秘匿ログ分析におけるデータの秘匿度合と分析効率とのトレードオフ評価

萩尾 玲太[1]　石原 靖哲[1]　矢内 直人[1]　唐崎 正史[2]　江口 勝彦[2]　藤原 融[1]

概要：不正ログインを目的とする攻撃を検知するための通信ログ分析問題を，ログの所有者，管理者，分析者が相異なる三者モデルのもとで考える．このモデルでは，所有者が，個人情報を秘匿したログを管理者に委託し，分析者がそのログに対して分析を行う．このような秘匿ログ分析は，分析に必要な個人情報を管理者に秘匿した状態で分析者に提供されれば実現できるが，管理者にもその情報の一部を開示することにより，分析者の計算量を軽減できる可能性がある．本稿では，異なるログレコード間における個人情報の等価性を用いる分析において，管理者への等価性情報の開示量と分析者の計算量とのトレードオフ評価を行う．

キーワード：秘匿ログ分析，トレードオフ評価，三者モデル，等価性情報

Trade-off Evaluation of Privacy-preserving Degree and Efficiency in Privacy-preserving Log Analysis

Ryota Hagio[1]　Yasunori Ishihara[1]　Naoto Yanai[1]　Masashi Karasaki[2]　Katsuhiko Eguchi[2]
Toru Fujiwara[1]

Abstract: This paper discusses communication log analysis problem to detect attacks aiming at unauthorized login under the three-party model, which consists of distinct three entities, namely, log owner, log administrator, and log analyst. In this model, the owner consigns privacy-preserved logs to the administrator, and the analyst analyzes the logs. Such privacy-preserving log analysis can be realized if the personal information necessary for analysis is provided to the analyst but kept secret from the administrator. However, by disclosing a part of the personal information to the administrator, the computational cost at the analyst can be reduced. In this paper, we focus on analysis where equivalence of personal information between different log records is used. Then, we evaluate the trade-off between the amount of equivalence information disclosed to the administrator and computational cost at the analyst.

Keywords: Privacy-preserving Log Analysis, Trade-off Evaluation, Three-Party Model, Equivalence Information

1. はじめに

1.1 研究背景

昨今，サイバー攻撃の著しい増加，高度化に伴い，サービスの死活監視を実施するだけではサービスの安定的な運用は実現が難しくなりつつある．このような不正アクセスやなりすましなどの高度なサーバ攻撃を検出するためには，サービスプロバイダやオペレーションセンタがネット

[1] 大阪大学大学院情報科学研究科
Graduate School of Information Science and Technology, Osaka University, 1-5 Yamadaoka, Suita, Osaka, 565-0871, Japan
[2] 株式会社 NTT ネオメイト
NTT Neomeit Corporation, 2-2-5 Uchihonmachi, chuo-ku, Osaka, Osaka, 540-0026, Japan

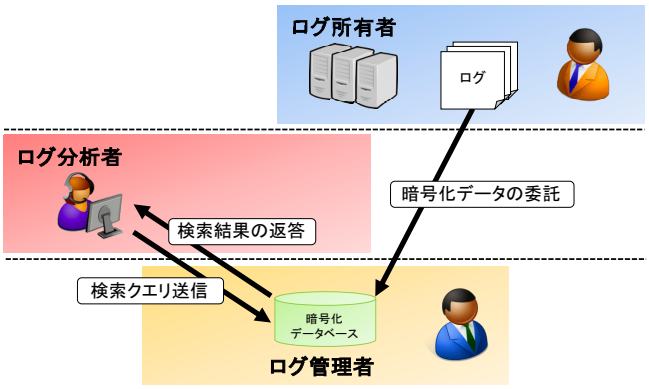

図 1 三者モデル

ワークを監視し，個人情報を含むログを収集し分析することが重要となっている．ここで個人情報とは，特定の個人を識別できるものをいい，一般にネットワークログ情報には，ユーザ ID や接続元 IP アドレスなどの個人情報が含まれている．これらは個人情報保護の観点から鑑みると，厳重に扱う必要があるため，暗号化して管理されている．またログを分析する側の観点からも，不用意に個人を特定し得る情報を知りたくないという心理的な問題もある．一方で障害や攻撃などが発生した際は，その対応に向け，ログを適切かつ迅速に分析できることが望ましい．

