小口研究室　研究紹介　(2022年度)
(お茶の水女子大学理学部情報科学科)

5G SA環境におけるIoTシステムの通信性能調査
（研究担当:伊藤　千紗）

背景/目的

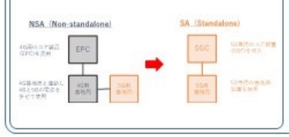
さまざまなモノに通信デバイスを組み込み、インターネットに接続しようとする試みをIoT (Internet of Things)と呼ぶ.
しかしモバイル環境にあるIoTデータの収集で要求される通信スループットや通信遅延を維持するには、現在のモバイル通信技術であるLTEでは難しい.高性能なモバイル通信技術として, 5Gの活用が期待されているが, IoT通信における性能は明らかでない.

5G SA (Stand Alone) 環境でのIoTデータ通信性能を調査

使用技術

5G SA (Stand Alone)

4G 用のコア装置を流用する 5G NSA (Non-Stand Alone) に対して, 5G 専用のコア装置と基地局を使用するもの.



SINETStream

国立情報学研究所 (NII) が開発しているIoTシステム開発支援ライブラリ. IoT 通信に適した Publisher/Subscriber 型の通信モデルを採用しており, 広域ネットワークを介してデータを欠損なく確実に収集・解析するための機能を提供する.

実験概要

国立情報学研究所に設置されている NTT docomo の 5G SA 環境と, LTE 環境でそれぞれスループットを測定する. クライアントからメッセージを送信し, ブローカを介してクライアント がメッセージを受信するまでの往復通信のスループットを計測する.

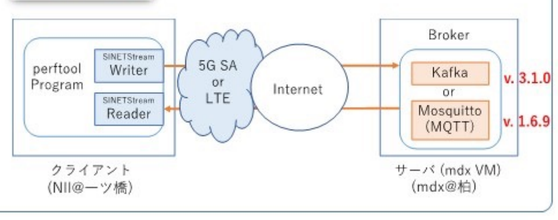
実験条件

ブローカ	Kafka / Mosquitto
暗号化	有 (AES) / 無
データサイズ (byte)	1024 / 10240 / 102400
スレッド数	1 / 2 / 4 / 8 /16

実験環境

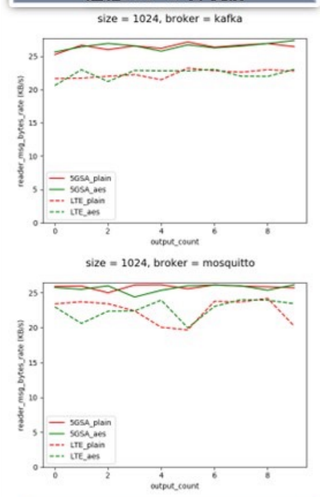
クライアント：Raspberry Pi	
機種名	Raspberry Pi 4 Computer Model B 4GB RAM
OS	Raspbian GNU / Linux 11
CPU	ARMv7 Processor rev 3 (v7l)
Main Memory	4GB
サーバ：mdx VM	
OS	Ubuntu 20.04.5 LTS
仮想CPUコア数	16
メモリ	24.19GB
仮想ディスクサイズ	100GB

実験構成図



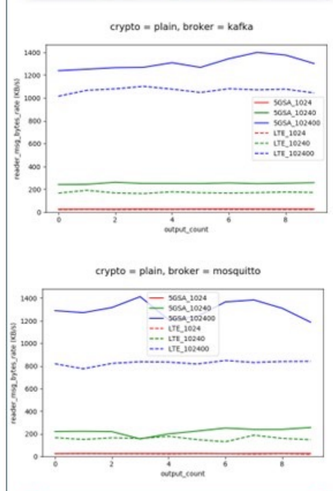
実験結果

暗号化の有無での通信スループット比較



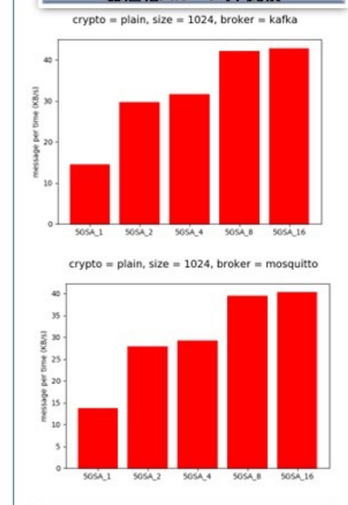
LTEと比べて5G SAの方が性能が良く, 暗号化による性能差は見られない

データサイズの違いによる通信スループット比較



どのデータサイズでも LTEと比べて5G SA の方が性能が良い

5G SAでスレッド数毎の単位時間あたり総通信スループット比較



スレッド数を増やすと総通信スループットが向上する

まとめ

- SINETStreamを用いて 5G SA 環境のスループット測定を行い, 従来のLTEと比べて高い性能を示すことを確認した.
- 今後は, 様々なデータサイズでマルチスレッドの実験を行うこと, 5G NSAでもスループットを測定して比較を行うことを考えている.

