資料１

2023年度 第１回 暗号技術検討会

令 和 ６ 年 ３ 月 ２ ６ 日
１０：００～
オ ン ラ イ ン 開 催

議事次第

１．開会

２．議事
(1) 2023年度暗号技術評価委員会 活動報告について【報告】
(2) 軽量暗号に関する外部評価報告書（案）及びCRYPTREC暗号技術ガイドライン（軽量暗号）（案）について【承認】
(3) 2023年度暗号技術活用委員会 活動報告について【報告】
(4) TLS暗号設定ガイドライン（案）について【承認】
(5) Triple DES等の取り扱いに係る暗号技術活用委員会からの意見について【報告】
(6) CRYPTREC暗号リストの更新について【承認】
(7) 電子署名法特定認証業務の暗号基準の改正スケジュールについて【報告】
(8) 2024年度暗号技術評価委員会活動計画（案）について【承認】
(9) 2024年度暗号技術活用委員会活動計画（案）について【承認】
(10) 暗号技術検討会 2023年度 報告書（案）について【承認】
(11) その他

３．閉会

配付資料一覧

資料１	議事次第・配付資料一覧
資料２	暗号技術検討会 開催要綱（構成員・オブザーバ名簿）
資料３－１	2023年度 暗号技術評価委員会 活動報告
資料３－２	監視状況報告
資料３－３	2023年度暗号技術調査ワーキンググループ（耐量子計算機暗号）活動報告
資料３－４	軽量暗号に関する技術動向調査 （別紙１）2023年度外部評価報告書（ASCON実装性能評価） （別紙２）2023年度外部評価報告書（ASCON標準化動向）
資料３－５	CRYPTREC暗号技術ガイドライン（軽量暗号）の更新について （別紙）CRYPTREC暗号技術ガイドライン（軽量暗号）（案）
資料４－１	2023年度 暗号技術活用委員会 活動報告
資料４－２	TLS暗号設定ガイドライン（案）の概要
資料４－３	TLS暗号設定ガイドライン（案）
資料４－４	Triple DES等の取り扱いに係る暗号技術活用委員会からの意見
資料５－１	CRYPTREC暗号リストの更新について
資料５－２	CRYPTREC暗号リスト（案）
資料６－１	電子署名法特定認証業務の暗号基準の改正スケジュールについて

資料６－２　　特定認証業務の基準の改正スケジュールについて（案）
資料７　　2024年度暗号技術評価委員会活動計画（案）
資料８　　2024年度暗号技術活用委員会活動計画（案）
資料９　　暗号技術検討会 2023年度 報告書（案）

以上

資料２

「暗号技術検討会」開催要綱

１　名　称

本検討会は「暗号技術検討会」（以下「検討会」という。）と称する。

２　開催の趣旨・目的

検討会は、デジタル庁統括官、総務省サイバーセキュリティ統括官及び経済産業省商務情報政策局長が有識者の参集を求め、暗号技術の普及による情報セキュリティ対策の推進を図る観点から、専門家による意見等を聴取することにより、デジタル庁、総務省及び経済産業省における施策の検討に資することを目的として開催する。

３　検討事項

（１）CRYPTREC暗号リスト掲載暗号技術の監視

（２）CRYPTREC暗号リスト掲載暗号技術の安全性及び信頼性確保のための調査・検討

（３）CRYPTREC暗号リストの改定に関する調査・検討

（４）CRYPTREC暗号リスト掲載暗号技術の普及促進及び暗号技術の利用促進・産業化に向けた取組の検討

（５）その他、システム全体のセキュリティ確保のために必要となる活動の検討等、暗号技術の評価及び利用に関すること

４　構成等

（１）検討会の構成は、別紙１のとおりとする。

（２）検討会には、座長１名を置く。

（３）座長は、構成員の互選により定める。

（４）座長は、検討会構成員の中から顧問及び座長代理を指名できる。

（５）構成員の任期は委嘱時に定めるものとし、再任を妨げないものとする。

５　運営

（１）座長は、検討会の議事を掌握する。

（２）座長が、緊急の理由によりやむを得ず不在となった場合、座長代理が座長に代わり議事を掌握する。

（３）関係する政府機関等で、座長が特に認めたものについては、オブザーバとして検討会に出席することができる。

（４）座長が必要と認めるときは、暗号技術の提案者、関連する利害関係者その他の参考人から意見を聴取することができる。

（５）座長は、検討会が調査する事項について特に専門的な調査を行う必要があると認めるときは、委員会等を置くことができる。
（６）座長は、必要があると認めるときは電子メールによる審議を行うことができる。なお、この審議を行った場合は、次の検討会において当該審議の結果を報告するものとする。
（７）その他検討会の運営に関し必要な事項は、座長が定めるところによる。

６　スケジュール

検討会は、年度内に１回以上開催する。

７　開催方法

検討会は、集合開催を原則とするが、必要に応じ、その一部又は全部をオンラインにより開催することができることとする。

８　議事・資料等の取扱い

別紙２のとおりとする。

９　庶　務

検討会の庶務は、デジタル庁デジタル社会共通機能グループ、総務省サイバーセキュリティ統括官室及び経済産業省商務情報政策局サイバーセキュリティ課において処理する。

（令和４年３月３０日　最終改訂）

別紙１

暗号技術検討会　構成員・オブザーバ名簿

2024.3.26現在

構成員

阿部　正幸	日本電信電話株式会社　社会情報研究所　上席特別研究員
石井　義則	一般社団法人情報通信ネットワーク産業協会　常務理事
上原哲太郎	立命館大学　情報理工学部　教授
太田　和夫	国立大学法人電気通信大学　名誉教授
高木　　剛	国立大学法人東京大学大学院　情報理工学系研究科　教授
田村　裕子	日本銀行　金融研究所　企画役
近澤　　武	三菱電機株式会社　情報技術総合研究所　開発戦略部　担当部長
手塚　　悟	慶應義塾大学　環境情報学部　教授
本間　尚文	国立大学法人東北大学　電気通信研究所　教授
松井　　充	三菱電機株式会社　開発本部　役員技監
松浦　幹太	国立大学法人東京大学 生産技術研究所　教授
松本　　勉	国立大学法人横浜国立大学大学院　環境情報研究院　教授
松本　　泰	日本ネットワークセキュリティ協会　フェロー
向山　友也	一般社団法人テレコムサービス協会　技術・サービス委員会　副委員長
吉田　博隆	国立研究開発法人産業技術総合研究所 サイバーフィジカルセキュリティ研究センター セキュリティ保証スキーム研究チーム 研究チーム長
渡邊　　創	国立研究開発法人産業技術総合研究所 サイバーフィジカルセキュリティ研究センター 副研究センター長

（五十音順、敬称略）

オブザーバ

内閣官房内閣サイバーセキュリティセンター　内閣参事官（政府機関総合対策担当）
個人情報保護委員会事務局　参事官
警察庁　長官官房　技術企画課 情報セキュリティ対策室長
総務省　自治行政局　住民制度課長
総務省　自治行政局　住民制度課　マイナンバー制度支援室長
法務省　民事局　商事課長
外務省　大臣官房　情報通信課長
財務省　大臣官房　文書課　業務企画室長
文部科学省　大臣官房　政策課　サイバーセキュリティ・情報化推進室長
厚生労働省　大臣官房参事官（サイバーセキュリティ・情報システム管理担当）
経済産業省　産業技術環境局　国際電気標準課長
防衛省　整備計画局　サイバー整備課　AI・サイバーセキュリティ政策調整官
国立研究開発法人情報通信研究機構　執行役/サイバーセキュリティ研究所長
国立研究開発法人産業技術総合研究所　サイバーフィジカルセキュリティ研究センター　首席研究員
独立行政法人情報処理推進機構　セキュリティセンター長
一般財団法人日本情報経済社会推進協会　デジタルトラスト評価センター　副センター長
公益財団法人金融情報システムセンター　監査安全部長

別紙２

暗号技術検討会の公開について

１　会議の公開について

（１）民間企業の暗号技術（既製品を含む）の解読方法等について議論を行う可能性があり、当事者又は第三者の権利、利益や公共の利益を害するおそれがあるため、検討会は原則非公開とする。

（２）検討会の出席者は、検討会において知り得た情報で、当事者又は第三者の権利、利益や公共の利益を害するおそれがあるものについては、検討会の出席者及び座長が特に認めた者以外に漏えいしてはならないものとする。

２　検討会の資料の公開について

（１）検討会の資料については、原則公開とする。

（２）ただし、検討会の資料を公開することにより、当事者又は第三者の権利、利益や公共の利益を害するおそれがある場合は、検討会は資料の公開を延期又は非公開とすることができる。

（３）資料は、ホームページ（cryptrec.go.jp）への掲載その他の方法により公開するものとする。

３　議事概要の公開について

（１）議事概要については、原則公開とする。

（２）ただし、議事概要を公開することにより、当事者又は第三者の権利、利益や公共の利益を害するおそれがある場合は、議事概要の該当部分を削除した上で公開することができる。

（３）議事概要は、ホームページ（cryptrec.go.jp）への掲載その他の方法により公開するものとする。

資料３－１

2023 年度　暗号技術評価委員会活動報告

1.　活動目的

CRYPTREC 暗号リストに掲載されている暗号技術や電子政府システム等で利用される暗号技術の安全性維持及び信頼性確保のために安全性及び実装に係る監視及び評価を行う。

2.　活動概要

2023 年度活動計画に沿って以下の内容を実施した。

1)　暗号技術の安全性及び実装に係る監視及び評価

暗号技術の安全性に係る監視・評価 及び 実装に係る技術の監視・評価を実施した。

①　CRYPTREC暗号リストの監視

国際会議等で発表されるCRYPTREC暗号リストの安全性及び実装に係る技術（暗号モジュールに対する攻撃とその対策も含む）に関する監視を行い、会議やMLを通して報告した。

・　2023年度は、電子政府推奨暗号リストの安全性に懸念を持たせるような事態は生じていない。今年度実施の監視報告の詳細については、CRYPTREC Report 2023で報告する。

②　電子政府推奨暗号リストからの運用監視暗号リストへの降格、並びに、推奨候補暗号リスト及び運用監視暗号リストからの危殆化が進んだ暗号の削除に係る検討

CRYPTREC暗号リストの安全性に係る監視活動を継続的に行い、急速に危殆化が進んだ暗号技術やその予兆のある暗号技術の安全性について評価を行う。また、リストからの降格や削除、注釈の改訂が必要か検討を行う。

・　2023年度は、運用監視暗号リストに含まれているTripleDESに関し、NISTによりFIPS SP800-67 Revision2 が2023年12月31日に削除された。但し、TripleDESに急速に危殆化が進んだわけではない。

③　CRYPTREC注意喚起レポートの発行

CRYPTREC暗号リストの安全性及び実装に係る技術の監視活動を通じて、国際会議等で発表された攻撃等の概要や想定される影響範囲、対処方法について早期に公開することが望ましいと判断された場合、注意喚起レポートを発行する。

・　現段階では、注意喚起レポートの発行は行っていない。

④　推奨候補暗号リストへの新規暗号（事務局選出）の追加に係る検討

標準化動向に鑑み電子政府システム等での利用が見込まれると判断される暗号技術の追加を検討する。

・　追加が必要となる暗号技術は無かった。

⑤　新技術等に関する調査及び評価

将来的に有用になると考えられる技術やリストに関わる技術について、安全性・性能評価を行う。必要に応じて、暗号技術調査ワーキンググループによる調査・評価、または、外部評価による安全性・性能評価などを行う。

(ア)NISTのPQC標準化において第4ラウンドが進行中であることから、引き続き、暗号技術調査ワーキンググループ（耐量子計算機暗号）（PQC WG）を設置して、耐量子計算機暗号に関する最新動向を把握する。また、「素因数分解の困難性に関する計算量評価」、「楕円曲線上の離散対数計算の困難性に関する計算量評価」の予測図の更新についてもPQC WGで検討し、更新を行う。

・　調査報告書・ガイドラインの記載内容は、2024年9月30日までの情報を可能な限り調査して掲載する等、調査報告書、および、ガイドラインの執筆方針が決定された。

・　NISTの標準化動向を調査しつつ、その内容を調査報告書、ガイドラインに反映することが承認された。

・　「素因数分解の困難性に関する計算量評価」、「楕円曲線上の離散対数計算の困難性に関する計算量評価」の予測図の更新を行い、承認された。
2023年度については、大きな変動はなかった。

（イ）2021年度に承認された「軽量暗号ガイドライン更新方針」に従って、「CRYPTREC暗号技術ガイドライン(軽量暗号)」2023 年度版の案を完成させる。

・　NISTで標準方式として選定されたASCONについて、実装、および、標準化動向について外部評価を実施し、報告書としてまとめた（資料３－４－別紙1、資料３－４－別紙２参照）

・　「CRYPTREC暗号技術ガイドライン（軽量暗号）」2023年度版の案を完成させた。（資料３－５－別紙参照）

2)　暗号技術の安全な利用方法に関する調査（技術ガイドラインの整備、学術的な安全性の調査・公表等）

暗号技術を利用する際の技術面での注意点に関する調査、新技術の安全性・性能に関する調査・評価を行う。

・　2-(1)-⑤-(イ）との重複項目：NISTの軽量暗号の標準方式であるAsconに関する実装評価、標準化動向調査を実施した。

3. スケジュールおよび各委員会における活動

1) 第1回暗号技術評価委員会（2023年7月3日：オンライン）の活動内容

- 委員長の選出を行い、高木剛委員が委員長として選出された。
- PQC WGについて、引き続き設置されることが承認された。そして、PQC WGの國廣昇委員が主査として指名された。また、PQC WGの活動計画を審議し、2024年9月30日までの情報に基づき、調査報告書とガイドラインを更新することが承認された。
- 軽量暗号ガイドラインの更新について、2023年度末までに更新することが承認された。また、ガイドラインの作成にあたり外部評価を行うことが承認された。
- 監視状況に関する報告が行われた。現在のCRYPTREC暗号リストに掲載の技術には問題がないことが確認された。

2) 第2回暗号技術評価委員会（2024年2月27日：オンライン）の活動内容

- PQC WGの活動が報告され、承認された。
- 軽量暗号の技術動向調査に関する報告が行われ、できる限り最新の情報に更新した上で公開することが承認された。
- 軽量暗号ガイドラインについて、2016年版の情報が混在している部分を整理し、体裁を整えた上で公開することが承認された。ガイドラインのレビュー結果を公開するか否かについては、再度、事務局で検討した後に、評価委員会によるメール審議を行い決定することになった。
- 監視状況に関する報告が行われた。現在のCRYPTREC暗号リストに掲載の技術には問題がないことが確認された。
- CRYPTREC Report 2023の目次案が承認された。

4. 評価委員会の構成（敬称略）

委員長	高木　剛	（東京大学）
委員	青木　和麻呂	（文教大学）
委員	岩田　哲	（名古屋大学）
委員	上原　哲太郎	（立命館大学）
委員	大東　俊博	（東海大学）
委員	國廣　昇	（筑波大学）
委員	四方　順司	（横浜国立大学）
委員	手塚　悟	（慶応義塾大学）
委員	花岡　悟一郎	（産業技術総合研究所）
委員	藤崎　英一郎	（北陸先端科学技術大学院大学）
委員	本間　尚文	（東北大学）
委員	松本　勉	（横浜国立大学）
委員	松本　泰	（セコム株式会社／ 日本ネットワークセキュリティ協会）*
委員	山村　明弘	（秋田大学）

＊2024 年 1 月より所属変更

資料３－２

監視状況報告

● 監視状況報告１ - Triple DES に関する NIST の情報

米国 NIST は Triple DES（TDES、米国では TDEA(Triple DATA Encryption Algorithm)）を標準化していた、 FIPS SP800-67 Revision 2 を 2023 年 12 月 31 日に削除いたしました。
そして、“Triple DES を容認されたアルゴリズムではない。すでに Triple DES で暗号化されて残っている暗号文などの復号等では使用を許容する”と述べ、“SP800-67 Rev.2 は歴史的な目的のためには、オンラインで利用できる”としている。*1

原文
“The scheduled withdrawal of SP 800-67 Rev. 2 will signify that TDEA is no longer an approved block cipher. TDEA will continue to be allowed for the decryption, key unwrapping, and verification of MACs of already-protected data, and SP 800-67 Rev. 2 will remain available online for historical purposes.”

*1 https://csrc.nist.gov/news/2023/nist-to-withdraw-sp-800-67-rev-2

● 監視状況報告２ － 学会関連の状況報告

1. 監視活動

2023 年 4 月より、2024 年 2 月までに、表 1 に示す国際会議に参加するとともに各種調査を行い、暗号解読技術等に関する研究動向を収集した。

表 1 国際会議への参加状況

学会名・会議名		開催国・都市	期間
FSE 2023	The 29th annual Fast Software Encryption Conference	(Beijing, China)	2023 年 3 月 20 日～3 月 24 日
Eurocrypt 2023	The 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques	Lyon, France (Hybrid Conference)	2023 年 4 月 23 日～4 月 27 日
PKC 2023	The 26th International Conference on Practice and Theory in Public Key Cryptography	Atlanta, USA (Hybrid Conference)	2023 年 5 月 7 日～5 月 10 日
PQCrypto 2023	The 14th International Conference on Post-Quantum Cryptography	(College Park, MD, USA)	2023 年 8 月 16 日～8 月 18 日
Crypto 2023	The 43rd Annual International Cryptology Conference	(Santa Barbara, USA)	2023 年 8 月 19 日～8 月 24 日
TCC 2023	Theory of Cryptography 20th International Conference	(Taipei, Taiwan)	2023 年 11 月 29 日～12 月 2 日
Asiacrypt 2023	The 29th Annual International Conference on the Theory and Application of Cryptology and Information Security	(Guangzhou, China)	2023 年 12 月 5 日～12 月 9 日

2. 解読技術等の動向

各国際会議より、具体的な暗号の攻撃に関する発表を抽出し、CRYPTREC 暗号リスト記載の暗号の安全性に直接関わる技術動向（2.1）およびその他の注視すべき技術動向（2.2）について分析した。

2.1. CRYPTREC 暗号リスト記載の暗号に直接関わる解読技術動向

CRYPTREC 暗号リスト（電子政府推奨暗号リスト）掲載の暗号に関して報告する。

フルラウンド 14 ラウンドの共通鍵暗号 AES については、Eurocrypt 2023 にて新たなブ

ーメラン攻撃が報告された。特筆すべき点として、6 ラウンドの AES における計算量 2^{61} の鍵回復攻撃が得られている。さらに、FSE 2023 にて、7 ラウンドの AES に対する新たな鍵回復攻撃が提案された。また CRYPTO 2023 にて、新しい関連鍵攻撃が報告された。中間一致攻撃と、差分攻撃を組み合わせた、差分中間一致攻撃という解読手法により、従来の関連鍵攻撃が 2 ラウンド更新されている。

また、フルラウンドが 20 ラウンドのストリーム暗号 ChaCha に対する新たな解読手法が FSE2023 や CRYPTO2023 にて報告された。FSE2023 では、6 ラウンド ChaCha20 に対する解読について、現在知られている時間計算量を 2^{40} 倍改善した手法が提案された。CRYPTO 2023 では、syncopation という新しい手法を導入することで、良い PNB (probabilistic neutrality bit, PNB) の集合をみつける効率的な手法が提案され、最後の XOR とローテーションを行わない 7.5 ラウンドの ChaCha 20 に対する鍵回復攻撃と、既存の 6 ラウンド及び 7 ラウンドに対する攻撃の高速化が実現された。

またハッシュ関数 RIPEMD-160 については、時間計算量 $2^{64.5}$ で 36 ラウンドまで到達できる、新しい衝突攻撃が、Eurocrypt 2023 にて提案された。

ハッシュ関数 SHA3 については、2 件の攻撃論文が Eurocrypt 2023 にて報告されている。特に、4 ラウンドの SHA3-512 に対して、2^{237} の時間計算量をもつ衝突攻撃と、$2^{504.58}$ の時間計算量を持つ原像攻撃が提案された。さらに、FSE 2023 にて新規の解析手法が提案され、$2^{59.64}$ の時間計算量および $2^{45.94}$ のメモリ計算量で実行される 4 ラウンド SHA-3-384 に対する衝突攻撃が報告された。SHA-3 のフルラウンドは 24 ラウンドである。

ハッシュ関数 SHAKE256 に対しても、Eurocrypt 2023 にて新たな攻撃が報告された。特に、5 ラウンドの SHAKE256 に対する衝突攻撃が、2^{185} の時間計算量で達成された。

いずれも現実的な脅威となるにはまだ十分なマージンがあるが、今後も動向を注視すべきである。

2.1.1. 共通鍵暗号に関する解読技術

- **Truncated Boomerang Attacks and Application to AES-based Ciphers [Eurocrypt 2023]**

  Augustin Bariant, Gaëtan Leurent

ブーメラン攻撃は、1 つの長い差分を使う代わりに、2 つの短い差分を組み合わせる暗号解読技術である。多くのプリミティブに適用されており、いくつかの AES ベースの暗号（Kiasu-BC、Deoxys-BC）に対する最も有名な攻撃である。本論文は、truncated differential を用いたブーメラン攻撃に関する一般的な枠組みを紹介し、文献にある最高のブーメラン攻撃よりも大幅に改善を得ている。特に、平文側と暗号文側の構造を考慮し、鍵回収のステップの分析を行っている。6 ラウンドの AES において、計算量 2^{87} の構造識別器と計算量 2^{61} の鍵回収攻撃を得ることができた。この truncated ブーメラン攻撃は、tweakable な AES の亜種に対して特に効果的である。とくに 8 ラウンドの Kiasu-BC に適

用すると、複雑さ 2^{83} の最もよく知られた攻撃が再現できる。また、6 ラウンドの AES を構成要素として使用した、tweakable ブロック暗号である TNT-AES の 6 ラウンド識別器への応用例も紹介されている。さらにこのフレームワークは Deoxys-BC に適用され、MILP モデルを使った最適 trail の自動的発見に応用されている。これにより著者らは、Deoxys-BC のすべての亜種のラウンド削減バージョンに対する最良の攻撃を得た。

- **Cryptanalysis of Reduced Round ChaCha -- New Attack & Deeper Analysis [FSE 2023]**

  Sabyasachi Dey, Hirendra kumar Garai, Subhamoy Maitra

  ストリーム暗号 ChaCha に対する攻撃論文である。本論文では、秘密鍵ビットに対する分割統治法によるアプローチが提案されている。この分割方法は複数の入出力差分に基づいていて、結果として、計算量 $2^{99.48}$ で 6 ラウンド ChaCha に対する攻撃が実現されている。これは現在知られている攻撃よりも 2^{40} 倍高速である。また、本攻撃のような PNB ベースの差分攻撃の成功確率の評価も行われている。

- **New Key Recovery Attack on Reduced-Round AES [FSE 2023]**

  Navid Ghaedi Bardeh, Vincent Rijmen

  AES の基本的な 4 ラウンドに対する、ゼロ差分性（zero-difference property）と呼ばれる性質が、Asiacrypt 2017 で Rønjom、Bardeh、Helleseth によって報告された。

  本論文はこの性質に対して、AES 設計者によって導入された関連する差分の概念を利用して分析し、簡単な特徴付けを与えている。さらに AES 線形層上の関連する差分の更なる性質を考慮して、このゼロ差分性を一般化することにより、4 ラウンドという性質を拡張する方法も示している。結果として、ゼロ差分性を利用した 7 ラウンド AES に対する新たな鍵回復攻撃を提案している。

- **Differential Meet-In-The-Middle Cryptanalysis [CRYPTO 2023]**

  Christina Boura, Nicolas David, Patrick Derbez, Gregor Leander, María Naya-Plasencia

  ブロック暗号 SKINNY および AES に対する攻撃論文である。本論文では、中間一致攻撃と、差分攻撃を組み合わせた、差分中間一致攻撃という解読手法を提案している。単一鍵モデルでは SKINNY-128-384 に、関連鍵モデルでは AES-256 に、提案された解読手法を適用することで、それぞれの解読が 2 ラウンド更新された。

- **Moving a Step of ChaCha in Syncopated Rhythm [CRYPTO 2023]**

  Shichang Wang, Meicheng Liu, Shiqi Hou, Dongdai Lin

  ストリーム暗号 ChaCha に対する解読論文である。本論文は、ChaCha を解析するため

に、確率的中立ビット（probabilistic neutrality bit, PNB）の差分解析を行っている。著者らは、syncopation という新しい手法を導入することで、良い PNB の集合をみつける効率的な手法を提案した。特に、最後の XOR とローテーションを行わない 7.5 ラウンドの ChaCha に対する鍵回復攻撃と、既存の 6 ラウンド及び 7 ラウンドに対する攻撃の高速化を実現している。

2.1.2. ハッシュ関数に関する解読技術

- **Analysis of RIPEMD-160: New Collision Attacks and Finding Characteristics with MILP [Eurocrypt 2023]**

  Fukang Liu, Gaoli Wang, Santanu Sarkar, Ravi Anand, Willi Meier, Yingxin Li, Takanori Isobe

ハッシュ関数 RIPEMD-160 は ISO/IEC 標準であり、SHA-256 とともにビットコインアドレスの生成に使用されている。MD-SHA ハッシュファミリーの多くのハッシュ関数が破られたにもかかわらず、RIPEMD-160 は安全性を保ち、CRYPTO 2019 で与えられた最高の衝突攻撃は 80 ラウンドのうち 34 ラウンドまでしか到達していない。

本論文では、時間計算量 $2^{64.5}$ で 36 ラウンドまで到達できる、RIPEMD-160 に対する新しい衝突攻撃が提案されている。この新しい攻撃は、メッセージの差分を選択する新しい戦略と、両方のブランチの差分条件を同時に処理する新しい技術による。RIPEMD-160 に関するすべての先行研究と異なる点として、差分特性を探索するために MILP ベースの方法を利用し、そのラウンド関数を通じて符号付き差分遷移を正確に記述するモデルが構築されている。これは、著者らの知る限り、MD-SHA ハッシュファミリーの符号差分遷移を対象とした最初のモデルだと報告されている。このモデルを設計する動機として、このような差分特性を検索するための多くの自動ツールが公開されておらず、それらをゼロから実装するのはあまりにも時間がかかり困難であるという課題があった。著者らは、このモデルが、いくつかの簡単な線形不等式を書き出すだけで済む、将来の研究のための代替的な簡単なツールになることを期待している。

- **Collision Attacks on Round-Reduced SHA-3 Using Conditional Internal Differentials [Eurocrypt 2023]**

  Zhongyi Zhang, Chengan Hou, Meicheng Liu

KECCAK ハッシュ関数は、2012 年に NIST によって SHA-3 コンペティションの勝者に選ばれ、2015 年に NIST の SHA-3 ハッシュ標準となった。SHA-3 ファミリーの中で、SHA3-512 は衝突攻撃に対して最も強い耐性を示している。SHA3-512 の理論的な攻撃は、誕生日攻撃の 64 倍の速さで多項式を解くことによっても、4 ラウンドまでしか伸びない。しかし、SHA-3 のインスタンスである SHAKE256 については、著者らが知る限り、衝突攻撃に関する結果は文献に存在していなかった。

本論文では、ラウンド削減された SHA-3 に対する衝突攻撃について研究される。2013 年の Dinur、Dunkelman、Shamir の研究に触発され、著者らは誕生日攻撃の変種を提案し、差分遷移条件や差分条件表などの新しい概念を抽象化して、内部差分暗号を改良する。これらの技術の助けを借りて、条件付き内部差分を用いたラウンド削減型 SHA-3 に対する新しい衝突攻撃を開発する。より正確には、線形条件で制約された初期メッセージは内部差分の最初の 2 ラウンドを通過し、最後の 2 ラウンドに入る対応する入力は線形条件の値に従って衝突検索のために異なる部分集合に分割される。改良されたターゲット内部差分アルゴリズム(TIDA)と共に、6 つの SHA-3 関数の最大 5 ラウンドに対する衝突攻撃を獲得した。特に、4 ラウンドの SHA3-512 と 5 ラウンドの SHAKE256 に対する衝突攻撃が、それぞれ 2^{237} と 2^{185} の計算量で達成された。著者らによれば、これはラウンド削減型 SHA3-512 に対する最良の衝突攻撃であり、ラウンド削減型 SHAKE256 に対する最初の衝突攻撃である。

- **Meet-in-the-Middle Preimage Attacks on Sponge-based Hashing [Eurocrypt 2023]**

  Lingyue Qin, Jialiang Hua, Xiaoyang Dong, Hailun Yan, Xiaoyun Wang

中間一致（MITM）攻撃は、Merkle-Damgård (MD) ハッシングに対する原像攻撃に広く適用されている。

本論文では、スポンジベースのハッシングに対する MITM 攻撃の一般的なフレームワークが紹介されている。結果として、Keccak-512/SHA3 に対する改良型 4 ラウンド原像攻撃が行われ、10 年近い暗号解読の記録を更新された。また、3-4 ラウンドの Ascon-XOF と 3 ラウンドの Xoodyak-XOF に対する最初の原像攻撃も与えられている。

- **Finding Collisions against 4-round SHA-3-384 in Practical Time [FSE 2023]**

  Senyang Huang, Orna Agmon Ben-Yehuda, Orr Dunkelman, Alexander Maximov

ハッシュ関数 SHA-3 に対する解析論文である。現在 SHA-3 に対する最良の衝突攻撃は Jian Guo の線形化手法であるが、これは SHA-3-384 のような亜種については実行できなかった。

本論文は、1 ブロックメッセージの代わりに 2 ブロックメッセージを使用した解の柔軟性の向上、線形化手法の代わりに接続性問題を充足可能性問題に還元する手法、そして非線形層の新しい 2 つの非ランダム性に基づいた効率的な deduce-and-sieve アルゴリズムの提案により、この課題を克服した。結果として、4 ラウンド SHA-3-384 上の衝突攻撃を、$2^{59.64}$ の時間計算量および $2^{45.94}$ のメモリ計算量で実現した。

2.2. その他の注視すべき技術動向

共通鍵暗号については、前述したAESに対する攻撃の他に、様々な軽量暗号およびブロック暗号などに対して、差分攻撃などを中心とした解読技術に関する研究が報告されている。

公開鍵暗号・署名においては、NIST PQC Standardizationと関わりのある耐量子計算機暗号への解析に、非常に大きな進展が見られた。特に、2022年8月にプレプリントとして公表された、第4ラウンドの候補であったSIKEを破った攻撃論文が、Eurocrypt 2023において報告された。また、この攻撃を一般化・改良した研究成果も複数発表された。また、小さなパラメータに対するFalconへの解析や、BIKEに対する鍵回復の検討などが進んでいる。他にも、実社会へのインパクトある話題として、昨年チューリッヒ大学が指摘した、クラウドストレージサービスMEGAにおけるRSA秘密鍵が漏洩する脆弱性に関連して、PKC 2023とEurocrypt 2023でそれぞれ1件ずつ、MEGAへのさらなる攻撃可能性に対する研究結果が発表されている。

第2.1節で報告したCRYPTREC暗号リストに関連する報告以外に、引用される機会が多かった、もしくは今後多くなると予想される解析報告を以下に列挙する。

2.2.1. 共通鍵暗号に関する技術動向

- **Better Steady than Speedy: Full Break of SPEEDY-7-192 [Eurocrypt 2023]**

  Christina Boura, Nicolas David, Rachelle Heim Boissier, Maria Naya-Plasencia

差分攻撃は、対称鍵暗号解読の最も重要な手法の一つであり、1990年の登場以来、基本的な手法に改良が加えられ、いくつかの専用攻撃も提案されてきた。提案されてきた改良のほとんどは、鍵の回復部分に関するものである。しかし、新しいプリミティブを設計する場合、差分攻撃に関する安全性解析は、多分岐と境界の技術を使って限られたラウンド数で最良の痕跡を見つけることに限られていて、そこからのヒューリスティックにより、差分攻撃が到達できるラウンド数を推測するものが多い。

本論文では、差分暗号に対するSPEEDYファミリーのブロック暗号の安全性を分析し、この種の攻撃に対する鍵回復手順の多くのステップを最適化する方法を示す。このために、この暗号の最適なtrailとそれに関連する多重確率をいくつかの制約の下で見つけるための探索を実装し、最適なデータと鍵ふるい（key-sieving）を得るための非自明な技法を適用した。これにより、192ビットのセキュリティを提供するとされるSPEEDYの7ラウンド変種であるSPEEDY-7-192の解読に成功している。

- **Exploiting Non-Full Key Additions: Full-Fledged Automatic Demirci-Selçuk Meet-in-the-Middle Cryptanalysis of SKINNY [Eurocrypt 2023]**

  Danping Shi, Siwei Sun, Ling Song, Lei Hu, Qianqian Yang

Demirci-Selçuk中間一致（DS-MITM）攻撃は、差分攻撃の高度な亜種である。その高度

さゆえに、AES を除くほとんどの暗号に対する最適な DS-MITM 攻撃を効率的に見つけることは困難である。さらに、現在の自動化ツールは DS-MITM 攻撃の最も基本的なバージョンしか捉えておらず、攻撃を強化するために開発された重要な技術（例えば、差分列挙や鍵依存ふるい（key-dependent-sieve））は、依然として手作業に頼っている。本論文では、DS-MITM 攻撃のための既知の技術（差分列挙、鍵依存ふるい、鍵ブリッジなど）を統合した本格的な自動フレームワークを開発し、識別器の探索だけでなく、鍵回復攻撃を直接生成できるようにする。さらに、部分的な鍵の追加を利用して、攻撃に有利な線形関係をより多く生成することができる新しい技術を開発する。このフレームワークをブロック暗号の SKINNY ファミリーに適用したところ、大幅に改善された結果が得られた。特に、SKINNY の各バージョンに対する既知の DS-MITM 攻撃は少なくとも 2 ラウンド改善されていて、また他の既存攻撃のデータ、メモリ、または時間計算量の削減にも成功している。

- **Efficient Detection of High Probability Statistical Properties of Cryptosystems via Surrogate Differentiation [Eurocrypt 2023]**

  Itai Dinur, Orr Dunkelman, Nathan Keller, Eyal Ronen, Adi Shamir

暗号解読の中心的な問題は、与えられたnビットの暗号プリミティブにおいて、ランダム性からの有意な逸脱をすべて見つけ出すことである。nが小さい場合（例えば 8 ビットの S-box）これは簡単だが、nが大きい場合、このような統計的性質を見つける唯一の実用的な方法は、プリミティブの内部構造を利用し、様々なヒューリスティックで検索を高速化することだった。しかし、このようなボトムアップ的な手法では、特にトラップドアを持っているような暗号システムにおいては、多くの特性を見逃す可能性がある。

本論文では、暗号プリミティブが構造のないブラックボックスとして与えられる問題のトップダウン版を考え、その重要な微分・線形特性をすべて見つけるための最もよく知られた技術の計算量を、$2^{n/2}$という大きな係数で低減させる方法を提案している。主な新しいツールは、surrogate differentiation を使うというアイデアである。差分特性を求める文脈では、ブラックボックス関数fと全ての可能な方向αにおいて、αと独立に任意に選ばれる方向γで差分をとり、$f(x) \oplus f(x \oplus \alpha)$の形の差分についての情報を同時に探索することが可能となる。線形特性を求める文脈では、surrogate differentiation は非常に効率的な方法で高速フーリエ変換と組み合わせることができる。この技術によって、64 ビット暗号プリミティブについて、2^{64} の時間計算量において、$p \geq 2^{-32}$の確率での全ての差分を自動探索し、バイアス$|p| \geq 2^{-16}$で全ての線形近似を自動探索することができる。以前の研究成果と比較すると、これらは2^{96}の時間計算量を要していた。同様の手法で、関連する鍵差分、2 階差分、ブーメラン攻撃を求める最もよく知られた時間計算量を大幅に改善することができる。さらに、メモリを必要としないこれらのアルゴリズムの変種を実行する方法を示し、トラップドア暗号システムにおいても、このような統計的特性を検出する方法を示すことができる。

- **Finding the Impossible: Automated Search for Full Impossible-Differential, Zero-Correlation, and Integral Attacks [Eurocrypt 2023]**

  Hosein Hadipour, Sadegh Sadeghi, Maria Eichlseder

不可能差分攻撃（Impossible-Differential: ID）、ゼロ相関攻撃（Zero-Correlation: ZC）、積分攻撃（Integral）は、ブロック暗号に対する重要な攻撃の一群である。例えば、不可能差分攻撃は、AES の 7 ラウンドに対する最初の暗号解読攻撃である。これらの攻撃に対するブロック暗号の安全性を評価することは、非常に重要であるが、同時に困難でもある：通常、これらの攻撃を見出すことは、手作業で解くことが困難な、多くのパラメータと制約を含む組み合わせ論的最適化問題を意味するためである。制約プログラミング（Constraint Programming: CP）ソルバーなどの自動化されたソルバーは、適切な攻撃を見出すのに有用である。しかし、これまでの CP ベースの手法は、ID、ZC、積分攻撃のみに焦点を当てているため、その探索空間は限られており、特に、効率的な鍵回収ステップを含む完全な攻撃を見つけるための統一的な最適化問題へと拡張できていない。

本論文では、ID、ZC、積分識別器を探索する新しい CP ベースの方法を提示し、ID、ZC、積分攻撃を見つけるための統一的な制約最適化問題へと拡張することが提案されている。その有効性と有用性を示すために、SKINNY、CRAFT、SKINNYe-v2、SKINNYee を含むいくつかのブロック暗号への応用結果が記載されている。SKINNY では、既存の ID 攻撃、ZC 攻撃、積分攻撃が大幅に改善されている。特に、SKINNY-n-3n と SKINNY-n-2n に対する積分攻撃をそれぞれ 3 ラウンドと 2 ラウンド改善し、単一鍵設定におけるこれらの変種に対する最高の暗号解析結果を得ている。また、SKINNY-n-n（SKINNY-n-2n）に対する ZC 攻撃を 2 ラウンド（1 ラウンド）改善されている。また、SKINNY のすべての変種に対する ID 攻撃も改善されている。特に、SKINNY-128-256（resp.SKINNY-128-384）に対する従来の最良の単一 tweakey（関連 tweakey）ID 攻撃の時間計算量を $2^{22.57}$ (resp.$2^{15.39}$) に向上させた。CRAFT では、21 ラウンド（20 ラウンド）の ID（resp.ZC）攻撃が提案されており、従来の単一 tweakey 攻撃を 2 ラウンド（resp.1 ラウンド）向上させている。他にも、SKINNY、CRAFT、Deoxys-BC のラウンド数を減らした場合の実用的な積分識別器も提供されている。

- **Vectorial Decoding Algorithm for Fast Correlation Attack and Its Applications to Stream Cipher Grain-128a [FSE 2023]**

  Zhaocun Zhou, Dengguo Feng, Bin Zhang

LFSR ベースのストリーム暗号に対する暗号解析手法の提案である。Meier と Staffelbach によって提案された高速相関攻撃は、LFSR ベースのストリーム暗号に対する重要な解析ツールである。この高速相関攻撃は、LFSR の状態と鍵ストリームの相関を利用して、復号アルゴリズムを通じて、LFSR の初期状態を回復する。

本論文では従来のバイナリアプローチの自然な一般化である、ベクトルに対する復号アルゴリズムを提案している。著者らはこの手法を良く解析されている Grain-128a に適用した。結果として、反復的な復号化の観点から安全性評価について新しい知見を与え、行列環上の LFSR 及びバイアスのある多次元の線形近似をもつ非線形関数に対する潜在的脆弱性を明らかにしている。

- **New Cryptanalysis of ZUC-256 Initialization Using Modular Differences [FSE 2023]**
  Fukang Liu, Willi Meier, Santanu Sarkar, Gaoli Wang, Ryoma Ito, Takanori Isobe

ストリーム暗号 ZUC-256 に対する暗号解析論文である。ZUC-256 は 5G アプリケーション用に設計されたストリーム暗号であり、AES-256、SNOW-V とともに、Security Algorithms Group of Experts (SAGE) による 5G 移動通信における標準化アルゴリズムの評価対象となっている。

本論文では、モジュラ差分、符号差分、XOR 差分のツールを用いることで、これらの演算間の相互作用の制御技術が提案されている。この技術は、従来 MD-SHA ハッシュ関数に対して Wang らが開発してものと似ているが、ZUC-256 はラウンド関数の複雑さなどが全く異なるため、新しい工夫が盛り込まれている。結果として、複雑な入力差を使う識別攻撃が、33 ラウンド中の 31 ラウンドの ZUC-256、33 ラウンド中の 30 ラウンドの ZUC-256-v2 に対して示された。また、関連鍵設定において、15 ラウンド ZUC-256、14 ラウンド ZUC-256-v2 に対して、16 ビットの鍵を効率的に回復する方法も示している。

- **Cryptanalysis of Draco [FSE 2023]**
  Subhadeep Banik

ストリーム暗号 Draco に関する解析論文である。Draco は Hamann らが IACR ToSC 2022 で報告した軽量ストリーム暗号であり、95 ビットと 33 ビットの二つの状態レジスタを持ち、Grain に似た構造を持つ。さらにこの暗号は、128 ビットの秘密鍵と 96 ビットの IV を使用しており、この秘密鍵と IV の最初の 32 ビットは、暗号が鍵ストリームビットを生成している間は変化しない不揮発性の内部状態を形成する。この Draco は、時間-メモリ-データ（TMD）トレードオフ攻撃に対して安全であると主張されている。

これに対し本論文は、Draco に対する 2 つの TMD トレードオフ攻撃を提案した。これらの攻撃は、慎重に選ばれた特定の IV について、状態更新関数が不揮発性内部状態のごく一部に依存するという事実を利用している。第一の攻撃は約 $2^{114.2}$ 回の Draco 処理が必要であり、攻撃者は 2^{32} 回選択された IV にアクセスできる必要がある。第二の攻撃は、攻撃者が攻撃パラメータを調整できる攻撃である。IV へのアクセスを 2^{20} 回に制限する場合、2^{126} 回の Draco 処理に比例した時間で実行できる。IV へのアクセス回数に制限がない場合は、IV へのアクセスを 2^{40} 回行うことで、2^{107} 回の Draco 処理に比例した時間で実行できる。

- **Attacks on the Firekite cipher [FSE 2023]**

  Vu Nguyen, Thomas Johansson, Willi Meier

ストリーム暗号 Firekite に対する暗号解析論文である。Firekite は擬似乱数生成器(PRNG) を使った同期ストリーム暗号で、その安全性は Learning Parity with Noise (LPN) 問題の困難性に依存するといわれている。これは数少ない LPN ベース対称暗号化方式の 1 つであり、ローエンド SoC FPGA 上で非常に効率的に実装できる。設計者である Bogos、Korolija、Locher、Vaudenay は、暗号的に強力なビットのソースを 1 つだけ必要とすること、小さな鍵サイズ、達成可能な高いスループット、選択された実際的なパラメータに依存するビットレベルのセキュリティの見積もりなど、Firekite の魅力的な特性を実証した。

本論文は、その PRNG の構造特性を利用することにより、 Firekite に対する識別および鍵回復攻撃を提案している。バースデイ攻撃のテクニックを用いることで、Firekite の出力の特定の和がランダムな場合よりも高い確率で低いハミング重みを持つことが示されている。結果として、80 ビット安全性及び 128 ビット安全性に相当するセキュリティパラメータを持つ Firekite に対して、それぞれ計算量 $2^{66.75}$ 及び $2^{106.75}$ を持つ識別攻撃が得られている。鍵回復攻撃については、80 ビット安全性に相当するセキュリティパラメータを持つ Firekite に対して、$2^{69.87}$ の計算量のものが提案されている。

- **SAT-aided Automatic Search of Boomerang Distinguishers for ARX Ciphers [FSE 2023]**

  Dachao Wang, Baocang Wang, Siwei Sun

軽量ブロック暗号 SPECK に対する解読論文である。ARX 暗号において、大きなドメインサイズはブーメラン接続表の作成を困難にし、接続表を利用した攻撃を困難にする運用を妨げることが知られている。

本論文は、モジュラ加算とブーメラン特性の自動探索についてのブーメラン接続表の計算問題を取り扱い、この表およびその亜種を効率的に計算するための動的計画法アルゴリズムを提案している。これは従来の計算手法と比較して 4 倍高速である。さらにこれらのアルゴリズムをブール表現で書き替えた後、対応する充足可能性問題モデルを構築し、それに対する 2 つの自動探索フレームワークも提案されている。結果として、SPECK32/64 に対する 10 ラウンドのブーメラン特性を確率 $2^{-29.15}$ で、SPECK48/72 に対する 12 ラウンドの特性を確率 $2^{-44.15}$ で発見した。

- **Automatic Search of Rectangle Attacks on Feistel Ciphers: Application to WARP [FSE 2023]**

  Virginie Lallemand, Marine Minier, Loïc Rouquette

軽量暗号 WARP に対する解析論文である。

本論文では、Delaune ら自動化ツールを Feistel 暗号の場合に適応させる方法が示されている。この手法によって、41 ラウンド中 23 ラウンドの WARP に対する確率 2^{-124} の識別器が得られている。これは今までの識別器で最良である。また鍵回復フェーズを追加して、時間計算量 $2^{115.9}$、データ計算量 $2^{120.6}$ の 26 ラウンド矩形攻撃も提案されている。

- **Throwing Boomerangs into Feistel Structures: Application to CLEFIA, WARP, LBlock, LBlock-s and TWINE [FSE 2023]**

  Hosein Hadipour, Marcel Nageler, Maria Eichlseder

Feistel 構造をもつ暗号に対するブーメラン識別器の解析論文である。

本論文では、Hadipour らにより最近提案されたブーメラン識別器を探索する方法を強化することにより、ブーメラン識別器を探索するための自動ツールを提供し、それを一般化 Feistel 構造に基づくブロック暗号に適用している。特に、WARP のブーメラン識別器を 2 ラウンド更新、CLEFIA に対するブーメラン識別器を 1 ラウンド更新した。この識別器を利用して、11 ラウンドの CLEFIA に対する鍵回復攻撃も提案されている。他にも、LBlock、Lblock-s、TWINE に対する最良のブーメラン識別器を得ている。

- **Integral Cryptanalysis of WARP based on Monomial Prediction [FSE 2023]**

  Hosein Hadipour, Maria Eichlseder

軽量暗号 WARP に対する解析論文である。WARP は AES に代わる軽量暗号として SAC 2020 で Banik らが提案した軽量 128 ビットブロック暗号であり、一般化 Feistel 構造に基づいている。今までに、41 ラウンド中の 21 ラウンドの WARP に対して積分鍵回復攻撃が得られている。

本論文では、積分識別器と鍵回復攻撃の双方を実質的に改善することで、32 ラウンドまでの積分鍵回復攻撃を得ている。積分識別器については、Hu らが ASIACRYPT 2020 で提案した単項式予測を SAT 問題にモデル化し、鍵スケジュールを考慮した WARP のビット指向モデルを作成した。これにより以前の識別器を古典的積分識別器として 2 ラウンド、一般化積分識別器として 4 ラウンド拡張している。

- **Practical Cube Attack against Nonce-Misused Ascon [FSE 2023]**

  Jules Baudrin, Anne Canteaut, Léo Perrin

軽量暗号 ASCON に対する解析論文である。ASCON は CAESAR コンペティションで採択され、また NIST で標準化された軽量暗号でもある。

本論文では、nonce-misuse 設定において、6 ラウンド ASCON に対する実用的な時間で実現できる cube 攻撃を提示している。著者らは、この攻撃の前提が nonce-misuse 設定であるため、ASCON 設計者により設定された安全性に違反するものではなく、鍵回復や鍵偽造攻撃が達成されているわけではないことも注釈している。

- **Truncated Differential Attacks on Contracting Feistel Ciphers [FSE 2023]**

  Tim Beyne, Yunwen Liu

Feistel 暗号に対する切り捨て（trucated）差分攻撃を改善する論文である。新しい切り捨て差分に基づき、$O(N^{t-1})$のデータ計算量および時間計算量で、t^2+t-2ラウンドに対する識別器を得ている。また、$\tilde{O}(N^{t-2})$のデータ計算量、$\tilde{O}(N^{t-1})$の時間計算量を用いて、t^2+1ラウンドの鍵回復攻撃を得ている。この一般的な攻撃は、多くのラウンドで、知られていた攻撃よりも低いデータ計算量を実現している。

さらにこの攻撃を、GMiMC-crf のいくつかのフルラウンドインスタンスや、中国のブロック暗号である SM4 の 17 ラウンドに適用している。

- **Breaking HALFLOOP-24 [FSE 2023]**

  Marcus Dansarie, Patrick Derbez, Gregor Leander, Lukas Stennes

Tweakable ブロック暗号 HALFLOOP-24 に対する解析論文である。HALFLOOP-24 は、長距離通信で用いられる高周波無線において、自動リンク確立(ALE)のためのメッセージを保護する tweakable ブロック暗号である。

本論文は HALFLOOP-24 に対する最初の公開暗号解析であり、暗号文単独攻撃、既知平文攻撃、選択平文攻撃、選択暗号文攻撃を行い、128 ビット安全性を破っている。
提案された攻撃の中で最も効率的なものはブーメラン鍵回復であり、2^10 回未満の暗号及び復号クエリにて最初のラウンド鍵を見つけることができる。結論として、HALFLOOP-24 を使用しないことを、著者らは強く推奨している。

- **Cryptanalysis of Rocca and Feasibility of Its Security Claim [FSE 2023]**

  Yosuke Todo, Akinori Hosoyamada, Akiko Inoue, Ryoma Ito, Tetsu Iwata, Kazuhiko Minematsu, Ferdinand Sibleyras

認証付き暗号 Rocca に対する解析論文である。Rocca は FSE 2022 及び ToSC 2021 で提案され、鍵回復および識別攻撃に対する 256 ビットの安全性と、偽造攻撃に対する 128 ビットの安全性が実現されていると主張されていた。これらの安全性主張において、鍵回復攻撃は、復号オラクルを通じた複数の偽造を取得することを攻撃者に許容している。

本論文はこの設定において、Rocca に対する完全鍵回復攻撃を提示している。本攻撃のデータ計算量は 2^{128}、時間計算量は 2^{128}、成功確率はほぼ 1 である。これは 256 ビット安全性主張を破るものである。

- **Practical Attacks on the Full-round FRIET [FSE 2023]**

  Senpeng Wang, Dengguo Feng, Bin Hu, Jie Guan, Tairong Shi

Duplex 構造に基づく軽量認証付き暗号 FRIET に対する解析論文である。EUROCRYPT

2020 で提案された FRIET は、故障攻撃に対する対策を内蔵していることを特徴とし、認証付き暗号 FRIET-AE と、それを構築するための置換 FRIET-PC が提案されている。

本論文は、FRIET-PC のラウンド関数による差分と線形マスクの対の伝搬を研究し、フルラウンド FRIET-PC に対して、確率が 1 である差分識別器と、相関の絶対値が 1 である線形識別器を構成している。この差分識別器を用いて、正当なユーザによって作成されていない有効なタグと暗号文から成る集合が構成することで、完全性と機密性に対する FRIET-AE の安全性主張を破っている。

- **Revisiting the Extension of Matsui's Algorithm 1 to Linear Hulls: Application to TinyJAMBU [FSE 2023]**

  Muzhou Li, Nicky Mouha, Ling Sun, Meiqin Wang

軽量暗号に対する解析論文である。EUROCRYPT 1993 で Matsui により導入された線形解読法において、アルゴリズム 1、アルゴリズム 2 は、特定の状態ビットを含む線形近似を使用する。アルゴリズム 2 は、追加の鍵ビットを推測した後、状態ビットを得るために部分的な暗号化または復号化を必要とするため、状態の一部しか取得できない場合は難しい。一方で、アルゴリズム 1 は、その一部の状態ビットから 1 つの線形近似特性を使用して鍵ビットを 1 ビット回復することができる。しかし、線形包現象と呼ばれる、同じ状態ビットを含むいくつかの強い特性がある場合では、アルゴリズム 1 も機能しない。Rock と Nyberg はこのアルゴリズム 1 を線形包現象が起きているケースにも拡張したが、その成功確率に関する評価及び解析は行われていなかった。

本論文では、理論的成功確率が実験結果と良く一致する新しい統計モデルを構築し、Matsui のアルゴリズム 1 の線形包現象に対する拡張の最初の正確な解析が提供された。またこの解析を、NIST 軽量暗号コンペティションのファイナリストの一つである、TinyJAMBU に適用し、部分鍵を回復する新しい暗号解析手法を提供している。

- **Correlation Cube Attack Revisited: Improved Cube Search and Superpoly Recovery Techniques [ASIACRYPT 2023]**

  Jianhua Wang, Lu Qin, Baofeng Wu

本研究ではある特別なキューブのインデックス集合（ISoC: Index Set of Cube）に関連する superpoly の低次数因子を効率的に抽出することでキューブ攻撃を発展させる。これは EUROCRYPT 2018 で提案された相関キューブ攻撃の特殊ケースと考えられるが、提案するフレームワークの下で、鍵変数に関するより有益な等式を鍵回復フェーズで取得できるという利点がある。提案攻撃を実行するためには、以下に示す 2 つの問題を解決する必要がある。1 つ目の問題は superpoly の代数標準形を効果的に復元するとともに、その低次数因子を抽出することである。2 つ目の問題は大量かつ良い性質を有する ISoC を効率的に探索することである。

これらの問題を解決するために、2 つの新しい技術を提案する。1 つ目の提案技術は変数置換技術である。これは鍵変数の複雑な表現を新しい変数に置換するとともに、中間ラウンドにおける内部状態の痕跡を排除するために使用される。新しい変数の導入によって superpoly をよりコンパクトに表現でき、結果として因数分解が容易になる。また、superpoly 復元の計算量を改善できるため、特殊な ISoC を効果的に識別できるようになる。2 つ目の提案技術はベクトル数値マッピング技術である。これは superpoly の次数評価における数値マッピング技術（CRYPTO 2019）の効率性と単項予測技術（ASIACRYPT 2020）の精度との間のトレードオフを模索した結果として得られた技術である。これらの技術を組み込むことで、高速枝刈り技術を MILP でモデル化できるようになり、代数次数が一定のしきい値を満たすような良い性質の ISoC をフィルタリングすることができる。また、自動化された MILP ソルバーのおかげで、探索空間全体にわたって適切なキューブを包括的に探索することが現実的に可能となった。

提案手法をストリーム暗号 Trivium に適用した結果、2020 年に Kesarwani らによって提示された 3 つのキューブに対する superpoly を復元できたが、Kesarwani らが主張するように、842 ラウンドまでのゼロサム特性が存在しないことを明らかにした。また、既存の最良かつ現実的な鍵回復攻撃は$2^{53.17}$の計算量で実行可能な 820 ラウンド Trivium に対するものであったが、本研究では 820、825、830 ラウンド Trivium に対して、それぞれ$2^{79.8}$、$2^{79.7}$、$2^{79.4}$個の秘密鍵を2^{60}の計算量で復元した。

- **Differential-Linear Approximation Semi-Unconstrained Searching and Partition Tree: Application to LEA and Speck [ASIACRYPT 2023]**

  Yi Chen, Zhenzhen Bao, Hongbo Yu

差分線形攻撃は ARX（Addition-Rotation-XOR）暗号に対する最も効果的な攻撃手法の 1 つである。しかしながら、次のような 2 つの技術的な問題により、この攻撃の効果を高めること、そしてこの攻撃をより多くのアプリケーションに対して適用することが困難であることが課題となっている。1 つ目の問題は、より良い差分線形近似を探索するための効果的な手法が存在しないことである。これは、既存手法を使用するとより良い差分線形近似の探索に多くの制約事項が必要となる、もしくは探索のための手法が非効率であることに起因する。2 つ目の問題は、差分線形攻撃のためのパーティション化技術には鍵回復攻撃の時間計算量を削減する可能性を秘めているが、ARX 暗号に対してパーティションを構築するための汎用的なツールが存在しないことである。

本研究はこれら 2 つの問題の解決を目指す。最初に、新たに考案した探索アルゴリズムに基づき、既知の差分線形近似からより良い差分線形近似を生成するための新しいアイデアを提案する。次に、ARX 暗号に対してパーティションを構築するために、パーティションツリーと呼ばれる汎用的なツールを提供する。これらの新しい技術に基づき、本研究では 2 つの ISO/IEC 標準暗号である LEA と Speck に対して、既存攻撃よりも優れた攻撃が実

行可能であることを示す。

LEA に対して初めて 17 ラウンドの識別子が構成できることを示した。これは既存の最良な識別子よりも 1 ラウンド長い結果となる。また、17 ラウンドの LEA-128、18 ラウンドの LEA-192、18 ラウンドの LEA-256 に対する初めての鍵回復攻撃も示した。これらの攻撃は既存の最良な鍵回復攻撃を 3〜4 ラウンド更新するものである。最後に、Speck48 と Speck64 に対して既存研究よりも優れた識別子が構成可能であることを発見した。さらに、Speck96 と Speck128 に対して初めての識別子を発見した。具体的には、11 ラウンドの Speck48、13 ラウンドの Speck64、15 ラウンドの Speck96、そして 18 ラウンドの Speck128 に対する識別子である。

- **Quantum Speed-Up for Multidimensional (Zero Correlation) Linear Distinguishers [ASIACRYPT 2023]**

  Akinori Hosoyamada

線形攻撃は共通鍵暗号解析の分野において最も重要な技術の 1 つである。Kaplan らは線形攻撃のための量子技術を用いた平方的高速化手法について提案した。しかしながら、彼らの提案手法は一般的な 1 次元の線形識別子を構成することを目的とした場合にのみ適用できることが知られている。古典的な攻撃において、攻撃にかかる計算量を削減するためにしばしば多次元線形近似が解析されることを踏まえると、Kaplan らの技術を多次元線形解析のために拡張可能かについて検討することは興味深いものと考えられる。

本研究では多次元（ゼロ相関）線形識別子と積分識別子を構成するための量子技術を用いた高速化手法について示す。1) Simon アルゴリズムのサブルーチンとフーリエ変換後の線形相関との間に深い関連性があることに着目する。Simon アルゴリズムはアダマール変換と対象となる関数へのオラクルクエリで構成されるサブルーチンを繰り返し実行するが、このサブルーチンを微修正することで、線形相関の二乗に比例する確率で線形マスクのペアを出力できることを発見した。このサブルーチンを相関抽出アルゴリズム（CEA）と呼ぶ。2) CEA と量子振幅増幅（QAA）技術を組み合わせて多次元線形識別子の構成が高速化可能であることを示す。例えば、メッセージサイズがnビット、ラウンド数をrとすると、Beyne による FEA-1 と FEA-2 への多次元線形識別攻撃にかかる計算量をそれぞれ$\mathcal{O}\left(2^{(r/4-3/4)n}\right)$から$\mathcal{O}\left(2^{(r/8-1/4)n}\right)$と$\mathcal{O}\left(2^{(r/6-3/4)n}\right)$から$\mathcal{O}\left(2^{(r/12-1/4)n}\right)$へ改善できる。3) CEA が多次元ゼロ相関線形識別の構成も高速化可能であることを示す。ラウンド関数が全単射でブロックサイズがnビットである 5 ラウンドのバランス型 Feistel 構造とタイプ I/II 一般化 Feistel 構造に対して、提案手法を用いると計算量$\mathcal{O}\left(2^{n/2}\right)$で量子識別子を構成することができる。4) 積分識別子の構成に関する高速化手法を示す。この高速化手法は積分特性とゼロ相関線形特性の深い関連性に依るものである。特に、単一の量子クエリによって、4 ビットセルの SPN 暗号における積分特性が 2.5 ラウンドの AES における積分特性と等価であることを明らかにした。5) CEA におけるアダマール変換を汎用的な量子フーリエ変換に

置き換えることで、提案手法は任意の有限アーベル群に対する汎用的な線形識別子の構成へと拡張できる。具体的には、Bayne による FF3-1 構造への識別攻撃を高速化できることを示した。

- **Exact Security Analysis of ASCON [ASIACRYPT 2023]**

  Bishwajit Chakraborty, Chandranan Dhar, Mridul Nandi

ASCON は NIST が主催する軽量暗号標準化プロセスの最終選考方式であり、Duplex 構造の認証暗号方式に加えて Sponge 構造のハッシュ関数を提供する。ASCON の認証暗号方式には ASCON-128 と ASCON-128a と呼ばれる 2 つのバリエーションがあり、過去には軽量な認証暗号アプリケーションとして CAESAR コンペティションの最終選考方式の 1 つに選出された。

この論文では、ランダム置換モデルにおける ASCON 認証暗号方式の安全性を厳密かつ網羅的に評価した結果を示す。ASCON（と Duplex 構造を持つ一般的な認証暗号方式）の偽造不可能性に関する既存の安全性は、Dをデータ計算量、Tを時間計算量、cをスポンジ構造のキャパシティとすると、$DT/2^c$で表すことができる。ここで、κを鍵サイズ、τをタグサイズ、bを基礎となる暗号学的置換のブロックサイズ（ASCON の場合は 320 ビット）とする。本研究では、Tの上界を2^κと2^cのうちの最小値、Dの上界を2^τと2^cのうちの最小値、DTの上界を2^bと限定した場合において、ASCON が AE 安全性（理想的な認証暗号との識別不可能性）を達成できることを示した。

軽量暗号標準化プロセスにおける NIST が提示した要求（$D \leq 2^{53}$、$T \leq 2^{112}$、$\kappa \geq 128$、$\tau \geq 64$）を考慮すると、キャパシティサイズが$c = 136$、タグサイズが$\tau = 64$において、ASCON の AE 安全性が保証されることを明らかにした。このパラメータ選択は 184 ビットという高いレートを実現できることから、ランダム置換モデルにおける ASCON の AE 安全性を損なうことなく、ASCON の効率性をさらに高めることができることを意味する。

- **Revisiting Higher-Order Differential-Linear Attacks from an Algebraic Perspective [ASIACRYPT 2023]**

  Kai Hu, Thomas Peyrin, Quan Quan Tan, Trevor Yap Hong Eng

本研究では、高階差分（HD）攻撃と高階差分線形（HDL）攻撃を代数的側面から見直すとともに、HD/HDL 識別子を検出するための 2 つの新しいツールを提供する。1 つ目のツールは Higher-order Algebraic Transitional Form (HATF) と呼ばれものであり、確率的 HD/HDL 近似を検出するために使用される。dを対象となる関数の代数次数とすると、一般的に HATF は計算量が$\mathcal{O}\left(2^{\ell+d2^\ell}\right)$となる$\ell$階 HDL 近似のバイアスを見積もることができる。もし対象となる関数が 2 次関数であるならば、計算量をさらに$\mathcal{O}\left(2^{3.8\ell}\right)$にまで削減可能できるため、ASCON や Xoodyak のようにラウンド関数を 2 次関数で表現可能な暗号方式に対

する HDL 攻撃に HATF が有効となる。また、2 つ目のツールは Differential Supporting Function (DSF) と呼ばれるものであり、暗号学的置換への入力に対して適切な線形近似を見つける便利な方法を提供することで、決定的 HD 識別子を容易に探索することを可能にする。

HATF を使用することで、ラウンドを削減した ASCON と Xoodyak の初期化フェーズにおける多くの HDL 近似を発見した。例えば、5 ラウンド ASCON に対しては 8 階 HDL 近似まで得ることができ、既存の最良な差分線形（DL）近似を使用した場合と比較して、識別攻撃の計算量を2^{16}から2^{12}まで削減した。また、単一鍵設定において、これまで 6 ラウンド ASCON と 5 ラウンド Xoodyak に対する DL 識別子は存在しなかったが、HATF を使用することで HDL 識別子が存在することを示した。HATF は DL 攻撃（つまり、1 階 HDL 攻撃）に対しても十分に機能し、ASCON や Xoodyak における既存の DL 近似を理論的観点から説明できるようになった。

DSF を使用することで、8 ラウンド ASCON-p（ASCON で使用される暗号学的置換）に対する新しい識別攻撃が実行可能であり、既存の最良な攻撃と比較して計算量を2^{130}から2^{48}まで削減した。また、フルラウンド ASCON-pに対するゼロサム識別攻撃も実行可能であり、既存の最良な攻撃と比較して計算量を2^{130}から2^{55}まで削減した。

- **More Insight on Deep Learning-aided Cryptanalysis [ASIACRYPT 2023]**

  Zhenzhen Bao, Jinyu Lu, Yiran Yao, Liu Zhang

CRYPTO 2019 において、Gohr は洗練されたニューラルネットワークを使用することで差分分布表（DDT）ベースの識別子よりも優れた暗号学的識別攻撃を実行できることを示した。これは差分ニューラル識別子（ND）が純粋な暗号文差分以外の追加情報を使用している可能性について提言するものである。しかしながら、この可能性に関する明示的な理論はまだ明らかとなっていない。

本研究では DDT と併用可能な明示的なルールを提供する。nをワードサイズとすると、この明示的なルールは法2^nの下での算術加算を通じて得られる XOR 差分伝搬の正しいペアにおけるビット値間の強い相関性に基づいており、純粋な DDT ベースの識別子と比べて差分ニューラル識別子の効果を高めることが期待できる。興味深いことに、これらのルールはマルチビット制約に関する先行研究（ASIACRYPT 2012、 CRYPTO 2013）や固定鍵差分確率の最新研究（CRYPTO 2022）と密接に関連している。対照的に、これらのルールの組み合わせでは ND のパフォーマンスは向上しないことが明らかになった。これはこれらのルールもしくはこれらのルールと同等の形式が ND によってすでに利用されているということを示唆するものであり、暗号解析分野におけるニューラルネットワークの威力が浮き彫りになったと考えられる。

さらに、本研究では、差分ニューラル識別子の精度と攻撃可能ラウンド数を向上させるためには、差分伝搬を制御することが不可欠であることを明らかにする。通常、秘密鍵に対し

て差分を埋め込むと、ブロック暗号のデータ処理部における内部状態の差分を打ち消すことができ、結果としてより強力な差分伝搬を得ることが可能となる。ただし、従来の攻撃とは異なり、差分ニューラル攻撃は出力差分を指定しないため、単一の差分伝搬に限定されないという特徴がある。つまり、鍵差分が差分ニューラル攻撃にとって有益かどうかが不明確であると言える。また、これに伴って、Speck が関連鍵設定での差分ニューラル攻撃に対してどの程度の耐性があるかも不明確であると言える。本研究では、14 ラウンド Speck32/64 に対する鍵回復攻撃を実行することにより、関連鍵設定での差分ニューラル攻撃が単一鍵設定の場合よりも強力であることを確認した。

2.2.2. 公開鍵暗号に関する解読技術

- **Post-Quantum Anonymity of Kyber [PKC 2023]**

  Varun Maram, Keita Xagawa

Kyber は、NIST の PQC 標準化プロセスで採用された鍵カプセル化メカニズム（KEM）であり、公開鍵暗号化（PKE）と key establishment の文脈で採用された唯一の方式でもある。NIST PQC の文脈における KEM とそれに関連する PKE 方式の主な安全性目標は、IND-CCA セキュリティであった。しかし、いくつかの重要な現代アプリケーションは、その基礎となる KEM/PKE スキームの匿名性を要求している（Bellare et al.） そのようなアプリケーションの例としては、匿名認証システム、暗号通貨、ブロードキャスト暗号化方式、認証された鍵交換、オークションプロトコルがある。よって、このような「IND-CCA を超える」アプリケーションに NIST の新しい PQC 標準が互換性を持つかを分析することは重要である。

Grubbs ら（EUROCRYPT 2022）と草川（EUROCRYPT 2022）は、ほとんどの NIST PQC 第 3 ラウンド候補 KEM の匿名性を研究してきた。しかし技術的な障壁のため、Kyber の匿名性を示すことができていなかった。

本論文ではこの障壁を克服し、Grubbs ら（EUROCRYPT 2022）と草川（EUROCRYPT 2022）が提起した未解決の問題を解決し、Kyber とそこから派生した（ハイブリッド）PKE スキームの匿名性を、ポスト量子設定において確立している。また、具体的評価付きの Kyber の IND-CCA 安全性証明を得るためのアプローチも提供される。これは、Kyber のポスト量子 IND-CCA 安全性の主張に関する前述の研究によって特定された別の問題を、証明可能安全性の観点から解決している。またこの結果は、NIST PQC 第 3 ラウンドのファイナリストである Saber にも同様の方法で適用されている。

- **An Efficient Key Recovery Attack on SIDH [Eurocrypt 2023]**

  Wouter Castryck, Thomas Decru

本論文で、超特異同種 Diffie-Hellman プロトコル(SIDH)に対する効率的な鍵回復攻撃が発表された。この攻撃は Kani の reducibility criterion に基づいており、アリスとボブがプ

ロトコル中に交換するねじれ点の像に強く依存している。開始曲線の自己準同型環の知識を前提とすれば、システムパラメータに依存する少数の整数の因数分解を除き、ヒューリスティックな多項式時間でこの攻撃は行われる。またこの攻撃は、パーティの一方が 2-同種を使用し、開始曲線が非常に小さな次数のスカラー倍ではない自己準同型を備えている場合、高速かつ簡単に実装可能である。これは、NIST 耐量子暗号標準化の第 4 ラウンドに進んだ SIDH のインスタンスである SIKE のケースに当てはまる。実際著者らは、セキュリティレベル 1 を目指す SIKEp434 をシングルコアのプロセッサにより 10 分程度で解読する Magma での攻撃の実装を行った。

- **A Direct Key Recovery Attack on SIDH [Eurocrypt 2023]**

  Luciano Maino, Chloe Martindale, Lorenz Panny, Giacomo Pope, Benjamin Wesolowski

本論文は、SIDH への攻撃論文である。任意の開始曲線の場合、提案されている攻撃は準指数計算量で実行可能である。開始曲線の自己準同型環が既知の場合、著者らの攻撃は、一般化リーマン予想を仮定した上で、多項式時間の複雑さを持つ。本攻撃は、Séta や B-SIDH など、秘密にしている同種の元での点の像を公開する同種ベース暗号システムにも適用できる。一方で、CSIDH、CSI-FiSh、SQISign には適用されないことも明記されている。

- **Breaking SIDH in Polynomial Time [Eurocrypt 2023]**

  Damien Robert

本論文では、仮に開始曲線がランダムな状況でも、SIDH は多項式時間で破ることができることを報告している。

- **New NTRU Records with Improved Lattice Bases [PQCrypto 2023]**

  Elena Kirshanova, Alexander May, Julian Nowakowski

NTRU に関する解析論文である。NTRU は 1998 年に生まれた格子ベース暗号の出発点である。NTRU の亜種である NTRU-HPS、NTRU-HRSS は NIST 耐量子暗号コンペティションのラウンド 3 のファイナリストであり、また CRYSTALS-KYBER や FALCON なども NTRU の影響を受けている。Coppersmith と Shamir は、格子基底の簡約を介して NTRU を攻撃することを提案し、その Coppersmith-Shamir 格子の亜種は、Security Innovations, Inc.による n=173 次元までの NTRU チャレンジを解読することに成功している。

本論文は、現代の NTRU バージョンを攻撃するツールを提供している。この攻撃ツールは、適切な格子の基底の設定と、G6K ライブラリの格子篩アルゴリズムを用いた現代の BKZ をチューニングすることで為される。この新しい格子基底を用いて、 NTRU-HPS に対して$n \in [101,171]$、NTRU-HRSS に対して$n \in [101,211]$の場合にn次元の暗号解析を実施している。特に既存手法では 172 日を必要とするのに対し、本方式では 83 日以内で NTRU –

HPS-171 インスタンスへの攻撃を実現している。

- **Do Not Bound to a Single Position: Near-Optimal Multi-positional Mismatch Attacks Against Kyber and Saber [PQCrypto 2023]**

  Qian Guo, Erik Mårtensson

耐量子暗号 Kyber と Saber についての暗号解析論文である。格子ベース KEM では、一時的な鍵の偶発的な再利用が生じた場合、その安全性を損ないう可能性があることが知られている。この鍵ペアを再利用する攻撃として、鍵不一致攻撃が知られている。

本論文では、NIST 耐量子暗号コンペティションにおける第 3 ラウンドの候補である Kyber と Saber に対する鍵不一致攻撃を提案している。著者らの方式は秘密鍵を回復するために必要なオラクルへのアクセス数を削減しており、これはサイドチャネル解析においても重要だと指摘している。

- **Rigorous Foundations for Dual Attacks in Coding Theory [TCC 2023]**

  Charles Meyer-Hilfiger, Jean-Pierre Tillich

コードベース暗号に対する dual 攻撃についての論文である。一般の線形符号の復号化やコードベース暗号のパラメータ選択において、情報集合復号（information set decoding, ISD）のテクニックは過去 60 年にわたり支配的であったが、近年、特定のパラメータについては、ISD のテクニックよりも、dual 攻撃の方が高性能であることがわかった。しかし、dual 攻撃の計算量解析はいくつかの仮定に依存しており、その仮定は実験的にも確認されているとは言い難い状況であった。この dual 攻撃は、格子ベース暗号における dual 攻撃をコードベース暗号に適応させた類似種として見ることができる。格子ベース暗号においても dual 攻撃は有力であることが指摘されており、ある種の確率変数の独立性が成立するかどうかについても同様にわかっていない。

本論文は、コードベース暗号における dual 攻撃において使用される基本的な量について、シンプルな代替表現を与えている。この代替表現によりは、上述の確率変数の独立性を考慮する必要がなくなる。そして、本論文によって、Asiacrypt 2022 で導入された dual 攻撃の問題点が明らかにされた。そして、この dual 攻撃で選択されたパラメータについて、独立性仮定に依存した解析では予想できない誤った候補を生成してしまうことを指摘している。さらにこの dual 攻撃の修正アルゴリズムの提案と、計算量の評価などを行っている。

2.2.3. ハッシュ関数に関する解読技術

- **Finding Collisions for Round-Reduced Romulus-H [FSE 2023]**

  Marcel Nageler, Felix Pallua, Maria Eichlseder

ハッシュ関数 Romulus-H に対する攻撃論文である。Romulus-H は NIST 軽量暗号コンペティションのファイナリストであり、Hirose DBL 構成に基づいている。これは理想的な

ブロック暗号で使用した場合は安全性が証明できるが、実際には理想的なブロック暗号は近似することしかできないため、具合的なインスタンスの暗号解析は注意深く行う必要がある。

本論文は、Romulus-H が SKINNY ブロック暗号で Hirose DBL を使用していることに着目して Romulus-H を解析し、衝突攻撃の新しい枠組みを構築している。結果として Romulus-H に対して、40 ラウンド中の 10 ラウンドまでのプラクティカルな衝突攻撃と、14 ラウンドまでのプラクティカルな semi-free-start 衝突攻撃を提案している。

- **Improved MITM Cryptanalysis on Streebog [FSE 2023]**

  Jialiang Hua, Xiaoyang Dong, Siwei Sun, Zhiyu Zhang Lei Hu Xiaoyun Wang

ISO 標準でもあるロシアの国家標準ハッシュ関数 Streebog に対する攻撃論文である。

本論文は、ASIACRYPT 2012 に Sasaki らによって導入された中間一致攻撃の拡張である guess-and-determine アプローチと、CRYPTO 2021 で Dong らによって導入された中間一致原像攻撃における手法を組み合わせている。さらに、既存の中間一致攻撃の MILP 自動ツールに基づき、この組み合わせによって得られる制約条件を加えた新しいモデルが構築されている。結果として、Streebog-512 に対する最初の 8.5 ラウンド原像攻撃と Streebog-256 に対する最初の 7.5 ラウンド原像攻撃が提案された。これは今までの攻撃を 1 ラウンド更新している。また時間計算量についても、既存の攻撃の時間計算量のより改善されている。

- **Quantum Attacks on Hash Constructions with Low Quantum Random Access Memory [ASIACRYPT 2023]**

  Xiaoyang Dong, Shun Li, Phuong Pham, Guoyan Zhang

ASIACRYPT 2022 において、Benedikt、Fischlin、Huppert は繰り返しハッシュ関数への量子ハーディング攻撃を初めて提案した。この攻撃は指数関数的な量子ランダムアクセスメモリ量（qRAM: quantum Random Access Memory）、具体的にはnをハッシュ関数の出力長とすると、$2^{0.43n}$の量子アクセス可能な古典メモリ（QRACM: Quantum Random Accessible Clasical Memory）が必要となる。大規模な qRAM の必要性は現実的ではなく、Benedikt らは少ない qRAM（low-qRAM）で実行可能な量子ハーディング攻撃の検討を今後の課題としている。本研究ではこの課題に対する回答を提示した。

1 つ目の貢献は、ASIACRYPT 2017 で提案された Chailloux らの衝突探索アルゴリズムに基づき、繰り返しハッシュ関数への新しい量子ハーディング攻撃アルゴリズムを提案したことである。提案する攻撃は Benedikt らの攻撃と比べて$2^{0.43n}$から$2^{0.46n}$とわずかに計算量が高くなるものの、qRAM を全く必要としないこと（no-qRAM）が特徴的である。

2 つ目の貢献は、hash XOR combiner、hash concatenation combiner、Hash-Twice、そして Zipper hash に対する様々な low-qRAM または no-qRAM な量子攻撃を提示したことである。hash XOR combiner に対する攻撃では、CRYPTO 2022 で提案された

Schrottenloher と Stevens の量子中間一致攻撃、2007 年に SIAM J. Comput.で提案された Ambainis の element distinctness アルゴリズム、そして SAC 2020 で提案された Jaques と Schrottenloher の衝突探索アルゴリズムに基づくものであり、これらの技術を発展させて 3 種類の異なる low-qRAM 量子原像攻撃を提案し、既存攻撃を改善できることを示した。hash concatenation combiner に対する攻撃では、新しい no-qRAM 量子衝突攻撃と no-qRAM 量子ハーディング攻撃を提案し、それぞれ既存攻撃を改善できることを示した。また、その他の重要な hash combiner である Hash-Twice と Zipper hash に対し、初めてとなる量子ハーディング攻撃を示した。

2.2.4. 署名に関する解読技術

- **A Key-Recovery Attack against Mitaka in the t-Probing Model [PKC 2023]**

  Thomas Prest

Mitaka は、Eurocrypt 2022 で提案された格子ベースの署名である。Mitaka の主な特徴は、高次で効率的にマスクできることで、サイドチャネル攻撃が懸念されるシナリオで耐性があるという点である。特に Mitaka は、t-probing モデルにおける安全性の証明が主張されてきた。

本論文では、Mitaka の安全性証明の欠陥を明らかにし、その後、t-probing モデルにおいて安全でないことを示す。4 以上の任意の共有数dについて、1 回の実行で$t < d$個の変数を probing することで、攻撃者は約 2^21 回の実行で効率的に秘密鍵を復元できる。さらに、攻撃者がd/tに線形な実行回数でアクセスできる限り、$t = 3$という定数値で十分であることも示される。

- **New Low-Memory Algebraic Attacks on LowMC in the Picnic Setting [FSE 2023]**

  Fukang Liu, Willi Meier, Santanu Sarkar, Takanori Isobe

耐量子署名形式 Picnic に対する安全性解析論文である。Picnic の安全性は、一つの平文と暗号文の対から LowMC の秘密鍵を回復することの困難性に強く関連している。特に Picnic3 で使用されている S-box 全レイヤをもつ LowMC に対して、既存の最高の攻撃は Dinur の多項式手法で得られている。また、部分的な非線形レイヤをもつ LowMC（例えば、Picnic2 で用いられているラウンドあたり 10 個の S-box を持つ LowMC）に対する最良攻撃である Banik らの攻撃は、中間一致法を用いている。

本論文では、3 ラウンドの S-box 全レイヤを持つ LowMC に対する鍵回復攻撃を提案している。この攻撃のメモリ計算量は非常に小さい一方、時間計算量は Dinur の手法で得られたものに近いため、既存の 3 ラウンド LowMC に対する攻撃の効率を改善している。また、提案手法の 4 ラウンド LowMC への拡張や、中間一致法で得られていた攻撃手法のメモリ計算量の削減なども提案している。

- **Faulting Winternitz One-Time Signatures to Forge LMS, XMSS, or SPHINCS+ Signatures [PQCrypto 2023]**

  Alexander Wagner, Vera Wesselkamp, Felix Oberhansl, Marc Schink, Emanuele Strieder

耐量子署名形式である SPHINCS+に対する故障注入攻撃に関する論文である。ハッシュベースの署名は耐量子署名形式として有力視されており、LMS、XMSS、SPHINCS+などが知られている。Winternitz ワンタイム署名（WOTS）は、これらすべてで使用される基本的なビルディングブロックの 1 つである。

本論文では、任意の平文に対する署名を偽造する攻撃を可能とする WOTS に標的とする故障注入攻撃を提案している。この故障注入攻撃は、WOTS 内のチェックサム計算を役に立たなくすることで、偽造攻撃を実現している。また本攻撃の実用性の推定や、署名生成または検証を実行する露出デバイスに備えるべき適切な対策も示している。

- **Algebraic Attacks on Round-Reduced Rain and Full AIM-III [ASIACRYPT 2023]**

  Kaiyi Zhang, Qingju Wang, Yu Yu, Chun Guo, Hongrui Cui

Picnic は MPC-in-the-Head パラダイムに従って設計された共通鍵暗号プリミティブに基づく署名アルゴリズムであり、NIST PQC における第 3 ラウンドの追加候補となっている。より安全で効率的な署名方式を設計するために、AES に基づく伝統的な一方向性関数を利用する、または LowMC、Rain、AIM のような低計算量の一方向性関数を利用することが最近の主流となっている。なお、LowMC、Rain、AIM はそれぞれ署名アルゴリズムの Picnic、Rainier、AIMer で使用される共通鍵暗号プリミティブである。Rainier と AIMer は MPC-in-the-Head に基づく署名方式の中でも最も効率的であると言われており、これらは耐量子デジタル署名アルゴリズムの有望な候補となっている。一方で、これらの署名アルゴリズムとその基礎となる共通鍵暗号プリミティブに対して安全性評価が十分ではなかった。

本研究では Rain と AIM に対する代数攻撃耐性の評価結果について示している。最初に、1 ラウンドに簡略化した Rain に対して、2^nの計算量で攻撃が実行できることを示した。なお、nはセキュリティパラメータであり、$n \in \{128, 192, 256\}$である。次に、2 ラウンドに簡略化した Rain に対して、それぞれ$2^{120.3}$、$2^{180.4}$、$2^{243.1}$の計算量で攻撃が実行できることを示した。さらに、192 ビット安全性を有する AIM バリエーションの AIM-III に対して、フルスペックにも関わらず$2^{186.5}$の計算量で攻撃が実行できることを示した。これらの攻撃は体上の冪函数の代数的構造に関する性質を利用したものである。最後に、提案する代数攻撃に対して AIM の安全性を保証するための対策案について提供した。

2.2.5. サイドチャネル攻撃の技術動向

- **Breaking and Protecting the Crystal: Side-Channel Analysis of Dilithium in Hardware [PQCrypto 2023]**

  Hauke Steffen, Georg Land, Lucie Kogelheide & Tim Güneysu

耐量子計算機署名方式 CRYSTALS-Dilithium に関するサイドチャネル攻撃論文である。CRYSTALS-Dilithium は NIST で標準化された耐量子計算機署名であり、ソフトウェア実装に対してはサイドチャネル攻撃による分析が行われてきた。

本論文では、ハードウェア実装に対する分析を実施し、脆弱性のある操作の解析と、Beckwith らによる最近のハードウェア実装に対する単純電力解析(Simple Power Analysis, SPA）と相関電力解析（Correlation Power Analysis, CPA）を行っている。SPA 攻撃は 700,000 のトレースのプロファイルを必要とし、プロファイリングが完了次第、1101 個のトレースの係数ペアを特定できる。CPA 攻撃は 66,000 のトレースに対して秘密係数を復元する。本論文はこれらの攻撃に対する具体的な対策も提示している。

2.2.6. その他に関する解読技術動向

- **The Hidden Number Problem with Small Unknown Multipliers: Cryptanalyzing MEGA in Six Queries and Other Applications [PKC 2023]**

  Nadia Heninger, Keegan Ryan

Backendal、Haller、Paterson は、クラウドストレージプロバイダの MEGA に悪用可能ないくつかの脆弱性を特定した。彼らは、悪意のあるサーバーが 512 回のログイン試行後にクライアントの RSA 秘密鍵を回復することができる RSA 鍵回復攻撃を実証した。

本論文では、MEGA のプロトコルの脆弱性によって明らかになった追加情報を利用し、秘密鍵を回復するために必要なクライアントのログイン回数がわずか 6 回である攻撃を行う方法が示される。この最適化された攻撃は、いくつかの暗号解析技術を組み合わせたものである。特に、未知の小さな乗数を持つ Hidden Number Problem の変種を定式化し、その解を与える。この問題に対する格子構成が、May と Ritzenhofen の因数分解問題に対する改善された結果を与えるために使用できることも示される。

- **Hull Attacks on the Lattice Isomorphism Problem [PKC 2023]**

  Léo Ducas, Shane Gibbons

格子同型問題(LIP)は、2 つの格子の間の同型性を求める問題であり、暗号の基礎として提案されている。この問題は、符号の等価性問題の格子変形であり、符号の hull という概念は壊滅的な攻撃につながる可能性がある。

本研究では、格子設定での hull、すなわちs-hull の適応における暗号解読的役割を研究する。まず、s-hull が算術識別器の作成に役立たないことが示される。これは、s-hull の genus が、sと元の genus から効率的に予測することができる、すなわち余分な情報を持たないた

めである。

その一方で、本論文では、hull は幾何学的な攻撃には利用可能であることが示される。これはある特定の格子では、hull の最小距離は元々の格子よりも比較的小さいことが原因となっている。これによる攻撃にかかる計算量は指数的なままだが、その指数に含まれる定数が半分になる。この 2 つ目の結果は、Ducas & van Woerden が提案した LIP の一般的な困難性予想に対する反例となっている。

以上の結果は、暗号のための LIP インスタンス化においては、hull の幾何に十分な配慮する必要があることを示唆する。また、unimodular 格子は、その自己双対性から、攻撃者に元の格子しか残さないため、興味のあるオプションとなることも指摘されている。特筆すべきこととして、これは提案されているインスタンス化、すなわちトリビアル格子Z^nと Barnes-Wall 格子がすでにそのケースに当てはまっていることも指摘されている。

- **Caveat Implementor! Key Recovery Attacks on MEGA [Eurocrypt 2023]**

  Martin R. Albrecht, Miro Haller, Lenka Mareková, Kenneth G. Paterson

MEGA は大規模なクラウドストレージおよび通信プラットフォームであり、保存データのエンドツーエンドの暗号化を提供することを目的としている。Backendal, Haller and Paterson による最近の分析（IEEE S&P 2023）では、MEGA サービスプロバイダが搭載可能な MEGA に対する実用的な攻撃を提示し、これらのセキュリティ主張を無効とした。これに対し、MEGA 開発者は、MEGA で使用されるユーザ RSA 秘密鍵に、以前の攻撃を防ぐのに十分な軽量のサニティチェックを追加している。

本論文では、これらの新しいサニティチェックを分析し、それ自体が、元の攻撃よりもわずかに高い攻撃計算量で、ターゲットユーザの RSA 秘密鍵を回復する MEGA への新しい攻撃を実行するために悪用する方法が紹介されている。このオラクルは、ターゲットユーザの RSA 秘密鍵を、選択したデータで部分的に上書きする能力を攻撃者に提供しうる。さらに、MEGA のユーザ認証手続き中のサニティチェックと、その後の暗号処理で生じる異なるエラー状態を利用する、2 つの異なるタイプの攻撃が紹介されている。1 つめは、$u = q^{-1}$を再計算する際に、MEGA コードがモジュール反転を処理する方法を悪用する。2 つ目は、small subgroup attack (van Oorschot and Wiener, EUROCRYPT 1996, Lim and Lee, CRYPTO 1998)の亜種と見做せる。これらの攻撃は本論文において試作され、実際に動作することも示されている。副次的貢献として、MEGA の未パッチ版に対する Backendal-Haller-Paterson の RSA 鍵回復攻撃を改良し、元の 512 回のログインではなく、2 回のログインしか必要としない方法も提案されている。

- **Generic Attack on Duplex-Based AEAD Modes using Random Function Statistics [Eurocrypt 2023]**

  Henri Gilbert, Rachelle Heim Boissier, Louiza Khati, Yann Rotella

十分に大きな鍵長を持つ二重認証暗号モードは、cをキャパシティとしたとき、誕生日境界$2^{c/2}$まで安全であることが証明されている。しかし、この境界は最適であることは知られていない。実際、最もよく知られた一般的な攻撃計算量は$2^c/\alpha$である（ここでαは小さなセキュリティ損失係数)。本論文では、いくつかの二重通信ベースの AEAD モードに対する新しい一般的な攻撃について説明される。提案された攻撃は計算量$O(2^{3c/4})$を持つ。さらに、無視できる量の追加計算で、秘密鍵を回復することもできる。特に、NIST の軽量化コンペティション候補である Xoodyak の設計者が主張した安全性を、本攻撃は破っている。

- **Context Discovery and Commitment Attacks: How to Break CCM, EAX, SIV, and More [Eurocrypt 2023]**

  Sanketh Menda, Julia Len, Paul Grubbs, Thomas Ristenpart

近年行われていた一連の研究は、コンテキストコミットメントのセキュリティの重要性を強調している。これは、認証付き暗号（AEAD）方式が、(秘密鍵、関連データ、ナンスのどれかから) 攻撃者により選ばれた二つの異なる文脈で、攻撃者が選択した暗号文を復号しないことを求めている。最近の攻撃にもかかわらず、コンテキストコミットメントに関する多くの未解決の問題が残っている。CCM、EAX、SIV などの重要なスキームのコミットメントセキュリティについても、ほとんど何も知られていなかった。

本論文ではこれらの未解決の疑問点が解決される。まず、文脈のどの部分が敵対的に制御されているかという観点から、コンテキストコミットメントの安全性をより詳細に定義するのに役立つ新しいフレームワークが導入される。さらに、文脈発見可能性（context discoverability）と呼ばれる新しいセキュリティ概念が定式化され、制限のないコンテキストコミットメント安全性（すなわち、攻撃者が 2 つの文脈のすべてを制御する）が、実際に使用されるほとんどの方式を含むクラスに対して、文脈発見可能性セキュリティを意味することを示す。さらに、CCM、EAX、SIV、GCM、OCB3 を含む幅広い AEAD 方式に対する新しいコンテキスト発見攻撃が示される。SIV モードに対する制限付きコンテキストコミットメントの安全性も研究され、一般化誕生日問題に対する Wagner の k-tree アルゴリズムを使って、計算量$O(2^{n/3})$の新しい攻撃も与えられている。

- **Disorientation Faults in CSIDH [Eurocrypt 2023]**

  Gustavo Banegas, Juliane Krämer, Tanja Lange, Michael Meyer, Lorenz Panny, Krijn Reijnders, Jana Sotáková, Monika Trimoska

本論文では、CSIDH の族に対する新しいクラスの故障注入攻撃について研究されている。この攻撃は、いくつかの同種ステップにおいて、その方向を効果的に反転させるものであり、群作用の評価中に行われる Legendre 記号または Elligator 計算に関連する特定のサブルーチンに障害を与えることで行われている。これらのサブルーチンは、ほぼすべての既知の CSIDH 実装に存在するものである。そして、故障を持つサンプルのセットを後処理するこ

とで、秘密鍵に対する制約が推測できる。詳細は実装に依存するが、多くの場合、わずかな数の故障注入の成功と、わずかな計算資源で、完全な秘密鍵を回復できることが示されている。また、オリジナルの CSIDH の PoC 実証ソフトウェアと、CTIDH の一定時間実装を攻撃するための完全な詳細も提供されている。また、この攻撃に対する一連の簡単な対策も提示され、その安全性についても議論されている。

- **On the Hardness of the Finite Field Isomorphism Problem [Eurocrypt 2023]**

  Dipayan Das, Antoine Joux

  有限体同型問題（FFI）は、平均計算量を安全性の根拠とする格子問題（LWE、SIS、NTRU）などの代替えとして PKC'18 で導入された。同論文ではその応用として、FFI 問題を用いた完全準同型暗号方式も構築している。

  本論文では、FFI 問題の決定亜種が、標数$q = \Omega(\beta n^2)$において多項式時間で解けることを証明する（ここでq, β, nは FFI 問題のパラメータ）。この FFI 識別器の結果を用いて、完全準同型暗号のセマンティックな安全性に対する多項式時間攻撃も提案されている。また、いくつかの FFI 問題の亜種を、これまで知られていなかったq-ary 格子問題として記述する方法を示す。その結果、これまで難解であったいくつかのパラメータに対する探索問題を、簡単な格子簡約アプローチで解くことができるようになった。

- **A New Algebraic Approach to the Regular Syndrome Decoding Problem and Implications for PCG Constructions [Eurocrypt 2023]**

  Pierre Briaud, Morten Øygarden

  約 20 年前に Augot らによって導入された正則シンドローム復号化（RSD）問題は、特定の誤差分布を持つシンドローム復号問題の変種である。この問題では、誤差ベクトルは等しい大きさのブロックに分割され、それぞれが 1 つのノイズ座標を含んでいる。最近、MPC や ZK のアプリケーションで使用されているため、この仮定への関心が高まっている。この文脈では「規則的なノイズを含む LPN」と呼ばれるこの仮定により、通常の LPN と比較してより優れた効率が達成されている。これまでの暗号解読の研究において、この問題の特殊性を利用した攻撃方法は示されていない。

  本論文では、RSD に対する最初の代数的攻撃が提示される。基礎となる多項式システムの入念な理論解析に基づき、規則的なノイズ分布を利用することができる具体的な攻撃が提案される。特に、他のアルゴリズムを凌駕するような具体的なパラメータの例がいくつか挙げられている。

- **Subverting Telegram's End-to-End Encryption [FSE 2023]**

  Benoît Cogliati, Jordan Ethan, Ashwin Jha

  セキュアメッセージングサービス Telegram に関する論文である。Telegram は人気の高

いセキュアなメッセージングサービスで、2021 年時点で 3 番目に大きなユーザベースを持っている。

本論文は、大規模監視の存在下における Telegram のエンドツーエンド暗号化（E2EE）プロトコルのセキュリティを解析している。具体的には、 Telegram の E2EE プロトコルは、効率的なアルゴリズム置換攻撃を受けやすいことを示す。基礎となる認証された暗号化スキーム MTProto2.0 に対して効率的なアルゴリズム置換攻撃を提供し、少数のクエリと低いレイテンシで、大量の暗号鍵材料を非常に高い確率で回復する。この攻撃はランダムなパディング長とパディング値の選択における MTProto2.0 の自由度を利用しており、著者らは Telegram が MTProto2.0 のパディング方法を改訂することを強く推奨している。本論文は同時に、パディング記述における軽微な変更がこの攻撃に対する耐性を与えていることも示している。さらに、MTProto2.0 における基本動作モードを一般化した MTProto-G を提案し、それがマルチユーザ安全決定論的認証暗号方式であることも示している。

- **・Classical and Quantum 3 and 4-Sieves to Solve SVP with Low Memory [PQCrypto 2023]**

  André Chailloux, Johanna Loyer

格子ベース暗号の安全性において重要である最短ベクトル問題（SVP）に関する論文である。d次元 SVP について既知の最速攻撃は、格子篩（lattice sieving）により行われ、定数t, mに対して時間計算量$2^{td+o(d)}$かつメモリ計算量$2^{md+o(d)}$で行われる。
格子篩における簡約ベクトルの探索は、与えられた制約を満たすk個のベクトルの配置問題に帰着される。

本論文は、このk-格子篩アルゴリズムのためのフレームワークを提示する。このフレームワークでは、入力となるベクトルのリストをフィルタリングし、総和が零ベクトルとなるk個の符号語を中心とするリストを得た後、フィルタされたリストにおけるシンプルな配置問題を解くことで為される。結果として、$k = 3$の時の時間計算量の改善と、$k = 4$の時の新しいトレードオフを与えている。

- **Fast Enumeration Algorithm for Multivariate Polynomials over General Finite Fields [PQCrypto 2023]**

  Hiroki Furue, Tsuyoshi Takagi

多変数多項式の出力を列挙するアルゴリズムに関する論文である。多変数多項式の出力の列挙は、多変数公開鍵暗号に対する代数的攻撃において重要な問題である。

一般に、有限体$\mathbb{F}_q$上のn変数d次多項式に対する出力列挙アルゴリズムは、$O(\binom{n}{k}q^n)$回の操作が必要である。Bouillaguet らは、CHES 2010 にて、$q = 2$の場合の計算量$O(d \cdot 2^n)$である高速列挙アルゴリズムを提案した。このアルゴリズムは Gray 符号の順序に従って与えられた多項式の入力をカバーしている。

本論文では、qが一般の場合にこの結果を拡張し、計算量$O(d \cdot q^n)$である高速列挙アルゴリズムを提案している。提案されたアルゴリズムは、すべての入力をカバーするためにGrayコードの代わりに辞書式順序を用いており、特に$q = 2$の場合でもBouillaguetらによるアルゴリズムとは異なる。

- **Breaking the Quadratic Barrier: Quantum Cryptanalysis of Milenage, Telecommunications' Cryptographic Backbone [PQCrypto 2023]**

  Vincent Quentin Ulitzsch, Jean-Pierre Seifert

携帯電話ネットワークに対する量子暗号解析の論文である。現在の携帯電話ネットワークの暗号化に用いられているMilenageアルゴリズムは、7つの秘密鍵アルゴリズムを中心に、認証と鍵同意が行われている。このアルゴリズムはまだ量子暗号解析の対象になっていなかったが、その一方で近年、量子計算機を用いた対称鍵暗号への攻撃の二次以上の高速化が進んでいる。

本論文では、Milenageアルゴリズムに対する量子解析を実施し、攻撃の二次以上高速化した。特に、異なる量子攻撃モデルによって識別可能な指数的高速化を含む、すべてのMilenageアルゴリズムに対する量子攻撃シナリオが提供され、Milenageの量子攻撃に対する構造的弱点を指摘している。

- **Time and Query Complexity Tradeoffs for the Dihedral Coset Problem [PQCrypto 2023]**

Maxime Remaud, André Schrottenloher, Jean-Pierre Tillich $\mathbb{Z}_N$における二面体群コセット問題（DCP）は、LWEが帰着できる問題であるため、耐量子暗号における安全性において重要である。Ettinger-Høyerアルゴリズムは線形回のクエリでDCPを解くことが知られているが、時間計算量は指数回数必要である。最初の時間効率の高いアルゴリズムはKuperbergによって導入された。これらのアルゴリズムは、準指数時間計算量と$\tilde{O}(2^{\sqrt{\cdot c \log N}})$回のクエリで実行される。ここで$c$はある定数である。このKuperberg流のふるい分けアルゴリズムは、量子及び古典における時間、メモリ、クエリの間のトレードオフを示している。これらのトレードオフは、クエリコストを抑えたい攻撃者にとってクエリ回数を削減することを可能にする。

それを踏まえ本論文は、Ettinger-HøyerアルゴリズムとKuperbergアルゴリズムを補間する方法を示している。この手法によって、線形クエリ-指数時間の攻撃と、準指数クエリ-準指数時間の攻撃を滑らかに補間することが可能となり、従って、クエリコストを考慮した攻撃のファインチューニングを可能としている。

- **Exploiting the Symmetry of $¥mathbb{Z}^n$: Randomization and the Automorphism Problem [ASIACRYPT 2023]**

Kaijie Jiang, Anyu Wang, Hengyi Luo, Guoxiao Liu, Yang Yu, Xiaoyun Wang 格子暗号の多くは、SVP や CVP 等の格子点探索問題の困難性を安全性の根拠としている。その一方で、他の計算問題を根拠とした暗号の開発も進められている。格子同型問題（LIP: Lattice Isomorphism Problem）は 2 つの格子L_1, L_2が与えられたときにそれらが直交変換により写像されるかどうかを問う問題である。インスタンスは 2 つの基底行列B_1, B_2の形で与えられ、$OB_1 = B_2$を満たす直行行列、つまり$O^T O = I$を満たすある行列Oを求めることになる。この問題の困難性を用いた公開鍵暗号（Bennett et al.、ePrint 2021/1548）、署名(Ducas et al.、Asiacrypt 2022)などの基本的な暗号プリミティブがここ数年で提案されている。特に、$L_2 = \mathbb{Z}^n$（整数格子）に固定した LIP はℤLIP と呼ばれ、この形の問題が過去の格子暗号の解析において暗に用いられてきた歴史がある。

$\mathbb{Z}^n$を直交変換により回転させた格子は著しい対称性を持ち、特に格子の自己同型群 Aut(L)は符号付き置換群と同型であることが知られる。この性質を用いることでℤLIP に関連する計算問題の困難性を調査したことがこの論文の主結果である。上に述べた LIP、ℤLIP 以外で扱われている計算問題と困難性に関する結果を以下にまとめる。

- ℤSVP、γ- ℤSVP : $\mathbb{Z}^n$と同型であることが知られている格子Lに対して、最短ベクトルを求める問題、およびそのγ近似を求める問題。$\gamma = O(1)$の場合には困難性はℤSVP と等しい。ℤSVP≧ℤLIP。
- SCVP、 ℤSCVP：自身とその双対格子が等しい格子をユニモジュラ―格子と呼び、ユニモジュラ―格子内のベクトル$w \in L$が任意の$v \in L$に対して$\langle w, v\rangle \equiv \langle v, v\rangle \bmod 2$を満たすときに特性ベクトル(characteristic vector)であると呼ばれる。SCVP は与えられたユニモジュラ―格子の中で最短の特性ベクトルを求める問題であり、ℤSCVP は格子の種類を$\mathbb{Z}^n$と同型なものに制限したものである。ℤSCVP≧ℤLIP。
- LAP、 ℤLAP : 格子基底 B が与えられたときに、その格子の自己同型群 Aut(L)の中で非自明なものを求める問題。ℤLAP は格子の種類を$\mathbb{Z}^n$と同型なものに制限している。ℤLAP≡ℤLIP、LAP≡LIP
- HSP:隠れ部分群問題。群Gとそれを定義域とする関数fが、ある部分群 H に対して$f(g_1) = f(g_2) \Leftrightarrow g_1 H = g_2 H$を満たすとき、$f$を計算するオラクルへのアクセスから$H$の生成元を求める問題。$G = GL_n(\mathbb{Z})$かつ$H = Stab(B^T B) \leq G$の場合にℤLIP からの帰着が存在する。

技術的には randomized reduction framework と呼ばれる、離散ガウス分布を用いた格子のランダム基底生成を利用したランダム帰着である。上に述べた帰着以外にも、n次元のℤLIP から$(n/2)$次元の SVP、ℤLIP およびℤSVP から O(1)-uSVP への帰着が知られている。

- **Non-Interactive Commitment from Non-Transitive Group Actions [ASIACRYPT 2023]**

  Giuseppe D'Alconzo, Andrea Flamini, Andrea Gangemi

群作用に関する困難性仮定は耐量子計算機暗号に良く用いられる。実際、CSIDH では、楕円曲線上の同種写像に基づいた仮定から、プリミティブを構成している。さらに、テンソルや、符号問題に基づいたプリミティブもある。

本論文では、群作用に基づいたビットコミットメント法を提案している。提案法は、非推移的な群作用に基づいて構成され、Decisional Group Action Inversion 問題を安全性の根拠とし、標準モデルで安全性が証明できる。

既存のビットコミットメント法は、コミットメントのフェーズで、送信者と受信者の間で通信を行う interactive な手法であったが、本提案方式は non-interactive な方式である。さらに、送信者が honest である場合は、2 つのコミットメントがあった場合に、同じ入力値から作られたコミットメントかどうかを、入力値を示すことなく示すことができる、という特徴（Linkable Commitment）も有する。そして、このようなコミットメントが持つべき安全性を新規に定義した。

- **Concrete Analysis of Quantum Lattice Enumeration [ASIACRYPT 2023]**

  Shi Bai, Maya-Iggy van Hoof, Floyd B. Johnson, Tanja Lange, Tran Ngo

格子の SVP、 CVP を解く基本的なアルゴリズムである格子点列挙(Lattice Enumeration)アルゴリズムは深さ優先の木探索アルゴリズムとして実装されることが多く、その量子版アルゴリズムを考えることは安全性評価にとって重要である。深さn、大きさTの木を探索する量子アルゴリズムは(Montanaro、 ToC 2018)および(Ambains-Kokainis、STOC 2017)による量子 Tree Backtrack を用いた$O(\sqrt{T} * poly(n))$時間のものが知られており、いくつかの先行研究において格子探索への応用が言及されていた。初期の具体的な計算量の見積もりとして(Aono-Nguyen-Shen、Asiacrypt 2018)による枝刈りのフレームワークと組み合わせた場合の漸近的な計算量の解析が存在する。論文の主な結果として、上記 Montanaro、 Ambains-Kokainis らのアルゴリズムを組み合わせた量子木探索の詳細な回路と計算量評価を与えている。類似研究として量子回路の深さを制限した場合の古典-量子ハイブリッドアルゴリズムによる格子点探索の計算量を扱った(Bindel et al., ePrint 2023/1423)が存在する。

量子回路の設計は“Clifford+T”と呼ばれる標準的な手法を用いている。これは量子万能回路が Clifford gates と呼ばれる量子ゲートのセット（H、S、CNOT が代表元として取られることが多い）と、非 Clifford ゲートである$T = \sqrt{S}$を用いて構成可能であることから、以上の 4 種のゲートを用いて回路を構成するモデルである。また、Tゲートの低誤り実装が他の Clifford gates と比べて困難であることからTゲートの個数、深さを用いて量子計算量の見積もりがなされることが多い。なお、論文内の実際の設計では非 Clifford ゲートとして

Toffoli ゲートも用いており、リソースの見積もりの段階で 1Toffoli=4T (Selinger、 Phys. Rev. A87 042302)の変換を行っている。

著者らはまず回路のTゲートの個数と深さを、木のサイズ$\mathcal{T}$、格子次元n、ノードから伸びる子の数の最大値d、浮動小数点の精度p、入力格子中の成分の最大値Tを含む複雑な式により評価し、次に先行研究および論文内の計算機実験によるヒューリスティックな評価を代入することで漸近計算量$\sqrt{\mathcal{T}n}$からのオーバーヘッドを正確に見積っている。深さとサイズのオーバーヘッドはそれぞれ$128cn^3 \log n$と$10752n^6$としている。ここで、c は$\log \mathcal{T} \approx c \cdot n \log n$となる支配的項であり、既存の研究では 0.125 程度と見積もられているが著者らは懐疑的である。

- **Solving the Hidden Number Problem for CSIDH and CSURF via Automated Coppersmith [ASIACRYPT 2023]**

  Jonas Meers, Julian Nowakowski

HNP(Hidden Number Problem)は$\mathbb{Z}_p$において複数の整数の組$\left(t_i, \mathrm{MSB}_k(\alpha \cdot t_i \bmod p)\right)$が与えられたときに整数$\alpha$を復元する問題である。ここで、$\mathrm{MSB}_k$は入力の上位$k$ビットを返す関数であり$k$の大きさにより問題の困難性が異なる。Diffilie-Hellman 鍵共有において、g^{ab}の上位$k \geq \sqrt{\log p}$ビットから HNP を経由して残りのビットを復元する古典的な結果(Boneh-Venkatesan、 CRYPTO 1996)をベースとして、ECDSA、SGX、DSA、qDSA などの様々な状況において Computational Diffie-Hellman 問題の部分解導出の困難性や、暗号方式のサイドチャネル攻撃への耐性を議論するために気論が拡張されている。これらは$\mathrm{MSB}_k(\alpha \cdot t_i \bmod p)$関数の$\alpha \cdot t_i$の部分を対象となる Diffie-Hellman 鍵共有の群に合わせて適切な多項式関数$f(\alpha, t_i)$とし、$\mathrm{MSB}_k(\cdot)$の値からαを復元する問題は mod Nにおける多変数連立方程式の小さい解を求める問題へと変換される。方程式の求解は Coppersmith 法と呼ばれるヒューリスティックな多項式時間アルゴリズムを用いて解かれる。

この論文では同種写像暗号の一種である CSIDH と CSURF で用いる楕円曲線群に合わせ HNP の変種である CI(Commutative Isogeny)- HNP を定義し、Coppersmith 法を用いた場合に多項式時間で解けるパラメータの範囲について議論している。主な結果として、CSIDH、CSURF に対してそれぞれ共有鍵となる曲線$E_{AB} = CDH(E_A, E_B)$を表現するそのモンゴメリ係数(M_0, M_1)の上位kビットを与えるオラクルにアクセス可能な場合に、それぞれ$k \geq 13/24$、 31/41 であれば多項式時間での完全復元が可能であることを示した。

技術的には CSIDH、CSURF に対する CI-HNP に対応する方程式はそれぞれ 3 変数 3 制約式の 2 次方程式と、2 変数 1 制約式の 3 次方程式であり、Coppersmith 法における多項式格子の構成における自動化手法の提案と Sage による計算機実験を行い、ソースコードを公開している。

- **Memory-Efficient Attacks on Small LWE Keys [ASIACRYPT 2023]**

Andre Esser, Rahul Girme, Arindam Mukherjee, Santanu Sarkar

LWE 問題は NIST 標準化方式の CRYSTALS-Kyber、 Dilithium をはじめとする多くの代表的な格子暗号の安全性の根拠として用いられている。近年ではオリジナルの LWE 暗号(Regev、 STOC 2005)のようにエラー分布に離散ガウス分布を用いることは少なく、実装上の都合から$\{-t,\dots,t\}^m$のような制限された区間上の確率分布を用いることが多い。この論文では、この種の LWE を small max norm LWE 問題として定式化し、nested collision search と呼ばれる新たな組み合わせ論的アルゴリズムを提案している。これは、先行研究で用いられてきた meet-in-the-middle 系列のアルゴリズムよりも少ない多項式サイズでのメモリで動作するが、衝突探索を並列化することで時間空間計算量トレードオフを考えることも可能である。

論文が対象としているのは Ternary LWE と呼ばれる、$\boldsymbol{s},\boldsymbol{e}$の成分がともに{-1,0,1}からサンプリングされたときに$(A,\boldsymbol{b}=A\boldsymbol{s}+\boldsymbol{e})$から$\boldsymbol{s}$を復元する問題である。

原理は以下のようになる。Odlyzko ハッシュ関数の変種で、一部の成分を取り出した関数$f_1(\boldsymbol{x})=h(A\boldsymbol{x}), f_2(\boldsymbol{y})=h(\boldsymbol{b}-A\boldsymbol{y})$ をランダムにサンプリングしたベクトル$\boldsymbol{x},\boldsymbol{y}$に対して計算を行い、衝突$f_1(\boldsymbol{x})=f_2(\boldsymbol{y})$かつ$\boldsymbol{x}+\boldsymbol{y}$が Ternary になっている時点で$\boldsymbol{x}+\boldsymbol{y}$が問題の解となっている確率が高いという事実を利用する。ここで、ハッシュ関数の衝突検索には通常 meet-in-the-middle 法のように大量のメモリ空間を必要とするが、 ρ 法を用いることにより多項式空間で動作するものが得られる。上記分割は$\boldsymbol{x}+\boldsymbol{y}$の 2 つのみであったが、4 分割およびそれ以上による nested collision search を用いることも可能である。この場合には最初のレイヤで$\boldsymbol{x}=\boldsymbol{x_1}+\boldsymbol{x_2},\ \boldsymbol{y}=\boldsymbol{y_1}+\boldsymbol{y_2}$の再分割を行い、それぞれの部分空間で発見した衝突の情報をもとに上のレイヤで結合を行う。合計 3 回の ρ 法の呼び出しが必要となるが、指数関数的な計算時間の改良が可能である。

同様のアイデアは$\boldsymbol{s},\boldsymbol{e}$を{-2,…,+2}や{-3,…,+3}からサンプリングしている CRYSTAL-Dilithium の解析にも適用可能であり、多項式空間アルゴリズムの計算量が$2^{1.742n}$から$2^{1.282n}$へと大幅に改善した。

- **Too Many Hints - When LLL Breaks LWE [ASIACRYPT 2023]**

  Alexander May, Julian Nowakowski

格子暗号へのサイドチャネル攻撃の文脈から、LWE 問題において秘密ベクトル$\boldsymbol{s}$およびエラーベクトル$\boldsymbol{e}$の部分情報を知ることでどの程度問題が簡単になるのかを調査することは暗号学的に興味深い課題である。この論文では、$\langle\boldsymbol{v},\boldsymbol{s}\rangle=\ell$を満たす組$(\boldsymbol{v},\ell)$を perfect hint、$\langle\boldsymbol{v},\boldsymbol{s}\rangle=\ell \bmod m$を満たす組$(\boldsymbol{v},\ell,m)$を modular hint と呼び、これらの値がどの程度集まれば元の LWE 問題の困難性が著しく下がるのかが議論されている。

特に、CRYSTALS-Kyber の 512 次元パラメータにおいて 449 個の modular hint があれば LLL アルゴリズムのみで鍵復元が可能である事、200 個程度の perfet hints があれば LLL もしくは BKZ-20 程度の弱い基底簡約アルゴリズムのみで鍵復元が可能であることが

実験的に示されている。技術的には Perfect hint、 Modular hint の列から LWE の次元削減を行い、また Perfect hint を用いて格子の体積を上げることで疎な格子を構成する。両者とも格子点探索の計算量を下げる方向に格子を変換する手法である。

資料３－３

2023 年度暗号技術調査 WG（耐量子計算機暗号）活動報告

1　2023 年度暗号技術調査 WG（耐量子計算機暗号）活動経緯と活動内容の概要

2020 年度第 2 回暗号技術検討会において、2021 年度から暗号技術評価委員会の活動計画として 2 年をかけて PQC の研究動向を調査し、ガイドラインを作成することが決定された。暗号技術評価委員会は 2021-2022 年度に暗号技術調査ワーキンググループ(耐量子計算機暗号) を設置し、ガイドライン及び調査報告書を作成、公開した。

その後も、PQC 関連の技術開発、標準化活動が世界的に活発であることから、引き続き、暗号技術調査ワーキンググループ（耐量子計算機暗号）（以下、PQC WG）を設置して下記 2 点の活動を行うことが 2023 年度第 1 回暗号技術評価委員会において承認された。

(1) NIST の PQC 標準化において第 4 ラウンドが進行中であることをはじめ耐量子計算機暗号に関する技術開発、標準化活動が引き続き世界的に活発であることから、動向を 2023 年度から 2 年間かけて調査・把握し、ガイドラインの改定を行う。

(2) 「素因数分解の困難性に関する計算量評価」、「楕円曲線上の離散対数計算の困難性に関する計算量評価」の予測図の更新についても検討し、更新する。

2　ＷＧ委員の構成（敬称略）

主査：國廣 昇　　(筑波大学)
委員：青木 和麻呂　　(文教大学)
委員：伊藤 忠彦　　(セコム株式会社)
委員：下山 武司　　(国立情報学研究所)
委員：高木 剛　　(東京大学)
委員：高島 克幸　　(早稲田大学)
委員：成定 真太郎　　(KDDI 総合研究所)
委員：廣瀬 勝一　　(福井大学)
委員：安田 貴徳　　(岡山理科大学)
委員：安田 雅哉　　(立教大学)

3　耐量子計算機暗号ガイドラインの作成

3.1　スケジュール（2023 年度第 1 回暗号技術評価委員会で承認）

年度	回	耐量子計算機暗号ガイドラインの議論・決定・報告
2023 年度	第 1 回 2023/9/13	✓ 追記・改定の方針について議論 ✓ 執筆担当者を議論
	第 2 回 2024/1/19	✓ 追記・改定すべき項目及びその章立ての決定 ✓ 調査の中間報告
2024 年度	第 1 回 (8 月頃を想定)	✓ 中間報告、追加及び削除すべき暗号方式があれば議論
	第 2 回 (2 月頃を想定)	✓ 内容の確定

3.2　第 1 回 WG（9/13）での実施内容及び決定事項

- ガイドライン及び調査報告書の作成
  - 2022 年度版ガイドライン、調査報告書をベースに 2024 年度版ガイドライン、調査報告書を作成する。改訂扱いではなく新規の扱いとすることで合意した。
  - 「PQC の活用方法」の章は 2022 年度版ではガイドラインのみであったが、2024 年度版からは調査報告書にも含めることで合意。それに合わせて内容を拡充し、ガイドラインには公知の事実のみを載せ、詳細は調査報告書に載せる。
  - 以下に例示したいくつかの暗号方式の扱いに関しては今後の動向を注視し、2023 年度第 2 回以降の WG で改めて議論を行うこととした。
    - ✧ NIST 標準化が決まっているが FIPS 文書が発行されていないため詳細が流動的なもの
    - ✧ NIST Additional Signatures 候補の中で、格子、符号、多変数、同種写像、ハッシュ関数のカテゴリに含まれるもの
    - ✧ MPC-in-the-Head など新たなカテゴリとして分類されているもの

- 記載すべき項目及び章立てと執筆担当者

	執筆担当者
i. はじめに	事務局（青野）
ii. PQC の活用方法	伊藤委員
iii. 格子に基づく暗号技術	下山委員、安田（雅）委員
iv. 符号に基づく暗号技術	成定委員
v. 多変数多項式に基づく暗号技術	安田（貴）委員
vi. 同種写像に基づく暗号技術	高島委員
vii. ハッシュ関数に基づく署名技術	廣瀬委員

- 調査活動と執筆活動の方針
  - PQC の研究成果が発表される主要な国際会議 Crypto、Eurocrypt、Asiacrypt、PQCrypto を中心に、開発・標準化の動向に関しても 2024 年 9 月 30 日までの情報を可能な限り調査す

る。その他主要な動向があれば可能な限り取り上げる。

- 2023 年度第 2 回 PQC WG での調査内容の報告
  各章の執筆担当者が 2023 年度第 2 回 PQC WG において、その時点までの調査内容を報告する。

3.3 第 2 回 WG（2024/1/19）での実施内容及び決定事項

- 各章の執筆担当者が 2023 年度第 2 回 PQC WG において、その時点までの調査内容を報告。各章の大まかな更新内容が確認された。
- 調査報告書及びガイドラインの執筆方針について以下の執筆方針が決定された。
  - 1 章についても他の章と同様に、調査報告書は専門的な内容、ガイドラインには調査報告書から技術的に複雑な内容を省略し抜粋した内容とする。
  - 米国で FIPS 化が決まっている方式に関して、2024 年 9 月 30 日までに正式版が出版された場合には FIPS 版と更新部分を、FIPS 化されない場合には出版時期によって対応が異なるが、Initial draft 版とその更新差分を記述する方針とする。
  - Additional Signatures の候補を各章に記載するかどうかは執筆担当者の判断とする。
  - MPC-in-the-Head、Additional Signatures 提案方式の中でガイドライン中の計算問題の分類に含まれないものについて、大きな動きは認知されていないことから新章とはしない。

4. 「素因数分解の困難性に関する計算量評価」、「楕円曲線上の離散対数計算の困難性に関する計算量評価」の予測図の更新

4.1 予測図における分解記録のプロットについて

第 1 回 WG におけるご意見に従い、分解記録のプロットについては、図の中の該当する参考文献のところに、「文献に基づいてプロットしています」と追記した。

4.2 2023 年度予測図の更新

- 「今後の予測図の取り扱い」に基づいて予測図の更新を行った(図 1、2)。素因数分解問題の困難性および楕円曲線上の離散対数問題の困難性に関する計算量評価に大幅な進展はなかったため、2023 年 6 月・11 月のベンチマーク結果を追加した。
- 図 1 と図 2 において TOP500.Org のアドレスに https と http の表記揺れがあるため、https に統一することで合意した。

＜今後の予測図の取り扱い＞

(1) いわゆるムーアの法則を仮定して外挿線を年度末からプラス20年後まで従来どおり直線で引き、評価に大きな変動がないと考えられる限りにおいては、安全サイドに倒した評価※として予測図を当面の間更新していく。

〈今後の公開鍵暗号のパラメータ選択〉

(2) 公開鍵暗号のパラメータ選択に関する対応方針については、安全性以外にも相互接続性など、運用上の観点もあるため、暗号技術評価委員会だけではなく、暗号技術検討会、暗号技術活用委員会や関係各所などを含めて検討する。

※各国・国際標準化機関等により示されている主要な暗号技術の安全性基準と比較すると、より現状に即した評価となっており、危殆化時期は他機関等が規定している暗号技術の利用期限よりも先に延びている。

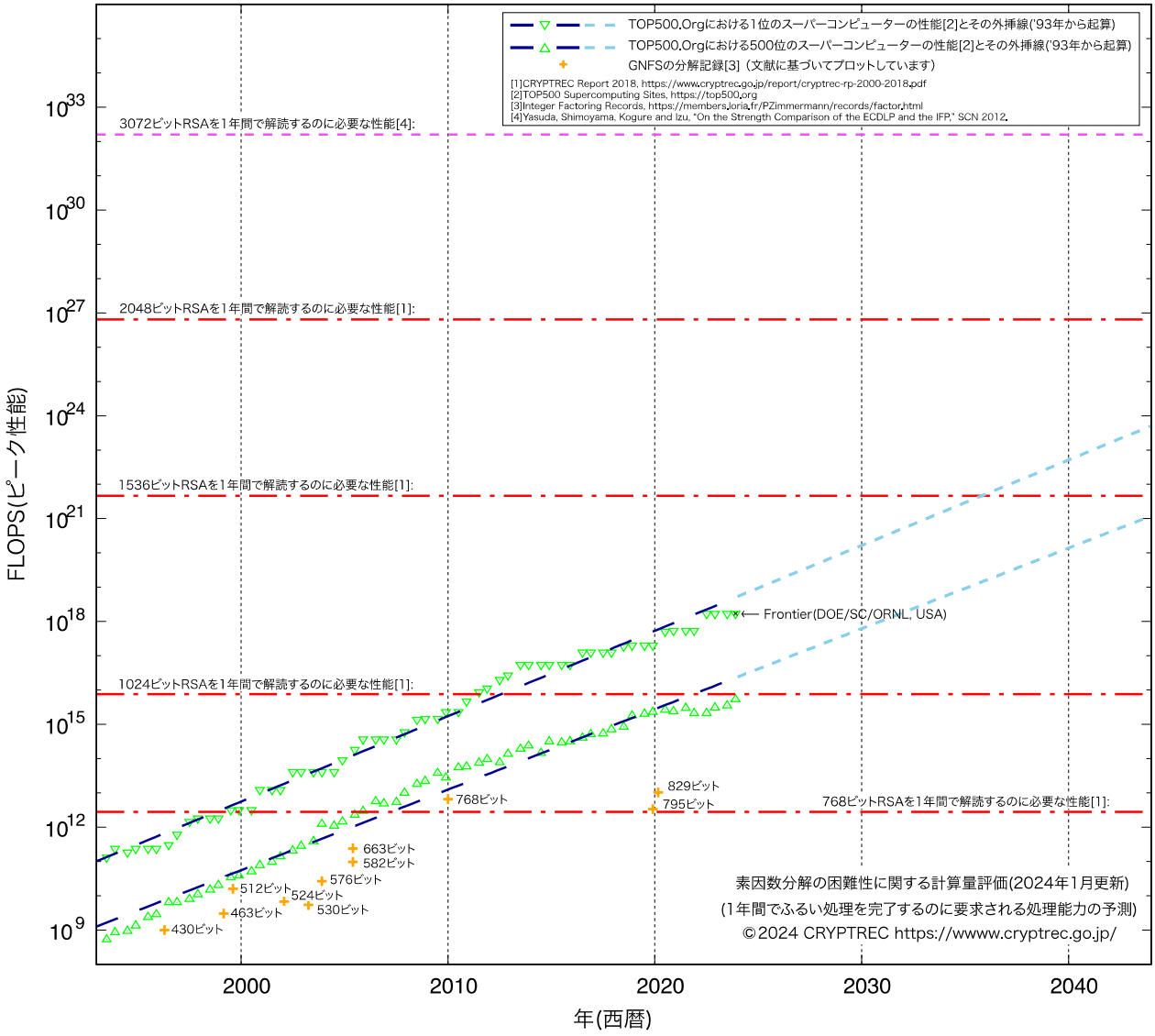


図 1：素因数分解の困難性に関する計算量評価（2024 年 1 月更新）[1]

[1] スーパーコンピューターの性能の伸びに関する外挿線は僅かであるが鈍化してきている。

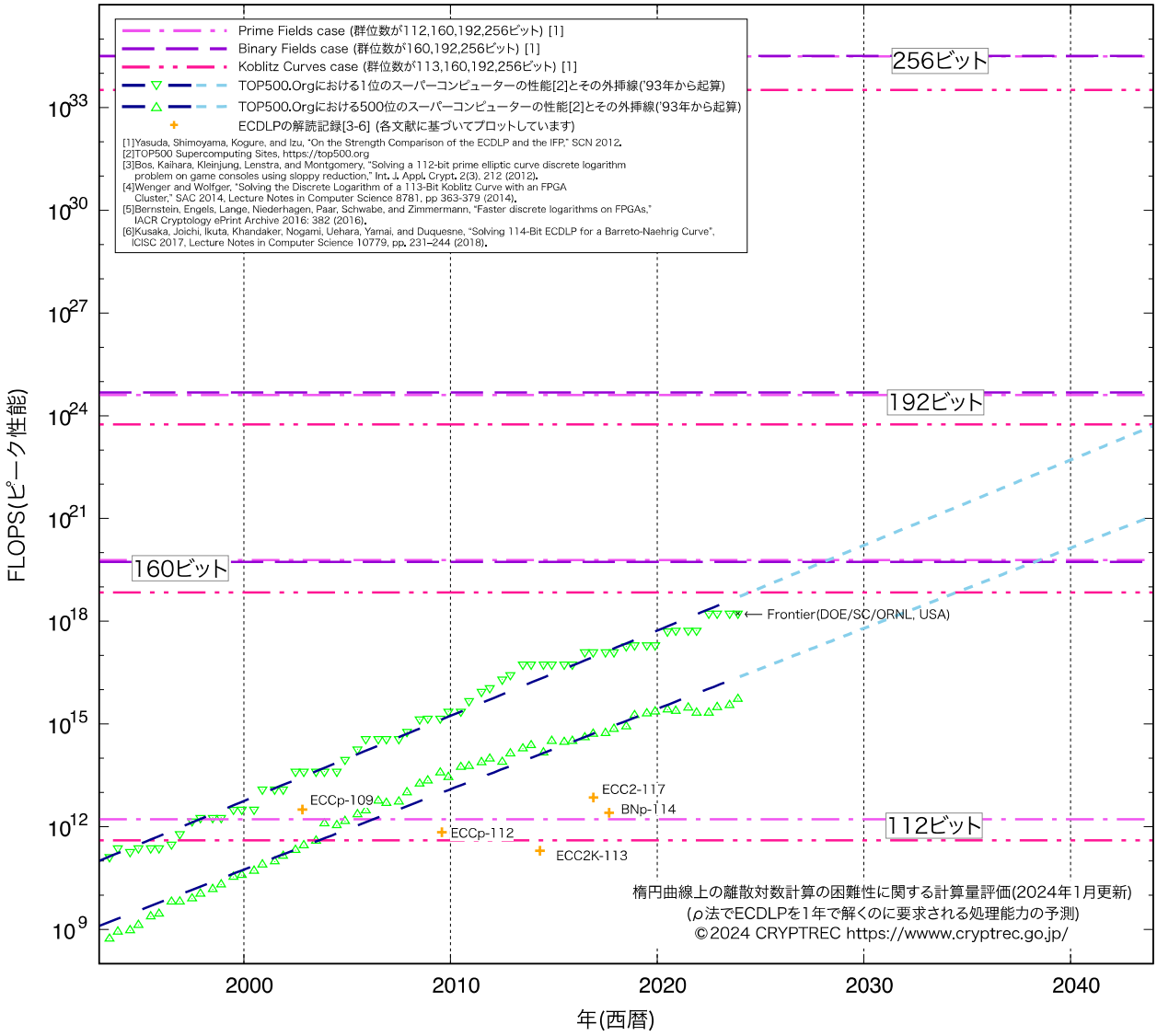


図 2：楕円曲線上の離散対数計算の困難性に関する計算量評価(2024 年 1 月更新)[2]

以上

[2] スーパーコンピューターの性能の伸びに関する外挿線は僅かであるが鈍化してきている。

軽量暗号 Ascon の実装性能に関する調査及び評価

電気通信大学 大学院情報理工学研究科

崎山 一男

2023 年 9 月

エグゼクティブサマリー

米国 National Institute of Standards and Technology (NIST) は，軽量暗号 (LWC: Lightweight Cryptography) コンペティション [34] で Ascon を選定した．本報告は，公開されている Ascon-128 の認証暗号モードにおける暗号化及び復号処理を行う実装研究を中心に論文を調査し，物理攻撃への耐性を持つハードウェア及びソフトウェア実装の性能評価結果をまとめ，考察を与えたものである．理論的安全性については，藤堂の文献 [48] を参照されたい．

Ascon は，認証暗号モードとハッシュモードに対応する軽量な暗号アルゴリズムであり，実装コストと処理パフォーマンスのトレードオフの点で高い柔軟性がある．つまり，ハードウェア実装でもソフトウェア実装でも，暗号機能を効率的に実現することができるため，様々なユースケースでの利用が期待される．高い柔軟性の理由は，以下のとおりである．

- 同じラウンド処理の繰り返しで実現できる
- ラウンド処理が並列実装にも対応できる
- 同じ 5 ビットの S-box が繰り返し使われている

5 ビットのコンパクトな S-box を同時に処理することで，実装コストの増加に見合う処理パフォーマンスを得ることができる．一方で，異なる時間に S-box を再利用して処理することで，処理パフォーマンスを犠牲にして実装コストを下げることができる．つまり，ハードウェア実装においては，インタフェースや求める性能に合わせてアーキテクチャを柔軟に変更することができ，ソフトウェア実装においては CPU のワードサイズに合わせたプログラミングが可能となる．また，複数のラウンド処理をまとめて計算することで，処理パフォーマンスのさらなる向上が実現できる．

Ascon が特に優れている理由のひとつは，ほとんど全ての処理を同じラウンド処理の繰り返して実現できる点にある．暗号化処理と復号処理の違いは，ラウンド処理における入出力データのインタフェース部分のみである．この極めて規則的な処理構造のおかげで，各モードの切り替えに対するオーバーヘッドは極めて小さくてすむ．AES 暗号にも，似たような実装上の性質はあるものの，暗号処理自体のデータパスが同じである Ascon は，実装における柔軟性がさらに高いと言える．

コンペティションで，AES 暗号が選定されたのは，1997 年 9 月であり，サイドチャネル攻撃の危険性を Kocher が最初に指摘したのが 1995 年 12 月である．そのため，AES 暗号に対してアルゴリズムレベルでの物理攻撃対策が十分に考慮できる状況ではなかった [24]．乱数によるマスキングや WDDL といった，サイドチャネル攻撃対策の研究が盛んになったのは 2000 年前後である [9, 43] ．つまり，Kocher によるサイドチャネル攻撃の論文や AES 暗号の選定により物理攻撃対策への研究者の意識が高まり，暗号アルゴリズムを新規に設計する場合においては，物理攻撃対策を含めた実装性は考慮すべきひとつの要素となった．Ascon はその最初の暗号アルゴ

リズムと言える．

その後，Nikova らによって Threshold Implementation (TI) が提案されたのが，2006 年である [32, 33]．現在 TI は，ハードウェア実装とソフトウエア実装の両方で多くの研究報告がなされており，現在も改良が進んでいる．Domain Oriented Masking (DOM) [20, 19] といった，TI よりもさらに効率的な実装を目指した対策技術が提案されるなど，暗号研究者内での理解は急速に進んだ．Ascon が最初に提案されたのは，2014 年の認証暗号のコンペティション CAESAR competition [4] である．Ascon を最初に提案したころは，最新の物理攻撃対策技術の成熟期にあった．実際に，Ascon-128 が物理攻撃対策との親和性が高い実装構造となっていることは興味深い事実である．

物理攻撃耐性を評価する手法にも大きな変化があった．サイドチャネル攻撃の発見直後は，鍵復元攻撃の成否や，少ない波形数での攻撃成功を目指すケーススタディが比較的多かった．非プロファイリング型の攻撃では選択関数やリーケージモデルの研究が，プロファイリング型の攻撃の場合ではテンプレートの作成方法に関する研究が研究の中心であった．これらは，攻撃者の能力に関するものである．適切な攻撃者が実装されていない場合には鍵が復元できないため，脆弱性は見つけることができない．さらに，実験における計算量の限界により攻撃を実装できない場合にも，脆弱性はないものとされてきた．つまり，攻撃が失敗したときには，暗号実装の安全性を判断することはできない．

現在では，Test Vector Leakage Assessment (TVLA) [18] による統計的に安全性を評価する手法が主流となっている．TVLA は，鍵が実際に導出できるかどうかを試すのではなく，サイドチャネルリークによる攻撃の可能性を判断するものである．未知の攻撃手法を含め，厳密に安全性評価が行えるようになった．現在の暗号アルゴリズムの実装研究では，TI といった乱数を用いたマスキング対策で物理攻撃を実装し，その安全性評価には TVLA を用いることが主流となっている．マスキング対策における実装上の問題は，冗長化した回路やプログラムのサイズによる実装コストと，マスキングに必要となる乱数コストである．実装コストを抑えるための研究成果は多く存在しているが，暗号アルゴリズム毎に最適な冗長化や乱数コストを狙う研究と，汎用的なコスト削減に向けた設計手法の確立を目指すものとに分かれる．

本報告では，最初に Ascon に適用する物理攻撃対策技術とその安全性評価手法に関する調査を行う．次に，実際のハードウェア実装及びソフトウェア実装における物理攻撃対策に関するケーススタディを 8 件を取り上げ，それらの内容をまとめた上で考察を与える．Ascon-128 の実装性能は非常に高く，特に物理攻撃対策については暗号研究者がこれまでに培った最新の技術を搭載しやすい構造である．一方で，今後 Ascon が，IoT デバイスとして様々なプラットフォームに実装されることを想定すると，対策を含めた暗号アルゴリズム実装について，その生産性の向上が重要となる．したがって，マスキング設計ツールやその安全性検証ツールを Ascon に適用した論文の調査を含め，今後の暗号実装研究の新たな方向性についても言及する．

目次

図目次

表目次

1 はじめに

モノのインターネット（IoT: Internet of Things）が社会実装され，実世界のデータをサイバー空間に取り込む IoT デバイスが新たな攻撃対象となっている．IoT デバイスは，センサ，通信モジュール，データ処理を行う CPU や専用ハードウェアで構成されている．IoT デバイスの多くは小型であり，バッテリー消費を抑えた省エネルギー実装やアンテナからの電力伝送で動作できるほどの低電力実装が求められる．軽量暗号アルゴリズムは，そのような低リソースの環境下においても安全に機能しなければならない．

米国 NIST (National Institute of Standards and Technology) は，社会のニーズを鑑みて NIST 軽量暗号 (LWC: Lightweight Cryptography) コンペティション [34] を実施し，2022 年 2 月に Ascon を選定した．Ascon はデータの秘匿性と認証性を担保する認証付き暗号（認証暗号）とハッシュ関数の機能をサポートすることができる軽量な暗号アルゴリズムである．

本報告では，今後世界中で広く使われていくであろう Ascon-128 の物理攻撃耐性を含めた実装性能に関する調査を行うものである．つまり，サイドチャネル攻撃や故障利用解析攻撃への対策技術の実装に関して，単なる実装コストや処理コストの議論ではなく，物理安全性に必要となるコストを調査する．社会基盤である IoT システムの安全性は，IoT デバイスの物理攻撃耐性で決まる．この報告では，物理攻撃研究が極めて活発な状況にある欧米中の研究機関，企業，及び大学から発表された Ascon の物理攻撃耐性に関する論文の中から，特に重要と思われるものを調査の対象とする．

2 本報告書の概要

2.1 調査対象

本報告において, 調査する対象の暗号アルゴリズムは, NIST 軽量暗号として選定された Ascon である．認証暗号モード時の暗号化及び復号処理の Ascon-128 に関する ASIC 実装，FPGA 実装及びソフトウェア実装に関して，物理安全性を中心とする性能評価を扱う文献を調査する．以下が主な情報源である．

- NIST 主催のワークショップ Lightweight Cryptography Workshop
  `https://csrc.nist.gov/Projects/lightweight-cryptography/workshops`
- IACR Transactions on Cryptographic Hardware and Embedded Systems (TCHES)
  `https://tches.iacr.org`
- IACR Cryptology ePrint Archive
  `https://eprint.iacr.org`

他にも，Springer 社の LNCS (Lecture Notes in Computer Science) [46] や米国電気電子学会 IEEE の会議プロシーディングスと学術論文誌 [50] を調査する．結果として，本報告には，NIST は，2023 年 6 月 21, 22 日に開催された第 6 回軽量暗号ワークショップ（バーチャル）で発表された研究成果が多く含まれている．その理由は，これまでの NIST 軽量暗号コンペティションの選考中においては，理論的安全性，実装コスト，及び処理パフォーマンスによる議論が中心であった一方で，第 6 回軽量暗号ワークショップでは，これまでの軽量暗号の標準化に向けた技術的課題の議論，つまり社会実装に向けた Ascon の物理攻撃耐性に関する発表が多く見られたためである．

本報告では CRYPTREC 外部評価報告書 [41] での実装性能調査と同様，Ascon の実装形態を以下のカテゴリに分けて調査を行う．

- ハードウェアアクセラレータ
  - コプロセッサ（FPGA 実装）
  - コプロセッサ（ASIC 実装）
- 命令拡張
- ソフトウェア実装

Ascon の命令拡張実装は，CHES 2023 に採択された論文 [8] で実装性能の報告がなされているが，物理攻撃対策がなされていないため，本報告の対象から外す．

2.2 Ascon の実装性能評価

Ascon-128 の物理攻撃対策のある実装（以下，セキュア実装と呼ぶ）における実装コスト，処理パフォーマンス，及び物理攻撃耐性について，公開されている論文の実験結果をまとめ，考察を与える．従来の暗号アルゴリズムの物理攻撃対策では，通常の設計フローで対応できない部分については，その都度，必要となる対策回路の設計や検証を人手で行うことが多かった．しかし，乱数を用いたマスキングによる物理攻撃対策は複雑であり，正しく対策技術を実装するためには設計と検証の自動化の必要がある．設計及び検証の自動化により，多少の実装コストや処理パフォーマンスのオーバヘッドが生じる可能性はあるが，人手による安全性上のバグを防ぐためにはこういったツールの活用は不可欠と言える．したがって，調査した論文に設計手法やツールが書かれているものについては，性能評価における重要な要素として付記する．

3 Ascon のセキュア実装

3.1 Ascon のアルゴリズム

Ascon の設計者（提出者）は，IAIK, Graz University of Technology に所属している次の 4 名の研究者である．

- Christoph Dobraunig
- Maria Eichlseder
- Florian Mendel
- Martin Schläffer

本方式に関する概要と仕様は以下の URL から参照できる．

- Web サイト：`https://ascon.iaik.tugraz.at`
- 仕様:`https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/finalist-round/updated-spec-doc/ascon-spec-final.pdf`

Ascon の認証暗号モードにおける処理は，大きく次の 3 つのステップから成る．以下で述べるワードは，64 ビットである．また，入力されるデータ長により，ステップ (2) の処理時間は異なる．ただし，Initialization と Finalization の処理時間は，入力されるデータに関係なく一定である．

ステップ (1): **Initialization**: ステートを鍵 K, ノンス N, 及び初期値 IV を用いて初期化する．

ステップ (2): **Iteration（データ処理部）**: Associated Data (AD) を分割し，A_i を入力とする処理を行う．その後，暗号化においては平文を決められたサイズに分割した P_i の処理を行い暗号文 C を出力する．復号においては分割された暗号分 C_i を入力とする処理を行い，平文 P を出力する．

ステップ (3): **Finalization**: 再び鍵を入力とする処理を行い，タグ T を出力する．

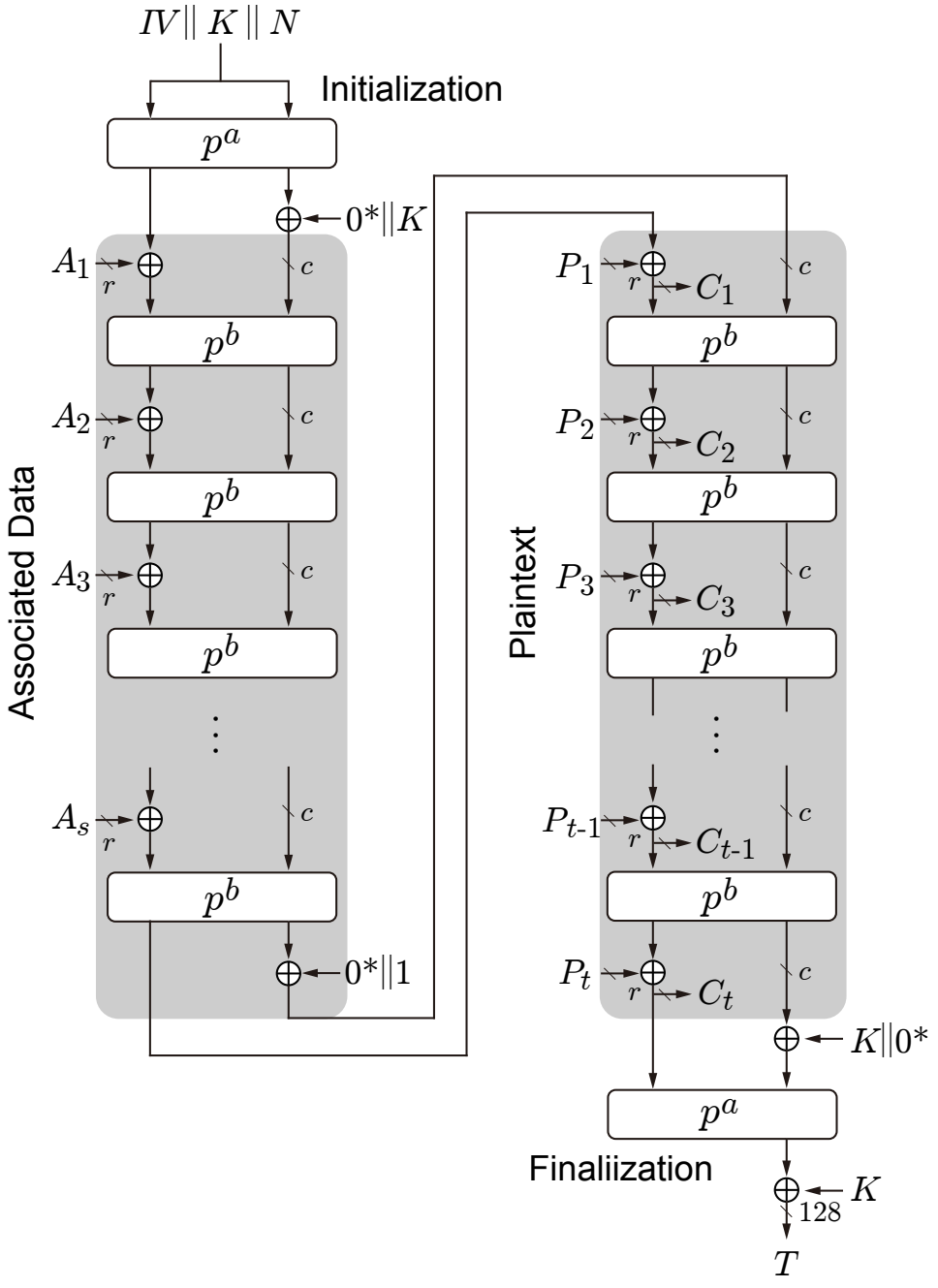


図 1: 認証暗号モードにおける Ascon の暗号化プロセス

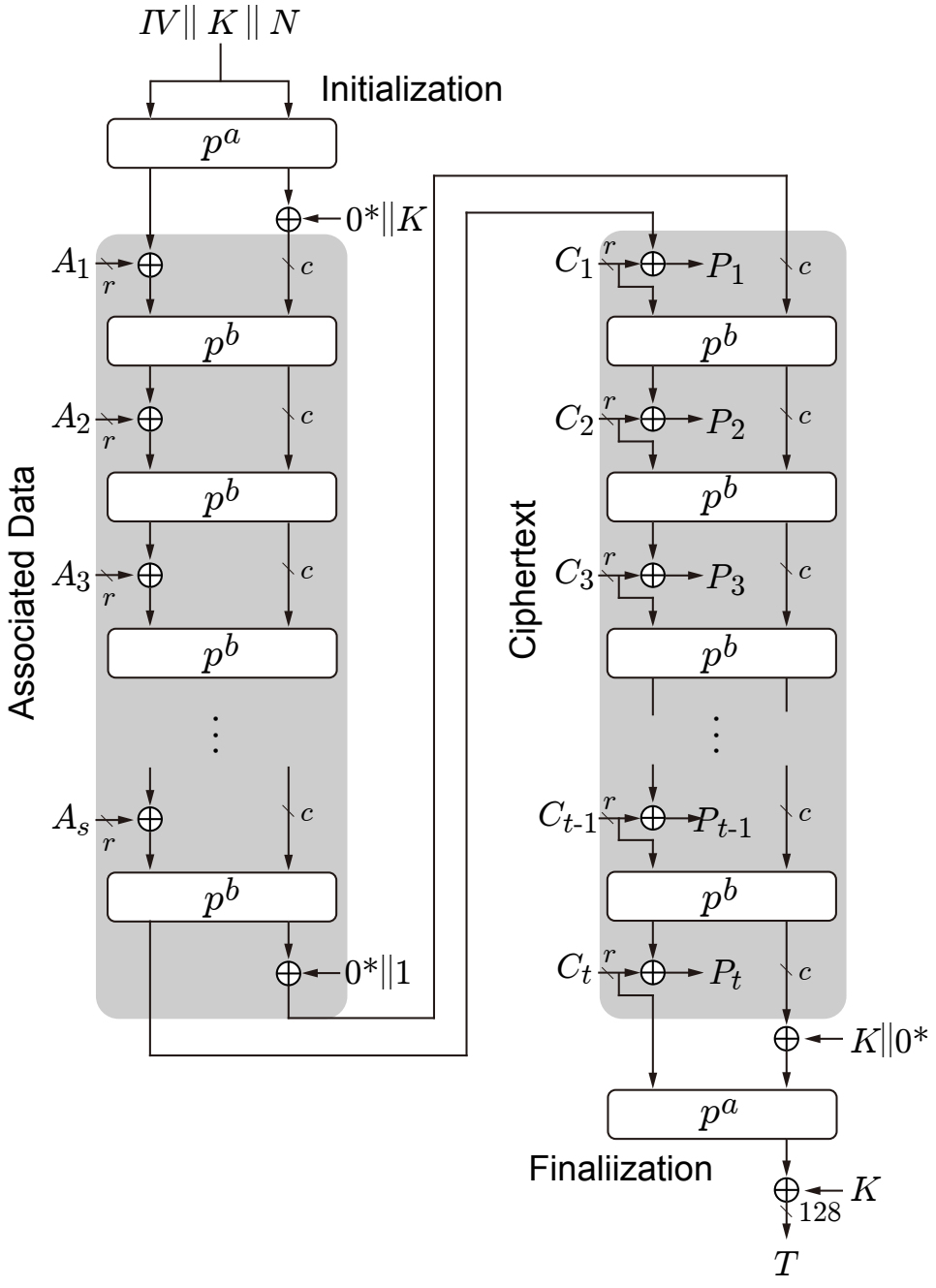


図 2: 認証暗号モードにおける Ascon の復号プロセス

3.2 Ascon の軽量 Permutation

Ascon-128 は，SPN（Substitution Permutation Network）型の Permutation p をラウンド関数として繰り返し使用している．先に，ASCON の認証暗号モードの処理の概要を述べたが，Initialization と Finalization の処理では，p を $a = 12$ 回繰り返す p^a の処理が行われ，AD 及び暗号化／復号処理においては p を $b = 8$ 回繰り返す p^b の処理が行われる．認証暗号モード時における Ascon の暗号化と復号の処理に対するブロック図を，それぞれ図 1 と図 2 に示す．

Ascon の Permutation p では， 5 個のワード（64 ビット長）の x_0, x_1, x_2, x_3, x_4 で構成される 320 ビットのステートに対して以下の処理を行う．

ステップ (1): **ラウンド定数加算** p_C: x_2 に対してラウンドごとに異なる 1 バイトの定数を XOR する．

ステップ (2): **非線形層** p_S: 5 個のワード x_0, x_1, x_2, x_3, x_4 に対してビットスライスを適用し，5 ビットの $x_{0,i}||x_{1,i}||x_{2,i}||x_{3,i}||x_{4,i}$ を入力とする Ascon S-box を 64 回適用する $(0 \leq i < 64)$.