また，クラウドコンピューティングの普及により，ログの所有者が個人情報を含むログを暗号化してクラウド上の暗号化データベースサービスなどに管理を委託し，その暗号化データベースの分析を他事業部が行うといった状況が増加してきている．このような暗号化データベースの利用形態は，ログの平文を持つログ所有者，暗号化データベースを管理するログ管理者，及び暗号化データベースを用いて分析を行うログ分析者の三者でモデル化される（**図 1**）．このような三者モデルでは，ログ分析者に関しては，平文を閲覧することなく委託されたログ分析に必要な情報（等価性情報など）だけを得られるようにすることを要件として定義している．また，ログ管理者が得られる情報は，ログ分析者が得られる情報の一部でなければならず，かつ，その情報から分析を正しく行えてはいけないことを要件として与える．

秘匿ログ分析は，ログ分析に必要な個人情報を管理者に秘匿した状態で分析者に提供されれば実現できる．しかし，ログ管理者にもその情報の一部を開示することにより，ログ分析者の計算量を軽減でき，効率的なログ分析を行うことができる可能性がある．

1.2 研究概要

本稿では，異なるログレコード間における個人情報（ユーザ ID や IP アドレスなど）の等価性を用いるログ分析において，ログ管理者への個人情報の等価性情報の開示量とログ分析者の計算量とのトレードオフ評価を行い，その評価・考察を行う．具体的には，前節で述べた三者モデルをもとに，ログ所有者は個人情報を含むログデータベースに対して個人情報の等価性情報を一部だけ開示する方式を施す．ログ分析者は，その等価性情報の開示量に従って，既存のデータベース分析ツールを用いたログの分析を行う．このような問題設定のもとで，等価性情報の開示量による計算量の変化を評価する．

1.3 本稿の構成

本稿の構成は次の通りである．2 節では関連研究を紹介し，3 節では本研究におけるモデルや要件等を定義する．4 節ではトレードオフ評価の対象方式を提案し，5 節でトレードオフ評価を行う．6 節で本稿を纏める．

2. 関連研究

2.1 検索可能暗号

検索可能暗号とは，暗号文生成時に，平文中に含まれるキーワードに関するインデックスをあらかじめ生成しておくことで，暗号化した状態でキーワードの検索を可能にした暗号化方式である [1] [2] [3] [4] [5] [6] [7]．検索可能暗号は，1.2 節のように暗号化データベースの分析を他事業部に委託している場合にも，検索クエリを暗号化しておくことでデータベース管理者に情報を漏らすことなく分析が可能となる．またクラウド上の暗号化データを多数のユーザ間で安全に利用できる暗号化方式も提案されている [8] [9]．

上記の研究に加えて，検索可能暗号を用いた，個人情報を秘匿しつつ個人が持つ属性（年齢，性別，収入，資格，経歴など）を確認する属性認証方式がある [10]．この方式では，ユーザの個人情報を暗号化した状態でクラウドサービス事業者に管理を委託し，必要に応じてユーザの情報から該当サービスの利用資格を確認するようなモデルが定義されている．このモデルは，ユーザの個人情報をクラウドサーバに登録する個人情報管理者，個人情報管理者が保持する個人情報の管理を委託されたクラウドサービス事業者，ユーザが利用するサービスを提供し，利用資格を確認するサービス提供者の三者に分類されている．また検索可能暗号の研究では，公開鍵暗号をベースとする方式や共通鍵暗号をベースとする方式など様々な方式が提案されており，坂崎ら [10] はこれらの方式と再暗号化技術を組み合わせることで上記の秘匿属性認証方式を開発している．

さらに，検索可能暗号を用いた暗号化データベースに対する漏洩情報量が評価されている [11]．一般に検索可能暗号により暗号化されたデータは確率的に暗号化されるが，キーワードとの完全一致が判明した暗号化データは，安全性が決定性暗号と同程度に低下することが知られている．この文献では，検索により徐々に低下する暗号化データの安全性について，情報エントロピーの観点から評価した結果が報告されている．

