検索可能暗号に対する SimGraph 攻撃のより詳細な性能評価

並木 拓海[1,a)]　岩本 貢[1]　渡邉 洋平[1]

概要：検索可能暗号は効率的な検索処理のために，安全性を損なわないと考えられる情報の漏洩を許容する．漏洩を許容した情報が本当に漏洩しても安全性を損なわないものかどうかは綿密に議論される必要があり，その議論を攻撃の観点で行う研究が盛んに進められている．このような攻撃研究は漏洩悪用攻撃と呼ばれ，検索可能暗号の漏洩情報を用いてクエリ復元等を試みるものである．もし漏洩情報を用いて攻撃ができるのであれば，その漏洩情報は本来漏洩してはならないものであるといえる．標準的な漏洩情報を用いた攻撃の中では非常に優れた性能を達成する攻撃の 1 つとして SimGraph 攻撃（Namiki et al, ESORICS 2025）が知られている．一般に，漏洩悪用攻撃では漏洩情報とは別に補助情報も必要とするが，SimGraph 攻撃は，他の攻撃アルゴリズムと比較して攻撃実行に必要な補助情報が少ない．本研究では，Namiki らの論文では扱われなかったデータセットを用いて，SimGraph 攻撃のより詳細な性能評価を行うとともに，データセットの特性が攻撃性能にどのような影響を与えるかを明らかにする．

キーワード：暗号化データベース，漏洩悪用攻撃，性能評価

Performance Evaluation of the SimGraph Attacks against Searchable Encryption

Takumi Namiki[1,a)]　Mitsugu Iwamoto[1]　Yohei Watanabe[1]

Abstract: Searchable encryption allows for the leakage of information that is considered not to compromise security in order to enable efficient search processing. Whether or not the leakage actually compromises security if it is disclosed must be carefully discussed, and research from an attack perspective is actively underway. Such attack research is referred to as leakage abuse attacks, which attempt to recover queries using leakage from searchable encryption. If attacks can be carried out using leakage, then that leakage should not have been disclosed in the first place. Among attacks using standard leakage, the SimGraph attack (Namiki et al., ESORICS 2025) is known to achieve exceptionally high performance. Generally, leakage abuse attacks require auxiliary information in addition to leakage, but the SimGraph attack requires less auxiliary information than other attack algorithms. In this study, we conduct a more detailed performance evaluation of the SimGraph attack using a dataset not covered in Namiki et al's paper, and clarify how the characteristics of the dataset affect attack performance.

Keywords: Encrypted Database, Leakage Abuse Attack, Performance Evaluation

1. はじめに

近年，オンラインサービスの普及に伴い，クラウドストレージサービスをはじめとして，様々なクラウドサービスが展開されている．特に，クアウドサービスは一般にクライアントとサーバの二者間の対話型システムとして運用されており，クライアントがサーバにファイルを保存しておき，好きなときにキーワード検索を行い，ファイルを取り出すことができる．サーバに保存されたファイルはデータ

[1] 電気通信大学
〒 182-8585 東京都調布市調布ヶ丘 1-5-1，The University of Electro-Communications
[a)] T.Namiki@uec.ac.jp

ベースとして管理されている．

検索可能暗号（Searchable Encryption: SE）[4], [18] はクライアントとサーバとの対話型オンラインストレージサービスにおいて，検索キーワードとファイルの双方を暗号化したまま検索を可能にする．SE の重要な性質の一つに，効率的な検索処理のために取るに足らないと考えられる情報の漏洩を許容する点にある，漏洩を許容した情報以上の漏洩情報が無いことを安全であると定義される．こうした漏洩情報は本当に漏洩しても安全性を損なわないかは綿密に議論される必要があり，この議論を攻撃の観点で行うのが漏洩悪用攻撃の研究である．

1.1 漏洩悪用攻撃

Islam ら [8] によって提案された漏洩悪用攻撃は，サーバが攻撃者となり，SE の漏洩情報を用いて検索キーワードやサーバに保存されているファイルの平文の復元を試みる攻撃である．漏洩悪用攻撃は，攻撃目標，攻撃者がデータベースに対して実行可能な操作，攻撃者が利用できる補助情報によっていくつかの種類に分類される．

- **攻撃の種類** 漏洩悪用攻撃は受動的攻撃と能動的攻撃に大分され，受動的攻撃は攻撃者がデータベース操作をする必要が無い攻撃のことで，SimGraph 攻撃 [12], VAL 攻撃 [10] 等が知られている．一方で，能動的攻撃は攻撃者がデータベース操作を行いながら実行する攻撃のことで，ファイル挿入攻撃 [21], [22] が知られている．本研究では受動的攻撃に着目する．
- **攻撃目標** 攻撃目標は**クエリ復元**と**データ復元**の二つがある．クエリ復元ではクライアントがクエリした検索キーワードを復元する攻撃であり，多くの攻撃アルゴリズム [2], [3], [4], [5], [6], [7], [9], [11], [15], [16] はクエリ復元を目標としている．一方で，データ復元はクライアントがサーバに保存した暗号化ファイルの平文を復元する攻撃である．LEAP[14] や VAL 攻撃 [10] はクエリ復元に加えて，データ復元も行う攻撃アルゴリズムである．本稿ではクエリ復元攻撃に着目する．
- **補助情報** 漏洩悪用攻撃（特に，能動的攻撃）では漏洩情報だけでは攻撃実行が難しいため，漏洩情報とは別に補助情報も必要とする．**サンプルデータ攻撃** [5], [6], [8], [9], [11], [13], [15], [16], [21] では，攻撃者が攻撃対象データベースと “似た” 分布をもつ別のデータベースから抽出したデータセットを補助情報として用いて攻撃実行する．**既知データ攻撃** [2], [3], [8], [10], [14], [21] では，攻撃者が攻撃対象データベースに保存された一部の平文ファイルを何らかの要因で入手し，補助情報として用いて攻撃実行する．本稿では特に後者の既知データ攻撃に着目する．

