データ収集における第三者機関を介さない匿名性保護の耐攻撃性の向上

山口　武志† 　隅　　崇佳†† 　上土井陽子†† 　若林　真一††

† 広島市立大学 情報科学部
†† 広島市立大学大学院 情報科学研究科

あらまし　近年，プライバシーの保護はデータ収集において重要な問題になっている．プライバシーが保護されていない場合，データを共有することを拒絶したり，正しくないデータを提供することが起こりやすい．したがって，各回答者から送信されたデータを解析者が回答者に結びつけることができない方法でオリジナル・データを集める必要がある．この問題に対処するため，Z.Yang らは，信頼された第三者機関の支援なしで，回答者からのデータを匿名で集めるオンライン・データ収集のための効率的な暗号手法を提案している．本稿ではそのプロトコルの semi-honest モデルの制限を弱めた時に行うことが可能な攻撃に耐えることができる新しいプロトコルを提案し，その匿名性を理論的に検証し，実験的に性能を評価する．

キーワード　匿名化，プライバシー保護，データ収集，semi-honest モデル

Anonymity-Preserving Data Collection against Malicious Participants without a Trusted Third Party

Takeshi YAMAGUCHI†, Takayoshi SUMI††, Yoko KAMIDOI††, and Shin'ichi WAKABAYASHI††

† Faculty of Information Sciences, Hiroshima City University
†† Graduate School of Information Sciences, Hiroshima City University

Abstract　In recent years, protection of privacy has become important in data collection. If protection of privacy does not preserve, individuals may either refuse to share their data or present incorrect data. Therefore, it is necessary to collect original data by the way the miner cannot link a respondent's data to the respondent. To solve this issue, Z.Yang, et al. proposed an efficient cryptographic technique for online data collection, in which data from many responders is collected anonymously, without a trusted third party. They assume that a miner and participants are semi-honest. In this paper, we propose a protocol based on Yang's protocol against malicious participants in a weaker semi-honest model. Additionally, we prove the anonymity theoretically and examine performance experimentally.

Key words　anominity, privacy protection, data collection, semi-honest model

1. ま え が き

コンピュータやネットワーク技術の急速な発展によって，莫大なデータは世界中で日々，収集され分析されている．そのうちのいくつかは，プライバシーに関わるものである．そして，データのプライバシー保護の問題は一般の人々から注目を受けており，データを提出する回答者の意欲にプライバシーの懸念が影響する．プライバシーが保護されていない場合，データを共有することを拒絶する，あるいは正しくないデータを提供するということが考えられる．従って，プライバシーを保護することは重要である．プライバシーを保護する手法として信頼された第三者機関が仲介する方法が考えられる．この方法では，各回答者はデータを収集する解析者の公開鍵で暗号化し，第三者機関に送信する．次に，第三者機関は送られてきたデータを並び替え，解析者に送信する．最後に解析者は秘密鍵を用いて暗号文を復号することで，データと回答者を結び付けることなく収集することができる．ここで，第三者機関はデータを提供する全ての回答者に信頼されていなければならない．しかしながら第三者機関を完全に信頼することは難しい．

近年，第三者機関を介さずに，多くの回答者から匿名でデータを収集する手法が文献 [6] で提案されている．文献 [6] のプロトコルでは，全ての参加者がプロトコルに従う semi-honest モデルが仮定されており，この制約が回答者の匿名性を保護するために不可欠な制約となっている．

本稿では，semi-honest モデルの制約を弱めた場合に実行可能となる攻撃に耐えるプロトコルを提案し，その匿名性を理論

的に検証し，実験的に性能を評価する．

2. データ収集問題

[問題の目的]

データ収集において，1 人のデータ解析者と N 人の回答者がいると考える．回答者が協力し解析者に各データの送信者を特定できないようにデータを送ることが目的である．ここで回答者の中から代表して匿名性を保護する役割を担うリーダを t 人選出しているとする．用いる暗号は ElGamal 暗号である．

[ElGamal 暗号]

ElGamal 暗号は鍵 $K = (q, g, x, y)$ を用いて，暗号化 $e_y(M, r)$ を行う公開鍵暗号手法である．ここで q は素数，g は原始元，x は秘密鍵，y は $g^x mod\, q$ より求められる公開鍵，M はメッセージ，r は乱数 $(0 \leqq r \leqq q-1)$ である．暗号 $e_y(M, r)$ では，二項組 $(C^{(1)}, C^{(2)})$ により，暗号文を構成する．ここで，

$$C^{(1)} = M \cdot y^r mod\, q,\ C^{(2)} = g^r mod\, q$$

とする．復号は，

$$D(e_y(M, r), x) = C^{(1)} \cdot (C^{(2)})^{-x} = M$$

により計算する．

3. 従来プロトコル

本節ではデータ収集問題に対する文献 [6] の従来プロトコルを紹介する．はじめに，プロトコルの敵モデルについて述べ，次に従来プロトコルで用いる鍵について述べる．そして，従来プロトコルについて述べる．

3.1 プロトコルの敵モデル

解析者と，悪意のある回答者は以下のような制約モデルに従うと始めに仮定する．

[semi-honest モデル]

プロトコルに参加する全ての機関は敵と協力する機関でさえ，正しくプロトコルに従うものとする．しかし，敵はすべての協力する機関の受信したすべてのメッセージの記録を含む内部ステートを得て解析することができる． □

