様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

令和　６ 年　 ５ 月　 ７ 日現在

機関番号：１５４０１
研究種目：基盤研究(C)（一般）
研究期間：2020～2023
課題番号：２０Ｋ１１８１１

研究課題名（和文）暗号アルゴリズムの代理変更が可能な共通鍵暗号型プロキシ再暗号化方式の開発応用研究

研究課題名（英文）Development and Application Research on a Proxy Re-encryption Scheme for Symmetric Key Cryptography with Support for Cipher Algorithm Delegation

研究代表者

渡邉　英伸（WATANABE, HIDENOBU）

広島大学・情報メディア教育研究センター・准教授

研究者番号：８０６１１６３１

交付決定額（研究期間全体）：（直接経費）　　3,300,000円

研究成果の概要（和文）：オンラインストレージ上の暗号文を復号することなくサーバ側で暗号アルゴリズムの変更を可能にする共通鍵暗号型のプロキシ再暗号化方式を提案し，再暗号化処理委託システムの開発を行った．ハードウェアセキュリティ機構Intel Software Guard Extension（Intel SGX）を活用することで，利用者が再暗号化処理を委託する際の処理に要する時間を約1秒程度に抑えつつ，暗号文を復号することなく暗号アルゴリズムの変更ができることも明らかにした．これにより，再暗号化処理の効率化および利用者の負担軽減の両立を示すことができた．

研究成果の学術的意義や社会的意義
本研究成果により，データを秘匿したまま利用者自身が自身の暗号文をいつ，どこで，再暗号化処理をさせるかを制御することが可能になるため，プライバシーやデータの安全性を強化できる．また，蓄積され続ける研究データの機密性を長期的に保護することが求められる研究データ管理の観点においても活用が期待できる．さらに，Society5.0の実現に向けて信頼のある自由なデータ流通のプラットフォームに展開も見込める点で社会的意義は大きい。

研究成果の概要（英文）：We proposed a proxy re-encryption scheme for symmetric key cryptography-based that enables the change of encryption algorithms on the server side without decrypting the ciphertext and developed a re-encryption processing delegation system. We clarified that the time required for users to delegate re-encryption processing can be kept to about one second by utilizing the hardware security mechanism Intel Software Guard Extension (Intel SGX), while changing the encryption algorithm without decrypting the ciphertext. This result shows the compatibility of improving the efficiency of re-encryption processing and reducing the burden on users.

研究分野：情報科学

キーワード：エッジコンピューティング　プロキシ再暗号化　Intel SGX

様　式　Ｃ－１９、Ｆ－１９－１（共通）

１．研究開始当初の背景

クラウドサービスの責任共有モデルにより、利用者がデータのセキュリティを確保することが要求され、長期データ保存に対するセキュリティ維持管理の重要性が増している。もし、暗号アルゴリズムの変更が求められる事態（暗号アルゴリズムの脆弱性が発見された場合、サービスの運用上のルール変更により鍵長やアルゴリズムを更新しなければならない場合など）が生じた場合、利用者はクラウドにあるストレージサーバ上の暗号文を一度端末にダウンロードしてから復号し、平文を取り出した後に新しいアルゴリズムで再度暗号化してアップロードする（以降、素朴な方法と呼ぶ）ことで暗号文の安全性は高度に保たれる。しかしながら、膨大な暗号文にこのような手段を用いると、多量のデータ出し入れと作業時間を必要とし利用者に甚大な負担を強いる。

暗号文を復号することなく別の利用者の暗号文に変換できる技術としてプロキシ再暗号化技術が知られている。公開鍵暗号の一種として提案され、再暗号化鍵の提供により第三者に復号権限の委譲やサーバに再暗号化処理の委託も可能である。プロキシ再暗号化技術の特性をうまく活用することができれば、データの出し入れと利用者の作業負担の軽減が期待できる。一方、プロキシ再暗号化は暗号鍵の変更を主目的としており、暗号アルゴリズムの変更を目的とする方式は殆ど無く、低速な公開鍵暗号に対する方式が理論的に検討されているのみである。実用的には高速に処理が可能である共通鍵暗号型のプロキシ再暗号化が Amazon Web Services 等で利用されているが、これらは自身の鍵を更新する目的であるため暗号アルゴリズムの更新はできない。また、プロキシ再暗号化の研究では、再暗号化鍵や暗号化ファイルの送受信の負荷を軽減するため、Intel SGX に代表されるハードウェアセキュリティ機構の活用に取り組んだ事例も無い。