2.2 データの等価性が判定可能な暗号化方式

データの等価性を判定が可能な暗号化方式として，Message Locked Encryption [12] (以下，MLE と呼ぶ) がある．MLE は，データの暗号化及び復号に用いる鍵を平文から生成する共通鍵暗号方式である．同じ平文であればその共通鍵も同じになるため，異なるユーザ間において暗号化されたデータの重複除外 (deduplication) に有効な方式とされている．主要なクラウドストレージサービスでは，ストレージ空間を有効活用するためにファイルレベルでの重複除外を行っている．しかし，ユーザが自身のファイルの中身をストレージのプロバイダに知られたくないため，暗号化して保存することを考えたとする．一般的な暗号技術では暗号化鍵がユーザ毎に異なるため，サーバが暗号文における平文の等価性を判定できない．しかし MLE を用いれば，同じファイルの暗号化鍵が同じになるためこの問題を解決でき，各ユーザは自身で生成した暗号化鍵でファイルの復号を行うことができる．一方，MLE の問題点は，データの等価性を判定するためのタグが確定的に生成されていることである．これにより，MLE を上記の三者モデルに適用することを考えた場合，ログ管理者によるデータの等価性がすべて判定可能となってしまう．

この問題を解決するために，Message Locked Encryption with Re-Encryption and Relational Search [13] (以下，MLE-RERS と呼ぶ) という技術を我々は提案している．MLE-RERS では，データの等価性を判定するために確定的に生成 (暗号化) されているタグを確率的な共通鍵暗号化方式で二重に暗号化することで，ログ管理者に平文に関する情報が漏れることを防いでいる．また，ログ分析者には二重暗号化から一段階復号する鍵をもたせておくことで，暗号文同士の関係の検索が可能となる．本稿では，MLE-RERS で二重暗号化されたタグを MLE-RERS タグ，ログ分析者がこのタグを一段階復号してデータの等価性が判定可能となるタグを MLE-EQ タグと呼ぶことにする．

3. 問題設定

3.1 三者モデル

本稿で想定する三者モデルは以下の 3 エンティティから構成される．

ログ所有者 ユーザ ID や IP アドレスなどの個人情報を含むログを所有し，ログ情報を暗号化してログ管理者に管理を委託する．

ログ管理者 ログ所有者から委託された暗号化ログデータベースを管理する．

ログ分析者 ログ管理者が管理している暗号化ログデータベースに対して，既存のデータベース分析ツールを用いて分析を行う．

表 1 対象ログデータ

項目名	内容
アクセス日時	該当ログが発生した日時
接続元 IP アドレス	該当ログにおいてアクセスのあった接続元の IP アドレス
ユーザ ID	該当ログにおいて入力されたユーザアカウント情報
認証判定	認証における成功可否
エラー事由	認証失敗の場合の原因
ユーザエージェント	接続元 IP アドレスにおいて利用されているクライアント情報

3.2 安全性要件

本節では，三者モデルにおけるログ管理者とログ分析者に関する安全性要件を定義する．まず，ログ分析者が得られる情報は，分析に必要な情報（本稿では異なるログレコードにおける個人情報の等価性情報）だけであり，特に平文そのものを知ることができてはいけない．また，ログ管理者が得られる情報は，ログ分析者が得られる情報の一部でなければならず，かつ，その情報から分析を正しく行えてはいけない．

3.3 対象ログデータ

本稿での対象ログデータは，ユーザ ID や IP アドレスなどの個人情報を含む認証ログである．対象ログデータの各レコードは**表 1** で示した項目で構成される．エラー事由に関しては，パスワード不一致と回線 ID 不一致，該当なしユーザ ID の 3 点の要素がある．また，個人情報保護の観点からユーザ ID と接続元 IP アドレスの 2 点を秘匿対象とする．評価実験で使用するログデータは，複数サービスで発生するログを約 10 日間収集した約 26 万レコードのものを用いる．

3.4 データベース分析ツール

本稿では，既存のデータベース分析ツール（以下，DB 分析ツールと呼ぶ）を用いて秘匿ログ分析を行う．具体的には，InfluxDB [14] [15] [16] [17] を採用する．InfluxDB はオープンソースの時系列データベースシステムの一つで，時系列データを格納するのに適しており，システムメトリクスの保存やアクセスログの集計や解析が可能となっている．さらに SQL (Structured Query Language) ライクな問合せ言語であり，時系列データに特化した様々な集約関数に対応している InfluxQL が使用できる．その中で “GROUP BY” という集約関数を用いることで，ユーザ ID や IP アドレスなどの特定の列をキーとした合計値や平均値などを集計することが可能となる．この GROUP BY を使用してデータベースに問い合わせることで，後述の検知対象セキュリティイベントのような，ユーザ ID や IP アドレスなどが等しいレコードグループ毎の認証失敗数を集計

```
SELECT COUNT(*)
INTO テーブル(event(a))
FROM テーブル
WHERE 認証判定 = 失敗
AND time >= 分析対象ログの最初の日時
AND time <= 分析対象ログの最後の日時
GROUP BY ユーザID, time(unit_time)

SELECT *
FROM テーブル(event(a))
WHERE 認証失敗数 >= 閾値
```