3.2 プロトコルに用いる鍵

文献 [1] のプロトコルは回答者がデータを公開鍵を用いて暗号化して解析者に送信し，解析者はそのデータをリーダに送り，各リーダがデータを並び替え暗号化する手法をとっている．このとき，プロトコルに用いる ElGamal 暗号の鍵として解析者の鍵や各回答者の鍵を用いることはできない．解析者の鍵を用いた場合，初めにデータを送信したときに復号され回答者とデータとの対応が漏洩してしまう．各回答者が自分の公開鍵で暗号化した場合，リーダが並び替えた後の暗号文の列から自分のデータを復号するときに自分のデータしか復号できないため，復号する際に回答者とデータとの対応が漏洩してしまう．そのため，文献 [1] のプロトコルでは以下のような鍵を用いている．

暗号化に用いる鍵は，すべてのリーダの公開鍵の積

$$Y = \prod_{i=1}^{t} y_i \ mod\, q$$

を用いる．鍵 $K = (q, g, X, Y)$ とするとその秘密鍵 X は

$$\begin{aligned} Y &= y_1 \cdot ... \cdot y_t \ mod\, q \\ &= g^{x_1} \cdot ... \cdot g^{x_t} \ mod\, q \\ &= g^{x_1 + ... + x_t} \ mod\, q \end{aligned}$$

より，

$$\begin{aligned} X &= x_1 + ... + x_t \\ &= \sum_{i=1}^{t} x_i \end{aligned}$$

となり，各リーダの秘密鍵の総和となる．したがって鍵 K で暗号化された $e_K(M, r)$ は 全てのリーダの秘密鍵を利用しなければ，復号することができないという性質を持つ．

3.3 従来プロトコル

文献 [1] で紹介されているプロトコルを以下に示す．

- Phase1：データ提出
  $i = 1, ..., N$ において，回答者 i は, 公開鍵 Y を使って自分のデータ d_iを暗号化する．値 $r_i (0 \leqq r_i \leqq q-1)$ がランダムに一様に選択される場合，

  $$\begin{aligned} e_Y(d_i, r_i) &= (C_i^{(1)}, C_i^{(2)}) \\ &= (d_i \cdot Y^{r_i} \ mod\, q, g^{r_i} \ mod\, q) \end{aligned}$$

  となる．その後，回答者 i は $e_Y(d_i, r_i)$ の出力を暗号文 $C_i = (C_i^{(1)}, C_i^{(2)})$ として解析者に送る．

- Phase2：匿名化
  $i = 1, ..., t$ において, 解析者とリーダ i は以下の (a) から (c) を t ラウンド繰り返す．
  (a) 解析者は暗号文の列 $(C_1, ..., C_N)$ をリーダ i に送る．
  (b) リーダ i はデータを並び替え，再度暗号化する以下の計算を行う．
  ラウンド i において，π_i を $1, ..., N$ に関するランダムな置換とする．各 $j = 1, ..., N$ において, $\pi_i(j)$ はランダムな並び替えの後に j 番目となった暗号文の元の暗号文列での順番を示し，各値 δ_j が $[0, q-1]$ から独立，かつ, 一様に選択されるとするとき，

  $$\begin{aligned} R_j &= (R_j^{(1)}, R_j^{(1)}) \\ &= (C_{\pi_i(j)}^{(1)} \cdot Y^{\delta_{\pi_i(j)}} \ mod\, q, C_{\pi_i(j)}^{(2)} \cdot g^{\delta_{\pi_i(j)}} \ mod\, q) \end{aligned}$$

  となる再暗号文を構成する．
  (c) リーダ i は $j = 1, ..., N$ において，再暗号文 R_jを暗号文 C_jとし，暗号文の列 (C_1 ,..., C_N) を解析者へ送り返す．

- Phase 3：復号
  (a) 解析者は Phase2 の最後に受け取った暗号文列の第二項のみからなる列 ($C_1^{(2)}$,..., $C_N^{(2)}$) を全てのリーダに送る．各リーダ i は $j = 1, ..., N$ において，

部分的な復号情報

$$p_{j,i} = (C_j^{(2)})^{q-1-x_i} \ mod\ q$$

を計算する. その後，各リーダ i は部分的な復号情報の列 ($p_{1,i}$,..., $p_{N,i}$) を解析者へ送る.

(b) 全てのリーダから部分的な復号情報を受け取った後，解析者は $j = 1, ..., N$ において，それら復号情報を統合し，

$$d'_j = C_j^{(1)} \cdot \prod_{i=1}^{t} p_{j,i} \ mod\ q$$

を計算し，復号する．なお，文献 [1] では

$$p_{j,i} = (C_j^{(2)})^{x_i} \ mod\ q$$

$$d'_j = C_j^{(1)} / \prod_{i=1}^{t} p_{j,i} \ mod\ q$$

となっているが，計算の都合上フェルマーの小定理 [2] により，

$$p_{j,i} = (C_j^{(2)})^{q-1-x_i} \ mod\ q$$

$$d'_j = C_j^{(1)} \cdot \prod_{i=1}^{t} p_{j,i} \ mod\ q$$

とした.

3.4 従来プロトコルの機密性

本節では，文献 [6] に概略しか記載されていない従来プロトコルのシミュレート可能性について説明する．ここでシミュレートとはプロトコルを介して知ることを許されている情報のみを用いて，解析者が一人で，プロトコルのやり取りを偽造することを表す．偽造したやり取りが本物のやり取りと区別することが困難であるとき，解析者は第三者に実行したやり取りが本物であることを証明することができない．ゼロ知識証明 [3] の概念に基づき，プロトコルの中でリーダや回答者の機密情報が漏れていないことを検証する．以下に偽造したやり取りと本物のやり取りが区別することが困難であることを説明する.

解析者はプロトコル終了時に知ることを許されたデータ $(d_1, ..., d_N)$，及び公開鍵 Y などの公開情報を使えるものとする.

Phase1

本物のやり取りでは，回答者がデータを公開鍵 Y を用いて暗号化し，暗号文を解析者に送信している．ここで，解析者はデータ $(d_1, ..., d_N)$，及び公開鍵 Y を用いることができるため，シミュレートできる.

