ヘテロジニアス IoT システムにおける暗号プロトコルに関する研究

代表研究者　　酒井　和哉　　東京都立大学大学院システムデザイン研究科　准教授

1 要約

本研究では、多種多様な IoT 端末で構成される IoT システムにおいて、セキュリティを担保するための暗号プロトコルの開発に取り組んだ。具体的には、1）アセット RFID 電子タグを想定したセキュアかつゼロ知識の性質を持つ認証プロトコルと 2）カメラセンサーなどから生成されるデータの完全性を担保するためのデータ検証プロトコルを開発した。提案手法は、シミュレーションによる性能評価と簡易プロトタイプ実装による実験によって、それらの有効性を検証した。

2 RFID ゼロ知識認証プロトコル

本研究では、計算能力が非力な IoT 端末の一種であるアセット RFID を対象として、ゼロ知識認証プロトコルを開発した。平方剰余を応用した認証プロトコルを設計し、プロトコル実行中に秘密鍵に関するいかなる情報も漏洩しない手法を開発した。また安全性に関しては、ランダムオラクルを用いた識別不可能生を証明するとともに、シミュレータによってゼロ知識の性質を担保することを証明した。さらに提案手法をラズベリーパイに実装し、提案手法が認証に要する処理時間を計測した。

2-1 予備知識

（1）平方剰余

群$\mathbb{G}$について、$x^2 = y$ となるような要素$x \in \mathbb{G}$が存在するとき、要素$y \in \mathbb{G}$を平方剰余 （quadratic residues）という。また素数pを法としたアーベル群$\mathbb{Z}_p^*$について、$x^2 = y \bmod p$ となるような要素$x \in \mathbb{G}$が存在するとき、要素$y \in \mathbb{G}$は平方剰余である。$\mathbb{Z}_p^*$内の半分の要素が平方剰余であり、その部分集合をQR_pと表す。素数pを知っていれば、要素yが平方剰余か否かを判定することは簡単であるが、複合モジュロの場合は計算困難な問題となる。

整数Nを大きな素数の積$p \cdot q$とする。このとき要素$y \in \mathbb{Z}_N^*$は$y \bmod p$がQR_pに含まれ、かつ$y \bmod q$がQR_qに含まれるとき、yはQR_Nに含まれる。平方剰余の計算困難性は以下のように定義できる。

Nとy、$\mathbb{Z}_N^*$が与えられたとき、yがQR_Nに含まれるか否かを判定することは困難である。

（2）ゼロ知識証明

証明者Pが検証者Vに対して、プロトコルを介してある秘密情報を知っていることを証明するときに、秘密情報に関する情報をいっさい漏洩しないならば、そのプロトコルはゼロ知識であるという。正規または悪意のある検証者を$\hat{V}$とする。Pと$\hat{V}$がプロトコルを実行したときのトランスクリプト（プロトコル実行時に互いにやり取りしたメッセージのリスト）を$tr_{P,\hat{V}}(\Pi)$とする。ここで、Πはプロトコルパラメータである。本質的に$tr_{P,\hat{V}}(\Pi)$は乱数のリストである。また$M(\hat{V},\Pi)$を多項式時間のプロトコルシミューレータとする。シミュレータで生成されたトランスクリプトを$tr_{M(\hat{V},\Pi)}$とする。

もしシミュレータが実際のプロトコル実行と同じ確率分布のトランスクリプトを生成できるならば、証明者Pを介さずに実際のプロトコルをシミュレーションできる。言い換えるとプロトコル実行時に秘密情報を使用せずにプロトコルをシミュレーションできるため、秘密情報が一切漏洩しない。このようなプロトコルをゼロ知識と呼ぶ。

2-1 関連研究

（１）軽量 RFID 認証プロトコル

電子タグは計算能力が非力であるため、簡単なオペレーションだけで構成する暗号技術を用いる。ハッシュロック[2]と呼ばれる手法では、秘密鍵のハッシュ値を用いることで電子タグを認証する。システム内の電子タグの数をlとした場合、計算速度は$O(l)$となる。

大規模な RFID システムでは、効率的に電子タグを認証するために鍵構造を構造化し、$O(\log l)$ の認証速度を測る。これまでに２分木[3]やグループ構造[4]、スキップリスト[5]、スキップグラフ[6]、K 近傍グラフ[7]を用いた鍵構造が提案されている。しかしながらこれらの手法は、それぞれの鍵が依存するため、一部の電子タグが危殆化されると、他の電子タグの安全性に影響する。