ステップ (3): **線形拡散層** p_L : 5 個のワード x_0, x_1, x_2, x_3, x_4 ごとに，右ローテションシフトと XOR 処理からなる線形処理でステートを攪拌する．

つまり，$p = p_L \circ p_S \circ p_C$ である．

設計者らは，非線形層 p_S のリードマラー標準形（ANF: Algebraic Normal Form）を示している [13]. 以下の通り 2 次である．

$$
\begin{aligned}
y_{0,i} &= x_{4,i}x_{1,i} \oplus x_{3,i} \oplus x_{2,i}x_{1,i} \oplus x_{2,i} \oplus x_{1,i}x_{0,i} \oplus x_{1,i} \oplus x_{0,i},\\
y_{1,i} &= x_{4,i} \oplus x_{3,i}x_{2,i} \oplus x_{3,i}x_{1,i} \oplus x_{3,i} \oplus x_{2,i}x_{1,i} \oplus x_{2,i} \oplus x_{1,i} \oplus x_{0,i},\\
y_{2,i} &= x_{4,i}x_{3,i} \oplus x_{4,i} \oplus x_{2,i} \oplus x_{1,i} \oplus 1,\\
y_{3,i} &= x_{4,i}x_{0,i} \oplus x_{4,i} \oplus x_{3,i}x_{0,i} \oplus x_{3,i} \oplus x_{2,i} \oplus x_{1,i} \oplus x_{0,i},\\
y_{4,i} &= x_{4,i}x_{1,i} \oplus x_{4,i} \oplus x_{3,i} \oplus x_{1,i}x_{0,i} \oplus x_{1,i}.
\end{aligned}
$$

Ascon-128 の S-box は 5 ビットであり，AES S-box のような簡単な代数学的表現はない．テーブル参照でハードウェア実装する場合には，表 3.1 に示す真理値表を用いて実装する．

単純に 64 個の S-box を並列に実装する場合には，5 ビットの真理値表に対応するメモリが 64 個分の容量，つまり $10,240$ ビットのメモリ領域が必要となる．

Ascon S-box のハードウェア実装では，図 3 に示すような組み合わせ回路での実装が効率的である [13]．特に，Ascon では同じ処理が繰り返し実行されるため，ループアーキテクチャにおける組み合わせ回路に対して，いわゆるループアンローリングを適用することで，処理パフォーマンスの向上が得られる場合がある．つまり，高スループットを得るために本来 1 サイクルで実

表 3.1: Ascon の S-box

x	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
$S(x)$	04	0b	1f	14	1a	15	09	02	1b	05	08	12	1d	03	06	1c
x	10	11	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
$S(x)$	1e	13	07	0e	00	0d	11	18	10	0c	01	19	16	0a	0f	17

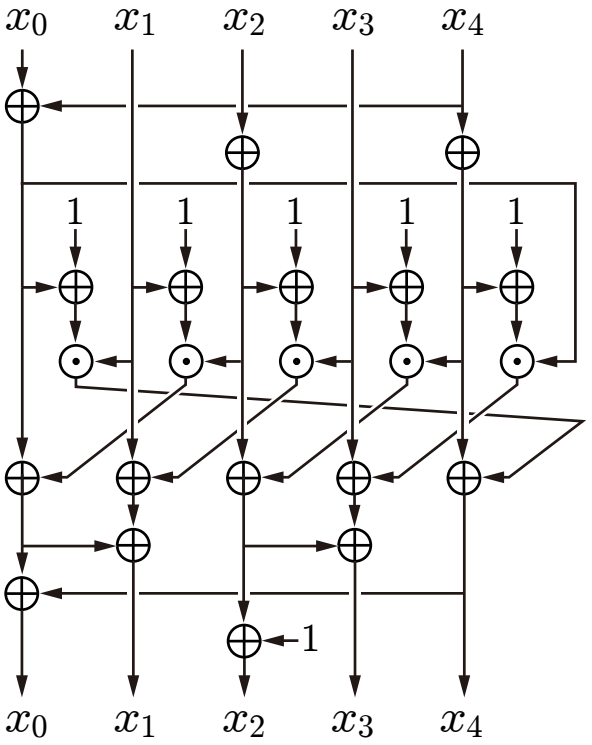


図 3: Ascon の S-box

行する組み合わせ回路のラウンド処理を，アンローリングにより複数ラウンドまとめて最適化することで，クリティカルパス遅延の短縮と同時にレイテンシ（クロックサイクル）の削減が狙える．このように，実装するシステムの様々な仕様に合わせて，面積コストと処理パフォーマンスのトレードオフを柔軟に変更することができる．これは，Ascon-128 の Permutation p のビルディングブロックである 5 ビット S-box の演算処理単位の小ささや，並列処理を可能とするデータ構造から説明できる．

一方，ソフトウェア実装では，ハードウェア実装とは異なり，プラットフォームに適した実装が求められる．特に，メモリ容量が少なく CPU の処理性能が低い IoT デバイスへの実装において十分なサイドチャネル対策を実現するためには，乱数生成に必要なコストと安全性のトレー

ドオフを慎重に模索する必要がある．CPU のワードサイズに合わせて x_0, x_1, x_2, x_3, x_4 に対する非線形演算を，いかに安全に効率よく実装するかが重要となる．

3.3 Ascon に対するサイドチャネル攻撃対策

3.3.1 TI: Threshold Implementation

TI は秘密分散法に基づく乱数マスキングである．2006 年に Nikova らによって提案された [32, 33]．TI では、計算対象の値 x はシェアと呼ばれる複数の値で表現される．ここで，$\mathrm{GF}(2^m)$ 上のある関数 $z = \mathrm{N}(x, y)$ を考える．$\mathrm{GF}(2^m)$ 上の非線形変換 $z = \mathrm{N}(x, y)$ に対しても，シェアの考え方を適用できる．

例えば，3つの関数 $\{\mathrm{f}_1, \mathrm{f}_2, \mathrm{f}_3\}$ が，以下に示す Correctness（正確性），Non-Completeness（不完全性），及び Uniformity（均一性）の性質を有する場合，$\mathrm{N}(x, y)$ はシェアに分けて，2次プロービングモデル[*1]に対して耐性のある計算処理が実現できる．2つ以下のシェアからは元の値を復元することはできないからである．

Correctness

例の場合，関数 $\{\mathrm{f}_1, \mathrm{f}_2, \mathrm{f}_3\}$ は，以下の関係が満たされる場合に Correctness を満たす．

$$
\begin{aligned}
z &= z_1 \oplus z_2 \oplus z_3 \\
&= \mathrm{f}_1(x_2, x_3, y_2, y_3) \oplus \mathrm{f}_2(x_3, x_1, y_3, y_1) \oplus \mathrm{f}_3(x_1, x_2, y_1, y_2) \\
&= \mathrm{N}(x, y).
\end{aligned}
$$

Non-completeness

それぞれの関数が x, y の少なくとも1つのシェア値に依存しないように，例えば次のように計算する．

$$
\begin{aligned}
z_1 &= \mathrm{f}_1(x_2, x_3, y_2, y_3), \\
z_2 &= \mathrm{f}_2(x_3, x_1, y_3, y_1), \\
z_3 &= \mathrm{f}_3(x_1, x_2, y_1, y_2).
\end{aligned}
$$

こうすることで，2つ以下のシェアに分けた関数の処理からは，元の値 x に関する情報を知ることはできない．

Uniformity

例えば x のシェアの発生確率が等しくない場合，攻撃者がその偏りを利用することで，全てのシェアが揃わなくても元の x を復元させることができる．したがって，全ての x のシェアにお

[*1] プロービングモデルは，暗号処理を行うハードウェアやソフトウェアに対して，攻撃者が本来観測することができない内部信号を1本あるいは複数のプローブ（針）を用いて観測可能とする攻撃者モデル [21]．d 次プロービングモデルの場合には，攻撃者は異なる d 本のプローブを用いて d 個の中間値を観測できると仮定する．ただし，同じ回路を使い回すシェア型のハードウェアアーキテクチャの場合，同じプローブで異なる時間の複数の中間値を取得することも想定できる [47]．

いて，そのとりうる値の発生確率は等しくなければならない．

例として $m = 1$ の場合，つまり GF(2) の乗算では，

$$\Pr[x_1, x_2, x_3] = 1/8,$$

を満たさなければならない．

なお，次数 t の関数が，d 次プロービングモデルに対してサイドチャネル攻撃耐性を持つためのシェア数は，最小で $td + 1$ である．

3.3.2 DOM: Domain Oriented Masking

TI の他にも d 次プロービングモデルに耐性を持つマスキング手法がある．シェア数をある規則にしたがって削減する手法として，DOM がよく知られている [19]．TI が，関数レベルで 3 つのプロパティ (Correctness, Non-completeness, Uniformity) の性質を考慮するマスキング方式であるのに対して，DOM では，非線形演算により増加するシェア数を抑制するために，ドメインと呼ばれる概念を導入し，ドメインごとにシェアを再構成する．

例えば DOM では，変数 x のシェア x_0, x_1 を、それぞれドメイン 0 と 1 に関連付ける，そして， d 次プロービングモデルへの耐性を実現するために，変数ごとに $d + 1$ 個のシェアを使用する．つまり，ドメインの数は，$d + 1$ 個である．ここで，1 次プロービングモデルに対して耐性のある DOM GF(2^m) 乗算器を考える．この乗算器を 1 次セキュアな DOM 乗算器と呼ぶ．入力値 x, y は，それぞれシェアで表現され，TI と同様，以下の計算を処理する．

$$\begin{aligned} xy &= (x_0 \oplus x_1)(y_0 \oplus y_1) \\ &= x_0y_0 \oplus x_0y_1 \oplus x_1y_0 \oplus x_1y_1. \end{aligned}$$

ドメイン 0 において，入力 x_0 と y_0 を入力とする AND 演算，つまり x_0y_0 の処理は安全である．なぜなら，どの中間値をプロービング（サイドチャネル情報）により読み出したとしても，x, y が復元できないからである．同様に，x_1y_1 の処理もドメイン 1 で安全に処理される．さらに，x_0y_1 と x_1y_0 の計算においても，x, y から独立しているものであるため，それぞれの処理だけでは x, y に関するサイドチャネルからのリーケージは観測できない．しかし，x_0y_1 をドメイン 0 に取り込み，$x_0y_0 \oplus x_0y_1$ の計算をした場合には問題が生じる可能性がある．異なるドメインのシェア y_0, y_1 が，直接的ではないが XOR 計算で発生するためである．これにより y が即座に復元できるわけではないが，サイドチャネルリークの危険性があると考えるべきである．そこで，x_0y_1 及び x_1y_0 は，いずれもクロスドメインで計算しなければならない処理とみなし，特定のドメイン 0 や 1 における計算とは切り離して考える．ここまでが，DOM における Calculation ステップである．

次に，クロスドメインの計算結果を特定のドメインに取り込むために，Resharing（再シェア）

とよばれるステップを実行する．具体的には，x_0y_1 と x_1y_0 の計算の後に，フレッシュな乱数でマスキングを行う．この Resharing においては，同じ乱数 r は使っても良いとしている．つまり，$x_0y_1 \oplus r$ と $x_1y_0 \oplus r$ のように処理できる．また，クロスドメインに関する一連の処理に起因して生じるグリッチの伝搬については，Rehareing の結果を FF (Flip-Flop) に格納することで情報漏洩を抑止する．パイプライン処理では，ドメイン 0 や 1 における計算のタイミングを揃える必要があるため，x_0y_0 と y_0y_1 の計算結果に対しても，FF (Flip-Flop) に格納する．

最後に，Integration ステップでは，以下のように特定ドメインとクロスドメインの 2 項の統合，つまり XOR 演算を行う．

$$q_0 = (x_0y_0) \oplus (x_0y_1 \oplus r),$$
$$q_1 = (x_1y_0 \oplus r) \oplus (x_1y_1).$$

より高次のセキュア DOM GF(2^m) 乗算器 も同様に設計することができる．また，上述の 3 ステップは，S-box などの非線形演算にも適用できる．

このように DOM は，ドメイン単位の管理によって，シェア数を適切に管理することができ，Resharing における乱数を工夫することで，対策実装コストの低減が期待できる．TI による回路サイズの削減には，数学的な処理の変換が必要となることが多い．一方，DOM による設計手法は，任意の回路に対して単純なステップの処理を繰り返せばよいため，設計の自動化がしやすいマスキング手法であると言える．つまり，DOM によるマスキング実装の設計生産性は高い．なお，DOM により生成した回路のコストは，最適化されていない TI 実装よりも低く，最適化された TI に匹敵するという結果も得られている．TI との実際の実装における安全性については，Ascon だけでなく，様々な暗号アルゴリズムに対して今後比較する必要がある．

3.4 Ascon に対するサイドチャネルからの漏洩評価

3.4.1 CPA: Correlation Power Analysis

電力のサイドチャネル情報を効率よく解析する方法として，最もよく知られているのが 相関電力解析 （CPA: Correlation Power Analysis）である [6]．電磁波サイドチャネルに対しては，CEMA（Correlation ElectroMagnetic Analysis）と呼ばれる．Ascon の実装に対しても，CPA による情報漏洩評価は重要である．

DPA（Differential Power Analysis) [25] では，特定の 1 ビットに対する電力モデルが採用される．一方で，CPA では複数ビットの電力消費をモデル化するため，測定ノイズや処理アルゴリズムに起因するノイズの影響を軽減することが期待できる．DPA では，鍵などの秘密情報の予測にもとづいて電力波形データを 2 つのグループに分け，2 つのデータの平均の差を調べるが，CPA は波形データをより多くのグループに分け，電力モデルとの相関を調べる．Ascon に限らず，多くの暗号アルゴリズムでは，予想した鍵によってレジスタに格納される複数ビットの中間値を選択関数により導出できる場合には，CPA が最適である．

本報告のケーススタディ（4 章）でも，情報漏洩評価として，CPA を用いた解析が採用されている．Ascon に対する CPA 評価は，AEAD 暗号化あるいは復号処理において，攻撃者が秘密鍵の復元ができるかどうかで判断する．選択関数（Selection Function）は，Initialization あるいは Finalization 処理から選ばれることが多い．これは，アルゴリズムの処理に鍵が直接関与しており，鍵予測によって中間値の導出が可能なためである．この選択関数の選び方については，文献 [42] を参照されたい．

3.4.2 TA: Template Attack

DPA とは対照的に，事前のプロファイリングが必要なテンプレート攻撃（TA: Template Attack）も Ascon の攻撃耐性評価では重要である．本報告でも，Ascon のソフトウェア実装への TA の報告をまとめている（4.8 章 [45].）

TA の前提として，暗号アルゴリズムを処理するデバイスが，攻撃者の完全な制御下になければならない．なぜならば，攻撃者が自由に平文や鍵情報をデバイスに設定し，デバイスから漏洩したサイドチャネル情報の確率分布から，デバイスの物理特性をプロファイリングするためである．つまり，TA は簡略化した電力モデルの代わりに，実際のデバイスの複雑な物理的な振る舞いに関するモデルを利用する．攻撃者が無制限にデバイスを実行することができれば，測定ノイズを十分削減することができるため，最も強力な攻撃手法となりうる．なお，プロファイリング型の攻撃が発展したものとして，教師あり機械学習を用いた攻撃やディープラーニング（DL: Deep Learning）などが登場している．Ascon がファイナリストとして選定される前の 2020 年の文献であるが，対策なしの Ascon に対して 24K 個の波形トレース[*2]で DL 攻撃に成功したと

[*2] オシロスコープ等で取得した物理情報の時系列変化の軌跡を波形トレースあるいは単にトレースと呼ぶ．波形ト

している [37].

3.4.3 TVLA: Test Vector Leakage Assessment (Welch's t-test)

ウエルチの t 検定は，さまざまな分野において，広く仮説検定に使用される統計手法である．サイドチャネルリークの評価における t 検定は，TVLA と名付けられている．その目的は，鍵復元や秘密情報の取得ではなく，暗号処理デバイスの内部データとサイドチャネル情報の依存性を評価し，潜在的な脆弱性を特定することにある．攻撃者の計算能力や攻撃手法に関係なく，暗号実装の安全性に関する汎用的な評価指標が提供できるツールとして，広く用いられるようになった．

具体的には，ある 2 つの基準に従って暗号アルゴリズムを実行し，その際に測定したサイドチャネル情報の波形データを，それぞれ集合 A と B に分ける．これらのデータセットの各サンプル点に対して，以下の式で t 値を算出する．

$$\mathrm{t} = \frac{\mu_A - \mu_B}{\sqrt{\frac{\sigma_A^2}{n_A} + \frac{\sigma_B^2}{n_B}}}. \tag{1}$$

ここで，μ、σ^2、n は，サンプル点における波形データ値の平均，分散，及び標本数であり，集合 A と B に対してそれぞれ求める．

サイドチャネルリークの評価で一般的に使われている基準は，鍵を固定にし，波形データの集合の 1 つを固定平文とし，もう 1 つをランダム平文とするものである [18]．$|\mathrm{t}| < 4.5\sigma$ を満たしていれば，そのサンプル点ではサイドチャネルリークがないものと判断する．

平易に表現すると，固定した平文の値に依存したサイドチャネールリークがあるかを調べるものである．もし，平文をランダムに入力した場合のサイドチャネル情報と比較して，なんらかの差異が見られるようであれば，攻撃者はその情報を使って内部の秘密情報を取得できる可能性があると判断する．したがって，t 検定において漏洩の可能性が示されたとしても，具体的な攻撃が実装できるかは不明であるが，未知の攻撃を含めて安全性評価をより厳格に行うことができると言える．

レースの単位として用いることもある．

4 Ascon の物理安全性と実装性に関するケーススタディ

4.1 Niels と Daemen による報告（2017.05） [42]

4.1.1 著者, 所属機関

- Niels Samwel, DiS Group, Radboud University
- Joan Daemen, DiS Group, Radboud University and ST Microelectronics

4.1.2 概要

この論文は，Ascon が，軽量暗号コンペティションの前に実施された CAESAR コンペティション [4] の候補であった時のものである．実際に，Ascon のハードウェア実装に対して，サイドチャネル攻撃 (CPA) を行った最初の論文として重要である．論文では，２つの CAESAR 候補である Keyak と Ascon の FPGA 実装におけるサイドチャネル攻撃が説明されているが，本報告書では，Ascon に絞り論文の要点をまとめる．Ascon は，繰り返し構造を持つスポンジ型の暗号アルゴリズムであり，非線形処理である S-box の出力に対して効率の良い選択関数を提案している．提案した選択関数により，FPGA 上に実装した対策のない Ascon-128 に対して， 50K トレースを用いた CPA 攻撃に成功している．さらに，３シェアの TI 実装 をした Ascon に対しては，シミュレーションベースで CPA 攻撃に成功したとしている．

4.1.3 CPA における選択関数

著者らは，ハードウェア実装におけるステート遷移 $S0_i(M, K^*), S1_i(M, K^*)$ に関する 64 ビットのレジスタレジスタ x'_0, x'_1 に着目し，以下に示す選択関数を提案している．

$$S0_i(M, K^*) = k_0^*(m'_i + 1) + m_i + k_1^*(m'_{i+45} + 1) + m_{i+45} + k_2^*(m'_{i+36}) + m_{i+36},$$

$$S1_i(M, K^*) = m_i(k_0^* + 1) + m'_i + m_{i+3}(k_1^* + 1) + m'_{i+3} + m_{i+25}(k_2^* + 1) + m'_{i+25}.$$

ここで，M は 128 ビットのノンスである．m, m' は，64 ビットのレジスタ，k_i^* は攻撃者が予測する鍵ビットである．一つ目の選択関数により，３ビットの鍵 k_0^*, k_1^*, k_2^* を予測することで，１ビットのステート値が導出できることが分かる．この値に対して，ハミング距離（HD: Hamming Distance）モデルを適用し，64 ビットの x'_0 レジスタからのリークとの相関を調べることで，鍵が復元できるとしている．しかし，一つ目の選択関数だけでは，攻撃成功により導出できる鍵は 高々 64 ビットであるため，二つ目の選択関数を用いて x'_1 のレジスタを攻撃し，全ての鍵が導出可能であるとしている．

4.1.4 攻撃の結果

Samwel と Daemen らは，SAKURA-G [49] に実装した Ascon に対して， 50K トレースで全ての鍵ビットの導出に成功したとしている．これにより，選択関数による電力モデルが効果的であることが分かるった．また，３シェア TI の Ascon に対しては．シミュレーションにより同様の CPA 攻撃を行い，900K トレースで全ての鍵を復元することに成功したとしている．TI 実装の詳細が不明ではあるが，この結果については漏洩の原因は明らかにはされていない．

4.1.5 まとめ

Ascon のハードウェア実装における Initialization へのノンスを用いた CPA 攻撃論文である．著者らは，Ascon のアルゴリズムとハードウェアレジスタのデータ遷移を良く考察した選択関数を導出している．このことは，Ascon のアルゴリズム処理における中間値データの格納の仕方は，バイト単位で処理を行う AES 暗号 [12] などとは異なり，アーキテクチャに強く依存することを意味している．つまり，AES 暗号では S-box の入力値のデータ遷移をモデルとすれば，アーキテクチャをあまり意識することなく攻撃ができたが，Ascon の場合にはそういった汎用的なモデルの構築に難しさがあるとも言える．アーキテクチャに適した選択関数を作成する必要があると思われる．なお，セキュア実装において，著者らはシミュレーションベースの CPA 攻撃に成功している．また，SAKURA-G 上に搭載された FPGA Spartan-6 を用いて実験が行われているが，実装コストや処理パフォーマンスについては説明がない．

本論文では考察されていないが，同様の手法で，Finalization に対しても CPA 攻撃は可能であると思われる．ただし，Initialization への攻撃ほどシンプルではなく，上述のとおりアーキテクチャにも依存するものと思われる．今後の課題として，Ascon に対する，より汎用的なサイドチャネル攻撃モデルの構築があげられる．さらに，Finalization への故障注入が効果的であると思われるため，その攻撃効率や対策技術のコストについても研究が必要であろう．

4.2 Groß の 学位論文（2018.06） [19]

4.2.1 著者, 所属機関

- Hannes Groß, IAIK, Graz University of Technology

4.2.2 概要

本学位論文は，Groß が著者として関与した複数の国際会議論文やジャーナル論文の内容が含まれており，DOM を体系的に理解できるものである．Groß は，TI よりも実装効率の良い暗号アルゴリズムのマスキング対策手法の確立を目指し， 3.3.2 章で紹介した Domain-Oriented Masking (DOM) を提案している．DOM には，Unified Masking (UMA) and Low-Latency Masking (LOLA) と呼ばれる 2 つのバリアントがある．UMA では，暗号アルゴリズムのデータパスにレジスタを追加することで，安全性上クリティカルとされるデータを適切に制御し，DOM の乱数コストを削減している．レジスタを追加することから，1 ラウンドの処理に必要となるサイクル数は増加するため，レイテンシは増加しスループットは低下する．しかし，必要となるフレッシュな乱数は少なくて済む．UMA とは対照的に，LOLA ではレジスタによるステージ数を減らし，処理パフォーマンスの向上を狙うものである．代わりに，非線形処理におけるシェア数が増加するため，より多くのデータの冗長性や追加の回路が必要になり，乱数コストも増加すると報告している．

4.2.3 実装結果

DOM による Ascon のセキュア実装結果を表 4.1 にまとめる．UMC-90nm Low-K の CMOS ライブラリで合成した結果である．UMA については，レイテンシとスループットをを犠牲にして，乱数コストが抑えられることが分かる．ただし，現実的な実装として捉えられる 1 次プロービングモデルに耐性のある 1 次セキュア UMA は，1 次セキュア DOM と比べて，乱数コストと回路の面積コストはほぼ同じである．シェア数が少ない場合には，UMA の実装コスト低下は限定的であると言える．5 次セキュア UMA では，5 次セキュア DOM と比べて，必要となる

表 4.1: Groß による対策付き Ascon AEAD 処理の ASIC 実装の報告（UMC-90nm Low-K）

デザイン	Area [KGE]	Cycle/Round	Throughput [Gbps]	Randomness [bit/cycles]
1 次セキュア DOM	28.89	3	2.25	320
1 次セキュア UMA	27.18	3	2.25	320
1 次セキュア LOLA	42.75	1	2.77	2,048
5 次セキュア DOM	161.87	3	1.86	4,800
5 次セキュア UMA	220.01	7	0.85	3,520
5 次セキュア LOLA	339.82	1	2.99	18,432

フレッシュ乱数を少なくすることに成功しているが，レジスタの追加により実装コスト自体は増加している．

一方，LOLA の実装については，１サイクルで１ラウンドの処理が行えるため，低レイテンシが実現できていることが分かる．面積コストは，DOM や UMA と比べて大きくなり，必要となる乱数が多い５次セキュア LOLA では，１サイクルあたり約 18K ビットと非常に多くのフレッシュな乱数を必要としている．

4.2.4 まとめ

通常の TI よりも少ないシェア数が実現できる DOM は，設計手法としても興味深い．DOM とそのバリアントにより，実装コスト，処理パフォーマンス，及び必要となる乱数におけるトレードオフは大幅に増えている．さらに設計者の選択肢が増えたことに加えて，設計手法自体が規則的であり汎用的なマスキングツールにできることは，生産性の向上につながると思われる．TI の実装においては，暗号アルゴリズムの数学的特徴をうまく利用して，シェア数を少なくする工夫が考えられている．Ascon のマスキング実装との比較については，今後の研究で模索されるべきであり，特に人手による最適化とツールによる最適化との比較を，生産性の観点から見ていく必要があると考える．

4.3 Batina らによる報告（2022.08） [5]

4.3.1 著者, 所属機関

- Lejla Batina, CESCA Radboud University
- Ileana Buhan, CESCA Radboud University
- Łukasz Chmielewski, CESCA Radboud University
- Ellen Gunnarsdóttir, CESCA Radboud University
- Vahid Jahandideh, CESCA Radboud University
- Tom Stock, CESCA Radboud University
- Léo Weissbart, CESCA Radboud University

4.3.2 概要

本論文は，NIST LWC のファイナリストのいくつかに対して，サイドチャネル解析の初期段階の結果をまとめたものである．2022 年 8 月 19 日に公開された Radboud 大の CESCA Lab の研究成果である．評価の対象とする暗号アルゴリズムは，Ascon, Xoodyak, 及び ISAP である．Ascon については，サイドチャネル対策のない実装とマスキング対策によるセキュア実装に対して TVLA と CPA を用いて物理攻撃に対する安全性を評価している．実装形態は，Arm-v6 上のソフトウェア実装であり，サイドチャネル情報として電力波形を用いている．以下，Ascon の安全性評価に絞って報告書の内容を紹介し考察を与える．

4.3.3 攻撃対象及び評価環境

評価に用いた Ascon は，Primary Recommendation である Ascon-128 である．ソフトウェアは，Ascon の開発チームが公開している C コードを用いている．電力測定には，Riscure 社の Piñata development board [39] を用いている．当該ボードには，32 ビットの Arm マイクロコントローラをベースとする SoC STM32F407IGT6 が搭載されている．動作周波数は 168 MHz である．電力波形は，Riscure 社のカレントプローブ（型番不明）と Picoscope 社のオシロスコープ（model 3206D） [40] を用いて取得している．著者らが行った CPA 攻撃では，4.1 章で紹介した Niels と Daemen により提案された選択関数を採用している．

4.3.4 評価結果

電力波形 50K トレースを用いて，対策なしの Ascon 全体の処理に対して TVLA を行った結果，Initialization で t 値をがしきい値を大きく超えていることが示されている．また，Initialization に特化して，100K トレースでの TVLA を実施し，CPA 攻撃に最適なサンプル時間を特定し，500K トーレスを用いた CPA 攻撃により，Niels と Daemen の攻撃手法に従い，正解鍵の復元に成功したことが示されている．２つある選択関数では，攻撃の成功率に差がある

という結果が得られている．

著者らは，マスキング対策のある Ascon に対しても同様の TVLA 及び CPA 攻撃を行っている．Ascon 設計者らによる公式コードの中でも，Arm-v6 向けに作成されたものを用いている．このソフトウェア実装は，乱数をほとんど使わない２～４シェアで対策されたセキュア実装である．この実装に対して，15M トレースを用いて Initialization の最初に処理される Permutation に対して，CPA 攻撃の攻撃箇所（サンプル時間）の特定を行った．ノンスはランダムに変化させ，その他のデータは全て固定としている．15M 波形トーレスを用いて CPA 攻撃を行った結果，暗号処理の２個の中間値を利用する２次 CPA でも攻撃は成功しなかった．攻撃が成功しなかったのは，波形数が少なかったことが原因であるとしている．

4.3.5 まとめ

Ascon-128 のソフトウェア実装について，電力サイドチャネル攻撃の結果を示す論文である．対策なしの実装では容易に鍵復元ができたが，セキュア実装に対しては２次の CPA 攻撃でも鍵を復元することができなかった．Niels と Daemen らのシミュレーションによる CPA の結果とは異なるものである．著者らは，トレース数の問題を指摘している．これは，TVLA の結果から妥当な指摘であるが，選択関数に改善の余地があるようにも思われる．なぜならば，Ascon の場合，ハードウェア実装に用いた選択関数をそのままソフトウェア実装に適用しても，効果的な解析ができるかどうかは明らかではないからである．たしかに，対策なしの Ascon では攻撃に成功しているため，ソフトウェア実装に対しても当該選択関数が一定の精度をもっていると言えるが，乱数を用いたセキュア実装の場合には，マスキングによってアルゴリズムの処理手順が大きく異なることを考慮しなければいけない．最適な選択関数として，アルゴリズムでのデータフローだけでなく，マスキング手法や CPU のアーキテクチャ，及び攻撃者能力の前提を考慮したものを用いるべきである．さらには，プロファイリング型の攻撃に対する脆弱性を調べる必要があると考える．

4.4 Mohajerani らによる報告（2023.06） [30]

4.4.1 著者, 所属機関

- Kamyar Mohajerani: CERG, George Mason University
- Luke Beckwith, CERG: George Mason University, PQSecure Technologies
- Abubakr Abdulgadir: PQSecure Technologies
- Eduardo Ferrufino: CERG, George Mason University
- Jens-Peter Kaps: CERG, George Mason University
- Kris Gaj: CERG, George Mason University

4.4.2 概要

本論文は，Lightweight Cryptography Workshop 2023 (Day 1) で公表されたものである．

NIST の軽量暗号 (LWC) 標準化プロセスにおいて，選定における主要な指標のひとつとされているサイドチャネル攻撃耐性について，複数の研究チームが調査やベンチマークを行った結果を報告している．サイドチャネル攻撃耐性の評価には，多くのマンパワーが必要となる．なぜなら，通常の実装よりも複雑な対策技術をアルゴリズムにあわせて実装し，膨大なサイドチャネル情報を長時間かけて収集し，解析をする必要があるためである．LWC コンペティションでは，多くの暗号アルゴリズムが候補として提出された．ラウンド 1 とラウンド 2 で，それぞれ 56 候補 と 32 候補まで絞られたものの，サイドチャネル耐性評価を公平に行うことは，時間的にも労力的負担が大きいため難しい．最終ラウンドでようやく，候補が 10 になったため，サイドチャネル攻撃耐性を評価する機運が高まったように思う．本論文では，そういったタイミングで，暗号アルゴリズム候補の物理耐性評価を行う機関を集め，物理攻撃耐性の一般的な評価フレームワークを提案している．参加した大学，研究機関，及び企業は次の七つの機関である．

1) IAIK, Graz University of Technology, Austria
2) CCSL, Shanghai Jiao Tong University, China
3) HSCP Lab, Tsinghua University, Beijing, China
4) Secure-IC, France
5) CERG, George Mason University, USA
6) Ruhr-Universitat Bochum, Germany
7) CESCA Lab, Radboud University, the Netherlands

対策技術の安全性の評価を行うとともに，対策技術の追加によって，実装コストと処理パフォーマンスに与える影響を実証実験している．サイドチャネル攻撃に対して耐性を持つセキュア実装に関して，安全性を含めた実装性能が報告されている．本論文では，対象を Ascon に絞り，当該論文の内容をまとめ考察を与える．

4.4.3 評価対象，手法，及び結果

Ascon-128 のハードウェアでのセキュア実装の安全性評価には，FPGA が用いられている．Ascon-128_Bochum_d1 は，Ascon-128_Graz-x1 をベースにマスキング対策が施されたものが使われている．対策のない HDL コードから，乱数マスキング対策のある HDL コードを半自動生成する AGEMA [23] と呼ばれるツールを利用している．このコードは、Ruhr-Universitat Bochum によって生成されたとしている．AGEMA は，合成後のネットリストに対して，サイドチャネル情報攻撃に対して保護する必要があるワイヤとゲートを特定し，それらに対して必要な乱数マスキングを施す．また，AGEMA は Probe Isolating Non-Interference (PINI) [7] とコンポーザビリティの概念に基づいている．しかしながら，AGEMA は，制御ロジックに対する制御ができないため，全てのコード変換を自動化することはできない．そのため，一部のコードに対しては手動でマスキング対策を施す必要がある．これが半自動生成ツールと呼ばれる理由である．Ascon-128_Graz_d1 に対しては，Domain Oriented Masking (DOM) [19] が採用されている．Ascon-128 ハードウェアに対するマスキングは全て 1 個の中間値に着目する 1 次攻撃に対して安全とされるものである．

Ascon の FPGA 上へのセキュア実装と Arm-Cortex-M4 上へのソフトウェア実装に対して，参加した研究機関が行なった安全性評価の結果を，表 4.1 と 表 4.2 にまとめる．ハードウェア評価では，電力と電磁波のサイドチャネル情報に対して，TVLA，χ^2-test, 及び CPA 攻撃で情報漏洩の可能性の有無を確認している．波形数はおよそ 100 万から 1,000 万程度である．CREG による評価だけが TVLA のしきい値である 4.5 を超えたと報告しているが，他の機関からは特段の情報漏洩の可能性はないと報告されている．CERG ラボによる Ascon-128_Bochum_d1 のテストでは，数個 (3〜10) のサンプルで しきい値 4.5 を超えたとしている．これらのテストにおいては，攻撃対象のクロックと同期したサンプリングクロックを使用していたことが原因としている．ただし，1M を超えるトレースが考慮されるまではしきい値を超えていないとしている．

一方，ソフトウェア実装は，すべて Arm Cortex-M4 [1] 上に実装されたものである．Ascon-128_Graz_d1 と Ascon-128_Graz_d2 が用いられている．いずれの安全性評価においても，EM サイドチャネルから取得した波形データ使っている．なお，オシロスコープのサンプリングクロックは，攻撃対象の CPU の動作クロックと同期していない．

評価の結果， CPA 攻撃による鍵復元は成功していない．CESCA グループ による 2 次 CPA 攻撃では，15M トレースを使用しても Ascon-128_Graz_d1 の鍵に関する情報を一切明らかにすることができないとしている．参考までに，対策なしの Ascon 実装に対する CPA 攻撃は，500K トレースで鍵復元に成功している．このことから，用いたコードのマスキング対策は正しく機能していることが分かる．

表 4.2: 対策済み Ascon の FPGA 実装に対する安全性評価結果

ソースコード	評価機関	評価プラットフォーム	オシロスコープ	サイドチャネル	評価手法	波形数（M トレース）	評価結果
Ascon-128_Bochum_d1	CERG	CW305 (Artix-7)	FOBOS3 ADC	電力	TVLA	10	リーク有 (1.5M トレース)
Ascon-128_Bochum_d1	IAIK	CW305 (Artix-7)	PicoScope 6404C	電力	TVLA	10	リーク無
Ascon-128_Bochum_d1	CCSL	SAKURA-X (Kintex-7)	LeCroy 610Zi	電磁波	TVLA	1	リーク無
Ascon-128_Bochum_d1	CCSL	SAKURA-X (Kintex-7)	LeCroy 610Zi	電磁波	χ^2-test	1	リーク無
Ascon-128_Bochum_d1	CCSL	SAKURA-X (Kintex-7)	LeCroy 610Zi	電磁波	CPA	11	リーク無
Ascon-128_Graz_d1	HSCP	SAKURA-G (Spartan-6)	WaveRunner 8404M	電力	TVLA	10	リーク無

表 4.3: 対策済み Ascon のソフトウェア実装に対する安全性評価結果

ソースコード	評価機関	評価プラットフォーム	オシロスコープ	サイドチャネル	評価手法	波形数（M トレース）	評価結果
Ascon-128_Graz_d1	CESCA	STM32F407 (Arm Cortex-M4)	Pico 3206D	電磁波	2次 CPA	15	リーク無
Ascon-128_Graz_d2	CCSL	STM32F303 (Arm Cortex-M4)	Pico 3203D	電磁波	TVLA	0.06	リーク無
Ascon-128_Graz_d2	CCSL	STM32F303 (Arm Cortex-M4)	Pico 3203D	電磁波	χ^2-test	0.06	リーク無
Ascon-128_Graz_d2	CCSL	STM32F303 (Arm Cortex-M4)	Pico 3203D	電磁波	CPA	0.06	リーク無

4.4.4 攻撃対象のクロックとサンプリングクロックとの同期について

オシロスコープのサンプリングと攻撃対象のデバイスのクロックの同期性は，鍵復元攻撃に必要となる波形数に影響を及ぼすことが報告されている [36]．本論文の実験においても，攻撃対象のクロックに同期したサンプリングクロックを使用した場合に、非同期クロックを使用する場合と比べ，大幅に少ない波形数で情報漏洩が検出できるとしている．原因として，データパスの乱数マスキング時に制御ロジックの一部のクロックサイクルで漏洩につながる問題があり，サイドチャネルリークが発生するとしている．オシロスコープのサンプリングを攻撃対象のクロックと同期させることで，少ない波形数でも TVLA の t 値が高くなることが実験的に示されている．サンプリングレートは 50MS/s と高くないにも関わらず，このような結果が得られているのは特筆に値する．単にサイドチャネル情報を取得する波形数を増やすのではなく，測定系での同期に注意を払うことが安全性評価として厳格にできる場合があることを示唆している．しかも，攻撃者が攻撃対象のクロックを観測することは可能であるため，現実的かつ厳密な評価を行う上で，サンプリングクロックの設定は詳細な議論が必要と考える．

4.4.5 対策による面積コストと処理パフォーマンスへの影響

本論文の図から，Ascon-128_Bochum_d1 FPGA 実装では，対策により面積コストがおよそ 3 倍程度増加し，スループットが約 1/3 倍に低下していることが読み取れる．一方，Ascon-128_Graz_d1 では，対策による面積コストの増加は 2 倍弱と Ascon-128_Bochum_d1 に比べて少なく，スループットも Ascon-128_Bochum_d1 ほど低下していないことが分かる．これは，DOM を人手により実装したことで，効率の良い対策技術が実現できているためと思われる．なお，ソフトウェア実装に関する実装結果は資料には掲載されていない．

4.4.6 まとめ

Ascon については，コンペティションの最終候補を選定する段階から，物理攻撃に対する実装上の脆弱性を知ることができている．これは，早期から Ascon に高い関心が高まり，軽量暗号アルゴリズムの選定の段階から，物理攻撃耐性を含めた実装性能評価で，世界中の研究者やエンジニアの協力があったためである．この安全性評価の取り組みに，日本から参加がなかったのは残念である．暗号技術に関する実装のノウハウは，企業で蓄積されるがあまり公にされることはない．このような物理攻撃に対する安全性の評価に関する取り組みでは，思いもよらぬ攻撃に対抗するためにも，グローバルな視野を持って学際的研究を進めていくことが重要と考える．

4.5 Kandi らによる報告（2023.06） [28]

4.5.1 著者, 所属機関

- Aneesh Kandi, Indian Institute of Technology Madras
- Anubhab Baksi, Nanyang Technological University
- Tomas Gerlich, Brno University of Technology
- Sylvain Guilley, Télécom Paris, Secure-IC
- Peizhou Gan, Nanyang Technological University
- Jakub Breier, Silicon Austria Labs
- Anupam Chattopadhyay, Nanyang Technological University
- Ritu Ranjan Shrivastwa, Télécom Paris, Secure-IC
- Zdenek Martinasek, Brno University of Technology
- Shivam Bhasin, Nanyang Technological University

4.5.2 概要

本論文は，Lightweight Cryptography Workshop 2023 (Day 1) で公表されたものである．著者らは，Ascon のハードウェア実装に関して，Ascon AEAD の暗号化とタグ生成，復号とタグ検証，及び Ascon のハッシュ関数としての処理性能について報告している．サイドチャネル攻撃への対策が施されていない実装に加えて，TI によるサイドチャネル攻撃対策と計算の 3 重化による故障利用攻撃対策が紹介されている．サイドチャネル攻撃対策と故障利用攻撃対策は，互いに直交した概念に基づき実装されているため，それぞれの対策が相互に影響しないとしている．つまり，要求仕様に応じて，どちらかの対策を実装することも，両方の対策を実装することも可能としている．

4.5.3 実装性能評価の結果

STM 130nm ライブラリによる ASIC 実装では，TI で保護された Ascon の 5 ビット S-box のゲートサイズは 56 gate で，線形層は 320 gate としている．このことから，1 サイクルで 1 ラウンドを処理する回路では，組み合わせのゲートサイズは少なくとも 12.6 Kgate 程度のコス

表 4.4: Kandi らによる Ascon AEAD 処理の ASIC 実装（STM 130nm）

コア	面積コスト [μm^2]	クリティカルパス遅延時間 [psec]
暗号化とタグ生成	73,803 (1.00)	8,595 (1.00)
復号とタグ検証	71,873 (0.97)	8,586 (1.00)
暗号化とタグ生成（3 シェア TI）	273,857 (3.71)	10,001 (1.16)
復号とタグ検証（3 シェア TI）	274,688 (3.72)	9,981 (1.16)

表 4.5: Ascon AEAD 処理の FPGA 実装（Kintex-7）

コア	面積コスト [LUT]	クロック周期* [psec]
暗号化とタグ生成	944 (1.00)	5,525 (1.00)
復号とタグ検証	1,05 8 (0.97)	5,525 (1.00)
暗号化とタグ生成（3 シェア TI）	3,977 (4.21)	6,024 (1.09)
復号とタグ検証（3 シェア TI）	3,795 (4.02)	6,010 (1.09)

∗ 表 4.4 及び 表 4.6 との比較のため，最大動作周波数からクロック周期を算出した.

トが必要なことが分かる．実際の実装では，ステートを保持する F/F，インターフェイス回路，乱数生成器が必要となる．

著者らは，まず Ascon AEAD の暗号化と復号処理の機能に関して，ASIC と FPGA（Kintex-7）の実装結果をまとめている（表 4.4 と 表 4.5）．ASIC 実装では，暗号化処理と復号処理での違いはほとんど見られないという結果を報告している．これは，暗号化処理と復号処理で Ascon のデーターパスの違いにほとんどないためである．3 シェアの TI による対策実装では，回路面積は 4 倍弱になっている．TI のシェア数は Ascon S-box の代数次数 2 に 1 を加えた 3 以上でなければならず，今回は 3 シェアを採用したとしている．

Kintex-7 は，28 nm テクノロジを採用した FPGA である．最先端テクノロジではないが，低電力でコストパフォーマンスの良い FPGA として広く利用されている．表 4.5 から分かるように，FPGA 実装においても，暗号化処理と復号処理での違いはほとんど見られない．3 シェア TI のセキュア実装では，暗号化処理と復号処理のいずれの回路サイズも 4 倍以上の LUT を必要としている．一方，クロック周期については，10% 程度の増加と少ない．残念ながら，レイテンシやスループットの処理パフォーマンスに対する考察は報告されていない．

4.5.4 S-Box に用いられた 3 シェア TI

本論文で採用された Ascon ハードウェアのアーキテクチャは，1 サイクルで Ascon の Permutation p を処理するものである．前述のとおり，非線形処理の S-box に対して，3 シェア TI が適用されている．その詳細を以下に示す．5 ビットの Ascon S-box の入力値 x_i $(0 \leq i \leq 4)$ を，x_{i0}, x_{i1}, x_{i2} の 3 つの値に分ける．$x_i = x_{i0} \oplus x_{i1} \oplus x_{i2}$ である．3 つのシェアに分けて以下の処理を行うことで，出力 y_i が得られる．ここで，$y_i = y_{i0} \oplus y_{i1} \oplus y_{i2}$ である．

【シェア０】

$$
\begin{aligned}
y_{00} &= x_{00} \oplus x_{01}x_{11} \oplus x_{01}x_{12} \oplus x_{01} \oplus x_{11}x_{21} \oplus x_{11}x_{41} \oplus x_{11}x_{02} \oplus x_{11}x_{22} \oplus x_{11}x_{42} \\
&\quad \oplus x_{11} \oplus x_{21}x_{12} \oplus x_{21} \oplus x_{31} \oplus x_{41}x_{12} \oplus x_{02}x_{12} \oplus x_{12}x_{22} \oplus x_{12}x_{42} \oplus x_{12} \oplus x_{22} \oplus x_{32}, \\
y_{20} &= x_{20} \oplus x_{11} \oplus x_{21} \oplus x_{31}x_{41} \oplus x_{31}x_{42} \oplus x_{41}x_{32} \oplus x_{41} \oplus x_{12} \oplus x_{32}x_{42} \oplus x_{42} \oplus 1, \\
y_{10} &= x_{10} \oplus x_{01} \oplus x_{11}x_{21} \oplus x_{11}x_{31} \oplus x_{11}x_{22} \oplus x_{11}x_{32} \oplus x_{11} \oplus x_{21}x_{31} \oplus x_{21}x_{12} \oplus x_{21}x_{32} \\
&\quad \oplus x_{21} \oplus x_{31}x_{12} \oplus x_{31}x_{22} \oplus x_{31} \oplus x_{41} \oplus x_{02} \oplus x_{12}x_{22} \oplus x_{12}x_{32} \oplus x_{22}x_{32} \oplus x_{22} \oplus x_{32} \oplus x_{42}, \\
y_{30} &= x_{30} \oplus x_{01}x_{31} \oplus x_{01}x_{41} \oplus x_{01}x_{32} \oplus x_{01}x_{42} \oplus x_{01} \oplus x_{11} \oplus x_{21} \oplus x_{31}x_{02} \oplus x_{31} \oplus x_{41}x_{02} \\
&\quad \oplus x_{41} \oplus x_{02}x_{32} \oplus x_{02}x_{42} \oplus x_{02} \oplus x_{12} \oplus x_{22} \oplus x_{42}, \\
y_{40} &= x_{40} \oplus x_{01}x_{11} \oplus x_{01}x_{12} \oplus x_{11}x_{41} \oplus x_{11}x_{02} \oplus x_{11}x_{42} \oplus x_{11} \oplus x_{31} \oplus x_{41}x_{12} \oplus x_{41} \\
&\quad \oplus x_{02}x_{12} \oplus x_{12}x_{42} \oplus x_{12} \oplus x_{32}.
\end{aligned}
$$

【シェア１】y_{01} の式にある $y_{20}x_{11}$ は $x_{20}x_{11}$ の誤りと思われる.

$$
\begin{aligned}
y_{01} &= x_{00}x_{10} \oplus x_{00}x_{11} \oplus x_{00}x_{12} \oplus x_{10}x_{20} \oplus x_{10}x_{40} \oplus x_{10}x_{01} \oplus x_{10}x_{21} \oplus x_{10}x_{41} \oplus x_{10}x_{02} \\
&\quad \oplus x_{10}x_{22} \oplus x_{10}x_{42} \oplus x_{10} \oplus x_{20}x_{11} \oplus x_{20}x_{12} \oplus x_{20} \oplus x_{30} \oplus x_{40}x_{11} \oplus x_{40}x_{12}, \\
y_{11} &= x_{00} \oplus x_{10}x_{20} \oplus x_{10}x_{30} \oplus x_{10}x_{21} \oplus x_{10}x_{31} \oplus x_{10}x_{22} \oplus x_{10}x_{32} \oplus x_{20}x_{30} \oplus x_{20}x_{11} \oplus x_{20}x_{31} \\
&\quad \oplus x_{20}x_{12} \oplus x_{20}x_{32} \oplus x_{20} \oplus x_{30}x_{11} \oplus x_{30}x_{21} \oplus x_{30}x_{12} \oplus x_{30}x_{22} \oplus x_{30} \oplus x_{40}, \\
y_{21} &= x_{10} \oplus x_{30}x_{40} \oplus x_{30}x_{41} \oplus x_{30}x_{42} \oplus x_{40}x_{31} \oplus x_{40}x_{32} \oplus x_{40}, \\
y_{31} &= x_{00}x_{30} \oplus x_{00}x_{40} \oplus x_{00}x_{31} \oplus x_{00}x_{41} \oplus x_{00}x_{32} \oplus x_{00}x_{42} \oplus x_{00} \oplus x_{10} \oplus x_{20} \oplus x_{30}x_{01} \\
&\quad \oplus x_{30}x_{02} \oplus x_{40}x_{01} \oplus x_{40}x_{02} \oplus x_{40}, \\
y_{41} &= x_{00}x_{10} \oplus x_{00}x_{11} \oplus x_{00}x_{12} \oplus x_{10}x_{40} \oplus x_{10}x_{01} \oplus x_{10}x_{41} \oplus x_{10}x_{02} \oplus x_{10}x_{42} \oplus x_{10} \oplus x_{30} \\
&\quad \oplus x_{40}x_{11} \oplus x_{40}x_{12}.
\end{aligned}
$$

【シェア２】

$$
\begin{aligned}
y_{02} &= x_{02}, \\
y_{12} &= x_{12}, \\
y_{22} &= x_{22}, \\
y_{32} &= x_{32}, \\
y_{42} &= x_{42}.
\end{aligned}
$$

本論文では，３シェア実装の別の例を紹介している．

【シェア０】

$$
\begin{aligned}
y_{00} &= x_{01}x_{30} \oplus x_{01}x_{31} \oplus x_{01}x_{32} \oplus x_{01} \oplus x_{02}x_{30} \oplus x_{02}x_{31} \oplus x_{02}x_{32} \oplus x_{02} \oplus x_{11} \oplus x_{12} \\
&\quad \oplus x_{30}x_{40} \oplus x_{30}x_{41} \oplus x_{30}x_{42} \oplus x_{30} \oplus x_{31}x_{40} \oplus x_{31}x_{41} \oplus x_{31}x_{42} \oplus x_{31} \oplus x_{32}x_{40} \oplus x_{32}x_{41} \\
&\quad \oplus x_{32}x_{42} \oplus x_{32}, \\
y_{10} &= x_{01}x_{40} \oplus x_{01}x_{41} \oplus x_{01}x_{42} \oplus x_{01} \oplus x_{02}x_{40} \oplus x_{02}x_{41} \oplus x_{02}x_{42} \oplus x_{02} \oplus x_{11}x_{40} \oplus x_{11}x_{41} \\
&\quad \oplus x_{11}x_{42} \oplus x_{11} \oplus x_{12}x_{40} \oplus x_{12}x_{41} \oplus x_{12}x_{42} \oplus x_{12} \oplus x_{21} \oplus x_{22} \oplus x_{30} \oplus x_{31} \oplus x_{32} \oplus x_{40} \\
&\quad \oplus x_{41} \oplus x_{42}, \\
y_{20} &= x_{01}x_{11} \oplus x_{01}x_{12} \oplus x_{01} \oplus x_{02}x_{11} \oplus x_{02}x_{12} \oplus x_{02} \oplus x_{21} \oplus x_{22} \oplus x_{30} \oplus x_{31} \oplus x_{32} \oplus 1, \\
y_{30} &= x_{01} \oplus x_{02} \oplus x_{11}x_{21} \oplus x_{11}x_{22} \oplus x_{11}x_{30} \oplus x_{11}x_{31} \oplus x_{11}x_{32} \oplus x_{11} \oplus x_{12}x_{21} \oplus x_{12}x_{22} \\
&\quad \oplus x_{12}x_{30} \oplus x_{12}x_{31} \oplus x_{12}x_{32} \oplus x_{12} \oplus x_{21}x_{30} \oplus x_{21}x_{31} \oplus x_{21}x_{32} \oplus x_{21} \oplus x_{22}x_{30} \oplus x_{22}x_{31} \\
&\quad \oplus x_{22}x_{32} \oplus x_{22} \oplus x_{30} \oplus x_{31} \oplus x_{32} \oplus x_{40} \oplus x_{41} \oplus x_{42}, \\
y_{40} &= x_{01}x_{30} \oplus x_{01}x_{31} \oplus x_{01}x_{32} \oplus x_{02}x_{30} \oplus x_{02}x_{31} \oplus x_{02}x_{32} \oplus x_{11} \oplus x_{12} \oplus x_{21}x_{30} \oplus x_{21}x_{31} \\
&\quad \oplus x_{21}x_{32} \oplus x_{21} \oplus x_{22}x_{30} \oplus x_{22}x_{31} \oplus x_{22}x_{32} \oplus x_{22} \oplus x_{30}x_{40} \oplus x_{30}x_{41} \oplus x_{30}x_{42} \oplus x_{30} \\
&\quad \oplus x_{31}x_{40} \oplus x_{31}x_{41} \oplus x_{31}x_{42} \oplus x_{31} \oplus x_{32}x_{40} \oplus x_{32}x_{41} \oplus x_{32}x_{42} \oplus x_{32} \oplus x_{40} \oplus x_{41} \oplus x_{42}.
\end{aligned}
$$

【シェア１】

$$
\begin{aligned}
y_{01} &= x_{00}x_{30} \oplus x_{00}x_{31} \oplus x_{00}x_{32} \oplus x_{00}, \\
y_{11} &= x_{00}x_{40} \oplus x_{00}x_{41} \oplus x_{00}x_{42} \oplus x_{00} \oplus x_{10}x_{40} \oplus x_{10}x_{41} \oplus x_{10}x_{42} \oplus x_{10}, \\
y_{21} &= x_{00}x_{10} \oplus x_{00}x_{12} \oplus x_{00} \oplus x_{02}x_{10} \oplus x_{20}, \\
y_{31} &= x_{00} \oplus x_{10}x_{20} \oplus x_{10}x_{22} \oplus x_{10}x_{30} \oplus x_{10}x_{31} \oplus x_{10}x_{32} \oplus x_{10} \oplus x_{12}x_{20} \oplus x_{20}x_{30} \oplus x_{20}x_{31} \\
&\quad \oplus x_{20}x_{32} \oplus x_{20}, \\
y_{41} &= x_{00}x_{30} \oplus x_{00}x_{31} \oplus x_{00}x_{32} \oplus x_{10} \oplus x_{20}x_{30} \oplus x_{20}x_{31} \oplus x_{20}x_{32}.
\end{aligned}
$$

【シェア２】

$$
\begin{aligned}
y_{02} &= x_{10}, \\
y_{12} &= x_{20}, \\
y_{22} &= x_{00}x_{11} \oplus x_{01}x_{10}, \\
y_{32} &= x_{10}x_{21} \oplus x_{11}x_{20}, \\
y_{42} &= x_{20}.
\end{aligned}
$$

4.5.5　３重化による故障利用攻撃対策

暗号モジュールに対して，これまでに数多くの故障攻撃が提案されている．差分故障解析（DFA: Differential Fault Analysis）[3] は最も知られている攻撃の一つであるが，暗号処理の２

表 4.6: Ascon AEAD の ASIC 実装（STM 130nm）

SCA 対策	FA 対策	面積コスト [μm^2]	クリティカルパス遅延時間 [psec]
-	-	98,524 (1.00)	8,520 (1.00)
-	3重化	258,224 (2.62)	8,518 (1.00)
3シェア TI	-	364,320 (3.70)	9,830 (1.15)
3シェア TI	3重化	948,544 (9.63)	9,832 (1.15)

重化，つまり同じ処理を2回行い，その結果を比較することで誤り暗号文を出力しないといった対策が存在する．Fault Sensitivity Analysis (FSA) [27] や Statistical Ineffective Fault Attack (SIFA) [10] といったさらに高度な攻撃では，単純な2重化の対策では不十分である．最新の共通鍵暗号への故障攻撃については，SoK 論文 [2] が詳しい．

著者らは，同じ処理を3回行い，結果の多数を出力とする対策を提案している．もし，3つの結果が全て異なる場合には，乱数を出力するとしている．故障利用攻撃に対する安全性評価の報告がないため詳細は不明であるが，3重化は原理的には DFA 対策としては十分であると考える．一方で，FSA 攻撃や SIFA 攻撃に対しては，一定の効果はあると思われるが，より強力な攻撃者に対しては厳密な安全性評価が必要と考える．

Ascon のコア部分の回路規模は，単純に3倍となる．より正確には，多数決により結果を決定する処理が必要となるため，3倍よりも大きくなる．表 4.6 から分かるように，実際にはインタフェイスなどの回路面積は増えないため，全体としては3倍弱となっている．また，空間的な3重化を行っているため，クリティカルパス遅延時間への影響はない．

4.5.6 まとめ

Ascon のハードウェア実装に関して，サイドチャネル対策と故障利用解析について実装結果を示している．3シェア TI については，乱数コストを下げる工夫が可能と思われる．また，故障利用解析対策として，3重化は一定の効果があるものの，協力な攻撃者を想定した評価についてはさらなる実験が必要と考える．元のサイズの約 10 倍となるコスト増については，現実的ではなく一層のコスト削減が必要である．故障利用解析においては，Daemen らのトフォリゲートを用いた効率的な対策 [11] が提案されている．また，暗号アルゴリズムだけの対策を進めるのではなく，レーザー検知などのセンサーの利用を考慮するべきである．例えば，文献 [29] では，AES 暗号ハードウェアのレーザーを利用した故障利用攻撃対策を 面積コスト 28% 増で実装できるとしている．処理パフォーマンスの低下はあるものの，センサーの感度を高くすることで，Ineffective Fault を用いた攻撃は全て無効とすることができる．アルゴリズムレベルでの対策とセンサーレベルでの対策を融合する研究が必要と考える．

4.6 Gigerl らによる報告（2023.06） [17]

4.6.1 著者, 所属機関

- Barbara Gigerl, Graz University of Technology
- Florian Mendel, Infineon Technologies
- Martin Schläffer, Infineon Technologies
- Robert Primas, Graz University of Technology

4.6.2 概要

この資料は，Lightweight Cryptography Workshop 2023 (Day 1) で公表されたものである．

この論文では，サイドチャネル攻撃耐性を持つ Ascon の効率的なソフトウェア実装が提案されている．２個の中間値を利用する２次の電力解析攻撃耐性を目指すものである．既存技術での Keccak S-box の１次マスキングを拡張し，コストの高いオンラインでの乱数性を必要としない効率的な２次マスキングが実装されている．また，処理パフォーマンスをさらに向上させるべく，マスクされたソフトウェアにおけるシェアを減らす実装のコツについて議論されている．ソフトウェアは，ARM Cortex-M4 をコアとする STM32F303 マイクロプロセッサに実装され，TVLA によって対策技術の安全性が評価されている．さらに，RISC-V Ibex コアのネットリストに対して，ゲートレベルの安全性検証ツール Coco (Co-Design and Co-Verification of Masked Software Implementations [16]) を使用して，ソフトウェア実装の安全性証明を与えている．マスキング対策の施された Ascon-128 認証暗号化ソフトウェアの ARM 及び RISC-V におけるベンチマークの結果，２シェアと３シェアで，それぞれ約 300， 550 Cycles/Byte のスループットが実現されたとしている．Initialization と Finalization のみに施す対策である Leveled Implementation のテクニックを利用すると，マスクされた実装のスループットは約 90 Cycles/Byte まで増加するとしている．

4.6.3 Coco

Coco は，CPU のネットリスト上に実装されたマスク対策の施されたソフトウェアの協調設計，及び検証ツールである [16]．Coco は，ゲートレベルのネットリストで記述された CPU 上で，マスクされたアセンブリを実行した際の安全性を検証することができ，d プローブを空間的及び時間的に実現することができる．つまり，同じクロックサイクル内の異なる位置で d 回のプロビングによる測定を実行したり，異なるクロックサイクルで同じ位置にプローブ測定を d 回実行したり，あるいは，その両方を組み合わせられる攻撃者を想定することができる．攻撃者が元の情報を復元できない場合，マスクされたソフトウェア実装はこの d プローブモデルにおいて d 次安全であるとみなしてる．

4.6.4 攻撃対象及び評価環境

攻撃対象は，Daeman らによる 2 シェアの 5 ビットの S-box [11] と Shahmirzadi と Moradi による 3 シェアの 5 ビットの S-box [38] である．さらに，Share-rotation と呼ばれる技術を追加している．ターゲットデバイスは，STM32F303 マイクロプロセッサと IBEX である．

STM32F303 は，32 ビットの ARM Cortex-M4 をコアとして搭載しており，ChipWhisperer UFO ボード [31] とツールチェーンの組み合わせで性能を評価を行っている．実験では，鍵，ノンス，平文を乱数マスキングしてから，ターゲットデバイスに送信している．また，本論文では，オンラインの乱数生成器を必要ではないため，ターゲットデバイス上の単純なソフトウェアで生成した乱数を使っている．TVLA の評価では，10M トレースの電力波形を用いて実施している．

一方，IBEX では，RISC-V IBEX コアのネットリスト上で，Ascon Permutation 1 ラウンド分を実装し，サイクルベースのシミュレーションを行っている．形式検証は，先述の Coco を用いて実施している．

4.6.5 評価結果

処理性能については，表 4.7 に示す結果が報告されている．

STM32F303 に対する TVLA の結果，2 シェア実装では，t 検定の一部でリークが見られる．実際のマスキングされたソフトウェア実装では，マイクロアーキテクチャレベルでのリークが存在することは知られており，今回も同様の結果となっている．3 シェア実装では，Share-rotation の効果もあってか，10M トレースの TVLA では，顕著なリークは見られなかった．

4.6.6 まとめ

本論文では，マスキング対策された Ascon のソフトウェア実装に対する安全性評価と性能評価が報告されている．STM32F303 上の実装では，検証ツールと実装結果で安全性評価の結果が一致しない例を紹介している．マイクロアーキテクチャからのリーケージは，これまでの研究でも報告があり，今後さらなら研究が必要と考える．また，協調設計／検証ツールである Coco を用いた RISC-V IBEX に対する評価結果についても言及されている．一般的に，実装前の安全性検証においては，現実よりも厳しい基準を満たすものでなければ意味をなさない．本論文では，追加の対策である Share-rotation により，TVLA でのリークはなくなったとしているが，

表 4.7: Daeman らによる Ascon のソフトウェア実装の結果 [Cycles/Byte]

実装	STM32F303	IBEX
対策なし	59	-
Leveled Implementation	89	-
2 シェア	318	260
3 シェア	542	500

理想的には Share-rotation の効果がシミュレートできるほどの精度を持つ検証ツールが必要と考える．つまり，Share-rotation により， TVLA でリークがなくなる理由をマイクロアーキテクチャと照らして明確にし，検証ツールにフィードバックするなどの取り組みが期待される．

4.7 Liu と Schaumont による報告（2023.06） [28]

4.7.1 著者, 所属機関

- Zhenyuan Liu, Worcester Polytechnic Institute
- Patrick Schaumont, Worcester Polytechnic Institute

4.7.2 概要

本論文は，Lightweight Cryptography Workshop 2023 (Day 1) で公表されたものである．

Ascon のサイドチャネルからの情報漏洩，つまりサイドチャネルリークを分析し，情報漏洩の根本原因の特定をハードウェアのコンポーネント単位で試みるものである．RISC-V SoC として実装された 反復型の Ascon のアクセラレータ実装と，RISC-V (RV32IMC) 上のソフトウェア実装に対して，サイドチャネルリーケージを分析している．ゲートレベルの電力シミュレーションを使用して，電力波形でリーケージが発生する箇所の当をつけ，ハードウェア及びソフトウェア実装のどの処理部分で秘密鍵の漏洩が最も多いかを特定している．シミュレーションによる電力波形と，同じ設計の 180 nm ASIC 実装から測定した波形とを比較し，シミュレーションの精度の違いなどを議論している．

4.7.3 攻撃対象及び評価環境

本研究では，著者らは，Ascon を二つの異なる実装で安全性評価を行っている．ひとつは，ハードウェアコプロセッサによる実装であり，もう一つは，ソフトウェア実装である．いずれの実装でも，180 nm スタンダードセル の RISC-V SoC が利用されている．ゲートレベルのネットリストと実際のシリコンに対して，情報漏洩の根本原因の特定に関する実験を行い，安全性を評価している．

Ascon のコプロセッサには，Fivez が設計した反復型アーキテクチャを使用している [15] ．コプロセッサは，PicoRV32 RISC-V の SoC の一部として組み込まれ，平文や AD などのデータ，IV，ノンス，及び鍵は，全てメモリマップインターフェイス経由で制御される．ハードウェア実装における情報漏洩の根本原因の解析は，アクセラレータへの鍵の取り込み時と Initialization 処理である．4 MHz で動作させたハードウェアに対して，クロック周波数の 4 倍のサンプリング周波数で電力波形を取得している．波形数は 2,000 トレースであり，鍵に対して Random-vs-Fix の評価（固定鍵とランダム鍵に対する評価）を行っている．鍵以外のパラメータは固定としている．

一方，Ascon のソフトウェア実装は，Ascon 設計者による ASCON128v12 のリファレンス実装を `-O2` の最適化オプションでコンパイルしたものを使用している．ソフトウェア実装に対しては，4 MHz 動作の RISC-V (RC32IMC) に対して，クロック周波数と同じサンプリング周波数で電力波形をシミュレーションで取得している．波形数は 1,000 トレースとハードウェアの評

価と比べて少ないが，鍵に対する Random-vs-Fix の評価の条件などは同じである．鍵の取り込み部と Initialization 処理に加え，Finalization 処理も評価の対象としている．

4.7.4 評価の結果

コプロセッサに対する評価

著者らの実装では，鍵を 32 ビットずつコプロセッサに書き込んでいる．RAM に格納された鍵データを読み出し，32 ビットのバスでコプロセッサに送っている．全ての鍵ビットの書き込みには 214 クロックサイクルが必要としている．この鍵の書き込みに関する処理において，サイドチャネルリークのシミュレーションを実施したところ，鍵の漏洩の可能性があるとされるゲート数は，最大で約 750 gates/cycle であるとしている．チップ全体のセル数が，57,671 であることと比べて，ゲート数は十分少ないとしている．

Initialization においても同様の実験を行い，12 クロックサイクルで 2,000 gate 以上のサイドチャネルリークが疑われる箇所が見つかったとしている．しかし，これに関しても，チップ全体からすればごく僅かであるとしている．

ソフトウェアに対する評価

ソフトウェア実装では，Ascon Permuation は 900 サイクルで処理される．Initialization における最初のラウンド処理では，Permutation 処理に加えて，鍵とノンスの読み出し部分に 957 クロックサイクルを消費し，そのうち，239 サイクルでリークの疑いのあるゲートが見つかったとしている[*3]．Initialization の 12 ラウンド目における鍵との XOR 演算には，973 クロックサイクルを消費し，そのうち，30 サイクルでリークの疑いのあるゲートが見つかったとしている．Finalization の最初のラウンドにおける鍵との XOR 演算には，941 クロックサイクルを消費し，そのうち，48 サイクルでリークの疑いのあるゲートが見つかったとしている．Finalization における鍵との XOR 演算には，922 クロックサイクルを消費し，そのうち，26 サイクルでリークの疑いのあるゲートが見つかったとしている．いずれのリークについても，コプロセッサでの評価と同様，チップ全体のセル数と比べて漏洩の可能性のあるゲート数は少ないとしている．

4.7.5 実測による妥当性の評価

最後に著者らは，Ascon ソフトウェア実装に対して，情報漏洩の根本原因に関する分析のシミュレーション結果と，同じ設計に基づくチップ実装から取得したサイドチャネル情報の実装値と比較している．実測では，測定ノイズが問題となるため，使用する電力波形数を 50,000 トレースとしている．

実測値での t 検定の結果は，鍵の読み出し，Initialization 処理，及び Finalization における t 検定の結果と類似している．著者らは，今回行った根本原因の分析に関して，シミュレーションと実測で，意味のあるつながりが見られるとしている．

[*3] クロックサイクルによってリークの疑いのあるゲート数は異なる．

4.7.6 まとめ

この論文の結果から，チップ作製前のシミュレーションによる t 検定においても，リークのある可能性があるゲートが見つけられる可能性が示されたと考える．対策技術が考慮していないリークの存在を解明するツールが期待できるため，研究の方向性として興味深い．しかしながら，今回の実験では，全体の回路サイズに対して検知できた漏洩原因の対象となるゲート数がそもそも少なかったため，漏洩原因の特定には至っていない．シリコン作製前後でのリーケージの検知を比較することで，設計フローにおける早期のサイドチャネルリーク対策につながりうる研究といえる．

プレシリコンにおける安全性解析の有効性は，セキュリティ上のマージンを含むものでなければならないと考える．つまり，実測では検知できない脆弱性をシミュレーションは検知できなければならない．シリコン作製前にリーケージが検知できれば，生産性が向上する．そういった意味においても，本論文を含めて，物理攻撃の安全性を担保する設計及び検証のツール開発にはさらなるエフォートが必要と考える．

4.8 You らによる報告（2023.09） [45]

4.8.1 著者, 所属機関

- Shih-Chun You, University of Cambridge
- Markus G. Kuhn, University of Cambridge
- Sumanta Sarkar, University of Warwick
- Feng Hao, University of Warwick

4.8.2 概要

この資料は，Transaction of Cryptographic Hardware and Embedded Systems (TCHES) 2023 に採択されたものである [45]．本論文では，Ascon 実装のサイドチャネル攻撃のリスクを評価をするために，Weatherley の Ascon-128 の 32 ビット実装 [44] を STM32F303 (Arm Cortex-M4) 上に実装し，電力ベースのテンプレート攻撃について評価結果を紹介するものである．著者らは，フラグメントテンプレート攻撃とビリーフプロパゲーション（Beleif propagatation: 確率伝播法）法とキーエニュメレーション（Key Enumeration: 鍵列挙）技術を組み合わせて安全性評価を行なった．主な成果として，大きく 3 つの報告がなされている．

1) サイドチャネル対策のないコードに対して，コンパイル時に `-Os` で最適化した場合，単一の波形トレースで攻撃成功率が 100% になった．
2) コンパイラの最適化オプションを `-O3` とした場合，電力波形 3 トレースで成功率が約 95 % になった．
3) マスキング対策のあるコードに対して `-Os` で最適化した場合は，最大 2^{24} 個の鍵候補を列挙した後に，電力波形を 20 トレースを用いると攻撃成功率が 90% 以上となった．

1 次マスキングで保護された Ascon 実装であっても，テンプレート攻撃によって鍵漏洩の危険性のあることが示されている．さらには、プログラミングのスタイルの違いや，コンパイラの最適化の設定でさえも結果に大きな影響を与える可能性があることが示されている．

4.8.3 攻撃対象及び評価環境

著者らは，Weatherley が作成した Ascon-128 の C コード [44] を SCA プラットフォームである ChipWhisperer-Lite 上の 32 ビットのミックスドシグナルのマイクロコントローラである STM32F303 に実装した．CPU コアは，Arm Cortex-M4 である．電力波形は，ターゲットボードに 5 MHz のクロックを供給し，10 ビットのオシロスコープ PXIe-5160 を使用して，サンプリングレート 2.5 GHz としている．

著者らの実験によると，フラグメントテンプレート攻撃の結果，異なる鍵を使うことで，より攻撃に有利に働くテンプレートが作成できることを示した．著者らは，ビリーフプロパゲーショ

ンとキーエニュメレーション技術の効果を調べるために，ビリーフプロパゲーションを使用しない場合や，ルーピービリーフプロパゲーションといったアルゴリズムに変更をして，最適化オプション -Os を使って実験を行い，鍵列挙の探索の深さに対する成功率の関係を明らかにした．結果は、単一のトレースでループ状の信念伝播を使用すると、2^{32} 個の鍵列挙がほぼ 100% の成功率で達成でき，ツリー状の確率伝播法を適用すると，2^{20} まで削減できることを示している．

さらに，コンパイラオプションの最適化がテンプレート攻撃に与える影響を調べるために，-O3 で実験を行った．ここで，Weatherley のコードは，Ascon Permutation の部分はアセンブリ記述であるため，最適化オプションの影響は受けないものとしている．実験の結果，攻撃効率は明らかに悪くなることが示されている．3 個の波形トレースで成功率が約 95% であり，ビリーフプロパゲーションとキーエニュメレーション技術の両方が揃わなければ，-O3 での攻撃はほとんど実用的ではないと結論づけている．

最後に，1 次攻撃に耐性のあるブーリアンマスキング対策をした Weatherley の C コードを用いて同様の実験を行った．最適化オプションは，-Os である．実験の結果，他の 2 種類の実験と比べて多くのトレースが必要であるものの，20 トレースで 2^{24} 個のキーエニュメレーションの成功率が 90% となったとしている．

4.8.4 まとめ

本論文の研究により，Ascon に対しても効率的なプロファイリングに基づくサイドチャネル攻撃が効果的であることが明らかとなった．Ascon のセキュア実装に対する効率的な性能評価手のひとつとして重要と思われる．Ascon のソフトウェア実装に対する安全性評価として，この論文でも取り扱われたテンプレート攻撃は，厳格な評価を実施する上で不可欠なものである．プロファイリング技術の深化を考慮しつつ，TVLA やプロファイリングを用いない鍵復元攻撃との関係性についても考慮する必要がある．また，コンパイラオプションによるサイドチャネルリークへの影響については，プラットフォーム毎に，安全性向上にむけた研究の取り組みや既存製品の評価実験が必要である．

参考文献

[1] Arm Cortex-M4, https://www.arm.com/ja/products/silicon-ip-cpu/cortex-m/cortex-m4.

[2] Anubhab Baksi, Shivam Bhasin, Jakub Breier, Dirmanto Jap, and Dhiman Saha, "A Survey On Fault Attacks On Symmetric Key Cryptosystems," *ACM Computing Surveys*, Vol.55, No.4, pages 1-34, 2022.

[3] Eli. Biham and Adi Shamir, "Differential fault analysis of secret key cryptosystems," In Burton S. Kaliski Jr., editor, *Advances in Cryptology — CRYPTO*, volume 1294 of Lecture Notes in Computer Science, pages 513–525 Springer, 1997. Available at https://link.springer.com/content/pdf/10.1007/BFb0052259.pdf.

[4] CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness. https://competitions.cr.yp.to/caesar.html.

[5] Lejla Batina, Ileana Buhan, Łukasz Chmielewski, Ellen Gunnarsdóttir, Vahid Jahandideh, Tom Stock, and Léo Weissbart, "Side-Channel Evaluation Report on Implementations of Several NIST LWC Finalists," Nijmegen : Cryptographic Engineering & Side-Channel Analysis (CESCA) Lab, 2022. Available at https://repository.ubn.ru.nl/bitstream/handle/2066/253567/253567.pdf?sequence=1&isAllowed=y.

[6] Eric Brier, Christophe Clavier, and Francis Olivier, "Correlation Power Analysis with a Leakage Model," In Marc Joye and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems (CHES)*, volume 3156 of Lecture Notes in Computer Science, pages 16–29. Springer, 2004. Available at https://link.springer.com/content/pdf/10.1007/978-3-540-28632-5_2.pdf.

[7] Gaëtan Cassiers and François-Xavier Standaert, "Trivially and Efficiently Composing Masked Gadgets With Probe Isolating Non-Interference," *IEEE Transactions on Information Forensics and Security*, Vol.15, pages 2542-2555, IEEE, 2020.

[8] Hao Cheng, Johann Großschädl, Ben Marshall, Dan Page, and Thinh Pham, "RISC-V Instruction Set Extensions for Lightweight Symmetric Cryptography," *IACR Transactions on Cryptographic Hardware and Embedded Systems (TCHES)*, Vol.2023, No.1, pages 192-237, 2023. Available at https://tches.iacr.org/index.php/TCHES/article/view/9951/9454.

[9] Jean-Sébastien Coron and Louis Goubin, "On Boolean and Arithmetic Masking against Differential Power Analysis," In Çetin Kaya Koç and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems (CHES)*, volume 1965 of Lecture Notes in Computer Science, pages 231-237, Springer, 2000. Available at https://link.

springer.com/content/pdf/10.1007/3-540-44499-8_18.pdf.

[10] Christoph Dobraunig, Maria Eichlseder, Thomas Korak, Stefan Mangard, Florian Mendel, and Robert Primas, "SIFA: Exploiting Ineffective Fault Inductions on Symmetric Cryptography," *IACR Transactions on Cryptographic Hardware and Embedded Systems (TCHES)*, Vol.2018, No.3, pages 547–572, 2018. Available at https://tches.iacr.org/index.php/TCHES/article/view/7286/6463.

[11] Joan Daemen, Christoph Dobraunig, Maria Eichlseder, Hannes Gross,Florian Mendel, and Robert Primas, "Protecting against Statistical Ineffective Fault Attacks," *IACR Transactions on Cryptographic Hardware and Embedded Systems (TCHES)*, Vol.2020, No.3, pages 508-543, 2020. Available at https://tches.iacr.org/index.php/TCHES/article/view/8599/8166.

[12] Joan Daemen, Vincent Rijmen, "The Design of Rijndael," Springer, 2002. Available at https://cs.ru.nl/~joan/papers/JDA_VRI_Rijndael_2002.pdf.

[13] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer, "Ascon v1.2," *Submission document to NIST*, 2021. Available at https://ascon.iaik.tugraz.at/files/asconv12-nist.pdf.

[14] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer, "Invited talk: The Ascon Family: Lightweight Authenticated Encryption, Hashing, and More," *Lightweight Cryptography Workshop 2023*. Available at https://csrc.nist.gov/csrc/media/Presentations/2023/the-ascon-family/images-media/june-21-mendel-the-ascon-family.pdf.

[15] Michael Fivez, "Energy efficient hardware implementations of CAESAR submissions," *Master's thesis, ESAT COSIC, KULeuven*, 2016. Available at https://www.esat.kuleuven.be/cosic/publications/thesis-279.pdf.

[16] Barbara Gigerl, Vedad Hadzic, Robert Primas, Stefan Mangard, and Roderick Bloem, "Coco: Co-Design and Co-Verification of Masked Software Implementations on CPUs," In Michael Bailey and Rachel Greenstadt, editors, *Proceedings of 30th USENIX Security Symposium, USENIX Security*, pages 1469-1468, ACM, 2021. Available at https://www.usenix.org/system/files/sec21fall-gigerl.pdf.

[17] Barbara Gigerl, Florian Mendel, Martin Schläffer, and Robert Primas, "Efficient Second-Order Masked Software Implementations of Ascon in Theory and Practice," *Lightweight Cryptography Workshop 2023*. Available at https://csrc.nist.gov/csrc/media/Events/2023/lightweight-cryptography-workshop-2023/documents/accepted-papers/04-efficient-second-order-masked-software.pdf.

[18] Gilbert Goodwill, Benjamin Jun, Josh Jaffe, and Pankaj Rohatgi, "A testing method-

ology for side-channel resistance validation," *NIST Non-invasive attack testing workshop*, pages 115-136, 2011. Available at https://csrc.nist.gov/csrc/media/events/non-invasive-attack-testing-workshop/documents/08_goodwill.pdf.

[19] Hannes Groß, "Domain-Oriented Masking—Generically Masked Hardware Implementations," *PhD Thesis, IAIK, Graz University of Technology*, 2018. Available at https://diglib.tugraz.at/download.php?id=5c80ea0c43a56&location=browse.

[20] Hannes Groß, Stefan Mangard, and Thomas Korak, "Domain-Oriented Masking: Compact Masked Hardware Implementations with Arbitrary Protection Order," *IACR Cryptology ePrint Archive, Paper 2023/484*, 2023. Available at http://eprint.iacr.org/2016/486.

[21] Yuval Ishai, Amit Sahai, and David Wagner , "Private Circuits: Securing Hardware against Probing Attacks," In Dan Boneh, editor, *Advances in Cryptology - CRYPTO*, volume 2729 of Lecture Notes in Computer Science, pages 463-481, Springer, 2003. Available at https://link.springer.com/content/pdf/10.1007/978-3-540-45146-4_27.pdf

[22] Aneesh Kandi, Anubhab Baksi, Tomas Gerlich, Sylvain Guilley, Peizhou Gan, Jakub Breier, Anupam Chattopadhyay, Ritu Ranjan Shrivastwa, Zdenek Martinasek, and Shivam Bhasin, "Hardware Implementation of ASCON," *Lightweight Cryptography Workshop 2023*. Available at https://csrc.nist.gov/csrc/media/Events/2023/lightweight-cryptography-workshop-2023/documents/accepted-papers/07-hardware-implementation-of-ascon.pdf.

[23] David Knichel, Pascal Sasdrich, and Amir Moradi, "Generic Hardware Private Circuits Towards Automated Generation of Composable Secure Gadgets," *IACR Transactions on Cryptographic Hardware and Embedded Systems (TCHES)*, Vol.2022, No.1, pages 323-344, 2022. Available at https://tches.iacr.org/index.php/TCHES/article/view/9299/8865.

[24] Paul C. Kocher, "Cryptanalysis of Diffie-Hellman, RSA, DSS, and Other Systems Using Timing Attacks," EXTENDED ABSTRACT, 1995. Available at https://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=466E62B26E868456707AE59A26CA7FFE?doi=10.1.1.397.192&rep=rep1&type=pdf.

[25] Paul Kocher, Joshua Jaffe, and Benjamin Jun, "Differential Power Analysis," In Neal Koblitz, editor, *Advances in Cryptology - CRYPTO*, volume 1666 of Lecture Notes in Computer Science, pages 388–397, Springer, 1996. Available at https://link.springer.com/content/pdf/10.1007/3-540-48405-1_25.pdf.

[26] Sinian Luo, Weibin Wu, Yanbin Li, Ruyun Zhang, and Zhe Liu, "An efficient soft

analytical side-channel attack on Ascon," In Lei Wang, Michael Segal, Jenhui Chen, and Tie Qiu, editors, *Wireless Algorithms, Systems, and Applications*, pages 389–400. Springer, 2022.

[27] Yang Li, Kazuo Sakiyama, Shigeto Gomisawa, Toshinori Fukunaga, Junko Takahashi, and Kazuo Ohta, "Fault Sensitivity Analysis," In Stefan Mangard and François-Xavier Standaert, editors, *Cryptographic Hardware and Embedded Systems (CHES)*, volume 6225 of Lecture Notes in Computer Science, pages 320-334, Springer, 2010. Available at `https://link.springer.com/content/pdf/10.1007/978-3-642-15031-9_22.pdf`.

[28] Zhenyuan Liu and Patrick Schaumont, "Root-cause Analysis of the Side Channel Leakage from ASCON Implementations," *Lightweight Cryptography Workshop 2023*. Available at `https://csrc.nist.gov/csrc/media/Events/2023/lightweight-cryptography-workshop-2023/documents/accepted-papers/13-root-cause-analysis-of-side-channel-leakage.pdf`.

[29] Kohei Matsuda, Tatsuya Fujii, Natsu Shoji, Takeshi Sugawara, Kazuo Sakiyama, Yuichi Hayashi, Makoto Nagata, and Noriyuki Miura, "A 286 F^2/Cell Distributed Bulk-Current Sensor and Secure Flush Code Eraser Against Laser Fault Injection Attack on Cryptographic Processor," *Journal of Solid-State Circuits*, Vol.53, No.11, pages 3174-3182, 2018. Available at `https://da.lib.kobe-u.ac.jp/da/kernel/90005512/90005512.pdf`.

[30] Kamyar Mohajerani, Luke Beckwith, Abubakr Abdulgadir, Eduardo Ferrufino, Jens-Peter Kaps, and Kris Gaj, "SCA Evaluation and Benchmarking of Finalists in the NIST Lightweight Cryptography Standardization Process," *IACR Cryptology ePrint Archive, Paper 2023/484*, 2023. Available at `https://eprint.iacr.org/2023/484.pdf`.

[31] NewAE, CW308 UFO, `https://rtfm.newae.com/Targets/CW308%20UFO/`.

[32] Svetla Nikova, Christian Rechberger, and Vincent Rijmen, "Threshold Implementations Against Side-Channel Attacks and Glitches," In Peng Ning, Sihan Qing, and Ninghui Li, editors, *International Conference and Communications Security (ICICS)*, volume 4307 of Lecture Notes in Computer Science, pages 529-545, Springer, 2006. Available at `https://link.springer.com/content/pdf/10.1007/11935308_38.pdf`.

[33] Svetla Nikova, Vincent Rijmen, and Martin Schläffer, "Secure Hardware Implementation of Nonlinear Functions in the Presence of Glitches,", *Journal of Cryptology*, Vol.24, No.2, pages 292-321, 2011. Available at `https://link.springer.com/content/pdf/10.1007/s00145-010-9085-7.pdf`.

[34] NIST: National Institute of Standards and Technology. `https://www.nist.gov`.

[35] NIST, "NIST IR 8454 Status Report on the Final Round of the NIST Lightweight

Cryptography Standardization Process," Available at https://nvlpubs.nist.gov/nistpubs/ir/2023/NIST.IR.8454.pdf

[36] Colun O' Flynn, "A Framework for Embedded Hardware Security Analysis," PhD thesis, Dalhousie University, 2017. Available at https://dalspace.library.dal.ca/bitstream/handle/10222/73002/OFlynn-Colin-PhD-ECED-June-2017.pdf.

[37] Keyvan Ramezanpour, Abubakr Abdulgadir, William Diehl, Jens-Peter Kaps, and Paul Ampadu, "Active and Passive Side-Channel Key Recovery Attacks on Ascon," *Lightweight Cryptography Workshop 2020*. Available at https://csrc.nist.gov/CSRC/media/Events/lightweight-cryptography-workshop-2020/documents/papers/active-passive-recovery-attacks-ascon-lwc2020.pdf.

[38] Aein Rezaei Shahmirzadi and Amir Moradi, "Re-consolidating first-order masking schemes nullifying fresh randomness," *IACR Transactions on Cryptographic Hardware and Embedded Systems (TCHES)*, Vol.2021, No.1, pages 305–342, 2021. Available at https://tches.iacr.org/index.php/TCHES/article/view/8736/8336.

[39] Riscure, Piñata (Training Target), https://www.riscure.com/products/pinata-training-target/.

[40] Pico Technology, PicoScope, https://www.pico-t.co.jp/product/picoscope/.

[41] 崎山 一男, "軽量暗号の実装性能に関する調査及び評価（NIST 軽量暗号コンペティションファイナリスト）," CRYPTREC 外部評価報告書, CRYPTREC EX-3205-2022, 2022. Available at https://www.cryptrec.go.jp/exreport/cryptrec-ex-3205-2022.pdf.

[42] Niels Samwel and Joan Daemen, "DPA on hardware implementations of Ascon and Keyak," In *Proceedings of the Computing Frontiers Conference*, pages 415–424, ACM, 2017.

[43] Kris Tiri and Ingrid Verbauwhede, "A Logic Level Design Methodology for a Secure DPA Resistant ASIC or FPGA Implementation," In Proceedings of *Design, Automation and Test in Europe Conference and Exposition (DATE*, pages 246-251, IEEE, 2004. Available at https://www.esat.kuleuven.be/cosic/publications/article-697.pdf.

[44] Rhys Weatherley, "Finalists to the NIST lightweight cryptography competition," *GitHub*, 2021. Available at https://github.com/rweather/lwc-finalists/tree/5d2b22c9ff7744be429cabda0c078ea5b7b6f79e.

[45] Shih-Chun You, Markus G. Kuhn, Sumanta Sarkar, and Feng Hao, "Low Trace-Count Template Attacks on 32-bit Implementations of ASCON AEAD," *IACR Transactions on Cryptographic Hardware and Embedded Systems (TCHES)*, Vol.2023, No.4, pages 344–366, 2023. Available at https://tches.iacr.org/index.php/TCHES/article/

view/11169/10608.

[46] Springer, Lecture Notes in Computer Science (LNCS). https://www.springer.com/gp/computer-science/lncs.

[47] Takeshi Sugawara, Yang Li, and Kazuo Sakiyama, "Probing attack of share-serial threshold implementation of advanced encryption standard," *IET Electronics Letters*, Vol.55, No.9, pages 517-519, 2019. Available at https://ietresearch.onlinelibrary.wiley.com/doi/epdf/10.1049/el.2018.7518.

[48] 藤堂 洋介, "軽量暗号の安全性に関する調査及び評価 (Ascon, Grain-128AEAD, Tiny-Jambu)," CRYPTREC 外部評価報告書, CRYPTREC EX-3203-2022, 2022. Available at https://www.cryptrec.go.jp/exreport/cryptrec-ex-3203-2022.pdf.

[49] TROCHE Co.,Ltd., SAKURA-G, SAKURA-X. http://www.troche.com/sakura/order.html.

[50] IEEE, IEEE Xplore. https://ieeexplore.ieee.org/Xplore/home.jsp.

資料３－４－別紙２

軽量暗号 Ascon などに関わる標準化動向調査

GMO サイバーセキュリティ by イエラエ株式会社

2023 年 9 月

エグゼクティブサマリー

本報告書では、NIST 軽量暗号コンペティションで 2023 年 2 月 7 日に選定された Ascon について標準化動向の調査を行った。Ascon の選定に関連する情報については、文献 [1]に詳しく記載されているため、この文献を中心に調査を行った。

Final Round では、最終候補として 10 のアルゴリズムが選択され、以下に示すような選考プロセスにおいて評価が行われた。

- 選考プロセスでのポイント
  - 様々な評価基準（安全性、ソフトウェアおよびハードウェアの性能、設計の成熟度、第三者による安全性評価の量、知的財産権の有無など）に異なる重み付けを割り当てて実施
  - 異なるセキュリティ要件、異なる機能性、異なる複雑性を持った攻撃などを踏まえた評価の実施
  - 限られたリソースにおける安全性評価および性能評価の実施

最終候補となったアルゴリズムの中から NIST が Ascon を選定したポイントについて整理を行うと以下の項目が挙げられる。

- 安全性
  - 高いセキュリティーマージン
  - 多数の第三者による安全性評価の数
- 設計/実装
  - 設計の微調整を行わないという設計の成熟度
  - 軽量暗号コンペティションである CAESAR プロジェクトにおいて軽量暗号の最終的なポートフォリオに選択されている実績
  - 漏えいに対するモードレベルでの保護メカニズムを有すること
  - 実装と設計の柔軟性
  - サイドチャネル攻撃に対する対策を行うための追加コストが低いこと
- 機能性
  - ハッシュに加えて XOF や MAC などの追加機能を有すること
- 性能
  - ソフトウェアおよびハードウェア環境において、現行の NIST 標準である AES-GCM や SHA-2 を上回る性能を有すること

また、NIST 以外の標準化団体における検討については、2023 年 9 月現在では大きな動きは見られなかったが、2023 年後半に予定されている NIST が発行する標準仕様の公開を受けて、本格的に様々な団体での検討が行われるものと考える。また、標準化された暗号技術が利用できる環境としてソフトウェアやハードウェア実装が公開されることが重要であるが、CAESAR プロジェクト等の実績などから実装がいくつか公開されているケースが見受けられた。これは Ascon の設計が成熟しており、NIST 軽量暗号コンペティションにおいて設計の微修正が行われていないことが背景にあると考える。

目次

1. はじめに

2017 年 3 月に公開された CRYPTREC 暗号技術ガイドライン（軽量暗号）（以下、「2016 年度ガイドライン」という） [2]では、「実装性能と安全性のトレードオフを勘案した上で、従来の暗号技術に対して特定の性能指標で優位性（軽量性）を持つように設計された共通鍵暗号技術」をスコープとし、軽量暗号の活用例、代表的な軽量暗号の性能比較、代表的な軽量暗号に関する基本情報について紹介している。しかしながら、暗号方式に対する安全性評価技術は日進月歩であり、2016 年度ガイドラインの公開から 5 年以上が経っているため、2016 年度ガイドラインには記載されていない。そのため、軽量暗号の安全性を脅かす新たな脅威が生じている可能性は十分に考えられる。そこで、2016 年度ガイドラインで紹介された暗号方式を中心とした代表的な軽量暗号の安全性評価に関する動向調査を行うことを目的とし、2021 年 9 月の時点における軽量暗号に対して現実的な脅威に繋がる脆弱性が指摘されているか否かを明らかにするための報告書として「CRYPTREC 暗号技術ガイドライン（軽量暗号）」掲載の暗号方式に関する安全性評価の動向調査 [3]が公開された。

また、2019 年度量子コンピュータ時代に向けた暗号の在り方検討タスクフォースにおいて、「CRYPTREC において、軽量暗号は CRYPTREC 暗号リストに組み込まず、別途ガイドラインという形で取り扱う」ことが決定された。この決定を踏まえて 2020 年度第 2 回暗号技術検討会において、2016 年度に作成した「CRYPTREC 暗号技術ガイドライン（軽量暗号）」について 2023 年度中を目処に更新することが承認された。そこで本報告書では、2020 年度第 2 回暗号技術検討会の承認内容を踏まえて、「NIST Lightweight コンペティション最終選考で採択された軽量暗号方式」や「軽量暗号として ISO/IEC 等で近年採録されたもしくは採録される予定の方式」に関する標準化動向について状況を整理した調査報告として、「軽量暗号の評価指標、標準化動向に関する調査（NIST 軽量暗号コンペティションファイナリストなど）」 [4]が報告された。しかしながら、調査報告は 2022 年 12 月であったため Lightweight Cryptography Project の結果を含めた内容にすることができなかった。Ascon が選出された。

本調査報告書では、前回の報告書の調査期間以降である 2023 年 2 月 7 日に選定された Ascon を中心とした標準化動向調査を行い、「CRYPTREC 暗号技術ガイドライン（軽量暗号）」の更新に向けた標準化動向調査結果を執筆する。

2. NIST 軽量暗号コンペティション

2016 年度ガイドラインの発行後、NIST*による軽量暗号コンペティション（以下、NIST 軽量暗号コンペティションとする。）[5]が開催された。前回の報告書 [4]の執筆時には「最終評価は 2022 年末に終了する予定である」と告げられていたが、2023 年 2 月 7 日に Ascon を選定したことがアナウンスされた。

なお、NIST 軽量暗号コンペティションの Web サイトでは、図 1 のような構成となっており、各 Round に関する情報や軽量暗号 Workshop に関する情報、制約のある環境下での実装性能など有益な情報へのリンクが整理されている。

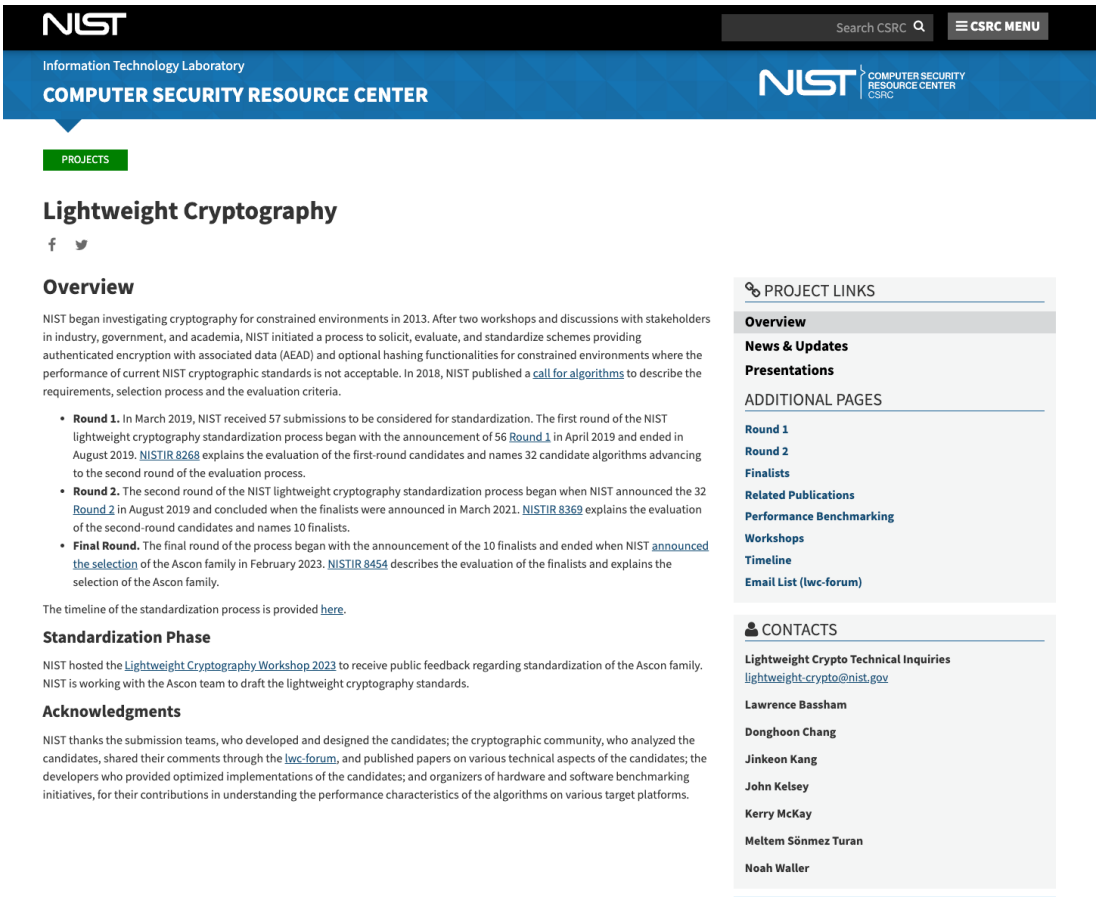


図 1　NIST 軽量暗号コンペティション

---

* NIST の正式名称は、National Institute of Standards and Technology であり、日本語では米国立標準技術研究所と呼ばれるアメリカの政府機関である。科学技術分野における計測と標準に関する研究が行われている。
URL： https://www.nist.gov/

以下に、2023 年 9 月現在の NIST 軽量暗号コンペティションにおける標準化動向に関する状況を整理する。NIST 軽量暗号コンペティションでは選定プロセスにおける Round 1 から Final Round の 3 回の選定が実施されている。それぞれの選定プロセスにおいて、どのような軽量暗号アルゴリズムが提案され、採択されたかについては表 1 を参照することで全体像を把握することができるように情報整理を行なっている。

	Round1	Round2	Final Round	Final Selection
#	候補アルゴリズム名	候補アルゴリズム名	候補アルゴリズム名	候補アルゴリズム名
1	ACE	ACE	ACE	ACE
2	ASCON	ASCON	ASCON	ASCON
3	Bleep64	Bleep64	Bleep64	Bleep64
4	CiliPadi	CiliPadi	CiliPadi	CiliPadi
5	CLAE	CLAE	CLAE	CLAE
6	CLX	CLX	CLX	CLX
7	COMET	COMET	COMET	COMET
8	DryGASCON	DryGASCON	DryGASCON	DryGASCON
9	Elephant	Elephant	Elephant	Elephant
10	ESTATE	ESTATE	ESTATE	ESTATE
11	FlexAEAD	FlexAEAD	FlexAEAD	FlexAEAD
12	ForkAE	ForkAE	ForkAE	ForkAE
13	Fountain	Fountain	Fountain	Fountain
14	GAGE and InGAGE	GAGE and InGAGE	GAGE and InGAGE	GAGE and InGAGE
15	GIFT-COFB	GIFT-COFB	GIFT-COFB	GIFT-COFB
16	Gimli	Gimli	Gimli	Gimli
17	Grain-128AEAD	Grain-128AEAD	Grain-128AEAD	Grain-128AEAD
18	HERN & HERON	HERN & HERON	HERN & HERON	HERN & HERON
19	HYENA	HyENA	HyENA	HyENA
20	ISAP	ISAP	ISAP	ISAP
21	KNOT	KNOT	KNOT	KNOT
22	LAEM	LAEM	LAEM	LAEM
23	Lilliput-AE	Lilliput-AE	Lilliput-AE	Lilliput-AE
24	Limdolen	Limdolen	Limdolen	Limdolen
25	LOTUS-AEAD and LOCUS-AEAD	LOTUS-AEAD and LOCUS-AEAD	LOTUS-AEAD and LOCUS-AEAD	LOTUS-AEAD and LOCUS-AEAD
26	mixFeed	mixFeed	mixFeed	mixFeed
27	ORANGE	ORANGE	ORANGE	ORANGE
28	Oribatida	Oribatida	Oribatida	Oribatida
29	PHOTON-Beetle	PHOTON-Beetle	PHOTON-Beetle	PHOTON-Beetle
30	Pyjamask	Pyjamask	Pyjamask	Pyjamask
31	Qameleon	Qameleon	Qameleon	Qameleon
32	Quartet	Quartet	Quartet	Quartet
33	REMUS	REMUS	REMUS	REMUS
34	Romulus	Romulus	Romulus	Romulus
35	SAEAES	SAEAES	SAEAES	SAEAES
36	Saturnin	Saturnin	Saturnin	Saturnin
37	Shamash & Shamashash	Shamash & Shamashash	Shamash & Shamashash	Shamash & Shamashash
38	SIMPLE	SIMPLE	SIMPLE	SIMPLE
39	SIV-Rijndael256	SIV-Rijndael256	SIV-Rijndael256	SIV-Rijndael256
40	SIV-TEM-PHOTON	SIV-TEM-PHOTON	SIV-TEM-PHOTON	SIV-TEM-PHOTON
41	SKINNY-AEAD/SKINNY-HASH	SKINNY-AEAD/SKINNY-HASH	SKINNY-AEAD/SKINNY-HASH	SKINNY-AEAD/SKINNY-HASH
42	SNEIK	SNEIK	SNEIK	SNEIK
43	SPARKLE (SCHWAEMM and ESCH)	SPARKLE (SCHWAEMM and ESCH)	SPARKLE (SCHWAEMM and ESCH)	SPARKLE (SCHWAEMM and ESCH)
44	SPIX	SPIX	SPIX	SPIX
45	SpoC	SpoC	SpoC	SpoC
46	Spook	Spook	Spook	Spook
47	Subterranean 2.0	Subterranean 2.0	Subterranean 2.0	Subterranean 2.0
48	SUNDAE-GIFT	SUNDAE-GIFT	SUNDAE-GIFT	SUNDAE-GIFT
49	Sycon	Sycon	Sycon	Sycon
50	Thank Goodness It's Friday (TGIF)	Thank Goodness It's Friday (TGIF)	Thank Goodness It's Friday (TGIF)	Thank Goodness It's Friday (TGIF)
51	TinyJambu	TinyJambu	TinyJambu	TinyJambu
52	Triad	Triad	Triad	Triad
53	TRIFLE	TRIFLE	TRIFLE	TRIFLE
54	WAGE	WAGE	WAGE	WAGE
55	Xoodyak	Xoodyak	Xoodyak	Xoodyak
56	Yarará and Coral	Yarará and Coral	Yarará and Coral	Yarará and Coral

表 1　NIST 軽量暗号コンペティション選定アルゴリズム（最終決定）

【Round 1】

2019 年 3 月に NIST は、NIST 軽量暗号コンペティションの Round 1 として 57 件の提出物を受け取り、“Submission Requirements and Evaluation Criteria for the Lightweight Cryptography Standardization Process”[6]で示した要件に基づき完全性と妥当性の観点から提出された軽量暗号アルゴリズムから Round 1 の候補アルゴリズムとして 56 個のアルゴリズム選定を 2019 年 4 月に行い、2019 年 8 月に Round 1 を終了した。

なお、Round 1 に関する詳細なステータスについては、“Status Report on the First Round of the NIST Lightweight Cryptography Standardization Process”[7]を参照してほしい。

なお、Round 1 で使用された評価基準は [7]において要約されているので、概要について整理を行う。この Round での評価基準として最重要なものは、「提出された暗号アルゴリズムの安全性」と言える。軽量暗号であることを評価するために制約のある環境下での実装特性（性能とコスト）も重要な基準となっていたことがわかる。また、実装での安全性の観点からは、サイドチャネル攻撃への対策に適しているかどうかについても評価されていた。

【Round2】

NIST 軽量暗号コンペティションの Round2 は、NIST が 2019 年 8 月に 32 個の候補アルゴリズムを発表し、2021 年 3 月に Finalist を公表したことで Round2 が終了した。なお、Round2 における選定に関する詳細なステータスについては、“Status Report on the Second Round of the NIST Lightweight Cryptography Standardization Process”[8]を参照してほしい。

[8]において Round2 で使用された評価基準が要約されているので、概要について整理を行う。この Round での評価基準は、前回の Round 1 と同様の評価観点である「第三者による分析や広く理解された設計原理と安全性証明に基づく要求」および「制約のあるデバイスを用いたアプリケーションにおける候補アルゴリズムの性能（制約のある環境における候補アルゴリズムのハードウェアおよびソフトウェアの性能)」というものであり、Round を経ることにより評価基準が詳細化されたという理解をした。なお、Round 1 と同様に候補アルゴリズムのサイドチャネル耐性についても評価基準となっていた。

【Final Round】

NIST 軽量暗号コンペティションの Final Round は、2021 年 3 月に Round2 での評価を踏まえて Ascon、Elephant、GIFT-COFB、Grain-128AEAD、ISAP、PHOTON-Beetle、Romulus、SPARKLE、TinyJAMBU、および Xoodyak の 10 個のアルゴリズムを選定し

た。当初は、NIST 軽量暗号コンペティションにおいて制約のある環境に適した AEAD とハッシュ機能として 1 つまたは複数の方式を選択するために標準化プロセスを開始したが、結果として 2023 年 2 月 7 日に Ascon ファミリーを選定したことを発表した。

なお、Final Round における選定に関する詳細なステータスについては、“Status Report on the Final Round of the NIST Lightweight Cryptography Standardization Process”[1]を参照してほしい。このドキュメントの目的は標準化プロセスとして Final Round の公開記録を提供し、選定された最終候補のアルゴリズムの評価について説明することである。

各 Round での候補アルゴリズムが選定されなかった理由については、NIST が公開している Status Report において示されているが、次の Round に進めなかった理由について概要を整理する。

【Round 1】

- 第三者による安全性に対する評価が公開されていないことや提出資料において、安全性の要求を裏付ける情報が不十分である提案については除外された。
- 第三者評価によって、Forgery Attacks、Length-extension Attacks や Distinguishing Attacks が存在する方式が整理された。
  - なお、指摘された懸念を払拭するために設計者が提案した修正は、評価時には考慮されなかったが、実装のバグによる実用的な攻撃（例えば、Forgery Attacks）は排除の理由とはされなかった。NIST の研究者は実装の更新をチェックし、元の仕様と整合性が取れているかを確認した。

【Round 2】

- Round 1 と同様に第三者による安全性評価が行われていることや安全性の要求を裏付ける情報が十分に情報公開されていること。
- 制約のあるデバイスを使用するアプリケーションにおける性能（制約のある環境におけるハードウェアおよびソフトウェアでの性能）がよいこと。
  - さまざまな性能とコストの指標で評価・比較され、現在の NIST 標準（特に AES-GCM [9]と SHA-2 [10]）より著しく性能がよいものが選定時に優遇されていた。
- 追加検討事項として、以下の項目について評価されている。
  - Side-Channel Resistance、Nonce-Misuse Security、RUP Security、Impacts of State Recovery および Post-Quantum Security

3. Ascon の選定に関する評価基準や評価観点

NIST 軽量暗号コンペティションの最終選考アルゴリズムとして Ascon が選定されたが、Final Round[†]における評価基準と選考プロセスについての概要を整理し、Ascon が選定された理由について調査結果を示す。詳細については、文献 [1]の「2. Evaluation Criteria and Selection Process」を参照することでより詳しく情報を得ることができる。

3.1. NIST 軽量暗号コンペティションにおける評価基準や評価観点

NIST 軽量暗号コンペティションにおける評価基準について、まとめると以下に示す 4 つが主な基準となっていると考えられる。また、重要度という観点から評価基準を見ると、最も重要な基準は「安全性」である。それに次ぐ重要な基準は「制約のある環境下におけるソフトウェアおよびハードウェアでの性能」であると考えられる。

- 暗号学的安全性
- 制約のある環境下におけるソフトウェアおよびハードウェアでの性能
- サイドチャネル攻撃や故障攻撃への耐性
- 知的財産

それぞれの評価基準について、具体例を挙げて詳しく解説を行う。

- 暗号学的安全性

  評価対象アルゴリズムにおける安全性は、提出された閲覧可能な自己による安全性解析結果、設計者による安全性に対する要求、安全性証明、広く閲覧可能な第三者による安全性評価などの情報を幅広く評価している。

  なお、明示的に提出が要求されていないが、Nonce-misuse シナリオや Releasing Unverified Plaintext (RUP) シナリオ、状態回復への影響、耐量子暗号としての安全性などが追加の考慮事項として挙げられている。

  なお、最終候補として選定されたアルゴリズムに対する安全性評価については、文献 [1]の「3. Finalists」に整理されている。

[†] 文献 [1]において、Final Round が Round 3 として明記されていることに注意されたい。

・ 制約のある環境下におけるソフトウェアおよびハードウェアでの性能

様々な性能やコストに関する測定基準において、Final Round に選定されたアルゴリズム同士や NIST 標準である AES-GCM [11] [12]（AEAD としての比較対象）と SHA-2 [13]（ハッシュ関数としての比較対象）との比較・評価が行われる。なお、現行アルゴリズムとして広く採用されている AES-GCM や SHA-2 に対しては大幅に優れた性能を発揮することが期待されている。

なお、最終候補として選定されたアルゴリズムの性能比較結果は、文献［1]の「4. Benchmarking Results」および「B. NIST Software Benchmarking Results」に整理されている。

・ サイドチャネル攻撃や故障攻撃への耐性

サイドチャネル攻撃への耐性を提供する必要はないと示されているが、簡単かつ低コストで実現できることが強く要望されている。

なお、最終候補として選定されたアルゴリズムのサイドチャネル攻撃や故障攻撃に関する結果は、文献［1]の「4.3. Resistance to Side-Channel and Fault Attacks」に整理されている。

・ 知的財産

知的財産について、特許請求の使用を必要とする可能性のあるアルゴリズムや実装に反対はしないが、技術的な理由によりこのアプローチが正当化される場合、評価プロセスでの選定を妨げる可能性のある要因であると示されている。

なお、知的財産に関する声明については、文献［1]の「2.2. Selection Process」に整理されている。

3.2. NIST 軽量暗号コンペティションにおける評価プロセス

最終候補を公正に評価し、標準化されたのちに長期にわたって利用されるアルゴリズムを選択することが困難な作業であることが示されており、困難な作業である理由として、以下の項目が挙げられている。

・ 最終候補の機能

- セキュリティの要求
- ベースとなる構成要素
- サポートされるパラメータサイズ
- 設計アプローチ
- バリエーションの数
- 利用可能な第三者による安全性評価の数
- 最適化された実装物の数

また、NIST 軽量暗号コンペティションの初期段階において、ターゲットアプリケーションに関して、一般からのフィードバックを踏まえて 2 つのプロファイル（図 2）を決定した。この部分が NIST 軽量暗号コンペティションにおいて 1 つまたは複数のアルゴリズムが選定される可能性が出た要因である。

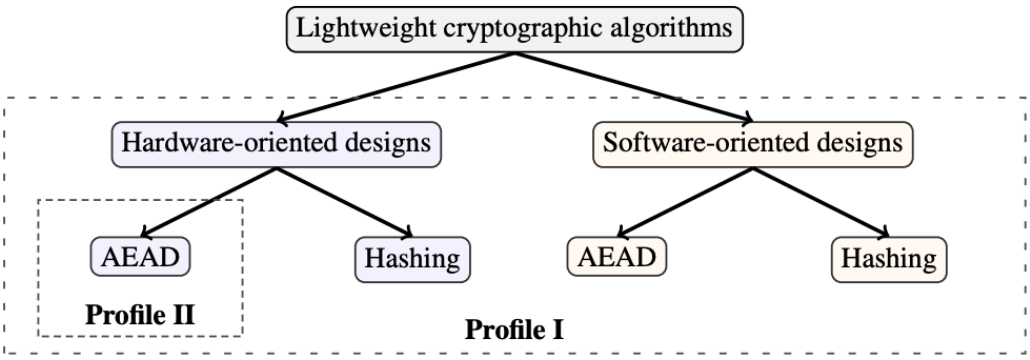


図 2 軽量暗号アプリケーションにおけるプロファイル

2 つのプロファイルが定められており、以下のとおりである。

- プロファイル 1
  - 制約のある環境でのソフトウェアとハードウェアのための AEAD およびハッシュ
- プロファイル 2
  - 制約のある環境でのハードウェアのための AEAD

最終候補に対する評価プロセスとして、第三者によるセキュリティ評価、バリエーション、設計の微調整、ベンチマーク、耐量子安全性、知的財産に関する声明の 6 つの観点から状況を整理し、まとめる。

・ 第三者によるセキュリティ評価

最終候補のアルゴリズムは、多くの第三者によるセキュリティ評価が行われた。それぞれの最終候補に対する評価結果は、文献 [1]の「3. Finalists」にまとめられている。また、単一鍵および Nonce-respecting において最終候補における安全性の要求を無効にするような評価はなく、ほとんどの候補は安全性のマージンがある状況であった。

・ バリエーション

AEAD とハッシュについては、異なる入出力サイズをサポートし、かつ／または異なるベースとなる構成ブロックを持つ、複数のバリアント（最大 10 個）の提出が許可されたが、NIST は公正な比較を行えるようにするため特定の入出力サイズを持つ AEAD とハッシュのバリエーションを各チームに求めた。この要望に対して、いくつかのチーム（Ascon、SPARKLE、Xoodyak など）には eXtendable Output Function (XOF)の亜種が含まれていたが、これらは正式なバリエーションとは考慮されなかったが、XOF 機能を提供できる柔軟性は選考過程において設計における有利な点としてみなされた。

・ 設計の微調整

Final Round の初期段階において、安全性や実装性能を向上させるための軽微な設計変更（NIST は以前に実施されたセキュリティ評価を無効にしない範囲を想定）が許可されたが、Ascon、GIFT-COFB、ISAP、PHOTON-Beetle および SPARKLE については、設計上の微調整は実施されなかった。一方、その他の候補については、性能向上や安全性改善のために設計が修正された。

・ ベンチマーク

NIST 標準である AES-GCM と SHA-2 よりも大幅に優れた性能を発揮することが期待されている。また、軽量暗号の重要な要素の 1 つとして、実装者が特定の用途に最適な実装を行うためのトレードオフ（コストと性能）を行えることである。

ソフトウェアでのベンチマーク結果として Ascon、GIFT-COFB、SPARKLE、TinyJAMBU および Xoodyak が、様々なプラットフォームで性能における優位性を示めした。詳細な情報については、文献 [1]の「4.1. Software Benchmarking」および「Appendix B. NIST Software Benchmarking Results」にまとめられている。また、ハードウェアでのベンチマーク結果として、Ascon、Xoodyak および TinyJAMBU が

最も優れた性能を示した。詳細な情報については、文献 [1]の「4.2. Hardware Benchmarking」にまとめられている。

また、サイドチャネル攻撃や故障攻撃への耐性やそのような攻撃を軽減させるための必要な実装オーバーヘッドについても評価が行われ、Ascon、ISAP、Xoodyak および TinyJAMBU はとても良い評価を示した。詳細な情報については、文献 [1]の「4.3.1. Protected Implementations and Side-Channel Security Evaluations」にまとめられている

- 耐量子安全性

軽量暗号の標準化プロセスにおける主要な関心事の一つではないが、量子コンピュータによる脅威に対する安全性の提供は長期利用の観点からも必要であるため、評価時には量子的な脅威に対するセキュリティも考慮された。一般的に共通鍵暗号関連の耐量子安全として、最も一般的な攻撃は Grover アルゴリズム [14]であり、網羅的な鍵探索(または、ハッシュ関数における衝突の発見)を 2 次関数的に高速化することが知られている。この攻撃を回避するためには、より大きな鍵サイズ（または、より大きなダイジェストサイズ）を持たせることになる。

これを踏まえると、3 つの候補が 128 ビットより長い鍵をサポートしている結果となった。特に SPARKLE ファミリーと TinyJAMBU ファミリーは 192 ビットと 256 ビットの鍵を持つ AEAD バリエーションを含み、また Ascon のバリエーションの 1 つが 160 ビットの鍵をサポートしている。

- 知的財産に関する声明

NIST は、最初のアルゴリズム提出時に、選択されたアルゴリズムに対して全世界でロイヤリティなしで利用できるようにするという目標が述べられていた。NIST は、アルゴリズム提出者に対し、候補アルゴリズムの実装によって侵害される可能性のある既知の知的財産をすべて特定するよう要求しており、結果として、最終候補の中で、該当する特許が特定されたのは PHOTON-Beetle のみという結果となった。しかしながら、この知的財産に関する事項は選考プロセスの決定には影響はなかったことが示されている。

3.3. Ascon に関する評価

NIST 軽量暗号コンペティションで行われた評価として、上記の基準に従って最終候補を評価した結果、NIST は Ascon ファミリーを標準化として選定した。Ascon ファミリーは、AEAD とハッシュ関数、そして追加された XOF を含むものとなっている。これにより、幅広いアプリケーションのニーズを満たすことができる。また、Permutation ベースの設計であるため追加機能を実装する際に追加コストが少なく済むことが期待されている。最終候補の中でも Ascon は、安全性という側面から見ると最終候補の中で最も成熟していると考えられている。理由としては、他の最終候補のいくつかは NIST 軽量暗号コンペティションの前から発表されていなかったが、Ascon ファミリーの AEAD バリエーションは CAESAR コンペティションの一環として発表され、安全性等について分析が行われていた。この CAESAR コンペティションでは、軽量認証暗号化を含む 3 つのプロファイルが作成されており、最終的に Ascon の AEAD バリエーションは、最終的な CAESAR ポートフォリオにおける軽量アプリケーションの主要な選択肢として選定された実績もある。Ascon の成熟度は、Final Round で行われた設計の微調整にも現れており、バリアントが追加されたが、Round 2 のバリエーションには設計の変更が行われていない状況であった。この事実を踏まえると、評価・分析で行われた攻撃に対処するために設計の微調整を行った他最終候補とは異なり、Ascon の高い成熟度を認識することができる事象と言える。

Ascon は公開されてから長い歴史があるため豊富な評価・分析が行われており、第三者による評価と実装が最も多いアルゴリズムであると言える。また、Ascon は暗号解析攻撃で先行しているにもかかわらず、高い安全性を維持している。さらに、Ascon ファミリーの AEAD バリエーションは、nonce-misuse resilience など AEAD におけるいくつかの高いセキュリティ機能を有する。

専用ハードウェアや組み込みシステムなど制約の多い環境での性能という基準は、最終選定の重要な要因となったと記されている。Ascon はソフトウェアおよびハードウェアで非常に優れた性能を発揮した。コストと性能の間のさまざまなトレードオフをサポートする実装の柔軟性を実証し、制約のあるリソース環境のさまざまなソフトウェアおよびハードウェアで、現在の NIST 標準である AEAD（AES-GCM）およびハッシュ（SHA-2）よりも優れた性能を示した。Ascon はまた、サイドチャネル攻撃等に対する対策が行われた保護された実装において、保護されていない実装よりも追加コストが低いことも示された。最終候補の 1 つである ISAP も、Ascon の Permutation に依存する AEAD のバリエーションを 2 つ持っていたが、最終的に Ascon よりも実装がより大きく、より遅くなるため実現性が低いと判断された。

軽量暗号の標準化プロセスで研究されたAsconのバリエーションの重要な制限事項の1つは、「256 ビット鍵のオプションがない」ことである。これは、量子アルゴリズムによる攻撃に対する 128 ビットのセキュリティが必要な場合に問題となる。しかしながら、この点に対して NIST は、この選定プロセスの主な目的を「軽量な AEAD とハッシュである」と強調している。もし仮に耐量子対策として 256 ビット鍵が必要な場合には、AES-GCM を使用することができると考えているようである。必要に応じて、より高い耐量子安全性を実現する追加のバリエーションの検討する可能性もあることが示唆されている。

なお、NIST の見解として、当面、Ascon ファミリーは制約のある環境下において十分なセキュリティを提供できると考えており、Ascon の性能はターゲット・デバイスやアプリケーションで許容されると予想されるため、現時点では第二候補のアルゴリズムは必要ないと判断しているとのことである。

4. 他標準化団体における軽量暗号 Ascon への検討状況

NIST 軽量暗号コンペティションにおいて、2023 年 2 月に Ascon が選定された結果を受けて、Ascon そのものや軽量暗号に関する採用に向けた検討が行われているかについて、NIST 以外の組織で標準化が行われているかを調査した。調査方法については以下のとおりである。

- 調査対象 標準化団体
  - IETF、W3C、ISO/IEC、ITU-T、Global Platform
- 調査方法
  - 調査手段：検索エンジン
  - 検索キーワード：Ascon、Light weight Crypto
  - 検索期間：2023 年 2 月 7 日 ～ 9 月 15 日

調査結果は以下のとおり[†]。

- IETF
  - Internet Draft “Secure UAS Network RID and C2 Transport” [15] の「5.3. Ciphers for Secure Transport」において、無人航空機で Ascon を選択するのが最善であることが示されている[‡]。その際には、ESP [16]や DTLS [17]の拡張が必要であるとも記述されている。
  - Internet Draft “Properties of AEAD algorithms” [18]の「4.4.2. Lightweight」において、NIST 軽量暗号コンペティションに関する参照が行われている。
  - IETF 117 で開催された TLS WG の発表である「New Post-Quantum Signatures on the Horizon」 [19]において、Ascon-Sign（SPHINCS+ with Ascon）が取り上げられていた。
- W3C
  - 調査した範囲では該当なし
- ISO/IEC
  - 調査した範囲では該当なし

[†] 2024 年 3 月 3 日現在、調査結果に関して追加情報が無いことを確認した。

[‡] Internet Draft 内では「NIST has selected a new lightweight cipher, Ascon, that may be the best choice for use on a UA. Work will be needed to develop full support for Ascon in both ESP and DTLS.」と記述されている。

- ITU-T
  - 調査した範囲では該当なし
- Global Platform
  - 調査した範囲では該当なし

以上のことから、標準化団体での検討状況については、大きな動きはあるように感じられなかった。しかしながら産業界において NIST 軽量暗号コンペティションの結果を受けて利用可能な環境を提供するような動向を把握したのでいくつか紹介する。NIST による最終的な標準化仕様が公開されることで、他標準化団体や産業界での活動が活性化されることが期待される。

- IP コア関連
  - Rambus 社 「Ascon-IP-41 暗号エンジン」 [20]
  - Xiphera 社 「XIP2201B: Ascon」 [21]
  - CAST 社 「Ascon-F」 [22]
- 暗号ライブラリ関連
  - Bouncy Castle 1.7.3 以降 [23]
  - CIRCL[§] [24]

[§] GitHub 上には記述されていないが、”https://pkg.go.dev/github.com/cloudflare/circl@v1.3.3/cipher/Ascon”には仕様が公開されている。

5. Ascon に関する考察

これまでの調査結果を踏まえて Ascon が選定されたことについて考察を行う。ポイントとなるのは、文献 [1]で示されている以下のような評価基準において、全体的に高い評価を得ることができている点であると考えられる。

- 暗号学的安全性
- 制約のある環境下におけるソフトウェアおよびハードウェアでの性能
- サイドチャネル攻撃や故障攻撃への耐性
- 知的財産

特に「暗号学的安全性」と「制約のある環境下におけるソフトウェアおよびハードウェアでの性能」が高く評価されているのではないかと考える。

5.1. 安全性

安全性については、Ascon のバリエーションが CAESAR コンペティションの最終ポートフォリオに含まれていることから、発表からの長い歴史があるため、第三者からのセキュリティ評価の数もかなり多い結果となっている。また、NIST 軽量暗号コンペティションにおいても Round 2 以降に設計を変更しないくらい設計が枯れていることも評価ポイントになっていたと考えられる。なお、文献 [1]の「3.1.2. Security Analysis」において、Ascon ファミリーに対するセキュリティ評価の概要がまとめられているので、具体的な内容を把握したい方は参照のこと。

5.2. 性能

制約のある環境下におけるソフトウェアおよびハードウェアでの性能については、ソフトウェアベンチマークおよびハードウェアベンチマークのそれぞれについて考察を行う。なお、文献 [1]の「4. Benchmarking Results」に注意点として、最終候補の特定の指標における最適化の可能性を完全に示すものでなく、すべての実装が同じ仮定や目標で設計されているわけではないことが示されている、さらに、注意点として、最終候補のより効率的な実装は実施可能であり、厳密な順位付けではないことに注意し一般的な指針として考慮することが示されている。

＜ソフトウェア・ベンチマーク＞

マイクロコントローラ上のソフトウェア性能は、最終候補の評価基準として重要である。評価実施については、複数の評価主体によって性能評価が行われた。評価主体とマイクロコントローラの環境については表 2 のとおりである。この評価では、メモリに制限のある 8 ビットマイコンから 32 ビットおよび 64 ビットマイコンまで幅広いターゲットプラットフォームを対象としている。

表 2 評価主体とベンチマークに使用したマイコンの仕様

Initiative	*Microcontroller*	*Processor*	*Word size*	*Clock speed*	*Flash*	*RAM*
NIST [253]	ATmega328P	AVR	8-bit	16 MHz	32 KB	2 KB
	ATmega4809	AVR	8-bit	16 MHz	48 KB	6 KB
	SAMD21G18A	ARM Cortex-M0+	32-bit	48 MHz	256 KB	32 KB
	nRF52840	ARM Cortex-M4	32-bit	64 MHz	1 MB	256 KB
	PIC32MX320F128H*	MIPS32 M4K	32-bit	80 MHz	128 KB	16 KB
	PIC32MX340F512H	MIPS32 M4K	32-bit	80 MHz	512 KB	32 KB
	ESP8266	Tensilica L106	32-bit	80 MHz	4 MB	80 KB
	AT91SAM3X8E	ARM Cortex-M3	32-bit	84 MHz	512 KB	96 KB
Renner et al. [254]	ATmega328P	AVR	8-bit	16 MHz	32 KB	2 KB
	STM32F103C8T6	ARM Cortex-M3	32-bit	72 MHz	64 KB	20 KB
	STM32F746ZG	ARM Cortex-M7	32-bit	216 MHz	1 MB	320 KB
	ESP32 WROOM	Tensilica Xtensa LX6	32-bit	240 MHz	4 MB	520 KB
	Kendryte K210	RISC-V (Dual Core)	64-bit	400 MHz	16 MB	8 MB
Weatherley [255]	ATmega2560	AVR	8-bit	16 MHz	256 KB	8 KB
	AT91SAM3X8E	ARM Cortex-M3	32-bit	84 MHz	512 KB	96 KB
	ESP32	Tensilica Xtensa LX6	32-bit	240 MHz	4 MB	520 KB

*PIC32MX340F512H microcontroller used with PlatformIO's PIC32MX320F128H board profile

以下に、NIST と Renner らによる評価結果の概要を示す。

- NIST による評価結果の概要
  - 評価環境での性能と PlatformIO でのコンパイルに成功した時に使用されたフラッシュサイズ（単位: バイト）で評価された。
  - AEAD 機能のサイズにおいて、Ascon は一貫してトップ・パフォーマーであり、AES-GCM よりもコンパクトな実装を実現した。
  - ハッシュ機能のサイズにおいて、Ascon はすべてのプラットフォームで一貫して SHA-256 より小さかったが、すべての環境で最速だったのは SHA-256 であった。
- Renner らによる評価結果の概要

- マイクロコントローラ上の AEAD アルゴリズムの性能を評価するためのベンチマークフレームワークを開発しており、この評価では、実行時間（マイクロ秒、テストベクタの平均生成時間)、コンパイル済みバイナリのサイズおよび RAM 使用量（STM32F7 のみ）を得ることができる。
- Arduino Uno の環境において、SPARKLE 、GIFT-COFB 、Xoodyak がトップ 3 であり、Ascon はかなり近接するような速度であったことから、この環境ではトップ集団でないことがわかる。ただし、コードサイズの観点では Ascon は最小サイズに達していると報告されている。

以上のことから、ソフトウェア・ベンチマークにおいて、Ascon は全体的にすべての環境で実行速度およびコードサイズにおけるトップ・パフォーマーであることが選定の決め手になったと考えられる。

＜ハードウェア・ベンチマーク＞

ハードウェア・ベンチマークにおいて、Round 2 で実施されたサイドチャネル攻撃対策が行われていない実装の性能評価結果の多くは Final Round に流用可能であることが示されていた。特に NIST は、ジョージ・メイソン大学（GMU）の暗号研究グループである CERG の評価結果に注目した。また、GMU はサイドチャネル攻撃等の対策を行った実装評価を行うには膨大な時間と専門知識が必要であり、単一グループが単独で行うことが困難であることを踏まえて、複数グループのリソースと専門知識を集結させるなどの貢献を行なっている。

GMU チームによる評価は、ベンチマーク環境として Xilinx 社 Artix-7 プラットフォームを使用した。保護された実装と保護されていない実装の両方が評価された。保護されていない実装と保護された実装のスループットや面積、あるいはマスキングに必要なランダム・ビット数などの性能比較により、保護されていない実装に保護手法を適用するコストに関する知見が得られたとされている。その評価結果として、非保護の AEAD 実装において Ascon は AES-GCM よりも平文を高速に処理した。サイズについて Ascon はトップ集団と比較するとよい結果は出ていない。なお、スループットから評価すると Ascon は最もスループットが高い結果となった。また、ハッシュ処理では、Ascon の非保護実装が最も高いスループットを示したが、サイズについて Ascon はトップ集団と比較するとよい結果は出ていない。

以上のことから、ハードウェア・ベンチマークにおいて、Ascon は全体的にすべての環境で実行速度においてはトップ・パフォーマーであるが、サイズの観点からは Ascon は最小

の実装を実現できていないが、GMU 以外の評価主体の評価によりエネルギー効率がよいことが報告されていることから選定の決め手になったと考えられる。

上記で考察した結果や文献 [1]で議論されている内容を踏まえると、Ascon は NIST 軽量暗号コンペティションの評価基準において高い評価結果を示していることから順当な判断によって選定されたと考える。

5.3. 標準化

NIST 軽量暗号コンペティション終了後である 2023 年 6 月 21～22 日にオンライン開催であったが、NIST 主催による 6th Lightweight Cryptography Workshop[**]が開催され、選定プロセスや軽量暗号の標準化に関する様々な側面について議論が行われた。18 個の発表が行われたが、その中から NIST 所属の Meltem Sönmez Turan 氏によって発表された「Evaluation of the Finalists and the Selection of Ascon」 [25]において、Ascon の標準化ドキュメントが公開されるタイミングに関する情報について言及されているためである。その標準化ドキュメントの公開時期として 2023 年後半（図 3）と共有されている。本報告書の 2023 年 9 月現在、標準ドキュメントの草案は公開されていない。

図 3 資料" Evaluation of the Finalists and the Selection of Ascon"

[**] https://csrc.nist.gov/Events/2023/lightweight-cryptography-workshop-2023

6. まとめ

本報告書では、NIST 軽量暗号コンペティションで 2023 年 2 月 7 日に選定された Ascon について標準化動向の調査を行った。Ascon の選定に関連する情報については、文献 [1]に詳しく記載されているため、この文献を中心に調査を行った。

Final Round では、最終候補として 10 のアルゴリズムが選択され、以下に示すような選考プロセスにおいて評価が行われた。

- 選考プロセスでのポイント
  - 様々な評価基準（安全性、ソフトウェアおよびハードウェアの性能、設計の成熟度、第三者による安全性評価の量、知的財産権の有無など）に異なる重み付けを割り当てて実施
  - 異なるセキュリティ要件、異なる機能性、異なる複雑性を持った攻撃などを踏まえた評価の実施
  - 限られたリソースにおける安全性評価および性能評価の実施

最終候補となったアルゴリズムの中から NIST が Ascon を選定したポイントについて整理を行うと以下の項目が挙げられる。

- 安全性
  - 高いセキュリティーマージン
  - 多数の第三者による安全性評価の数
- 設計/実装
  - 設計の微調整を行わないという設計の成熟度
  - 軽量暗号コンペティションである CAESAR プロジェクトにおいて軽量暗号の最終的なポートフォリオに選択されている実績
  - 漏えいに対するモードレベルでの保護メカニズムを有すること
  - 実装と設計の柔軟性
  - サイドチャネル攻撃に対する対策を行うための追加コストが低いこと
- 機能性
  - ハッシュに加えて XOF や MAC などの追加機能を有すること
- 性能
  - ソフトウェアおよびハードウェア環境において、現行の NIST 標準である AES-GCM や SHA-2 を上回る性能を有すること

また、NIST 以外の標準化団体における検討については、2023 年 9 月現在では大きな動き

は見られなかったが、2023 年後半に予定されている NIST が発行する標準仕様の公開を受けて、本格的に様々な団体での検討が行われるものと考える。また、標準化された暗号技術が利用できる環境としてソフトウェアやハードウェア実装が公開されることが重要であるが、CAESAR プロジェクト等の実績などから実装がいくつか公開されているケースが見受けられた。これは Ascon の設計が成熟しており、NIST 軽量暗号コンペティションにおいて設計の微修正が行われていないことが背景にあると考える。

参考文献

[1] NIST, “Status Report on the Final Round of the NIST Lightweight Cryptography Standardization Process:,” NIST, 2023.

[2] CRYPTREC 軽量暗号ワーキンググループ, “CRYPTREC 暗号技術ガイドライン（軽量暗号),” 3 2017.. Available: https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf.

[3] 伊藤竜馬, “「CRYPTREC 暗号技術ガイドライン（軽量暗号)」掲載の暗号方式に関する安全性評価の動向調査,” 2021. . Available: https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf.

[4] GMO サイバーセキュリティ by イエラエ株式会社, “軽量暗号の評価指標、標準化動向に関する調査（NIST 軽量暗号コンペティションファイナリストなど）,” 17 4 2023. . Available: https://www.cryptrec.go.jp/exreport/cryptrec-ex-3206-2022.pdf.

[5] National Institute of Standards and Technology, “Lightweight Cryptography,” . Available: https://csrc.nist.gov/projects/lightweight-cryptography.

[6] NIST, “Submission Requirements and Evaluation Criteria for the Lightweight Cryptography Standardization Process,” NIST.

[7] K. M. Ç. Ç. D. C. ,. L. B. Meltem Sönmez Turan, “Status Report on the First Round of the NIST Lightweight Cryptography Standardization Process,” 10 2019. . Available: https://csrc.nist.gov/publications/detail/nistir/8268/final.

[8] NIST, “Status Report on the Second Round of the NIST Lightweight Cryptography Standardization Process,” NIST, 2021.

[9] M. Dworkin, “NIST Special Publication 800-38D: Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC,” 11 2007. . Available: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38d.pdf.

[10] National Institute of Standards and Technology, "FIPS PUB 180-4 Secure Hash Standard (SHS)," 2015. . Available: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf.

[11] U.S. Department of Commerce , "Advanced Encryption Standard (AES)," 2001. . Available: https://doi.org/10.6028/NIST.FIPS.197- upd1.

[12] M. Dworkin, "Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC," 11 2007. . Available: https://doi.org/10.6028/NIST.SP.800-38D.

[13] U.S. Department of Commerce , "Secure Hash Standard (SHS)," 8 2015. . Available: https://doi.org/10.6028/NIST.FIPS.180-4.

[14] L. K. Grover, "A fast quantum mechanical algorithm for database search," 1996.

[15] R. Moskowitz, "Secure UAS Network RID and C2 Transport," 3 2023. . Available: https://datatracker.ietf.org/doc/draft-moskowitz-drip-secure-nrid-c2/.

[16] P. Jokela, "Using the Encapsulating Security Payload (ESP) Transport Format with the Host Identity Protocol (HIP)," 4 2015. . Available: https://www.rfc-editor.org/rfc/rfc7402.html.

[17] E. Rescorla, "The Datagram Transport Layer Security (DTLS) Protocol Version 1.3," 4 2022. . Available: https://www.rfc-editor.org/rfc/rfc9147.html.

[18] A. Bozhko, "Properties of AEAD algorithms," 3 2023. . Available: https://datatracker.ietf.org/doc/draft-irtf-cfrg-aead-properties/.

[19] T. W. Bas Westerbaan, "New Post-Quantum Signatures on the Horizon," 7 2023. . Available: https://datatracker.ietf.org/meeting/117/materials/slides-117-tls-new-post-quantum-signature-algorithms-on-the-horizon-00.

[20] Rambus, "Rambus IP Solution Supports New NIST Lightweight Cryptography Algorithm," 22 2 2023. . Available: https://www.rambus.com/blogs/rambus-ip-solution-supports-new-nist-lightweight-cryptography-algorithm/.

[21] Xiphera, “XIP2201B: ASCON, A Lightweight Cryptographic Suite for AEAD and Hashing,” 1 8 2023. . Available: https://xiphera.com/products/pdf/XIP2201B_PB.pdf.

[22] CAST, “ASCON-F, ASCON Authenticated Encryption & Hashing Engine,” 11 9 2023. . Available: https://www.cast-inc.com/security/encryption-primitives/Ascon-f.

[23] Bouncy Castle, “The Legion of the Bouncy Castle,” 8 4 2023. . Available: https://www.bouncycastle.org/releasenotes.html#r1rv73.

[24] Cloudflare, “CIRCL (Cloudflare Interoperable, Reusable Cryptographic Library),” . Available: https://github.com/cloudflare/circl.

[25] Meltem Sönmez Turan - NIST, “Evaluation of the Finalists and the Selection of Ascon,” 21 6 2023. . Available: https://csrc.nist.gov/Presentations/2023/evaluation-of-the-finalists-and-the-selection.

[26] “CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness,” . Available: https://competitions.cr.yp.to/caesar.html.

[27] “FELICS - Fair Evaluation of Lightweight Cryptographic Systems,” . Available: https://www.cryptolux.org/index.php/FELICS.

[28] “CAESAR submissions,” . Available: https://competitions.cr.yp.to/caesar-submissions.html.

[29] T. A. &. A. Luykx, An Account of the ISO/IEC Standardization of the Simon and Speck Block Cipher Families, Springer.

