暗号・認証分科会

2025年度　活動報告

目次

- 分科会紹介
- 分科会活動内容
- 暗号・認証とは
- 高機能暗号とは
- 各メンバー活動テーマ報告

分科会紹介

- 指導教員
  - 趙 晋輝 先生（リーダー）、花岡悟一郎 先生（指導教員）

- メンバー ６人
  - 情セ大　：百瀬耕平、村上誠樹、阿部草太
  - 中大　　：佐藤佑哉、浜崎昂多、吉田幸貴

分科会活動内容

・安心・安全な情報社会の実現には、暗号・認証技術の適切な理解が重要。

・新たな暗号・認証技術の設計の際、第三者に対して正確でわかりやすい説明が必要

目的

最先端暗号・認証技術の安全性を、一般の人々が理解できるような説明手法を構築

暗号・認証とは

- 暗号（Cryptography）：情報を隠す
  - 役割：第三者にデータの内容を知られないようにする（機密性の確保）
  - 例：共通鍵暗号、公開鍵暗号

- 認証（Authentication）：正しさを確かめる
  - 役割：通信相手が「本人」であること、データが「改竄されていないこと」を証明する
  - 例：デジタル署名やパスワード、生体認証

高機能暗号とは

- クラウド化やAIの発展→ 既存の暗号・認証技術では不便

↓

- 「既存の暗号・認証技術」 ＋「追加機能」＝「**高機能暗号**」

- 追加機能の例
  - 暗号化したまま検索
  - 暗号化の際にアクセス権限を設定
  - ㊙情報を持っていることを、情報を明かさずに証明

各メンバー活動テーマ報告

- 吉田 幸貴：準同型暗号
- 阿部 草太：ゼロ知識証明
- 村上 誠樹：マルチパーティ計算
- 浜崎 昂多：秘密分散（(k, n)閾値法）
- 百瀬 耕平：秘密分散（Shamirの秘密分散法）

準同型暗号

中央大学 M1 吉田幸貴

準同型暗号とは

- 準同型性： $f(a)+f(b)=f(a+b)$

- f：暗号化関数
  - $f(a)$：a の暗号化データ
  - 暗号化されたデータ同士の演算が、もとの平文の情報を保つ

↓ ↓ ↓

  - 暗号文のまま計算できて、それを復号しても計算が成り立つ

準同型でない暗号方式（イメージ）

- 「２」＋「３」＝「?」

↓（暗号化…３乗、復号…$\sqrt[3]{\ }$）

- 「８」＋「２７」＝「３５」

↓復号

$\sqrt[3]{35} \neq 5$

正しい答えを得るためには、「８」「２７」を復号してから計算する必要がある

準同型暗号（イメージ）

- 「２」＋「３」＝「?」

↓（暗号化…３倍、復号…1/3倍）

- 「６」＋「９」＝「１５」

↓復号

5

復号しなくても、元の式と同じ計算ができる

ユースケース

- 医療データの解析
  - 大勢の患者の診療データや遺伝子情報の統計をとりたい
  - データを暗号化したまま収集し、解析
  - 患者のプライバシーを保護しつつ、医療研究ができる

- クラウドでの秘匿計算
  - ユーザーデータを暗号化したままクラウドに送信
  - クラウド側は内容を知ることなく計算し、返却
  - 返ってきたデータを復号
  - 個人情報や機密情報を守りながらクラウド利用ができる