図 2 イベント (a) の発生を検知する InfluxQL 問合せ例

することができる．

3.5 検知対象セキュリティイベント

本節では，検知対象とするセキュリティイベントを以下の (a)–(g) で定義する．

(a) ID 毎の単位時間当たりの認証失敗数が閾値を超える
(b) ID 毎の単位時間当たりの認証失敗数の増加数が閾値を超える
(c) ID 毎の連続認証失敗数が閾値を超える
(d) 単位時間当たりの未登録 ID による認証失敗数が閾値を超える
(e) IP アドレス毎の単位時間当たりの異なる ID の数が閾値を超える
(f) IP アドレス毎の単位時間当たりの認証失敗数が閾値を超える
(g) IP アドレス毎の連続認証失敗数が閾値を超える

暗号化していないログデータベースに対して，イベント (a) の発生を問い合わせる InfluxQL 問合せの例を**図 2** に示す．ここで unit_time は単位時間を表す．イベント (d) 以外の 6 つのイベントについては，図 2 もしくはそれに類似した形式の問合せとなるため，本稿ではイベント (a) に焦点をあててトレードオフ評価を行う．なお，イベント (d) の発生を問い合わせる InfluxQL 問合せの例を**図 3** に示す．イベント (d) およびそれに類似した形式の問合せについてのトレードオフ評価は今後の課題とする．

4. トレードオフ評価対象方式

この評価対象方式では，2.2 節で定義した MLE-RERS タグに加えて新たに細分化タグと集約化タグという 2 種類のタグを付与する．以下で，細分化タグと集約化タグを定義する．A と B を個人情報 (ユーザ ID など) の平文とし，それぞれから生成したタグを T_A，T_B とする．細分化タグとは，$T_A = T_B$ ならば $A = B$ が成立するタグである．集約化タグとは，$T_A \neq T_B$ ならば $A \neq B$ が成立するタグである．

```
SELECT COUNT(*)
INTO テーブル(event(d))
FROM テーブル
WHERE エラー事由 = 該当なしユーザID
AND 認証判定 = 失敗
AND time >= 分析対象ログの最初の日時
AND time <= 分析対象ログの最後の日時
GROUP BY time(unit_time)

SELECT *
FROM テーブル(event(d))
WHERE 認証失敗数 >= 閾値
```

図 3 イベント (d) の発生を検知する InfluxQL 問合せ例

この方式を用いたログ分析のフローの概要を説明する．まず集約化タグをキーとして GROUP BY を行い，各グループの認証失敗数をカウントする．次に，このカウント数が閾値を超えたグループに対して細分化タグをキーとして GROUP BY を行い，各グループの認証失敗数をカウントする．そして，本来同じユーザ ID であるが異なるタグに暗号化されているものについて MLE-EQ タグを用いて本来のユーザ ID 毎に認証失敗数を合計する．最後にその数が閾値を超えたログレコードを検知結果として出力する．

4.1 節では，等価性情報を一部だけログ管理者に開示する方式として，細分化方式と集約化方式を示す．次に 4.2 節で上記の 3 種のタグを用いた評価対象方式におけるログ分析フローの詳細を示す．

4.1 等価性情報を一部だけ開示するタグの生成方式

本節では，ユーザ ID などの秘匿すべきデータの等価性情報を一部だけ開示する方式を提案する．具体的には，同一の個人情報から複数種類のタグを生成することで個人情報の等価性を「細分化」する方式，また逆に複数種類の個人情報から同一のタグを生成することで等価性を「集約化」する方式である．この 2 つの方式を組み合わせて用いることでログ管理者への等価性情報の開示量，すなわちデータの秘匿度合を調節する．