資料３―４

軽量暗号に関する技術動向調査（外部評価）

1. 背景

(1) 2020 年度第 2 回暗号技術検討会において、2016 年度に作成した「CRYPTREC 暗号技術ガイドライン（軽量暗号)」（文書番号：CRYPTREC GL-2003-2016）（以下、「2016 年度版ガイドライン」という）を 2023 年度中に更新することが承認された。

(2) 2021 年度第 2 回暗号技術評価委員会において、「CRYPTREC 暗号技術ガイドライン（軽量暗号)」2023 年度版（以下、「2023 年度版ガイドライン」という）の作成に向けた更新方針が承認された。

(3) 2022 年度の外部評価として、軽量暗号の安全性と実装性能に関する調査及び評価、並びに軽量暗号の標準化動向に関する調査を実施した。

(4) NIST 軽量暗号標準化プロセス（以下、「NIST LWC」という）では、2023 年 2 月に最終選考結果が発表され、ASCON が選定された。

2. 外部評価に関する実施事項（概要報告）

(1) 2023 年度第 1 回暗号技術評価委員会において、ASCON に焦点を当てた実装性能に関する評価及び調査、並びに ASCON を選出するに至った選考過程や選考理由に関する調査を外部評価によって実施することが承認された。

(2) 軽量暗号 ASCON の実装性能に関する調査及び評価

崎山 一男 様（電気通信大学）に外部評価を依頼した。選出理由と依頼内容は次のとおり。

ア　選出理由

近年の軽量暗号に関する技術動向に対して広い知見をお持ちであり、実装性能評価（ハードウェア及びソフトウェア）に関する多くの研究業績をお持ちであるため。

イ　依頼内容

軽量暗号 ASCON に関する実装性能評価について、物理攻撃[1]耐性を持つ実装性能評価も含め、公開されている評価結果を調査し、評価結果についてまとめ、考察などを行い、評価報告書を作成する。

(3) 軽量暗号 ASCON などに関わる標準化動向調査

菅野 哲 様（GMO サイバーセキュリティ by イエラエ株式会社）に外部評価を依頼した。選出理由と依頼内容は次のとおり。

ア　選出理由

暗号技術の標準化動向を含む調査・評価報告書執筆の多くの実績があり、近年の軽量暗号の技術動向調査に関する実績もお持ちであるため。

イ　依頼内容

NIST LWC の公募により、ASCON が選出されるに至った選定指標や評価の観点をまとめるとともに、軽量暗号に関する標準化動向について、公開情報を基にまとめ、考察などを行い、報告書を作成する。

---

[1] サイドチャネル攻撃等を含めた物理的な攻撃という意味で、本資料では物理攻撃という用語を使用する。

3. 外部評価報告書の概要

(1) 軽量暗号 ASCON の実装性能に関する調査及び評価

外部評価報告書の概要は以下のとおり。詳細は資料３－４－別紙１のとおり。

ア ASCON の実装面における特徴

ASCON は認証暗号モードとハッシュモードに対応する軽量な暗号アルゴリズムであり、

- 5 ビットのコンパクトな S-box を繰り返し使用
- ラウンド処理を並列実装可能
- 概ね全ての処理を同じラウンド処理の繰り返しで実現可能

という理由から、実装コストと処理性能のトレードオフの観点で高い柔軟性があることが特徴的である。例えば、コンパクトな S-box を同時に処理することで、実装コストを犠牲にして高い処理性能を得ることができる一方、異なる時間に S-box を再利用することで、処理性能を犠牲にして実装コストを下げることができる。

イ 物理攻撃対策を施した ASCON の実装性能

代表的な物理攻撃対策技術として、Threshold Implementation (TI)[2] とその発展的技術である Domain Oriented Masking (DOM)[3] がある。本報告書では、これらの物理攻撃対策を施した ASCON の実装評価に関する調査結果をまとめている。調査対象は次の表のとおり。

調査対象	著者	会議等
TI による物理攻撃対策を施した ASCON	Kandi ら	NIST LWC workshop 2023
DOM による物理攻撃対策を施した ASCON	Gross	学位論文（2018 年 6 月）

ウ ASCON に対する物理攻撃耐性評価

代表的な物理攻撃耐性評価技術として、相関電力解析（CPA）[4]、Test Vector Leakage Assessment (TVLA)[5]、テンプレート攻撃（TA）[6]、などがある。本報告書では、これらの評価技術を使用した ASCON の物理攻撃耐性評価に関する調査結果をまとめている。調査対象は次の表のとおり。

評価技術	調査対象 (対策の有無)	著者	会議等
CPA	両方	Niels ら	ACM CF 2017
TVLA, CPA	両方	Batina ら	Radbound 大学リポジトリ
TVLA, CPA	有	Mohajerani ら	NIST LWC workshop 2023
TVLA	有	Gigerl ら	NIST LWC workshop 2023
CPA	無	Liu ら	NIST LWC workshop 2023
TA	両方	You ら	TCHSE 2023

(2) 軽量暗号 ASCON などに関わる標準化動向調査

外部評価報告書の概要は以下のとおり。詳細は資料３－４－別紙２のとおり。

ア 最終ラウンドにおける評価基準と選定プロセス

---

[2] 秘密分散法に基づくマスキング手法

[3] d次プロービングモデルに対して耐性のあるマスキング手法

[4] 電力のサイドチャネル情報を効率よく解析する手法

[5] サイドチャネルからの漏洩評価における統計的手法

[6] 事前に攻撃対象モジュールの特性を評価したテンプレートを準備し、このテンプレートを使用してパラメータを操作できない攻撃対象モジュールの秘密鍵を推定する手法

2023年2月7日にNIST LWC最終ラウンドの評価フェーズが終了し、最終選考方式としてASCONが選出された。その後、NISTはステータスレポートNISTIR 8454[7]を発行し、最終ラウンドにおける評価基準や選定プロセスについて明らかにした。評価基準と選定プロセスの対応関係は次の表のとおり。

評価基準	選定プロセス
暗号学的安全性	第三者による安全性評価、耐量子安全性
制約のある環境下におけるソフトウェア及びハードウェアでの実装性能	ベンチマーク
サイドチャネル攻撃	ベンチマーク
知的財産	知的財産に関する声明
その他	バリエーション、設計の微調整

イ　ASCONが選出されるに至った選定指標や評価の観点

最終ラウンドにおける評価基準と選定プロセスに従い、標準化対象の暗号アルゴリズムとしてASCONが選出された。具体的には、次の表に示す観点でASCONが評価されている。

選定指標	評価
機能	● 認証暗号モードとハッシュモードに加え、XOF[8]機能を含む。 ● 暗号学的置換ベースの設計により、追加機能の実装コストが少ない。
成熟度	● CAESARコンペティションのユースケース1（軽量アプリケーション） ● 最終ラウンドにおける設計の微調整なし
安全性	● 第三者による安全性評価が最も多いファイナリスト ● 他ファイナリストよりも先行的に安全性評価が行われているにも関わらず依然として高い安全性を維持
実装性能	● ソフトウェアとハードウェアの両面で非常に優れた性能を発揮 ● 実装コストと処理性能の様々なトレードオフをサポートする柔軟性 ● 物理攻撃対策にかかる追加コストが低い。

ウ　軽量暗号に関する標準化動向

ASCONに関するNIST以外の組織での標準化動向についてまとめる。調査対象の標準化団体は、IETF[9]、W3C[10]、ISO/IEC[11]、ITU-T[12]、Global Platformの5団体である。2023年9月現在、IETFを除く4団体においてASCONに関する標準化動向は確認できなかった。IETFでは以下でASCONが取り上げられている。

- インターネットドラフト“Secure UAS Network RID and C2 Transport”
- インターネットドラフト“Properties of AEAD algorithms”
- IETF 117におけるTLS WGでの発表“New Post-Quantum Signatures on the Horizon”

✧　その他、産業界でもASCONを利用可能な環境を提供するような動向がある。

[7] https://csrc.nist.gov/pubs/ir/8454/final
[8] eXtendable Output Function：可変長出力関数
[9] Internet Engineering Task Force
[10] World Wide Web Consortium
[11] International Organization for Standardization/International Electrotechnical Commission
[12] International Telecommunication Union Telecommunication Standardization Sector

4.　今後の予定（外部評価報告書発行に向けた手順）

作成した 2 件の外部評価報告書を資料３－４－別紙１、資料３－４－別紙２として配布。今年度の調査対象である ASCON の実装性能及び標準化動向に関する技術動向調査として十分な内容を含んでいると考えられる。

2024 年 4 月上旬に本報告書を CRYPTREC の技術調査報告書として公開することを承認していただきたい。

以上

資料３－５－別紙

CRYPTREC 暗号技術ガイドライン（軽量暗号）2023 年度版

CRYPTREC 暗号技術評価委員会

2024 年 3 月 13 日

CRYPTREC 軽量暗号 WG 委員構成（2013 年度〜2016 年度）

主査	本間　尚文	国立大学法人東北大学
委員	青木　和麻呂	日本電信電話株式会社
委員	岩田　哲	国立大学法人名古屋大学
委員	小川　一人	日本放送協会
委員	小熊　寿	株式会社トヨタ IT 開発センター
委員	崎山　一男	国立大学法人電気通信大学
委員	渋谷　香士	ソニーグローバルマニュファクチャリング＆オペレーションズ株式会社
委員	鈴木　大輔	三菱電機株式会社
委員	成吉　雄一郎	ルネサスエレクトロニクス株式会社
委員	峯松　一彦	日本電気株式会社
委員	三宅　秀享	株式会社東芝
委員	渡辺　大	株式会社日立製作所

ガイドライン改定のための外部評価依頼先（2021 年度〜2023 年度）

井上　明子	日本電気株式会社
伊藤　竜馬	国立研究開発法人情報通信研究機構
岩田　哲	国立大学法人東海国立大学機構 名古屋大学
菅野　哲	GMO サイバーセキュリティ by イエラエ株式会社
崎山　一男	国立大学法人電気通信大学
藤堂　洋介	日本電信電話株式会社
内藤　祐介	三菱電機株式会社
本間　尚文	国立大学法人東北大学
峯松　一彦	日本電気株式会社

改定履歴

「CRYPTREC 暗号技術ガイドライン（軽量暗号）」（以下、「2016 年度版ガイドライン」という）

- 新規発行 – 2017 年 6 月 30 日（第 1 版）

「CRYPTREC 暗号技術ガイドライン（軽量暗号）2023 年度版」（以下、「2023 年度版ガイドライン」という）

- 追加
  - 2.2 節：軽量暗号の標準化動向（CAESAR コンペティション、NIST LWC プロジェクト）を追加
  - 3.3 節：NIST LWC 最終選考方式 Ascon の実装性能に関する評価結果を追加
  - 4.1 節：軽量ブロック暗号 LEA に関する情報を追加
  - 4.3 節：軽量ハッシュ関数 Lesamnta-LW に関する情報を追加
  - 4.4 節：軽量メッセージ認証コード Chaskey、LightMAC、Tsudik's keymode に関する情報を追加
  - 4.5 節：軽量認証暗号 Grain-128A に関する情報を追加
  - 付録 A：Ascon に対するサイドチャネル攻撃対策手法と物理攻撃手法に関する情報を追加
  - 付録 B：CAESAR final portfolio である AEGIS、COLM に関する情報を追加
  - 付録 C：NIST LWC ファイナリスト（Ascon を除く 9 方式）に関する情報を追加
- 改定
  - 2.1 節：本ガイドラインで紹介する代表的な軽量暗号技術（表 2.1）を改定
  - 4 章：2016 年度版ガイドライン掲載の代表的な軽量暗号に関する安全性解析状況と標準化状況を改定
  - 4.5 節：軽量認証暗号 Ascon に関する情報を改定

目次

第 1 章

はじめに

限られた実装環境でも安全で高性能な暗号技術を搭載したいというニーズは古くからあり、このニーズに応えて小型で高速な暗号技術の研究開発が進められてきた。昨今は、IoT (Internet of Things) の発展により、センサーやアクチュエータ等の計算リソースの限られたデバイスがネットワークに接続され、セキュリティやプライバシー上の脅威が高まっていることから、暗号技術に対してより多様な実装要件が求められている。

計算リソースの限られたデバイスにも実装可能な「軽量暗号」は、車載機器や医療機器をはじめとする様々な用途での利用が期待されているが、どの軽量暗号を選べばいいのか、運用時にどのようなことに注意すればいいのか、など実際に利用する際には専門家以外では判断が困難な場合も多い。

CRYPTREC では、主として電子政府で利用する暗号技術について検討を行っているが、それに加えて、今後さまざまな領域で利用が想定される暗号技術について技術調査を行い、社会に役立つ形で情報提供を行うことを目指している。特に、軽量暗号技術が求められる製品やサービスにおいて、利用者が最適な暗号方式を選択でき、容易に調達できることを目指し、2013 年度より CRYPTREC 暗号技術評価委員会の下に軽量暗号ワーキンググループ（WG）が設置された。2016 年度版ガイドライン [1, 2] は、軽量暗号の方式を選択・利用する際の技術的判断に資すること、今後の利用促進をはかることを目的として、軽量暗号 WG が作成したものである。その後、軽量暗号に関する研究開発の最新動向や米国 NIST (National Institute of Standards and Technology) 等の標準化動向を踏まえ、2021 年度から 2023 年度にかけて軽量暗号の安全性、実装性能、標準化動向に関する技術動向調査・評価を国内有識者に依頼し、これらの調査結果・評価結果を外部評価報告書として CRYPTREC ホームページに公開した [3, 4, 5, 6, 7, 8, 9, 10, 11]。2023 年度版ガイドラインは、これらの外部評価報告書に基づいて CRYPTREC 事務局が執筆・編集し、2016 年度版ガイドラインの改定版として公開するものである。本ガイドラインの主たる読者として、情報システムのセキュリティ機能の設計・開発・実装において暗号技術を活用する技術者を想定しているが、軽量暗号技術に興味をもつ方に広く読んで頂ければ幸いである。

1 章は、本ガイドラインの総説である。2 章では、軽量暗号を概説する。本ガイドラインの対象となる軽量暗号技術の概要、軽量暗号の標準化動向、そして軽量暗号の活用例を示し、その上で、軽量暗号を実際に活用する際の手引きを示している。特に、軽量暗号の特徴、代表的なユースケース、方式、パラメータの選択方法、使用時の留意点などを記載している。3 章では、代表的な軽量暗号の実装性能を示す。多くの軽量暗号方式が提案されているブロック暗号と認証暗号の技術分野において、代表的ないくつかの方式を取り上げ、これらの方式の実装性能を比較している。性能比較は、ハードウェア実装とソフトウェア実装で、それぞれ同一の実装環境下で行っている。ハードウェア実装では回路規模、消費電力、レイテンシの比較結果を、ソフトウェア実装では必要なメモリサイズの比較結果を示している。加えて、NIST が主催する軽量暗号標準化プロジェクトで最終選考方式に選定された Ascon の実装性能を示す。特に、サイドチャネル攻撃等への対策を施さない場合と対策を施した場合におけるソフトウェア実装とハードウェア実装の性能、そしてサイドチャネル攻撃耐性の評価結果について、公開されている情報に基づき記載している。4 章では、代表的な軽量暗号技術の基本情報をブロック暗号、ストリーム暗号、ハッシュ関数、メッセージ認証コード、認証暗号の技術分野別に示す。

本ガイドラインで紹介している軽量暗号技術は、執筆時点までに主要国際学会で発表されており、有力な攻撃法が発見されておらず、かつ十分な実装性能を持つと考えられる方式を選んでいる。その上で、可能な限り最新の情報に基づき安全性や実装性能、標準化動向等を紹介している。しかしながら、軽量暗号の研究開発は今まさに盛んに行われており、年々新たな方式や評価結果が出ていることから、記載内容が執筆時点（特に断りがない限り、2023 年 9 月現在）のものであることに

留意いただきたい。

2016 年度版ガイドライン策定に関する謝辞

2016 年度版ガイドラインは以下に示す軽量暗号 WG 委員および CRYPTREC 事務局で執筆・編集を行いました。所属は 2016 年 10 月時点のものです。また、2016 年度版ガイドラインを執筆する上で、軽量暗号の応用に関して株式会社日立製作所の大和田徹氏に、軽量暗号の実装評価に関して三菱電機株式会社の松井充氏、菅原健氏、村上ユミコ氏、梨本翔永氏に、それぞれ多大なご貢献をいただきました。この場を借りて皆様に深く感謝申し上げます。

主査	本間　尚文	国立大学法人東北大学
委員	青木　和麻呂	日本電信電話株式会社
委員	岩田　哲	国立大学法人名古屋大学
委員	小川　一人	日本放送協会
委員	小熊　寿	株式会社トヨタ IT 開発センター
委員	崎山　一男	国立大学法人電気通信大学
委員	渋谷　香士	ソニーグローバルマニュファクチャリング＆オペレーションズ株式会社
委員	鈴木　大輔	三菱電機株式会社
委員	成吉　雄一郎	ルネサスエレクトロニクス株式会社
委員	峯松　一彦	日本電気株式会社
委員	三宅　秀享	株式会社東芝
委員	渡辺　大	株式会社日立製作所
事務局	盛合　志帆	国立研究開発法人情報通信研究機構
事務局	大久保　美也子	国立研究開発法人情報通信研究機構
事務局	金森　祥子	国立研究開発法人情報通信研究機構

2023 年度版ガイドライン策定に関する謝辞

2023 年度版ガイドラインは以下に示す有識者の皆様にご執筆いただいた外部評価報告書に基づき、CRYPTREC 事務局にて執筆・編集を行いました。また、以下に示す有識者の皆様に 2023 年度版ガイドライン（ドラフト版）の改定内容についてレビューいただくとともに、改定内容の認識誤りなどについてご指摘いただきました。所属は 2024 年 3 月時点のものです。この場を借りて皆様に深く感謝申し上げます。

外部評価 [6]	井上　明子	日本電気株式会社
外部評価 [5]、ガイドライン執筆・編集	伊藤　竜馬	国立研究開発法人情報通信研究機構
外部評価 [7]	岩田　哲	国立大学法人東海国立大学機構 名古屋大学
外部評価 [3, 4]	菅野　哲	GMO サイバーセキュリティ by イエラエ株式会社
外部評価 [8, 9]	崎山　一男	国立大学法人電気通信大学
外部評価 [3, 4]	酒見　由美	GMO サイバーセキュリティ by イエラエ株式会社
外部評価 [10]	藤堂　洋介	日本電信電話株式会社
外部評価 [11]	内藤　祐介	三菱電機株式会社
ガイドライン（ドラフト版）のレビュー	本間　尚文	国立大学法人東北大学
ガイドライン（ドラフト版）のレビュー	峯松　一彦	日本電気株式会社

参考文献

[1] CRYPTREC Lightweight Cryptography Working Group: CRYPTREC Cryptographic Technology Guideline (Lightweight Cryptography) (Document ID: CRYPTREC GL-2003-2016EN) (2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016en.pdf

[2] CRYPTREC 軽量暗号ワーキンググループ: CRYPTREC 暗号技術ガイドライン（軽量暗号）（文書番号：CRYPTREC GL-2003-2016JP）(2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf

[3] GMO サイバーセキュリティ by イエラエ株式会社: 軽量暗号の評価指標、標準化動向に関する調査（NIST 軽量暗号コンペティションファイナリストなど）（文書番号：CRYPTREC EX-3206-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3206-2022.pdf

[4] GMO サイバーセキュリティ by イエラエ株式会社: 軽量暗号 Ascon などに関わる標準化動向調査（文書番号：CRYPTREC EX-3302-2023）(2023)

[5] 伊藤竜馬:「CRYPTREC 暗号技術ガイドライン（軽量暗号）」掲載の暗号方式に関する安全性評価の動向調査（文書番号：CRYPTREC EX-3101-2021）(2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf

[6] 井上明子: 軽量暗号の安全性に関する調査及び評価（Elephant,ISAP,Romulus）（文書番号：CRYPTREC EX-3204-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3204-2022.pdf

[7] 岩田哲: 軽量暗号の安全性に関する調査及び評価（Photon-Beetle,Sparkle,Tsudik's keymode）（文書番号：CRYPTREC EX-3201-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3201-2022.pdf

[8] 崎山一男: 軽量暗号の実装性能に関する調査及び評価（NIST 軽量暗号コンペティションファイナリスト）（文書番号：CRYPTREC EX-3205-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3205-2022.pdf

[9] 崎山一男: 軽量暗号 Ascon の実装性能に関する調査及び評価（文書番号：CRYPTREC EX-3301-2023）(2023)

[10] 藤堂洋介: 軽量暗号の安全性に関する調査及び評価（Ascon,Grain-128AEAD,TinyJambu）（文書番号：CRYPTREC EX-3203-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3203-2022.pdf

[11] 内藤祐介: 軽量暗号の安全性に関する調査及び評価（GIFT-COFB,Xoodyak）（文書番号：CRYPTREC EX-3202-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3202-2022.pdf

第 2 章

軽量暗号とその活用法

2.1 軽量暗号とは

近年、計算リソースの限られたデバイスにも実装可能な「軽量暗号」の研究開発が進展しており、多くの方式が学会等で提案されている。欧州では 2004 年から European Commission の第 6–7 次 Framework Programme の研究プロジェクト ECRYPT I と ECRYPT II のテーマとしても取り上げられてきた。日本も小型実装に適した暗号技術等で強みを持っている分野である。軽量暗号の標準化も進んでおり、軽量暗号アルゴリズムを技術分野毎に定めた ISO/IEC 29192 や RFID 向けの暗号技術を定めた ISO/IEC 29167 が策定され、米国 NIST も 2015 年より軽量暗号の標準化の検討を開始している。

低コスト・低消費電力で動作可能な軽量暗号技術は、今後も車載機器や医療機器など様々な機器で利用される可能性があり、IoT や CPS (Cyber Physical System) といった次世代のネットワークサービスを構築する上で有効なセキュリティ技術の一つとなることが期待されている。

一方で、一般的に合意されている軽量暗号の定義はない。これまで提案されてきた軽量暗号技術には、ハードウェア実装のサイズ・消費電力量の観点で軽量性を追求したもの、組み込みソフトウェア実装で必要なメモリサイズの軽量性を追求したもの、などの様々な性能指標で最適化された方式が存在する。また、性能と安全性のトレードオフもあり、実際の性能は多岐に渡っている。このような状況を踏まえ、本ガイドラインでは「実装性能と安全性のトレードオフを勘案した上で、従来の暗号技術に対して特定の性能指標で優位性（軽量性）を持つように設計された暗号技術」をスコープとし、用途が想定される代表的な性能指標に対して優位性を主張する暗号を主な対象とする。また、現時点で、公開鍵暗号系において軽量暗号として広くコンセンサスがとれている方式はほとんどないため、本ガイドラインでは共通鍵暗号系の軽量暗号を対象としている。

本ガイドラインでは、下記を軽量暗号の代表的な性能指標とする。

- ハードウェア実装における回路規模、消費電力量、レイテンシ
- 組み込みソフトウェア実装におけるメモリサイズ（ROM/RAM）

■回路規模　ハードウェア実装の回路規模は半導体のコストに直結し、また、消費電力の指標にもなり得ることが知られている。回路規模の小型化は、RFID をはじめとする回路実装面積の要求条件が厳しいアプリケーションで重要な要件である。また、非接触 IC カードのようにバッテリーや外部供給電源がなく、電磁誘導等で駆動するデバイスにおいても重要な要件である。

■消費電力量　消費電力量の低減は、人体へ埋め込まれたり密着装備される医療機器をはじめ、バッテリーで駆動するあらゆるデバイスで求められる要件である。

■レイテンシ　本ガイドラインにおいて、レイテンシ（遅延時間）は 1 回の暗号化（復号）処理に必要な時間を意味する。低遅延性はメモリ暗号化や車載機器などのリアルタイム性が求められるアプリケーションで必須の要件である。

■メモリサイズ　組み込みソフトウェア実装では、組み込みマイコン上で実現される様々なアプリケーションの一部として、暗号機能を実装することが多い。組み込みマイコンでは、ROM や RAM のサイズが限られており、小さく実装できる暗号ほど、選択できるマイコンの幅が広がり、コストを下げられる等の利点がある。組み込みマイコンは家電機器やセンサー、

車載向け等で広く利用されており、実装に必要なメモリサイズ（ROM/RAM）が少ないことはこれらのアプリケーションで重要な要件である。

性能指標	アプリケーションの例
回路規模（消費電力、コスト）	RFID、低コストセンサー
消費電力量	医療機器、バッテリ駆動デバイス
レイテンシ（リアルタイム性能）	メモリ暗号化、車載機器、産業向け I/O デバイス制御
メモリサイズ（ROM/RAM）	家電機器、センサー、車載機器

本ガイドラインでは、軽量暗号の代表的な方式を、ブロック暗号、ストリーム暗号、ハッシュ関数、メッセージ認証コード、認証暗号の技術分野別に分類している。それぞれの技術で実現できる機能（秘匿、認証など）は既存の暗号技術と同じである。

本ガイドラインで紹介する代表的な軽量暗号技術を表 2.1 に示す。これらの軽量暗号技術は、執筆時点までに主要国際学会等で発表されており、有力な攻撃法が発見されておらず、十分な実装性能を持ち、計算リソースの限られた実装条件下で有用と考えられるアルゴリズムを選んでいる。軽量暗号技術の概要については、4 章、付録 C にまとめている。

表 2.1　本ガイドラインで紹介する代表的な軽量暗号技術

ブロック暗号	CLEFIA, LED, Midori, Piccolo、PRESENT, PRINCE, SIMON, SPECK, TWINE, LEA
ストリーム暗号	ChaCha20, Enocoro, Grain v1, MICKEY 2.0, Trivium
ハッシュ関数	Keccak, PHOTON, QUARK, SPONGENT, Lesamnta-LW
メッセージ認証コード	SipHash, Chaskey, LightMAC, Tsudik's keymode
認証暗号	Ascon, ACORN, AES-JAMBU, AES-OTR, CLOC and SILC, Deoxys, Joltik, Ketje, Minalpher, OCB, PRIMATEs, Grain-128A, Elephant, GIFT-COFB, Grain-128AEAD, ISAP, PHOTON-Beetle, Romulus, Sparkle, TinyJAMBU, Xoodyak

なお、AEGIS と COLM の 2 方式については軽量性を謳った方式ではないものの、CAESAR final portfolio に選出された方式であるということを鑑み、付録 B で調査結果をまとめている。

以降、2.2 節では、軽量暗号の選定に関する標準化プロジェクトの動向を記載する。具体的には、CAESAR（Competition for Authenticated Encryption: Security, Applicability, and Robustness）コンペティションと NIST 軽量暗号（LWC: Lightweight Cryptography）プロジェクトについてまとめる。その他、表 2.1 で示す代表的な軽量暗号技術の標準化動向について、標準化されている場合に限り、その標準化状況を 4 章と付録で記載している。次に、2.3 節で軽量暗号がどのような場面で使えるかという活用例を示し、2.4 節で方式や鍵長、ブロック長を選ぶ際の留意点を記載する。最後に、2.5 節では、2.3 節で取り上げる活用例において、どのような点に着目して軽量暗号を選定すればよいか具体例を示す。なお、2.3–2.5 節の記載内容は 2016 年度版ガイドライン執筆時点（2017 年 3 月現在）のものであることに留意いただきたい。

2.2　軽量暗号の標準化動向

軽量暗号の選定に関する代表的なプロジェクトとして広く知られている CAESAR コンペティションと NIST LWC プロジェクトの標準化動向について紹介する。また、NIST LWC プロジェクトの最終選考方式として Ascon が選出されたことを受け、標準化団体における Ascon の検討状況についても紹介する。これらの標準化動向については、2022 年度と 2023 年度に公開された CRYPTREC 外部評価報告書 [11, 12] に基づき、2023 年 9 月現在の調査結果を記載している。

2.2.1　CAESAR コンペティション

CAESAR（Competition for Authenticated Encryption: Security, Applicability, and Robustness）コンペティション [4] は、認証暗号技術に関する設計の促進を図るために国際的な暗号研究コミュニティによって運営されたコンペティショ

ンである。2013 年 1 月に開催された Early Symmetric Crypto workshop[*1]でコンペティション開催についてアナウンスされ、2014 年 3 月に 57 件の応募があった。その後、第 1 ラウンド候補として 48 件に絞り込まれた上で、第 1 ラウンド、第 2 ラウンド、第 3 ラウンド、そして最終ラウンドの 4 回の評価フェーズを経て 2019 年 2 月に最終的なポートフォリオが 6 件発表された。表 2.2 は、CAESAR コンペティションにおける各評価フェーズの期間と評価対象アルゴリズムの数をまとめたものである。

表 2.2　CAESAR コンペティションにおける選定プロセスの動向

評価フェーズ	期間	対象アルゴリズム数
1	2014 年 3 月–2015 年 7 月	57 → 48
2	2015 年 7 月–2016 年 8 月	30
3	2016 年 8 月–2018 年 3 月	15
最終	2018 年 3 月–2019 年 2 月	7

最終的なポートフォリオ数：6

最終的なポートフォリオは、ユースケース 1 の軽量アプリケーション（リソースに制約のある環境）、ユースケース 2 の高性能アプリケーション、ユースケース 3 の多層防御、という 3 部構成となっており、軽量暗号技術に該当するユースケース 1 の第 1 候補として Ascon、第 2 候補として ACORN が選定された。参考までに、ユースケース 2 の第 1 候補は AEGIS-128、第 2 候補は OCB であり、ユースケース 3 の第 1 候補は Deoxis-II、第 2 候補は COLM である。これら 6 件の最終的なポートフォリオの概要については、4.5 節と付録 B でまとめている。

CAESAR コンペティションでは評価基準が明確に示されていないものの、AES-GCM よりも優れた利点を有し、幅広い領域で採用される認証暗号を選択することを目的としている[*2]。また、ソフトウェアとハードウェアの実装性能評価に関し、統一されたフレームワークを使用して測定できる仕組みを導入している[*3]。

2.2.2　NIST LWC プロジェクト

NIST 軽量暗号（LWC: Lightweight Cryptography）プロジェクト [3] は、制約のあるデバイス上などで限定的に使用が認められる軽量暗号アルゴリズムと暗号利用モードのポートフォリオを開発および維持することを目的として開催されたコンペティション形式のプロジェクトである。2013 年、NIST は軽量暗号の標準化に向けたプロジェクトを開始し、2015 年 7 月開催の第 1 回ワークショップと 2016 年 10 月開催の第 2 回ワークショップを経て、2017 年 3 月に NIST は軽量暗号に関するレポート NISTIR 8114 [16] を発行するとともに、オープンプロセスを通じて軽量暗号アルゴリズムのポートフォリオを作成することを決定したとアナウンスした。2018 年 8 月、NIST は軽量暗号標準化プロセスのための応募要件と評価基準 [19] を公開して新しい軽量暗号アルゴリズムの募集を開始し、2019 年 3 月の締切時に 57 件の応募を受け取った。その後、第 1 ラウンド候補として 56 件に絞り込まれた上で、第 1 ラウンド、第 2 ラウンド、そして最終ラウンドの 3 回の評価フェーズを経て、2023 年 2 月 7 日に最終選考方式として Ascon が選定された。表 2.3 は、NIST LWC プロジェクトにおける各評価フェーズの期間と評価対象アルゴリズムの数をまとめたものである。以下、これら 3 回の評価フェーズを概説する。

[*1] https://www.cryptolux.org/mediawiki-esc2013/index.php/ESC_2013

[*2] Call for submissions のページ（https://competitions.cr.yp.to/caesar-call.html）において、“CAESAR (Competition for Authenticated Encryption: Security, Applicability, and Robustness) will identify a portfolio of authenticated ciphers that (1) offer advantages over AES-GCM and (2) are suitable for widespread adoption. Cryptographic algorithm designers are invited to submit proposals of authenticated ciphers to CAESAR. All proposals will be made public for evaluation.” と記載されている。その他、同ページには応募に際して様々な要件（機能要件、ソフトウェア要件、ハードウェア要件、応募要件）が指定されている。

[*3] Submissions のページ（https://competitions.cr.yp.to/caesar-submissions.html）において、“See https://bench.cr.yp.to/supercop.html for software implementations, and https://cryptography.gmu.edu/athena/index.php?id=CAESAR_source_codes for VHDL implementations.” と記載されている。

表 2.3　NIST LWC における選定プロセスの動向

評価フェーズ	期間	対象アルゴリズム数
1	2019 年 3 月–2019 年 8 月	57 → 56
2	2019 年 8 月–2021 年 3 月	32
最終	2021 年 3 月–2023 年 2 月	10

最終選考方式の数：1

2.2.2.1　第 1 ラウンド

NIST は、2019 年 3 月に 57 件の応募を受け取った後、軽量暗号標準化プロセスのための応募要件と評価基準 [19] で示した要件に基づき、この要件に対する完全性と妥当性の観点から選定を行った。2019 年 4 月、NIST は 57 件の応募のうち、第 1 ラウンド候補のアルゴリズムとして 56 件の応募を承認し、第 1 ラウンドの評価フェーズを開始した。2019 年 8 月に第 1 ラウンドの評価フェーズが終了し、第 2 ラウンド候補として 32 件のアルゴリズムが発表された。その後、NIST はステータスレポート NISTIR 8268 [23] を発行し、第 1 ラウンドにおける評価基準や選定プロセスを明確にした。

NISTIR 8268 [23] によると、第 1 ラウンドにおける評価基準や選定プロセスの中で最も重要な観点は暗号アルゴリズムの安全性である。この理由は、以下の 2 点が挙げられる。

1. 第三者による安全性評価が公開されていない、または応募資料において安全性の主張を裏付ける情報が不十分である暗号アルゴリズムについては第 2 ラウンド候補から除外された。
2. 第三者による安全性評価によって安全性上の懸念が生じた暗号アルゴリズムについては第 2 ラウンド候補から除外された。具体的には、偽造攻撃、Length-extension attack、識別攻撃、その他の予期しない性質（Undesirable properties）が存在する暗号アルゴリズムが整理された。

なお、実装のバグによる実用的な攻撃（例えば、偽造攻撃）については、第 2 ラウンド候補から除外されていない。

2.2.2.2　第 2 ラウンド

NIST は、2019 年 8 月に第 2 ラウンド候補となる 32 件のアルゴリズムを発表した後、第 2 ラウンドの評価フェーズを開始した。2021 年 3 月に第 2 ラウンドの評価フェーズが終了し、最終ラウンド候補のアルゴリズム（ファイナリスト）として Ascon、Elephant、GIFT-COFB、Grain-128AEAD、ISAP、Romulus、PHOTON-Beetle、SPARKLE、TinyJAMBU、Xoodyak の 10 件が発表された。その後、NIST はステータスレポート NISTIR 8369 [24] を発行し、第 2 ラウンドにおける評価基準や選定プロセスを明確にした。なお、これら 10 件のファイナリストの概要については、4.5 節と付録 C でまとめている。

NISTIR 8369 [24] によると、第 1 ラウンドと同様、第 2 ラウンドにおいても暗号学的な安全性が最も重要な評価基準となっている。具体的には、以下のいずれかに該当するアルゴリズムがファイナリスト選定時において重要な評価指標となった。

1. 第三者による安全性評価が十分に行われたアルゴリズム
2. 十分に認知されている設計原理と安全性証明に基づく安全性主張が明確であるアルゴリズム

つまり、第三者による安全性評価の結果により、安全性主張の妥当性について懸念が生じたアルゴリズムについては、ファイナリストから除外されている。特に、除外されたアルゴリズムについては、暗号プリミティブ（例えば、暗号学的置換、ブロック暗号、tweakable ブロック暗号、ストリーム暗号、など）に対する識別攻撃や弱鍵クラスの存在が懸念された。加えて、現実的な偽造攻撃や鍵回復攻撃など、安全性主張を無効にする評価結果も提供された。

制約のある環境でのハードウェアとソフトウェアの実装性能についても重要な評価基準となっている。第 2 ラウンド候補は様々な実装性能とコストの観点から評価・比較され、現在の NIST 標準である AES-GCM [9] や SHA-2 [18] よりも著しく性能に優れたアルゴリズムであるということがファイナリスト選定時において重要な評価指標となった。なお、NIST は

公開されている様々なハードウェア・ソフトウェアのベンチマークに加えて、独自のソフトウェアベンチマークを用いて評価した。

追加の評価基準として、サイドチャネル攻撃耐性、nonce-misuse 安全性、releasing unverified plaintext (RUP) 安全性、内部状態復元の影響、耐量子安全性を含む様々な性質を満たすかについても考慮された。

その他、候補の多様性についても考慮された。具体的には、アルゴリズムの根底を担う暗号プリミティブをベースとした複数の有望な候補が存在する場合、暗号プリミティブの種類によってグループ分けされるとともに、グループ内で相互比較され、候補が絞り込まれた。

2.2.2.3 最終ラウンド

NIST は、2021 年 3 月にファイナリストとなる 10 件のアルゴリズムを発表した後、最終ラウンドの評価フェーズを開始した。当初、NIST LWC プロジェクトにおいて、制約のある環境に適した認証暗号とハッシュ機能として 1 つまたは複数のアルゴリズムを選定するために標準化プロセスが開始されたが、2023 年 2 月 7 日に最終ラウンドの評価フェーズが終了し、最終選考方式として Ascon を選定したことが発表された。その後、NIST はステータスレポート NISTIR 8454 [25] を発行し、最終ラウンドにおける評価基準や選定プロセスを明確にした。なお、Ascon の概要と実装性能評価については、それぞれ 4.5 節と 3.3 節でまとめている。

以下、NISTIR 8454 [25] に基づき、最終ラウンドにおける評価基準と選定プロセスについてまとめる。

■評価基準 以下の 4 項目が主な評価基準となっている。

1. 暗号学的安全性
2. 制約のある環境下におけるソフトウェア及びハードウェアでの実装性能
3. サイドチャネル攻撃や故障利用攻撃への耐性
4. 知的財産

最も重要な評価基準は、暗号学的安全性である。設計者自身の安全性評価、安全性主張、安全性証明、公開されている第三者による安全性評価などの情報を幅広く評価している。また、明示的に要求されていないが、第 2 ラウンドと同様、nonce-misuse 安全性、RUP 安全性、内部状態復元の影響、耐量子安全性などが追加の考慮事項として挙げられている。なお、ファイナリストに対する安全性評価については、NISTIR 8454 [25] の第 3 章で整理されている。

もう 1 つの重要な評価基準は、制約のある環境下におけるソフトウェア及びハードウェアでの実装性能である。様々な性能やコストに関する測定基準において、ファイナリスト同士や現在の NIST 標準である AES-GCM [9]（認証暗号としての比較対象）と SHA-2 [18]（ハッシュ関数としての比較対象）との比較、評価が行われた。広く採用されている AES-GCM や SHA-2 に対し、大幅に優れた性能を発揮することが期待されている。なお、ファイナリストの性能比較結果については、NISTIR 8454 [25] の第 4 章と付録 B で整理されている。

その他、サイドチャネル攻撃への耐性を提供する必要はないが、容易かつ低コストで実現できることが要望されている（細部は、NISTIR 8454 [25] の第 4.3 節を参照されたい）。知的財産についてもまた、アルゴリズムの使用や実装における特許請求の必要性について反対しないものの、評価フェーズでの選定を妨げる可能性があると示されている（細部は、NISTIR 8454 [25] の第 2.2 節を参照されたい）。

■選定プロセス NIST LWC プロジェクトの初期段階において、NIST はターゲットアプリケーションに関するパブリックフィードバックに基づき、次の 2 つのプロファイルを指定した。

- プロファイル 1：制約のある環境でのソフトウェア及びハードウェア環境向けの認証暗号とハッシュ
- プロファイル 2：制約のある環境でのハードウェア環境向けの認証暗号

当初、両方のプロファイルをカバーするアルゴリズムの提出が求められたが、NIST は標準化のために複数のアルゴリズム（例えば、各プロファイルに対して 1 つのアルゴリズム）を選定することも検討した。

NISTIR 8454 [25] によると、ファイナリストに対する評価プロセスは、第三者による安全性評価、バリエーション、設計の微調整、ベンチマーク、耐量子安全性、知的財産に関する声明の 6 つの観点から検討されていることがわかる。なお、

表 2.4　NIST LWC 最終ラウンドにおける評価基準と選定プロセスの関係

番号	評価基準	選定プロセス
1	暗号学的安全性	第三者による安全性評価、耐量子安全性
2	制約のある環境下におけるソフトウェア及びハードウェアでの実装性能	ベンチマーク
3	サイドチャネル攻撃や故障利用攻撃への耐性	ベンチマーク
4	知的財産	知的財産に関する声明
5	その他	バリエーション、設計の微調整

表 2.4 は NIST LWC 最終ラウンドにおける評価基準と選定プロセスの対応関係をまとめたものである。

第三者による安全性評価　ファイナリストに対し、第三者による多くの安全性評価が行われた。その細部については、NISTIR 8454 [25] の第 3 章でまとめられている。公開されている安全性評価は、いずれも単一鍵または nonce-respecting 設定での安全性主張を無効にするものではなく、ファイナリストの多くは十分に安全性マージンが確保されている状況である。

バリエーション　応募されたアルゴリズムの公正な比較を行うために、NIST は各設計チームに対して特定の入出力サイズを持つ認証暗号とハッシュのバリエーションを提出するように求めた。一方で、NIST は、異なる入出力サイズ、異なる種類のベースとなる構成要素など、最大 10 種類までの複数のバリエーションを提出することも許容した。このような要望に対し、いくつかの設計チーム（例えば、Ascon、SPARKLE、Xoodyak、など）は eXtendable Output Function (XOF) のバリエーションを提出した。XOF は正式なバリエーションとはみなされないものの、XOF 機能を提供できる柔軟性は選定プロセスにおいて設計の利点であるとみなされた。

設計の微調整　最終ラウンドの初期段階において、安全性や実装性能を向上させるための軽微な設計変更が許容されたが、Ascon、GIFT-COFB、ISAP、PHOTON-Beetle、そして SPARKLE については、設計上の微調整が実施されなかった。

ベンチマーク　ベンチマーク結果に関して、NIST 標準である AES-GCM [9] と SHA-2 [18] よりも大幅に優れた性能を発揮することが期待された。具体的には、ソフトウェアでのベンチマーク結果、ハードウェアでのベンチマーク結果、サイドチャネル攻撃や故障利用攻撃への耐性、そしてこれらの攻撃を軽減させるために必要な実装オーバーヘッドに関する評価が行われた。また、特定の用途に応じて最適な実装を行うために、実装者がコストと性能のトレードオフを考慮できる柔軟性もまた、軽量暗号の重要な要素の 1 つとされた。

耐量子安全性　暗号アルゴリズムの長期利用の観点から、量子計算機の脅威に対する安全性が考慮された。共通鍵暗号アルゴリズムの耐量子安全性として最も一般的な量子アルゴリズムは Grover のアルゴリズム [13] である。例えば、量子計算機ではない現在の計算機（古典計算機）では計算量 $\mathcal{O}(2^n)$ で秘密鍵の全数探索が実行可能であるのに対し、量子計算機では Grover のアルゴリズムを使用して計算量 $\mathcal{O}(2^{n/2})$ で秘密鍵の全数探索が実行可能となる。また、古典計算機では計算量 $\mathcal{O}(2^{n/2})$ でハッシュ関数の衝突探索が可能であるのに対し、量子計算機では Grover のアルゴリズムを使用して計算量 $\mathcal{O}(2^{n/3})$ でハッシュ関数の衝突探索が可能となる。この量子アルゴリズムに対して安全性を確保するために、より大きな鍵サイズ（又は、ハッシュ関数におけるより大きな出力サイズ）が必要となる。なお、ファイナリストのうち、Ascon、SPARKLE、TinyJAMBU が 128 ビットよりも長い秘密鍵をサポートしている。

知的財産に関する声明　各設計チームに対し、NIST は候補アルゴリズムの実装によって侵害の恐れがある知的財産を全て特定するよう要求していた。結果として、PHOTON-Beetle のみが既存の知的財産を有することとなった。一方で、この知的財産に関する事項は、選定プロセスの決定には影響を及ぼしていないと示されている。

2.2.2.4　Ascon に関する評価

最終ラウンドにおける評価基準と選定プロセスに従い、NIST は標準化対象のアルゴリズムとして Ascon を選定した。具体的には、以下の観点が特に評価されている。

機能　Ascon ファミリーには、認証暗号とハッシュに加えて、追加の XOF が含まれている。これにより、幅広いアプリケー

ションのニーズを満たすことができる。また、暗号学的置換ベースの設計であることから、追加機能を実装するための追加コストが少ないことが期待されている。

成熟度　認証暗号のバリエーションは、CAESAR コンペティションで最終的なポートフォリオの 1 つである軽量アプリケーション（リソースに制約のある環境）の第 1 候補として選定された実績がある。また、最終ラウンドで行われた設計の微調整において、バリエーションが追加されたものの、第 2 ラウンドのバリエーションに関しては設計の変更が行われなかった。これらの事実から、安全性や実装性能を向上させるために設計の微調整を行なった他ファイナリストよりも高い成熟度を満たしていると言える。

安全性　初期バージョンの仕様が公開されてから長い歴史があり、豊富な評価・分析が行われてきたことから、第三者による安全性評価が最も多いファイナリストである。他ファイナリストと比較すると、安全性評価が先行して行われているにも関わらず、依然として高い安全性を維持している。特に、認証暗号のバリエーションは、nonce-respecting 設定で高い安全性マージンを提供するとともに、nonce-misuse 設定でも高い完全性を保証する。さらに、認証暗号モードは、耐漏洩安全性のためにモードレベルの保護メカニズムも提供している。

実装性能　ソフトウェア及びハードウェアで非常に優れた性能を発揮するとともに、コストと性能の間の様々なトレードオフをサポートする実装の柔軟性を実証した。特に、リソースが限られている様々なソフトウェア及びハードウェア上で、現在の NIST 標準である AES-GCM と SHA-2 より性能が優れている。また、サイドチャネル攻撃等への対策を施すことで保護された実装は、保護されていない実装よりも追加コストが低いことも示された。

■制約事項　重要な制約事項の 1 つは、256 ビット鍵のオプションがないことである。これは、量子攻撃に対する 128 ビット安全性が必要な場合に問題となる可能性がある。現状、耐量子安全性として 256 ビット鍵が必要となる場合、AES-GCM を使用することが推奨されており、NIST は今後必要に応じてより高い耐量子安全性を満たすバリエーションの追加を検討する可能性があると示唆している。

NISTIR 8454 [25] の発行時点において、NIST は第二候補のアルゴリズムを選定する必要がないと判断している。当面の間、Ascon ファミリーが制約のある環境下で十分な安全性を提供でき、実装性能においてもターゲットデバイスやターゲットアプリケーションで許容されると予想しているためである。

2.2.3　他標準化団体における Ascon の検討状況

NIST LWC プロジェクトの最終選考方式として選定された Ascon に関し、NIST 以外の組織での標準化動向についてまとめる。調査対象の標準化団体は、Internet Engineering Task Force (IETF)、World Wide Web Consortium (W3C)、International Organization for Standardization/International Electrotechnical Commission (ISO/IEC)、International Telecommunication Union Telecommunication Standardization Sector (ITU-T)、Global Platform の 5 団体である。

2023 年 9 月現在、IETF を除く 4 団体において標準化が行われていない。IETF では、以下のとおり Ascon が取り上げられている。

- インターネットドラフト “Secure UAS Network RID and C2 Transport” [17] の第 5.3 節において、無人航空機で Ascon を選択することが最善である、と記載されている。その際、Encapsulating Security Payload (ESP) [15] や Datagram Transport Layer Security (DTLS) [22] の拡張が必要であると記載されている。
- インターネットドラフト “Properties of AEAD algorithms” [5] の第 4.4.2 節において、NIST LWC プロジェクトに関する参照が行われている。
- IETF 117 で開催された TLS ワーキンググループでの発表 “New Post-Quantum Signatures on the Horizon” [26] において、Ascon-Sign (SPHINCS$^+$ with Ascon) が取り上げられていた。

その他、産業界では、NIST LWC プロジェクトの結果を受け、Ascon を利用可能な環境を提供するような動向がある。例えば、IP コア関連では、Rambus 社の「Ascon-IP-41 暗号エンジン」 [21]、Xiphera 社の「XIP2201B: Ascon」 [27]、CAST 社の「Ascon-F」 [6] が提供されており、暗号ライブラリ関連では、Bouncy Castle 1.7.3 以降 [7]、CIRCL [8] などで提供されている。NIST による Ascon の最終的な標準化仕様が公開されることで、他標準化団体や産業界での活動が活性化されることが期待される。

2.3 軽量暗号はどこに使えるのか

「いつでも、どこでも、何でも、誰でも」ネットワークにつながるユビキタスネットワークの構築は、IoT というキーワードで表現されるようになっている。IoT というコンセプトの下、パソコンやスマートフォン、タブレットといった従来型の ICT 端末だけでなく、自動車、家電、ロボット、施設などがインターネットに繋がることとなる [30]。ただし、どのようなデバイスがネットワークに接続されるかについて明言することはできず、デバイス上でどのような処理が行われるかも不明である。IoT の時代だからこそ、今までは考えられない状況を考えなければならない。

これら IoT 端末は、先述したユビキタスネットワーク構築のために、我々の生活空間へシームレスに浸透しつつあるため、プライバシーや情報秘匿、また情報の完全性担保を目的としたセキュリティ機能が必須となる。加えて、サービス利用者の目的はセキュリティ機能によるメリットを享受することではないため、IoT 端末を利用した円滑なサービス提供をセキュリティ機能が妨げるべきではない。また、全てのデバイスに高機能の CPU が搭載されるとは考えにくい。従来型 ICT 端末に比べて処理能力、回路規模、消費電力、そしてメモリサイズなどの制約を含め、計算機資源が乏しいデバイスも想定しなければならない。例えば、IoT 端末の 1 つである自動車において、図 2.1 のような要求条件が述べられている。

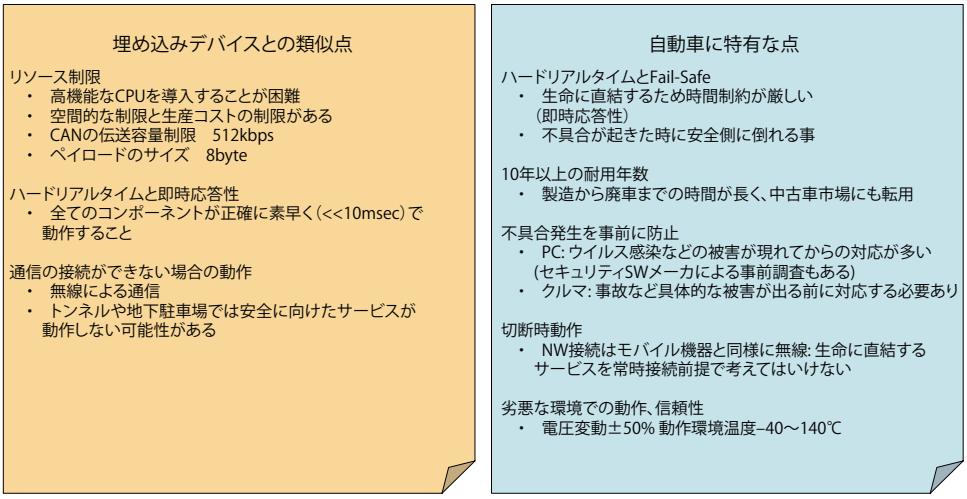


図 2.1 自動車の機能に関する要求条件

そこで、有望となるのが軽量暗号である。軽量暗号には、CPU 負荷が軽い、使用するメモリサイズが小さい、低レイテンシ性などの特徴がある。このような軽量暗号は、計算リソースが比較的乏しい IoT 端末に適していると考えられており、特徴に応じて、例えば次のような場面での使用が期待されている。

- CPU やメモリなどを多くのアプリケーションで共有しなければならない場合、CPU コストが小さく、メモリ使用量が少ない暗号として軽量暗号の使用が期待される。例えば、スマートフォンやタブレット端末、スマートテレビのような高機能化されたデバイスなどがこの場合に該当する。
- 装置そのものがバッテリーで動作している場合、消費電力が少ない暗号として軽量暗号の使用が期待される。例えば、電気が通っていない場所に置かれることが多い環境測定用のデバイスなどがこの場合に該当する。また、埋め込み型の医療機器に関して、人体に埋め込まれるデバイスであればバッテリー駆動以外は考えられない。埋め込み機器として可能な限り小さいデバイス、かつ人体への影響が少ないデバイスであることが望まれることから、このような場合において軽量暗号の使用が期待される。
- 即時性が求められる場合、低レイテンシ性を持つ暗号として軽量暗号の使用が期待される。例えば、バッテリーの消費を極力抑えるために、データを送信する場合にのみ電源スイッチが ON になり、一瞬だけ動作してデータを伝送し、伝送が終了次第電源スイッチが OFF になるように運用している機器などがこの場合に該当する。また、自動車に関しても、制御するためにかかる時間が遅い場合に安全性に影響が出る可能性が高いため、この場合に該当する。

これら軽量暗号の活用例について、スマートテレビ、RFID、農地での環境測定、医療機器、工場の機器制御、自動車を例

にして紹介する。

2.3.1 家電・スマートテレビ

スマートフォンやタブレット端末、スマートテレビのようなデバイスに搭載されている CPU には様々な種類があり、低機能なものから高機能なものまである。高機能の CPU を使用している場合は問題ないが、低機能の CPU を使用している場合は当然、実現できる機能が制限される。

例えば、テレビ内の処理は、スクランブルの解除、圧縮の復号、映像・音声の提示のみがハードウェア処理であり、それ以外の処理は内蔵されている CPU で処理される。また、CPU は実に多くの作業をしており、ほとんど負荷 100% で使用されている。さらに、今後の発展としてネットワークと接続して通信アプリケーションをテレビ上で動作させようとしている。テレビを少しでも安価にするためには、搭載する CPU も安価なものが好ましく、このため高機能な CPU が使われていないのが現状である。そのようなテレビにおいて、テレビ機能だけでなく、アプリケーションをシームレスで動作させる場合には、CPU やメモリリソースの取り合いが生じる。その上、暗号化すべきデータが存在するならば、暗号化は使用メモリ（ROM/RAM）サイズが小さく、CPU 負荷が小さいものが推奨される。テレビ内で暗号用途の特別なチップを導入することはなく、ソフトウェアで暗号化処理を行うこととなるため、ソフトウェアで CPU 使用時間が短く ROM の使用サイズが小さい暗号の利用価値が高いと考えられる。

IoT 時代では、テレビ以外にも多くの家電がネットワークに接続される。ネットワークを通じてやりとりするデータの中には秘匿しなければならないデータもあり得る。仮に、エアコン、ガスコンロなどがネットワークに接続された場合、その制御信号がネットワーク上を流れるサービスでは、制御信号が外部から不正アクセスされて改ざんされたり、任意のコマンドが入力されることで異常動作しないようにしなければならない。また、家庭内のロボットが個人データを持つ場合、プライバシー保護の観点からデータを秘匿しなければならない。一方、これらのデバイスは 10 年以上使用し続けることも考えなければならない。その上、小さな・安価なデバイスにおいては特殊なハードウェアが搭載されることは考えにくい。デバイスによっては、スマートテレビと同じようにリソースの取り合いが生じることもあり得る。これらのデバイスでは、アップデートが可能なソフトウェア処理がメインとなり、それを司る CPU は低機能で安価なものになるであろう。このような CPU でデータ保護を実現するためには、ソフトウェアで CPU 使用時間が短いタイプの暗号が必要となる。

2.3.2 RFID タグ利用のアプリケーション（物流管理等）

RFID とは、無線を利用して物を認識するシステムのことであり、様々な用途で使用されている。例えば、倉庫内の在庫管理、物流における物品管理、CD/DVD ショップでの盗難防止、物品の履歴管理、電子マネー、交通用カード、社員証カードなどである。動物の生態を調べるための追跡用にも使われることがある。同様に、人の居場所を追跡するために利用することなども考えられている。さらには、IoT 時代の家庭内の物品識別、例えば、冷蔵庫内に入っているものを冷蔵庫が認識するために利用することも考えられている。

RFID による無線電波は強くはないため、近距離伝送で利用されることが多い。したがって、例えば、倉庫内にどのような物品が保管されているかを別の同業者が知るために、RFID で発信されるデータを倉庫の外部から盗聴するようなことは困難である。倉庫内での在庫管理においては、暗号を実装する必要性は少ないと考えられる。

これに対し、CD ショップでの万引き防止、電子マネー、交通用カード、社員証カードの偽造防止、人の追跡や人の持ち物の追跡におけるプライバシー保護、動物の追跡における密漁防止等の目的で使用される場合は、データは秘匿されることが必須である。

さて、RFID タグは RFID で用いられるチップのことであり、そのサイズは数十 μm 角から、数 cm 角のものまである。このチップの中にプロセッサ、メモリなどが凝縮されている。図 2.2 に RFID の構造を示す。
送信部と書かれている部分はアンテナであり、使用されている周波数（kHz〜MHz）に応じてその大きさは様々である。送信部以外は小型化が可能であるが、データ記憶容量や使用されるプロセスによって大きさが決まる。

RFID では、全体の回路規模に対して暗号機能に使える回路サイズには限界がある。RFID がパッシブタグ、すなわち電池を内蔵していない場合、電波から給電するため消費電力に大きな制約が存在する。このような制約の下でデータを保護するため、ハードウェアが小さく、消費電力の少ない暗号が求められる。

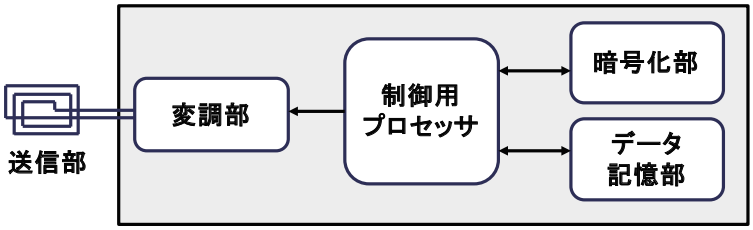


図 2.2　RFID の構造

2.3.3　センサーを利用したスマート農業

農作物を栽培する際に、気候の変化に応じた対応を講じることで、収穫量を安定させるなど生産性を向上させたり、作物の品質を向上させることが可能になると言われている。これまで多くの農家では、このような対応を熟練者の経験や勘で補ってきた。例えば、気候（気温、湿度、日照時間、日照量、土壌水分、風向、風速、降水量など）をモニタリングし、これまで経験・勘に頼っていた部分を数値化・データ化することができれば、熟練者のみならず経験の浅い方々でも安定した農業を営むことができる。具体的には、これらのモニターにより、水まき時間、量を制御したり、ビニールハウスの窓の開け閉めを自動化するなどが可能となる。熟練者にとってもこのようなモニタリングは、気候への対応を容易にし、作業計画・スケジューリング、病害虫駆除対策等が容易になるだけでなく、農地そのものを評価し・改良するための指針を与えることになり、安定した収穫への手助けとなる。

モニタリングするデータは細かければ細かいほど分析精度が高くなる。特に、農地の評価を行う上では、広大な農地を区画に分けて区画ごとのデータを取ることが望ましい。農家の方々は、少ない労力で長期的に、安価に利用できるモニタリング装置を要望することとなる。このため、徐々に環境センサーネットワークの利用が始まっている。

このセンサーへの条件として、自律駆動、小型、省電力、大量設置、などがある。また、細かいデータ取得には、膨大な数のセンサーが必要となるため、これらの条件を満たすためにも軽量暗号の使用が望まれる。

同じような状況が防災上も考えられる。天候変化だけでなく、地殻の変動をセンシングすることで地震予知に役立てたり、火山噴火予知、土砂崩れなどの予測に役立てることが考えられている。これらの用途のセンサーも、人の出入りが困難であり、給電が困難な場所に置かれることがしばしばある。また、防災関連のデータは重要であり、盗聴されることは許容できるが、改ざんなどがあってはならない。データが改ざんされることで、不要な警報が発せられることがあってはならないからである。このような秘匿性のあるデータの改ざん防止、保護のためには、認証付きの軽量暗号が有望と考えられる。

2.3.4　医療

入院経験のある方は少なくないと思うが、入院患者には心電図・血圧・脈拍・血糖値・酸素濃度などを測定する様々なセンサーが取り付けられる。基本的にこのようなセンサーは有線で接続されており、電源に接続されている。ただし、患者が移動する場合、このようなセンサーも患者とともに移動できることが望ましい。特に、帰宅等の外出が可能な方、在宅・職場であっても種々のデータをモニタリングする必要のある方もいる。さらに、体内に測定装置を埋め込まなければならない場合も考えられる。定期的なデータ測定、投薬時間の連絡などのためである。特に、デバイスを埋め込む場合、何度もデバイスを取り出すことは考えにくい。このため、無線利用のデータ通信、バッテリーによる長時間動作が可能なデバイスでなければならない。埋め込み型ではなくとも、有線では移動の自由度が制限されるため無線であることが望ましく、必然的にバッテリーで長時間動作するセンサーであることが必須となる。

これらのセンサーデータは個人情報そのものであり、プライバシー保護の観点から全てのデータが秘匿されなければならない。現在、特に埋め込み型センサーでは小型化の研究・開発が進められており、nm サイズのデバイスが開発されているところである。当然、それに付随する暗号化処理部分も小型化が必須である。

最近ではウェアラブルデバイスの進展に伴い、mHealth と呼ばれる概念が出てきている。Mobile Health の略とも言われるが、確立した定義は存在していない。例えば、mHealth として次のようなことが考えられている。ウェアラブルデバイスを生体データの観測装置として利用し、日々の健康維持のために心肺活動などのデータを観測する。家庭内で日々のデー

タを記録し、健康診断や通院の際に利用するものである。また、日常生活の中のデータ保存のみならず、健康維持のために行っている活動の記録としてもしばしば利用する。従来からの健康機器である歩数計も GPS による位置情報の取得など、個人情報を取り扱うものが出てきている。

この mHealth では、人の動作により発電する発電デバイスを組み込んだデバイスもあるが、基本的にはウェアラブルデバイスが測定装置となるため、バッテリー運用となる。使用方法にも依存するが、常時データを測定し、観測、集計、分析するのであれば、無線による伝送が必要となる。これらのデータも個人情報そのものであり、プライバシー保護の観点から全てのデータが秘匿されなければならない。

2.3.5 産業用システム

工場などでは資材の運搬や加工、組み立てなどの工程を自動化し、効率的に運用することが考えられている。例えば、工作機械やロボットを動作させるにあたって、ネットワークを通じて情報を共有したり、センサーを用いた工程管理を行うことが可能となる。さらに、ネットワークを通じて全ての情報を一か所に集めることで、集中管理が可能となる。

もっとも先進的にこのような自動化に取り組んでいたのが、ドイツであると思われる。国家プロジェクトとして多額の予算を組み、インダストリー 4.0 として種々の機械・装置がセンサーを持ち、そのデータに応じて考えながら動作するスマートファクトリ（考える工場）という概念を実現しようとしてきた。

データを集中管理するにあたり、ネットワークについても EtherCAT と呼ばれる方式が提案された。工場のデータ管理では、個々のデータが重要な意味を持っている。従来のインターネットプロトコルである TCP/IP ではデータのネットワーク上での衝突等が起こり、データの遅延、消失などが生じてしまうため、工場データの管理に利用するには弊害があった。そこで、EtherCAT では、個々の機器をスレーブ化し、その接続をシリアルにする方式としている。

これらのセンサーは工場内のあらゆる場所に配置される。当然、人の手が入りにくい場所もある。電源が装置に備え付けてある場合が多いものの、全てをシリアルに接続することが困難な状況も生じる。これに対応するためには、無線でデータを送ることも考えられている。無線でデータを送る場合は、工場内で送受信できるようにするため、ある程度の距離を無線伝送することになる。このためには、暗号化してデータを送受信することが必要である。

2.3.6 自動車

自動車では、車内でのデータ通信だけでなく、車外との通信も行うようになってきた。この自動車の通信においては、自動車と自動車（車車間）ならびに自動車と信号機や道路標識などのインフラ（路車間）との相互通信により安全運転の支援を行う自動運転支援システム（Car2X communication）に対応するために大量の情報を処理し、かつクラウドと連携してコンテンツなどのサービスと連携する車載情報システム、ならびにレーダなど外部からの情報を各種センサーで取得しボディ系、シャシー系など複数の電子制御ユニット（ECU）間において互いに情報をやりとりしながら協調制御をおこなう車載制御などがある。

車内の ECU 間の協調制御をおこなうため、車内は CAN (Controller Area Network)、LIN (Local Interconnect Network)、Ethernet など各種方式によるネットワークが張り巡らされている。CAN は車載において基幹となるネットワークであり、パワートレイン系、シャシー系、ボディ系などとの協調制御に広く使用されている。例えば、車間をミリ波レーダで計測し、車間が狭くなると警告表示あるいは警告ブザーを開始、ブレーキ制御、シートベルト制御といった衝突検知システムを CAN を通じて情報共有を行うことで実現している。車載 LAN においても攻撃の実証例があり、CAN に誤ったメッセージが流れるとブレーキ制御が誤るなどの重大な事態につながりかねないため、メッセージを暗号化するとともに改ざんされていないか認証を行う必要がある。車車間、路車間のように非常に高い暗号処理性能、低レイテンシ暗号までは要求されないが、リアルタイム性は求められており、高速な暗号処理の実現が必須である。

車の自動運転支援システムにおいては車車間、路車間と相互通信しなければいけない機器の数が非常に多く、かつ個人につながる情報の漏洩を防ぐため暗号処理にも対応するため、回路規模が小さく、低レイテンシな暗号を実現する必要がある。

車載情報端末においては、ほかにもクラウドを経由した様々なサービス、渋滞予測などの交通情報あるいはコンテンツなども取り扱っており、情報の保護あるいは改ざんの防止が求められる。特に、コンテンツ保護においては高スループットの暗号処理が求められる。

規格関連の状況では、欧州 AUTOSAR では車内通信用にメッセージ認証技術の必要性が述べられている。すなわち、多くの自動車で採用されている CAN に、AUTOSAR 規格の R4.2.2 における Secure Onboard Communication (SecOC) としてカウンタとメッセージ認証コード (MAC) によるメッセージ認証が明記されている [1]。車外通信では、欧州の Car 2 Car Communication Consortium (C2C-CC) による車車間通信へ軽量暗号の活用が期待できる [2]。

2.4 どんな軽量暗号、パラメータを選べばいいか

2.4.1 一般的方針

2.1 節で挙げたように、軽量暗号は既存暗号と比べて実装時に回路規模、消費電力量、レイテンシ、メモリサイズのいずれか、もしくは複数の性能指標において優位性を持つ暗号である。逆にいえば、ある軽量暗号について既存暗号と比べて回路規模については小さく実装可能であるものの、その回路規模が小さな実装では既存暗号の実装より消費電力量は増えている可能性もあり得る。このように軽量暗号は万能のものではない。暗号利用システムにおいて暗号に対する性能指標の要求条件を明らかにすることは困難であることが多いが、やみくもに軽量暗号を利用するのではなく、まずは要求条件をある程度は明らかにしなければならない。その上で、従来暗号、特に CRYPTREC 暗号リストに掲載されている暗号の利用を検討し、要求される性能指標に対して達成困難な場合には軽量暗号の利用を検討するのが望ましい。

軽量暗号を使用するにあたり、秘匿のみが必要な場合、データ認証のみが必要な場合、両方が必要な場合、といった場合が考えられ、目的に応じて方式を選択する必要がある。例えば、ブロック暗号を使用する場合、使用する利用モードにブロック暗号の復号が必要ないのであればその分の実装コストを削減できる。秘密鍵をハードコードするような利用形態であれば、その分の実装コストも削減可能である。利用目的と実装からの制約に照らし合わせ、適した方式を選択することが求められる。

2.4.2 鍵長の選択

鍵長は安全性の基準となる最も重要なパラメータであり、慎重な選択が求められる。ブロック暗号においては一組、あるいは少数の入出力があれば全数探索が可能であり、次節において述べる多くのユースケースにおいても全数探索のシナリオが成立する。例えば、鍵長を 128 ビットから 80 ビットに減らしたとする。回路集積効率に関する 3 年で 4 ($= 2^2$) 倍というムーアの法則が今後も続くとすると、$(128 - 80)/2 \times 3 = 72$ 年寿命が短くなることに注意しなければならない。

2.4.3 ブロック長の選択

ブロック暗号におけるブロック長も安全性に直結する重要なパラメータである。特に、ブロック暗号利用モードや認証暗号にブロック長の短いブロック暗号を使用した場合、安全性が保たれるデータ量に厳しい制限が加わり、このための対策が必要となる。

例として CTR モードの安全性の評価法を紹介する。CTR モードの安全性は、利用するブロック暗号のブロック長を n ビット、同じ鍵のもとブロック暗号が呼ばれる回数を σ とおくと、一様ランダムなビット列との識別が確率 $\sigma^2/2^{n+1}$ 以下であることが示されている（例えば、CRYPTREC 技術報告書 No.2012 (2011/3/4 更新版) 47 頁 [20])。この確率に基づき、システムの利用者のうち一人程度は暗号文とランダムな文字列との識別が可能となるリスクまで受容出来る場合の同一の鍵で処理できる最大データ長を求めると表 2.5 の通りとなる。それほど大きなデータではないことに注意が必要である。

別の注意点として、ブロック暗号を構成要素として用いてハッシュ関数を構成する方法が知られている。これらは十分に長いブロック長を有するブロック暗号を用いた場合に安全性が保たれるものであり、ブロック長の短い軽量ブロック暗号はこのような用途には適さないと考えられる。

2.4.4 処理データ量と鍵更新、その他の対策

利用形態に応じて、例えば秘密鍵を更新可能な環境であれば、頻繁に更新するといった対策が考えられる。あるいは、秘密鍵をハードウェアとしてハードコードするような実装においては秘密鍵を更新できないため、処理するデータ量に制限を

表 2.5　利用者一人の暗号文がランダムな文字列と識別可能となる最大データ長

ブロック長 n (ビット)	利用者数	データ長 $n\sigma$
64	10^3	1.4 Gbyte
	10^6	46.3 Mbyte
	10^9	1.4 Mbyte
48	10^3	4.3 Mbyte
	10^6	139.0 Kbyte
	10^9	4.4 Kbyte

設け、それを超過する前にデバイス自体を破棄するといった運用が考えられる。

鍵更新のタイミングについて考えると、一般的にデータを処理すればするほど暗号方式の安全性は徐々に低下する。このため、任意の攻撃者による攻撃成功確率が十分に許容できるほど小さい範囲にある間に鍵を更新することが望まれる。例えば、[10] にある CMAC では、一般的なアプリケーションにおいて、ブロック長 128 ビットの AES を利用した場合は 2^{48} ブロックのデータ（2^{22}G バイト）を処理する前の鍵更新を、あるいは 64 ビットブロック暗号の TDEA を利用した場合は 2^{21} ブロックのデータ（16M バイト）を処理する前の鍵更新を推奨している。これらの制限により、攻撃者の攻撃成功確率は AES の場合は 10 億分の 1、TDEA の場合は 100 万分の 1 以下となることが期待される。許容できる攻撃成功確率は暗号方式を使用するアプリケーションに依存し、慎重な選択が求められる。

鍵更新の方法についても、アプリケーションに応じてそれぞれに適した方法を選択することが必要である。鍵共有プロトコルを実行できる環境であれば、鍵更新は問題とはならない。あるいは、マスター鍵からセッション鍵を生成し、鍵更新をしながら同期をとるような利用形態が考えられる。

いずれにせよあるタイミングで鍵を更新、あるいは破棄する必要がある。この頻度を遅らせることができるような暗号方式も提案されており、例えば MAC であれば SUM-ECBC [28] や PMAC_Plus [29]、暗号化であれば CENC [14] といった例が挙げられる。これらの方式では、利用するブロック暗号のブロック長が n ビット、ブロック暗号が呼ばれる回数が σ であれば、攻撃者の攻撃成功確率はおおよそ $\sigma^3/2^{2n}$ 以下となる。先述の方式と比べ、64 ビットブロック暗号であれば敵の攻撃成功確率は $\sigma/2^{64}$ 倍小さくなり、攻撃成功確率の閾値に達するまでにより多くのデータを処理することができるようになる。

2.4.5　利用シナリオ

軽量ブロック暗号は一般的に線形攻撃や差分攻撃などの暗号学的な攻撃に対して、十分な耐性を有するように設計されている。これらは軽量ブロック暗号に限らず、通常のブロック暗号についても考慮される安全性である。一方で、通常のブロック暗号では関連鍵攻撃や既知鍵攻撃、選択鍵攻撃といった攻撃者側にとりわけ有利な状況を考え、その安全性を評価することが行われている。実装効率の観点から、軽量ブロック暗号では簡素な設計を採用する方式が多くあり、必ずしもこれらの攻撃に対する安全性が十分ではない、あるいは十分な評価が実施されていない、といったケースが考えられる。関連鍵攻撃や選択鍵攻撃に対する耐性が十分ではないことが分かっている方式も存在し、これらの方式を採用する場合には攻撃シナリオが成立しないような運用が求められる。

2.4.6　その他の留意点

ソフトウェア実装に適した方式やハードウェアに特化した方式など様々な選択肢があり、実装環境に応じて使用する方式を選択する。このとき、暗号学的な攻撃手法のみならず、サイドチャネル攻撃に対する対策の必要性について検討することが重要である。一般的に多くのユースケースにおいてサイドチャネル攻撃が可能な環境が考えられ、実装レベルでの対策の必要性を検討する必要がある。

2.4.7 CRYPTREC 暗号リストの暗号との違い

ブロック暗号を例にして考える。CRYPTREC 暗号リストに掲載されているブロック暗号はブロック長は 64 ビットもしくは 128 ビット、鍵長は 128 ビット以上となっている。

一方、軽量ブロック暗号については、ブロック長は 32 ビット、また鍵長も 80 ビットなど CRYPTREC 暗号リスト掲載の方式より短いものが数多く提案されている。ブロック長と鍵長は安全性に直結するパラメータであり、ブロック長や鍵長を短くすることにより健在化するリスクが利用するシステムにおいて受容可能かどうかを判断しなければならない。

CRYPTREC 暗号リストにある注釈無しの暗号の利用で安全に利用できる範囲であっても小さなパラメータの軽量暗号を採用する場合には上述の例のように利用データ量などについて、再評価する必要がある。

軽量暗号に限らず、CRYPTREC 暗号リストの暗号と同様、無条件に永遠に安全である効率的な暗号はない。利用目的とリスク管理を適切に行ない、従来暗号の利用が困難であるが軽量暗号が使える場面では積極的な利用を推奨する。

2.5 軽量暗号活用例と効果

2.3 節で取り上げた活用例において、軽量暗号、その中でもブロック暗号、あるいはメッセージの改ざんも検出できるような認証暗号を利用する場合において、どのような点に着目して軽量暗号を選定していけばよいかについて本節で例示する。以下、選び方の一例として本ガイドラインの 3 章に記載している軽量暗号の性能比較を元に記載する。軽量暗号を適用する際には 2.4 節に記載されている鍵長の選択などの安全性にも配慮することに留意されたい。

2.5.1 家電・スマートテレビ

スマートテレビのように計算リソースの取り合いが生じる場合、3 章の図 3.40 に基づき、例えばソフトウェアで CPU 使用時間が短く ROM の使用サイズが小さい SPECK、SIMON、Piccolo、TWINE のような軽量暗号を選定することを検討できる。

また、家電におけるデータ保護については、アップデートが可能なソフトウェア処理がメインとなり、それを司る CPU は低機能で安価なものになるであろう。このような CPU でデータ保護を実現するものの一例として、3 章の図 3.34 に基づき SPECK のようなソフトウェアで CPU 使用時間が短いタイプの軽量暗号を選定することも考えられる。

2.5.2 RFID タグ利用のアプリケーション（物流管理等）

RFID では、全体の回路規模に対して暗号機能に使える回路サイズの限界もさることながら、消費電力に大きな制約が存在する。消費電力の低減が可能な暗号方式として、例えば 3 章の図 3.18 から SIMON、SPECK、Piccoro、PRINCE などの軽量暗号を選定することが考えられる。特に 180nm 以上のレガシーのプロセスでは、この差がクリティカルである。40nm 世代のプロセスであっても、50μm 角クラスの極めて小さなチップであれば、この差が搭載可否に影響を与える。

2.5.3 センサーを利用したスマート農業

農作物の生産向上のためには、細かいデータ取得が必要であり、膨大な数のセンサーが必要となる。これらのデータのすべてを暗号化するためには、安価な軽量暗号が望まれる。

また、防災上では、データを暗号化し、メッセージ認証コード（MAC）を付与する必要がある場合も考えられる。認証暗号を実装するにあたり、3 章の図 3.45 によると、ROM サイズの使用量が少なく小型実装に向いている JAMBU-SIMON、SILC-PRESENT、ACORN、Ascon、Minalpher などの軽量暗号の活用が例として考えられる。ブロック暗号の実装を考えるのであれば、例えば 3 章の図 3.42 から AES と比較して 1 回あたりの処理 cycle 数が少ないことから消費電力量が小さく、バッテリー寿命を長くすることができる SPECK、SIMON、PRESENT、TWINE、Midori などの軽量暗号の使用も検討に値する。メッセージが短く秘匿を要しない場合は、4.4 節に記載のメッセージ認証コードの利用、あるいはブロック暗号に軽量暗号を適用し CMAC モードを使用することで小型化が図られる。

2.5.4 医療

医療用センサーデータは個人の情報そのものであり、プライバシー保護の観点からは全てのデータが秘匿されなければならない。現在、特に埋め込み型センサーでは小型化の研究・開発が進められており、nm サイズのデバイスが開発中である。当然、それに付随する部分も小型化が必須である。これらを併せ持つ暗号としては、ハードウェアで省電力の軽量暗号、例えば 3 章の図 3.15 を参考にすると、SIMON、SPECK、Piccolo、PRESENT などが候補となりうる。

また、mHealth の場合は、健常者のウェアラブル端末を利用することが主な想定となっているため、小型化、長時間化ということはあまり大きな問題とはならないが、ウェアラブル端末の CPU の小型化・低廉化のために、軽量暗号が望ましい。

2.5.5 産業用システム

工場などの産業用オートメーションにおいて、フィールドネットワークのオープン化が進む中、EtherCAT などの超高速の産業用オープンネットワークが注目されている。例えば、このネットワークにつながるノードの 1000 点デジタル I/O の読み書きに求められる速度は 30μs であり、1 つのイーサネットフレームでは 1486 バイトまでのプロセスデータを交換できる。この通信路の秘匿と改ざん防止を AES で実装しようとした場合、MAC 検証、復号、（解釈、書き換え）暗号化、MAC 生成で 1 ブロックあたり 4 回の暗号化回路を call する必要がある。AES 1 ブロック暗号化に 100ns、処理速度にして 1.3Gbps かかるとすると 37.2μs 必要になり、AES では厳しい条件となる。これに対して、3 章の図 3.8 によると、例えば軽量暗号の Midori あるいは PRINCE を Unrolled 実装することで回路規模を AES より抑えた上でそれぞれ 3.9Gbps、3.6Gbps の処理速度で演算が出来るため、リアルタイム性が求められる用途での活用が期待される。

2.5.6 自動車

自動車内部にハードウェア実装するための暗号方式を選定する例として、3 章の図 3.15 と図 3.16 を参考に、AES と比較し回路規模を抑えた上で処理速度が高速である Midori、PRINCE、PRESENT、SIMON などの実装が一例として候補としてあげることができる。

車の自動運転支援システムにおける暗号方式としては、レイテンシを低くするため、1 回分のラウンド処理ではなく複数のラウンド処理をハードウェアに実装することになる。このため、例えば 3 章の図 3.7 と図 3.8 を参考に、AES と比較し回路規模を抑えた上で処理速度が高速である Midori、PRINCE、PRESENT、SIMON などの利用も考えられる。

車載情報端末における情報を保護するため、特にコンテンツ保護においては高スループットの暗号処理が求められることから、例えば 3 章の図 3.16 をもとに軽量暗号である Midori の実装も考えられる。

参考文献

[1] AUTOSAR Specification of Secure Onboard Communication, https://www.autosar.org/fileadmin/standards/R4-3/CP/AUTOSAR_SWS_SecureOnboardCommunication.pdf（2023-10-07 閲覧）

[2] CAR 2 CAR Communication Consortium, https://www.car-2-car.org/index.php?id=5（2023-10-07 閲覧）

[3] Bassham, L., Chang, D., Kang, J., Kelsey, J., McKay, K., Turan, M.S., Waller, N.: NIST Lightweight Cryptography Project, https://csrc.nist.gov/projects/lightweight-cryptography

[4] Bernstein, D.J.: CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness, https://competitions.cr.yp.to/caesar.html（2023-10-04 閲覧）

[5] Bozhko, A.: Properties of AEAD algorithms. Internet-Draft draft-irtf-cfrg-aead-properties-01, Internet Engineering Task Force (Mar 2023), https://datatracker.ietf.org/doc/draft-irtf-cfrg-aead-properties/01/, work in Progress

[6] CAST: Ascon-F, Ascon Authenticated Encryption & Hashing Engine, https://www.cast-inc.com/security/encryption-primitives/ascon-f（2023-10-07 閲覧）

[7] Castle, B.: The Legion of the Bouncy Castle, https://www.bouncycastle.org/releasenotes.html#r1rv73（2023-10-07 閲覧）

[8] Cloudflare: CIRCL (Cloudflare Interoperable, Reusable Cryptographic Library), https://github.com/cloudflare/circl（2023-10-07 閲覧）

[9] Dworkin, M.: NIST SP800-38D – Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC (August 2015), https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38d.pdf

[10] Dworkin, M.: Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication (May 2015), https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-38b.pdf

[11] GMO サイバーセキュリティ by イエラエ株式会社: 軽量暗号の評価指標、標準化動向に関する調査（NIST 軽量暗号コンペティションファイナリストなど）（文書番号：CRYPTREC EX-3206-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3206-2022.pdf

[12] GMO サイバーセキュリティ by イエラエ株式会社: 軽量暗号 Ascon などに関わる標準化動向調査（文書番号：CRYPTREC EX-3302-2023）(2023)

[13] Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Miller, G.L. (ed.) Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996. pp. 212–219. ACM (1996), https://doi.org/10.1145/237814.237866

[14] Iwata, T.: New Blockcipher Modes of Operation with Beyond the Birthday Bound Security. In: Robshaw, M.J.B. (ed.) Fast Software Encryption, 13th International Workshop, FSE 2006, Graz, Austria, March 15-17, 2006, Revised Selected Papers. Lecture Notes in Computer Science, vol. 4047, pp. 310–327. Springer (2006), https://doi.org/10.1007/11799313_20

[15] Jokela, P., Moskowitz, R.G., Melén, J.: Using the Encapsulating Security Payload (ESP) Transport Format with the Host Identity Protocol (HIP). RFC 7402, 1–40 (2015), https://doi.org/10.17487/RFC7402

[16] McKay, K.A., Bassham, L., Turan, M.S., Mouha, N.: NISTIR 8114: Report on Lightweight Cryptography, https://nvlpubs.nist.gov/nistpubs/ir/2017/NIST.IR.8114.pdf

[17] Moskowitz, R., Card, S.W., Wiethuechter, A., Gurtov, A.: Secure UAS Network RID and C2 Transport. Internet-Draft draft-moskowitz-drip-secure-nrid-c2-13, Internet Engineering Task Force (Sep 2023), https://datatracker.ietf.org/doc/draft-moskowitz-drip-secure-nrid-c2/13/, work in Progress

[18] National Institute of Standards and Technology: FIPS 180-4 – Secure Hash Standard (SHS) (August 2015), https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf

[19] NIST: Submission Requirements and Evaluation Criteria for the Lightweight Cryptography Standardization Process, https://csrc.nist.gov/CSRC/media/Projects/Lightweight-Cryptography/documents/final-lwc-submission-requirements-august2018.pdf

[20] Phillip Rogaway: Evaluation of some Blockcipher Modes of Operation（文書番号：CRYPTREC EX-2012-2010R1）(2010), https://www.cryptrec.go.jp/exreport/cryptrec-ex-2012-2010r1.pdf

[21] Rambus: Rambus IP Solution Supports New NIST Lightweight Cryptography Algorithm, https://www.rambus.com/blogs/rambus-ip-solution-supports-new-nist-lightweight-cryptography-algorithm/（2023-10-07 閲覧）

[22] Rescorla, E., Tschofenig, H., Modadugu, N.: The Datagram Transport Layer Security (DTLS) Protocol Version 1.3. RFC 9147, 1–61 (2022), https://doi.org/10.17487/RFC9147

[23] Turan, M.S., McKay, K.A., Çalik, Ç., Chang, D., Bassham, L.: NISTIR 8268: Status Report on the First Round of the NIST Lightweight Cryptography Standardization Process, https://nvlpubs.nist.gov/nistpubs/ir/2019/NIST.IR.8268.pdf

[24] Turan, M.S., McKay, K.A., Çalik, Ç., Chang, D., Bassham, L., Kang, J., Kelsey, J.: NISTIR 8369: Status Report on the Second Round of the NIST Lightweight Cryptography Standardization Process, https://nvlpubs.nist.gov/nistpubs/ir/2021/NIST.IR.8369.pdf

[25] Turan, M.S., McKay, K.A., Chang, D., Bassham, L., Kang, J., Waller, N.D., Kelsey, J., Hong, D.: NISTIR 8454: Status Report on the Final Round of the NIST Lightweight Cryptography Standardization Process, https://nvlpubs.nist.gov/nistpubs/ir/2023/NIST.IR.8454.pdf

[26] Westerbaan, B., Wiggers, T.: New Post-Quantum Signatures on the Horizon, https://datatracker.ietf.org/meeting/117/materials/slides-117-tls-new-post-quantum-signature-algorithms-on-the-horizon-00

[27] Xiphera: XIP2201B: Ascon, A Lightweight Cryptographic Suite for AEAD and Hashing, https://xiphera.com/products/pdf/XIP2201B_PB.pdf（2023-10-07 閲覧）

[28] Yasuda, K.: The Sum of CBC MACs Is a Secure PRF. In: Pieprzyk, J. (ed.) Topics in Cryptology - CT-RSA 2010, The Cryptographers' Track at the RSA Conference 2010, San Francisco, CA, USA, March 1-5, 2010. Proceedings. Lecture Notes in Computer Science, vol. 5985, pp. 366–381. Springer (2010), https://doi.org/10.1007/978-3-642-11925-5_25

[29] Yasuda, K.: A New Variant of PMAC: Beyond the Birthday Bound. In: Rogaway, P. (ed.) Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6841, pp. 596–609. Springer (2011), https://doi.org/10.1007/978-3-642-22792-9_34

[30] 総務省: 平成 27 年度版 情報通信白書 (2015), https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h27/pdf/index.html

第 3 章

軽量暗号の実装性能

3.1 節と 3.2 節では、同一プラットフォーム上で、同一の実装者または統一的な実装ポリシーによりハードウェア実装・ソフトウェア実装を行い、統一的な評価環境で性能比較を行った結果を示す。ハードウェア実装についてはブロック暗号を、ソフトウェア実装についてはブロック暗号及び認証暗号を評価対象とした。評価対象としたブロック暗号は表 3.1 に示す 12 種類である。また、評価対象とした認証暗号は表 3.2 に示す 10 種類である。なお、3.1 節と 3.2 節の記載内容は 2016 年度版ガイドライン執筆時点（2017 年 3 月現在）のものであることに留意いただきたい。

表 3.1　評価対象としたブロック暗号

ブロック暗号	ブロック長/鍵長	参照仕様書
AES	128/128	[37]
Camellia	128/128	[1]
CLEFIA	128/128	[45]
TDES	64/168	[4]
LED	64/128	[20]
PRINCE	64/128	[8]
PRESENT	64/80	[28]
Piccolo	64/80	[44]
TWINE	64/80	[46]
Simon	32/64, 64/96, 64/128, 128/128	[6]
Speck	32/64, 64/96, 64/128, 128/128	[6]
Midori	64/128, 128/128	[3]

表 3.2　評価対象とした認証暗号

認証暗号	参照仕様書
ACORN	[49]
AES-GCM	[13]
AES-OTR	[33]
Ascon	[12]
CLOC	[24]
SILC	[23]
JAMBU	[50]
Ketje	[11]
Minalpher	[43]
OCB	[29]

3.3 節では、NIST LWC プロジェクトの最終選考方式として選定された Ascon の実装性能に関する評価結果を紹介する。具体的には、3.3.1 節でサイドチャネル攻撃対策を施していない Ascon のハードウェア実装に関する評価結果、3.3.2 節でサイドチャネル攻撃対策を施していない Ascon のソフトウェア実装に関する評価結果、そして 3.3.3 節で Ascon-128 の物理攻撃耐性を含めた実装性能についてまとめている。これらの Ascon の実装性能については、2022 年度と 2023 年度に公開された CRYPTREC 外部評価報告書 [53, 54] に基づき、2023 年 9 月現在の調査結果を記載している。

3.1 ブロック暗号の実装性能

3.1.1 ハードウェア実装評価

暗号回路の実装方式は用途に応じて様々な形態が考えられるが、本ガイドラインでは図 3.1 記載の「Unrolled 実装」、「Round 実装」、「Serial 実装」の 3 つの基本実装方式を採る。図 3.1 中、Round Function は、各暗号アルゴリズムで規定される基本関数の演算を行う組み合わせ回路を指す。12 種類のアルゴリズムに対するハードウェア実装評価では、暗号化演算のみと暗号化・復号演算の双方を同一のモジュールで実行し、その切り替えは制御信号でのみ行う実装の 2 通りに対して実装評価を行う。なお、本ガイドラインでは前者の実装を「Enc」と後者を「Enc/Dec」と表記する。

ブロック暗号アルゴリズムは一般に鍵スケジューリング機能と、暗号化・復号機能に分割できる。本ガイドラインにおける評価では、前記の両機能を持つ暗号回路を構成する。当該暗号化回路は、鍵スケジューリング機能、暗号化・復号機能ともに同一のクロックで動作する仕様で構成する。また、鍵スケジューリングにレジスタが不要なアルゴリズムでは、当該レジスタを削除して実装する。

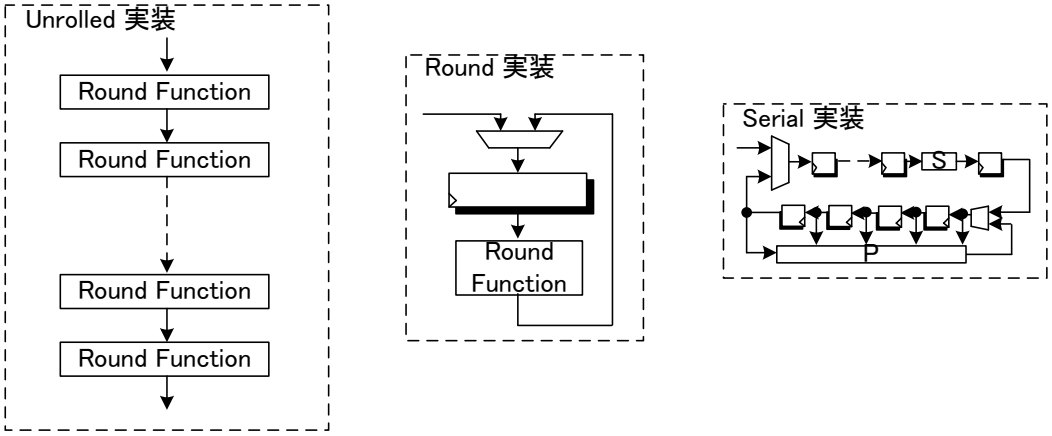


図 3.1　基本実装方式

3.1.1.1 性能比較

以下、表 3.4 から表 3.6 に実装評価結果のデータを示す。また、インターフェースの回路規模を除いた各実装における回路規模の比較結果を表 3.7 に示す。さらに、各実装に対する回路規模、処理速度、ピーク電流、リーク電流についてグラフによる比較結果を図 3.3 から図 3.26 に示す。表中、(comp) はブロック暗号の構成要素である S-box を合成体上の演算回路として実装したことを表し、(table) は S-box をテーブル参照として記述し、合成ツール依存で回路を構成したことを表す。

まず、表 3.4 の注目点について述べる。Unrolled 実装の回路規模は AES を含む CRYPTREC 暗号リスト掲載のブロック暗号が軽量暗号・低遅延暗号（LED は除く）に対して突出して大きい。この要因は表 3.8 に示されるように、S-box の性能差が支配的である。8-bit S-box は遅延優先の Table 実装の場合、PRESENT や PRINCE などの 4-bit S-box に対して 100 倍以上大きくなる一方、遅延は 5 倍程度大きい。仮に 8-bit S-box で 4-bit S-box と同等の遅延性能を達成しようとした場合、4-bit S-box よりもラウンド数を 1/4 程度にすることができないと遅延の観点からは効率が悪く、遅延が同等になっても S-box に関する回路規模は 100 倍近く大きくなることを意味する。この視点から PRINCE はほぼ AES と同じラウンド数となっているため、S-box の性能差がダイレクトに全体性能の差になって表れている。PRINCE は PRESENT に対してラウンド数がおよそ 1/3 であるが、表 3.9 に示すように P 層は PRESENT の方が高速であるため、3 倍の差はなく、回路規模、遅延ともに 2 倍程度 PRINCE が優れる結果となる。復号の影響については、GFN (Generalized Feistel Networks) 型や α-reflection property の効果により、PRINCE、TWINE、Piccolo と PRESENT や LED の回路規模の差はさらに広がる。また、SIMON と Midori については PRINCE と同程度の処理性能を持つ。

Unrolled 実装の遅延についてもう一つ着目すべき点は、AES、LED など復号鍵を一端生成しないと復号できないアルゴリズムや Camellia や CLEFIA のような中間鍵を生成するアルゴリズムはクリティカルパスにその分の遅延が乗るため、

Unrolled 実装では不利になる。Piccolo や PRINCE は暗号化と暗号化・復号回路の最大動作周波数がほぼ同じように構成できる。

次に、round 実装と serial 実装との性能差を比較する。AES では 9kgate 程度の削減が可能であるのに対して、PRESENT、PRINCE は数百 gate から 1kgate の削減に留まる。S-box や P 層などの演算器を削減しても、表 3.8 と表 3.9 からわかるように、その削減効果が限定的であるためである。その一方で、処理性能は 1/10 以下なるため、例えば処理速度 / 回路規模などの指標を導入すれば効率が悪い。コードとしての可読性も悪いため、回路規模に対してなんらかの強い実装制約がない限り、軽量暗号で serial 実装を採用する必要はないであろう。

最後に、serial 実装について述べる。文献 [36] において AES は 2.4kgate で実装されていたが、本実装では 1kgate 程度増加している。その要因は、フリップフロップ 1 つあたりのゲート換算が文献 [36] よりもおよそ 1 ゲート程度大きいことや、合成によって挿入されるバッファ、制御回路の構成などが差分として挙げられる。文献 [36] に記載されるような Scan-FF を積極的に利用するような最適化を実施していないことも差分になる。PRESENT、PRINCE については本ガイドラインの結果でも AES より 1〜2kgate 程度小さい回路となっている。PRESENT と PRINCE との間に回路規模としての差はないが、サイクル数は PRINCE が PRESENT の 1/2 程度で実装できる。

3.1.1.2 評価方法の概要

本節では、評価方法の概要を示す。今回の評価では、各種軽量暗号を、オープンソースの CMOS セルライブラリを利用して、回路リソースの使用量及び最大動作周波数などに関するデータを計測した。実装環境を表 3.3 に示す。

表 3.3　実装環境

論理合成ツール	Design Compiler (Version G-2012.06-SP5)
パワー解析ツール	PrimeTime PX (Version G-2012.06-SP3-2)
合成制約	面積最小
ライブラリ	NANGATE Open Cell Library (45-nm CMOS) `https://www.nangate.com/`
遅延条件	NangateOpenCellLibrary_slow (最悪条件の仮想遅延)

以下に実装する論理回路の機能概略を述べる。

F1. 鍵長は 80bit 以上でかつ規定される最小のパラメータで評価を行う。但し LED に関してはテストベクタが提供されている 128bit 鍵長で評価する。
F2. 暗号化、暗号化・復号回路の実装とする。
F3. CPU のコプロセッサとしての利用を想定し、コンパクトで低電力とされる APB バス [2] 接続が可能な設計とする。

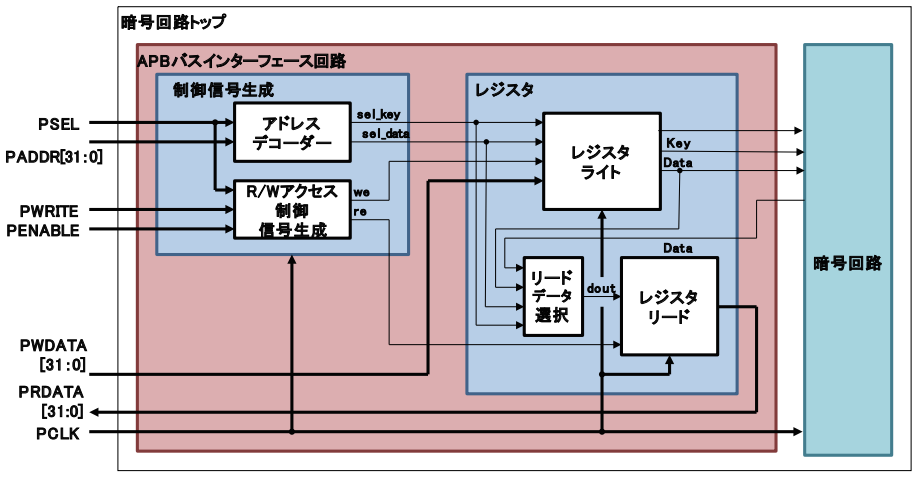


図 3.2　APB バスと暗号回路

APB バスと暗号回路のブロック図を図 3.2 に示す。図中の信号の意味は以下の通りである。

* PCLK: バスクロック信号
* PRESETn: 非同期リセット信号
* PADDR[31:0]: アドレス信号
* PSEL:IP 選択信号
* PENABLE: イネーブル信号
* PWRITE: ライト信号、1: ライト, 0: リード
* PWDATA[31:0]: ライトデータ
* PRDATA[31:0]: リードデータ

このほか、APB 信号規定としては、PSTRB[3:0]（ライトストローブ信号）や PREADY（APB 転送延長信号）があるが、今回の評価では使用していない。

次に設計方針を述べる。

P1. 各アルゴリズムに対して 3 種類の実装を行う: (i) 典型的な round ベースの実装に加え、(ii) 1 サイクルで処理が完了する Unrolled 実装、(iii) データパスを S-box のサイズとする serial 実装を行う。
P2. 鍵スケジュールは on-the-fly で実装する。
P3. CMOS セルライブラリを直接インスタンスするような最適化は行わず、ライブラリ非依存で合成可能な記述とする。

以上の方針に基づき設計した論理回路に対してサイクル数、最大動作周波数（最大遅延)、スループット、ゲート数、ピーク電力、リーク電力を評価している。

表 3.4　Unrolled 実装の評価結果

アルゴリズム	ブロック長 [bit]	鍵長 [bit]	1ブロックあたりの処理サイクル数	最大動作周波数 [MHz]	処理速度 [Gbps]	回路規模 [kgate]	ピーク電力 [mW]	リーク電力 [uW]
Unrolled, Enc								
AES(table)(128/128)	128	128	1	25.7	3.3	112.4	-	-
AES（comp)(128/128)	128	128	1	13.4	1.7	78.8	175.6	939.6
Camellia(comp)(128/128)	128	128	1	7.8	1.0	60.2	136.5	706.7
CLEFIA(128/128)	128	128	1	5.7	0.7	74.6	195.5	891.0
SIMON(128/128)	128	128	1	24.7	3.2	63.2	172.2	685.9
SPECK(128/128)	128	128	1	3.2	0.4	44.4	73.0	417.0
Midori(128/128)	128	128	1	38.5	4.9	34.6	118.2	446.1
TDES(64/168)	64	168	1	10.0	0.6	55.4	111.9	652.2
LED(64/128)	64	128	1	6.9	0.4	74.5	99.1	824.0
PRINCE(64/128)	64	128	1	57.1	3.7	9.8	28.1	107.4
SIMON(64/128)	64	128	1	27.8	1.8	23.8	71.5	260.4
SPECK(64/128)	64	128	1	7.3	0.5	19.5	35.6	183.0
Midori(64/128)	64	128	1	46.5	3.0	12.3	34.9	149.0
SIMON(64/96)	64	96	1	41.3	2.6	20.3	56.7	218.1
SPECK(64/96)	64	96	1	7.6	0.5	18.6	35.4	174.7
PRESENT(64/80)	64	80	1	34.3	2.2	23.9	57.8	259.6
Piccolo(64/80)	64	80	1	18.0	1.2	19.1	61.0	224.8
TWINE(64/80)	64	80	1	24.8	1.6	19.5	43.8	221.2
SIMON(32/64)	32	64	1	39.4	1.3	9.0	30.5	97.4
SPECK(32/64)	32	64	1	15.3	0.5	8.2	17.3	78.0
Unrolled, Enc/Dec								
AES(table)(128/128)	128	128	1	11.4	1.5	208.4	337.2	2612.0
AES（comp)(128/128)	128	128	1	6.4	0.8	144.2	294.3	1734.3
Camellia(comp)(128/128)	128	128	1	7.7	1.0	63.4	133.8	754.9
CLEFIA(128/128)	128	128	1	5.7	0.7	74.3	195.5	891.0
SIMON(128/128)	128	128	1	13.0	1.7	74.1	187.0	803.7
SPECK(128/128)	128	128	1	1.1	0.1	69.1	127.1	672.5
Midori(128/128)	128	128	1	30.7	3.9	55.6	123.7	720.2
TDES(64/168)	64	168	1	9.6	0.6	56.5	112.9	673.3
LED(64/128)	64	128	1	3.1	0.2	215.4	103.1	815.6
PRINCE(64/128)	64	128	1	56.1	3.6	10.1	29.1	108.2
SIMON(64/128)	64	128	1	16.8	1.1	27.5	83.2	299.1
SPECK(64/128)	64	128	1	2.7	0.2	29.9	62.3	290.8
Midori(64/128)	64	128	1	37.7	2.4	20.6	37.1	256.4
SIMON(64/96)	64	96	1	21.5	1.4	23.8	62.9	255.3
SPECK(64/96)	64	96	1	2.9	0.2	28.6	57.8	278.0
PRESENT(64/80)	64	80	1	26.8	1.7	43.8	127.8	505.4
Piccolo(64/80)	64	80	1	16.3	1.0	22.8	64.8	264.0
TWINE(64/80)	64	80	1	13.1	0.8	25.6	50.9	292.2
SIMON(32/64)	32	64	1	23.6	0.8	10.4	30.9	111.8
SPECK(32/64)	32	64	1	6.9	0.2	12.4	27.5	121.7

表 3.5　round 実装の評価結果

アルゴリズム	ブロック長 [bit]	鍵長 [bit]	1ブロックあたりの処理サイクル数	最大動作周波数 [MHz]	処理速度 [Gbps]	回路規模 [kgate]	ピーク電力 [mW]	リーク電力 [uW]
Round, Enc								
AES（comp)(128/128)	128	128	11	108.2	1.259	15.4	36.1	152.6
Camellia(comp)(128/128)	128	128	23	103.0	0.573	10.8	46.6	107.7
CLEFIA(128/128)	128	128	19	145.8	0.982	10.1	39.8	99.6
SIMON(128/128)	128	128	68	371.7	0.700	7.0	17.4	69.9
SPECK(128/128)	128	128	32	50.3	0.201	7.2	11.4	66.2
Midori(128/128)	128	128	20	386.1	2.471	7.1	11.9	79.7
TDES(64/168)	64	168	48	164.2	0.219	7.9	13.9	76.2
LED(64/128)	64	128	48	208.3	0.278	6.3	5.3	52.5
PRINCE(64/128)	64	128	13	234.2	1.153	5.1	16.4	47.1
SIMON(64/128)	64	128	44	371.7	0.541	5.3	12.4	51.1
SPECK(64/128)	64	128	27	95.8	0.227	5.3	10.5	48.3
Midori(64/128)	64	128	16	340.1	1.361	4.7	11.4	49.1
SIMON(64/96)	64	96	42	392.2	0.598	4.5	11.8	44.1
SPECK(64/96)	64	96	26	95.8	0.236	4.6	10.0	42.4
PRESENT(64/80)	64	80	33	326.8	0.634	4.1	4.7	33.4
Piccolo(64/80)	64	80	27	262.5	0.622	3.5	3.4	34.2
TWINE(64/80)	64	80	36	311.5	0.554	4.4	4.6	40.0
SIMON(32/64)	32	64	32	369.0	0.369	2.9	9.8	28.0
SPECK(32/64)	32	64	22	175.1	0.255	2.9	8.4	26.8
Round, Enc/Dec								
AES（comp)(128/128)	128	128	11	107.0	1.245	18.7	44.1	193.6
Camellia(comp)(128/128)	128	128	23	103.0	0.573	11.8	44.6	121.9
CLEFIA(128/128)	128	128	19	143.1	0.964	9.9	38.1	99.0
SIMON(128/128)	128	128	68	310.6	0.585	7.8	17.2	78.4
SPECK(128/128)	128	128	32	49.9	0.200	9.6	11.2	92.7
Midori(128/128)	128	128	20	271.0	1.734	8.4	11.9	96.9
TDES(64/168)	64	168	48	161.6	0.215	10.6	13.9	114.0
LED(64/128)	64	128	48	188.7	0.252	7.2	6.5	66.6
PRINCE(64/80)	64	128	13	224.7	1.106	5.3	18.7	50.3
SIMON(64/128)	64	128	44	342.5	0.498	6.0	12.4	58.2
SPECK(64/128)	64	128	27	93.5	0.222	6.7	10.6	63.2
Midori(64/128)	64	128	16	266.7	1.067	5.3	11.4	57.5
SIMON(64/96)	64	96	42	342.5	0.522	5.1	11.6	49.9
SPECK(64/96)	64	96	26	93.5	0.230	5.9	9.9	55.7
PRESENT(64/80)	64	80	33	280.9	0.545	4.7	4.9	44.8
Piccolo(64/80)	64	80	27	261.8	0.621	3.8	3.3	38.5
TWINE(64/80)	64	80	36	302.1	0.537	4.7	4.5	42.8
SIMON(32/64)	32	64	32	359.7	0.360	3.3	9.9	31.8
SPECK(32/64)	32	64	22	167.5	0.244	3.6	8.7	34.1

表 3.6　serial 実装の評価結果

アルゴリズム	ブロック長 [bit]	鍵長 [bit]	1ブロックあたりの処理サイクル数	最大動作周波数 [MHz]	処理速度 [Gbps]	回路規模 [kgate]	ピーク電力 [mW]	リーク電力 [uW]
Serial, Enc								
AES（comp)(128/128)	128	128	226	112.2	63.6	6.3	18.5	76.8
Camellia(comp)(128/128)	128	128	360	109.5	38.9	6.6	14.4	66.1
CLEFIA(128/128)	128	128	175	114.2	83.5	6.2	13.1	61.3
SIMON(128/128)	128	128	4481	269.5	7.7	4.8	8.2	47.1
SPECK(128/128)	128	128	2177	291.5	17.1	5.0	8.2	48.4
Midori(128/128)	128	128	489	254.5	66.6	4.9	11.9	49.2
LED(64/128)	64	128	1872	344.8	11.8	5.6	2.2	50.0
PRINCE(64/128)	64	128	247	246.3	63.8	3.9	8.7	40.0
SIMON(64/128)	64	128	1537	309.6	12.9	3.7	4.8	36.2
SPECK(64/128)	64	128	993	339.0	21.8	3.9	5.4	37.4
Midori(64/128)	64	128	393	253.2	41.2	3.5	11.4	35.3
SIMON(64/96)	64	96	1441	328.9	14.6	3.3	4.5	31.7
SPECK(64/96)	64	96	929	314.5	21.7	3.4	5.1	33.1
PRESENT(64/80)	64	80	563	186.9	21.2	3.9	3.4	36.4
Piccolo(64/80)	64	80	433	300.3	44.4	3.5	2.0	28.5
TWINE(64/80)	64	80	324	277.8	54.9	4.1	2.8	29.6
SIMON(32/64)	32	64	577	389.1	21.6	2.2	3.7	20.8
SPECK(32/64)	32	64	417	390.6	30.0	2.3	5.5	21.9
Serial, Enc/Dec								
AES（comp)(128/128)	128	128	226	108.6	61.5	7.2	14.5	61.2
Camellia(comp)(128/128)	128	128	360	108.3	38.5	7.3	14.8	63.1
CLEFIA(128/128)	128	128	175	113.1	82.7	6.8	12.5	59.3
SIMON(128/128)	128	128	4481	277.8	7.9	5.6	9.7	57.4
SPECK(128/128)	128	128	2177	316.5	18.6	5.9	8.3	57.2
Midori(128/128)	128	128	489	204.1	53.4	5.3	11.9	53.9
LED(64/128)	64	128	1872	303.0	10.4	6.9	1.4	34.5
PRINCE(64/128)	64	128	247	245.7	63.7	3.8	8.4	36.2
SIMON(64/128)	64	128	1537	277.0	11.5	4.5	5.6	45.3
SPECK(64/128)	64	128	993	317.5	20.5	4.8	7.6	46.2
Midori(64/128)	64	128	393	220.3	35.9	3.8	11.4	37.7
SIMON(64/96)	64	96	1441	298.5	13.3	3.9	5.1	39.0
SPECK(64/96)	64	96	929	280.1	19.3	4.1	7.6	40.1
PRESENT(64/80)	64	80	563	170.9	19.4	4.5	2.4	25.8
Piccolo(64/80)	64	80	433	292.4	43.2	3.7	2.0	23.4
TWINE(64/80)	64	80	324	270.3	53.4	4.2	2.6	28.4
SIMON(32/64)	32	64	577	299.4	16.6	2.6	4.1	25.7
SPECK(32/64)	32	64	417	295.9	22.7	2.8	6.3	27.3

表 3.7　各実装の回路規模

アルゴリズム	インターフェースの除いた暗号回路のみの回路規模[kgate]					
	Unrolled, Enc	Unrolled, Enc/Dec	Round, Enc	Round, Enc/Dec	Serial, Enc	Serial, Enc/Dec
AES(table)(128/128)	109.7	205.6	—	—	—	—
AES（comp)(128/128)	76.1	141.5	12.4	15.6	3.2	4.1
Camellia(comp)(128/128)	57.4	60.6	8.0	9.0	4.1	4.3
CLEFIA(128/128)	71.5	71.5	7.3	7.1	3.6	3.8
SIMON(128/128)	60.4	71.3	4.3	5.0	2.1	2.9
SPECK(128/128)	41.6	66.4	4.4	6.8	2.2	3.1
Midori(128/128)	31.8	52.9	4.3	5.6	2.2	2.6
TDES(64/168)	52.8	53.8	5.3	7.9	—	—
LED(64/128)	71.9	212.9	3.8	4.7	3.0	4.3
PRINCE(64/128)	7.8	8.1	2.7	3.0	1.6	1.8
SIMON(64/128)	21.8	25.4	3.2	3.9	1.7	2.5
SPECK(64/128)	17.4	27.8	3.2	4.6	1.8	2.7
Midori(64/128)	10.2	18.5	2.6	3.2	1.5	1.7
SIMON(64/96)	18.4	21.9	2.7	3.2	1.4	2.0
SPECK(64/96)	16.8	26.8	2.8	4.1	1.6	2.3
PRESENT(64/80)	22.0	42.1	2.2	2.9	2.0	2.8
Piccolo(64/80)	17.4	21.1	1.6	1.9	1.1	1.3
TWINE(64/80)	17.8	23.9	2.7	2.9	2.4	2.5
SIMON(32/64)	7.8	9.2	1.7	2.1	1.0	1.4
SPECK(32/64)	7.0	11.2	1.7	2.4	1.1	1.6

表 3.8　S-box の比較

Module	Area [gate]	Path delay [ns]
AES 8-bit S-box (Table)	3,194	2.43
AES 8=bit S-box (Composite)	315	5.75
PRESENT 4-bit S-box (Table)	26	0.57
PRINCE 4-bit S-box (Table)	18	0.48

表 3.9　P 層の比較

Module	Area [gate]	Path delay [ns]
AES 128-bit permutation	864	0.89
PRESENT 64-bit permutation	0	0
PRINCE 64-bit permutation	192	0.51

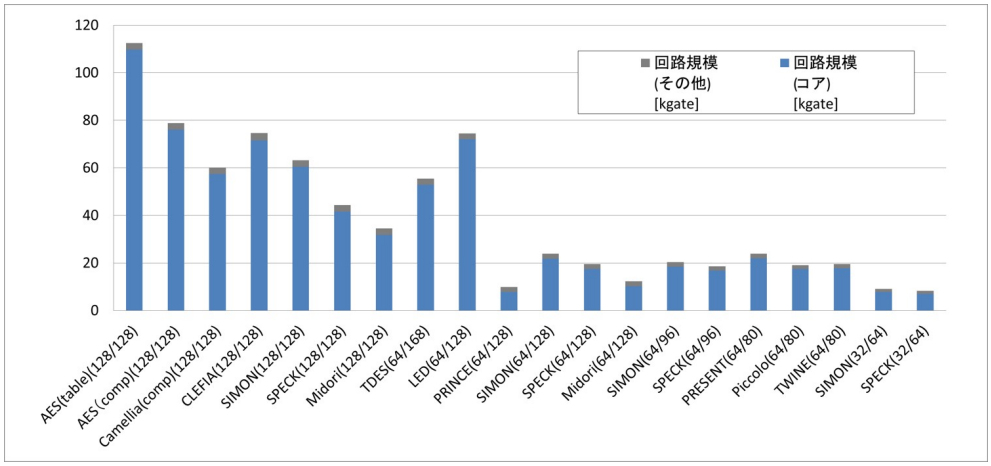


図 3.3　Enc, Unrolled 実装の回路規模

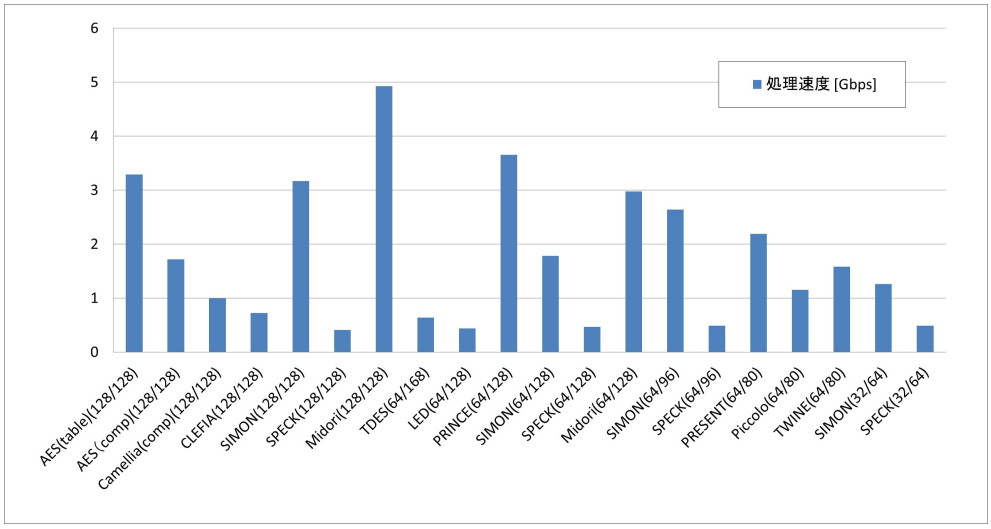


図 3.4　Enc, Unrolled 実装の処理速度

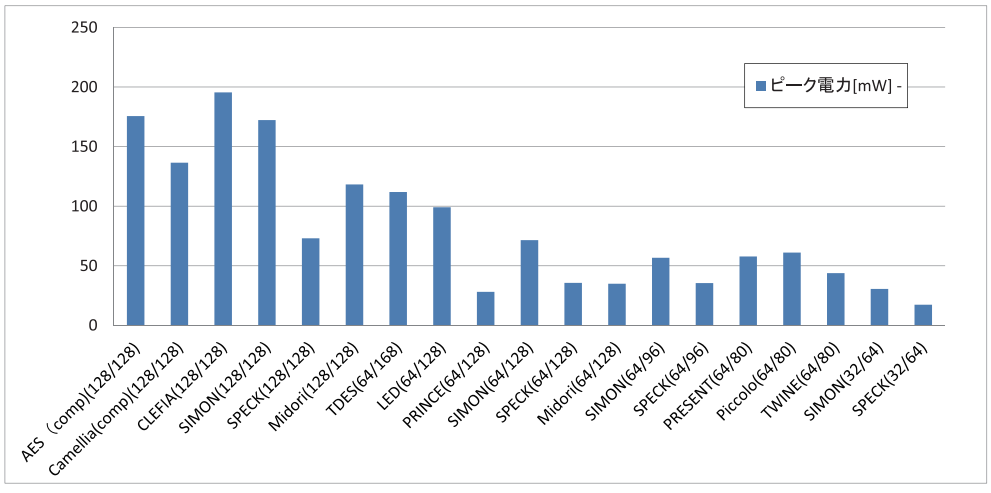


図 3.5　Enc, Unrolled 実装のピーク電流

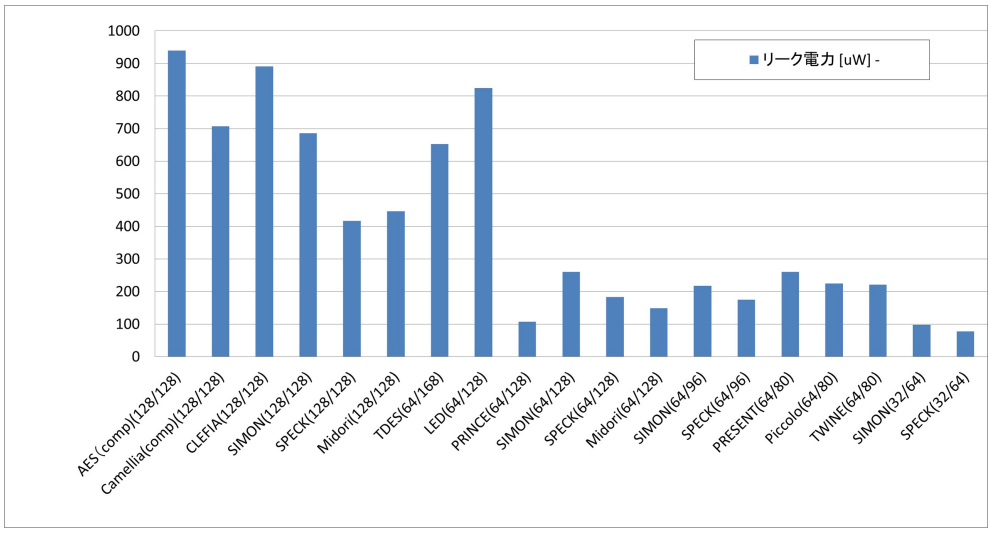


図 3.6　Enc, Unrolled 実装のリーク電流

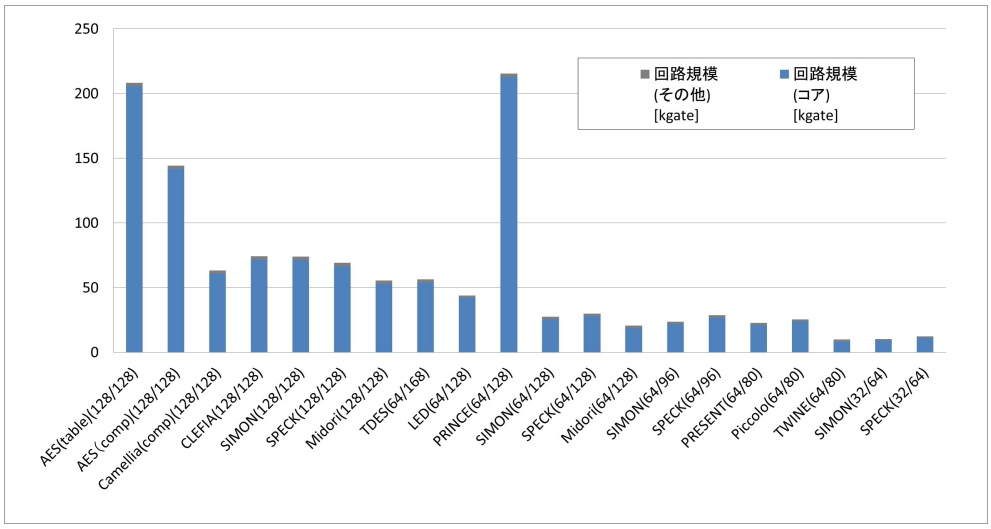


図 3.7　Enc/Dec, Unrolled 実装の回路規模

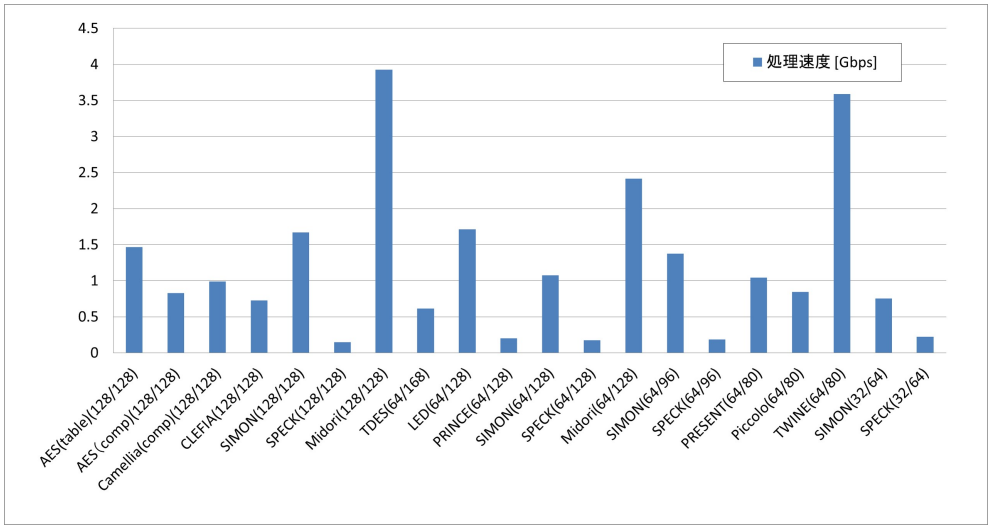


図 3.8　Enc/Dec, Unrolled 実装の処理速度

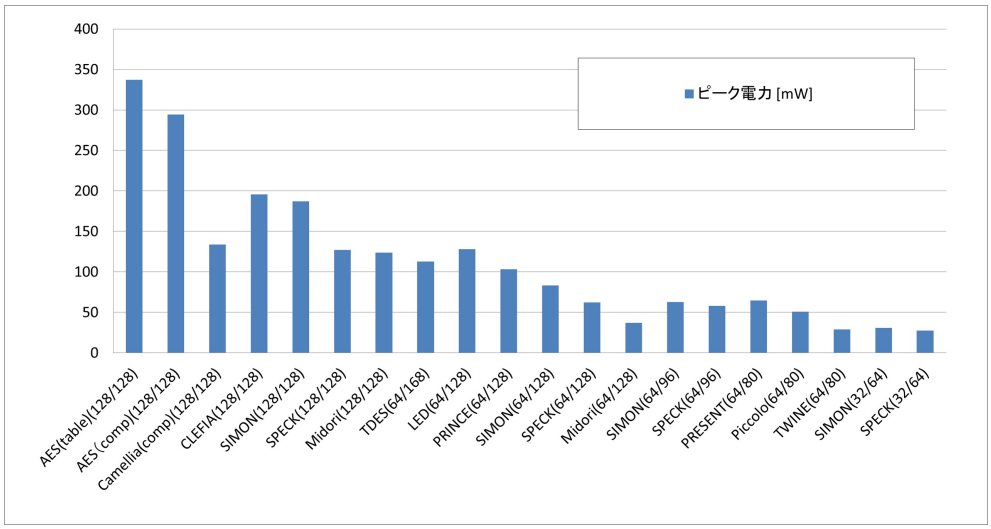


図 3.9　Enc/Dec, Unrolled 実装のピーク電流

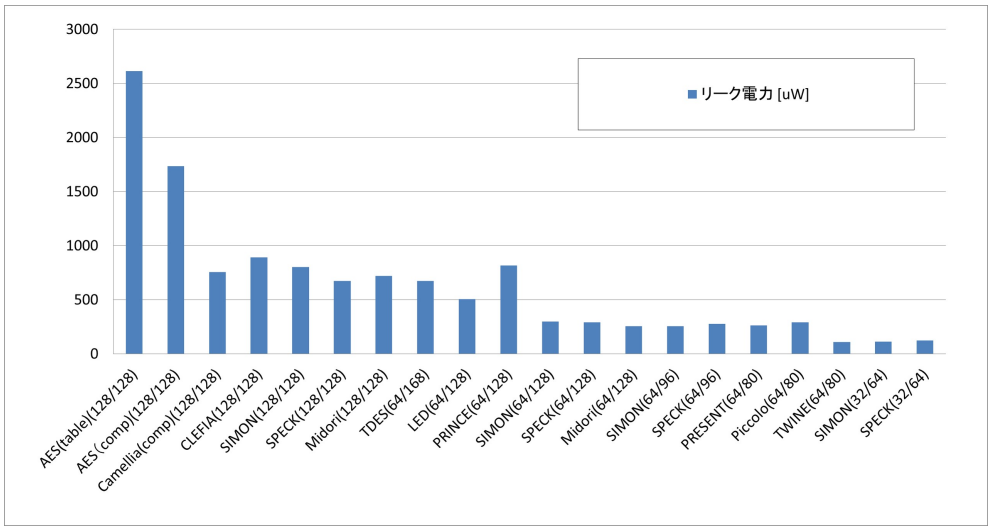


図 3.10　Enc/Dec, Unrolled 実装のリーク電流

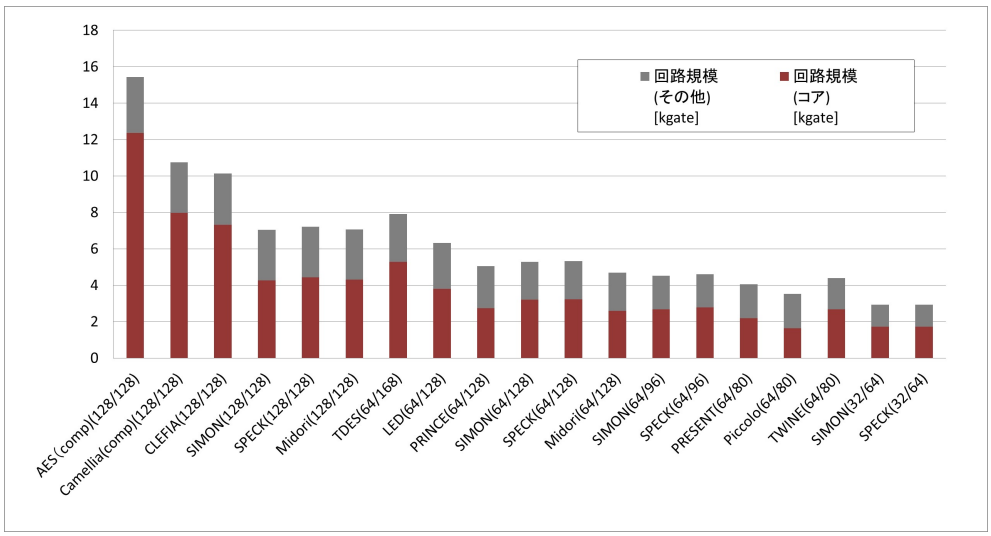


図 3.11　Enc,Round 実装の回路規模

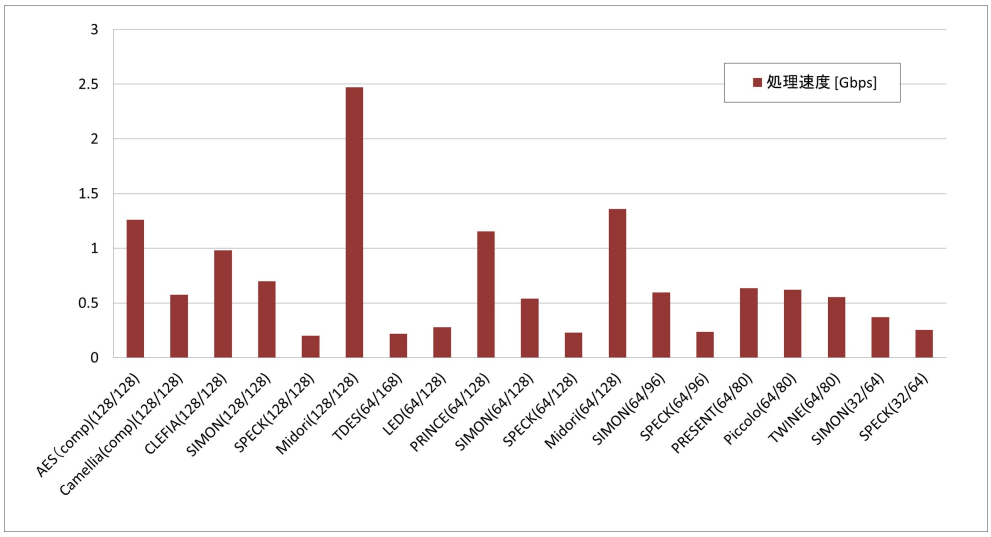


図 3.12　Enc,Round 実装の処理速度

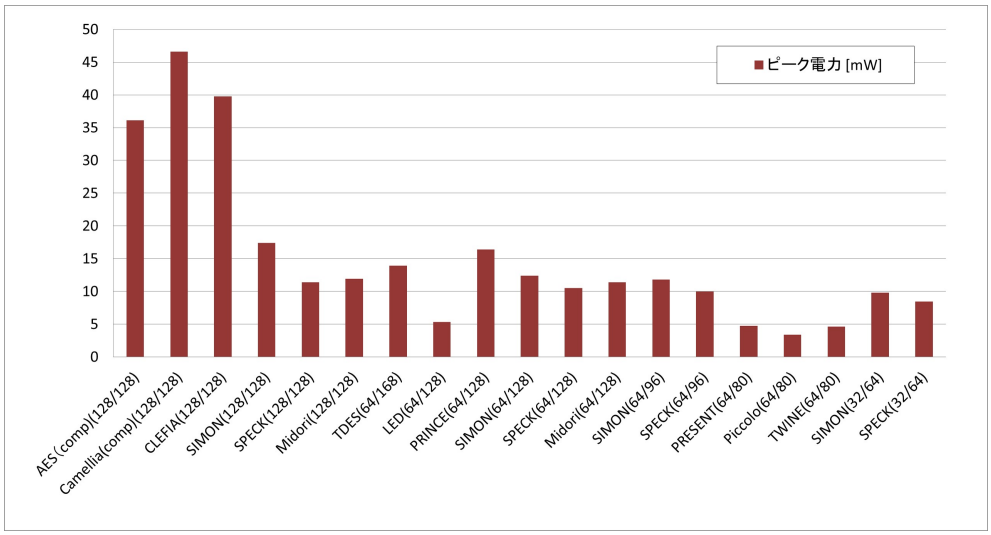


図 3.13　Enc,Round 実装のピーク電流

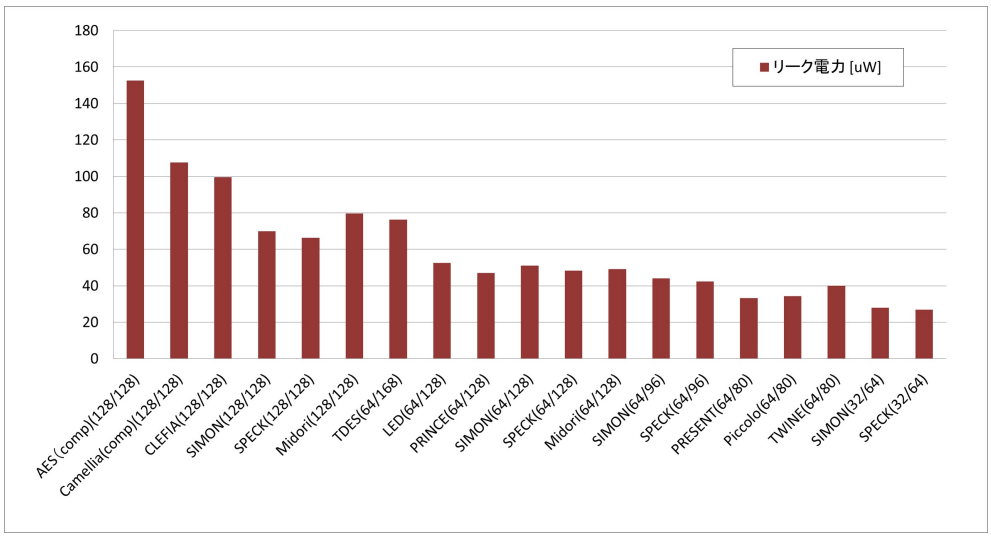


図 3.14　Enc,Round 実装のリーク電流

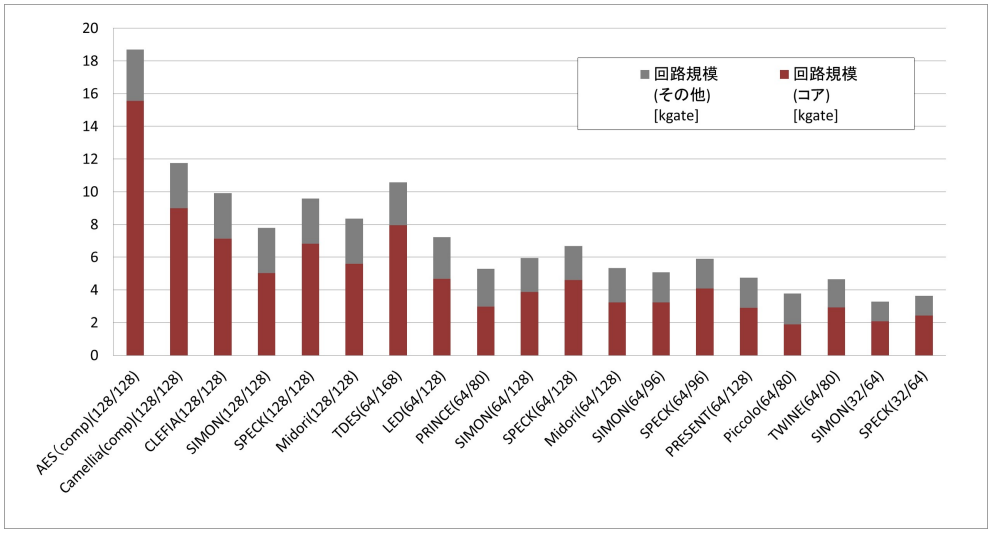


図 3.15　Enc/Dec,Round 実装の回路規模

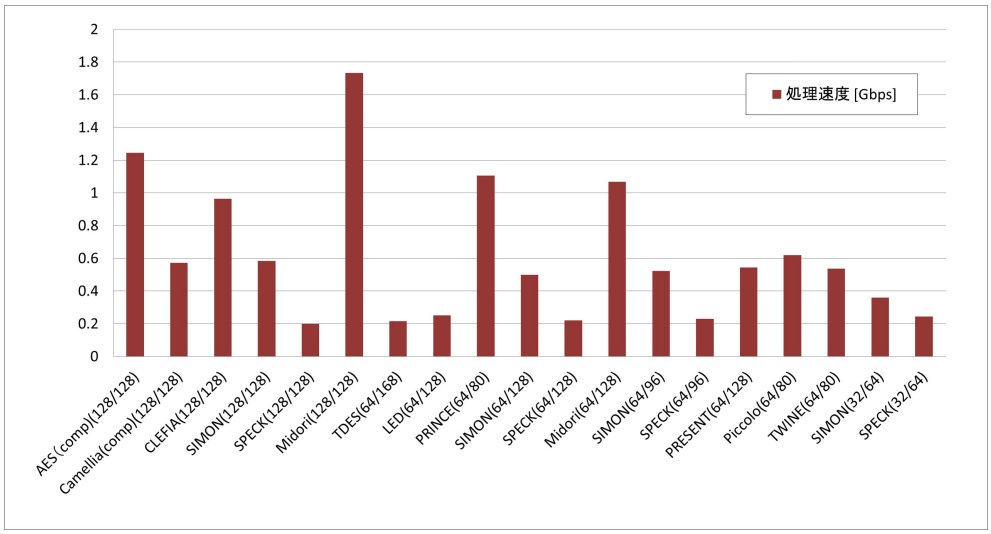


図 3.16　Enc/Dec,Round 実装の処理速度

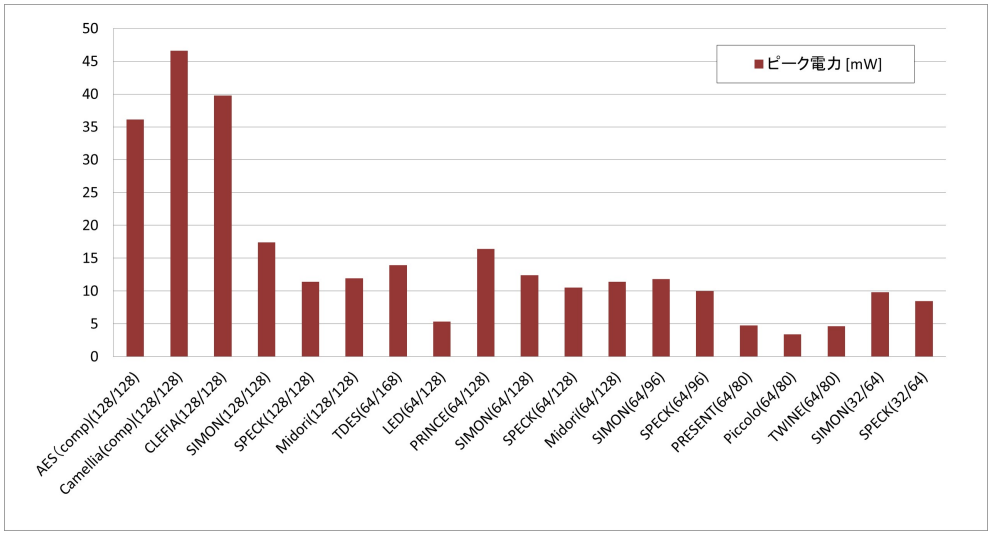


図 3.17　Enc/Dec,Round 実装のピーク電流

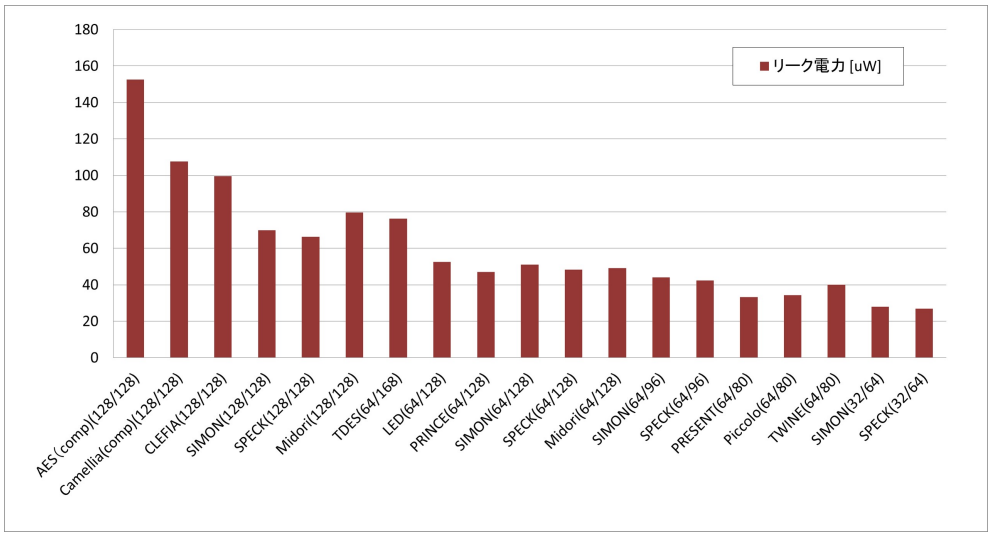


図 3.18　Enc/Dec,Round 実装のリーク電流

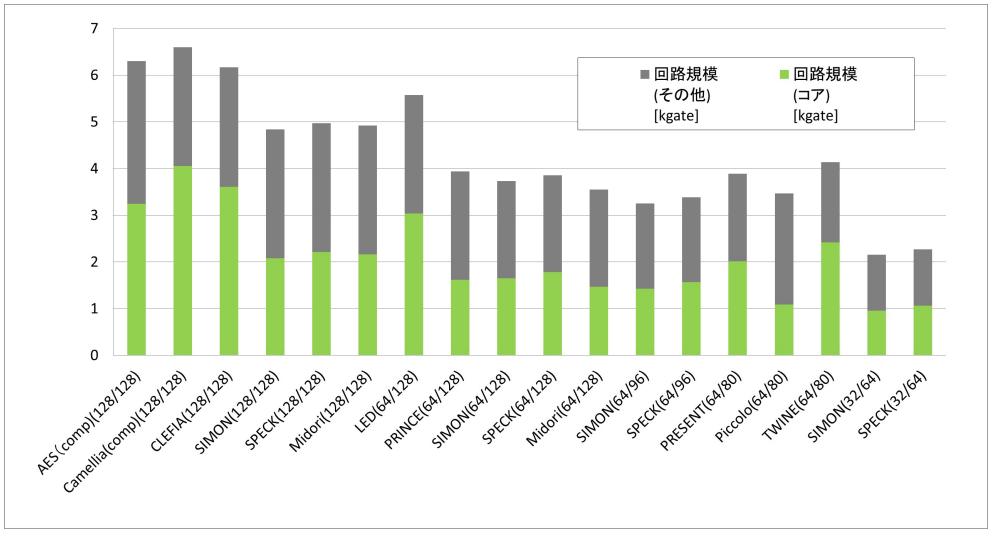


図 3.19　Enc,Serial 実装の回路規模

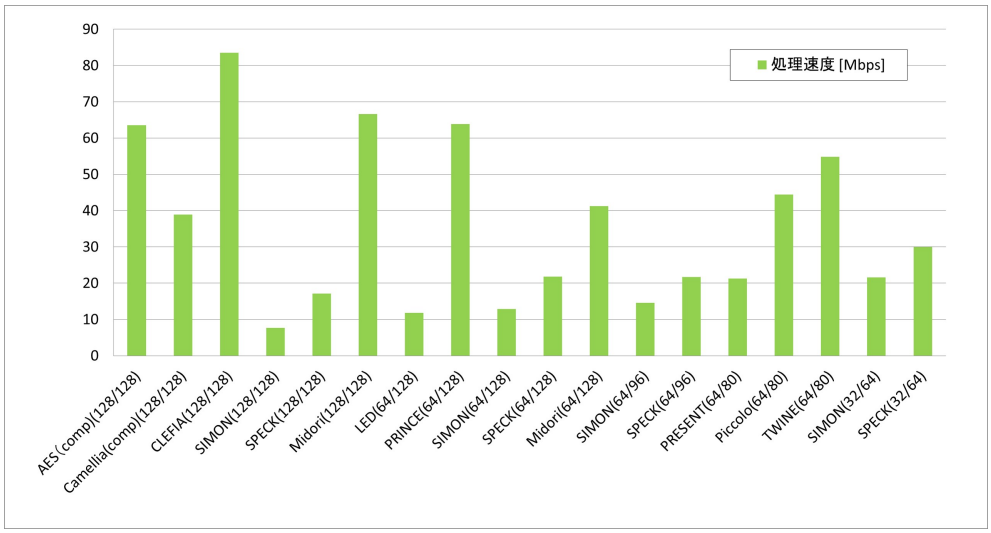


図 3.20　Enc,Serial 実装の処理速度

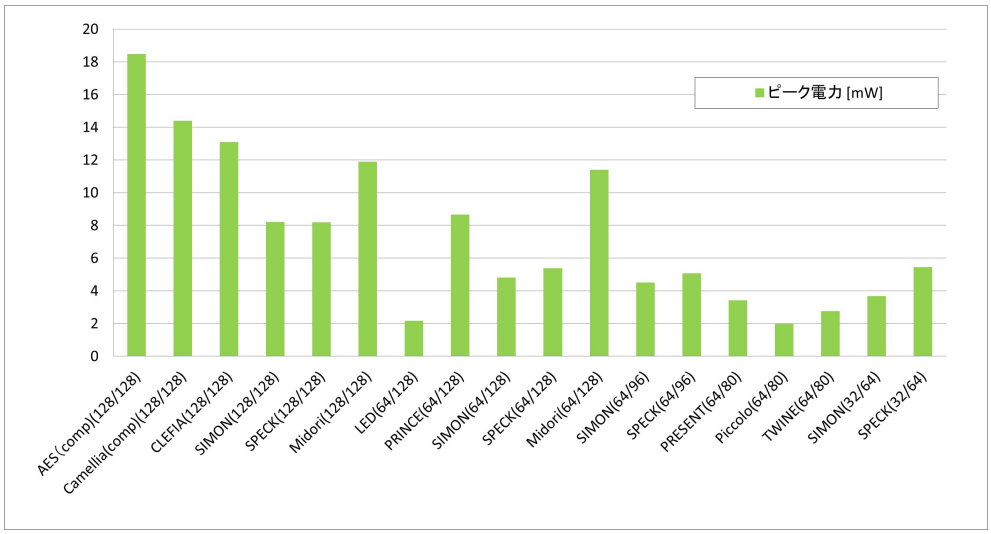


図 3.21　Enc,Serial 実装のピーク電流

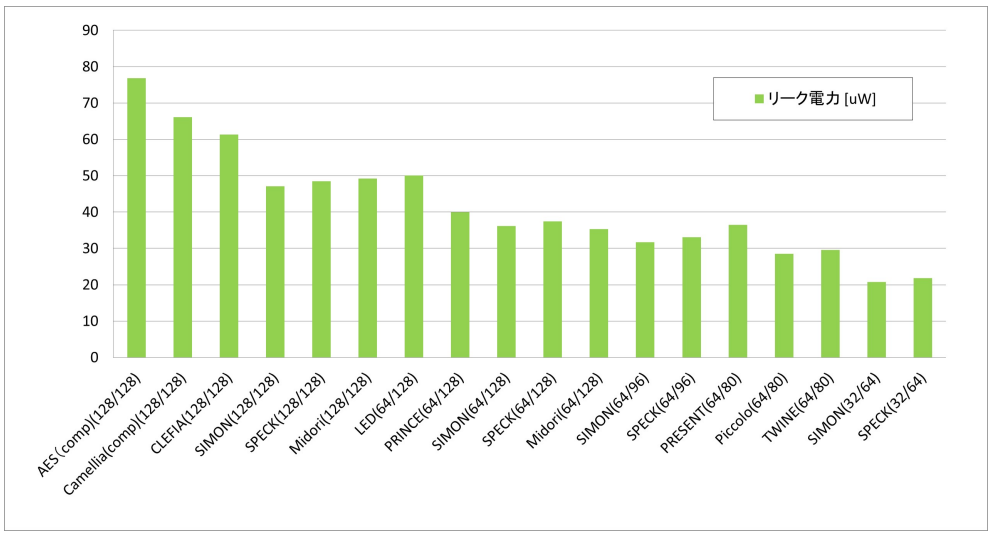


図 3.22　Enc,Serial 実装のリーク電流

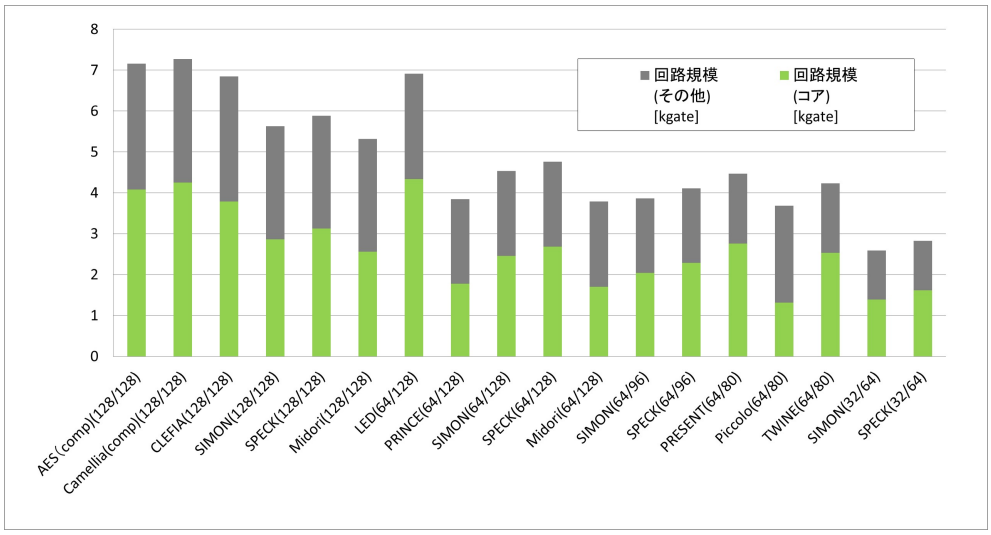


図 3.23　Enc/Dec,Serial 実装の回路規模

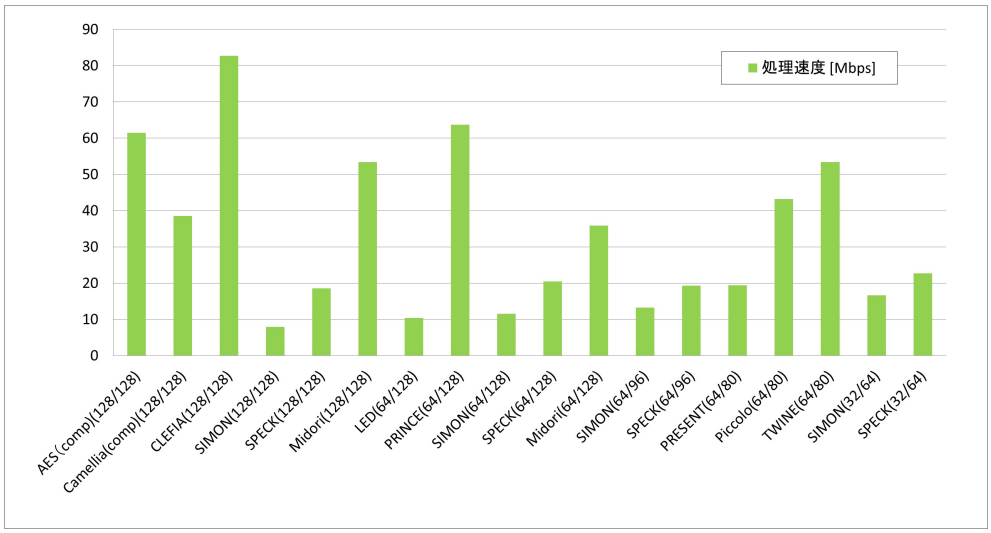


図 3.24　Enc/Dec,Serial 実装の処理速度

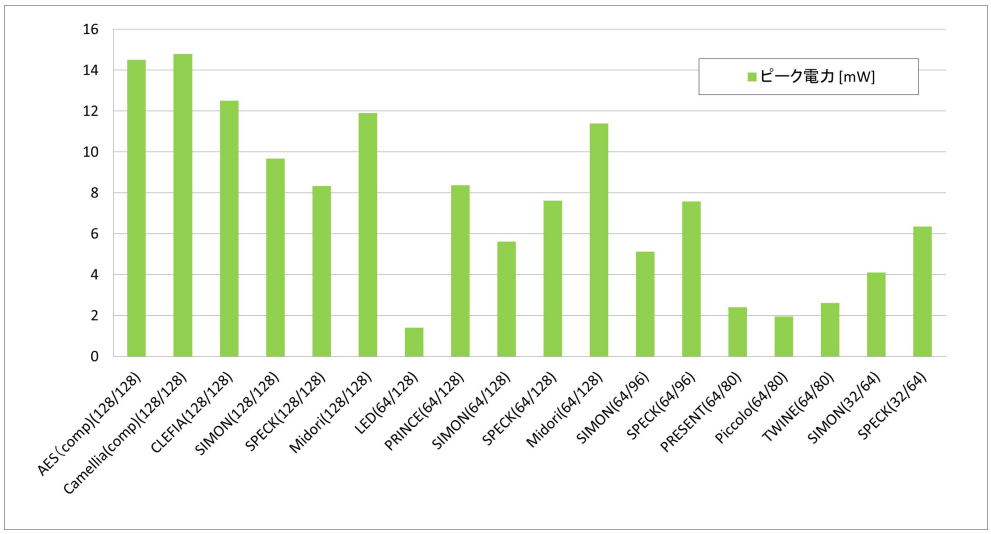


図 3.25　Enc/Dec,Serial 実装のピーク電流

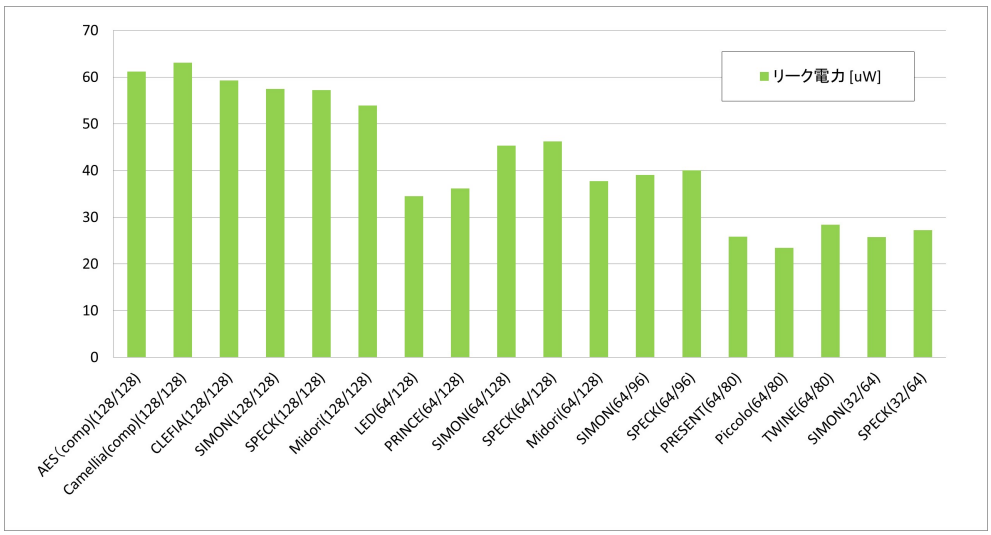


図 3.26　Enc/Dec,Serial 実装のリーク電流

3.1.2 ソフトウェア実装評価

本節では、組み込みマイコン上の限定されたメモリリソースのもとで、軽量ブロック暗号をソフトウェア実装した場合の速度性能について計測、比較した結果を示す。

3.1.2.1 性能評価

本節ではルネサスエレクトロニクス社製 16 ビットマイコン RL78 において、9 つのブロック暗号を実装した結果を示す。表 3.10〜表 3.21 の Category の欄について、ROM-Min は ROM サイズを最小化する実装を行ったもの、(n, m) は ROM サイズが n バイト以下 $(n = 512, 1024)$、RAM サイズが m バイト以下（$m = 64, 128$) で実装を行ったもの、Fast は ROM サイズが約 2K バイト以下で高速化を目指して実装を行ったものである。すべての実装は、暗号化・復号と並行して鍵スケジュールを行う実装方式（on-the-fly key scheduling）を採用している。

■AES

表 3.10 に、RL78 にて AES を実装した結果を示す。暗号化のみを実装する場合、RAM サイズは 64 バイトで十分であったため、RAM サイズ 128 バイトでの実装は省略した。また、暗号化・復号の両方を実装する場合は、S-box だけで ROM サイズを 512 バイト消費するので、ROM サイズ 1024 バイトのカテゴリのみ実装を行っている。

実装方法は、基本的にメモリサイズの制約が厳しくなるにつれ、ラウンド内のループを増やしていくものであるが、参考までにコードサイズに大きく影響する MixColumns の実装方法を一番右の欄に示した。ここで、M4 は MixColumns の行列乗算 4 つを独立にコードとして持っているもの、M1 は行列乗算 1 つだけのコードを持ち、各ラウンド M1 を 4 回ループさせて MixColumns を実行するもの、また MQ は行列の 1 行分だけの演算をするコードを持ち、この二重ループ合計 16 回により 1 ラウンド分の MixColumns 演算をするものである。

ROM 最小化に関しては、文献 [31] で示されている実装よりもさらに小型化されており、暗号化で ROM サイズ 430 バイトは現在知られている最も小さい実装であると思われる。一方、AES の暗号化・復号を実装する場合に ROM サイズ 1024 バイトというのは大きな制約であり、内部でループを多用することによる性能低下が避けられない。実際、ATtiny での既存実装がほとんど ROM サイズ 1500 バイト以上であることも、このような事情が背景にある。高速化に関しては、ROM サイズが 2K バイトあれば、ほぼすべてのループがアンロールできるので、3500 サイクル/ブロック程度が RL78 プロセッサにおける AES の最高性能であると考えられる。

表 3.10　RL78 での AES の実装結果

Algorithm	Category	ROM	RAM static+stack	Enc Speed	Dec Speed	Method
AES (E)	ROM-Min	430	66 + 14	8,753n	–	MQ
AES (E)	(512,64)	510	48 + 10	5,302n	–	M1
AES (E)	(1024,64)	926	48 + 8	3,554n	–	M4
AES (ED)	(1024,64)	1,020	48 + 14	8,193n	9,719n	M1
AES (ED)	(1024,128)	1,020	66 + 14	6,946n	1,380+8,490n	M1
AES (ED)	Fast	2,044	50 + 10	3,554n	753+5,527n	M4

■Camellia

表 3.11 に、RL78 にて Camellia を実装した結果を示す。Camellia には 128 ビットの回転シフト演算が多数含まれているが、その回転数に規則性がないため、ROM サイズが大きくなる。FL 関数や定数 Σ のサイズも小さくなく、S-box を 1 個だけもった場合でも最小 ROM サイズは 749 バイトであった。また、暗号化と復号両方を実装する場合には、コードサイズを減少させるためにサブルーチン化が必要となり、結果として利用するスタックが増加するため RAM サイズ 64 バイトでは実装が困難であった。

一方、ROM サイズが 2K バイトという条件で暗号化だけを行う場合には、AES と同程度の速度が得られている。した

がって、さらに ROM を利用できるならば、復号においては Feistel である Camellia は AES よりも高速となることが期待される。このように Camellia はメモリに比較的余裕がある場合に高性能となる方式である。

これらの実装のうち、ROM-Min 以外で暗号化のみを実装したコードは、必要な回転シフトサブルーチン群を個別に内部で持っているが、暗号化・復号の両方を実装したコードは 8 ビット回転シフトルーチンと 1 ビット回転シフトルーチンだけをコードとしてプログラムの内部に持ち、オンラインで必要なビット数の回転シフトを実現するものである。ROM-Min 実装は 1 ビット回転シフトルーチンだけを内部にコードとして持ち、オンラインで必要なビット数の回転シフトを実現するものである。参考までに、表 3.11 の一番右の欄には、それぞれの実装が何ビットの回転シフトルーチンを独立に持っているかを示した。

表 3.11　RL78 での Camellia の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
Camellia (E)	ROM-Min	749	56 + 16	58,382n	–	1
Camellia (E)	(1024,64)	1,024	54 + 10	889+4,709n	–	17,15
Camellia (E)	(1024,128)	1,018	56 + 14	884+4,520n	–	17,15
Camellia (E)	Fast	1,995	66 + 12	740+3,638n	–	34,30,17,15
Camellia (ED)	(1024,128)	1,021	58 + 22	3,034+25,470n	3,907+25,498n	8,1

■CLEFIA

表 3.12 に、RL78 にて CLEFIA を実装した結果を示す。CLEFIA は鍵スケジュール部において、中間鍵を格納するメモリサイズが多いため RAM サイズ 64 バイトで実装することは困難である。また、S-box や MixColumns が 2 つあることに加え、定数が 384 バイトあるなど ROM サイズも大きく、ROM サイズ 512 バイトで実装することは不可能である。

一方、2 つの S-box のうちの 1 つと定数は実行中に動的に生成することも可能であり、ROM-Min 実装と暗号化・復号両方を実装したものについては、これらを実際動的に生成させている。暗号のみの実装で (1024,128) のものは、定数だけを動的に生成させ、S-box は 2 つ ROM に持つ実装を行っている。この結果、ROM 最小実装でも 800 バイト必要であった。参考までに、表 3.12 の一番右の欄にどの実装方法を採用したかを記載している。S は S-box を動的生成、C は定数を動的生成させたことを意味している。

表 3.12　RL78 での CLEFIA の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
CLEFIA (E)	ROM-Min	800	58 + 22	23,854n	–	SC
CLEFIA (E)	(1024,128)	1,021	58 + 16	12,351n	–	C
CLEFIA (E)	Fast	1,681	74 + 14	3,010+5,899n	–	
CLEFIA (ED)	(1024,128)	1,018	90 + 26	19,879n	20,797n	SC

■TDES

表 3.13 に、RL78 にて TDES を実装した結果を示す。TDES は RAM サイズは 64 バイトで十分であるものの、不規則なビット演算が中心のアルゴリズムであるため、S-box やビット位置を示す表などだけで 400 バイト以上の ROM を占有する。したがって、ROM サイズ 512 バイトで全体を実装することはできない。

ROM サイズ 1832 バイトの実装は、速度にかかわる部分はほぼアンロールしているので、このプロセッサでのほぼ最高性能に近い速度が出ていると考えられる。この結果から、最高性能で比較すると TDES は AES の 1/15 程度の速度性能と言うことができる。

表 3.13　RL78 での TDES の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed
			static+stack		
TDES (E)	ROM-Min	958	50 + 14	111,183n	–
TDES (E)	(1024,64)	1,024	50 + 14	77,708n	–
TDES (E)	Fast	1,832	50 + 8	26,697n	–
TDES (ED)	(1024,64)	1,019	50 + 14	87,879n	87,543n

■LED

表 3.14 に、RL78 にて LED を実装した結果を示す。LED は 4 ビットを 1 ワードとする AES の構造に近い。このような構造の暗号の場合 8 ビットの平文や鍵をどこかの段階で 2 つの 4 ビットデータに分割する必要がある。これをどの段階で行うか（暗号化前におこなっておくか、実行時に分割するか）でメモリサイズと速度のトレードオフが存在する。また、4 ビット S-box をあらかじめ ROM から RAM に転送しておくと、コードサイズと RAM サイズの増加と引き換えに速度が向上するという別のトレードオフがある。これは、RL78 は ROM データの読み出しに 4 サイクルかかるのに対して、RAM データの読み出しは 1 サイクルで済むからである。さらに、GF(16) 上の 2 倍算を実行時に行うのか RAM に搭載した表で行うかのトレードオフも存在する。

このようなさまざまのトレードオフの中で、それぞれの与えられたメモリサイズ条件に対してどれが最も高速になるかは複雑なパズルである。参考までに、表 3.14 の一番右の欄にどの実装を採用したかを示す記号を示した。ここで、S は S-box を RAM 転送していること、G は GF(16) 上の二倍算のテーブルを RAM 転送していること、T は平文（暗号文）を最初に 4 ビット分割していること、K は鍵を最初に 4 ビット分割していることをそれぞれ示している。

なお、暗号化のみの (1024,128) 実装は、主要な部分をすべてアンロールしているものなので、この実装が RL78 での最高速性能に近いと考えられる。

表 3.14　RL78 での LED の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
LED (E)	ROM-Min	298	54 + 12	36,779n	–	T
LED (E)	(512,64)	510	54 + 10	18,055n	–	S
LED (E)	(512,128)	504	100 + 12	17,207n	–	SGTK
LED (E)	(1024,64)	956	54 + 10	15,899n	–	S
LED (E)	(1024,128)	1,023	100 + 8	14,478n	–	SGTK
LED (ED)	(512,64)	508	54 + 10	35,726n	32,219n	T
LED (ED)	(512,128)	508	54 + 14	33,950n	31,787n	T
LED (ED)	(1024,64)	1,007	54 + 10	17,717n	17,788n	S
LED (ED)	(1024,128)	1,023	100 + 8	16,753n	17,472n	SGT

■PRINCE

表 3.15 に、RL78 にて PRINCE を実装した結果を示す。このうち、RAM サイズが 128 バイトの実装はすべて 2 つの S-box の合計 32 バイトを RAM 転送することにより高速化を目指したもので、表 3.15 の一番右に記号 S で示している。また、高速実装のものは S-box2 つを並列化した 256 バイトのテーブルを 2 つ持つことで高速化を目指した実装であり、記号 S8 で示している。

PRINCE は鍵スケジュール処理がほとんどなく、しかも暗号化と復号がほとんど同じ処理で実現できるという特長をもっているが、一方で定数が少なくないことと Matrix 演算のコードのオーバーヘッドから、最小実装の ROM サイズは他の 64 ビット軽量ブロック暗号に比べると大きくなっている。

表 3.15　RL78 での PRINCE の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
PRINCE (E)	ROM-Min	424	42 + 22	9,905n	–	
PRINCE (E)	(512,64)	512	42 + 12	7,611n	–	
PRINCE (E)	(512,128)	511	74 + 12	7,320n	–	S
PRINCE (E)	(1024,64)	1,019	42 + 12	4,928n	–	
PRINCE (E)	(1024,128)	1,020	74 + 12	4,541n	–	S
PRINCE (E)	Fast	1,789	42 + 8	3,307n	–	S8
PRINCE (ED)	(512,64)	511	44 + 20	9,925n	10,050n	
PRINCE (ED)	(512,128)	511	76 + 24	9,541n	9,810n	S
PRINCE (ED)	(1024,64)	1,007	42 + 12	5,117n	5,214n	
PRINCE (ED)	(1024,128)	1,017	74 + 12	4,745n	4,832n	S

■PRESENT

表 3.16 に、RL78 にて PRESENT を実装した結果を示す。PRESENT は規則正しい構造を持つため、この構造を利用して極めて小さいコードを作ることが可能である。暗号化での ROM 最小実装は 164 バイトを達成した。この実装は文献 [40] や文献 [31] で示されている実装よりはるかに小さいものである上、速度的にもこれらの結果よりも優れている。

PRESENT の実装方法は、基本的にはいずれも入力レジスタのデータを 1 ビットシフトし、そのキャリービットを出力レジスタに取り込むという簡単な処理の繰り返しである。RL78 の 16 ビット命令を使うことで、このキャリービットの移動が一命令でできることが小型化に貢献している。

PRESENT もテーブルの作り方、またそれを RAM に転送するかどうかでサイズと速度のトレードオフが存在する。表 3.16 の一番右の欄の Sn-m は、16 バイトのテーブルを ROM に n 個持ち、そのうち m 個を RAM に転送する実装であることを意味している。また、S8 はこのテーブル 2 つを並列に参照する 256 バイトのテーブルを 2 つ持つ実装であることを示している。

暗号化における (1024,64) 実装は 1 段を完全にアンロールしているものであり、RL78 における速度の限界を示していると考えられる。このように PRESENT は速度は遅いものの、メモリサイズの小型化で優位性のあるアルゴリズムである。

表 3.16　RL78 での PRESENT の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
PRESENT (E)	ROM-Min	164	38 + 22	93,412n	–	S1-0
PRESENT (E)	(512,64)	491	44 + 16	11,344n	–	S2-1
PRESENT (E)	(512,128)	499	60 + 16	10,560n	–	S2-2
PRESENT (E)	(1024,64)	952	28 + 10	9,007n	–	S8
PRESENT (ED)	(512,64)	512	42 + 18	16,924n	3,736+19,131n	S2-0
PRESENT (ED)	(512,128)	509	74 + 18	16,407n	3,643+18,614n	S2-2
PRESENT (ED)	(1024,64)	989	38 + 18	12,048n	1,996+12,367n	S4-0
PRESENT (ED)	(1024,128)	1,003	102 + 18	10,691n	1,903+11,010n	S4-4

■Piccolo

表 3.17 に、RL78 にて Piccolo を実装した結果を示す。表 3.17 の右端の見方は PRESENT の場合と同様である。Piccolo は RAM メモリの使用が少なく実装できるため、すべてのカテゴリにおいて 64 バイトの RAM メモリがあれば十分である。

最小実装のサイズでは PRESENT に及ばないものの、全体的に Piccolo は高速に実装できるアルゴリズムと言える。

表 3.17　RL78 での Piccolo の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
Piccolo (E)	ROM-Min	275	24 + 18	12,220n	–	S1-0
Piccolo (E)	(512,64)	498	52 + 8	5,779n	–	S2-2
Piccolo (E)	(1024,64)	1,018	40 + 8	4,961n	–	S8
Piccolo (E)	Fast	1,172	40 + 8	4,636n	–	S8
Piccolo (ED)	(512,64)	512	54 + 8	6,186n	6,084n	S2-2
Piccolo (ED)	(1024,64)	966	52 + 8	5,779n	5,779n	S2-2

■TWINE

表 3.18 に、RL78 にて TWINE を実装した結果を示す。一番右の欄の見方は PRESENT、Piccolo と同様である。TWINE はソフトウエアでオーバーヘッドが少なく、きわめて小型化が可能なアルゴリズムである。また、Piccolo と同じく RAM メモリの使用が少なく実装できるため、すべてのカテゴリにおいて 64 バイトの RAM メモリがあれば十分である。

暗号化のみの場合、ROM サイズが 512 バイト、RAM サイズが 64 バイトですでにほぼ最高速の実装が可能となっている。速度的には、TWINE は Piccolo とほぼ同程度を達成している。

表 3.18　RL78 での TWINE の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
TWINE (E)	ROM-Min	232	52 + 8	11,043n	–	S1-0
TWINE (E)	(512,64)	468	52 + 6	4,957n	–	S1-1
TWINE (ED)	(512,64)	510	54 + 10	6,132n	2,463+5,570n	S1-1
TWINE (ED)	(1024,64)	972	54 + 6	4,957n	1,727+4,892n	S1-1

■SIMON

表 3.19 に、RL78 にて SIMON を実装した結果を示す。小型実装は、共通部分のループ化やサブルーチン化を積極的に行って、できる限り ROM サイズを小さくしたものである。SIMON は Feistel 構造であるため、暗号化モジュールと復号モジュールを両方持つプログラムにおいては、データ暗号化部の暗号/復号共用を行っている。表 3.19 の Category で One と表記する一段実装は、ラウンド関数 1 つの内部をアンロールし、これをラウンド回ループさせた実装を行ったものある。データ暗号化部と鍵スケジュール部の共用は行っていないが、暗号化モジュールと復号モジュールを両方持つプログラムにおいては、初期化と鍵スケジュール部はサブルーチン化して暗号化と復号で共用している。

高速実装は、複数ラウンドをまとめてアンロールし、これを必要回数ループさせる実装を行うとともに、一切の共用化やサブルーチン化を抑制することにより、さらなる高速化を目指したものである。高速実装においてまとめてアンロールする最適なラウンド数は、m = (key size)/(word size) とするとき、SIMON の場合 LCM$(2, m)$、である。これは、SIMON がデータ暗号化部は 2 ラウンド周期、鍵スケジュール部が m ラウンド周期であるためである。

■SPECK

表 3.20 に、RL78 にて SPECK を実装した結果を示す。小型実装は、共通部分のループ化やサブルーチン化を積極的に行って、できる限り ROM サイズを小さくしたものであり、データ暗号化部と鍵スケジュール部の共用を行っている。一段実装は、ラウンド関数 1 つの内部をアンロールし、これをラウンド回ループさせた実装を行ったものある。データ暗号化部と鍵スケジュール部の共用は行っていないが、暗号化モジュールと復号モジュールを両方持つプログラムにおいては、初期化と鍵スケジュール部はサブルーチン化して暗号化と復号で共用している。高速実装は、複数ラウンドをまとめてアンロールし、これを必要回数ループさせる実装を行うとともに、一切の共用化やサブルーチン化を抑制することにより、さらなる

表 3.19　RL78 での SIMON の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed
			static+stack		
SIMON(32/64)(E)	ROM-Min	127	20 + 8	3,706n	–
SIMON(32/64)(E)	One	171	20 + 6	2,480n	–
SIMON(32/64)(E)	Fast	413	20 + 6	1,872n	–
SIMON(64/96)(E)	ROM-Min	112	32 + 8	7,354n	–
SIMON(64/96)(E)	One	243	32 + 6	4,598n	–
SIMON(64/96)(E)	Fast	859	32 + 6	3,450n	–
SIMON(64/128)(E)	ROM-Min	128	40 + 8	9,094n	–
SIMON(64/128)(E)	One	303	40 + 6	6,404n	–
SIMON(64/128)(E)	Fast	753	40 + 6	4,688n	–
SIMON(128/128)(E)	ROM-Min	111	48 + 8	21,050n	–
SIMON(128/128)(E)	One	415	48 + 6	13,148n	–
SIMON(128/128)(E)	Fast	629	48 + 6	10,836n	–
SIMON(32/64)(ED)	ROM-Min	273	20 + 14	4,227n	6,586n
SIMON(32/64)(ED)	One	310	30 + 10	2,777n	4,473n
SIMON(32/64)(ED)	Fast	1,035	20 + 6	1,872n	3,069n
SIMON(64/96)(ED)	ROM-Min	244	32 + 14	8,035n	12,063n
SIMON(64/96)(ED)	One	436	32 + 10	4,985n	7,559n
SIMON(64/96)(ED)	Fast	1,888	32 + 6	3,450n	5,217n
SIMON(64/128)(ED)	ROM-Min	277	40 + 14	9,807n	15,408n
SIMON(64/128)(ED)	One	546	40 + 10	6,809n	11,057n
SIMON(64/128)(ED)	Fast	1,883	40 + 6	4,688n	7,551n
SIMON(128/128)(ED)	ROM-Min	203	48 + 14	22,147n	34,005n
SIMON(128/128)(ED)	One	506	48 + 10	13,767n	21,023n
SIMON(128/128)(ED)	Fast	1,457	48 + 6	10,836n	16,116n

高速化を目指したものである。高速実装においてまとめてアンロールする最適なラウンド数は、$m = (\text{key size})/(\text{word size})$ とするとき、SPECK の場合 $m-1$ である。である。これは SPECK がデータ暗号化部に周期はなく、鍵スケジュール部が $m-1$ ラウンド周期であることから導き出される。

■Midori

表 3.21 に、RL78 にて Midori を実装した結果を示す。Midori64 については 4 ビットの S-box で構成されるため、実装方法は PRESENT などと同様の方針をとる。したがって、一番右の欄の見方は PRESENT、Piccolo、TWINE と同様である。Midori128 について、速度優先の実装については 8 ビット S-Box のテーブル実装とループ展開を行う。また、小型実装においては、Midori128 における 8 ビット S-Box の 4 ビット S-Box から計算処理による作成、関数共通化と関数呼び出しの多用、ループ処理化を行っている。