ゼロ知識証明

- 情報セキュリティ大学院大学 M1 阿部草太

ゼロ知識証明とは・・・？

■ ゼロ知識証明とは、暗号分野に用いられる技術で、

ある主張が正しいことを証明しながらも、
その主張に関する追加情報を一切明かすことなく証明する手法

＝秘密の内容を一切しゃべらず、「秘密を知っているよ」と証明する

■ 英語で

「Zero-Knowledge Interactive Proof」（ZKIP）と言われたりもする

ゼロ知識証明の構成要素（求められる機能）

■ゼロ知識証明が成り立つためには、以下の３要素が必要とされる。

１．完全性・・・証明者が正しい主張している→高い確率で正しいと判断

２．健全性・・・偽の主張をした→正しいとする確率が極めて低い

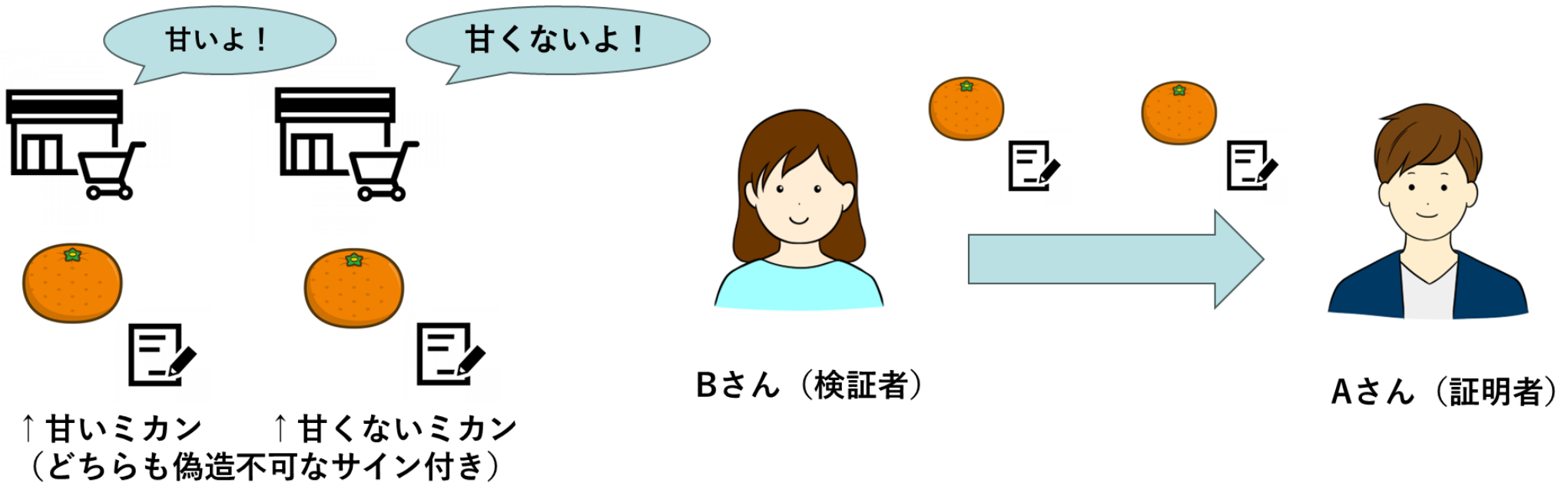
３．ゼロ知識証明性・・・検証者が知るのは、「主張が正しい」という事実のみ

ゼロ知識証明とは・・・？

■ゼロ知識証明をわかりやすく

⭐例「Aさんは甘いみかんの見分け方を知っている」を証明①

前提：「あまいミカン」と「あまくないミカン」が売っているスーパーがあり、この表示に誤りはない



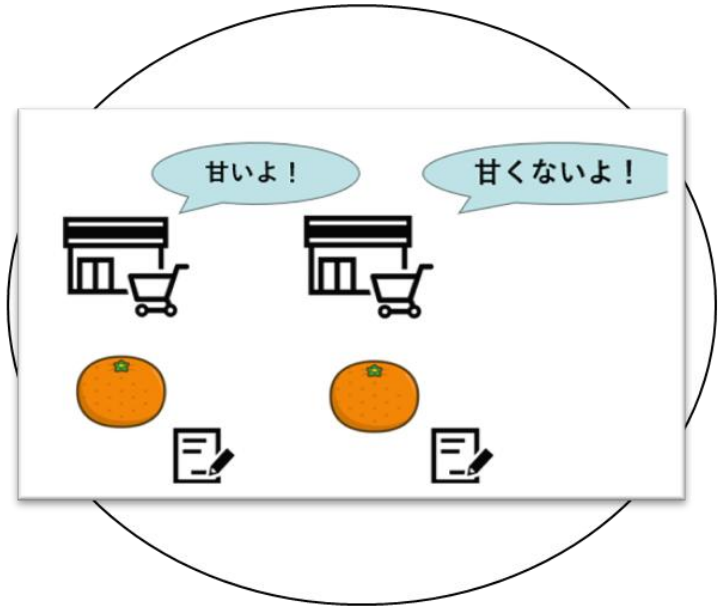
ゼロ知識証明を破壊する要因	要因に対する対策（条件）
・Aさんがあらかじめ甘いミカンを用意する	・Bさんがミカンにサインを書いた
・Aさんがミカンのサインを偽造する	・Bさんが世界でただ一つのペンでサインを書いた
・送るミカンに甘いものが含まれている保証がない	・各専門店から購入している
・識別方法のヒントが得られてしまう可能性がある	・ミカンを送りあうのみで、判定は食べて確認
・偶然Aさんが甘いミカンを返している	・複数回繰り返し、偶然の成功の確率を下げる