（２）平方剰余を用いた RFID 認証プロトコル

平方剰余を用いた RFID 認証プロトコルは、比較的計算能力が高いアセット電子タグを想定している。基本的なアイディアは文献[8]で提案されているが、リプライ攻撃などの基本的な攻撃に対処できない。また文献[9]で提案された手法は、なりすまし攻撃に対処できない。一方、文献[10]の手法では安全性の担保できるが、暗号プロトコルの性質上、正規の電子タグの認証に失敗することがある。

（３）ゼロ知識証明システム

ゼロ知識の概念が提案されたのは、秘密情報をいっさい漏洩させないインタラクティブな証明システム［11]である。ゼロ知識の概念が RFID セキュリティ分野に適応されたのは文献[12]であるが、悪意のあるリーダに対してゼロ知識を保証するものではない。

したがって、悪意のあるリーダ（検証者）に対してゼロ知識の性質を持つ RFID 認証プロトコルは未だに報告されていない。

2-3 提案手法

（１） 問題の定義

本研究において、IoT システムは図１に示すとおり、複数の電子タグとリーダ、サーバから構成される。それぞれ$T = \{t_1, t_2, \cdots, t_l\}$と$R$、$S$と記述する。また攻撃者は、サーバとリーダ間、リーダと電子タグ間の通信データにアクセスすることができる。

各々の電子タグは、識別子TID_iとn-bit の秘密鍵sk_iに関連付けられており、平方剰余の法を$N_1 = p_1 \cdot q_1$並びに$N_2 = p_2 \cdot q_2$とする。なおN_1とN_2は公開鍵であるが、それらの因数は秘密情報として保管する。

暗号プリミティブとして、暗号論的ハッシュ関数$H: \{0,1\}^* \rightarrow \{0,1\}^n$を定義する。また$Gen$と$Enc$、$Dec$をそれぞれ鍵生成アルゴリズムと暗号化アルゴリズム、復号化アルゴリズムとする。本研究で用いる暗号スキームを$\Pi \coloneqq (Gen, Enc, Dec, H)$とする。

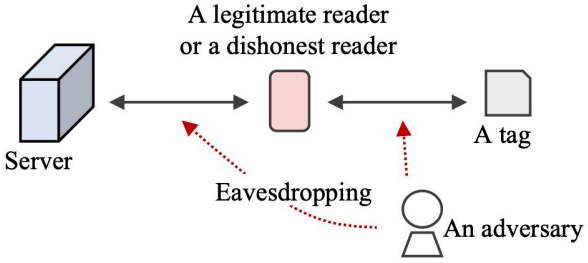

図１．システムアーキテクチャ.

（2）攻撃モデル

本研究では、以下の攻撃を検討した。

- 盗聴（eavesdropping）：攻撃者は、電子タグとリーダ、サーバ間で送受信されるデータを得ることができる。
- 危殆化攻撃（compromise attacks）：攻撃者は、一部の電子タグを危殆化し、関連する秘密情報を得ることができる。
- トレース攻撃（tracing attacks）：攻撃者は、複数のインテロゲーションをまたいで電子タグのリプライをトレースすることができる。
- クローン攻撃（cloning attacks）：攻撃者は、電子タグのリプライをコピーし、異なるセッションで当該電子タグのなりすましを行う。
- 中間者攻撃（man-in-the-middle）：攻撃者は、サーバと電子タグ間の通信データを仲介し、セッションハイジャッキングを実施する。

（3）平方剰余を用いたゼロ知識認証プロトコル（ZKAP）

本研究では、平方剰余の計算困難性に基づき、ゼロ知識認証プロトコルを提案する。提案手法は、初期化フェーズ、認証フェーズから構成される。

初期化フェーズ：サーバSは、二つの大きな素数のペア(p_1, q_1), (p_2, q_2)をランダムに生成し、$N_1 = p_1 \cdot q_1$並びに$N_2 = p_2 \cdot q_2$を計算する。各々の電子タグtについて、識別子TIDと秘密鍵skを初期化する。サーバは、タプル$(sk, TID, N_1, p_1, q_1, N_2, p_2, q_2)$を保管し、電子タグのメモリーに$(TID, sk, N_1, N_2)$を記録する。