表 3.20　RL78 での SPECK の実装結果

Algorithm	Category	ROM	RAM static+stack	Enc Speed	Dec Speed
SPECK(32/64)(E)	ROM-Min	96	24 + 8	1,817n	–
SPECK(32/64)(E)	One	115	20 + 6	1,249n	–
SPECK(32/64)(E)	ROM-Min	261	20 + 6	1,006n	–
SPECK(64/96)(E)	ROM-Min	90	44 + 8	6,645n	–
SPECK(64/96)(E)	One	185	32 + 6	2,335n	–
SPECK(64/96)(E)	Fast	308	32 + 6	2,062n	–
SPECK(64/128)(E)	ROM-Min	89	52 + 8	7,448n	–
SPECK(64/128)(E)	One	205	40 + 6	2,644n	–
SPECK(64/128)(E)	Fast	451	40 + 6	2,122n	–
SPECK(128/128)(E)	ROM-Min	71	67 + 8	11,432n	–
SPECK(128/128)(E)	One	205	64 + 6	5,662n	–
SPECK(128/128)(E)	Fast	309	48 + 6	4,793n	–
SPECK(32/64)(ED)	ROM-Min	211	24 + 10	2,308n	3,684n
SPECK(32/64)(ED)	One	283	20 + 6	1,249n	1,918n
SPECK(32/64)(ED)	Fast	623	20 + 6	1,006n	1,392n
SPECK(64/96)(ED)	ROM-Min	211	44 + 10	6,600n	10,837n
SPECK(64/96)(ED)	One	447	32 + 6	2,335n	3,585n
SPECK(64/96)(ED)	Fast	742	32 + 6	2,062n	3,088n
SPECK(64/128)(ED)	ROM-Min	210	52 + 10	7,078n	11,690n
SPECK(64/128)(ED)	One	499	40 + 6	2,644n	4,152n
SPECK(64/128)(ED)	Fast	1,087	40 + 6	2,122n	3,165n
SPECK(128/128)(ED)	ROM-Min	157	67 + 10	11,471n	18,074n
SPECK(128/128)(ED)	One	391	64 + 10	5,702n	8,726n
SPECK(128/128)(ED)	Fast	746	48 + 6	4,793n	7,343n

表 3.21　RL78 での Midori の実装結果

Algorithm	Category	ROM	RAM static+stack	Enc Speed	Dec Speed	Method
Midori64 (E)	Fast	871	64 + 8	6,768n	–	S2-0
Midori64 (E)	ROM-Min	232	96 + 8	16,979n	–	S2-0
Midori64 (ED)	Fast	1,576	64 + 10	6,768n	8,360n	S2-0
Midori64 (ED)	ROM-Min	374	96 + 6	17,867n	27,966n	S2-0
Midori128 (E)	Fast	1,346	64 + 8	9,217n	–	–
Midori128 (E)	ROM-Min	560	64 + 8	31,794n	–	–
Midori128 (ED)	Fast	1,745	64 + 10	9,217n	10,166n	–
Midori128 (ED)	ROM-Min	605	64 + 6	32,495n	45,586n	–

3.1.2.2 性能比較

以下、これまでの実装結果をもとに、評価対象アルゴリズムをいくつかの軸で比較する。

■メモリサイズを限定した実装（暗号化のみ）

図 3.27 は ROM サイズ 1024 バイト以下、RAM サイズ 128 バイト以下の条件で、暗号化のみを実装した場合の速度性能の比較である。図 3.28 は、見やすさのため図 3.27 から TDES を除いたものである。この程度のメモリリソースがある場合には AES が最も高速となり、SPECK がそれに続くとの結果を得た。

図 3.29 は ROM サイズ 1024 バイト以下、RAM サイズ 64 バイト以下の条件で、暗号化のみを実装した場合の速度性能の比較である。図 3.30 は、見やすさのため図 3.29 から TDES を除いたものである。CLEFIA を除けば、ROM サイズ 1024 バイト以下、RAM サイズ 128 バイト以下の場合と同様であるが、CLEFIA だけは RAM64 バイトでの実装が不可能である。これを図では値 0 として示している。

図 3.31 は ROM サイズ 512 バイト以下、RAM サイズ 128 バイト以下の条件で、暗号化のみを実装した場合の速度性能の比較、図 3.32 は ROM サイズ 512 バイト以下、RAM サイズ 64 バイト以下の条件で暗号化のみ実装した場合の速度性能の比較である。ROM サイズが 512 バイト以下になると CLEFIA 以外にも Camellia や TDES も実装が不可能となる。その他のアルゴリズムでは AES、SPECK が依然高速である。

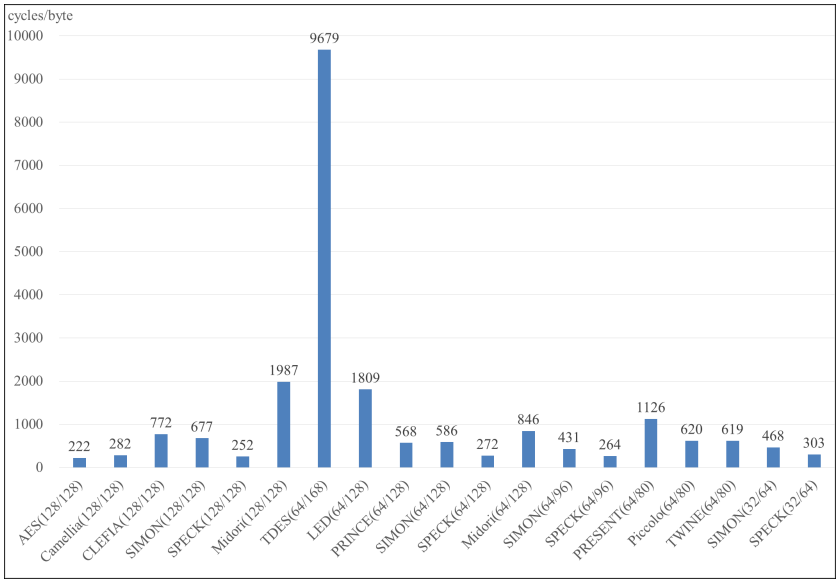


図 3.27　ROM 1024 バイト、RAM 128 バイトでの速度性能

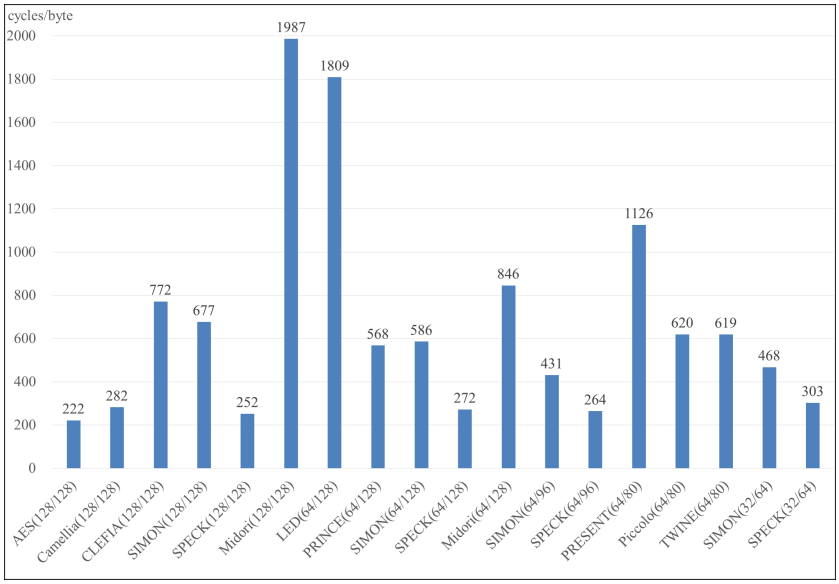


図 3.28　ROM 1024 バイト、RAM 128 バイトでの速度性能（TDES を除いた図）

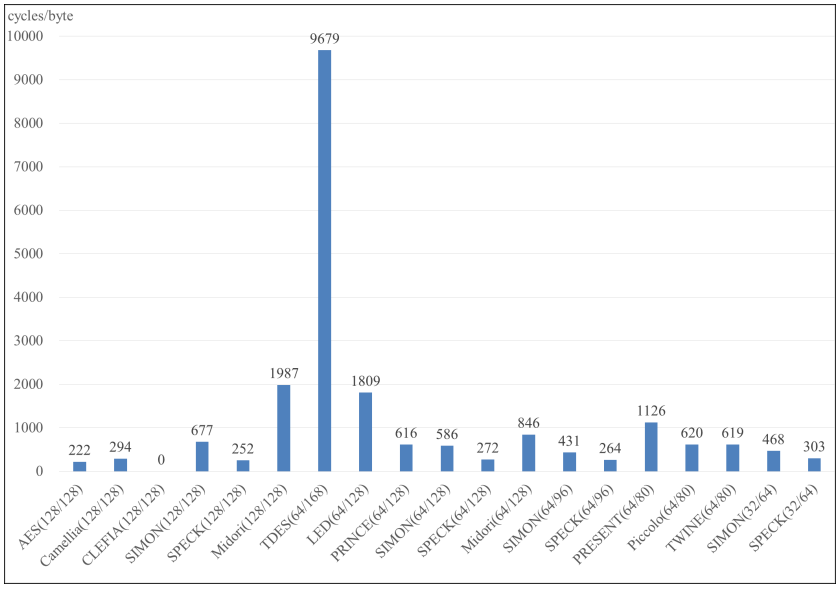


図 3.29　ROM 1024 バイト、RAM 64 バイトでの速度性能

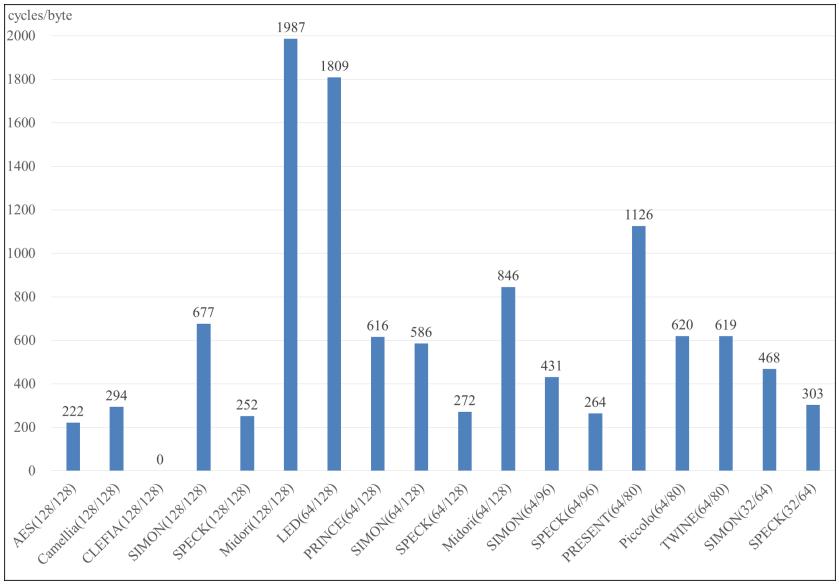


図 3.30　ROM 1024 バイト、RAM 64 バイトでの速度性能（TDES を除いた図）

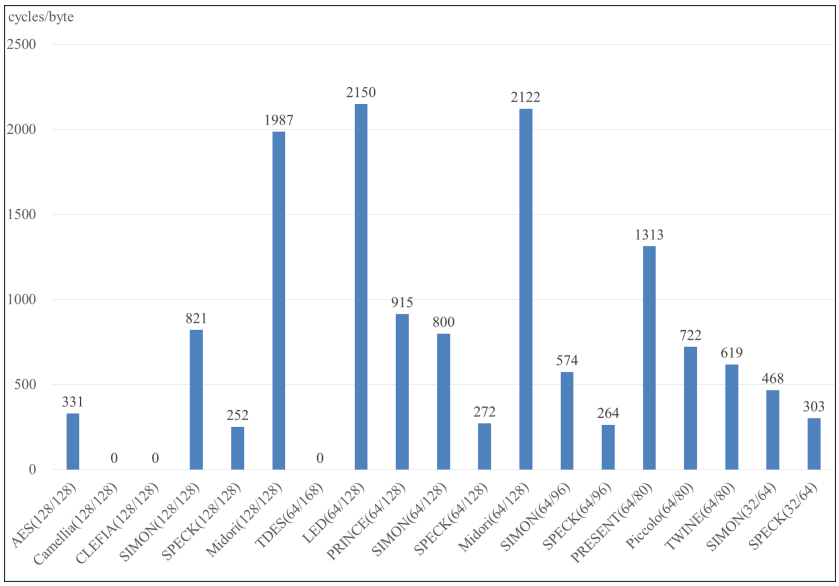


図 3.31　ROM 512 バイト、RAM 128 バイトでの速度性能

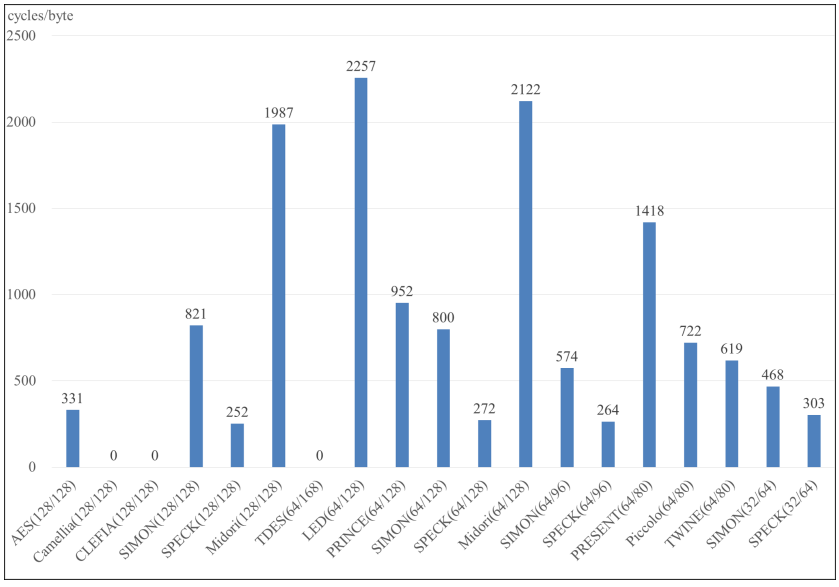


図 3.32　ROM 512 バイト、RAM 64 バイトでの速度性能

■メモリサイズを限定した実装（暗号化・復号）

図 3.33 は ROM サイズ 1024 バイト以下、RAM サイズ 128 バイト以下の条件で、暗号化・復号両方を実装した場合の速度性能の比較である。図 3.34 は、見やすさのため図 3.33 から TDES を除いたものである。ここでも SPECK が最高速であるが、暗号化のみの実装の場合と比べて Piccolo、PRINCE、TWINE との差は縮まっている。

図 3.35 は ROM サイズ 1024 バイト以下、RAM サイズ 64 バイト以下の条件で、暗号化・復号両方を実装した場合の速度性能の比較である。図 3.36 は、見やすさのため図 3.35 から TDES を除いたものである。このカテゴリでは CLEFIA と Camellia、Midori128 が実装不可能となった。また、AES の速度と Piccolo、PRINCE、TWINE との差があまりなくなっている。これは AES の速度がメモリリソース不足のため低下していることを示している。ここでも SPECK が最高速である。

図 3.37 は ROM サイズ 512 バイト以下、RAM サイズ 128 バイト以下の条件で暗号化・復号両方を実装した場合の速度性能の比較、図 3.38 は ROM サイズ 512 バイト以下、RAM サイズ 64 バイト以下の条件で暗号化・復号両方を実装した場合の速度性能の比較である。ここでは AES も実装不可能となり、結果的に 5 つのアルゴリズムだけが生き残るという結果となった。なかでも SPECK は ROM サイズの制限によってほとんど影響を受けない高速性能を示している。

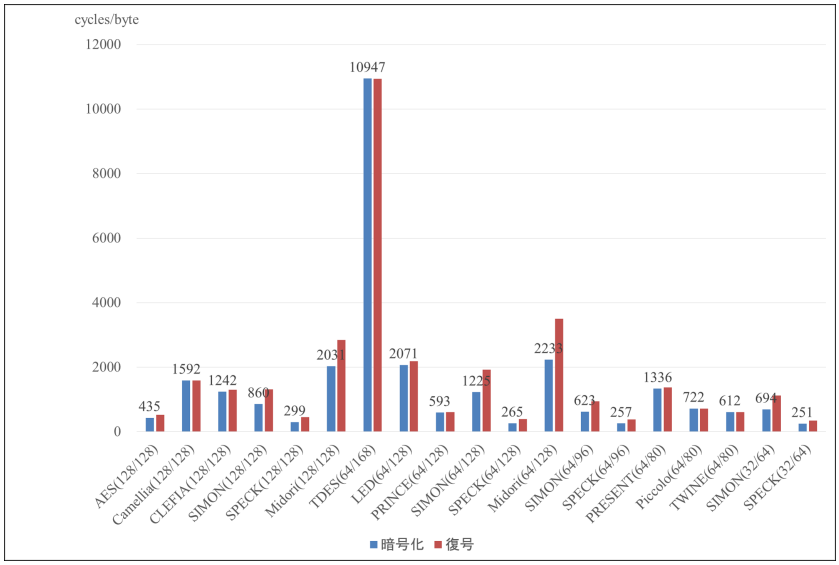


図 3.33　ROM 1024 バイト、RAM 128 バイトでの速度性能

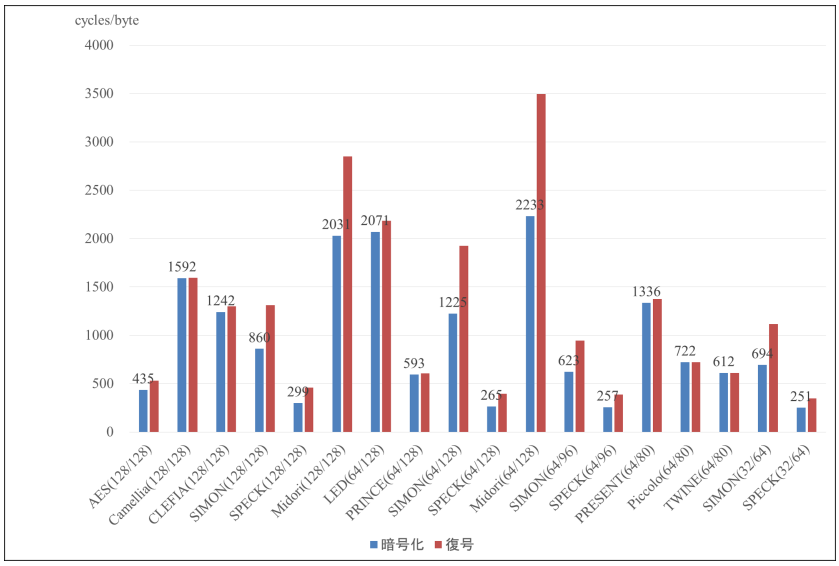


図 3.34　ROM 1024 バイト、RAM 128 バイトでの速度性能（TDES を除いた図）

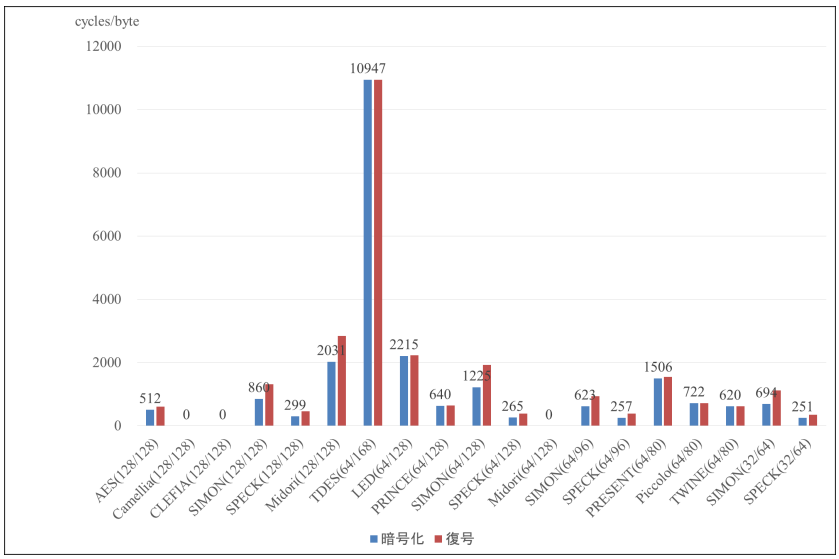


図 3.35　ROM 1024 バイト、RAM 64 バイトでの速度性能

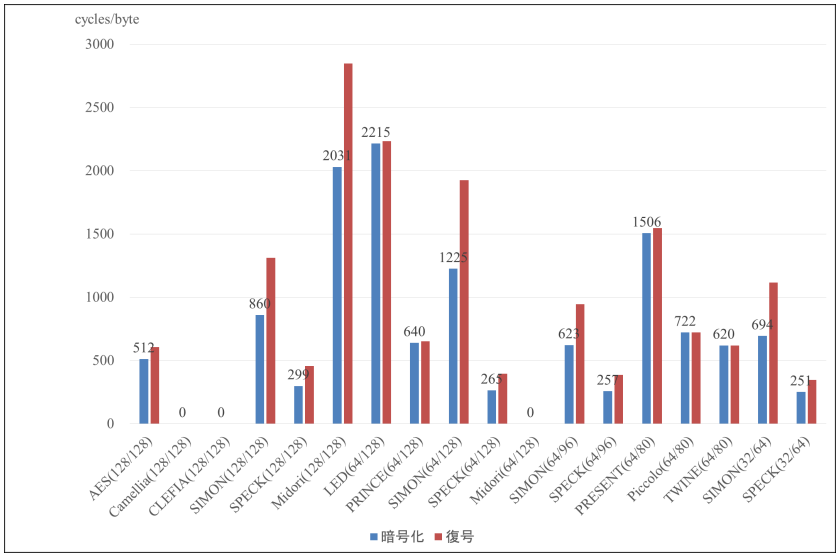


図 3.36　ROM 1024 バイト、RAM 64 バイトでの速度性能（TDES を除いた図）

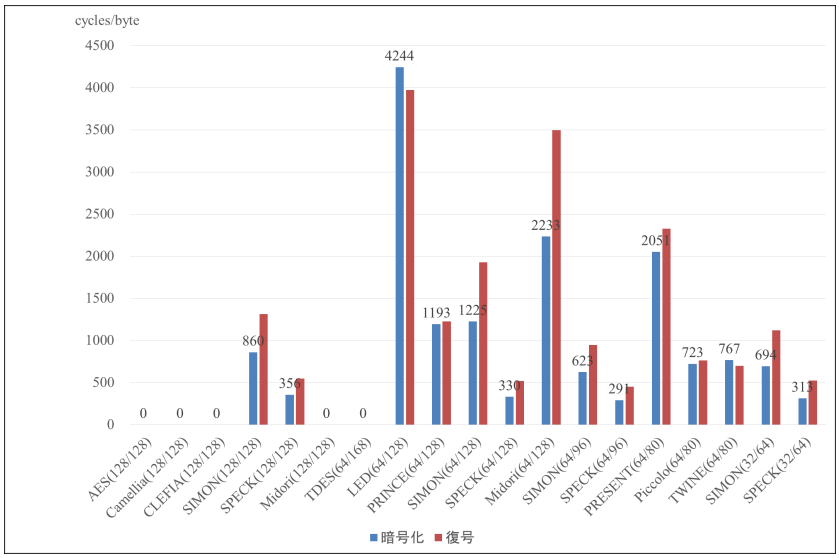


図 3.37　ROM 512 バイト、RAM 128 バイトでの速度性能

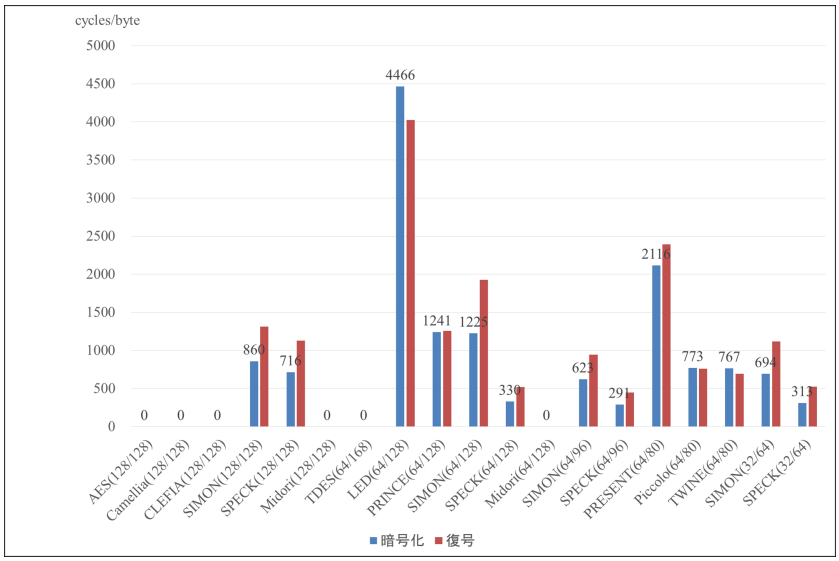


図 3.38　ROM 512 バイト、RAM 64 バイトでの速度性能

■メモリサイズを限定した実装（まとめ）

以上の結果を一つの図にまとめたものを図 3.39 ならびに図 3.40 に示す。後者は前者から TDES を除いたものである。一般的にメモリサイズの制約が厳しくなる右側に行けばいくほど性能は低下する。この性能低下があまり見られない SIMON、SPECK、Piccolo 及び TWINE はメモリサイズにまだ余裕があることを示している。

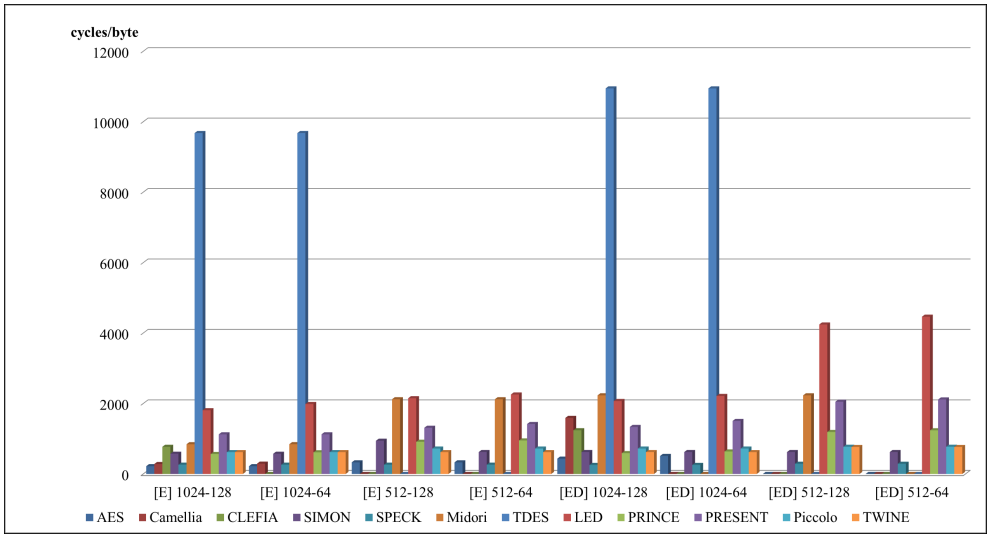


図 3.39　メモリサイズ限定速度性能一覧

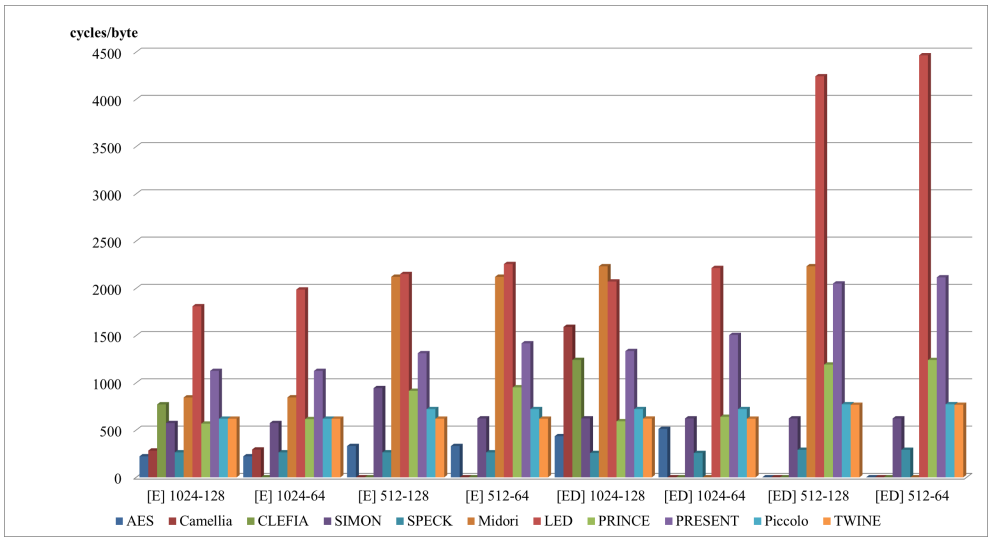


図 3.40　メモリサイズ限定速度性能一覧（TDES を除いた図）

■高速実装

図 3.41 は ROM 2KB 程度で高速実装を目指した場合の速度性能の比較である。これは RL78 プロセッサで達成できる各アルゴリズムの最高性能に近い値と考えられる。この評価では AES、Camellia、SPECK がほぼ同等の性能となることが知られる。

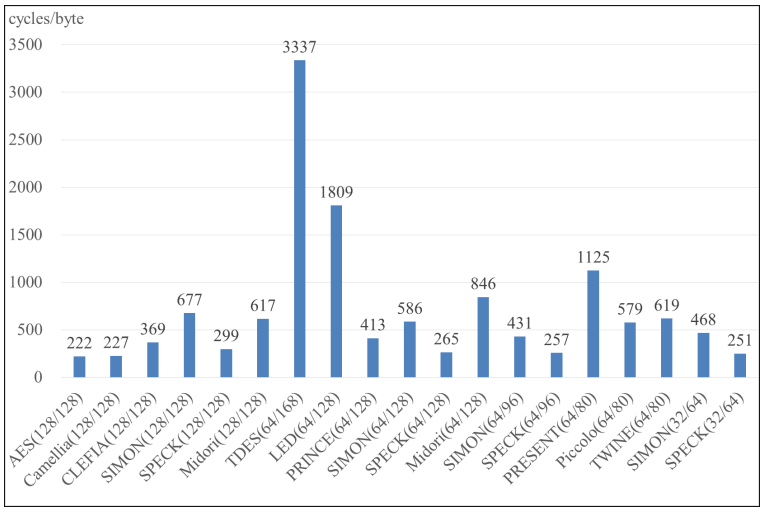


図 3.41　高速実装速度性能一覧

■最小実装

図 3.42 は ROM サイズを最小にする実装（暗号化のみ）において、そのサイズがどこまで小さくなるかを評価したものである。ここでは、最近の軽量ブロック暗号アルゴリズムと、それ以外のものの差がはっきりあらわれている。

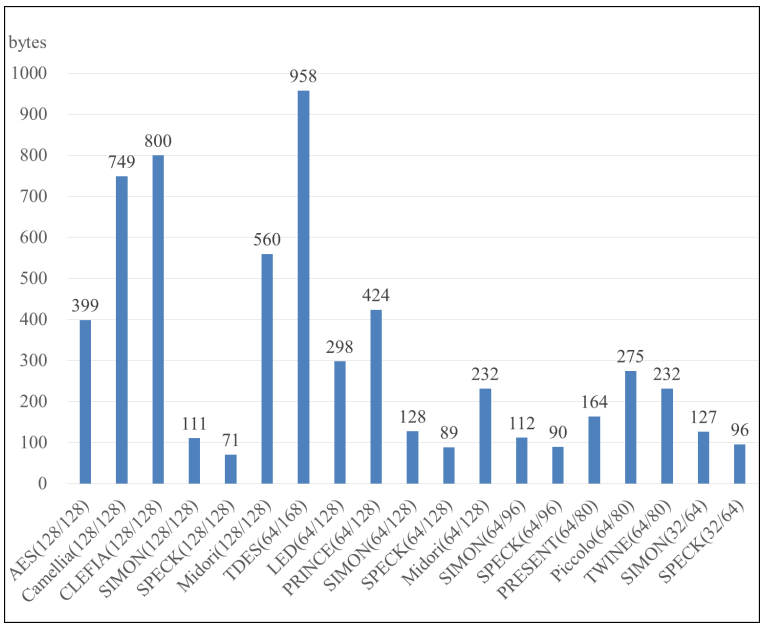


図 3.42　最小実装速度性能一覧

■その他の考察

図 3.43 は暗号化のみを実装したもののうち、ROM サイズが 512 バイト以下になるものについてすべてをプロットしたものである。ここで横軸は ROM サイズ、縦軸は速度を示している。AES については S-box テーブルを持つ限り ROM サイズ 400 バイト前後が限界であり、それ以下の領域は、より小さい S-box を持つアルゴリズムあるいは S-box を持たないアルゴリズムで可能となる。また、この図が示すように、ROM サイズが 200 バイト以下で 2000 サイクル/バイト以下は、SIMON、SPECK だけが達成できる域であり、今後のソフトウエア軽量暗号が目指す一つの方向性を示していると思われる。

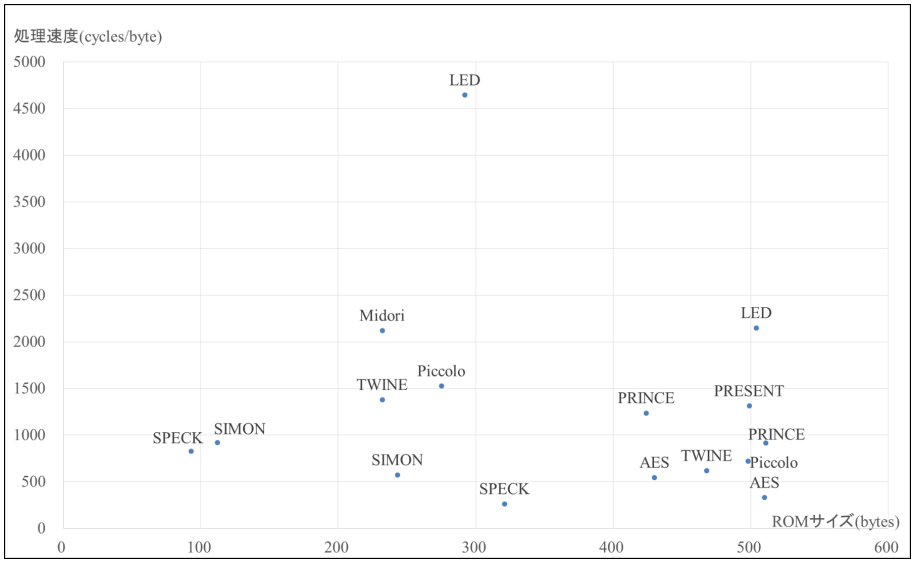


図 3.43　サイズと速度のトレードオフ

一方、今回の実装をアルゴリズム構造の観点から見ると、小型化するにはアルゴリズム全体が数少ない単純な繰り返し構造をもっている必要がある。小型化実装を行う場合、単なるデータの移動や定数も無視できないオーバーヘッドとなる。また、RAM サイズの制約がある場合 on-the-fly で鍵スケジュールを行う必要があるが、アルゴリズムによってはこの鍵スケジュールが小型化のボトルネックになることも少なくない。

さらに、プロセッサ構造の観点から見ると、回転シフト命令の効率はプロセッサに大きく依存すること、最近のプロセッサはほとんどが little endian なので big endian を前提としたアルゴリズムはバイト順の変更に伴うオーバーヘッドが発生する可能性があることなどは、軽量暗号設計において留意すべき点であると言える。

3.1.2.3 評価方法の概要

本節では、軽量ブロック暗号のソフトウェア実装性能評価において採用した評価環境と実装条件について述べる。

■組み込みマイコン RL78 と評価環境

本評価では、ルネサスエレクトロニクス社製の組み込みマイコン RL78 上で評価対象ブロック暗号の実装を行った [10]。RL78 はアキュムレータベースの 16 ビット CISC プロセッサである。その命令セットには多くの 1 バイト命令が存在し、また Load-Modify 命令がサポートされているなど、ROM サイズの小型化に適したプロセッサであると言える。一方、RL78 はすべての命令で 16 ビットデータを扱えるわけではなく、例えばブロック暗号で頻繁に必要となる論理演算や回転シフト演算は 8 ビット命令だけがサポートされている。

RL78 にはいくつかのシリーズが存在するが、例えばそのうち汎用である G1x シリーズのローエンド製品は ROM サイズが 1K バイト、RAM サイズは 128 バイトである。また、車載用に開発された F1x シリーズのローエンド製品は ROM サイズ 8K バイト、RAM サイズ 512 バイトを搭載している。

RL78 の命令セットはシリーズ共通であり（ただし乗算命令のサポートの有無は品種により異なる）、その命令長も同じである。したがって、乗算命令以外の汎用命令を使用する限り、RL78 のコードはすべての品種で動作し、またそのコードが占有するメモリサイズは同じである。実行速度については一部採用するハードウェアコア（S1、S2、S3 の 3 種類が存在する [10]）によって若干異なるが、ここではほとんどの品種で利用されている S2 コアでの速度を計測した。

なお、開発環境はルネサスエレクトロニクス社製の CubeSuite+ を使用した。

■実装条件

さまざまなメモリリソース環境での性能を評価するため、ROM サイズは 512 バイトと 1024 バイトの 2 種類、RAM サイズは 64 バイトと 128 バイトの 2 種類の合計 4 通りのメモリ制約条件のもので、暗号化機能だけを実装した場合と、暗号化機能と復号機能の両方を実装した場合の速度性能を調べた。さらにこれに加え、速度性能は考慮せずに ROM サイズを最小化する実装や、逆に ROM サイズを 2K バイト程度まで許した場合に速度がどこまで向上するかなどの観点でも評価を行った。

評価対象とするブロック暗号アルゴリズムによっては、特定のメモリ制約条件ではそもそも実装が不可能なものや、逆に少ないリソースで充分な性能が出ており、それ以上のメモリ容量が与えられてもそれ以上の性能向上が認められないものも存在する。そのような場合は個別の実装を省略した。なお、RAM サイズが最大 128 バイトでの評価であるので、結果的にすべての評価対象アルゴリズムで、鍵スケジュールは on-the-fly 実装になっている。

本ライブラリのプログラムインターフェースや ROM、RAM サイズの計算方法は文献 [31] と同様である。すなわち、プログラムはアセンブリ言語で記述し、1 ブロックのデータを暗号化および復号する機能を持つ。C 言語から呼び出し可能なサブルーチン形式とし、このサブルーチンは引数を一つだけ取り、その引数が指すアドレスに、平文、暗号文、鍵、一時利用データを格納する。また、RAM メモリ最小化ならびに実用的な観点から、以下の条件での実装を行っている。

- 平文と暗号文の領域は共用する。
- 鍵の領域はサブルーチン呼び出し時と終了時で同じ内容とする（ただし実行中に一時的に変更してもよい）。
- システムが主に利用するゼロページ領域（絶対アドレス 256 未満の領域）は利用しない。
- リロケータブルなプログラムとする（絶対アドレスをハードコーディングしない）。
- システムに依存するコーディングは行わない（例えばレジスタバンクの切り替えをしない、特殊レジスタを直接操作することはしないなど）。

ROM、RAM サイズの計算には、このサブルーチンを実行するために必要なすべてのリソースを含めている。具体的には、ROM サイズはコードならびに固定データテーブルを含み、RAM サイズは、平文（暗号文と領域を共用する）・鍵・一時データ・スタックをすべて含んでいる。したがって、例えば 128 ビットブロック、128 ビット鍵のブロック暗号においては、平文と鍵の領域だけで 32 バイトを占有する。また、関数呼び出し時に必要となるスタックフレームは 6 バイト（call 命令 4 バイト +callee save レジスタの保存 2 バイト）であるので、例えば RAM サイズ 64 バイト以下で実装する場合、自由に使える RAM はスタックを含めて 26 バイトしか存在せず、かなり厳しい制約となる。

3.2 認証暗号の実装性能

3.2.1 ソフトウェア実装評価

本節では、組み込みマイコン上の限定されたメモリリソースのもとで、認証暗号をソフトウェア実装した場合の速度性能について計測、比較した結果を示す。評価対象とした認証暗号を図 3.44 に示す。

	セキュリティレベル	鍵長	ブロック長	ナンス長	タグ長
CLOC-TWINE	32	80	64	48	32
SILC-PRESENT	32	80	64	48	32
JAMBU-SIMON	48	96	96	48	48
CLOC-AES	64	128	128	96	64
SILC-AES	64	128	128	96	64
AES-OTR	64	128	128	96	128
AES-OCB	64	128	128	96	128
JAMBU-AES	64	128	128	96	128
AES-GCM	64	128	128	96	128
ACORN	128	128	128	128	128
Minalpher	128	128	256	128	128
Ketje-SR	128	128	32	128	128
Ascon	128	128	128	128	128

図 3.44　評価対象暗号パラメータ比較

なお、ハードウェア実装については、ジョージメイソン大学の研究チームによって同一プラットフォームによる評価が行われているため、本ガイドラインでは扱わない。詳細は以下の URL を参照されたい。

Authenticated Encryption FPGA Ranking

`https://cryptography.gmu.edu/athenadb/fpga_auth_cipher/rankings_view`

3.2.1.1 性能比較

■認証暗号の実装評価結果

図 3.45 に認証暗号を小型実装した場合の評価結果を示す。図 3.45 における上の表は左から、アルゴリズム名、セキュリティレベル（bit）、ROM サイズ（bytes）、RAM サイズ（bytes）、関連データ処理に関する各関数の処理速度（赤字が cycles/byte、黒字は cycles）、内部で利用されている Core Function 名、Core Function の ROM サイズ（bytes）、Core Functions の速度（cycles/byte）である。図 3.45 における下の表は暗号化処理ならびに復号処理に関する各関数の処理速度を表している。

なお、CLOC-TWINE における ENC_NULL、DEC_NULL の速度はそれぞれ 11160、11099 cycles、CLOC-AES における ENC_NULL、DEC_NULL の速度はそれぞれ 8895、8790 cycles である。

図 3.46 に認証暗号を高速実装した場合の評価結果を示す。図の見方は図 3.45 と同様である。

なお、CLOC-TWINE における ENC_NULL、DEC_NULL の速度はそれぞれ 5074、5013 cycles、CLOC-AES における ENC_NULL、DEC_NULL の速度はそれぞれ 3748、3643 cycles である。

以上は暗号化（とタグ生成）モジュールならびに復号（とタグ検証）モジュールをともに含むプログラムの評価結果であったが、ここから復号部の関数群（DEC_0, DEC_1, DEC_2, DEC_3, DEC_4）を取り除いて暗号化（とタグ生成）機能だけにした場合は、その他の関数の性能は変わらず、ROM サイズだけが表 3.22 で示すバイト数だけ減少する。このサイズは Core Function の種類や小型版、高速版に依存せず一定である（そのように設計されている）。

ここで、Core Function で利用されるブロック暗号の暗号化・復号機能は、認証暗号としての暗号化・復号機能とは異なることに注意されたい。今回の実装評価対象とした認証暗号のうち、Minalpher と AES-OCB 以外はすべて、認証暗号としての復号機能においても Core Function の逆関数（Core Function がブロック暗号の場合はブロック暗号としての復号機能）

	Sec Level	Size		Associate Data Processing					Core Function		
		ROM	RAM	AD_0	AD_1	AD_2	AD_3	AD_4		ROM	Speed
CLOC-TWINE	32	811	108	-	11199	1409		365	TWINE	234	1380
SILC-PRESENT		537	98	93413	11706			312	PRESENT	164	11677
		600	96	38357	4796			312	PRESENT	227	4768
JAMBU-SIMON	48	522	90	12407	1050			12602	Simon (96/96)	109	1030
CLOC-AES	64	963	150	-	8966	570		637	AES	399	544
SILC-AES		772	144	9071	569			544			
AES-OTR		1202	178	142/9638	594		10134	-			
JAMBU-AES		803	128	8885	563		9010	-			
AES-OCB		1705	224	9252	564		8921		AES	1020	512/607
AES-GCM		760	172	8943	2557			-	Mul128	64	2534
ACORN	128	589	129	109075	489			15522	State	327	478
Minalpher		665	193	607/41361	1276		41220	-	P	295	1241
Ketje-SR		724	114	46827	988		3955	-	F	385	969
Ascon		617	132	40332	2491		19919	11	p^6	299	2475

	Plaintext Data Processing					Ciphertext Data Processing				
	ENC_0	ENC_1	ENC_2	ENC_3	ENC_4	DEC_0	DEC_1	DEC_2	DEC_3	DEC_4
CLOC-TWINE	11160	22365	2799	22593	-	11160	22365	2799	22532	64
SILC-PRESENT	93816	187276	23424		93786	93816	187276	23424		93778
	38343	76730	9606		38513	38543	76730	9606		38505
JAMBU-SIMON	-	1057		12619	25048	-	1057		12633	25099
CLOC-AES	9373	18211	1151	18096	-	9373	18211	1151	17991	108
SILC-AES	9443	18210	1151		9353	9443	18210	1151		9345
AES-OTR	9708	616		19839	10166	9708	609		19841	10158
JAMBU-AES	-	573		9035	17952	-	573		9042	17960
AES-OCB	9736	585		8804	8861	9736	680		8804	8885
AES-GCM	182	3124			49823	182	3124			50008
ACORN		490			62053		493			62085
Minalpher	40729	2560		81942	-	40729	2549		81525	196
Ketje-SR	-	992		23298	11724	-	993		23302	11747
Ascon		2503		206	40220		2517		300	40173

図 3.45　認証暗号の実装結果（小型実装）

表 3.22　認証暗号の暗号化モジュールのみのプログラムの ROM 削減バイト数

アルゴリズム名	AES-GCM	CLOC	SILC	AES-OTR	Ketje	Minalpher
削減されるバイト数	47	102	59	255	108	93

は必要ではない。また、Minalpher の Core Function は、その逆関数とは厳密には異なるが、Involution 構造を持つので非常に類似した構造を持っている。

本評価におけるインターフェースの作り方から、認証暗号としての暗号化部の関数群と復号部の関数群は明確に分けられているので、暗号化（とタグ生成）機能だけを持つ認証暗号モジュールのサイズは、単純に復号部の関数群のサイズを全体から引くだけで計算でき、速度性能については変わることはない。

図 3.47 に小型実装を行った場合の ROM サイズを比較したグラフを示す。図 3.48 はこの ROM サイズを、Core Function の ROM サイズ（下部）と Mode に相当する、それ以外の部分の ROM サイズ（上部）に分割したものである。また、図 3.49 に高速実装を行った場合の速度（正確には平文サイズが十分大きい時の漸近的速度）を比較したグラフを示す。図 3.50 はこの速度を、Core Function の速度（下部）と Mode に相当する、それ以外の部分の速度（上部）に分割したものである。

これらのグラフでは、いずれも認証暗号アルゴリズムのセキュリティレベルを（32 ビット、64 ビット、128 ビット）色で表現している。

	Sec Level	Size		Associate Data Processing					Core Function		
		ROM	RAM	AD_0	AD_1	AD_2	AD_3	AD_4		ROM	Speed
CLOC-TWINE	32	1049	106	-	5113	649		365	TWINE	470	620
SILC-PRESENT		896	118	10777	1349			262	PRESENT	499	1320
JAMBU-SIMON	48	1709	80	6342	531		6435	-	Simon (96/96)	477	527
CLOC-AES	64	1521	128	-	3819	248		637	AES	928	222
SILC-AES		1369	124	3924	248			438			
AES-OTR		1958	156	142/4081	239		4163	-			
JAMBU-AES		2466	102	3589	227		3710	-			
AES-OCB		2962	216	3809	238		3837		AES	2007	222/391
AES-GCM		1489	150	3669	1034			-	Mul128	239	1011
ACORN	128	871	116	55296	246			7752	State	446	237
Minalpher		1926	169	49/9580	308		9898	-	P	1455	289
		1977	425	866/8669	254		8170	-	P	1455	235
Ketje-SR		1482	114	15699	340		1361	-	f	927	321
Ascon		1966	116	11173	696		5639	11	p^6	1015	691

	Plaintext Data Processing					Ciphertext Data Processing				
	ENC_0	ENC_1	ENC_2	ENC_3	ENC_4	DEC_0	DEC_1	DEC_2	DEC_3	DEC_4
CLOC-TWINE	5074	10193	1278	10421	-	5074	10193	1278	10307	64
SILC-PRESENT	10657	21485	2689		10883	10657	21485	2689		10875
JAMBU-SIMON	-	532		6455	12699	-	532		6460	12767
CLOC-AES	3728	7585	476	7971	-	3728	7585	476	7866	108
SILC-AES	3694	7751	487		4100	3694	7751	487		4092
AES-OTR	4151	245		8140	4147	4151	243		8142	4139
JAMBU-AES	-	228		3738	7203	-	228		3745	7293
AES-OCB	4923	243		3893	3737	4923	412		3893	3888
AES-GCM	173	1271			20172	173	1271			20357
ACORN		247			31034		250			31082
Minalpher	9713	616		19739	-	9713	602		19554	196
	7985	508		16283	-	7985	494		16098	196
Ketje-SR	-	343		7734	3942	-	344		7738	3965
Ascon		698		188	11184		699		273	11312

図 3.46　認証暗号の実装結果（高速実装）

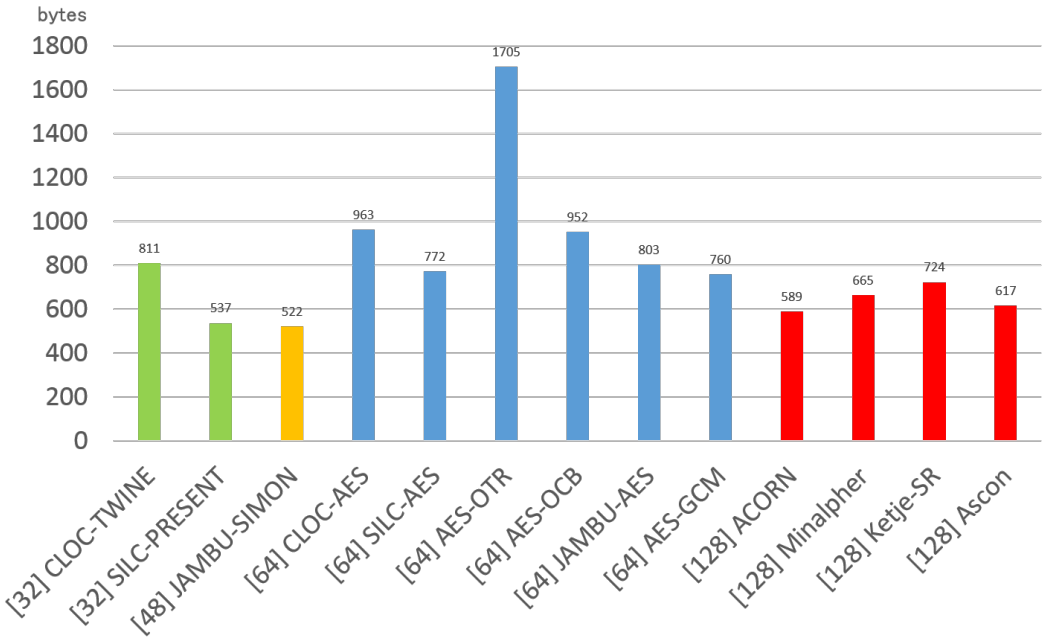


図 3.47　小型実装における ROM サイズ比較

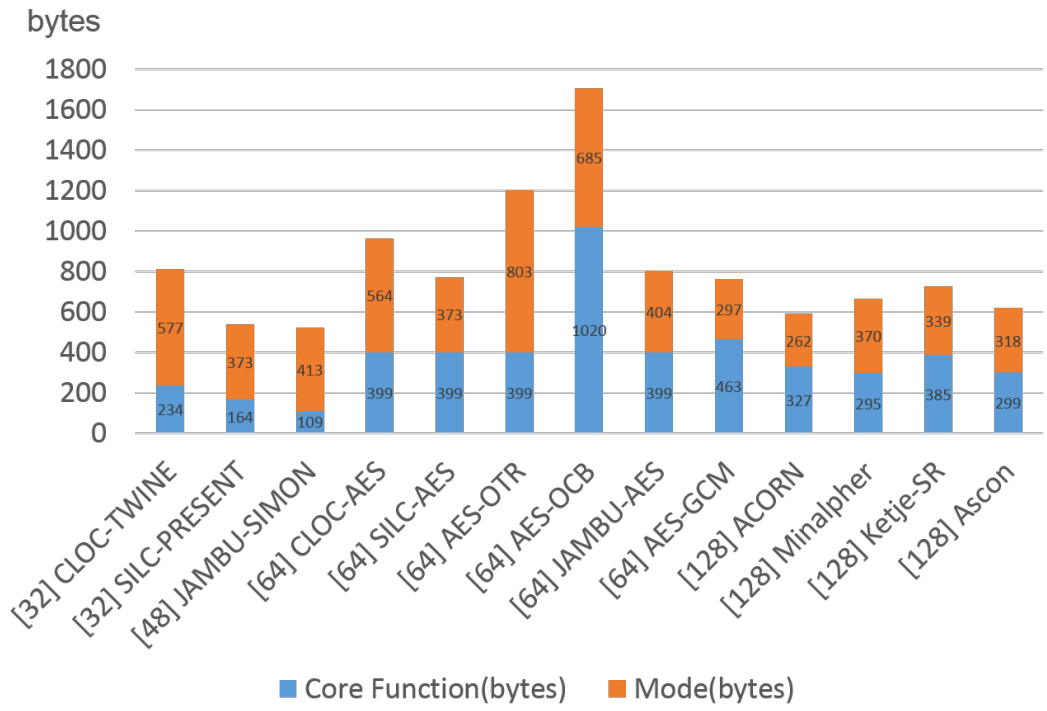


図 3.48　小型実装における ROM サイズ比較（Core Function と Mode に分割）

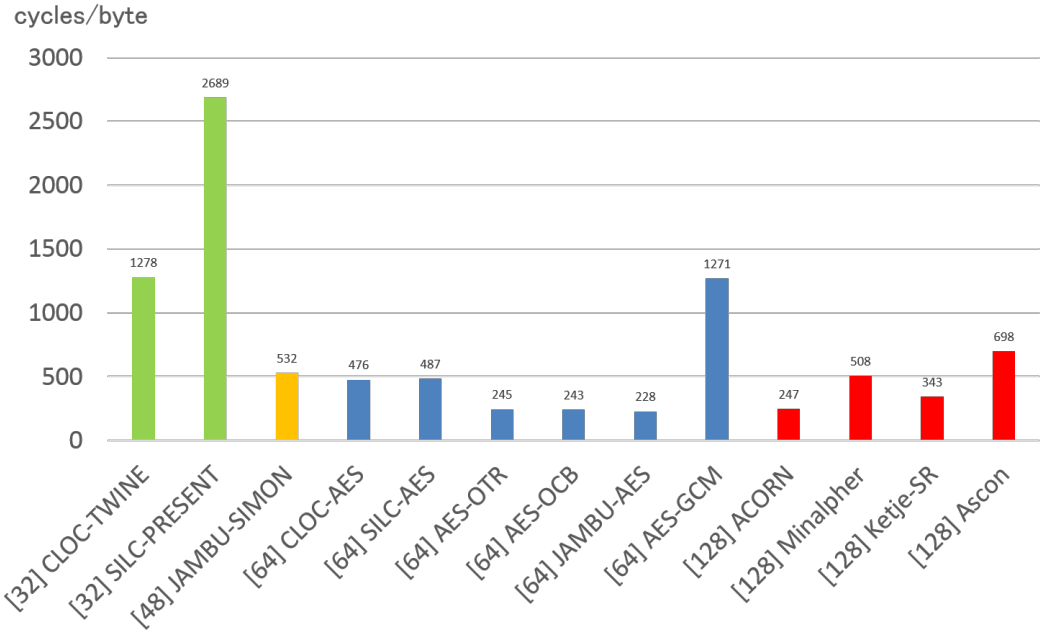


図 3.49　高速実装における暗号化速度（漸近速度）比較

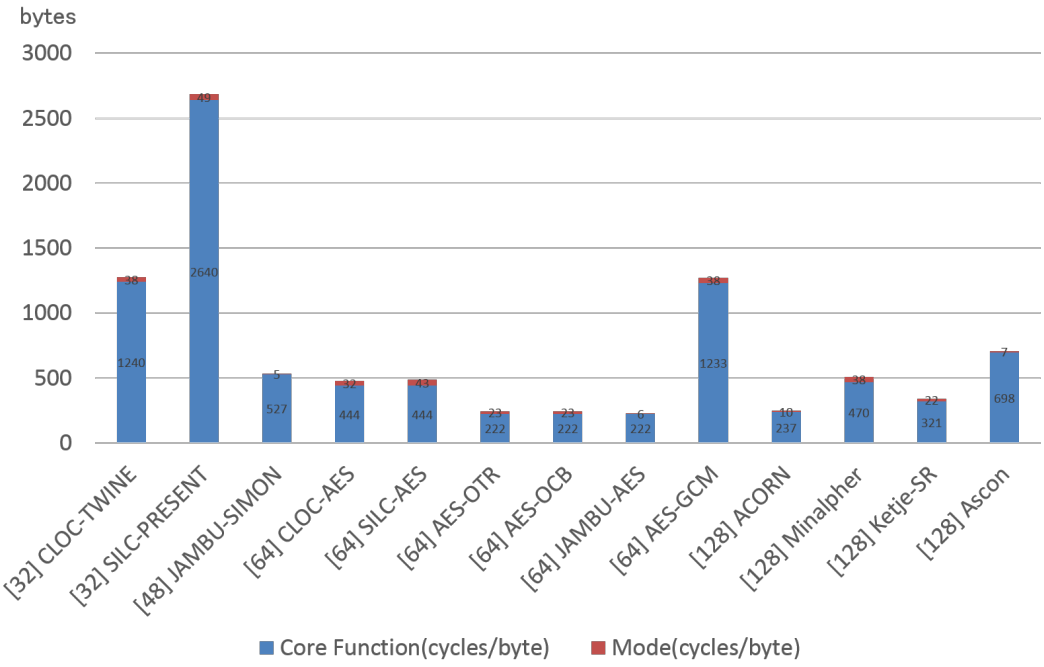


図 3.50　高速実装における暗号化速度（漸近速度）比較（Core Function と Mode に分割）

3.2.1.2 評価方法の概要

■コーディングの方針とインターフェース仕様

認証暗号は平文、暗号文、鍵に加え、関連データ（Associate Data)、ナンス（Nonce）等、入出力パラメータが多く、このため軽量暗号のように速度性能を一次元的に評価することは困難である。そこで本評価では、文献 [32] で示されたコーディングの方針に沿いつつ、各々の認証暗号アルゴリズムをいくつかの部分に分割し、その分割単位ごとに速度性能を評価する方針とした。これにより平文や関連データの長さが与えられれば、本報告で作成された表から、誰もが実際の計算にかかるサイクル数を自分で計算できる。

一方で、このようにアルゴリズムを細かく分割すると、認証暗号全体の処理を行うためには全体のフローをコントロールする上位プログラムが必要であり、そのプログラムのオーバーヘッドが大きくなると、分割単位ごとの性能データを集積しても全体の速度を正しく見積もることができなくなる。

そこで、ここでは上位プログラムのオーバーヘッドができるだけ少なくなるような、ブロック単位での分割の方法を提案する。具体的には、認証暗号のアルゴリズムを、関連データ処理部（以後 AD)、暗号化部（以後 ENC)、復号部（以後 DEC）に分割し、この 3 つの部分それぞれについて、さらに次の 5 つの機能に分割を行った。

1 初期計算（関連データや平文、暗号文を入力する前の処理）
2 第 1 ブロック計算
3 中間ブロック計算（第 2 ブロックから最終ブロックの前までの各 1 ブロックの処理）
4 最終ブロック計算
5 終了計算（関連データや平文、暗号文を入力し終わった後の処理）

これら各機能をひとつの関数としてコーディングしたが、実際には不必要な関数や同じ機能を持つ複数の関数があるため、すべての認証暗号アルゴリズムについて合計 15 個の関数を別々に実装する必要はない。また、一つの機能と別の機能との境界は一意的ではないが、アルゴリズムごとにもっとも自然と考えられる境界を設定した。

以降、これらの関数を次のように記述する。

AD_0: 関連データ処理部の初期計算関数
ENC_123: 暗号化部の第 1、中間、最終ブロック計算関数、これは ENC_1、ENC_2、ENC_3 が共通化できることを意味する。

この方法で記述した関数群を用いて認証暗号全体を記述した上位プログラムを、AES-GCM の暗号化モードを例として図 3.51 に示す。ここで、alen と mlen はそれぞれ関連データ、平文のバイト数である。BLEN はブロック長でこの場合 16 である。赤字が今回作成した関数に対応する。

```
int crypto_aead_encrypt(int mlen, int alen)
{
        int   clen=0,size;

        // Associate Data Handling

        AESGCM_AD_0();                          // Initialization

        while(alen > 0){
                size = (alen >= BLEN) ? BLEN : alen;
                AESGCM_AD_123(size);            // one block processing
                aadr += BLEN;                   // aadr = address of AD
                alen -= BLEN;
        }

        // Message Handling

        AESGCM_ENC_0();                         // Initialization

        while(mlen > 0){
                size = (mlen >= BLEN) ? BLEN : mlen;
                clen += AESGCM_ENC_123(size);   // one block processing
                madr += BLEN;                   // madr = address of MSG
                cadr += BLEN;                   // cadr = address of OUT
                mlen -= BLEN;
        }

        AESGCM_ENC_4(alen, clen);               // Tag Generation

        return clen;
}
```

図 3.51　AES-GCM の暗号化モードのコード例

■AES-GCM

本評価では、文献 [13] に記述されている AES-GCM アルゴリズムを、最も一般的と考えられる次のパラメータで実装した。

表 3.23　本評価で用いた AES-GCM のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
AES-GCM	128 ビット	128 ビット	96 ビット	64 ビット

また、図 3.52 に示すように AES-GCM アルゴリズムを機能分割した。

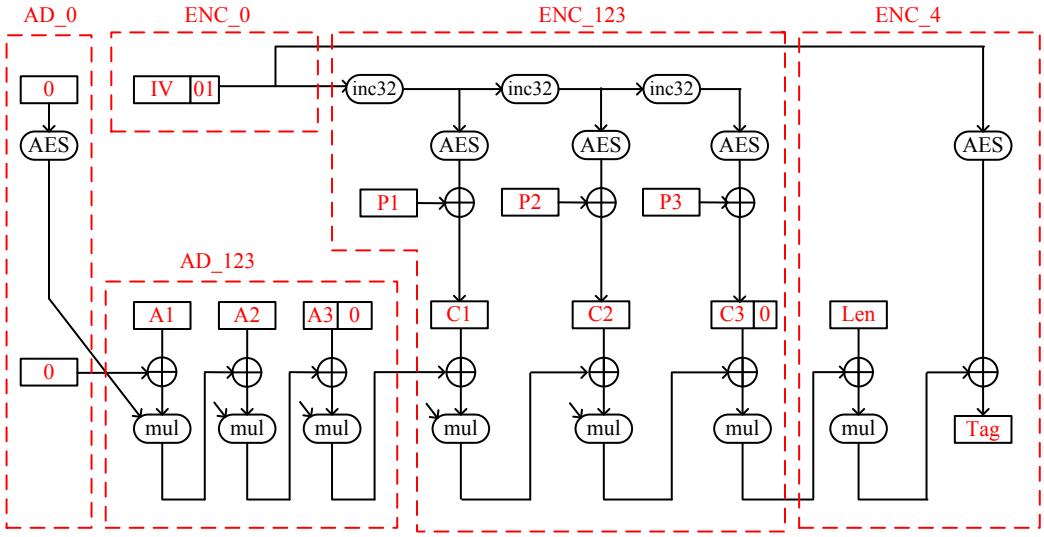


図 3.52　本評価で用いた AES-GCM の機能分割

■CLOC

CLOC アルゴリズム v2 [24] には推奨パラメータが 3 つ示されている。本評価ではこのうち 2 つを実装した。このうちひとつは Core Function として TWINE-64-80 を用いるもの、もうひとつは Core Function として AES-128-128 を用いるものである。

表 3.24　本評価で用いた CLOC のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
CLOC-TWINE	80 ビット	64 ビット	48 ビット	32 ビット
CLOC-AES	128 ビット	128 ビット	96 ビット	64 ビット

また、図 3.53 に示すように CLOC アルゴリズムを機能分割した。なお、CLOC は平文のサイズが 0 の時に特別の処理を行う仕様となっている。この特別な処理を ENC_NULL と名付けている。

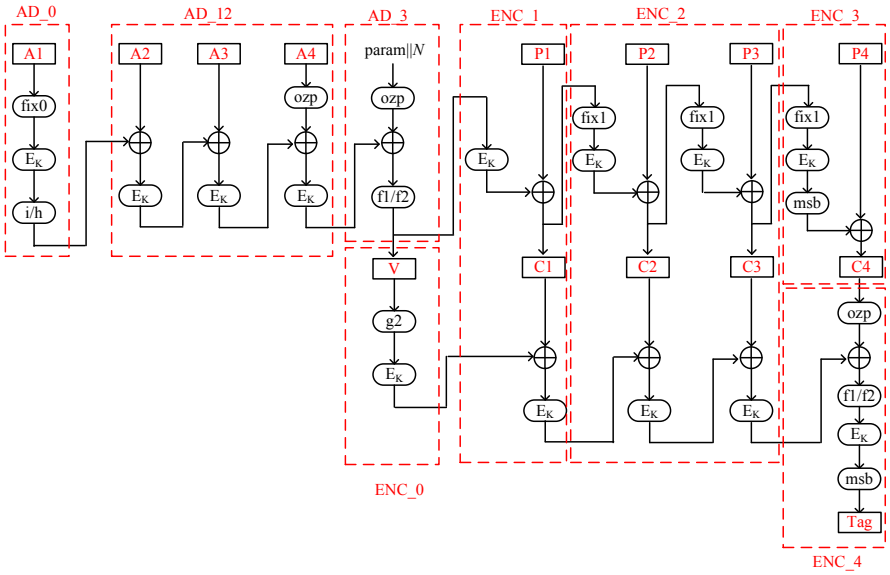


図 3.53　本評価で用いた CLOC の機能分割

■SILC

SILC アルゴリズム v2 [23] には推奨パラメータが 4 つ示されている。本評価ではこのうち 2 つを実装した。このうちひとつは Core Function として PRESENT-64-80 を用いるもの、もうひとつは Core Function として AES-128-128 を用いるものである。

表 3.25　本評価で用いた SILC のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
SILC-PRESENT	80 ビット	64 ビット	48 ビット	32 ビット
SILC-AES	128 ビット	128 ビット	96 ビット	64 ビット

また、図 3.54 に示すように SILC アルゴリズムを機能分割した。

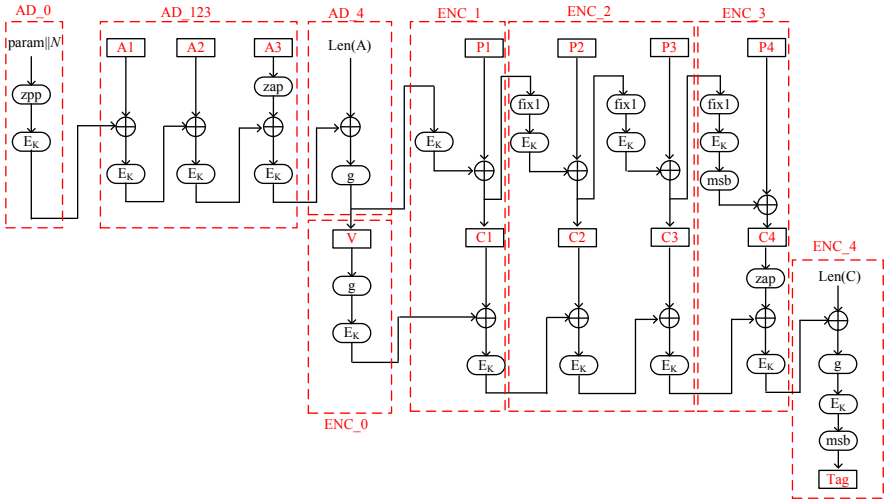


図 3.54　本評価で用いた SILC の機能分割

■Minalpher

Minalpher アルゴリズム [43] は Core Function として、Minalpher-P と呼ばれる置換が用いられている。パラメータは以下の一種類である。

表 3.26　本評価で用いた Minalpher のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
Minalpher	128 ビット	256 ビット	104 ビット	128 ビット

また、図 3.55 に示すように Minalpher アルゴリズムを機能分割した。

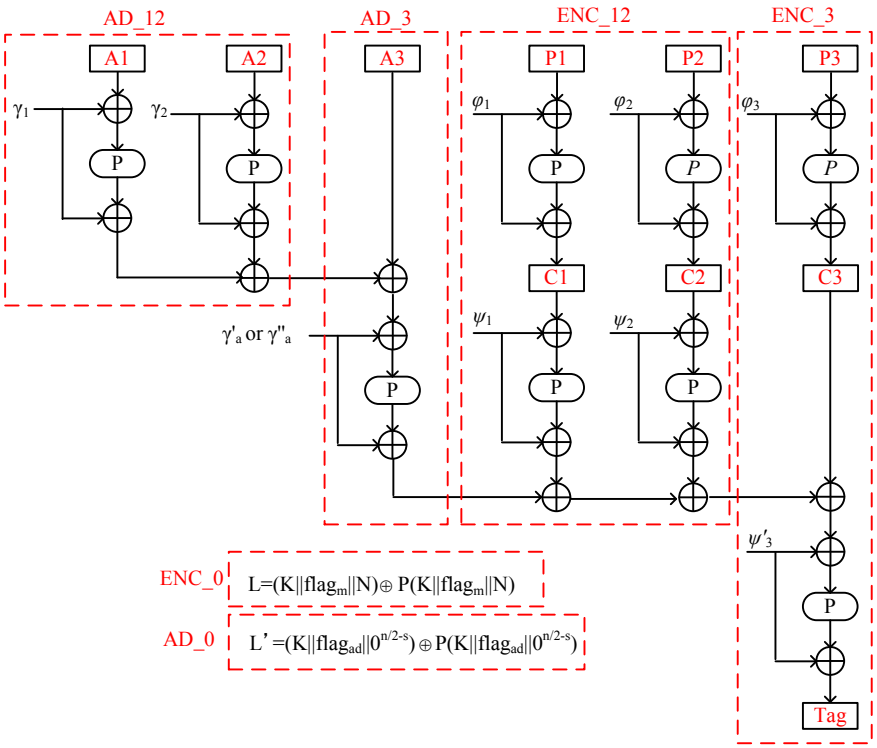


図 3.55　本評価で用いた Minalpher の機能分割

■AES-OTR

AES-OTR アルゴリズム [33] は Core Function として AES-128-128 あるいは AES-128-256 が用いられる。AES-OTR にはいくつかのパラメータが定義されているが、今回実装したのは Primary Parameter と呼ばれる以下のものであり、Core Function として AES-128-128 が使われている。

表 3.27　本評価で用いた AES-OTR のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
AES-OTR	128 ビット	128 ビット	96 ビット	128 ビット

また、図 3.56 に示すように AES-OTR アルゴリズムを機能分割した.

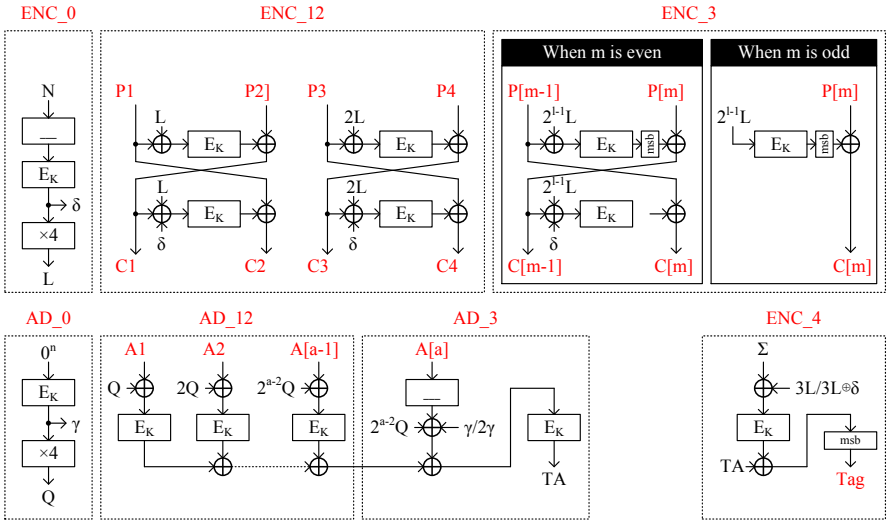


図 3.56　本評価で用いた AES-OTR の機能分割

■Ketje

Ketje アルゴリズム [11] は、Sponge 型認証暗号であり、Core Function として独自の関数 f が用いられている．Ketje には Ketje-SR と Ketje-JR の 2 つのパラメータが定義されているが、今回実装したのは Primary Recommendation とされている、50 バイトの入出力を持つ関数 f を用いたものである．

表 3.28　本評価で用いた Ketje のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
Ketje-SR	128 ビット	32 ビット	128 ビット	128 ビット

また、図 3.57 に示すように Ketje アルゴリズムを機能分割した．

なお、AD_12 と AD_3 は定数が異なる以外は同じ機能であり、関数 f の下に書かれた数字は関数内部の繰り返し回数を表している。

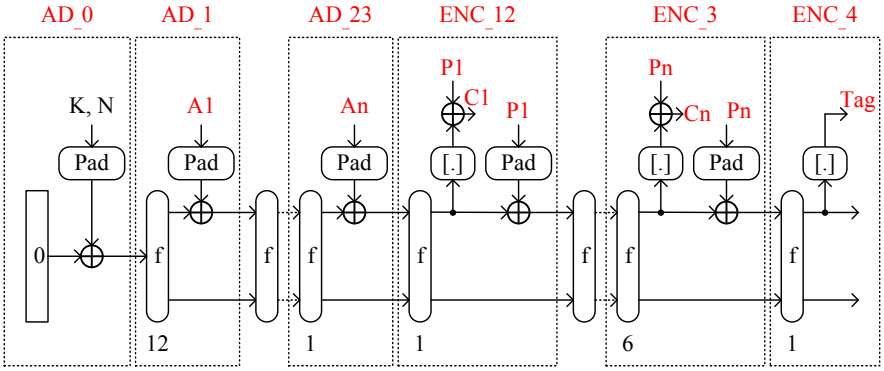


図 3.57　本評価で用いた Ketje の機能分割

■ACORN

ACORN [49] は、ストリーム暗号型の認証暗号である。ACORN では、Core function となる StateUpdate を、制御ビットと入力を変更しながら繰り返し実行することにより、293 ビットの内部状態 state を変更しながら暗号処理を実行する。本評価では、ACORN アルゴリズムを表 3.29 に示すパラメータで実装した。

表 3.29　本評価で用いた ACORN のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
ACORN	128 ビット	128 ビット	128 ビット	128 ビット

また、図 3.58 に示すように ACORN アルゴリズムを機能分割した.

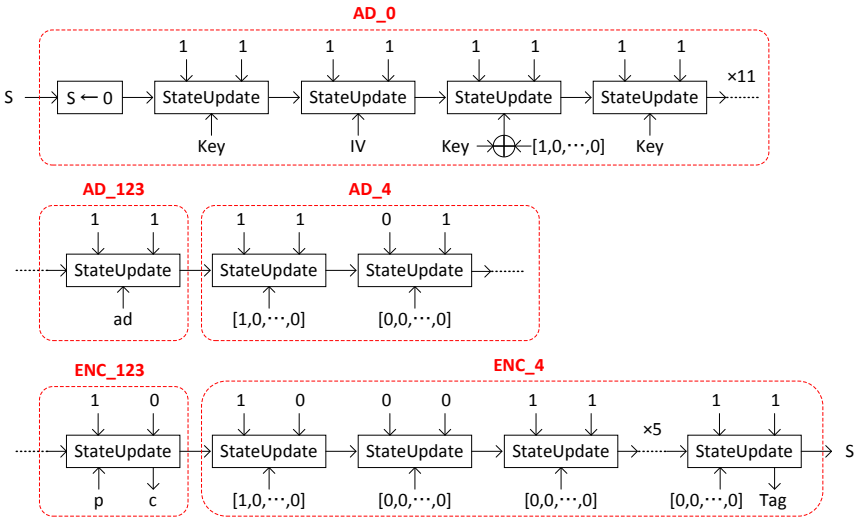


図 3.58　本評価で用いた ACORN の機能分割

■AES-OCB

AES-OCB アルゴリズム [29] は Core Function として AES-128-128、AES-128-192、あるいは AES-128-256 が用いられる。ここでは Primary Recommendation と考えられる、Core Function として AES-128-128 を用いた以下のパラメータのものを実装した。なお、AES-OCB は本報告書でとりあげた他の AES ベースの認証暗号の中で唯一（認証暗号としての）復号時に AES の復号機能を必要とする。

表 3.30　本評価で用いた AES-OCB のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
AES-OCB	128 ビット	128 ビット	96 ビット	128 ビット

また、図 3.59 に示すように AES-OCB アルゴリズムを機能分割した。

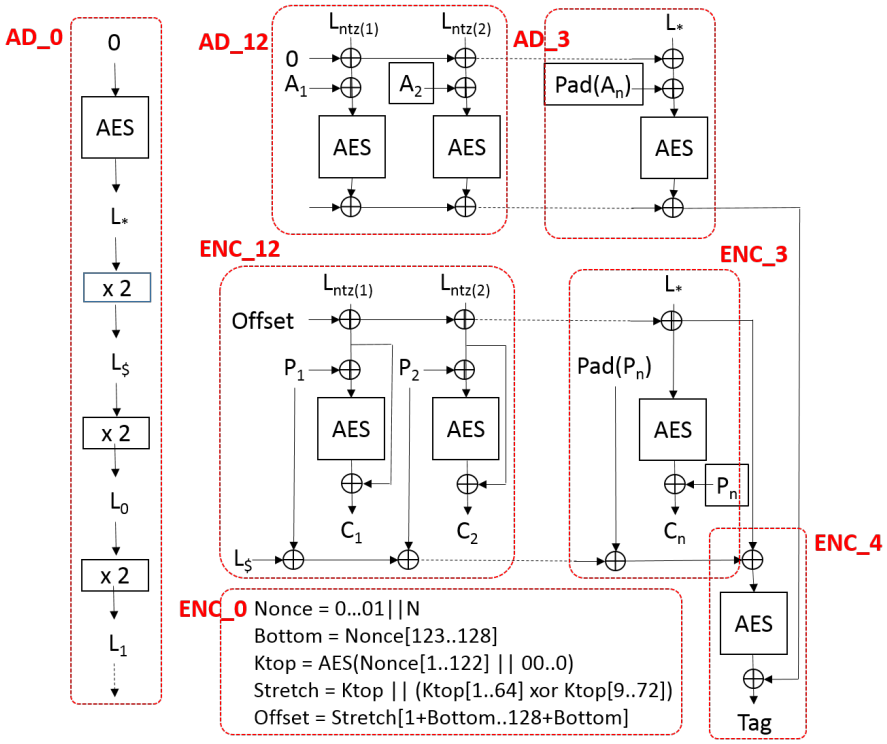


図 3.59　本評価で用いた AES-OCB の機能分割

■JAMBU

JAMBU アルゴリズム [50] は、Core Function として Simon-96-96、Simon-64-96、Simon-128-128 あるいは AES-128-128 が用いられる。ここでは Primary recommendation とされている Simon-96-96 を用いたものに加えて、AES-128-128 を用いたものの 2 種類を実装した。これらはそれぞれ以下のパラメータを持つものである。

表 3.31　本評価で用いた JAMBU のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
JAMBU-SIMON	96 ビット	96 ビット	48 ビット	48 ビット
JAMBU-AES	128 ビット	128 ビット	64 ビット	64 ビット

また、図 3.60 に示すように JAMBU アルゴリズムを機能分割した。

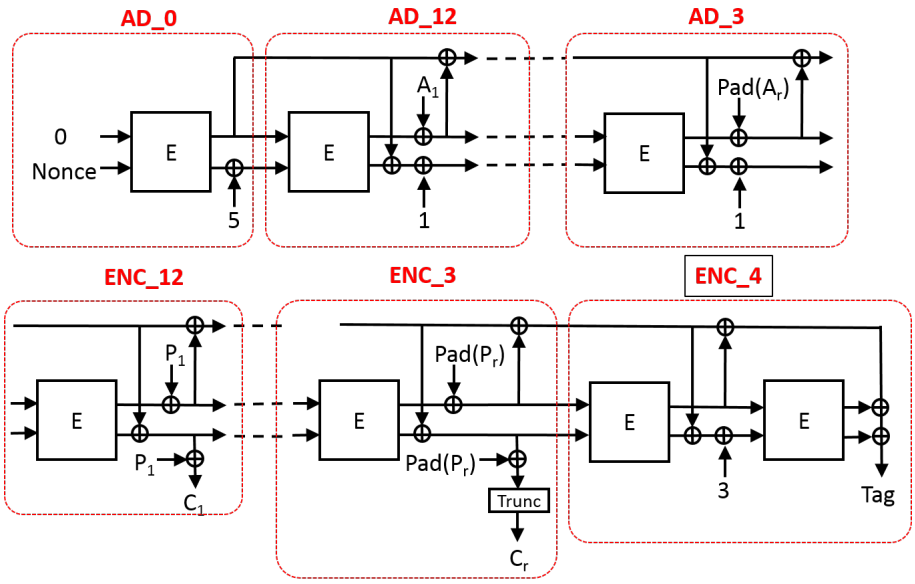


図 3.60　本評価で用いた JAMBU の機能分割

■Ascon

Ascon アルゴリズム [12] は Ketje と同じく Sponge 型認証暗号であり、独自の関数 p が内部で用いられている。ここでは Ascon の Primary Recommendation である、以下のパラメータを持つものを実装した。なお、関数 p は 40 バイトの入出力をもつものである。

表 3.32　本評価で用いた Ascon のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
Ascon	128 ビット	128 ビット	96 ビット	128 ビット

また、図 3.61 に示すように Ascon アルゴリズムを機能分割した。ここで、p^12, p^6 と示されている関数は、それぞれ関数 p を 12 回、6 回実行させることを示している。

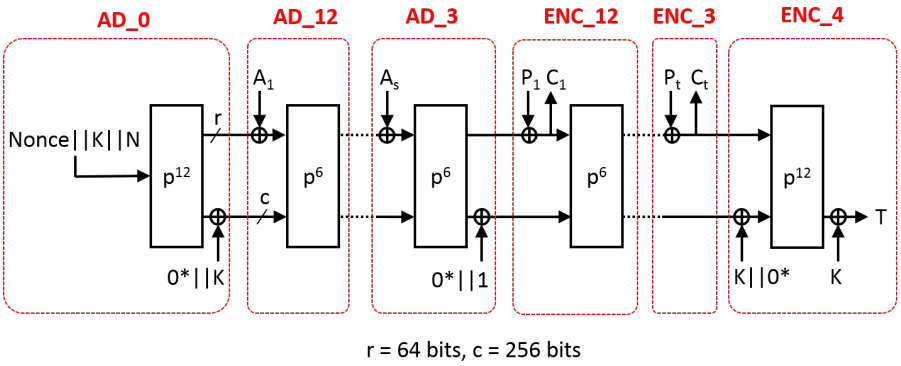


図 3.61　本評価で用いた Ascon の機能分割

3.3 Ascon の実装性能

3.3.1 ハードウェア実装性能

本節では、Ascon のハードウェア実装（特に、FPGA 実装と ASIC 実装）性能について、2022 年度に公開された CRYPTREC 外部評価報告書 [53] に基づき、2022 年 9 月現在の調査結果を掲載する。

3.3.1.1 調査対象と性能評価環境

ハードウェア実装（またはハードウェアアクセラレータ）とは、いわゆる専用回路実装と呼ばれるものであり、処理に必要なデータを供給して結果を出力させるものである。専用回路の処理ではインタフェース部がボトルネックとなる場合があるため、入出力データのインタフェース部も含めて実装する必要がある。NIST LWC ファイナリストの実装性能を可能な限り公平に評価する目的で、CAESAR コンペティションで使用された CAESAR HW API [22] と NIST LWC プロジェクトで提案された LWC HW API [26] が使用されることが多い。いずれのインタフェースも暗号処理性能を評価する上で大きな違いはないが、入力されるデータを適切に処理するために、ある程度のバッファメモリを用意する必要がある。これに伴い、全体の回路面積が増えることになるが、ハードウェア上で安定したデータ転送の実現を優先するために欠かせないものである。その他、アクセラレータがバスに対して優先的にデータを転送するためのバスアービトレーションを考慮することも重要である。このような詳細な検討は、ハードウェアアーキテクチャの仕様を厳密に策定する場合に必要となる。

■FPGA 実装　NIST LWC ファイナリストの FPGA 実装については、CAESAR コンペティションでの評価を含め、いくつかの報告がある [15, 41, 47, 51]。CAESAR コンペティションでの評価対象は、NIST LWC プロジェクトでの評価対象と仕様が異なる場合があるものの、実装性能に関する貴重な情報であることから、本ガイドラインでも紹介する。

FPGA 上の回路規模（面積コスト）の単位は統一されていない。AMD 社（旧 Xilinx 社）の FPGA は、LUT (Look-up Table) 数で評価することが一般的であり、Spartan-6、Artix-7、そして Zynq-7000 はいずれも 6 入力の LUT で評価される。一方、Intel（旧 Altera 社）の FPGA は、Cyclone-V の ALM (Adaptive Logic Module) や Cyclone 10 の LE (Logic Element) と呼ばれる複数の LUT を含むモジュールをビルディングブロックとし、その数で評価している。いずれも、2 入力 NAND ゲートを 1 単位とする GE (Gate Equivalent) に換算することが可能である。

Mohajerani らの研究 [35] では、複数の異なる FPGA に対し、入力データの違いによる認証暗号とハッシュ関数の処理性能を網羅的に比較している。インタフェースには LWC HW API [26] が使用されている。本ガイドラインでは、Mohajerani らによる評価結果の中から、特に重要と考えられる実装コストとスループット性能について紹介する。

■ASIC 実装　ASIC 実装の性能評価では、回路規模やスループット性能[*1]に加え、消費電力やエネルギー（電力量）効率がより重要となる。これはリソースの限られた IoT デバイスなどを想定しているためである。例えば、バッテリーを持たないデバイスでは消費電力が他の指標よりも重要となり、バッテリー駆動のデバイスではデバイスの寿命に直結するエネルギー効率が重要となる。エネルギーは電力を時間積分したものであるため、デバイスの使用率や消費電力の管理手法によりエネルギー効率は大きく変わる。理想的には、認証暗号が組み込まれるデバイス全体の消費電力やエネルギー効率が評価されるべきであるものの、このような実使用下のフィールドテストによる評価は難しく、アプリケーションにも大きく依存することから、暗号処理中に特化して消費電力とエネルギー効率を評価する研究が多い。

高スループットを得るために、本来であれば 1 サイクルで実行する関数（ラウンド関数など）をまとめて実行する Unrolled 実装型アーキテクチャを設計し、そのアーキテクチャの実装性能を評価するという研究が盛んに行われている。暗号アルゴリズムの種類によって Unrolled 実装に適さないものもあるが、Ascon に関しては Unrolled 実装に対して柔軟に適用可能であることが知られている。Unrolled 実装型アーキテクチャの特徴は、次のとおりである。

- アンロールするラウンド数が多くなるほど組合せ回路の面積が大きくなり、結果として全体の回路面積が大きくなる。一方、順序回路の規模は変わらない。
- 組合せ回路のクリティカルパス遅延時間が長くなり、結果として最大動作周波数が低下する。一方、組合せ回路を最

[*1] ここではレイテンシの逆数としての評価指標と位置づけ、パイプライン化による向上は想定しない。

適化し、最大動作周波数の低下を抑制することが可能である。

- 消費電力が増加する。一方、所望の処理を短時間で実行でき、エネルギー効率を向上できる。

本ガイドラインでは、Großらの研究 [19] と Elsadek らの研究 [14] で報告された評価結果を紹介する。Großらは、高いスループット性能を達成するために 1 種類の非 Unrolled 実装と 3 種類の Unrolled 実装で性能評価を実施するとともに、面積コストを抑制するために 2 種類のコンパクト実装で性能評価を実施している。また、Elsadek らはスループット性能とエネルギー効率に焦点を当てて性能評価を実施している。

3.3.1.2 実装性能

■FPGA 実装 表 3.33 は文献 [15, 41, 47, 51] で報告された FPGA 実装の評価結果をまとめたものである。この表では、FPGA の種類（プラットフォーム）、インタフェース、面積コスト、スループット性能を掲載している。

表 3.33 Ascon の FPGA 実装性能評価結果 [15, 41, 47, 51]

名称	プラットフォーム	インタフェース	面積コスト	スループット
Ascon-128 [15]	Spartan-6	CAESAR HW API	1,402 LUTs	1,906 Mbps
Ascon-128a [15]			1,712 LUTs	2,884 Mbps
Ascon-128 [51]	Spartan-6	CAESAR HW API	684 LUTs	60 Mbps
Ascon-128a [51]			684 LUTs	119 Mbps
Ascon-128 [41]	Artix-7	LWC HW API	1,898 LUTs	1,683 Mbps
	Spartan-6		1,913 LUTs	1,116 Mbps
	Cyclone-V		1,051 ALMs	1,295 Mbps
Ascon-Hash [41]	Artix-7	LWC HW API	2,181 LUTs	1,032 Mbps
	Spartan-6		2,188 LUTs	678 Mbps
	Cyclone-V		1,064 ALMs	898 Mbps
Ascon-128 [47]	Zynq-7000-6	CAESAR HW API	6,325 LUTs	–

表 3.33 の中で最もコンパクトな実装は、文献 [51] で報告された結果であり、その面積コストは 684 LUTs である。最も高速な実装は，文献 [15] で報告された結果であり、約 2.9 Gbps のスループット性能を 1,712 LUTs の面積コストで達成した。また、文献 [41] では、ハッシュ関数として機能させた場合、Artix-7 上で 1.0 Gbps の処理性能を 2,181 LUTs の回路面積で達成したと報告されている。

インタフェースの使用はハードウェアモジュールの面積コストと処理性能に大きな影響を及ぼすため、表の数値だけで Ascon の正確な実装性能を評価することは難しい。一方で、表 3.33 の結果から、Ascon が軽量実装可能であることや高い処理性能を実現可能であることを読み取ることができる。このため、Ascon には実装性能のトレードオフを模索できる柔軟性があると言える。

その他、Mohajerani らの研究 [35] を紹介する。Mohajerani らは、複数の異なる種類の FPGA に対し、入力データの違いによる認証暗号とハッシュ関数の処理性能を網羅的に比較している。評価結果が膨大であるため、面積コストとスループット性能に限定し、評価結果の一部を表 3.34 に掲載する。

表 3.34: Ascon の FPGA 実装性能評価結果 [35]

名称	プラットフォーム	データ量	面積コスト	スループット
Ascon-GMU-v1	Artix-7	AD+PT (Long)	2,410 LUTs	6,297 Mbp
		Hash (Long)	–	–
	Cyclone 10	AD+PT (Long)	4,552 LEs	3,031 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
		Hash (Long)	2,415 LEs	864 Mbps
	ECP5	AD+PT (Long)	5,909 LUTs	2,158 Mbps
		Hash (Long)	–	–
	Artix-7	AD+PT (1,536 Bytes)	2,410 LUTs	3,022 Mbps
		AD+PT (64 Bytes)		1,574 Mbps
		AD+PT (16 Bytes)		629 Mbps
		Hash	–	–
	Cyclone 10	AD+PT (1,536 Bytes)	4,552 LEs	1,454 Mbps
		AD+PT (64 Bytes)		757 Mbps
		AD+PT (16 Bytes)		303 Mbps
		Hash	–	–
	ECP5	AD+PT (1,536 Bytes)	5,909 LUTs	1,035 Mbps
		AD+PT (64 Bytes)		539 Mbps
		AD+PT (16 Bytes)		215 Mbps
		Hash	–	–
Ascon-GMU-v2	Artix-7	AD+PT (Long)	1,790 LUTs	4,366 Mbps
		Hash (Long)	–	–
	Cyclone 10	AD+PT (Long)	3,113 LEs	2,284 Mbps
		Hash (Long)	3,215 LEs	1,232 Mbps
	ECP5	AD+PT (Long)	4,641 LUTs	1,666 Mbps
		Hash (Long)	–	–
	Artix-7	AD+PT (1,536 Bytes)	1,790 LUTs	2,115 Mbps
		AD+PT (64 Bytes)		1,237 Mbps
		AD+PT (16 Bytes)		538 Mbps
		Hash	–	–
	Cyclone 10	AD+PT (1.536 Bytes)	3,113 LEs	1,107 Mbps
		AD+PT (64 Bytes)		647 Mbps
		AD+PT (16 Bytes)		281 Mbps
		Hash	–	–
	ECP5	AD+PT (1,536 Bytes)	4,641 LUTs	807 Mbps
		AD+PT (64 Bytes)		472 Mbps
		AD+PT (16 Bytes)		205 Mbps
		Hash	–	–
Ascon-GMU2-v1h	Artix-7	AD+PT (Long)	1,375 LUTs	2,523 Mbps
		Hash (Long)		1,358 Mbps
	Cyclone 10	AD+PT (Long)	2,415 LEs	1,605 Mbps
		Hash (Long)	4,161 LEs	1,173 Mbps
	ECP5	AD+PT (Long)	2,928 LUTs	1,006 Mbps
		Hash (Long)		541 Mbps
	Artix-7	AD+PT (1,536 Bytes)	1,375 LUTs	1,236 Mbps
		AD+PT (64 Bytes)		851 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
		AD+PT (16 Bytes)		430 Mbps
		Hash (1,536 Bytes)		1,321 Mbps
		Hash (64 Bytes)		812 Mbps
		Hash (16 Bytes)		368 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	2,415 LEs	786 Mbps
		AD+PT (64 Bytes)		541 Mbps
		AD+PT (16 Bytes)		274 Mbps
		Hash (1,536 Bytes)		840 Mbps
		Hash (64 Bytes)		516 Mbps
		Hash (16 Bytes)		234 Mbps
	ECP5	AD+PT (1,536 Bytes)	2,928 LUTs	493 Mbps
		AD+PT (64 Bytes)		339 Mbps
		AD+PT (16 Bytes)		171 Mbps
		Hash (1,536 Bytes)		527 Mbps
		Hash (64 Bytes)		323 Mbps
		Hash (16 Bytes)		146 Mbps
Ascon-GMU2-v2h	Artix-7	AD+PT (Long)	2,126 LUTs	3,744 Mbps
		Hash (Long)		2,139 Mbps
	Cyclone 10	AD+PT (Long)	3,215 LEs	2,157 Mbps
		Hash (Long)	–	–
	ECP5	AD+PT (Long)	3,764 LUTs	1,427 Mbps
		Hash (Long)		815 Mbps
	Artix-7	AD+PT (1,536 Bytes)	2,126 LUTs	1,825 Mbps
		AD+PT (64 Bytes)		1,163 Mbps
		AD+PT (16 Bytes)		544 Mbps
		Hash (1,536 Bytes)		2,077 Mbps
		Hash (64 Bytes)		1,248 Mbps
		Hash (16 Bytes)		554 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	3,215 LEs	1,051 Mbps
		AD+PT (64 Bytes)		670 Mbps
		AD+PT (16 Bytes)		313 Mbps
		Hash (1,536 Bytes)		1,196 Mbps
		Hash (64 Bytes)		719 Mbps
		Hash (16 Bytes)		319 Mbps
	ECP5	AD+PT (1,536 Bytes)	3,764 LUTs	696 Mbps
		AD+PT (64 Bytes)		443 Mbps
		AD+PT (16 Bytes)		207 Mbps
		Hash (1,536 Bytes)		792 Mbps
		Hash (64 Bytes)		475 Mbps
		Hash (16 Bytes)		211 Mbps
Ascon-GMU2-v3h	Artix-7	AD+PT (Long)	2,493 LUTs	3,029 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
		Hash (Long)		1,817 Mbps
	Cyclone 10	AD+PT (Long)	4,161 LEs	1,955 Mbps
		Hash (Long)	–	–
	ECP5	AD+PT (Long)	4,925 LUTs	1,305 Mbps
		Hash (Long)		783 Mbps
	Artix-7	AD+PT (1,536 Bytes)	2,493 LUTs	1,470 Mbps
		AD+PT (64 Bytes)		876 Mbps
		AD+PT (16 Bytes)		386 Mbps
		Hash (1,536 Bytes)		1,762 Mbps
		Hash (64 Bytes)		1,038 Mbps
		Hash (16 Bytes)		454 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	4,161 LEs	949 Mbps
		AD+PT (64 Bytes)		565 Mbps
		AD+PT (16 Bytes)		249 Mbps
		Hash (1,536 Bytes)		1,137 Mbps
		Hash (64 Bytes)		670 Mbps
		Hash (16 Bytes)		293 Mbps
	ECP5	AD+PT (1,536 Bytes)	4,925 LUTs	633 Mbps
		AD+PT (64 Bytes)		377 Mbps
		AD+PT (16 Bytes)		166 Mbps
		Hash (1,536 Bytes)		759 Mbps
		Hash (64 Bytes)		447 Mbps
		Hash (16 Bytes)		195 Mbps
Ascon-Graz-v1	Artix-7	AD+PT (Long)	1,465 LUTs	1,528 Mbps
		Hash (Long)		873 Mbps
	Cyclone 10	AD+PT (Long)	2,517 LEs	1,131 Mbps
		Hash (Long)		646 Mbps
	ECP5	AD+PT (Long)	2,544 LUTs	474 Mbps
		Hash (Long)		271 Mbps
	Artix-7	AD+PT (1,536 Bytes)	1,465 LUTs	752 Mbps
		AD+PT (64 Bytes)		552 Mbps
		AD+PT (16 Bytes)		301 Mbps
		Hash (1,536 Bytes)		850 Mbps
		Hash (64 Bytes)		528 Mbps
		Hash (16 Bytes)		242 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	2,517 LEs	556 Mbps
		AD+PT (64 Bytes)		409 Mbps
		AD+PT (16 Bytes)		223 Mbps
		Hash (1,536 Bytes)		629 Mbps
		Hash (64 Bytes)		391 Mbps
		Hash (16 Bytes)		179 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
	ECP5	AD+PT (1,536 Bytes)	2,544 LUTs	233 Mbps
		AD+PT (64 Bytes)		171 Mbps
		AD+PT (16 Bytes)		93 Mbps
		Hash (1,536 Bytes)		263 Mbps
		Hash (64 Bytes)		164 Mbps
		Hash (16 Bytes)		75 Mbps
Ascon-Graz-v2	Artix-7	AD+PT (Long)	1,541 LUTs	2,272 Mbps
		Hash (Long)		973 Mbps
	Cyclone 10	AD+PT (Long)	2,634 LEs	1,529 Mbps
		Hash (Long)		655 Mbps
	ECP5	AD+PT (Long)	2,603 LUTs	683 Mbps
		Hash (Long)		292 Mbps
	Artix-7	AD+PT (1,536 Bytes)	1,541 LUTs	1,108 Mbps
		AD+PT (64 Bytes)		712 Mbps
		AD+PT (16 Bytes)		336 Mbps
		Hash (1,536 Bytes)		948 Mbps
		Hash (64 Bytes)		589 Mbps
		Hash (16 Bytes)		269 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	2,634 LEs	746 Mbps
		AD+PT (64 Bytes)		479 Mbps
		AD+PT (16 Bytes)		226 Mbps
		Hash (1,536 Bytes)		638 Mbps
		Hash (64 Bytes)		396 Mbps
		Hash (16 Bytes)		181 Mbps
	ECP5	AD+PT (1,536 Bytes)	2,603 LUTs	333 Mbps
		AD+PT (64 Bytes)		214 Mbps
		AD+PT (16 Bytes)		101 Mbps
		Hash (1,536 Bytes)		285 Mbps
		Hash (64 Bytes)		177 Mbps
		Hash (16 Bytes)		81 Mbps
Ascon-Graz-v3	Artix-7	AD+PT (Long)	2,142 LUTs	2,572 Mbps
		Hash (Long)		1,608 Mbps
	Cyclone 10	AD+PT (Long)	3,716 LEs	1,403 Mbps
		Hash (Long)		877 Mbps
	ECP5	AD+PT (Long)	3,305 LUTs	815 Mbps
		Hash (Long)		509 Mbps
	Artix-7	AD+PT (1,536 Bytes)	2,142 LUTs	1,260 Mbps
		AD+PT (64 Bytes)		857 Mbps
		AD+PT (16 Bytes)		428 Mbps
		Hash (1,536 Bytes)		1,564 Mbps
		Hash (64 Bytes)		961 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
		Hash (16 Bytes)		436 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	3,716 LEs	687 Mbps
		AD+PT (64 Bytes)		467 Mbps
		AD+PT (16 Bytes)		233 Mbps
		Hash (1,536 Bytes)		853 Mbps
		Hash (64 Bytes)		524 Mbps
		Hash (16 Bytes)		237 Mbps
	ECP5	AD+PT (1,536 Bytes)	3,305 LUTs	399 Mbps
		AD+PT (64 Bytes)		271 Mbps
		AD+PT (16 Bytes)		135 Mbps
		Hash (1,536 Bytes)		495 Mbps
		Hash (64 Bytes)		304 Mbps
		Hash (16 Bytes)		138 Mbps
Ascon-Graz-v4	Artix-7	AD+PT (Long)	2,249 LUTs	3,296 Mbps
		Hash (Long)		1,648 Mbps
	Cyclone 10	AD+PT (Long)	3,730 LEs	1,738 Mbps
		Hash (Long)		869 Mbps
	ECP5	AD+PT (Long)	3,379 LUTs	989 Mbps
		Hash (Long)		494 Mbps
	Artix-7	AD+PT (1,536 Bytes)	2,249 LUTs	1,605 Mbps
		AD+PT (64 Bytes)		1,004 Mbps
		AD+PT (16 Bytes)		462 Mbps
		Hash (1,536 Bytes)		1,603 Mbps
		Hash (64 Bytes)		985 Mbps
		Hash (16 Bytes)		446 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	3,730 LEs	846 Mbps
		AD+PT (64 Bytes)		529 Mbps
		AD+PT (16 Bytes)		244 Mbps
		Hash (1,536 Bytes)		845 Mbps
		Hash (64 Bytes)		519 Mbps
		Hash (16 Bytes)		235 Mbps
	ECP5	AD+PT (1,536 Bytes)	3,379 LUTs	481 Mbps
		AD+PT (64 Bytes)		301 Mbps
		AD+PT (16 Bytes)		138 Mbps
		Hash (1,536 Bytes)		481 Mbps
		Hash (64 Bytes)		296 Mbps
		Hash (16 Bytes)		134 Mbps
Ascon-Graz-v5	Artix-7	AD+PT (Long)	2,797 LUTs	2,400 Mbps
		Hash (Long)		1,600 Mbps
	Cyclone 10	AD+PT (Long)	4,905 LEs	1,281 Mbps
		Hash (Long)		854 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
	ECP5	AD+PT (Long)	4,646 LUTs	889 Mbps
		Hash (Long)		593 Mbps
	Artix-7	AD+PT (1,536 Bytes)	2,797 LUTs	1,173 Mbps
		AD+PT (64 Bytes)		775 Mbps
		AD+PT (16 Bytes)		376 Mbps
		Hash (1,536 Bytes)		1,555 Mbps
		Hash (64 Bytes)		948 Mbps
		Hash (16 Bytes)		426 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	4,905 LEs	626 Mbps
		AD+PT (64 Bytes)		414 Mbps
		AD+PT (16 Bytes)		201 Mbps
		Hash (1,536 Bytes)		830 Mbps
		Hash (64 Bytes)		506 Mbps
		Hash (16 Bytes)		227 Mbps
	ECP5	AD+PT (1,536 Bytes)	4,646 LUTs	435 Mbps
		AD+PT (64 Bytes)		287 Mbps
		AD+PT (16 Bytes)		139 Mbps
		Hash (1,536 Bytes)		576 Mbps
		Hash (64 Bytes)		351 Mbps
		Hash (16 Bytes)		158 Mbps
Ascon-Graz-v6	Artix-7	AD+PT	–	–
		Hash	–	–
	Cyclone 10	AD+PT	–	–
		Hash	–	–
	ECP5	AD+PT (Long)	5,346 LUTs	827 Mbps
		Hash (Long)		496 Mbps
		AD+PT (1,536 Bytes)		402 Mbps
		AD+PT (64 Bytes)		245 Mbps
		AD+PT (16 Bytes)		110 Mbps
		Hash (1,536 Bytes)		482 Mbps
		Hash (64 Bytes)		292 Mbps
		Hash (16 Bytes)		130 Mbps
Ascon-VT-v1	Artix-7	AD+PT (Long)	1,913 LUTs	1,491 Mbps
		Hash (Long)	–	–
	Cyclone 10	AD+PT (Long)	2,432 LEs	1,130 Mbps
		Hash (Long)	–	–
	ECP5	AD+PT (Long)	3,130 LUTs	543 Mbps
		Hash (Long)	–	–
	Artix-7	AD+PT (1,536 Bytes)	1,913 LUTs	735 Mbps
		AD+PT (64 Bytes)		560 Mbps
		AD+PT (16 Bytes)		320 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
	Cyclone 10	AD+PT (1,536 Bytes)	2,432 LEs	557 Mbps
		AD+PT (64 Bytes)		424 Mbps
		AD+PT (16 Bytes)		243 Mbps
	ECP5	AD+PT (1,536 Bytes)	3,130 LUTs	268 Mbps
		AD+PT (64 Bytes)		204 Mbps
		AD+PT (16 Bytes)		116 Mbps
Ascon-VT-v2	Artix-7	AD+PT (Long)	1,928 LUTs	1,475 Mbps
		Hash (Long)		934 Mbps
	Cyclone 10	AD+PT (Long)	2,695 LEs	1,158 Mbps
		Hash (Long)		733 Mbps
	ECP5	AD+PT (Long)	3,041 LUTs	508 Mbps
		Hash (Long)		321 Mbps
	Artix-7	AD+PT (1,536 Bytes)	1,928 LUTs	726 Mbps
		AD+PT (64 Bytes)		544 Mbps
		AD+PT (16 Bytes)		304 Mbps
		Hash (1,536 Bytes)		910 Mbps
		Hash (64 Bytes)		572 Mbps
		Hash (16 Bytes)		264 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	2,695 LEs	570 Mbps
		AD+PT (64 Bytes)		427 Mbps
		AD+PT (16 Bytes)		239 Mbps
		Hash (1,536 Bytes)		715 Mbps
		Hash (64 Bytes)		449 Mbps
		Hash (16 Bytes)		207 Mbps
	ECP5	AD+PT (1,536 Bytes)	3,041 LUTs	250 Mbps
		AD+PT (64 Bytes)		187 Mbps
		AD+PT (16 Bytes)		104 Mbps
		Hash (1,536 Bytes)		313 Mbps
		Hash (64 Bytes)		197 Mbps
		Hash (16 Bytes)		91 Mbps

表 3.34 の結果について、認証暗号の性能は暗号化時のものであり、インタフェースは LWC HW API を使用している。名称の項目は RTL (Register Transfer Level) コードの名前であり、設計者とバージョンの違いで区別されている。データ量の項目は入力データの違いを記載しており、例えば “AD+PT (Long)” は十分にデータ長の長い関連データ (AD: associated data) と平文 (PT: plaintext) に対して暗号化処理を行う場合を表している。括弧内がバイト数の場合は、入力長を表している。表 3.34 から、同じ FPGA 実装でも入力長の違いによって処理性能が異なることがわかる。最も高速な実装性能は、2,410 LUTs の面積コストで 6 Gbps を超えるスループット性能を達成している。これは Artix-7 上で、入力データ量が AD+PT (Long) の場合の評価結果である[*2]。

■ASIC 実装　表 3.35 は文献 [19] で報告された Großらによる ASIC 実装の評価結果をまとめたものである。Großらの研究では、90 nm UMC low-K ライブラリを使用して Ascon の ASIC 実装に対し、面積コスト、スループット、消費電力、エネ

[*2] 実際に高いスループット性能を達成するためには、データの入出力がボトルネックにならないことが条件となる。

ルギー消費量を評価している。なお、インタフェース部（鍵レジスタと 64 ビットのバスインタフェース）の回路コストは、0.87 kGE から 1.18 kGE のゲートサイズである。

表 3.35　Großらによる Ascon の ASIC 実装評価結果 [19]

名称	インタフェース	面積コスト	スループット @f_{max}	消費電力 @1MHz	エネルギー消費量
Ascon-fast 1 round	no	7.08 kGE	5,524 Mbps	43 μW	33 μJ/byte
	custom	7.95 kGE			
Ascon-fast 2 rounds	no	10.61 kGE	8,425 Mbps	72 μW	27 μJ/byte
	custom	11.48 kGE			
Ascon-fast 3 rounds	no	14.26 kGE	10,407 Mbps	102 μW	25 μJ/byte
	custom	15.13 kGE			
Ascon-fast 6 rounds	no	24.93 kGE	13,218 Mbps	184 μW	23 μJ/byte
	custom	25.80 kGE			
Ascon-64-bit	no	4.99 kGE	72 Mbps	32 μW	1,397 μJ/byte
	custom	5.86 kGE			
Ascon-x-low-area	no	2.57 kGE	14 Mbps	15 μW	5,706 μJ/byte
	custom	3.75 kGE			

Ascon-fast は、高いスループット性能を実現するための実装である。Ascon-fast 1 round は、Ascon-fast をベースとして設計された非 Unrolled 実装である。また、Ascon-fast 2 rounds、Ascon-fast 3 rounds、Ascon-fast 6 rounds は、それぞれ 2、3、6 ラウンド分を 1 サイクルで実行する Unrolled 実装である。表 3.35 からも、Ascon には面積コストと処理性能のトレードオフを模索できる柔軟性があると言える。アンロール数が多くなるほど処理性能が高くなり、特に Ascon-fast 6 rounds で約 13 Gbps のスループット性能を達成していることがわかる。また、Ascon-fast 6 rounds において消費電力が最大になるものの、処理時間が短くなるためにエネルギー効率が最も良いことがわかる。

Ascon-64-bit は、64 ビットの算術論理ユニット（ALU）に基づくデータパスとして設計されたものである。ソフトウェア実装のようにラウンド処理を複数のサイクルに分けて実行するため、処理性能は低下するものの面積コストを抑制できるという利点がある。また、Ascon-x-low-area は面積コストをさらに抑制するために設計されたものである。Ascon-64-bit よりもさらに小さいデータパスを利用することでコンパクト実装を実現している。Ascon-64-bit と Ascon-x-low-area は、いずれも消費電力を抑制できたが、Ascon-fast と比べてエネルギー消費量は大幅に増加する。

表 3.36 は文献 [14] で報告された Elsadek らによる ASIC 実装の評価結果をまとめたものである。Elsadek らの研究では、GF 22 nm CMOS (GF22FDx) で合成した Ascon の ASIC 実装に対し、スループット性能とエネルギー効率を評価している。なお、インターフェース回路とレイアウトについて考慮されておらず、コア関数のみの評価となっていることに注意されたい。

入力データ長が短い（Short）場合では、16 バイトのデータを間隔を空けて送信することを想定している。また、入力データ長が長い（Long）場合では、1,536 バイトの連続した入力データの処理を想定している。Short の場合と比べると、Long の場合の方がスループット性能は高くなり、500 Mbps を超えていることがわかる。

エネルギー効率については、以下の式が利用されている。

$$\text{エネルギー効率 [bit/J]} = \frac{\text{スループット [bit/sec]}}{\text{消費電力 [W]}}$$

例えば、表 3.36 の 1 行目のデータに対する平均消費電力は、$39.1/407.2 = 0.0960$ となるため、96 μW と算出できる。また、1 ビットの処理に必要なエネルギーを求める場合は、エネルギー効率と処理データ数を用いて、$128/407.2 = 0.3144$ となるため、0.3144 μJ/bit（2.51 μJ/byte）と算出できる。Großらによる研究 [19] で報告されているエネルギー消費量と比

表 3.36　Elsadek らによる Ascon の ASIC 実装評価結果 [14]

名称	インタフェース	データ量	面積コスト	スループット @f_{max}	エネルギー効率
Ascon-128 (Enc.)	no	Short/PT	11 kGE	39 Mbps	407.2 Mbit/mJ
		Short/AD		37 Mbps	371.7 Mbit/mJ
		Short/PT+AD		70 Mbps	640.8 Mbit/mJ
Ascon-128 (Enc.)	no	Long/PT	11 kGE	522 Mbps	2,614.7 Mbit/mJ
		Long/AD		522 Mbps	2,531.3 Mbit/mJ
		Long/PT+AD		531 Mbps	2,600.4 Mbit/mJ

べると大きな違いがあることがわかる。どちらの性能評価も合成結果後のシミュレーションによる見積もりであることが原因として考えられる。電力やエネルギー効率の正確な測定には、レイアウト後の正確なシミュレーションや実チップでの測定が不可欠と言える。

上記の式から、エネルギー効率（1 ジュールのエネルギーで処理できるビット数）を高めるためには、スループット性能を高めるか消費電力を抑制することで達成できると言える。また、スループット性能を高めるためには電力が必要であることから、最適なエネルギー効率はスループット性能と消費電力のトレードオフで決まる。設計段階においては、Unrolled 実装型アーキテクチャでのトレードオフの模索が効果的であり、クロック周波数や供給電力を変更することによるエネルギー効率の最適化も有効な手段となる。

3.3.2　ソフトウェア実装性能

本節では、Ascon のソフトウェア実装性能について、2022 年度に公開された CRYPTREC 外部評価報告書 [53] に基づき、2022 年 9 月現在の調査結果を掲載する。

3.3.2.1　調査対象と性能評価環境

CAESAR コンペティションの最終的なポートフォリオや NIST LWC ファイナリストなどを対象としたソフトウェア実装性能について、eBACS (ECRYPT Benchmarking of Cryptographic Systems)[*3]で幅広い評価結果がまとめられている。ただし、eBACS では Intel Xeon や Arm Cortex-M7 のような処理性能の高い CPU 上での評価結果を中心としており、IoT デバイス向けの低消費電力 CPU 上での評価結果は掲載されていない。本ガイドラインでは、IoT デバイス向けの低消費電力 CPU 上で Ascon のソフトウェア実装性能を評価した 2 つの文献 [21, 48] における評価結果を紹介する。

一般的に、ソフトウェア実装における処理性能は、1 バイトのデータを処理するために必要なサイクル数（cycles/byte）で評価する方法とレイテンシで評価する方法がある。認証暗号やハッシュ関数の処理では、初期化処理などのオーバヘッド時間が必要となり、データ長が短い場合には処理性能が低くなる傾向にある。このため、少ないデータ量に対して暗号化処理を行うようなアプリケーションを対象としてソフトウェア実装性能を評価する場合には、サイクル数を評価するよりもレイテンシを評価する方が適している場合が多い。

一方で、十分な量のデータ量に対して暗号化処理を行う場合にはオーバーヘッドを無視することができるため、必要サイクル数の測定により対象となる暗号アルゴリズムの最適な処理性能を取得することができる。つまり、この場合にはスループット性能が重視されるべきであるため、サイクル数を評価することが適していると言える。

3.3.2.2　実装性能

表 3.37 は文献 [21] で報告された Hira らによる Arm Cortex-M0 上でのレイテンシ評価結果をまとめたものである。Hira らの研究では、Ascon 設計チームが提出したリファレンスコードを Arm Cortex-M0 に移植し、レイテンシ、ROM サイズ、そしてコードサイズを評価している。CPU の動作周波数は 48 MHz である。

[*3] https://bench.cr.yp.to/

測定では、関連データと平文を 0 バイトから 32 バイトまで変化させ、暗号化と復号にかかるレイテンシを分けて評価されている。なお、測定に使用したテストベクトルは、関連データと平文をそれぞれ 2 バイトずつ変化させ、$17 \times 17 = 289$ 通りの組み合わせで構成されている。

表 3.37　Hira らによる Arm Cortex-M0 上での Ascon のレイテンシ評価結果 [21]

名称	暗号化	復号	ROM サイズ	コードサイズ
Ascon-128a	153 msec (0.529 msec)	155 msec (0.536 msec)	30.6 Kbytes	28.6 Kbytes
Ascon-128	183 msec (0.633 msec)	185 msec (0.640 msec)	31.4 Kbytes	29.4 Kbytes
Ascon-80pq	185 msec (0.640 msec)	188 msec (0.650 msec)	31.3 Kbytes	29.3 Kbytes

表中の結果は、289 通りのテストベクトル全ての処理にかかるレイテンシの総和を表している。また、括弧内の数値は、1 つのテストベクトルを処理するために必要となるレイテンシの平均値を表している。

表 3.38 は文献 [48] で報告された Watanabe らによる Arm Cortex-M3 上と AVR ATmega 上でのレイテンシ評価結果をまとめたものである。Watanabe らの研究では、16 バイトの関連データと 16 バイトの平文に対して、暗号化と復号にかかるレイテンシを分けて評価している。CPU の動作周波数は、Arm Cortex-M3 が 84 MHz、AVR ATmega が 16 MHz である。

表 3.38　Watanabe らによる Arm Cortex-M3 上と AVR ATmega 上での Ascon のレイテンシ評価結果 [48]

名称	プラットフォーム	レイテンシ	ROM サイズ	RAM サイズ
Ascon-128 (暗号化)	AVR ATmega @16 Mhz	5.84 msec	9,732 bytes	157 bytes
Ascon-128 (復号)		5.86 msec		181 bytes
Ascon-128 (暗号化)	Arm Cortex-M3 @84 MHz	0.30 msec	4,764 bytes	196 bytes
Ascon-128 (復号)		0.31 msec		121 bytes

テストベクトルや動作周波数が異なるものの、いずれの結果でも数十バイト程度のデータであれば、数 msec 程度でのレイテンシで暗号化処理が可能であることがわかる。また、文献 [48] ではコードサイズが最適化されていることが読み取れる。実装性能をさらに向上させるためには、アルゴリズムの特徴を理解し、CPU に合わせて最適化を図る必要がある。

3.3.3　物理攻撃耐性

本節では、Ascon-128 の物理攻撃耐性を含めた実装性能について、2023 年度に公開された CRYPTREC 外部評価報告書 [54] に基づき、2023 年 9 月現在の調査結果を掲載する。

3.3.3.1　用語

本節で取り扱うサイドチャネル攻撃対策手法とサイドチャネル解析・漏えい評価手法に関する用語を表 3.39 で示す。詳細は付録 A.1 と付録 A.2 を参照されたい。

3.3.3.2　サイドチャネル攻撃対策が施された実装への評価結果

本節では、Kandi らによる Threshold Implementation (TI) を使用した評価結果 [25] と Groß による Domain Oriented Masking (DOM) を使用した評価結果 [17] を紹介する。

表 3.39　3.3.3 節で取り扱う用語

用語	説明	詳細
Threshold Implementation (TI)	2006 年に Nikova らによって提案された秘密分散法に基づくマスキング手法 [38, 39]	付録 A.1.1
Domain Oriented Masking (DOM)	2016 年に Großらによって提案された d 次のプロービングモデルに対して耐性のあるマスキング手法 [17, 18]	付録 A.1.2
相関電力解析	電力のサイドチャネル情報を効率よく解析する手法 [9]、電磁波サイドチャネルに対する解析手法は相関電磁波解析と呼ばれる。	付録 A.2.1
故障利用攻撃	暗号機能を実装したハードウェアの動作中に故意に故障を起こし、故障によって生じた計算誤りを利用して解析を行う手法 [7]	付録 A.2.2
Test Vector Leakage Assessment (TVLA)	サイドチャネルからの漏洩評価における統計的手法、ウェルチの t 検定（Welch's t-test）が利用される。	付録 A.2.3
テンプレート攻撃	事前に攻撃対象モジュールの特性を評価したテンプレートを準備し、このテンプレートを使用してパラメータを操作できない攻撃対象モジュールの秘密鍵を推定する手法	付録 A.2.4

■Kandi らによる TI を使用したサイドチャネル攻撃対策と三重化による故障利用攻撃対策 [25]　2023 年 6 月、Kandi らは Ascon のハードウェア実装性能に関する評価結果を報告した [25]。具体的には、サイドチャネル攻撃対策が施されていない実装に加え、TI を使用したサイドチャネル攻撃対策と計算の三重化による故障利用攻撃対策が施された実装への評価結果が紹介されている。サイドチャネル攻撃対策と故障利用攻撃対策は互いに独立した概念に基づき実装されることから、それぞれの対策が相互に影響しないと言われている。つまり、要求仕様に応じて、いずれかの対策を施して実装することも、両方の対策を施して実装することも可能である。

最初に、TI を使用したサイドチャネル攻撃対策が施された実装への評価結果を紹介する。表 3.40 と表 3.41 は、それぞれ Ascon の暗号化処理と復号処理に関する FPGA 実装と ASIC 実装の評価結果をまとめている。なお、括弧内の数値は、対策を施していない実装における暗号化とタグ生成の評価結果を基準とした割合を表している。FPGA 実装では 28nm テクノロジを有する Kintex-7 が使用されている。表 3.40 から、暗号化処理と復号処理での実装性能の違いはほとんど見られないことがわかる。3 シェア TI を使用した実装では、対策を施していない実装と比べ、暗号化処理と復号処理のいずれの回路サイズも 4 倍以上の LUT を必要としている。一方、クロック周期については、10% 程度の増加に抑えることができる。

表 3.40　Kandi らによる Kintex-7 上での Ascon の FPGA 実装評価結果

コア	面積コスト [LUT]	クロック周期 [psec]
暗号化とタグ生成	944 (1.00)	5,525 (1.00)
復号とタグ検証	1,058 (1.12)	5,525 (1.00)
暗号化とタグ生成（3 シェア TI）	3,977 (4.21)	6,024 (1.09)
復号とタグ検証（3 シェア TI）	3,795 (4.02)	6,010 (1.09)

ASIC 実装では STM 130nm ライブラリが使用され、TI で保護された Ascon S-box のゲートサイズが 56 gates、線形層のゲートサイズが 320 gates となっている。このことから、1 サイクルで 1 ラウンドを処理する回路において、組合せのゲートサイズは少なくとも 12.6 Kgates 程度の面積コストが必要となる。なお、実際の ASIC 実装においては、内部状態を保持するフリップフロップ回路、インタフェース回路、乱数生成器が必要となる。また、表 3.41 から、FPGA と同様に、暗号化処理と復号処理での違いがほとんど見られないことがわかる。

文献 [25] において 3 シェア TI を採用した理由は、Ascon S-box の代数次数が 2 であり、TI のシェアの数が Ascon S-box

表 3.41　Kandi らによる STM 130nm 上での Ascon の ASIC 実装評価結果（サイドチャネル攻撃対策との比較結果）

コア	面積コスト [μm^2]	クリティカルパス遅延時間 [psec]
暗号化とタグ生成	73,803 (1.00)	8,595 (1.00)
復号とタグ検証	71,873 (0.97)	8,586 (1.00)
暗号化とタグ生成（3 シェア TI）	273,857 (3.71)	10,001 (1.16)
復号とタグ検証（3 シェア TI）	274,688 (3.72)	9,981 (1.16)

表 3.42　Kandi らによる STM 130nm 上での Ascon の ASIC 実装評価結果（サイドチャネル攻撃対策、故障利用攻撃対策との比較結果）

サイドチャネル攻撃対策	故障利用攻撃対策	面積コスト [μm^2]	クリティカルパス遅延時間 [psec]
未対策	未対策	98,524 (1.00)	8,520 (1.00)
未対策	三重化	258,224 (2.62)	8,518 (1.00)
3 シェア TI	未対策	364,320 (3.70)	9,830 (1.15)
3 シェア TI	三重化	948,544 (9.63)	9,832 (1.15)

の代数次数に 1 を加えた 3 である必要があるためである。Ascon S-box に対する 3 シェア TI の構成方法の詳細は、文献 [25] の 4.3 節、または文献 [54] の 4.5.4 節を参照されたい。

次に、三重化による故障利用攻撃対策が施された実装への評価結果を紹介する。差分故障解析への有効な対策の 1 つとして暗号化処理の二重化を紹介したが、文献 [25] では暗号化処理の三重化、つまり同じ暗号化処理を 3 回行い、3 つの出力結果が全て異なる場合には乱数を出力するという対策を提案した。

暗号化処理の三重化により、Ascon のコア部分における面積コストは単純に 3 倍となる。より正確には、多数決により出力結果を決定する処理が追加されるため、3 倍よりも大きくなる。表 3.42 は、STM 130nm 上での Ascon の ASIC 実装に関し、対策が施されていない場合、三重化による対策を施した場合、3 シェア TI による対策を施した場合、3 シェア TI による対策と三重化による対策の両方を施した場合における性能評価（面積コスト、クリティカルパス遅延時間）をまとめている。この表からわかるように、インタフェースなどの面積コストが増えないため、全体としては面積コストは 3 倍弱の増加となっている。また、空間的な三重化を施しているため、クリティカル遅延時間への影響はない。

■Großによる DOM を使用したサイドチャネル攻撃対策 [17]　2018 年 6 月、Großは自身の学位論文 [17] で DOM によるサイドチャネル攻撃対策を施した Ascon の実装性能評価結果を体系的にまとめている。本学位論文では、DOM のバリエーションである Unified Masking (UMA) と Low-Latency Masking (LOLA) も提案されている。UMA は、暗号アルゴリズムのデータパスにレジスタを追加することで、安全性の観点でクリティカルとされるデータを適切に制御し、DOM の乱数コストの削減を目指したものである。レジスタを追加することから 1 ラウンドの処理に必要となるサイクル数が増加するため、レイテンシは増加しスループットは低下するものの、必要となるフレッシュな乱数は少なくて済む。UMA とは対照的に、LOLA ではレジスタによるステージ数を減らし、処理パフォーマンスの向上を目指したものである。代わりに、非線形処理におけるシェア数が増加するため、より多くのデータの冗長性や追加の回路が必要となり、乱数コストも増加する。

表 3.43 は、DOM を使用したサイドチャネル攻撃対策を施した ASIC 実装の評価結果をまとめたものである。ここで、1 次（又は 5 次）安全な DOM、UMA、LOLA とは、それぞれ 1 次（又は 5 次）プロービングモデルに耐性のある DOM、UMA、LOLA を使用した実装のことを意味する。本実装では UMC-90nm Low-K の CMOS ライブラリが使用されている。

UMA については、レイテンシとスループット性能を犠牲にすることで、乱数コストを抑えられることがわかる。ただし、現実的な実装となる 1 次 UMA は、1 次 DOM と比べて面積コストと乱数コストがほぼ同じである。シェア数が少ない場合には、UMA の実装コスト低下は限定的であると言える。5 次 UMA では、5 次 DOM と比べて必要となるフレッシュな乱数のコストを削減することに成功しているが、レジスタの追加などによって面積コストが増加してしまう。

LOLA 実装については、1 ラウンドの処理を 1 サイクルで実行可能となるため、低レイテンシが実現できていることがわかる。ただし、面積コストは DOM や UMA と比べて大きくなり、必要となる乱数のコストが多い 5 次 LOLA では、1 サイ

表 3.43　Großによる UMC-90nm Low-K 上での Ascon の ASIC 実装評価結果（DOM、UMA、LOLA との比較結果）

デザイン	面積コスト [KGE]	サイクル数 [cycle/round]	スループット性能 [Mbps]	乱数コスト [bit/cycle]
1 次安全な DOM	28.89	3	2,250	320
1 次安全な UMA	27.18	3	2,250	320
1 次安全な LOLA	42.75	1	2,770	2,048
5 次安全な DOM	161.87	3	1,860	4,800
5 次安全な UMA	220.01	7	850	3,520
5 次安全な LOLA	339.82	1	2,990	18,432

クル当たり約 18K ビットと非常に多くのフレッシュな乱数を必要としている。

通常の TI と比べて少ないシェア数でサイドチャネル攻撃対策を実現できる DOM は、設計手法としても興味深いものとなっている。DOM とその 2 つのバリエーションにより、実装コスト、処理パフォーマンス、そして必要となる乱数コストのトレードオフは大幅に広がっている。また、設計者の選択肢が増えたことに加え、設計手法自体が規則的、かつ汎用的なマスキングツールで対策を実現できることは、生産性の向上に繋がると考えられる。

3.3.3.3　物理攻撃耐性評価

本節では、Samwel らによる相関電力解析の評価結果 [42]、Betina らによる相関電力解析と TVLA の評価結果 [5]、そして Mohajerani らによる相関電力解析と TVLA の評価結果 [34] を紹介する。

その他の最新動向として、Gigerl らによる TVLA の評価結果 [16]、Liu らによるサイドチャネル情報漏洩の評価結果 [30]、そして You らによるテンプレート攻撃の評価結果 [52] が報告されている。これらの評価結果について本ガイドラインでは取り扱わないため、詳細は文献 [54] を参照されたい。

■Samwel らによる相関電力解析 [42]　2017 年 5 月、Samwel らは Ascon のハードウェア実装に対して相関電力解析を実施した結果を初めて報告した [42]。具体的には、Ascon の非線形処理である S-box の出力に対して効率の良い選択関数を提案し、この選択関数を使用してサイドチャネル攻撃対策を施していない FPGA 実装と 3 シェア TI によるサイドチャネル攻撃対策を施した FPGA 実装に対して、相関電力攻撃に成功したと報告されている。選択関数の構成方法の詳細については、文献 [42] の 5.1 節、または文献 [54] の 4.1.3 節を参照されたい。

Samwel らは、サイドチャネル攻撃対策を施していない Ascon を SAKURA-G 上に搭載された FPGA Spartan-6 に実装し、この実装に対して 50K 個の波形トレース[*4]から全ての秘密鍵ビットの導出に成功したと報告している。これにより、提案された選択関数による電力モデルが効果的であることが明らかとなった。また、3 シェア TI によるサイドチャネル攻撃対策を施した Ascon に対して、シミュレーションにて同様の相関電力攻撃を実行し、900K 個の波形トレースで全ての秘密鍵ビットの導出に成功したと報告している。

■Betina らによる相関電力解析と TVLA [5]　2022 年 8 月、Betina らは Ascon のソフトウェア実装に対する相関電力解析と TVLA を用いた安全性評価の結果を報告した [5]。Betina らは、Ascon の設計者チームが公開している Ascon-128 のソースコードを Arm-V6 上に実装し、サイドチャネル情報として電力波形を使用した。電力測定には、Riscure 社の Piñata development board を使用している。当該ボードには、32 ビットの Arm マイクロコントローラをベースとする SoC STM32F407IGT6 が搭載されており、その動作周波数は 168 MHz である。電力波形の取得には、Riscure 社のカレントプローブ（型番不明）と Picoscope 社のオシロスコープ（model 3206D）を使用している。なお、本報告での相関電力解析では、Samwel ら [42] が提案した選択関数を使用している。

最初に、Betina らは 50K 個の波形トレースを使用し、サイドチャネル攻撃対策を施していない Ascon の暗号化処理に対して TVLA を行った。その結果、Ascon の初期化処理フェーズにおいて t 値が閾値を大きく超えていることが示された。次

[*4] オシロスコープ等で取得した物理情報の時系列変化の軌跡を波形トレース、あるいは単にトレースと呼ぶ。波形トレースの単位として用いることもある。

に、Ascon の初期化処理フェーズに特化し、Samwel ら [42] の攻撃手法に従い、100K 個の波形トレースを使用した相関電力攻撃を実施して正しい秘密鍵の復元に成功したことが示された。なお、2 つある選択関数の使用において、攻撃の成功確率に差が生じることが明らかとなった。

Betina らはサイドチャネル攻撃（マスキング）対策を施した Ascon に対しても同様に TVLA と相関電力攻撃を実施した。このソフトウェア実装では、乱数をほとんど使用しない 2〜4 個のシェアで対策が施されている。最初に、15K 個の波形トレースを使用し、Ascon の初期化処理フェーズの最初で処理される Ascon permutation に対して、相関電力解析で使用する攻撃箇所（サンプル時間）の特定が行われた。この際、ナンスはランダムに変化させ、その他のパラメータは全て固定としている。その後、15K 個の波形トレースを使用して相関電力解析を実施した結果、暗号化処理の 2 個の中間値を利用する 2 次相関電力解析でも攻撃は成功しないことが示された。その原因は、使用した波形トレースの数が少なかったことにあると考察されている。

■Mohajerani らによる相関電力解析と TVLA [34]　2023 年 6 月、Mohajerani らは NIST LWC ファイナリスト 10 方式に対する物理耐性評価を行う機関を集め、サイドチャネル攻撃耐性に関する調査やベンチマーク評価行った結果を報告するとともに、物理攻撃耐性に関する一般的な評価フレームワークを提案した [34]。本研究プロジェクトに参画した大学、研究所、そして企業は以下に示す 7 つの機関である。

1. IAIK, Graz University of Technology, Austria
2. CCSL, Shanghai Jiao Tong University, China
3. HSCP Lab, Tsinghua University, Beijing, China
4. Secure-IC, France
5. CERG, George Mason University, USA
6. Ruhr-Universitat Bochum, Germany
7. CESCA Lab, Radboud University, the Netherlands

また、サイドチャネル攻撃対策技術の安全性を評価するとともに、対策技術を追加することによって実装コストと処理パフォーマンスに与える影響について実証実験を行っている。

表 3.44 は、サイドチャネル攻撃対策を施した Ascon の FPGA 実装に対する安全性評価結果をまとめたものである。ソースコード Ascon-128_Graz_d1 の実装では DOM [17] によるサイドチャネル攻撃対策が施されており、ソースコード Ascon-128_Bochum_d1 の実装ではサイドチャネル攻撃対策が施されていない Ascon-128_Graz_x1 のソースコードをベースとしてマスキング対策が施されている。また、サイドチャネル攻撃対策を施していない HDL (Hardware Description Language) コードから乱数マスキング対策を施した HDL コードを半自動生成するために、AGEMA [27] と呼ばれるツールが使用されている[*5]。

サイドチャネル攻撃対策を施した Ascon の FPGA 実装に対する安全性評価結果では、サイドチャネル情報として電力と電磁波が使用され、これらのサイドチャネル情報に対して TVLA、χ^2 検定、そして相関電力解析による情報漏洩の可能性の有無が解析された。波形トレース数は約 100 万から 1000 万程度である。CERG による評価のみ TVLA の閾値である 4.5σ を超えたと報告されているが、他の機関からは情報漏洩の可能性がないと報告されている。CERG による Ascon-128_Graz_d1 のテストでは、数個（3〜10）のサンプルで閾値の 4.5σ を超えたものの、これらのテストでは攻撃対象のクロックと同期したサンプリングクロックを使用していたことが原因であると考察されている。

表 3.45 は、サイドチャネル攻撃対策を施した Ascon のソフトウェア実装に対する安全性評価結果をまとめたものである。全て Arm Cortex-M4 上で実装されたものであり、ソースコードは Ascon-128_Graz_d1 と Ascon-128_Graz_d2 が使用されている。

いずれの安全性評価においても、電磁波に関するサイドチャネル情報から取得した波形データが使用されている。評価の結果、相関電力解析による秘密鍵の復元には成功していない。CESCA グループによる 2 次の相関電力解析では、15M 個の波形トレースを使用しても Ascon-128_Graz_d1 の秘密鍵に関する情報を明らかにできないと報告されている。参考までに、

[*5] AGEMA はサイドチャネル攻撃に対して保護する必要があるワイヤとゲートを特定し、これらに対して必要な乱数マスキング対策を施すことが可能であるものの、制御ロジックに対して乱数マスキング対策を施すことができないため、一部のコードに対しては手動でマスキング対策を施す必要がある。これが半自動生成ツールと呼ばれる理由である。

表 3.44　サイドチャネル攻撃対策を施した Ascon の FPGA 実装に対する安全性評価結果

ソースコード（評価機関）	プラットフォーム	オシロスコープ	評価手法（サイドチャネル）	波形数（トレース）	評価結果
Ascon-128_Bochum_d1（CERG）	Artix-7 (CW305)	FOBOS3 ADC	TVLA（電力）	10M	リーク有 (1.5 M トレース)
Ascon-128_Bochum_d1（IAIK）	Artix-7 (CW305)	PicoScope 6404C	TVLA（電力）	10M	リーク無
Ascon-128_Bochum_d1（CCSL）	Kintex-7 (SAKURA-X)	LeCroy 610Zi	TVLA（電磁波）	1M	リーク無
Ascon-128_Bochum_d1（CCSL）	Kintex-7 (SAKURA-X)	LeCroy 610Zi	χ^2 検定（電磁波）	1M	リーク無
Ascon-128_Bochum_d1（CCSL）	Kintex-7 (SAKURA-X)	LeCroy 610Zi	相関電力解析（電磁波）	11M	リーク無
Ascon-128_Graz_d1（HSCP）	Spartan-6 (SAKURA-G)	WaveRunner 8404M	TVLA（電力）	10M	リーク無

表 3.45　サイドチャネル攻撃対策を施した Ascon のソフトウェア実装に対する安全性評価結果

ソースコード（評価機関）	評価プラットフォーム	オシロスコープ	評価手法（サイドチャネル）	波形数（トレース）	評価結果
Ascon-128_Graz_d1（CESCA）	Arm Cortex-M4 (STM32F407)	Pico 3206D	2 次相関電力解析（電磁波）	15M	リーク無
Ascon-128_Graz_d2（CCSL）	Arm Cortex-M4 (STM32F303)	Pico 3203D	TVLA（電磁波）	60K	リーク無
Ascon-128_Graz_d2（CCSL）	Arm Cortex-M4 (STM32F303)	Pico 3203D	χ^2 検定（電磁波）	60K	リーク無
Ascon-128_Graz_d2（CCSL）	Arm Cortex-M4 (STM32F303)	Pico 3203D	相関電力解析（電磁波）	60K	リーク無

サイドチャネル攻撃対策を施していない Ascon のソフトウェア実装に対する相関電力解析では 500K 個の波形トレースを使用して秘密鍵の復元に成功している。このことから、使用したソースコードのマスキング対策が正常に機能していることがわかる。

Mohajerani らは、サイドチャネル攻撃対策を施した FPGA 実装における面積コストと処理パフォーマンスへの影響についても考察している。Ascon-128_Bochum_d1 における FPGA 実装では、サイドチャネル攻撃対策を施すことにより、面積コストが約 3 倍程度増加し、スループット性能が約 1/3 倍に低下していることが報告されている。一方、Ascon-128_Graz_d1 における FPGA 実装では、サイドチャネル攻撃対策を施すことにより、面積コストの増加が約 2 倍程度と Ascon-128_Bochum_d1 に比べて少なく、スループット性能も Ascon-128_Bochum_d1 ほど低下していないことが報告されている。これは、DOM を人手で実装したことにより、効率の良い対策技術が実現できたものと考えられる。なお、ソフトウェア実装に関する実験結果は文献 [34] には記載されていない。

参考文献

[1] Kazumaro Aoki, Tetsuya Ichikawa, Masayuki Kanda, Mitsuru Matsui, Shiho Moriai, and Junko Nakajima Toshio Tokita. Specification of Camellia - a 128-bit Block Cipher, 2001. https://info.isl.ntt.co.jp/crypt/camellia/dl/01espec.pdf（2023-10-07 閲覧）.

[2] ARM. AMBA 3 APB Protocol Specification v2.0, 2008. https://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ihi0024c/index.html（2023-10-07 閲覧）.

[3] Subhadeep Banik, Andrey Bogdanov, Takanori Isobe, Kyoji Shibutani, Harunaga Hiwatari, Toru Akishita, and Francesco Regazzoni. Midori: A block cipher for low energy. In *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II*, pages 411–436, 2015.

[4] William C. Barker and Elaine Barker. Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, January 2012. https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-67r1.pdf.

[5] Lejla Batina, Ileana Buhan, Lukasz Chmielewski, Ellen Gunnarsdóttir, Vahid Jahandideh, Tom Stock, and Léo Weissbart. Side-Channel Evaluation Report on Implementations of Several NIST LWC Finalists, 2022. Nijmegen : Cryptographic Engineering & Side-Channel Analysis (CESCA) Lab, https://github.com/rweather/lwc-finalists/tree/5d2b22c9ff7744be429cabda0c078ea5b7b6f79e（2023-10-07 閲覧）.

[6] Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. SIMON and SPECK: block ciphers for the internet of things. *IACR Cryptology ePrint Archive*, 2015:585, 2015.

[7] Eli Biham and Adi Shamir. Differential Fault Analysis of Secret Key Cryptosystems. In Burton S. Kaliski Jr., editor, *Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 1997, Proceedings*, volume 1294 of *Lecture Notes in Computer Science*, pages 513–525. Springer, 1997.

[8] Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knezevic, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and Tolga Yalçin. PRINCE - A low-latency block cipher for pervasive computing applications - extended abstract. In *Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings*, pages 208–225, 2012.

[9] Eric Brier, Christophe Clavier, and Francis Olivier. Correlation Power Analysis with a Leakage Model. In Marc Joye and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems - CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2004.

[10] Renesas Electronics Corporation. RL78 ファミリ. https://japan.renesas.com/products/mpumcu/rl78/index.jsp?campaign=tb_prod（2023-10-07 閲覧）.

[11] Joan Daemen, Seth Hoffert, Michaël Peeters, Gilles Van Assche, Ronny Van Keer, and Silvia Mella. Ketje. https://ketje.noekeon.org/（2023-10-07 閲覧）.

[12] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon v1.2. https://competitions.cr.yp.to/round3/asconv12.pdf（2023-10-07 閲覧）.

[13] Morris Dworkin. NIST SP800-38D – Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, August 2015. https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38d.pdf.

[14] Islam Elsadek, Sohrab Aftabjahani, Doug Gardner, Erik MacLean, John Ross Wallrabenstein, and Eslam Yahya Tawfik. Hardware and Energy Efficiency Evaluation of NIST Lightweight Cryptography Standardization Finalists. In *IEEE International Symposium on Circuits and Systems, ISCAS 2022, Austin, TX, USA, May 27 - June 1, 2022*, pages 133–137. IEEE, 2022.

[15] Farnoud Farahmand, William Diehl, Abubakr Abdulgadir, Jens-Peter Kaps, and Kris Gaj. Improved Lightweight Implementations of CAESAR Authenticated Ciphers. In *26th IEEE Annual International Symposium on Field-Programmable Custom Computing Machines, FCCM 2018, Boulder, CO, USA, April 29 - May 1, 2018*, pages 29–36. IEEE Computer Society, 2018.

[16] Barbara Gigerl, Florian Mendel, Martin Schläffer, and Robert Primas. Efficient Second-Order Masked Software Implementations of Ascon in Theory and Practice, 2023. NIST, Lightweight Cryptography Workshop 2023. https://csrc.nist.gov/csrc/media/Events/2023/lightweight-cryptography-workshop-2023/documents/accepted-papers/04-efficient-second-order-masked-software.pdf.

[17] Hannes Groß. Domain-Oriented Masking–Generically Masked Hardware Implementations, 2018. PhD Thesis, IAIK, Graz University of Technology. https://diglib.tugraz.at/download.php?id=5c80ea0c43a56&location=browse（2023-10-07 閲覧）.

[18] Hannes Groß, Stefan Mangard, and Thomas Korak. Domain-Oriented Masking: Compact Masked Hardware Implementations with Arbitrary Protection Order. *IACR Cryptol. ePrint Arch.*, page 486, 2016.

[19] Hannes Groß, Erich Wenger, Christoph Dobraunig, and Christoph Ehrenhöfer. Suit up! - Made-to-Measure Hardware Implementations of Ascon. In *2015 Euromicro Conference on Digital System Design, DSD 2015, Madeira, Portugal, August 26-28, 2015*, pages 645–652. IEEE Computer Society, 2015.

[20] Jian Guo, Thomas Peyrin, Axel Poschmann, and Matthew J. B. Robshaw. The LED block cipher. In *Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings*, pages 326–341, 2011.

[21] Ryota Hira, Tomoaki Kitahara, Daiki Miyahara, Yuko Hara-Azumi, Yang Li, and Kazuo Sakiyama. Software Evaluation for Second Round Candidates in NIST Lightweight Cryptography. *IACR Cryptol. ePrint Arch.*, page 591, 2022.

[22] Ekawat Homsirikamol, William Diehl, Ahmed Ferozpuri, Farnoud Farahmand, Panasayya Yalla, Jens-Peter Kaps, and Kris Gaj. CAESAR Hardware API. *IACR Cryptol. ePrint Arch.*, page 626, 2016.

[23] Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, and Sumio Morioka. CLOC and SILC—Authenticated Encryption Schemes for Constrained Devices, 2014. https://www.nuee.nagoya-u.ac.jp/labs/tiwata/AE/（2023-10-07 閲覧）.

[24] Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, and Sumio Morioka. CLOC: authenticated encryption for short input. In *Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers*, pages 149–167, 2014.

[25] Aneesh Kandi, Anubhab Baksi, Tomas Gerlich, Sylvain Guilley, Peizhou Gan, Jakub Breier, Anupam Chattopadhyay, Ritu Ranjan Shrivastwa, Zdenek Martinasek, and Shivam Bhasin. Hardware Implementation of Ascon, 2023. NIST, Lightweight Cryptography Workshop 2023. https://csrc.nist.gov/csrc/media/Events/2023/lightweight-cryptography-workshop-2023/documents/accepted-papers/07-hardware-implementation-of-ascon.pdf.

[26] Jens-Peter Kaps, William Diehl, Michael Tempelmeier, Ekawat Homsirikamol, and Kris Gaj. Hardware API for Lightweight Cryptography, 2019. https://cryptography.gmu.edu/athena/LWC/LWC_HW_API.pdf（2023-10-07 閲覧）.

[27] David Knichel, Pascal Sasdrich, and Amir Moradi. Generic Hardware Private Circuits Towards Automated Generation of Composable Secure Gadgets. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2022(1):323–344, 2022.

[28] Lars R. Knudsen and Gregor Leander. PRESENT - block cipher. In *Encyclopedia of Cryptography and Security, 2nd Ed.*, pages 953–955. 2011.

[29] Ted Krovetz and Phillip Rogaway. OCB (v1.1). `https://competitions.cr.yp.to/round3/ocbv11.pdf`（2023-10-07 閲覧）.

[30] Zhenyuan Liu and Patrick Schaumont. Root-cause Analysis of the Side Channel Leakage from Ascon Implementations, 2023. NIST, Lightweight Cryptography Workshop 2023. `https://csrc.nist.gov/csrc/media/Events/2023/lightweight-cryptography-workshop-2023/documents/accepted-papers/13-root-cause-analysis-of-side-channel-leakage.pdf`.

[31] Mitsuru Matsui and Yumiko Murakami. Minimalism of software implementation - extensive performance analysis of symmetric primitives on the RL78 microcontroller. In *Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers*, pages 393–409, 2013.

[32] Mitsuru Matsui and Yumiko Murakami. AES smaller than s-box - minimalism in software design on low end microcontrollers. In *Lightweight Cryptography for Security and Privacy - Third International Workshop, LightSec 2014, Istanbul, Turkey, September 1-2, 2014, Revised Selected Papers*, pages 51–66, 2014.

[33] Kazuhiko Minematsu. AES-OTR v1. `https://competitions.cr.yp.to/round1/aesotrv1.pdf`（2023-10-07 閲覧）.

[34] Kamyar Mohajerani, Luke Beckwith, Abubakr Abdulgadir, Eduardo Ferrufino, Jens-Peter Kaps, and Kris Gaj. SCA Evaluation and Benchmarking of Finalists in the NIST Lightweight Cryptography Standardization Process. *IACR Cryptol. ePrint Arch.*, page 484, 2023.

[35] Kamyar Mohajerani, Richard Haeussler, Rishub Nagpal, Farnoud Farahmand, Abubakr Abdulgadir, Jens-Peter Kaps, and Kris Gaj. FPGA Benchmarking of Round 2 Candidates in the NIST Lightweight Cryptography Standardization Process: Methodology, Metrics, Tools, and Results. *IACR Cryptol. ePrint Arch.*, page 1207, 2020.

[36] Amir Moradi, Axel Poschmann, San Ling, Christof Paar, and Huaxiong Wang. Pushing the limits: A very compact and a threshold implementation of AES. In *Advances in Cryptology - EUROCRYPT 2011 - 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011. Proceedings*, pages 69–88, 2011.

[37] National Institute of Standards and Technology. FIPS 197-4 – Secure Hash Standard (SHS), November 2001. `https://csrc.nist.gov/publications/fips/fips197/fips-197.pdf`.

[38] Svetla Nikova, Christian Rechberger, and Vincent Rijmen. Threshold Implementations Against Side-Channel Attacks and Glitches. In Peng Ning, Sihan Qing, and Ninghui Li, editors, *Information and Communications Security, 8th International Conference, ICICS 2006, Raleigh, NC, USA, December 4-7, 2006, Proceedings*, volume 4307 of *Lecture Notes in Computer Science*, pages 529–545. Springer, 2006.

[39] Svetla Nikova, Vincent Rijmen, and Martin Schläffer. Secure hardware implementation of nonlinear functions in the presence of glitches. *J. Cryptol.*, 24(2):292–321, 2011.

[40] Konstantinos Papagiannopoulos and Aram Verstegen. Speed and size-optimized implementations of the PRESENT cipher for tiny AVR devices. In *Radio Frequency Identification - Security and Privacy Issues 9th International Workshop, RFIDsec 2013, Graz, Austria, July 9-11, 2013, Revised Selected Papers*, pages 161–175, 2013.

[41] Behnaz Rezvani and William Diehl. Hardware Implementations of NIST Lightweight Cryptographic Candidates: A First Look. *IACR Cryptol. ePrint Arch.*, page 824, 2019.

[42] Niels Samwel and Joan Daemen. DPA on hardware implementations of Ascon and Keyak. In *Proceedings of the Computing Frontiers Conference, CF'17, Siena, Italy, May 15-17, 2017*, pages 415–424. ACM, 2017.

[43] Yu Sasaki, Yosuke Todo, Kazumaro Aoki, Yusuke Naito, Takeshi Sugawara, Yumiko Murakami, Mitsuru Matsui, and Shoichi Hirose. Minalpher. https://info.isl.ntt.co.jp/crypt/minalpher/index.html（2023-10-07 閲覧）.

[44] Kyoji Shibutani, Takanori Isobe, Harunaga Hiwatari, Atsushi Mitsuda, Toru Akishita, and Taizo Shirai. Piccolo: An ultra-lightweight blockcipher. In *Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings*, pages 342–357, 2011.

[45] Taizo Shirai, Kyoji Shibutani, Toru Akishita, Shiho Moriai, and Tetsu Iwata. The 128-bit blockcipher CLEFIA (extended abstract). In *Fast Software Encryption, 14th International Workshop, FSE 2007, Luxembourg, Luxembourg, March 26-28, 2007, Revised Selected Papers*, pages 181–195, 2007.

[46] Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, and Eita Kobayashi. TWINE : A lightweight block cipher for multiple platforms. In *Selected Areas in Cryptography, 19th International Conference, SAC 2012, Windsor, ON, Canada, August 15-16, 2012, Revised Selected Papers*, pages 339–354, 2012.

[47] Michael Tempelmeier, Fabrizio De Santis, Georg Sigl, and Jens-Peter Kaps. The CAESAR-API in the real world - Towards a fair evaluation of hardware CAESAR candidates. In *2018 IEEE International Symposium on Hardware Oriented Security and Trust, HOST 2018, Washington, DC, USA, April 30 - May 4, 2018*, pages 73–80. IEEE Computer Society, 2018.

[48] Yuhei Watanabe, Hideki Yamamoto, and Hirotaka Yoshida. Performance Evaluation of NIST LWC Finalists on AVR ATmega and ARM Cortex-M3 Microcontrollers. *IACR Cryptol. ePrint Arch.*, page 1071, 2022.

[49] Hongjun Wu. ACORN v2. https://competitions.cr.yp.to/caesar-submissions.html/（2023-10-07 閲覧）.

[50] Hongjun Wu and Tao Huang. The JAMBU Lightweight Authentication Encryption Mode. https://competitions.cr.yp.to/round3/jambuv21.pdf（2023-10-07 閲覧）.

[51] Panasayya Yalla and Jens-Peter Kaps. Evaluation of the CAESAR hardware API for lightweight implementations. In *International Conference on ReConFigurable Computing and FPGAs, ReConFig 2017, Cancun, Mexico, December 4-6, 2017*, pages 1–6. IEEE, 2017.

[52] Shih-Chun You, Markus G. Kuhn, Sumanta Sarkar, and Feng Hao. Low Trace-Count Template Attacks on 32-bit Implementations of Ascon AEAD. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2023(4):344–366, 2023.

[53] 崎山一男. 軽量暗号の実装性能に関する調査及び評価（NIST 軽量暗号コンペティションファイナリスト）（文書番号：CRYPTREC EX-3205-2022）, 2022. https://www.cryptrec.go.jp/exreport/cryptrec-ex-3205-2022.pdf.

[54] 崎山一男. 軽量暗号 Ascon の実装性能に関する調査及び評価（文書番号：CRYPTREC EX-3301-2023）, 2023.