ミカンの例における３要素

・完全性

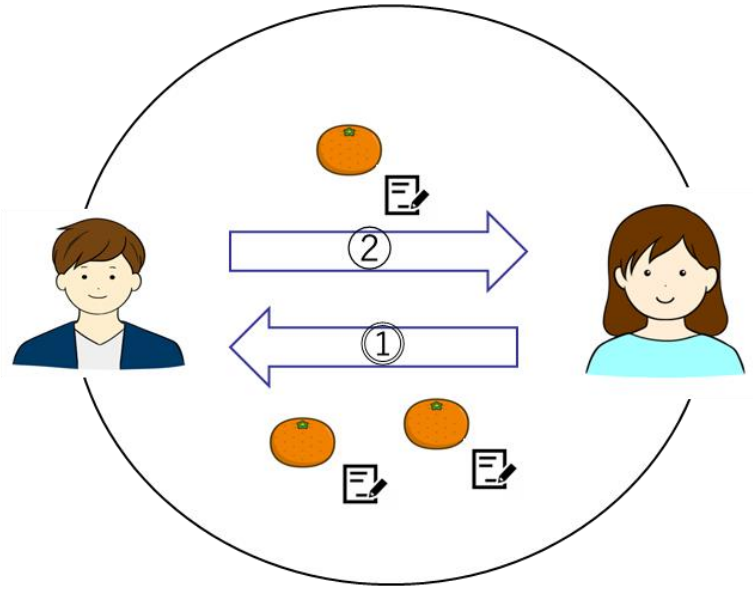
検証者が確認し、証明者の嘘は見抜ける

・健全性



証明者は不正する余地がない

・ゼロ知識性



ミカンの見極め方のヒントは
一切得られていない

マルチパーティ計算について

情報セキュリティ大学院大学 修士課程2年 村上誠樹

身近なたとえで理解する

年齢・立場が異なる趣味友５人で飲み会。
でも、誰がいくら払えるか知られたくない…

- 現金を見せたくない
- クレジットカードを見られると社会的ステータスがバレる
- でも、傾斜はつけつつ公平に支払いを分担したい…

秘密計算を使えば、解決できる！

秘密計算があれば…!