Phase2

本物のやり取りでは，各リーダはデータを匿名化するために，暗号文の列 $(C_1, ..., C_N)$ を並び替え，公開鍵 Y を用いて再度暗号化している．Phase1 と同様に，公開鍵 Y を用いることができるため，シミュレートできる.

Phase3

本物のやり取りは，以下の二つである.

(a) 解析者は各暗号文の第二項 $C_j^{(2)} (1 \leqq j \leqq N)$ を各リーダ $i (1 \leqq i \leqq t)$ へ送信し，各リーダ i は自分の秘密鍵 x_i を用いて部分的に復号する.

(b) 解析者は部分的な復号情報を元に暗号文を復号する.

(a) について，解析者は各リーダ i の秘密鍵 x_i を知らないので i の値により以下のやり取りを行う.

(1)$1 \leqq i \leqq t-1$ のとき

各リーダ i の秘密鍵 x_i の代わりに解析者は乱数 $X_{r_i} (1 \leqq X_{r_i} \leqq q-1)$ を用いてあたかも秘密鍵として部分的な復号情報 $p_{j,i} = (C_j^2)^{X_r}$ を計算する.

(2)$i = t$ のとき

解析者は以下の計算を行い，部分的な復号情報 $p_{j,t}$ を計算する.

$$p_{j,t} = d_k / C_i^{(1)} \prod_{i=1}^{t-1} p_{j,i}$$

(b) では解析者は $j = 1, ..., N$ において，復号情報を統合しているが，全て解析者が知ることが許されている値であるので，シミュレートできる.

上記のシミュレーションによって作成されたプロトコルのやり取りの記録と本物のやり取りの記録が区別がつくかどうかを考える．Phase1，Phase2，Phase3(b) に関しては，本物と同じ動作をシミュレーションでも行っているため，上記のやり取りの記録では本物とシミュレーションの区別はつかない．Phase3(a) でのやり取りのみがシミュレーションと本物を識別できるかどうかを左右するカギとなる.

(1) では，乱数 X_{r_i} をあたかも秘密鍵 x_i として復号しているため，区別することが困難であることは明らかである．よって，(2) で本物のやり取りと偽造のやり取りを区別することが困難であるならば，プロトコルの機密性は保証できる.

(2) では，$p_{j,t} = d_k / C_i^{(1)} \prod_{i=1}^{t-1} p_{j,i}$ を計算し，帳尻を合わしている．そのため，この $p_{j,t}$ を用いて復号した場合の平文 Z と本物のやり取りの場合の $p_{j,t}$ で得られる平文 m を区別することができるかどうかが問題となる．つまり，図 1 のような判定

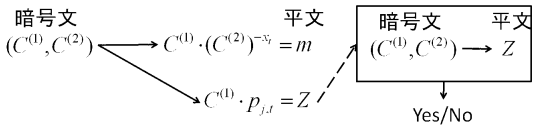

図 1 Z と m を区別する判定問題

問題の難しさとプロトコルのシミュレートを見破る難しさは同じである.

この判定問題を効率的に解くことが可能であるとする．ここで，暗号文の第一項 $C^{(1)}$ を平文 Z で割るとする.

$$C^{(1)} = Z \cdot y^r \ mod\ q$$
$$= Z \cdot (g^x)^r \ mod\ q$$
$$= Z \cdot g^{rx} \ mod\ q$$
$$C^{(1)}/Z = g^{rx} \ mod\ q$$

となる．したがって，これは $(q, g, g^x, g^r(= C^{(2)}), g^c(= C^{(1)}/Z))$ が与えられ，$g^c = g^{rx}$ かどうかという DDH 問題 [5] と等価である．しかしながら，DDH 問題を効率的に解くアルゴリズムは現在知られておらず，DDH 問題を効率的に解くアルゴリズムが存在しないという仮定が機密性の基準として用いられている．よってこの判定問題を効率的に解くことは難しいと予測される．

以上より，プロトコルの機密性は DDH 仮定が成り立つ限り保証できる．

3.5 従来プロトコルの脆弱性

本節では semi-honest モデルの制約を弱めた時に文献 [1] のプロトコルに対して実行可能な攻撃について考える．始めに解析者と，悪意のある回答者は以下のような制約モデルに従うと仮定する．

[弱い semi-honest モデル]

敵と協力する機関は互いにデータを交換しあたかも自分のデータとして用いることはできるが，それ以外は正しくプロトコルに従うものとする．しかし，敵はすべての協力する機関の受信したすべてのメッセージの記録を含む内部ステートを得て解析することができる．上記のモデルの場合，semi-honest モデルと異なり，解析者と悪意のある回答者は互いにデータを交換し，あたかも自分のデータとして用いることが可能となる．

□

弱い semi-honest モデルにおいて，文献 [1] のプロトコルに対して実行可能な攻撃を以下に述べる．プロトコルの最後に匿名化を行うリーダ t が悪意のあるリーダであると仮定する．もし，解析者がこのリーダに本来送る暗号データ

$$C_i : e_Y(M_j, r_j + \delta_1 + ... + \delta_{t-1})$$

(ここで各$\delta_i (i = 1, ...t-1)$ は各リーダが用いた乱数, r_jは回答者 j が用いた乱数とする.)
とは別に，Phase1 で各回答者から受け取った暗号データ

$$C_i : e_Y(M_i, r_i)$$

を秘密裏に送ったならば，リーダ t は前者の暗号データを使わず，後者の暗号データを用いて並び替え再度暗号化することが可能である．プロトコルに従って送られてきた暗号データを並び替えて再度暗号化した場合は

