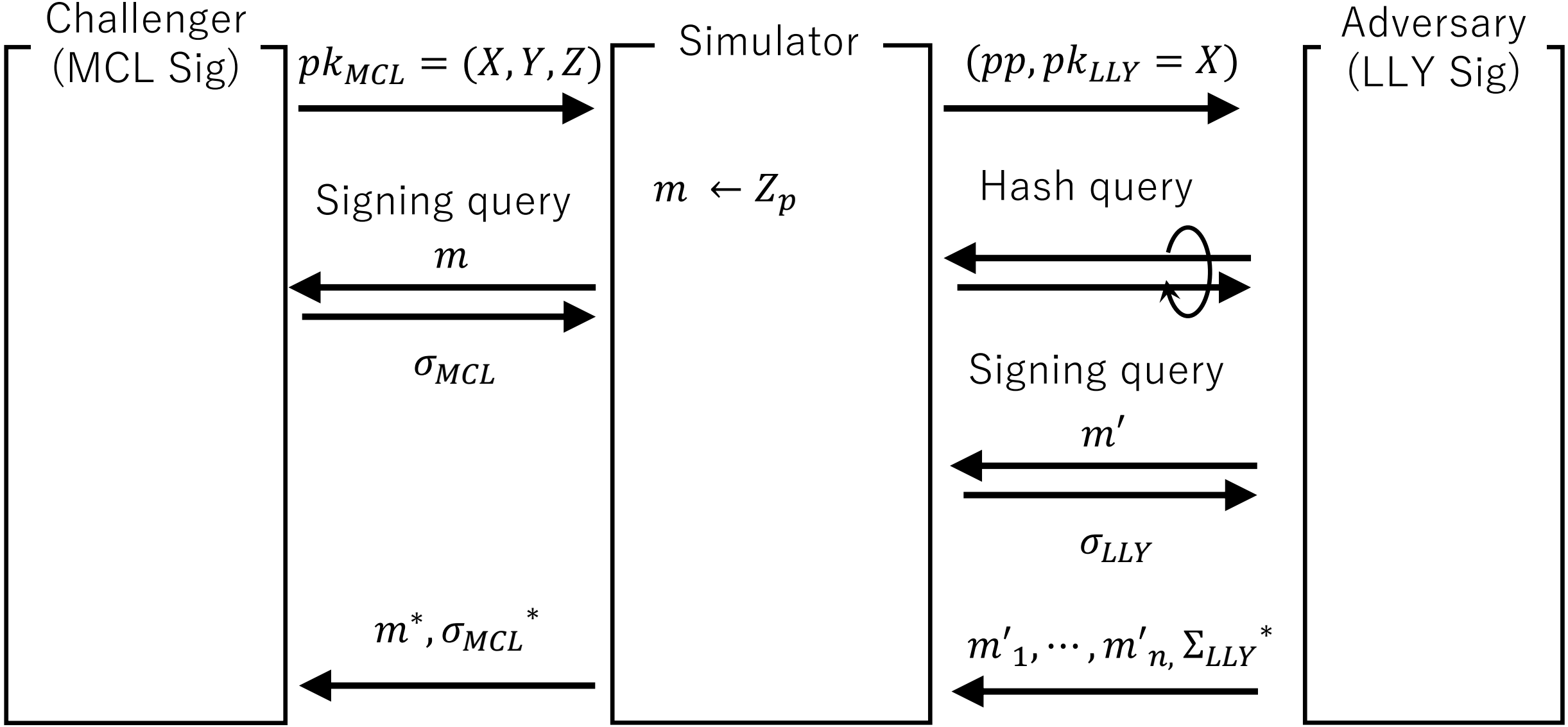
$m'_i = H_3(t, m_i)$
$e(E, g) = e(H_1(t), \prod_{i=1}^n X_i)$
$\cdot\, e\left(H_2(t), \prod_{i=1}^n {X_i}^{m'_i}\right)$?