守りたいデータ

- 各自の支払い能力
- 所持品からわかる状況
- 持っている現金・クレカ

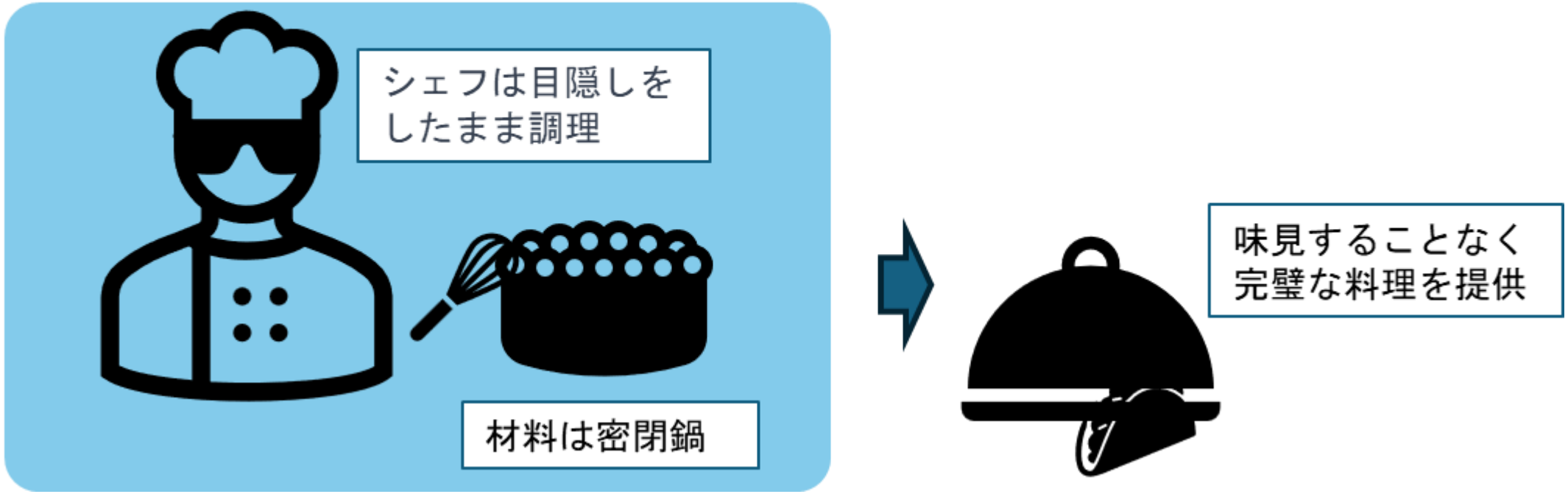
- 趣味友の友情関係

実行したいこと

- 合計金額だけを計算
- 一人当たり金額を算出
- 互いの財布事情は秘密

絵解きでわかるマルチパーティ計算

料理人の調理工程で例える「中身を見ず計算」のイメージ



絵解きでわかるマルチパーティ計算

ステップ1：秘密レシピ分割（シェア化）

材料を5つの密閉鍋（シェア）に小分けする
3鍋あれば復元可能。2鍋以下では味は全くわからない

密閉鍋1 ◆〇‰
密閉鍋2 kϵ〒
密閉鍋3 ζωΞ
密閉鍋4 жюд
密閉鍋5 ゞ‡∀

ステップ2：各シェフが目隠しのまま手順実行

それぞれが自分の鍋にだけレシピ操作を実施（「混ぜる」・「加える」等が演算処理）
鍋は不透明なので味見できず、シェフはほかの鍋を見れない

ステップ3：断片を統合することで、完成料理を復元

計算過程では、個々の入力値は互いに一切不明のまま

完成調理1 ‰〇◆
完成調理2 〒ϵk
完成調理3 Ξωζ