認証フェーズ：認証フェーズは、五つのムーブ（エンティティ間でデータを送受信する回数）で構成される。

- ムーブ 1（リーダから電子タグ）：リーダは二つのn-bit の乱数$r_r \in \{0,1\}^n$並びにベクトル$\vec{e} \leftarrow \{e_1, e_2, \cdots, e_c\}$を一様分布で生成する。ここで、$1 \le i \le c$について、$e_i \in \{0,1\}$とする。メッセージ$m_1 \coloneqq \langle r_r, \vec{e} \rangle$を構成し、$m_1$を電子タグ$t$に送信する。
- ムーブ 2（電子タグからリーダ）：電子タグは、リーダから$m_1 \coloneqq \langle r_r, \vec{e} \rangle$を受信する。$n$-bit の乱数$r_t \in \{0,1\}^n$を一様分布で生成し、受信したデータと自身が持つ秘密鍵から$y \leftarrow sk \oplus r_t \oplus r_r$を計算する。もし$y$が$\mathbb{Z}_{N_2}^*$に含まれない場合は、乱数$r_t$の生成からやり直す。次に平方剰余を用いた暗号化を次の通りに行う。

$$U \leftarrow Enc(r_t) \coloneqq r_t^2 \bmod N_2$$
$$Y \leftarrow Enc(y) \coloneqq y^2 \bmod N_2$$
$$K \leftarrow H(TID || sk)$$
$$x \leftarrow ENc(K) \oplus r_t \coloneqq (K^2 \bmod N_1) \oplus r_t$$
$$\vec{b} \coloneqq [b_1, b_2, \cdots, b_c], where\ b_i \leftarrow r_t K^{e_i} \bmod N_1$$

  最後に、$H(r_t)$と$H(y)$を計算し、メッセージ$m_2 \coloneqq \langle U, Y, x, \vec{b}, H(r_t), H(y) \rangle$を構成しる。電子タグ$t$は、$m_2$をリーダ$R$に送信する。

- ムーブ 3（リーダからサーバ）：リーダは、電子タグから$m_2 \coloneqq \langle U, Y, x, \vec{b}, H(r_t), H(y) \rangle$を受信する。ムーブ 1 で生成した情報を加えて、メッセージ$m_3 \coloneqq \langle m_2, r_r, \vec{e} \rangle$を構成し、これをサーバ$S$に送信する。

- ムーブ４（サーバからリーダ）：サーバは、リーダから$m_3 \coloneqq \langle m_2, r_r, \vec{e} \rangle$を受信する。まず$c$個の要素について、$(b_i)^2 = r_t^2(x \oplus r_t)^{e_i} \bmod N_1$（ここで$1 \le i \le c$）を計算し、等式が成立しない$i$が存在すれば、認証を却下する。パスすれば、次に示すとおりに暗号化したデータの復号化を行う。

$$(r_{t,1}, r_{t,2}, r_{t,3}, r_{t,4}) \leftarrow Dec(U)$$
$$r_t \leftarrow r_{t,i}\ s.t. H(r_{t,i}) = H(r) \text{ for } 1 \le i \le 4$$
$$(y_1, y_2, y_3, y_4) \leftarrow Dec(Y)$$
$$y \leftarrow y_i\ s.t. H(y_i) = H(y) \text{ for } 1 \le i \le 4$$
$$sk \leftarrow y \oplus r_t \oplus r_r$$

なお複合平方剰余を用いているため、UとYを復号化したときに、四つの平方剰余の候補が存在する。ムーブ２でr_tと$H(y)$を計算しておくことで、正しい平方剰余を求めることができる。サーバは復元した秘密鍵skがデータベース内に存在するか否かを確認し、存在しなければ却下する。正規の電子タグである場合は、相互認証のためにリプライを生成する。ACK として、$Y_{ack} \leftarrow H(sk||r_t||r_r)$並びに$H(Y_{ack})$を計算し、メッセージ$m_4 \coloneqq \langle H(Y_{ack}) \rangle$を構成する。サーバは、リーダに$m_4$を送信する。またインテローゲーション毎に電子タグの秘密鍵を更新するため、$sk \leftarrow H(TID||r_r||r_t)$を実行する。

- ムーブ５（リーダから電子タグ）：リーダRは、サーバSから$m_4 \coloneqq \langle H(Y_{ack}) \rangle$を受信し、メッセージ$m_5 \coloneqq m_4 \coloneqq \langle H(Y_{ack}) \rangle$をそのまま電子タグ$t$に転送する。タグは$m_5$を受信し、ACK が正しいか否かを判定する。$H(TID \oplus r_t \oplus y)$を計算し、それが受信した$m_5$内の$Y_{ack}$と同じであれば、サーバを認証し、$sk \leftarrow H(TID||r_r||r_t)$を計算することで、自身の秘密鍵を更新する。また異なれば、ACK を却下する。