安全性証明のアプローチ

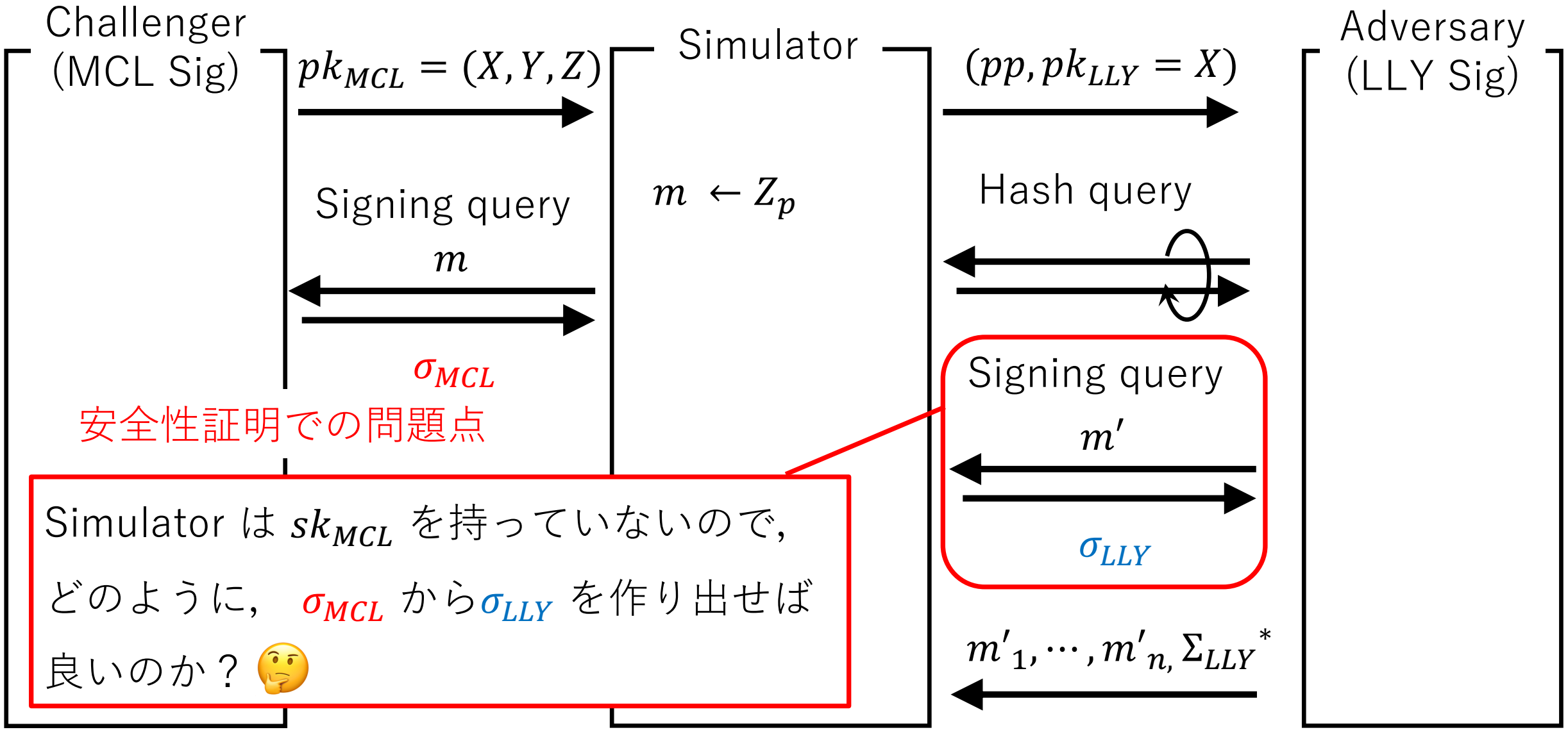
安全性証明のアプローチ



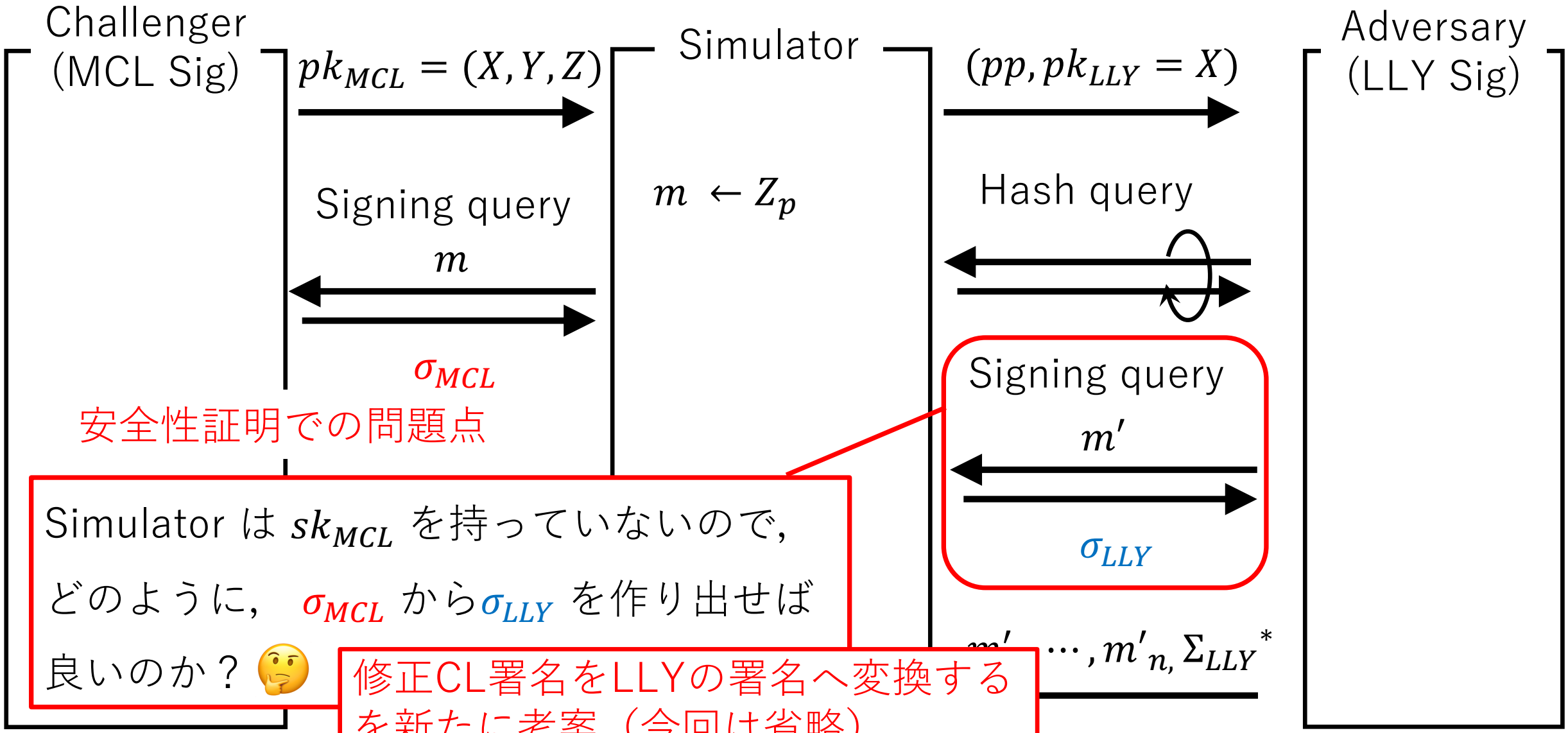
安全性証明の流れ


Challenger
(MCL Sig)
Simulator
Adversary
(LLY Sig)
$pk_{MCL} = (X, Y, Z)$
$(pp, pk_{LLY} = X)$
$m \leftarrow Z_p$
Signing query
m
σ_{MCL}
Hash query
Signing query
m'
σ_{LLY}
$m^*, {\sigma_{MCL}}^*$
${m'}_1, \cdots, {m'}_n, {\Sigma_{LLY}}^*$