ステップ4：安全性を確認

全工程で、料理人の誰も元レシピを知らず、他の料理人の断片は不明のまま
最終的な料理だけが残る **途中のプロセスは隠蔽されたまま、完璧な料理が作れる！**

将来に必須な技術「秘密計算」

- まとめ

データを「鍵がかかった箱」に入れたまま、分析・計算できる技術。

- 何が嬉しいか

「便利さ」と「プライバシー」を両立させ、
個人の安心と社会の発展を同時に実現できる。

データ・AI時代の現代
秘密を保ったまま情報を扱える夢のような技術

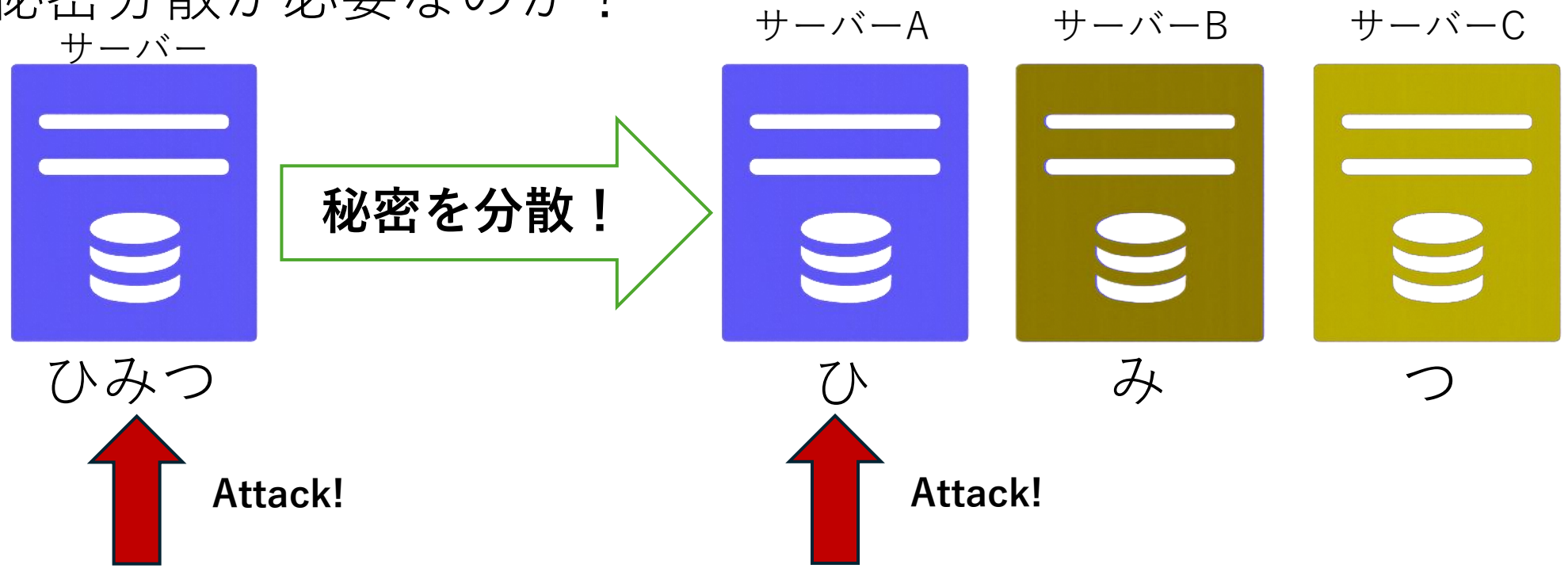
秘密計算の技術が、未来の私たちの生活を安全で便利にします

秘密分散（(２，３)閾値法）

中央大学 M1 浜崎昂多

(2，3)閾値法

➢なぜ秘密分散が必要なのか？



防御が破られる
と全部漏れる！

防御が破られても分割された
情報でわからない！（「ひ」のみ）

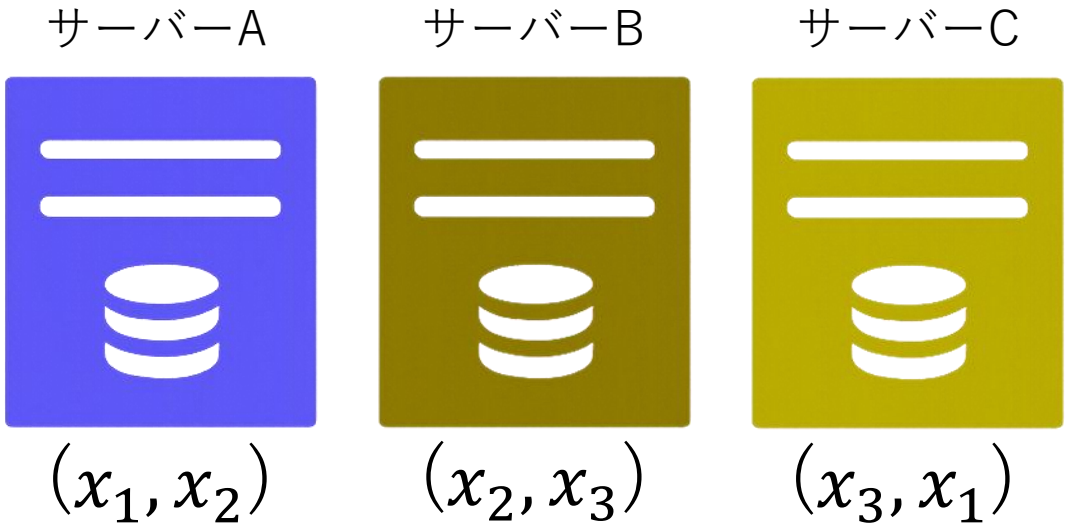
(2，3)閾値法

➢仕組み

秘密を3つのシェアに
ランダムで分割

$X = x_1 + x_2 + x_3$