2-4 性能評価

（１） 安全性とゼロ知識

ランダムオラクルを用いたプライバシー実験を設定し、識別不可能性を示すことによって、提案した認証プロトコルの安全性を証明した。安全性の定義は、RFID 分野で一般的に用いられる指標[5]を用いた。

ゼロ知識に関しては、提案手法の各ムーブで生成されるメッセージと同じ確率分布のデータを生成するシミュレータを設計することで証明する。各々のムーブで生成されるメッセージは、以下の要領で構成する。

- ムーブ１：シミュレータMは悪意のあるリーダRとやり取りをして、r_r'と$\overrightarrow{e'}$を得る。また擬似的な秘密鍵$psk \leftarrow \{0,1\}^n$をランダムに生成する。リーダが生成したr_r'は使用せずに、ランダムに生成した$r_M' \leftarrow_u \{0,1\}^n$をを乱数として使用する。シミュレータはメッセージ$\tilde{m}_1 \coloneqq \langle r_M', \overrightarrow{e'} \rangle$を構成する。
- ムーブ２：シミュレータMは、乱数$r_r' \leftarrow_u \{0,1\}^n$をランダムに生成し、$y' \leftarrow psk \oplus r_t' \oplus r_r'$を計算する。$y' \in \mathbb{Z}_{N_2}^*$でなければ、再度乱数を生成して$y$を計算し、$y' \in \mathbb{Z}_{N_2}^*$となるまで繰り返す。提案手法と同様に$U'$、$Y'$、$K'$、$x'$、$\vec{b}'$、$H(y')$、$H(y')$を計算する。シミュレータはメッセージ$\tilde{m}_2 \coloneqq \left(U', Y', x', \overrightarrow{b'}, H(y'), H(r_t')\right)$を構成する。
- ムーブ３：シミュレータMはリーダRとやり取りをするが、リーダからのデータは破棄する。すでに計算済みのデータから、メッセージ$\tilde{m}_3 \coloneqq \left(U', Y', x', \overrightarrow{b'}, H(y'), H(r_t'), r_M', \overrightarrow{e'}\right)$を構成する。
- ムーブ４：シミュレータMは、$Y_{ack} \leftarrow H(PID) \oplus r_t' \oplus y'$と$H(Y_{ack})$を計算し、メッセージ$\tilde{m}_4 \coloneqq \langle H(Y_{ack}) \rangle$を構成する。

- ムーブ５：$m_5 = m_4$であるため、$\tilde{m}_5 = \tilde{m}_4$とする。最後に乱数$\vec{e}$を一様分布によってランダムに生成する。リーダが生成した$\vec{e'}$と一様分布で生成した$\vec{e}$の全ての要素が同じなるまで、ループ構造によって、ムーブ１からムーブ５の処理を繰り返す。これによってリーダがどのような$\vec{e'}$を生成しようとも、$\vec{e'}$が一様分布になることを保証する。

以上のように、シミュレータで生成した各々のメッセージは、実際のプロトコルで送受信されるメッセージと同じ確率分布を持ち、多項式時間で動作する。そのため提案手法はゼロ知識の性質を持つ。

（２）提案手法の実装と実験結果

本研究では、アセット電子タグとしての機能をラズベリーパイに実装し、提案手法と既存法[10]を実装した。またサーバとリーダの機能は、Ubuntu マシンに実装した。セキュリティーパラメータは$n \in [12,28]$、ベクトル$\vec{e}$の大きさは$c \in [4,9]$と設定した。

実験結果の一例を図２と図３に示す。図２は、公開情報のビット数（X 軸）に対する提案手法の実行速度（Y 軸）を示す。ビット数が大きいほど、セキュリティ強度が高くなるが、それと同時にプロトコルの計算処理が大きくなる。図３はプロトコルの成功率を示す。従来法[10]は一定の確率で認証が失敗するが、提案手法では、ムーブ２で$y \in \mathbb{Z}_{N_2}^*$であることを担保するため、１００％の確率で正規の電子タグを認証することができる。

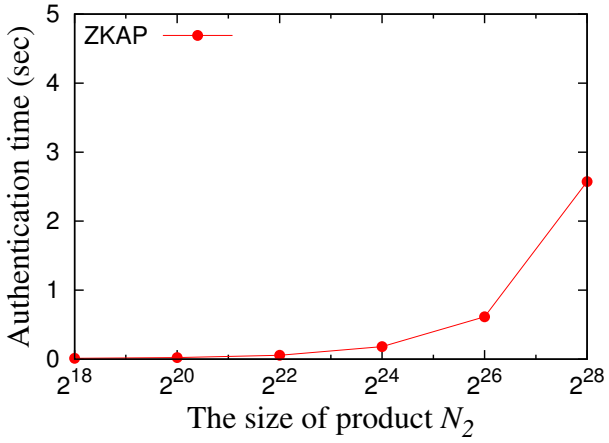

図２．認証プロトコル実行速度.