安全性証明の流れ


Challenger
(MCL Sig)
Simulator
Adversary
(LLY Sig)
$pk_{MCL} = (X, Y, Z)$
$(pp, pk_{LLY} = X)$
$m \leftarrow Z_p$
Signing query
m
Hash query
σ_{MCL}
安全性証明での問題点
Signing query
m'
σ_{LLY}
Simulator は sk_{MCL} を持っていないので,
どのように, σ_{MCL} からσ_{LLY} を作り出せば
良いのか？
$m'_1, \cdots, m'_n, \Sigma_{LLY}{}^*$

安全性証明の流れ


Challenger
(MCL Sig)
$pk_{MCL} = (X, Y, Z)$
Simulator
$(pp, pk_{LLY} = X)$
Adversary
(LLY Sig)
$m \leftarrow Z_p$
Signing query
m
σ_{MCL}
Hash query
安全性証明での問題点
Signing query
m'
σ_{LLY}
Simulator は sk_{MCL} を持っていないので,
どのように, σ_{MCL} からσ_{LLY} を作り出せば
良いのか？
$m', \cdots, m'_n, \Sigma_{LLY}{}^*$
修正CL署名をLLYの署名へ変換する
を新たに考案（今回は省略）

まとめと今後の課題

まとめ

対話型の計算仮定（LRSW）により安全が証明された同期集約署名方式[LLY13]に対し非対話型の計算仮定（1-MSDH-2）仮定から安全性を証明した.

今後の課題

LRSW仮定を用いて安全性が証明がされた他の署名方式に対し，安全性を非対話型計算仮定で証明する統一的なフレームワークを発見する.

同期集約署名方式 [LLY13]，あるいは効率をあまり落とさない新たな同期集約署名を構成し，標準的な計算仮定から安全性を証明する.

本発表の内容

Masayuki Tezuka and Keisuke Tanaka. Improved Security Proof for the Camenisch-Lysyanskaya Signature-Based Synchronized Aggregate Signature Scheme. （ACISP 2020）

Appendix

安全性証明について

Modified Camenisch-Lysyanskaya Signature

$pp = (p, G, G_T, e)$

$(pk, sk) \leftarrow \mathrm{KeyGen}(pp)$

$sk = (x, y, z) \leftarrow Z_p$
$g \leftarrow G^*, X = g^x, Y = g^y, Z = g^z$
$pk = (g, X, Y, Z)$