第 4 章

代表的な軽量暗号

4.1 ブロック暗号

本節では、主要な軽量ブロック暗号として CLEFIA、LED、Midori、Piccolo、PRESENT、PRINCE、SIMON、SPECK、TWINE の調査結果をまとめる。調査対象は、主要国際学会で発表されており、現段階で有力な攻撃法が発見されておらず、かつ十分な実装性能を持つと考えられるアルゴリズムとした。 また、CLEFIA、PRESENT、LEA が軽量ブロック暗号に関係する ISO/IEC（ISO/IEC 29192-2）[66] で規格化されている状況を鑑み、本稿では 2016 年度版ガイドライン [38] に掲載されていない LEA を新たな調査対象とし、その調査結果をまとめる。

各アルゴリズムのブロック長、鍵長といった基本入出力情報に加え、全体構造、および構成段数を記載している。鍵長やブロック長によって個別の名称が与えられているアルゴリズムについては、それぞれ個別の名称も記載した。アルゴリズムの特徴としては、主に提案論文で述べられている設計者らの主張を可能な限りそのまま記載した。

各アルゴリズムの安全性解析状況については、2021 年度に公開された CRYPTREC 外部評価報告書 [136] に基づき、2021 年 9 月時点の状況を記載している。文献 [136] は、2016 年度版ガイドライン [38] に掲載されている暗号アルゴリズムを中心とした代表的な軽量暗号の安全性評価に関する動向調査を行い、2021 年 9 月時点でこれらの軽量暗号に対し現実的な脅威に繋がる脆弱性が指摘されているか否かを明らかにしたものである。なお、新たに調査対象として追加した LEA について、文献 [136] では安全性解析状況だけでなく、仕様等（設計者、発表年、仕様参照先、特徴、主な実装評価結果、標準化状況）もまとめられているため、文献 [136] の記載内容に従って調査結果をまとめた。さらに、文献 [136] ではバイクリーク攻撃とその派生攻撃が提案された軽量ブロック暗号（Midori、Piccolo、PRESENT、TWINE が該当）に関し、これらの攻撃が提案された事実について記載されているものの、これらの攻撃を除いた解析手法の中から最大の攻撃可能段数を達成するものを最良の攻撃としてラベル付けされている。本ガイドラインにおいても文献 [136] の方針に従うものとする。

ハードウェア実装性能調査では、主に十分な評価が行われていると考えられる ASIC での実装性能評価を調査し、実装ゲート規模（GE）、1 ブロックの演算に必要なサイクル数（cycles/block）、および 100kHz におけるスループットを記載している。また、ソフトウェア実装性能調査では、ハイエンド CPU での実装結果として 1 バイトの処理に必要なサイクル数（cycles/byte）を記載し、ローエンド CPU での実装結果として cycles/byte に加えて ROM、RAM 使用量を記載した。

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>CLEFIA</td></tr>
<tr><td>設計者</td><td>Taizo Shirai[1], Kyoji Shibutani[1], Toru Akishita[1], Shiho Moriai[1], Tetsu Iwata[2]
(1: Sony Corporation/Japan, 2: Nagoya University/Japan)</td></tr>
<tr><td>発表年</td><td>2007 (FSE 2007 [109])</td></tr>
<tr><td>仕様参照先</td><td>FSE 2007 [109]、設計者ウェブサイト [37]</td></tr>
<tr><td>特徴</td><td>設計者らは、高い安全性を保ちつつ、ハードウェア、ソフトウェアの両実装形態で高い実装性能を持つと主張している。また、AES と同じインタフェースに対応している点も特長である。
<table>
<tr><td>全体構造</td><td colspan="3">4-line type-II 一般化 Feistel 型</td></tr>
<tr><td>ブロック長 [bit]</td><td colspan="3">128</td></tr>
<tr><td>鍵長 [bit]</td><td>128
(CLEFIA-128)</td><td>192
(CLEFIA-192)</td><td>256
(CLEFIA-256)</td></tr>
<tr><td>構成段数 [段]</td><td>18</td><td>22</td><td>26</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [23, 27, 81, 82, 92, 119, 125, 131] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
最良の攻撃は、2015 年に提案された Li ら [81] による切り詰め差分攻撃であり、14 段に簡略化した CLEFIA-128、14 段に簡略化した CLEFIA-192、15 段に簡略化した CLEFIA-256 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><td>Algorithm</td><td>Area
[GE]</td><td>Cycles/block</td><td>Throughput@100kHz
[kbps]</td><td>Ref.</td></tr>
<tr><td>CLEFIA-128 (Enc)</td><td>2,488</td><td>328</td><td>39.0</td><td>[4]</td></tr>
<tr><td>CLEFIA-128 (Enc/Dec)</td><td>2,604</td><td>328/320</td><td>39.0/40.0</td><td>[4]</td></tr>
<tr><td>CLEFIA-128 (Enc/Dec)</td><td>5,979</td><td>18</td><td>711.1</td><td>[109]</td></tr>
</table>
ソフトウェア実装評価結果 (ローエンド CPU)
<table>
<tr><td>Algorithm</td><td>ROM
[byte]</td><td>RAM
[byte]</td><td>Cycles/byte
[Enc/Dec]</td><td>Platform</td><td>Ref.</td></tr>
<tr><td>CLEFIA-128</td><td>1,309</td><td>78</td><td>39,357/152,023</td><td>RL78</td><td>[94]</td></tr>
<tr><td>CLEFIA-128</td><td>2,026</td><td>64</td><td>4,337/4,477</td><td>RL78</td><td>[94]</td></tr>
</table></td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-2 [66]、IETF RFC 6114 [74]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>LED</td></tr>
<tr><td>設計者</td><td>Jian Guo[1], Thomas Peyrin[2], Axel Poschmann[2], Matt Robshaw[3]
(1: Institute for Infocomm Research/Singapore, 2: Nanyang Technological University/ Singapore, 3: Orange Labs/France)</td></tr>
<tr><td>発表年</td><td>2011（CHES 2011 [54]）</td></tr>
<tr><td>仕様参照先</td><td>CHES 2011 [54]</td></tr>
<tr><td>特徴</td><td>設計者らは、鍵スケジュールがなく、関連鍵攻撃耐性を持ち、ハードウェア実装での軽量性に特化しながらも十分なソフトウェア実装性能を持つと主張している。軽量ハッシュ関数 PHOTON と同様、serialized MDS を内部構造として採用している。
<table>
<tr><td>全体構造</td><td colspan="2">SPN 型</td></tr>
<tr><td>ブロック長 [bit]</td><td colspan="2">64</td></tr>
<tr><td>鍵長 [bit]</td><td>64 (LED-64)</td><td>128 (LED-128)</td></tr>
<tr><td>構成段数 [段]</td><td>32 (8 ステップ)</td><td>48 (12 ステップ)</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [44, 45, 65, 95, 98, 110, 115] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2013 年に提案された Dinur ら [44] による Even-Mansour 暗号への汎用的な攻撃であり、3 ステップに簡略化した LED-64 と 8 ステップに簡略化した LED-128 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。関連鍵設定における最良の攻撃は、2012 年に提案された Mendel ら [95] による差分攻撃であり、4 ステップに簡略化した LED-64 に対して、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>LED-64 (Enc)</td><td>966</td><td>1,248</td><td>5.1</td><td>[54]</td></tr>
<tr><td>LED-64 (Enc)</td><td>2,695</td><td>32</td><td>200.0</td><td>[1]</td></tr>
<tr><td>LED-128 (Enc)</td><td>1,265</td><td>1,872</td><td>3.4</td><td>[54]</td></tr>
<tr><td>LED-128 (Enc)</td><td>3,036</td><td>48</td><td>133.3</td><td>[1]</td></tr>
</table>
ソフトウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Type</th><th>Cycles/byte</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>LED-64</td><td>Table/VPI/Bitslice</td><td>76.0/48.1/13.1</td><td>Core i3 2367M</td><td>[13]</td></tr>
<tr><td>LED-128</td><td>Table/VPI/Bitslice</td><td>113.3/54.6/17.6</td><td>Core i3 2367M</td><td>[13]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>Midori</td></tr>
<tr><td>設計者</td><td>Subhadeep Banik[1], Andrey Bogdanov[1], Takanori Isobe[2], Kyoji Shibutani[2], Harunaga Hiwatari[2], Toru Akishita[2], Francesco Regazzoni[3]
(1: Technical University of Denmark/Denmark, 2: Sony Corporation/Japan, 3: University of Lugano/Switzerland)</td></tr>
<tr><td>発表年</td><td>2015（ASIACRYPT 2015 [8]）</td></tr>
<tr><td>仕様参照先</td><td>ASIACRYPT 2015 [8]</td></tr>
<tr><td>特徴</td><td>設計者らは、ハードウェア実装における小型実装性能、低レイテンシ性能に加え、低エネルギー消費性能に優れたアルゴリズムであると主張している。
<table>
<tr><td>全体構造</td><td colspan="2">SPN 型</td></tr>
<tr><td>ブロック長 [bit]</td><td>64 (Midori64)</td><td>128 (Midori128)</td></tr>
<tr><td>鍵長 [bit]</td><td colspan="2">128</td></tr>
<tr><td>構成段数 [段]</td><td>16</td><td>20</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [5, 12, 15, 16, 33, 49, 52, 53, 56, 83, 84, 114, 115, 120, 121, 123, 132, 133] が発表されているが、弱鍵設定と関連鍵設定を除き、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2015 年に提案された Liu ら [83, 84] による Midori64 への中間一致攻撃と、2016 年に提案された Tolba ら [123] による Midori128 への切り詰め差分攻撃であり、12 段に簡略化した Midori64 と 13 段に簡略化した Midori128 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。Midori64 には弱鍵が存在し、弱鍵設定では仕様段数であっても効率的な鍵回復攻撃 [52, 53] とメッセージ復元攻撃 [120, 121] が可能となる。関連鍵設定における最良の攻撃は、2016 年に提案された Gérault ら [49] による差分攻撃であり、Midori64 と Midori128 に対して、それぞれ仕様段数において秘密鍵の全数探索より効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>Midori64 (Enc)</td><td>1,542</td><td>16</td><td>400.0</td><td>[8]</td></tr>
<tr><td>Midori64 (Enc/Dec)</td><td>2,450</td><td>16</td><td>400.0</td><td>[8]</td></tr>
<tr><td>Midori128 (Enc)</td><td>2,522</td><td>20</td><td>640.0</td><td>[8]</td></tr>
<tr><td>Midori128 (Enc/Dec)</td><td>3,661</td><td>20</td><td>640.0</td><td>[8]</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>Piccolo</td></tr>
<tr><td>設計者</td><td>Kyoji Shibutani, Takanori Isobe, Harunaga Hiwatari, Atsushi Mitsuda, Toru Akishita, Taizo Shirai (Sony Corporation/Japan)</td></tr>
<tr><td>発表年</td><td>2011（CHES 2011 [108]）</td></tr>
<tr><td>仕様参照先</td><td>CHES 2011 [108]</td></tr>
<tr><td>特徴</td><td>設計者らは、従来の攻撃に加え、関連鍵攻撃、中間一致攻撃に対して十分な安全性を持ち、特にハードウェア実装での性能が高く、構造上、復号関数を実装したとしても大きなオーバーヘッドはなく、軽量性のみならずエネルギー効率も高いと主張している。
<table>
<tr><td>全体構造</td><td colspan="2">4-line 変形一般化 Feistel 型</td></tr>
<tr><td>ブロック長 [bit]</td><td colspan="2">64</td></tr>
<tr><td>鍵長 [bit]</td><td>80 (Piccolo-80)</td><td>128 (Piccolo-128)</td></tr>
<tr><td>構成段数 [段]</td><td>25</td><td>31</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [6, 55, 65, 86, 96, 108, 111, 122] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2012 年に提案された Isobe ら [65] による中間一致攻撃と 2018 年に提案された Liu ら [86] による中間一致攻撃である。また、関連鍵設定における最良の攻撃は、2013 年に提案された Minier [96] による不能差分攻撃である。これらの攻撃により、14 段に簡略化した Piccolo-80 と 21 段に簡略化した Piccolo-128 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>Piccolo-80 (Enc)</td><td>1,048</td><td>432</td><td>14.8</td><td>[108]</td></tr>
<tr><td>Piccolo-80 (Enc)</td><td>1,499</td><td>27</td><td>237.0</td><td>[108]</td></tr>
<tr><td>Piccolo-80 (Enc/Dec)</td><td>1,109</td><td>432</td><td>14.8</td><td>[108]</td></tr>
<tr><td>Piccolo-128 (Enc)</td><td>1,338</td><td>528</td><td>12.1</td><td>[108]</td></tr>
<tr><td>Piccolo-128 (Enc)</td><td>1,776</td><td>33</td><td>193.9</td><td>[108]</td></tr>
<tr><td>Piccolo-128 (Enc/Dec)</td><td>1,397</td><td>528</td><td>12.1</td><td>[108]</td></tr>
</table>
ソフトウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Type</th><th>Cycles/byte</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>Piccolo-80</td><td>Bitslice</td><td>4.57</td><td>Core i7 870</td><td>[93]</td></tr>
<tr><td>Piccolo-128</td><td>Bitslice</td><td>5.52</td><td>Core i7 870</td><td>[93]</td></tr>
<tr><td>Piccolo-80</td><td>Table/VPI/Bitslice</td><td>89.3/33.3/9.2</td><td>Core i3 2367M</td><td>[13]</td></tr>
<tr><td>Piccolo-128</td><td>Table/VPI/Bitslice</td><td>103.6/41.6/10.9</td><td>Core i3 2367M</td><td>[13]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>PRESENT</td></tr>
<tr><td>設計者</td><td>Andrey Bogdanov[1], Lars R. Knudsen[2], Gregor Leander[1], Christof Paar[1], Axel Poschmann[1], Matthew J. B. Robshaw[3], Yannick Seurin[3], C. Vikkelsoe[2]
(1: Ruhr-University Bochum/Germany, 2: Technical University Denmark/Denmark, 3: France Telecom/France)</td></tr>
<tr><td>発表年</td><td>2007（CHES 2007 [24]）</td></tr>
<tr><td>仕様参照先</td><td>CHES 2007 [24]</td></tr>
<tr><td>特徴</td><td>軽量ブロック暗号の草分け的アルゴリズムであり、特にハードウェアの小型実装において高い実装性能を持つ。
<table>
<tr><td>全体構造</td><td colspan="2">SPN 型</td></tr>
<tr><td>ブロック長 [bit]</td><td colspan="2">64</td></tr>
<tr><td>鍵長 [bit]</td><td>80 (PRESENT-80)</td><td>128 (PRESENT-128)</td></tr>
<tr><td>構成段数 [段]</td><td colspan="2">31</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [2, 19, 20, 21, 25, 34, 47, 71, 72, 134] が発表されているが、既知鍵設定を除き、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2020 年に提案された Flórez-Gutiérrez ら [47] による多次元線形攻撃であり、28 段に簡略化した PRESENT-80 と PRESENT-128 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。既知鍵設定における最良の攻撃は、2015 年に提案された Blondeau ら [21] による切り詰め差分攻撃であり、PRESENT-80 と PRESENT-128 に対して、それぞれ仕様段数において効率的に識別攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>PRESENT-80 (Enc)</td><td>1,000</td><td>563</td><td>11.4</td><td>[107]</td></tr>
<tr><td>PRESENT-80 (Enc)</td><td>1,570</td><td>32</td><td>200.0</td><td>[24]</td></tr>
<tr><td>PRESENT-128 (Enc)</td><td>1,391</td><td>559</td><td>11.4</td><td>[101]</td></tr>
<tr><td>PRESENT-128 (Enc)</td><td>1,886</td><td>32</td><td>200.0</td><td>[24]</td></tr>
</table>
ソフトウェア実装評価結果 (ハイエンド CPU)
<table>
<tr><th>Algorithm</th><th>Type</th><th>Cycles/byte</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>PRESENT-80/128</td><td>Bitslice</td><td>5.79</td><td>Core i7 870</td><td>[93]</td></tr>
<tr><td>PRESENT-80</td><td>Table/VPI/Bitslice</td><td>72.6/35.0/17.4</td><td>Core i3 2367M</td><td>[13]</td></tr>
<tr><td>PRESENT-128</td><td>Table/VPI/Bitslice</td><td>72.5/35.0/18.9</td><td>Core i3 2367M</td><td>[13]</td></tr>
</table>
ソフトウェア実装評価結果 (ローエンド CPU)
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>PRESENT-80</td><td>512</td><td>62</td><td>61,634/60,834</td><td>RL78</td><td>[94]</td></tr>
<tr><td>PRESENT-80</td><td>1,855</td><td>48</td><td>9,007/8,920</td><td>RL78</td><td>[94]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42, 105] で報告されている。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-2 [66]、ISO/IEC 29167-11 [67]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>PRINCE</td></tr>
<tr><td>設計者</td><td>Julia Borghoff[1], Anne Canteaut[1,2], Tim Guneysu[3], Elif Bilge Kavun[3], Miroslav Knezevic[4], Lars R. Knudsen[1], Gregor Leander[1], Ventzislav Nikov[4], Christof Paar[3], Christian Rechberger[1], Peter Rombouts[4], Soren S. Thomsen[1], Tolga Yalcin[3]
(1: Technical University of Denmark/Denmark, 2: INRIA/France, 3: Ruhr-University Bochum/Germany, 4: NXP Semiconductors/Belgium)</td></tr>
<tr><td>発表年</td><td>2012（ASIACRYPT 2012 [26]）</td></tr>
<tr><td>仕様参照先</td><td>ASIACRYPT 2012 [26]</td></tr>
<tr><td>特徴</td><td>設計者らは、ハードウェア実装における小型実装性能に加え、低レイテンシ性能にも優れたアルゴリズムであると主張している。
α-reflection と呼ばれる対称性を持つことにより、通常のブロック暗号とは異なり、128 ビット鍵を利用していても攻撃者が 2^n の平文暗号文ペアを使える場合、$(127-n)$ ビットの安全性しか主張できていない。
<table>
<tr><td>全体構造</td><td>SPN 型</td></tr>
<tr><td>ブロック長 [bit]</td><td>64</td></tr>
<tr><td>鍵長 [bit]</td><td>128</td></tr>
<tr><td>構成段数 [段]</td><td>12</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [5, 28, 29, 40, 41, 47, 51, 70, 80, 104] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2014 年に提案された Canteaut ら [28] による多重差分攻撃であり、10 段に簡略化した PRINCE に対して、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>PRINCE (Enc/Dec)</td><td>2,953</td><td>12</td><td>533.3</td><td>[10]</td></tr>
<tr><td>PRINCE (Enc/Dec)</td><td>8,577</td><td>1</td><td>6,400.0</td><td>[10]</td></tr>
</table>
ソフトウェア実装評価結果 (ローエンド CPU)
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>PRINCE</td><td>2,382</td><td>220</td><td>225.4</td><td>ATtiny85</td><td>[100]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>SIMON</td></tr>
<tr><td>設計者</td><td>Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, Louis Wingers（National Security Agency/USA）</td></tr>
<tr><td>発表年</td><td>2013（Cryptology ePrint Archive [11]）</td></tr>
<tr><td>仕様参照先</td><td>Cryptology ePrint Archive [11]</td></tr>
<tr><td>特徴</td><td>設計者らは、様々なブロック長、鍵長に対応したアルゴリズムであり、軽量性において、ハードウェア、ソフトウェア両方で高い実装性能を持つが、特にハードウェアでの実装性能に優れると主張している。
ブロック長 $2n$ ビット、鍵長 m ワードの SIMON を SIMON$2n/mn$ と表記する。例えば、SIMON64/128 はブロック長 64 ビット、鍵長 128 ビットの SIMON を表す。
<table>
<tr><td>全体構造</td><td colspan="10">Feistel 型</td></tr>
<tr><td>ブロック長 [bit]</td><td>32</td><td colspan="2">48</td><td colspan="2">64</td><td colspan="2">96</td><td colspan="3">128</td></tr>
<tr><td>鍵長 [bit]</td><td>64</td><td>72</td><td>96</td><td>96</td><td>128</td><td>96</td><td>144</td><td>128</td><td>192</td><td>256</td></tr>
<tr><td>構成段数 [段]</td><td>32</td><td colspan="2">36</td><td>42</td><td>44</td><td>52</td><td>54</td><td>68</td><td>69</td><td>72</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [9, 30, 35, 39, 57, 59, 60, 61, 76, 78, 79, 90, 91, 103, 106, 127, 128] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2016 年に提案された Chen ら [30] による線形攻撃、2018 年に提案された Rohit ら [106] による correlated sequence attack、2021 年に提案された Leurent ら [79] による線形攻撃であり、27、25、31、45、56 段に簡略化したブロック長 32、48、64、96、128 ビットの SIMON に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。既知鍵設定における最良の攻撃は、2017 年に提案された Hao ら [57] による切り詰め差分攻撃であり、29、32、37、47、63 段に簡略化したブロック長 32、48、64、96、128 ビットの SIMON に対して、それぞれ効率的に識別攻撃が実行できる。関連鍵設定における最良の攻撃は、2019 年に提案された Lee ら [78] による線形攻撃であり、23、28、34、62 段に簡略化したブロック長 32、48、64、128 ビットの SIMON に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>SIMON64/96</td><td>809</td><td>1,455</td><td>4.4</td><td>[11]</td></tr>
<tr><td>SIMON64/128</td><td>958</td><td>1,524</td><td>4.2</td><td>[11]</td></tr>
<tr><td>SIMON128/128</td><td>1,234</td><td>4,414</td><td>2.9</td><td>[11]</td></tr>
<tr><td>SIMON128/256</td><td>1,782</td><td>4,923</td><td>2.6</td><td>[11]</td></tr>
</table>
ソフトウェア実装評価結果 (ローエンド CPU)
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>SIMON64/96</td><td>274</td><td>0</td><td>239</td><td>ATtiny45</td><td>[11]</td></tr>
<tr><td>SIMON64/128</td><td>282</td><td>0</td><td>250</td><td>ATtiny45</td><td>[11]</td></tr>
<tr><td>SIMON128/128</td><td>732</td><td>0</td><td>376</td><td>ATtiny45</td><td>[11]</td></tr>
<tr><td>SIMON128/256</td><td>764</td><td>0</td><td>398</td><td>ATtiny45</td><td>[11]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29167-21 [68]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>SPECK</td></tr>
<tr><td>設計者</td><td>Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, Louis Wingers（National Security Agency/USA）</td></tr>
<tr><td>発表年</td><td>2013（Cryptology ePrint Archive [11]）</td></tr>
<tr><td>仕様参照先</td><td>Cryptology ePrint Archive [11]</td></tr>
<tr><td>特徴</td><td>設計者らは、様々なブロック長、鍵長に対応したアルゴリズムであり、軽量性において、ハードウェア、ソフトウェア両方で高い実装性能を持つが、特にソフトウェアでの実装性能に優れると主張している。
SIMON 同様、ブロック長 $2n$ ビット、鍵長 m ワードの SPECK を SPECK$2n/mn$ と表記する。例えば、SPECK64/128 はブロック長 64 ビット、鍵長 128 ビットの SPECK を表す。
<table>
<tr><td>全体構造</td><td colspan="10">変形 Feistel 型</td></tr>
<tr><td>ブロック長 [bit]</td><td>32</td><td colspan="2">48</td><td colspan="2">64</td><td colspan="2">96</td><td colspan="3">128</td></tr>
<tr><td>鍵長 [bit]</td><td>64</td><td>72</td><td>96</td><td>96</td><td>128</td><td>96</td><td>144</td><td>128</td><td>192</td><td>256</td></tr>
<tr><td>構成段数 [段]</td><td colspan="2">22</td><td>23</td><td>26</td><td>27</td><td>28</td><td>29</td><td>32</td><td>33</td><td>34</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [9, 14, 18, 31, 32, 43, 48, 50, 62, 63, 75, 87, 88, 89, 112, 116, 126] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2016 年に提案された Song ら [112] による差分攻撃であり、14、16、20、21、25 段に簡略化したブロック長 32、48、64、96、128 ビットの SPECK に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>SPECK64/96</td><td>860</td><td>1,778</td><td>3.6</td><td>[11]</td></tr>
<tr><td>SPECK64/128</td><td>996</td><td>1,882</td><td>3.4</td><td>[11]</td></tr>
<tr><td>SPECK128/128</td><td>1,280</td><td>4,267</td><td>3.0</td><td>[11]</td></tr>
<tr><td>SPECK128/256</td><td>1,840</td><td>4,571</td><td>2.8</td><td>[11]</td></tr>
</table>
ソフトウェア実装評価結果 (ローエンド CPU)
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>SPECK64/96</td><td>182</td><td>0</td><td>144</td><td>ATtiny45</td><td>[11]</td></tr>
<tr><td>SPECK64/128</td><td>186</td><td>0</td><td>150</td><td>ATtiny45</td><td>[11]</td></tr>
<tr><td>SPECK128/128</td><td>396</td><td>0</td><td>167</td><td>ATtiny45</td><td>[11]</td></tr>
<tr><td>SPECK128/256</td><td>412</td><td>0</td><td>177</td><td>ATtiny45</td><td>[11]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29167-22 [69]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>TWINE</td></tr>
<tr><td>設計者</td><td>Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, Eita Kobayashi（NEC Corporation/Japan）</td></tr>
<tr><td>発表年</td><td>2011（ECRYPT Workshop on Lightweight Cryptography、SAC 2012 [118]）</td></tr>
<tr><td>仕様参照先</td><td>SAC 2012 [118]</td></tr>
<tr><td>特徴</td><td>設計者らは、ハードウェアでの軽量性のみならず、ローエンド CPU からハイエンド CPU までの幅広いソフトウェアにおいても高い実装性能を持つと主張している。FSE 2010 [117] で設計者らにより提案された改良ブロックシャッフルを採用し、安全性を高めている。
<table>
<tr><td>全体構造</td><td colspan="2">16-line 変形一般化 Feistel 型</td></tr>
<tr><td>ブロック長 [bit]</td><td colspan="2">64</td></tr>
<tr><td>鍵長 [bit]</td><td>80 (TWINE-80)</td><td>128 (TWINE-128)</td></tr>
<tr><td>構成段数 [段]</td><td colspan="2">36</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [3, 17, 22, 36, 73, 85, 97, 99, 124, 129, 130, 135] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2016 年に提案された Lin ら [85] による多次元零相関線形攻撃であり、23 段に簡略化した TWINE-80 と 25 段に簡略化した TWINE-128 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>TWINE-80 (Enc)</td><td>1,503</td><td>36</td><td>177.8</td><td>[118]</td></tr>
<tr><td>TWINE-80 (Enc)</td><td>1,011</td><td>393</td><td>16.3</td><td>[118]</td></tr>
<tr><td>TWINE-80 (Enc/Dec)</td><td>1,799</td><td>36</td><td>177.8</td><td>[118]</td></tr>
<tr><td>TWINE-128 (Enc)</td><td>1,866</td><td>36</td><td>177.8</td><td>[118]</td></tr>
<tr><td>TWINE-128 (Enc/Dec)</td><td>2,285</td><td>36</td><td>177.8</td><td>[118]</td></tr>
</table>
ソフトウェア実装評価結果 (ハイエンド CPU)
<table>
<tr><th>Algorithm</th><th>Type</th><th>Cycles/byte</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>TWINE-80/128</td><td>Bitslice (Single/Double)</td><td>11.10/5.55</td><td>Core i7 2600S</td><td>[118]</td></tr>
</table>
ソフトウェア実装評価結果 (ローエンド CPU)
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>TWINE-80</td><td>2,294</td><td>386</td><td>163/163</td><td>ATmega163</td><td>[118]</td></tr>
<tr><td>TWINE-80</td><td>792</td><td>191</td><td>2,350/2,337</td><td>ATmega163</td><td>[118]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>LEA</td></tr>
<tr><td>設計者</td><td>Deukjo Hong[1], Jung-Keun Lee[1], Dong-Chan Kim[1], Daesung Kwon[1], Kwon Ho Ryu[1], Dong-Geon Lee[2]
(1: Attached Institute of ETRI/Korea, 2: Pusan National University/Korea)</td></tr>
<tr><td>発表年</td><td>2013（WISA 2013 [58]）</td></tr>
<tr><td>仕様参照先</td><td>WISA 2013 [58]</td></tr>
<tr><td>特徴</td><td>設計者らは、ソフトウェア実装における高速な暗号化処理が可能であり、オーバーヘッドの軽減による低消費電力性能を持つとともに、コードサイズの小さいコンパクトな実装が可能であると主張している。また、構成段数の設定においては未知の攻撃への対策として 1.5 倍のセキュリティマージンを設けることにより、ブロック暗号に対する全ての既存攻撃に対して十分な安全性を持つと主張している。
<table>
<tr><td>全体構造</td><td colspan="3">Addition-Rotation-XOR (ARX) 型</td></tr>
<tr><td>ブロック長 [bit]</td><td colspan="3">128</td></tr>
<tr><td>鍵長 [bit]</td><td>128 (LEA-128)</td><td>192 (LEA-192)</td><td>256 (LEA-256)</td></tr>
<tr><td>構成段数 [段]</td><td>24</td><td>28</td><td>32</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [7, 46, 75, 112, 113] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2016 年に提案された Song ら [112] による差分攻撃であり、14、14、15 段に簡略化した LEA-128、LEA-192、LEA-256 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。また、最良の識別攻撃は、2020 年に提案された Kim ら [75] によるブーメラン攻撃であり、16 段に簡略化した LEA-128 に対して、効率的に識別攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100KHz [kbps]</th><th>Ref.</th></tr>
<tr><td>LEA-128 (Enc)</td><td>3,826</td><td>168</td><td>76.19</td><td>[58]</td></tr>
<tr><td>LEA-128 (Enc)</td><td>5,426</td><td>24</td><td>533.33</td><td>[58]</td></tr>
</table>
ソフトウェア実装評価結果（ハイエンド CPU）
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>LEA-128</td><td>–</td><td>–</td><td>9.29/14.83</td><td>Intel Core 2 Quad Q6600</td><td>[58]</td></tr>
<tr><td>LEA-128</td><td>–</td><td>–</td><td>9.29/14.52</td><td>Intel Core i5-2500</td><td>[58]</td></tr>
<tr><td>LEA-128</td><td>–</td><td>–</td><td>8.85/14.50</td><td>AMD Phenom II X4 965</td><td>[58]</td></tr>
<tr><td>LEA-128</td><td>–</td><td>–</td><td>8.55/14.05</td><td>AMD Opteron 6176 SE</td><td>[58]</td></tr>
</table>
ソフトウェア実装評価結果（ローエンド CPU）
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>LEA-128</td><td>590</td><td>32</td><td>326.94/–</td><td>Arm926EJ-S</td><td>[58]</td></tr>
<tr><td>LEA-128</td><td>–</td><td>–</td><td>20.06/–</td><td>Arm926EJ-S</td><td>[58]</td></tr>
<tr><td>LEA-128</td><td>9,674</td><td>832</td><td>103.59/–</td><td>MCF5213</td><td>[58]</td></tr>
<tr><td>LEA-128</td><td>704</td><td>32</td><td>829.25/–</td><td>MCF5213</td><td>[58]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-2 [66]、KS X 3246 [64]</td></tr>
</table>

参考文献

[1] The LED block cipher (Dec 2013), available from https://sites.google.com/site/ledblockcipher/hardware（2023-10-04 閲覧不可）

[2] Abed, F., Forler, C., List, E., Lucks, S., Wenzel, J.: Biclique Cryptanalysis of the PRESENT and LED Lightweight Ciphers. IACR Cryptol. ePrint Arch. 2012, 591 (2012), https://eprint.iacr.org/2012/591

[3] Ahmadi, S., Ahmadian, Z., Mohajeri, J., Aref, M.R.: Biclique Cryptanalysis of Block Ciphers LBlock and TWINE-80 with Practical Data Complexity. ISC Int. J. Inf. Secur. 11(1), 57–74 (2019), https://doi.org/10.22042/isecure.2018.138036.420

[4] Akishita, T., Hiwatari, H.: Very Compact Hardware Implementations of the Blockcipher CLEFIA. In: Miri, A., Vaudenay, S. (eds.) Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7118, pp. 278–292. Springer (2011), https://dx.doi.org/10.1007/978-3-642-28496-0_17

[5] Ankele, R., Kölbl, S.: Mind the Gap - A Closer Look at the Security of Block Ciphers against Differential Cryptanalysis. In: Cid, C., Jr., M.J.J. (eds.) Selected Areas in Cryptography - SAC 2018 - 25th International Conference, Calgary, AB, Canada, August 15-17, 2018, Revised Selected Papers. Lecture Notes in Computer Science, vol. 11349, pp. 163–190. Springer (2018), https://doi.org/10.1007/978-3-030-10970-7_8

[6] Azimi, S.A., Ahmadian, Z., Mohajeri, J., Aref, M.R.: Impossible differential cryptanalysis of Piccolo lightweight block cipher. In: 11th International ISC Conference on Information Security and Cryptology, ISCISC 2014, Tehran, Iran, September 3-4, 2014. pp. 89–94. IEEE (2014), https://doi.org/10.1109/ISCISC.2014.6994028

[7] Bagherzadeh, E., Ahmadian, Z.: MILP-based automatic differential search for LEA and HIGHT block ciphers. IET Inf. Secur. 14(5), 595–603 (2020), https://doi.org/10.1049/iet-ifs.2018.5539

[8] Banik, S., Bogdanov, A., Isobe, T., Shibutani, K., Hiwatari, H., Akishita, T., Regazzoni, F.: Midori: A Block Cipher for Low Energy. In: Iwata, T., Cheon, J.H. (eds.) Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II. Lecture Notes in Computer Science, vol. 9453, pp. 411–436. Springer (2015), https://dx.doi.org/10.1007/978-3-662-48800-3_17

[9] Bao, Z., Guo, J., Liu, M., Ma, L., Tu, Y.: Conditional Differential-Neural Cryptanalysis. IACR Cryptol. ePrint Arch. 2021, 719 (2021), https://eprint.iacr.org/2021/719

[10] Batina, L., Das, A., Ege, B., Kavun, E.B., Mentens, N., Paar, C., Verbauwhede, I., Yalçin, T.: Dietary Recommendations for Lightweight Block Ciphers: Power, Energy and Area Analysis of Recently Developed Architectures. In: Hutter, M., Schmidt, J. (eds.) Radio Frequency Identification - Security and Privacy Issues 9th International Workshop, RFIDsec 2013, Graz, Austria, July 9-11, 2013, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8262, pp. 103–112. Springer (2013), https://dx.doi.org/10.1007/978-3-642-41332-2_7

[11] Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., Wingers, L.: The SIMON and SPECK Families of Lightweight Block Ciphers. IACR Cryptology ePrint Archive 2013, 404 (2013), https://eprint.iacr.org/2013/404

[12] Beierle, C., Canteaut, A., Leander, G.: Nonlinear Approximations in Cryptanalysis Revisited. IACR Trans. Symmetric Cryptol. 2018(4), 80–101 (2018), https://doi.org/10.13154/tosc.v2018.i4.80-101

[13] Benadjila, R., Guo, J., Lomné, V., Peyrin, T.: Implementing Lightweight Block Ciphers on x86 Architectures. In: Lange et al. [77], pp. 324–351, https://dx.doi.org/10.1007/978-3-662-43414-7_17

[14] Benamira, A., Gérault, D., Peyrin, T., Tan, Q.Q.: A Deeper Look at Machine Learning-Based Cryptanalysis. In: Canteaut, A., Standaert, F. (eds.) Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12696, pp. 805–835. Springer (2021), https://doi.org/10.1007/978-3-030-77870-5_28

[15] Beyne, T.: Block Cipher Invariants as Eigenvectors of Correlation Matrices. In: Peyrin, T., Galbraith, S.D. (eds.) Advances in Cryptology - ASIACRYPT 2018 - 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2-6, 2018, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11272, pp. 3–31. Springer (2018), https://doi.org/10.1007/978-3-030-03326-2_1

[16] Beyne, T.: Block Cipher Invariants as Eigenvectors of Correlation Matrices. J. Cryptol. 33(3), 1156–1183 (2020), https://doi.org/10.1007/s00145-020-09344-1

[17] Biryukov, A., Derbez, P., Perrin, L.: Differential Analysis and Meet-in-the-Middle Attack Against Round-Reduced TWINE. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9054, pp. 3–27. Springer (2015), https://doi.org/10.1007/978-3-662-48116-5_1

[18] Biryukov, A., Velichkov, V., Corre, Y.L.: Automatic Search for the Best Trails in ARX: Application to Block Cipher Speck. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 289–310. Springer (2016), https://doi.org/10.1007/978-3-662-52993-5_15

[19] Blondeau, C., Nyberg, K.: Links between Truncated Differential and Multidimensional Linear Properties of Block Ciphers and Underlying Attack Complexities. In: Nguyen, P.Q., Oswald, E. (eds.) Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings. Lecture Notes in Computer Science, vol. 8441, pp. 165–182. Springer (2014), https://dx.doi.org/10.1007/978-3-642-55220-5_10

[20] Blondeau, C., Nyberg, K.: Improved Parameter Estimates for Correlation and Capacity Deviates in Linear Cryptanalysis. IACR Trans. Symmetric Cryptol. 2016(2), 162–191 (2016), https://doi.org/10.13154/tosc.v2016.i2.162-191

[21] Blondeau, C., Peyrin, T., Wang, L.: Known-Key Distinguisher on Full PRESENT. In: Gennaro, R., Robshaw, M. (eds.) Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9215, pp. 455–474. Springer (2015), https://doi.org/10.1007/978-3-662-47989-6_22

[22] Bogdanov, A., Boura, C., Rijmen, V., Wang, M., Wen, L., Zhao, J.: Key Difference Invariant Bias in Block Ciphers. In: Sako, K., Sarkar, P. (eds.) Advances in Cryptology - ASIACRYPT 2013 - 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part I. Lecture Notes in Computer Science, vol. 8269, pp. 357–376. Springer (2013), https://doi.org/10.1007/978-3-642-42033-7_19

[23] Bogdanov, A., Geng, H., Wang, M., Wen, L., Collard, B.: Zero-Correlation Linear Cryptanalysis with FFT and Improved Attacks on ISO Standards Camellia and CLEFIA. In: Lange et al. [77], pp. 306–323, https://dx.doi.org/10.1007/978-3-662-43414-7_16

[24] Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: Paillier, P., Verbauwhede, I. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-

13, 2007, Proceedings. Lecture Notes in Computer Science, vol. 4727, pp. 450–466. Springer (2007), https://dx.doi.org/10.1007/978-3-540-74735-2_31

[25] Bogdanov, A., Tischhauser, E., Vejre, P.S.: Multivariate Profiling of Hulls for Linear Cryptanalysis. IACR Trans. Symmetric Cryptol. 2018(1), 101–125 (2018), https://doi.org/10.13154/tosc.v2018.i1.101-125

[26] Borghoff, J., Canteaut, A., Güneysu, T., Kavun, E.B., Knezevic, M., Knudsen, L.R., Leander, G., Nikov, V., Paar, C., Rechberger, C., Rombouts, P., Thomsen, S.S., Yalçin, T.: PRINCE - A Low-Latency Block Cipher for Pervasive Computing Applications - Extended Abstract. In: Wang, X., Sako, K. (eds.) Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7658, pp. 208–225. Springer (2012), https://dx.doi.org/10.1007/978-3-642-34961-4_14

[27] Boura, C., Naya-Plasencia, M., Suder, V.: Scrutinizing and Improving Impossible Differential Attacks: Applications to CLEFIA, Camellia, LBlock and Simon. In: Sarkar, P., Iwata, T. (eds.) Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I. Lecture Notes in Computer Science, vol. 8873, pp. 179–199. Springer (2014), https://doi.org/10.1007/978-3-662-45611-8_10

[28] Canteaut, A., Fuhr, T., Gilbert, H., Naya-Plasencia, M., Reinhard, J.: Multiple Differential Cryptanalysis of Round-Reduced PRINCE. In: Cid, C., Rechberger, C. (eds.) Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8540, pp. 591–610. Springer (2014), https://dx.doi.org/10.1007/978-3-662-46706-0_30

[29] Canteaut, A., Naya-Plasencia, M., Vayssière, B.: Sieve-in-the-Middle: Improved MITM Attacks. In: Canetti, R., Garay, J.A. (eds.) Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I. Lecture Notes in Computer Science, vol. 8042, pp. 222–240. Springer (2013), https://doi.org/10.1007/978-3-642-40041-4_13

[30] Chen, H., Wang, X.: Improved Linear Hull Attack on Round-Reduced Simon with Dynamic Key-Guessing Techniques. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 428–449. Springer (2016), https://dx.doi.org/10.1007/978-3-662-52993-5_22

[31] Chen, Y., Yu, H.: Bridging Machine Learning and Cryptanalysis via EDLCT. IACR Cryptol. ePrint Arch. 2021, 705 (2021), https://eprint.iacr.org/2021/705

[32] Chen, Y., Yu, H.: Improved Neural Aided Statistical Attack for Cryptanalysis. IACR Cryptol. ePrint Arch. 2021, 311 (2021), https://eprint.iacr.org/2021/311

[33] Chen, Z., Chen, H., Wang, X.: Cryptanalysis of Midori128 Using Impossible Differential Techniques. In: Bao, F., Chen, L., Deng, R.H., Wang, G. (eds.) Information Security Practice and Experience - 12th International Conference, ISPEC 2016, Zhangjiajie, China, November 16-18, 2016, Proceedings. Lecture Notes in Computer Science, vol. 10060, pp. 1–12 (2016), https://doi.org/10.1007/978-3-319-49151-6_1

[34] Cho, J.Y.: Linear Cryptanalysis of Reduced-Round PRESENT. In: Pieprzyk, J. (ed.) Topics in Cryptology - CT-RSA 2010, The Cryptographers' Track at the RSA Conference 2010, San Francisco, CA, USA, March 1-5, 2010. Proceedings. Lecture Notes in Computer Science, vol. 5985, pp. 302–317. Springer (2010), https://dx.doi.org/10.1007/978-3-642-11925-5_21

[35] Chu, Z., Chen, H., Wang, X., Dong, X., Li, L.: Improved Integral Attacks on SIMON32 and SIMON48 with Dynamic Key-Guessing Techniques. Secur. Commun. Networks 2018, 5160237:1–5160237:11 (2018), https://doi.org/10.1155/2018/5160237

[36] Çoban, M., Karakoç, F., Boztas, Ö.: Biclique Cryptanalysis of TWINE. In: Pieprzyk, J., Sadeghi, A., Manulis, M. (eds.) Cryptology and Network Security, 11th International Conference, CANS 2012, Darmstadt, Germany, December 12-14, 2012. Proceedings. vol. 7712, pp. 43–55. Springer (2012), https://doi.org/10.1007/

978-3-642-35404-5_5

[37] Corporation, S.: CLEFIA: The 128-bit Blockcipher, https://www.sony.net/Products/cryptography/clefia/（2023-10-04 閲覧）

[38] CRYPTREC 軽量暗号ワーキンググループ: CRYPTREC 暗号技術ガイドライン（軽量暗号）（文書番号：CRYPTREC GL-2003-2016JP）(2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf

[39] Derbez, P., Fouque, P.: Automatic Search of Meet-in-the-Middle and Impossible Differential Attacks. In: Robshaw, M., Katz, J. (eds.) Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II. Lecture Notes in Computer Science, vol. 9815, pp. 157–184. Springer (2016), https://doi.org/10.1007/978-3-662-53008-5_6

[40] Derbez, P., Perrin, L.: Meet-in-the-Middle Attacks and Structural Analysis of Round-Reduced PRINCE. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9054, pp. 190–216. Springer (2015), https://doi.org/10.1007/978-3-662-48116-5_10

[41] Derbez, P., Perrin, L.: Meet-in-the-Middle Attacks and Structural Analysis of Round-Reduced PRINCE. J. Cryptol. 33(3), 1184–1215 (2020), https://doi.org/10.1007/s00145-020-09345-0

[42] Dinu, D., Corre, Y.L., Khovratovich, D., Perrin, L., Großschädl, J., Biryukov, A.: Triathlon of lightweight block ciphers for the Internet of things. J. Cryptogr. Eng. 9(3), 283–302 (2019), https://doi.org/10.1007/s13389-018-0193-x

[43] Dinur, I.: Improved Differential Cryptanalysis of Round-Reduced Speck. In: Joux, A., Youssef, A.M. (eds.) Selected Areas in Cryptography - SAC 2014 - 21st International Conference, Montreal, QC, Canada, August 14-15, 2014, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8781, pp. 147–164. Springer (2014), https://dx.doi.org/10.1007/978-3-319-13051-4_9

[44] Dinur, I., Dunkelman, O., Keller, N., Shamir, A.: Key Recovery Attacks on 3-round Even-Mansour, 8-step LED-128, and Full AES^2. In: Sako, K., Sarkar, P. (eds.) Advances in Cryptology - ASIACRYPT 2013 - 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part I. Lecture Notes in Computer Science, vol. 8269, pp. 337–356. Springer (2013), https://dx.doi.org/10.1007/978-3-642-42033-7_18

[45] Dinur, I., Dunkelman, O., Keller, N., Shamir, A.: Improved Linear Sieving Techniques with Applications to Step-Reduced LED-64. In: Cid, C., Rechberger, C. (eds.) Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8540, pp. 390–410. Springer (2014), https://doi.org/10.1007/978-3-662-46706-0_20

[46] Dwivedi, A.D., Srivastava, G.: Differential Cryptanalysis of Round-Reduced LEA. IEEE Access 6, 79105–79113 (2018), https://doi.org/10.1109/ACCESS.2018.2881130

[47] Flórez-Gutiérrez, A., Naya-Plasencia, M.: Improving Key-Recovery in Linear Attacks: Application to 28-Round PRESENT. In: Canteaut, A., Ishai, Y. (eds.) Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12105, pp. 221–249. Springer (2020), https://doi.org/10.1007/978-3-030-45721-1_9

[48] Fu, K., Wang, M., Guo, Y., Sun, S., Hu, L.: MILP-Based Automatic Search Algorithms for Differential and Linear Trails for Speck. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 268–288. Springer (2016), https://doi.org/10.1007/978-3-662-52993-5_14

[49] Gérault, D., Lafourcade, P.: Related-Key Cryptanalysis of Midori. In: Dunkelman, O., Sanadhya, S.K. (eds.) Progress in Cryptology - INDOCRYPT 2016 - 17th International Conference on Cryptology in India, Kolkata, India, December 11-14, 2016, Proceedings. Lecture Notes in Computer Science, vol. 10095, pp. 287–304 (2016),

https://doi.org/10.1007/978-3-319-49890-4_16

[50] Gohr, A.: Improving Attacks on Round-Reduced Speck32/64 Using Deep Learning. In: Boldyreva, A., Micciancio, D. (eds.) Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part II. Lecture Notes in Computer Science, vol. 11693, pp. 150–179. Springer (2019), https://doi.org/10.1007/978-3-030-26951-7_6

[51] Grassi, L., Rechberger, C.: Practical Low Data-Complexity Subspace-Trail Cryptanalysis of Round-Reduced PRINCE. In: Dunkelman, O., Sanadhya, S.K. (eds.) Progress in Cryptology - INDOCRYPT 2016 - 17th International Conference on Cryptology in India, Kolkata, India, December 11-14, 2016, Proceedings. Lecture Notes in Computer Science, vol. 10095, pp. 322–342 (2016), https://doi.org/10.1007/978-3-319-49890-4_18

[52] Guo, J., Jean, J., Nikolic, I., Qiao, K., Sasaki, Y., Sim, S.M.: Invariant Subspace Attack Against Midori64 and The Resistance Criteria for S-box Designs. IACR Trans. Symmetric Cryptol. 2016(1), 33–56 (2016), https://doi.org/10.13154/tosc.v2016.i1.33-56

[53] Guo, J., Jean, J., Nikolic, I., Sasaki, Y., Sim, S.M.: Invariant Subspace Attack Against Midori64 and The Resistance Criteria for S-box Designs. IACR Cryptol. ePrint Arch. 2016, 973 (2016), https://eprint.iacr.org/2016/973

[54] Guo, J., Peyrin, T., Poschmann, A., Robshaw, M.J.B.: The LED Block Cipher. In: Preneel and Takagi [102], pp. 326–341, https://dx.doi.org/10.1007/978-3-642-23951-9_22

[55] Han, G., Zhang, W.: Improved Biclique Cryptanalysis of the Lightweight Block Cipher Piccolo. Secur. Commun. Networks 2017, 7589306:1–7589306:12 (2017), https://doi.org/10.1155/2017/7589306

[56] Han, G., Zhang, W., Xing, Z., Zhao, H., Lian, J.: Unbalanced biclique cryptanalysis of a full round Midori. IET Commun. 13(5), 505–511 (2019), https://doi.org/10.1049/iet-com.2018.5343

[57] Hao, Y., Meier, W.: Truncated differential based known-key attacks on round-reduced SIMON. Des. Codes Cryptogr. 83(2), 467–492 (2017), https://doi.org/10.1007/s10623-016-0242-3

[58] Hong, D., Lee, J., Kim, D., Kwon, D., Ryu, K.H., Lee, D.: LEA: A 128-Bit Block Cipher for Fast Encryption on Common Processors. In: Kim, Y., Lee, H., Perrig, A. (eds.) Information Security Applications - 14th International Workshop, WISA 2013, Jeju Island, Korea, August 19-21, 2013, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8267, pp. 3–27. Springer (2013), https://doi.org/10.1007/978-3-319-05149-9_1

[59] Hou, Z., Ren, J., Chen, S.: Cryptanalysis of Round-Reduced SIMON32 Based on Deep Learning. IACR Cryptol. ePrint Arch. 2021, 362 (2021), https://eprint.iacr.org/2021/362

[60] Hou, Z., Ren, J., Chen, S.: Improve Neural Distinguisher for Cryptanalysis. IACR Cryptol. ePrint Arch. 2021, 1017 (2021), https://eprint.iacr.org/2021/1017

[61] Hou, Z., Ren, J., Chen, S.: SAT-based Method to Improve Neural Distinguisher and Applications to SIMON. IACR Cryptol. ePrint Arch. 2021, 452 (2021), https://eprint.iacr.org/2021/452

[62] Huang, M., Wang, L.: Automatic Tool for Searching for Differential Characteristics in ARX Ciphers and Applications. In: Hao, F., Ruj, S., Gupta, S.S. (eds.) Progress in Cryptology - INDOCRYPT 2019 - 20th International Conference on Cryptology in India, Hyderabad, India, December 15-18, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11898, pp. 115–138. Springer (2019), https://doi.org/10.1007/978-3-030-35423-7_6

[63] Huang, M., Wang, L.: Automatic Search for the Linear (Hull) Characteristics of ARX Ciphers: Applied to SPECK, SPARX, Chaskey, and CHAM-64. Secur. Commun. Networks 2020, 4898612:1–4898612:14 (2020), https://doi.org/10.1155/2020/4898612

[64] Internet, K., (KISA), S.A.: 128-bit block cipher LEA (2016), kS X 3246, https://www.rra.go.kr/ko/reference/kcsList_view.do?nb_seq=1923&cpage=4&nb_type=6&searchCon=&searchTxt=&sortOrder= (in Korean)

[65] Isobe, T., Shibutani, K.: Security Analysis of the Lightweight Block Ciphers XTEA, LED and Piccolo. In: Susilo, W., Mu, Y., Seberry, J. (eds.) Information Security and Privacy - 17th Australasian Conference, ACISP 2012,

Wollongong, NSW, Australia, July 9-11, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7372, pp. 71–86. Springer (2012), https://doi.org/10.1007/978-3-642-31448-3_6

[66] ISO/IEC: Information security – Lightweight cryptography – Part 2: Block ciphers (ISO/IEC 29192-2:2019), https://www.iso.org/standard/78477.html

[67] ISO/IEC: Information technology — Automatic identification and data capture techniques – Part 11: Crypto suite PRESENT-80 security services for air interface communications (ISO/IEC 29167-11: 2023), https://www.iso.org/standard/81489.html

[68] ISO/IEC: Information technology — Automatic identification and data capture techniques – Part 21: Crypto suite SIMON security services for air interface communications (ISO/IEC 29167-21: 2018), https://www.iso.org/standard/70388.html

[69] ISO/IEC: Information technology — Automatic identification and data capture techniques – Part 22: Crypto suite SPECK security services for air interface communications (ISO/IEC 29167-22: 2018), https://www.iso.org/standard/70389.html

[70] Jean, J., Nikolic, I., Peyrin, T., Wang, L., Wu, S.: Security Analysis of PRINCE. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8424, pp. 92–111. Springer (2013), https://doi.org/10.1007/978-3-662-43933-3_6

[71] Jeong, K., Kang, H., Lee, C., Sung, J., Hong, S.: Biclique Cryptanalysis of Lightweight Block Ciphers PRESENT, Piccolo and LED. IACR Cryptol. ePrint Arch. 2012, 621 (2012), https://eprint.iacr.org/2012/621

[72] Jithendra, K.B., Kassim, S.T.: New Biclique Cryptanalysis on Full-Round PRESENT-80 Block Cipher. SN Comput. Sci. 1(2), 94 (2020), https://doi.org/10.1007/s42979-020-0103-z

[73] Karakoç, F., Demirci, H., Harmanci, A.E.: Biclique cryptanalysis of LBlock and TWINE. Inf. Process. Lett. 113(12), 423–429 (2013), https://doi.org/10.1016/j.ipl.2013.03.011

[74] Katagi, M.: The 128-Bit Blockcipher CLEFIA. RFC 6114 (Mar 2011), https://www.rfc-editor.org/info/rfc6114

[75] Kim, D., Kwon, D., Song, J.: Efficient Computation of Boomerang Connection Probability for ARX-Based Block Ciphers with Application to SPECK and LEA. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. 103-A(4), 677–685 (2020), https://doi.org/10.1587/transfun.2019EAP1083

[76] Koo, B., Jung, Y., Kim, W.: Rotational-XOR Rectangle Cryptanalysis on Round-Reduced Simon. Secur. Commun. Networks 2020, 5968584:1–5968584:12 (2020), https://doi.org/10.1155/2020/5968584

[77] Lange, T., Lauter, K.E., Lisonek, P. (eds.): Selected Areas in Cryptography - SAC 2013 - 20th International Conference, Burnaby, BC, Canada, August 14-16, 2013, Revised Selected Papers, Lecture Notes in Computer Science, vol. 8282. Springer (2014), https://dx.doi.org/10.1007/978-3-662-43414-7

[78] Lee, J., Koo, B., Kim, W.: A General Framework for the Related-Key Linear Attack Against Block Ciphers with Linear Key Schedules. In: Paterson, K.G., Stebila, D. (eds.) Selected Areas in Cryptography - SAC 2019 - 26th International Conference, Waterloo, ON, Canada, August 12-16, 2019, Revised Selected Papers. Lecture Notes in Computer Science, vol. 11959, pp. 194–224. Springer (2019), https://doi.org/10.1007/978-3-030-38471-5_9

[79] Leurent, G., Pernot, C., Schrottenloher, A.: Clustering Effect in Simon and Simeck. In: Tibouchi, M., Wang, H. (eds.) Advances in Cryptology - ASIACRYPT 2021 - 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6-10, 2021, Proceedings, Part I. Lecture Notes in Computer Science, vol. 13090, pp. 272–302. Springer (2021), https://doi.org/10.1007/978-3-030-92062-3_10

[80] Li, L., Jia, K., Wang, X.: Improved Meet-in-the-Middle Attacks on AES-192 and PRINCE. IACR Cryptol. ePrint Arch. 2013, 573 (2013), https://eprint.iacr.org/2013/573

[81] Li, L., Jia, K., Wang, X., Dong, X.: Meet-in-the-Middle Technique for Truncated Differential and Its Applications

to CLEFIA and Camellia. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9054, pp. 48–70. Springer (2015), https://doi.org/10.1007/978-3-662-48116-5_3

[82] Li, Y., Wu, W., Zhang, L.: Improved Integral Attacks on Reduced-Round CLEFIA Block Cipher. In: Jung, S., Yung, M. (eds.) Information Security Applications - 12th International Workshop, WISA 2011, Jeju Island, Korea, August 22-24, 2011. Revised Selected Papers. Lecture Notes in Computer Science, vol. 7115, pp. 28–39. Springer (2011), https://doi.org/10.1007/978-3-642-27890-7_3

[83] Lin, L., Wu, W.: Meet-in-the-Middle Attacks on Reduced-Round Midori-64. IACR Cryptology ePrint Archive 2015, 1165 (2015), https://eprint.iacr.org/2015/1165

[84] Lin, L., Wu, W.: Meet-in-the-Middle Attacks on Reduced-Round Midori64. IACR Trans. Symmetric Cryptol. 2017(1), 215–239 (2017), https://doi.org/10.13154/tosc.v2017.i1.215-239

[85] Lin, L., Wu, W., Zheng, Y.: Automatic Search for Key-Bridging Technique: Applications to LBlock and TWINE. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 247–267. Springer (2016), https://doi.org/10.1007/978-3-662-52993-5_13

[86] Liu, Y., Cheng, L., Liu, Z., Li, W., Wang, Q., Gu, D.: Improved meet-in-the-middle attacks on reduced-round Piccolo. Sci. China Inf. Sci. 61(3), 032108:1–032108:13 (2018), https://doi.org/10.1007/s11432-016-9157-y

[87] Liu, Y., Wang, Q., Rijmen, V.: Automatic Search of Linear Trails in ARX with Applications to SPECK and Chaskey. In: Manulis, M., Sadeghi, A., Schneider, S.A. (eds.) Applied Cryptography and Network Security - 14th International Conference, ACNS 2016, Guildford, UK, June 19-22, 2016. Proceedings. Lecture Notes in Computer Science, vol. 9696, pp. 485–499. Springer (2016), https://doi.org/10.1007/978-3-319-39555-5_26

[88] Liu, Y., Witte, G.D., Ranea, A., Ashur, T.: Rotational-XOR Cryptanalysis of Reduced-round SPECK. IACR Trans. Symmetric Cryptol. 2017(3), 24–36 (2017), https://doi.org/10.13154/tosc.v2017.i3.24-36

[89] Liu, Z., Li, Y., Jiao, L., Wang, M.: A New Method for Searching Optimal Differential and Linear Trails in ARX Ciphers. IEEE Trans. Inf. Theory 67(2), 1054–1068 (2021), https://doi.org/10.1109/TIT.2020.3040543

[90] Liu, Z., Li, Y., Wang, M.: Optimal Differential Trails in SIMON-like Ciphers. IACR Trans. Symmetric Cryptol. 2017(1), 358–379 (2017), https://doi.org/10.13154/tosc.v2017.i1.358-379

[91] Lu, J., Liu, Y., Ashur, T., Sun, B., Li, C.: Rotational-XOR Cryptanalysis of Simon-Like Block Ciphers. In: Liu, J.K., Cui, H. (eds.) Information Security and Privacy - 25th Australasian Conference, ACISP 2020, Perth, WA, Australia, November 30 - December 2, 2020, Proceedings. Lecture Notes in Computer Science, vol. 12248, pp. 105–124. Springer (2020), https://doi.org/10.1007/978-3-030-55304-3_6

[92] Mala, H., Dakhilalian, M., Shakiba, M.: Impossible differential attacks on 13-round CLEFIA-128. J. Comput. Sci. Technol. 26(4), 744–750 (2011), https://dx.doi.org/10.1007/s11390-011-1173-0

[93] Matsuda, S., Moriai, S.: Lightweight Cryptography for the Cloud: Exploit the Power of Bitslice Implementation. In: Prouff, E., Schaumont, P. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2012 - 14th International Workshop, Leuven, Belgium, September 9-12, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7428, pp. 408–425. Springer (2012), https://dx.doi.org/10.1007/978-3-642-33027-8_24

[94] Matsui, M., Murakami, Y.: Minimalism of Software Implementation - Extensive Performance Analysis of Symmetric Primitives on the RL78 Microcontroller. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8424, pp. 393–409. Springer (2013), https://dx.doi.org/10.1007/978-3-662-43933-3_20

[95] Mendel, F., Rijmen, V., Toz, D., Varici, K.: Differential Analysis of the LED Block Cipher. In: Wang, X., Sako, K. (eds.) Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7658, pp. 190–207. Springer (2012), https://doi.org/10.1007/978-3-642-34961-4_

13

[96] Minier, M.: On the Security of Piccolo Lightweight Block Cipher against Related-Key Impossible Differentials. In: Paul, G., Vaudenay, S. (eds.) Progress in Cryptology - INDOCRYPT 2013 - 14th International Conference on Cryptology in India, Mumbai, India, December 7-10, 2013. Proceedings. Lecture Notes in Computer Science, vol. 8250, pp. 308–318. Springer (2013), https://dx.doi.org/10.1007/978-3-319-03515-4_21

[97] Najarkolaei, S.R.H., Ahangarkolaei, M.Z., Ahmadi, S., Aref, M.R.: Biclique cryptanalysis of TWINE-128. In: 13th International Iranian Society of Cryptology Conference on Information Security and Cryptology, ISCISC 2016, Tehran, Iran, September 7-8, 2016. pp. 46–51. IEEE (2016), https://doi.org/10.1109/ISCISC.2016.7736450

[98] Nikolic, I., Wang, L., Wu, S.: Cryptanalysis of Round-Reduced LED. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8424, pp. 112–129. Springer (2013), https://doi.org/10.1007/978-3-662-43933-3_7

[99] Niu, C., Li, M., Sun, S., Wang, M.: Zero-Correlation Linear Cryptanalysis with Equal Treatment for Plaintexts and Tweakeys. In: Paterson, K.G. (ed.) Topics in Cryptology - CT-RSA 2021 - Cryptographers' Track at the RSA Conference 2021, Virtual Event, May 17-20, 2021, Proceedings. Lecture Notes in Computer Science, vol. 12704, pp. 126–147. Springer (2021), https://doi.org/10.1007/978-3-030-75539-3_6

[100] Papapagiannopoulos, K.: High Throughput in Slices: The Case of PRESENT, PRINCE and KATAN64 Ciphers. In: Saxena, N., Sadeghi, A. (eds.) Radio Frequency Identification: Security and Privacy Issues - 10th International Workshop, RFIDSec 2014, Oxford, UK, July 21-23, 2014, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8651, pp. 137–155. Springer (2014), https://dx.doi.org/10.1007/978-3-319-13066-8_9

[101] Poschmann, A.: Lightweight Cryptography - Cryptographic Engineering for a Pervasive World. IACR Cryptology ePrint Archive 2009, 516 (2009), https://eprint.iacr.org/2009/516

[102] Preneel, B., Takagi, T. (eds.): Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings, Lecture Notes in Computer Science, vol. 6917. Springer (2011), https://dx.doi.org/10.1007/978-3-642-23951-9

[103] Qiao, K., Hu, L., Sun, S.: Differential Analysis on Simeck and SIMON with Dynamic Key-Guessing Techniques. In: Camp, O., Furnell, S., Mori, P. (eds.) Information Systems Security and Privacy - Second International Conference, ICISSP 2016, Rome, Italy, February 19-21, 2016, Revised Selected Papers. Communications in Computer and Information Science, vol. 691, pp. 64–85. Springer (2016), https://doi.org/10.1007/978-3-319-54433-5_5

[104] Rasoolzadeh, S., Raddum, H.: Cryptanalysis of PRINCE with Minimal Data. In: Pointcheval, D., Nitaj, A., Rachidi, T. (eds.) Progress in Cryptology - AFRICACRYPT 2016 - 8th International Conference on Cryptology in Africa, Fes, Morocco, April 13-15, 2016, Proceedings. Lecture Notes in Computer Science, vol. 9646, pp. 109–126. Springer (2016), https://doi.org/10.1007/978-3-319-31517-1_6

[105] Reis, T.B.S., Aranha, D.F., López-Hernández, J.C.: PRESENT Runs Fast - Efficient and Secure Implementation in Software. In: Fischer, W., Homma, N. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10529, pp. 644–664. Springer (2017), https://doi.org/10.1007/978-3-319-66787-4_31

[106] Rohit, R., Gong, G.: Correlated Sequence Attack on Reduced-Round Simon-32/64 and Simeck-32/64. IACR Cryptol. ePrint Arch. 2018, 699 (2018), https://eprint.iacr.org/2018/699

[107] Rolfes, C., Poschmann, A., Leander, G., Paar, C.: Ultra-Lightweight Implementations for Smart Devices - Security for 1000 Gate Equivalents. In: Grimaud, G., Standaert, F. (eds.) Smart Card Research and Advanced Applications, 8th IFIP WG 8.8/11.2 International Conference, CARDIS 2008, London, UK, September 8-11, 2008. Proceedings. Lecture Notes in Computer Science, vol. 5189, pp. 89–103. Springer (2008), https://dx.

doi.org/10.1007/978-3-540-85893-5_7

[108] Shibutani, K., Isobe, T., Hiwatari, H., Mitsuda, A., Akishita, T., Shirai, T.: Piccolo: An ultra-lightweight blockcipher. In: Preneel and Takagi [102], pp. 342–357, https://dx.doi.org/10.1007/978-3-642-23951-9_23

[109] Shirai, T., Shibutani, K., Akishita, T., Moriai, S., Iwata, T.: The 128-bit blockcipher CLEFIA (extended abstract). In: Biryukov, A. (ed.) Fast Software Encryption, 14th International Workshop, FSE 2007, Luxembourg, Luxembourg, March 26-28, 2007, Revised Selected Papers. Lecture Notes in Computer Science, vol. 4593, pp. 181–195. Springer (2007), https://dx.doi.org/10.1007/978-3-540-74619-5_12

[110] Soleimany, H.: Probabilistic Slide Cryptanalysis and Its Applications to LED-64 and Zorro. In: Cid, C., Rechberger, C. (eds.) Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8540, pp. 373–389. Springer (2014), https://doi.org/10.1007/978-3-662-46706-0_19

[111] Song, J., Lee, K., Lee, H.: Biclique cryptanalysis on lightweight block cipher: HIGHT and Piccolo. Int. J. Comput. Math. 90(12), 2564–2580 (2013), https://doi.org/10.1080/00207160.2013.767445

[112] Song, L., Huang, Z., Yang, Q.: Automatic Differential Analysis of ARX Block Ciphers with Application to SPECK and LEA. In: Liu, J.K., Steinfeld, R. (eds.) Information Security and Privacy - 21st Australasian Conference, ACISP 2016, Melbourne, VIC, Australia, July 4-6, 2016, Proceedings, Part II. Lecture Notes in Computer Science, vol. 9723, pp. 379–394. Springer (2016), https://doi.org/10.1007/978-3-319-40367-0_24

[113] Sun, L., Wang, W., Wang, M.: Automatic Search of Bit-Based Division Property for ARX Ciphers and Word-Based Division Property. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10624, pp. 128–157. Springer (2017), https://doi.org/10.1007/978-3-319-70694-8_5

[114] Sun, L., Wang, W., Wang, M.: More Accurate Differential Properties of LED64 and Midori64. IACR Trans. Symmetric Cryptol. 2018(3), 93–123 (2018), https://doi.org/10.13154/tosc.v2018.i3.93-123

[115] Sun, L., Wang, W., Wang, M.: MILP-aided bit-based division property for primitives with non-bit-permutation linear layers. IET Inf. Secur. 14(1), 12–20 (2020), https://doi.org/10.1049/iet-ifs.2018.5283

[116] Sun, L., Wang, W., Wang, M.: Accelerating the Search of Differential and Linear Characteristics with the SAT Method. IACR Trans. Symmetric Cryptol. 2021(1), 269–315 (2021), https://doi.org/10.46586/tosc.v2021.i1.269-315

[117] Suzaki, T., Minematsu, K.: Improving the Generalized Feistel. In: Hong, S., Iwata, T. (eds.) Fast Software Encryption, 17th International Workshop, FSE 2010, Seoul, Korea, February 7-10, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6147, pp. 19–39. Springer (2010), https://doi.org/10.1007/978-3-642-13858-4_2

[118] Suzaki, T., Minematsu, K., Morioka, S., Kobayashi, E.: TWINE : A Lightweight Block Cipher for Multiple Platforms. In: Knudsen, L.R., Wu, H. (eds.) Selected Areas in Cryptography, 19th International Conference, SAC 2012, Windsor, ON, Canada, August 15-16, 2012, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7707, pp. 339–354. Springer (2012), https://dx.doi.org/10.1007/978-3-642-35999-6_22

[119] Tezcan, C., Selçuk, A.A.: Improved improbable differential attacks on ISO standard CLEFIA: Expansion technique revisited. Inf. Process. Lett. 116(2), 136–143 (2016), https://doi.org/10.1016/j.ipl.2015.09.010

[120] Todo, Y., Leander, G., Sasaki, Y.: Nonlinear Invariant Attack - Practical Attack on Full SCREAM, iSCREAM, and Midori64. In: Cheon, J.H., Takagi, T. (eds.) Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10032, pp. 3–33 (2016), https://doi.org/10.1007/978-3-662-53890-6_1

[121] Todo, Y., Leander, G., Sasaki, Y.: Nonlinear Invariant Attack: Practical Attack on Full SCREAM, iSCREAM, and Midori64. J. Cryptol. 32(4), 1383–1422 (2019), https://doi.org/10.1007/s00145-018-9285-0

[122] Tolba, M., Abdelkhalek, A., Youssef, A.M.: Meet-in-the-Middle Attacks on Reduced Round Piccolo. In: Güneysu, T., Leander, G., Moradi, A. (eds.) Lightweight Cryptography for Security and Privacy - 4th International Workshop, LightSec 2015, Bochum, Germany, September 10-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9542, pp. 3–20. Springer (2015), https://doi.org/10.1007/978-3-319-29078-2_1

[123] Tolba, M., Abdelkhalek, A., Youssef, A.M.: Truncated and Multiple Differential Cryptanalysis of Reduced Round Midori128. In: Bishop, M., Nascimento, A.C.A. (eds.) Information Security - 19th International Conference, ISC 2016, Honolulu, HI, USA, September 3-6, 2016, Proceedings. Lecture Notes in Computer Science, vol. 9866, pp. 3–17. Springer (2016), https://doi.org/10.1007/978-3-319-45871-7_1

[124] Tolba, M., Youssef, A.M.: Generalized MitM attacks on full TWINE. Inf. Process. Lett. 116(2), 128–135 (2016), https://doi.org/10.1016/j.ipl.2015.09.011

[125] Tsunoo, Y., Tsujihara, E., Shigeri, M., Saito, T., Suzaki, T., Kubo, H.: Impossible Differential Cryptanalysis of CLEFIA. In: Nyberg, K. (ed.) Fast Software Encryption, 15th International Workshop, FSE 2008, Lausanne, Switzerland, February 10-13, 2008, Revised Selected Papers. Lecture Notes in Computer Science, vol. 5086, pp. 398–411. Springer (2008), https://doi.org/10.1007/978-3-540-71039-4_25

[126] Wang, G., Wang, G.: Improved Differential-ML Distinguisher: Machine Learning Based Generic Extension for Differential Analysis. In: Gao, D., Li, Q., Guan, X., Liao, X. (eds.) Information and Communications Security - 23rd International Conference, ICICS 2021, Chongqing, China, November 19-21, 2021, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12919, pp. 21–38. Springer (2021), https://doi.org/10.1007/978-3-030-88052-1_2

[127] Wang, N., Wang, X., Jia, K., Zhao, J.: Differential attacks on reduced SIMON versions with dynamic key-guessing techniques. Sci. China Inf. Sci. 61(9), 098103:1–098103:3 (2018), https://doi.org/10.1007/s11432-017-9231-5

[128] Wang, X., Wu, B., Hou, L., Lin, D.: Automatic Search for Related-Key Differential Trails in SIMON-like Block Ciphers Based on MILP. In: Chen, L., Manulis, M., Schneider, S.A. (eds.) Information Security - 21st International Conference, ISC 2018, Guildford, UK, September 9-12, 2018, Proceedings. Lecture Notes in Computer Science, vol. 11060, pp. 116–131. Springer (2018), https://doi.org/10.1007/978-3-319-99136-8_7

[129] Wang, Y., Wu, W.: Improved Multidimensional Zero-Correlation Linear Cryptanalysis and Applications to LBlock and TWINE. In: Susilo, W., Mu, Y. (eds.) Information Security and Privacy - 19th Australasian Conference, ACISP 2014, Wollongong, NSW, Australia, July 7-9, 2014. Proceedings. Lecture Notes in Computer Science, vol. 8544, pp. 1–16. Springer (2014), https://doi.org/10.1007/978-3-319-08344-5_1

[130] Wei, Y., Xu, P., Rong, Y.: Related-key impossible differential cryptanalysis on lightweight cipher TWINE. J. Ambient Intell. Humaniz. Comput. 10(2), 509–517 (2019), https://doi.org/10.1007/s12652-017-0675-1

[131] Yi, W., Wu, B., Chen, S., Lin, D.: Improved Integral and Zero-correlation Linear Cryptanalysis of CLEFIA Block Cipher. In: Chen, K., Lin, D., Yung, M. (eds.) Information Security and Cryptology - 12th International Conference, Inscrypt 2016, Beijing, China, November 4-6, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 10143, pp. 33–46. Springer (2016), https://doi.org/10.1007/978-3-319-54705-3_3

[132] Zhao, H., Han, G.: Biclique cryptanalysis on Midori block cipher. Int. J. Embed. Syst. 11(2), 229–239 (2019), https://doi.org/10.1504/IJES.2019.098299

[133] Zhao, H., Han, G., Wang, L., Wang, W.: MILP-Based Differential Cryptanalysis on Round-Reduced Midori64. IEEE Access 8, 95888–95896 (2020), https://doi.org/10.1109/ACCESS.2020.2995795

[134] Zheng, L., Zhang, S.: FFT-based multidimensional linear attack on PRESENT using the 2-bit-fixed characteristic. Secur. Commun. Networks 8(18), 3535–3545 (2015), https://doi.org/10.1002/sec.1278

[135] Zheng, X., Jia, K.: Impossible Differential Attack on Reduced-Round TWINE. In: Lee, H., Han, D. (eds.) Information Security and Cryptology - ICISC 2013 - 16th International Conference, Seoul, Korea, November 27-29, 2013, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8565, pp. 123–143. Springer (2013), https://doi.org/10.1007/978-3-319-12160-4_8

[136] 伊藤竜馬:「CRYPTREC 暗号技術ガイドライン（軽量暗号）」掲載の暗号方式に関する安全性評価の動向調査（文書番号：CRYPTREC EX-3101-2021）(2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf

4.2 ストリーム暗号

本節では軽量なストリーム暗号を取り上げる。ストリーム暗号はデータの秘匿を行うための関数だが、同じ機能を提供するブロック暗号と異なり、機能を秘匿のみに絞っている。したがって、（ソフトウェア実装での）プログラム領域や、（ハードウェア実装での）回路規模などリソースに関する制約が強く、複数の暗号化関数を実装できないような場合、たとえば単一の暗号化関数を用いてメッセージ認証子も生成したい、といった場合には、ブロック暗号を選択するのが適当である。また、多くのストリーム暗号アルゴリズムは、初期化に要する時間が長くなりがちであり、短いデータを処理する場合にはブロック暗号が適する場合が多い。逆に、秘匿機能だけで良いが、省リソースで高速な処理を行いたい、といった場合にはストリーム暗号が適する。

軽量暗号の中では、ストリーム暗号の提案／評価が先行して実施された経緯もあり、他に比べて成熟しているアルゴリズムが多い。そこで、本節では、安全性評価が十分に行われたと考えられる eStream portfolio [30] と ISO/IEC 29192-3 [58] に掲載されたアルゴリズムを中心に紹介する。eStream プロジェクトでは、ソフトウェア向けの Profile 1 とハードウェア向けの Profile 2 に分けてアルゴリズムの公募、評価を行っているが、本節では Profile 2 で portfolio に掲載された 3 つのアルゴリズム Grain v1 [37]、MICKEY 2.0 [4]、Trivium [12] を取り上げる。ソフトウェア向けの Profile 1 では 4 つのアルゴリズム HC-128、Rabbit、Salsa20/12、SOSEMANUK が portfolio に掲載されている。eBACS の PC、サーバ向け CPU を使った処理性能比較によれば、長いメッセージの処理については Salsa20/12 がもっとも優れている。ここでは、Salsa20/12 のかわりに、2015 年に RFC 化された ChaCha20 [10] を紹介する。ChaCha20 は、Salsa20/12 の改良版として開発されたソフトウェア向けのストリーム暗号であり、Salsa20/12 に比べると処理速度の点で幾分劣る。しかし、多くのオープンソースに採用されており、利用しやすさという点で優れている。ISO/IEC 29292-3 [58] は 2 つのアルゴリズム Trivium と Enocoro を掲載している。そこで、本節では、上記のアルゴリズムに加えて Enocoro を取り上げる。

各アルゴリズムの安全性解析状況については、4.1 節と同様、2021 年度に公開された CRYPTREC 外部評価報告書 [71] に基づき、2021 年 9 月時点の状況を記載している。

その他、軽量暗号に関する ISO 標準としては、RFID 向けの暗号技術を取り扱う ISO/IEC 29167 があるが、安全性の観点で望ましくない「XOR 暗号」が含まれており、CRYPTREC では一般に使用を奨めない。

技術分野	ストリーム暗号
名称	ChaCha20
設計者	Daniel J. Bernstein (The University of Illinois at Chicago/USA)
発表年	2008（SASC 2008 [10]）
仕様参照先	SASC 2008 [10]、IETF RFC 7539 [54]
特徴	鍵長 256 ビット、IV 長 96 ビットのストリーム暗号であり、秘密鍵、IV に加えて 128 ビットの定数、32 ビットのブロックカウンタを入力として 512 ビットの擬似乱数を出力する。アルゴリズムは 32 ビットワードの算術加算、排他的論理和、巡回シフトで構成されており、ソフトウェア実装に適する。特に、初期化処理がほとんど無いため、短いメッセージに対しても高速であるという特徴を持つ。また、アルゴリズム中でテーブル参照を行わないため、素直に実装してもキャッシュタイミング攻撃に対して安全である。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [3, 7, 8, 13, 14, 15, 16, 19, 21, 22, 23, 51, 52, 53, 57] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 20 段のうち 7 段に簡略化した ChaCha20 に対する最良の攻撃は、2020 年に提案された Beierle ら [8] による差分線形攻撃と 2021 年に提案された Coutinho ら [15, 16] による差分線形攻撃であり、Beierle ら [8] による攻撃では秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行でき、Coutinho ら [15, 16] による攻撃では効率的に識別攻撃が実行できる。また、20 段のうち 7.25 段に簡略化した ChaCha20 に対する最良の攻撃は、2021 年に提案された Miyashita ら [52] による差分攻撃であり、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。
主な実装評価結果	eBASC [1] での評価（2016 年 6 月現在）によれば、Intel Core i5 上で 1.2 cycles/byte を達成している。また、FELICS [2] での評価（2016 年 6 月現在）によれば、Arm Cortex-M3 上の評価で、初期化に 144 cycles を要し、スループットは 54.3 cycles/byte である。
標準化状況	IETF RFC 7539 [54]
利用実績等	主に多項式型メッセージ認証子の Poly-1305 と組み合わせた認証暗号として利用される。Google が提供するサービスの通信路 (https) 保護に利用されている [11]。
オープンソース	OpenSSL, Google Chrome, Mozilla Firefox, OpenSSH など。

技術分野	ストリーム暗号
名称	Enocoro
設計者	Hitachi, Ltd.
発表年	2008（WAIS 2008）、2010（ISITA 2010 [66]）
仕様参照先	ISITA 2010 [66]、設計者ウェブサイト [72]
特徴	鍵長 80 ビットの Enocoro-80、鍵長 128 ビットの Enocoro-128v2 の 2 つのアルゴリズムから成る。いずれのアルゴリズムについても鍵長相当の安全性を謳っているが、鍵、IV を固定するごとに出力するデータはそれぞれ 2^{32} バイト、2^{64} バイトに制限されている。軽量ストリーム暗号には珍しく、8 ビット単位の処理で構成されており、ソフトウェア実装でも AES と同等の処理速度が得られる。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [26, 66, 73, 74, 76, 77] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 Enocoro-80 に対する最良の攻撃は、2015 年に提案された Ding ら [26] による弱鍵攻撃であり、Enocoro-80 の秘密鍵は確率 2^{-8} で弱鍵であり、2^{17} 個の選択 IV を用いることで、計算量 2^{48} で鍵回復攻撃を実行できる。Enocoro-128v2 に対する最良の攻撃は、2019 年に提案された船引ら [77] によるキューブ攻撃と 2021 年に提案された芝山ら [76] による高階差分攻撃であり、96 段のうち 11 段に簡略化した Enocoro-128v2 に対して秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行でき、96 段のうち 22 段に簡略化した Enocoro-128v2 に対して効率的に識別攻撃が実行できる [76]。
主な実装評価結果	Enocoro-80 [65]. Pentium 4 上での実装では、初期化に 1,335 cycles を要し、スループットが 27 cycles/byte である。また、ハードウェア実装 (ASIC) での性能は回路規模が 2.7 KGE, 処理速度が 2,197.6 Mbps（180nm プロセス、最大動作周波数 274.7 MHz）である。 Enocoro-128v2 [75]. Intel Core2 Duo 上での実装では、初期化に 1,530 cycles を要し、スループットが 14.8 cycles/byte である。また、ハードウェア実装 (ASIC) での性能は回路規模が 2.4 KGE, 処理速度が 6,250 Mbps（90 nm プロセス、最大動作周波数 781.3 MHz）である。
標準化状況	CRYPTREC 推奨候補暗号（Enocoro-128v2） [70]、 ISO/IEC 29192-3 [42]

技術分野	ストリーム暗号
名称	Grain v1
設計者	Martin Hell[1], Thomas Johansson[1], Willi Meier[2] (1: Lund University/Sweden, 2: FH Aargau/Switzerland)
発表年	2005（eSTREAM Project）
仕様参照先	eSTREAM ウェブサイト [29]
特徴	eSTREAM portfolio に選ばれた鍵長 80 ビット（IV 長 64 ビット）、鍵長 128 ビット（IV 長 80 ビット）の 2 つのアルゴリズムから成る、ハードウェア実装向けのストリーム暗号である。ビット単位で処理を行う線形フィードバックシフトレジスタ 1 個と非線形フィードバックシフトレジスタ 1 個を組み合わせている。軽量ストリーム暗号の中でも特にハードウェア実装の軽量性に優れている。ある程度の並列処理が可能であり、ソフトウェア実装でも実用に耐える。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [5, 6, 9, 17, 18, 28, 45, 46, 48, 49, 50, 55, 56, 62, 68, 69] が発表されている。 単一鍵設定における最良の攻撃は、2018 年に提案された Todo ら [62] による高速相関攻撃であり、仕様段数において効率的に内部状態復元攻撃が実行できる。なお、本攻撃では $2^{76.7}$ の計算量と $2^{75.1}$ のデータ量が必要となる。
主な実装評価結果	Grain v1 のハードウェア実装性能については、Good らの評価 [33] が詳しい。この評価では、0.13 μm スタンダードセルライブラリを用いて評価を行っている。 **鍵長 80 ビット**. 回路規模 1,294 GE で最大動作周波数は 724.6 MHz、スループットは 724.6 MHz である。また、回路を 16 個まで並列処理することが可能である。 **鍵長 128 ビット**. 回路規模 1,857 GE で最大動作周波数は 925.9 MHz、スループットは 925.9 MHz である。また、回路を 32 個まで並列処理することが可能である。
標準化状況	Grain v1 をベースにした認証暗号 Grain-128A が ISO/IEC 29192-8 [43] と ISO/IEC 29167-13 [44] にて標準化されている。

技術分野	ストリーム暗号
名称	MICKEY 2.0
設計者	Steve Babbage[1], Matthew Dodd[2] (1: Vodafone/UK, 2: Independent consultant)
発表年	2005（eSTREAM Project）
仕様参照先	eSTREAM ウェブサイト [29]
特徴	eSTREAM portfolio に選ばれた鍵長 80 ビット、IV 長 80 ビットのハードウェア実装向けストリーム暗号である。1 つの鍵に対して、利用可能な IV の数は最大 2^{40} に制限されている。また、鍵と IV のペアに対して、利用可能な鍵ストリームは 2^{40} ビットに制限されている。線形フィードバックシフトレジスタ 1 個と非線形フィードバックシフトレジスタ 1 個を組み合わせており、不規則なクロック制御を行うことを特徴としている。このクロック制御機構が原因で、並列処理は困難である。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [24, 25, 27, 38, 39] が発表されている。 単一鍵設定における最良の攻撃は、2019 年に提案された Ding ら [24] によって提案されたタイムメモリデータトレードオフ攻撃であり、仕様段数において秘密鍵の全数探索より効率的に鍵回復攻撃が実行できる。ただし、本攻撃は IV 長が 64 ビットの場合に成立し、$2^{79.0}$ の計算量、2^{80} のデータ量、$2^{45.0}$ のメモリ量が必要となる。
主な実装評価結果	MICKEY 2.0 のハードウェア実装性能については、Good らの評価 [33] が詳しい。この評価では、0.13 μm スタンダードセルライブラリを用いて評価を行っている。MICKEY 2.0 の回路規模は 3,188 GE であり、最大動作周波数は 454.5 MHz、スループットは 454.5 MHz である。

技術分野	ストリーム暗号
名称	Trivium
設計者	Christophe De Cannière, Bart Preneel (KU Leuven/Belgium)
発表年	2005（eSTREAM Project）
仕様参照先	eSTREAM ウェブサイト [29]
特徴	eSTREAM portfolio に掲載された鍵長 80 ビット、IV 長 80 ビットのハードウェア実装向けのストリーム暗号である。鍵と IV のペアごとに生成される鍵ストリームは 2^{64} ビットに制限される。3 つの非線形フィードバックシフトレジスタを直列した特徴的なアルゴリズムである。ビット単位の処理を基本としながら、高い並列性を持ち、ハードウェア実装での軽量性とソフトウェア実装での高速性を両立している。ただし、初期化に要する時間が長いため、短いデータの処理には適さない。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [20, 31, 32, 34, 35, 36, 40, 41, 47, 59, 60, 61, 63, 64, 67] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 最良の攻撃は、2021 年に提案された Hu ら [40] によるキューブ攻撃であり、1152 段のうち 845 段に簡略化した Trivium の初期化フェーズに対して、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。
主な実装評価結果	Trivium のハードウェア実装性能については、Good らの評価 [33] が詳しい。この評価では、0.13 μm スタンダードセルライブラリを用いて評価を行っている。Trivium の回路規模は 2,580 GE であり、最大動作周波数は 327.9 MHz、スループットは 327.9 Mbps である。また、Trivium のアルゴリズムは最大で 64 並列で実行することが可能であり、このときの回路規模は 4,921 GE、スループットは 22,299.6 Mbps である。 また、ソフトウェア実装については、FELICS [2] で評価が行われている。Arm Cortex-M3 上でのスループットは 49.4 cycles/byte であり、ChaCha20 よりも高速である。ただし、初期化に 7,195 cycles を要するため、短いデータの処理には適さない。
標準化状況	ISO/IEC 29192-3 [42]

参考文献

[1] eBACS: ECRYPT Benchmarking of Cryptographic Systems, https://bench.cr.yp.to/results-stream.html（2023-10-04 閲覧）

[2] FELICS Stream Ciphers Brief Results, https://www.cryptolux.org/index.php/FELICS_Stream_Ciphers_Brief_Results（2023-10-04 閲覧）

[3] Aumasson, J., Fischer, S., Khazaei, S., Meier, W., Rechberger, C.: New Features of Latin Dances: Analysis of Salsa, ChaCha, and Rumba. In: Nyberg, K. (ed.) Fast Software Encryption, 15th International Workshop, FSE 2008, Lausanne, Switzerland, February 10-13, 2008, Revised Selected Papers. Lecture Notes in Computer Science, vol. 5086, pp. 470–488. Springer (2008), https://doi.org/10.1007/978-3-540-71039-4_30

[4] Babbage, S., Dodd, M.: The MICKEY Stream Ciphers. In: Robshaw, M., O. Billet, e. (eds.) New Stream Cipher Designs: The eSTREAM Finalists. Lecture Notes in Computer Science, vol. 4986, pp. 191–209. Springer (2008)

[5] Banik, S.: Some Insights into Differential Cryptanalysis of Grain v1. In: Susilo, W., Mu, Y. (eds.) Information Security and Privacy - 19th Australasian Conference, ACISP 2014, Wollongong, NSW, Australia, July 7-9, 2014. Proceedings. Lecture Notes in Computer Science, vol. 8544, pp. 34–49. Springer (2014), https://doi.org/10.1007/978-3-319-08344-5_3

[6] Banik, S.: Conditional differential cryptanalysis of 105 round Grain v1. Cryptogr. Commun. 8(1), 113–137 (2016), https://doi.org/10.1007/s12095-015-0146-5

[7] Barbero, S., Bellini, E., Makarim, R.H.: Rotational analysis of ChaCha permutation. IACR Cryptol. ePrint Arch. 2020, 1049 (2020), https://eprint.iacr.org/2020/1049

[8] Beierle, C., Leander, G., Todo, Y.: Improved Differential-Linear Attacks with Applications to ARX Ciphers. In: Micciancio, D., Ristenpart, T. (eds.) Advances in Cryptology - CRYPTO 2020 - 40th Annual International Cryptology Conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17-21, 2020, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12172, pp. 329–358. Springer (2020), https://doi.org/10.1007/978-3-030-56877-1_12

[9] Berbain, C., Gilbert, H., Maximov, A.: Cryptanalysis of Grain. In: Robshaw, M.J.B. (ed.) Fast Software Encryption, 13th International Workshop, FSE 2006, Graz, Austria, March 15-17, 2006, Revised Selected Papers. Lecture Notes in Computer Science, vol. 4047, pp. 15–29. Springer (2006), https://doi.org/10.1007/11799313_2

[10] Bernstein, D.J.: ChaCha, a variant of Salsa20. In: The State of the Art of Stream Ciphers, SASC 2008. ECRYPT (2008)

[11] Bursztein, E.: Google Security Blog: Speeding up and strengthening HTTPS connections for Chrome on Android (April 24, 2014) (2014), https://security.googleblog.com/2014/04/speeding-up-and-strengthening-https.html（2023-10-04 閲覧）

[12] Cannière, C.D.: Trivium: A Stream Cipher Construction Inspired by Block Cipher Design Principles. In: Katsikas, S.K., López, J., Backes, M., Gritzalis, S., Preneel, B. (eds.) Information Security, ISC 2006. Lecture Notes in Computer Science, vol. 4176, pp. 171–186. Springer Berlin Heidelberg (2006)

[13] Choudhuri, A.R., Maitra, S.: Significantly Improved Multi-bit Differentials for Reduced Round Salsa and ChaCha. IACR Trans. Symmetric Cryptol. 2016(2), 261–287 (2016), https://doi.org/10.13154/tosc.v2016.i2.261-287

[14] Coutinho, M., Neto, T.C.S.: New Multi-bit Differentials to Improve Attacks Against ChaCha. IACR Cryptol. ePrint Arch. 2020, 350 (2020), https://eprint.iacr.org/2020/350

[15] Coutinho, M., Neto, T.C.S.: Improved Linear Approximations to ARX Ciphers and Attacks Against ChaCha. IACR Cryptol. ePrint Arch. p. 224 (2021), https://eprint.iacr.org/2021/224

[16] Coutinho, M., Neto, T.C.S.: Improved Linear Approximations to ARX Ciphers and Attacks Against ChaCha. In: Canteaut, A., Standaert, F. (eds.) Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12696, pp. 711–740. Springer (2021), https://doi.org/10.1007/978-3-030-77870-5_25

[17] Dalai, D.K., Maitra, S., Pal, S., Roy, D.: Distinguisher and non-randomness of Grain-v1 for 112, 114 and 116 initialisation rounds with multiple-bit difference in IVs. IET Inf. Secur. 13(6), 603–613 (2019), https://doi.org/10.1049/iet-ifs.2018.5276

[18] Dalai, D.K., Pal, S.: Recovering Internal States of Grain-v1. In: Heng, S., López, J. (eds.) Information Security Practice and Experience - 15th International Conference, ISPEC 2019, Kuala Lumpur, Malaysia, November 26-28, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11879, pp. 325–337. Springer (2019), https://doi.org/10.1007/978-3-030-34339-2_18

[19] Deepthi, K.K.C., Singh, K.: Cryptanalysis for reduced round Salsa and ChaCha: revisited. IET Inf. Secur. 13(6), 591–602 (2019), https://doi.org/10.1049/iet-ifs.2018.5328

[20] Delaune, S., Derbez, P., Gontier, A., Prudhomme, C.: A Simpler Model for Recovering Superpoly on Trivium. IACR Cryptol. ePrint Arch. 2021, 1191 (2021), https://eprint.iacr.org/2021/1191, (accepted on *Selected Areas in Cryptography - 28th International Workshop, SAC 2021*)

[21] Dey, S., Sarkar, S.: Improved analysis for reduced round Salsa and Chacha. Discret. Appl. Math. 227, 58–69 (2017), https://doi.org/10.1016/j.dam.2017.04.034

[22] Dey, S., Sarkar, S.: Proving the biases of Salsa and ChaCha in differential attack. Des. Codes Cryptogr. 88(9), 1827–1856 (2020), https://doi.org/10.1007/s10623-020-00736-9

[23] Dey, S., Sarkar, S.: A theoretical investigation on the distinguishers of Salsa and ChaCha. Discret. Appl. Math. 302, 147–162 (2021), https://doi.org/10.1016/j.dam.2021.06.017

[24] Ding, L., Gu, D., Wang, L.: New Key Recovery Attack on the MICKEY Family of Stream Ciphers. In: Shen, B., Wang, B., Han, J., Yu, Y. (eds.) International Conference on Frontiers in Cyber Security - FCS 2019. Communications in Computer and Information Science, vol. 1105, pp. 239–249. Springer (2019), https://doi.org/10.1007/978-981-15-0818-9_16

[25] Ding, L., Guan, J.: Cryptanalysis of MICKEY family of stream ciphers. Secur. Commun. Networks 6(8), 936–941 (2013), https://doi.org/10.1002/sec.637

[26] Ding, L., Jin, C., Guan, J.: Slide attack on standard stream cipher Enocoro-80 in the related-key chosen IV setting. Pervasive Mob. Comput. 24, 224–230 (2015), https://doi.org/10.1016/j.pmcj.2015.08.002

[27] Ding, L., Jin, C., Guan, J., Qi, C.: New Treatment of the BSW Sampling and Its Applications to Stream Ciphers. In: Pointcheval, D., Vergnaud, D. (eds.) Progress in Cryptology - AFRICACRYPT 2014 - 7th International Conference on Cryptology in Africa, Marrakesh, Morocco, May 28-30, 2014. Proceedings. Lecture Notes in Computer Science, vol. 8469, pp. 136–146. Springer (2014), https://doi.org/10.1007/978-3-319-06734-6_9

[28] Dinur, I., Shamir, A.: Breaking Grain-128 with Dynamic Cube Attacks. In: Joux, A. (ed.) Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6733, pp. 167–187. Springer (2011), https://doi.org/10.1007/978-3-642-21702-9_10

[29] ECRYPT: FELICS – Fair Evaluation of Lightweight Cryptographic Systems, https://www.ecrypt.eu.org/stream/project.html（2023-10-04 閲覧）

[30] of Excellence, E.N.: The eSTREAM Project, https://www.ecrypt.eu.org/stream/（2023-10-04 閲覧）

[31] Fouque, P., Vannet, T.: Improving Key Recovery to 784 and 799 Rounds of Trivium Using Optimized Cube Attacks. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8424, pp. 502–517. Springer (2013), https://doi.org/10.1007/978-3-662-43933-3_26

[32] Fu, X., Wang, X., Dong, X., Meier, W.: A Key-Recovery Attack on 855-round Trivium. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10992, pp. 160–184. Springer (2018), https://doi.org/10.1007/978-3-319-96881-0_6

[33] Good, T., Benaissa, M.: Hardware performance of eStream phase-III stream cipher candidates. In: The State of the Art of Stream Ciphers, SASC 2008 (2008)

[34] Hao, Y., Isobe, T., Jiao, L., Li, C., Meier, W., Todo, Y., Wang, Q.: Improved Division Property Based Cube Attacks Exploiting Algebraic Properties of Superpoly. IEEE Trans. Computers 68(10), 1470–1486 (2019), https://doi.org/10.1109/TC.2019.2909871

[35] Hao, Y., Leander, G., Meier, W., Todo, Y., Wang, Q.: Modeling for Three-Subset Division Property Without Unknown Subset - Improved Cube Attacks Against Trivium and Grain-128AEAD. In: Canteaut, A., Ishai, Y. (eds.) Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12105, pp. 466–495. Springer (2020), https://doi.org/10.1007/978-3-030-45721-1_17

[36] Hao, Y., Leander, G., Meier, W., Todo, Y., Wang, Q.: Modeling for Three-Subset Division Property without Unknown Subset. J. Cryptol. 34(3), 22 (2021), https://doi.org/10.1007/s00145-021-09383-2

[37] Hell, M., Johansson, T., Maximov, A., Meier, W.: The Grain Family of Stream Ciphers. In: Robshaw, M., O. Billet, e. (eds.) New Stream Cipher Designs: The eSTREAM Finalists. Lecture Notes in Computer Science, vol. 4986, pp. 179–190. Springer (2008)

[38] Helleseth, T., Jansen, C.J.A., Kazymyrov, O., Kholosha, A.: State space cryptanalysis of the MICKEY cipher. In: 2013 Information Theory and Applications Workshop, ITA 2013, San Diego, CA, USA, February 10-15, 2013. pp. 1–10. IEEE (2013), https://doi.org/10.1109/ITA.2013.6502941

[39] Hong, J., Kim, W.: TMD-Tradeoff and State Entropy Loss Considerations of Streamcipher MICKEY. In: Maitra, S., Madhavan, C.E.V., Venkatesan, R. (eds.) Progress in Cryptology - INDOCRYPT 2005, 6th International Conference on Cryptology in India, Bangalore, India, December 10-12, 2005, Proceedings. Lecture Notes in Computer Science, vol. 3797, pp. 169–182. Springer (2005), https://doi.org/10.1007/11596219_14

[40] Hu, K., Sun, S., Todo, Y., Wang, M., Wang, Q.: Massive Superpoly Recovery with Nested Monomial Predictions. In: Tibouchi, M., Wang, H. (eds.) Advances in Cryptology - ASIACRYPT 2021 - 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6-10, 2021, Proceedings, Part I. Lecture Notes in Computer Science, vol. 13090, pp. 392–421. Springer (2021), https://doi.org/10.1007/978-3-030-92062-3_14

[41] Hu, K., Sun, S., Wang, M., Wang, Q.: An Algebraic Formulation of the Division Property: Revisiting Degree Evaluations, Cube Attacks, and Key-Independent Sums. In: Moriai, S., Wang, H. (eds.) Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12491, pp. 446–476. Springer (2020), https://doi.org/10.1007/978-3-030-64837-4_15

[42] ISO/IEC: Information security – Lightweight cryptography – Part 3: Stream ciphers (ISO/IEC 29192-3:2012), https://www.iso.org/standard/56426.html

[43] ISO/IEC: Information security – Lightweight cryptography – Part 8: Authenticated encryption (ISO/IEC 29192-8:2022), https://www.iso.org/standard/80114.html

[44] ISO/IEC: Information technology – Automatic identification and data capture techniques – Part 13: Crypto suite Grain-128A security services for air interface communications (ISO/IEC 29167-13: 2015), https://www.iso.org/standard/60682.html

[45] Li, J., Guan, J.: Advanced conditional differential attack on Grain-like stream cipher and application on Grain v1. IET Inf. Secur. 13(2), 141–148 (2019), https://doi.org/10.1049/iet-ifs.2018.5180

[46] Li, J., Guan, J.: Improved Conditional Differential Attacks on Round-Reduced Grain v1. KSII Trans. Internet Inf. Syst. 12(9), 4548–4559 (2018), https://doi.org/10.3837/tiis.2018.09.023

[47] Liu, M., Yang, J., Wang, W., Lin, D.: Correlation Cube Attacks: From Weak-Key Distinguisher to Key Recovery. In: Nielsen, J.B., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2018 - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29 - May 3, 2018 Proceedings, Part II. Lecture Notes in Computer Science, vol. 10821, pp. 715–744. Springer (2018), https://doi.org/10.1007/978-3-319-78375-8_23

[48] Ma, Z., Tian, T., Qi, W.: Improved conditional differential attacks on Grain v1. IET Inf. Secur. 11(1), 46–53 (2017), https://doi.org/10.1049/iet-ifs.2015.0427

[49] Ma, Z., Tian, T., Qi, W.: Internal state recovery of Grain v1 employing guess-and-determine attack. IET Inf. Secur. 11(6), 363–368 (2017), https://doi.org/10.1049/iet-ifs.2017.0232

[50] Ma, Z., Tian, T., Qi, W.: A New Distinguishing Attack on Grain-V1 with 111 Initialization Rounds. J. Syst. Sci. Complex. 32(3), 970–984 (2019), https://doi.org/10.1007/s11424-018-7170-4

[51] Maitra, S.: Chosen IV cryptanalysis on reduced round ChaCha and Salsa. Discret. Appl. Math. 208, 88–97 (2016), https://doi.org/10.1016/j.dam.2016.02.020

[52] Miyashita, S., Ito, R., Miyaji, A.: PNB-based Differential Cryptanalysis of ChaCha Stream Cipher. IACR Cryptol. ePrint Arch. 2021, 1537 (2021), https://eprint.iacr.org/2021/1537

[53] Neves, S., Araújo, F.: An observation on NORX, BLAKE2, and ChaCha. Inf. Process. Lett. 149, 1–5 (2019), https://doi.org/10.1016/j.ipl.2019.05.001

[54] Nir, Y., Langley, A.: ChaCha20 and Poly1305 for IETF Protocols, Request For Comments, vol. RFC7539 (May), https://tools.ietf.org/html/rfc7539

[55] Pan, S., Wu, Y., Wang, L.: Optimizing Fast Near Collision Attack on Grain Using Linear Programming. IEEE Access 7, 181191–181201 (2019), https://doi.org/10.1109/ACCESS.2019.2959334

[56] Rahimi, M., Barmshory, M., Mansouri, M.H., Aref, M.R.: Dynamic cube attack on Grain-v1. IET Inf. Secur. 10(4), 165–172 (2016), https://doi.org/10.1049/iet-ifs.2014.0239

[57] Shi, Z., Zhang, B., Feng, D., Wu, W.: Improved Key Recovery Attacks on Reduced-Round Salsa20 and ChaCha. In: Kwon, T., Lee, M., Kwon, D. (eds.) Information Security and Cryptology - ICISC 2012 - 15th International Conference, Seoul, Korea, November 28-30, 2012, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7839, pp. 337–351. Springer (2012), https://doi.org/10.1007/978-3-642-37682-5_24

[58] for Standards, I.O.: ISO/IEC 29192-3:2012 Information technology – Security techniques – Lightweight cryptography – Part 3: Stream ciphers (October 2012)

[59] Sun, Y.: Cube Attack against 843-Round Trivium. IACR Cryptol. ePrint Arch. p. 547 (2021), https://eprint.iacr.org/2021/547

[60] Todo, Y., Isobe, T., Hao, Y., Meier, W.: Cube Attacks on Non-Blackbox Polynomials Based on Division Property. In: Katz, J., Shacham, H. (eds.) Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10403, pp. 250–279. Springer (2017), https://doi.org/10.1007/978-3-319-63697-9_9

[61] Todo, Y., Isobe, T., Hao, Y., Meier, W.: Cube Attacks on Non-Blackbox Polynomials Based on Division Property. IEEE Trans. Computers 67(12), 1720–1736 (2018), https://doi.org/10.1109/TC.2018.2835480

[62] Todo, Y., Isobe, T., Meier, W., Aoki, K., Zhang, B.: Fast Correlation Attack Revisited - Cryptanalysis on

Full Grain-128a, Grain-128, and Grain-v1. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10992, pp. 129–159. Springer (2018), https://doi.org/10.1007/978-3-319-96881-0_5

[63] Wang, Q., Hao, Y., Todo, Y., Li, C., Isobe, T., Meier, W.: Improved Division Property Based Cube Attacks Exploiting Algebraic Properties of Superpoly. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10991, pp. 275–305. Springer (2018), https://doi.org/10.1007/978-3-319-96884-1_10

[64] Wang, S., Hu, B., Guan, J., Zhang, K., Shi, T.: MILP-aided Method of Searching Division Property Using Three Subsets and Applications. In: Galbraith, S.D., Moriai, S. (eds.) Advances in Cryptology - ASIACRYPT 2019 - 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8-12, 2019, Proceedings, Part III. Lecture Notes in Computer Science, vol. 11923, pp. 398–427. Springer (2019), https://doi.org/10.1007/978-3-030-34618-8_14

[65] Watanabe, D., Ideguchi, K., Kitahara, J., Muto, K., Furuichi, H., Kaneko, T.: Enocoro-80: A Hardware Oriented Stream Cipher. In: Second International Workshop on Advances in Information Security (2008)

[66] Watanabe, D., Owada, T., Okamoto, K., Igarashi, Y., Kaneko, T.: Update on Enocoro stream cipher. In: Proceedings of the International Symposium on Information Theory and its Applications, ISITA 2010, 17-20 October 2010, Taichung, Taiwan. pp. 778–783. IEEE (2010), https://doi.org/10.1109/ISITA.2010.5649627

[67] Ye, C., Tian, T.: Revisit Division Property Based Cube Attacks: Key-Recovery or Distinguishing Attacks? IACR Trans. Symmetric Cryptol. 2019(3), 81–102 (2019), https://doi.org/10.13154/tosc.v2019.i3.81-102

[68] Zhang, B., Li, Z., Feng, D., Lin, D.: Near Collision Attack on the Grain v1 Stream Cipher. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8424, pp. 518–538. Springer (2013), https://doi.org/10.1007/978-3-662-43933-3_27

[69] Zhang, B., Xu, C., Meier, W.: Fast Near Collision Attack on the Grain v1 Stream Cipher. In: Nielsen, J.B., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2018 - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29 - May 3, 2018 Proceedings, Part II. Lecture Notes in Computer Science, vol. 10821, pp. 771–802. Springer (2018), https://doi.org/10.1007/978-3-319-78375-8_25

[70] デジタル庁・総務省・経済産業省: 電子政府における調達のために参照すべき暗号のリスト（CRYPTREC 暗号リスト）（文書番号：CRYPTREC LS-0001-2022）(2023), https://www.cryptrec.go.jp/list/cryptrec-ls-0001-2022.pdf

[71] 伊藤竜馬:「CRYPTREC 暗号技術ガイドライン（軽量暗号）」掲載の暗号方式に関する安全性評価の動向調査（文書番号：CRYPTREC EX-3101-2021）(2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf

[72] 株式会社日立製作所: 擬似乱数生成器 Enocoro, https://www.hitachi.co.jp/rd/yrl/crypto/enocoro/（2023-10-04 閲覧）

[73] 五十嵐保隆, 岡本和人, 金子敏信: 関連鍵攻撃による Enocoro-128v1.1 の弱鍵復元の検討 (II). In: 電子情報通信学会総合大会講演論文集 (2010)

[74] 五十嵐保隆, 岡本和人, 金子敏信: 関連鍵攻撃による Enocoro の弱鍵復元の検討. pp. 275–280 (2010)

[75] 三上修吾, 渡辺大: ストリーム暗号 Enocoro-128v2 のソフトウェアおよびハードウェア実装と評価. In: コンピュータセキュリティシンポジウム 2012 論文集. pp. 742–748 (2012)

[76] 芝山直喜, 五十嵐保隆, 金子敏信: ストリーム暗号 Enocoro-128v2 の高階差分特性. In: 暗号と情報セキュリティシンポジウム, SCIS2021, 1B1-4 (2021)

[77] 船引悠生, 藤堂洋介, 五十部孝典, 森井昌克: Enocoro-128v2 の Cube 攻撃に対する安全性評価. In: 暗号と情報セキュリティシンポジウム, SCIS2019, 2B1-1 (2019)

4.3 ハッシュ関数

本節では軽量ハッシュ関数について記載する。ここで取り上げるアルゴリズムは、軽量ハッシュ関数として主要国際会議等に採録実績のある Keccak、PHOTON、QUARK、SPONGENT を調査対象とする。ただし、軽量という観点から、Keccak は SHA-3 として選定されたフルスペックのものでは無く、置換関数のビット幅が小さいもののみを対象とする。 また、PHOTON、SPONGENT、Lesamnta-LW が軽量ハッシュ関数に関係する ISO/IEC（ISO/IEC 29192-5）[17] で規格化されている状況を鑑み、本稿では 2016 年度版ガイドライン [9] に掲載されていない Lesamnta-LW を新たな調査対象とし、その調査結果をまとめる。

各アルゴリズムの安全性解析状況については、4.1 節と同様、2021 年度に公開された CRYPTREC 外部評価報告書 [42] に基づき、2021 年 9 月時点の状況を記載している。なお、新たに調査対象として追加した Lesamnta-LW について、文献 [42] では安全性解析状況だけでなく、仕様等（設計者、発表年、仕様参照先、特徴、主な実装評価結果、標準化状況）もまとめられているため、文献 [42] の記載内容に従って調査結果をまとめた。 また、実装評価結果は基本的に提案論文から抽出しており、それぞれが同一環境で評価されたものではないことに注意されたい。

<table>
<tr><td>技術分野</td><td>ハッシュ関数</td></tr>
<tr><td>名称</td><td>Keccak</td></tr>
<tr><td>設計者</td><td>Guido Bertoni[1], Joan Daemen[1], Michaël Peeters[2], Gilles Van Assche[1]
(1: STMicroelectronics/Switzerland, 2: NXP Semiconductors/Belgium)</td></tr>
<tr><td>発表年</td><td>2008（NIST SHA-3 Competition）</td></tr>
<tr><td>仕様参照先</td><td>設計者ウェブサイト [10]</td></tr>
<tr><td>特徴</td><td>Keccak はスポンジ構造から成り、内部パラメータにより様々なバリエーションを構成できる。置換関数は 7 種類が定義されており、それぞれ Keccak-$f[b]$ ($b \in$ 25、50、100、200、400、800、1600) と表される。ここでは、軽量暗号の観点から、Keccak-$f[100]$、Keccak-$f[200]$、Keccak-$f[400]$ を利用した方式について掲載する。
<table>
<tr><th>Keccak-$f[b]$</th><th>n</th><th>r</th><th>r'</th><th>構成段数 [段]</th></tr>
<tr><td>Keccak-$f[100]$</td><td>80</td><td>20</td><td>20</td><td>16</td></tr>
<tr><td>Keccak-$f[200]$</td><td>64</td><td>72</td><td>72</td><td>18</td></tr>
<tr><td>Keccak-$f[400]$</td><td>128</td><td>144</td><td>144</td><td>20</td></tr>
</table>
$*$ n: 出力長、r: 入力ブロック長、r': 出力ブロック長</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、SHA-3 として標準化された方式も含め、様々な解析論文 [5, 7, 8, 11, 12, 14, 19, 21, 22, 23, 24, 25, 26, 28, 29, 32, 33, 34, 35, 36] が発表されているが、仕様段数においてハッシュ関数の安全性基準を脅かす攻撃は発表されていない。
最良の衝突攻撃は、2021 年に提案された Boissier ら [5] による代数攻撃であり、2 段に簡略化した Keccak-$f[200]$ と 2 段に簡略化した Keccak-$f[400]$ に対して効率的に衝突計算を実行できる。
また、最良の原像攻撃は、2017 年に提案された Li ら [22] による攻撃であり、3 段に簡略化した Keccak-$f[400]$ に対して効率的に原像計算を実行できる。
最良の識別攻撃は、2011 年に Boura ら [8] によって提案されたゼロサム攻撃である。最大 24 段までの Keccak-f に対してゼロサム識別子が構成可能であると報告されているが、提案者 [8] が述べているように、この攻撃はハッシュ関数の安全性を脅かすものではない。その他、積分攻撃 [8, 36]、リバウンド攻撃 [11]、ブーメラン攻撃 [36]、差分攻撃 [26] などが報告されているが、これらの攻撃もまたハッシュ関数の安全性を脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装 [20]（130nm process）
<table>
<tr><th></th><th>Area [GE]</th><th>Latency [cycles/block]</th><th>Throughput [kbps]</th></tr>
<tr><td>Keccak-$f[100]$</td><td>1,250</td><td>800</td><td>2.5</td></tr>
<tr><td>Keccak-$f[200]$</td><td>2,520</td><td>900</td><td>8.0</td></tr>
<tr><td>Keccak-$f[400]$</td><td>5,090</td><td>1,000</td><td>14.4</td></tr>
</table></td></tr>
<tr><td>標準化状況</td><td>Keccak-$f[1600]$ を利用した方式は SHA-3 (FIPS 202 [27]) に採用されている。</td></tr>
<tr><td>利用実績等</td><td>SHA-3 としては多くのアプリケーションで導入されつつある。
https://csrc.nist.gov/groups/STM/cavp/documents/sha3/sha3val.html
https://www.3gpp.org/DynaReport/35-series.html
（いずれも 2023-10-04 閲覧）</td></tr>
<tr><td>オープンソース</td><td>https://keccak.team/archives.html
https://github.com/gvanas/KeccakCodePackage
（いずれも 2023-10-04 閲覧）</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ハッシュ関数</td></tr>
<tr><td>名称</td><td>PHOTON</td></tr>
<tr><td>設計者</td><td>Jian Guo[1], Thomas Peyrin[2], Axel Poschmann[2]
(1: Institute for Infocomm Research/Singapore, 2: Nanyang Technological University/Singapore)</td></tr>
<tr><td>発表年</td><td>2011（CRYPTO 2011 [13]）</td></tr>
<tr><td>仕様参照先</td><td>CRYPTO 2011 [13]</td></tr>
<tr><td>特徴</td><td>PHOTON はスポンジ構造から成り、内部パラメータにより様々なバリエーションを構成できる。ISO/IEC 29192 では 5 種のバリエーションが示されている（下表）。
使用する暗号学的置換は AES と似た構成であり、AddConstants、SubCells、ShiftRows、MixColumnsSerial の 4 ステップを 12 ラウンド繰り返す。SubCells での変換には PRESENT の S-box を利用する。
<table>
<tr><th>PHOTON-$n/r/r'$</th><th>n</th><th>r</th><th>r'</th><th>構成段数 [段]</th></tr>
<tr><td>PHOTON-80/20/16</td><td>80</td><td>20</td><td>16</td><td>12</td></tr>
<tr><td>PHOTON-128/16/16</td><td>128</td><td>16</td><td>16</td><td>12</td></tr>
<tr><td>PHOTON-160/36/36</td><td>160</td><td>36</td><td>36</td><td>12</td></tr>
<tr><td>PHOTON-224/32/32</td><td>224</td><td>32</td><td>32</td><td>12</td></tr>
<tr><td>PHOTON-256/32/32</td><td>256</td><td>32</td><td>32</td><td>12</td></tr>
</table>
∗ n: 出力長、r: 入力ブロック長、r': 出力ブロック長</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [18, 37, 38] が発表されているが、仕様段数においてハッシュ関数の安全性基準を脅かす攻撃は発表されていない。
最良の攻撃は、2017 年に提案された Wang ら [37, 38] によって提案されたゼロサム攻撃であり、11 段に簡略化した PHOTON-80 の暗号学的置換とフルスペックの PHOTON-128/160/224 の暗号学的置換に対して、それぞれ効率的に識別攻撃を実行できる。なお、PHOTON に対する識別攻撃は原則的にハッシュ関数の必須安全性基準（原像計算困難性、第 2 原像計算困難性、衝突困難性）を脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装 [13]（180nm process）
<table>
<tr><th></th><th>Area [GE]</th><th>Latency [cycles/block]</th><th>Throughput [kbps]</th></tr>
<tr><td>PHOTON-80/20/16</td><td>865/1,168</td><td>708/132</td><td>2.82/15.15</td></tr>
<tr><td>PHOTON-128/16/16</td><td>1,122/1,708</td><td>996/156</td><td>1.61/10.26</td></tr>
<tr><td>PHOTON-160/36/36</td><td>1,396/2,117</td><td>1,332/180</td><td>2.70/20.00</td></tr>
<tr><td>PHOTON-224/32/32</td><td>1,735/2,786</td><td>1,716/204</td><td>1.86/15.69</td></tr>
<tr><td>PHOTON-256/32/32</td><td>2,177/4,362</td><td>996/156</td><td>3.21/20.51</td></tr>
</table>
ソフトウェア実装 [13] (Intel Core i7 @1.6GHz)
<table>
<tr><th></th><th>32-bit optimized implementation [cycles/byte]</th></tr>
<tr><td>PHOTON-80/20/16</td><td>95</td></tr>
<tr><td>PHOTON-128/16/16</td><td>156</td></tr>
<tr><td>PHOTON-160/36/36</td><td>116</td></tr>
<tr><td>PHOTON-224/32/32</td><td>227</td></tr>
<tr><td>PHOTON-256/32/32</td><td>135</td></tr>
</table></td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-5 [17]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ハッシュ関数</td></tr>
<tr><td>名称</td><td>QUARK</td></tr>
<tr><td>設計者</td><td>Jean-Philippe Aumasson[1], Luca Henzen[2], Willi Meier[3], Maria Naya-Plasencia[3]
(1: Nagravision SA/Switzerland, 2: ETH Zurich/Switzerland, 3: FHNW/Switzerland)</td></tr>
<tr><td>発表年</td><td>2010（CHES 2010 [3]）</td></tr>
<tr><td>仕様参照先</td><td>CHES 2010 [3]、設計者 Web ページ [2]</td></tr>
<tr><td>特徴</td><td>QUARK はスポンジ構造から成り、内部パラメータにより様々なバリエーションを構成できる。パラメータの違いにより、U-QUARK、D-QUARK、S-QUARK の 3 種類が示されている。使用する暗号学的置換はストリーム暗号 Grain とブロック暗号 KATAN の利点を組み合わせた構成となっている。ラウンド数はそれぞれ 544、704、1024 である。
<table>
<tr><th></th><th>n</th><th>r</th><th>r'</th><th>構成段数 [段]</th></tr>
<tr><td>U-QUARK</td><td>128</td><td>8</td><td>8</td><td>544</td></tr>
<tr><td>D-QUARK</td><td>160</td><td>16</td><td>16</td><td>704</td></tr>
<tr><td>S-QUARK</td><td>224</td><td>32</td><td>32</td><td>1024</td></tr>
</table>
$*$ n: 出力長、r: 入力ブロック長、r': 出力ブロック長</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [39, 41] が発表されているが、仕様段数においてハッシュ関数の安全性基準を脅かす攻撃は発表されていない。
最良の攻撃は、2018 年に提案された Yang ら [39] によって提案された条件付き差分攻撃であり、155、166、259 段に簡略化した U/D/S-QUARK に対して、それぞれ効率的に識別攻撃を実行できる。なお、QUARK に対する識別攻撃は原則的にハッシュ関数の必須安全性基準（原像計算困難性、第 2 原像計算困難性、衝突困難性）を脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装 [3]（180nm process）
<table>
<tr><th></th><th>Area [GE]</th><th>Latency [cycles/block]</th><th>Throughput [kbps]</th></tr>
<tr><td>U-QUARK</td><td>1,379/2,392</td><td>544/68</td><td>1.47/11.76</td></tr>
<tr><td>D-QUARK</td><td>1,702/2,819</td><td>704/88</td><td>2.27/18.18</td></tr>
<tr><td>S-QUARK</td><td>2,296/4,640</td><td>1,024/64</td><td>3.13/50.00</td></tr>
</table></td></tr>
<tr><td>オープンソース</td><td><code>https://aumasson.jp/quark/</code>（2023-10-04 閲覧）
<code>https://github.com/veorq/Quark</code>（2023-10-04 閲覧）</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ハッシュ関数</td></tr>
<tr><td>名称</td><td>SPONGENT</td></tr>
<tr><td>設計者</td><td>Andrey Bogdanov[1], Miroslav Knežević[2], Gregor Leander[3], Deniz Toz[1], Kerem Varıcı[1], Ingrid Verbauwhede[1]
(1: KU Leuven/Belgium, 2: NXP Semiconductors/Belgium, 3: Technical University of Denmark/Denmark)</td></tr>
<tr><td>発表年</td><td>2011（CHES 2011 [4]）</td></tr>
<tr><td>仕様参照先</td><td>CHES 2011 [4]</td></tr>
<tr><td>特徴</td><td>SPONGENT は PRESENT タイプの暗号学的置換を用いたスポンジ構造から成り、内部パラメータにより様々なバリエーションを構成できる。提案者により 13 種のバリエーションが示されており、そのうち 5 種が ISO/IEC 29192-5 として標準化されている（表中の*印）。
<table>
<tr><th>SPONGENT-$n/c/r$</th><th>n</th><th>c</th><th>r</th><th>構成段数 [段]</th></tr>
<tr><td>SPONGENT-88/80/8*</td><td>88</td><td>80</td><td>8</td><td>45</td></tr>
<tr><td>SPONGENT-88/176/88</td><td>88</td><td>176</td><td>88</td><td>135</td></tr>
<tr><td>SPONGENT-128/128/8*</td><td>128</td><td>128</td><td>8</td><td>70</td></tr>
<tr><td>SPONGENT-128/256/128</td><td>128</td><td>256</td><td>128</td><td>195</td></tr>
<tr><td>SPONGENT-160/160/16*</td><td>160</td><td>160</td><td>16</td><td>90</td></tr>
<tr><td>SPONGENT-160/160/80</td><td>160</td><td>160</td><td>80</td><td>120</td></tr>
<tr><td>SPONGENT-160/320/160</td><td>160</td><td>320</td><td>160</td><td>240</td></tr>
<tr><td>SPONGENT-224/224/16*</td><td>224</td><td>224</td><td>16</td><td>120</td></tr>
<tr><td>SPONGENT-224/224/112</td><td>224</td><td>224</td><td>112</td><td>170</td></tr>
<tr><td>SPONGENT-224/448/224</td><td>224</td><td>448</td><td>224</td><td>340</td></tr>
<tr><td>SPONGENT-256/256/16*</td><td>256</td><td>256</td><td>16</td><td>140</td></tr>
<tr><td>SPONGENT-256/256/128</td><td>256</td><td>256</td><td>128</td><td>195</td></tr>
<tr><td>SPONGENT-256/512/256</td><td>256</td><td>512</td><td>256</td><td>385</td></tr>
</table>
∗ n: 出力長、c: capacity、r: rate(入力ブロック長)</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、いくつかの解析論文 [1, 40] が発表されているが、仕様段数においてハッシュ関数の安全性基準を脅かす攻撃は発表されていない。
最良の攻撃は、2017 年に提案された Zhang ら [40] による切り詰め差分攻撃であり、簡略化した全バリエーションの SPONGENT の暗号学的置換に対して効率的に識別攻撃が実行できる。なお、SPONGENT に対する識別攻撃は原則的にハッシュ関数の必須安全性基準（原像計算困難性、第 2 原像計算困難性、衝突困難性）を脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装 [4]（130nm process）
<table>
<tr><th></th><th>Area [GE]</th><th>Latency [cycles/block]</th><th>Throughput [kbps]</th></tr>
<tr><td>SPONGENT-88/80/8</td><td>738/1,127</td><td>990/45</td><td>0.81/17.78</td></tr>
<tr><td>SPONGENT-128/128/8</td><td>1,060/1,687</td><td>2,380/70</td><td>0.34/11.43</td></tr>
<tr><td>SPONGENT-160/160/16</td><td>1,329/2,190</td><td>3,960/90</td><td>0.40/17.78</td></tr>
<tr><td>SPONGENT-224/224/16</td><td>1,728/2,903</td><td>7,200/120</td><td>0.22/13.33</td></tr>
<tr><td>SPONGENT-256/256/16</td><td>1,950/3,281</td><td>9,520/140</td><td>0.17/11.43</td></tr>
</table></td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-5 [17]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ハッシュ関数</td></tr>
<tr><td>名称</td><td>Lesamnta-LW</td></tr>
<tr><td>設計者</td><td>Shoichi Hirose[1], Kota Ideguchi[2], Hidenori Kuwakado[3], Toru Owada[2], Bart Preneel[4], Hirotaka Yoshida[2,4]
(1: University of Fukui/Japan, 2: Hitachi, Ltd./Japan, 3: Kobe University/Japan, 4: KU Leuven/Belgium)</td></tr>
<tr><td>発表年</td><td>2010（ICISC 2010 [15]）</td></tr>
<tr><td>仕様参照先</td><td>ICISC 2010 [15]</td></tr>
<tr><td>特徴</td><td>Lesamnta-LW は LW1 モードと呼ばれるドメイン拡張型の Merkle-Damgård 構造から成り、その基礎となるコンポーネントは AES ベースのブロック暗号（Lesamnta-LW-BC）を利用する。出力長は 256 ビットであり、原像攻撃や衝突攻撃に対して 2^{120} のセキュリティレベルを有するよう設計されている。なお、Lesamnta-LW は SHA-3 competition に応募された Lesamnta の軽量版として提案された。
Lesamnta-LW-BC は 4-branch type-1 一般化 Feistel network 型のブロック暗号であり、仕様段数は 64 段、ブロックサイズは 256 ビット、秘密鍵サイズは 128 ビット、ラウンド関数は AES のコンポーネントである MixColumns と SubBytes を使用する。</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [6, 16, 30, 31] が発表されているが、仕様段数においてハッシュ関数の安全性基準を脅かす攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2021 年に提案された Shiba ら [31] による積分攻撃であり、20 段に簡略化した Lesamnta-LW-BC に対して、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。また、2020 年に提案された Hirose ら [16] による差分攻撃では、29 段に簡略化した Lesamnta-LW-BC に対して、効率的に識別攻撃が実行できる。既知鍵設定における最良の攻撃は、2021 年に提案された Shiba ら [31] によるゼロサム攻撃であり、47 段に簡略化した Lesamnta-LW-BC に対して、効率的に識別攻撃が実行できる。なお、Lesamnta-LW-BC に対する識別攻撃は原則的にハッシュ関数の必須安全性基準（原像計算困難性、第 2 原像計算困難性、衝突困難性）を脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装 [15] (90nm Logic Process)
<table>
<tr><td>lgorithm</td><td>Area [GE]</td><td>Clock [MHz]</td><td>Throughput@30MHz [Mbit/s]</td></tr>
<tr><td>Lesamnta-LW</td><td>8,240</td><td>188.3</td><td>20.00</td></tr>
</table>
ソフトウェア実装評価結果 [15]
<table>
<tr><td>Algorithm</td><td>RAM [byte]</td><td>Cycles/byte</td><td>Platform</td></tr>
<tr><td>Lesamnta-LW</td><td>50</td><td>1,650.9</td><td>Renesas H8 (8-bit CPU)</td></tr>
<tr><td>Lesamnta-LW</td><td>–</td><td>39.5</td><td>Intel Core i5 (32-bit CPU)</td></tr>
</table></td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-5 [17]</td></tr>
<tr><td>オープンソース</td><td><code>https://github.com/kuwakado/Lesamnta-LW</code>（2023-10-04 閲覧）</td></tr>
</table>

参考文献

[1] Abdelraheem, M.A.: Estimating the Probabilities of Low-Weight Differential and Linear Approximations on PRESENT-Like Ciphers. In: Kwon, T., Lee, M., Kwon, D. (eds.) Information Security and Cryptology - ICISC 2012 - 15th International Conference, Seoul, Korea, November 28-30, 2012, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7839, pp. 368–382. Springer (2012), https://doi.org/10.1007/978-3-642-37682-5_26

[2] Aumasson, J., Henzen, L., Meier, W., Naya-Plasencia, M.: Quark: a lightweight hash, https://www.aumasson.jp/quark/（2023-10-04 閲覧）

[3] Aumasson, J., Henzen, L., Meier, W., Naya-Plasencia, M.: Quark: A Lightweight Hash. In: Mangard, S., Standaert, F. (eds.) Cryptographic Hardware and Embedded Systems, CHES 2010, 12th International Workshop, Santa Barbara, CA, USA, August 17-20, 2010. Proceedings. Lecture Notes in Computer Science, vol. 6225, pp. 1–15. Springer (2010), https://doi.org/10.1007/978-3-642-15031-9_1

[4] Bogdanov, A., Knezevic, M., Leander, G., Toz, D., Varici, K., Verbauwhede, I.: SPONGENT: A Lightweight Hash Function. In: Preneel, B., Takagi, T. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6917, pp. 312–325. Springer (2011), https://doi.org/10.1007/978-3-642-23951-9_21

[5] Boissier, R.H., Noûs, C., Rotella, Y.: Algebraic Collision Attacks on Keccak. IACR Trans. Symmetric Cryptol. 2021(1), 239–268 (2021), https://doi.org/10.46586/tosc.v2021.i1.239-268

[6] Bouillaguet, C., Dunkelman, O., Leurent, G., Fouque, P.: Attacks on Hash Functions Based on Generalized Feistel: Application to Reduced-Round *Lesamnta* and *SHAvite-3*$_{512}$. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) Selected Areas in Cryptography - 17th International Workshop, SAC 2010, Waterloo, Ontario, Canada, August 12-13, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6544, pp. 18–35. Springer (2010), https://doi.org/10.1007/978-3-642-19574-7_2

[7] Boura, C., Canteaut, A.: Zero-sum distinguishers for iterated permutations and application to keccak-f and hamsi-256. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) Selected Areas in Cryptography - 17th International Workshop, SAC 2010, Waterloo, Ontario, Canada, August 12-13, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6544, pp. 1–17. Springer (2010), https://doi.org/10.1007/978-3-642-19574-7_1

[8] Boura, C., Canteaut, A., Cannière, C.D.: Higher-order differential properties of keccak and *Luffa*. In: Joux, A. (ed.) Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6733, pp. 252–269. Springer (2011), https://doi.org/10.1007/978-3-642-21702-9_15

[9] CRYPTREC 軽量暗号ワーキンググループ: CRYPTREC 暗号技術ガイドライン（軽量暗号）（文書番号：CRYPTREC GL-2003-2016JP）(2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf

[10] Daemen, J., Hoffert, S., Peeters, M., Assche, G.V., Keer, R.V., Mella, S.: Team Keccak: Xoodyak, https://keccak.team/keccak.html

[11] Duc, A., Guo, J., Peyrin, T., Wei, L.: Unaligned rebound attack: Application to keccak. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. Lecture Notes in Computer Science, vol. 7549, pp. 402–421. Springer (2012),

https://doi.org/10.1007/978-3-642-34047-5_23

[12] Guo, J., Liao, G., Liu, G., Liu, M., Qiao, K., Song, L.: Practical Collision Attacks against Round-Reduced SHA-3. J. Cryptol. 33(1), 228–270 (2020), https://doi.org/10.1007/s00145-019-09313-3

[13] Guo, J., Peyrin, T., Poschmann, A.: The PHOTON Family of Lightweight Hash Functions. In: Rogaway, P. (ed.) Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6841, pp. 222–239. Springer (2011), https://doi.org/10.1007/978-3-642-22792-9_13

[14] He, L., Lin, X., Yu, H.: Improved Preimage Attacks on 4-Round Keccak-224/256. IACR Trans. Symmetric Cryptol. 2021(1), 217–238 (2021), https://doi.org/10.46586/tosc.v2021.i1.217-238

[15] Hirose, S., Ideguchi, K., Kuwakado, H., Owada, T., Preneel, B., Yoshida, H.: A Lightweight 256-Bit Hash Function for Hardware and Low-End Devices: Lesamnta-LW. In: Rhee, K.H., Nyang, D. (eds.) Information Security and Cryptology - ICISC 2010 - 13th International Conference, Seoul, Korea, December 1-3, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6829, pp. 151–168. Springer (2010), https://doi.org/10.1007/978-3-642-24209-0_10

[16] Hirose, S., Sasaki, Y., Yoshida, H.: Lesamnta-LW Revisited: Improved Security Analysis of Primitive and New PRF Mode. In: Conti, M., Zhou, J., Casalicchio, E., Spognardi, A. (eds.) Applied Cryptography and Network Security - 18th International Conference, ACNS 2020, Rome, Italy, October 19-22, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12146, pp. 89–109. Springer (2020), https://doi.org/10.1007/978-3-030-57808-4_5

[17] ISO/IEC: Information security – Security techniques – Lightweight cryptography – Part 5: Hash-functions (ISO/IEC 29192-5:2016), https://www.iso.org/standard/67173.html

[18] Jean, J., Naya-Plasencia, M., Peyrin, T.: Improved Rebound Attack on the Finalist Grøstl. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. Lecture Notes in Computer Science, vol. 7549, pp. 110–126. Springer (2012), https://doi.org/10.1007/978-3-642-34047-5_7

[19] Jean, J., Nikolic, I.: Internal differential boomerangs: Practical analysis of the round-reduced keccak- f f permutation. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9054, pp. 537–556. Springer (2015), https://doi.org/10.1007/978-3-662-48116-5_26

[20] Kavun, E.B., Yalçin, T.: A Lightweight Implementation of Keccak Hash Function for Radio-Frequency Identification Applications. In: Radio Frequency Identification: Security and Privacy Issues - 6th International Workshop, RFIDSec 2010, Istanbul, Turkey, June 8-9, 2010, Revised Selected Papers. pp. 258–269 (2010), https://dx.doi.org/10.1007/978-3-642-16822-2_20

[21] Li, T., Sun, Y.: Preimage Attacks on Round-Reduced Keccak-224/256 via an Allocating Approach. In: Ishai, Y., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part III. Lecture Notes in Computer Science, vol. 11478, pp. 556–584. Springer (2019), https://doi.org/10.1007/978-3-030-17659-4_19

[22] Li, T., Sun, Y., Liao, M., Wang, D.: Preimage Attacks on the Round-reduced Keccak with Cross-linear Structures. IACR Trans. Symmetric Cryptol. 2017(4), 39–57 (2017), https://doi.org/10.13154/tosc.v2017.i4.39-57

[23] Li, Z., Dong, X., Bi, W., Jia, K., Wang, X., Meier, W.: New Conditional Cube Attack on Keccak Keyed Modes. IACR Trans. Symmetric Cryptol. 2019(2), 94–124 (2019), https://doi.org/10.13154/tosc.v2019.i2.94-124

[24] Lin, X., He, L., Yu, H.: Improved Preimage Attacks on 3-Round Keccak-224/256. IACR Trans. Symmetric Cryptol. 2021(3), 84–101 (2021), https://doi.org/10.46586/tosc.v2021.i3.84-101

[25] Liu, G., Qiu, W., Tu, Y.: New Techniques for Searching Differential Trails in Keccak. IACR Trans. Symmetric

Cryptol. 2019(4), 407–437 (2019), https://doi.org/10.13154/tosc.v2019.i4.407-437

[26] Mella, S., Daemen, J., Assche, G.V.: New techniques for trail bounds and application to differential trails in keccak. IACR Trans. Symmetric Cryptol. 2017(1), 329–357 (2017), https://doi.org/10.13154/tosc.v2017.i1.329-357

[27] National Institute of Standards and Technology: FIPS 202 – SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions, https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf

[28] Qiao, K., Song, L., Liu, M., Guo, J.: New Collision Attacks on Round-Reduced Keccak. In: Coron, J., Nielsen, J.B. (eds.) Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10212, pp. 216–243 (2017), https://doi.org/10.1007/978-3-319-56617-7_8

[29] Rajasree, M.S.: Cryptanalysis of Round-Reduced KECCAK Using Non-linear Structures. In: Hao, F., Ruj, S., Gupta, S.S. (eds.) Progress in Cryptology - INDOCRYPT 2019 - 20th International Conference on Cryptology in India, Hyderabad, India, December 15-18, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11898, pp. 175–192. Springer (2019), https://doi.org/10.1007/978-3-030-35423-7_9

[30] Sasaki, Y., Aoki, K.: Improved Integral Analysis on Tweaked Lesamnta. In: Kim, H. (ed.) Information Security and Cryptology - ICISC 2011 - 14th International Conference, Seoul, Korea, November 30 - December 2, 2011. Revised Selected Papers. Lecture Notes in Computer Science, vol. 7259, pp. 1–17. Springer (2011), https://doi.org/10.1007/978-3-642-31912-9_1

[31] Shiba, R., Sakamoto, K., Liu, F., Minematsu, K., Isobe, T.: Integral and Impossible Differential Attacks on the Reduced-Round Lesamnta-LW-BC. In: 暗号と情報セキュリティシンポジウム, SCIS2021, 1B1-2 (2021)

[32] Song, L., Guo, J.: Cube-Attack-Like Cryptanalysis of Round-Reduced Keccak Using MILP. IACR Trans. Symmetric Cryptol. 2018(3), 182–214 (2018), https://doi.org/10.13154/tosc.v2018.i3.182-214

[33] Song, L., Guo, J., Shi, D., Ling, S.: New MILP Modeling: Improved Conditional Cube Attacks on Keccak-Based Constructions. In: Peyrin, T., Galbraith, S.D. (eds.) Advances in Cryptology - ASIACRYPT 2018 - 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2-6, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 11273, pp. 65–95. Springer (2018), https://doi.org/10.1007/978-3-030-03329-3_3

[34] Song, L., Liao, G., Guo, J.: Non-full Sbox Linearization: Applications to Collision Attacks on Round-Reduced Keccak. In: Katz, J., Shacham, H. (eds.) Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10402, pp. 428–451. Springer (2017), https://doi.org/10.1007/978-3-319-63715-0_15

[35] Suryawanshi, S., Saha, D., Sachan, S.: New Results on the SymSum Distinguisher on Round-Reduced SHA3. In: Nitaj, A., Youssef, A.M. (eds.) Progress in Cryptology - AFRICACRYPT 2020 - 12th International Conference on Cryptology in Africa, Cairo, Egypt, July 20-22, 2020, Proceedings. Lecture Notes in Computer Science, vol. 12174, pp. 132–151. Springer (2020), https://doi.org/10.1007/978-3-030-51938-4_7

[36] Todo, Y.: Structural evaluation by generalized integral property. In: Oswald, E., Fischlin, M. (eds.) Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9056, pp. 287–314. Springer (2015), https://doi.org/10.1007/978-3-662-46800-5_12

[37] Wang, Q., Grassi, L., Rechberger, C.: Zero-Sum Partitions of PHOTON Permutations. IACR Cryptol. ePrint Arch. 2017, 1211 (2017), https://eprint.iacr.org/2017/1211

[38] Wang, Q., Grassi, L., Rechberger, C.: Zero-Sum Partitions of PHOTON Permutations. In: Smart, N.P. (ed.) Topics in Cryptology - CT-RSA 2018 - The Cryptographers' Track at the RSA Conference 2018, San Francisco, CA, USA, April 16-20, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10808, pp. 279–299. Springer

(2018), https://doi.org/10.1007/978-3-319-76953-0_15

[39] Yang, J., Liu, M., Lin, D., Wang, W.: Symbolic-Like Computation and Conditional Differential Cryptanalysis of QUARK. In: Inomata, A., Yasuda, K. (eds.) Advances in Information and Computer Security - 13th International Workshop on Security, IWSEC 2018, Sendai, Japan, September 3-5, 2018, Proceedings. Lecture Notes in Computer Science, vol. 11049, pp. 244–261. Springer (2018), https://doi.org/10.1007/978-3-319-97916-8_16

[40] Zhang, G., Liu, M.: A distinguisher on PRESENT-like permutations with application to SPONGENT. Sci. China Inf. Sci. 60(7), 72101 (2017), https://doi.org/10.1007/s11432-016-0165-6

[41] Zhang, K., Guan, J., Fei, X.: Improved conditional differential cryptanalysis. Secur. Commun. Networks 8(9), 1801–1811 (2015), https://doi.org/10.1002/sec.1144

[42] 伊藤竜馬:「CRYPTREC 暗号技術ガイドライン(軽量暗号)」掲載の暗号方式に関する安全性評価の動向調査(文書番号:CRYPTREC EX-3101-2021) (2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf

4.4 メッセージ認証コード

本節では、軽量なメッセージ認証コード (Message Authentication Code: MAC) を取り上げる。汎用的に用いられている MAC は、主にブロック暗号のモード (CMAC [38]) やハッシュ関数のモード (HMAC [19, 39]) である。CMAC や HMAC は、モード自体のオーバーヘッドがそれほど大きくないので、4.1 節に掲載されている軽量ブロック暗号や、4.3 節に掲載されている軽量ハッシュ関数と組み合せることで、軽量な MAC を構成することができる。HMAC はハッシュ関数を 2 回呼び出すため、処理するメッセージ長が非常に短い場合には、ブロック暗号ベースの CMAC を用いる方が効率的である可能性が高い。

ソフトウェア実装の性能に限れば、軽量暗号のベンチマークを行っている FELICS (Fair Evaluation of Lightweight Cryptographic Systems) プロジェクト [12] がブロック暗号やハッシュ関数を選択する参考になる。FELICS では、Atmel AVR ATmega128 (8-bit)、Texas Instruments MSP430F1611 (16-bit)、Arduino Due Arm Cortex-M3 (32-bit) 上で多数のアルゴリズム (ブロック暗号、ストリーム暗号、ハッシュ関数) を比較している。

ブロック暗号やハッシュ関数のモードではない、専用に設計された軽量 MAC はそれほど多く知られていないが、短いメッセージの処理に特化した擬似ランダム関数 SipHash [1, 2] は MAC として利用可能であり、多くの利用実績がある。ただし、SipHash は内部処理で 64-bit 加算などを利用しているので、「比較的ハイエンドの CPU 上で高速」という意味での軽量 MAC であり、8〜32-bit CPU での使用には適さない。

この他に、ローエンド CPU 向けの軽量 MAC として Chaskey [26, 28] がある。FELICS では Chaskey は軽量ブロック暗号に分類されており、多くの項目で最も優秀な成績を収めている。その一方で、8 ラウンド中 7 ラウンドについて鍵回復攻撃が可能であることが報告されており [20]、安全性の観点ではセキュリティマージンが小さい。この点を改善するために、ラウンド数を 8 から 12 に増やした Chaskey-12 [27] が提案されている。

本節では、2016 年度版ガイドライン [8] で掲載されている SipHash に加え、Chaskey、LightMAC、Tsudik's keymode が軽量 MAC に関係する ISO/IEC（ISO/IEC 29192-6）[15] で規格化されている状況を鑑み、これら 3 方式も新たな調査対象とし、その調査結果をまとめる。 なお、CMAC, HMAC については、本節では特に取り上げない。

SipHash の安全性解析状況については、4.1 節と同様、2021 年度に公開された CRYPTREC 外部評価報告書 [49] に基づき、2021 年 9 月時点の状況を記載している。なお、新たに調査対象として追加した Chaskey、LightMAC、Tsudik's keymode について、文献 [49] では安全性解析状況だけでなく、仕様等（設計者、発表年、仕様参照先、特徴、主な実装性能結果、標準化状況）もまとめられているため、文献 [49] の記載内容に従って調査結果をまとめた。また、Tsudik's keymode の安全性解析状況については、2022 年度に公開された CRYPTREC 外部評価報告書 [50] に基づき、2022 年 9 月現在の解析状況を記載している。

技術分野	メッセージ認証コード
名称	SipHash
設計者	Jean-Philippe Aumasson[1], Daniel J. Bernstein[2] (1: Kudelski Security/Switzerland, 2: University of Illinois at Chicago/USA)
発表年	2012（INDOCRYPT 2012 [2]）
仕様参照先	INDOCRYPT 2012 [2]、設計者ウェブサイト [1]
特徴	SipHash は、連想配列に用いるハッシュ関数として開発された鍵長 128 ビット、出力長 64 ビットの鍵付きハッシュ関数である。入力されるメッセージ長の上限は 2039 バイトであり、汎用のハッシュ関数に比べて短い。SipHash のアルゴリズムは c ラウンドの圧縮フェーズと d ラウンドの最終処理フェーズからなり、SipHash-c-d と表される。一般に利用されているのは、$c = 2, d = 4$ の SipHash-2-4 である。64 ビットワードを単位とし、算術加算、排他的論理和、巡回シフトを組み合わせたアルゴリズムであり、64 ビット演算をサポートする CPU 上で高速に動作する。また、アルゴリズム中でテーブル参照を行わないため、素直に実装してもキャッシュタイミング攻撃に対して安全である。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [11, 22, 44] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 SipHash-2-4 に対するに対する最良の鍵回復攻撃は、2014 年に提案された Dobraunig ら [11] による差分攻撃であり、差分特性確率 $2^{-236.3}$ の差分パスが発見されたが、鍵の総当たり攻撃の方がはるかに効率的であるので、この結果は SipHash の安全性を脅かすものではない。圧縮フェーズのみを簡略化した SipHash に対する最良の攻撃は、2019 年に提案された Xin ら [44] による差分攻撃であるが、この結果においても SipHash の安全性を脅かすものではない。
主な実装評価結果	提案論文 [2] によれば、SipHash のスループットは amd64 アーキテクチャ上で 1.5〜3.0 cycles/byte である。メッセージ長が短い場合には最終処理のオーバーヘッドが大きく、8 バイトのデータでは 10〜30 cycles/byte となる。
標準化状況	現時点では、(デジュールの) 標準化には提案されていない。しかし、多くのオープンソースライブラリに実装されており、デファクト標準の地位を固めつつある。
利用実績等	オープンソース、特に軽量プログラミング言語（Perl, Python, Ruby 等）の連想配列で用いるハッシュ関数として広く採用されている。この他にも、[1] によれば Wireguard, Bloomberg, OpenBSD, Shardmap, SoundHound, FreeBSD, Hashable, Rubinius, JRuby, Redis, OpenDNS, Rust, Sodium が SipHash を採用している。
オープンソース	前項を参照のこと。

<table>
<tr><td>技術分野</td><td>メッセージ認証コード</td></tr>
<tr><td>名称</td><td>Chaskey</td></tr>
<tr><td>設計者</td><td>Nicky Mouha[1], Bart Mennink[1], Anthony Van Herrewege[1], Dai Watanabe[2], Bart Preneel[1], Ingrid Verbauwhede[1]
(1: KU Leuven/Belgium, 2: Hitachi, Ltd./Japan)</td></tr>
<tr><td>発表年</td><td>2014（SAC 2014 [29])、2015（Cryptology ePrint Archive [27]）</td></tr>
<tr><td>仕様参照先</td><td>SAC 2014 [29]、Cryptology ePrint Archive [27]</td></tr>
<tr><td>特徴</td><td>Chaskey は、算術加算、排他的論理和、巡回シフトの組み合わせで構成される暗号学的置換を用いたメッセージ認証コードアルゴリズムである。暗号学的置換の仕様段数は 8 段又は 12 段、鍵長とブロックサイズは 128 ビット、タグ長は 64 ビット以上が推奨されている。なお、ISO/IEC 29192-6 [15] では 12 段の Chaskey が規格化されている。32 ビットワードを単位として演算が実行されることから、32 ビット演算をサポートするマイクロコントローラー上で効率的に動作する。また、全ての演算にかかる実行時間が一定であり、サイクル数がメッセージ長のみに依存するため、Chaskey はタイミング攻撃に対して安全である。</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、いくつかの解析論文 [3, 6, 18, 20, 25, 45] が発表されている。
単一鍵設定における最良の鍵回復攻撃は、2021 年に提案された Broll ら [6] による差分線形攻撃であり、7.5 段に簡略化した Chaskey に対して、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。また、Chaskey については弱鍵が存在し、関連鍵設定においてその弱鍵を使用している場合、仕様段数において効率的に鍵回復攻撃と偽造攻撃が実行できる [18]。</td></tr>
<tr><td>主な実装評価結果</td><td>ソフトウェア実装評価結果 [29]
<table>
<tr><th>Data [byte]</th><th>ROM [byte]</th><th>Cycles/byte</th><th>Platform</th></tr>
<tr><td>16</td><td>414</td><td>21.8</td><td>Cortex-M0</td></tr>
<tr><td>16</td><td>1,308</td><td>21.3</td><td>Cortex-M0</td></tr>
<tr><td>128</td><td>414</td><td>16.9</td><td>Cortex-M0</td></tr>
<tr><td>128</td><td>1,308</td><td>18.3</td><td>Cortex-M0</td></tr>
<tr><td>16</td><td>402</td><td>16.1</td><td>Cortex-M3/M4</td></tr>
<tr><td>16</td><td>908</td><td>10.6</td><td>Cortex-M3/M4</td></tr>
<tr><td>128</td><td>402</td><td>11.2</td><td>Cortex-M3/M4</td></tr>
<tr><td>128</td><td>908</td><td>7.0</td><td>Cortex-M3/M4</td></tr>
</table>
なお、Chaskey-12 [27] は Chaskey [29] と比較すると 32-bit Arm Cortex-M microcontrollers で 15 % 低速であると言及されている。
その他、効率的なソフトウェア実装の結果が文献 [10] で報告されている。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-6 [15]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>メッセージ認証コード</td></tr>
<tr><td>名称</td><td>LightMAC</td></tr>
<tr><td>設計者</td><td>Atul Luykx[1,2], Bart Preneel[1,2], Elmar Tischhauser[3], Kan Yasuda[4]
(1: KU Leuven/Belgium, 2: iMinds/Belgium, 3: Technical University of Denmark/ Denmark, 4: NTT/Japan)</td></tr>
<tr><td>発表年</td><td>2016（FSE 2016 [23]）</td></tr>
<tr><td>仕様参照先</td><td>FSE 2016 [23]</td></tr>
<tr><td>特徴</td><td>LightMAC は、ブロック暗号を利用した暗号利用モードによるメッセージ認証コードアルゴリズムである。従来のメッセージ認証技術では、ブロック長の短い軽量ブロック暗号を利用した場合、大きなデータを処理すると安全性が低下してしまうという課題があったが、LightMAC ではブロック暗号に対して独特の繰り返し方法を用いることにより、この課題を解決した。これにより LightMAC は既存の軽量ブロック暗号の実装を有効活用しつつ必要な安全性を確保することができる（参考記事：NTT 持株会社ニュースリリース [51]）。</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [9, 30, 31, 43] が発表されている。
基礎となるブロック暗号として Simeck32/64 を利用した LightMAC に対し、3 種類の偽造攻撃が現実的な計算量で実行可能であることが報告されている [9, 43]。しかしながら、これらの攻撃は設計者 [23] が示す安全性上界のバウンドを脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ソフトウェア実装評価結果（Intel Core i7-6700 CPU） [23]
<table>
<tr><td rowspan="2">Underlying Block Cipher</td><td rowspan="2">Rate</td><td colspan="7">Message length [bytes]</td></tr>
<tr><td>128</td><td>256</td><td>512</td><td>1,024</td><td>2,048</td><td>4,096</td><td>8,192</td></tr>
<tr><td>PRESENT</td><td>1/2</td><td>25.50</td><td>23.67</td><td>22.75</td><td>22.32</td><td>22.08</td><td>21.97</td><td>21.92</td></tr>
<tr><td>PRESENT</td><td>2/3</td><td>25.70</td><td>21.21</td><td>20.17</td><td>19.03</td><td>18.09</td><td>17.80</td><td>17.80</td></tr>
<tr><td>PRESENT</td><td>7/8</td><td>20.31</td><td>18.34</td><td>14.65</td><td>13.48</td><td>–</td><td>–</td><td>–</td></tr>
<tr><td>AES</td><td>1/2</td><td>1.33</td><td>1.29</td><td>1.27</td><td>1.26</td><td>1.26</td><td>1.26</td><td>1.25</td></tr>
<tr><td>AES</td><td>2/3</td><td>1.37</td><td>1.31</td><td>1.12</td><td>1.04</td><td>0.95</td><td>0.95</td><td>0.92</td></tr>
<tr><td>AES</td><td>15/16</td><td>1.38</td><td>1.00</td><td>0.82</td><td>0.80</td><td>0.72</td><td>–</td><td>–</td></tr>
</table>
なお、数値は cycles/byte である。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-6 [15]</td></tr>
</table>

技術分野	メッセージ認証コード
名称	Tsudik's keymode
設計者	Gene Tsudik (University of Southern California/USA)
発表年	1992（ACM INFOCOM 1992 [40]）
仕様参照先	ACM INFOCOM 1992 [40]
特徴	Tsudik's keymode は、一方向性ハッシュ関数を用いた MAC であり、提案論文 [40] では MD4 を用いてアルゴリズムを紹介している。 $H : \{0,1\}^* \to \{0,1\}^n$ を出力長 n ビットのハッシュ関数とする。鍵長 k ビット、タグ長 t ビットの Tsudik's keymode $\mathsf{TKM} : \{0,1\}^k \times \{0,1\}^* \to \{0,1\}^t$ は、 1. $\mathsf{TKM}_K(M) = \lfloor H(K \parallel M) \rfloor_t$（secret prefix 方式） 2. $\mathsf{TKM}_K(M) = \lfloor H(M \parallel K) \rfloor_t$（secret suffix 方式） 3. $\mathsf{TKM}_{K,K'}(M) = \lfloor H(K \parallel M \parallel K') \rfloor_t$（ハイブリッド方式） の 3 種類が定義されている。なお、ISO/IEC 29192-6 [15] では secret prefix 方式のみ標準化されており、使用するハッシュ関数は ISO/IEC 29192-5 [16] で標準化されている PHOTON、SPONGENT、Lesamnta-LW の 3 方式が推奨されている。
安全性解析状況	ISO/IEC 29192-6 [15] において、国際標準方式である secret prefix 方式の安全性が述べられている。使用するハッシュ関数が衝突困難性を有すること、length-extension attack が実行できないことが要件として挙げられている。また、可変長入力のランダムオラクル [5] から強識別不可能性 [7, 24] を有するハッシュ関数であれば、Tsudik's keymode での使用に適していることが言及されている。推奨されているハッシュ関数の PHOTON と SPONGENT はこれに該当する。Lesamnta-LW は length-extension attack が実行可能な方式であるものの、設計者 [13] が Tsudik's keymode で使用した場合の擬似ランダム性を証明している。したがって、これら 3 方式は Tsudik's keymode での使用に適している。 Tsudik's keymode に対する第三者評価として、Preneel ら [17, 32, 34] による現実的な鍵回復攻撃と偽造攻撃が報告されている。この攻撃では length-extension attack が実行可能なハッシュ関数を使用した場合を想定しており、適切なハッシュ関数を使用することで攻撃を回避できる。その他、類似した方式に対するいくつかの解析結果 [4, 14, 21, 32, 33, 35, 36, 37, 41, 42, 46, 47, 48] が報告されている。これらの解析結果は Tsudik's keymode（特に、国際標準方式の secret prefix 方式）の安全性を脅かすものではない。
主な実装評価結果	Tsudik's keymode の実装性能は使用する一方向性ハッシュ関数に依存する。
標準化状況	ISO/IEC 29192-6 [15]

参考文献

[1] Aumasson, J.P.: SipHash: A Fast Short-input PRF, https://131002.net/siphash/（2023-10-04 閲覧）

[2] Aumasson, J.P., Bernstein, D.J.: SipHash: A Fast Short-Input PRF. In: Galbraith, S., Nandi, M. (eds.) Progress in Cryptology – INDOCRYPT 2012. Lecture Notes in Computer Science, vol. 7668, pp. 489–508. Springer-Verlag Berlin Heidelberg (2012)