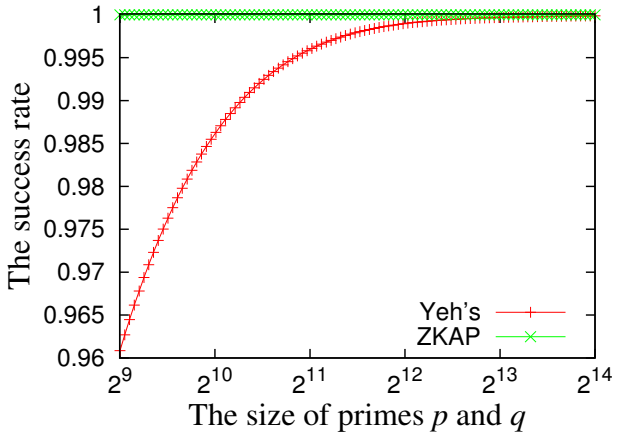

図３．プロトコルの成功率.

3 データ検証プロトコル

本研究課題では、IoT システムのエッジにおいて、IoT 端末が生成するデータの完全性を担保するためのデータ検証プロトコルを開発する。IoT システムにおけるデータ完全生問題の特徴としては、１）センサー機能を持つ IoT 端末はストリーム型のデータを生成する、２）IoT 端末自体は非力なデバイスであるため重たい処理はサーバ側（IoT ブローカー）で実施する、３）複数のセンサー端末がデータを生成するため join オペーレーションをサポートする必要がある、といった点である。これらの問題を解決するために、本研究ではカメレオンハッシュ関数を応用した認証データ構造を提案し、データ検証プロトコルを提案する。

3-1 予備知識

（1） カメレオンハッシュ関数

衝突困難性を持つハッシュ関数を暗号論的ハッシュ関数（Cryptographic hash functions）と呼ぶ。すなわち関数$H:\{0,1\}^* \rightarrow \{0,1\}^n$について、$H(x) = H(y)$となるような異なる入力$x$と$y$を計算することは困難であるという。

またトラップドア付きの暗号論的ハッシュ関数をカメレオンハッシュ関数（Chameleon hash functions）と呼ぶ。ハッシュ関数Chに関する秘密情報（トラップドア）を持つ場合にのみ、衝突を発見することができる。すなわちトラップドアcskが与えられた時、メッセージxとあるランダムな値r 、さらに別のメッセージ x' について、カメレオンハッシュアルゴリズムChを用いて、$Ch_{csk}(x,r) = Ch_{csk}(x',r')$となるような異なる入力$r'$を効率的に計算することができる。

（2） カメレオン認証木（Chameleon Authentication Tree）

カメレオン認証木（CAT: Chameleon Authentication Tree）は［13］にて提案された認証可能なストリーミングプロトコルのデータ認証手法として提案されたデータ構造である。CAT は根と全ての右ノードがカメレオンハッシュ関数で、また、全ての左ノードが衝突困難なハッシュ関数で計算された、葉の要素数が事前に決められていないマークル木である。

文献[13]において提案されているストリーミングデータ検証プロトコル VDS において、CAT は認証のためのデータ構造として利用されている。サーバ側では CAT 全体を保持していることに対し、クライアント側ではある要素に対する CAT の認証パスのみを保持することで対数オーダーでの要素の検証を行うことができ、かつクライアント側ではストリーミングデータすべての保持をする必要がなく、計算資源の少ない環境でもストリーミングデータの検証を行うことができる。

3-2 関連研究

本研究と最も密接に関連する研究は、IoT アプリケーションにおけるデータの検証である。VERID［14]は、IoT サービスのための検証可能なデータ管理システムである。この研究では、プレフィックス木とマークル木を統合した PrefixMHT を認証のためのデータ構造として用いることで、データの利用者が範囲選択・集計クエリを行うことができるようになっている。文献[15] では、ブロックチェーンベースのデータ整合性サービスのフレームワークが提案されており、スマートコントラクトを実装することで、第三者に頼ることなくデータの整合性を確保することができている。文献[16]では、ウェアラブルデバイス向けのデータ完全性スキームを提案しており、コンピュータのリソースを大量に消費する計算を IoT 対応のゲートウェイに委任することで、公開鍵基盤ベースの暗号化行うことができる。