$\sigma \leftarrow \mathrm{Sign}\,(sk, m)$

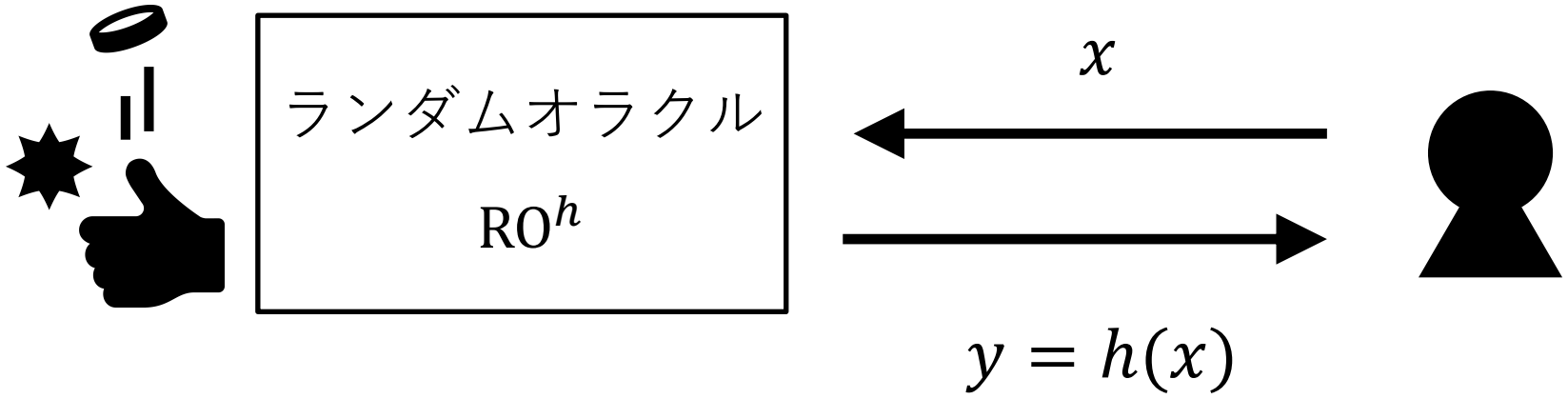
$w \leftarrow Z_p, A \leftarrow G^*, B = A^y$
$C = A^z, D = C^y, E = A^x B^{mx} D^{wx}$
$\sigma \leftarrow (w, A, B, C, D, E)$

$1 \text{ or } 0 \leftarrow \mathrm{Vefiry}(pk, m, \sigma)$

$e(A, Y) = e(B, g)$?
$e(A, Z) = e(C, g)$?
$e(C, Y) = e(D, g)$?
$e(AB^m D^w, X) = e(E, g)$?

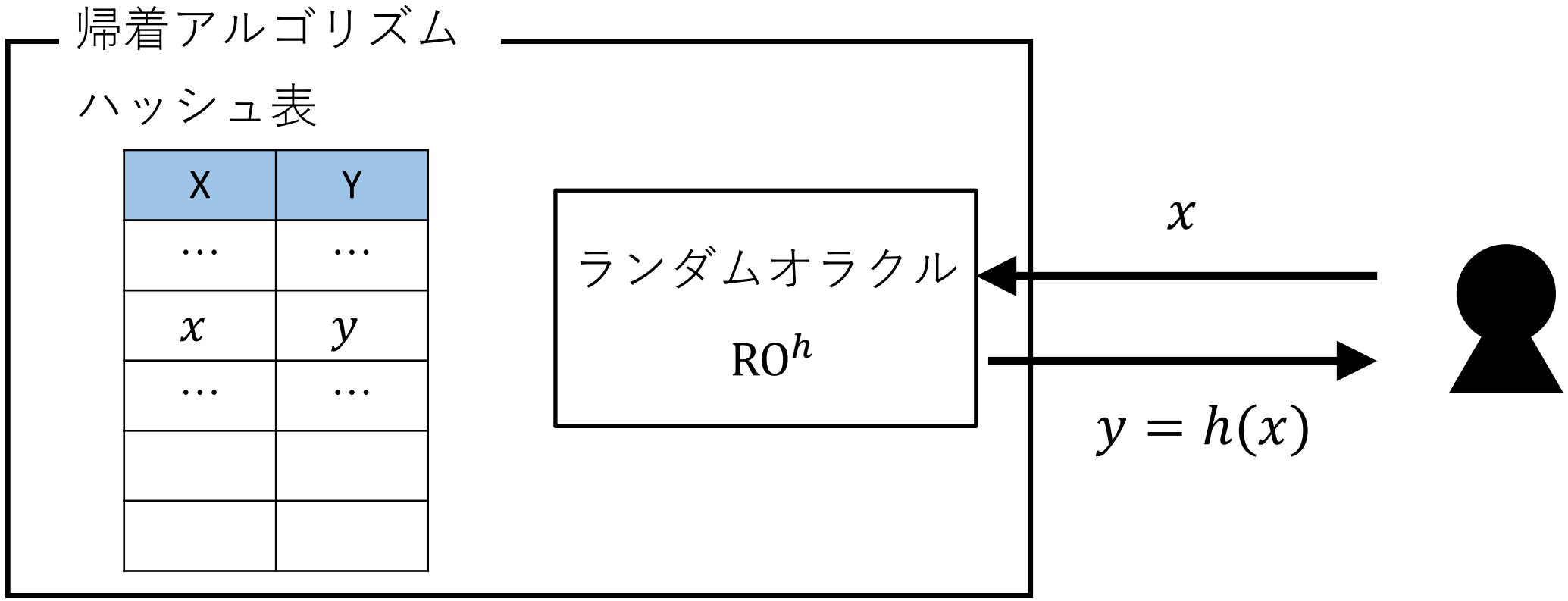
MCL Sig において q 個の署名を敵が所持していても．新たなメッセージに対する偽造不可能性は q-MSDH-2 仮定から証明できる．[PS18]