各シェアの組み合わせを配布



➢復元

任意の2台で秘密を復元可能（ x_1, x_2, x_3がそろう！）

✓ちなみに３は配る人数，２は復元に必要な人数を表す。
例えば（k,n）はn人に配り，k人で復元できる

(2, 3)閾値法

➢分割したまま計算可能

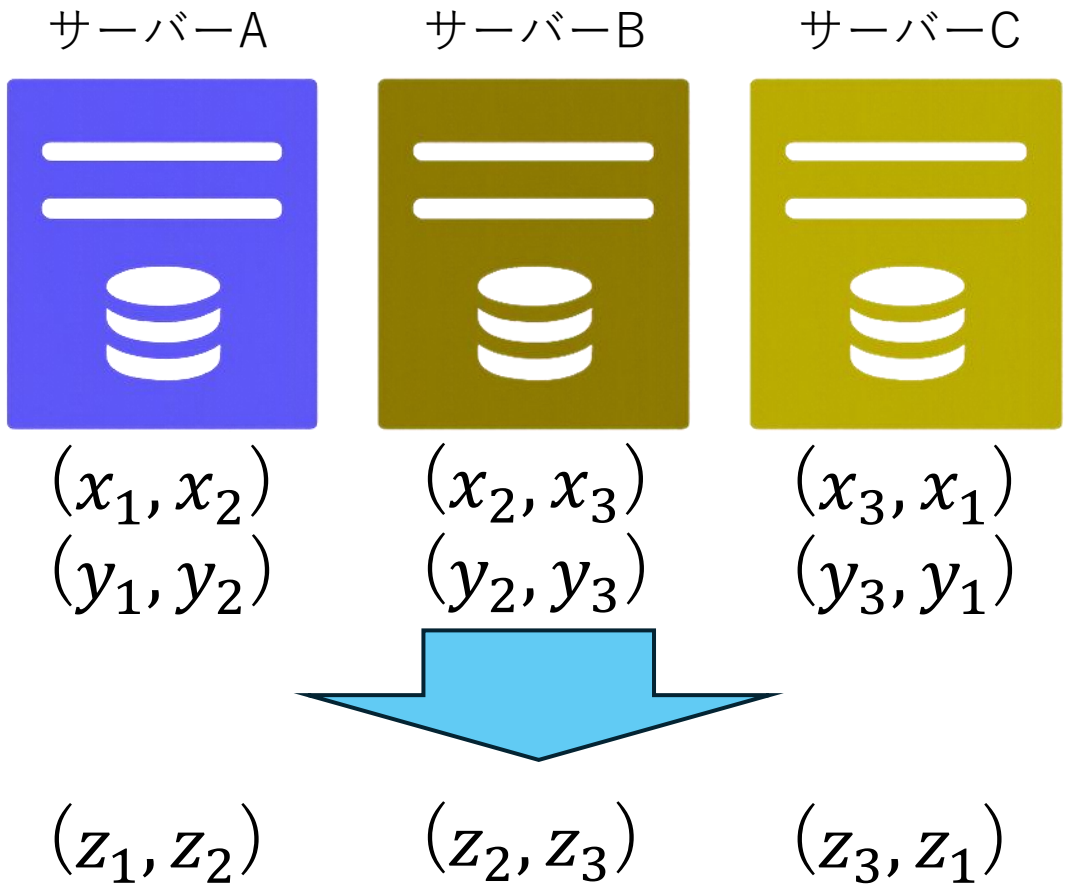
秘密X, Yの足し算を行いたい！

計算済みのシェアをZとすると

$Z_1 = x_1 + y_1$,（Z_2, Z_3も同様）

✓このZを復元することで
計算結果が得られる

✓加減算・乗算などの計算が可能で
秘密を公開することなく
データ分析やAI構築に利用できる



秘密分散（Shamirの秘密分散法）

情報セキュリティ大学院大学 M1 百瀬耕平

はじめに

- これまで説明があった(k, n)閾値型の秘密分散を実現する代表的手法であり、マルチパーティ計算の構成要素としても広く使われるものとして「**Shamir（シャミア）の秘密分散法**」があげられる
- Shamirの秘密分散法のような手法によって、実社会での活用例は実現される