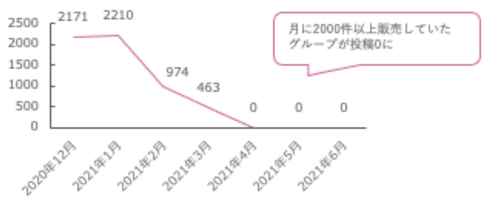
SNSコミュニケーション分析手法を用いた金融犯罪情報の早期検出に関する研究
（研究担当:大原　望乃）

研究背景　フィッシング攻撃とTelegram

- クレジットカードの不正利用被害は年々増加
- その主な手口はフィッシング攻撃
  → 様々な対策が検討されてきた
- フィッシングサイトで窃取されたカード情報がSNS上で売買されている
  → 特に売買の場にされているのが Telegram（テレグラム）というメッセージアプリケーション

先行研究と課題　モニタリング / 個人情報漏洩

- 検索APIを使い，カード情報売買に使われる特徴的な文字列を検知可能なモニタリングツールを整備
- 結果はカード会社に連携し、利用確認・利用停止などで対応



→ モニタリングによる犯罪抑止効果が見られたが，監視用アカウントに紐付けられている電話番号が漏洩し，DoS攻撃を受けた

個人情報の秘匿化／それを保証する手段 が必要

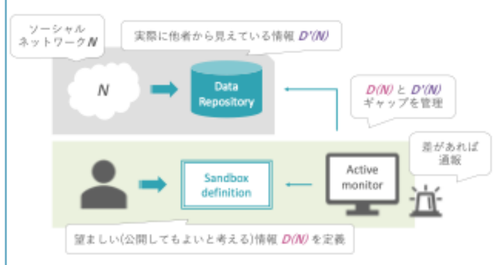
提案　秘匿化と保証 – プライバシサンドボックス

秘匿化：Telethon (Telegram API) を用いて電話番号の公開範囲パラメータを強制変更

→ モニタリング開始前や定期的に実施することで意図せぬ漏洩の防止

保証：プライバシサンドボックスによるプライバシの監視

プライバシサンドボックス
独立した環境下で、ユーザが他者から見られると想定している個人情報の範囲と実際に見られる範囲のギャップを管理する概念機構

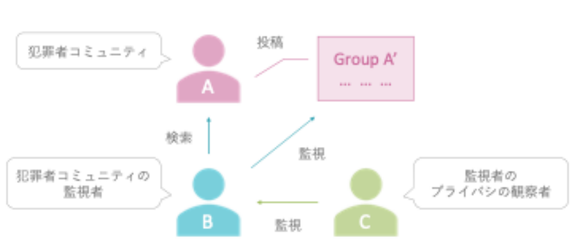


- 今回はそれぞれ以下を採用

N　：Telegram
D(N)　：電話番号の公開設定が非公開であること
D'(N)　：ユーザデータ取得API (GetFullUserRequest) で得られたユーザデータ

実験　APIとプライバシサンドボックスを用いた実験 / 予備実験から見るTelegramの脆弱性

- 3つのアカウントが 5~20分間隔でそれぞれ役割に沿った動作を行うよう設定



- 秘匿化が成功しているか／漏洩した際正常に検知するか をテストするため，以下のパターンについてモニタ

① APIにより電話番号を 非公開（正常状態）
② APIにより電話番号を 公開（漏洩状態）
③ 再度APIにより電話番号を 非公開（正常状態）

予備実験：観察の間隔を1分ごとにし同様の実験を行ったところ，③の直後に電話番号が公開と非公開を繰り返していた
→ ある条件と手順で設定を変更すると，変更した設定が読み込まれない脆弱性を発見
→ プライバシサンドボックスがあればこのような場面でもプライバシの保護が期待できる