また、データの完全性を保証するもう一つのアプローチは、MAC を使用することである。例えば、EPPDA［17］では、同形暗号と同形 MAC を組み合わせて、IoT ベースのヘルスケアアプリケーションにデータの完全性を提供している。しかしながら、これらの既存のデータ検証スキームはストリーミング形式のデータを保持することを想定していないため、ストリーミングデータアプリケーションには適用することができない。

3-3 提案手法

（1）問題定義

本研究で想定する IoT システムは、IoT 端末（センサーなど）と IoT ブローカー（サーバ)、クライアント（パソコン、スマホなど）で構成される。IoT 端末はデータを生成し、それらを IoT ブローカーに送信する。IoT ブローカーは生成されたデータの認証データ構造を構成し、データの完全性を担保する。IoT ブロ

ーカーは、クライアントからのクエリーを送信することによって、データサービスを提供する。

（2）攻撃モデル

データ検証プロトコルの安全性については、選択メッセージ攻撃（Chosen message attacks）に対して偽装不可能生（Unforgeability）に基づいて実証する。また偽装不可能生については、どのようなメッセージでも良いので、正規の認証コードを生成することを目的とした、存在的偽装不可能生（Existential forgery）を想定する。当該攻撃モデルでは、攻撃者はセキュリティパラメータ（鍵の長さ）とオラクル（クエリーに対してメッセージ認証コードを返す）を与えられ、自ら選択したメッセージに対応する認証コードを学習することができる。オラクルにメッセージを送信することができる回数は多項式回数に制限される。攻撃者は学習後に任意のメッセージとそれに対応する正規の認証コードを出力する。

提案プロトコルの安全性に関しては、ランダムオラクルを用いて、秘密鍵を持たない攻撃者がデータを注入し、セグメントCATに認証コードを追加と認証コードの改ざんができないことを証明する。

（3）セグメントCATデータ構造

本研究では、データ検証のためのデータ構造として、セグメントCAT を提案する。これは、前述の CAT をセグメント木に応用したものである。セグメント木は完全二分木であり、各節点によって区間を管理するデータ構造である。根は区間全体を管理し、各節点の子は親の区間を二等分した２つの区間の片方を管理する。これにより、区間に対する操作をn個の要素に対して、$O(\log n)$時間で行うことができるという特徴がある。このセグメント木において、根と全ての右ノードをカメレオンハッシュ関数で、また、その他全ての左ノードを衝突困難なハッシュ関数で計算する。これにより、木の深さをDとした場合に、2^D個の要素を認証することができ、またデータの消費者から送信される各区間クエリについて、$O(\log n)$で応答することができる。各節点は、値とその値をもつ葉のインデックスを管理することで、データ消費者からの要求に対して、認証パスを計算することができる。

セグメントCATの主なオペレーションは以下のとおりである。

- セグメントCATの生成：$gen(1^\lambda, D)$
  セグメント CAT は入力にランダムなセキュリティパラメータλと木の深さを表すDを受け取り秘密鍵spと検証鍵vpを返却する。
- 要素の追加： $addLeaf(sp, v)$
  要素の追加を行うには、秘密鍵spと新たに葉に追加する値vを入力として受け取り新たな秘密鍵sp'、葉のインデックスi、認証パス$aPath$を出力する 。
- 要素の認証：$verify(vp, i, v, aPath)$
  要素の認証を行うには、検証鍵vpと葉のインデックスi、葉の値v、認証パス$aPath$を入力として受け取り、葉i番目の要素がvであると検証できた場合$true$を、そうでなければ$false$を返却する。

（4）セグメントCATを用いたデータ検証プロトコル

セグメントCATをデータ構造としてデータ検証プロトコルを開発する。 IoT 環境において、データの消費者が一度にメモリ上に保存することができない長さのストリーミングデータを扱うことを目的としている。

（4－1）データベースの初期化

サーバはデータベースの初期化に際して、セキュリティパラメータ1^λ、木の深Dを受け取り、$gen(1^\lambda, D)$を実行する。その結果として、秘密鍵 $sp = (csk, csk_1, st)$ と検証鍵 $vp = (cpk, cpk_1, \rho)$ が出力される。ここでρは初期の空の木の頂点を表している。その後、クライアントは秘密鍵spをサーバは検証鍵vpを保持し、