$$C_i : e_Y(M_{\pi(j)}, r_{\pi(j)} + \delta_1 + ... + \delta_{t-1} + \delta_t)$$

となる．ここで，$\pi(i)$ は $1, ..., N$ に関するランダムな置換である．一方，Phase1 で各回答者から受け取ったデータを並び替えずに再度暗号化した場合は

$$C_i : e_Y(M_i, r_i + \delta'_t) \quad (1)$$

となる．ここで，$Y^{r_{\pi(j)}+\delta_1+...+\delta_{t-1}+\delta_t} mod\, q$ と $Y^{r_i+\delta'_t} mod\, q$ が区別可能かどうか考える．r_i，$r_{\pi(j)} (i, j = 1, ..., N)$，$\delta_k (k = 1, ..., t)$ がとる値域は 0 から $q-1$ である．また，フェルマーの小定理より $Y^{q-1} mod\, q = 1$ である．従って，$Y^{r_{\pi(j)}+\delta_1+...+\delta_{t-1}+\delta_t} mod\, q$ と等しくなるような δ'_tの値が必ず存在する．よって，$Y^{r_{\pi(j)}+\delta_1+...+\delta_{t-1}+\delta_t} mod\, q$ と $Y^{r_i+\delta'_t} mod\, q$ は区別できないといえる．(1) のデータを解析者が Phase3 の始めに暗号文の列 $(C_1, ..., C_N)$ として受け取ったとする．そのとき，各暗号文 C_i を二項組 $(C_i^{(1)}, C_i^{(2)})$ とすると，

$$C_i^{(1)} = M_i \cdot Y^{r_i+\delta'_t} mod\, q$$
$$C_i^{(2)} = g^{r_i+\delta'_t} mod\, q$$

となる．各暗号文の第二項 $C_i^{(2)}$ を全てのリーダに送ると，$k = 1, ..., t$ において，各リーダ k から部分的な復号情報の列 $(C_i^{(2)})^{q-1-x_k}$ が返送されてくる．その後，それらの復号情報を統合し復号すると，

$$C_i^{(1)} \cdot \prod_{k=1}^{t} (C_i^{(2)})^{q-1-x_k}$$
$$= M_i \cdot Y^{r_i+\delta'_t} \cdot (g^{r_i+\delta'_t})^{\sum_{k=1}^{N} q-1-x_k}$$
$$= M_i$$

となり，リーダ t が受け取ったそのままの順のデータを得ることができる．従って，解析者は回答者と復号したデータを結び付けることが可能となる．次節にこの攻撃に耐えることのできるプロトコルを提案する．

4. 提案プロトコル

従来プロトコルでは，プロトコルに従わず，最後のリーダが悪意のあるリーダで，本来使わなければならないデータを使わず解析者が回答者と対応付けできるデータを使ってリーダが再度暗号化することができる．この場合, 乱数の項しか違わず区別をつけることができないことにより，後者のデータを復号でき，回答者とデータを結び付けできることが問題だと考えられる．その要因として各リーダは匿名化を行ったデータが最終的に復号するデータであることを保証できないことが挙げられる．

この問題を解決するために，我々は Phase2 と Phase3 で別々に行われる匿名化と復号を交互に行う．Phase2 で回答者のデータの暗号文の列を受け取ったデータをリーダはまず自分の秘密鍵を用いて部分的に復号し，その後並び替え再度暗号化することを考える．この手順の場合，別のデータへ差し替える等の行為があったとしても復号に必要な秘密鍵を用いた情報を交換後の暗号文が持たなくなるため，最終的に復号する際正しい値に復号できないと考えられる．

4.1 暗号化に用いる鍵

各リーダ j $(1 \leq j \leq t)$ において，

$$Y_{j \cdots t} = \prod_{i=j}^{t} y_i$$
$$X_{j \cdots t} = \sum_{i=j}^{t} x_i$$

とする．リーダ j が自分の秘密鍵 x_j で復号するとき，

$$\begin{aligned} D(e_{Y_j\cdots t}(M,r),x_j) &= C^{(1)}\cdot(C^{(2)})^{-x_j} \\ &= M\cdot Y_{j\cdots t}^r/(g^r)^{x_j} \\ &= M\cdot Y_{j+1\cdots t}^r\cdot y_j^r/(g^{x_j})^r \\ &= M\cdot Y_{j+1\cdots t}^r \end{aligned}$$

であることを利用して，

$$\begin{aligned} (C^{(1)},C^{(2)}) &= (D(e_{Y_j\cdots t}(M,r),x_j),C^{(2)}) \\ &= e_{Y_{j+1\cdots t}}(M,r) \end{aligned}$$

なる j の鍵を必要とせずに復号できるメッセージを作成する．最後のリーダがデータを知ってしまう問題が生じるため，回答者がデータを暗号化する際に用いる公開鍵は各リーダの公開鍵 $y_j(j=1,...,t)$ と解析者の公開鍵 y_m の積 $Y_{1\cdots t,m}$ とする．最後のリーダが再度暗号化するときに用いる鍵は解析者の公開鍵 y_m となる．

4.2 提案プロトコル

提案プロトコルを以下に示す．

- Phase1：データ提出
  $i=1,...,N$ において，回答者 i は，各リーダの公開鍵と解析者の公開鍵の積である公開鍵 $Y_{1\cdots t,m}$ を使って自分のデータを暗号化する．値 $r_i(0\leqq r_i\leqq q-1)$ がランダムに一様に選択される場合，

  $$\begin{aligned} e_{Y_{1\cdots t,m}}(d_i,r_i) &= (C_i^{(1)},C_i^{(2)}) \\ &= (d_i\cdot Y_{1\cdots t,m}^{r_i}\ mod\ q, g^{r_i}\ mod\ q) \end{aligned}$$