1.2 関連研究

本稿では，Blackstone らの Subgraph 攻撃 [2] と，Namiki らの SimGraph 攻撃 [12] の 2 つの既知データ・クエリ復元攻撃に着目する．Islam ら [8] の IKK 攻撃や Cash らの Count 攻撃 [3] では，攻撃者が暗号化データベースのほぼ全ての内容を知っている必要があった他，一部のクエリが既に復元済みであることを仮定していた．つまり，攻撃実行には非常に強い仮定が必要であった．これに対して，Blackstone らは Subgraph 攻撃を提案することで，復元済みクエリを必要とせず，また，攻撃者が暗号化データベースの 5% の内容を知っていれば，クエリ復元攻撃が実行可能であることを示し，その当時存在する他の攻撃と比べて非常に優れた性能を達成することも示した．その後，Namiki らは Subgraph 攻撃のアルゴリズムを解析し，Subgraph 攻撃がクエリ復元に失敗する場合とその原因を明らかにした．また，彼らはその原因となるアルゴリズムを修正することで Subgraph 攻撃を改良した SimGraph 攻撃を提案し，Subgraph 攻撃の約 2 倍の攻撃性能を達成することを示した．したがって，Subgraph 攻撃や SimGraph 攻撃の研究を通じて，攻撃実行に必要な仮定は大きく弱まっており，漏洩情報が安全性を損なわないかをより正確に分析することが可能になっている．

1.3 本研究の貢献

Namiki ら [12] の研究では，SimGraph 攻撃の性能が，Subgraph 攻撃の約 2 倍の性能を達成することを実装実験を通じて明らかにしたが，その事実は Enron 社のメールデータセット [19][*1] を用いた際の結果に対してのみ言及されている．したがって，他のデータセットを用いた場合に同様の結果が得られるのかが明らかにされていない．そこで本研究では，Enron 社のデータセットと同様に漏洩悪用攻撃の研究で広く用いられている Wikipedia の英語記事からなるデータセット [20] を用いて，SimGraph 攻撃と Subgraph 攻撃との間にクエリ復元率の違いがどの程度現れるかを明らかにする．結果として，Enron データセットを用いた場合と比較して，検索結果件数が 5 から 14 件となるようなクエリの復元を試みたときに，クエリ復元率が高くなる結果が得られた．この結果を受け，Enron，Wiki の両データセットを分析し，データセットの特性が攻撃性能にどのような影響を与えるかを明らかにする．

2. 準備

2.1 記法

数学的準備. 任意の整数 n に対して，$[n] := \{1, 2, \ldots, n\}$ とする．有限集合 $\mathcal{X}$ の大きさを $\#\mathcal{X}$ と書く．$\mathcal{X}$ の部分集合 $\mathcal{A}$ の写像 $f\colon \mathcal{X} \to \mathcal{Y}$ による像を $f(\mathcal{A})$ と表す．多重集合及

[*1] データセットの内訳は 4.3 節にて詳説する．

び多重集合の重複度はそれぞれ $\{\!\{\cdot\}\!\}$，m を用いて表す．例えば，多重集合 $\mathcal{X} = \{\!\{a, a, b\}\!\}$ について，a, b の重複度はそれぞれ $\mathsf{m}_{\mathcal{X}}(a) = 2, \mathsf{m}_{\mathcal{X}}(b) = 1$ と表される．

キーワードとクエリ．集合 $\mathbb{W} \coloneqq \{w_1, w_2, \ldots, w_m\}$ をキーワード空間とする．キーワード w_j から生成された検索クエリを $q_{j'}$ とし，集合 $\mathbb{Q}$ をクエリされたクエリされたキーワードの集合とする．すなわち，集合 $\mathbb{Q}$ は集合 $\mathbb{W}$ の部分集合である．また，キーワード w_j 及び検索クエリ $q_{j'}$ それぞれの添字 j と j' は必ずしも等しいとは限らない．

ドキュメント．集合 $\mathbb{D} \coloneqq \{\mathrm{D}_1, \mathrm{D}_2, \ldots, \mathrm{D}_n\}$ をキーワード空間 $\mathbb{W}$ 上のドキュメント集合とし，$\mathbb{ED} \coloneqq \{\mathrm{ED}_1, \mathrm{ED}_2, \ldots, \mathrm{ED}_n\}$ を集合 $\mathbb{D}$ に属するドキュメントの暗号文集合とする．ただし，平文ドキュメント D_i と暗号化ドキュメント $\mathrm{ED}_{i'}$ のそれぞれの添字 i と i' は必ずしも一対一対応しているわけではない．つまり，暗号化ドキュメント ED_i は必ずしも平文ドキュメント D_i に対応する暗号文であるとは限らない．便宜上，本稿ではしばしば任意のドキュメント $\mathrm{D} \in \mathbb{D}$ はいくつかのキーワードからなる集合，すなわち $\mathrm{D} \subseteq \mathbb{W}$ として扱う．つまり，適切なステミングアルゴリズム（例えば，Porter Stemming アルゴリズム [17]）を用いて，キーワードが抽出されていることを想定する．

識別子．任意のドキュメント $\mathrm{D} \in \mathbb{D}$ は固有の識別子が割り振られており，関数 $\mathbf{id}\colon \mathbb{D} \to [n]$ を用いて $\mathbf{id}(\mathrm{D})$ と表す．特に，$\mathrm{D} = \mathrm{D}_i$ のとき $\mathbf{id}(\mathrm{D}_i) = i$ とする．また，$\mathbb{D} = \{\mathrm{D}_1, \mathrm{D}_2, \ldots, \mathrm{D}_n\}$ に対して，$\mathbf{id}(\mathbb{D}) \coloneqq \{\mathbf{id}(\mathrm{D}_1), \mathbf{id}(\mathrm{D}_2), \ldots, \mathbf{id}(\mathrm{D}_n)\}$ とする．キーワード w が元の一つであるようなドキュメントの集合は $\mathbf{D}(w)$ と表す．