空のデータベースを作成する。

（4－2）要素をデータベースに追加するとき

サーバ上のデータベースに要素を追加する際、クライアントは $addLeaf(sp, v)$ を自身の端末上で実行し秘密鍵sp'、葉のインデックスi、認証パス$aPath$を得る。その後、i、$aPath$、新たに追加する要素の値v、をサーバに送付する。サーバは$aPath$を受け取り、それを自身のセグメント CAT に新たなランダムな値Rと共に追加する。

（4－3）要素を取得するとき

サーバ上のデータベースからクライアントが要素を取得する際、クライアントは求める要素のインデックスiをサーバに送付する。サーバは該当する葉の値vと認証パス$\mathrm{aPath_v}$を返却する。クライアントは、$verify(vp, i, v, aPath)$を実行し、該当する要素が変更や改ざんされていないことを確認する。

（4－4）要素を更新するとき

サーバ上のデータベースを更新する際、初めにクライアントは セグメント CAT のトラップドアsp、状態 st、トラップドアが適用されていない値のペア $(x_{i,j}, r_{i,j})$ を取得する。続いて更新したい要素のインデックスiをサーバに送付し、サーバから更新前の要素 v_{pre} と認証パス $aPath_{pre}$ を取得する。クライアントは $verify(sp, i, v_{pre}, aPath_{pre})$ を実行し要素が正しいものであることを検証する。要素が正当なものであった場合、葉の値を更新し新たな認証パスを計算し新たな根 ρ'を得て状態stを更新する。最後にクライアントは新たな認証パス $aPath_i'$、新たな要素 v_{new}、更新後の検証鍵 vp' をサーバに送付する。サーバは受け取った認証パスの正当性を確認した後、正当なものであった場合は要素と自身の検証鍵の更新を行い、正当でなかった場合は更新をせずに処理を終了する。

3-4 性能評価

本研究では、提案手法の効率性を検証するために、シミュレーションを行った。提案手法であるセグメント CAT を MacBook Pro 13 インチ（Core i7）に実装し、クエリ処理を発生させて、処理時間を計測した。生成したデータ数は、10 個から 100 万個とした。

図１に範囲最小値クエリに対するシミュレーション結果を示す。縦軸には応答時間、横軸には葉の要素数が表されている。図４に示すとおり、要素数が 1000 程度までは、従来手法である CAT は応答時間が短く、要素数が 1000 を超えると、提案手法の応答時間が短くなっていることがわかる。これは、要素数が 1000 程度までは、従来の手法のように葉の要素全てを巡回し最小値を計算した場合の方がセグメント木において区間クエリを処理することよりも効果的であることに起因する。しかし、提案手法において想定している、データの消費者がメモリ上に保存しておくことができない量のデータ、例えば 100 万データなどの場合は、従来手法の約 1000 倍速くクエリに応答することができていることがわかる。範囲最大値クエリや平均値を求めるようなクエリに対しても同様の結果を得ることができている。

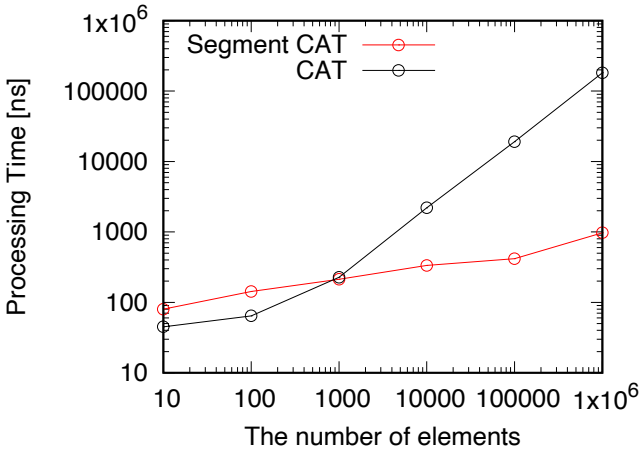

図４．範囲最小値クエリに対する応答時間.