ランダムオラクルモデル (ROM)



ハッシュ値はランダムオラクルにクエリすることでしか計算できない.

Programmability (ROM)



Programmability

ハッシュ値の分布が一様ランダムと識別できない範囲において,

ハッシュ値を決めることでランダムオラクルをシミュレートが可能

Modified CL Sig から LLY Sig への変換

Modified CL Sig　署名者 i の鍵 $pk_i = (g^{x_i}, g^{y_i}, g^{z_i}),\ sk_i = (x_i, y_i, z_i)$

$$\sigma_i = (w_i, A_i, B_i = {A_i}^{y_i}, C_i = {A_i}^{z_i}, D_i = {A_i}^{y_i z_i}, E_i = {A_i}^{x_i} {B_i}^{m_i x_i} {D_i}^{w_i x_i})$$

署名の要素数 6個

LLY 同期集約署名　署名者 i の鍵 $pk_i = g^{x_i},\ sk_i = x_i$

$$\sigma_i = (E_i = H_1(t)^{x_i} H_2(t)^{H_3(t, m_i) x_i}, t)$$　署名の要素数 2個

・鍵の要素数を3個から1個にするにはどうするか？
・署名を構成する要素数を6個から2個にどう減らすか？
・ハッシュ関数をどのように利用（プログラム）するか？ 🤔

Modified CL Sig から LLY Sig への変換

変換方法

Step1

Modified CL Sig において，全ての署名者が同じ期間 t で共通の

$$w_i, A_i, {A_i}^{y_i}, {A_i}^{z_i}, {A_i}^{y_i z_i}$$

を用いて署名を生成する中間の変種方式を考える.

Step2

LLY Sig で用いるハッシュ関数 H_1, H_2, H_3 ハッシュ値をランダムオラクモデルのプログラマビリティを用いることで，中間の変種方式の署名からLLY Sig の署名をシミュレートする.

Modified CL から LLY Sig への変換 Step1

Modified CL Sig　署名者 i の署名鍵 $sk_i = (x_i, y_i, z_i)$

$$\sigma_i = (w_i, A_i, B_i = {A_i}^{y_i}, C_i = {A_i}^{z_i}, D_i = {A_i}^{y_i z_i}, E_i = {A_i}^{x_i} {B_i}^{m_i x_i} {D_i}^{w_i x_i})$$

↓

Step 1
すべての署名者が期間 t の署名で用いる w_i, A_i, B_i, C_i, D_i を同じにする.