ボリューム．しばしばドキュメント D はキーワードの集合として扱うが，$|\mathrm{D}|$ は D のバイト長を表す．つまり，ドキュメント D のボリューム $|\mathrm{D}|$ は D の集合としての大きさのことでは無いことに注意されたい．また，ある二つのドキュメント $\mathrm{D}_i, \mathrm{D}_j \in \mathbb{D}$（ただし，$\mathrm{D}_i \neq \mathrm{D}_j$）が存在して，$|\mathrm{D}_i| = |\mathrm{D}_j|$ となることがあるため，$\mathbb{D} = \{\mathrm{D}_1, \mathrm{D}_2, \ldots, \mathrm{D}_n\}$ に対して，$|\mathbb{D}|$ を多重集合とし，$|\mathbb{D}| \coloneqq \{\!\{|\mathrm{D}_1|, |\mathrm{D}_2|, \ldots, |\mathrm{D}_n|\}\!\}$ とする．

2.2 検索可能暗号

先行研究 [2], [3], [8], [12] と同様に，本研究ではデータベース更新が無い non-dynamic な SE[4] に着目する．SE は三つの確率的多項式時間アルゴリズム, Setup, QueryGen, 及び Search からなる．Setup はセキュリティパラメータ κ とドキュメント集合 $\mathbb{D}$ を入力にとり，クライアントに秘密鍵 k を，サーバに暗号化データベース EDB をそれぞれ出力する．QueryGen は秘密鍵 k とキーワード $w \in \mathbb{W}$ を入力にとり，検索クエリ q を出力する．Search はキーワード w から生成された検索クエリ q と暗号化データベース EDB を入力にとり，検索結果 $\mathcal{X}(q)$ と，更新済み暗号化データベース EDB' を出力する．ここで，Search の正当性とは，検索結果 $\mathcal{X}(q)$ がキーワード w を含むドキュメントの識別子集合と等しいこと，すなわち，$\mathcal{X}(q) = \mathbf{id}(\mathbf{D}(w))$ が成り立つことをいう．

2.3 漏洩関数

SE は効率的な検索処理のために，その情報からは暗号化データベースに関する情報が漏れないと考えられる情報（しばしば “取るに足らない情報” と呼ぶ）の漏洩を許容する．このような漏洩情報は SE の具体的な構成によって異なり，漏洩関数としてあらかじめ定義される．SE の安全性は漏洩関数以上の漏洩情報が無いことを保証する．一般に，より多くの取るに足らないとされる情報の漏洩を許容すれば，より効率的な検索処理が可能になる．一方で，そのような漏洩情報は真に「取るに足らない情報」でなければ安全性を損なう可能性がある．したがって，効率性と安全性のトレードオフを考慮した，バランスの良い SE の方式を提案することが研究課題の中心となっており，同時に，漏洩悪用攻撃の研究はどのような情報が「取るに足らない情報」といえるかを攻撃の観点で議論する研究分野である．

SE で許容される漏洩情報は入力に対応する関数として定義され，**漏洩関数**と呼ばれる．Cash ら [3] は漏洩関数を L1 から L4 の四段階に分類し，L1 が最も漏洩情報が少なく，L4 が最も漏洩情報が多い定義となっている．本稿では L1 の漏洩関数に含まれる二つの漏洩情報**アクセスパターン**と**ボリュームパターン**に着目する．

アクセスパターン．アクセスパターンとはキーワード w を用いて検索した際に，検索結果として返答されたドキュメントの識別子集合，すなわち，キーワード w を含むドキュメントの識別子集合のことをいう．具体的には，関数族 $\mathsf{AP} \coloneqq \{\mathsf{AP}_t\}_{t\in\mathbb{N}}$ で表され，関数 $\mathsf{AP}_t\colon \left(2^{\mathbb{W}}\right) \times \mathbb{W}^t \to \left(2^{[n]}\right)^t$ は

$$\mathsf{AP}_t(\mathbb{D}, \boldsymbol{w}_t) \coloneqq (\mathbf{id}(\mathbf{D}(w_1)), \ldots, \mathbf{D}(w_t)) = (\mathcal{X}(q_1), \ldots, \mathcal{X}(q_t))$$

と定義される．ただし，$\boldsymbol{w}_t \coloneqq (w_1, \ldots, w_t)$ とし，クエリ $q_1, \ldots, q_t$ はそれぞれキーワード $w_1, \ldots, w_t$ から生成されたとし，Search の正当性を仮定する．

ボリュームパターン．ボリュームパターンとはキーワード w を用いて検索した際に，検索結果として返答されたドキュメントのバイト長の多重集合，すなわち，キーワード w を含むドキュメントのバイト長の多重集合のことをいう．具体的には，関数族 $\mathsf{VP} \coloneqq \{\mathsf{VP}_t\}_{t\in\mathbb{N}}$ で表され，関数 $\mathsf{VP}_t\colon \left(2^{\mathbb{W}}\right) \times \mathbb{W}^t \to \mathbb{N}^t$ は

$$\begin{aligned}\mathsf{VP}_t(\mathbb{D}, \boldsymbol{w}_t) &\coloneqq \left(\{\!\{|\mathrm{D}|\}\!\}_{\mathrm{D}\in\mathbf{D}(w_1)}, \ldots, \{\!\{|\mathrm{D}|\}\!\}_{\mathrm{D}\in\mathbf{D}(w_t)}\right) \\ &= \left(|\mathcal{X}(q_1)|, \ldots, |\mathcal{X}(q_t)|\right)\end{aligned}$$

と定義される．ただし，$\boldsymbol{w}_t \coloneqq (w_1, \ldots, w_t)$ とし，クエリ $q_1, \ldots, q_t$ はそれぞれキーワード $w_1, \ldots, w_t$ から生成されたとし，Search の正当性を仮定する．また，$|\mathcal{X}(q_i)|$ はクエリ q_i $(i \in [t])$ に対して返答されたドキュメントのバイト長の多重集合，すなわち，$|\mathcal{X}(q_i)| \coloneqq \{\!\{|\mathrm{D}|\}\!\}_{\mathbf{id}(\mathrm{D}) \in \mathcal{X}(q_i)} = \{\!\{|\mathrm{D}|\}\!\}_{\mathrm{D} \in \mathbf{D}(w_i)}$ とする．