  となる．その後，回答者 i は $e_{Y_{1\cdots t,m}}(d_i,r_i)$ の出力を暗号文 $C_i=(C_i^{(1)},C_i^{(2)})$ として解析者に送る．

- Phase2：匿名化と部分的復号
  $i=1,...,t$ において，解析者とリーダ i は以下の (a) から (d) を t ラウンド繰り返す．
  (a) 解析者は暗号文の列 $(C_1,...,C_N)$ をリーダ i に送る．
  (b) $j=1,...,N$ において，リーダ i は自分の秘密鍵 x_i を用いて部分的復号

  $$C_j^{(1)}=C_j^{(1)}\cdot(C_j^{(2)})^{q-1-x_i}\ mod\ q$$

  を行う．このとき，C_j は $Y_{j+1\cdots t,\ m}$ で暗号化されたものになる．
  (c) ラウンド i において，π_i を $1,...,N$ に関するランダムな置換とする．各 $j=1,...,N$ において，$\pi_i(j)$ はランダムな並び替えの後に j 番目となった暗号文の元の暗号文列での順番を示し，各値 δ_j が $[0,q-1]$ から独立，かつ，一様に選択されるとするとき，

  $$\begin{aligned} R_j &= (R_j^{(1)},R_j^{(1)}) \\ &= (C_{\pi_i(j)}^{(1)}\cdot Y_{j+1\cdots t,m}^{\delta_{\pi_i(j)}}\ mod\ q, C_{\pi_i(j)}^{(2)}\cdot g^{\delta_{\pi_i(j)}}\ mod\ q) \end{aligned}$$

  を計算する．
  (d) リーダ i は $j=1,...,N$ において，再暗号文 R_j を暗号文 C_j とし暗号文の列（C_1 ,..., C_N ）を解析者へ送り返す。

- Phase3：復号
  解析者は Phase2 の最後に受け取った暗号文の列
  (C_1 , ..., C_N) に対し，各 $i=1,...,N$ において自分の秘密鍵を用いて，

  $$d_i'=C_i^{(1)}\cdot(C_i^{(2)})^{q-1-x_m}\ mod\ q$$

  を計算し，復号する．

4.3 プロトコルの正当性

本節では，提案プロトコルの正当性について考える．初めにプロトコルの正当性の定義を述べ，その後プロトコルの正当性が満たされていることを証明する．

[定義 1]

全ての参加者がプロトコルに従うと仮定したとき，各回答者が送信するデータの集合と最終的に解析者が受け取るデータの集合が一致するとき，プロトコルの正当性が満たされているという．

[補題 1]

全ての回答者のデータの集合を $(d_1,...,d_N)$ としたとき，解析者が得る結果は $(d_1,...,d_N)$ の置換である．

[証明]

Phase1 の終わりに，解析者は $(d_1,...,d_N)$ の暗号文の列 $(C_1,...,C_N)$ を受け取っている．ここで，受け取った暗号文の一つに注目し，それを C_I とする．C_I は Phase1 で述べられているように公開鍵 $Y_{1\cdots t,m}$ を用いた暗号文 $e_{Y_{1\cdots t,m}}(d_I,r_I)$ である．

Phase2 では，C_I はまずリーダ i の秘密鍵 x_i で復号され，その後，並び替えを行い，まだ並び替えが終わっていないリーダと解析者の公開鍵で再度暗号化されている．例えば，一人目のリーダ 1 は，C_I をまずリーダ 1 の秘密鍵によって部分的に復号し，C_I' に変換する．この時，公開鍵 $Y_{j+1\cdots t,m}$ を用いて暗号化 $e_{Y_{j+1\cdots t,m}}(d_I,r_I)$ を行ったものに対応させることができる．C_I の二項組を $(C^{(1)},C^{(2)})$ ，C_I' の二項組を $(C'^{(1)},C'^{(2)})$ とすると

$$\begin{aligned} C'^{(1)} &= C^{(1)}\cdot(C^{(2)})^{q-1-x_1} \\ &= d_I\cdot Y_{j\cdots t,m}^{r_I}\cdot a_2^{q-1-x_1} \\ &= d_I\cdot(g^{x_1+,...,+x_t+x_m})^{r_I}\cdot(g^{r_I})^{q-1-x_1} \\ &= d_I\cdot(g^{x_2+,...,+x_t+x_m})^{r_I} \\ C'^{(2)} &= C^{(2)} \end{aligned}$$

であるので，秘密鍵 $X_{j+1\cdots t,m}$ と公開鍵 $Y_{j+1\cdots t,m}$ は

$$X_{j+1\cdots t,m}=\sum_{i=j}^{t}x_i+x_m$$

$$Y_{j+1\cdots t,m}=\prod_{i=j}^{t}y_i\cdot y_m$$

となる．これらの鍵はまだ行っていないリーダと解析者のものである．

次にデータを並び替え公開鍵 $Y_{j+1\cdots t,m}$ で再度暗号化する．

$$e_{Y_{j+1\cdots t,m}}(e_{Y_{j+1\cdots t,m}}(d_I, r_I), \delta_{\pi_1(I)})$$
$$= e_{Y_{j+1\cdots t,m}}(d_I, r_I + \delta_{\pi_1(I)})$$