２．研究の目的

本研究では、暗号化状態を維持したまま復号時の暗号アルゴリズムの変更を可能にする共通鍵暗号型プロキシ再暗号化方式ならびにストレージサーバに提案方式を採用した再暗号化処理委託システムを提案する。加えて、Intel SGX の特性を活用した再暗号化処理委託システムの改善手法および有用性について評価する。これにより、ストレージサーバ上の膨大な暗号文に対する再暗号化処理の効率化および利用者の負担軽減を両立し、長期的なデータセキュリティの維持管理を進展させることを目的とする。

３．研究の方法

本研究では、暗号アルゴリズム変換型の共通鍵暗号型プロキシ再暗号化処理委託システムを開発した。具体的には(1)～(3)の内容を実施した。

(1) 暗号アルゴリズム変換型共有鍵プロキシ再暗号化方式の提案

既存の共通鍵暗号型プロキシ再暗号化技術では、ストリーム暗号の暗号化処理で用いる擬似乱数を別の秘密鍵（共通鍵）から生成された擬似乱数に置き換えることによって、異なる利用者に復号させることを実現している。本研究は、擬似乱数の置き換えによって再暗号化を実現していることに注目し、更新対象を共通鍵ではなく暗号アルゴリズムとして更新できるように改良した。

ストリーム暗号は疑似乱数生成器を利用する方式である。共通鍵 K と初期化ベクトル IV を疑似乱数生成器 f()の入力情報としたとき、疑似乱数列 Z=f(K,IV)を得る。平文を M、暗号文を C としたとき、暗号化は C=M⊕f(K,IV)=M⊕Z となり、復号は M=C⊕Z=M⊕Z⊕Z となる。あるストリーム暗号の疑似乱数生成器を関数 f()と表記するとき、異なるストリーム暗号の疑似乱数生成器を別の関数 f'()を用いた方式と定義する。このとき、共通鍵 K で暗号化された暗号文の暗号アルゴリズムを f()から f'()に変換したい場合、それぞれの関数から生成される疑似乱数列 Z、Z'から再暗号化鍵 Z⊕Z'を生成すればよい。

変更前の暗号アルゴリズムで作成した暗号文 C=M⊕f(K,IV)=M⊕Z から更新後アルゴリズムで作成した暗号文 C'=M⊕f'(K,IV)=M⊕Z'へ変換するには、再暗号化鍵を用いて C'=C⊕(Z⊕Z')=M⊕Z⊕(Z⊕Z')=M⊕Z'のように計算する。このように鍵を変更するときや暗号アルゴリズムを変更するときも本質的には疑似乱数列の置き換えであることに注目して暗号アルゴリズムの変換機能を実現した。今回の提案方式はストリーム暗号用であるが、CTR モードなどを用いることでブロック暗号のアルゴリズムにも対応可能である。なお、暗号アルゴリズムの変換と同時に共通鍵の更新をすることも原理的には可能である。