[3] Beierle, C., Leander, G., Todo, Y.: Improved Differential-Linear Attacks with Applications to ARX Ciphers. In: Micciancio, D., Ristenpart, T. (eds.) Advances in Cryptology - CRYPTO 2020 - 40th Annual International Cryptology Conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17-21, 2020, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12172, pp. 329–358. Springer (2020), https://doi.org/10.1007/978-3-030-56877-1_12

[4] Bellare, M., Canetti, R., Krawczyk, H.: Pseudorandom Functions Revisited: The Cascade Construction and Its Concrete Security. In: 37th Annual Symposium on Foundations of Computer Science, FOCS '96, Burlington, Vermont, USA, 14-16 October, 1996. pp. 514–523. IEEE Computer Society (1996), https://doi.org/10.1109/SFCS.1996.548510

[5] Bellare, M., Rogaway, P.: Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. In: Denning, D.E., Pyle, R., Ganesan, R., Sandhu, R.S., Ashby, V. (eds.) CCS '93, Proceedings of the 1st ACM Conference on Computer and Communications Security, Fairfax, Virginia, USA, November 3-5, 1993. pp. 62–73. ACM (1993), https://doi.org/10.1145/168588.168596

[6] Broll, M., Canale, F., David, N., Flórez-Gutiérrez, A., Leander, G., Naya-Plasencia, M., Todo, Y.: Further Improving Differential-Linear Attacks: Applications to Chaskey and Serpent. IACR Cryptol. ePrint Arch. 2021, 820 (2021), https://eprint.iacr.org/2021/820

[7] Coron, J., Dodis, Y., Malinaud, C., Puniya, P.: Merkle-Damgård Revisited: How to Construct a Hash Function. In: Shoup, V. (ed.) Advances in Cryptology - CRYPTO 2005: 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18, 2005, Proceedings. Lecture Notes in Computer Science, vol. 3621, pp. 430–448. Springer (2005), https://doi.org/10.1007/11535218_26

[8] CRYPTREC 軽量暗号ワーキンググループ: CRYPTREC 暗号技術ガイドライン（軽量暗号）（文書番号：CRYPTREC GL-2003-2016JP）(2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf

[9] Darumaya, T.A., Susanti, B.H.: Forgery Attack on LightMAC Hash Function Scheme using SIMECK 32/64 Lightweight Block Cipher. IOP Conference Series: Materials Science and Engineering 453, 012014 (nov 2018), https://doi.org/10.1088/1757-899x/453/1/012014

[10] Dinu, D., Corre, Y.L., Khovratovich, D., Perrin, L., Großschädl, J., Biryukov, A.: Triathlon of lightweight block ciphers for the Internet of things. J. Cryptogr. Eng. 9(3), 283–302 (2019), https://doi.org/10.1007/s13389-018-0193-x

[11] Dobraunig, C., Mendel, F., Schläffer, M.: Differential Cryptanalysis of SipHash. In: Joux, A., Youssef, A.M. (eds.) Selected Areas in Cryptography - SAC 2014 - 21st International Conference, Montreal, QC, Canada, August 14-15, 2014, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8781, pp. 165–182. Springer (2014), https://doi.org/10.1007/978-3-319-13051-4_10

[12] FELICS: Fair Evaluation of Lightweight Cryptographic Systems, https://www.cryptolux.org/index.php/

[13] Hirose, S., Ideguchi, K., Kuwakado, H., Owada, T., Preneel, B., Yoshida, H.: An AES Based 256-bit Hash Function for Lightweight Applications: Lesamnta-LW. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. 95-A(1), 89–99 (2012), https://doi.org/10.1587/transfun.E95.A.89

[14] Hosoyamada, A., Sasaki, Y.: Cryptanalysis Against Symmetric-Key Schemes with Online Classical Queries and Offline Quantum Computations. In: Smart, N.P. (ed.) Topics in Cryptology - CT-RSA 2018 - The Cryptographers' Track at the RSA Conference 2018, San Francisco, CA, USA, April 16-20, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10808, pp. 198–218. Springer (2018), https://doi.org/10.1007/978-3-319-76953-0_11

[15] ISO/IEC: Information security – Lightweight cryptography – Part 6: Message authentication codes (MACs) (ISO/IEC 29192-6:2019), https://www.iso.org/standard/71116.html

[16] ISO/IEC: Information security – Security techniques – Lightweight cryptography – Part 5: Hash-functions (ISO/IEC 29192-5:2016), https://www.iso.org/standard/67173.html

[17] Koblitz, N., Menezes, A.: Another Look at Security Theorems for 1-Key Nested MACs. In: Koç, Ç.K. (ed.) Open Problems in Mathematics and Computational Science, pp. 69–89. Springer (2014), https://doi.org/10.1007/978-3-319-10683-0_4

[18] Kraleva, L., Ashur, T., Rijmen, V.: Rotational Cryptanalysis on MAC Algorithm Chaskey. In: Conti, M., Zhou, J., Casalicchio, E., Spognardi, A. (eds.) Applied Cryptography and Network Security - 18th International Conference, ACNS 2020, Rome, Italy, October 19-22, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12146, pp. 153–168. Springer (2020), https://doi.org/10.1007/978-3-030-57808-4_8

[19] Krawczyk, H., Bellare, M., Canetti, R.: HMAC: Keyed-Hashing for Message Authentication, Request For Comments, vol. RFC2104 (February), https://tools.ietf.org/html/rfc2104

[20] Leurent, G.: Improved Differential-Linear Cryptanalysis of 7-Round Chaskey with Partitioning. In: Fischlin, M., Coron, J. (eds.) Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9665, pp. 344–371. Springer (2016), https://doi.org/10.1007/978-3-662-49890-3_14

[21] Liu, F., Xie, T., Shen, C.: Breaking H^2-MAC Using Birthday Paradox. IACR Cryptol. ePrint Arch. p. 647 (2011), https://eprint.iacr.org/2011/647

[22] Liu, Y., Sun, S., Li, C.: Rotational Cryptanalysis from a Differential-Linear Perspective - Practical Distinguishers for Round-Reduced FRIET, Xoodoo, and Alzette. In: Canteaut, A., Standaert, F. (eds.) Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12696, pp. 741–770. Springer (2021), https://doi.org/10.1007/978-3-030-77870-5_26

[23] Luykx, A., Preneel, B., Tischhauser, E., Yasuda, K.: A MAC Mode for Lightweight Block Ciphers. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 43–59. Springer (2016), https://doi.org/10.1007/978-3-662-52993-5_3

[24] Maurer, U.M., Renner, R., Holenstein, C.: Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology. In: Naor, M. (ed.) Theory of Cryptography, First Theory of Cryptography Conference, TCC 2004, Cambridge, MA, USA, February 19-21, 2004, Proceedings. Lecture Notes in Computer Science, vol. 2951, pp. 21–39. Springer (2004), https://doi.org/10.1007/978-3-540-24638-1_2

[25] Mavromati, C.: Key-Recovery Attacks Against the MAC Algorithm Chaskey. In: Dunkelman, O., Keliher, L. (eds.) Selected Areas in Cryptography - SAC 2015 - 22nd International Conference, Sackville, NB, Canada, August 12-14, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9566, pp. 205–216. Springer (2015),

https://doi.org/10.1007/978-3-319-31301-6_12

[26] Mouha, N.: Chaskey, https://mouha.be/chaskey/（2023-10-04 閲覧）

[27] Mouha, N.: Chaskey: A MAC Algorithm for Microcontrollers – Status Update and Proposal of Chaskey-12 –, https://eprint.iacr.org/2015/1182

[28] Mouha, N., Mennink, B., Herrewege, A.V., Watanabe, D., Preneel, B., Verbauwhede, I.: Chaskey: An Efficient MAC Algorithm for 32-bit Microcontrollers. In: Joux, A., Youssef, A. (eds.) Selected Areas in Cryptography – SAC 2014. Lecture Notes in Computer Science, vol. 8781, pp. 306–323. Springer (2014)

[29] Mouha, N., Mennink, B., Herrewege, A.V., Watanabe, D., Preneel, B., Verbauwhede, I.: Chaskey: An Efficient MAC Algorithm for 32-bit Microcontrollers. In: Joux, A., Youssef, A.M. (eds.) Selected Areas in Cryptography - SAC 2014 - 21st International Conference, Montreal, QC, Canada, August 14-15, 2014, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8781, pp. 306–323. Springer (2014), https://doi.org/10.1007/978-3-319-13051-4_19

[30] Naito, Y.: Blockcipher-Based MACs: Beyond the Birthday Bound Without Message Length. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10626, pp. 446–470. Springer (2017), https://doi.org/10.1007/978-3-319-70700-6_16

[31] Naito, Y.: Improved Security Bound of LightMAC_Plus and Its Single-Key Variant. In: Smart, N.P. (ed.) Topics in Cryptology - CT-RSA 2018 - The Cryptographers' Track at the RSA Conference 2018, San Francisco, CA, USA, April 16-20, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10808, pp. 300–318. Springer (2018), https://doi.org/10.1007/978-3-319-76953-0_16

[32] Preneel, B., van Oorschot, P.C.: MDx-MAC and Building Fast MACs from Hash Functions. In: Coppersmith, D. (ed.) Advances in Cryptology - CRYPTO '95, 15th Annual International Cryptology Conference, Santa Barbara, California, USA, August 27-31, 1995, Proceedings. Lecture Notes in Computer Science, vol. 963, pp. 1–14. Springer (1995), https://doi.org/10.1007/3-540-44750-4_1

[33] Preneel, B., van Oorschot, P.C.: On the Security of Two MAC Algorithms. In: Maurer, U.M. (ed.) Advances in Cryptology - EUROCRYPT '96, International Conference on the Theory and Application of Cryptographic Techniques, Saragossa, Spain, May 12-16, 1996, Proceeding. Lecture Notes in Computer Science, vol. 1070, pp. 19–32. Springer (1996), https://doi.org/10.1007/3-540-68339-9_3

[34] Preneel, B., van Oorschot, P.C.: On the Security of Iterated Message Authentication Codes. IEEE Trans. Inf. Theory 45(1), 188–199 (1999), https://doi.org/10.1109/18.746787

[35] Qiao, S., Wang, W., Jia, K.: Distinguishing Attack on Secret Prefix MAC Instantiated with Reduced SHA-1. In: Lee, D.H., Hong, S. (eds.) Information, Security and Cryptology - ICISC 2009, 12th International Conference, Seoul, Korea, December 2-4, 2009, Revised Selected Papers. Lecture Notes in Computer Science, vol. 5984, pp. 349–361. Springer (2009), https://doi.org/10.1007/978-3-642-14423-3_23

[36] Sasaki, Y.: Cryptanalyses on a Merkle-Damgård Based MAC - Almost Universal Forgery and Distinguishing-H Attacks. In: Pointcheval, D., Johansson, T. (eds.) Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7237, pp. 411–427. Springer (2012), https://doi.org/10.1007/978-3-642-29011-4_25

[37] Sasaki, Y.: Cryptanalyses on a Merkle-Damgård Based MAC - Almost Universal Forgery and Distinguishing-H Attacks. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. 97-A(1), 167–176 (2014), https://doi.org/10.1587/transfun.E97.A.167

[38] of Standards, N.I., Technology: Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication. NIST Special Publication 800-38B (May 2005), https://csrc.nist.gov/publications/

nistpubs/800-38B/SP_800-38B.pdf

[39] of Standards, N.I., Technology: The Keyed-Hash Message Authentication Code (HMAC). Federal Information Processing Standards Publication FIPS 198-1 (July 2008), https://csrc.nist.gov/publications/fips/fips198-1/FIPS-198-1_final.pdf

[40] Tsudik, G.: Message Authentication with One-Way Hash Functions. In: Proceedings IEEE INFOCOM '92, The Conference on Computer Communications, Eleventh Annual Joint Conference of the IEEE Computer and Communications Societies, One World through Communications, Florence, Italy, May 4-8, 1992. pp. 2055–2059. IEEE Computer Society (1992), https://doi.org/10.1109/INFCOM.1992.263477

[41] Wang, G.: Distinguishing Attacks on LPMAC Based on the Full RIPEMD and Reduced-Step RIPEMD-{256, 320}. In: Lai, X., Yung, M., Lin, D. (eds.) Information Security and Cryptology - 6th International Conference, Inscrypt 2010, Shanghai, China, October 20-24, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6584, pp. 199–217. Springer (2010), https://doi.org/10.1007/978-3-642-21518-6_15

[42] Wang, X., Wang, W., Jia, K., Wang, M.: New Distinguishing Attack on MAC Using Secret-Prefix Method. In: Dunkelman, O. (ed.) Fast Software Encryption, 16th International Workshop, FSE 2009, Leuven, Belgium, February 22-25, 2009, Revised Selected Papers. Lecture Notes in Computer Science, vol. 5665, pp. 363–374. Springer (2009), https://doi.org/10.1007/978-3-642-03317-9_22

[43] Windarta, S., Ramli, K., Sudiana, D.: Security Evaluation of LIGHTMAC: Second Preimage Attack using Existential Forgery. In: 2020 1st International Conference on Information Technology, Advanced Mechanical and Electrical Engineering (ICITAMEE). pp. 265–269. IEEE (2020)

[44] Xin, W., Liu, Y., Sun, B., Li, C.: Improved Cryptanalysis on SipHash. In: Mu, Y., Deng, R.H., Huang, X. (eds.) Cryptology and Network Security - 18th International Conference, CANS 2019, Fuzhou, China, October 25-27, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11829, pp. 61–79. Springer (2019), https://doi.org/10.1007/978-3-030-31578-8_4

[45] Xu, Y., Wu, B., Lin, D.: Rotational-Linear Attack: A New Framework of Cryptanalysis on ARX Ciphers with Applications to Chaskey. In: Gao, D., Li, Q., Guan, X., Liao, X. (eds.) Information and Communications Security - 23rd International Conference, ICICS 2021, Chongqing, China, November 19-21, 2021, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12919, pp. 192–209. Springer (2021), https://doi.org/10.1007/978-3-030-88052-1_12

[46] Yasuda, K.: "Sandwich" Is Indeed Secure: How to Authenticate a Message with Just One Hashing. In: Pieprzyk, J., Ghodosi, H., Dawson, E. (eds.) Information Security and Privacy, 12th Australasian Conference, ACISP 2007, Townsville, Australia, July 2-4, 2007, Proceedings. Lecture Notes in Computer Science, vol. 4586, pp. 355–369. Springer (2007), https://doi.org/10.1007/978-3-540-73458-1_26

[47] Yasuda, K.: HMAC without the "Second" Key. In: Samarati, P., Yung, M., Martinelli, F., Ardagna, C.A. (eds.) Information Security, 12th International Conference, ISC 2009, Pisa, Italy, September 7-9, 2009. Proceedings. Lecture Notes in Computer Science, vol. 5735, pp. 443–458. Springer (2009), https://doi.org/10.1007/978-3-642-04474-8_35

[48] Yu, H., Wang, X.: Distinguishing Attack on the Secret-Prefix MAC Based on the 39-Step SHA-256. In: Boyd, C., Nieto, J.M.G. (eds.) Information Security and Privacy, 14th Australasian Conference, ACISP 2009, Brisbane, Australia, July 1-3, 2009, Proceedings. Lecture Notes in Computer Science, vol. 5594, pp. 185–201. Springer (2009), https://doi.org/10.1007/978-3-642-02620-1_13

[49] 伊藤竜馬:「CRYPTREC 暗号技術ガイドライン(軽量暗号)」掲載の暗号方式に関する安全性評価の動向調査(文書番号:CRYPTREC EX-3101-2021) (2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf

[50] 岩田哲: 軽量暗号の安全性に関する調査及び評価(Photon-Beetle,Sparkle,Tsudik's keymode)(文書番号:CRYPTREC EX-3201-2022) (2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3201-2022.pdf

[51] 日本電信電話株式会社: NTT 持株会社ニュースリリース – IoT 向けメッセージ認証技術 LightMAC が ISO 標準に採

択 –, https://journal.ntt.co.jp/article/1004（2023-10-04 閲覧）

4.5 認証暗号

本節では、CAESAR コンペティション [10] において提案された方式のうち、軽量性を謳い、かつ安全性の観点で問題が見つかっていない方式を中心に調査結果をまとめる（2021 年 9 月現在)。 加えて、Grain-128A が軽量認証暗号に関係する ISO/IEC（ISO/IEC 29192-8）[49] で規格化されている状況を鑑み、本稿では 2016 年度版ガイドライン [23] に掲載されていない Grain-128A を新たな調査対象とし、その調査結果をまとめる。

各方式（Ascon を除く）の安全性解析状況については、2021 年度に公開された CRYPTREC 外部評価報告書 [114] に基づき、2021 年 9 月時点の状況を記載している。なお、新たに調査対象として追加した Grain-128A について、文献 [114] では安全性解析状況だけでなく、仕様等（設計者、発表年、仕様参照先、特徴、主な実装評価結果、標準化状況）もまとめられているため、文献 [114] の記載内容に従って調査結果をまとめた。

ここで示す方式には、ブロック暗号ないし tweakable ブロック暗号を用いているものも多い。これらの方式については理論的速度を測る指標としてレートを導入する。レートは 1 ブロック暗号で処理可能な入力ブロック数を表す。ソフトウェアの実装評価値は特に断りのない限り eBACS 内の Supercop ベンチマークシステム [11] での十分長いメッセージ処理の結果、ハードウェアの実装評価値も同様に特に断りのない限り ATHENA ベンチマークシステム [22] の結果である。ソフトウェアの評価尺度は十分長いメッセージでのバイトあたりの処理サイクル数（Cycles/Byte、C/B と略す)、ハードウェアでの評価は FPGA のスライス数（slices）と最大動作周波数（fmax)、ASIC ハードウェア実装の場合はサイズの評価尺度は Gate equivalent (GE) を用いるものとする。その他、これらの公式ベンチマークに当てはまらない注目すべき実装についても適宜報告する。いずれの場合も最適化実装の有無・最適化の度合いにより結果は大きく変わりうるため注意が必要である。著者の所属については提案時点のものである。

2019 年 2 月 20 日に CAESAR final portfolio が発表され、Use Case 1 (Lightweight Applications) として Ascon と ACORN、Use Case 2 (High-performance Applications) として AEGIS-128 と OCB、Use Case 3 (Defense in Depth) として Deoxys-II と COLM の合計 6 方式が選出された。final portfolio に選出された方式についてはその旨を記載している。なお、2016 年度版ガイドライン [23] ではこれら 6 方式のうち AEGIS-128 と COLM の 2 方式について掲載していない。これら 2 方式については軽量性を謳った方式ではないものの、CAESAR final portfolio に選出された方式であるということを鑑み、付録 B で調査結果をまとめた。

2021 年 3 月 30 日に NIST 軽量暗号（NIST LWC）プロジェクトのファイナリストが発表され、Ascon、Elephant、GIFT-COFB、Grain-128AEAD、ISAP、PHOTON-Beetle、Romulus、Sparkle、TinyJAMBU、Xoodyak の 10 方式がファイナリストとして選出された。その後、2023 年 2 月 7 日に最終選考結果が発表され、Ascon が最終選考方式として選出された。NIST LWC プロジェクトの動向を踏まえ、Ascon の記載内容については、2016 年度版ガイドライン [23] から大幅に更新されていることに注意が必要である。具体的には、2022 年度に公開された CRYPTREC 外部評価報告書 [115, 116] に基づき、2022 年 9 月現在の内容を記載している。また、Ascon を除くファイナリスト 9 方式についても軽量性の観点で優れており、かつ安全性の観点で問題が見つかっていない方式であることから、これらの方式も同様に付録 C で調査結果をまとめた。

Ascon の実装評価結果については文献 [115] に基づき更新している。文献 [115] で多くの実装評価結果がまとめられているものの、紙面の都合上、次の項目に限定している。ハードウェア実装評価結果については、FPGA 実装に着目し、回路面積の観点からコンパクト実装である結果、またはスループットの観点で高速実装である結果を抽出している。回路面積の評価尺度は、ルックアップテーブル数（LUTs）である。ソフトウェア実装評価結果については、IoT 向けローエンド CPU、特に Arm Cortex-M0 上での実装に着目し、設計者が作成したリファレンスコードを使用した場合のレイテンシ（暗号化・復号)、ROM サイズ、コードサイズの結果をまとめている。レイテンシの評価尺度は、テストベクトルを実行した際の 1 回の処理にかかる実行時間（msec）の平均値である。その他、文献 [115] では、ASIC 実装、命令拡張のハードウェア実装、ハイエンド CPU 上でのソフトウェア実装の結果がまとめられている。

<table>
<tr><td>技術分野</td><td>認証暗号、ハッシュ関数</td></tr>
<tr><td>名称</td><td>Ascon</td></tr>
<tr><td>設計者</td><td>Christoph Dobraunig[1,2], Maria Eichlseder[2], Florian Mendel[3], Martin Schlaffer[3]
(1: Radboud University/Netherlands, 2: Graz University of Technology/Austria, 3: Infineon Technologies AG/Germany)</td></tr>
<tr><td>発表年</td><td>2014（DIAC 2014 [29]）、2019（NIST LWC ウェブサイト [7]）</td></tr>
<tr><td>仕様参照先</td><td>CAESAR ウェブサイト [10]、NIST LWC ウェブサイト [31]、設計者ウェブサイト [28]</td></tr>
<tr><td>特徴</td><td>Ascon は暗号学的置換をプリミティブとして用いた MonkeyDuplex 構造 [13, 26] に基づく 2 つの認証暗号 Ascon-128、Ascon-128a と Sponge 構造 [12] に基づく 2 つのハッシュ関数 Ascon-Hash、Ascon-Hasha をまとめた総称である。ソフトウェアとハードウェアの両面で軽量性があること、そしてサイドチャネル耐性があることを主張している。
使用する暗号学的置換 p は SPN 型のラウンド関数であり、定数加算、非線形部（5 ビット S-box）、線形部（64 ビット単位の巡回シフトと XOR）で構成されている。ブロックサイズは 320 ビット、段数の異なる 2 種類の暗号学的置換（p^a、p^b）が使用される。また、認証暗号とハッシュ関数におけるパラメータの違いは下表のとおりであり、設計者が推奨する認証暗号は Ascon-128 で、ハッシュ関数は Ascon-Hash である。
<table>
<tr><th>名称</th><th>鍵長</th><th>nonce 長</th><th>タグ長</th><th>出力長</th><th>レート</th><th>p^a の段数</th><th>p^b の段数</th></tr>
<tr><td>Ascon-128</td><td>128</td><td>128</td><td>128</td><td>–</td><td>64</td><td>12</td><td>6</td></tr>
<tr><td>Ascon-128a</td><td>128</td><td>128</td><td>128</td><td>–</td><td>128</td><td>12</td><td>8</td></tr>
<tr><td>Ascon-Hash</td><td>–</td><td>–</td><td>–</td><td>256</td><td>64</td><td>12</td><td>12</td></tr>
<tr><td>Ascon-Hasha</td><td>–</td><td>–</td><td>–</td><td>256</td><td>64</td><td>12</td><td>8</td></tr>
</table>
なお、CAESAR final portfolio の Use Case 1 (Lightweight Applications) と NIST LWC プロジェクトの最終選考方式に選出された。</td></tr>
<tr><td>安全性解析状況</td><td>2022 年 9 月現在、いくつかの解析論文 [6, 30, 34, 37, 45, 66, 68, 71, 74, 86, 87, 112] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。
設計者 [31] は認証暗号とハッシュ関数の安全性がそれぞれ MonkeyDuplex 構造 [13, 26] と Sponge 構造 [12] の安全性に帰着できると主張している。さらに、認証暗号は一般的な MonkeyDuplex 構造とは異なり、初期化・最終処理フェーズで秘密鍵をそれぞれ 2 回適用していることから、安全性がさらに向上していると主張している。
認証暗号に対する最良の攻撃は、2021 年に提案された Rohit ら [86] によるキューブ攻撃と 2021 年に提案された Gerault ら [37] による差分攻撃であり、Rohit ら [86] は 12 段中 7 段の Ascon に対する鍵回復攻撃、Gerault ら [37] は 12 段中 4 段の Ascon に対する偽造攻撃を示した。ハッシュ関数に対する最良の攻撃は、2021 年に提案された Gerault ら [37] による差分攻撃であり、12 段中 2 段の Ascon に対して衝突攻撃が実行できる。プリミティブに対する最良の攻撃は設計者ら [30] によるゼロサム識別攻撃であり、フルラウンドの識別攻撃が可能であるが、この攻撃が認証暗号とハッシュ関数の安全性を脅かすものではないと主張されている。その他、Gerault ら [37] は 12 段のうち 7 段に簡略化したプリミティブに対して制限付き誕生日識別攻撃が実行できると報告している。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果（FPGA 実装）
<table>
<tr><th>Algorithm</th><th>Platform</th><th>Area</th><th>Throughput</th><th>Ref.</th></tr>
<tr><td>Ascon-128</td><td>Spartan-6</td><td>1,712 LUTs</td><td>2.88 Gbps</td><td>[35]</td></tr>
<tr><td>Ascon-128</td><td>Spartan-6</td><td>684 LUTs</td><td>60.10 Mbps</td><td>[104]</td></tr>
<tr><td>Ascon-Hash</td><td>Artix-7</td><td>2,181 LUTs</td><td>1.03 Gbps</td><td>[81]</td></tr>
</table>
ソフトウェア実装評価結果（Arm Cortex-M0）
<table>
<tr><th>Algorithm</th><th>Enc</th><th>Dec</th><th>ROM</th><th>Code</th><th>Ref.</th></tr>
<tr><td>Ascon-128</td><td>0.529 msec</td><td>0.536 msec</td><td>31.4 Kbyte</td><td>29.4 Kbyte</td><td>[44]</td></tr>
</table></td></tr>
</table>

技術分野	認証暗号
名称	ACORN
設計者	Hongjun Wu (Nanyang Technological University/Singapore)
発表年	2014 (DIAC 2014 [102])
仕様参照先	CAESAR ウェブサイト [10]
特徴	LFSR と単純な非線形処理を利用した方式。ハードウェア向けのストリーム暗号である Grain や Trivium と類似したシンプルな構造を持つ。 鍵は 128 ビットであり、LFSR を 6 つを組み合わせ、293 ビットを内部状態として保持する。 Grain や Trivium と同様にハードウェアに向いている。 なお、CAESAR final portfolio の Use Case 1 (Lightweight Applications) に選出された。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [27, 40, 41, 42, 43, 57, 60, 88, 89, 101, 105, 107] が発表されているが、仕様において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 最良の攻撃は、2021 年に提案された Hao ら [43] によるキューブ攻撃であり、1792 段のうち 775 段に簡略化した ACORN v3 の初期化フェーズに対しては、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。
主な実装評価結果	(SW) Intel Core i5-6600 (Skylake 3.31 GHz) で 8.46 C/B。 (HW) Virtex 6 で 135 slices、fmax 389 MHz。

技術分野	認証暗号
名称	AES-JAMBU
設計者	Hongjun Wu, Tao Huang (Nanyang Technological University/Singapore)
発表年	2014 (DIAC 2014 [103])
仕様参照先	CAESAR ウェブサイト [10]
特徴	ブロック暗号利用モードである。ブロック暗号として AES-128 と SIMON [8] を採用。SIMON はブロックサイズ/鍵長（ビット）で 64/96、96/96、128/128 の 3 バージョンを指定。 ブロック暗号の入出力以外にブロックサイズの半分を状態変数として用いてシリアルに処理を行う。ブロック暗号 1 回ごとにブロックサイズの半分の暗号化を行う。状態変数のサイズが小さいため小規模ハードウェアに向いている。
安全性解析状況	一般的な暗号利用モードとは異なり、ブロック暗号の計算量的安全性に基づく安全性帰着を提案者は示していない。提案者の主張では k ビット鍵、$2n$ ビットブロック暗号のときに、暗号化の安全性で k ビット、認証の安全性で n ビットとしている。 2021 年 9 月現在、Peyrin ら [80] による解析論文の他、目立った解析論文は発表されていない。 2015 年に Peyrin ら [80] は、nonce-misuse シナリオにおいて $2^{n/2}$ 回の暗号化による攻撃と、nonce-respecting シナリオにおける CCA2 (adaptive chosen-ciphertext attack) 安全性 [9] に対する計算量 $2^{3n/2}$ の攻撃を報告している。なお、$n = 64$ である。
主な実装評価結果	(SW) AES-128 利用、 Intel Core i5-6600 (Skylake 3.31 GHz) で 5.71 C/B。 (HW) Virtex 6 で 453 slices、fmax 209.8 MHz。

技術分野	認証暗号
名称	AES-OTR
設計者	Kazuhiko Minematsu（NEC Corporation/Japan）
発表年	2014 (EUROCRYPT 2014 [76])
仕様参照先	CAESAR ウェブサイト [10]
特徴	ブロック暗号利用モードである。CAESAR 提案は AES を利用している。 OCB と類似した構造を持つ。2 ラウンドのフェイステル置換を用いており、ほぼ暗号化のみの計算量で処理が可能。並列処理も可能。OCB と異なり、認証暗号としての復号処理も AES 暗号化関数のみで実行可能であり、AES 復号を用いない。
安全性解析状況	提案論文にて、OTR の安全性がブロック暗号の擬似ランダム性（Pseudorandomness）へ帰着可能なことが示されている。n ビットブロック暗号の利用において $n/2$ ビットの証明可能安全性を有する。 2021 年 9 月現在、Bost ら [19] による解析論文の他、目立った解析論文は発表されていない。 2016 年に Bost ら [19] により内部のマスク生成における安全性証明との齟齬が指摘され、提案者により修正版が提案されている。
主な実装評価結果	(SW) AES-128 利用、Intel Core i5-6600 (Skylake 3.31 GHz) で 0.68 C/B。 (HW) Virtex 6 で 1,385 slices、fmax 256.9 MHz。 Arm v7 実装 [77]：1GHz Cortex-A8 マイコンボード上で 23.5 C/B (42.5 MByte/sec)。 Banik らによる ASIC 実装 [5]：部分的に外部メモリ利用、入力長の制約など加えた特殊条件下で実装し 6,000 GE 台

利用実績等	`https://www.nec-solutioninnovators.co.jp/ss/mobility/control.html`
	`https://www.nec-solutioninnovators.co.jp/sl/emb/pdf/automotive.pdf`
	（いずれも 2023-10-04 閲覧）

技術分野	認証暗号
名称	CLOC and SILC
設計者	Tetsu Iwata[1], Kazuhiko Minematsu[2], Jian Guo[3], Sumio Morioka[4], Eita Kobayashi[2] (1: Nagoya University/Japan, 2: NEC Corporation/Japan, 3: Nanyang Technological University/Singapore, 4: NEC Europe Ltd./UK)
発表年	2014（FSE 2014 [52]、DIAC 2014 [53]）
仕様参照先	CAESAR ウェブサイト [10]、FSE 2014 [52]、設計者ウェブサイト [51]
特徴	ブロック暗号利用モードである。CFB と CBC-MAC をベースにしたレート 1/2 の方式。鍵以外に必要なメモリ量が小さいのが特徴（n ビットブロック暗号利用で約 $2n$ ビット）。CLOC は処理のオーバーヘッドを削減し短い入力での性能向上を狙っており、組み込みソフトウェア向き。SILC は CLOC の処理を簡素化したハードウェア向けの方式。 CLOC、SILC ともに 128 ビットブロック暗号として AES を採用。64 ビットブロック暗号として CLOC は TWINE [96] を採用。SILC は PRESENT [18] および LED [38] を採用。
安全性解析状況	2021 年 9 月現在、目立った解析論文は発表されていない。 提案論文にて、CLOC と SILC の安全性が用いるブロック暗号の疑似ランダム性に帰着可能であることが示されている。n ビットブロック暗号を用いたとき $n/2$ ビットの安全性を有する。 nonce を誤って暗号化で重複させた場合でも暗号文の改ざんに対する安全性が保証されている。
主な実装評価結果	(SW) Intel Core i5-6600 (Skylake 3.31 GHz) で CLOC が 2.82 C/B。SILC は 2.78 C/B。 (HW) Virtex 6、 CLOC が 891 slices、fmax 280.9 MHz。SILC が 989 slices、fmax 280.7 MHz。 CLOC の 8 ビットマイコン実装 [52]: AVR ATmega128 (16 Mhz)。初期化に 2,000 サイクル、32 バイト暗号化に 550 C/B。 Banik らによる ASIC 実装 [5]：部分的に外部メモリ利用、入力長の制約など加えた特殊条件下で実装し、CLOC-AES、SILC-AES ともに約 3,100 GE。

技術分野	認証暗号
名称	Deoxys
設計者	Jérémy Jean, Ivica Nikolić, Thomas Peyrin (Nanyang Technological University/Singapore)
発表年	2014 (DIAC 2014 [54]、ASIACRYPT 2014 [55])
仕様参照先	CAESAR ウェブサイト [10]、ASIACRYPT 2014 [55]
特徴	専用 tweakable ブロック暗号 Deoxys-BC を利用するブロック暗号利用モード。 Deoxys-BC は 128 ビットブロック、256 ビット tweak+key、ラウンド関数は AES そのものであり、段数は 14 から 16 のいずれか。 ブロック暗号利用モードは TAE [70] と SCT [79] の 2 種類。TAE モードを用いる場合は 128 ビット安全性を有する。 TAE モードでは OCB 同様の実装面の特徴を有し、レート 1 での並列処理が可能である。一方の SCT は 2 パス、レート 1/2 のオフライン処理だが、SCT モードが deterministic AE (あるいは misuse-resistant AE) [85] の機能を有することにより、nonce の重複に対する安全性を持つ。 なお、Deoxys のバリエーションの 1 つである Deoxys-II が CAESAR final portfolio の Use Case 3 (Defense in Depth) に選出された。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [21, 33, 63, 64, 72, 78, 79, 90, 108, 109, 113] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 関連 tweakey 設定における最良の攻撃は、2019 年に提案された Zhao ら [108] と 2019 年に提案された Zhao ら [109] によって提案された rectangle attack であり、14 段のうち 13 段に簡略化した Deoxys-BC-256、16 段のうち 14 段に簡略化した Deoxys-BC-384、14 段のうち 10 段に簡略化した Deoxys-I-128-128、16 段のうち 13 段に簡略化した Deoxys-I-256-128 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。 なお、Deoxys-BC と Deoxys-I に関する解析論文がいくつか発表されているが、Deoxys-II に関する解析論文は発表されていない。
主な実装評価結果	(SW) Intel Core i5-6600 (Skylake 3.31 GHz) で 0.87 C/B。 (HW) Virtex 6 で 993 slices、fmax 330 MHz。 Deoxys-BC 単体の ASIC 実装が 2,860 GE [56]。

技術分野	認証暗号
名称	Joltik
設計者	Jérémy Jean, Ivica Nikolić, Thomas Peyrin (Nanyang Technological University/Singapore)
発表年 (発表学会等)	2014 (DIAC 2014 [54]、ASIACRYPT 2014 [55])
仕様参照先	CAESAR ウェブサイト [10]、ASIACRYPT 2014 [55]
特徴	専用 tweakable ブロック暗号 Joltik-BC を利用。 Joltik-BC は 64 ビットブロック、128 ビット tweak+key、ラウンド関数は 4 ビット S-box を用いた SPN 構造、段数は 24 から 32 のいずれか。 Deoxys 同様、モードは TAE と SCT の 2 種類である。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [62, 65, 73, 111] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 単一鍵設定における最良の攻撃、2019 年に提案されたは Li ら [65] による中間一致攻撃であり、24 段のうち 8 段に簡略化した Joltik-BC-128、32 段のうち 10 段に簡略化した Joltik-BC-192 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。関連 tweakey 設定における最長の攻撃は、2021 年に提案された Li ら [62] による中間一致攻撃であり、24 段のうち 9 段に簡略化した Joltik-BC-128、32 段のうち 11 段に簡略化した Joltik-BC-192 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。 なお、Joltik-BC に関する解析論文がいくつか発表されているが、認証暗号としての Joltik に関する解析論文は発表されていない。
主な実装評価結果	(SW) Intel Core i5-6600 (Skylake 3.31 GHz) で 13.32 C/B。 (HW) Virtex 6 で 494 slices、fmax 430 MHz。

技術分野	認証暗号
名称	Ketje
設計者	Guido Bertoni[1], Joan Daemen[1], Michael Peeters[2], Gilles Van Assche[1], Ronny Van Keer[1] (1: STMicroelectronics/Switzerland, 2: NXP Semiconductors/Belgium)
発表年	2014 (DIAC 2014 [15])
仕様参照先	CAESAR ウェブサイト [10]、設計者ウェブサイト [25]
特徴	Sponge 構造を持つ。利用モードは MonkeyDuplex [24] がベースとなる。 内部の暗号学的置換は Keccak-p と呼ばれ、SHA-3 関数で用いられる Keccak-f 置換 [14] をベースとしたものである。200 ビット幅の置換を利用するものを Ketje-JR、400 ビット幅のものを Ketje-SR と呼ぶ。 メモリサイズの小ささと計算量の少なさによる、ハード・ソフト両面での軽量性を謳っている。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [17, 32, 36, 67, 94, 95, 110] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 最良の攻撃は、2018 年に提案された Song [94] によるキューブ攻撃、2018 年に提案された Song ら [95] によるキューブ攻撃、2019 年に提案された Li ら [67] による条件付きキューブ攻撃、2021 年に提案された Zhao ら [110] によるキューブ攻撃であり、12 段のうち 5 段に簡略化した Ketje Jr、12 段のうち 7 段に簡略化した Ketje Sr/Minor/Major に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。また、Ketje Jr に対する内部状態復元攻撃が Fuhr ら [36] によって 2018 年に提案され、ビットレートが 40 の場合には仕様段数であっても秘密鍵の全数探索より効率的に内部状態復元攻撃が実行できるものの、推奨パラメータであるビットレートが 16 の場合には秘密鍵の全数探索より効率的に内部状態復元攻撃が実行できない。
主な実装評価結果	(SW) Ketje-SR、 Intel Core i5-6600 (Skylake 3.31 GHz) で 42.57 C/B。 (HW) Virtex 6 で 456 slices、fmax 229.5 MHz。

技術分野	認証暗号
名称	Minalpher
設計者	Yu Sasaki[1], Yosuke Todo[1], Kazumaro Aoki[1], Yusuke Naito[2], Takeshi Sugawara[2], Yumiko Murakami[2], Mitsuru Matsui[2], Shoichi Hirose[3] (1: NTT/Japan, 2: Mitsubishi Electric Corporation/Japan, 3: University of Fukui/Japan)
発表年	2014 (DIAC 2014 [93])
仕様参照先	CAESAR ウェブサイト [10]
特徴	256 ビットの暗号学的置換 Minalpher-P を用いた専用 256 ビット Tweakable Even-Mansour ブロック暗号 (TEM) を利用、モードは独自方式。 TEM が用いる内部の置換は 4 ビット S-box 利用の SPN 構造、暗号化・復号関数の統合が容易となる構造を採用している。nonce の重複に対し部分的な安全性を有する。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [20, 39, 91] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 最良の攻撃は、2017 年に提案された佐々木ら [91] による不能差分攻撃であり 17.5 段のうち 7.5 段に簡略化した Minalpher に対して、効率的に識別攻撃を実行できる。
主な実装評価結果	(SW) Intel Core i5-6600 (Skylake 3.31 GHz) で 5.81 C/B。 (HW) Virtex 6 で 1,104 slices、fmax 280.9 MHz。 SIMD 実装 [92] : Intel CPU Core i7 (Haswell) で 5.6 C/B。 8 ビット RL78 マイコン実装 [92]: 510 ROM、214 RAM バイトの利用で約 2,800 C/B、1,275 ROM、470 RAM バイトの利用で 514 C/B。

技術分野	認証暗号
名称	OCB
設計者	Ted Krovetz[1], Phillip Rogaway[2] (1: California State University/USA, 2: University of California/USA)
発表年	2001 (ACM CCS 2001 [84])、 2004 (ASIACRYPT 2004 [82])、 2011 (FSE 2011 [58])
仕様参照先	CAESAR ウェブサイト [10] 公式ウェブサイト `http://web.cs.ucdavis.edu/~rogaway/ocb/`
特徴	ブロック暗号利用モードである。AES を利用したバージョンが IETF RFC 7253 [59] にて規定されている。CAESAR 提案は IETF RFC 7253 と同じ。ECB モードと類似した構造だが、メッセージ認証は平文ブロックのチェックサム、実際は排他的論理和をとり、これを暗号化するのみで実現しており、全体の計算量はほぼ暗号化のみの計算量と同等である。さらにブロックごとの並列処理が可能である。 基本的な構造は 2001 年に提案されており、マスク生成の違いなどで後に複数のバージョンが提案されている。AES を用いるケースでは、特に AESNI 命令が利用可能な CPU において顕著な高速性を有する。 なお、CAESAR final portfolio の Use Case 2 (High-performance Applications) に選出された。OCB には 3 種類のバリエーション（OCB1、OCB2、OCB3）があるが、CAESAR への提案方式は OCB3 である。
安全性解析状況	2021 年 9 月現在、いくつかの解析論文 [16, 46, 47, 48, 106] が発表されている。 提案論文 [58, 82, 83, 84] にて、OCB の安全性がブロック暗号の強擬似ランダム性（Strong Pseudorandomness）へ帰着可能なことが示されている。n ビットブロック暗号の利用において $n/2$ ビットの証明可能安全性を有する。 2020 年に Inoue ら [47, 48] は、OCB2 の基礎となる tweakable ブロック暗号 XEX*に欠陥があることを示すとともに、既存の安全性証明にも欠陥があることを指摘した。これらの欠陥を悪用することにより、現実的な攻撃として universal forgeries と full plaintext recovery が可能となる。結果として、OCB2 が ISO/IEC 19772:2009-02 規格から除外された [1]。なお、本攻撃は OCB1 と OCB3 には影響がない。 2023 年に Liénardy と Lafitte [69] は、OCB3 の nonce 長が 6 ビット未満の場合に現実的な攻撃が可能であることを報告している。文献 [69] では OCB3 の仕様変更が提案されているが、仕様が変更されない場合でも 6 ビット以上の nonce を使用すれば安全であると主張されている。このような短い nonce の実用性に議論はあるものの、仕様において最低 nonce 長の記載がない以上はリスクがあることに注意が必要である。
主な実装評価結果	(SW) AES-128 利用、 Intel Core i5-6600 (Skylake 3.31 GHz) で 0.64 C/B。 (HW) Virtex 6 で 1,348 slices、fmax 292.7 MHz。 その他多様な CPU での実装結果が報告されている [58]。
標準化状況	IETF RFC 7253 [59]

技術分野	認証暗号
名称	PRIMATEs
設計者	Elena Andreeva[1], Begul Bilgin[1], Andrey Bogdanov[2], Atul Luykx[1], Florian Mendel[3], Bart Mennink[1], Nicky Mouha[1], Qingju Wang[1], Kan Yasuda[4] (1: KU Leuven/Belgium, 2: Technical University of Denmark/Denmark, 3: Graz University of Technology/Austria, 4: NTT/Japan)
発表年	2014 (DIAC 2014 [3]、FSE 2014 [4])
仕様参照先	CAESAR ウェブサイト [10]、FSE 2014 [4]
特徴	Sponge 構造をもつ。具体的には、それぞれ異なる利用モードを持つ HANUMAN、GIBBON、APE の 3 つの方式で構成される。いずれの方式も 200 ないし 280 ビットの置換を内部要素とし、この置換を公開ランダム置換と仮定した場合に 80 ビットないし 120 ビットセキュリティを持つことが保証されている。内部の置換は AES ないし Rijndael と類似した SPN だが S-box は 5 ビットである。HANUMAN、GIBBON はそれぞれ既知の利用モード（SpongeWrap、MonkeyWrap）をベースとするが、APE は nonce 重複など考慮した独自の利用モードである。
安全性解析状況	HANUMAN に対して、Associated Data がないときの処理の問題点を利用した現実的な偽造作成攻撃が報告されている [100]。提案者による修正が提案されている。 その他、2021 年 9 月現在において目立った解析論文は発表されていない。
主な実装評価結果	GIBBON について、 (SW) Intel Core i5-6600 (Skylake 3.31 GHz) で 1,712 C/B。 (HW) Virtex 6 で 419 slices、fmax 333.4 MHz。

<table>
<tr><td>技術分野</td><td>認証暗号</td></tr>
<tr><td>名称</td><td>Grain-128A</td></tr>
<tr><td>設計者</td><td>Martin Agren[1], Martin Hell[1], Thomas Johansson[1], Willi Meier[2]
(1: Lund University/Sweden, 2: FHNW/Switzerland)</td></tr>
<tr><td>発表年</td><td>2011（IJWMC 2011 [2]）</td></tr>
<tr><td>仕様参照先</td><td>IJWMC 2011 [2]</td></tr>
<tr><td>特徴</td><td>Grain-128A は、eSTREAM portfolio に選出された Grain v1、その派生版である Grain-128 と同様の構造を有するハードウェア実装向けのストリーム暗号であるが、認証機能をサポートしていることが大きな違いである。初期化フェーズは 256 段、鍵長は 128 ビット、nonce 長は 96 ビットであり、タグ長は任意に設定できるものの 32 ビットが推奨されている。Grain-128A は、Grain-128 に対する既存攻撃に耐性を持つよう非線形関数に改良が施されている。
なお、NIST LWC プロジェクトのファイナリストの 1 つである Grain-128AEAD も同様の構造を有しており、Grain-128A に対する既存攻撃に耐性を持つようさらに改良が施されている。</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [40, 61, 97, 98, 99, 101] が発表されている。
単一鍵設定における最良の攻撃は、2018 年に提案された Todo ら [99] による高速相関攻撃であり、Grain-128A に対して仕様段数であっても秘密鍵の全数探索より効率的に内部状態復元攻撃が実行できる。なお、本攻撃では $2^{115.4}$ の計算量と $2^{113.8}$ のデータ量が必要となる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果（Cadence RTL Compiler, TSMC 90 nm ASIC）[75]
<table>
<tr><th># of parallel</th><th>Frequency [GHz]</th><th>Throughput [Gbps]</th><th>Area [μm^2]</th><th>Power [μW]</th></tr>
<tr><td>1</td><td>2.1</td><td>1.1</td><td>5,876</td><td>96.9</td></tr>
<tr><td>2</td><td>2.0</td><td>2.0</td><td>6.972</td><td>106.1</td></tr>
<tr><td>4</td><td>2.0</td><td>4.0</td><td>8,299</td><td>120.6</td></tr>
<tr><td>8</td><td>1.9</td><td>7.6</td><td>10,778</td><td>176.4</td></tr>
<tr><td>16</td><td>1.7</td><td>13.6</td><td>15,709</td><td>247.8</td></tr>
<tr><td>32</td><td>1.5</td><td>24.0</td><td>23,430</td><td>417.9</td></tr>
</table>
なお、全てオリジナル実装の結果である。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29167-13 [50]、ISO/IEC 29192-8 [49]</td></tr>
</table>

参考文献

[1] ISO/IEC JTC 1/SC 27 STATEMENT ON OCB2.0 – Major weakness found in a standardised cipher scheme (2019-01-09, press release), https://www.din.de/resource/blob/321470/da3d9bce7116deb510f6aded2ed0b4df/20190107-press-release-19772-2009-1st-ed-ocb2-0-data.pdf（2023-10-04 閲覧）

[2] Ågren, M., Hell, M., Johansson, T., Meier, W.: Grain-128a: a new version of Grain-128 with optional authentication. Int. J. Wirel. Mob. Comput. 5(1), 48–59 (2011), https://doi.org/10.1504/IJWMC.2011.044106

[3] Andreeva, E., Bilgin, B., Bogdanov, A., Luykx, A., Mendel, F., Mennink, B., Mouha, N., Wang, Q., Yasuda, K.: CAESAR candidates PRIMATES. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[4] Andreeva, E., Bilgin, B., Bogdanov, A., Luykx, A., Mennink, B., Mouha, N., Yasuda, K.: APE: Authenticated Permutation-Based Encryption for Lightweight Cryptography. In: FSE. Lecture Notes in Computer Science, vol. 8540, pp. 168–186. Springer (2014)

[5] Banik, S., Bogdanov, A., Minematsu, K.: Low-area hardware implementations of CLOC, SILC and AES-OTR. In: HOST. pp. 71–74. IEEE Computer Society (2016)

[6] Bar-On, A., Dunkelman, O., Keller, N., Weizman, A.: DLCT: A New Tool for Differential-Linear Cryptanalysis. In: Ishai, Y., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11476, pp. 313–342. Springer (2019), https://doi.org/10.1007/978-3-030-17653-2_11

[7] Bassham, L., Chang, D., Kang, J., Kelsey, J., McKay, K., Turan, M.S., Waller, N.: NIST Lightweight Cryptography Project, https://csrc.nist.gov/projects/lightweight-cryptography

[8] Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., Wingers, L.: The SIMON and SPECK Families of Lightweight Block Ciphers. Cryptology ePrint Archive, Report 2013/404 (2013), https://eprint.iacr.org/2013/404

[9] Bellare, M., Desai, A., Jokipii, E., Rogaway, P.: A Concrete Security Treatment of Symmetric Encryption. In: FOCS. pp. 394–403. IEEE Computer Society (1997)

[10] Bernstein, D.J.: CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness, https://competitions.cr.yp.to/caesar.html（2023-10-04 閲覧）

[11] Bernstein, D.J.: eBACS: ECRYPT Benchmarking of Cryptographic Systems, https://bench.cr.yp.to/results-caesar.html（2023-10-04 閲覧）

[12] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: On the Indifferentiability of the Sponge Construction. In: Smart, N.P. (ed.) Advances in Cryptology - EUROCRYPT 2008, 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Istanbul, Turkey, April 13-17, 2008. Proceedings. Lecture Notes in Computer Science, vol. 4965, pp. 181–197. Springer (2008), https://doi.org/10.1007/978-3-540-78967-3_11

[13] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: Duplexing the Sponge: Single-Pass Authenticated Encryption and Other Applications. In: Miri, A., Vaudenay, S. (eds.) Selected Areas in Cryptography - 18th

International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7118, pp. 320–337. Springer (2011), https://doi.org/10.1007/978-3-642-28496-0_19

[14] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: The Keccak reference (2011), https://keccak.noekeon.org/（2023-10-04 閲覧）

[15] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V., Keer, R.V.: CAESAR candidates Ketje + Keyak. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[16] Bhaumik, R., Nandi, M.: Improved Security for OCB3. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10625, pp. 638–666. Springer (2017), https://doi.org/10.1007/978-3-319-70697-9_22

[17] Bi, W., Dong, X., Li, Z., Zong, R., Wang, X.: MILP-aided cube-attack-like cryptanalysis on Keccak Keyed modes. Des. Codes Cryptogr. 87(6), 1271–1296 (2019), https://doi.org/10.1007/s10623-018-0526-x

[18] Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: CHES. Lecture Notes in Computer Science, vol. 4727, pp. 450–466. Springer (2007)

[19] Bost, R., Sanders, O.: Trick or Tweak: On the (In)security of OTR's Tweaks. In: Cheon, J.H., Takagi, T. (eds.) Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10031, pp. 333–353 (2016), https://doi.org/10.1007/978-3-662-53887-6_12

[20] Canteaut, A., Lambooij, E., Neves, S., Rasoolzadeh, S., Sasaki, Y., Stevens, M.: Refined Probability of Differential Characteristics Including Dependency Between Multiple Rounds. IACR Trans. Symmetric Cryptol. 2017(2), 203–227 (2017), https://doi.org/10.13154/tosc.v2017.i2.203-227

[21] Cid, C., Huang, T., Peyrin, T., Sasaki, Y., Song, L.: A Security Analysis of Deoxys and its Internal Tweakable Block Ciphers. IACR Trans. Symmetric Cryptol. 2017(3), 73–107 (2017), https://doi.org/10.13154/tosc.v2017.i3.73-107

[22] Cryptographic Engineering Research Group at George Mason University: ATHENa: Automated Tools for Hardware EvaluatioN, https://cryptography.gmu.edu/athena/（2023-10-04 閲覧）

[23] CRYPTREC 軽量暗号ワーキンググループ: CRYPTREC 暗号技術ガイドライン（軽量暗号）（文書番号：CRYPTREC GL-2003-2016JP）（2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf

[24] Daemen, J.: Permutation-based encryption, authentication and authenticated encryption. DIAC - Directions in Authenticated Ciphers (2012), https://hyperelliptic.org/DIAC/

[25] Daemen, J., Hoffert, S., Peeters, M., Assche, G.V., Keer, R.V., Mella, S.: Team Keccak: The Ketje authenticated encryption scheme, https://keccak.team/ketje.html（2023-10-04 閲覧）

[26] Daemen, J., Mennink, B., Assche, G.V.: Full-State Keyed Duplex with Built-In Multi-user Support. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10625, pp. 606–637. Springer (2017), https://doi.org/10.1007/978-3-319-70697-9_21

[27] Ding, L., Wang, L., Gu, D., Jin, C., Guan, J.: Algebraic Degree Estimation of ACORN v3 Using Numeric Mapping. Secur. Commun. Networks 2019, 7429320:1–7429320:5 (2019), https://doi.org/10.1155/2019/7429320

[28] Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: ASCON: Lightweight Authenticated Encryption & Hashing, https://ascon.iaik.tugraz.at/（2023-10-04 閲覧）

[29] Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: CAESAR candidates Ascon. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[30] Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Cryptanalysis of Ascon. In: CT-RSA. Lecture Notes in Computer Science, vol. 9048, pp. 371–387. Springer (2015)

[31] Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Ascon v1.2. Submission to the NIST Lightweight Cryptography project (2021)

[32] Dong, X., Li, Z., Wang, X., Qin, L.: Cube-like Attack on Round-Reduced Initialization of Ketje Sr. IACR Trans. Symmetric Cryptol. 2017(1), 259–280 (2017), https://doi.org/10.13154/tosc.v2017.i1.259-280

[33] Dong, X., Qin, L., Sun, S., Wang, X.: Key Guessing Strategies for Linear Key-Schedule Algorithms in Rectangle Attacks. IACR Cryptol. ePrint Arch. 2021, 856 (2021), https://eprint.iacr.org/2021/856

[34] Erlacher, J., Mendel, F., Eichlseder, M.: Bounds for the Security of Ascon against Differential and Linear Cryptanalysis. IACR Trans. Symmetric Cryptol. 2022(1), 64–87 (2022), https://doi.org/10.46586/tosc.v2022.i1.64-87

[35] Farahmand, F., Diehl, W., Abdulgadir, A., Kaps, J., Gaj, K.: Improved Lightweight Implementations of CAESAR Authenticated Ciphers. In: 26th IEEE Annual International Symposium on Field-Programmable Custom Computing Machines, FCCM 2018, Boulder, CO, USA, April 29 - May 1, 2018. pp. 29–36. IEEE Computer Society (2018), https://doi.org/10.1109/FCCM.2018.00014

[36] Fuhr, T., Naya-Plasencia, M., Rotella, Y.: State-Recovery Attacks on Modified Ketje Jr. IACR Trans. Symmetric Cryptol. 2018(1), 29–56 (2018), https://doi.org/10.13154/tosc.v2018.i1.29-56

[37] Gérault, D., Peyrin, T., Tan, Q.Q.: Exploring differential-based distinguishers and forgeries for ASCON. IACR Cryptol. ePrint Arch. 2021, 1103 (2021), https://eprint.iacr.org/2021/1103, accepted to *IACR Trans. Symmetric Cryptol., 2021(3)*

[38] Guo, J., Peyrin, T., Poschmann, A., Robshaw, M.J.B.: The LED Block Cipher. In: CHES. Lecture Notes in Computer Science, vol. 6917, pp. 326–341. Springer (2011)

[39] Guo, Z., Wu, W., Liu, R., Zhang, L.: Multi-key Analysis of Tweakable Even-Mansour with Applications to Minalpher and OPP. IACR Trans. Symmetric Cryptol. 2016(2), 288–306 (2016), https://doi.org/10.13154/tosc.v2016.i2.288-306

[40] Hao, Y., Isobe, T., Jiao, L., Li, C., Meier, W., Todo, Y., Wang, Q.: Improved Division Property Based Cube Attacks Exploiting Algebraic Properties of Superpoly. IEEE Trans. Computers 68(10), 1470–1486 (2019), https://doi.org/10.1109/TC.2019.2909871

[41] Hao, Y., Jiao, L., Li, C., Meier, W., Todo, Y., Wang, Q.: Links between Division Property and Other Cube Attack Variants. IACR Trans. Symmetric Cryptol. 2020(1), 363–395 (2020), https://doi.org/10.13154/tosc.v2020.i1.363-395

[42] Hao, Y., Leander, G., Meier, W., Todo, Y., Wang, Q.: Modeling for Three-Subset Division Property Without Unknown Subset - Improved Cube Attacks Against Trivium and Grain-128AEAD. In: Canteaut, A., Ishai, Y. (eds.) Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12105, pp. 466–495. Springer (2020), https://doi.org/10.1007/978-3-030-45721-1_17

[43] Hao, Y., Leander, G., Meier, W., Todo, Y., Wang, Q.: Modeling for Three-Subset Division Property without Unknown Subset. J. Cryptol. 34(3), 22 (2021), https://doi.org/10.1007/s00145-021-09383-2

[44] Hira, R., Kitahara, T., Miyahara, D., Hara-Azumi, Y., Li, Y., Sakiyama, K.: Software Evaluation for Second Round Candidates in NIST Lightweight Cryptography. IACR Cryptol. ePrint Arch. p. 591 (2022), https://eprint.iacr.org/2022/591

[45] Hirch, S.E., Mella, S., Mehrdad, A., Daemen, J.: Improved Differential and Linear Trail Bounds for ASCON. IACR Trans. Symmetric Cryptol. 2022(4), 145–178 (2022), https://doi.org/10.46586/tosc.v2022.i4.145-178

[46] Hirose, S., Sasaki, Y., Yasuda, K.: Rate-One AE with Security Under RUP. In: Nguyen, P.Q., Zhou, J. (eds.) Information Security - 20th International Conference, ISC 2017, Ho Chi Minh City, Vietnam, November 22-24, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10599, pp. 3–20. Springer (2017), https://doi.org/10.1007/978-3-319-69659-1_1

[47] Inoue, A., Iwata, T., Minematsu, K., Poettering, B.: Cryptanalysis of OCB2: Attacks on Authenticity and Confidentiality. In: Boldyreva, A., Micciancio, D. (eds.) Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11692, pp. 3–31. Springer (2019), https://doi.org/10.1007/978-3-030-26948-7_1

[48] Inoue, A., Iwata, T., Minematsu, K., Poettering, B.: Cryptanalysis of OCB2: Attacks on Authenticity and Confidentiality. J. Cryptol. 33(4), 1871–1913 (2020), https://doi.org/10.1007/s00145-020-09359-8

[49] ISO/IEC: Information security – Lightweight cryptography – Part 8: Authenticated encryption (ISO/IEC 29192-8:2022), https://www.iso.org/standard/80114.html

[50] ISO/IEC: Information technology – Automatic identification and data capture techniques – Part 13: Crypto suite Grain-128A security services for air interface communications (ISO/IEC 29167-13: 2015), https://www.iso.org/standard/60682.html

[51] Iwata, T., Minematsu, K., Guo, J., Morioka, S.: CLOC and SILC – Authenticated Encryption Schemes for Constrained Devices, hhttps://www.nuee.nagoya-u.ac.jp/labs/tiwata/AE/

[52] Iwata, T., Minematsu, K., Guo, J., Morioka, S.: CLOC: Authenticated Encryption for Short Input. In: FSE. Lecture Notes in Computer Science, vol. 8540, pp. 149–167. Springer (2014)

[53] Iwata, T., Minematsu, K., Guo, J., Morioka, S., Kobayashi, E.: CAESAR candidates SILC. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[54] Jean, J., Nikolić, I., Peyrin, T.: CAESAR candidates DEOXYS + Joltik. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[55] Jean, J., Nikolic, I., Peyrin, T.: Tweaks and Keys for Block Ciphers: The TWEAKEY Framework. In: ASIACRYPT (2). Lecture Notes in Computer Science, vol. 8874, pp. 274–288. Springer (2014)

[56] Jean, J., Nikolić, I., Peyrin, T.: Deoxys and Joltik. DIAC - Directions in Authenticated Ciphers (2015)

[57] Kesarwani, A., Roy, D., Sarkar, S., Meier, W.: New cube distinguishers on NFSR-based stream ciphers. Des. Codes Cryptogr. 88(1), 173–199 (2020), https://doi.org/10.1007/s10623-019-00674-1

[58] Krovetz, T., Rogaway, P.: The Software Performance of Authenticated-Encryption Modes. In: FSE. Lecture Notes in Computer Science, vol. 6733, pp. 306–327. Springer (2011)

[59] Krovetz, T., Rogaway, P.: The OCB Authenticated-Encryption Algorithm. RFC 7253 (May 2014), https://www.rfc-editor.org/info/rfc7253

[60] Lafitte, F., Lerman, L., Markowitch, O., Heule, D.V.: SAT-based cryptanalysis of ACORN. IACR Cryptol. ePrint Arch. 2016, 521 (2016), https://eprint.iacr.org/2016/521

[61] Lehmann, M., Meier, W.: Conditional Differential Cryptanalysis of Grain-128a. In: Pieprzyk, J., Sadeghi, A., Manulis, M. (eds.) Cryptology and Network Security, 11th International Conference, CANS 2012, Darmstadt, Germany, December 12-14, 2012. Proceedings. vol. 7712, pp. 1–11. Springer (2012), https://doi.org/10.1007/978-3-642-35404-5_1

[62] Li, M., Chen, S.: Improved meet-in-the-middle attacks on reduced-round Joltik-BC. IET Information Security 15(3), 247–255 (2021)

[63] Li, M., Chen, S.: Improved Meet-in-the-Middle Attacks on Reduced-Round Tweakable Block Cipher Deoxys-BC. The Computer Journal (06 2021), https://doi.org/10.1093/comjnl/bxab076

[64] Li, R., Jin, C.: Meet-in-the-middle attacks on round-reduced tweakable block cipher Deoxys-BC. IET Inf. Secur. 13(1), 70–75 (2019), https://doi.org/10.1049/iet-ifs.2018.5091

[65] Li, R., Jin, C., Pan, H.: Key recovery attacks on reduced-round Joltik-BC in the single-key setting. Inf. Process. Lett. 151 (2019), https://doi.org/10.1016/j.ipl.2019.105834

[66] Li, Y., Zhang, G., Wang, W., Wang, M.: Cryptanalysis of round-reduced ASCON. Sci. China Inf. Sci. 60(3), 38102 (2017), https://doi.org/10.1007/s11432-016-0283-3

[67] Li, Z., Dong, X., Bi, W., Jia, K., Wang, X., Meier, W.: New Conditional Cube Attack on Keccak Keyed Modes. IACR Trans. Symmetric Cryptol. 2019(2), 94–124 (2019), https://doi.org/10.13154/tosc.v2019.i2.94-124

[68] Li, Z., Dong, X., Wang, X.: Conditional Cube Attack on Round-Reduced ASCON. IACR Trans. Symmetric Cryptol. 2017(1), 175–202 (2017), https://doi.org/10.13154/tosc.v2017.i1.175-202

[69] Liénardy, J., Lafitte, F.: A weakness in OCB3 used with short nonces allowing for a break of authenticity and confidentiality. Inf. Process. Lett. 183, 106404 (2024), https://doi.org/10.1016/j.ipl.2023.106404

[70] Liskov, M., Rivest, R.L., Wagner, D.: Tweakable Block Ciphers. In: Yung, M. (ed.) CRYPTO. Lecture Notes in Computer Science, vol. 2442, pp. 31–46. Springer (2002)

[71] Liu, M., Lu, X., Lin, D.: Differential-Linear Cryptanalysis from an Algebraic Perspective. In: Malkin, T., Peikert, C. (eds.) Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12827, pp. 247–277. Springer (2021), https://doi.org/10.1007/978-3-030-84252-9_9

[72] Liu, Y., Shi, B., Gu, D., Zhao, F., Li, W., Liu, Z.: Improved Meet-in-the-Middle Attacks on Reduced-Round Deoxys-BC-256. Comput. J. 63(12), 1859–1870 (2020), https://doi.org/10.1093/comjnl/bxaa028

[73] Liu, Y., Shi, Y., Gu, D., Zeng, Z., Zhao, F., Li, W., Liu, Z., Bao, Y.: Improved Meet-in-the-Middle Attacks on Reduced-Round Kiasu-BC and Joltik-BC. Comput. J. 62(12), 1761–1776 (2019), https://doi.org/10.1093/comjnl/bxz059

[74] Makarim, R.H., Rohit, R.: Towards Tight Differential Bounds of Ascon A Hybrid Usage of SMT and MILP. IACR Trans. Symmetric Cryptol. 2022(3), 303–340 (2022), https://doi.org/10.46586/tosc.v2022.i3.303-340

[75] Mansouri, S.S., Dubrova, E.: An Improved Hardware Implementation of the Grain-128a Stream Cipher. In: Kwon, T., Lee, M., Kwon, D. (eds.) Information Security and Cryptology - ICISC 2012 - 15th International Conference, Seoul, Korea, November 28-30, 2012, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7839, pp. 278–292. Springer (2012), https://doi.org/10.1007/978-3-642-37682-5_20

[76] Minematsu, K.: Parallelizable Rate-1 Authenticated Encryption from Pseudorandom Functions. In: EUROCRYPT. Lecture Notes in Computer Science, vol. 8441, pp. 275–292. Springer (2014)

[77] Minematsu, K., Shigeri, M., Kubo, H.: AES-OTR v2. DIAC - Directions in Authenticated Ciphers (2015)

[78] Moazami, F., Mehrdad, A., Soleimany, H.: Impossible Differential Cryptanalysis on Deoxys-BC-256. ISC Int. J. Inf. Secur. 10(2), 93–105 (2018), https://doi.org/10.22042/isecure.2018.114245.405

[79] Peyrin, T., Seurin, Y.: Counter-in-Tweak: Authenticated Encryption Modes for Tweakable Block Ciphers. In: Robshaw, M., Katz, J. (eds.) Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9814, pp. 33–63. Springer (2016)

[80] Peyrin, T., Sim, S.M., Wang, L., Zhang, G.: Cryptanalysis of JAMBU. In: FSE. Lecture Notes in Computer Science, vol. 9054, pp. 264–281. Springer (2015)

[81] Rezvani, B., Diehl, W.: Hardware Implementations of NIST Lightweight Cryptographic Candidates: A First Look. IACR Cryptol. ePrint Arch. p. 824 (2019), https://eprint.iacr.org/2019/824

[82] Rogaway, P.: Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC. In: ASIACRYPT. Lecture Notes in Computer Science, vol. 3329, pp. 16–31. Springer (2004)

[83] Rogaway, P., Bellare, M., Black, J.: OCB: A block-cipher mode of operation for efficient authenticated encryption. ACM Trans. Inf. Syst. Secur. 6(3), 365–403 (2003)

[84] Rogaway, P., Bellare, M., Black, J., Krovetz, T.: OCB: a block-cipher mode of operation for efficient authenti-

cated encryption. In: ACM Conference on Computer and Communications Security. pp. 196–205. ACM (2001)

[85] Rogaway, P., Shrimpton, T.: A Provable-Security Treatment of the Key-Wrap Problem. In: Vaudenay, S. (ed.) EUROCRYPT. Lecture Notes in Computer Science, vol. 4004, pp. 373–390. Springer (2006)

[86] Rohit, R., Hu, K., Sarkar, S., Sun, S.: Misuse-Free Key-Recovery and Distinguishing Attacks on 7-Round Ascon. IACR Trans. Symmetric Cryptol. 2021(1), 130–155 (2021), https://doi.org/10.46586/tosc.v2021.i1.130-155

[87] Rohit, R., Sarkar, S.: Diving Deep into the Weak Keys of Round Reduced Ascon. IACR Trans. Symmetric Cryptol. 2021(4), 74–99 (2021), https://doi.org/10.46586/tosc.v2021.i4.74-99

[88] Roy, D., Mukhopadhyay, S.: Some results on ACORN. IACR Cryptol. ePrint Arch. 2016, 1132 (2016), https://eprint.iacr.org/2016/1132

[89] Salam, M.I., Bartlett, H., Dawson, E., Pieprzyk, J., Simpson, L., Wong, K.K.: Investigating Cube Attacks on the Authenticated Encryption Stream Cipher ACORN. In: Batten, L., Li, G. (eds.) Applications and Techniques in Information Security - 6th International Conference, ATIS 2016, Cairns, QLD, Australia, October 26-28, 2016, Proceedings. Communications in Computer and Information Science, vol. 651, pp. 15–26 (2016), https://doi.org/10.1007/978-981-10-2741-3_2

[90] Sasaki, Y.: Improved Related-Tweakey Boomerang Attacks on Deoxys-BC. In: Joux, A., Nitaj, A., Rachidi, T. (eds.) Progress in Cryptology - AFRICACRYPT 2018 - 10th International Conference on Cryptology in Africa, Marrakesh, Morocco, May 7-9, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10831, pp. 87–106. Springer (2018), https://doi.org/10.1007/978-3-319-89339-6_6

[91] Sasaki, Y., Todo, Y.: New Impossible Differential Search Tool from Design and Cryptanalysis Aspects - Revealing Structural Properties of Several Ciphers. In: Coron, J., Nielsen, J.B. (eds.) Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10212, pp. 185–215 (2017), https://doi.org/10.1007/978-3-319-56617-7_7

[92] Sasaki, Y., Todo, Y., Aoki, K., Naito, Y., Sugawara, T., Murakami, Y., Matsui, M., Hirose, S.: Minalpher v1.1, https://competitions.cr.yp.to/caesar-submissions.html

[93] Sasaki, Y., Todo, Y., Aoki, K., Naito, Y., Sugawara, T., Murakami, Y., Matsui, M., Hirose, S.: CAESAR candidates Minalpher. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[94] Song, L., Guo, J.: Cube-Attack-Like Cryptanalysis of Round-Reduced Keccak Using MILP. IACR Trans. Symmetric Cryptol. 2018(3), 182–214 (2018), https://doi.org/10.13154/tosc.v2018.i3.182-214

[95] Song, L., Guo, J., Shi, D., Ling, S.: New MILP Modeling: Improved Conditional Cube Attacks on Keccak-Based Constructions. In: Peyrin, T., Galbraith, S.D. (eds.) Advances in Cryptology - ASIACRYPT 2018 - 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2-6, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 11273, pp. 65–95. Springer (2018), https://doi.org/10.1007/978-3-030-03329-3_3

[96] Suzaki, T., Minematsu, K., Morioka, S., Kobayashi, E.: TWINE : A Lightweight Block Cipher for Multiple Platforms. In: Selected Areas in Cryptography. Lecture Notes in Computer Science, vol. 7707, pp. 339–354. Springer (2012)

[97] Todo, Y., Isobe, T., Hao, Y., Meier, W.: Cube Attacks on Non-Blackbox Polynomials Based on Division Property. In: Katz, J., Shacham, H. (eds.) Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10403, pp. 250–279. Springer (2017), https://doi.org/10.1007/978-3-319-63697-9_9

[98] Todo, Y., Isobe, T., Hao, Y., Meier, W.: Cube Attacks on Non-Blackbox Polynomials Based on Division Property. IEEE Trans. Computers 67(12), 1720–1736 (2018), https://doi.org/10.1109/TC.2018.2835480

[99] Todo, Y., Isobe, T., Meier, W., Aoki, K., Zhang, B.: Fast Correlation Attack Revisited - Cryptanalysis on

Full Grain-128a, Grain-128, and Grain-v1. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10992, pp. 129–159. Springer (2018), https://doi.org/10.1007/978-3-319-96881-0_5

[100] Vizár, D.: Ciphertext Forgery on HANUMAN. Cryptology ePrint Archive, Report 2016/697 (2016), https://eprint.iacr.org/2016/697

[101] Wang, Q., Hao, Y., Todo, Y., Li, C., Isobe, T., Meier, W.: Improved Division Property Based Cube Attacks Exploiting Algebraic Properties of Superpoly. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10991, pp. 275–305. Springer (2018), https://doi.org/10.1007/978-3-319-96884-1_10

[102] Wu, H.: CAESAR candidates Acorn + MORUS. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[103] Wu, H., Huang, T.: CAESAR candidates AEGIS + Jambu. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[104] Yalla, P., Kaps, J.: Evaluation of the CAESAR hardware API for lightweight implementations. In: International Conference on ReConFigurable Computing and FPGAs, ReConFig 2017, Cancun, Mexico, December 4-6, 2017. pp. 1–6. IEEE (2017), https://doi.org/10.1109/RECONFIG.2017.8279790

[105] Yang, J., Liu, M., Lin, D.: Cube Cryptanalysis of Round-Reduced ACORN. In: Lin, Z., Papamanthou, C., Polychronakis, M. (eds.) Information Security - 22nd International Conference, ISC 2019, New York City, NY, USA, September 16-18, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11723, pp. 44–64. Springer (2019), https://doi.org/10.1007/978-3-030-30215-3_3

[106] Zhang, P., Wang, P., Hu, H., Cheng, C., Kuai, W.: INT-RUP Security of Checksum-Based Authenticated Encryption. In: Okamoto, T., Yu, Y., Au, M.H., Li, Y. (eds.) Provable Security - 11th International Conference, ProvSec 2017, Xi'an, China, October 23-25, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10592, pp. 147–166. Springer (2017), https://doi.org/10.1007/978-3-319-68637-0_9

[107] Zhang, X., Lin, D.: Cryptanalysis of Acorn in Nonce-Reuse Setting. In: Chen, X., Lin, D., Yung, M. (eds.) Information Security and Cryptology - 13th International Conference, Inscrypt 2017, Xi'an, China, November 3-5, 2017, Revised Selected Papers. Lecture Notes in Computer Science, vol. 10726, pp. 342–361. Springer (2017), https://doi.org/10.1007/978-3-319-75160-3_21

[108] Zhao, B., Dong, X., Jia, K.: New Related-Tweakey Boomerang and Rectangle Attacks on Deoxys-BC Including BDT Effect. IACR Trans. Symmetric Cryptol. 2019(3), 121–151 (2019), https://doi.org/10.13154/tosc.v2019.i3.121-151

[109] Zhao, B., Dong, X., Jia, K., Meier, W.: Improved Related-Tweakey Rectangle Attacks on Reduced-Round Deoxys-BC-384 and Deoxys-I-256-128. In: Hao, F., Ruj, S., Gupta, S.S. (eds.) Progress in Cryptology - INDOCRYPT 2019 - 20th International Conference on Cryptology in India, Hyderabad, India, December 15-18, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11898, pp. 139–159. Springer (2019), https://doi.org/10.1007/978-3-030-35423-7_7

[110] Zhao, Z., Chen, S., Wang, M., Wang, W.: Improved cube-attack-like cryptanalysis of reduced-round Ketje-Jr and Keccak-MAC. Inf. Process. Lett. 171, 106124 (2021), https://doi.org/10.1016/j.ipl.2021.106124

[111] Zong, R., Dong, X.: MILP-Aided Related-Tweak/Key Impossible Differential Attack and its Applications to QARMA, Joltik-BC. IEEE Access 7, 153683–153693 (2019), https://doi.org/10.1109/ACCESS.2019.2946638

[112] Zong, R., Dong, X., Wang, X.: Collision Attacks on Round-Reduced Gimli-Hash/Ascon-Xof/Ascon-Hash. IACR Cryptol. ePrint Arch. 2019, 1115 (2019), https://eprint.iacr.org/2019/1115

[113] Zong, R., Dong, X., Wang, X.: Related-tweakey impossible differential attack on reduced-round Deoxys-BC-256.

Sci. China Inf. Sci. 62(3), 32102:1–32102:12 (2019), https://doi.org/10.1007/s11432-017-9382-2
[114] 伊藤竜馬: 「CRYPTREC 暗号技術ガイドライン（軽量暗号）」掲載の暗号方式に関する安全性評価の動向調査（文書番号：CRYPTREC EX-3101-2021）(2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf
[115] 崎山一男: 軽量暗号の実装性能に関する調査及び評価（NIST 軽量暗号コンペティションファイナリスト）（文書番号：CRYPTREC EX-3205-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3205-2022.pdf
[116] 藤堂洋介: 軽量暗号の安全性に関する調査及び評価（Ascon,Grain-128AEAD,TinyJambu）（文書番号：CRYPTREC EX-3203-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3203-2022.pdf

付録 A

Ascon の物理攻撃耐性

A.1 サイドチャネル攻撃対策手法

Ascon のサイドチャネル攻撃対策として有効な Threshold Implementation (TI) と Domain Oriented Masking (DOM) について概説する。

A.1.1 Threshold Implementation (TI)

TI は秘密分散法に基づくマスキング手法であり、2006 年に Nikova らによって提案された [11, 12]。TI において、計算対象の値 x は (x_1, x_2, x_3) のようにシェアと呼ばれる複数の値で表現される。

ここで、GF(2^m) 上の非線形変換 $z = \mathrm{N}(x, y)$ を考える。この際、GF(2^m) 上の非線形変換 $z = \mathrm{N}(x, y)$ に対しても、シェアの考え方を適用することができる。例えば、3 つの関数 f_1, f_2, f_3 が、以下に示す Non-Completeness（不完全性）、Correctness（正確性）、そして Uniformity（均一性）の性質を有する場合、z を 3 つのシェアに分け、2 次のプロービングモデル[*1]に耐性のある計算処理を実現できることが知られている。

Non-Completeness　各関数 f_1, f_2, f_3 は、入力変数 x, y の少なくとも 1 つのシェア値に依存しないよう、例えば、次のように計算する。

$$
\begin{aligned}
z_1 &= f_1(x_2, x_3, y_2, y_3), \\
z_2 &= f_2(x_3, x_1, y_3, y_1), \\
z_3 &= f_3(x_1, x_2, y_1, y_2).
\end{aligned}
$$

このように計算することで、2 つ以下のシェアに分けた関数の処理から、元の値 x, y の値に関する情報を知ることができない。

Correctness　各関数 f_1, f_2, f_3 は、以下の関係が満たされる場合に Correctness を満たす。

$$
\begin{aligned}
z &= z_1 \oplus z_2 \oplus z_3 \\
&= f_1(x_2, x_3, y_2, y_3) \oplus f_2(x_3, x_1, y_3, y_1) \oplus f_3(x_1, x_2, y_1, y_2) \\
&= \mathrm{N}(x, y).
\end{aligned}
$$

Uniformity　入力変数 x, y の発生確率は均一でなければならない。これは、発生確率に偏りが生じる場合、攻撃者はその偏りを利用することで、全てのシェア値を取得しなくとも正しい x, y の値を復元できる可能性があるからである。例えば、$m = 1$ の場合、つまり GF(2) の乗算において、入力変数 x の発生確率は以下を満たさなければならない。

$$
\Pr[x_1, x_2, x_3] = \frac{1}{8}.
$$

[*1] プロービングモデルとは、暗号化処理を行うハードウェアやソフトウェアに対し、攻撃者が本来観測することができない内部信号を 1 本あるいは複数のプローブ（針）を用いて観測可能とする攻撃者モデルである [8]。d 次プロービングモデルの場合、攻撃者は異なる d 本のプローブを用いて d 個の中間値を観測できると仮定する。ただし、同じ回路を使い回すシェア型のハードウェアアーキテクチャの場合、同じプローブで異なる時間の複数の中間値を取得することも想定できる [14]。

なお、関数が次数 t である場合、d 次のプロービングモデルに対してサイドチャネル攻撃耐性を持つためのシェアの数は、最小で $td+1$ であることが知られている。

A.1.2 Domain Oriented Masking (DOM)

DOM は d 次のプロービングモデルに対して耐性のあるマスキング手法であり、2016 年に Großらによって提案された [6, 7]。ドメインと呼ばれる概念を導入し、ドメインごとにシェアを構成することで、非線形演算によって増加するシェアの数を抑制することを可能にした。なお、d プロービングモデルへの耐性を実現するためには、変数ごとに $d+1$ 個のシェアを使用する必要があり、この場合におけるドメインの数は $d+1$ 個となる。

例えば、1 次プロービングモデルへの耐性を実現するために、変数 x, y のシェア $(x_0, y_0), (x_1, y_1)$ をそれぞれドメイン 0 とドメイン 1 に関連づける。また、GF(2^m) 上の 1 次安全な DOM 乗算器を考える。この時、TI と同様、入力値 x, y に関して以下の計算処理を実行する。

$$\begin{aligned} x \cdot y &= (x_0 \oplus x_1) \cdot (y_0 \oplus y_1) \\ &= x_0 \cdot y_0 \oplus x_0 \cdot y_1 \oplus x_1 \cdot y_0 \oplus x_1 \cdot y_1. \end{aligned}$$

ここで、‘·’ の演算記号は AND 演算を意味する。

$x_0 \cdot y_0$ の演算処理は、ドメイン 0 で安全に実行可能である。なぜならば、どの中間値（サイドチャネル情報）をプロービングによって読み出したとしても、入力値 x, y を復元することができないからである。同様に、$x_1 \cdot y_1$ の演算処理も、ドメイン 1 で安全に実行可能である。$x_0 \cdot y_1$ と $x_1 \cdot y_0$ の演算処理に関しても、入力値 x, y から独立しているため、これらの各処理からだけで入力値 x, y に関するサイドチャネルからの漏洩情報は観測できない。

一方、$x_0 \cdot y_1$ の演算処理をドメイン 0 に取り込み、$x_0 \cdot y_0 \oplus x_0 \cdot y_1$ を計算した場合には問題が生じる可能性がある。直接的ではないものの、異なるドメインのシェア y_0, y_1 が XOR で演算処理できるからである。これにより y の値が即座に復元できるわけではないものの、サイドチャネル情報に関する漏洩の危険性があると考えるべきである。そこで、$x_0 \cdot y_1$ と $x_1 \cdot y_0$ の演算処理は、いずれもクロスドメインで計算しなければならない処理とみなし、特定のドメイン 0 やドメイン 1 における演算処理とは切り離して考える必要がある。ここまでの手順が、DOM における計算（Calculation）ステップとなる。

次に、クロスドメインでの計算結果を特定のドメインに取り込むために、再シェア（Resharing）と呼ばれるステップを実行する。具体的には、$x_0 \cdot y_1$ と $x_1 \cdot y_0$ の演算処理後に、フレッシュな乱数 r でマスキングを行う。この再シェアでは、同じ乱数 r を使っても問題ないと知られている。つまり、$x_0 \cdot y_1 \oplus r$ と $x_1 \cdot y_0 \oplus r$ のようにマスキング処理を行うことができる。また、クロスドメインに関する一連の処理に起因して生じるグリッチの伝搬については、再シェアの結果をフリップフロップ回路に格納することで情報漏洩を抑止する。パイプライン処理では、ドメイン 0 やドメイン 0 における演算処理のタイミングを揃える必要があるため、$x_0 \cdot y_0$ と $y_0 \cdot y_1$ の計算結果もフリップフロップ回路に格納する。

最後に、統合（Integration）ステップでは、次のように特定のドメインとクロスドメインの演算結果の統合、つまり XOR 演算を行う。

$$\begin{aligned} q_0 &= (x_0 \cdot y_0) \oplus (x_0 \cdot y_1 \oplus r), \\ q_1 &= (x_1 \cdot y_1) \oplus (x_1 \cdot y_0 \oplus r). \end{aligned}$$

上記の 3 つのステップ（計算ステップ、再シェアステップ、統合ステップ）について、文献 [6] の Figure 1.1 又は文献 [7] の Figure 2 で概要図が示されているため、参考にされたい。

同様に、より高次（2 次以上）のプロービングモデルに耐性のある安全な DOM 乗算器も設計することが可能である。また、上述の 3 つのステップは、S-box などの非線形演算にも適用できる。

DOM の最大の特徴は、ドメイン単位の管理によってシェア数を適切に管理することができるとともに、再シェアステップにおいて使用する乱数を工夫することによってサイドチャネル攻撃対策にかかる実装コストの削減が期待できることである。TI による回路サイズの削減には数学的な処理の変換が必要になることが多いものの、DOM による設計手法は任意の回路に対して単純な処理ステップを繰り返すことで実現可能なため、設計の自動化が容易なマスキング手法であると言える。つまり、DOM によるマスキング実装の設計生産性は高い。なお、DOM によって生成された回路の実装コストは、最適化されていない TI 実装よりも低く、最適化された TI に匹敵する結果も得られている。

A.2 サイドチャネル解析・漏えい評価手法

Ascon に対するサイドチャネル解析・漏えい評価手法として報告されている相関電力解析（CPA: Correlation Power Analysis)、故障利用攻撃 (FA: Fault Attack)、Test Vector Leakage Assessment (TVLA) 、そしてテンプレート攻撃 (TA: Templete Attack）について概説する。

A.2.1 相関電力解析（CPA: Correlation Power Analysis）

相関電力解析は、電力のサイドチャネル情報を効率よく解析する方法として最もよく知られている [3]。なお、電磁波サイドチャネルに対する解析手法は、相関電磁波解析（CEMA: Correlation ElectroMagnetic Analysis）と呼ばれている。

差分電力解析（DPA: Differential Power Analysis） [9] では特定の 1 ビットに対する電力モデルが採用されるのに対し、相関電力解析では複数ビットの電力消費をモデル化するため、測定ノイズや処理アルゴリズムに起因するノイズの影響を軽減することが期待できる。また、差分電力解析では秘密鍵などの秘密情報の推測結果に基づいて電力波形データを 2 つのグループに分け、これら 2 つのデータの平均の差を解析するのに対し、相関電力解析では電力波形データをより多くのグループに分け、電力モデルとの相関関係を解析する。Ascon に限らず多くの暗号アルゴリズムにおいて、推測した秘密情報によってレジスタに格納される中間値の複数ビットを導出できる場合には、相関電力解析が最適である。

Ascon に対する既存の漏洩評価においても、相関電力解析を用いた安全性評価手法が採用されている。Ascon に対する相関電力解析では、認証暗号アルゴリズムの暗号化または復号処理において、攻撃者が秘密鍵を復元できるかどうかで評価されている。相関電力解析において中間値を導出するために選択関数（Selection Function）という概念を導入するが、この選択関数は初期化処理または最終処理（タグ生成処理）から構成される場合が多い。これは、初期化処理や最終処理に秘密鍵が直接関与しており、秘密鍵の予測によって中間値の予測が可能になるためである。選択関数の構成方法については、文献 [13] を参照されたい。

A.2.2 故障利用攻撃（FA: Fault Attack）

故障利用攻撃では、暗号機能を実装したハードウェアの動作中に故意に故障 (fault) を起こし、故障によって生じた計算誤りを利用して解析を行う手法である。故障利用解析の中でも差分故障解析 (DFA: Differential Fault Analysis) [2] が最もよく知られている解析手法の 1 つであり、正しい暗号文と誤りが生じた暗号文の差分を利用し、秘密鍵候補の探索空間を削減することで秘密鍵を推定する。

差分故障解析への有効な対策の 1 つとして、暗号化処理の二重化、つまり同じ暗号化処理を 2 回行い、その結果を比較することで誤った暗号文を出力しないといった対策が施される。しかし、Fault Sensitivity Analysis (FSA) [10] や Statistical Ineffective Fault Attack (SIFA) [4] といった高度な解析手法に対し、暗号化処理の二重化という単純な対策では不十分と言われている。

共通鍵暗号に対する最新の故障利用攻撃については、Baksi らによる SoK 論文 [1] を参照されたい。

A.2.3 Test Vector Leakage Assessment (TVLA)

ウェルチの t 検定（Welch's t-test）は様々な分野において幅広く利用されている統計的手法の 1 つであり、サイドチャネルからの漏洩評価における t 検定は、Test Vector Leakage Assessment (TVLA) と呼ばれている。TVLA の目的は、秘密鍵の復元や秘密情報の取得ではなく、暗号化処理デバイスの内部データとサイドチャネル情報との依存関係を評価し、潜在的な脆弱性を特定することにある。攻撃者の計算能力や攻撃手法に関係なく、暗号実装の安全性に関する汎用的な評価指標が提供できるツールとして、幅広く使用されるようになった。

具体的には、ある 2 つの基準に従って暗号アルゴリズムの処理を実行し、その際に測定したサイドチャネル情報の波形データをそれぞれデータセット A とデータセット B に分類する。これらのデータセットに含まれる各サンプル点に対し、

以下の式を用いて t 値を導出する。

$$t = \frac{\mu_A - \mu_B}{\sqrt{\frac{\sigma_A^2}{n_A} + \frac{\sigma_B^2}{n_B}}}.$$

ここで、μ、σ^2、n は、サンプル点における波形データ値の平均、分散、そして標本数であり、データセット A とデータセット B に対してそれぞれを求める。

サイドチャネルからの漏洩評価で一般的に使用されている評価基準は、秘密鍵を固定して、波形データのデータセットの 1 つを固定平文とし、もう 1 つをランダム平文とするものである [5]。$t < 4.5\sigma$ を満たす場合、そのサンプル点ではサイドチャネルからの漏洩がないものと判断される。

平易に表現すると、固定した平文の値に依存したサイドチャネルからの漏洩があるかを解析するものである。平文をランダムに入力した場合のサイドチャネル情報と比較してなんらかの差異が見られるならば、攻撃者はそのような情報を使用して内部の秘密情報を取得できる可能性があると判断する。つまり、t 検定においてサイドチャネルからの漏洩の可能性が示されたとしても、具体的な攻撃を実行できるかは不明であり、未知の攻撃を含めて安全性評価をより厳格に行う必要があると言える。

A.2.4 テンプレート攻撃（TA: Templete Attack）

テンプレート攻撃は、プロファイリングフェーズと攻撃フェーズから構成される。プロファイリングフェーズでは、攻撃対象と同種類のモジュールを使用し、入力値などのパラメータを操作しながら対象となるモジュールの特性を評価するフェーズである。攻撃フェーズでは、パラメータを操作できない攻撃対象モジュールに対して秘密鍵の推定を行うフェーズである。

上述のとおり、テンプレート攻撃の前提として、攻撃者は暗号アルゴリズムを処理するデバイスを完全に制御できなくてはならない。なぜならば、攻撃者が自由に平文や秘密鍵などの情報をデバイスに設定し、デバイスから漏洩したサイドチャネル情報の確率分布からデバイスの物理特性をプロファイリングしなければならないからである。つまり、テンプレート攻撃では、簡略化した電力モデルの代わりに、実際のデバイスから得られる物理的な振る舞いを用いて複雑なモデルを構築し、攻撃に利用する。攻撃者がデバイスから無制限に情報をプロファイリングすることができれば、測定ノイズを十分に削減することができるため、非常に強力な攻撃になりうる。

参考文献

[1] Anubhab Baksi, Shivam Bhasin, Jakub Breier, Dirmanto Jap, and Dhiman Saha. A Survey on Fault Attacks on Symmetric Key Cryptosystems. *ACM Comput. Surv.*, 55(4):86:1–86:34, 2023.

[2] Eli Biham and Adi Shamir. Differential Fault Analysis of Secret Key Cryptosystems. In Burton S. Kaliski Jr., editor, *Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 1997, Proceedings*, volume 1294 of *Lecture Notes in Computer Science*, pages 513–525. Springer, 1997.

[3] Eric Brier, Christophe Clavier, and Francis Olivier. Correlation Power Analysis with a Leakage Model. In Marc Joye and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems - CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2004.

[4] Christoph Dobraunig, Maria Eichlseder, Thomas Korak, Stefan Mangard, Florian Mendel, and Robert Primas. SIFA: Exploiting Ineffective Fault Inductions on Symmetric Cryptography. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2018(3):547–572, 2018.

[5] Gilbert Goodwill, Benjamin Jun, Josh Jaffe, and Pankaj Rohatgi. A testing methodology for side channel resistance validation, 2011. NIST, Non-Invasive Attack Testing Workshop. `https://csrc.nist.gov/CSRC/media/Events/Non-Invasive-Attack-Testing-Workshop/documents/08_Goodwill.pdf`.

[6] Hannes Groß. Domain-Oriented Masking–Generically Masked Hardware Implementations, 2018. PhD Thesis, IAIK, Graz University of Technology. `https://diglib.tugraz.at/download.php?id=5c80ea0c43a56&location=browse`（2023-10-07 閲覧）.

[7] Hannes Groß, Stefan Mangard, and Thomas Korak. Domain-Oriented Masking: Compact Masked Hardware Implementations with Arbitrary Protection Order. *IACR Cryptol. ePrint Arch.*, page 486, 2016.

[8] Yuval Ishai, Amit Sahai, and David A. Wagner. Private Circuits: Securing Hardware against Probing Attacks. In Dan Boneh, editor, *Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings*, volume 2729 of *Lecture Notes in Computer Science*, pages 463–481. Springer, 2003.

[9] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential Power Analysis. In Michael J. Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 388–397. Springer, 1999.

[10] Yang Li, Kazuo Sakiyama, Shigeto Gomisawa, Toshinori Fukunaga, Junko Takahashi, and Kazuo Ohta. Fault Sensitivity Analysis. In Stefan Mangard and François-Xavier Standaert, editors, *Cryptographic Hardware and Embedded Systems, CHES 2010, 12th International Workshop, Santa Barbara, CA, USA, August 17-20, 2010. Proceedings*, volume 6225 of *Lecture Notes in Computer Science*, pages 320–334. Springer, 2010.

[11] Svetla Nikova, Christian Rechberger, and Vincent Rijmen. Threshold Implementations Against Side-Channel Attacks and Glitches. In Peng Ning, Sihan Qing, and Ninghui Li, editors, *Information and Communications Security, 8th International Conference, ICICS 2006, Raleigh, NC, USA, December 4-7, 2006, Proceedings*, volume 4307 of *Lecture Notes in Computer Science*, pages 529–545. Springer, 2006.

[12] Svetla Nikova, Vincent Rijmen, and Martin Schläffer. Secure hardware implementation of nonlinear functions in the presence of glitches. *J. Cryptol.*, 24(2):292–321, 2011.

[13] Niels Samwel and Joan Daemen. DPA on hardware implementations of Ascon and Keyak. In *Proceedings of the Computing Frontiers Conference, CF'17, Siena, Italy, May 15-17, 2017*, pages 415–424. ACM, 2017.

[14] Takeshi Sugawara, Yang Li, and Kazuo Sakiyama. Probing attack of share-serial threshold implementation of advanced encryption standard. *IET Electronics Letters*, 55(9):517–519, 2019.

付録 B

CAESAR final portfolio: AEGIS, COLM

4.5 節の冒頭で述べたとおり、2016 年度版ガイドライン [2] では CAESAR final portfolio に選出された 6 方式のうち AEGIS-128 と COLM の 2 方式について掲載していない。これら 2 方式については軽量性を謳った方式ではないものの、CAESAR final portfolio に選出された方式であるということを鑑み、本節でこれらの方式の調査結果をまとめる。なお、調査結果については、2021 年度に公開された CRYPTREC 外部評価報告書 [8] に基づいて記載した。

技術分野	認証暗号
名称	AEGIS
設計者	Hongjun Wu[1], Bart Preneel[2] (1: Nanyang Technological University/Singapore, 2: KU Leuven/Belgium)
発表年	2013（SAC 2013 [7]）
仕様参照先	CAESAR ウェブサイト [1]、SAC 2013 [7]
特徴	AEGIS は、AEGIS-128L、AEGIS-128、AEGIS-256 の 3 種のバリエーションが提案されており、AEGIS-128 が final portfolio の Use Case 2 (High-performance Applications) に選出された。AEGIS-128 は、640 (128×5) ビットの内部状態を持ち、5 つの AES ラウンド関数を並列に実行することで内部状態を更新する。鍵長、nonce 長、タグ長はそれぞれ 128 ビットを推奨している。 なお、ソフト・ハード両面での高速性が特徴として挙げられる。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [4, 5, 6] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 単一鍵設定における最良の攻撃は、2019 年に提案された Eichlseder ら [4] による線形攻撃であり、AEGIS-256 に対して仕様段数であっても効率的に識別攻撃を実行できる。なお、AEGIS-128 に対しては仕様段数であっても 2^{132} から 2^{140} の範囲の計算量で識別攻撃を実行できる。弱鍵設定における最良の攻撃は、2021 年に提案された Liu ら [5] による積分攻撃であり、10 段のうち 5 段に簡略化した AEGIS-128 に対して、効率的に鍵回復攻撃と識別攻撃が実行できる。
主な実装評価結果	(SW) AEGIS-128、Intel Core i5-1030NG7 (Icelake 4×1.1 GHz) で 0.41 C/B。 (HW) AEGIS-128L、Virtex 6 で 1,025 slices、fmax 320.8 MHz。

技術分野	認証暗号
名称	COLM
設計者	Elena Andreeva[1], Andrey Bogdanov[2], Nilanjan Datta[3], Atul Luykx[1], Bart Mennink[1], Mridul Nandi[3], Elmar Tischhauser[2], Kan Yasuda[4] (1: KU Leuven/Belgium, 2: Technical University of Denmark/Denmark, 3: Indian Statistical Institute/India, 4: NTT/Japan)
発表年	2016（CAESAR ウェブサイト [1]）
仕様参照先	CAESAR ウェブサイト [1]
特徴	COLM は、COLM_0 と COLM_{127} の 2 種のバリエーションが提案されており、いずれも final portfolio の Use Case 3 (Defense in Depth) に選出された。当初、CAESAR submissions として AES-COPA と ELmD が投稿されたが、それぞれの長所を活かした形として COLM が設計された。 COLM は、ブロック暗号ベースの Encrypt-Linearmix-Encrypt 構造を採用しており、ブロック暗号として AES-128 を利用する。鍵長とタグ長は 128 ビット、nonce 長は 64 ビットが推奨されている。COLM_0 と COLM_{127} の主な違いはタグ生成の手順であり、COLM_{127} では暗号化処理の途中で中間タグ値を生成した後、これらの中間タグ値を用いてタグ生成が実行される。
安全性解析状況	2021 年 9 月現在、Datta ら [3] の他、目立った解析論文は発表されていない。 COLM タイプの認証暗号に対し、nonce-misuse シナリオと nonce-respecting シナリオにおける INT-RUP（タグ未検証において取得された平文の整合性）を考慮した攻撃について、2017 年に議論されている [3]。n をブロックサイズとすると、nonce-misuse シナリオにおいて暗号化・復号クエリが各 $4n$ 回、メッセージブロックサイズが $3n$ ブロックの場合に偽造攻撃が成立し、nonce-respecting シナリオにおいて暗号化クエリが 1 回、復号クエリが $2n$ 回、メッセージブロックサイズが $(n+1)n$ ブロックの場合に偽造攻撃が成立する。なお、これらの攻撃については COLM_{127} に影響しない。
主な実装評価結果	(SW) COLM_0、Intel Core i5-1030NG7 (Icelake 4 × 1.1 GHz) で 1.10 C/B。 (SW) COLM_{127}、Intel Core i5-1030NG7 (Icelake 4 × 1.1 GHz) で 30.06 C/B。 (HW) COLM_0、Virtex 6 で 2,060 slices、fmax 241.8 MHz。

参考文献

[1] Bernstein, D.J.: CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness, https://competitions.cr.yp.to/caesar.html（2023-10-04 閲覧）

[2] CRYPTREC 軽量暗号ワーキンググループ: CRYPTREC 暗号技術ガイドライン（軽量暗号）（文書番号：CRYPTREC GL-2003-2016JP）(2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf

[3] Datta, N., Luykx, A., Mennink, B., Nandi, M.: Understanding RUP Integrity of COLM. IACR Trans. Symmetric Cryptol. 2017(2), 143–161 (2017), https://doi.org/10.13154/tosc.v2017.i2.143-161

[4] Eichlseder, M., Nageler, M., Primas, R.: Analyzing the Linear Keystream Biases in AEGIS. IACR Trans. Symmetric Cryptol. 2019(4), 348–368 (2019), https://doi.org/10.13154/tosc.v2019.i4.348-368

[5] Liu, F., Isobe, T., Meier, W., Sakamoto, K.: Weak Keys in Reduced AEGIS and Tiaoxin. IACR Trans. Symmetric Cryptol. 2021(2), 104–139 (2021), https://doi.org/10.46586/tosc.v2021.i2.104-139

[6] Minaud, B.: Linear Biases in AEGIS Keystream. In: Joux, A., Youssef, A.M. (eds.) Selected Areas in Cryptography - SAC 2014 - 21st International Conference, Montreal, QC, Canada, August 14-15, 2014, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8781, pp. 290–305. Springer (2014), https://doi.org/10.1007/978-3-319-13051-4_18

[7] Wu, H., Preneel, B.: AEGIS: A Fast Authenticated Encryption Algorithm. In: Lange, T., Lauter, K.E., Lisonek, P. (eds.) Selected Areas in Cryptography - SAC 2013 - 20th International Conference, Burnaby, BC, Canada, August 14-16, 2013, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8282, pp. 185–201. Springer (2013), https://doi.org/10.1007/978-3-662-43414-7_10

[8] 伊藤竜馬:「CRYPTREC 暗号技術ガイドライン（軽量暗号）」掲載の暗号方式に関する安全性評価の動向調査（文書番号：CRYPTREC EX-3101-2021）（2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf

付録 C

NIST LWC ファイナリスト（Ascon を除く）

4.5 節の冒頭で述べたとおり、Ascon を除く NIST 軽量暗号（NIST LWC）プロジェクトのファイナリスト 9 方式（Elephant、GIFT-COFB、Grain-128AEAD、ISAP、PHOTON-Beetle、Romulus、Sparkle、TinyJAMBU、Xoodyak）についても軽量性の観点で優れており、かつ安全性の観点で問題が見つかっていない方式であることから、本節でこれらの方式の調査結果をまとめる。

各方式の仕様（設計者、発表年、仕様参照先、特徴）、安全性解析状況、そして主な実装評価結果については、2022 年度に公開された CRYPTREC 外部評価報告書 [149, 151, 152, 155, 156] に基づき、2022 年 9 月現在の調査結果を記載した。なお、文献 [152] で多くの実装評価結果がまとめられているものの、紙面の都合上、次の項目に限定している。ハードウェア実装評価結果については、FPGA 実装に着目し、回路面積の観点からコンパクト実装である結果、またはスループットの観点で高速実装である結果を抽出している。回路面積の評価尺度は、ルックアップテーブル数（LUTs）である。一部、紙面に余裕がある場合には、GlobalFoundries 社の GF 22nm CMOS で合成した ASIC 実装の評価結果も掲載している。評価尺度は FPGA 実装の場合と同じである。ソフトウェア実装評価結果については、IoT 向けローエンド CPU、特に Arm Cortex-M0 上での実装に着目し、設計者が作成したリファレンスコードを使用した場合のレイテンシ（暗号化・復号）、ROM サイズ、コードサイズの結果をまとめている。レイテンシの評価尺度は、テストベクトルを実行した際の 1 回の処理にかかる実行時間（msec）の平均値である。その他、文献 [152] では、ASIC 実装、命令拡張のハードウェア実装、ハイエンド CPU 上でのソフトウェア実装の結果がまとめられている。

技術分野	認証暗号
名称	Elephant
設計者	Tim Beyne[1], Yu Long Chen[1], Christoph Dobraunig[2], Bart Mennink[2] (1: KU Leuven/Belgium, 2: Radboud University/Netherlands)
発表年	2019（NIST LWC ウェブサイト [13]）
仕様参照先	NIST LWC ウェブサイト [26]、設計者ウェブサイト [25]
特徴	Elephant は暗号学的置換をプリミティブとして用いた認証暗号モードの名称であり、3 つの認証暗号 Dumbo、Jumbo、Delirium をまとめた総称である。認証暗号モードとしての構成は Enc-then-MAC 構造であり、暗号化部分は CTR モード、MAC 部分は Protected counter sum [21, 107] と同様の構成である。また、暗号化や MAC の内部構造は Masked Even-Mansour [66] を簡易にしたもので構成されている。NIST LWC プロジェクトのファイナリストのうち入力全体での並列化が可能な唯一の方式であるという特徴がある。 3 つの認証暗号は全てモード構成が Elephant であり、使用する暗号学的置換 P がそれぞれ異なる。各方式における鍵長、nonce 長、P のサイズ、タグ長、P の違いについては、下表のとおり。なお、設計者が推奨する方式は Dumbo である。

方式	鍵長	nonce 長	P のサイズ	タグ長	P
Dumbo	128	96	160	64	Spongent-π[160]
Jumbo	128	96	176	64	Spongent-π[176]
Delirium	128	96	200	128	Keccak-f[200]

安全性解析状況	2022 年 9 月現在、いくつかの解析論文 [4, 27, 28, 97, 122, 129, 135, 147, 153] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。 設計者 [26, 27] が Elephant モードの安全性証明を示しており、シングルユーザーとマルチユーザーのいずれの場合においても、仕様書で記載される安全性を担保することが示されている。Elephant モードへの第三者評価としては、2022 年に提案された土生ら [153] による鍵回復、識別及び偽造攻撃が提案されているものの、これらの攻撃は仕様書で主張される安全性バウンドがタイトであることを示す結果となっている。 暗号プリミティブへの安全性解析状況については、Keccak（4.3 節、文献 [29, 30, 58, 96, 112, 131]）と SPONGENT（4.3 節、文献 [1, 144]）の安全性解析状況を参照されたい。その他、耐量子安全性に関する解析論文 [4, 28, 122]、サイドチャネル攻撃耐性に関する解析論文 [97, 129, 135] が報告されているが、これらの安全性について設計者は主張していないため、仕様上の安全性とは矛盾しない。
主な実装評価結果	ハードウェア実装評価結果（FPGA 実装）

Algorithm	Platform	Data	Area	Throughput	Ref.
Dumbo	Artix-7	1,536 Bytes	1,291 LUTs	214.30 Mbps	[2]
Dumbo	Artix-7	Long	2,645 LUTs	1.54 Gbps	[113]

ハードウェア実装評価結果（ASIC 実装：GF 22nm CMOS）

Algorithm	Data	Area	Throughput	Ref.
Elephant	16 Bytes	17.3 kGE	24.0 Mbps	[62]
Elephant	1,536 Bytes	17.3 kGE	70.9 Mbps	[62]

ソフトウェア実装評価結果（Arm Cortex-M0）

Algorithm	Enc	Dec	ROM	Code	Ref.
Dumbo	1,069 msec	1,069 msec	16.4 Kbyte	14.4 Kbyte	[84]
Jumbo	1,255 msec	1,255 msec	16.4 Kbyte	14.4 Kbyte	[84]
Delirium	38.39 msec	38.39 msec	17.0 Kbyte	14.9 Kbyte	[84]

<table>
<tr><td>技術分野</td><td>認証暗号</td></tr>
<tr><td>名称</td><td>GIFT-COFB</td></tr>
<tr><td>設計者</td><td>Subhadeep Banik[1], Avik Chakraborti[2], Akiko Inoue[3], Tetsu Iwata[4], Kazuhiko Minematsu[3], Mridul Nandi[5], Thomas Peyrin[6], Yu Sasaki[7], Siang Meng Sim[6], Yosuke Todo[7]
(1: FHNW/Switzerland, 2: TCG CREST/India, 3: NEC Corporation/Japan, 4: Nagoya University/Japan, 5: Indian Statistical Institute/India, 6: Nanyang Technological University/Singapore, 7: NTT/Japan)</td></tr>
<tr><td>発表年</td><td>2019（NIST LWC ウェブサイト [13]）</td></tr>
<tr><td>仕様参照先</td><td>NIST LWC ウェブサイト [7]、NIST LWC メーリングリスト [6]、設計者ウェブサイト [5]</td></tr>
<tr><td>特徴</td><td>GIFT-COFB はブロック暗号 GIFT-128 [9] をプリミティブとして用いた暗号利用モード COFB [33, 34] に基づく認証暗号である。GIFT-128 は SPN 型ブロック暗号であり、鍵長とブロックサイズが 128 ビット、ラウンド関数を 40 段繰り返す構造を持つ。COFB は n ビットブロック暗号をプリミティブとして用いた認証暗号利用モードであり、実装サイズ、特にハードウェアゲートやソフトウェア上の動作メモリを最小化することに焦点を合わせて提案されたという特徴がある。
GIFT-COFB における推奨パラメータは、鍵長、nonce 長、タグ長がそれぞれ 128 ビットである。また、設計者が主張する安全性レベルは IND-CPA（選択平文攻撃に対する識別不可能性）が 64 ビット、INT-CTXT（暗号文の整合性）が 58 ビットである。</td></tr>
<tr><td>安全性解析状況</td><td>2022 年 9 月現在、いくつかの解析論文 [89, 98, 127, 128, 148] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。
認証暗号に対する第三者評価として、2022 年に Khairallah [98] は COFB モードの提案論文 [34] で主張される安全性バウンドに矛盾があることを指摘したが、これは仕様書で記載される安全性とは矛盾がない。また、2022 年に Inoue ら [89] は GIFT-COFB v1.1 [7] で主張される安全性バウンドに矛盾があることを指摘したが、最新版の GIFT-COFB v1.2 [6] において Inoue らの指摘が反映されており、仕様上の安全性とは矛盾しない。
暗号プリミティブに対する第三者評価として、Zong ら [148] による差分攻撃、Sun ら [127, 128] による線形攻撃が提案されているが、攻撃可能段数の最大値は 40 段のうち 27 段であり、安全性マージンが十分に確保されている。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果（FPGA 実装）
<table>
<tr><th>Platform</th><th>Data</th><th>Area</th><th>Throughput</th><th>Ref.</th></tr>
<tr><td>Artix-7</td><td>Long</td><td>1,041 LUTs</td><td>733.3 Mbps</td><td>[113]</td></tr>
<tr><td>Artix-7</td><td>1,536 Bytes</td><td>1,041 LUTs</td><td>364.3 Mbps</td><td>[113]</td></tr>
<tr><td>Artix-7</td><td>Long</td><td>1,730 LUTs</td><td>3.02 Gbps</td><td>[113]</td></tr>
<tr><td>Artix-7</td><td>1,536 Bytes</td><td>1,730 LUTs</td><td>1.48 Gbps</td><td>[113]</td></tr>
</table>
ハードウェア実装評価結果（ASIC 実装：GF 22nm CMOS）
<table>
<tr><th>Data</th><th>Area</th><th>Throughput</th><th>Ref.</th></tr>
<tr><td>16 Bytes</td><td>8.1 kGE</td><td>50.4 Mbps</td><td>[62]</td></tr>
<tr><td>1,536 Bytes</td><td>8.1 kGE</td><td>159.6 Mbps</td><td>[62]</td></tr>
</table>
ソフトウェア実装評価結果（Arm Cortex-M0）
<table>
<tr><th>Enc</th><th>Dec</th><th>ROM</th><th>Code</th><th>Ref.</th></tr>
<tr><td>6.25 msec</td><td>6.25 msec</td><td>17.1 Kbyte</td><td>15.1 Kbyte</td><td>[84]</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>技術分野</td><td>認証暗号</td></tr>
<tr><td>名称</td><td>Grain-128AEAD</td></tr>
<tr><td>設計者</td><td>Martin Hell[1], Thomas Johansson[1], Alexander Maximov[2], Willi Meier[3], Jonathan Sönnerup[1], Hirotaka Yoshida[4]
(1: Lund University/Sweden, 2: Ericsson AB/Sweden, 3: FHNW/Switzerland, 4: AIST/Japan)</td></tr>
<tr><td>発表年</td><td>2019（NIST LWC ウェブサイト [13]）</td></tr>
<tr><td>仕様参照先</td><td>NIST LWC ウェブサイト [81]、設計者ウェブサイト [80]</td></tr>
<tr><td>特徴</td><td>Grain は eSTREAM プロジェクトに応募された初期バージョンの Grain v0 [82] から始まり、Grain v1 [83]、Grain-128 [79]、Grain-128A [3] と、既知の脆弱性を補完 [20, 45, 47]、128 ビット安全性の確保、認証暗号モードの追加、などを経て系譜を継いできた暗号方式である。Grain-128AEAD もまたこの系譜を継ぐ認証暗号であり、Grain v1、Grain-128、Grain-128A（ストリーム暗号モードのみ）に対する既知の脆弱性 [133] に対策を施す形で提案された。また、NIST LWC 選考期間中、Grain-128AEAD の初期バージョンに対する脆弱性 [38] が指摘され、バージョン 2（Grain-128AEADv2）へと仕様が更新されている。
Grain-128AEADv2 は LFSR 型ストリーム暗号ベースの認証暗号であり、128 ビットの LFSR、128 ビットの NFSR、タグ生成用の 64 ビット Accumulator と 64 ビットレジスタから構成されている。鍵長は 128 ビット、nonce 長は 96 ビット、タグ長は 64 ビットであり、鍵と nonce をそれぞれ NFSR と LFSR にロードした後、512 段の初期化フェーズを経て内部状態を初期化する。初期化後、キーストリームを出力するが、奇数番目のキーストリームをタグ生成用として、偶数番目のキーストリームを暗号化用として利用する。</td></tr>
<tr><td>安全性解析状況</td><td>2022 年 9 月現在、Grain-128AEADv2 に対する解析論文は発表されていない。
藤堂 [155] は、Grain 型ストリーム暗号に対する強力な解読法として知られている高速相関攻撃とキューブ攻撃に着目し、Grain-128AEADv2 に対する高速相関攻撃 [20, 133] とキューブ攻撃 [46, 47, 76, 77, 78, 132, 138, 139] の適用可能性について考察した。結果として、最新の高速相関攻撃とキューブ攻撃に対し、Grain-128AEADv2 が十分に大きな安全性マージンを有していることを明らかにした。また、Grain-128A の初期化フェーズ（256 段）に対する条件付き差分攻撃 [101, 108] や関連鍵攻撃 [8, 44] が示されているものの、Grain-128AEADv2 の初期化フェーズは 512 段であり、これらの攻撃もまた安全性を脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果（FPGA 実装）
<table><tr><th>Platform</th><th>Area</th><th>Throughput</th><th>Ref.</th></tr><tr><td>Spartan-3</td><td>161 LUTs</td><td>152.2 Mbps</td><td>[150]</td></tr><tr><td>Spartan-6</td><td>174 LUTs</td><td>196.8 Mbps</td><td>[150]</td></tr></table>ハードウェア実装評価結果（ASIC 実装：GF 22nm CMOS）
<table><tr><th>Library</th><th>Data</th><th>Unroll</th><th>Area</th><th>Throughput</th><th>Power</th><th>Ref.</th></tr><tr><td>GF 22nm CMOS</td><td>16 Bytes</td><td>–</td><td>4.3 kGE</td><td>9.6 Mbps</td><td>–</td><td>[62]</td></tr><tr><td>GF 22nm CMOS</td><td>1,536 Bytes</td><td>–</td><td>4.3 kGE</td><td>17.7 Mbps</td><td>–</td><td>[62]</td></tr><tr><td>STM 65nm</td><td>–</td><td>1</td><td>2.6 kGE</td><td>1.25 Gbps</td><td>0.25mW</td><td>[125]</td></tr><tr><td>STM 65nm</td><td>–</td><td>64</td><td>16.9 kGE</td><td>33.6 Gbps</td><td>2.76mW</td><td>[125]</td></tr></table>ソフトウェア実装評価結果（Arm Cortex-M0）
<table><tr><th>Enc</th><th>Dec</th><th>ROM</th><th>Code</th><th>Ref.</th></tr><tr><td>82.54 msec</td><td>82.46 msec</td><td>17.8 Kbyte</td><td>15.8 Kbyte</td><td>[84]</td></tr></table></td></tr>
</table>

技術分野	認証暗号
名称	ISAP
設計者	Christoph Dobraunig[1], Maria Eichlseder[1], Stefan Mangard[1], Florian Mendel[2], Bart Mennink[3], Robert Primas[1], Thomas Unterluggauer[1] (1: Graz University of Technology, 2: Infineon Technologies AG/Germany, 3: Radboud University/Netherlands)
発表年	2017 (IACR ToSC 2017 [50])、2019（NIST LWC ウェブサイト [13]）
仕様参照先	NIST LWC ウェブサイト [49]、設計者ウェブサイト [48]
特徴	ISAP は暗号学的置換をプリミティブとして用いた認証暗号モードの名称であり、4 つの認証暗号 ISAP-A-128A、ISAP-K-128A、ISAP-A-128、ISAP-K-128 をまとめた総称である。認証暗号モードとしての構成は Sponge 構造を採用するとともに、Fresh Rekeying [110] と呼ばれる技術から着想を得てサイドチャネル攻撃に対して堅牢となるような設計であることが特徴的であり、Rekey 関数、暗号化関数、MAC 関数で構成されている。 4 つの認証暗号は全てモード構成が ISAP であり、使用する暗号学的置換、レートサイズ、各フェーズにおけるラウンド数などのパラメータがそれぞれ異なる。パラメータの違いについては、下表のとおり（k は安全性レベル（ビット）、その他の細部は仕様参照先を確認されたい）。なお、設計者が推奨する方式は ISAP-A-128A である。

方式	暗号学的置換	ビットサイズ				ラウンド数			
		k	n	r_H	r_B	s_H	s_B	s_E	s_K
ISAP-A-128A	Ascon-p	128	320	64	1	12	1	6	12
ISAP-K-128A	Keccak-p[400]	128	400	144	1	16	1	8	8
ISAP-A-128	Ascon-p	128	320	64	1	12	12	12	12
ISAP-K-128	Keccak-p[400]	128	400	144	1	20	12	12	12

安全性解析状況	2022 年 9 月現在、いくつかの解析論文 [14, 52, 53, 55, 71, 94, 134, 145] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。 Rekey 関数と暗号化関数の安全性については、Daemen ら [42] による Keyed duplex の安全性証明に依拠していることが設計者 [49] によって述べられている。MAC 関数の安全性については、Dobraunig ら [53, 55] によって Suffix keyed sponge の安全性証明が与えられており、128 ビット安全性がタイトであると述べられている。 暗号プリミティブへの安全性解析状況については、Keccak（4.3 節、文献 [29, 30, 58, 96, 112, 131]）と Ascon（4.5 節、文献 [12, 51, 63, 64, 85, 104, 109]）の安全性解析状況を参照されたい。その他、耐漏洩安全性に関する解析論文 [52, 53, 71]、サイドチャネル攻撃耐性に関する解析論文 [14, 134, 145]、量子識別攻撃 [94] に関する解析論文が報告されているが、これらの解析に関しても仕様上の安全性とは矛盾しない。
主な実装評価結果	ハードウェア実装評価結果（FPGA 実装）

Algorithm	Platform	Data	Area	Throughput	Ref.
ISAP-A-128A	Artix-7	1,536 Bytes	3,491 LUTs	389.4 Mbps	[113]
ISAP-A-128A	Artix-7	Long	3,491 LUTs	829.6 Mbps	[113]

ソフトウェア実装評価結果（Arm Cortex-M0）

Algorithm	Enc	Dec	ROM	Code	Ref.
ISAP-A-128A	9.66 msec	9.66 msec	16.5 Kbyte	14.5 Kbyte	[84]
ISAP-A-128	39.49 msec	39.50 msec	16.5 Kbyte	14.5 Kbyte	[84]
ISAP-K-128A	161.9 msec	161.9 msec	16.6 Kbyte	14.5 Kbyte	[84]
ISAP-K-128	1,366 msec	1,366 msec	16.6 Kbyte	14.5 Kbyte	[84]

<table>
<tr><td>技術分野</td><td>認証暗号、ハッシュ関数</td></tr>
<tr><td>名称</td><td>PHOTON-Beetle</td></tr>
<tr><td>設計者</td><td>Zhenzhen Bao[1], Avik Chakraborti[2], Nilanjan Datta[3], Jian Guo[1], Mridul Nandi[3], Thomas Peyrin[1], Kan Yasuda[4]
(1: Nanyang Technological University/Singapore., 2: University of Exeter/UK, 3: Indian Statistical Institute/India 4: NTT/Japan)</td></tr>
<tr><td>発表年</td><td>2019（NIST LWC ウェブサイト [13]）</td></tr>
<tr><td>仕様参照先</td><td>NIST LWC ウェブサイト [11]、設計者ウェブサイト [10]</td></tr>
<tr><td>特徴</td><td>PHOTON-Beetle は暗号学的置換をプリミティブとして用いた認証暗号 PHOTON-Beetle-AEAD[r] とハッシュ関数 PHOTON-Beetle-Hash をまとめた総称である。なお、パラメータ r はレートサイズ（ビット）を表す。これらの方式はハッシュ関数 PHOTON [72] で使用されている 256 ビットブロックの暗号学的置換を構成要素としている。また、認証暗号は Duplex Sponge [24] を改良した Beetle モード [32] に基づいて設計され、ハッシュ関数は Sponge 構造 [23] に基づいて設計されている。
認証暗号ではレートサイズとして $r = 32$ あるいは $r = 128$ を選択でき、いずれの選択においても鍵長、nonce 長、タグ長は 128 ビットである。なお、設計者が推奨する方式は $r = 128$ の PHOTON-Beetle-AEAD[128] である。ハッシュ関数ではレートサイズとして $r = 32$ 以外の場合を推奨しておらず、任意長の入力から 256 ビットのハッシュ値を出力する。</td></tr>
<tr><td>安全性解析状況</td><td>2022 年 9 月現在、いくつかの解析論文 [36, 37, 54, 89, 93, 100, 111] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。
設計者 [11] は認証暗号とハッシュ関数に関する安全性バウンドを示している。これらの安全性バウンドは第三者評価の結果 [54, 111] が反映されている。認証暗号に対する第三者評価として、2020 年に報告された Chakraborty ら [36, 37] による安全性証明、2020 年に提案された Dobraunig ら [54] による鍵回復攻撃、2022 年に提案された Inoue ら [89] による偽造攻撃と識別攻撃が示されているが、これらは仕様書で主張される安全性に矛盾がないことを示す結果となっている。Inoue ら [89] は関連鍵設定における効率的な偽造攻撃も示したが、この攻撃は限定的なシナリオでのみ成立するものであり、このシナリオが成立しないように実装することで回避できる。ハッシュ関数に対する第三者評価として、2021 年に提案された Mége [111] による衝突攻撃、2022 年に提案された Lefevre ら [100] による原像攻撃が示されているが、これらも仕様書で主張される安全性に矛盾がないことを示す結果となっている。
暗号プリミティブへの安全性解析状況については、PHOTON（4.3 節、文献 [95, 136, 137]）の安全性解析状況を参照されたい。その他、Jana ら [93] による効率的なサイドチャネル攻撃が示されているものの、この攻撃に対しても実装面での対策が有効である。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果（FPGA 実装）
<table>
<tr><th>Algorithm</th><th>Rate</th><th>Platform</th><th>Data</th><th>Area</th><th>Throughput</th><th>Ref.</th></tr>
<tr><td>認証暗号</td><td>128</td><td>Artix-7</td><td>1,536 Bytes</td><td>2,065 LUTs</td><td>370.4 Mbps</td><td>[113]</td></tr>
<tr><td>認証暗号</td><td>128</td><td>Artix-7</td><td>Long</td><td>2,065 LUTs</td><td>747.0 Mbps</td><td>[113]</td></tr>
<tr><td>ハッシュ関数</td><td>32</td><td>Artix-7</td><td>1,536 Bytes</td><td>2,065 LUTs</td><td>228.6 Mbps</td><td>[113]</td></tr>
<tr><td>ハッシュ関数</td><td>32</td><td>Artix-7</td><td>Long</td><td>2,065 LUTs</td><td>227.8 Mbps</td><td>[113]</td></tr>
</table>
ソフトウェア実装評価結果（Arm Cortex-M0）
<table>
<tr><th>Algorithm</th><th>Rate</th><th>Enc</th><th>Dec</th><th>ROM</th><th>Code</th><th>Ref.</th></tr>
<tr><td>認証暗号</td><td>128</td><td>42.39 msec</td><td>42.40 msec</td><td>17.5 Kbyte</td><td>15.4 Kbyte</td><td>[84]</td></tr>
<tr><td>認証暗号</td><td>32</td><td>102.6 msec</td><td>102.6 msec</td><td>17.6 Kbyte</td><td>15.5 Kbyte</td><td>[84]</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>技術分野</td><td>認証暗号、ハッシュ関数</td></tr>
<tr><td>名称</td><td>Romulus</td></tr>
<tr><td>設計者</td><td>Chun Guo[1], Tetsu Iwata[2], Mustafa Khairallah[3], Kazuhiko Minematsu[4], Thomas Peyrin[3]
(1: Shandong University/China, 2: Nagoya University/Japan, 3: Nanyang Technological University/Singapore, 4: NEC Corporation/Japan)</td></tr>
<tr><td>発表年</td><td>2019（NIST LWC ウェブサイト [13])、2020 (IACR ToSC 2020 [91])</td></tr>
<tr><td>仕様参照先</td><td>NIST LWC ウェブサイト [68]、設計者ウェブサイト [67]</td></tr>
<tr><td>特徴</td><td>Romulus は tweakable ブロック暗号（TBC）をプリミティブとして用いた暗号利用モードの名称であり、認証暗号 Romulus-N、Romulus-M、Romulus-T とハッシュ関数 Romulus-H をまとめた総称である。認証暗号の 3 方式はそれぞれ達成する安全性の種類が異なり、それに応じてモードの構成も変わることが特徴である。
Romulus-N は nonce-respecting 設定下で安全性が保証され、COFB モード [33, 34] をベースとしている。Romulus-M は nonce-misuse 設定下で安全性が保証され、SIV モード [119] をベースとしている。Romulus-T は耐漏洩安全性が保証され、TEDT モード [22] をベースとしている。3 方式とも nonce 長、鍵長、タグ長は 128 ビット、tweak 長は 256 ビットである。なお、設計者が推奨する方式は Romulus-N である。Romulus-H は MDPH 構造 [115] を採用しており、任意長の入力から 256 ビットのハッシュ値を出力する。
使用するプリミティブは Skinny-128-384+ であり、この方式は Skinny [19, 90] のインスタンスの 1 つである Skinny-128-384 の段数を 56 段から 40 段に削減したものである。</td></tr>
<tr><td>安全性解析状況</td><td>2022 年 9 月現在、いくつかの解析論文 [35, 56, 73, 99, 114, 145, 154] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。
設計者 [70, 91] は全モードの安全性証明を示している。認証暗号に対する第三者評価として、2020 年に報告された Lee [99] による安全性証明、2022 年に提案された Habu ら [73, 154] による識別攻撃、偽造攻撃、マッチング攻撃が示されているが、これらは仕様書で主張される安全性バウンドがタイトであることを示す結果となっている。また、ハッシュ関数に対する第三者評価として、2021 年に提案された Dong ら [56] による衝突攻撃と 2023 年に提案された Nageler ら [114] による衝突攻撃が示されているが、これらはプリミティブを簡略化した場合にのみ有効であり、ハッシュ関数の安全性を脅かすものではない。
プリミティブに対していくつかの解析論文 [31, 43, 56, 57, 74, 75, 87, 118, 124] が発表されているが、その仕様上の安全性を脅かす攻撃については提案されていない。オリジナルの Skinny-128-384 と仕様が異なるため、Skinny-128-384 に対する攻撃が必ずしも Skinny-128-384+ に適用できるとは限らない。設計者 [69] の分析によると、40 段のうち単一鍵設定では 22 段まで、関連鍵設定では 26 段まで鍵回復攻撃が可能であると見積もっている。ただし、これらは 256 ビット鍵を使用した場合における結果であることに注意されたい。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果（FPGA 実装）
<table>
<tr><th>Algorithm</th><th>Platform</th><th>Data</th><th>Area</th><th>Throughput</th><th>Ref.</th></tr>
<tr><td>Romulus-N</td><td>Artix-7</td><td>1,536 Bytes</td><td>1,280 LUTs</td><td>542.0 Mbps</td><td>[113]</td></tr>
<tr><td>Romulus-N</td><td>Artix-7</td><td>Long</td><td>1,280 LUTs</td><td>1.09 Gbps</td><td>[113]</td></tr>
<tr><td>Romulus-N</td><td>Artix-7</td><td>1,536 Bytes</td><td>953 LUTs</td><td>315.7 Mbps</td><td>[113]</td></tr>
<tr><td>Romulus-N</td><td>Artix-7</td><td>Long</td><td>953 LUTs</td><td>637.2 Mbps</td><td>[113]</td></tr>
</table>
ソフトウェア実装評価結果（Arm Cortex-M0）
<table>
<tr><th>Algorithm</th><th>Enc</th><th>Dec</th><th>ROM</th><th>Code</th><th>Ref.</th></tr>
<tr><td>Romulus-N</td><td>11.12 msec</td><td>11.13 msec</td><td>19.5 Kbyte</td><td>16.9 Kbyte</td><td>[84]</td></tr>
<tr><td>Romulus-M</td><td>14.48 msec</td><td>14.49 msec</td><td>19.7 Kbyte</td><td>17.1 Kbyte</td><td>[84]</td></tr>
</table></td></tr>
</table>

技術分野	認証暗号、ハッシュ関数
名称	Sparkle
設計者	Christof Beierle[1,2], Alex Biryukov[1], Luan Cardoso dos Santos[1], Johann Großschädl[1], Amir Moradi[2], Léo Perrin[3], Aein Rezaei Shahmirzadi[2], Aleksei Udovenko[1,4], Vesselin Velichkov[5], Qingju Wang[1] (1: University of Luxembourg/Luxembourg, 2: Ruhr University Bochum/Germany, 3: INRIA/France, 4: CryptoExperts/France, 5: University of Edinburgh/UK)
発表年	2019（NIST LWC ウェブサイト [13])、2020 (IACR ToSC 2020 [18])
仕様参照先	NIST LWC ウェブサイト [16]、設計者ウェブサイト [15]
特徴	Sparkle はブロック暗号 Alzette [17] を構成要素とした暗号学的置換であり、Sparkle を暗号プリミティブとして用いた Sponge 構造に基づく認証暗号 Schwaemm とハッシュ関数 Esch が定義されている。 Sparkle は Sparkle256、Sparkle384、Sparkle512 の 3 種類の暗号学的置換をまとめた総称である（数値：入出力サイズ)。それぞれ段数の異なる 2 つのインスタンス（slim、big）が定義されている。例えば、Sparkle384 の slim は 7 段、big は 11 段である。Schwaemm は nonce ベースの認証暗号であり、4 通りのパラメータが定義されている。設計者が推奨する方式は Sparkle384 を使用した Schwaemm256-128 であり、鍵長、タグ長、キャパシティが 128 ビット、nonce 長とレートが 256 ビット、主張する安全性が 120 ビットである。ハッシュ関数 Esch は Esch256 と Esch384 の 2 通りが定義されている。設計者が推奨する方式は Sparkle384 を使用した Esch256 であり、キャパシティが 128 ビット、レートが 256 ビット、任意長の入力から 256 ビットのハッシュ値を出力する。
安全性解析状況	2022 年 9 月現在、いくつかの解析論文 [88, 92, 105, 106, 117, 121, 126, 142, 143] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。 設計者 [16, 18] は Sparkle に対して広範な攻撃手法に対する安全性解析結果を示している。第三者評価として、2022 年に提案された Schrottenloher ら [121] による推測決定攻撃が示されているが、これは仕様上の安全性を脅かすものではない。Alzette に対する識別攻撃 [88, 105, 106, 117, 142] がいくつか発表されているものの、Alzette は Sparkle において S-box として機能しており、Alzette への識別攻撃が Sparkle の安全性を直ちに脅かすものではない。 認証暗号は Beetle モード [32] に基づいて設計されており、設計者 [16] はその安全性が Beetle の安全性バウンドに帰着できると主張している。また、ハッシュ関数の安全性は Hirose [86] が示す証明可能安全性の結果に基づいている。なお、これらの方式の安全性を脅かす第三者評価は発表されていない。岩田 [151] によると、設計者 [16] の安全性証明において一部精査が必要な箇所があるものの、大部分において問題とならないと結論付けられている。
主な実装評価結果	ハードウェア実装評価結果（FPGA 実装）

Algorithm	Platform	Data	Area	Throughput	Ref.
Schwaemm256-128	Artix-7	1,536 Bytes	3,071 LUTs	396.8 Mbps	[113]
Schwaemm256-128	Artix-7	Long	3,071 LUTs	831.2 Mbps	[113]
Esch256	Artix-7	1,536 Bytes	3,740 LUTs	481.2 Mbps	[113]
Esch256	Artix-7	Long	3,740 LUTs	489.4 Mbps	[113]

ソフトウェア実装評価結果（Arm Cortex-M0）

Algorithm	Enc	Dec	ROM	Code	Ref.
Schwaemm128-128	0.76 msec	0.77 msec	16.9 Kbyte	14.9 Kbyte	[84]
Schwaemm256-128	0.93 msec	0.93 msec	17.1 Kbyte	15.1 Kbyte	[84]

技術分野	認証暗号
名称	TinyJAMBU
設計者	Hongjun Wu, Tao Huang (Nanyang Technological University/Singapore)
発表年	2019（NIST LWC ウェブサイト [13]）
仕様参照先	NIST LWC ウェブサイト [141]
特徴	TinyJAMBU は鍵付き暗号学的置換をプリミティブとして用いた Sponge 構造に基づく認証暗号であり、TinyJAMBU-128、TinyJAMBU-192、TinyJAMBU-256 の 3 方式をまとめた総称である。CAESAR competition の第 3 ラウンド候補の 1 つである JAMBU [140] の軽量版として提案された。 鍵付き暗号学的置換の内部状態は 128 ビットの NFSR で構成されており、秘密鍵をロードしながら内部状態を更新する。また、認証暗号において 2 種類の暗号学的置換（$P1$、$P2$）を使用するが、これらは内部状態の更新回数（段数）が異なるのみである。3 種類の認証暗号方式におけるパラメータの違いについては、下表のとおり。TinyJAMBU は NIST LWC の最終ラウンドにおいて仕様が更新されたが、主な違いは $P1$ の段数であり、更新前の仕様段数は 384 段であった。なお、設計者が推奨する方式は TinyJAMBU-128 である。 方式 / 鍵長 / nonce 長 / タグ長 / $P1$ の段数 / $P2$ の段数 TinyJAMBU-128 / 128 / 96 / 64 / 640 / 1024 TinyJAMBU-192 / 192 / 96 / 64 / 640 / 1152 TinyJAMBU-256 / 256 / 96 / 64 / 640 / 1280
安全性解析状況	2022 年 9 月現在、いくつかの解析論文 [59, 60, 102, 120, 123, 130] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。 認証暗号に対する第三者評価として、2020 年に提案された Saha ら [120] の差分攻撃と線形攻撃、2022 年に提案された Li ら [102] の線形攻撃が示されている。Saha ら [120] の攻撃では $P1$ を 338 段に簡略化した場合に偽造攻撃が可能であり、この報告を受け設計者は $P1$ の段数を 640 段に修正した。Li ら [102] の攻撃では $P1$ を 387 段に簡略化した場合に鍵回復攻撃が可能であり、これは仕様修正前の TinyJAMBU が安全でなかったことを示している。最新の TinyJAMBU は安全性マージンが十分に確保されており、この攻撃が仕様上の安全性を脅かすものではない。 その他、2022 年に Sibleyras ら [123] は鍵付き暗号学的置換をブロック暗号とみなし、$P2$ の段数に関係なくスライド攻撃によって解読可能であることを示した。この攻撃はブロック暗号としての安全性を有していないことを示すものであり、認証暗号としての TinyJAMBU に対してこの攻撃は有効ではない。2022 年に Dunkelman ら [59, 60] は関連鍵設定において TinyJAMBU-192 と TinyJAMBU-256 に対する現実的な計算量での偽造攻撃が実行可能であることを示した。本攻撃が成立するような関連鍵（と nonce）の使用を避けることで安全性を確保できる。
主な実装評価結果	ハードウェア実装評価結果（FPGA 実装） Algorithm / Platform / Data / Area / Throughput / Ref. TinyJAMBU-128 / Artix-7 / 1,536 Bytes / 591 LUTs / 250.4 Mbps / [2] TinyJAMBU-128 / Artix-7 / 1,536 Bytes / 591 LUTs / 176.1 Mbps / [113] TinyJAMBU-128 / Artix-7 / Long / 591 LUTs / 354.7 Mbps / [113] ソフトウェア実装評価結果（Arm Cortex-M0） Algorithm / Enc / Dec / ROM / Code / Ref. TinyJAMBU-128 / 0.39 msec / 0.39 msec / 15.7 Kbyte / 13.7 Kbyte / [84] TinyJAMBU-192 / 4.63 msec / 4.63 msec / 15.7 Kbyte / 13.7 Kbyte / [84] TinyJAMBU-256 / 0.44 msec / 0.44 msec / 15.7 Kbyte / 13.7 Kbyte / [84]

技術分野	認証暗号、ハッシュ関数
名称	Xoodyak
設計者	Joan Daemen[1], Seth Hoffert[1,2], Michaël Peeters[2], Gilles Van Assche[2], Ronny Van Keer[2], Silvia Mella[1,2] (1: Radboud University/Netherlands, 2: STMicroelectronics/Switzerland)
発表年	2019（NIST LWC ウェブサイト [13]）
仕様参照先	NIST LWC ウェブサイト [41]、設計者ウェブサイト [40]
特徴	Xoodyak は暗号学的置換 Xoodoo [39] をプリミティブとして用いた Duplex 構造 [24, 42] に基づく認証暗号と Sponge 構造 [23] に基づくハッシュ関数の総称である。Xoodoo は Keccak-p [116] から着想を得て設計された暗号学的置換であり、ブロックサイズが 384 ビット、ラウンド関数を 12 段繰り返す構造を持つ。 認証暗号ではレートサイズが 256 ビット、鍵長、nonce 長、タグ長はそれぞれ 128 ビットが推奨されている。ハッシュ関数ではレートサイズが 128 ビットであり、任意長の入力から 256 ビットのハッシュ値を出力する。いずれも 128 ビット安全性が主張されている。
安全性解析状況	2022 年 9 月現在、いくつかの解析論文 [61, 103, 146] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。 認証暗号に関して、文献 [42] によると、推奨パラメータを使用した場合の安全性レベル 64 ビットであると証明されている。設計者の安全性主張を破る攻撃は存在しないが、安全性証明と設計者の主張に差があることには注意が必要である。その他、2020 年に提案された Zhou ら [146] による条件付きキューブ攻撃、2022 年に提案された Dunkelman ら [61] による差分線形攻撃があるが、7 段以上の Xoodoo に対して適用可能な攻撃が存在しないため、これらの攻撃が認証暗号の安全性を脅かすものではない。 ハッシュ関数に関して、文献 [23] の安全性証明と設計者の安全性主張に矛盾はない。その他、ハッシュ関数に対する第三者評価は報告されていない。 暗号プリミティブに対する第三者評価として、2020 年に提案された Liu ら [103] によるゼロサム識別攻撃がある。仕様段数の Xoodoo に対して 2^{33} の計算量でゼロサム識別子が構成可能であることが報告されているが、設計者 [41] が考察しているように、この攻撃が Xoodyak の安全性に直接影響を及ぼすものではない。 2023 年に Gilbert ら [65] は、Duplex ベースの認証暗号モードに対する汎用的な攻撃手法を提案した。この攻撃を Xoodyak に適用した場合、秘密鍵やタグの長さに依存せず、2^{148} の計算量で平文回復攻撃と偽造攻撃が可能となる。この攻撃は設計者 [41] が提供する安全性主張を破るものであるが、NIST が要求する 112 ビット安全性レベルを脅かすものではない。
主な実装評価結果	ハードウェア実装評価結果（FPGA 実装）

Algorithm	Platform	Data	Area	Throughput	Ref.
認証暗号	Artix-7	1,536 Bytes	1,808 LUTs	1.71 Gbps	[2]
認証暗号	Artix-7	1,536 Bytes	1,608 LUTs	2.89 Gbps	[113]
認証暗号	Artix-7	Long	1,608 LUTs	6.56 Gbps	[113]
ハッシュ関数	Artix-7	1,536 Bytes	1,608 LUTs	3.01 Gbps	[113]
ハッシュ関数	Artix-7	Long	1,608 LUTs	3.09 Gbps	[113]

ハードウェア実装評価結果（ASIC 実装：GF 22nm CMOS）

Algorithm	Data	Area	Throughput	Ref.
認証暗号	16 Bytes	11.9 kGE	79.2 Mbps	[62]
認証暗号	1,536 Bytes	11.9 kGE	1.02 Gbps	[62]

ソフトウェア実装評価結果（Arm Cortex-M0）

Algorithm	Enc	Dec	ROM	Code	Ref.
認証暗号	0.59 msec	0.60 msec	16.3 Kbyte	14.3 Kbyte	[84]

参考文献

[1] Abdelraheem, M.A.: Estimating the Probabilities of Low-Weight Differential and Linear Approximations on PRESENT-Like Ciphers. In: Kwon, T., Lee, M., Kwon, D. (eds.) Information Security and Cryptology - ICISC 2012 - 15th International Conference, Seoul, Korea, November 28-30, 2012, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7839, pp. 368–382. Springer (2012), https://doi.org/10.1007/978-3-642-37682-5_26

[2] Abdulgadir, A., Haeussler, R., Lin, S., Kaps, J.P., Gaj, K.: Side-Channel Resistant Implementations of Three Finalists of the NIST Lightweight Cryptography Standardization Process: Elephant, TinyJAMBU, and Xoodyak. Lightweight Cryptography Workshop 2022 pp. 1–7 (2022)

[3] Ågren, M., Hell, M., Johansson, T., Meier, W.: Grain-128a: a new version of Grain-128 with optional authentication. Int. J. Wirel. Mob. Comput. 5(1), 48–59 (2011), https://doi.org/10.1504/IJWMC.2011.044106

[4] Alagic, G., Bai, C., Katz, J., Majenz, C., Struck, P.: Post-Quantum Security of the (Tweakable) FX Construction, and Applications. IACR Cryptol. ePrint Arch. p. 1097 (2022), https://eprint.iacr.org/2022/1097

[5] Banik, S., Chakraborti, A., Inoue, A., Iwata, T., Minematsu, K., Nandi, M., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT-COFB Authenticated Encryption, https://www.isical.ac.in/~lightweight/COFB/（2023-10-04 閲覧）

[6] Banik, S., Chakraborti, A., Inoue, A., Iwata, T., Minematsu, K., Nandi, M., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT-COFB v1.2. Submission to the NIST Lightweight Cryptography project (2022), https://groups.google.com/a/list.nist.gov/g/lwc-forum/c/7BmjTeE-NsY?pli=1（2023-10-04 閲覧）

[7] Banik, S., Chakraborti, A., Iwata, T., Minematsu, K., Nandi, M., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT-COFB v1.1. Submission to the NIST Lightweight Cryptography project (2021)

[8] Banik, S., Maitra, S., Sarkar, S., Turan, M.S.: A Chosen IV Related Key Attack on Grain-128a. In: Boyd, C., Simpson, L. (eds.) Information Security and Privacy - 18th Australasian Conference, ACISP 2013, Brisbane, Australia, July 1-3, 2013. Proceedings. Lecture Notes in Computer Science, vol. 7959, pp. 13–26. Springer (2013), https://doi.org/10.1007/978-3-642-39059-3_2

[9] Banik, S., Pandey, S.K., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT: A Small Present - Towards Reaching the Limit of Lightweight Encryption. In: Fischer, W., Homma, N. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10529, pp. 321–345. Springer (2017), https://doi.org/10.1007/978-3-319-66787-4_16

[10] Bao, Z., Chakraborti, A., Datta, N., Guo, J., Nandi, M., Peyrin, T., Yasuda, K.: PHOTON-Beetle Authenticated Encryption, https://www.isical.ac.in/~lightweight/beetle/（2023-10-04 閲覧）

[11] Bao, Z., Chakraborti, A., Datta, N., Guo, J., Nandi, M., Peyrin, T., Yasuda, K.: PHOTON-Beetle Authenticated Encryption and Hash Family. Submission to the NIST Lightweight Cryptography project (2021)

[12] Bar-On, A., Dunkelman, O., Keller, N., Weizman, A.: DLCT: A New Tool for Differential-Linear Cryptanalysis. In: Ishai, Y., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11476, pp. 313–342. Springer (2019), https:

//doi.org/10.1007/978-3-030-17653-2_11

[13] Bassham, L., Chang, D., Kang, J., Kelsey, J., McKay, K., Turan, M.S., Waller, N.: NIST Lightweight Cryptography Project, https://csrc.nist.gov/projects/lightweight-cryptography

[14] Batina, L., Buhan, I., Chmielewski, L., Gunnarsdóttir, E., Jahandideh, V., Stock, T., Weissbart, L.: Side-Channel Evaluation Report on Implementations of Several NIST LWC Finalists (2022), https://cryptography.gmu.edu/athena/LWC/Reports/Radboud/Radboud_Report_SW_3_candidates.pdf（2023-10-04 閲覧）

[15] Beierle, C., Biryukov, A., dos Santos, L.C., Großschädl, J., Moradi, A., Perrin, L., Shahmirzadi, A.R., Udovenko, A., Velichkov, V., Wang, Q.: Sparkle Suite: A collection of lightweight symmetric cryptographic primitives, finalist of the ongoing NIST lightweight standardisation effort, https://sparkle-lwc.github.io/（2023-10-04 閲覧）

[16] Beierle, C., Biryukov, A., dos Santos, L.C., Großschädl, J., Moradi, A., Perrin, L., Shahmirzadi, A.R., Udovenko, A., Velichkov, V., Wang, Q.: Schwaemm and Esch: Lightweight Authenticated Encryption and Hashing using the Sparkle Permutation Family. Submission to the NIST Lightweight Cryptography project (2021)

[17] Beierle, C., Biryukov, A., dos Santos, L.C., Großschädl, J., Perrin, L., Udovenko, A., Velichkov, V., Wang, Q.: Alzette: A 64-Bit ARX-box - (Feat. CRAX and TRAX). In: Micciancio, D., Ristenpart, T. (eds.) Advances in Cryptology - CRYPTO 2020 - 40th Annual International Cryptology Conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17-21, 2020, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12172, pp. 419–448. Springer (2020), https://doi.org/10.1007/978-3-030-56877-1_15

[18] Beierle, C., Biryukov, A., dos Santos, L.C., Großschädl, J., Perrin, L., Udovenko, A., Velichkov, V., Wang, Q.: Lightweight AEAD and Hashing using the Sparkle Permutation Family. IACR Trans. Symmetric Cryptol. 2020(S1), 208–261 (2020), https://doi.org/10.13154/tosc.v2020.iS1.208-261

[19] Beierle, C., Jean, J., Kölbl, S., Leander, G., Moradi, A., Peyrin, T., Sasaki, Y., Sasdrich, P., Sim, S.M.: The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS. In: Robshaw, M., Katz, J. (eds.) Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II. Lecture Notes in Computer Science, vol. 9815, pp. 123–153. Springer (2016), https://doi.org/10.1007/978-3-662-53008-5_5

[20] Berbain, C., Gilbert, H., Maximov, A.: Cryptanalysis of Grain. In: Robshaw, M.J.B. (ed.) Fast Software Encryption, 13th International Workshop, FSE 2006, Graz, Austria, March 15-17, 2006, Revised Selected Papers. Lecture Notes in Computer Science, vol. 4047, pp. 15–29. Springer (2006), https://doi.org/10.1007/11799313_2

[21] Bernstein, D.J.: How to Stretch Random Functions: The Security of Protected Counter Sums. J. Cryptol. 12(3), 185–192 (1999), https://doi.org/10.1007/s001459900051

[22] Berti, F., Guo, C., Pereira, O., Peters, T., Standaert, F.: TEDT, a Leakage-Resist AEAD Mode for High Physical Security Applications. IACR Trans. Cryptogr. Hardw. Embed. Syst. 2020(1), 256–320 (2020), https://doi.org/10.13154/tches.v2020.i1.256-320

[23] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: On the Indifferentiability of the Sponge Construction. In: Smart, N.P. (ed.) Advances in Cryptology - EUROCRYPT 2008, 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Istanbul, Turkey, April 13-17, 2008. Proceedings. Lecture Notes in Computer Science, vol. 4965, pp. 181–197. Springer (2008), https://doi.org/10.1007/978-3-540-78967-3_11

[24] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: Duplexing the Sponge: Single-Pass Authenticated Encryption and Other Applications. In: Miri, A., Vaudenay, S. (eds.) Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7118, pp. 320–337. Springer (2011), https://doi.org/10.1007/978-3-642-28496-0_19

[25] Beyne, T., Chen, Y.L., Dobraunig, C., Mennink, B.: Elephant, https://www.esat.kuleuven.be/cosic/

elephant/（2023-10-04 閲覧）

[26] Beyne, T., Chen, Y.L., Dobraunig, C., Mennink, B.: Elephant v2.0. Submission to the NIST Lightweight Cryptography project (2021)

[27] Beyne, T., Chen, Y.L., Dobraunig, C., Mennink, B.: Multi-user Security of the Elephant v2 Authenticated Encryption Mode. In: AlTawy, R., Hülsing, A. (eds.) Selected Areas in Cryptography - 28th International Conference, SAC 2021, Virtual Event, September 29 - October 1, 2021, Revised Selected Papers. Lecture Notes in Computer Science, vol. 13203, pp. 155–178. Springer (2021), https://doi.org/10.1007/978-3-030-99277-4_8

[28] Bonnetain, X., Jaques, S.: Quantum Period Finding against Symmetric Primitives in Practice. IACR Trans. Cryptogr. Hardw. Embed. Syst. 2022(1), 1–27 (2022), https://doi.org/10.46586/tches.v2022.i1.1-27

[29] Boura, C., Canteaut, A.: Zero-sum distinguishers for iterated permutations and application to keccak-f and hamsi-256. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) Selected Areas in Cryptography - 17th International Workshop, SAC 2010, Waterloo, Ontario, Canada, August 12-13, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6544, pp. 1–17. Springer (2010), https://doi.org/10.1007/978-3-642-19574-7_1

[30] Boura, C., Canteaut, A., Cannière, C.D.: Higher-order differential properties of keccak and *Luffa*. In: Joux, A. (ed.) Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6733, pp. 252–269. Springer (2011), https://doi.org/10.1007/978-3-642-21702-9_15

[31] Boura, C., David, N., Derbez, P., Leander, G., Naya-Plasencia, M.: Differential Meet-In-The-Middle Cryptanalysis. IACR Cryptol. ePrint Arch. p. 1640 (2022), https://eprint.iacr.org/2022/1640

[32] Chakraborti, A., Datta, N., Nandi, M., Yasuda, K.: Beetle Family of Lightweight and Secure Authenticated Encryption Ciphers. IACR Trans. Cryptogr. Hardw. Embed. Syst. 2018(2), 218–241 (2018), https://doi.org/10.13154/tches.v2018.i2.218-241

[33] Chakraborti, A., Iwata, T., Minematsu, K., Nandi, M.: Blockcipher-Based Authenticated Encryption: How Small Can We Go? In: Fischer, W., Homma, N. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10529, pp. 277–298. Springer (2017), https://doi.org/10.1007/978-3-319-66787-4_14

[34] Chakraborti, A., Iwata, T., Minematsu, K., Nandi, M.: Blockcipher-Based Authenticated Encryption: How Small Can We Go? J. Cryptol. 33(3), 703–741 (2020), https://doi.org/10.1007/s00145-019-09325-z

[35] Chakraborty, A., Singh, N., Bhattacharya, S., Rebeiro, C., Mukhopadhyay, D.: Timed speculative attacks exploiting store-to-load forwarding bypassing cache-based countermeasures. In: Oshana, R. (ed.) DAC '22: 59th ACM/IEEE Design Automation Conference, San Francisco, California, USA, July 10 - 14, 2022. pp. 553–558. ACM (2022), https://doi.org/10.1145/3489517.3530493

[36] Chakraborty, B., Jha, A., Nandi, M.: On the Security of Sponge-type Authenticated Encryption Modes. IACR Cryptol. ePrint Arch. p. 1475 (2019), https://eprint.iacr.org/2019/1475

[37] Chakraborty, B., Jha, A., Nandi, M.: On the Security of Sponge-type Authenticated Encryption Modes. IACR Trans. Symmetric Cryptol. 2020(2), 93–119 (2020), https://doi.org/10.13154/tosc.v2020.i2.93-119

[38] Chang, D., Turan, M.S.: Recovering the Key from the Internal State of Grain-128AEAD. IACR Cryptol. ePrint Arch. p. 439 (2021), https://eprint.iacr.org/2021/439

[39] Daemen, J., Hoffert, S., Assche, G.V., Keer, R.V.: The design of Xoodoo and Xoofff. IACR Trans. Symmetric Cryptol. 2018(4), 1–38 (2018), https://doi.org/10.13154/tosc.v2018.i4.1-38

[40] Daemen, J., Hoffert, S., Peeters, M., Assche, G.V., Keer, R.V., Mella, S.: Team Keccak: Xoodyak, https://keccak.team/xoodyak.html（2023-10-04 閲覧）

[41] Daemen, J., Hoffert, S., Peeters, M., Assche, G.V., Keer, R.V., Mella, S.: Xoodyak, a lightweight cryptographic scheme. Submission to the NIST Lightweight Cryptography project (2021)

[42] Daemen, J., Mennink, B., Assche, G.V.: Full-State Keyed Duplex with Built-In Multi-user Support. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10625, pp. 606–637. Springer (2017), https://doi.org/10.1007/978-3-319-70697-9_21

[43] Delaune, S., Derbez, P., Vavrille, M.: Catching the Fastest Boomerangs Application to SKINNY. IACR Trans. Symmetric Cryptol. 2020(4), 104–129 (2020), https://doi.org/10.46586/tosc.v2020.i4.104-129

[44] Ding, L., Guan, J.: Related Key Chosen IV Attack on Grain-128a Stream Cipher. IEEE Trans. Inf. Forensics Secur. 8(5), 803–809 (2013), https://doi.org/10.1109/TIFS.2013.2256419

[45] Dinur, I., Güneysu, T., Paar, C., Shamir, A., Zimmermann, R.: An Experimentally Verified Attack on Full Grain-128 Using Dedicated Reconfigurable Hardware. In: Lee, D.H., Wang, X. (eds.) Advances in Cryptology - ASIACRYPT 2011 - 17th International Conference on the Theory and Application of Cryptology and Information Security, Seoul, South Korea, December 4-8, 2011. Proceedings. Lecture Notes in Computer Science, vol. 7073, pp. 327–343. Springer (2011), https://doi.org/10.1007/978-3-642-25385-0_18

[46] Dinur, I., Shamir, A.: Cube Attacks on Tweakable Black Box Polynomials. In: Joux, A. (ed.) Advances in Cryptology - EUROCRYPT 2009, 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings. Lecture Notes in Computer Science, vol. 5479, pp. 278–299. Springer (2009), https://doi.org/10.1007/978-3-642-01001-9_16

[47] Dinur, I., Shamir, A.: Breaking Grain-128 with Dynamic Cube Attacks. In: Joux, A. (ed.) Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6733, pp. 167–187. Springer (2011), https://doi.org/10.1007/978-3-642-21702-9_10

[48] Dobraunig, C., Eichlseder, M., Mangard, S., Mendel, F., Mennink, B., Primas, R., Unterluggauer, T.: ISAP: Lightweight Authenticated Encryption, https://isap.iaik.tugraz.at/index.html（2023-10-04 閲覧）

[49] Dobraunig, C., Eichlseder, M., Mangard, S., Mendel, F., Mennink, B., Primas, R., Unterluggauer, T.: ISAP v2.0. Submission to the NIST Lightweight Cryptography project (2021)

[50] Dobraunig, C., Eichlseder, M., Mangard, S., Mendel, F., Unterluggauer, T.: ISAP - Towards Side-Channel Secure Authenticated Encryption. IACR Trans. Symmetric Cryptol. 2017(1), 80–105 (2017), https://doi.org/10.13154/tosc.v2017.i1.80-105

[51] Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Cryptanalysis of Ascon. In: CT-RSA. Lecture Notes in Computer Science, vol. 9048, pp. 371–387. Springer (2015)

[52] Dobraunig, C., Mennink, B.: Leakage Resilience of the Duplex Construction. In: Galbraith, S.D., Moriai, S. (eds.) Advances in Cryptology - ASIACRYPT 2019 - 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8-12, 2019, Proceedings, Part III. Lecture Notes in Computer Science, vol. 11923, pp. 225–255. Springer (2019), https://doi.org/10.1007/978-3-030-34618-8_8

[53] Dobraunig, C., Mennink, B.: Security of the Suffix Keyed Sponge. IACR Trans. Symmetric Cryptol. 2019(4), 223–248 (2019), https://doi.org/10.13154/tosc.v2019.i4.223-248

[54] Dobraunig, C., Mennink, B.: Key Recovery Attack on PHOTON-Beetle. NIST lightweight-crypto mailing list, ROUND 2 OFFICIAL COMMENT: PHOTON-Beetle (2020)

[55] Dobraunig, C., Mennink, B.: Tightness of the Suffix Keyed Sponge Bound. IACR Trans. Symmetric Cryptol. 2020(4), 195–212 (2020), https://doi.org/10.46586/tosc.v2020.i4.195-212

[56] Dong, X., Hua, J., Sun, S., Li, Z., Wang, X., Hu, L.: Meet-in-the-Middle Attacks Revisited: Key-Recovery, Collision, and Preimage Attacks. In: Malkin, T., Peikert, C. (eds.) Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12827, pp. 278–308. Springer (2021), https://doi.org/10.