となる．その後，このデータを解析者へ送り返す．残りのリーダも同様に行われる．

$Phase3$ では，解析者は $Phase2$ の終わりで鍵 $K" = (q, g, x_m, y_m)$ を用いた暗号文 $e_{y_m}(d_I, r_I + \sum_{i=1}^{t} \delta_{\pi_i(i)})$ を受け取っている．$e_{y_m}(d_I, r_I + \sum_{i=1}^{t} \delta_{\pi_i(i)})$ の二項組を $(C"^{(1)}, C"^{(2)})$ とし，解析者の秘密鍵 x_m を用いて復号すると，

$$C"^{(1)} \cdot (C"^{(2)})^{q-1-x_m}$$
$$= d_I \cdot y_m^{r_I + \sum_{i=1}^{t} \delta_{\pi_i(i)}} \cdot (g^{r_I + \sum_{i=1}^{t} \delta_{\pi_i(i)}})^{q-1-x_m}$$
$$= d_I \cdot (g^{x_m})^{r_I + \sum_{i=1}^{t} \delta_{\pi_i(i)}} \cdot (g^{r_I + \sum_{i=1}^{t} \delta_{\pi_i(i)}})^{q-1-x_m}$$
$$= d_I$$

となり d_I が得られることが分かる．同様な議論が任意の $I = 1, ..., N$ において成り立つことが分かる．また，d_I は各リーダによって順番を並び替えられているので，解析者が得る時の I の順番を I' とすると，

$$I' = \pi_t(\pi_{t-1}(...(\pi_1(I)...))$$

となる．従ってプロトコルの正当性は満たされている． □

5. 匿名性の保証

本節では提案プロトコルの匿名性を満たすことを保証できるかについて考える．はじめに，提案プロトコルの機密性について述べ，その後，提案プロトコルの匿名性を保証する．

5.1 提案プロトコルの機密性

本節では，提案プロトコルのシミュレート可能性について説明する．以下に偽造したやり取りと本物のやり取りを区別することが困難であることを証明する．

解析者はプロトコル終了時に知ることを許されたデータ $(d_1, ..., d_N)$，及び公開鍵 $Y_{j\cdots t,m}$ などの公開情報，自分の秘密鍵 x_m を使えるものとする．

Phase1

本物のやり取りでは，回答者がデータを公開鍵 $Y_{j\cdots t,m}$ を用いて暗号化し，暗号文を解析者に送信している．ここで，解析者はデータ $(d_1, ..., d_N)$，及び公開鍵 $Y_{j\cdots t,m}$ を用いることができるため，シミュレートできる．

Phase2

本物のやり取りでは，リーダ i は自分の秘密鍵 x_i を用いて部分的に復号し，その後公開鍵 $Y_{j+1\cdots t,m}$ でデータを暗号化している．つまり，通信路には公開鍵 $Y_{j+1\cdots t,m}$ を用いて暗号化したものが送信されている．よって，解析者は回答者のデータの列 $(d_1, ..., d_N)$ を並び替え，公開鍵 $Y_{j+1\cdots t,m}$ で暗号化することでシミュレートすることができる．

Phase3

本物のやり取りでは，解析者は自分の秘密鍵 x_i で復号している．従って，解析者は自分の秘密鍵 x_i を用いることができるため，シミュレートできる．

上記のシミュレーションによって作成されたプロトコルのやり取りの記録と本物のやり取りの記録が区別がつくかどうかを考える．Phase1，Phase3 に関しては，本物と同じ動作をシミュレーションでも行っているため，上記のやり取りの記録では本物とシミュレーションの区別はつかない．Phase2 でのやり取りのみがシミュレーションと本物を識別できるかどうかを左右する鍵となる．ここで，本物のやり取りで送信するデータは公開鍵 $Y_{j+1\cdots t,m}$ で再度暗号化されたものであり，偽造のやりとりも $Y_{j+1\cdots t,m}$ で暗号化されたものである．つまり，用いられている乱数の値が違うだけである．これは，3.5 節で述べた議論と同様に，本物のやり取りで送信するデータが偽造したやり取りで送信するデータと一致する乱数の値が必ず存在する．従って，プロトコルの機密性は保証できる．

5.2 匿名性を保証する証明

I を悪意のあるリーダの集合とし，$|I| = t - 1$ とする．すなわち，1 人を除いて他の全員は悪意のあるリーダとする．ここで，入力データを $(d_1, ..., d_N)$ としてプロトコルを行ったときの解析者と悪意のあるリーダから見たやり取りの記録を $view_{miner,I}(d_1, ..., d_N)$ とする．このとき，σ を任意の置換としたときに $(d_{\sigma(1)}, ..., d_{\sigma(N)})$ を入力データとしてプロトコルを行ったときの解析者と悪意のあるリーダから見たやり取りの記録を $view_{miner,I}(d_{\sigma(1)}, ..., d_{\sigma(N)})$ とする．ここでのプロトコルが匿名性を満たすとは，$view_{miner,I}(d_1, ..., d_N)$ と $view_{miner,I}(d_{\sigma(1)}, ..., d_{\sigma(N)})$ が計算上区別がつかないことを意味する．

背理法に従い，プロトコルが匿名性を保護しないと仮定する．このとき，$view_{miner,I}(d_1, ..., d_N)$ と $view_{miner,I}(d_{\sigma(1)}, ..., d_{\sigma(N)})$ を区別するアルゴリズム D が存在する．つまり，

$$Pr[D(view_{miner,I}(d_1, d_2, ..., d_N)) = 1]$$
$$- Pr[D(view_{miner,I}(d_{\sigma(1)}, ..., d_{\sigma(N)})) = 1)]$$

において差が出ることを意味する．ここで，アルゴリズム D は入力したやり取りに対して 1 か 0 を出力するアルゴリズムであり，$D(view_{miner,I}(d_1, ..., d_N))$ が 1 を出力した場合には，$D(view_{miner,I}(d_{\sigma(1)}, ..., d_{\sigma(N)}))$ は 0 を出力し，$D(view_{miner,I}(d_{\sigma(1)}, ..., d_{\sigma(N)}))$ が 1 を出力した場合には，$D(view_{miner,I}(d_1, ..., d_N))$ は 0 を出力する．また，Pr は確率を表す．ここで，σ が二つのデータ d_A と d_B $(A < B)$ のみ入れ替え $(\sigma(d_A) = d_B,\ \sigma(d_B) = d_A)$，それ以外は入れ替えないとする $(\sigma(d_i) = d_i,\ i = 1, ..., N(i \neq A, i \neq B))$.