4.1.1 細分化方式

この方式では，あらかじめ x 種類（$x > 1$）の鍵を用意しておき，ログレコード毎に擬似ランダムに鍵を選択することで，細分化タグを生成する．同一の個人情報に対して生成されるタグの種類数を増加させることで，ログ管理者に対する等価性情報の開示量を減少させる．具体的には，使用鍵を決定するための鍵 k と x 個の鍵 $k_1, \ldots, k_x$ を準備し，n 番目のログレコードに現れる個人情報の平文 M に対して以下の式で細分化タグを生成する．

$$h(M||k_{h(n||k) \bmod x})$$

ただし，h は一方向性ハッシュ関数である．

4.1.2 集約化方式

この方式では，本来 z 種類存在する個人情報に対し，y 種類 $(y < z)$ のタグを確定的に生成することで，集約化タグを生成する．同一タグに対応する個人情報の種類数を増加させることで，ログ管理者に対する等価性情報の開示量を減少させる．具体的には，個人情報の平文 M に対して以下の式で集約化タグを生成する．

$$h(M) \bmod y$$

ただし，h は一方向性ハッシュ関数である．

4.2 3 種類のタグを用いたトレードオフ評価対象方式

本方式では，前節で生成した細分化タグと集約化タグ，また MLE-RERS タグの 3 種類のタグを各ログレコードに付与する．トレードオフ評価対象方式を用いたログ分析フローの詳細を以下に示す．また，ログ分析フローを説明するための例を**図 4**，**図 5** に，各フェーズでの InfluxQL 問合せを**図 6**，**図 7** に示す．本ログ分析フロー図では，閾値を 20 回に設定している．

1. 集約化タグでの GROUP BY フェーズ

集約化タグをキーにして GROUP BY を用いた検索クエリで問合せ（図 6）を行い，集約化タグ毎の単位時間当たりの認証失敗数を DB 分析ツールから取得する．図 4 では，認証失敗数が閾値 20 回を超えた#ABC と#JKL，#MNO の 3 つの集約化タグが取得できる．この問合せでは，本来異なるユーザ ID が同じタグとなっているため，この時点で認証失敗数の閾値を超えていないグループはフェーズ 2 での検知対象から除外することができる．

2. 細分化タグでの GROUP BY フェーズ

フェーズ 1 の問合せ結果のログデータに対して，細分化タグをキーにして GROUP BY で問合せ（図 7）を行い，細分化タグ毎の単位時間当たりの認証失敗数を DB 分析ツールから取得する．図 5 では，フェーズ 1 で認証失敗数が閾値を超えた#ABC が 9 種類の細分化タグをもち，各タグの認証失敗数を取得している．

3. MLE-RERS タグでの COUNT フェーズ

フェーズ 2 の問合せ結果のログデータは，本来同じユーザ ID が違うタグとなっている．ログ分析者は MLE-RERS タグを一段階復号した MLE-EQ タグを用いて，本来のユーザ ID 毎の認証失敗数を合計することで，最終的に失敗認証数が閾値を超えた本来のユーザ ID を取得する．図 5 では，本来の等価性情報を表す MLE-EQ タグを用いて，タグ A,B,C 毎の認証失敗数を計算することで，閾値を超える本来のユーザ ID を取得する．この例では，閾値 20 回を超える A のみがイベント (a) の検知結果として出力される．

このトレードオフ評価対象方式について，各ログレコードに細分化タグだけを付与する場合は，ログ分析者は本来同じユーザ ID 毎の認証失敗数を合計する必要がある．しかし，細分化タグ毎の認証失敗数を合計しても閾値を超えない場合があるため，分析にかかる計算量が増加してしまうことが懸念される．

また集約化タグだけを付与する場合は，問合せ結果で残ったログレコードに関して，本来同じユーザ ID の認証失敗数をログ分析者自身で計算しなければならなくなる．

以上より，各ログレコードに 3 種類のタグを付与することで，ログ分析者は認証失敗数の合計を計算するだけをするだけでよくなるため，分析効率の向上が期待される．

time	...	認証失敗数 (集約化)	集約化タグ	細分化タグ	MLE-RERS タグ
00:00:00	...	46	#ABC	-	-
00:00:00	...	15	#DEF	-	-
00:00:00	...	18	#GHI	-	-
00:00:00	...	37	#JKL	-	-
00:00:00	...	25	#MNO	-	-
00:00:00	...	12	#PQR	-	-