暗号化ドキュメント ED から，対応する平文ドキュメント D のバイト長を得られると仮定すると，ボリュームパターンはアクセスパターンから導出することができる．したがって，ボリュームパターンはアクセスパターンよりも少ない漏洩といえる．

3. 漏洩悪用攻撃

本節では，漏洩悪用攻撃の攻撃者モデルについて詳説し，今回特に着目する二つの攻撃アルゴリズム，Subgraph 攻撃 [2] と SimGraph 攻撃 [12] の概要を紹介する．これら両攻撃アルゴリズムはともに，2.3 節にて紹介したアクセスパターン AP，あるいはボリュームパターン VP と，3.1 節にて後述する補助情報それぞれの二部グラフを作成し，２つのグラフを比較しながらクエリ復元を試みる攻撃である．二部グラフの構成や攻撃アルゴリズムの詳細に関しては Blackstone らの論文 [2] や Namiki らの論文 [12] を参照されたい．

3.1 攻撃者モデル

一般に，SE に対する攻撃者はプロトコルを忠実に実行するが，実行した結果得られる情報を収集し，収集した情報をもとにクエリ復元 [2], [3], [8], [9], [12], [15], [16][*2]を試みる．具体的には，攻撃者はすべてのクエリとそれぞれのクエリから漏洩する情報を収集し，これらの情報を用いてクエリに対応するキーワードを推測する．

このような推測のために，多くの攻撃研究では [2], [3], [8], [10], [12], [14], [21] では，攻撃者はクライアントがサーバに預けた一部の平文ドキュメントを補助情報として保有している状況を仮定する．このような補助情報は，例えば，SE の Setup アルゴリズムの実行時のセキュリティ侵害や，広く公開されているデータ（例えば，暗号化データベースに保管されている公開された E メール，Web サイトの記事など）から収集することができる．こうした状況設定では実用的な攻撃実行を可能にするが，SE のプロトコルを実行するシナリオに沿って考えると現実的でない可能性がある．したがって，使用する補助情報は可能な限り少ないことが要求され，より少ない補助情報でより強力な攻撃アルゴリズムを提案することが研究目標の一つである．

つまり，本稿では攻撃者は以下の情報を保有し，利用しながら攻撃実行することを仮定する．

- **漏洩情報**．t 個の検索クエリ $q_1, q_2, \ldots, q_t \in \mathbb{Q}$ を受信することで得られる漏洩情報．攻撃の種類によって，アクセスパターン AP_t やボリュームパターン VP_t を得ることができる．
- **補助情報**．$\mathbb{D}$ 全体の件数の δ の割合だけ攻撃者は平文ドキュメントを保有している．以降，δ のことを補助情報の割合と呼ぶ．補助ドキュメント集合は全体のドキュメント集合の部分集合であり，補助情報の割合は $\delta \coloneqq \#\widetilde{\mathbb{D}}/\#\mathbb{D}$ である．また，攻撃者は補助ドキュメント集合 $\widetilde{\mathbb{D}}$ から補助キーワード集合 $\widetilde{\mathbb{W}} \coloneqq \bigcup_{\mathrm{D} \in \widetilde{\mathbb{D}}} \mathrm{D}$ も導出することができる．

これらの情報を用いて，攻撃者はクエリ $q \in \mathbb{Q}$ に対するキーワードの推測結果 $C(q) \subseteq \widetilde{\mathbb{W}}$ を出力する．$C(q)$ は，クエリ q を生成する際に入力候補となるキーワードの集合である．ここで，クエリ復元成功の条件は $C(q) = \{w\}$ であり，かつ $q \leftarrow \mathsf{QueryGen}(k, w)$ のときとする．ただし，キーワード w の探索範囲は $\widetilde{\mathbb{W}}$ であり，$\mathbb{W}$ ではないことに注意されたい．

3.2 Subgraph 攻撃

Subgraph 攻撃は Blackstone ら [2] によって提案され，L1 の漏洩を用いた攻撃の中では非常に優れた性能を達成する攻撃アルゴリズムの一つであり，アクセスパターンの漏洩を用いる $\mathsf{Subgraph}^{\mathsf{ID}}$ 攻撃と，ボリュームパターンの漏洩を用いる $\mathsf{Subgraph}^{\mathsf{VL}}$ の二種類が提案されている．一般に，漏洩悪用攻撃では利用できる漏洩情報が少ないほど攻撃性能が低くなるが，アクセスパターン AP より少ない漏洩情報であるボリュームパターン VP を用いる $\mathsf{Subgraph}^{\mathsf{VL}}$ 攻撃の性能が，アクセスパターン AP を用いる $\mathsf{Subgraph}^{\mathsf{ID}}$ 攻撃と同程度であることが知られている．また，Blackstone らは，検索結果件数が多いクエリほど，対応する検索キーワードを推測しやすいことも明らかにした．一方で，Blackstone らの実験では現実的でない状況設定が行われており，具体的には，攻撃者は既知キーワード空間 $\widetilde{\mathbb{W}}$ の部分集合に属するキーワードがクエリされることを仮定していた．そのため，キーワードの探索範囲が狭くなり，クエリ復元が容易になっていた．Namiki ら [12] はこうした事実を明らかにするとともに，本来用いるべき状況設定を提案した．

3.3 SimGraph 攻撃

Subgraph 攻撃では，クエリに対応するキーワードが補助情報内に存在するのにもかかわらず，そのようなキーワード復元先の候補から除外する偽陰性と呼ばれる事象が多く発生していた．Namiki ら [12] は Subgraph 攻撃で起こる偽陰性の原因を明らかにし，その点を修正することで，よりシンプルなアルゴリズムで，より高い攻撃性能を達成する攻撃として $\mathsf{Subgraph+}^{\mathsf{ID}}$ 攻撃と SimGraph 攻撃を提案した．また，$\mathsf{Subgraph+}^{\mathsf{ID}}$ 攻撃や SimGraph 攻撃は Subgraph