次に，アルゴリズム D を用いて，d_A と d_B の ElGamal 暗号文を区別するアルゴリズム A を以下のように作成する．

（1） 与えられた d_A(あるいは d_B) の暗号文である入力暗号文 e において，$e \cdot e'$=($d_A \cdot d_B$ の暗号化) となる暗号文 e' を計算する．

（2） 暗号文 e' を再度暗号化し，それを e” とする．

（3） データの順番を $(d_1, ..., d_{A-1}, Dec(e), d_{A+1}, ..., d_{B-1}, Dec(e"), d_{B+1}, ..., d_N)$ として，$view_{miner,I}$ をシミュレートする．(d_B の暗号文を入力とした場合，$(d_1, ..., d_{A-1}, Dec(e"), d_{A+1}, ..., d_{B-1}, Dec(e), d_{B+1}, ..., d_N)$ をシミュレートする．) ここで，$Dec(e)$ は暗号文 e の復号を表すとする．

（4） $D(view_{miner,I})$ の出力を o_1 とする．

（5） データの順番を $(d_1, ..., d_{A-1}, Dec(e"), d_{A+1}, ..., d_{B-1}, \mathrm{Dec(e)}, d_{B+1}, ..., d_N)$ として，$view_{miner,I}$ をシミュレートする．(d_B の暗号文を入力とした場合，$(d_1, ..., d_{A-1}, Dec(e), d_{A+1}, ..., d_{B-1}, Dec(e"), d_{B+1}, ..., d_N)$ をシミュレートする．)

（6） $D(view_{miner,I})$ の出力を o_2 とする．

（7） o_1=1 かつ o_2=0 ならば，1 を出力．o_1=0 かつ o_2=1 ならば，0 を出力．それ以外は 0.5 とする．

ここで，便宜上

$$p_1 = Pr[D(view_{miner,I}(d_1, ..., d_{A-1}, Dec(e), d_{A+1}, ..., d_{B-1}, Dec(e"), d_{B+1}, ..., d_N)) = 1]$$
$$p_2 = Pr[D(view_{miner,I}(d_1, ..., d_{A-1}, Dec(e"), d_{A+1}, ..., d_{B-1}, Dec(e), d_{B+1}, ..., d_N)) = 1]$$

とする．d_A の暗号文を入力として与えたときアルゴリズム A が 1 を出力するのは，暗号文 e を復号した平文 $Dec(e)$ が d_A の箇所に該当するときのみである．従って，アルゴリズム A の o_1=1，o_2=0 のときのみアルゴリズム A は 1 を出力するので，その確率は，

$$Pr[A(d_A\text{の暗号文}) = 1] = p_1(1 - p_2) + p_1p_2/2 + (1 - p_1)(1 - p_2)/2$$

である．同様に，d_B の暗号文を入力として与えたときアルゴリズム A が 1 を出力するのは，暗号文 e を復号した平文 $Dec(e)$ が d_B の箇所に該当するときのみである．従って，アルゴリズム A が 1 を出力する確率は，

$$Pr[A(d_A\text{の暗号文}) = 1] = p_2(1 - p_1) + p_2p_1/2 + (1 - p_2)(1 - p_1)/2$$

である．上記二つの式の差をとると，

$$\begin{aligned} &Pr[A(d_A\text{の暗号文}) = 1] - Pr[A(d_A\text{の暗号文}) = 1] \\ &= p_1(1 - p_2) + p_1p_2/2 + (1 - p_1)(1 - p_2)/2 \\ &\quad - p_2(1 - p_1) + p_2p_1/2 + (1 - p_2)(1 - p_1)/2 \\ &= p_1 - p_2 \end{aligned}$$

と差がでることが分かる．しかしながら，これは 2.4 節で述べた 2 つの異なる平文の暗号文を計算上区別することは困難であるという ElGamal 暗号の識別不可能性に矛盾する．従って，プロトコルの匿名性は保証される． □

5.3 提案プロトコルに関する考察

従来プロトコルでは semi-honest モデルに従うと仮定していた．それによりデータ収集の匿名性を保証していたが，弱い semi-honest モデルに従うと仮定した場合に解析者と悪意のあるリーダがデータを交換し，あたかもそれを正規のプロトコル動作で受け取ったデータとして用いることによって，回答者とデータを結び付けられる問題が生じてしまっていた．その問題に対処するために，提案プロトコルにおいては別の Phase で行われていた匿名化と復号を交互に行い，匿名化の際に用いる鍵を変えることでデータを交換してそのデータを用いたときは正しく復号できないようにした．また，提案プロトコルの正当性と匿名性を保証することにより，データ収集において正しくデータを送信できることと回答者の匿名性について保証することができた．したがって，提案プロトコルは従来プロトコルの耐攻撃性に加え，上記の攻撃に対しての耐性を持ち，かつ正当性と匿名性を保証できることから本研究によりデータ収集プロトコルの耐攻撃性を向上できたといえる．

6. 実験結果

提案プロトコルと従来プロトコルでの各参加者の手続きを OpenSSL ライブラリ，C 言語を用いて計算機上に実現し，シミュレーション実験を行った．ここで，ElGamal 暗号で用いた素数 q は 1024bit である．実験に使用した計算機環境は，Intel(R) Core(TM)2 Duo 2.4GHz プロセッサで，メモリは 2GB である．ここで，ランダムな並び替えのアルゴリズムとしては Josephus 選出 [4] を用いた．