図 4 ログ分析フロー 1

図 5 ログ分析フロー 2

```
SELECT COUNT(*)
INTO テーブル2
FROM テーブル1
WHERE 認証判定 = 失敗
AND time >= 分析対象ログの最初の日時
AND time <= 分析対象ログの最後の日時
GROUP BY 集約化タグ,time(unit_time)

SELECT *
FROM テーブル2
WHERE 認証失敗数 >= 閾値
```

図 6 フェーズ 1 の InfluxQL 問合せ

```
SELECT COUNT(*)
INTO テーブル3
FROM テーブル1
WHERE 認証判定 = 失敗
AND 集約化タグ = 認証失敗数が閾値を超えた集約化タグ
AND time >= 分析対象ログの最初の日時
AND time <= 分析対象ログの最後の日時
GROUP BY 細分化タグ,time(unit_time)

SELECT *
FROM テーブル3
```

図 7 フェーズ 2 の InfluxQL 問合せ

表 2 実験環境

OS	OS X EI Capitan
CPU	2.2 GHz Intel Core i7
メモリ	16GB
言語	Python 2.7
コンパイラ	PyCrypto
InfluxDB	v1.3.2

5. トレードオフ評価に向けての予備実験

上記のトレードオフ評価対象方式に対し，ログデータの秘匿度合に応じて分析効率を評価し，このトレードオフ評価における考察を行う．ただし，本稿執筆時点ではフェーズ 3 の実装が完了していないため，本節ではフェーズ 2 までの予備的評価について述べる．

本方式の検知対象セキュリティイベントと対象ログデータの詳細を以下に示す．また，本評価の実験環境について**表 2** に示す．

検知対象セキュリティイベント 「(a) ユーザ ID 毎の 10 分間当たりの認証失敗数が 6 回を超える」．

対象ログデータ 約 26 万ログ（複数サービスで発生するログを約 10 日間収集したもの）．本評価では，上記のイベント (a) でトレードオフ評価を行うため，ユーザ ID のみを秘匿対象とする．

細分化タグ 3 種と集約化タグ 3 種を用いた計 9 パターンの組み合わせについて評価を行った．これらのパターンの詳細について**表 3** に示す．この表では，細分化のパラメータ x，集約化のパラメータ y の値とそれぞれの場合におけるログ全体の細分化タグ数及び集約化タグ数を示している．また，平文のユーザ ID の種類数は 148,732 となっている．

5.1 実験結果

それぞれの評価パターンに基づく分析時間を**図 8**，**図 9**，**図 10** に示す．各評価パターンにおける試行回数は 5 回とし，その平均を分析時間としている．

上述の結果より，集約化のパラメータ y を変化させるこ

表 3 評価パターン

	細分化		集約化	
	x	種類数	y	種類数
①	10	237,530	100	81
②			1,000	729
③			10,000	6,561
④	100	260,043	100	81
⑤			1,000	729
⑥			10,000	6,561
⑦	500	263,105	100	81
⑧			1,000	729
⑨			10,000	6,561