[*2] いくつかの攻撃 [10], [14] では暗号化ドキュメントから平文ドキュメントを得る，ドキュメント復元に主眼を置いている．

攻撃と同様に検索結果件数が多いクエリほど，対応する検索キーワードを推測しやすいことが示されており，アクセスパターン AP を用いる Subgraph+$^{\mathsf{ID}}$, SimGraph$^{\mathsf{ID}}$ 攻撃とボリュームパターン VP を用いる SimGraph$^{\mathsf{VL}}$ 攻撃の性能が同程度であることも知られている．Subgraph+$^{\mathsf{ID}}$ 攻撃と SimGraph$^{\mathsf{ID}}$ 攻撃は攻撃性能が同等であることが示されているため，本稿では SimGraph 攻撃にのみ着目する．

4. 攻撃性能の評価

4.1 目的

3.2 節，3.3 節で述べた通り，Subgraph 攻撃と SimGraph 攻撃はともに，検索結果のドキュメントの件数が多いクエリほど，対応する検索キーワードを正しく見つけやすいことがわかっている．ただし，既存研究 [2], [12] では，この事実は Enron メールデータセット [19] にのみ言及されており，他のデータセット（例えば，Wiki データセット）を用いた性能評価が行われていないため，本研究では Enron メールデータセットに加えて，Wiki データセットを用いて実験を行い，検索結果件数が多いクエリほど，対応するキーワードを推測しやすいかを明らかにし，特に，Wiki データセットを用いた場合に，SimGraph$^{\mathsf{ID}}$ 攻撃が Subgraph$^{\mathsf{ID}}$ 攻撃と比べてどれだけ多くのクエリを復元可能かを明らかにする．

4.2 Selectivity

検索クエリ q（検索キーワード w）に対してレスポンスされるドキュメントの数 $\#\mathbf{D}(w)$ を Selectivity と呼ぶ．Selectivity を用いて，検索クエリを以下条件に従って 3 つの区分に分ける．

- **High Selectivity**: $\#\mathbf{D}(w) \geq 15$
- **Middle Selectivity**: $5 \leq \#\mathbf{D}(w) \leq 14$
- **Low Selectivity**: $\#\mathbf{D}(w) \leq 4$

4.3 データセット

本研究では 2 つのデータセット *Enron* **メールデータセット**と *Wiki* **データセット**を用いる．いずれのデータセットも PyPI にて公開されている NLTK ライブラリ [1] を用いて，Porter Stemming アルゴリズム [17] による互換抽出と Stopwords の削除を行った．

Enron メールデータセット．Enron メールデータセット [19] は，Enron 社に所属する社員が実際に送受信したメールの集合であり，多くの攻撃研究で用いられている．また，倫理的な配慮として，このデータセットは社員からの要望に応じてメールが削除されている．このデータセットはメールデータセットであるため，複数のドキュメント（ファイル）で繰り返し用いられている特定のキーワードの組み合わせ（便宜上，定型文と呼ぶ）が多く出現する．すなわち，ある相異なる 2 つのクエリが存在して，それらの検索結果が等しくなる確率が比較的高い．

本稿では Enron 社のメールデータセットのうち，arnold-j ユーザディレクトリに含まれる 4,897 件のファイルを用いる．ただし，データセットの各ファイルからメールヘッダを削除した．キーワードの総数は 21,903 個で，4.2 節の分類方法に従うと，High Selectivity のキーワードが 3,658 個，Middle Selectivity のキーワードが 3,998 個，Low Selectivity のキーワードが 14,247 個である．

Wiki データセット．Wiki データセットは，Wikipedia に投稿された記事の集合であり，Enron 社メールデータセットと同様に過去の攻撃研究で用いられている．このデータセットは，メールデータセットとは異なり定型文が少ない．すなわち，ある相異なる 2 つのクエリが存在して，それらの検索結果が等しくなる確率が比較的低い．

本稿では 2025 年 8 月 1 日時点で公開されている Wikipedia の英語記事を収集したデータセット [20] から 5,000 件の記事データを抽出したものを用いる．キーワードの総数は 315,904 個で，4.2 節の分類方法に従うと，High Selectivity のキーワードが 26,049 個，Middle Selectivity のキーワードが 30,884 個，Low Selectivity のキーワードが 258,971 個である．

4.4 実験手順

攻撃アルゴリズムの実装にはプログラミング言語 Rust を用い，以下の二つの計算機環境を用いて実験を行った．

- OS が Arch Linux，CPU が Intel Core i9–12900K，RAM が 128GB の計算機環境
- OS が Ubuntu 24.04 LTS，CPU が AMD Ryzen 7 9800X3D，RAM が 32GB の計算機環境

4.3 節にて言及した 2 つのデータセットを用いるいずれの実験も，150 個のキーワードがクエリされたと仮定する．すなわち，攻撃者は 150 個のクエリそれぞれに対して，対応するキーワードを既知キーワード空間 $\widetilde{\mathbb{W}}$ から探索し，推測する状況を考え，Subgraph 攻撃，SimGraph 攻撃の両アルゴリズムの攻撃性能を評価する．具体的には，補助情報の割合を 5% から 100% に 5% ずつ増加させ，各補助情報の割合でクエリ復元率を計測して得られた最大値，中央値，最小値を計算する．得られた結果は 4.5 節にて示す．実験の状況設定は Namiki らの方法 [12] に従う．

4.5 実験結果

Enron データセットと Wiki データセットを用いて，Subgraph 攻撃，SimGraph 攻撃によるクエリ復元の実験結果を図 1–4 に示し，補助情報の割合を 5% から 100% に 5% ずつ増加させ，各補助情報の割合でクエリ復元率を計測して得られた最大値，中央値，最小値を図示する．