1007/978-3-030-84252-9_10

[57] Dong, X., Qin, L., Sun, S., Wang, X.: Key Guessing Strategies for Linear Key-Schedule Algorithms in Rectangle Attacks. In: Dunkelman, O., Dziembowski, S. (eds.) Advances in Cryptology - EUROCRYPT 2022 - 41st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Trondheim, Norway, May 30 - June 3, 2022, Proceedings, Part III. Lecture Notes in Computer Science, vol. 13277, pp. 3–33. Springer (2022), https://doi.org/10.1007/978-3-031-07082-2_1

[58] Duc, A., Guo, J., Peyrin, T., Wei, L.: Unaligned rebound attack: Application to keccak. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. Lecture Notes in Computer Science, vol. 7549, pp. 402–421. Springer (2012), https://doi.org/10.1007/978-3-642-34047-5_23

[59] Dunkelman, O., Ghosh, S., Lambooij, E.: Practical related-key forgery attacks on full-round tinyjambu-192/256. IACR Trans. Symmetric Cryptol. 2023(2), 176–188 (2023), https://doi.org/10.46586/tosc.v2023.i2.176-188

[60] Dunkelman, O., Lambooij, E., Ghosh, S.: Practical Related-Key Forgery Attacks on the Full TinyJAMBU-192/256. IACR Cryptol. ePrint Arch. p. 1122 (2022), https://eprint.iacr.org/2022/1122

[61] Dunkelman, O., Weizman, A.: Differential-Linear Cryptanalysis on Xoodyak. In: NIST LWC Workshop 2022 (2022), https://csrc.nist.gov/csrc/media/Events/2022/lightweight-cryptography-workshop-2022/documents/papers/differential-linear-cryptanalysis-on-xoodyak.pdf

[62] Elsadek, I., Aftabjahani, S., Gardner, D., MacLean, E., Wallrabenstein, J.R., Tawfik, E.Y.: Hardware and Energy Efficiency Evaluation of NIST Lightweight Cryptography Standardization Finalists. In: IEEE International Symposium on Circuits and Systems, ISCAS 2022, Austin, TX, USA, May 27 - June 1, 2022. pp. 133–137. IEEE (2022), https://doi.org/10.1109/ISCAS48785.2022.9937643

[63] Erlacher, J., Mendel, F., Eichlseder, M.: Bounds for the Security of Ascon against Differential and Linear Cryptanalysis. IACR Trans. Symmetric Cryptol. 2022(1), 64–87 (2022), https://doi.org/10.46586/tosc.v2022.i1.64-87

[64] Gérault, D., Peyrin, T., Tan, Q.Q.: Exploring differential-based distinguishers and forgeries for ASCON. IACR Cryptol. ePrint Arch. 2021, 1103 (2021), https://eprint.iacr.org/2021/1103, accepted to *IACR Trans. Symmetric Cryptol., 2021(3)*

[65] Gilbert, H., Boissier, R.H., Khati, L., Rotella, Y.: Generic attack on duplex-based AEAD modes using random function statistics. In: Hazay, C., Stam, M. (eds.) Advances in Cryptology - EUROCRYPT 2023 - 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part IV. Lecture Notes in Computer Science, vol. 14007, pp. 348–378. Springer (2023), https://doi.org/10.1007/978-3-031-30634-1_12

[66] Granger, R., Jovanovic, P., Mennink, B., Neves, S.: Improved Masking for Tweakable Blockciphers with Applications to Authenticated Encryption. In: Fischlin, M., Coron, J. (eds.) Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9665, pp. 263–293. Springer (2016), https://doi.org/10.1007/978-3-662-49890-3_11

[67] Guo, C., Iwata, T., Khairallah, M., Minematsu, K., Peyrin, T.: Romulus: Authenticated Encryption / Hash, https://romulusae.github.io/romulus/（2023-10-04 閲覧）

[68] Guo, C., Iwata, T., Khairallah, M., Minematsu, K., Peyrin, T.: Romulus v1.3. Submission to the NIST Lightweight Cryptography project (2021)

[69] Guo, C., Iwata, T., Khairallah, M., Minematsu, K., Peyrin, T.: Final-round updates on Romulus (2022), https://csrc.nist.gov/csrc/media/Projects/lightweight-cryptography/documents/finalist-round/status-updates/romulus-update.pdf（2023-10-04 閲覧）

[70] Guo, C., Iwata, T., Khairallah, M., Minematsu, K., Peyrin, T.: Security Proof for Romulus-T (2022), https://romulusae.github.io/romulus/docs/Romulus_T_proof.pdf（2023-10-04 閲覧）

[71] Guo, C., Pereira, O., Peters, T., Standaert, F.: Towards Low-Energy Leakage-Resistant Authenticated Encryption from the Duplex Sponge Construction. IACR Trans. Symmetric Cryptol. 2020(1), 6–42 (2020), https://doi.org/10.13154/tosc.v2020.i1.6-42

[72] Guo, J., Peyrin, T., Poschmann, A.: The PHOTON Family of Lightweight Hash Functions. In: Rogaway, P. (ed.) Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6841, pp. 222–239. Springer (2011), https://doi.org/10.1007/978-3-642-22792-9_13

[73] Habu, M., Minematsu, K., Iwata, T.: Matching attacks on Romulus-M. IET Inf. Secur. 16(6), 459–469 (2022), https://doi.org/10.1049/ise2.12075

[74] Hadipour, H., Bagheri, N., Song, L.: Improved Rectangle Attacks on SKINNY and CRAFT. IACR Trans. Symmetric Cryptol. 2021(2), 140–198 (2021), https://doi.org/10.46586/tosc.v2021.i2.140-198

[75] Hadipour, H., Sadeghi, S., Eichlseder, M.: Finding the Impossible: Automated Search for Full Impossible-Differential, Zero-Correlation, and Integral Attacks. In: Hazay, C., Stam, M. (eds.) Advances in Cryptology - EUROCRYPT 2023 - 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part IV. Lecture Notes in Computer Science, vol. 14007, pp. 128–157. Springer (2023), https://doi.org/10.1007/978-3-031-30634-1_5

[76] Hao, Y., Jiao, L., Li, C., Meier, W., Todo, Y., Wang, Q.: Links between Division Property and Other Cube Attack Variants. IACR Trans. Symmetric Cryptol. 2020(1), 363–395 (2020), https://doi.org/10.13154/tosc.v2020.i1.363-395

[77] Hao, Y., Leander, G., Meier, W., Todo, Y., Wang, Q.: Modeling for Three-Subset Division Property Without Unknown Subset - Improved Cube Attacks Against Trivium and Grain-128AEAD. In: Canteaut, A., Ishai, Y. (eds.) Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12105, pp. 466–495. Springer (2020), https://doi.org/10.1007/978-3-030-45721-1_17

[78] Hao, Y., Leander, G., Meier, W., Todo, Y., Wang, Q.: Modeling for Three-Subset Division Property without Unknown Subset. J. Cryptol. 34(3), 22 (2021), https://doi.org/10.1007/s00145-021-09383-2

[79] Hell, M., Johansson, T., Maximov, A., Meier, W.: A Stream Cipher Proposal: Grain-128. In: Proceedings 2006 IEEE International Symposium on Information Theory, ISIT 2006, The Westin Seattle, Seattle, Washington, USA, July 9-14, 2006. pp. 1614–1618. IEEE (2006), https://doi.org/10.1109/ISIT.2006.261549

[80] Hell, M., Johansson, T., Maximov, A., Meier, W., Sönnerup, J., Yoshida, H.: Grain-128AEAD, https://grain-128aead.github.io/（2023-10-04 閲覧）

[81] Hell, M., Johansson, T., Maximov, A., Meier, W., Sönnerup, J., Yoshida, H.: Grain-128AEADv2 – A lightweight AEAD stream ciphe. Submission to the NIST Lightweight Cryptography project (2021)

[82] Hell, M., Johansson, T., Meier, W.: Grain – A Stream Cipher for Constrained Environments (2005), https://www.ecrypt.eu.org/stream/

[83] Hell, M., Johansson, T., Meier, W.: Grain – A Stream Cipher for Constrained Environments. Int. J. Wirel. Mob. Comput. 2(1), 86–93 (2007), https://doi.org/10.1504/IJWMC.2007.013798

[84] Hira, R., Kitahara, T., Miyahara, D., Hara-Azumi, Y., Li, Y., Sakiyama, K.: Software Evaluation for Second Round Candidates in NIST Lightweight Cryptography. IACR Cryptol. ePrint Arch. p. 591 (2022), https://eprint.iacr.org/2022/591

[85] Hirch, S.E., Mella, S., Mehrdad, A., Daemen, J.: Improved Differential and Linear Trail Bounds for ASCON. IACR Trans. Symmetric Cryptol. 2022(4), 145–178 (2022), https://doi.org/10.46586/tosc.v2022.

i4.145-178

[86] Hirose, S.: Sequential Hashing with Minimum Padding. Cryptogr. 2(2), 11 (2018), https://doi.org/10.3390/cryptography2020011

[87] Hua, J., Liu, T., Cui, Y., Qin, L., Dong, X., Cui, H.: Low-Data Cryptanalysis On SKINNY Block Cipher. Comput. J. 66(4), 970–986 (2023), https://doi.org/10.1093/comjnl/bxab208

[88] Huang, M., Xu, Z., Wang, L.: On the Probability and Automatic Search of Rotational-XOR Cryptanalysis on ARX Ciphers. Comput. J. 65(12), 3062–3080 (2022), https://doi.org/10.1093/comjnl/bxab126

[89] Inoue, A., Iwata, T., Minematsu, K.: Analyzing the Provable Security Bounds of GIFT-COFB and Photon-Beetle. In: Ateniese, G., Venturi, D. (eds.) Applied Cryptography and Network Security - 20th International Conference, ACNS 2022, Rome, Italy, June 20-23, 2022, Proceedings. Lecture Notes in Computer Science, vol. 13269, pp. 67–84. Springer (2022), https://doi.org/10.1007/978-3-031-09234-3_4

[90] ISO/IEC: Information security – Encryption algorithms – Part 7: Tweakable block ciphers (ISO/IEC 18033-7:2022), https://www.iso.org/standard/80505.html

[91] Iwata, T., Khairallah, M., Minematsu, K., Peyrin, T.: Duel of the Titans: The Romulus and Remus Families of Lightweight AEAD Algorithms. IACR Trans. Symmetric Cryptol. 2020(1), 43–120 (2020), https://doi.org/10.13154/tosc.v2020.i1.43-120

[92] Jagielski, A., Kanciak, K.: Grover on sparkle quantum resource estimation for a NIST LWC call finalist. Quantum Inf. Comput. 22(13&14), 1132–1143 (2022), https://doi.org/10.26421/QIC22.13-14-3

[93] Jana, A., Paul, G.: Differential Fault Attack on PHOTON-Beetle. In: Chang, C., Rührmair, U., Mukhopadhyay, D., Forte, D. (eds.) Proceedings of the 2022 Workshop on Attacks and Solutions in Hardware Security, ASHES 2022, Los Angeles, CA, USA, 11 November 2022. pp. 25–34. ACM (2022), https://doi.org/10.1145/3560834.3563824

[94] Janson, C., Struck, P.: Sponge-Based Authenticated Encryption: Security Against Quantum Attackers. In: Cheon, J.H., Johansson, T. (eds.) Post-Quantum Cryptography - 13th International Workshop, PQCrypto 2022, Virtual Event, September 28-30, 2022, Proceedings. Lecture Notes in Computer Science, vol. 13512, pp. 230–259. Springer (2022), https://doi.org/10.1007/978-3-031-17234-2_12

[95] Jean, J., Naya-Plasencia, M., Peyrin, T.: Improved Rebound Attack on the Finalist Grøstl. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. Lecture Notes in Computer Science, vol. 7549, pp. 110–126. Springer (2012), https://doi.org/10.1007/978-3-642-34047-5_7

[96] Jean, J., Nikolic, I.: Internal differential boomerangs: Practical analysis of the round-reduced keccak- f f permutation. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9054, pp. 537–556. Springer (2015), https://doi.org/10.1007/978-3-662-48116-5_26

[97] Joshi, P., Mazumdar, B.: Single Event Transient Fault Analysis of ELEPHANT cipher. CoRR abs/2106.09536 (2021), https://arxiv.org/abs/2106.09536

[98] Khairallah, M.: Security of COFB against Chosen Ciphertext Attacks. IACR Trans. Symmetric Cryptol. 2022(1), 138–157 (2022), https://doi.org/10.46586/tosc.v2022.i1.138-157

[99] Lee, J.: Security evaluation of romulus, https://romulusae.github.io/romulus/docs/Security_evaluation_Romulus_Jooyoung_Lee.pdf（2023-10-04 閲覧）

[100] Lefevre, C., Mennink, B.: Tight Preimage Resistance of the Sponge Construction. In: Dodis, Y., Shrimpton, T. (eds.) Advances in Cryptology - CRYPTO 2022 - 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15-18, 2022, Proceedings, Part IV. Lecture Notes in Computer Science, vol. 13510, pp. 185–204. Springer (2022), https://doi.org/10.1007/978-3-031-15985-5_7

[101] Lehmann, M., Meier, W.: Conditional Differential Cryptanalysis of Grain-128a. In: Pieprzyk, J., Sadeghi, A.,

Manulis, M. (eds.) Cryptology and Network Security, 11th International Conference, CANS 2012, Darmstadt, Germany, December 12-14, 2012. Proceedings. vol. 7712, pp. 1–11. Springer (2012), https://doi.org/10.1007/978-3-642-35404-5_1

[102] Li, M., Mouha, N., Sun, L., Wang, M.: Revisiting the Extension of Matsui's Algorithm 1 to Linear Hulls: Application to TinyJAMBU. IACR Trans. Symmetric Cryptol. 2022(2), 161–200 (2022), https://doi.org/10.46586/tosc.v2022.i2.161-200

[103] Liu, F., Isobe, T., Meier, W., Yang, Z.: Algebraic Attacks on Round-Reduced Keccak/Xoodoo. IACR Cryptol. ePrint Arch. p. 346 (2020), https://eprint.iacr.org/2020/346

[104] Liu, M., Lu, X., Lin, D.: Differential-Linear Cryptanalysis from an Algebraic Perspective. In: Malkin, T., Peikert, C. (eds.) Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12827, pp. 247–277. Springer (2021), https://doi.org/10.1007/978-3-030-84252-9_9

[105] Liu, Y., Niu, Z., Sun, S., Li, C., Hu, L.: Rotational Differential-Linear Cryptanalysis Revisited. J. Cryptol. 36(1), 3 (2023), https://doi.org/10.1007/s00145-022-09440-4

[106] Liu, Y., Sun, S., Li, C.: Rotational Cryptanalysis from a Differential-Linear Perspective - Practical Distinguishers for Round-Reduced FRIET, Xoodoo, and Alzette. In: Canteaut, A., Standaert, F. (eds.) Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12696, pp. 741–770. Springer (2021), https://doi.org/10.1007/978-3-030-77870-5_26

[107] Luykx, A., Preneel, B., Tischhauser, E., Yasuda, K.: A MAC Mode for Lightweight Block Ciphers. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 43–59. Springer (2016), https://doi.org/10.1007/978-3-662-52993-5_3

[108] Ma, Z., Tian, T., Qi, W.: Conditional differential attacks on Grain-128a stream cipher. IET Inf. Secur. 11(3), 139–145 (2017), https://doi.org/10.1049/iet-ifs.2016.0060

[109] Makarim, R.H., Rohit, R.: Towards Tight Differential Bounds of Ascon A Hybrid Usage of SMT and MILP. IACR Trans. Symmetric Cryptol. 2022(3), 303–340 (2022), https://doi.org/10.46586/tosc.v2022.i3.303-340

[110] Medwed, M., Standaert, F., Großschädl, J., Regazzoni, F.: Fresh Re-keying: Security against Side-Channel and Fault Attacks for Low-Cost Devices. In: Bernstein, D.J., Lange, T. (eds.) Progress in Cryptology - AFRICACRYPT 2010, Third International Conference on Cryptology in Africa, Stellenbosch, South Africa, May 3-6, 2010. Proceedings. Lecture Notes in Computer Science, vol. 6055, pp. 279–296. Springer (2010), https://doi.org/10.1007/978-3-642-12678-9_17

[111] Mége, A.: Official comment: PHOTON-Beetle. NIST lightweight-crypto mailing list, ROUND 2 OFFICIAL COMMENT: PHOTON-Beetle (2021)

[112] Mella, S., Daemen, J., Assche, G.V.: New techniques for trail bounds and application to differential trails in keccak. IACR Trans. Symmetric Cryptol. 2017(1), 329–357 (2017), https://doi.org/10.13154/tosc.v2017.i1.329-357

[113] Mohajerani, K., Haeussler, R., Nagpal, R., Farahmand, F., Abdulgadir, A., Kaps, J., Gaj, K.: FPGA Benchmarking of Round 2 Candidates in the NIST Lightweight Cryptography Standardization Process: Methodology, Metrics, Tools, and Results. IACR Cryptol. ePrint Arch. p. 1207 (2020), https://eprint.iacr.org/2020/1207

[114] Nageler, M., Pallua, F., Eichlseder, M.: Finding Collisions for Round-Reduced Romulus-H. IACR Trans. Symmetric Cryptol. 2023(1), 67–88 (2023), https://doi.org/10.46586/tosc.v2023.i1.67-88

[115] Naito, Y.: Optimally Indifferentiable Double-Block-Length Hashing Without Post-processing and with Support for Longer Key Than Single Block. In: Schwabe, P., Thériault, N. (eds.) Progress in Cryptology - LATINCRYPT 2019 - 6th International Conference on Cryptology and Information Security in Latin America, Santiago de Chile,

Chile, October 2-4, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11774, pp. 65–85. Springer (2019), https://doi.org/10.1007/978-3-030-30530-7_4

[116] National Institute of Standards and Technology: FIPS 202 – SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions, https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf

[117] Niu, Z., Sun, S., Liu, Y., Li, C.: Rotational Differential-Linear Distinguishers of ARX Ciphers with Arbitrary Output Linear Masks. In: Dodis, Y., Shrimpton, T. (eds.) Advances in Cryptology - CRYPTO 2022 - 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15-18, 2022, Proceedings, Part I. Lecture Notes in Computer Science, vol. 13507, pp. 3–32. Springer (2022), https://doi.org/10.1007/978-3-031-15802-5_1

[118] Qin, L., Dong, X., Wang, X., Jia, K., Liu, Y.: Automated Search Oriented to Key Recovery on Ciphers with Linear Key Schedule Applications to Boomerangs in SKINNY and ForkSkinny. IACR Trans. Symmetric Cryptol. 2021(2), 249–291 (2021), https://doi.org/10.46586/tosc.v2021.i2.249-291

[119] Rogaway, P., Shrimpton, T.: A Provable-Security Treatment of the Key-Wrap Problem. In: Vaudenay, S. (ed.) EUROCRYPT. Lecture Notes in Computer Science, vol. 4004, pp. 373–390. Springer (2006)

[120] Saha, D., Sasaki, Y., Shi, D., Sibleyras, F., Sun, S., Zhang, Y.: On the Security Margin of TinyJAMBU with Refined Differential and Linear Cryptanalysis. IACR Trans. Symmetric Cryptol. 2020(3), 152–174 (2020), https://doi.org/10.13154/tosc.v2020.i3.152-174

[121] Schrottenloher, A., Stevens, M.: Simplified MITM Modeling for Permutations: New (Quantum) Attacks. In: Dodis, Y., Shrimpton, T. (eds.) Advances in Cryptology - CRYPTO 2022 - 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15-18, 2022, Proceedings, Part III. Lecture Notes in Computer Science, vol. 13509, pp. 717–747. Springer (2022), https://doi.org/10.1007/978-3-031-15982-4_24

[122] Shi, T., Wu, W., Hu, B., Guan, J., Wang, S.: Breaking LWC candidates: sESTATE and Elephant in quantum setting. Des. Codes Cryptogr. 89(7), 1405–1432 (2021), https://doi.org/10.1007/s10623-021-00875-7

[123] Sibleyras, F., Sasaki, Y., Todo, Y., Hosoyamada, A., Yasuda, K.: Birthday-Bound Slide Attacks on TinyJAMBU's Keyed-Permutations for All Key Sizes. In: Cheng, C., Akiyama, M. (eds.) Advances in Information and Computer Security - 17th International Workshop on Security, IWSEC 2022, Tokyo, Japan, August 31 - September 2, 2022, Proceedings. Lecture Notes in Computer Science, vol. 13504, pp. 107–127. Springer (2022), https://doi.org/10.1007/978-3-031-15255-9_6

[124] Song, L., Zhang, N., Yang, Q., Shi, D., Zhao, J., Hu, L., Weng, J.: Optimizing Rectangle Attacks: A Unified and Generic Framework for Key Recovery. In: Agrawal, S., Lin, D. (eds.) Advances in Cryptology - ASIACRYPT 2022 - 28th International Conference on the Theory and Application of Cryptology and Information Security, Taipei, Taiwan, December 5-9, 2022, Proceedings, Part I. Lecture Notes in Computer Science, vol. 13791, pp. 410–440. Springer (2022), https://doi.org/10.1007/978-3-031-22963-3_14

[125] Sönnerup, J., Hell, M., Sönnerup, M., Khattar, R.: Efficient Hardware Implementations of Grain-128AEAD. In: Hao, F., Ruj, S., Gupta, S.S. (eds.) Progress in Cryptology - INDOCRYPT 2019 - 20th International Conference on Cryptology in India, Hyderabad, India, December 15-18, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11898, pp. 495–513. Springer (2019), https://doi.org/10.1007/978-3-030-35423-7_25

[126] Speel, T.: Cryptanalysis of SPARKLE's ARX-box Alzette. Bachelor Thesis, Radboud University (2022)

[127] Sun, L., Wang, W., Wang, M.: Linear Cryptanalyses of Three AEADs with GIFT-128 as Underlying Primitives. IACR Trans. Symmetric Cryptol. 2021(2), 199–221 (2021), https://doi.org/10.46586/tosc.v2021.i2.199-221

[128] Sun, L., Wang, W., Wang, M.: Addendum to Linear Cryptanalyses of Three AEADs with GIFT-128 as Underlying Primitives. IACR Trans. Symmetric Cryptol. 2022(1), 212–219 (2022), https://doi.org/10.46586/tosc.v2022.i1.212-219

[129] Takemoto, S., Ikezaki, Y., Nozaki, Y., Yoshikawa, M.: Hardware Trojan for Lightweight Cryptoraphy Elephant. In: 10th IEEE Global Conference on Consumer Electronics, GCCE 2021, Kyoto, Japan, October 12-15, 2021. pp. 944–945. IEEE (2021), https://doi.org/10.1109/GCCE53005.2021.9622003

[130] Teng, W.L., Salam, M.I., Yau, W., Pieprzyk, J., Phan, R.C.: Cube Attacks on Round-Reduced TinyJAMBU. IACR Cryptol. ePrint Arch. p. 1164 (2021), https://eprint.iacr.org/2021/1164

[131] Todo, Y.: Structural evaluation by generalized integral property. In: Oswald, E., Fischlin, M. (eds.) Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9056, pp. 287–314. Springer (2015), https://doi.org/10.1007/978-3-662-46800-5_12

[132] Todo, Y., Isobe, T., Hao, Y., Meier, W.: Cube Attacks on Non-Blackbox Polynomials Based on Division Property. In: Katz, J., Shacham, H. (eds.) Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10403, pp. 250–279. Springer (2017), https://doi.org/10.1007/978-3-319-63697-9_9

[133] Todo, Y., Isobe, T., Meier, W., Aoki, K., Zhang, B.: Fast Correlation Attack Revisited - Cryptanalysis on Full Grain-128a, Grain-128, and Grain-v1. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10992, pp. 129–159. Springer (2018), https://doi.org/10.1007/978-3-319-96881-0_5

[134] Udvarhelyi, B., Bronchain, O., Standaert, F.: Security Analysis of Deterministic Re-keying with Masking and Shuffling: Application to ISAP. In: Bhasin, S., Santis, F.D. (eds.) Constructive Side-Channel Analysis and Secure Design - 12th International Workshop, COSADE 2021, Lugano, Switzerland, October 25-27, 2021, Proceedings. Lecture Notes in Computer Science, vol. 12910, pp. 168–183. Springer (2021), https://doi.org/10.1007/978-3-030-89915-8_8

[135] Vialar, L.: Fast Side-Channel Key-Recovery Attack against Elephant Dumbo. IACR Cryptol. ePrint Arch. p. 446 (2022), https://eprint.iacr.org/2022/446

[136] Wang, Q., Grassi, L., Rechberger, C.: Zero-Sum Partitions of PHOTON Permutations. IACR Cryptol. ePrint Arch. 2017, 1211 (2017), https://eprint.iacr.org/2017/1211

[137] Wang, Q., Grassi, L., Rechberger, C.: Zero-Sum Partitions of PHOTON Permutations. In: Smart, N.P. (ed.) Topics in Cryptology - CT-RSA 2018 - The Cryptographers' Track at the RSA Conference 2018, San Francisco, CA, USA, April 16-20, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10808, pp. 279–299. Springer (2018), https://doi.org/10.1007/978-3-319-76953-0_15

[138] Wang, Q., Hao, Y., Todo, Y., Li, C., Isobe, T., Meier, W.: Improved Division Property Based Cube Attacks Exploiting Algebraic Properties of Superpoly. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10991, pp. 275–305. Springer (2018), https://doi.org/10.1007/978-3-319-96884-1_10

[139] Wang, S., Hu, B., Guan, J., Zhang, K., Shi, T.: MILP-aided Method of Searching Division Property Using Three Subsets and Applications. In: Galbraith, S.D., Moriai, S. (eds.) Advances in Cryptology - ASIACRYPT 2019 - 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8-12, 2019, Proceedings, Part III. Lecture Notes in Computer Science, vol. 11923, pp. 398–427. Springer (2019), https://doi.org/10.1007/978-3-030-34618-8_14

[140] Wu, H., Huang, T.: CAESAR candidates AEGIS + Jambu. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[141] Wu, H., Huang, T.: TinyJAMBU: A Family of Lightweight Authenticated Encryption Algorithms (Version 2). Submission to the NIST Lightweight Cryptography project (2021)

[142] Xu, Z., Li, Y., Jiao, L., Wang, M., Meier, W.: Do NOT Misuse the Markov Cipher Assumption - Automatic Search for Differential and Impossible Differential Characteristics in ARX Ciphers. IACR Cryptol. ePrint Arch. p. 135 (2022), https://eprint.iacr.org/2022/135

[143] Yang, Y., Jang, K., Kim, H., Song, G., Seo, H.: Grover on SPARKLE. In: You, I., Youn, T. (eds.) Information Security Applications - 23rd International Conference, WISA 2022, Jeju Island, South Korea, August 24-26, 2022, Revised Selected Papers. Lecture Notes in Computer Science, vol. 13720, pp. 44–59. Springer (2022), https://doi.org/10.1007/978-3-031-25659-2_4

[144] Zhang, G., Liu, M.: A distinguisher on PRESENT-like permutations with application to SPONGENT. Sci. China Inf. Sci. 60(7), 72101 (2017), https://doi.org/10.1007/s11432-016-0165-6

[145] Zhang, X., Wang, T., Cao, P.: Side-Channel Evaluation on Protected Implementations of Several NIST LWC Finalists (2022), https://cryptography.gmu.edu/athena/LWC/Reports/SJTU/SJTU_Report_HW_4_candidates_RUB.pdf（2023-10-04 閲覧）

[146] Zhou, H., Li, Z., Dong, X., Jia, K., Meier, W.: Practical Key-Recovery Attacks On Round-Reduced Ketje Jr, Xoodoo-AE And Xoodyak. Comput. J. 63(8), 1231–1246 (2020), https://doi.org/10.1093/comjnl/bxz152

[147] Zhou, H., Zong, R., Dong, X., Jia, K., Meier, W.: Interpolation Attacks on Round-Reduced Elephant, Kravatte and Xoofff. Comput. J. 64(4), 628–638 (2021), https://doi.org/10.1093/comjnl/bxaa101

[148] Zong, R., Dong, X., Chen, H., Luo, Y., Wang, S., Li, Z.: Towards Key-recovery-attack Friendly Distinguishers: Application to GIFT-128. IACR Trans. Symmetric Cryptol. 2021(1), 156–184 (2021), https://doi.org/10.46586/tosc.v2021.i1.156-184

[149] 井上明子: 軽量暗号の安全性に関する調査及び評価（Elephant,ISAP,Romulus）（文書番号：CRYPTREC EX-3204-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3204-2022.pdf

[150] 岡部忠: 軽量ストリーム暗号のハードウェア実装 〜 FPGA を対象デバイスとした実装性能の比較 〜. In: 情報処理学会講演論文集. p. 2 (2022)

[151] 岩田哲: 軽量暗号の安全性に関する調査及び評価（Photon-Beetle,Sparkle,Tsudik's keymode）（文書番号：CRYPTREC EX-3201-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3201-2022.pdf

[152] 崎山一男: 軽量暗号の実装性能に関する調査及び評価（NIST 軽量暗号コンペティションファイナリスト）（文書番号：CRYPTREC EX-3205-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3205-2022.pdf

[153] 土生亮, 岩田哲: Elephant に対する鍵回復, 識別及び偽造攻撃. In: 暗号と情報セキュリティシンポジウム, SCIS2022, 1F2-5 (2022)

[154] 土生亮, 峯松一彦, 岩田哲: Romulus-N 及び Romulus-M に対する識別攻撃及び偽造攻撃. 信学技報 121(22, ISEC2021-6), 25–31 (2022)

[155] 藤堂洋介: 軽量暗号の安全性に関する調査及び評価（Ascon,Grain-128AEAD,TinyJambu）（文書番号：CRYPTREC EX-3203-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3203-2022.pdf

[156] 内藤祐介: 軽量暗号の安全性に関する調査及び評価（GIFT-COFB,Xoodyak）（文書番号：CRYPTREC EX-3202-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3202-2022.pdf

資料３－５

軽量暗号ガイドラインの更新

1. 背景

(1) 2020 年度第 2 回暗号技術検討会において、2016 年度版ガイドライン（文書番号：CRYPTREC GL-2003-2016）を 2023 年度中に更新することが承認された。

(2) 2021 年度の外部評価として、2016 年度版ガイドラインに掲載された軽量暗号の安全性評価に関する動向調査を実施した。

(3) 2021 年度第 2 回暗号技術評価委員会において、2023 年度版ガイドラインの作成に向けた更新方針が承認された。

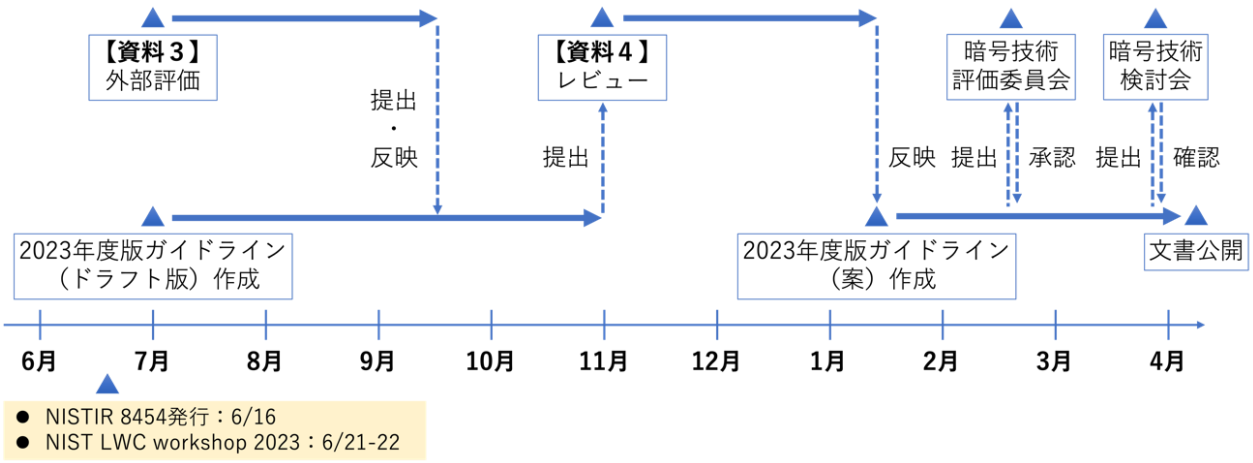
(4) ガイドラインの更新方針に基づき、2022 年度及び 2023 年度の外部評価として、軽量暗号の安全性と実装性能に関する調査及び評価、並びに軽量暗号の標準化動向に関する調査を実施した。

2. ガイドライン更新に向けた実施事項（概要報告）

(1) 2023 年度第 1 回暗号技術評価委員会において、ガイドラインの更新方針に基づき、事務局で 2023 年度版ガイドライン（ドラフト版）を作成するとともに、その掲載内容の適切性や情報の過不足などについて外部有識者にレビューいただき、2023 年度版ガイドライン（案）を作成することが承認された

(2) 2023 年度スケジュール

ガイドライン更新に向けた 2023 年度スケジュールは下の図のとおり。



(3) 2023 年度版ガイドライン（ドラフト版）の作成

ガイドラインの更新方針及び 2021 年度から 2023 年度にかけて実施した外部評価に基づき、事務局にて 2016 年度版ガイドラインの更新を行い、完成したものを 2023 年度版ガイドライン（ドラフト版）とする。

(4) 外部有識者によるレビュー

事務局が作成した 2023 年度版ガイドライン（ドラフト版）について、掲載内容の適切性や情報の過不足などを 2 名の外部有識者によりレビューいただくとともに、第 2 回暗号技術評価委員会にてレビュー結果を報告いただいた。

ア　本間 尚文 様（東北大学）

（ア）選出理由

2016 年度版ガイドライン作成時に、暗号技術調査ワーキンググループ（軽量暗号）の主査を担って頂き、ガイドラインの内容を熟知頂いている。また、軽量暗号に関わる幅広い知識をお持ちであるため。

（イ）依頼内容

CRYPTREC 事務局が作成した軽量暗号ガイドラインの更新案について、主に第 1 章から第 3 章（はじめに、軽量暗号とその活用法、軽量暗号の性能比較、の各章）に記載された更新内容の妥当性等を評価し、レビュー報告書を作成する。また、CRYPTREC 事務局が開催するレビュー結果に関する報告会にて、レビュー結果を報告する。

イ　峯松 一彦 様（日本電気株式会社）

（ア）選出理由

2016 年度版ガイドライン作成時に、主として認証暗号の分野をご担当頂いた方であり、共通鍵暗号に関わる幅広い知識をお持ちであるため。

（イ）依頼内容

CRYPTREC 事務局が作成した軽量暗号ガイドラインの更新案について、主に第 4 章及び新たに追加する付録(代表的な軽量暗号の章)に記載された更新内容の妥当性等を評価し、レビュー報告書を作成する。また、CRYPTREC 事務局が開催するレビュー結果に関する報告会にて、レビュー結果を報告する。

(5) 事務局による 2023 年度版ガイドライン（案）の作成

レビュー結果に基づき、事務局にて 2023 年度版ガイドライン（ドラフト版）の更新を行う。更新内容について外部有識者に了解頂いたものを最終的な 2023 年度版ガイドライン（案）とする。

3. 2023 年度版ガイドラインの概要

2023 年度第 2 回暗号技術評価委員会において、2023 年度ガイドライン（案）の承認をいただき、2023 年度ガイドラインとした。2023 年度版ガイドライン（資料３－５－別紙）の概要は以下のとおり。

(1) 目次

2023 年度版ガイドラインの目次は次の表のとおり。なお、表内において 2016 年度版ガイドラインとの差分箇所を赤字で示す。

章	章タイトル	概要
第 1 章	はじめに	導入、謝辞
第 2 章	軽量暗号とその活用法	
2.1	軽量暗号とは	定義、代表的な軽量暗号
2.2	軽量暗号の標準化動向	CAESAR コンペティション、NIST LWC、他標準化団体における ASCON の検討状況
2.3	軽量暗号はどこに使えるのか	家電、スマートテレビ、スマート農業、医療、自動車、等での活用例
2.4	どんな軽量暗号、パラメータを選べばいいか	一般的方針、鍵長・ブロック長の選択、利用シナリオ、等
2.5	軽量暗号活用例と効果	家電、スマートテレビ、スマート農業、医療、自動車、等での効果
第 3 章	軽量暗号の実装性能	
3.1	ブロック暗号の実装性能	12 種類の軽量ブロック暗号に対するハードウェア・ソフトウェア実装評価
3.2	認証暗号の実装性能	10 種類の軽量認証暗号に対するソフトウェア実装評価
3.3	ASCON の実装性能	ASCON のハードウェア・ソフトウェア実装評価と物理攻撃耐性評価
第 4 章	代表的な軽量暗号	
4.1	ブロック暗号	各技術分野の各方式に関する仕様等（設計者、発表年、仕様参照先、特徴、安全性解析状況、主な実装性能評価、標準化状況、等）の調査結果
4.2	ストリーム暗号	
4.3	ハッシュ関数	
4.4	メッセージ認証コード	
4.5	認証暗号	
付録 A	ASCON の物理攻撃耐性	
A.1	サイドチャネル攻撃対策	ASCON に対して有効なサイドチャネル攻撃対策として Threshold Implementation と Domain Oriented Masking の紹介
A.2	サイドチャネル解析・漏洩評価	ASCON に対して有効なサイドチャネル解析・漏洩評価として相関電力解析、故障利用攻撃、テンプレート攻撃、等の紹介
付録 B	CAESAR final portfolio: AEGIS, COLM	AEGIS、COLM に関する仕様等の調査結果
付録 C	NIST LWC ファイナリスト（ASCON を除く）	ASCON を除く NIST LWC ファイナリスト 9 方式に関する仕様等の調査結果

(2) 主な更新内容

2023 年度版ガイドラインの主な更新内容は次の表のとおり。

章		章タイトル	内容
第 2 章		軽量暗号とその活用法	
	2.1	軽量暗号とは	本ガイドラインで紹介する代表的な軽量暗号技術（表 2.1）を**改定**
	2.2	軽量暗号の標準化動向	2022 年度・2023 年度外部評価（菅野様）に基づく**追加** ● CAESAR コンペティション ● NIST LWC ● 他標準化団体における ASCON の検討状況
第 3 章		軽量暗号の実装性能	
	3.3	ASCON の実装性能	2022 年度外部評価（崎山様）に基づく**追加** ● ASCON のハードウェア実装評価 ● ASCON のソフトウェア実装評価 2023 年度外部評価（崎山様）に基づく**追加** ● ASCON の物理攻撃耐性評価
第 4 章		代表的な軽量暗号	
	共通		2021 年度外部評価（事務局）に基づく**改定** ● 各方式における安全性解析状況 2022 年度外部評価（菅野様）に基づく**改定** ● 各方式における標準化動向
	4.1	ブロック暗号	2021 年度外部評価（事務局）に基づく**追加** ● LEA に関する仕様等
	4.3	ハッシュ関数	2021 年度外部評価（事務局）に基づく**追加** ● Lesamnta-LW に関する仕様等
	4.4	メッセージ認証コード	2021 年度外部評価（事務局）に基づく**追加** ● Chaskey、LightMAC に関する仕様等 2022 年度外部評価（岩田様）に基づく**追加** ● Tsudik's keymode に関する仕様等
	4.5	認証暗号	2021 年度外部評価（事務局）に基づく**追加** ● Grain-128A に関する仕様等 2022 年度外部評価（藤堂様）に基づく**改定** ● ASCON の安全性解析状況
付録 A		ASCON の物理攻撃耐性	2023 年度外部評価（崎山様）に基づく**追加** ● ASCON に対するサイドチャネル攻撃対策 ● ASCON に対する物理攻撃手法
付録 B		CAESAR final portfolio: AEGIS, COLM	2021 年度外部評価（事務局）に基づく**追加** ● AEGIS、COLM に関する仕様等
付録 C		NIST LWC ファイナリスト（ASCON を除く）	2022 年度外部評価（岩田様、内藤様、藤堂様、井上様、崎山様）に基づく**追加** ● Elephant、GIFT-COFB、Grain-128AEAS、ISAP、PHOTON-Beetle、Romulus、Sparkle、TinyJambu、Xoodyak に関する仕様等

4. 2023 年度版ガイドライン（ドラフト）版に対するレビュー結果報告

(1) 第 1 章から第 3 章の更新内容に関するレビュー（報告者：本間様）

1 箇所の構成変更といくつかの確認が必要と思われる箇所を除き、第 1 章から第 3 章における改定内容・構成が妥当であると報告する。主なレビュー内容は次のとおり。詳細は資料３－５別紙２のとおり。

節	ページ	レビュー内容
2.1	5	CAESAR コンペティションにおける軽量暗号アプリケーション以外のユースケース候補の取扱いについて確認・整理するよう提案（表 2.1）
2.2	8-9	NIST LWC プロジェクトにおける評価基準と選定プロセスの対応関係について整理するよう提案
2.2	9-10	ASCON に関する評価について個別に整理するよう提案
3.3	73	スループット性能の正しい意図が読者に伝わるよう「ここではレイテンシの逆数としての評価指標と位置づけ、パイプライン化による向上は想定しない」などの注釈をつけるよう提案
3.3	84-85	本ガイドラインの想定読者に合わせた改定が望まれるため、サイドチャネル攻撃対策手法とサイドチャネル解析・漏えい評価手法の詳細については付録とし、各手法の説明を各 1～2 行程度で本文に記載するよう提案
共通		軽微な修正を提案

(2) 第 4 章の更新内容に関するレビュー（報告者：峯松様）

全般的に記載内容に関して大きな疑義を呈する箇所はなく、改定内容が妥当であると報告する。主なレビュー内容は次のとおり。詳細は資料３－５別紙３のとおり。

節	ページ	レビュー内容
4.3	128	Keccak を除くスポンジベースハッシュの暗号学的置換への識別攻撃に関する言及はあるが、Keccak への識別攻撃に関する言及が無かったため、記載することが望ましいと提案
4.4	139	Chaskey に関し、仕様段数が 8 段と 12 段の 2 つのバリアントがあるが、ISO/IEC 標準は 12 段の Chaskey であることを記載することが望ましいと提案
4.4	140	LightMAC には証明可能安全性が存在し、Simeck32/64 への有意な識別攻撃が存在しない限り LightMAC を解読することが不可能であるため、LightMAC で Simeck32/64 を利用した場合に解読可能というような記述は適切ではないことを提言
4.5	157	2022 年 9 月以降、OCB3 のナンス長が 6 ビット未満の場合におけるシンプルな攻撃が提案されており、仕様に最低ナンス長が記載されていない以上、一定のリスクがあることから、本ガイドラインに反映させた方が良いことを提案
付録 B	187	2022 年 9 月以降、Xoodyak の仕様で記載されている安全性主張の誤りが指摘されており、新しい安全性主張に関して本ガイドラインに反映させた方が良いことを提案
共通		軽微な修正、引用先文献の追加を提案

5. 今後の予定

作成した2023年度ガイドラインは軽量暗号に関する最新動向を踏まえて2016年度版ガイドラインを更新したものであり、暗号技術ガイドラインとして十分な内容を含んでいると考えられる。また、外部有識者によるレビュー結果で更新内容の妥当性が評価されている。

この後、2024年3月31日までにガイドラインを完成させ、4月上旬にCRYPTRECホームページで公開しますので、ご承認をお願いします。

以上

資料４－１

2023 年度　暗号技術活用委員会活動報告

1.　2023 年度の活動概要

1.1　活動目的

活用委員会では、情報システム全般のセキュリティ確保に寄与することを目的として、暗号の取り扱いに関する観点から、運用ガイドライン／ガイダンスの作成を行っている。

2023 年度は、TLS 暗号設定ガイドラインの改訂を行う。また、2022 年度と同様に暗号鍵管理ガイダンス WG を設置して、2022 年度の成果として発行した暗号鍵管理ガイダンスの拡充を行う。暗号鍵管理ガイダンスの拡充は 2024 年度の完成を目標とする。

1.2　活動概要

今年度の活動概要は以下の通りである。

(1) TLS 暗号設定ガイドラインの改訂

現在の「TLS 暗号設定ガイドライン（Ver3.0.1)」の公開（2020 年 7 月）以降、CRYPTREC では CRYPTREC 暗号リストの改定、「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準（以降「強度要件設定基準」と表記)」の策定を行っている。このため、これら CRYPTREC 成果の取り込み及び 3 年間の TLS に関する RFC 規格化や技術環境の変化なども踏まえ、本ガイドラインを改訂する。

(2) 暗号鍵管理ガイダンスの拡充

暗号鍵管理ガイドラインの拡充を目的として進めていた暗号鍵管理ガイダンスについて、2021 年度・2022 年度に引き続いて暗号鍵管理ガイダンス WG を設置し、2022 年度発行版では記載を見送った部分の拡充を行う。2022 年度版の内容見直しも含め、2024 年度完成を目標とする。

1.3　暗号技術活用委員会の委員構成及び開催状況

暗号技術活用委員会の委員構成は表 1-1 のとおりである。また、2023 年度に開催された暗号技術活用委員会での議案は表 1-2 のとおりである。

表 1-1 暗号技術活用委員会　委員構成

委員長	松本　勉	横浜国立大学　教授
委員	上原　哲太郎	立命館大学　教授
委員	垣内　由梨香	Microsoft Corporation　セキュリティプログラムマネージャー
委員	菊池　浩明	明治大学　教授

委員	佐藤　直之	SCSK 株式会社　シニアプロフェッショナルコンサルタント
委員	佐藤　雅史	セコム株式会社　主幹研究員
委員	須賀　祐治	株式会社インターネットイニシアティブ　シニアエンジニア
委員	田村　裕子	日本銀行金融研究所　企画役
委員	手塚　悟	慶應義塾大学　教授
委員	寺村　亮一	GMOサイバーセキュリティbyイエラエ株式会社　執行役員
委員	三澤　学	三菱電機株式会社　グループマネージャ
委員	満塩　尚史	デジタル庁　セキュリティアーキテクト
委員	山口　利恵	東京大学　准教授
委員	渡邊　創	国立研究開発法人産業技術総合研究所　副研究センター長

（2024 年 3 月 5 日現在）

表 1-2　暗号技術活用委員会　開催状況

回	開催日	議案
第一回	2023 年 7 月 11 日	● 2023 年度暗号技術活用委員会活動計画について ● 2023 年度暗号鍵管理ガイダンス WG 活動計画について ● TLS 暗号設定ガイドライン改訂について
メール	2023 年 1 月 12 日 ～2 月 15 日	● TLS 暗号設定ガイドライン改訂案 v3.1 のメール審議
第二回	2024 年 3 月 5 日	● TLS 暗号設定ガイドライン改訂内容について ● 2023 年度暗号鍵管理ガイダンス WG 活動報告 ● Triple DES に関する扱いについて ● 2023 年度暗号技術活用委員会活動報告案について

2.　成果概要

以下に成果概要の要約を記載する。詳細については、CRYPTREC Report 2023 暗号技術活用委員会報告[1]（以下「活用委員会報告書」）を参照されたい。

2.1　TLS 暗号設定ガイドラインの改訂

現行の TLS 暗号設定ガイドライン（v3.0.1）からの一番大きな変更点は、強度要件設定基準の策定に伴い、安全性の基準として「鍵長」で表現されていた部分を「ビットセキュリティ」で表現するようにしたところである。これにより、「鍵長 256 ビットの楕円曲線」との要件に「X25519 の楕円曲線」が許容されるか否かについて、明確に許容されることとなった。

[1] CRYPTREC Report 2023 暗号技術活用委員会報告, https://www.cryptrec.go.jp/promo_cmte.html

また、3 年間の TLS 規格化や技術環境の変化なども踏まえ、主に以下の観点での議論を行い、必要な改訂を行うこととした。

① CRYPTREC 暗号リスト改定等を踏まえた TLS での利用を推奨／禁止する暗号アルゴリズムの改訂
② 「セキュリティ例外型」の取り扱い
③ DHE の強度設定について推奨要件の改訂要否
④ その他、改訂することが望ましい項目

作成したガイドライン v3.1 ドラフト案に対する主な改訂内容を表 1-3 にまとめる。なお、今回の改訂では、推奨の設定内容に大きな影響を与える項目がないことから、バージョン名は v3.1 とすることとした。

議論の詳細については活用委員会報告書を参照されたい。

表 1-3　TLS 暗号設定ガイドラインの主な改訂内容

項目	改訂内容概要
「鍵長」基準から「ビットセキュリティ」基準への変更	強度要件設定基準に従い、現行版（v3.0.1）の鍵長をそのままビットセキュリティ基準に置き換えた。なお、利用する楕円曲線は「強度要件設定基準」に記載のものから選択することを明記した。 また、セキュリティ例外型の DH／DHE の 1024 ビット鍵長は、対応するビットセキュリティ基準が存在しないため、鍵長表現のままとした。
CRYPTREC 暗号リスト改定等を踏まえた TLS での利用を推奨／禁止する暗号アルゴリズムの更新	改定された CRYPTREC 暗号リストによりリストの位置づけが変更されたアルゴリズム、及び 3 年間の TLS に関する RFC 規格化や技術環境の変化などにより変更が必要と考えるアルゴリズムについて、以下のように改訂する。 ● サーバ証明書における DSA の利用推奨を削除する ● サーバ証明書における RSA-PSS の利用推奨を追加する ● EdDSA はサーバ証明書、暗号スイートとも利用推奨をしない ● 暗号スイートでの利用禁止暗号アルゴリズムに SM2（署名）、SM3、SM4 を追加する
「セキュリティ例外型」の取り扱い	移行を明確に促す観点から移行期限を明記した以下の表現に強化する。 「本ガイドラインで記載されているセキュリティ例外型の設定内容は、2029 年度を目途とした改訂時に終了させる予定である。速やかに推奨セキュリティ型への移行を完了させるべきである。」
DHE の強度設定について推奨要件の改訂要否	以下のように改訂する。 ● 高セキュリティ型は 112 ビットセキュリティから 128 ビットセキュリティ以上に変更する。推奨セキュリティ型とセキュリティ例外型は変更しない。
その他	その他の主な改訂内容として以下のものがある。 ● 「Certificate Transparency」に関する節の追加 ● 「ブラウザを利用する際に注意すべきポイント」について、Microsoft、Google、Mozilla、Apple の各ブラウザの最新情報を反映

	なお、IoT の普及という観点から組み込み系に向けた補足ドキュメントを検討してはどうかとの意見があったが、本ガイドラインの主たる読者層とは対象が異なると想定されることから、今後の新規ガイドラインの作成や拡充の候補として検討することになった。

2.2 暗号鍵管理ガイダンスの拡充

2022 年度に発行した「暗号鍵管理ガイダンス Ver.1.0」と今回作成中の「暗号鍵管理ガイダンス拡充分」は、「暗号鍵管理システム設計指針（基本編）」の章構成に対応して表 1-4 のとおりである。なお、下表はガイダンス拡充分を別冊とした場合の章構成であり、暗号鍵管理ガイダンス Ver.1.0 にマージするか別冊とするかは 2024 年度の執筆状況を踏まえて決定する。

表 1-4　暗号鍵管理ガイダンスの章構成

暗号鍵管理システム設計指針（基本編）	暗号鍵管理ガイダンス Ver.1.0（2022 年度発行）	暗号鍵管理ガイダンス拡充分（別冊時）
1. はじめに	1. はじめに	1. はじめに
2. 暗号鍵管理の在り方	（1 章に集約）	（1 章に集約）
3. 本設計指針の活用方法	（1 章に集約）	（1 章に集約）
4. 暗号鍵管理システム（CKMS）の設計原理と運用ポリシー		2. 暗号鍵管理システム（CKMS）の設計原理と運用ポリシー
5. 暗号アルゴリズム運用のための暗号鍵管理オペレーション対策	2. 暗号アルゴリズム運用のための暗号鍵管理オペレーション対策	
6. 暗号アルゴリズムの選択	3. 暗号アルゴリズムの選択	
7. 暗号アルゴリズム運用に必要な鍵情報の管理	4. 暗号アルゴリズム運用に必要な鍵情報の管理	
8. 暗号鍵管理デバイスへのセキュリティ対策		3. 暗号鍵管理デバイスへのセキュリティ対策
9. 暗号鍵管理システム（CKMS）のオペレーション対策		4. 暗号鍵管理システム（CKMS）のオペレーション対策

2023 年度は、「暗号鍵管理システムの設計原理と運用ポリシー」及び「暗号鍵管理デバイスへのセキュリティ対策」について、記載すべき内容をダイジェスト形式で整理した。整理した主な概要は以下のとおりである。

議論の詳細については、活用委員会報告書中の暗号鍵管理ガイダンス WG 活動報告を参照されたい。

① トイモデル

暗号鍵管理システムのシンプルなモデル（トイモデル）を例示し、それに対する各検討項目への対応例を説明するためのモデルとして、「暗号鍵管理システムの設計原理と運用ポリシー」及び「暗号鍵管理デバイスへのセキュリティ対策」の節に記載するトイモデルを、IoT 製品向けのプライベート CA システム（図 1）とすることに決定した。

トイモデルで扱う「プライベート CA システム」の構成

- CKMS の範囲を CA サーバと HSM までとする
- プライベート CA システムは IoT 製品（家電想定）向けに公開鍵証明書の発行及び証明書の失効管理に使われる CRL の発行を行う
- 証明書のトラストアンカーはプライベート CA である
- IoT 製品向けの ID 管理、プライベート鍵生成、発行された証明書の機器埋め込みは工場内で行う
- IoT 製品は運用時にネット接続され、スマホ内専用アプリから IoT 製品ハブ経由でセンシングや制御が行われる。証明書は専用アプリと IoT 機器の接続（TLS での認証と秘匿通信確立）に利用される

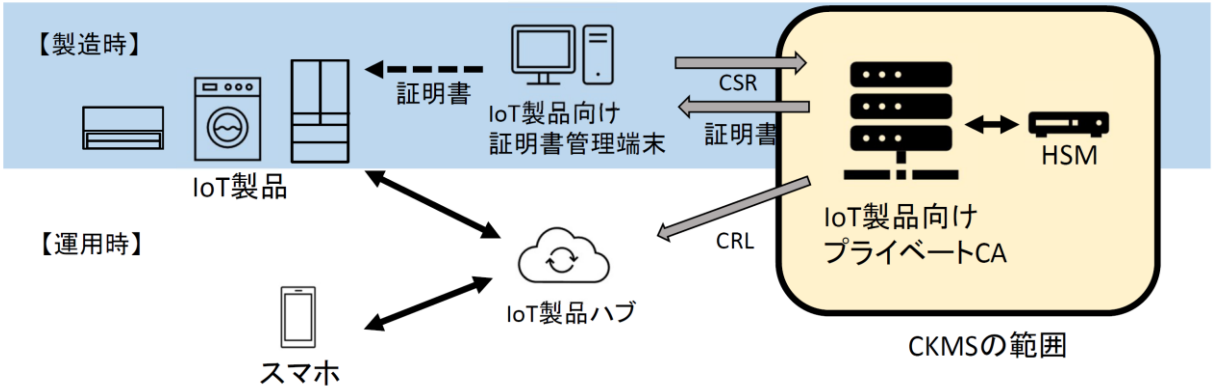


図 1　トイモデルで扱う「プライベート CA システム」の構成図

② 「暗号鍵管理システムの設計原理と運用ポリシー」に記載する「解説・考慮点」の主な概要（表 1-5）

表 1-5　「暗号鍵管理システムの設計原理と運用ポリシー」での「解説・考慮点」の主な概要

節番号	FR 番号	「解説・考慮点」の説明概要
4.1 節 CKMS セキュリティポリシー	A.01-A.05	セキュリティポリシーとは CKMS が実現するセキュリティ機能や運用方針の概要を定めたもの。CKMS を利用するシステムや CKMS が構築される IT 環境のポリシーなどと矛盾がないことが前提

4.2 節 情報管理ポリシー等からの要求事項	A.06	個人の説明責任が求められるケース（監査、リスクマネジメントの観点）を想定して CKMS でのサポートメカニズムを記載
	A.07-A.13	匿名性、連結不可能性、観測不可能性のサポート有無とサポートする場合のメカニズムを記載。一般に、匿名性、連結不可能性、観測不可能性を要求するのは特殊なケース
4.3 節 ドメインのセキュリティポリシー	A.14-A.19	異なるセキュリティドメイン間での鍵情報の交換がなければ対象外。GPKI は異なるセキュリティドメイン間での鍵交換の事例
	A.22-A.26	マルチレベルのセキュリティドメインでの鍵情報の交換がなければ対象外。一般に、マルチレベルのセキュリティドメインでの鍵情報の交換は特殊なケース
4.4 節 CKMS における役割と責任	A.27-A.28	CKMS の運用に関わるエンティティを定め、エンティティに割り当てる役割と実行できる鍵情報の管理機能へのアクセス権（権限）を定義する。
	A.29-A.31	不必要な権限の割り当てや権限の分離が不十分な場合、内部犯行を誘発するリスクがある
4.5 節 CKMS の構築環境及び実現目標	A.32	CKMS を構成する主要なデバイスおよびコンポーネントの一式を定める
	A.33-A.36	CKMS が要求する時刻の精度や利用する権威時刻ソース、第三者タイムスタンプの要求有無を定める
	A.39-A.42	初期及び将来を想定してユーザ数や CKMS 性能面の目標、負荷増大時の対応策を定める
	A.43-A.46	CKMS 内デバイスや CKMS 間の相互運用を可能とするため、インタフェース、プロトコル、コマンド仕様を定める
	A.47-A.50	使いやすいユーザインタフェースを検討し、ヒューマンエラーを防止する
	A.51-A.53	どのような商用既製品を利用してどのようなセキュリティ機能を実行するかを定める
4.6 節 標準／規制に対する適合性	A.54-A.55	暗号アルゴリズム、暗号モジュール、セキュリティ認証などの標準への準拠性を明確にする
	A.57	CKMS が使用される国家・地域の法的規制を明確にする。欧州のサイバーセキュリティ法、中国のデータセキュリティ法、各国のデータ規制など
4.7 節 将来的な移行対策の必要性	A.58-A.61	CKMS は暗号アルゴリズムのセキュリティライフタイムを超えたサービス提供や、危殆化により、暗号アルゴリズムの置き換えが必要になる。そのため、複数の暗号アルゴリズムや異なる鍵長をサポートするケースも多い
	A.62-A.69	技術の進歩をウォッチすると共に、予め潜在的な脅威に対する影響評価の実施を推奨する

③ 「暗号鍵管理デバイスへのセキュリティ対策」に記載する「解説・考慮点」の主な概要（表 1-6）

表 1-6　「暗号鍵管理デバイスへのセキュリティ対策」での「解説・考慮点」の主な概要

節番号	FR 番号	「解説・考慮点」の説明概要
8.1 節 鍵情報へのアクセスコントロール	E.01-E.04	暗号モジュールの各機能の実行を認可されたエンティティに限定する。実行権を管理するアクセスコントロールシステム(ACS)は暗号モジュールと連動して動作する
	E.05	ACS によるエンティティ識別、認証、認可の粒度や機能を明確にする
	E.07-E.20	暗号モジュールとは、暗号境界内で暗号処理を実行するハードウェアもしくはソフトウェアの集合。暗号境界内で利用される暗号鍵の保護機能を有する
	E.08-E.14	暗号モジュールへの鍵情報の入出力を平文形式で行うことは望ましくない。出力は暗号化して行うことが望ましく、主に外部での保管（バックアップなど）目的である
	E.21	鍵情報の入力を人間が行う場合、その正確さとセキュリティ面の問題がある。こうした入力がない場合は対象外
	E.22-E.25	マルチパーティコントロールを利用する機能を明確にする。暗号鍵分割(K out of N 秘密分散)やマルチパーティ機能をベンダに確認する
8.2 節 セキュリティ評価・試験	E.26-E.34	いずれもシステムレベルの試験項目であるが、特に暗号モジュール(HSM)にも関連するものはベンダテスト、機能テスト、セキュリティテスト、環境テスト、セルフチェックテスト、第三者テストである
	E.26-E.34	FIPS140 などの認証試験で上記テストをカバーするものが多い
8.3 節 暗号モジュールの障害時の BCP 対策	E.35	暗号モジュールはセルフテスト機能を備えることが望ましい。FIPS140-2/-3 の要件に動作前や条件付きのセルフテスト機能がある
	E.37	回復可能なエラー発生時のセルフテストを含む回復の手順、回復困難なエラー発生時の暗号モジュールの交換手順(鍵情報のバックアップや破壊を含む)を明確にする

2.3　Triple DES 等の取り扱いについて

NIST が Triple DES を規定していた SP 800-67 Revision 2 を 2023 年 12 月 31 日に（予定通り）廃止したことに伴い、暗号技術検討会事務局からの Triple DES の取り扱いについての意見聴取の依頼に対し、暗号技術活用委員会としては検討の結果、以下のように回答した。

【Triple DES の扱いに対する意見】

- 現時点では、「運用監視暗号リスト」からの削除を検討する必要性はない
- 現時点では、「運用監視暗号リスト」の条件である「互換性維持以外の目的での利用は推奨しない。」が実質的かつ十分な制約になっており、特段の利用制限を付加する必要性もない
- 「SP 800-67 Revision 2 が 2023 年 12 月に廃止されたが、それ以外は、運用監視暗号リストに移行した時点での状況とほとんど変わっていないため、Triple DES の位置づけに変更はない。」

との注釈を付記する

【上記意見に至った理由】

① 廃止理由が、安全性が著しく低下したわけではなく、NIST のスケジュールに基づく動きであること
② 利用実績調査結果からは依然として極めて高い実装率であること
③ すでに運用監視暗号リストに掲載されており、互換性維持以外の目的での利用が推奨されていないこと
④ NIST も、Triple DES ですでに暗号化されたデータに対する処理は引き続き許容していること
⑤ 「電子政府推奨暗号リスト」に掲載されている DSA は、現在の FIPS PUB 186-5 では廃止されているが、FIPS PUB 186-5 になるときに削除すべきとの議論はなかったこと

【DSA の扱いに対する意見】

- 今回、Triple DES の取り扱いについて検討することになった理由が「SP 800-67 Revision 2 が廃止された」ことが契機になっていると承知している。その場合、上記⑤に記載の通り、DSA も「現在の FIPS PUB 186-5 では廃止されている」ことから、Triple DES との注釈と同様の注釈を追記すべきではないか。

3. 今後に向けて

2024 年度は、暗号鍵管理ガイダンス WG にて検討中の暗号鍵管理ガイダンスを完成させる予定である。また、暗号利活用のための新たなガイダンスの作成について検討する予定である。

資料４－２

TLS暗号設定ガイドラインの改訂内容について

改訂の背景

「TLS暗号設定ガイドライン（Ver3.0.1）」の作成時点（2020年7月）以降の動向を踏まえ、以下の対応をするための改訂を行う

- 「CRYPTREC暗号リスト（電子政府推奨暗号リスト）」の改定
  - 「電子政府推奨暗号」への昇格：
    EdDSA, SHA-512/256, SHA3-256, SHA3-384, SHA3-512, SHAKE128, SHAKE256, XTS, ChaCha20-Poly1305, ISO/IEC 9798-4
  - 「運用監視暗号」への降格：3key-Triple DES
  - 「運用監視暗号」からリスト外への降格：RC4, SC2000

  ※下線のアルゴリズムはIANA TLS registryに登録されているもの
- 「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」の策定
  - 「ビットセキュリティ基準」を導入し、これに基づいてアルゴリズムごとに強度要件を規定
- TLSに関連するIETFでの動向や利用（サポート）状況の変化

主な改訂の内容

主な改訂内容は以下のとおり。

① 「鍵長」基準から「ビットセキュリティ」基準への変更

② CRYPTREC暗号リスト改定等を踏まえたTLSでの利用を推奨／禁止する暗号アルゴリズムの更新

③ 「セキュリティ例外型」の取り扱い

④ DHEの強度設定について推奨要件の改訂要否

⑤ その他、改訂することが望ましい項目

①「鍵長」基準から「ビットセキュリティ」基準への変更

- 暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準に従い、現行版（v3.0.1）の「鍵長」をそのまま「ビットセキュリティ」に置き換え
- 利用する楕円曲線は「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」に記載のものから選択することを明記
- セキュリティ例外型のDH／DHEの1024ビット鍵長は、対応するビットセキュリティ基準が存在しないため、鍵長表現のまま

表 11　表 6、表 7、表 8 に記載の暗号アルゴリズム・パラメータに対する「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」での推定セキュリティ強度

セキュリティ強度（ビットセキュリティ）		112	128	192	256
公開鍵暗号（署名・守秘・鍵共有）	素因数分解型（RSA 暗号・RSA 署名）	鍵長 2048 ビット	鍵長 3072 ビット	鍵長 7680 ビット	鍵長 15360 ビット
	離散対数型（DH(E)、DSA）	鍵長 2048 ビット (L, N) = (2048, 224)	鍵長 3072 ビット (L, N) = (3072, 256)	鍵長 7680 ビット (L, N) = (7680, 384)	鍵長 15360 ビット (L, N) = (15360, 512)
	楕円曲線暗号（ECDH(E)、ECDSA、EdDSA）	P-224 B-233 K-233	P-256 B-283 K-283 W-25519 Curve25519 Edwards25519	P-384 B-409 K-409 W-448 Curve448 Edwards448	P-521 B-571 K-571
共通鍵暗号	ブロック暗号	なし	鍵長 128 ビットの AES、Camellia	鍵長 192 ビットの AES、Camellia	鍵長 256 ビットの AES、Camellia
	認証暗号	なし	なし	なし	ChaCha20-Poly1305
ハッシュ関数	HMAC で使う場合	なし	SHA-1	なし	SHA-256 SHA-384 SHA-512

② CRYPTREC暗号リスト改定等を踏まえたTLSでの利用を推奨／禁止する暗号アルゴリズムの更新

暗号アルゴリズム	改訂内容案
DSA	FIPS186-5から削除されたので一段階下げる表現にする →「本ガイドラインでは積極的には利用を勧めない」から「今後、新規・更新時にDSAを利用すべきではない」に修正
RSA-PSS	「サーバ証明書で利用可能な署名アルゴリズム」として、「推奨セキュリティ型」および「高セキュリティ型」に追加する ● 現行版はサーバ証明書でのRSA-PSSの記載が漏れていたため
EdDSA	「サーバ証明書で利用可能な暗号」及び「暗号スイートでの利用推奨暗号アルゴリズム」への追加は行わない ● サーバ証明書ではCA/ブラウザフォーラムの規定によりEdDSAはCA署名アルゴリズムとしても、subject公開鍵としても設定できない状況にある。このため、TLSでの鍵交換時に付与する署名アルゴリズムとしてもEdDSAを利用できないため。2.5.2節にこの旨を記載
ChaCha20-Poly1305	現行版ですでに記載済み → 対応不要（「リストの種類」の修正及び「ストリーム暗号」から「認証暗号」に変更のみ）
3-key Triple DES	現行版ですでに削除済み → 対応不要
RC4	現行版ですでに削除済み → 対応不要（「リストの種類」の修正のみ）
SM2（署名）、SM3、SM4	CRYPTREC暗号リストに記載されていない → 利用禁止リストに追記

③「セキュリティ例外型」の取り扱い

- 移行を明確に促す観点から移行期限を明記した以下の表現に強化

【原文】
※推奨セキュリティ型への移行完了までの暫定運用を想定している。

【改訂文】
※本ガイドラインで記載されているセキュリティ例外型の設定内容は、2029年度を目途とした改訂時に終了させる予定である。速やかに推奨セキュリティ型への移行を完了させるべきである。

④ DHEの強度設定について推奨要件の改訂要否

設定基準	改訂内容案
高セキュリティ型	「128ビットセキュリティ以上の鍵長（3072ビット以上）」に変更する ● DHEでは依然として2048ビット鍵が主流だが、高セキュリティ型では、先行して「128ビットセキュリティ以上の鍵長（3072ビット以上）」にすべき
推奨セキュリティ型	変更しない ● DHEでは依然として2048ビット鍵が主流であるため
セキュリティ例外型	変更しない **※ 1024ビット鍵長利用（112ビットセキュリティよりはるかに弱い設定）に関して、以下の理由により変更しない** ➢ 設定内容を変更させるなら、セキュリティ例外型で設定内容を変更させるよりも、推奨セキュリティ型への変更に誘導するほうがよい ➢ このタイミングで設定内容を見直すと、設定内容を変更しさえすれば「セキュリティ例外型」の継続利用をむしろ容認したかのように誤解される恐れがある

⑤ その他、改訂することが望ましい項目

- 「Certificate Transparency（全世界のデジタル証明書が確認できる仕組み）」に関する節の追加
  - デジタル証明書の誤発行や不正発行のインシデントが相次いで発生した。Certificate Transparency（CT）は、それらのインシデントを受けて実施された様々な取り組みの1つ
- 「ブラウザを利用する際に注意すべきポイント」について、Microsoft、Google、Mozilla、Appleの各ブラウザの最新情報を反映
  - サポート中のバージョン情報
- IoT の普及という観点から組み込み系に向けた補足ドキュメントを検討してはどうかとの意見があったが、本ガイドラインの主たる読者層とは対象が異なると想定されることから、今後の新規ガイドラインの作成や拡充の候補として検討することになった

資料４－３
CRYPTREC GL-3001-3.1.0

TLS 暗号設定ガイドライン

2024 年 4 月
（2024 年 3 月 21 日版－委員最終確認版）

独立行政法人 情報処理推進機構
国立研究開発法人 情報通信研究機構

目次

【修正履歴】

修正日	修正内容
2024.4.xx（Ver.3.1）	● CRYPTREC 暗号リストを 2013 年発行版から 2023 年発行版に変更 ● 安全性評価尺度の基準を「鍵長」から「ビットセキュリティ」に変更 ● 「セキュリティ例外型」の利用停止（「推奨セキュリティ型への移行完了」）を勧告 ● 2023 年 11 月時点での情報にアップデート
2020.7.8（Ver.3.0.1）	● Appendix A の URL 誤植を修正
2020.7.7（Ver.3.0）	● ガイドラインの名称を「SSL/TLS 暗号設定ガイドライン」から「TLS 暗号設定ガイドライン」に変更 ● 最新動向を踏まえ、内容を全面改訂（大きな差分の説明は Appendix E 参照）
2018.5.8（Ver.2.0）	● 最新動向を踏まえ、「セキュリティ例外型」を中心とした設定基準の見直しを実施 ● 最新データへの更新を実施
2015.8.3（Ver.1.1）	Appendix B.6 での誤記を修正
2015.5.8（Ver.1.0）	初版発行

1. はじめに

1.1 本書の内容及び位置付け

本ガイドラインは、2024 年 3 月時点における TLS 通信での安全性と相互接続性のバランスを踏まえた TLS サーバの設定方法を示すものである。

特に、SSL/TLS 暗号設定ガイドライン（version 1.x/2.x）発行以降、TLS1.3 発行[RFC8446]や SSL3.0 禁止[RFC7525]、ChaCha20-Poly1305 追加[RFC7905]、RC4 禁止[RFC7465]など、同ガイドラインに記載されている内容に大きく影響する規格化が相次いで行われており、それに伴い SSL/TLS の利用環境も大きく変化した（2.7 節参照）。

今回の改訂にあたっては、このような規格化状況及びサポート状況等の各種動向を踏まえ、プロトコルバージョンの要求設定において TLS1.3 の採用及び SSL3.0 の禁止を行った。これに伴い、各設定基準における要求設定についても大幅な変更が行われており、SSL/TLS 暗号設定ガイドライン（version 1.x/2.x）における設定基準から一段階高い安全性を求めるようになった項目も多い。例えば、推奨セキュリティ型で利用が認められていた TLS1.0 や TLS1.1 は、本ガイドラインではセキュリティ例外型のみで利用可能となった。また、鍵交換では Perfect Forward Secrecy の特性をもつ ECDHE や DHE をさらに強く推奨するようにした。

このため、SSL/TLS 暗号設定ガイドライン（version 1.x/2.x）を利用している場合には、本ガイドラインでの要求設定に基づいた見直しを速やかに行い、必要に応じて設定変更を実施することを強く推奨する。

本ガイドラインは 9 章で構成されており、章立ては以下のとおりである。

2 章では、本ガイドラインを理解するうえで助けとなる技術的な基礎知識をまとめている。特に高度な内容は含んでおらず、TLS 及び暗号についての技術的な基礎知識を有している読者は本章を飛ばしてもらって構わない。

3 章では、TLS サーバに要求される設定基準の概要について説明しており、4 章から 6 章で実現すべき要求設定の考え方を示す。

4 章から 6 章では、3 章で定めた設定基準に基づき、具体的な TLS サーバの要求設定について示す。安全性と相互接続性を踏まえたうえで、選択した設定基準としての最低限の安全性を確保するために必ず満たさなければならない項目である「遵守項目」と、当該設定基準としてよりよい安全性を実現するために満たすことが望ましい項目である「推奨項目」を決めている。

7 章では、チェックリストの対象には含めていないが、TLS を安全に使うために考慮すべきことをまとめている。本章の内容は、「情報提供」の位置づけとして記載している。

8 章は、クライアントの一つであるブラウザの設定に関する事項を説明しており、ブラウザの利用者に対して啓発するべき事項を取り上げている。本章の内容は、7 章と同様、「情報提供」の位置づけのものである。

9 章は、そのほかのトピックとして、TLS を用いたリモートアクセス技術（“SSL-VPN”とも言われる）について記載している。本章の内容も「情報提供」の位置づけのものである。

3 章から 6 章が本ガイドラインの最大の特長ともいえ、「暗号技術以外の様々な利用上の判断材料も加味した合理的な根拠」を重視して現実的な利用方法を目指している。具体的には、実現すべき安全性と必要となる相互接続性とのトレードオフを考慮する観点から、安全性と相互接続性を踏まえたうえで設定すべき要求設定として 3 つの設定基準（「高セキュリティ型」「推奨セキュリティ型」「セキュリティ例外型」）を提示している。

実際にどの設定基準を採用するかは、安全性の確保と相互接続の必要性の両面を鑑みて、サーバ管理やサービス提供に責任を持つ管理者が最終的に決定すべきことではあるが、本ガイドラインでは、安全性もしくは相互接続性についての特段の要求がなければ「推奨セキュリティ型」の採用を強く勧める。本ガイドラインの作成時点（2024 年 3 月）では、「推奨セキュリティ型」がもっとも安全性と相互接続性のバランスが取れている要求設定であると考えている。

Appendix には、4 章から 6 章までの設定状況を確認するためのチェックリスト等を記載している。チェックリストの目的は、「選択した設定基準に対応した要求設定の設定忘れの防止」と「サーバ構築の作業受託先が適切に要求設定を遵守したことの確認」を行うための手段となるものである。

1.2 本書が対象とする読者

本ガイドラインでは、主に Web に TLS を利用するシステムを対象に、TLS サーバを実際に構築するにあたって具体的な設定を行うサーバ構築者、実際のサーバ管理やサービス提供に責任を持つことになるサーバ管理者、並びに TLS サーバの構築を発注するシステム担当者を想定読者としている。一部の内容については、ブラウザを使う一般利用者への注意喚起も対象とする。

なお、Web 以外の TLS プロトコルの利用については、参照できる部分も多いと考えているが、例えば鍵事前共有型（PSK: Pre-Shared Key）の利用方法など、十分に検討されていない項目があることに留意されたい。

1.3 ガイドラインの検討体制

本ガイドラインの Ver.3.0.x への改訂にあたっては、CRYPTREC 暗号技術活用委員会の配下に TLS 暗号設定ガイドラインワーキンググループを設置した。Ver.3.0.x は同ワーキンググループに参加する委員の知見を集約したベストプラクティスとして作成されたものであり、暗号技術活用委員会の承認を得て発行された。

TLS 暗号設定ガイドラインワーキンググループは表 1 のメンバーにより構成されている。

さらに、Ver. 3.1 への改訂は、CRYPTREC 暗号技術活用委員会（表 2）にて審議を行い、その結果を受けて発行するものである。

表 1　TLS 暗号設定ガイドラインワーキンググループの構成（2020 年 6 月時点）

主査	須賀　祐治	株式会社インターネットイニシアティブ　セキュリティ本部 セキュリティ情報統括室　シニアエンジニア
委員	漆嶌　賢二	GMO グローバルサイン株式会社 プロダクトマネジメント部　部長
委員	垣内　由梨香	マイクロソフト株式会社　セキュリティレスポンスチーム セキュリティプログラムマネージャー
委員	菅野　哲	株式会社レピダム　代表取締役
委員	菊池　浩明	明治大学　総合数理学部　先端メディアサイエンス学科 教授
委員	北村　英志	グーグル合同会社　デベロッパーリレーションズ デベロッパーアドボケイト
委員	島岡　政基	セコム株式会社　IS 研究所 コミュニケーションプラットフォームディビジョン 主任研究員
委員	杉尾　信行	株式会社 NTT ドコモ　情報セキュリティ部
委員	杉原　弘祐	セコムトラストシステムズ株式会社 情報セキュリティサービス本部　セキュアサービス 1 部
委員	松本　照吾	アマゾンウェブサービスジャパン株式会社 パブリックセクター コンサルティング本部 シニアセキュリティコンサルタント

表 2　2023 年度暗号技術活用委員会の構成（2024 年 3 月時点）

委員長	松本　勉	横浜国立大学　大学院環境情報研究院 教授
委員	上原　哲太郎	立命館大学　情報理工学部　情報理工学科 教授
委員	垣内　由梨香	Microsoft Corporation セキュリティレスポンスチーム セキュリティプログラムマネージャー
委員	菊池　浩明	明治大学　総合数理学部　先端メディアサイエンス学科 教授
委員	佐藤　直之	SCSK 株式会社 ソリューション事業グループ クラウドサービス事業本部　セキュリティサービス部 シニアプロフェッショナルコンサルタント
委員	佐藤　雅史	セコム株式会社　IS 研究所 デジタルプラットフォームディビジョン 主幹研究員
委員	須賀　祐治	株式会社インターネットイニシアティブ セキュリティ本部　セキュリティ情報統括室 シニアエンジニア
委員	田村　裕子	日本銀行金融研究所　情報技術研究センター 企画役
委員	手塚　悟	慶應義塾大学　環境情報学部 教授
委員	寺村　亮一	GMO サイバーセキュリティ by イエラエ株式会社 執行役員
委員	三澤　学	三菱電機株式会社 情報技術総合研究所　情報ネットワークシステム技術部 グループマネージャ
委員	満塩　尚史	デジタル庁 戦略・組織グループ　セキュリティ危機管理チーム セキュリティアーキテクト
委員	山口　利恵	東京大学　附属情報理工学教育センター 知能社会創造研究部門 准教授
委員	渡邊　創	国立研究開発法人産業技術総合研究所 サイバーフィジカルセキュリティ研究センター 副研究センター長

2. 本ガイドラインの理解を助ける技術的な基礎知識

2.1 TLS の概要

2.1.1 TLS の歴史

Secure Sockets Layer（SSL）はブラウザベンダであった Netscape 社により開発されたクライアントーサーバモデルにおけるセキュアプロトコルである。SSL には 3 つのバージョンが存在するがバージョン 1.0 は非公開である。SSL2.0 が 1995 年にリリースされたが、その後すぐに脆弱性が発見され、翌 1996 年に SSL3.0 [RFC6101]が公開されている。

標準化団体 Internet Engineering Task Force（IETF）[1]はベンダ間での非互換性の問題を解決するために、Transport Layer Security Protocol Version 1.0（TLS1.0）[RFC2246]を策定した。TLS1.0 は SSL3.0 をベースにしている。TLS1.0 で定められているプロトコルバージョンからも分かるように TLS1.0 は SSL3.1 とも呼ばれる。

TLS1.1 [RFC4346]は、TLS1.0 における暗号利用モードの一つである CBC[2]モードで利用される初期ベクトルの選択とパディングエラー処理に関連する脆弱性に対処するために仕様策定が行われた。具体的には BEAST[3]攻撃を回避することができる。

TLS1.2 [RFC5246]は、特にハッシュ関数 SHA-2 family（SHA-256 や SHA-384）の利用など、より強い暗号アルゴリズムの利用が可能になった。例えばメッセージ認証コード（MAC[4]）や擬似乱数関数にて SHA-2 family が利用可能になっている。また認証暗号が利用可能な暗号スイートのサポートがなされており、具体的には GCM[5]や CCM[6]モードの利用が可能になった。

TLS1.3 [RFC8446]は、TLS1.2 策定以降に見つかった新たな脆弱性や攻撃手法への対策を施すと共に、QUIC（現在 IETF で標準化が進められているトランスポートプロトコル。内部的に TLS1.3 を利用する）等のプロトコルに対応するための性能向上を狙いとして、プロトコルと暗号アルゴリズムの抜本的な再設計が行われた。

表 3 に TLS のバージョンの概要をまとめる。なお、SSL/TLS に対する攻撃方法の技術的な詳細については、「CRYPTREC 暗号技術ガイドライン（SSL/TLS における近年の攻撃への対応状況）[7]」を参照されたい。

[1] https://ietf.org/
[2] CBC: Cipher Block Chaining
[3] BEAST: Browser Exploit Against SSL/TLS
[4] MAC: Message Authentication Code
[5] GCM: Galois/Counter Mode
[6] CCM: Counter with CBC-MAC
[7] https://www.cryptrec.go.jp/report/cryptrec-gl-2002-2013.pdf

表 3　TLS のバージョン概要

バージョン	概要
SSL2.0 （1994）	● いくつかの脆弱性が発見されており、なかでも「ダウングレード攻撃（最弱のアルゴリズムを強制的に使わせることができる）」と「バージョンロールバック攻撃（SSL2.0 を強制的に使わせることができる）」は致命的な脆弱性といえる ● SSL2.0 は利用すべきではなく、2005 年頃以降に出荷されているサーバやブラウザでは SSL2.0 は初期状態で利用不可となっている
SSL3.0 （RFC6101） （1995）	● SSL2.0 での脆弱性に対処したバージョン ● 2014 年 10 月に POODLE[8]攻撃が発表されたことにより特定の環境下での CBC モードの利用は危険であることが認識されている。POODLE 攻撃は、SSL3.0 におけるパディングチェックの仕方の脆弱性に起因しているため、この攻撃に対する回避策は現在のところ存在していない ● SSL3.0 は利用すべきではなく、2018 年頃以降に出荷されているサーバやブラウザでは SSL3.0 は初期状態で利用不可となっている
TLS1.0 （RFC2246） （1999）	● ブロック暗号を CBC モードで利用した時の脆弱性を利用した攻撃（BEAST 攻撃など）が広く知られているが、容易な攻撃回避策が存在し、すでにセキュリティパッチも提供されている。また、SSL3.0 で問題となった POODLE 攻撃は、プロトコルの仕様上、TLS1.0 には適用できない ● 暗号スイートとして、より安全なブロック暗号の AES と Camellia、並びに公開鍵暗号・署名に楕円曲線暗号が利用できるようになった ● 秘密鍵の生成などに擬似乱数関数を採用 ● MAC の計算方法を HMAC に変更 ● 2023 年 11 月時点での SSL Pulse の調査結果によれば、約 15 万の主要なサイトについて TLS1.0 が利用できるのは 29.5%
TLS1.1 （RFC4346） （2006）	● ブロック暗号を CBC モードで利用した時の脆弱性を利用した攻撃（BEAST 攻撃等）への対策を予め仕様に組み入れる等、TLS1.0 の安全性強化を図っている ● 2023 年 11 月時点での SSL Pulse の調査結果によれば約 15 万の主要なサイトについて TLS1.1 が利用できるのは 31.8%
TLS1.2 （RFC5246） （2008）	● 暗号スイートとして、より安全なハッシュ関数 SHA-256 と SHA-384、CBC モードより安全な認証付き秘匿モード（GCM、CCM）が利用できるようになった。特に、認証付き秘匿モードでは、利用するブロック暗号が同じであっても、CBC モードの脆弱性を利用した攻撃（BEAST 攻撃等）がそもそも適用できない ● 必須の暗号スイートを TLS_RSA_WITH_AES_128_CBC_SHA に変更 ● IDEA と DES を使う暗号スイートを削除 ● 擬似乱数関数の構成を MD5/SHA-1 ベースから SHA-256 ベースに変更 ● 2023 年 11 月時点での SSL Pulse の調査結果によれば約 15 万の主要なサイトについて TLS1.2 が利用できるのは 99.9%

[8] POODLE: Padding Oracle On Downgraded Legacy Encryption

バージョン	概要
TLS1.3 （RFC8446） （2018）	● 暗号スイートの表記方法が変更。署名と鍵交換を暗号スイートから分離 ● ハンドシェイク性能の向上のため、1-RTT、0-RTT (Round Trip Time)になるようにシーケンスが簡素化された ● Perfect Forward Secrecy (PFS)を実現するため、静的な RSA、DH を削除 ● ハンドシェイクのデータを暗号化して保護 ● HMAC ベースの導出関数（HKDF-Expand(・), HKDF-Extract(・)）を使った鍵導出に変更 ● TLS1.2 互換に配慮し、ClientHello、ServerHello、ChangeCipherSpec が規定された ● リネゴシエーション、圧縮が削除 ● Triple DES、DSA、RC4、MD5、SHA-1、SHA-224、認証暗号（AEAD: Authenticated Encryption with Associated Data）でない CBC モードを削除 ● 共通鍵暗号は AES-GCM、AES-CCM、ChaCha20-Poly1305 のみが規定された。このうち、AES-GCM が必須、ChaCha20-Poly1305 が推奨になった ● 鍵交換は DHE、ECDHE、PSK のみが規定され、いずれかの利用が必須になった ● 署名は RSA-PSS、RSASSA-PKCS1-v1_5、ECDSA が必須になった ● ハッシュ関数は SHA-256 以上が規定された。このうち、SHA-256 が必須になった ● 楕円曲線は secp256r1（P-256）が必須に、X25519（Curve25519）が推奨になった ● 2023 年 11 月時点での SSL Pulse の調査結果によれば約 15 万の主要なサイトについて TLS1.3 が利用できるのは 66.2%

2.1.2 SSL/TLS プロトコル概要

SSL/TLS はセッション層に位置するセキュアプロトコルで、通信の暗号化、データ完全性の確保、サーバ（場合によりクライアント）の認証を行うことができる。セッション層に位置することで、アプリケーション層ごとにセキュリティ確保のための仕組みを実装する必要がなく、HTTP、SMTP、POP など様々なプロトコルの下位に位置して、上記の機能を提供することができる。

SSL/TLS では、暗号通信を始めるに先立って、ハンドシェイクが実行される（図 1 参照）。

ハンドシェイクでは、①ブラウザ（クライアント）とサーバが暗号通信するために利用する暗号アルゴリズムとプロトコルバージョンを決定し、②サーバ証明書によってサーバの認証を行い、③そのセッションで利用するセッション鍵を共有する、までの一連の動作を行う。

その際、SSL/TLS では相互接続性の確保を優先してきたため、一般には複数の暗号アルゴリズムとプロトコルバージョンが実装されている。結果として、暗号通信における安全性強度は、ハンドシェイクの①の処理でどの暗号アルゴリズムとプロトコルバージョンを選択したかに大きく依存する。

サイトの身分証明ともいえるサーバ証明書は、Trusted Third Party である認証局（CA[9]）によっ

[9] Certificate Authority

て発行されるのが一般的であり、特に Web Trust for CA などの一定の基準を満たしている代表的な認証局の証明書はルート CA として予めブラウザに登録されている。図 1 の(4)の検証では、ブラウザに登録された認証局の証明書を信頼の起点として、当該サーバ証明書の正当性を確認する。

なお、本書では、TLS サーバ認証用途向けデジタル証明書を「サーバ証明書」と呼んでいることに注意されたい。

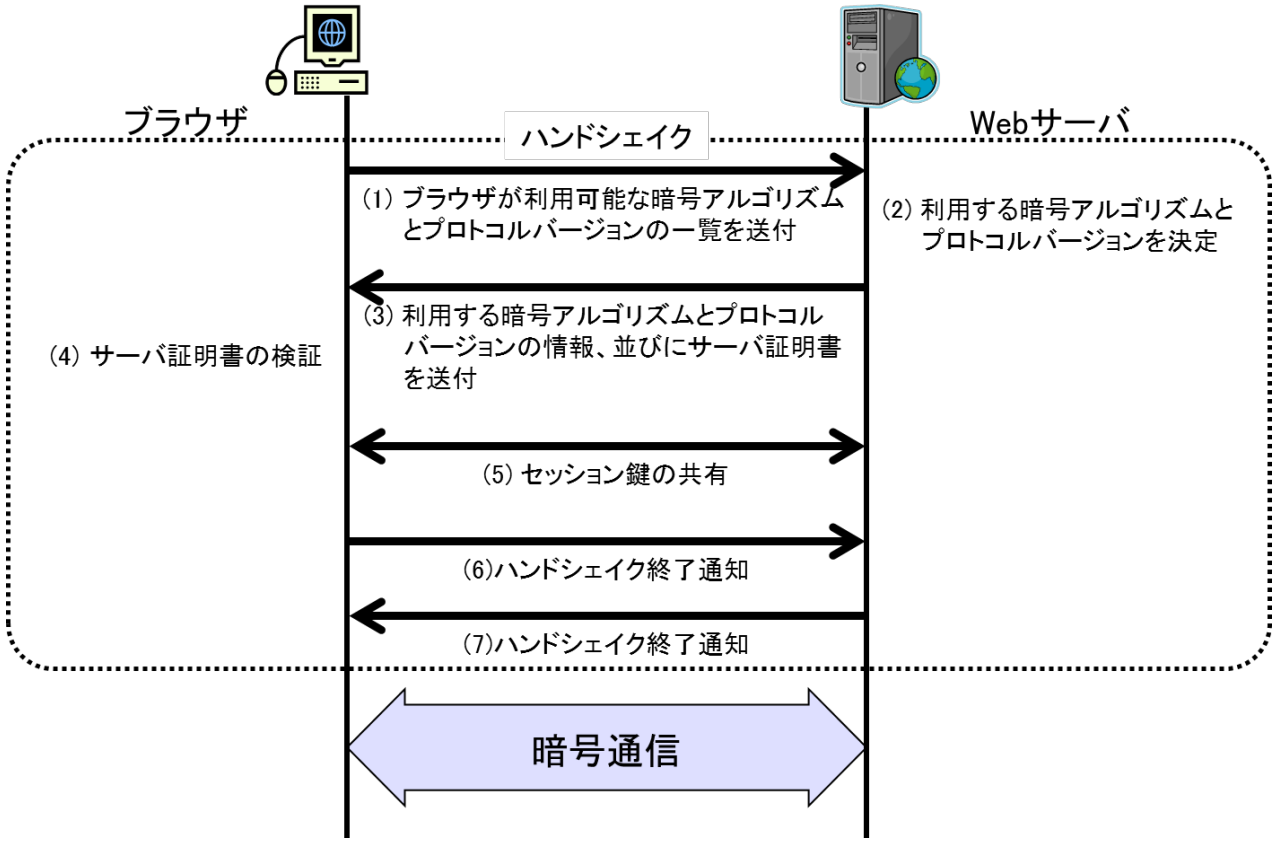


図 1　TLS プロトコル概要

2.1.3 TLS1.3 の概要

TLS1.3 は、TLS1.2 策定以降に見つかった新たな脆弱性や攻撃手法への対策を施すと共に、QUIC 等のプロトコルに対応するための性能向上を狙いとして、プロトコルと暗号アルゴリズムの抜本的な再設計が行われた。

(1) ServerHello 以降のハンドシェイクパラメータを暗号化して保護する。

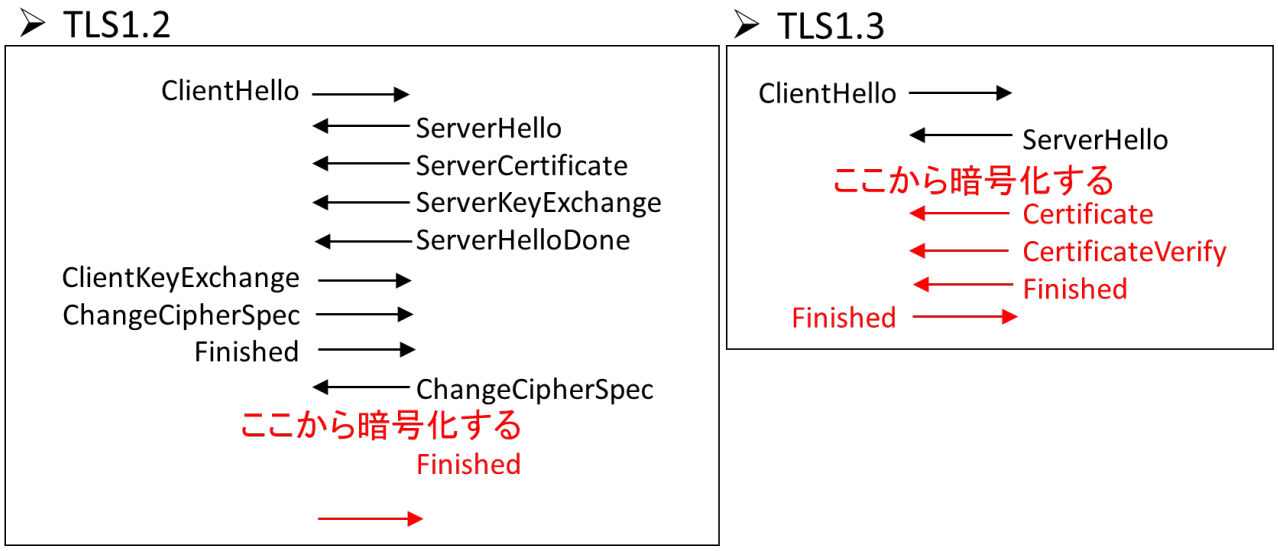


図 2　TLS1.2 と TLS1.3 との暗号化開始個所の比較

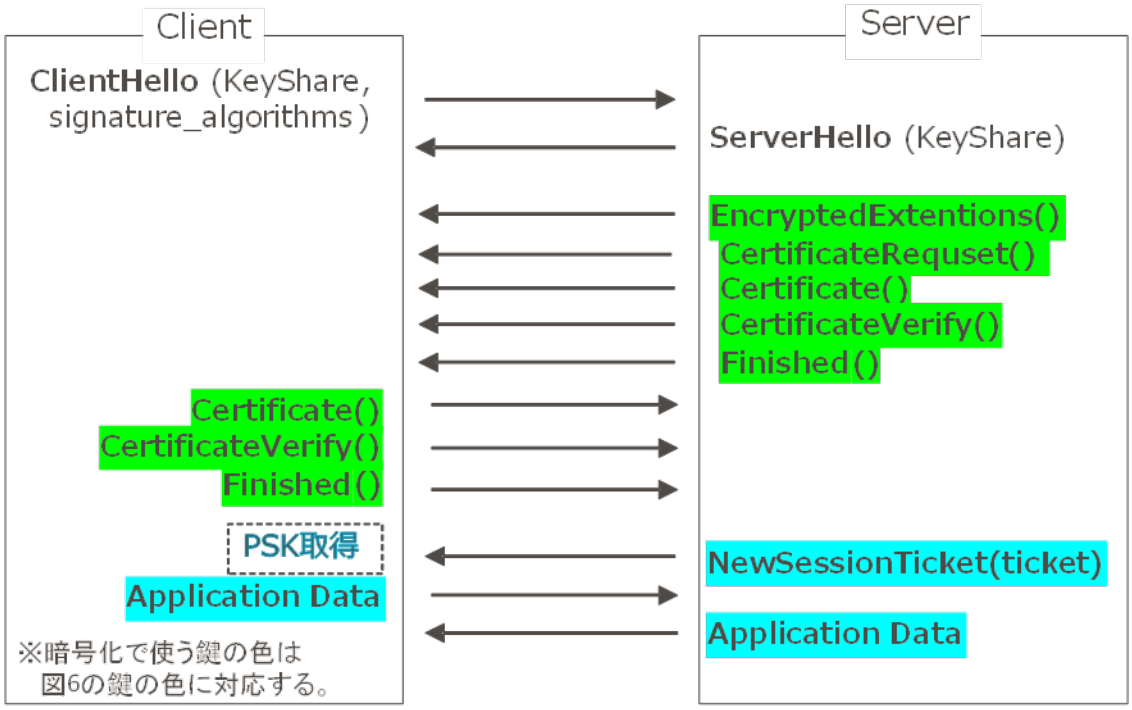


図 3　TLS1.3 のシーケンス図

(2) 性能向上のため、1-RTT でハンドシェイクが完了するようにメッセージ及び拡張が見直された。

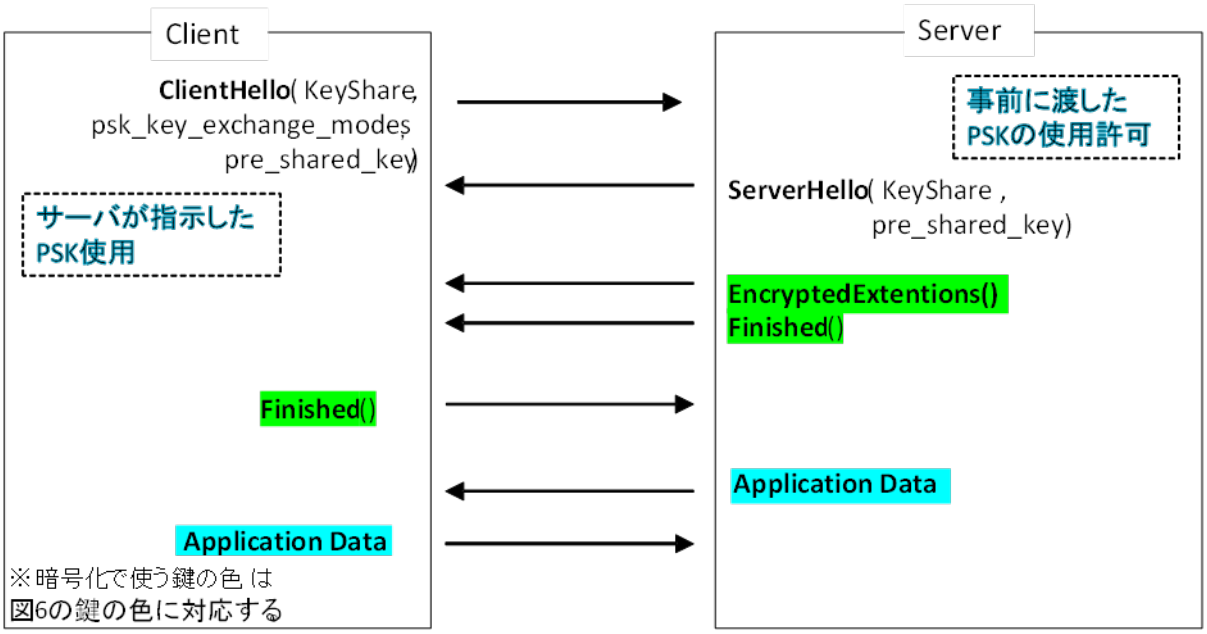


図 4　1-RTT のシーケンス図

(3) 0-RTT でアプリケーションデータを送信する機能が追加された。

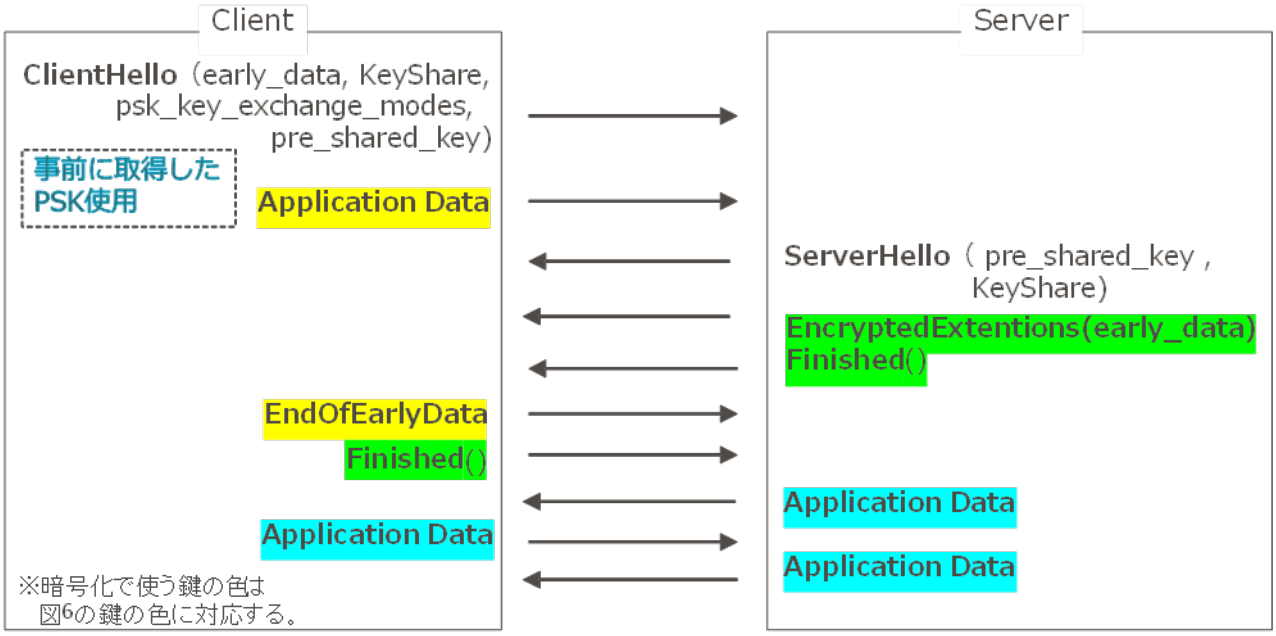


図 5　0-RTT のシーケンス図

(4) ClientHello、ServerHello、ChangeCipherSpec の TLS1.2 互換性を保つことにより、中間ノードによる接続性を向上した。

(5) HMAC ベースの導出関数（HKDF-Expand(・), HKDF-Extract(・)）を使った鍵導出に変更された。

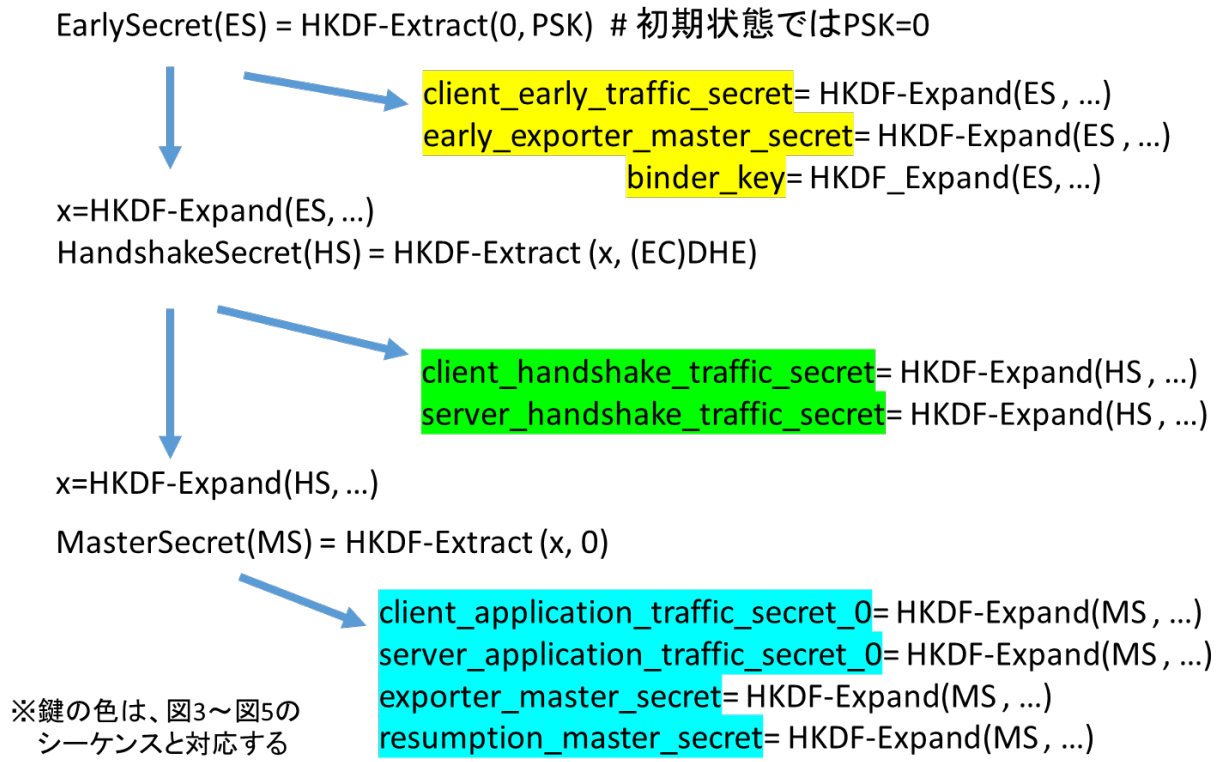


図 6　鍵の導出方法

2.2 プロトコルバージョンごとの安全性の違い

SSL/TLS は、1994 年に SSL2.0 が実装され始めた後、2024 年 3 月現在の最新版となる TLS1.3 まで 6 つのプロトコルバージョンが実装されている。基本的に、プロトコルのバージョンが後になるほど、以前の攻撃に対する対策が盛り込まれるため、より安全性が高くなる。しかし、相互接続性も確保する観点から、多くの場合、複数のプロトコルバージョンが利用できるように実装されているので、プロトコルバージョンの選択順位を正しく設定しておかなければ、予想外のプロトコルバージョンで SSL/TLS 通信を始めることになりかねないことに留意されたい。なお、プロトコルバージョンの詳細な要求設定については、設定基準に対応する該当章を参照すること。

SSL2.0 から TLS1.3 までの各プロトコルバージョンにおける安全性の違いを表 4 にまとめる。なお、表 4 では BEAST 攻撃のように「プロトコルの仕様上の脆弱性」のみを対象とすることと

し、OpenSSL Heartbleed Bug のように「実装に伴う脆弱性」は対象外としている。また、仕様の記載内容があいまいで当該プロトコルバージョン中の一部のパラメータ等の使用の可否が実装者の解釈に依存し、その結果、実装上の脆弱性が発生した、または発生する余地があったものであって、その後に仕様の補正・明確化が行われたような、必ずしも「仕様上の脆弱性」とまでは言い切れないものも対象外としている。

表 4　プロトコルバージョンでの安全性の違い

SSL/TLS 攻撃方法に対する耐性	TLS1.3	TLS1.2	TLS1.1	TLS1.0	SSL3.0	SSL2.0
ダウングレード攻撃（最弱の暗号アルゴリズムを強制的に使わせることができる）	安全	安全	脆弱	脆弱	脆弱	脆弱
バージョンロールバック攻撃（意図したよりも古いバージョンを強制的に使わせることができる）	安全	安全	安全	安全	脆弱	脆弱
ブロック暗号の CBC モード利用時の脆弱性を利用した攻撃（BEAST/POODLE 攻撃など）	安全	安全	安全	パッチ適用要	脆弱	脆弱

凡例：

- 濃い赤の「脆弱」：
  2015 年 5 月以前に発見され、修正パッチによっても回避できない「仕様上の脆弱性」
- オレンジの「脆弱」：
  2015 年 5 月以降に発見され、修正パッチによっても回避できない「仕様上の脆弱性」
- 黄色の「パッチ適用要」：
  多くの製品実装に影響を及ぼすが、修正パッチによって対策が可能な「仕様上の脆弱性」
- 緑色の「安全」：
  2024 年 3 月時点で上記に該当する「仕様上の脆弱性」が発見されていない状態。なお、「実装に伴う（製品の）脆弱性」は上記の「仕様上の脆弱性」に含めない。

2.3 サーバ証明書についての概要

サーバ証明書は、①当該サーバが、意図する相手によって管理されているサーバであることを確認する手段をクライアントに対して提供することと、②SSL/TLS による暗号通信を行うために必要なサーバの公開鍵情報をクライアントに正しく伝えること、の 2 つの役割を持っている。

サーバ証明書には、表 5 に示す情報が記載されている。これらの情報は、証明書プロパティの「詳細」で見ることができる。

これらのうち、本ガイドラインにおいて重要な項目は以下の 3 点である。

- 署名アルゴリズム
- 公開鍵情報（subject Public Key Info）に含まれる公開鍵の暗号アルゴリズム、パラメータ、鍵長
- 拡張情報のひとつ subjectAltName.dNSName に記載される、サーバの FQDN

表 5　サーバ証明書に記載される情報

証明書のバージョン	Version
シリアル番号	Serial Number
署名アルゴリズム	Certificate Signature Algorithm
発行者	Issuer
有効期間（開始～終了）	Validity (Not Before ～ Not After)
サブジェクト（発行対象）	Subject
（サブジェクトが使う）公開鍵情報[10]	Subject Public Key Info (Algorithm, Public Key Value)
拡張情報	Extensions
キー使用法	Certificate Key Usage
署名	Certificate Signature Value

2.4 暗号スイートについての概要

TLS1.2 までの暗号スイートは「鍵交換__署名__暗号化__ハッシュ関数」の組によって構成される。例えば、「TLS_DHE_RSA_WITH_CAMELLIA_256_GCM_SHA384」であれば、鍵交換には「DHE」、署名には「RSA」、暗号化には「鍵長 256 ビット GCM モードの Camellia（CAMELLIA_256_GCM）」メッセージ認証コード（HMAC）を作るハッシュ関数には「SHA-384」が使われることを意味する。「TLS_RSA_WITH_AES_128_CBC_SHA」であれば、鍵交換と署名には「RSA」、暗号化には「鍵長 128 ビット CBC モードの AES（AES_128_CBC）」、HMAC に使うハッシュ関数には「SHA-1」が使われることを意味する。

ただし、TLS 用の暗号スイートは IETF で規格化されており、任意に暗号アルゴリズムを選択して「鍵交換__署名__暗号化__ハッシュ関数」の組を自由に作れるわけではない。また、IETF で規格化されている暗号スイートだけでも数多くあるため、実際の製品には実装されていない暗号スイートも多い。

TLS1.3 の暗号スイートでは、TLS1.2 までの暗号スイートの組から「鍵交換」と「署名」が外さ

[10] Windows の証明書プロパティでは『公開キー』と表記されているが、本文中では『公開鍵』で表記を統一する。

れ、「暗号化__ハッシュ関数」[11]だけの構成に変更となった。

実際の TLS 通信においては、サーバとクライアント間での事前通信（ハンドシェイク）時に、両者の合意により一つの暗号スイート（TLS1.3 の場合は「暗号スイート」の他、「鍵交換」と「署名」も含めて）を選択する。暗号スイートが選択された後は、選択された暗号スイートに記載の鍵交換、署名、暗号化、ハッシュ関数の方式により TLS における各種処理が行われる。

このため、TLS における安全性にとって、暗号スイートをどのように設定するかが最も重要なファクタであることを意味する。一般に、暗号スイートの優先順位の上位から順にサーバとクライアントの両者が合意できる暗号スイートを見つけていくので、暗号スイートの選択のみならず、優先順位の設定が重要となる。

多くのブラウザ（クライアント）との相互接続性を確保するためには、対象とするブラウザ（クライアント）に実装されている暗号スイートを幅広く受け入れる設定をすることになる。ただし、一定以上の安全性を確保するためには、一部の旧式のブラウザ（クライアント）との相互接続性を断念してでも、一定以上の安全性を持つ暗号アルゴリズムで構成される暗号スイートを設定する必要があることにも留意されたい。

2.5 本ガイドラインでの暗号アルゴリズムに対する考え方

2.5.1 サーバ証明書で利用する暗号アルゴリズムに対する考え方

本ガイドラインにおいては、サーバ証明書の仕様に合致するものに採用されている「署名」及び「ハッシュ関数」のうち、CRYPTREC 暗号リスト（2.6.1 節参照）の電子政府推奨暗号リストまたは推奨暗号候補リストに掲載されているものの中から原則としてサーバ証明書で利用する暗号アルゴリズムを選択する。

具体的には、表 6 に示した「署名」及び「ハッシュ関数」がサーバ証明書で利用可能な暗号アルゴリズムである。例外は DSA と EdDSA である。

DSA については、電子政府推奨暗号リストに選定されており安全性上の問題はないが、デジタル署名の米国政府標準暗号 FIPS186-5 からも仕様が削除された[12]ことから、今後、新規利用時や更新時は DSA を利用すべきではない。また、EdDSA については、パブリック認証局が発行するサーバ証明書の要件を規定する CA/ブラウザフォーラムのベースライン要求（2023 年 8 月 17 日発行分）[13]において利用できる署名アルゴリズムに指定されていないことから、現時点ではサーバ証明書で利用可能な暗号アルゴリズムに EdDSA を含めない。

[11] TLS1.3 の暗号スイートで指定するハッシュ関数は HMAC ベースの鍵導出関数(HKDF)に使われる（2.1.3 節参照）。
[12] FIPS186-5 では DSA を用いた新たな署名の生成は承認されず、既存の署名の検証にのみ利用が認められている。
[13] https://cabforum.org/wp-content/uploads/CA-Browser-Forum-BR-v2.0.1.pdf

表 6　サーバ証明書で利用可能な暗号アルゴリズム

技術分類	リストの種類	アルゴリズム名	備考
署名	電子政府推奨暗号リスト	RSASSA-PKCS1-v1_5	
		RSA-PSS	
		ECDSA	
		DSA	今後、新規利用時や更新時は利用すべきでない
ハッシュ関数	電子政府推奨暗号リスト	SHA-256	
		SHA-384	
		SHA-512	現在までに利用実績はほとんどない

2.5.2 暗号スイートで利用する暗号アルゴリズムに対する考え方

本ガイドラインにおいては、SSL/TLS 用の暗号スイートとして IANA TLS Cipher Suites に登録されているもの及び RFC8446 に採用されている暗号アルゴリズムのうち、CRYPTREC 暗号リスト（2.6.1 節参照）に掲載されているものの中から原則として暗号スイートで利用する暗号アルゴリズムを選択する。

本節では、本ガイドラインにおける暗号スイートで利用する暗号アルゴリズムに対する原則的な考え方を示す。最終的にどの暗号アルゴリズムが利用できるかについては、どの設定基準（3.1 節参照）を選択したかにより異なるため、詳細については該当章を参照されたい。

■（全ての設定基準における）暗号スイートでの利用推奨暗号アルゴリズム

原則的には電子政府推奨暗号リストまたは推奨暗号候補リストに掲載されている暗号アルゴリズムを「(全ての設定基準における) 暗号スイートでの利用推奨暗号アルゴリズム」として規定する。具体的には、表 7 にその一覧を示す。

例外的に、本ガイドラインでは、以下の理由により、DSA 及び EdDSA を利用推奨アルゴリズムとせず、表 7 から除外した。ただし、自らの責任において利用することを妨げるものではない。

DSA については、電子政府推奨暗号リストに選定されており安全性上の問題はないが、デジタル署名の米国政府標準暗号 FIPS186-5 から仕様が削除されたことから、今後 DSA を利用すべきではない。また、EdDSA については、CA/ブラウザフォーラムのベースライン要求（2023 年 8 月 17 日発行分）の規定に従う場合には、サブジェクトが使う公開鍵情報に EdDSA を設定できない。このため、TLS において鍵交換時に付与する署名のアルゴリズムとしても EdDSA を利用できないことに留意されたい。

表 7　暗号スイートでの利用推奨暗号アルゴリズム

	CRYPTREC 暗号リストでの標記			備考
	技術分類	リストの種類	アルゴリズム名	
鍵交換	鍵共有・守秘	電子政府推奨暗号リスト	DHE (Ephemeral DH)	PFS 特性をもつ DHE を選択するのがセキュリティ上望ましい
			ECDHE (Ephemeral ECDH)	PFS 特性をもつ ECDHE を選択するのがセキュリティ上望ましい
署名	署名	電子政府推奨暗号リスト	ECDSA	
			RSASSA-PKCS1-v1_5	
			RSA-PSS	TLS1.3 のみ利用可能
暗号化	128 ビットブロック暗号	電子政府推奨暗号リスト	AES	
			Camellia	TLS1.2 までで利用可能
	暗号利用モード	電子政府推奨暗号リスト	CCM	
			CCM_8	CCM の縮退版なので、CCM が利用できるのであれば、CCM を利用することが望ましい
			GCM	
	認証暗号	電子政府推奨暗号リスト	ChaCha20-Poly1305	
ハッシュ関数	ハッシュ関数	電子政府推奨暗号リスト	SHA-256	
			SHA-384	

■設定基準により暗号スイートでの取り扱いが異なる暗号アルゴリズム

運用監視暗号リストに掲載されている暗号アルゴリズムは、安全性に問題があり互換性維持での利用を前提としているため、原則として、本ガイドラインの「設定基準により暗号スイートでの取り扱いが異なる暗号アルゴリズム」として位置付ける。

例外的に、CBC、DH、ECDH については電子政府推奨暗号リストに記載されているが、以下の理由により、「設定基準により暗号スイートでの取り扱いが異なる暗号アルゴリズム」に含めるものとする。

- CBC については、暗号利用モード単体としては一定の安全性を有しており電子政府推奨暗号リストに掲載されているが、実装上の問題によっては TLS1.0 以前のプロトコルバージョンで利用する場合に TLS としての脆弱性につながる要因が存在することが分かっている。
- DHE と ECDHE は、それぞれ電子政府推奨暗号リストに記載されている DH、ECDH とアルゴリズムとしては同じものであるが、プロトコル中での利用方法に違いがあるため、区別し

た表記になっている。2.5.3 節にあるように TLS での利用にあたっては Perfect Forward Secrecy の性質を有する使い方である DHE や ECDHE を含む暗号スイートを選択するほうがセキュリティ上望ましい。

ここに該当する暗号アルゴリズムは具体的には表 8 に示したものであり、選択する設定基準によって「利用推奨」されることも「利用禁止」されることもあり得る。どちらに該当するかは選択する設定基準の該当章を参照されたい。なお、「利用推奨」とも「利用禁止」とも言及されない場合もあるが、その場合には「利用を推奨しないものの、自らの責任において利用することは妨げない」と理解されたい。

表 8 設定基準により暗号スイートでの取り扱いが異なる暗号アルゴリズム

	CRYPTREC 暗号リストでの標記			備考
	技術分類	リストの種類	アルゴリズム名	
鍵交換	鍵共有・守秘	電子政府推奨暗号リスト	DH	DH は PFS 特性を持たないので、PFS 特性をもつ DHE を選択するほうがセキュリティ上望ましい
			ECDH	ECDH は PFS 特性を持たないので、PFS 特性をもつ ECDHE を選択するほうがセキュリティ上望ましい
		運用監視暗号リスト	RSAES-PKCS1-v1_5	脆弱性が見つかっているが、利用実績上、無視できない
暗号化	暗号利用モード	電子政府推奨暗号リスト	CBC	TLS1.1 以前の実装時に脆弱性が見つかっている
ハッシュ関数	ハッシュ関数	運用監視暗号リスト	SHA-1	HMAC での利用のため、耐衝突困難性のリスクは SHA-1 単体の場合より小さい

■利用禁止暗号アルゴリズム

本ガイドラインでは、IANA TLS Cipher Suites に登録及び RFC8446 に採用されている暗号スイートで使われている暗号アルゴリズムのうち、CRYPTREC として安全性評価を実施しておらず安全性が不明であったり実際に暗号解読されたりするなどの技術的要因、もしくは輸出規制や暗号政策をはじめとする政策的要因などにより、**いかなる設定基準であっても利用すべきではない暗号アルゴリズムを「利用禁止暗号アルゴリズム」として新たに規定する。原則的には CRYPTREC 暗号リストに掲載されていない暗号アルゴリズムが該当し、具体的には、表 9 に示した暗号アルゴリズム（2023 年 11 月 30 日時点）である。**

例外は以下である。

- 3-key Triple DES は CRYPTREC 暗号リストの運用監視暗号リストに掲載されているが、SSL3.0

以前で主に使われていた暗号アルゴリズムであり、TLS1.0 以降では AES や Camellia などが利用可能である。本ガイドラインで SSL3.0 を利用禁止とすることから、3-key Triple DES を表 9 に含めるものとする。

表 9　暗号スイートでの利用禁止暗号アルゴリズム（2023 年 11 月 30 日時点）

	CRYPTREC 暗号リストでの標記			備考
	技術分類	リストの種類	アルゴリズム名	
署名	署名	対象外	GOST R 34.10-2012	
			SM2（署名）	
暗号化	64 ビット ブロック暗号	対象外	RC2, EXPORT-RC2	
			IDEA	
			DES, EXPORT-DES	
			GOST 28147-89	
			Magma	
		運用監視暗号リスト	3-key Triple DES	SSL3.0 を利用禁止にしたため
	128 ビット ブロック暗号	対象外	Kuznyechik	
			SM4	
			ARIA	
			SEED	
	暗号利用モード	対象外	CTR_OMAC	
	ストリーム暗号	対象外	RC4, EXPORT-RC4	
ハッシュ関数	ハッシュ関数	対象外	MD5	
			GOST R 34.11-2012	
			SM3	

2.5.3 Perfect Forward Secrecy の重要性－秘密鍵漏えい時の影響範囲を狭める手法

TLS の仕様では、実際のデータを暗号化する際に利用する“セッション鍵”はセッションごとに（あるいは任意の要求時点で）更新される。したがって、何らかの理由により、ある時点でのセッション鍵が漏えいした場合でも、当該セッション以外のデータは依然として保護された状態にある。

一方、セッション鍵は暗号通信を始める前にサーバとクライアントとで共有しておく必要があるため、事前通信（ハンドシェイク）の段階でセッション鍵を共有するための処理が行われる。この処理のために使われるのが、「鍵共有・守秘」に掲載されている暗号アルゴリズムである。秘密鍵が漏えいする原因は暗号アルゴリズムの解読によるものばかりではない。むしろ、プログラムなどの実装ミスや秘密鍵の運用・管理ミス、あるいはサイバー攻撃やウイルス感染によるものなど、暗号アルゴリズムの解読以外が原因となって秘密鍵が漏えいする場合のほうが圧倒的に多い。

過去には、OpenSSL Heartbleed Bug や Dual_EC_DRBG の脆弱性などが原因による秘密鍵の漏えいといった事件も起きており、“秘密鍵が漏えいする”リスクそのものは決して無視できるものではない。スノーデン事件でも話題になったように、秘密鍵の運用・管理そのものに問題がある場合も想定される。

上述した通り、TLS では、毎回変わるセッション鍵をサーバとクライアントが共有することでセッションごとに違った秘密鍵を使って暗号通信をしており、仮にある時点でのセッション鍵が漏えいした場合でも当該セッション以外のデータは依然として保護されている。

しかし、多くの場合、セッション鍵の交換には固定の鍵情報を使って行っている。このため、どんな理由であれ、もし仮に鍵交換で使う暗号アルゴリズムの“秘密鍵”が漏えいした場合、当該秘密鍵で復号できるセッション鍵はすべて漏えいしたことと同義となる。つまり、TLS での通信データをためておき、年月が経って、当時の鍵交換で使った暗号アルゴリズムの“秘密鍵”が入手できたならば、過去にさかのぼって、ためておいた通信データの中身が読み出せることを意味している。

そこで、過去の TLS での通信データの秘匿を確保する観点から、鍵交換で使った暗号アルゴリズムの“秘密鍵”に毎回異なる乱数を付加することにより、見かけ上、毎回異なる秘密鍵を使ってセッション鍵の共有を行うようにする方法がある。これによって、仮に鍵交換で使う暗号アルゴリズムの“秘密鍵”が何らかの理由で漏えいしたとしても、当該セッション鍵の共有のために利用した乱数がわからなければ、当該セッション鍵そのものは求められず、過去に遡及して通信データの中身が読まれる危険性を回避することができる。

このような性質のことを、Perfect Forward Secrecy、または単に Forward Secrecy と呼んでいる。なお、本ガイドラインでは Perfect Forward Secrecy（あるいは PFS）に統一して呼ぶこととする。

現在の TLS で使う暗号スイートの中で、Perfect Forward Secrecy の特性を持つのは Ephemeral DH と Ephemeral ECDH と呼ばれる方式であり、それぞれ DHE、ECDHE と表記される。

2.5.4 DH(E)/ECDH(E)での鍵長設定についての注意

鍵交換について、暗号スイート上は鍵長の規定がない。これは、鍵交換の安全性は鍵長にも大きく影響されるためであり、通常同じアルゴリズムであれば鍵長が長いほど安全性を高くすることができる。ただし、あまりにも長くしすぎると処理時間や消費リソース等が過大にかかるようになり、実用性を損なうことにつながる。このため、安全性と処理性能、消費リソース等のトレードオフを考えて適切な鍵長を選択する必要がある。

また、様々な鍵長に対応できるように、同じ暗号スイートを使っていても利用可能な鍵長は製品依存になっている。特に、鍵交換で RSA を使う場合と、DH(E)や ECDH(E)を使う場合とでは、鍵長の扱いが全く異なることに留意する必要がある。

RSA での鍵交換を行う場合にはサーバ証明書に記載された公開鍵を使うことになっており、現在では鍵長 2048 ビットの公開鍵がサーバ証明書に通常記載されている。このことは、RSA での鍵交換を行う場合、サーバ証明書を正当に受理する限り、どのサーバもブラウザも当該サーバ証

明書によって利用する鍵長が 2048 ビットにコントロールされていることを意味する。例え鍵長 2048 ビットの RSA が使えないブラウザがあったとしても、鍵交換が不成立・通信エラーになるだけであり、2048 ビット以外の鍵長が使われることはない。

一方、DH(E)と ECDH(E)については、利用する鍵長がサーバ証明書で明示的にコントロールされるのではなく、個々のサーバやブラウザでの鍵パラメータの設定によって決められる。このため、どの鍵長が利用されるかは、使用する製品での鍵パラメータの設定状況に大きく依存する。

このうち、ECDH(E)については比較的最近になって急速に普及してきたことから、当初からセキュリティを考慮して通常 256 ビット以上の鍵長を利用するように実装されている。このため、鍵長をデフォルト設定のままで利用したとしても、セキュリティ上の問題につながるような短い鍵長が使われる可能性はかなり低いと言える。

ところが、DH(E)については、古く SSL の時代から利用されていたこともあり、デフォルト設定での鍵長が製品やバージョンによって大きく異なることが知られている。実際、2.7 節の図 10 の DH 鍵交換の強度からもわかるように、現在の主要なブラウザは鍵長 2048 ビットの DH(E)が利用可能であるにも関わらず、2023 年 11 月時点で 3.5%のサイトで鍵長 1024 ビット以下の DH(E)が使われているものと考えられる。このことは、デフォルト設定で鍵長 1024 ビットが利用可能になっている製品を利用し続けているサイトやレガシーシステムとの相互接続性確保のためにあえて鍵長 1024 ビット以下も利用可能に設定しているサイトが存在していることを示している。

このように、DH(E)を利用する場合には鍵長の設定を正しく行っておかなければ予期せぬタイミングで 1024 ビット以下の弱い鍵長が使われる可能性が依然として残っていることから、利用を許可する鍵長を明示的に適切に設定できない製品を利用する場合には、DH(E)を含む暗号スイートは選択しないようにすべきである。

2.6 暗号アルゴリズムの安全性

2.6.1 CRYPTREC 暗号リスト

デジタル庁と総務省、経済産業省は、暗号技術に関する有識者で構成される CRYPTREC 活動を通して、電子政府で利用される暗号技術の評価を行っている。

2023 年 3 月に、2013 年 3 月に策定した「電子政府における調達のために参照すべき暗号のリスト（CRYPTREC 暗号リスト）」を改定した[14]。CRYPTREC 暗号リストは、「電子政府推奨暗号リスト」、「推奨候補暗号リスト」及び「運用監視暗号リスト」で構成される。それぞれのリストの位置づけは以下の通りである。

- 電子政府推奨暗号リスト：
  CRYPTREC により安全性及び実装性能が確認された暗号技術について、市場における利用実績が十分であるか今後の普及が見込まれると判断され、当該技術の利用を推奨するもののリスト。なお、利用する鍵長について、「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」の規定に合致しない鍵長を用いた場合には、電子政府推奨暗号リストの暗号技術を利用してい

[14] https://www.cryptrec.go.jp/list.html

るとは見なされないことに留意すること。

- 推奨候補暗号リスト：
  CRYPTREC により安全性及び実装性能が確認され、今後、電子政府推奨暗号リストに掲載される可能性のある暗号技術のリスト。なお、本リストに記載されている暗号技術を利用する際は、「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」の規定に合致する鍵長を用いることが求められることに留意すること。
- 運用監視暗号リスト：
  実際に解読されるリスクが高まるなど、推奨すべき状態ではなくなったと CRYPTREC により確認された暗号技術のうち、互換性維持のために継続利用を容認するもののリスト。互換性維持以外の目的での利用は推奨しない。なお、本リストに記載されている暗号技術を利用する際は、「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」の規定に合致する鍵長を用いることが求められることに留意すること。

「政府機関の情報セキュリティ対策のための統一基準（令和 5 年度版）[15]」（令和 5 年 7 月 4 日、サイバーセキュリティ戦略本部）では以下のように記載されており、政府機関における情報システムの調達及び利用において、CRYPTREC 暗号リストのうち「電子政府推奨暗号リスト」が原則的に利用される。

政府機関の情報セキュリティ対策のための統一基準（抄）

7.1.5 暗号・電子署名－遵守事項(1)(b)

情報システムセキュリティ責任者は、暗号技術検討会及び関連委員会（CRYPTREC）により安全性及び実装性能が確認された「電子政府推奨暗号リスト」に基づき、情報システムで使用する暗号及び電子署名のアルゴリズム及び鍵長並びにそれらを利用した安全なプロトコルを定めること。また、その運用方法について実施手順を定めること。

2.6.2 異なる暗号アルゴリズムにおける安全性の見方

異なる技術分類の暗号アルゴリズムを組合せて利用する際、ある技術分類の暗号アルゴリズムの安全性が極めて高いものであっても、別の技術分類の暗号アルゴリズムの安全性が低ければ、結果として、低い安全性の暗号アルゴリズムに引きずられる形で全体の安全性が決まる。逆に言えば、異なる技術分類の暗号アルゴリズムであっても、同程度の安全性とみなされている暗号アルゴリズムを組合せれば、全体としても同程度の安全性が実現できることになる。

異なる技術分類の暗号アルゴリズムについて同程度の安全性を持つかどうかを判断する目安として、“ビットセキュリティ（等価安全性ということもある）”という指標がある。具体的には、評価対象とする暗号アルゴリズムに対してもっとも効率的な攻撃手法を用いたときに、どの程度

[15] https://www.nisc.go.jp/pdf/policy/general/kijyunr5.pdf

の計算量があれば解読できるか（解読計算量[16]）で表現され、鍵長[17]とは別に求められる。表記上、解読計算量が 2^x である場合に“x ビットセキュリティ”という。例えば、共通鍵暗号においては、全数探索する際の鍵空間の大きさで 2^x（x は共通鍵のビット長）、ハッシュ関数の例としては、一方向性で 2^x、衝突困難性で $2^{(x/2)}$（x はハッシュ長）が解読計算量の（最大）理論値である。

“ビットセキュリティ”による評価では、技術分類に関わらず、どの暗号アルゴリズムであっても、解読計算量が大きければ安全性が高く、逆に小さければ安全性が低い。また、解読計算量が実現可能と考えられる計算量を大幅に上回っていれば、少なくとも現在知られているような攻撃手法ではその暗号アルゴリズムを破ることは現実的に不可能であると予測される。

そこで、暗号アルゴリズムの選択においては、“x ビットセキュリティ”の“x ビット”に着目して、長期的な利用期間の目安とする使い方ができる。

「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準[18]」では、CRYPTREC 暗号リストに記載の各アルゴリズムについて、2.2 節にビットセキュリティに対応する鍵長パラメータが規定されている。また、セキュリティ強度要件設定の考え方が 3 節に記載されている（表 10、表 11 参照）。

本ガイドラインでは、この基準に従って、適切な鍵長を設定することを求めることとしている。

表 10 「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」での セキュリティ強度要件の基本設定方針

ビットセキュリティ	利用上の条件	利用期間		
		2022～2030 年	2031～2040 年	2041～2050 年
112 ビット	新規に処理をする場合	移行完遂期間**	利用不可	利用不可
	過去に処理したものを利用する場合		許容*	
128 ビット	新規に処理をする場合	利用可	利用可	移行完遂期間**
	過去に処理したものを利用する場合			
192 ビット	特になし	利用可	利用可	利用可
256 ビット	特になし	利用可	利用可	利用可

* “許容”とは、そのセキュリティ強度の暗号技術では必要なセキュリティ（暗号学的安全性）を確保するには必ずしも十分ではないレベルであると想定され得るが、その正当性を担保又は確認するための何らかの技術的又は運用的な対策やルール等（暗号技術によるものとは限らない）を

[16] 直感的には、基本となる暗号化処理の繰り返し回数のことである。例えば、解読計算量 2^{20} といえば、暗号化処理 2^{20} 回相当の演算を繰り返し行えば解読できることを意味する
[17] ハッシュ関数の場合はハッシュ長（ダイジェスト長ともいう）に相当する
[18] https://www.cryptrec.go.jp/list/cryptrec-ls-0003-2022r1.pdf

併用している場合に、過去に暗号保護が施された保護済みのデータに対して復号や検証の処理を行うことを許容する期間であることを示す。

** “移行完遂期間”とは、そのセキュリティ強度の暗号技術では必要なセキュリティ（暗号学的安全性）を確保するには必ずしも十分ではないレベルになりつつあると想定され、この期間中に、**よりセキュリティ強度の高い暗号技術及び鍵長への移行を完遂させなければならない**期間であることを示す。そのため、利用する暗号処理が短期間で完結する場合（例：エンティティ認証）、又は既存の電子政府システムの継続利用やそれらとの互換性・相互接続性維持のための利用に**限定すべき**であり、新規調達や更新調達を行うシステムにおいて、**既存の電子政府システムとの互換性・相互接続性維持が必要でない場合や代替手段がある場合には、利用を許容すべきではない**ことに留意されたい。

表 11　表 6、表 7、表 8 に記載の暗号アルゴリズム・パラメータに対する「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」での推定セキュリティ強度

セキュリティ強度（ビットセキュリティ）		112	128	192	256
公開鍵暗号（署名・守秘・鍵共有）	素因数分解型（RSA 暗号・RSA 署名）	鍵長 2048 ビット	鍵長 3072 ビット	鍵長 7680 ビット	鍵長 15360 ビット
	離散対数型（DH(E)、DSA）	鍵長 2048 ビット (L, N) = (2048, 224)	鍵長 3072 ビット (L, N) = (3072, 256)	鍵長 7680 ビット (L, N) = (7680, 384)	鍵長 15360 ビット (L, N) = (15360, 512)
	楕円曲線暗号（ECDH(E)、ECDSA、EdDSA）	P-224 B-233 K-233	P-256 B-283 K-283 W-25519 Curve25519 Edwards25519	P-384 B-409 K-409 W-448 Curve448 Edwards448	P-521 B-571 K-571
共通鍵暗号	ブロック暗号	なし	鍵長 128 ビットの AES、Camellia	鍵長 192 ビットの AES、Camellia	鍵長 256 ビットの AES、Camellia
	認証暗号	なし	なし	なし	ChaCha20-Poly1305
ハッシュ関数	HMAC で使う場合	なし	SHA-1	なし	SHA-256 SHA-384 SHA-512

2.7 SSL/TLS の利用環境の変化

2015 年 1 月から 2023 年 11 月の期間に発行された SSL/TLS に関する RFC のうち、「プロトコルバージョン」「サーバ証明書」「暗号スイート（暗号アルゴリズム）」の 3 つの観点から、利用可否

や利用期間などの記述が含まれるものは以下のとおりである。例えば、TLS1.3 の規格化のほか、TLS1.2 までのプロトコルに対して SSL3.0 の無効化や RC4 の無効化など、プロトコルの脆弱性の排除に関するものが規格化されている。

また、2019 年 8 月に NIST は TLS に関する新たなガイドライン SP800-52 Revision 2[19]を発表した。Rev.2 では、①2024 年 1 月 1 日までに連邦政府で利用する全てのサーバ及びクライアント（ブラウザ）で TLS1.3 をサポートすること、②TLS1.2 以上の利用を原則とすること、③RSA での鍵交換を止める、などが規定された。

表 12　2015 年 1 月から 2023 年 11 月の期間に「プロトコルバージョン」「サーバ証明書」「暗号スイート（暗号アルゴリズム）」に関連して発行された RFC

発効年	RFC	Title	プロトコル	サーバ証明書	暗号スイート	内容
2015.2	7465	Prohibiting RC4 Cipher Suites	×	×	○	RC4 禁止
2015.4	7507	TLS Fallback Signaling Cipher Suite Value (SCSV) for Preventing Protocol Downgrade Attacks	×	×	○	新暗号スイートの定義
2015.5	7525	Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)	○	×	×	SSL2.0, SSL3.0 禁止 TLS1.0, TLS1.1 非推奨
2015.6	7568	Deprecating Secure Sockets Layer Version 3.0	○	×	×	SSL3.0 禁止
2016.3	7836	Guidelines on the Cryptographic Algorithms to Accompany the Usage of Standards GOST R 34.10-2012 and GOST R 34.11-2012	×	×	△	GOST 暗号の追加
2016.6	7905	ChaCha20-Poly1305 Cipher Suites for Transport Layer Security (TLS)	×	×	○	ChaCha20-Poly1305 の暗号スイート追加
2016.8	7919	Negotiated Finite Field Diffie-Hellman Ephemeral Parameters for Transport Layer Security (TLS)	×	×	△	DHE で利用するパラメータのネゴシエーションを整理
2018.8	8422	Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS) Versions 1.2 and Earlier	×	×	○	TLS1.2 以前での ECDHE、ECDSA、EdDSA の利用方法を規定

[19] NIST SP 800-52 Rev. 2, Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations

発効年	RFC	Title	プロトコル	サーバ証明書	暗号スイート	内容
2018.9	8442	ECDHE_PSK with AES-GCM and AES-CCM Cipher Suites for TLS 1.2 and DTLS 1.2	×	×	○	TLS1.2 での ECDHE_PSK の暗号スイートの追加 ※本ガイドラインの対象外の暗号スイート
2018.8	8446	The Transport Layer Security (TLS) Protocol Version 1.3	○	×	○	TLS1.3 の規格化
2018.8	8447	IANA Registry Updates for TLS and DTLS	×	×	△	暗号スイートのスイート番号の管理
2020.2	8734	Elliptic Curve Cryptography (ECC) Brainpool Curves for Transport Layer Security (TLS) Version 1.3	×	×	△	TLS1.3 での Brainpool 曲線の追加。ただし、IETF として勧めた方式ではない
2021.3	8996	Deprecating TLS 1.0 and TLS 1.1	○	×	×	TLS1.0, TLS1.1 禁止
2021.3	8998	ShangMi (SM) Cipher Suites for TLS 1.3	×	×	△	ShangMi（SM）暗号スイートの追加
2022.4	9151	Commercial National Security Algorithm (CNSA) Suite Profile for TLS and DTLS 1.2 and 1.3	×	×	○	Commercial National Security Algorithm (CNSA) 1.0 に準拠した Suite Profile を追加
2021.12	9155	Deprecating MD5 and SHA-1 Signature Hashes in TLS 1.2 and DTLS 1.2	×	×	△	デジタル署名での MD5,SHA-1 の利用禁止 ※HMAC での SHA-1 利用を禁止するものではない
2022.3	9189	GOST Cipher Suites for Transport Layer Security (TLS) Protocol Version 1.2	×	×	△	GOST 暗号スイートの追加
2022.12	9325	Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)	○	○	○	TLS の安全な利用上の推奨事項をまとめた Best Current Practice
2023.7	9345	Delegated Credentials for TLS and DTLS	△	○	○	エンドサーバで CA 証明書に紐づく Delegated credential を発行する仕様
2023.2	9367	GOST Cipher Suites for Transport Layer Security (TLS) Protocol Version 1.3	×	×	△	GOST 暗号スイートの追加

凡例：　○：影響あり　　×：影響なし　　△：関連はあるものの直接的な影響はなし

上記のような規格化の流れもあり、SSL Pulse[20]の集計データによれば、図 7 から図 10 のように、2014 年から 2023 年の間にプロトコルバージョンや暗号アルゴリズム等のサポート状況に大きな変動が発生していることがわかる。

例えば、SSL3.0 禁止の RFC が発行された 2015 年前後でサポート状況が全く異なること、TLS1.1 以前のバージョンはいずれもサポート率が低下し、TLS1.2 に移行していること、などが図 7 から読み取れる。RC4 についても同様で、RC4 禁止の RFC が発行された 2015 年前後でサポート状況が全く異なることが図 8 から読み取れる。

鍵交換に関しては、2013 年のスノーデン事件をきっかけに 2.5.3 節に記載したように PFS の重要性が認識され、2014 年にはわずか 5%しかサポートされていなかったものが瞬く間にサポートされるようになったことが図 9 からわかる。また、DH(E)の鍵長については鍵長 2048 ビット以上への移行が進んでいることが図 10 からわかる。なお、図 10 は、サーバがサポートする最も弱い鍵交換の強度を等価強度の RSA 鍵長で表したものであり、「1024 ビット以下」のほとんどは DH(E)のパラメータに由来すると考えられる。

[20] https://www.ssllabs.com/ssl-pulse/

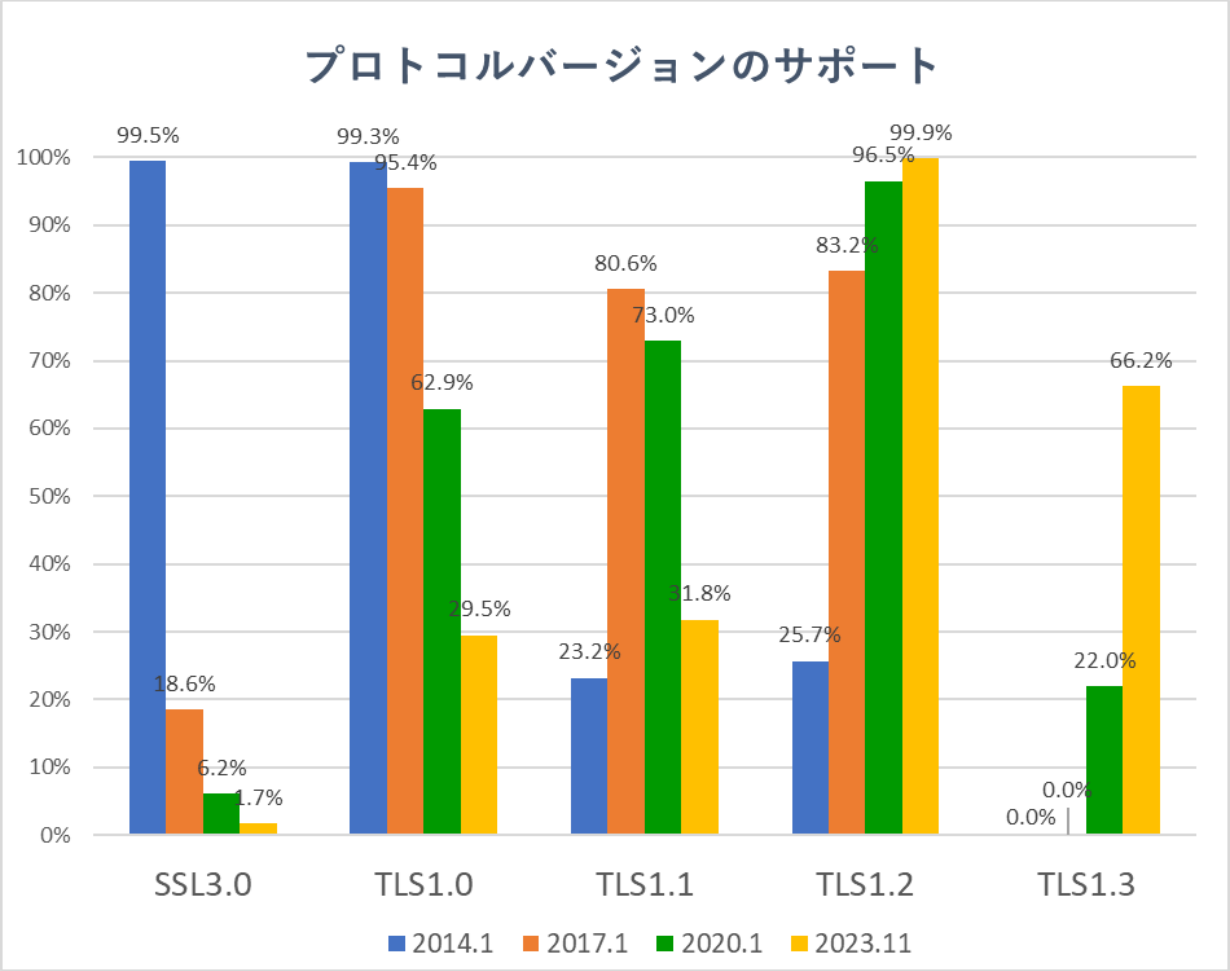


図 7　サポートされるプロトコルバージョンの推移（SSL Pulse による集計データを加工）

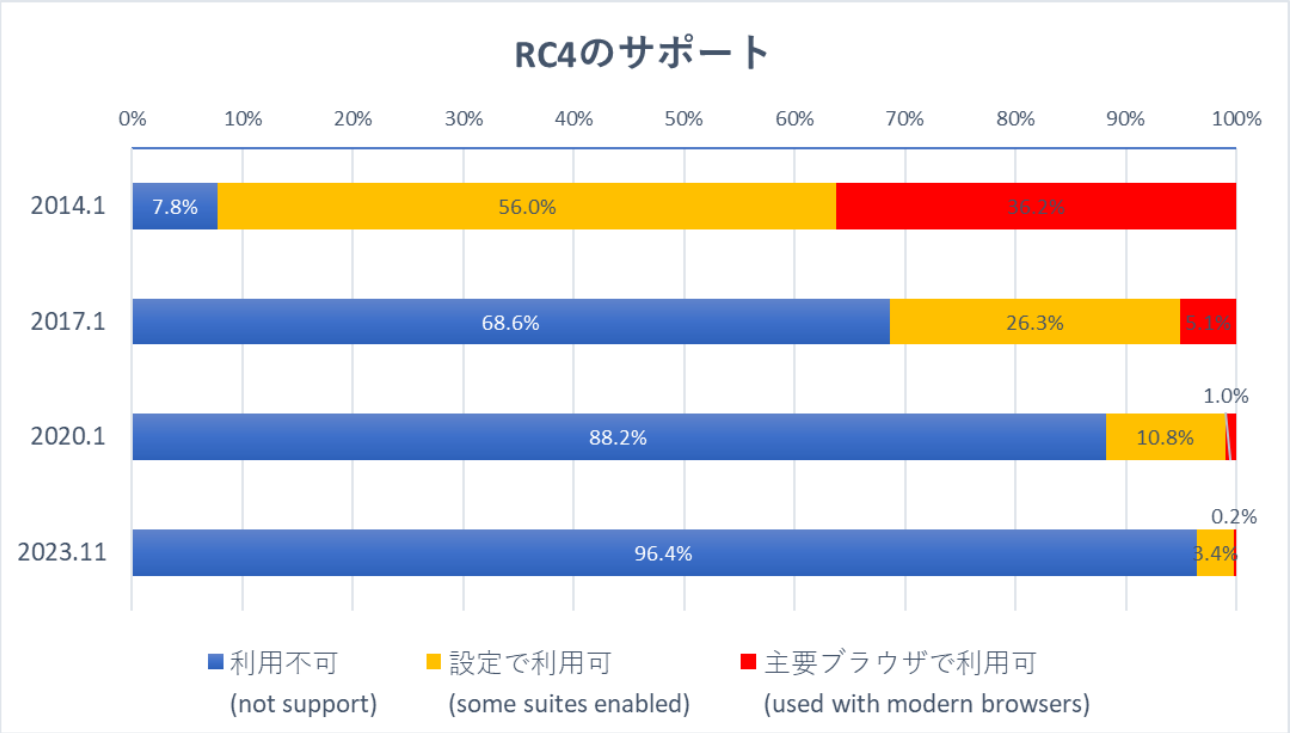


図 8　RC4 のサポート状況（SSL Pulse による集計データを加工）

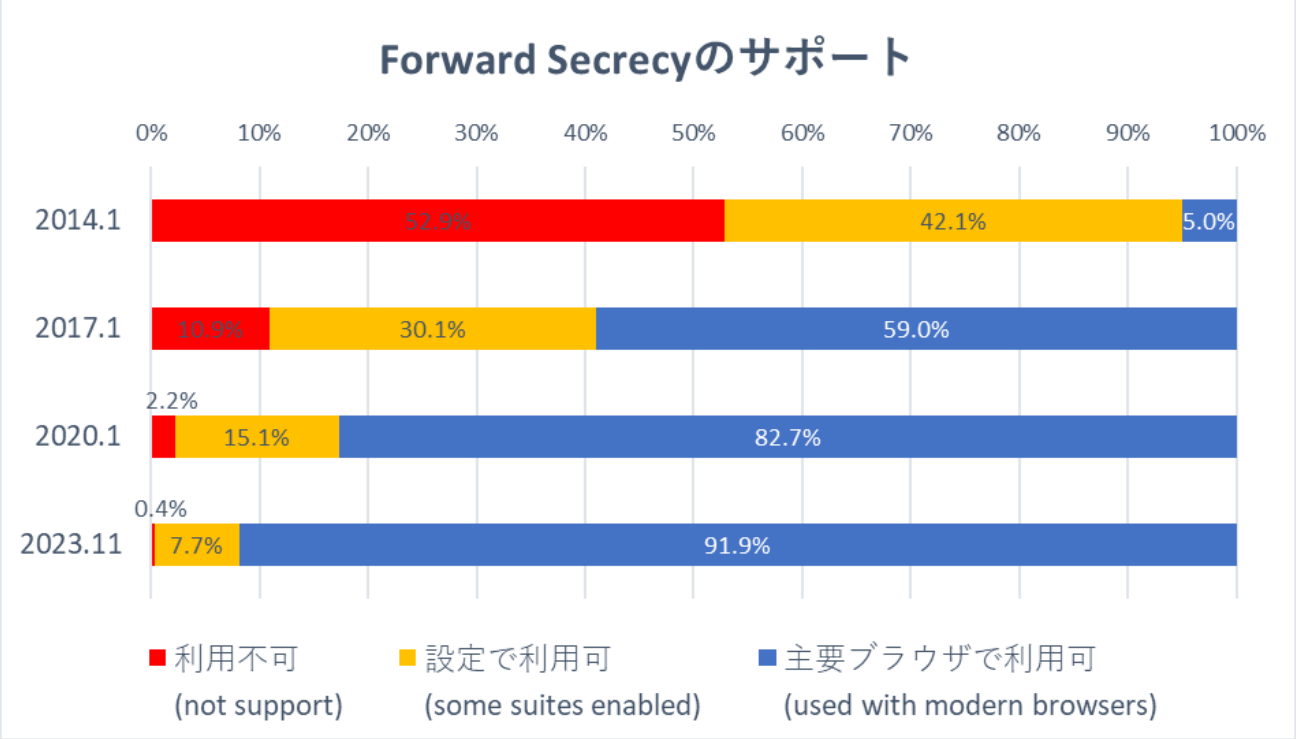


図 9　Perfect Forward Secrecy のサポート状況（SSL Pulse による集計データを加工）

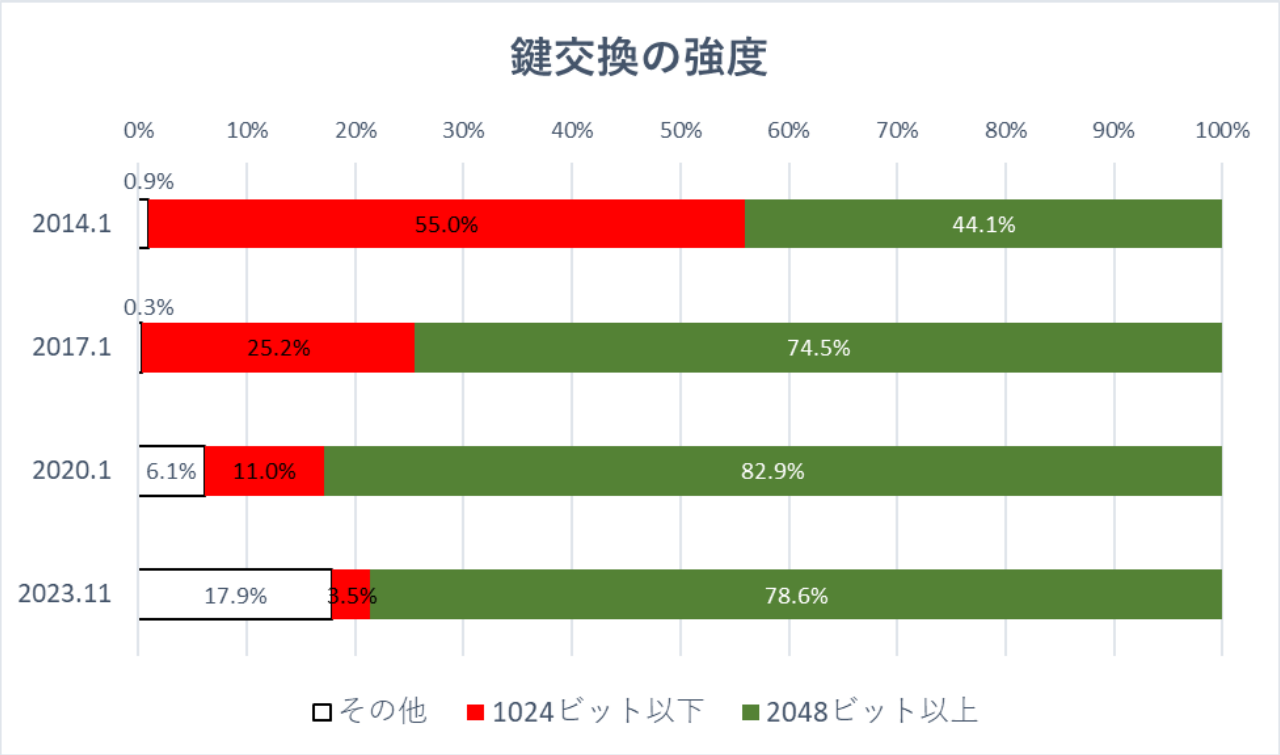


図 10　DH(E)鍵交換の鍵長の推移（SSL Pulse による集計データを加工）

【コラム①】 常時 HTTPS 化に伴う留意点

2015 年頃からウェブの全 HTTPS 化が叫ばれてきた。これは、基本は HTTP、ログイン画面や支払い画面など機密情報を扱うページのみ HTTPS にするという慣習を止め、すべてのウェブページを HTTPS で配信しようというものである。この動きは一部のブラウザベンダのみが提唱してきた話ではなく、すべてのメジャーなブラウザベンダや IETF も巻き込んだインターネット全体の動きに発展してきた。