中間生成物の同期集約署名　署名者 i の署名鍵 $sk_i = x_i$ （鍵を構成する要素数が1個）

$$\sigma_i = (w, A, B = A^y, C = A^z, D = A^{yz}, E_i = (AD^w)^{x_i} B^{m_i x_i}, t)$$

$$\Sigma \leftarrow \left(w, A, B, C, D, E = \prod_{i=1}^{n} E_i = \prod_{i=1}^{n} \left((AD^w)^{x_i} B^{m_i x_i} \right), t \right)$$

Modified CL から LLY Sig への変換 Step2

中間生成物の同期集約署名

$$\sigma_i = (w, A, B = A^y, C = A^z, D = A^{yz}, E_i = (AD^w)^{x_i} B^{m_i x_i}, t)$$

Step 2

$H_1(t) = AD^w$, $H_2(t) = B$ とプログラム，m_iの部分を $H_3(t, m_i)$ に変更する． w, A, B, C, D を捨てる．

LLY 同期集約署名

$$\sigma_i = (E_i = H_1(t)^{x_i} H_2(t)^{H_3(t, m_i) x_i}, t)$$

😀 この変換の技術とROMのProgrammability用いることにより安全性証明できる．

集約署名の変種・対話型計算仮定の補足

集約署名の変種

・集約署名 [BGLS03]
　複数の署名を集約できる．さらに複数の集約署名も集約できる

・順次集約署名(Sequential Aggregate Signature) [LMRS04]
　署名を一つずつ逐次的に集約できる

・同期集約署名(Synchronized Aggregate Signature) [AGH10]
　同じ時刻に生成された署名を集約できる

対話型計算仮定が用いられている例

- Lysyanskaya-Rivest-Sahai-Wolf (LRSW) 仮定
  Camenisch-Lysyanskaya (CL) 署名方式 [CL04]
  の偽造不可能性そのもの.
- Modified-LRSW (M-LRSW) 仮定
  IDベース順次集約署名 [BGOY07] の偽造不可能性の証明に用いられた.
  しかし M-LRSW仮定は[HLY09] により破られた.
- Abe-Groth-Halevi-Ohkubo (AGHO) 仮定
  AGHO構造維持署名方式 [AGHO11]の偽造不可能性そのもの.
- Pointcheval-Sanders (PS) 仮定
  PS署名方式 [PS16] の偽造不可能安全性そのもの.

References (1/2)

[AGH10] Ahn, Green, and Hohenberger. Synchronized aggregate signatures: new definitions, constructions and applications. (ACM CCS 2010)

[AGHO11] Abe, Groth, Haralambiev, and Ohkubo. Optimal structure-preserving signatures in asymmetric bilinear groups (9CRYPTO 2011)

[BGLS03] Boneh, Gentry, Lynn, and Shacham. Aggregate and verifiably encrypted signatures from bilinear maps. (EUROCRYPT 2003)

[BGOY07] Boldyreva, Gentry, O'Neill, and Yum. Ordered Multisignatures and Identity-Based Sequential Aggregate Signatures, with Applications to Secure Routing. (ACM CCS 2007)

[CL04] Camenisch and Lysyanskaya. Signature schemes and anonymous credentials from bilinear maps. (CRYPTO 2004)

[HKW15] Hohenberger, Koppula, Waters. Universal signature aggregators. (EUROCRYPT 2015)

[HLY09] Hwang, Lee, and Yung, Universal forgery of the Identity-Based Sequential Aggregate Signature Scheme (ASIACCS 2009)

[HSW13] Hohenberger, Sahai, and Waters. Full domain hash from (leveled) multilinear maps and identity-based aggregate signatures . (CRYPTO 2013)

References (2/2)

[LLY13] Lee, Lee, and Yung. Aggregating CL-signatures revisited: Extended functionality and better efficiency. (FC 2013)

[LMRS04] Lysyanskaya, Micali, Reyzin, and Shacham. Sequential aggregate signatures from trapdoor permutations. (EUROCRYPT 2004)

[LRSW99] Lysyanskaya, Rivest, Sahai, and Wolf. Pseudonym systems. (SAC 1999)

[PS16] Pointcheval and Sanders. Short Randomizable Signatures. (CT-RSA 2016)

[PS18] Pointcheval and Sanders. Reassessing security of randomizable signatures. (CT-RSA 2018)