(2) 提案方式を用いた再暗号化処理委託システムの開発

ストレージサーバに提案方式を採用した再暗号化処理委託システムの概要を図 1 に示す。

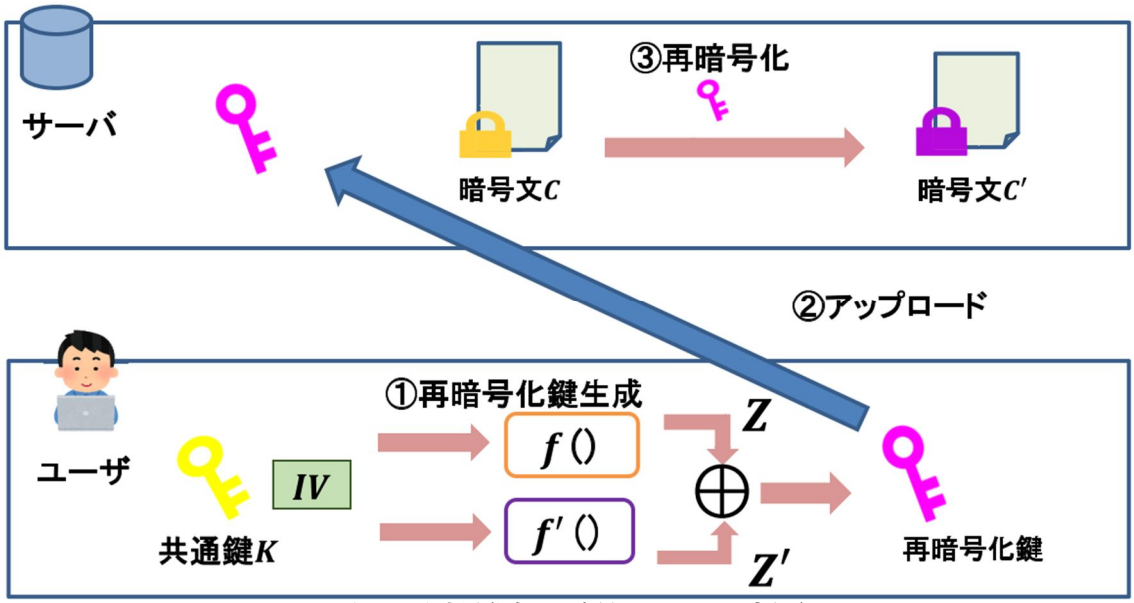

図 1　再暗号化処理委託システムの概要

提案方式で暗号アルゴリズムの更新を行う場合の具体的な手順を示す。ユーザは自身の共通鍵を用いて更新前の暗号アルゴリズムと更新後の暗号アルゴリズムから生成した疑似乱数列の組を排他的論理和することで再暗号化鍵を生成し、サーバに送信する。サーバは再暗号化鍵と暗号文を排他的論理和することで更新後の暗号アルゴリズムで暗号化された暗号文に変換する。本研究では、評価実験のために RC4 で暗号化された暗号文を AES-CTR モードで再暗号化する実装を行った。変換する共通鍵暗号に関しては、RC4 は独自実装、AES-CTR モードは OpenSSL ライブラリを用いて共通鍵 K に対応する疑似乱数列を得ている。ユーザとサーバ間の送受信については cURL を使用し、これらの通信は HTTPS で行っている。

(3) Intel SGX を活用した再暗号化処理委託システムの改善

図 1 に示した提案システムでは、使用するプロキシ再暗号化方式の性質により、再暗号化時に送信する再暗号化鍵が暗号文のサイズに依存してしまい、暗号文のサイズの増加にともないアップロードに要する処理時間が増大することが明らかになった。そこで、本研究では Intel によって提供されるハードウェアセキュリティ機構である Intel SGX を用いてストレージサーバ上の安全な領域にユーザの共通鍵を配送し、再暗号化演算を実施できることから再暗号化時に送信するデータサイズを暗号化鍵と同程度まで小さくする改善を試みた。Intel SGX を用いた再暗号化処理委託システムの概要を図 2 に示す。