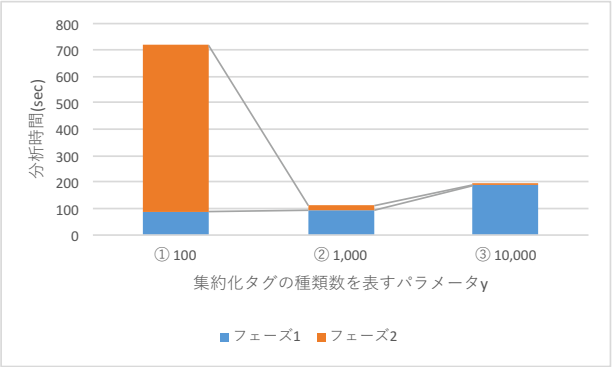

図 8 ①-③（細分化のパラメータ $x = 10$）におけるトレードオフ評価

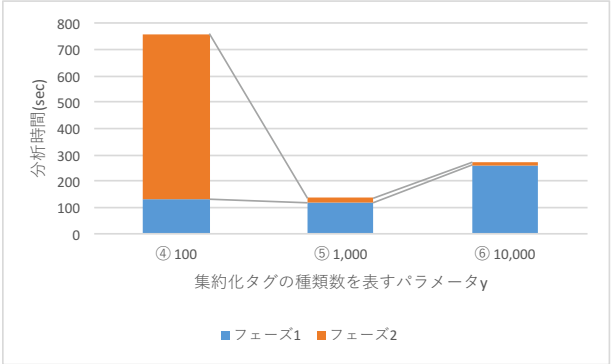

図 9 ④-⑥（細分化のパラメータ $x = 100$）におけるトレードオフ評価

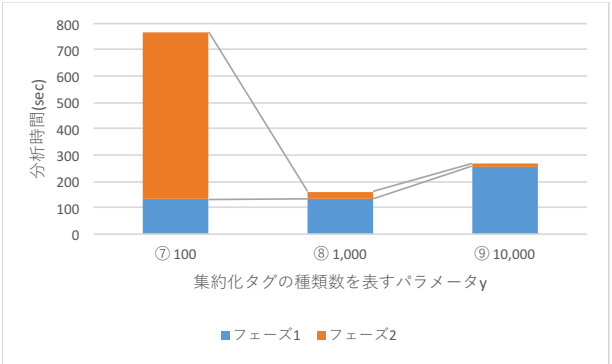

図 10 ⑦-⑨（細分化のパラメータ $x = 500$）におけるトレードオフ評価

とで，フェーズ 1，2 の両方の分析時間が変化しており，特にフェーズ 2 の変化率が著しい．フェーズ 1 では，パラメータ y を大きくすることで分析時間が約 10〜100 秒増加している．またフェーズ 2 では，パラメータ y を大きくすることで分析時間が減少しており，特に $y = 100$ から $y = 1,000$ の変化では，分析時間が約 600 秒減少している．

また細分化のパラメータ x を変化させることで，フェーズ 2 の分析時間が変化している．具体的に $x = 10$ から $x = 500$ に変化させた時に分析時間が約 2〜10 秒増加している．

5.2 評価・考察

まず，集約化のパラメータ y を大きくすることでフェーズ 1 での分析時間が増加していることに関しては，問合せに含まれる GROUP BY の計算対象となるグループが増加してしまうためであると考えられる．$y = 100$ から $y = 10,000$ の変化では，分析時間が約 100〜120 秒増加している．また，パラメータ y を大きくすることでフェーズ 2 の分析時間が減少していることに関しては，問合せ対象である認証失敗数が閾値を超えないグループを多く検出できているためであると考えられる．$y = 100$ から $y = 10,000$ の変化では，分析時間が約 620〜630 秒減少している．また，細分化のパラメータ x を大きくすることでフェーズ 2 での分析時間が増加していることに関しては，パラメータ x を大きくすることで，上記の集約化と同様，問合せに含まれる GROUP BY の計算対象となるグループが増加してしまうためであると考えられる．しかし $x = 10$ から $x = 500$ の変化では，集約化の場合ほど顕著な増加はなく，約 2〜10 秒の増加である．

本評価から，ログ分析者の分析時間は，ログ管理者への等価性情報の開示量，つまり問合せに含まれる GROUP BY の計算量に依存するが，このログ管理者への等価性情報の開示量とログ分析者の分析時間の 2 つの要素が比例関係ではないことが分かる．具体的に $y = 100$ というように集約化のパラメータ y を，使用するログデータ数に対して極端に小さくした場合，フェーズ 2 での分析時間が $y = 1,000$ や $y = 10,000$ の場合と比較して約 20〜30 倍と大幅に増加してしまう点である．またフェーズ 1，2 を合計した分析時間で見ると②，⑤，⑧のパターンが約 100〜150 秒での時間での分析が可能となっており，$y = 1,000$ の場合が他のパターンと比較して効率の良い分析が行えることが分かる．つまり，単にログ管理者への等価性情報の開示量を増やしてもログ分析者の分析効率が上がるわけではなく，細分化と集約化のパラメータをうまく選択することで分析効率を上げられる可能性があるということである．

さらに，本稿では評価していないフェーズ 3 に関しては，分析時間が細分化のパラメータに依存すると考える．これは上記と同様にパラメータが増加することで，最後に本来の等価性情報における認証失敗数を合計する計算が増加するためである．

6. まとめ

本稿では，異なるログレコード間における個人情報の等価性を用いる秘匿ログ分析において，ログ管理者への等価性情報の開示量とログ分析者の計算量とのトレードオフ評価を行った．また，予備的評価ではあるが，細分化と集約化のパラメータをうまく選択することで，ログ管理者への等価性情報の開示量を最大にせずとも分析効率を上げられる可能性があることを示した．