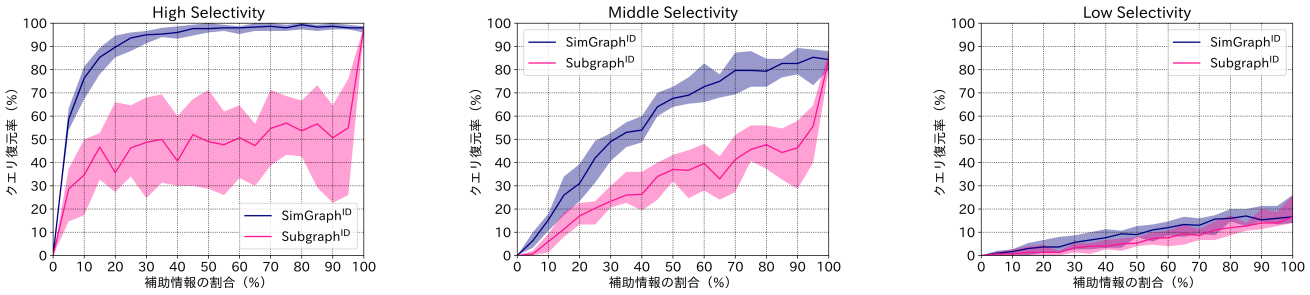

図 1 Enron データセットを用いた Subgraph^{ID} 攻撃と SimGraph^{ID} 攻撃の性能比較 [12]

Fig. 1 Performance comparisons between Subgraph^{ID} and SimGraph^{ID} attacks with the Enron dataset [12]

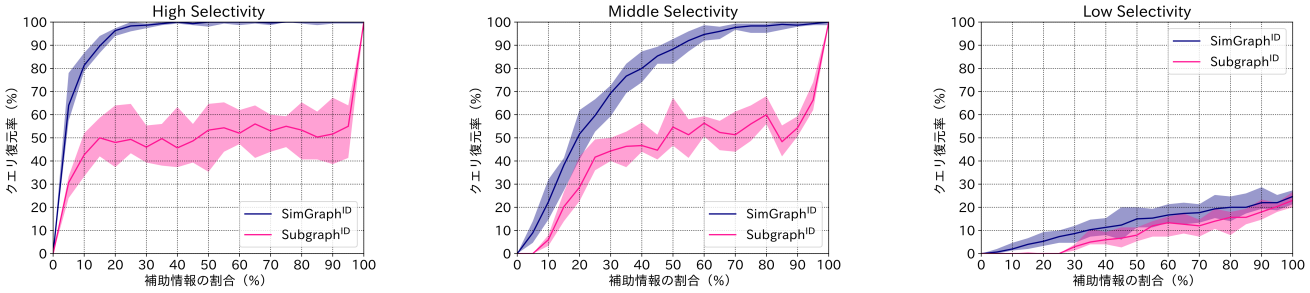

図 2 Wiki データセットを用いた Subgraph^{ID} 攻撃と SimGraph^{ID} 攻撃の性能比較

Fig. 2 Performance comparisons between Subgraph^{ID} and SimGraph^{ID} attacks with the Wiki dataset

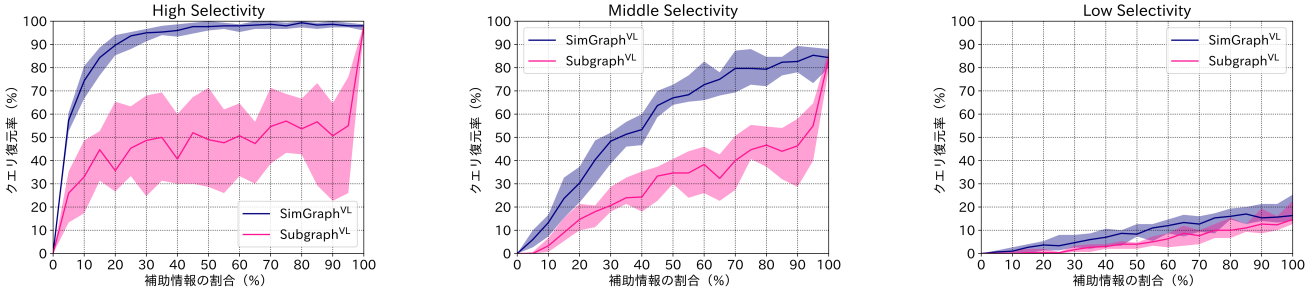

図 3 Enron データセットを用いた Subgraph^{VL} 攻撃と SimGraph^{VL} 攻撃の性能比較 [12]

Fig. 3 Performance comparisons between Subgraph^{VL} and SimGraph^{VL} attacks with the Enron dataset [12]

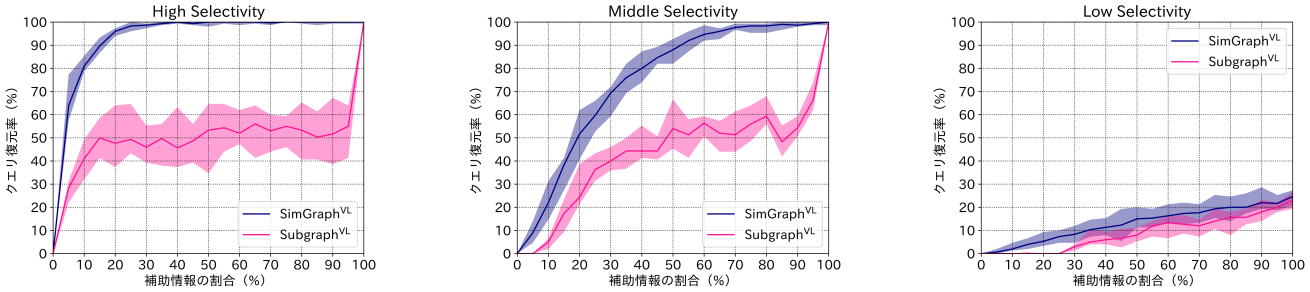

図 4 Wiki データセットを用いた Subgraph^{VL} 攻撃と SimGraph^{VL} 攻撃の性能比較

Fig. 4 Performance comparisons between Subgraph^{VL} and SimGraph^{VL} attacks with the Wiki dataset

図 1，3 は Enron データセットを用いた実験結果であり，Namiki らの論文 [12] で示されたものと同じである．図 2，4 は Wiki データセットを用いた実験結果である．