4 まとめ

本研究では、ヘテロジニアス IoT システムにおける暗号プロトコルの研究に取組んだ。まずアセット RFID 電子タグを想定した認証プロトコルの開発に取組んだ。平方剰余を用いることによってゼロ知識の性質を持つプロトコルを設計することに成功した。また提案手法の安全性を証明するとともに、センサー端末（ラズベリーパイ）に提案手法を実装・実験することで、その有効性を示した。研究成果は電子情報通信学会の ISEC 研究会で発表するとともに、IEEE Internet of Things Journals に採択された。

またデータ検証プロトコルの開発にも取り組んだ。提案手法では、トラップドア付きのハッシュ関数であるカメレオンハッシュ関数を用いてデータ認証構造を設計し、ストリームデータの完全性を保証する手法を設計した。提案手法は、従来の手法ではサポートしていない join オペレーションを含む範囲最小クエリを対数時間で処理できることを示した。またミュレーションによって性能評価を行い、実時間でも十分な性能を有することを示した。今後は、提案手法をサーバに実装し、カメラセンサーから生成されたストリームデータからデータ認証構造を生成する簡易プロトタイプを実装する。さらに今年度中に、研究成果を国内研究会で発表するとともに、IEEE などの国際論文誌に論文を投稿する。

【参考文献】

[1] S. A. Ahson and M. Ilyas, RFID Handbook: Applications, Technology, Security, and Privacy. CRC press, 2017.

[2] S. A. Weis, “Security and Privacy in Radio-frequency Identification Devices,” Ph.D. dissertation, Massachusetts Institute of Technology, 2003.

[3] D. Molnar and D. Wagner, “Privacy and Security in Library RFID: Issues, Practices, and Architectures,” in CCS, 2004, pp. 210–219.

[4] M. E. Hoque, F. Rahman, and S. I. Ahamed, “Anonpri: An Efficient Anonymous Private Authentication Protocol,” in PerCom, 2011, pp. 102– 110.

[5] M.-T. Sun, K. Sakai, W.-S. Ku, T. H. Lai, and A. V. Vasilakos, “Private and Secure Tag Access for Large Scale RFID Systems,” IEEE Trans. Dependable Secure Comput., vol. 13, no. 6, pp. 657–671, 2015.

[6] Y. Komori, K. Sakai, and S. Fukumoto, “Fast and Secure Tag Authentication in Large-scale RFID Systems Using Skip Graphs,” Comput. Commun., vol. 116, pp. 77–89, 2018.

[7] K. Sakai, M.-T. Sun, W.-S. Ku, and T. H. Lai, “On The Performance Bound of Structured Key-based RFID Authentication,” in PerCom, 2019, pp. 1–10.
[8] Y. Chen, J.-S. Chou, and H.-M. Sun, “A Novel Mutual Authentication Scheme Based on Quadratic Residues for RFID Systems,” Comput. Netw., vol. 52, no. 12, pp. 2373–2380, 2008.
[9] T. Cao, P. Shen, and E. Bertino, “Cryptanalysis of Some RFID Authentication Protocols,” J. Commun., vol. 3, no. 7, pp. 20–27, 2008.
[10] T. Y. Yeh TC, Wu CH, “Improvement of The RFID Authentication Scheme Based on Quadratic Residues,” Comput. Commun., vol. 31, pp. 337–341, 2011.
[11] S. Goldwasser, S. Micali, and C. Rackoff, “The Knowledge Complexity of Interactive Proof Systems,” SIAM J. Sci. Comput., vol. 18, no. 1, pp.186–208, 1989.
[12] H. Liu and H. Ning, “Zero-knowledge Authentication Protocol Based on Alternative Mode in RFID Systems,” IEEE Sens. J., vol. 11, no. 12, pp. 3235–3245, 2011.
[13] D. Schroder and H. Schroder. “Verifiable Data Streaming,” In ACM CCS, pp. 953-964, 2012.
[14] X. Li, M. Wang, S. Shi, and C. Qian, “VERID: Towards Verifiable IoT Data Management,” In IoTDI, pp. 118-129, 2019.
[15] B. Liu, X. Yu, S. Chen, X. Wu, and L. Zhu, “Blockchain Based Data Integrity Service Framework for IoT Data,” In ICWS. pp. 468-475, 2017.
[16] Ch. Doukas, I. Maglogiannis, V. Koufi, F. Malamateniou, and G. Vassilacopoulos, “Enabling Data Protection through PKI Encryption in IoT m-Health Devices,” In BIBE, 2012. p. 25-29. 2012.
[17] F. Almalki; SOUFIENE and B. Othman, “EPPDA: An Efficient and Privacy-Preserving Data Aggregation Scheme with Authentication and Authorization for IoT-Based Healthcare Applications,” In WCMC, 2021.

〈発　表　資　料〉

題　　名	掲載誌・学会名等	発表年月
Quadratic Residues-Based Private Authentication for RFID Systems	電子情報通信学会・ISEC 研究会	2020 年 11 月.
An RFID Zero-knowledge Authentication Protocol based on Quadratic Residues	IEEE Internet of Things Journals	2021 年度採択.