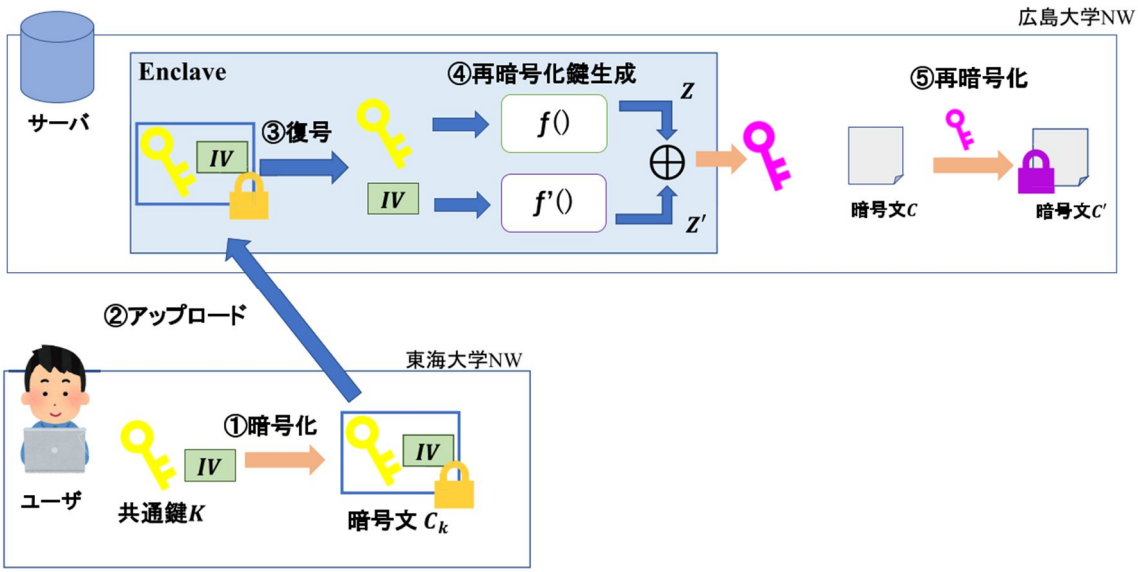

図 2　Intel SGX を用いた再暗号化処理委託システムの概要

まずユーザとサーバで Remote Attestation を行い、Intel SGX を搭載したサーバの CPU と暗号化されたメモリ領域（Enclave）の完全性を検証したうえで、ユーザと Enclave 内で同じ秘密情報を共有する。次にユーザは再暗号化に用いる共通鍵 K を平文と見なし、Remote Attestation

によって共有された秘密情報を鍵として共通鍵暗号で暗号化する。この暗号文はサーバに送信された後に Enclave に入力され、Enclave 内部で Remote Attestation によって共有された秘密情報により復号することで共通鍵 K が Enclave 内にセットされる。さらに Enclave 内で共通鍵 K を用いて更新前の暗号アルゴリズムと更新後の暗号アルゴリズムを実行し、それぞれのアルゴリズムの疑似乱数列の排他的論理和をとることで再暗号化鍵を生成し、Enclave 外に出力する。最後に再暗号化鍵とサーバに保存されている暗号文を排他的論理和することで、暗号文を更新後の暗号アルゴリズムに対応したものに変換する。

本実装では Remote Attestation のサンプルコードをベースに実装を行った。また Intel SGX は Intel SGX Driver、Intel SGX SDK、Intel SGX PSW で構成している。アルゴリズム更新に用いる共通鍵の暗号化には、OpenSSL ライブラリの AES-GCM モードを用いている。送信を行う場合は、Intel が提供する関数である msgio->send()を用い、受信する場合は msgio->read()を使用する。暗号化された共通鍵は Enclave 内で Intel SGX SDK が用意する AES-GCM モードで復号される。具体的には sgx rijndael128GCM decrypt()という Enclave 内で使用される Intel SGX SDK の関数を用いている。RC4 の処理は独自で実装を行った。再暗号化鍵生成を行う場合、Enclave 内で AES-CTR モードと RC4 疑似乱数列を用いて、再暗号化鍵を生成する。AES-CTR モードについては sgx aes ctr encrypt()という関数を使用した。Enclave 内の動作は EPC サイズとして 96 MB の制限があり、巨大なサイズのデータを扱うことができない。そのため本実装では Enclave 外のメモリ上に再暗号化鍵格納用の配列を確保し、ECALL を用いて再暗号化鍵格納用の配列のアドレスを Enclave 内に渡す。次に Encave 内で 10KB など Enclave のサイズ制限より小さいサイズずつ再暗号化鍵を生成し、再暗号化鍵格納用の配列のアドレスに該当する場所に都度書き込むように実装している。

４．研究成果

評価実験では、再暗号化するデータサイズを変化させて素朴な方法と提案手法ならびに改善手法における暗号アルゴリズムの更新に係る処理時間を評価し、ユーザが要する処理時間をどの程度短縮できるかを確認した。ユーザの機器諸元は、Intel core i3、 4GB RAM、 ubuntu OS を搭載し、Openssl や Intel SGX SDK などのライブラリを導入した機器である。サーバは、Intel Xeon E-2224G、 8GB RAM、 ubuntu OS を搭載し、こちらも Openssl や Intel SGX SDK などのライブラリを導入した機器となっている。今回、ユーザ機器は東海大学のネットワーク、サーバ機器は広島大学のネットワークに配置し、アクセス制限を行った上でインターネット通信が可能になっている。評価実験として、1MB、10MB、100MB、200MB、300MB のファイルを用意し、素朴な方法、提案手法、改善手法の処理時間を計測した。計測では再暗号化を 100 回実行した平均時間を求めた。各結果を表 1、表 2、表 3 に示す。

表 1　素朴な方法の処理時間[sec]

	ユーザ				合計
	ダウンロード	復号	暗号化	アップロード	
1MB	0.31049	0.00548	0.00104	0.42968	0.74668
10MB	1.15173	0.03294	0.00714	2.37526	3.56707
100MB	9.70914	0.31280	0.07015	21.70935	31.80145
200MB	19.08252	0.62545	0.14191	43.54734	63.39722
300MB	28.44144	0.93843	0.21159	64.57598	94.16744