提案プロトコルと従来プロトコルの処理時間を表 5.1，表 5.2 に示す．本実験では，各プロトコルの通信手続きを除き逐次的に処理を行ったときの計算時間を計測した．したがって，表での総処理時間の項の逐次実行時間とは全てのリーダの作業を連続で行ったときにかかる時間である（つまり，並列に行える作業を並列にしないときの時間である）．実際には，Phase1 の回答者がデータを提供する作業は並列に実行可能である．加えて，従来プロトコルでは Phase3 の各リーダが部分的に復号する作業も並列に実行可能である．並列に実行可能な処理を実際に並列に実行したと仮定した時の計算上の処理時間を並列処理時間とする．例えば，従来プロトコルの回答者数 20 人，リーダ数 10 人のときの並列処理時間は

$$\begin{aligned} (\text{並列処理時間}) =& (\text{逐次実行時間}) \\ & - (\text{回答者一人あたり}) \times (\text{回答者数} - 1) \\ & - (\text{各リーダの復号時間}) \times (\text{リーダ数} - 1) \\ =& 3.40 - 0.011 \times 19 - 0.11 \times 9 \\ =& 2.20 \end{aligned}$$

のように計算する．

表 5.1　提案プロトコルの処理時間 (通信時間を除く)

		総処理時間 [s]		リーダ一人あたり [s]		
回答者数	リーダ数	逐次実行	並列処理	匿名化	復号	合計
20	10	3.51	3.30	0.21	0.11	0.32
50	10	8.84	8.30	0.54	0.26	0.80
50	20	16.72	16.17	0.54	0.26	0.80
100	10	17.54	16.45	1.07	0.53	1.60
100	20	33.58	32.48	1.07	0.53	1.60
100	40	65.47	64.38	1.07	0.53	1.60
100	50	81.81	80.72	1.07	0.53	1.60
回答者一人あたりの処理時間 [s]				0.011		

表 5.2　従来プロトコルの処理時間 (通信時間を除く)

		総処理時間 [s]		リーダ一人あたり [s]		
回答者数	リーダ数	逐次実行	並列処理	匿名化	復号	合計
20	10	3.40	2.20	0.21	0.11	0.32
50	10	8.46	5.55	0.54	0.26	0.80
50	20	16.53	10.97	0.54	0.26	0.80
100	10	17.05	11.25	1.07	0.53	1.60
100	20	32.98	21.94	1.07	0.53	1.60
100	40	65.03	43.49	1.07	0.53	1.60
100	50	81.18	54.29	1.07	0.53	1.60
回答者一人あたりの処理時間 [s]				0.011		

表 5.3　各参加者の暗号化および復号の回数

	提案プロトコル		従来プロトコル	
	暗号化回数	復号回数	暗号化回数	復号回数
回答者	1	0	1	0
リーダ	N	N	N	N
解析者	0	N	0	0

表 5.1，表 5.2 で総処理時間で差が出たのは，表 5.3 より提案プロトコルは解析者も自分の秘密鍵を用いて復号を行うため，従来プロトコルより N 回多く復号を行っているためである．また，提案プロトコルに比べ，従来プロトコルでは Phase3 のリーダの部分的復号の作業が並列に行えることから，並列に行った際には $(t-1)N$ 回の復号時間ほど差が生じる．従って，従来プロトコルの $Phase3$ のリーダの部分的復号の作業を並列に行う際には提案プロトコルは tN 回の復号時間分が従来プロトコルに比べて余分にかかることが分かる．

一方，データ通信量について考える．データ伝送時のデータのビット幅を B とした場合のデータ通信量を表 5.4 に示す．

表 5.4　データ通信量

	提案プロトコル [bit]	従来プロトコル [bit]
Phase1	$2NB$	$2NB$
Phase2	$4tNB$	$4tNB$
Phase3	-	$2tNB$
合計	$N(4t+2)B$	$N(6t+2)B$

表 5.4 より，提案プロトコルのほうが従来プロトコルに比べ，$2tNB$ ビット分通信量が少なくなることが分かる．よって，t, N の値や通信速度によっては，提案プロトコルの方が実行時間が短くなる可能性があると考えられる．

7. おわりに

本稿では，データ収集においてプライバシーを保護する従来プロトコルに注目した．その従来プロトコルで定義されている semi-honest モデルより弱い制約モデルを定義したときに，従来プロトコルにどのような攻撃を行うことでプライバシーを保護できない問題が生じるかについて考察した．さらに，従来プロトコルに対して行うことのできる考察した攻撃に耐えることのできるプロトコルを提案した．加えて，提案プロトコルの正当性と匿名性を証明することで，プライバシーを保護しつつデータ収集が行えることを示し，プロトコルの耐攻撃性が向上していることを確認した．

今回は従来プロトコルで用いられている ElGamal 暗号を提案プロトコルでも用いることにしたが，プロトコルの構成が変化したことにより，別の暗号手法でも構成可能である可能性がある．よって，今後の課題として，提案プロトコルに必要な最小限の制約を考察することで，プロトコルに用いることができる ElGamal 暗号以外の他の暗号手法を探し，効率的なプロトコルを構築することが挙げられる．

文　献

[1] 黒沢，尾形，現代暗号の基礎数理，コロナ社, 2004.

[2] 岡本，太田，暗号・ゼロ知識証明・数論，共立出版, 1995.

[3] B. Schneier, Applied Cryptography, Second Edition, John Wiley&Sons, Inc., 1996.

[4] R. Sedgewick, Algorithms in C++, Third Edition, Addison - Wesley, 1998.

[5] D. R. Stinson, Cryptography Theory and Practice, Third Edition, Chapman&Hall/CRC, 2006.

[6] Z. Yang, S. Zhong and R. Wright, “Anonymity-preserving data collection,” Proc. of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining (KDD’05), pp. 334-343, 2005.