Selectivity の影響．Wiki データセットを用いた実験の結

果（図 2，4）から，SimGraph 攻撃は Enron データセットを用いた実験の結果（図 1，3，[12]）と同様に，Selectivity が高いほどクエリ復元率が高くなる傾向が見られる．Enron のデータセットを用いた場合に関して，Namiki ら [12] は，Selectivity が高いほど漏洩情報がユニークなものになりやすいため，クエリ復元がより正確になると考察している．Wiki データセットを用いた場合も同様に，Selectivity が高いほど漏洩情報がユニークなものになりやすいため，クエリ復元がより正確になると考えられる．

データセットの影響．Namiki ら [12] は Subgraph+$^{\mathrm{ID}}$ 攻撃に関して，ユニークな漏洩情報が多いほど，クエリ復元率が高くなることを確認しており，その派生である SimGraph 攻撃もそのような傾向が見られることが実験結果から確認されている．本研究における実験結果から，Subgraph 攻撃，SimGraph 攻撃の両方で，Selectivity に関係なく，Wiki データセットを用いた場合のクエリ復元率が高くなっている．また，両攻撃のクエリ復元率の最大値と最小値の差に関しても，Enron データセットは大きいが，Wiki データセットでは小さい．これらの理由として，Enron データセットでは定型文が多く，複数のクエリ（キーワード）の間で検索結果や補助情報が等しくなるケースが多く，逆に，Wiki データセットでは定型文が少なく，複数のクエリ（キーワード）の間で検索結果や補助情報が等しくなるケースが少ないからと考えられる．つまり，等しい漏洩情報，補助情報を持つキーワード，クエリの割合が大きくなるほど，一定の補助情報の割合におけるクエリ復元率の差が大きくなると考えられる．4.3 節で述べたことから，Wiki データセットは Enron データセットではドキュメント件数に大きな差は無い．一方で，キーワードの数では Wiki データセットは Enron データセットの約 15 倍の数になるため，直感的に考えれば等しい漏洩情報，補助情報となるような 2 つのキーワードの組み合わせの個数が多く現れるのは Wiki データセットの方であると推測するのが自然である．しかしながら，実験結果ではむしろ Enron データセットの方でそのようなキーワードの組み合わせの個数が多いことが示唆されている．したがって，ユニークな情報を持つキーワードをより多く含んでいるのは Wiki データセットであると考えられる．

こうした点を踏まえて，クエリ復元率の変化が顕著な SimGraph 攻撃（特に，SimGraph$^{\mathrm{ID}}$ 攻撃）の Middle Selectivity の実験結果に着目する．前述の通り Wiki データセットを用いた場合のクエリ復元率が Enron データセットを用いた場合のクエリ復元率よりも高い結果が得られている．それぞれのデータセットの Middle Selectivity のクエリ復元実験の結果の比較表を表 1 に示す．表 1 から，補助情報の割合が 20% 以上のときに特にクエリ復元率の差が大きく，Wiki データセットを用いた場合のクエリ復元率が

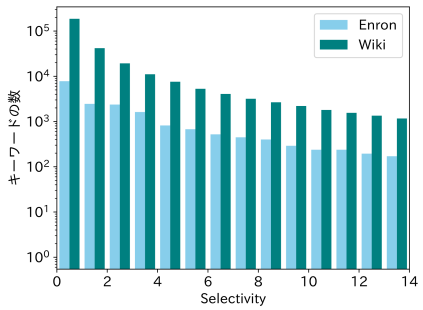

図 5 Low/Middle Selectivity のキーワードの数．縦軸が対数スケールの片対数グラフであることに注意されたい．

Fig. 5 The number of low and middle selectivity keywords in each dataset. Note that the graph is a semi-logarithmic graph with a logarithmic scale vertical axis.

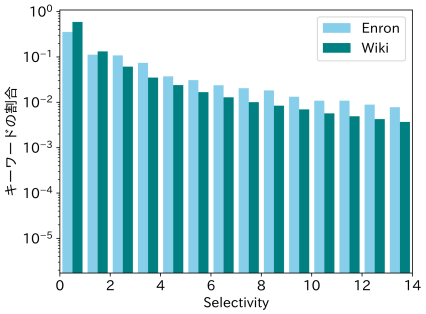

図 6 Low/Middle Selectivity のキーワードの割合．縦軸が対数スケールの片対数グラフであることに注意されたい．

Fig. 6 The ratio of low and middle selectivity keywords in each dataset. Note that the graph is a semi-logarithmic graph with a logarithmic scale vertical axis.

Enron データセットを用いた場合の 20% 前後高い結果となっている．次に，Middle Selectivity のキーワードの数の比較を行う．Enron，Wiki の両データセットに含まれる Middle 以下の Selectivity のキーワードの個数を図 5 に示す．各 Selectivity において，Wiki データセットの方が 10 倍程度の数のキーワードを含んでいることがわかる．さらに，このデータを割合にして図示した結果を図 6 に示す．割合で比較すると，Selectivity が 1 または 2 のキーワードの割合が Wiki の方が大きいが，Middle Selectivity のキーワードの割合は Enron の方が大きいことがわかる．これらのことからも，攻撃性能はキーワードの数ではなく，キーワードの割合に依存し，特定の情報[*3]をもつキーワードの割合が低いほど，攻撃性能が高くなると考えられる．

5. まとめと今後の課題

本研究では Enron メールデータセットと Wiki データセットの二つのデータセットを用いて，Subgraph 攻撃と SimGraph 攻撃の性能評価を行い，データセットの “ユニークさ” が攻撃性能に大きな影響を与えることを明らかにし

[*3] Selectivity や漏洩情報，補助情報を指す．

表 1　SimGraph^{ID} 攻撃の Middle Selectivity のクエリ復元率（中央値）の比較

Table 1　Comparison of SimGraph^{ID} attack's query recovery rate (median) between Enron and Wiki datasets for middle selectivity queries.