表 2　提案手法の処理時間[sec]

	ユーザ			サーバ	合計
	再暗号鍵生成	アップロード	計	再暗号化処理	
1MB	0.02415	0.42110	0.44526	0.00254	0.44780
10MB	0.16768	2.36335	2.53103	0.00260	2.55563
100MB	1.66838	21.70853	23.37691	0.23273	23.60964
200MB	3.33335	43.22398	46.55734	0.46033	47.01767
300MB	5.00456	64.55509	69.55965	0.68730	70.24695

表 3　Intel SGX を用いた改善手法の処理時間[sec]

	ユーザ				サーバ			合計
	暗号化	アップロード	Remote Attestation	計	復号+再暗号化鍵生成	再暗号化処理	計	
1MB	0.00002	0.00003	1.63253	1.63258	0.00590	0.00098	0.00688	1.63946
10MB	0.00002	0.00003	1.65508	1.65508	0.04476	0.00807	0.05283	1.70796

100MB	0.00002	0.00003	1.63800	1.63805	0.42369	0.07950	0.50319	2.14124
200MB	0.00002	0.00003	1.66180	1.66185	0.83555	0.15897	0.99452	2.65637
300MB	0.00002	0.00003	1.66401	1.66406	1.23808	0.23494	1.47302	3.13708

素朴な方法は、ユーザの機器側にダウンロード・復号・暗号化・アップロードの処理を有し、暗号文サイズが 100MB であれば合計処理時間が 30 秒以上かかることが分かった。

提案手法では、ユーザの機器側の処理は再暗号化鍵生成・アップロードだけとなり、サーバ機器側の処理に再暗号化処理を実施させることで、暗号文サイズが 100MB の際は合計処理時間が約 23 秒となり、3/4 倍程度まで削減できることが確認できた。一方で、(3)で述べた通り、暗号文のサイズの増加にともないアップロードに要する処理時間が増大する傾向も明らかになった。

改善手法では、ユーザ機器側の処理は暗号化・アップロード・Remote Attestation となるが、アップロードの処理を含めても暗号文のサイズに依存することなく約 1.6 秒の一定した処理時間を達成していることを確認した。サーバ機器側の処理においても、復号+再暗号化鍵生成・再暗号化となるものの 1 秒程度の処理時間であることを確認した。合計処理時間についても暗号文サイズが 300MB だったとしても数秒程度であり、改善手法によって 1/20 倍以上の削減効果を得ることができ有効性を示すことができた。

これらの結果より、本研究の目標であった膨大な暗号文に対する再暗号化処理の効率化および利用者の負担軽減の両立は達成できたと考える。本研究成果により、データを秘匿したまま利用者自身が自身の暗号文をいつ、誰に、再暗号化処理をさせるかを制御することが可能になるため、プライバシーやデータの安全性を強化できる。また、蓄積され続ける研究データの機密性を長期的に保護することが求められる研究データ管理の観点においても活用が期待できる。さらに、Society5.0 の実現に向けて信頼のある自由なデータ流通のプラットフォームに展開も見込める点で社会的意義は大きい。

＜引用文献＞

① 西平 侑磨、 鈴木 達也、 渡邉 英伸、 大東俊博、“ アルゴリズム変換型共通鍵プロキシ再暗号化とその実装、” 情報処理学会論文誌、 vol.64、 no.9: pp.1241-1254 2023 年 9 月.

５．主な発表論文等

〔雑誌論文〕　計7件（うち査読付論文　6件／うち国際共著　0件／うちオープンアクセス　6件）

１．著者名 西平 侑磨，鈴木 達也，渡邉 英伸，大東俊博	４．巻 64
２．論文標題 アルゴリズム変換型共通鍵プロキシ再暗号化とその実装	５．発行年 2023年
３．雑誌名 情報処理学会論文誌	６．最初と最後の頁 1241-1254
掲載論文のDOI（デジタルオブジェクト識別子） 10.20729/00227592	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 －