今後の予定としては，フェーズ 3 の実装を行い，本稿でのトレードオフ評価対象方式における総括的な評価・考察を検討している．またパラメータのパターン数を増やし，より精密なトレードオフ評価を行いたい．さらに，細分化タグと集約化タグを用いたトレードオフ評価を行う際，2 種類のタグを統一的に扱えるようなデータの秘匿度合の定義を行いたい．

謝辞 本研究に関しまして多大なるご支援をいただきました西日本電信電話株式会社・山内泰介氏に深く感謝いたします．

参考文献

[1] 黒澤馨. クラウドストレージサービスにおける安全なキーワード検索. 電子情報通信学会 基礎・境界ソサイエティ Fundamentals Review, Vol. 9, No. 1, pp. 47–57, 2015.

[2] R. A. Popa and N. Zeldovich. Multi-key searchable encryption. Cryptology ePrint Archive, Report 2013/508, 2013.

[3] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky. Searchable symmetric encryption: Improved definitions and efficient constructions. In *Proceedings of the 13th ACM Conference on Computer and Communications Security*, CCS '06, pp. 79–88, New York, NY, USA, 2006. ACM.

[4] D. X. Song, D. Wagner, and A. Perrig. Practical techniques for searches on encrypted data. In *Proceedings of the 2000 IEEE Symposium on Security and Privacy*, SP '00, pp. 44–, Washington, DC, USA, 2000. IEEE Computer Society.

[5] M. Yoshino, K. Naganuma, and H. Satoh. Symmetric searchable encryption for database applications. In *2011 14th International Conference on Network-Based Information Systems*, pp. 657–662, Sept 2011.

[6] Y. Unagami, N. Matsuzaki, S. Yamada, N. Attrapadung, T. Matsuda, and G. Hanaoka. Private similarity searchable encryption for euclidean distance. In *2016 International Symposium on Information Theory and Its Applications (ISITA)*, pp. 718–722, Oct 2016.

[7] M. Yoshino, H. Sato, and K. Naganuma. Searchable encryption processing system, March 1 2016. US Patent 9,275,250.

[8] A. Lopez-Alt, E. Tromer, and V. Vaikuntanathan. On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. Cryptology ePrint Archive, Report 2013/094, 2013.

[9] K. Naganuma, M. Yoshino, H. Sato, and Y. Sato. Privacy preserving analysis technique for secure, cloud based big data analytics. *Hitachi Review*, Vol. 63, No. 9, p. 578, 2014.

[10] 坂崎尚生, 安細康介, 吉野雅之, 長沼健. 検索可能暗号による秘匿属性認証の提案. In *2016 Symposium on Cryptography and Information Security*, SCIS 2016, Kumamoto, Japan, 2016.

[11] 吉野雅之, 國廣昇, 長沼健, 小野澤綜大. 検索可能型暗号化データベースに対する漏洩情報量の評価. In *2017 Symposium on Cryptography and Information Security*, SCIS 2017, Naha, Japan, 2017.

[12] M. Bellare, S. Keelveedhi, and T. Ristenpart. *Message-Locked Encryption and Secure Deduplication*, pp. 296–312. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[13] Y. Furuta, N. Yanai, M. Karasaki, K. Eguchi, Y. Ishihara, and T. Fujiwara. Towards efficient and secure encrypted databases: Extending message-locked encryption in three-party model. In *12th International Workshop on Data Privacy Management (DPM 2017)*, 2017.

[14] Influxdb.com. Influxdb - open source time series, metrics, and analytics database, 2014. http://influxdb.com/.

[15] J. Ganz, M. Beyer, and C. Plotzky. Time-series based solution using influxdb, 2017. https://beyermatthias.de/papers/2017/Time-series-based-solution-using-influxdb.pdf.

[16] B. Leighton, S. J. D. Cox, N. J. Car, M. P. Stenson, J. Vleeshouwer, and J. Hodge. *A Best of Both Worlds Approach to Complex, Efficient, Time Series Data Delivery*, pp. 371–379. Springer International Publishing, Cham, 2015.

[17] K. Gäfvert and R. Linusson. Improving a network monitoring system by examining a new data storage alternative, 2016. http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-205351.