補助情報の割合	5%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
クエリ復元率（Enron）	6%	15%	31%	49%	54%	68%	72%	80%	79%	83%	84%
クエリ復元率（Wiki）	9%	22%	52%	69%	80%	88%	95%	98%	98%	99%	100%

た．具体的には，Subgraph と SimGraph 攻撃の両アルゴリズムで Wiki データセットを用いるとクエリ復元率が大きくなる傾向が見られた．メールの場合，定型文とも呼べる決まったフレーズが多用されていることから，そうしたフレーズで使われている単語（キーワード）の復元が難しいと考えれ，一方で，Wiki データセットではそうした定型文が少ないため，比較的クエリ復元が容易であったことが考えられる．特に，SimGraph 攻撃の Middle Selectivity のクエリ復元率が Wiki データセットを用いた場合に高くなった点は非常に興味深い結果であり，SimGraph 攻撃がよりユニークな情報をもつキーワードの割合が高いほど攻撃性能が高くなることがわかった．

今後は，使用するデータセットのドキュメント件数を増加させた場合にどのような変化が現れるかを明らかにするとともに，攻撃アルゴリズムの実行時間やメモリ使用量を理論的に評価し，攻撃性能と効率の関係性を明らかにする．

謝辞　本研究は JSPS 科研費 JP23H00468, JP23H00479, JP23K17455, JP23K21644, JP23K21668, JP23K24846 の助成，および JST 経済安全保障重要技術育成プログラム【JPMJKP24U2】 の支援を受けたものです．

参考文献

[1] Bird, S., Klein, E. and Loper, E.: *Natural language processing with Python: analyzing text with the natural language toolkit*, " O'Reilly Media, Inc." (2009).

[2] Blackstone, L., Kamara, S. and Moataz, T.: Revisiting Leakage Abuse Attacks, *NDSS 2020*, The Internet Society (2020).

[3] Cash, D., Grubbs, P., Perry, J. and Ristenpart, T.: Leakage-Abuse Attacks Against Searchable Encryption, *ACM CCS 2015*, ACM, pp. 668–679 (2015).

[4] Curtmola, R., Garay, J. A., Kamara, S. and Ostrovsky, R.: Searchable symmetric encryption: improved definitions and efficient constructions, *ACM CCS 2006*, ACM, pp. 79–88 (2006).

[5] Damie, M., Hahn, F. and Peter, A.: A Highly Accurate Query-Recovery Attack against Searchable Encryption using Non-Indexed Documents, *USENIX Security 2021*, USENIX Association, pp. 143–160 (2021).

[6] Gui, Z., Paterson, K. G. and Patranabis, S.: Rethinking Searchable Symmetric Encryption, *IEEE SP 2023*, pp. 1401–1418 (2023).

[7] Hoover, A., Ng, R., Khu, D., Li, Y., Lim, J., Ng, D., Lim, J. and Song, Y.: Leakage-Abuse Attacks Against Structured Encryption for SQL, *USENIX Security 2024* (Balzarotti, D. and Xu, W., eds.), USENIX Association (2024).

[8] Islam, M. S., Kuzu, M. and Kantarcioglu, M.: Access Pattern disclosure on Searchable Encryption: Ramification, Attack and Mitigation, *NDSS 2012*, The Internet Society (2012).

[9] Kamara, S., Kati, A., Moataz, T., DeMaria, J., Park, A. and Treiber, A.: MAPLE: MArkov Process Leakage attacks on Encrypted Search, *Proc. Priv. Enhancing Technol.* (2024).

[10] Lambregts, S., Chen, H., Ning, J. and Liang, K.: VAL: Volume and Access Pattern Leakage-Abuse Attack with Leaked Documents, *ESORICS 2022*, Vol. 13554, Springer, pp. 653–676 (2022).

[11] Liu, H., Xu, L., Liu, X., Mei, L. and Xu, C.: Query Correlation Attack Against Searchable Symmetric Encryption With Supporting for Conjunctive Queries, *IEEE Trans. Inf. Forensics Secur.*, pp. 1924–1936 (2025).

[12] Namiki, T., Amada, T., Iwamoto, M. and Watanabe, Y.: Crrecting the Record on Leakage Abuse Attacks: Revisiting the Subgraph Attacks with Sound Evaluation, *ESORICS 2025*, Springer (2025).

[13] Nie, H., Wang, W., Xu, P., Zhang, X., Yang, L. T. and Liang, K.: Query Recovery from Easy to Hard: Jigsaw Attack against SSE, *USENIX Security 2024*, USENIX Association, pp. 2599–2616 (2024).

[14] Ning, J., Huang, X., Poh, G. S., Yuan, J., Li, Y., Weng, J. and Deng, R. H.: LEAP: Leakage-Abuse Attack on Efficiently Deployable, Efficiently Searchable Encryption with Partially Known Dataset, *ACM CCS 2021*, ACM, pp. 2307–2320 (2021).

[15] Oya, S. and Kerschbaum, F.: Hiding the Access Pattern is Not Enough: Exploiting Search Pattern Leakage in Searchable Encryption, *USENIX Security'21*, USENIX Association, pp. 127–142 (2021).

[16] Oya, S. and Kerschbaum, F.: IHOP: Improved Statistical Query Recovery against Searchable Symmetric Encryption through Quadratic Optimization, *USENIX Security'22*, USENIX Association, pp. 2407–2424 (2022).

[17] Porter, M. F.: An algorithm for suffix stripping, *Program*, Vol. 14, No. 3, pp. 130–137 (1980).

[18] Song, D. X., Wagner, D. A. and Perrig, A.: Practical Techniques for Searches on Encrypted Data, *IEEE S&P 2000*, IEEE, pp. 44–55 (2000).

[19] The CALO Project: Enron Email Dataset (May 7, 2015 Version), `https://www.cs.cmu.edu/~enron/` (2015).

[20] Wikimedia Commons: enwiki dump progress on 20250801, `https://dumps.wikimedia.org/enwiki/20250801/` (2025).

[21] Xu, L., Zheng, L., Xu, C., Yuan, X. and Wang, C.: Leakage-Abuse Attacks Against Forward and Backward Private Searchable Symmetric Encryption, *ACM CCS 2023*, ACM, p. 3003–3017 (2023).

[22] Zhang, Y., Katz, J. and Papamanthou, C.: All Your Queries Are Belong to Us: The Power of File-Injection Attacks on Searchable Encryption, *USENIX Security 2016*, USENIX Association, pp. 707–720 (2016).