Shamirの秘密分散法の仕組み

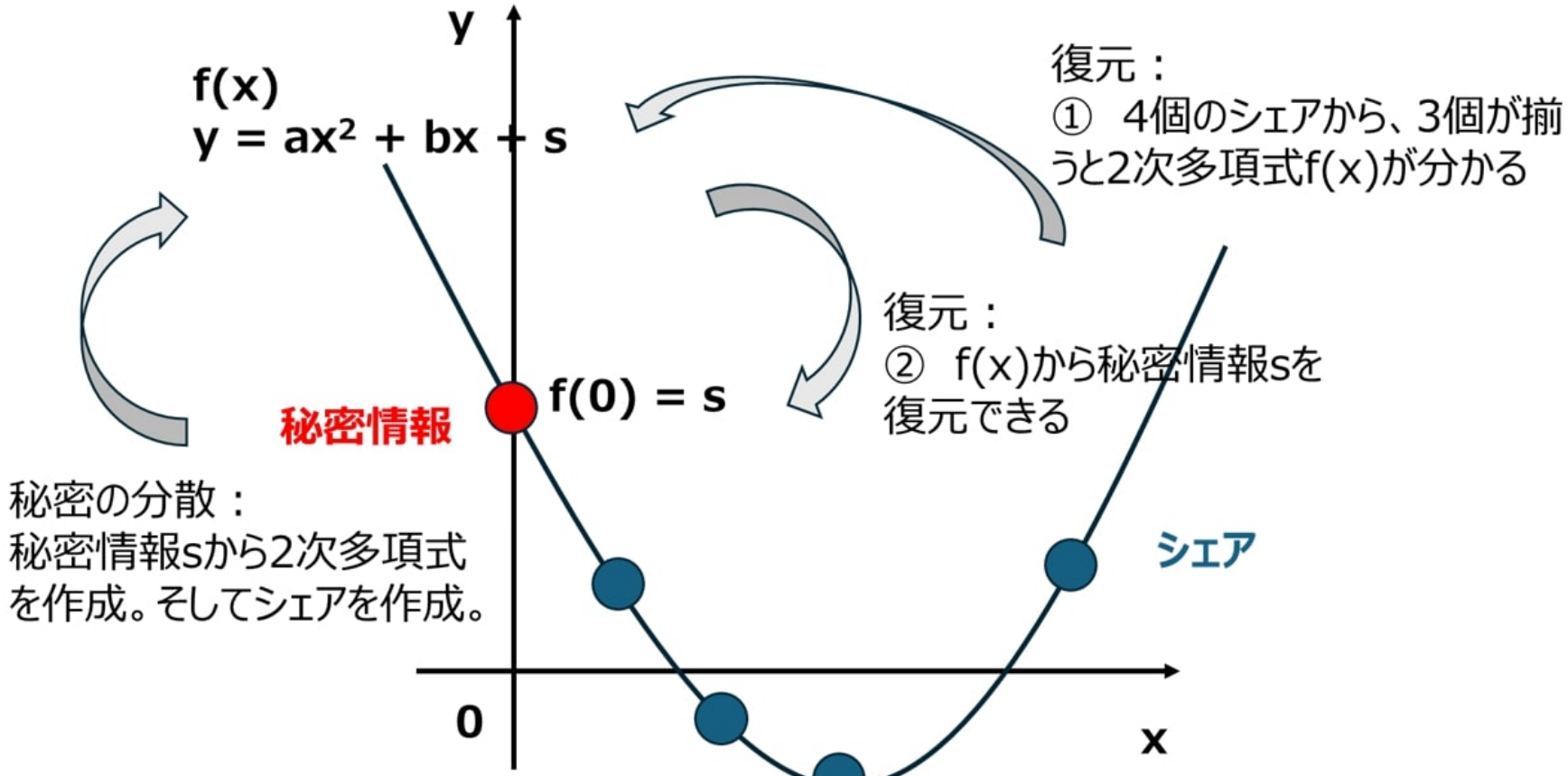
＜秘密分散の実現＞
(k, n)閾値法において
①　秘密情報を**s**とする
②　定数項をsとし、残りの係数をランダムに選んで、**k－1次多項式 f(x)**を作る
③　通る点の座標 $(x_i, f(x_i))(i=1,2,\cdots,\mathbf{n})$ をシェアとして各参加者に配布する

＜復元方法＞
- k個以上のシェアが手元にあると、

→ f(x)が通る点が k個以上分かる
→ 多項式補間によりf(x)を特定できる
→ 秘密情報 f(0)を特定できる
となり、秘密情報sを復元できる（逆に、k－1 個以下では秘密情報は不明）

Shamirの秘密分散法の仕組み　図解

(k, n) = (3, 4)の例



秘密情報の足し算

- Shamirの秘密分散法において、各参加者が対応するシェアを足し算してその結果を復元すると、誰も各秘密情報の中身を知ることが無く、**元の秘密情報の和を得ることができる**

秘密情報の足し算　図解

(k, n) = (3, 4)の例