１．著者名 Yuma NISHIHIRA, Takuya ISHIBASHI, Yoshio KAKIZAKI, Toshihiro OHIGASHI, Hidenobu WATANABE, Tohru KONDO and Reiji AIBARA	４．巻 -
２．論文標題 Development of the Edge Computing Platform based on Modular Architecture using Intel SGX	５．発行年 2023年
３．雑誌名 Proc. the 18th Asia Joint Conference on Information Security (AsiaJCIS 2023)	６．最初と最後の頁 59-65
掲載論文のDOI（デジタルオブジェクト識別子） 10.1109/AsiaJCIS60284.2023.00020	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 －

１．著者名 石橋拓哉，鈴木智也，大東俊博，土田 光，金岡 晃，柿崎 淑郎，相原 玲二	４．巻 16
２．論文標題 複数組織対応属性ベース暗号を用いたファイル共有システムの設計	５．発行年 2024年
３．雑誌名 東海大学紀要情報通信学部	６．最初と最後の頁 11-20
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 －

１．著者名 石橋拓哉，小林海，大東俊博，土田 光，金岡 晃，柿崎 淑郎，相原玲二	４．巻 64
２．論文標題 複数組織対応属性ベース暗号を用いたファイル共有システムの実現可能性に関する考察	５．発行年 2023年
３．雑誌名 情報処理学会論文誌	６．最初と最後の頁 670-686
掲載論文のDOI（デジタルオブジェクト識別子） 10.20729/00225261	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 －

1．著者名 Ema Shuntaro、Sato Yuta、Emura Keita、Ohigashi Toshihiro	4．巻 無
2．論文標題 Implementation and Evaluation of an Identity-Based Encryption with Security Against the KGC	5．発行年 2021年
3．雑誌名 Proc. CANDAR Workshops 2021	6．最初と最後の頁 320-325
掲載論文のDOI（デジタルオブジェクト識別子） 10.1109/CANDARW53999.2021.00061	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 －

1．著者名 Tatsuya SUZUKI, Keita EMURA, Toshihiro OHIGASHI, and Kazumasa OMOTE	4．巻 無
2．論文標題 Verifiable Functional Encryption using Intel SGX	5．発行年 2021年
3．雑誌名 The 15th International Conference on Provable and Practical Security (ProvSec 2021)	6．最初と最後の頁 215-240
掲載論文のDOI（デジタルオブジェクト識別子） 10.1007/978-3-030-90402-9_12	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 －

1．著者名 Tatsuya SUZUKI, Keita EMURA, Toshihiro OHIGASHI, Kazumasa OMOTE	4．巻 無
2．論文標題 Verifiable Functional Encryption using Intel SGX	5．発行年 2020年
3．雑誌名 Cryptology ePrint Archive: Report 2020/1221	6．最初と最後の頁 1-25
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 無
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 －

〔学会発表〕　計4件（うち招待講演　1件／うち国際学会　0件）

1．発表者名 西平侑磨，鈴木達也，石橋拓哉，柿崎淑朗，大東俊博，渡邉英伸，近堂徹，相原玲二
2．発表標題 Intel SGXを用いたモジューラ型エッジコンピューティング基盤の提案および評価
3．学会等名 電子情報通信学会技術研究報告，情報通信システムセキュリティ(ICSS)研究会
4．発表年 2023年

１．発表者名
佐藤友哉，大東俊博，近堂　徹，渡邉英伸，稲村勝樹

２．発表標題
Pub/Subプラットフォームにおける委託型秘密分散法の実装

３．学会等名
情報処理学会第84回全国大会

４．発表年
2022年

１．発表者名
西平侑磨，鈴木達也，渡邉英伸，大東俊博

２．発表標題
Intel SGXを用いたアルゴリズム変換型プロキシ再暗号化システムの実装・評価

３．学会等名
2022年暗号と情報セキュリティシンポジウム（SCIS2022）

４．発表年
2022年

１．発表者名
渡邉英伸，近堂徹，大東俊博

２．発表標題
データ特性に応じて組み換え可能なモジュール型エッジコンピューティング基盤の開発と高機能暗号化技術の応用

３．学会等名
電子情報通信学会コミュニケーションクオリティ研究会（招待講演）

４．発表年
2020年

〔図書〕　計0件

〔産業財産権〕

〔その他〕

―

６．研究組織

	氏名 （ローマ字氏名） （研究者番号）	所属研究機関・部局・職 （機関番号）	備考
研究分担者	大東　俊博 (Ohigashi Toshihiro) (80508127)	東海大学・情報通信学部・教授 (32644)	

７．科研費を使用して開催した国際研究集会

〔国際研究集会〕　計0件

８．本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関