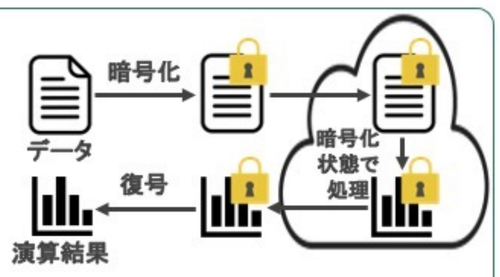
今後の展望

・Telegram以外の他のSNSへの応用　・BOTを用いてより安全面を追求したシステムを構築　等

Paillier暗号を用いたデータベース演算実装方式における 性能解析に関する検討
（研究担当:内藤　華）

研究背景

- ◆ 個人情報などの電子データが第三者のクラウドへ委託されており、情報漏洩や改ざんを防ぐために暗号化が必要
- ◆ 準同型暗号を用いると暗号化状態で加算や乗算が可能であり、第三者の目に触れることなく処理結果を取り出すことができる
- ◆ 暗号化データベースはサイズや処理コストが大きい

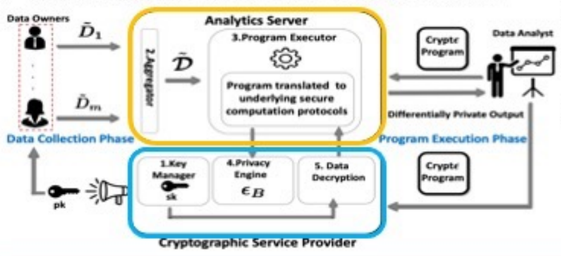


準同型暗号を用いてデータベースを暗号化し、安全で高速なデータベース演算プリミティブを実現することを目指す

関連研究 - Crypt ε

Chowdhury, A. R., et al. (2020).
Crypte: Crypto-Assisted Differential Privacy on Untrusted Servers.

- ◆ 暗号化状態で差分プライバシープログラム実行
- ◆ 加法準同型を持つPaillier暗号により低コスト実現



AS(Analytics Server)：暗号化状態で演算実行
CSP(Cryptographic Service Provider)：公開鍵・秘密鍵の生成や管理

① CSPが公開鍵と秘密鍵生成、データ所有者が秘密鍵で自分のデータを暗号化しASへ送信
② ASがデータ解析者から送られたプログラムを暗号化状態で実行、演算結果をCSPへ送信
③ データ所有者から指定されたプライバシーコストを超えていないことを確認し秘密鍵で復号、差分プライバシーを満たした結果を解析者へ送信

実験概要

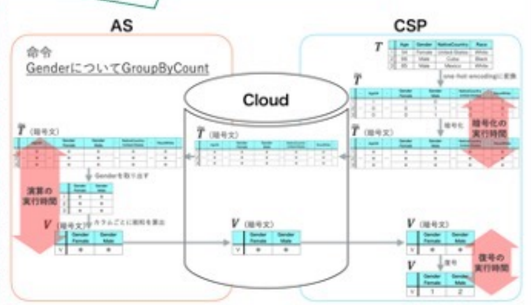
7種類のデータベース演算プリミティブを実装し、処理時間の計測を行った

処理の手順

① CSPがマスタ公開鍵/秘密鍵とユーザ秘密鍵を生成、データをone-hot-encodingに変換した上でPaillier暗号を用いてマスタ公開鍵とユーザ秘密鍵で暗号化しクラウドへ送信
② ASが暗号化状態でプログラム実行、結果をクラウドへ送信
③ CSPが演算結果をマスタ秘密鍵で復号

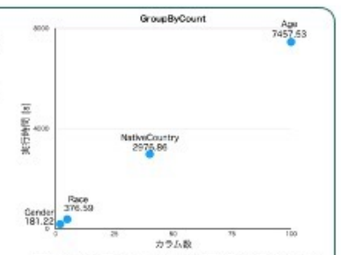
one-hot-encoding
データを0と1のダミー変数で表す
e.g., <22, *Female*> → <[0, ..., 0, 1, 0,..., 0], [1, 0]>
（1の位置：21, 78）

GenderについてGroupByCountする際のデータ暗号化、演算、復号の流れ



実験結果

- ◆ どの演算においてもone-hot-encodingで表現した時のテーブルのカラム数（値の種類の多さ）と処理時間は比例
- ◆ 処理負荷の少ないPaillier暗号を用いることで準同型暗号としては小さいデータサイズと処理時間を実現
- ◆ 暗号文同士の乗算にはASとCSP間での通信が必要であるため処理時間増加
- ◆ 複数の属性の直積を求める演算では処理時間増大
  →直積演算の効率化が必要
- ◆ データの暗号化に850秒程度要する（10000レコードの場合）
  →鍵の生成・管理、暗号化手法、データ構造などの全体的な効率化必要



属性のカラム数（値の種類の多さ）とGroupByCountの実行時間の関係

https://www.is.ocha.ac.jp/~oguchi_lab/　　oguchi@is.ocha.ac.jp