ウェブサイトの常時 HTTPS 化は手間のかかる作業だが、ユーザの安全性を守るだけでなく、プライバシーに配慮してよりパワフルな機能を実現することができる下地となる。まだ常時 HTTPS 化を行っていないようであれば検討することを勧める。

なぜ今、全 HTTPS 化が叫ばれているのか？

HTTP は通信内容を保証しないため、例えば通信経路の第三者がサービスを改ざんしても、それが本来配信されるべき内容であるかどうかはユーザには分からない。これは大きな問題を引き起こす可能性がある。

例えば、インターネットプロバイダやホテル、ショップ等が提供する Wi-Fi サービス、公共機関等に設置された Wi-Fi サービスなどを利用して通信を行う場合、中継地点に悪意のある第三者が存在する可能性が否定できない。そういった第三者は、コンテンツに広告を挿入したり、マルウェアを埋め込んでユーザを危険にさらしたりするだけでなく、ユーザをトラッキングしたり、盗聴する可能性もある。

通信を HTTPS 化することによって、通信相手が意図したものであることを保証し、通信内容が改ざんされていないことを保証し、そして通信内容を傍受されないことを保証することができるようになる。

また、HTTPS 化することによって、様々なブラウザのパワフルで新しい機能が利用可能になる。例えば Service Worker は、ブラウザのタブを開いていなくてもスクリプトを動かすことを可能にし、ウェブページのオフライン動作やプッシュ通知を可能にする。その他にも生体認証やセキュリティキーを利用可能にする WebAuthn、ブラウザ上の支払いを便利にする Payment Request API なども含まれる。ネットワーク面でも、HTTP の新しいバージョンである HTTP/2 では通信をシリアライズかつ並列化することでパフォーマンスを強化し、さらに次の世代となる HTTP/3 の基礎となる QUIC についても HTTPS が前提となっている。

一方、元々パワフルだった古い機能も HTTP では利用できなくなるような移行が行われた。例えば、位置情報を取得する Geolocation、デバイスの傾きを取得する Device motion / orientation などがこれに該当する。

このように、ウェブ全体を HTTPS 化する動きに伴って、「HTTPS だから安全」と捉えるのではなく、「HTTP を危険」と捉える方向にシフトしつつある。このため、例えば各種ブラウザでは、これまで HTTPS の場合に錠前を表示してきたが、現在は HTTPS ではデフォルト表示を行い、反対に HTTP の場合はユーザに注意を促す表示を行うようになっている。

図 11 は Google の公開している Transparency Report から引用したものである。日本の HTTPS 対応は他の国々に大きく遅れを取っていたが、2016 年時点では飛び抜けて低かった HTTPS 対応状況が 2017 年を境にここ数年で目覚ましい挽回を見せていることが分かる。

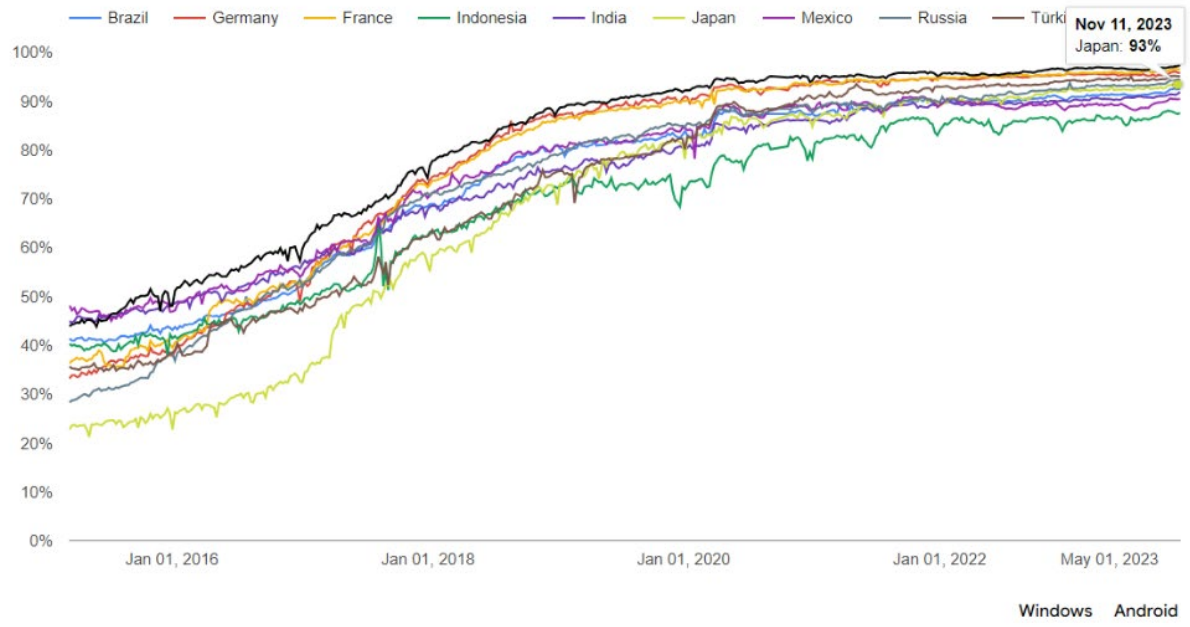


図 11　HTTPS 対応状況（[出典] Google Transparency Report）

常時 HTTPS 化に伴う留意事項

サーバから配信するすべてのリソースを HTTPS 化するには、以下の留意点がある。

- Mixed-Contents

  ブラウザは、表示しているページに暗号化されていない（HTTP の）リソースが含まれている場合、これを Mixed-Contents、つまり「暗号化されているものとされていないものが混在しているコンテンツ」であり、「安全ではない」ものとして扱う。その結果、ブラウザによっては URL バーに「保護されていない通信」などと表示されることがある。さらに、ブラウザによっては、HTTP 通信の部分のロードが最終的にすべてブロックされるようになる予定がある。

  Mixed-Contents を避けるためには、ウェブサイトは、ページに掲載するすべてのコンテンツやリソースについて HTTPS 化されている状態にしなければならない。そのため、自分自身ですべてのコンテンツやリソースを管理していれば対応しやすいが、問題は広告や埋め込みコンテンツなどが自分以外のオーナーによってリソースが管理されている場合である。全 HTTPS 化が叫ばれて久しいので、広告や埋め込みコンテンツの多くはすでに HTTPS 化されているが、もしそうではない場合は、提供元に働きかける必要がある。

  https://developers.google.com/web/fundamentals/security/prevent-mixed-content/what-is-mixed-content

また、すべてのコンテンツやリソースが HTTPS で配信されているかどうかをマニュアルで確認するのは簡単なことではない。そのようなときには、ブラウザの提供する Reporting API を使えば、ブラウザが Mixed Contents を発見次第、レポートを送るという機能を利用することができる。

https://developers.google.com/web/fundamentals/security/prevent-mixed-content/fixing-mixed-content

- ストレージの移行

元々HTTP だったウェブサイトを HTTPS に変更する際、ブラウザ上のストレージにも注意が必要である。ブラウザは同じドメインでもスキーマが異なれば別オリジンとして扱うため、HTTP と HTTPS の間ではストレージが共有されない。そのため、移行に際しては Local Storage や Indexed DB などのストレージを移し替える必要が出てくる。

- Cookie に Secure オプションを追加

Cookie には"Secure"属性という HTTPS 通信に限って送信する機能がある。セッション ID などの機微な情報が誤って HTTP 上に流れてしまうと、セッションを盗まれてしまう危険性もあるので、必ず“Secure”属性を付けるようにすべきである。

- HSTS で強制的にデフォルト HTTPS 化

全 HTTPS 化を進めるにしても、ウェブサイトに残された既存のリンクをすべて残らず変更するのは容易ではない。そうして残った HTTP 経由のリンクでアクセスした場合でも、HTTPS にリダイレクトすれば問題ないと思えるかもしれないが、その（リダイレクトされる前の）アクセスでセッション情報など、重要な情報が平文で送られてしまうのは好ましくない。

その対策として、HTTP Strict Transport Security（HSTS）ヘッダーを送ることで、ブラウザに強制的に HTTPS でアクセスさせることを記憶させることができる。これを利用することで、それ以降一定期間、同じブラウザからのアクセスは、リンクが HTTP で記述されていても、自動的に HTTPS に変換した上でリクエストが送信される。HSTS については、7.4.1 節も参照されたい。

なお、HTTP のリンクが残っている限り、初めてそのウェブサイトに訪れる際には HTTP でアクセスせざるを得ないケースは存在する。このため、既存のリンクについても、可能な限り HTTPS 化することを怠らないようにすべきである。

https://developers.google.com/web/fundamentals/security/encrypt-in-transit/enable-https

PART I：

サーバ構築における設定要求について

3. 設定基準の概要

本章では、TLS サーバの構築時に、主に暗号通信に関わる設定に関しての要求事項を決めるために考慮すべきポイントについて取りまとめる。

3.1 実現すべき設定基準の考え方

SSL/TLS は、1994 年に SSL2.0 が実装されて以来、その利便性から多くの製品に実装され、利用されている。一方、プロトコルの脆弱性に対応するため、何度かプロトコルとしてのバージョンアップが行われている影響で、製品の違いによってサポートしているプロトコルバージョンや暗号スイート等が異なり、相互接続性上の問題が生じる可能性がある。

このため、SSL/TLS における暗号通信に関わる設定には以下のものがある。

- プロトコルバージョンの設定
- サーバ証明書の設定
- 暗号スイートの設定
- TLS を安全に使うために考慮すべきこと

本ガイドラインでは、安全性の確保と相互接続の必要性のトレードオフにより、**「高セキュリティ型」「推奨セキュリティ型」「セキュリティ例外型」**の 3 段階の設定基準を設ける。それぞれの設定基準における、安全性と相互接続性についての関係は表 13 のとおりである。

また、4 章から 6 章にかけて、それぞれの設定基準に対応する形で、プロトコルバージョン、サーバ証明書、暗号スイートの 3 つの項目についての詳細な要求設定を定める。表 14 に要求設定の概要を記す。

実際にどの設定基準を採用するかは、安全性の確保と相互接続の必要性の両面を鑑みて、サーバ管理やサービス提供に責任を持つ管理者が最終的に決定すべきことではあるが、**本ガイドラインでは、安全性もしくは相互接続性についての特段の要求がなければ「推奨セキュリティ型」の採用を強く勧める**。本ガイドラインの作成時点（2024 年 3 月）では、「推奨セキュリティ型」がもっとも安全性と相互接続性のバランスが取れている要求設定であると考えている。

「セキュリティ例外型」は、システム等の制約上、安全性が低いプロトコルバージョンである TLS1.0、TLS1.1 の利用を全面禁止することが現実的ではなく、安全性上のリスクを受容してでも TLS1.0、TLS1.1 を継続利用せざるを得ないと判断される場合にのみ採用すべきである。なお、セキュリティ例外型であっても、TLS1.0、TLS1.1 の無期限の継続利用を認めているわけではない。**本ガイドラインで記載されているセキュリティ例外型の設定内容は、2029 年度を目途とした改訂時に終了させる予定である。速やかに推奨セキュリティ型への移行を完了させるべきである。**

表 13　安全性と相互接続性についての比較

設定基準	概要	安全性	相互接続性の確保
高セキュリティ型 ＜5 章＞	情報が漏えいした際、組織の運営や資産、個人の資産やプライバシー等に悪影響を及ぼすと予想される情報を、安全性を優先して TLS 通信を行うような場合に採用する設定基準 **※ 高い安全性の確保を必要とするケースでの利用を想定している** ＜利用例＞ • セキュリティが重要視されるシステムを構築する場合	本ガイドライン作成時点（2024 年 3 月）において、標準的な水準を上回る高い安全性水準を達成している	本ガイドラインで対象とするブラウザが搭載されている PC、スマートフォン等であれば相互接続性を確保できると期待される。
推奨セキュリティ型 ＜4 章＞	情報が漏えいした際、組織の運営や資産、個人の資産やプライバシー等に悪影響を及ぼすと予想される情報を、安全性確保と利便性（相互接続性）の実現をバランスさせて TLS 通信を行うための標準的な設定基準 **※ほぼすべての一般的な利用形態で使うことを想定している** ＜利用例＞ • 電子申請など、企業・国民と役所等との電子行政サービスを提供する場合 • 金融サービスや電子商取引サービス、多様な個人情報の入力を必須とするサービス等を広範囲（不特定多数）に提供する場合 • 新規に社内システムを構築する場合	本ガイドライン作成時点（2024 年 3 月）における標準的な安全性水準を実現している	本ガイドラインで対象とするブラウザが搭載されている PC、スマートフォンを含め、多くの製品・システムで相互接続性を確保できると期待される。 ※サポートが終了しているバージョンが古い OS やブラウザ、発売開始から長期間経過している古い機器、IoT 用途などの一部の機器類については接続できない可能性がある。
セキュリティ例外型 ＜6 章＞	脆弱なプロトコルバージョンや暗号が使われるリスクを受容したうえで、**安全性よりも相互接続性に対する要求をやむなく優先**させて TLS 通信を行う場合に採用する設定基準 **※本ガイドラインで記載されているセキュリティ例外型の設定内容は、2029 年度を目途とした改訂時に終了させる予定である。速やかに推奨セキュリティ型への移行を完了させるべきである。**	本ガイドライン作成時点（2024 年 3 月）において、標準的な安全性水準を満たしていないプロトコルバージョンや暗号スイート等が使用される可能性があることを認識する必要がある	既存システムを含め、ほぼすべての機器に対して相互接続性が確保されると期待される。

	＜利用例＞ • 利用するサーバやクライアントの実装上の制約、又は既存システムとの相互接続上の制約により、推奨セキュリティ型（以上）の設定が事実上できない場合		

表 14　要求設定の概要

要件		高セキュリティ型	推奨セキュリティ型	セキュリティ例外型
【遵守】想定対象		高い安全性の確保を必要とするケース	一般的な利用形態	推奨セキュリティ型への移行完了までの暫定運用
【遵守】暗号アルゴリズム		高セキュリティ型での利用禁止暗号を利用不可	推奨セキュリティ型での利用禁止暗号を利用不可	セキュリティ例外型での利用禁止暗号を利用不可
【遵守】バージョン		TLS1.3 (必須) 及び TLS1.2 (オプション)	TLS1.2 (必須) 及び TLS1.3 (オプション)	TLS1.3 ～ TLS1.0 のいずれか
【推奨】推奨暗号アルゴリズム設定	鍵交換	①128 ビットセキュリティ以上を満たす曲線の ECDHE ②128 ビットセキュリティ以上を満たす鍵長の DHE	①128 ビットセキュリティ以上を満たす曲線の ECDHE ②112 ビットセキュリティ以上を満たす鍵長の DHE	① 1024 ビット以上の鍵長の DHE 又は DH ② 112 ビットセキュリティ以上を満たす鍵長の RSA ③ 128 ビットセキュリティ以上を満たす曲線の ECDHE 又は ECDH
	暗号化	128 ビットセキュリティ及び 256 ビットセキュリティを満たす鍵長の AES 又は Camellia、もしくは ChaCha20-Poly1305		128 ビットセキュリティ及び 256 ビットセキュリティを満たす鍵長の AES 又は Camellia、もしくは ChaCha20-Poly1305
	暗号利用モード	GCM, CCM, CCM_8 のいずれか	①GCM, CCM, CCM_8 のいずれか ②CBC	①GCM, CCM, CCM_8 のいずれか ②CBC
	ハッシュ関数	SHA-384 又は SHA-256	①SHA-384 又は SHA-256 ②SHA-1	SHA-384, SHA-256, SHA-1 のいずれか
【遵守】サーバ証明書	公開鍵情報	112 ビットセキュリティ*以上を満たす鍵長の RSA、もしくは 128 ビットセキュリティ以上を満たす曲線の楕円曲線暗号		112 ビットセキュリティ以上を満たす鍵長の RSA
	署名アルゴリズム	112 ビットセキュリティ*以上を満たす鍵長の RSA、もしくは 128 ビットセキュリティ以上を満たす曲線の ECDSA		112 ビットセキュリティ以上を満たす鍵長の RSA
	ハッシュ関数	SHA-256 以上	SHA-256	SHA-256

* 高セキュリティ型の場合には、可能であれば、128 ビットセキュリティ以上を満たす鍵長（3072 ビット以上）とすること

3.2 要求設定における遵守項目と推奨項目

本ガイドラインでは、それぞれの設定基準における要求設定として「遵守項目」と「推奨項目」が決められている。

「遵守項目」とは、選択した設定基準としての最低限の安全性を確保するために必ず満たさなければならない項目のことであり、「・・・しなければならない」と記載されている。この項目を一つでも満たさないものがあれば当該設定基準を達成したとは見なさない。

「推奨項目」とは、当該設定基準としてよりよい安全性を実現するために満たすことが望ましい項目のことであり、「・・・すべきである」と記載されている。この項目は、できるだけ設定するように推奨するものであるが、強制するものではない。状況に応じて判断されたい。

表 15　要求設定における遵守項目と推奨項目

要求設定	遵守項目	プロトコルバージョン	利用禁止プロトコルバージョンを利用不可にする設定
		サーバ証明書	利用する暗号アルゴリズムと鍵長の設定
			発行・更新時の鍵情報の生成方法の明確化
			警告表示の回避方法の明確化
		暗号スイート	利用禁止暗号アルゴリズムを利用不可にする設定
			公開鍵暗号の鍵長の設定
	推奨項目	プロトコルバージョン	利用プロトコルバージョンの優先順位付け
		暗号スイート	利用推奨暗号アルゴリズムのみでの設定
			推奨暗号スイートの優先順位付け

3.3 チェックリスト

図 12 に高セキュリティ型のチェックリストのイメージを示す。

チェックリストの目的は、

- 選択した設定基準に対応した要求設定を実施したことを確認し、設定忘れを防止すること
- サーバ構築の作業受託先が適切に要求設定を遵守したことを、発注者が文書として確認する手段を与えること

である。選択した設定基準に応じたチェックリストを用い、すべてのチェック項目について、以下の方針でチェックの確認を行う。Appendix A には、各々の設定基準に対応したチェックリストを載せる。

- 「要求設定確認」は選択した設定基準を採用してよいかの確認項目であり、「該当」にチェッ

クが入る場合に限り、当該設定基準を採用してよい

- 「遵守項目」については該当章に記載の要求設定に合致していることを確認して「済」にチェックが入ることが必要である
- 「遵守項目」以外については該当章の記載内容と照らし合わせ、設定の実態に即して適切なほうのチェックボックスにチェックを入れる。

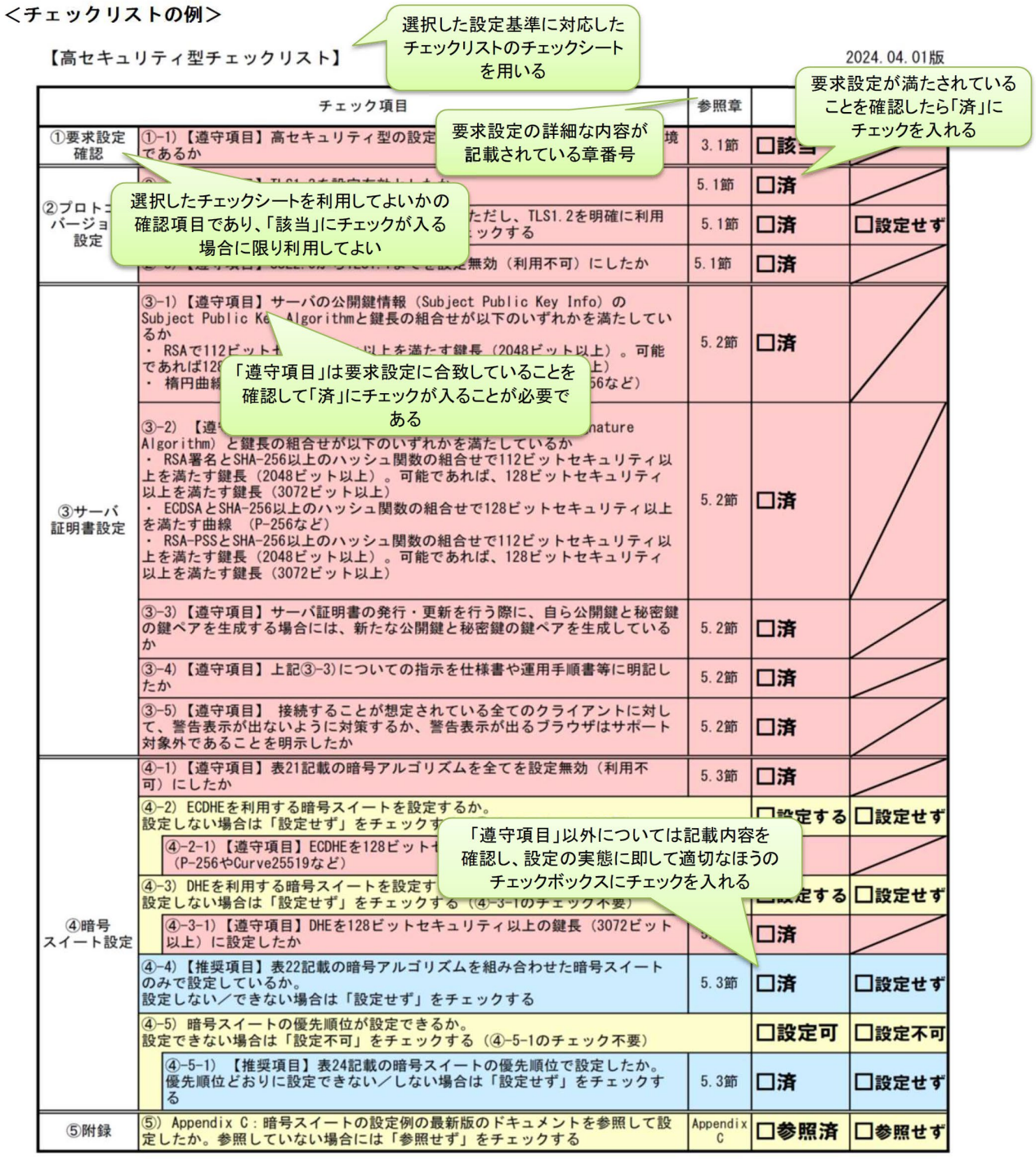


図 12　チェックリスト（高セキュリティ型の例）

4. 推奨セキュリティ型の要求設定

4.1 プロトコルバージョン

SSL2.0 から TLS1.3 までの安全性の違い（2.2 節参照）を踏まえ、TLS サーバがサポートするプロトコルバージョンについての「遵守項目」を以下のように定める。

【プロトコルバージョンの遵守項目】

- TLS1.2 を設定有効としなければならない。
- SSL2.0 から TLS1.1 までを設定無効（利用不可）にしなければならない。

【プロトコルバージョンの推奨項目】

- TLS1.3 については、実装されているのであれば、設定有効にすべきである。ただし、TLS1.3 が実装されている場合であっても、TLS1.3 を明確に利用しないと判明している場合には TLS1.3 の設定有効化を必須とするものではない。

表 16　推奨セキュリティ型でのプロトコルバージョン設定

	TLS1.3*	TLS1.2	TLS1.1	TLS1.0	SSL3.0	SSL2.0
2 つのうちのいずれか	○	○	×	×	×	×
	－	○	×	×	×	×

○：設定有効　　×：設定無効化　　－：実装なし

* TLS1.3 を明確に利用しないと判明している場合には、TLS1.3 設定有効化を必須としない

4.2 サーバ証明書

サーバ証明書についての「遵守項目」を以下のように定める。

現在発行されているサーバ証明書は、大多数が RSA と SHA-256 との組合せによるものである。

また、RSA では 112 ビットセキュリティの鍵長（2048 ビット）が多いのに対し、処理性能の低下を避けるために 128 ビットセキュリティを満たす曲線（P-256 など）を利用した ECDSA を採用するケースも増えてきている。実際に、従来 RSA しかサーバ証明書を発行しなかった認証局でも、ECDSA に対応したサーバ証明書を発行するようになってきている。

【サーバ証明書の遵守項目】

- 本ガイドライン作成時点（2024 年 3 月）で、多くのパブリック認証局から入手可能なサーバ証明書のうち、安全性が高いものを利用しなければならない。

表 17　推奨セキュリティ型でのサーバ証明書設定

サーバ証明書のアルゴリズムと鍵長	サーバ証明書の発行・更新を要求する際に生成する公開鍵情報（Subject Public Key Info）でのアルゴリズム（Subject Public Key Algorithm）と鍵長としては、以下のいずれかを必須とする。 ● RSA（OID = 1.2.840.113549.1.1.1）で 112 ビットセキュリティ以上を満たす鍵長（2048 ビット以上） ● 楕円曲線暗号（ecPublicKey; OID = 1.2.840.10045.2.1）で 128 ビットセキュリティ以上を満たす曲線（例：P-256 の場合、OID = 1.2.840.10045.3.1.7） また、認証局が発行するサーバ証明書での署名アルゴリズム（Certificate Signature Algorithm）と鍵長については、以下のいずれかを必須とする。 ● RSA 署名と SHA-256 の組合せ（sha256WithRSAEncryption; OID = 1.2.840.113549.1.1.11）で 112 ビットセキュリティ以上を満たす鍵長（2048 ビット以上） ● ECDSA と SHA-256 の組合せ（ecdsa-with-SHA256; OID = 1.2.840.10045.4.3.2）で 128 ビットセキュリティ以上を満たす曲線（例：P-256 以上） ● RSA-PSS（id-RSASSA-PSS; OID=1.2.840.113549.1.1.10）と SHA-256 の組合せで 112 ビットセキュリティ以上を満たす鍵長（2048 ビット以上）
サーバ証明書の発行・更新時の鍵情報の生成	● サーバ証明書の発行・更新を行う際に、自ら公開鍵と秘密鍵の鍵ペアを生成する場合には、既存の公開鍵と秘密鍵の鍵ペアを再利用せず、新たな公開鍵と秘密鍵の鍵ペアを生成しなければならない。 ● 上記の指示をサーバ管理者への仕様書、運用手順書、ガイドライン等に明示しなければならない。
クライアントでの警告表示の回避	● 当該サーバに接続することが想定されている全てのブラウザ（クライアント）に対して、以下のいずれかの手段を用いて警告表示が出ないようにしなければならない。 ➢ パブリック認証局からサーバ証明書を入手する ➢ 警告表示が出るブラウザ（クライアント）はサポート対象外であることを明示する、または警告表示が出ないサポート対象のブラウザ（クライアント）を明示する ➢ 7.2.9 節に従って、信頼できるプライベート認証局のルート CA 証明書を予めインストールする

4.3 暗号スイート

暗号スイートについての「遵守項目」及び「推奨項目」を以下のように定める。
なお、鍵交換に PSK または KRB が含まれる暗号スイートは、サーバとクライアントの両方で

特別な設定をしなければ利用することができないため、本ガイドラインの対象外とする。

【暗号スイートの遵守項目】

- 以下の条件に該当する暗号アルゴリズムのいずれかを含む暗号スイートが選択されないようにするため、表 18 に記載される暗号アルゴリズム全てを設定無効（利用不可）としなければならない。
  - 2.5.2 節の表 9 に掲載されている暗号アルゴリズム
  - 2.5.2 節の表 8 に掲載されている暗号アルゴリズムのうち、DH 及び ECDH

表 18　推奨セキュリティ型での利用禁止暗号アルゴリズム一覧

鍵交換		DH, ECDH
署名		GOST R 34.10-2012, SM2（署名）
暗号化	ブロック暗号	RC2, EXPORT-RC2, IDEA, DES, EXPORT-DES, GOST 28147-89, Magma, 3-key Triple DES, Kuznyechik, ARIA, SEED, SM4
	暗号利用モード	CTR_OMAC
	ストリーム暗号	RC4, EXPORT-RC4
ハッシュ関数		MD5, GOST R 34.11-2012, SM3

- 鍵交換で ECDHE を利用する場合には 128 ビットセキュリティ以上を満たす曲線（P-256 や、Curve25519 を用いた X25519 など）[21]を、DHE を利用する場合には 112 ビットセキュリティ以上を満たす鍵長（2048 ビット以上）での設定をしなければならない。なお、DHE の鍵長を明示的に適切に設定できない製品を利用する場合には、DHE を含む暗号スイートは選定してはならない。

【暗号スイートの推奨項目】

- 以下の条件に該当する表 19 に記載される暗号アルゴリズムを組み合わせた暗号スイートのみが選択されるようにすべきである。
  - 2.5.2 節の表 7 に掲載されている暗号アルゴリズム
  - 2.5.2 節の表 8 に掲載されている暗号アルゴリズムのうち、CBC 及び SHA-1

[21] 2.6.2 節 表 11 に記載されている楕円曲線から選択すること。

表 19　推奨セキュリティ型での利用推奨暗号アルゴリズム一覧

鍵交換		ECDHE, DHE
署名		ECDSA, RSASSA-PKCS1-v1_5, RSASSA-PSS（TLS1.3 のみ）
暗号化	ブロック暗号	AES, Camellia（TLS1.2 のみ）
	暗号利用モード	GCM, CCM, CCM_8, CBC
	認証暗号	ChaCha20-Poly1305
ハッシュ関数		SHA-256, SHA-384, SHA-1

- 安全性が高い暗号スイートから優先的に接続するようにするため、表 20 のようにグループ A、グループ B、・・・の順に暗号スイートの優先順位を設定すべきである。

表 20　推奨セキュリティ型での暗号スイートの優先順位

	【第一優先】 暗号利用モードの種類	【第二優先】 鍵交換の種類	【第三優先】 ハッシュ関数
グループ A	GCM, CCM, CCM_8 認証暗号	ECDHE	SHA-256, SHA-384
グループ B	GCM, CCM, CCM_8 認証暗号	DHE	SHA-256, SHA-384
グループ C	CBC	ECDHE	SHA-256, SHA-384
グループ D	CBC	ECDHE	SHA-1*)
グループ E	CBC	DHE	SHA-256, SHA-384
グループ F	CBC	DHE	SHA-1*)

*) 暗号スイートとしての SHA-1 は HMAC の構成部品として利用されており、2024 年 3 月時点においても HMAC-SHA1 の安全性について問題はない。しかしながら、実運用上は、HMAC の構成部品であっても SHA-1 の利用を避けるなどの世の中の動きもあり、今後の動向次第では推奨から外す可能性があることに留意されたい。

上記の方針に従った暗号スイートの推奨設定を表 21 に示す。表 21 は、表 19 に記載される暗号アルゴリズムのみで組み合わせた暗号スイートの全てが網羅されており、さらに表 20 に従ってグループ A、グループ B、グループ C の 3 つにグループ分けされている。

優先順位の設定ができない場合は表 21 全体の暗号スイートから、優先順位の設定ができる場合には各グループ内での暗号スイートから全部または一部を選択して設定する。

優先順位を設定する際には、グループ A に含まれる暗号スイート、グループ B に含まれる暗号スイート、グループ C に含まれる暗号スイートの順になるように調整する。この際、各グループ内での暗号スイートの優先順位は任意に定めてよい。また、グループ B 以降の暗号スイートについては選択しなくてもよい。

表 21　推奨セキュリティ型での暗号スイートの推奨設定

	TLS1.2 を利用する場合
グループ A	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_ECDHE_RSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8
	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_ECDHE_RSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM_8
	TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256
	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256
グループ B	TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
	TLS_DHE_RSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_DHE_RSA_WITH_AES_128_CCM
	TLS_DHE_RSA_WITH_AES_128_CCM_8
	TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
	TLS_DHE_RSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_DHE_RSA_WITH_AES_256_CCM
	TLS_DHE_RSA_WITH_AES_256_CCM_8
	TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256
グループ C	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_CBC_SHA256
	TLS_ECDHE_RSA_WITH_CAMELLIA_128_CBC_SHA256
	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_CBC_SHA384
	TLS_ECDHE_RSA_WITH_CAMELLIA_256_CBC_SHA384

グループ D	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
グループ E	TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
	TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA256
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA256
	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA256
グループ F	TLS_DHE_RSA_WITH_AES_128_CBC_SHA
	TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA
	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA

	TLS1.3 を利用する場合
グループ A	TLS_AES_128_GCM_SHA256
	TLS_AES_128_CCM_SHA256
	TLS_AES_128_CCM_8_SHA256
	TLS_AES_256_GCM_SHA384
	TLS_CHACHA20_POLY1305_SHA256
鍵交換	ECDHE（優先）
	DHE
署名	ECDSA
	RSA-PSS
	RSASSA-PKCS1-v1_5

5. 高セキュリティ型の要求設定

5.1 プロトコルバージョン

SSL2.0 から TLS1.3 までの安全性の違い（2.2 節参照）を踏まえ、TLS サーバがサポートするプロトコルバージョンについての「遵守項目」を以下のように定める。なお、TLS1.3 のみを利用する設定にする場合には、サーバとクライアントの両方が TLS1.3 をサポートしていることが必須となることに注意されたい。

【プロトコルバージョンの遵守項目】

- SSL2.0 から TLS1.1 までを設定無効（利用不可）にしなければならない。
- TLS1.3 及び TLS1.2 を設定有効にしなければならない。ただし、TLS1.2 を明確に利用しないと判明している場合には TLS1.2 の設定有効化を必須とするものではない。

表 22　高セキュリティ型でのプロトコルバージョン設定

TLS1.3	TLS1.2*	TLS1.1	TLS1.0	SSL3.0	SSL2.0
○	○	×	×	×	×

○：設定有効　　×：設定無効化　　－：実装なし

* TLS1.2 を明確に利用しないと判明している場合には、TLS1.2 設定有効化を必須としない

5.2 サーバ証明書

サーバ証明書についての「遵守項目」を以下のように定める。

現在発行されているサーバ証明書は、大多数が RSA と SHA-256 との組合せによるものである。

また、RSA では 112 ビットセキュリティの鍵長（2048 ビット）が多いのに対し、処理性能の低下を避けるために 128 ビットセキュリティを満たす曲線（P-256 など）を利用した ECDSA を採用するケースも増えてきている。実際に、従来 RSA しかサーバ証明書を発行しなかった認証局でも、ECDSA に対応したサーバ証明書を発行するようになってきている。

【サーバ証明書の遵守項目】

- 本ガイドライン作成時点（2024 年 3 月）で、パブリック認証局から入手可能なサーバ証明書のうち、非常に安全性が高いものを利用しなければならない。

表 23　高セキュリティ型でのサーバ証明書設定

サーバ証明書のアルゴリズムと鍵長	サーバ証明書の発行・更新を要求する際に生成する公開鍵情報（Subject Public Key Info）でのアルゴリズム（Subject Public Key Algorithm）と鍵長としては、以下のいずれかを必須とする。 ● RSA（OID = 1.2.840.113549.1.1.1）で 112 ビットセキュリティ以上を満たす鍵長（2048 ビット以上）。なお、可能であれば、128 ビットセキュリティ以上を満たす鍵長（3072 ビット以上）とすること ● 楕円曲線暗号（ecPublicKey; OID = 1.2.840.10045.2.1）で 128 ビットセキュリティ以上を満たす曲線（例：P-256 の場合、OID = 1.2.840.10045.3.1.7） また、認証局が発行するサーバ証明書での署名アルゴリズム（Certificate Signature Algorithm）と鍵長については、以下のいずれかを必須とする。 ● RSA 署名と SHA-256 以上のハッシュ関数の組合せ（例：SHA-256 の場合、sha256WithRSAEncryption; OID = 1.2.840.113549.1.1.11）で 112 ビットセキュリティ以上を満たす鍵長（2048 ビット以上）。なお、可能であれば、128 ビットセキュリティ以上を満たす鍵長（3072 ビット以上）とすること ● ECDSA と SHA-256 以上のハッシュ関数の組合せ（例：SHA-256 の場合、ecdsa-with-SHA256; OID = 1.2.840.10045.4.3.2）で 128 ビットセキュリティ以上を満たす曲線（例：P-256 以上） ● RSA-PSS（id-RSASSA-PSS; OID=1.2.840.113549.1.1.10）と SHA-256 以上のハッシュ関数の組合せで 112 ビットセキュリティ以上を満たす鍵長（2048 ビット以上）。なお、可能であれば、128 ビットセキュリティ以上を満たす鍵長（3072 ビット以上）とすること
サーバ証明書の発行・更新時の鍵情報の生成	● サーバ証明書の発行・更新を行う際に、自ら公開鍵と秘密鍵の鍵ペアを生成する場合には、既存の公開鍵と秘密鍵の鍵ペアを再利用せず、新たな公開鍵と秘密鍵の鍵ペアを生成しなければならない。 ● 上記の指示をサーバ管理者への仕様書、運用手順書、ガイドライン等に明示しなければならない。
クライアントでの警告表示の回避	● 当該サーバに接続することが想定されている全てのブラウザ（クライアント）に対して、以下のいずれかの手段を用いて警告表示が出ないようにしなければならない。 ➢ パブリック認証局からサーバ証明書を入手する ➢ 警告表示が出るブラウザ（クライアント）はサポート対象外であることを明示する、または警告表示が出ないサポート対象のブラウザ（クライアント）を明示する ➢ 7.2.9 節に従って、信頼できるプライベート認証局のルート CA 証明書を予めインストールする

5.3 暗号スイート

暗号スイートについての「遵守項目」及び「推奨項目」を以下のように定める。

なお、鍵交換に PSK または KRB が含まれる暗号スイートは、サーバとクライアントの両方で特別な設定をしなければ利用することができないため、本ガイドラインの対象外とする。

【暗号スイートの遵守項目】

- 以下の条件に該当する暗号アルゴリズムのいずれかを含む暗号スイートが選択されないようにするため、表 24 に記載される暗号アルゴリズム全てを設定無効（利用不可）としなければならない。
  - 2.5.2 節の表 8 に掲載されている暗号アルゴリズム
  - 2.5.2 節の表 9 に掲載されている暗号アルゴリズム

表 24　高セキュリティ型での利用禁止暗号アルゴリズム一覧

鍵交換		DH, ECDH, RSAES-PKCS1-v1_5
署名		GOST R 34.10-2012, SM2（署名）
暗号化	ブロック暗号	RC2, EXPORT-RC2, IDEA, DES, EXPORT-DES, GOST 28147-89, Magma, 3-key Triple DES, Kuznyechik, ARIA, SEED, SM4
	暗号利用モード	CBC, CTR_OMAC
	ストリーム暗号	RC4, EXPORT-RC4
ハッシュ関数		MD5, SHA-1, GOST R 34.11-2012, SM3

- 鍵交換で ECDHE を利用する場合には 128 ビットセキュリティ以上を満たす曲線（P-256 や Curve25519 を用いた X25519 など）[22]を、DHE を利用する場合には 128 ビットセキュリティ以上を満たす鍵長（3072 ビット以上）での設定をしなければならない。なお、DHE の鍵長を明示的に適切に設定できない製品を利用する場合には、DHE を含む暗号スイートは選定してはならない。

【暗号スイートの推奨項目】

- 以下の条件に該当する表 25 に記載される暗号アルゴリズムを組み合わせた暗号スイートのみが選択されるようにすべきである。
  - 2.5.2 節の表 7 に掲載されている暗号アルゴリズム

表 25　高セキュリティ型での利用推奨暗号アルゴリズム一覧

鍵交換	ECDHE, DHE
署名	ECDSA, RSASSA-PKCS1-v1_5, RSASSA-PSS（TLS1.3 のみ）

[22] 2.6.2 節 表 11 に記載されている楕円曲線から選択すること。

暗号化	ブロック暗号	AES, Camellia（TLS1.2 のみ）
	暗号利用モード	GCM, CCM, CCM_8
	認証暗号	ChaCha20-Poly1305
ハッシュ関数		SHA-256, SHA-384

- 安全性が高い暗号スイートから優先的に接続するようにするため、表 26 のようにグループ A、グループ B の順に暗号スイートの優先順位を設定すべきである。

表 26　高セキュリティ型での暗号スイートの優先順位

	【第一優先】 鍵交換の方式
グループ A	ECDHE
グループ B	DHE

上記の方針に従った暗号スイートの推奨設定を表 27 に示す。表 27 は、表 25 に記載される暗号アルゴリズムのみで組み合わせた暗号スイートの全てが網羅されており、さらに表 26 に従ってグループ A、グループ B の 2 つにグループ分けされている。

優先順位の設定ができない場合は表 27 全体の暗号スイートから、優先順位の設定ができる場合には各グループ内での暗号スイートから全部または一部を選択して設定する。

優先順位を設定する際には、グループ A に含まれる暗号スイート、グループ B に含まれる暗号スイートの順になるように調整する。この際、各グループ内での暗号スイートの優先順位は任意に定めてよい。また、グループ B の暗号スイートについては選択しなくてもよい。

表 27　高セキュリティ型での暗号スイートの推奨設定

	TLS1.3 を利用する場合
グループ A	TLS_AES_256_GCM_SHA384
	TLS_CHACHA20_POLY1305_SHA256
	TLS_AES_128_GCM_SHA256
	TLS_AES_128_CCM_SHA256
	TLS_AES_128_CCM_8_SHA256
鍵交換	ECDHE（優先）
	DHE
署名	ECDSA
	RSA-PSS
	RSASSA-PKCS1-v1_5

	TLS1.2 を利用する場合
グループ A	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_ECDHE_RSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM_8
	TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256
	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256
	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_ECDHE_RSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8

グループ B	TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
	TLS_DHE_RSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_DHE_RSA_WITH_AES_256_CCM
	TLS_DHE_RSA_WITH_AES_256_CCM_8
	TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256
	TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
	TLS_DHE_RSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_DHE_RSA_WITH_AES_128_CCM
	TLS_DHE_RSA_WITH_AES_128_CCM_8

6. セキュリティ例外型の要求設定

6.1 プロトコルバージョン

SSL2.0 から TLS1.3 までの安全性の違い（2.2 節参照）を踏まえ、TLS サーバがサポートするプロトコルバージョンについての「遵守項目」を以下のように定める。

【プロトコルバージョンの遵守項目】

- SSL3.0 及び SSL2.0 を設定無効（利用不可）にしなければならない
- TLS1.2 については、実装されているのであれば、設定有効にしなければならない。
- TLS1.0 及び TLS1.1 については、やむを得ず利用する場合に限り設定を容認する。必要性に応じて設定可否の判断を行わなければならない。

【プロトコルバージョンの推奨項目】

- TLS1.3 については、実装されているのであれば、設定有効にすべきである。ただし、TLS1.3 が実装されている場合であっても、TLS1.3 を明確に利用しないと判明している場合には TLS1.3 の設定有効化を必須とするものではない。
- プロトコルバージョンの優先順位が設定できる場合には、設定有効になっているプロトコルバージョンのうち、最も新しいバージョンによる接続を最優先とし、接続できない場合に順番に一つずつ前のプロトコルバージョンで接続するように設定すべきである。

表 28　セキュリティ例外型でのプロトコルバージョン設定

	TLS1.3*	TLS1.2	TLS1.1	TLS1.0	SSL3.0	SSL2.0
4 つのうちのいずれか	○	○	△	△	×	×
	－	○	△	△	×	×
	－	－	△	△	×	×
	－	－	－	△	×	×

○：設定有効　△：やむを得ず利用する場合に限り設定容認　×：設定無効化　－：実装なし

* TLS1.3 を明確に利用しないと判明している場合には、TLS1.3 設定有効化を必須としない

6.2 サーバ証明書

サーバ証明書についての「遵守項目」を以下のように定める。

現在発行されているサーバ証明書は、大多数が RSA と SHA-256 との組合せによるものである。RSA では 112 ビットセキュリティの鍵長（2048 ビット）が多い。

【サーバ証明書の遵守項目】

- 本ガイドライン作成時点（2024 年 3 月）で、多くのパブリック認証局から入手可能なサー

バ証明書のうち、安全性が高いものを利用しなければならない。なお、セキュリティ例外型においては、楕円曲線暗号を利用したサーバ証明書の場合、十分な相互接続性が確保できるとは必ずしも言えないため、RSA を利用すべきである。

表 29　セキュリティ例外型でのサーバ証明書設定

サーバ証明書の暗号アルゴリズムと鍵長	サーバ証明書の発行・更新を要求する際に生成する公開鍵情報（Subject Public Key Info）でのアルゴリズム（Subject Public Key Algorithm）と鍵長としては、以下を必須とする。 ● RSA（OID = 1.2.840.113549.1.1.1）で 112 ビットセキュリティ以上を満たす鍵長（2048 ビット以上） また、認証局が発行するサーバ証明書での署名アルゴリズム（Certificate Signature Algorithm）と鍵長については、以下を必須とする。 ● RSA 署名と SHA-256 の組合せ（sha256WithRSAEncryption; OID = 1.2.840.113549.1.1.11）で 112 ビットセキュリティ以上を満たす鍵長（2048 ビット以上）
サーバ証明書の発行・更新時の鍵情報の生成	● サーバ証明書の発行・更新を行う際に、自ら公開鍵と秘密鍵の鍵ペアを生成する場合には、既存の公開鍵と秘密鍵の鍵ペアを再利用せず、新たな公開鍵と秘密鍵の鍵ペアを生成しなければならない。 ● 上記の指示をサーバ管理者への仕様書、運用手順書、ガイドライン等に明示しなければならない。
クライアントでの警告表示の回避	● 当該サーバに接続することが想定されている全てのブラウザ（クライアント）に対して、以下のいずれかの手段を用いて警告表示が出ないようにしなければならない。 ➢ パブリック認証局からサーバ証明書を入手する ➢ 警告表示が出るブラウザ（クライアント）はサポート対象外であることを明示する、または警告表示が出ないサポート対象のブラウザ（クライアント）を明示する ➢ 7.2.9 節に従って、信頼できるプライベート認証局のルート CA 証明書を予めインストールする

6.3 暗号スイート

暗号スイートについての「遵守項目」及び「推奨項目」を以下のように定める。

なお、鍵交換に PSK または KRB が含まれる暗号スイートは、サーバとクライアントの両方で特別な設定をしなければ利用することができないため、本ガイドラインの対象外とする。

【暗号スイートの遵守項目】

- 以下の条件に該当する暗号アルゴリズムのいずれかを含む暗号スイートが選択されないようにするため、表 30 に記載される暗号アルゴリズム全てを設定無効（利用不可）としなければならない。
  - 2.5.2 節の表 9 に掲載されている暗号アルゴリズム

表 30　セキュリティ例外型での利用禁止暗号アルゴリズム一覧

署名		GOST R 34.10-2012, SM2（署名）
暗号化	ブロック暗号	RC2, EXPORT-RC2, IDEA, DES, EXPORT-DES, GOST 28147-89, Magma, 3-key Triple DES, Kuznyechik, ARIA, SEED, SM4
	暗号利用モード	CTR_OMAC
	ストリーム暗号	RC4, EXPORT-RC4
ハッシュ関数		MD5, GOST R 34.11-2012, SM3

- 鍵交換で DHE・DH を利用する場合には鍵長 1024 ビット以上を、RSA を利用する場合には 112 ビットセキュリティ以上を満たす鍵長（2048 ビット以上）を、ECDHE・ECDH を利用する場合には 128 ビットセキュリティ以上を満たす曲線（P-256 や Curve25519 を用いた X25519 など）[23]での設定をしなければならない。なお、DHE・DH の鍵長を明示的に適切に設定できない製品を利用する場合には、DHE・DH を含む暗号スイートは選定してはならない。

【暗号スイートの推奨項目】

- 以下の条件に該当する表 31 に記載される暗号アルゴリズムを組み合わせた暗号スイートのみが選択されるようにすべきである。
  - 2.5.2 節の表 7 に掲載されている暗号アルゴリズム
  - 2.5.2 節の表 8 に掲載されている暗号アルゴリズム

表 31　セキュリティ例外型での利用推奨暗号アルゴリズム一覧

鍵交換		DHE, ECDHE, RSAES-PKCS1-v1_5, DH, ECDH
署名		RSASSA-PKCS1-v1_5, RSASSA-PSS（TLS1.3 のみ）, ECDSA
暗号化	ブロック暗号	AES, Camellia（TLS1.2 まで）
	暗号利用モード	GCM, CCM, CCM_8, CBC
	認証暗号	ChaCha20-Poly1305
ハッシュ関数		SHA-256, SHA-384, SHA-1

[23] 2.6.2 節 表 11 に記載されている楕円曲線から選択すること。

- 安全性が高い暗号スイートから優先的に接続するようにするため、表 32 のようにグループ X、グループ Y、グループ Z の順に暗号スイートの優先順位を設定すべきである。

表 32　セキュリティ例外型での暗号スイートの優先順位

	【第一優先】 暗号利用モードの種類	【第二優先】 Perfect Forward Secrecy (PFS)の特性有無
グループ X	GCM, CCM, CCM_8 認証暗号	PFS あり
グループ Y	GCM, CCM, CCM_8 認証暗号	PFS なし
グループ Z	CBC	PFS あり・PFS なし

上記の方針に従った暗号スイートの推奨設定を表 33 に示す。表 33 は、表 31 に記載される暗号アルゴリズムのみで組み合わせた暗号スイートの全てが網羅されており、さらに表 32 に従ってグループ X、グループ Y、グループ Z の 3 つにグループ分けされている。

優先順位の設定ができない場合は表 33 全体の暗号スイートから、優先順位の設定ができる場合には各グループ内での暗号スイートから全部または一部を選択して設定する。

優先順位を設定する際には、グループ X に含まれる暗号スイート、グループ Y に含まれる暗号スイート、グループ Z に含まれる暗号スイートの順になるように調整する。この際、各グループ内での暗号スイートの優先順位は任意に定めてよい。また、グループ Y 以降の暗号スイートについては選択しなくてもよい。

表 33　セキュリティ例外型での暗号スイートの推奨設定

	TLS1.2 を利用する場合
グループ X	TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
	TLS_DHE_RSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_ECDHE_RSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_DHE_RSA_WITH_AES_128_CCM
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM
	TLS_DHE_RSA_WITH_AES_128_CCM_8
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8
	TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
	TLS_DHE_RSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_ECDHE_RSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_DHE_RSA_WITH_AES_256_CCM
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM
	TLS_DHE_RSA_WITH_AES_256_CCM_8
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM_8
	TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256
	TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256
	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256
グループ Y	TLS_RSA_WITH_AES_128_GCM_SHA256
	TLS_DH_RSA_WITH_AES_128_GCM_SHA256
	TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256
	TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256
	TLS_RSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_DH_RSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_ECDH_ECDSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_ECDH_RSA_WITH_CAMELLIA_128_GCM_SHA256
	TLS_RSA_WITH_AES_128_CCM
	TLS_RSA_WITH_AES_128_CCM_8
	TLS_RSA_WITH_AES_256_GCM_SHA384
	TLS_DH_RSA_WITH_AES_256_GCM_SHA384

グループ Y（続）	TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384
	TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384
	TLS_RSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_DH_RSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_ECDH_ECDSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_ECDH_RSA_WITH_CAMELLIA_256_GCM_SHA384
	TLS_RSA_WITH_AES_256_CCM
	TLS_RSA_WITH_AES_256_CCM_8
グループ Z	TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
	TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA256
	TLS_DHE_RSA_WITH_AES_128_CBC_SHA
	TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA
	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_CBC_SHA256
	TLS_ECDHE_RSA_WITH_CAMELLIA_128_CBC_SHA256
	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA256
	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA256
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA
	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA
	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_CBC_SHA384
	TLS_ECDHE_RSA_WITH_CAMELLIA_256_CBC_SHA384
	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
	TLS_RSA_WITH_AES_128_CBC_SHA256
	TLS_DH_RSA_WITH_AES_128_CBC_SHA256
	TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256
	TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256
	TLS_RSA_WITH_CAMELLIA_128_CBC_SHA256
	TLS_DH_RSA_WITH_CAMELLIA_128_CBC_SHA256
	TLS_ECDH_ECDSA_WITH_CAMELLIA_128_CBC_SHA256
	TLS_ECDH_RSA_WITH_CAMELLIA_128_CBC_SHA256
	TLS_RSA_WITH_AES_128_CBC_SHA

グループ Z（続）	TLS_DH_RSA_WITH_AES_128_CBC_SHA
	TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA
	TLS_ECDH_RSA_WITH_AES_128_CBC_SHA
	TLS_RSA_WITH_CAMELLIA_128_CBC_SHA
	TLS_DH_RSA_WITH_CAMELLIA_128_CBC_SHA
	TLS_RSA_WITH_AES_256_CBC_SHA256
	TLS_DH_RSA_WITH_AES_256_CBC_SHA256
	TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384
	TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384
	TLS_RSA_WITH_CAMELLIA_256_CBC_SHA256
	TLS_DH_RSA_WITH_CAMELLIA_256_CBC_SHA256
	TLS_ECDH_ECDSA_WITH_CAMELLIA_256_CBC_SHA384
	TLS_ECDH_RSA_WITH_CAMELLIA_256_CBC_SHA384
	TLS_RSA_WITH_AES_256_CBC_SHA
	TLS_DH_RSA_WITH_AES_256_CBC_SHA
	TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA
	TLS_ECDH_RSA_WITH_AES_256_CBC_SHA
	TLS_RSA_WITH_CAMELLIA_256_CBC_SHA
	TLS_DH_RSA_WITH_CAMELLIA_256_CBC_SHA

	TLS1.3 を利用する場合
グループ A	TLS_AES_128_GCM_SHA256
	TLS_AES_128_CCM_SHA256
	TLS_AES_128_CCM_8_SHA256
	TLS_AES_256_GCM_SHA384
	TLS_CHACHA20_POLY1305_SHA256
鍵交換	DHE
	ECDHE
署名	RSASSA-PKCS1-v1_5
	RSA-PSS
	ECDSA

7. TLS を安全に使うために考慮すべきこと

TLS をより安全に使うために、以下の項目についても考慮すべきである。

7.1 最新のセキュリティパッチの適用

プロトコルとしての脆弱性だけでなく、実装上の脆弱性が発見されることも時おり起きる。

そのような脆弱性が発見されると、基本的にはベンダからセキュリティパッチが提供されるので、ベンダが提供するセキュリティパッチを入手可能な状態とし、常にセキュリティパッチを適用して最新の状態にしておくべきである。

7.2 サーバ証明書の作成・管理について注意すべきこと

7.2.1 サーバ証明書での脆弱な鍵ペアの使用の回避

擬似乱数生成機能にエントロピー不足などの脆弱性が存在する場合、これを用いて鍵配送・共有や署名で使う公開鍵と秘密鍵の鍵ペアを生成した際に、結果的に解読容易な鍵ペアが生成されてしまうリスクがある。

こうしたリスクを防ぐためには、サーバ管理者は、鍵ペアの生成時点で脆弱性が指摘されていない暗号モジュールを利用するよう注意すべきである。

7.2.2 サーバ証明書を発行・更新する際に新しい鍵情報を生成する重要性

サーバ証明書を取得する際に、(鍵ペアにおける) 秘密鍵の生成・運用・管理が正しく行われないと、暗号化された通信データが第三者に復号されてしまうなどの問題が発生するリスクがある。例えば、とりわけ危険なのは、サーバ機器の出荷時に設定されているデフォルト鍵や、デフォルト設定のまま生成した鍵ペアを利用した場合、意図せず第三者と同じ秘密鍵を共有してしまうリスクがある。

また、サーバ証明書を再発行する必要性に迫られた時に、前回使用した CSR (Certificate Signing Request : サーバ証明書を発行するための署名要求) を使い回すと、同じ公開鍵と秘密鍵の鍵ペアのまま新しいサーバ証明書が作られることになり、以前の秘密鍵が漏えいした場合に最新の暗号化通信も復号できてしまうリスクがある。

こうしたリスクを防ぐためには、サーバ管理者は、サーバ証明書を取得・更新する際に既存の鍵ペアを使い回すことを厳に慎み、毎回新しく生成した鍵ペアを使った CSR でサーバ証明書を取得・更新することが望ましい。よって、本ガイドラインではサーバ証明書の遵守項目として位置付けている。

7.2.3 サーバ証明書の更新忘れ防止に対する対策例

サーバ管理者は、サーバ証明書の更新漏れによって自社のサービスに障害を発生させることが

ないように、サーバ証明書の有効期間を管理し、更新作業のために必要なリードタイムを考慮した上で、適切な管理方法（例えば、更新作業開始時期の明文化など）を定めることが求められる。

市販されているサーバ証明書の有効期間は、90 日程度のもの、1 年程度のもの等様々である[24]。一般に、有効期間が長いほど、サーバ証明書の更新頻度が少なく更新作業の工数を削減できる。その反面、鍵危殆化（秘密鍵の漏えい）リスクの増大、サーバ証明書に記載されたサーバの運営組織情報が（組織名変更などにより）正確でなくなるリスクの増大、アルゴリズムアジリティ（暗号アルゴリズムの危殆化に対して、別の安全な暗号アルゴリズムに移行するための対策に要する容易性）の低下などが危惧されるようになる。一方、サーバ証明書の有効期間を 90 日等の短期とする場合においては、更新作業が頻繁に発生する。そのため、更新作業を人力に頼っている場合においては、単純なミスによる更新忘れ、組織改編・担当者異動時の引き継ぎ不備による更新漏れ等のリスクが大きく上昇することとなる。そのため、短い有効期間のサーバ証明書を利用する場合においては自動的に証明書を更新できる体制の整備を行った上で実施することが現実的となる。

これらを総合的に勘案し、通常は有効期限の 3 ヶ月程度前からサーバ証明書の発行会社から有効期限切れに関するリマインドメールなどが送られてくるようになるため、リマインドメールに対して適切に対応する体制（責任者・担当者を明確にする、更新手順を決めるなど）を明確化しておくべきである。また、特段の制約が存在しない限り、1 年程度の有効期間を持つサーバ証明書を選択し、リマインダメールに依存することなく、サーバ証明書の更新作業を年次の定型業務と位置付けることが望ましい。

また、その他の手段として、証明書の自動発行・更新プロトコルである ACME を実装した発行サービスを利用し、自動で証明書を更新する設定を組み込むことも対策として挙げられる。これにより、更新作業コストの削減が見込める。ただし、自動更新機能を利用するにあたり、導入先のサーバ環境に適合するクライアントツールの選定など初期設定が必要であることを注意する必要がある。なお、ACME に関する詳細はコラム②を参照されたい。

[24] CA/ブラウザフォーラムによる「Baseline Requirement」でサーバ証明書の有効期限についての要件が規定されている。2011 年 11 月以降に発行するサーバ証明書の有効期限は 60 ヶ月以内とされていたが、その後、2015 年 4 月以降の発行では 39 ヶ月以内、2018 年 3 月以降の発行では 825 日（約 27 ヶ月）以内、2020 年 9 月以降の発行では 398 日（約 13 ヶ月）以内と、徐々に有効期限が短くなってきている。

【コラム②】 サーバ証明書の自動発行・更新プロトコル

スノーデン事件以降、政府機関等による広域盗聴等によるプライバシー侵害の懸念が顕在化し、全ての Web サーバへの接続を暗号化する機運が高まった。そのような背景のもと、より多くの Web サーバが暗号通信を使えるように、既存のサーバ証明書発行で必要であった支払い処理を排除し、サーバの設定やサーバ証明書の発行及び更新処理等を自動化するプロジェクトである Let's Encrypt プロジェクトが盛り上がりを見せた。Let's Encrypt プロジェクトの成果物のうち、自動化に係る仕組みは、IETF にて ACME（Automated Certificate Management Environment）プロトコルとして Standard Track の RFC8555 として刊行された。ACME プロトコルの普及は、Web サーバにおけるサーバ証明書の利用率の急激な上昇を招き、Web ブラウザがサーバ証明書の利用を必須化する後押しとなった。

ACME を利用した実装の多くは、サーバ証明書を導入するサーバとサーバ証明書を発行する認証局との間のやり取りをサポートしたものであり、両者が ACME プロトコルに対応し、発行するサーバ証明書が DV 証明書である時に特に有用となる。これは、RFC 8555 において認証局が行うべき身元確認スキームとして「ドメイン確認」にフォーカスしているためである。しかしながら、その他の用途のサーバ証明書の発行プロセスの全部又は一部を自動化する用途にも、ACME プロトコルは利用されつつある。

ACME プロトコルを用いたサービスも多数存在しており、それらを利用することでサーバ証明書の発行から更新も含めて自動化可能となるが、以下の点は少なくとも留意しておくべきであろう。

- 実装に瑕疵や脆弱性があれば、証明書の発行・更新が適切に処理されずに停止する可能性がある。
- 暗号スイートの設定変更などは ACME の対象外であり、サーバ管理者にて別途対応する必要がある。
- 仕様変更やアップデート等が発生した場合に、適宜対処が必要となることもある。

なお、ACME の導入はサーバ証明書の更新忘れの防止といった面での効果もあるが、計画通りに更新が行われていることはログなどにより確認するべきである。

7.2.4 サーバで使用する鍵ペアの適切な管理

サーバ管理者は、サーバ証明書に対応する秘密鍵について、紛失、漏えい等が発生しないように適切に管理しなければならない。秘密鍵の紛失（データ破壊を含む）に備えバックアップを作成し保管する場合には、秘密鍵の危殆化（漏えいなど）が発生しないようにするために、バックアップの方法や保管場所、その他の保管の要件について注意深く設計することが求められる。

サーバ管理者は、秘密鍵が危殆化した際に遅滞なく適切な対処を行うため、次のような事項について、あらかじめ方針及び手順を整理し、文書化しておくべきである。

- 秘密鍵の危殆化に対応するための体制（関係者と役割、委託先との連携を含む）
- 秘密鍵が危殆化した、またはその恐れがあると判断するための基準
- 秘密鍵の危殆化の原因を調べること、及び、原因の解消を図ること
- 当該サーバ証明書の利用を停止すること（実施の判断基準、手順を含む）
- 当該サーバ証明書を遅滞なく失効すること（実施の判断基準、手順を含む）
- 新しい鍵ペアを生成し、新鍵に対して新しくサーバ証明書の発行を行うこと
- 秘密鍵の危殆化についての情報の開示（通知先、通知の方法、公表の方針等）

CRYPTREC では、「暗号鍵管理システム設計指針（基本編）」という暗号鍵管理に関するガイドラインを発行[25]している。上記の検討にあたっては、特に同ガイドラインの 2 章（暗号鍵管理の在り方）、5 章（暗号アルゴリズム運用のための暗号鍵管理オペレーション対策）、6 章（暗号アルゴリズムの選択）、7 章（暗号アルゴリズム運用に必要な鍵情報の管理）も参考にされたい。

7.2.5 推奨されるサーバ証明書の種類

ブラウザなどをはじめとするサーバ証明書を検証するアプリケーションには、一定の基準に準拠した認証局の証明書（ルート CA 証明書）があらかじめ登録されており、これらの認証局（とその下位認証局）はパブリック認証局と呼ばれている。一般に、パブリック認証局が、第三者の立場から確認したサーバの運営組織等の情報を記載したサーバ証明書を発行し、ブラウザに予め搭載されたルート CA 証明書と組合せて検証を行うことで、サーバ証明書の信頼性を確保する。これにより、当該サーバ証明書の正当性が確認できれば、ブラウザは警告表示することなく、当該サーバへの接続を行う。

パブリック認証局から発行されるサーバ証明書は、その用途や利用範囲に応じて表 34 に示す 3 種類に分類される。

これらのサーバ証明書のうち、不特定多数の利用者がアクセスする一般的な Web サーバ用途であれば、運営サイトの法的実在性の確認が行われる EV 証明書が利用者にとって本来一番安心できるサーバ証明書である。実際、欧州では、信頼されるサービス（Trusted Service）であることを示すために、Qualified Website Authentication Certificate（QWAC）という EV と同等の実在確認と組織認証を行う証明書が利用されている。厳密な実在確認を行うため取得コストはかかるものの、以前はブラウザのアドレス表示部分等が緑色になる「グリーンバー」表示による視認性の高さが

[25] https://www.ipa.go.jp/security/vuln/ckms.html
https://www.cryptrec.go.jp/report/cryptrec-gl-3002-1.0.pdf

EV 証明書の大きなアドバンテージであった。しかし、現在の主要ブラウザではグリーンバー表示を廃止する傾向にあり、OV 証明書や DV 証明書との視認性の差は低下している。もっとも、アドレスバーにある「錠前」マークをクリックすると表示される「証明書の簡易ビューア」では、EV 証明書は OV 証明書や DV 証明書よりも発行先を含む情報が表示されるなど、依然として視認性の差は存在している。

図 13　証明書の簡易ビューアの表示例（左：EV 証明書の場合、右：EV 証明書でない場合）

OV 証明書は、EV 証明書と同様、不特定多数の利用者がアクセスする一般的な Web サーバ用途で用いられることが多く、運営サイトの実在性の確認も行われる。

表 34 に示した通り、EV 証明書ほどは厳格な実在確認は行われないため、EV 証明書と比較すると取得コストが安価である。その一方、ブラウザにおける視認性はやや低く、DV 証明書との識別が難しいものもある[26]。

DV 証明書は、個人でも取得することができる証明書である。しかも、Let's Encrypt プロジェクト[27]が DV 証明書を無料発行するなど、非常に入手コストが安い。一方で、運営サイトの実在性の確認は行われないため、フィッシングサイトなどに利用されるサーバ証明書のほぼすべてが DV 証明書を使っている。したがって、企業が DV 証明書を使うのは、不特定多数の利用者がアクセスする一般的な Web サーバよりも、特定少数の利用者向けの Web サーバや実証実験などのテストサーバなどにとどめることが望ましい。

そこで、不特定多数の利用者がブラウザでアクセスする一般的な Web サーバ、とりわけドメイン名のなりすましリスクや運営組織の誤認リスクを避けたい場合（例：EC サイトや企業の公式 HP など）については、EV 証明書の利用をまず検討すべきである。それ以外の利用ケースにおいては、入手コストと各々の証明書で実現される効用とのバランスを考慮して決めるべきである。例えば、ブラウザ以外のスマートフォン用アプリケーションでの利用など、アドレスバー視認性

[26] 「証明書の簡易ビューア」から「証明書ビューア」を表示して、「対象者（subject）」フィールドの内容や「証明書ポリシー」の OID を確認することで EV、OV、DV の各証明書の判別ができる。
[27] https://letsencrypt.org/

の高さが求められないようなケースでは、EV 証明書のメリットが十分に生かせないので、OV 証明書や DV 証明書も有力な選択肢となる。

表 34　サーバ証明書の種類と違い

サーバ証明書の種類	内容の違い
DV 証明書 (Domain Validation)	サーバの運営組織が、サーバ証明書に記載されるドメインの利用権を有することを確認したうえで発行される証明書。 オンライン申請による短時間発行や低コストで入手できるものが多く、コラム②にある ACME プロトコルとの親和性が高い、などのメリットがある。 一方、サーバの運営組織の実在性や、ドメイン名と運営組織の関係については確認されないので、ドメイン名以外の主体者名（運営組織名などの発行先情報）は基本的に表示されない。このため、自らのドメイン名と非常によく似たドメイン名の DV 証明書を、異なる運営組織に入手・利用されやすいことを念頭に置いておく必要がある。場合によっては、不特定の利用者にサーバの運営組織をあえて誤認させる手段に利用される可能性もあることに留意されたい（コラム④も参照のこと）。
OV 証明書 (Organization Validation)	ドメイン名の利用権に加えて、サーバ運営組織の実在性の確認やドメイン名と運営組織との関係などについても確認した上で発行される証明書。 不特定多数の利用者がアクセスするような一般的な Web サーバの用途で利用される。しかし、①証明書ビューアで確認しない限り、ブラウザのアドレス表示部分では OV 証明書と DV 証明書を識別できない、②サーバ運営組織等の確認項目や確認方法は個々の認証局によって異なる、という課題もある。
EV 証明書 (Extended Validation)	OV 証明書と同様、ドメイン名の利用権に加えて、サーバ運営組織の実在性等の確認やドメイン名と運営組織との関係などについても確認した上で発行される証明書。 3 つの証明書のなかでは発行コストが最もかかるが、以下の点で DV 証明書や OV 証明書に対して優位点を持つ。 ● 運営組織の法的実在性について、CA/ブラウザフォーラムが規定した国際的な認定基準にもとづいて確認が行われる。このため認証局に依らず一定レベルの確認が保証される ● 多くのブラウザでは、証明書の簡易ビューアで主体者名（運営組織名などの発行先情報）が表示され、EV 証明書であることが識別できる。

【コラム③】サーバ証明書解析からフィッシングサイトを見つけ出せるか？

コラム④にもあるように、フィッシングサイトの HTTPS 化が急速に進んでいる。その際、必ず必要となるのが「（証明書の仕組みとしては正当なフィッシングサイト用の）サーバ証明書」である。

ここで注目すべきは、フィッシングサイトは、その性質上、短期間の使い捨てで変化していくため、フィッシングサイト用のサーバ証明書の有効期間は長くする必要がない。また、金銭目的であることが多いことを考えればコストメリットを重視し不必要なコストはかけたくないはずである。実際、F5 Labs が公表した記事[28]によれば、多くのフィッシングサイトでは cPanel と Let's Encrypt が発行する「**無料のサーバ証明書を使い**」、フィッシングサイトの 36%は「**90 日しか活動しておらず**」、サーバ証明書の「**自動発行プロセスを活用している**」と強く推定されるとのことである。

このようなフィッシングサイト構築者の行動パターンを逆手に取り、実際のフィッシングサイトで使われたサーバ証明書をもとに、今後フィッシングサイトに使われる可能性があるサーバ証明書を検知しよう、という研究発表[29]が CSS2019 であった。

本研究では、OpenPhish が 2018 年 10 月～2019 年 1 月に収集したデータセットの中から実際のフィッシングサイトで使用されたサーバ証明書 1,634 個を抽出し、先生役としてのフィッシングデータとしている。ちなみに、1,634 個のうち、cPanel と Let's Encrypt が発行するものは合計で 1,566 個、95.8%にも達していることが示されている。

これらフィッシングサイトで使われたサーバ証明書のコモンネームの類似性を分析して作成した検知用テンプレートから未知のフィッシングサイトで使われるサーバ証明書を事前に検出できる可能性があることを明らかにした。実際、69 個の検知用テンプレートを作成した後、Censys が保有する cPanel と Let's Encrypt が同時期に発行した約 3,800 万個のサーバ証明書について探索したところ、1,650 個の証明書が検出された。特にある 1 つのテンプレートは 900 を超えるサーバ証明書のコモンネームと合致しており、それらをより詳細に調べたところ、93%の証明書が Let's Encrypt から発行されていた。加えて、フィッシングサイト作成サービスの存在を発見するに至った。

フィッシングサイトの検出手法といえば、今までドメイン名や URL、メール等の分析結果を基にしたものが一般的であったが、フィッシングサイトの HTTPS 化に伴い必然的にサーバ証明書が必要となることから「サーバ証明書解析に基づく検出手法」が有効な対策の一つになる可能性があり、今後の研究が期待されるところである。

一方、サーバ運営者は、サーバ証明書の選択にあたって、無料のサーバ証明書がフィッシングサイトにも多く使われていることに十分注意を払うべきであり、場合によっては無料のサーバ証明書は使わないといった配慮が必要となる。

---

[28] F5 Labs, "2019 PHISHING AND FRAUD REPORT," https://www.f5.com/content/dam/f5-labs-v2/article/pdfs/F5Labs_2019_Phishing_and_Fraud_Report.pdf

[29] 櫻井、渡邉、奥田、秋山、森、「サーバ証明書解析によるフィッシングサイトの発見手法」、CSS2019

7.2.6 DNS の CAA（Certification Authority Authorization）設定による証明書不正発行の防止

Web サイト管理者は、DNS リソースレコードの一種である CAA に、1 つ以上の認証局事業者（の所有する DNS ドメインネーム）を記載することにより、所有する DNS ドメインネームに対し証明書を発行可能な認証局事業者を指定できる。

DNS の CAA リソースレコード（以下、CAA）は 2013 年に RFC 6844 として定められ、2019 年 11 月に RFC 8659 として改訂された。2017 年 9 月に CA 及びブラウザベンダの業界団体である「CA/ブラウザフォーラム」が、認証局事業者に対し CAA の確認を必須化したことにより、徐々に利用されつつある。なお、SSL Pulse[30]によると、CAA の普及率は 2023 年 11 月時点で 14.3%となっている。

CAA の第一の目的は、他の認証局事業者の意図しない証明書誤発行を削減することである。証明書発行後に、その証明書が適切か否かを判断する為の TLSA リソースレコード（RFC 6698[31]で利用される）とは目的が異なる点に注意されたい。

CAA の設定は、①証明書を発行する認証局事業者のドメインネームを、②DNS ドメインネーム所有者が、③所定のタグの値へ記載する、ことにより行われる。以上の三つのプロセスについて、順に説明を行う。

① 証明書を発行する認証局事業者のドメインネームを、各認証局事業者の案内ページ等[32]で確認する。
② DNS リソースレコードを管理している主体（例えば DNS サービスプロバイダ）に、CAA を設定するよう依頼を行う。設定方法は各 DNS サービスプロバイダの案内ページ等を参照する。
③ 証明書を発行する認証局事業者のドメインネームを issue タグの値へ記載する。ワイルドカード証明書を発行する認証局事業者を別に指定したい時は issuewild タグの値へ記載する。なお、ワイルドカード証明書の発行を完全に禁止したい場合は、issuewild タグの値へ空文字（""）を記載する。

ここで、CAA に記載がない場合は、任意の認証局事業者が証明書を発行できることとなる。もっとも、そのドメインに CAA が設定されていなくても、上位ドメインに CAA が設定されている場合は、その設定が反映されるので注意が必要となる。

[30] https://www.ssllabs.com/ssl-pulse/
[31] RFC 6698 DNS-based Authentication of Named Entity (DANE): Transport Layer Security (TLS) Protocol: TLSA
[32] CA/ブラウザフォーラムに登録されたドメインネーム一覧は以下で確認できる。
https://ccadb-public.secure.force.com/mozilla/AllCAAIdentifiersReport

7.2.7 Certificate Transparency（CT）：全世界のサーバ証明書が確認できる仕組み

2010 年代前半、Web サーバ向けのサーバ証明書を発行する複数の認証局事業者において、サーバ証明書の誤発行や不正発行のインシデントが相次いで発生した。Certificate Transparency（CT）は、それらのインシデントを受けて実施された様々な取り組みの 1 つであり、IETF にて Experimental な RFC として刊行された[RFC 6962][RFC 9162]。

2023 年 11 月時点において、Apple 社は、全ての“Public Trusted”なサーバ証明書の発行に際して、各サーバ証明書が複数の CT ログに記録されることを要求している[33]。また、それら CT ログに保管された情報を収集・統合してモニタリングするような公開サービス等も存在し、そのようなサービスを利用することで、全ての発行済み証明書の一元的検索を行うこと等も可能となっている。CT の登場以前では、サーバ証明書の誤発行や不正発行を速やかに確認することは当該サーバ証明書を発行する認証局事業者以外には難しかった。しかしながら、現在においては CT ログの内容を検証することで、誰もが速やかに不適切なサーバ証明書の発行を確認できることとなった。

CT ログへの登録は、以下のようになされる。

- Web サイト運営者が、認証局（CA）に対してサーバ証明書を要求する。
- CA は、発行するサーバ証明書の情報を CT ログサーバに送付する。
- CT ログサーバは、受け取った情報をログとして記憶する。
- CT ログサーバにサーバ証明書が登録されると、CT ログサーバはそのサーバ証明書が確かに登録されたことを示す情報である SCT（Signed Certificate Timestamp）を生成し、CA に送付する。
- CA は、サーバ証明書と SCT を併せて、Web サイト運営者に送付する。

上記のような手続きで CT ログサーバに登録された情報は、Merkle ツリー技術を利用して登録されており、過去に登録された情報を削除又は変更することはできない。加えて、CT ログが適切に機能していることは、モニタリングサービスにより監視されている。CT ログサーバに登録された情報は一般に公開されており、CA 事業者以外が閲覧することもできる。また、これらの情報を検索可能なサービス[34]や、統計情報を提供するサイト[35]も存在する。

一般に、Web サーバ向けのサーバ証明書を発行すると、そのサーバ証明書に記載される情報は CT ログサーバに登録され、速やかに公開されることとなる。そのため、サーバ証明書の発行に際しては、記載事項に秘密の情報が含まれていないように配慮する必要がある。例えば、ある製造事業者のドメイン名から未発表の商品が近い将来発売されることが容易に推測できる場合[36]において、製造事業者が商品の発売を発表する前にそのドメイン名が記載されたサーバ証明書を発行したとすると、CT ログサーバに記載される情報から発表前商品が販売される事実が推測されてしまう。そのような状況にならないようにするためには、未発表の新商品のキャンペーンペー

[33] https://support.apple.com/en-us/HT205280
[34] https://crt.sh/
[35] https://ct.cloudflare.com/
[36] 例えば、<商品名>.example.com のようなドメイン名。

ジ等を作成する段において、サーバ証明書に記載するドメイン名から商品名等を推測することが難しくなるような配慮が必要となる。

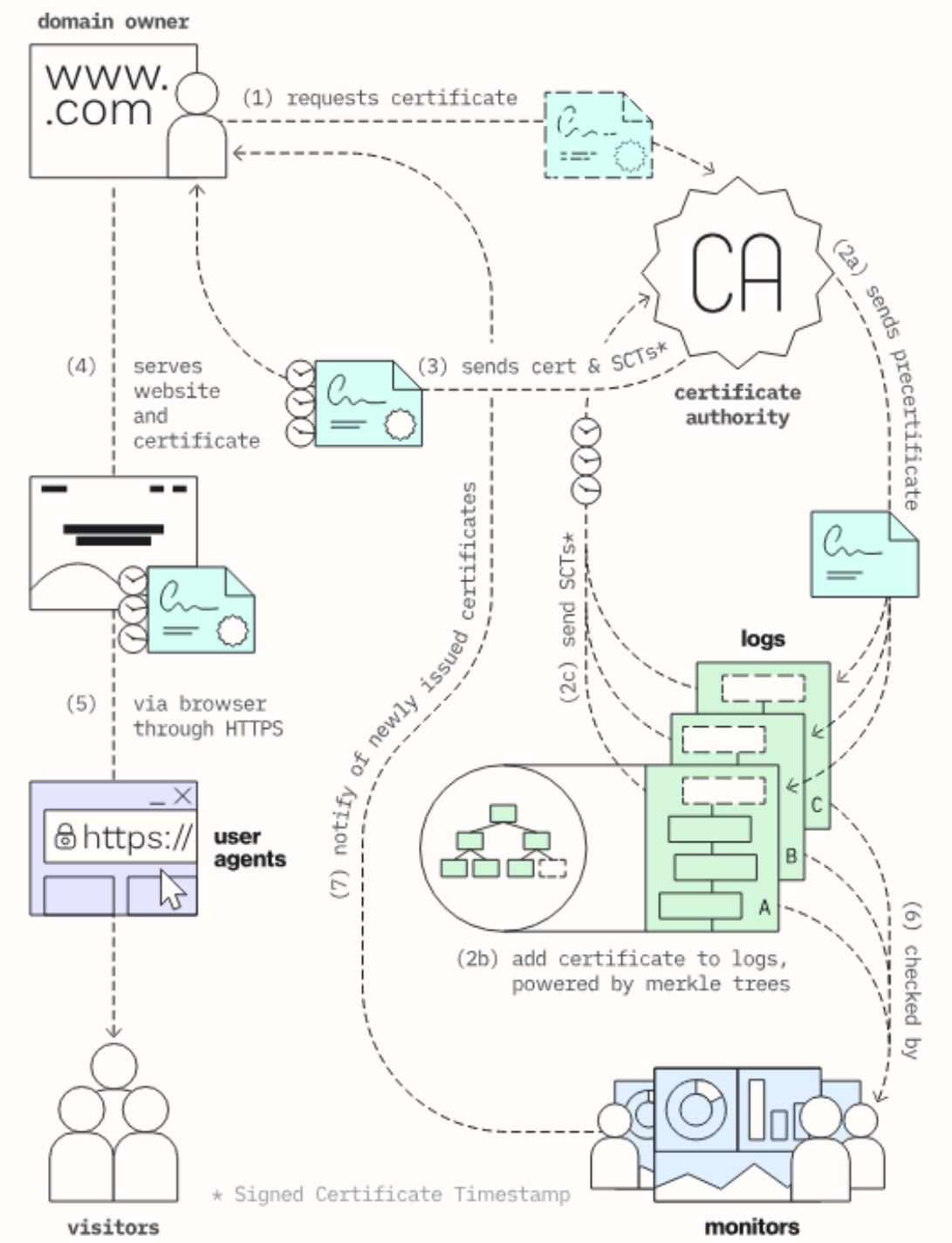


図 14　CT ログのエコシステム（Certificate Transparency Community Site[37]より引用）

[37] https://certificate.transparency.dev/
https://github.com/google/certificate-transparency-community-site

7.2.8 複数サーバに同一のサーバ証明書（ワイルドカード証明書／マルチドメイン証明書）を利用する場合の注意点

サーバ証明書では 1 枚の証明書に複数のドメイン名（FQDN）あるいはワイルドカードを用いたドメイン名を記載することができ、前者はマルチドメイン証明書、後者はワイルドカード証明書と呼ばれている。

これらのサーバ証明書は、各サーバに同じ鍵ペアと証明書を共有することができるため、負荷分散や冗長化による可用性向上などを目的とした複数サーバの構成管理などに有用である。

サーバ管理者としては、鍵ペアとそれにひもづく複数のサーバとを両方管理するケースと、（鍵ペアはホスティング業者などで別途管理されており）サーバだけを管理するケースが考えられ、本節では前者にフォーカスする。

サーバ管理者は、複数のサーバで同一の鍵ペア・証明書を共有することになるため、以下の点に留意する必要がある[38]。

- サーバ証明書の更新や再発行の際には、入替作業の対象となるすべてのサーバについて漏れなく鍵ペア・証明書を入れ替えること
- サーバ証明書の入替に伴って暗号スイートの設定変更などを行う場合は、対象とするすべてのサーバについて漏れなく適用すること

これらの運用を円滑に行うために、サーバ管理者は、サーバ証明書の作業対象となるサーバに漏れが発生しないよう、各サーバに対応する鍵ペアや証明書、暗号スイートなどの情報を一覧管理し、管理方法についても規定・文書化することが推奨される。

7.2.9 プライベート認証局の利用の注意点

証明書の発行プログラムさえあれば誰もがサーバ証明書を作ることができるが、これではサーバ構築者が“自分は正当なサーバ”であると自己主張しているに過ぎない。このようなサーバ証明書は“オレオレ証明書”ともいわれ、ブラウザでは当該サーバ証明書の正当性が確認できない“危険なサーバ”として警告が表示される。

本来、TLS における重要な役割の一つが接続するサーバの認証であり、その認証をサーバ証明書で行う仕組みである以上、“危険なサーバ”との警告表示が出るにもかかわらず、その警告を無視して接続するよう指示しなければならないサーバ構築の仕方をすべきではない。

例外的に、社内向けシステムやクローズドシステムなど、利用者や利用端末が限定される環境である場合には、プライベート認証局を立ち上げ、その管理下でシステムが運用されることがある。このような場合、プライベート認証局が発行したサーバ証明書を使ったサーバを“危険なサーバ”として認識させない手段として、当該サーバ証明書の正当性を確認するためのプライベート認証局のルート CA 証明書を、ブラウザ（クライアント）の「信頼できるルート CA」に手動で

[38] 特に、異なるポリシーによって管理される複数のドメイン名や組織を同一証明書に混在させる場合は、ポリシー間の整合や各組織との調整等も含めて留意する必要がある。

インストールする方法がある。

ただし、パブリック認証局のルート CA 証明書とは異なり、これら手動インストールしたプライベート認証局のルート CA 証明書はブラウザベンダによって管理されていないため、万が一、当該ルート CA 証明書の安全性に問題が生じた場合でも、ブラウザベンダによって自動的に無効化されることはなく、ブラウザ（クライアント）にインストールした当該ルート CA 証明書を手動で削除する必要がある。もし削除を怠ると、ブラウザ（クライアント）側では不正なサーバ証明書であっても正しいサーバ証明書と誤認するリスクが増大する。

このため、例外的にプライベート認証局のルート CA 証明書を手動インストールする必要があるシステムの場合には、当該ルート CA 証明書の安全性に問題が生じた場合にインストールを実施・管理する主体によって、全ての対象ブラウザ（クライアント）から当該ルート CA 証明書の無効化や削除ができるようにする等、具体的な対策を実施する体制を整えるべきである。例えば、企業が自身の管理する端末に対してシステム管理部門が運用しているプライベート認証局のルート CA 証明書を搭載し、一定のポリシーに基づいて端末管理を行っている場合、管理システムなどから各端末にルート CA 証明書を自動更新（インストール及び削除）する仕組みを提供するなどである。一例として Windows クライアントに対して Active Directory から自動更新する場合の構成例を Appendix D.2 に示す。

このような仕組みを用いたシステムにおいて、当該ルート CA 証明書の安全性に問題が生じた場合には、速やかにシステム管理部門が各端末に対して当該ルート CA 証明書を無効化する措置を講じなければならない。

7.3 委託先のサーバ（PaaS／SaaS）を利用する場合の注意点

本節では、委託先のサーバ（PaaS／SaaS）を利用する場合の注意点について記載する。

A) 提供されるレイヤにより、利用者がコントロール可能な範囲が異なることに留意

クラウド等のサービスにコンピューティングリソースを配置する場合、クラウドサービスプロバイダ（CSP）によって提供されるレイヤによって、利用者による管理の要否が異なる場合がある。CSP が管理すべきレイヤが多い場合、運用上の負荷やコストを軽減することができるが、統制上のオーナーシップをより利用者が保有することが重要な場合もあるため、サービスの選択において、組織の統制モデル、実現可能性、運用やコストへの負担を踏まえたうえで選択することが望まれる。

- **管理レイヤの差異による違い**

  SaaS に近いサービスであれば、経路暗号化がサービスに組み込み済みのもの（利用者がコントロールせず、CSP がコントロール）が存在する。IaaS や PaaS レイヤにおいても、利用者が自らサードパーティの証明書を導入するケースもあれば、CSP が証明書サービスを提供しており、利用可能である場合もある。こうしたサービスは、更新管理の自動化や通知の実装

は可能なケースもある一方、利用可能な証明書の制限（EV 等の利用制限）も存在する。特にサービスを利用する場合、利用を禁止とすべき技術（危殆化したバージョン等）等をどのように管理可能かを留意すべきである。

B) CSP が提供する証明書の制限

CSP が証明書管理サービスを提供している場合、適用可能なバージョンや対象はサービスの機能に依存する（例えば、CSP のプラットフォーム上のサーバインスタンスにのみ証明書が適用可能等）。コストや管理面によるメリットを理解したうえで、適用範囲を決定する必要がある。

- **主な考慮事項**

  CSP が提供する証明書に対する考慮事項は主に以下の項目があげられる。
  - 適用対象（証明書を導入可能なサーバインスタンスの特定や、連携可能なサービス）
  - コスト（有償、無償）
  - 証明書の種類（EV、OV、DV）
  - 管理範囲（更新処理の自動化等）

C) アーキテクチャによる実装の違いに留意

IaaS や PaaS などを利用する場合、ネットワークアーキテクチャを踏まえて、利用者が設計するケースがある。その場合、ネットワーク上のどの範囲までを経路暗号化の対象とすべきかを考慮したうえで設計する必要がある。一般にインターネットなどの公衆網を暗号化通信の適用範囲とすることは多くのケースで推奨される。その一方、CSP が統制するネットワーク範囲をどの程度、暗号化通信の対象とすべきかは、利用者の設計に委ねられる。

CSP の提供するサービスにもよるが、処理を行うサーバで終端処理を行うパターン、ロードバランサーやゲートウェイサービスで終端処理を行うパターン、CDN（Content Delivery Network）のエッジで終端処理を行うパターンがある。

i) 処理を行うサーバで終端処理を行う

Web サーバ（Web アプリケーションサーバ）で終端処理を行う。

利点： End to End の暗号化を実現する

欠点： コスト及び管理の複雑性、サーバのリソース処理が増加する。処理を行うサーバがスケールアウト（負荷に対応するため、弾力的に増加をする設計）する場合、終端処理を行うサーバの台数分の証明書管理が必要となる場合がある。

ii) ロードバランサーやゲートウェイサービスで終端処理を行う

実際に処理を行うサーバのフロントに配置されたロードバランサー等の疎結合層において終端処理を行う。

利点： 処理のオフロード及び管理の集中化により、運用やコストの負荷が軽減される

欠点： プロバイダによっては、CSP 内の通信網は平文の通信となる（ただし、CSP 内における通信に対する盗聴リスクを他のコントロールで評価することは可能である）

iii) CDN（Content Delivery Network）のエッジで終端処理を行う

利用者に近いエッジで終端処理を行うことでより暗号化及び復号処理の分散や処理の高速化を実現する。

利点： 処理のオフロード及び管理の集中化により、運用やコストの負荷が軽減される

欠点： プロバイダによっては、エッジから CSP 及び CSP 内の通信網は平文の通信となる（ただし、CSP 内における通信に対する盗聴リスクを他のコントロールで評価することは可能である）

7.4 さらに安全性を高めるために

7.4.1 HTTP Strict Transport Security（HSTS）の設定有効化

例えばオンラインショッピングサイトのトップページが暗号化なしの HTTP サイトで、ショッピングを開始する際に HTTPS へリダイレクトされるような構成になっていた場合、最初の HTTP 通信は通信上の攻撃者によって改ざんされる可能性がある。リダイレクトして悪意のあるサイトに誘導し情報を抜き取ったり、SSL strip というツールを用いて HTTPS へのリダイレクトを防ぎつつあたかも本物のサイトのように振る舞い情報を盗んだりといった手法の報告が Moxie Marlinspike によってなされた。

この攻撃のように、サーバが HTTPS に対応していてもユーザが http://で始まる URL でサイトにアクセスすると、通信は平文となり、メッセージを改ざんされうる。Web サイトが暗号化なしの HTTP をサポートしている限り、攻撃者がメールで誘導するなど http://のリンクを攻撃対象者に開かせることで、平文通信を行わせることが出来る。

これを防ぐため RFC 6797 で規定されている HTTP Strict Transport Security（HSTS）では、サーバ側から以降 HTTP でのアクセスは行わず HTTPS でアクセスするようブラウザに指示できる。具体的には、HTTPS 応答に以下のような HTTP レスポンスヘッダを含めることでブラウザに指示を送る。

Strict-Transport-Security:max-age=有効期間秒数; includeSubDomains

この HTTP レスポンスヘッダを受け取った HSTS 対応のブラウザは、有効期間中は当該ドメインへのアクセスは HTTP ではなく全て HTTPS で通信するように自動設定しておく。includeSubDomains が指定されていた場合、サブドメインへのアクセスも同様である。

これにより、以前接続したときに HSTS が有効になっているドメインであれば、何らかの理由で、ブラウザが HTTP で接続しようとしても自動的に HTTPS に切り替えて接続する[39]。

以上のように、HTTPS で安全にサービスを提供したい場合などでは、ユーザに意識させること

[39] HSTS はサーバ側の仕組みであるが、ブラウザ側の仕組みとして Chrome では HTTPS-FIRST mode という取り組みを進めている。
https://developers-jp.googleblog.com/2023/09/https.html

なくミスを防止でき、ユーザの利便性を向上させることができるので、HSTS の機能を持っているならば有効にすることを推奨する。

7.4.2 OCSP Stapling の設定有効化

サービス提供の終了やサーバの秘密鍵の漏えいなど、何らかの理由で、サーバ証明書の有効期間内であっても当該サーバ証明書が失効している場合がある。そのため、サーバ証明書の正当性を確認する時には、当該サーバ証明書が失効していないかどうかもあわせて確認すべきである。

サーバ証明書が失効されていないか確認する方法として、CRL（Certificate Revocation List）とOCSP（Online Certificate Status Protocol）の二つの方法があるが、CRL はサイト数の増大に伴ってファイルサイズが増大しており、近年では OCSP のみに依存するブラウザが多くを占めている。

ただ、OCSP を使用した場合には 2 つの問題がある。

i) OCSP 実行時の通信エラー処理について明確な規定がなく、ブラウザの実装に依存する。このため、OCSP レスポンダの通信障害等で適切な OCSP 応答が得られない場合にサーバ証明書の失効検証を正しく行わないまま TLS 通信を許可してしまうブラウザも少なくない。そのようなブラウザに対しては、あるサイトのサーバ証明書が失効していたとしても、DDoS 攻撃などにより意図的に OCSP レスポンダに接続させないことにより、当該サイトが有効であるとして TLS 通信をさせることができる

ii) OCSP を使った場合には、あるサイトにアクセスがあったことを OCSP レスポンダも知り得てしまうため、プライバシー上の懸念がある。例えば、ある利用者が、ある会員制のサイトにアクセスした場合、ブラウザはサーバ証明書の失効検証のために当該サイトの OCSP 応答を取得する。そこで、OCSP レスポンダのアクセス履歴から、ある接続元 IP の利用者は、当該サイトの会員であると OCSP レスポンダが知り得ることになる

上記の問題を解決するために、RFC 6066 Transport Layer Security (TLS) Extension: Extension Definition の 8 節で、Certificate Status Request という TLS 拡張が規定されている。この拡張には以下の特長があり、これを使うことにより OCSP 応答を OCSP レスポンダからではなく、アクセス先サイトの Web サーバから TLS ハンドシェイク中に OCSP レスポンスもできる。

- OCSP レスポンダからの OCSP 応答を Web サーバがキャッシュしている限り、ブラウザは OCSP 応答による失効検証を行うことができる
- OCSP 応答を、OCSP レスポンダからではなく、Web サーバから取得するので、当該サイトへのアクセス履歴を OCSP レスポンダが知ることはない

7.4.3 Public Key Pinning のサポート終了について

HPKP（HTTP Public Key Pinning）は、不正に発行されたサーバ証明書による通信を検知するための仕組みとして 2011 年に Google により提案され[40]、2015 年に RFC 7469 として標準化された

[40] ImperialViolet: Public Key Pinning, May 2011, https://www.imperialviolet.org/2011/05/04/pinning.html

[41]。過去には、Chrome、Firefox、Opera など一部のウェブブラウザで HPKP をサポートしていたが、

- 仕組みが複雑で、運用や設定ミスが発生した場合、数か月単位で指定された FQDN での TLS 暗号通信ができなくなるケースが発生した
- HPKP を設定しているサーバの比率はかなり低く、今なお減少傾向にある

などの理由により、2019 年 1 月ごろ Chrome、Firefox は HPKP のサポートを終了した[42][43]。これに伴い、今後 HPKP が広く使われることはないと判断し、本ガイドラインでも HPKP の設定方法の記述は削除することとした。

[41] RFC 7469 Public Key Pinning Extension for HTTP, Apr 2015, https://tools.ietf.org/html/rfc7469
[42] Can I use HPKP https://caniuse.com/#search=hpkp
[43] Chrome Platform Status: Remove HTTP-Based Public Key Pinning (removed) https://www.chromestatus.com/feature/5903385005916160

PART II：

ブラウザ＆リモートアクセスの利用について

8. ブラウザを利用する際に注意すべきポイント

本ガイドラインに記載の情報に関し、マイクロソフト及び Google についての情報は 2023 年 11 月時点で各社から直接提供していただいたものである。また、Mozilla 及び Apple についての情報は各社の公式ホームページから入手したものである。

8.1 本ガイドラインが対象とするブラウザ

ブラウザは、少なくとも提供ベンダがメンテナンスしているバージョンのものを利用すべきである。

マイクロソフト

- Edge　Windows のサポート期限[*1] に準じてメンテナンスされる

  [*1)] Windows のサポート期限はモダン ライフサイクル ポリシーに従い半年ごとのメジャーアップデートを適用することでサポートを受けることができる
- Internet Explorer 11　一部の企業向けの Windows[*2] を除いて、Internet Explorer 11 のサポートは、2022 年 6 月 15 日に終了している。

  [*2)] 現在サポートされているすべての Windows 10 LTSC リリース及び、企業向け延長セキュリティアップデート Extended Security Updates (ESU)を利用している Windows Server 2012 及び Windows Server 2012 R2

Google

- Chrome　最新バージョン（2023 年 11 月時点）は以下のシステム要件を満たす場合に利用できる

  参考：https://support.google.com/chrome/a/answer/7100626
  - Windows
    - Windows 10 or later or Windows Server 2016 or lator
    - An Intel Pentium 4 processor or later that's SSE3 capable
  - Mac
    - macOS Catalina 10.15 or later
  - Linux
    - 64-bit Ubuntu 18.04+, Debian 10+, openSUSE 15.2+, or Fedora Linux 32+
    - An Intel Pentium 4 processor or later that's SSE3 capable
  - Android
    - Android 7.0 Nougat 以上
  - iPhone & iPad
    - iOS 15 or later / iPadOS 15 or lator

Mozilla

- Firefox　最新バージョン（2023 年 11 月時点）は以下のシステム要件を満たす場合に利

用できる

参考：https://www.mozilla.org/en-US/firefox/120.0/system-requirements/

- Windows（32-bit and 64-bit）
  - Windows 10 or later
  - 1 gigahertz (GHz) or faster compatible processor or System on a Chip (SoC)
  - 1GB of RAM / 2GB of RAM for the 64-bit version
  - 500MB of hard drive space
- Mac
  - macOS 10.15 or later
  - Macintosh computer with an Intel x86 or Apple silicon processor
  - 512 MB of RAM
  - 200 MB hard drive space
- Linux
  - glibc 2.17 or higher
  - GTK+ 3.14 or higher
  - libdbus-glib 0.6.0 or higher
  - libglib 2.42 or higher
  - libstdc++ 4.8.1 or higher
  - X.Org 1.0 or higher (1.7 or higher is recommended)
- Android

  参考：https://www.mozilla.org/en-US/firefox/android/120.0/system-requirements/
  - Android 5.0 or newer
- iPhone / iPad

  参考：https://www.mozilla.org/en-US/firefox/ios/120.0/system-requirements/
  - iOS 15 or later

Apple

- Safari　最新バージョン（2023 年 11 月時点）は以下のシステム要件を満たす場合に利用できる

  参考：https://developer.apple.com/documentation/safari-release-notes/safari-17_1-release-notes
  - Mac
    - macOS Sonoma（macOS 14）
    - macOS Ventura（macOS 13）
    - macOS Monterey（macOS 12）
  - iOS
    - iOS 17.1
    - iPadOS 17.1

8.2 設定に関する確認項目

8.2.1 基本原則

8.1 節で対象とするブラウザは、インストール時のデフォルト設定で利用することを各ベンダは推奨しているので、企業の情報システム担当からの特別な指示がある場合などを除き、原則としてデフォルト設定を変えずに利用することを強く推奨する。

【基本原則】

- ベンダがサポートしているブラウザであって、更新プログラムを必ず適用し、最新状態にして利用する
- 自動更新を有効化しておく
- 企業の情報システム担当からの特別な指示がある場合などに限り、社内ポリシーに従う

8.2.2 設定項目

以下のブラウザは、プロトコルバージョンや暗号スイート、証明書処理などに関する設定変更を通常の設定機能では提供していない。

- PC（デスクトップ）版 Web ブラウザ
  - Google Chrome
  - Safari
    参考：https://support.apple.com/ja-jp/guide/safari/welcome/mac

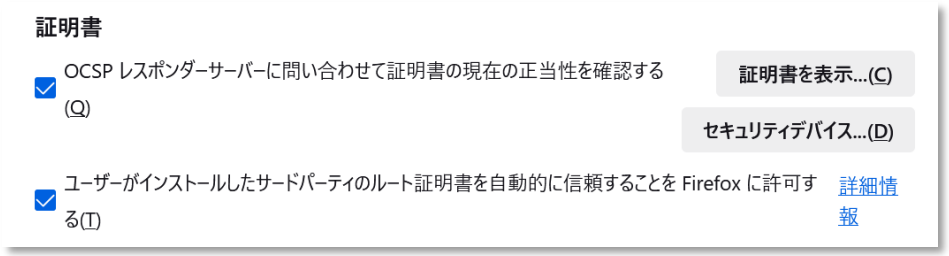
- スマートフォンに含まれる Web ブラウザ
  - Google Chrome
  - Mozilla Firefox

Microsoft Edge では、Internet Explorer mode 利用時のみ、Internet Explorer で利用可能な設定項目を標準機能で提供している。ただし、特別な指示がない限り、Internet Explorer モードの有効化をすべきではない。

Mozilla Firefox（デスクトップ版）では、サーバ証明書の検証・失効機能及びユーザがインストールしたルート証明書の信頼に関して、どのように処理するかの動作についてのみ設定方法を提供している。この設定については、

“メニュー”→“プライバシーとセキュリティ”

を選択した画面に設定方法がある。デフォルトの設定は以下のようになっており、特段の理由がない場合に変更することは推奨しない。



設定項目の強制管理機能

以下のブラウザでは、条件を満たす場合に、ユーザが利用するブラウザの設定変更オプションを強制的に管理者が設定できる。

- Microsoft Edge
  ローカル管理者権限、Active Dirctory グループポリシーの機能、Itune による構成管理のいずれかにより設定が可能

- Google Chrome
  エンタープライズ環境であれば、管理者は下記サイトのガイドに従って設定が可能
  https://support.google.com/chrome/a/answer/9710898
  https://cloud.google.com/blog/products/chrome-enterprise/new-benchmarks-securing-chrome-center-internet-security

- Mozilla Firefox（デスクトップ版のみ）
  エンタープライズ用途においては、管理者は以下の資料に基づいて Policy Engine を経由した設定が可能
  https://support.mozilla.org/products/firefox-enterprise/policies-customization-enterprise

8.3 ブラウザ利用時の注意点

2010 年代中頃以前のブラウザは、SSL/TLS を利用しない通信、すなわち URL が「http://」から始まるサイトへのアクセスが基本形になっており、SSL/TLS を利用する「https://」から始まるサイトにアクセスするとブラウザ上部のアドレスバーなどに「錠前」マークや「保護された通信」といった表示がされるようになっていた。この表示方法は、ブラウザの開発ベンダやバージョンの違いといったものに依存せず、すべてに共通したものであった。

しかし、この数年で常時 HTTPS 化の対応を行ったサイトが急増したことを受け、主要ブラウザ

は、SSL/TLS を利用する URL が「https://」から始まるサイトへのアクセスのほうを基本形とするようになっている。すなわち、以前のブラウザとは反対に、「http://」から始まるサイトにアクセスすると「保護されていない通信」や「安全ではありません」といった表示、あるいは「錠前マークに“／”が付いている」表示などの警告表示が出るようになった。

また、サーバ証明書の表示に関しても最近変化があり、アドレスバーが緑色になったり企業名や団体名を表示したりといった「EV 証明書」特有の表示を取りやめる傾向がある。そのような対応をした理由は、「EV 証明書」特有の表示を行っても行わなくても利用者の行動に変化はなく、期待した効果が得られていないという Chrome のセキュリティ UX チームの調査結果に基づく判断[44]としている。

EV 証明書特有の表示を取りやめたブラウザでは、どの種類のサーバ証明書を利用していたとしてもアドレスバーでの表示に変わりがなく、錠前マークをクリックして証明書の内容を表示するまで違いが判らない状態になった（7.2.5 節も参照されたい）[45]。

このような状況を踏まえ、本ガイドラインではアドレスバーでの表示方法についての説明を取りやめる。

代わりに、主要ブラウザについてアドレスバーの表示方法の違いや変更について、フィッシング対策協議会が適宜調査を実施し公表しているので、フィッシング対策協議会の情報を参照されたい。URL は以下の通り。

- 各ブラウザによる SSL/TLS サーバ証明書の表示の違い
  （フィッシング対策協議会 証明書普及促進 WG 成果物）
  https://member.antiphishing.jp/about_ap/wg.html#certificate

【参考】フィッシング対策協議会とは：

フィッシング詐欺に関する事例情報、技術情報の収集及び提供を中心に行うことで、日本国内におけるフィッシング詐欺被害の抑制を目的として、2005 年 4 月に発足した団体である。JPCERT/CC を事務局に、2023 年 11 月時点で 131 の会社や関連団体等が参加している。

https://www.antiphishing.jp/

[44] Google, EV UI Moving to Page Info,
https://chromium.googlesource.com/chromium/src/+/HEAD/docs/security/ev-to-page-info.md
[45] Chrome では「https://」から始まるサイトへのアクセス時にアドレスバーに表示するマークを、錠前から「調整アイコン」と呼ばれる設定項目のようなものに変更を始めている。これをクリックしてサイトの設定やセキュリティ状態が確認できることを明確に表すと共に、セキュリティ的にニュートラルなものと認識できる表現にしたものである。

【コラム④】 TLS ではフィッシングが防げない？－TLS で守られる限界を知ろう

「TLS は、通信の暗号化、データ完全性の確保、サーバ（場合によりクライアント）の認証を行うプロトコルであり、オンラインショッピングやネットバンキング等のサービスには不可欠である。」と説明されている。本ガイドラインでも、利用例として「金融サービスや電子商取引サービス、多様な個人情報の入力を必須とするサービス等を広範囲（不特定多数）に提供する場合」を挙げている。

ところが、一方で残念なデータもある。**「半数以上のフィッシングサイトが HTTPS を使っている」**という記事[46]が公開されている。2017 年以降のフィッシングサイトでの HTTPS 利用率の伸びは驚異的で、その後も増え続け、2016 年 4Q には 5%に過ぎなかったのに 2019 年 1Q に 58%、2020 年 3Q に 80%超にもなっている。フィッシングサイトがこのような対応を進めた理由は明らかで、このころから世界中で常時 HTTPS 化が始まるとともにブラウザのアドレスバー表示が変わったからに他ならない。HTTP のままでは「保護されていない通信」とか「安全ではない」と表示されるようになったので、例え本物そっくりのサイト画面を作ったとしてもすぐにフィッシングサイトだと見破られてしまうためである。

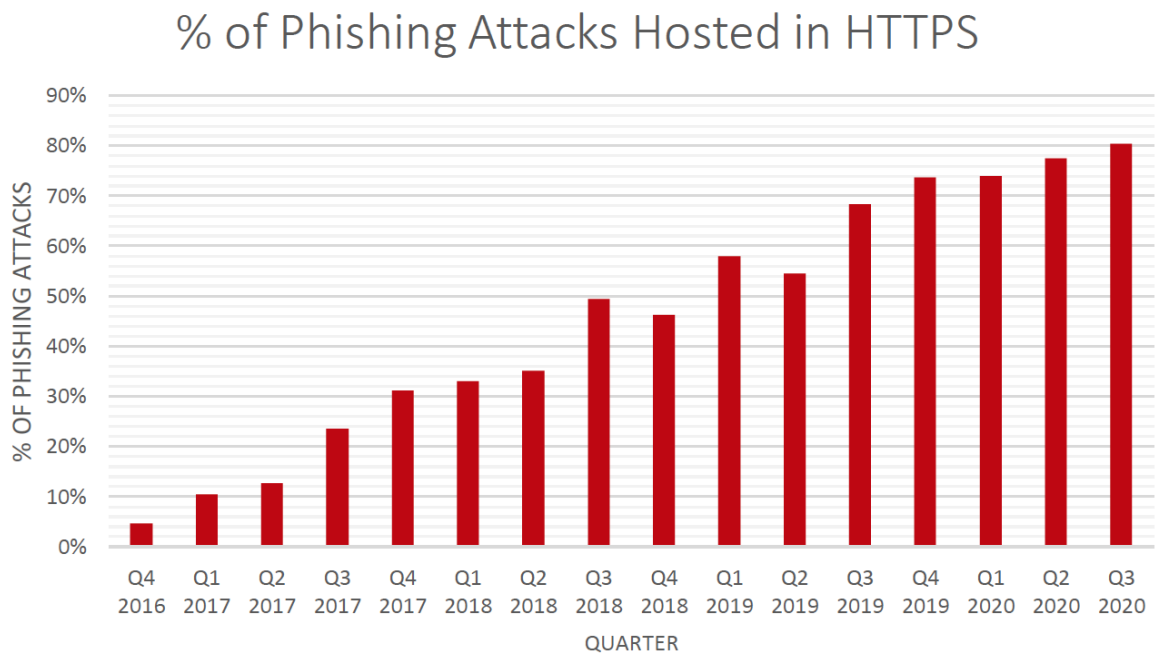


図 15 フィッシングサイトの HTTPS 利用率（[出典] The PhishLabs Blog[47]）

しかし、なぜこれほど HTTPS を使ったフィッシングサイトが作れるのかと疑問をもつ読者がいるかもしれない。そのポイントは「証明書によるサーバ認証」にかかるコストメリットが大きく変化したことにある。

一昔前にフィッシングサイトが HTTPS 化をしていなかった理由は、鍵マークを確認しない人が多く HTTP のままでもフィッシングサイトに誘導される可能性がそれなりにあったた

[46] https://info.phishlabs.com/blog/more-than-half-of-phishing-sites-use-https
[47]https://www.phishlabs.com/blog/apwg-q3-report-four-out-of-five-criminals-prefer-https

めである。つまり、コストをかけてサーバ証明書を購入してまでフィッシングサイトを HTTPS 化する必要性がなく、HTTPS 化することのコストメリットがほとんどなかったからにすぎない。

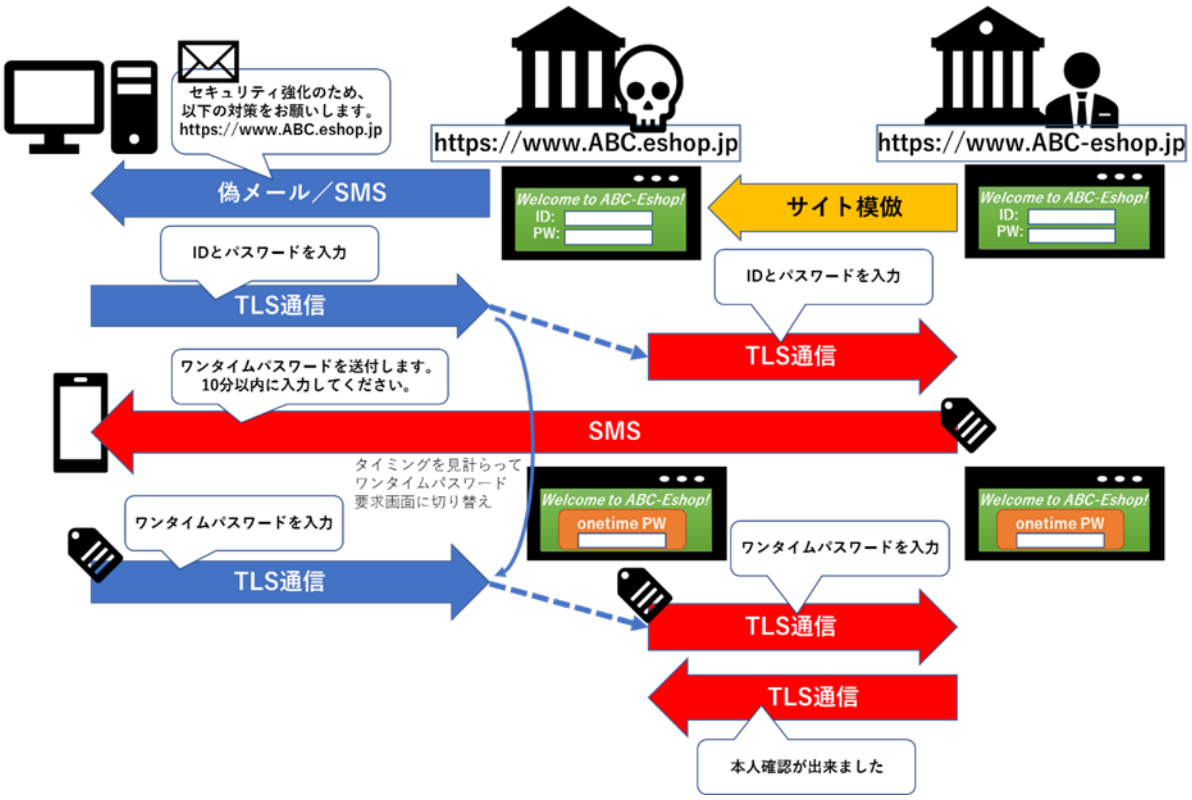


図 16　2 段階認証を突破するフィッシングの手法例

ところが、今やその状況は逆転した。HTTP のままではブラウザの警告表示によってフィッシングサイトへの誘導が難しくなった一方、常時 HTTPS 化の動きの中で Let’s Encrypt のような無料（あるいは相当安価なコスト）でサーバ証明書を準備できるようになった。しかも、心理的に「HTTPS なら安心」という警戒感が薄まる効果も期待できる。このため、攻撃者から見ると「フィッシングサイトを HTTPS 化することのコストメリットが一気に高まった」のである。

実際、日本サイバー犯罪対策センターから注意喚起[48]が出ているように、最近のフィッシングでは偽メールだけでなく SMS（ショートメッセージサービス）も多用されており、その中には書かれる URL が「https://」から始まるなど手口が巧妙になっているものもある。例えば、フィッシング対策協議会の「フィッシングに関するニュース[49]」の各種金融機関をかたるフィッシング情報、IPA 安心相談窓口が公表している「偽 SMS を使ったフィッシング[50]」

[48] 日本サイバー犯罪対策センター, https://www.jc3.or.jp/topics/banking/phishing.html
[49] フィッシング対策協議会, https://www.antiphishing.jp/news/alert/
[50] IPA 安心相談窓口だより、「宅配便業者をかたる偽ショートメッセージに引き続き注意！」, https://www.ipa.go.jp/security/anshin/mgdayori20200220.html

の事例が該当する。

技術的には、フィッシングサイト運営者は“フィッシングサイト”の“ドメインに対するサーバ証明書（DV 証明書)”を使う。DV 証明書では、ドメインが実在するかどうかしか検証せず、サーバの運営組織の実在性やドメイン名と運営組織の関係については確認しないので、（たとえフィッシングサイトであったとしても）ドメインが実在さえすれば DV 証明書の発行は可能である。また、ブラウザは、DV 証明書が有効であり、かつその証明書に記載のドメインがありさえれば TLS 接続を自動的に行う。

この時、ブラウザはそのサイトがフィッシングサイトであるかどうかの判断はしないため、利用者がサーバ証明書の中身を自ら確認して、接続したい本物のサイトのサーバ証明書でなければ接続しない（接続を閉じる）という行為をしない限り、TLS 接続されたからといってフィッシングを防ぐ手段にはならない。具体的には、偽メールや SMS に書かれているフィッシングサイト誘導用のリンクをそのまま使うと**フィッシングサイトが終端となる TLS 通信路が出来上がる**結果となる（図 16 の青矢印)。そうなると、**たとえ SMS を使った 2 段階認証を採用していたとしても、ワンタイムパスワードすらフィッシングサイト運営者に筒抜けになり、2 段階認証を突破**されるリスクがある（図 16 の赤矢印)。

このように、フィッシング対策としては「錠前マーク」や「https://になっているか」の確認だけではもはや不十分であり、「事前に正しいウェブサイトの URL をブックマークに登録してブックマークからアクセスする」等、新しい対策を取ることが必要になっていることに留意されたい。特に SMS は、便利である反面、送信元が偽装され本物の企業から届いている SMS のスレッドに送信元が偽装された SMS が紛れこむ事例が報道[51]されており、十分な注意が必要である。

また、もし利用者が本物の企業から SMS でリンク先を周知されるようなサービスを普段から日常的に受けていれば、偽装された SMS に記載されたフィッシングサイト誘導用のリンク先でもいつもと同じように疑わずに接続してしまう可能性が非常に高いと考えられる。このことから、サイト運営者のほうも、SMS を使ってリンク先を利用者に周知するといった方法を安易に使うべきではないといえるだろう。

[51] ケータイ Watch, 「送信元を偽装する SMS――注意すべき点とキャリアの対策は？」, https://k-tai.watch.impress.co.jp/docs/review/1220916.html

9. その他のトピック

9.1 リモートアクセス VPN over SSL （いわゆる SSL-VPN）

SSL-VPN と呼ばれるものは、正確には TLS を使った“リモートアクセス VPN”の実現方法といえる。SSL-VPN 装置を介して SSL-VPN 装置の奥にあるサーバ（インターネットからは直接アクセスできないサーバ）とクライアント端末をつなぐ形での VPN であり、IPsec-VPN のような特定端末間だけで VPN を構成する、いわゆる拠点間 VPN とは異なる。

したがって、あくまでリモートアクセスでの通信経路上が TLS で保護されているにすぎないと考え、本ガイドラインの推奨セキュリティ型（または高セキュリティ型）の設定を適用することとし、Appendix A.2（または Appendix A.3）のチェックリストを用いて確認すべきである。

なお、一口に SSL-VPN といっても、実現形態が製品によって全く異なることに注意がいる。実現形態としては、大きく以下の 3 通りに分かれる。

- 通常のブラウザを利用する“クライアントレス型”
- 接続時に自動的に Java や Active X をインストールすることでブラウザだけでなく、アプリケーションでも利用できるようにした“on-demand インストール型”
- 専用のクライアントソフト（通信アダプタなどを含む）をインストール・設定してから利用する“クライアント型”がある。

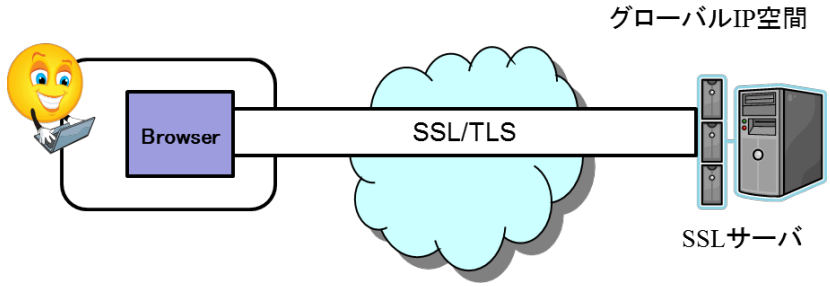
クライアントレス型は、ブラウザさえあればどの端末からでもアクセス可能であり、利便性に優れる一方、TLS との最大の差はグローバル IP をインターネットに公開しているか否か程度の違いといえる。結果として、最初のクライアント認証を TLS サーバが受け持つか、SSL-VPN 装置が受け持つか程度の差でしかなく、VPN というよりも、本質的には TLS と同じものとみるべきである。

On-demand インストール型も、接続時に自動的にインストールされることから、特に利用端末に制限を加えるものではなく、クライアントレス型と大きく異なるわけではない。むしろ、ブラウザでしか使えなかったクライアントレス型を、他のアプリケーションでも利用できるように拡張したという位置づけのものである。

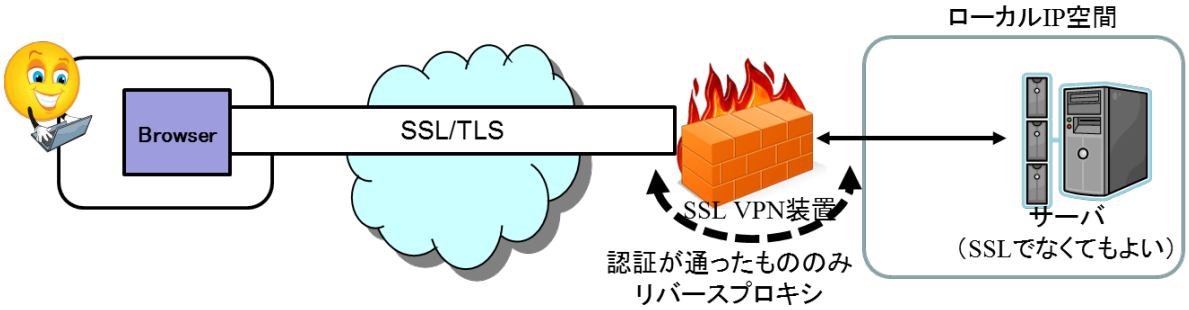
一方、クライアント型は上記の 2 つのタイプとは明らかに異なり、専用のクライアントソフトがインストールされた端末との間でのみアクセスする。つまり、誤って偽サーバに接続することがなく、また内部サーバにアクセスできる端末も厳格に制限できるため、端末に IPsec-VPN ソフトをインストールして構成するモバイル型の IPsec-VPN に近い形での運用形態となる。

機密度の高い情報を扱うのだとすれば、少なくともクライアント型での SSL-VPN を利用すべきである。

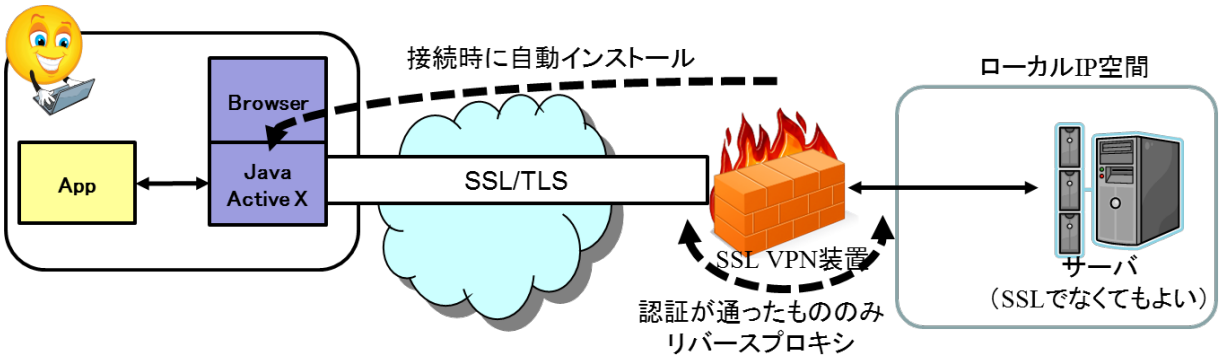
【参考：通常の TLS】



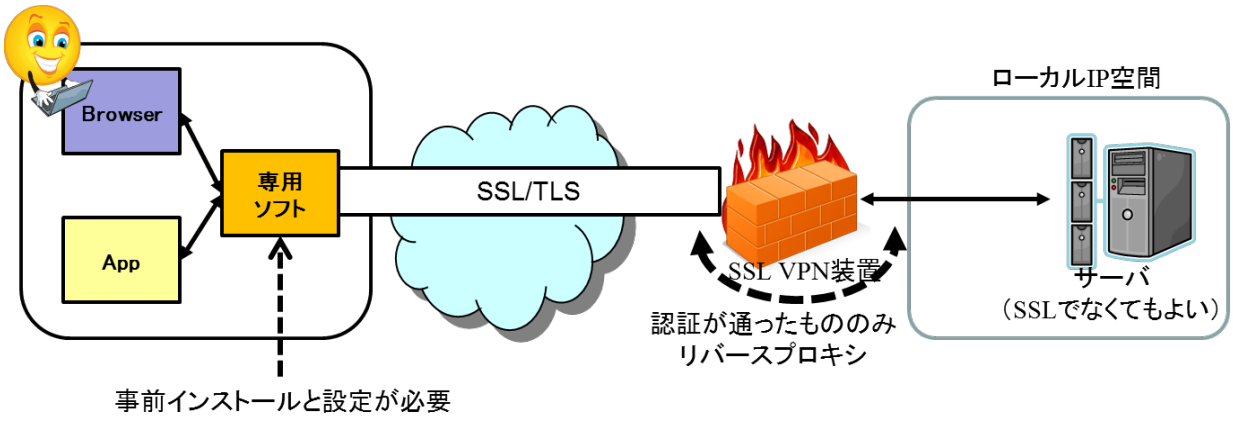
【クライアントレス型（ブラウザベース）】



【On-demand インストール型（Java や Active X を使ってブラウザ以外でも利用可能）】



【クライアント型（専用ソフトベース）】



【コラム⑤】 ローカルネットワークでの HTTPS 通信問題

IoT の普及が進むとともに、ローカルネットワークに接続されたデバイス（プリンタ、テレビ等）に対して、同じローカルネットワーク上の（例えばスマホなどの）Web ブラウザからアクセスするユースケースが増えてきた。一方、8.3 節に示すように Web ブラウザのなかには HTTP 通信に警告が表示されるものも出てきているため、こうしたローカルネットワーク内の通信においても HTTPS 通信が必要となってくる。

しかし、パブリック認証局では CA/Browser Forum の規準により、パブリック DNS で一意性を確認できないサーバ名、いわゆる Internal Name を含む証明書を発行することが原則として禁じられており、前述のデバイスにおける HTTPS 通信は実現が難しい状況にある。

このような問題を解決するため、W3C（World Wide Web Consortium）の httpslocal CG（HTTPS in Local Network Community Group）において、ユースケースが整理され、これを踏まえてローカルネットワーク上の HTTPS 通信としてサーバ証明書を用いない通信方式、及びサーバ証明書を用いる通信方式の両面で検討された。これらの課題について、現時点では利用環境等に応じて個別に適切な技術を組み合わせて解決する必要がある。一方で、それらの方法の中には環境に例外的な設定を設けるものもあり、その場合は他のプロセスへの影響や安全性への影響に配慮して設定を行う必要があることには留意されたい。

なお、2023 年 11 月現在、WICG（Web Platform Incubator Community Group）にて、Private Network Access について整理するドキュメントが提案されている。ローカルネットワークでの通信を検討する場合は、WICG のドキュメントも参照されたい。

Appendix：

付録

Appendix A：チェックリスト

チェックリストの原本は以下の URL からも入手可能である。

[PDF 版]　https://www.ipa.go.jp/security/ipg/documents/ipa-cryptrec-gl-3001-3.1-checklists.pdf

[Excel 版] https://www.ipa.go.jp/security/ipg/documents/ipa-cryptrec-gl-3001-3.1-checklists.xlsx

A.1.　チェックリストの利用方法

【チェックリストの使い方】
本チェックリストは、選択した設定基準に対応した要求設定を実施したことを確認するためのチェックリストである
○ 選択した設定基準に応じたチェックリストのチェックシートを下部の「タグ」から選択する
○ 当該チェックシートに記載のチェック項目全てについて参照章の記載を参考に設定内容を確認する
○ 「要求設定確認」は選択したチェックシートを利用してよいかの確認項目であり、「該当」にチェックが入る場合に限り、当該チェックシートを利用してよい
○ 「遵守項目」については要求設定に合致していることを確認して「済」にチェックが入ることが必要である
○ 「遵守項目」以外については記載内容を確認し、設定の実態に即して適切なほうのチェックボックスにチェックを入れる

＜チェックリストの例＞

【高セキュリティ型チェックリスト】

選択した設定基準に対応したチェックリストのチェックシートを用いる

2024.04.01版

要求設定が満たされていることを確認したら「済」にチェックを入れる

要求設定の詳細な内容が記載されている章番号

選択したチェックシートを利用してよいかの確認項目であり、「該当」にチェックが入る場合に限り利用してよい

「遵守項目」は要求設定に合致していることを確認して「済」にチェックが入ることが必要である

「遵守項目」以外については記載内容を確認し、設定の実態に即して適切なほうのチェックボックスにチェックを入れる

チェック項目		参照章		
①要求設定確認	①-1)【遵守項目】高セキュリティ型の設定 境であるか	3.1節	□該当	
②プロトコルバージョン設定	[redacted]	5.1節	□済	
	[redacted]ただし、TLS1.2を明確に利用 ックする	5.1節	□済	□設定せず
	[redacted]無効（利用不可）にしたか	5.1節	□済	
③サーバ証明書設定	③-1)【遵守項目】サーバの公開鍵情報（Subject Public Key Info）の Subject Public Ke Algorithmと鍵長の組合せが以下のいずれかを満たしているか ・ RSAで112ビット[redacted]以上を満たす鍵長（2048ビット以上）。可能であれば128[redacted]上） ・ 楕円曲線[redacted]56など）	5.2節	□済	
	③-2)【遵[redacted]nature Algorithm）と鍵長の組合せが以下のいずれかを満たしているか ・ RSA署名とSHA-256以上のハッシュ関数の組合せで112ビットセキュリティ以上を満たす鍵長（2048ビット以上）。可能であれば、128ビットセキュリティ以上を満たす鍵長（3072ビット以上） ・ ECDSAとSHA-256以上のハッシュ関数の組合せで128ビットセキュリティ以上を満たす曲線　(P-256など) ・ RSA-PSSとSHA-256以上のハッシュ関数の組合せで112ビットセキュリティ以上を満たす鍵長（2048ビット以上）。可能であれば、128ビットセキュリティ以上を満たす鍵長（3072ビット以上）	5.2節	□済	
	③-3)【遵守項目】サーバ証明書の発行・更新を行う際に、自ら公開鍵と秘密鍵の鍵ペアを生成する場合には、新たな公開鍵と秘密鍵の鍵ペアを生成しているか	5.2節	□済	
	③-4)【遵守項目】上記③-3)についての指示を仕様書や運用手順書等に明記したか	5.2節	□済	
	③-5)【遵守項目】 接続することが想定されている全てのクライアントに対して、警告表示が出ないように対策するか、警告表示が出るブラウザはサポート対象外であることを明示したか	5.2節	□済	
④暗号スイート設定	④-1)【遵守項目】表21記載の暗号アルゴリズムを全てを設定無効（利用不可）にしたか	5.3節	□済	
	④-2) ECDHEを利用する暗号スイートを設定するか。 設定しない場合は「設定せず」をチェックす		□設定する	□設定せず
	④-2-1)【遵守項目】ECDHEを128ビット[redacted]（P-256やCurve25519など）			
	④-3) DHEを利用する暗号スイートを設定す 設定しない場合は「設定せず」をチェックする（④-3-1のチェック不要）		[redacted]定する	□設定せず
	④-3-1)【遵守項目】DHEを128ビットセキュリティ以上の鍵長（3072ビット以上）に設定したか	[redacted]	□済	
	④-4)【推奨項目】表22記載の暗号アルゴリズムを組み合わせた暗号スイートのみで設定しているか。 設定しない／できない場合は「設定せず」をチェックする	5.3節	□済	□設定せず
	④-5) 暗号スイートの優先順位が設定できるか。 設定できない場合は「設定不可」をチェックする（④-5-1のチェック不要）		□設定可	□設定不可
	④-5-1)【推奨項目】表24記載の暗号スイートの優先順位で設定したか。優先順位どおりに設定できない／しない場合は「設定せず」をチェックする	5.3節	□済	□設定せず
⑤附録	⑤) Appendix C：暗号スイートの設定例の最新版のドキュメントを参照して設定したか。参照していない場合には「参照せず」をチェックする	Appendix C	□参照済	□参照せず

A.2.　推奨セキュリティ型のチェックリスト

【推奨セキュリティ型チェックリスト】　　2024.04.01版

チェック項目		参照章		
①要求設定確認	チェック項目なし			
②プロトコルバージョン設定	②-1)【遵守項目】TLS1.2を設定有効としたか	4.1節	□済	
	②-2)【遵守項目】SSL2.0からTLS1.1までを設定無効（利用不可）にしたか	4.1節	□済	
	②-3) TLS1.3が実装されているか。 実装されていない場合は「未実装」をチェックする（②-3-1のチェック不要）		□実装済	□未実装
	②-3-1)【推奨項目】TLS1.3について設定を有効にしたか。ただし、TLS1.3を明確に利用しない場合は「設定せず」をチェックする	4.1節	□済	□設定せず
③サーバ証明書設定	③-1)【遵守項目】サーバの公開鍵情報（Subject Public Key Info）のSubject Public Key Algorithmと鍵長の組合せが以下のいずれかを満たしているか ・ RSAで112ビットセキュリティ以上を満たす鍵長（2048ビット以上） ・ 楕円曲線暗号で128ビットセキュリティ以上を満たす曲線（P-256など）	4.2節	□済	
	③-2)【遵守項目】認証局の署名アルゴリズム（Certificate Signature Algorithm）と鍵長の組合せが以下のいずれかを満たしているか ・ RSA署名とSHA-256の組合せで112ビットセキュリティ以上を満たす鍵長（2048ビット以上） ・ ECDSAとSHA-256の組合せで128ビットセキュリティ以上を満たす曲線（P-256など） ・ RSA-PSSとSHA-256の組合せで112ビットセキュリティ以上を満たす鍵長（2048ビット以上）	4.2節	□済	
	③-3)【遵守項目】サーバ証明書の発行・更新を行う際に、自ら公開鍵と秘密鍵の鍵ペアを生成する場合には、新たな公開鍵と秘密鍵の鍵ペアを生成しているか	4.2節	□済	
	③-4)【遵守項目】上記③-3)についての指示を仕様書や運用手順書等に明記したか	4.2節	□済	
	③-5)【遵守項目】 接続することが想定されている全てのクライアントに対して、警告表示が出ないように対策するか、警告表示が出るブラウザはサポート対象外であることを明示したか	4.2節	□済	
④暗号スイート設定	④-1)【遵守項目】表15記載の暗号アルゴリズムを全てを設定無効（利用不可）にしたか	4.3節	□済	
	④-2) ECDHEを利用する暗号スイートを設定するか。 設定しない場合は「設定せず」をチェックする（④-2-1のチェック不要）		□設定する	□設定せず
	④-2-1)【遵守項目】ECDHEを128ビットセキュリティ以上の曲線にしたか（P-256やCurve25519など）	4.3節	□済	
	④-3) DHEを利用する暗号スイートを設定するか。 設定しない場合は「設定せず」をチェックする（④-3-1のチェック不要）		□設定する	□設定せず
	④-3-1)【遵守項目】DHEを112ビットセキュリティ以上の鍵長（2048ビット以上）に設定したか	4.3節	□済	
	④-4)【推奨項目】表16記載の暗号アルゴリズムを組み合わせた暗号スイートのみで設定しているか。 設定しない／できない場合は「設定せず」をチェックする	4.3節	□済	□設定せず
	④-5) 暗号スイートの優先順位が設定できるか。 設定できない場合は「設定不可」をチェックする（④-5-1のチェック不要）		□設定可	□設定不可
	④-5-1)【推奨項目】表18記載の暗号スイートの優先順位で設定したか。優先順位どおりに設定できない／しない場合は「設定せず」をチェックする	4.3節	□済	□設定せず
⑤附録	⑤) Appendix C：暗号スイートの設定例の最新版のドキュメントを参照して設定したか。参照していない場合には「参照せず」をチェックする	Appendix C	□参照済	□参照せず

【表18】（TLS暗号設定ガイドライン 4.3節）
※下記の暗号アルゴリズムのいずれかを含む暗号スイートは
「全種類」設定無効化（利用不可）とすること

【遵守項目】利用禁止暗号アルゴリズム一覧（2024年4月1日時点）		
鍵交換		DH
		ECDH
署名		GOST R 34.10-2012
		SM2（署名）
暗号化	ブロック暗号	RC2
		EXPORT-RC2
		IDEA
		DES
		EXPORT-DES
		GOST 28147-89
		Magma
		3-key Triple DES
		Kuznyechik
		SM4
		ARIA
		SEED
	暗号利用モード	CTR_OMAC
	ストリーム暗号	RC4
		EXPORT-RC4
ハッシュ関数		MD5
		GOST R 34.11-2012
		SM3

【表19】（TLS暗号設定ガイドライン 4.3節）
※下記の暗号アルゴリズムだけを組み合わせた暗号スイートのみで
設定（利用可）されている

【推奨項目】利用推奨暗号アルゴリズム一覧		
鍵交換		ECDHE
		DHE
署名		ECDSA
		RSASSA-PKCS1-v1_5
		RSASSA-PSS（TLS1.3のみ）
暗号化	ブロック暗号	AES
		Camellia（TLS1.2のみ）
	暗号利用モード	GCM
		CCM
		CCM_8
		CBC
	ストリーム暗号	ChaCha20-Poly1305
ハッシュ関数		SHA-256
		SHA-384
		SHA-1

【表21】（TLS暗号設定ガイドライン 4.3節）

【推奨項目】TLS1.2を利用する場合の優先順位		
優先順位グループ	暗号スイート名	スイート番号
グループA	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	(0xC0,0x2B)
	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	(0xC0,0x2F)
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x86)
	TLS_ECDHE_RSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x8A)
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	(0xC0,0xAC)
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8	(0xC0,0xAE)
	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	(0xC0,0x2C)
	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	(0xC0,0x30)
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x87)
	TLS_ECDHE_RSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x8B)
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	(0xC0,0xAD)
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM_8	(0xC0,0xAF)
	TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256	(0xCC,0xA9)
	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	(0xCC,0xA8)
グループB	TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	(0x00,0x9E)
	TLS_DHE_RSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x7C)
	TLS_DHE_RSA_WITH_AES_128_CCM	(0xC0,0x9E)
	TLS_DHE_RSA_WITH_AES_128_CCM_8	(0xC0,0xA2)
	TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	(0x00,0x9F)
	TLS_DHE_RSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x7D)
	TLS_DHE_RSA_WITH_AES_256_CCM	(0xC0,0x9F)
	TLS_DHE_RSA_WITH_AES_256_CCM_8	(0xC0,0xA3)
	TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256	(0xCC,0xAA)
グループC	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	(0xC0,0x23)
	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	(0xC0,0x27)
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_CBC_SHA256	(0xC0,0x72)
	TLS_ECDHE_RSA_WITH_CAMELLIA_128_CBC_SHA256	(0xC0,0x76)
	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	(0xC0,0x24)
	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	(0xC0,0x28)
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_CBC_SHA384	(0xC0,0x73)
	TLS_ECDHE_RSA_WITH_CAMELLIA_256_CBC_SHA384	(0xC0,0x77)
グループD	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA	(0xC0,0x09)
	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	(0xC0,0x13)
	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA	(0xC0,0x0A)
	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA	(0xC0,0x14)
グループE	TLS_DHE_RSA_WITH_AES_128_CBC_SHA256	(0x00,0x67)
	TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA256	(0x00,0xBE)
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA256	(0x00,0x6B)
	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA256	(0x00,0xC4)
グループF	TLS_DHE_RSA_WITH_AES_128_CBC_SHA	(0x00,0x33)
	TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA	(0x00,0x45)
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA	(0x00,0x39)
	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA	(0x00,0x88)

【推奨項目】TLS1.3を利用する場合の優先順位		
優先順位グループ	暗号スイート名	スイート番号
グループA	TLS_AES_128_GCM_SHA256	(0x13,0x01)
	TLS_AES_128_CCM_SHA256	(0x13,0x04)
	TLS_AES_128_CCM_8_SHA256	(0x13,0x05)
	TLS_AES_256_GCM_SHA384	(0x13,0x02)
	TLS_CHACHA20_POLY1305_SHA256	(0x13,0x03)
鍵交換	ECDHE（優先）	RFC8446に規定
	DHE	RFC8446に規定
署名	ECDSA	RFC8446に規定
	RSA-PSS	RFC8446に規定
	RSASSA-PKCS1-v1_5	RFC8446に規定

A.3. 高セキュリティ型のチェックリスト

【高セキュリティ型チェックリスト】　　2024.04.01版

チェック項目		参照章		
①要求設定確認	①-1)【遵守項目】高セキュリティ型の設定基準を満たすことが必要な利用環境であるか	3.1節	□該当	
②プロトコルバージョン設定	②-1)【遵守項目】TLS1.3を設定有効としたか	5.1節	□済	
	②-2)【遵守項目】TLS1.2を設定有効としたか。ただし、TLS1.2を明確に利用しないと判明している場合は「設定せず」をチェックする	5.1節	□済	□設定せず
	②-3)【遵守項目】SSL2.0からTLS1.1までを設定無効（利用不可）にしたか	5.1節	□済	
③サーバ証明書設定	③-1)【遵守項目】サーバの公開鍵情報（Subject Public Key Info）のSubject Public Key Algorithmと鍵長の組合せが以下のいずれかを満たしているか ・ RSAで112ビットセキュリティ以上を満たす鍵長（2048ビット以上）。可能であれば128ビットセキュリティ以上を満たす鍵長（3072ビット以上） ・ 楕円曲線暗号で128ビットセキュリティ以上を満たす曲線（P-256など）	5.2節	□済	
	③-2)【遵守項目】認証局の署名アルゴリズム（Certificate Signature Algorithm）と鍵長の組合せが以下のいずれかを満たしているか ・ RSA署名とSHA-256以上のハッシュ関数の組合せで112ビットセキュリティ以上を満たす鍵長（2048ビット以上）。可能であれば、128ビットセキュリティ以上を満たす鍵長（3072ビット以上） ・ ECDSAとSHA-256以上のハッシュ関数の組合せで128ビットセキュリティ以上を満たす曲線（P-256など） ・ RSA-PSSとSHA-256以上のハッシュ関数の組合せで112ビットセキュリティ以上を満たす鍵長（2048ビット以上）。可能であれば、128ビットセキュリティ以上を満たす鍵長（3072ビット以上）	5.2節	□済	
	③-3)【遵守項目】サーバ証明書の発行・更新を行う際に、自ら公開鍵と秘密鍵の鍵ペアを生成する場合には、新たな公開鍵と秘密鍵の鍵ペアを生成しているか	5.2節	□済	
	③-4)【遵守項目】上記③-3)についての指示を仕様書や運用手順書等に明記したか	5.2節	□済	
	③-5)【遵守項目】 接続することが想定されている全てのクライアントに対して、警告表示が出ないように対策するか、警告表示が出るブラウザはサポート対象外であることを明示したか	5.2節	□済	
④暗号スイート設定	④-1)【遵守項目】表21記載の暗号アルゴリズムを全てを設定無効（利用不可）にしたか	5.3節	□済	
	④-2) ECDHEを利用する暗号スイートを設定するか。 設定しない場合は「設定せず」をチェックする（④-2-1のチェック不要）		□設定する	□設定せず
	④-2-1)【遵守項目】ECDHEを128ビットセキュリティ以上の曲線にしたか（P-256やCurve25519など）	5.3節	□済	
	④-3) DHEを利用する暗号スイートを設定するか。 設定しない場合は「設定せず」をチェックする（④-3-1のチェック不要）		□設定する	□設定せず
	④-3-1)【遵守項目】DHEを128ビットセキュリティ以上の鍵長（3072ビット以上）に設定したか	5.3節	□済	
	④-4)【推奨項目】表22記載の暗号アルゴリズムを組み合わせた暗号スイートのみで設定しているか。 設定しない／できない場合は「設定せず」をチェックする	5.3節	□済	□設定せず
	④-5) 暗号スイートの優先順位が設定できるか。 設定できない場合は「設定不可」をチェックする（④-5-1のチェック不要）		□設定可	□設定不可
	④-5-1) 【推奨項目】表24記載の暗号スイートの優先順位で設定したか。 優先順位どおりに設定できない／しない場合は「設定せず」をチェックする	5.3節	□済	□設定せず
⑤附録	⑤) Appendix C：暗号スイートの設定例の最新版のドキュメントを参照して設定したか。参照していない場合には「参照せず」をチェックする	Appendix C	□参照済	□参照せず

A.4.

【表24】（TLS暗号設定ガイドライン 5.3節）
※下記の暗号アルゴリズムのいずれかを含む暗号スイートは
「全種類」設定無効化（利用不可）とすること

【遵守項目】利用禁止暗号アルゴリズム一覧（2024年4月1日時点）		
鍵交換		DH
		ECDH
		RSAES-PKCS1-v1_5
署名		GOST R 34.10-2012
		SM2（署名）
暗号化	ブロック暗号	RC2
		EXPORT-RC2
		IDEA
		DES
		EXPORT-DES
		GOST 28147-89
		Magma
		3-key Triple DES
		Kuznyechik
		SM4
		ARIA
		SEED
	暗号利用モード	CBC
		CTR_OMAC
	ストリーム暗号	RC4
		EXPORT-RC4
ハッシュ関数		MD5
		SHA-1
		GOST R 34.11-2012
		SM3

【表25】（TLS暗号設定ガイドライン 5.3節）
※下記の暗号アルゴリズムだけを組み合わせた暗号スイートのみで
設定（利用可）されている

【推奨項目】利用推奨暗号アルゴリズム一覧		
鍵交換		ECDHE
		DHE
署名		ECDSA
		RSASSA-PKCS1-v1_5
		RSASSA-PSS（TLS1.3のみ）
暗号化	ブロック暗号	AES
		Camellia（TLS1.2のみ）
	暗号利用モード	GCM
		CCM
		CCM_8
	ストリーム暗号	ChaCha20-Poly1305
ハッシュ関数		SHA-256
		SHA-384

【表27】（TLS暗号設定ガイドライン 5.3節）

【推奨項目】TLS1.3を利用する場合の優先順位		
優先順位グループ	暗号スイート名	スイート番号
グループA	TLS_AES_256_GCM_SHA384	(0x13,0x02)
	TLS_CHACHA20_POLY1305_SHA256	(0x13,0x03)
	TLS_AES_128_GCM_SHA256	(0x13,0x01)
	TLS_AES_128_CCM_SHA256	(0x13,0x04)
	TLS_AES_128_CCM_8_SHA256	(0x13,0x05)
鍵交換	ECDHE（優先）	RFC8446に規定
	DHE	RFC8446に規定
署名	ECDSA	RFC8446に規定
	RSA-PSS	RFC8446に規定
	RSASSA-PKCS1-v1_5	RFC8446に規定

【推奨項目】TLS1.2を利用する場合の優先順位		
優先順位グループ	暗号スイート名	スイート番号
グループA	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	(0xC0,0x2C)
	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	(0xC0,0x30)
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x87)
	TLS_ECDHE_RSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x8B)
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	(0xC0,0xAD)
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM_8	(0xC0,0xAF)
	TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256	(0xCC,0xA9)
	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	(0xCC,0xA8)
	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	(0xC0,0x2B)
	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	(0xC0,0x2F)
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x86)
	TLS_ECDHE_RSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x8A)
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	(0xC0,0xAC)
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8	(0xC0,0xAE)
グループB	TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	(0x00,0x9F)
	TLS_DHE_RSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x7D)
	TLS_DHE_RSA_WITH_AES_256_CCM	(0xC0,0x9F)
	TLS_DHE_RSA_WITH_AES_256_CCM_8	(0xC0,0xA3)
	TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256	(0xCC,0xAA)
	TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	(0x00,0x9E)
	TLS_DHE_RSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x7C)
	TLS_DHE_RSA_WITH_AES_128_CCM	(0xC0,0x9E)
	TLS_DHE_RSA_WITH_AES_128_CCM_8	(0xC0,0xA2)

セキュリティ例外型のチェックリスト

【セキュリティ例外型チェックリスト】　2024.04.01版

	チェック項目	参照章		
①要求設定確認	①-1) 【遵守項目】推奨セキュリティ型以上の設定が現実的ではない等の特殊事情があるケースに該当するか	3.1節	□該当	
	①-2) 【遵守項目】推奨セキュリティ型への移行完了までの短期暫定運用を前提とし、早期の利用終了期限を含む移行計画を策定するなど、今後の対処方針を具体的に策定しているか	3.1節	□済	
②プロトコルバージョン設定	②-1)【遵守項目】SSL3.0及びSSL2.0を設定無効（利用不可）にしたか	6.1節	□済	
	②-2) TLS1.2が実装されているか。 実装されていない場合は「未実装」をチェックする（②-2-1のチェック不要）		□実装済	□未実装
	②-2-1)【遵守項目】TLS1.2について設定を有効にしたか	6.1節	□済	
	②-3) TLS1.1とTLS1.0のいずれか、または両方の設定を有効にするか。 両方とも有効にしない場合は「設定せず」をチェックする（②-3-1のチェック不要）		□設定する	□設定せず
	②-3-1)【遵守項目】TLS1.1とTLS1.0のいずれか、または両方を設定有効とする必要性を確認したか	6.1節	□済	
	②-4) TLS1.3が実装されているか。 実装されていない場合は未実装にチェックする（②-4-1のチェック不要）		□実装済	□未実装
	②-4-1)【推奨項目】TLS1.3について設定を有効にしたか。 ただし、TLS1.3を明確に利用しないと判断している場合には「設定せず」をチェックする	6.1節	□済	□設定せず
	②-5) プロトコルバージョンの優先順位が設定できるか。 設定できない場合は「設定不可」にチェックする（②-5-1のチェック不要）		□設定可	□設定不可
	②-5-1)【推奨項目】最も新しいバージョンによる接続を最優先とし、接続できない場合に順番に一つずつ前のバージョンで接続するように設定したか。設定しない場合は「設定せず」をチェックする	6.1節	□済	□設定せず
③サーバ証明書設定	③-1)【遵守項目】サーバの公開鍵情報（Subject Public Key Info）のSubject Public Key Algorithmと鍵長の組合せが以下のいずれかを満たしているか ・ RSAで112ビットセキュリティ以上を満たす鍵長（2048ビット以上）	6.2節	□済	
	③-2) 【遵守項目】認証局の署名アルゴリズム（Certificate Signature Algorithm）と鍵長の組合せが以下のいずれかを満たしているか ・ RSA署名とSHA-256の組合せで112ビットセキュリティ以上を満たす鍵長（2048ビット以上）	6.2節	□済	
	③-3)【遵守項目】サーバ証明書の発行・更新を行う際に、自ら公開鍵と秘密鍵の鍵ペアを生成する場合には、新たな公開鍵と秘密鍵の鍵ペアを生成しているか	6.2節	□済	
	③-4)【遵守項目】上記③-3)についての指示を仕様書や運用手順書等に明記したか	6.2節	□済	
	③-5)【遵守項目】 接続することが想定されている全てのクライアントに対して、警告表示が出ないように対策するか、警告表示が出るブラウザはサポート対象外であることを明示したか	6.2節	□済	
④暗号スイート設定	④-1)【遵守項目】表27記載の暗号アルゴリズムを全てを設定無効（利用不可）にしたか	6.3節	□済	
	④-2) ECDHE／ECDHを利用する暗号スイートを設定するか。 設定しない場合は「設定せず」をチェックする（④-2-1のチェック不要）		□設定する	□設定せず
	④-2-1)【遵守項目】ECDHE/ECDHを128ビットセキュリティ以上の曲線にしたか（P-256やCurve25519など）	6.3節	□済	
	④-3) DHE/DHを利用する暗号スイートを設定するか。 設定しない場合は「設定せず」をチェックする（④-3-1のチェック不要）		□設定する	□設定せず
	④-3-1)【遵守項目】DHE/DHを1024ビット以上の鍵長に設定したか	6.3節	□済	
	④-4)【推奨項目】表28記載の暗号アルゴリズムを組み合わせた暗号スイートのみで設定しているか。 設定しない／できない場合には「設定せず」をチェックする	6.3節	□済	□設定せず
	④-5) 暗号スイートの優先順位が設定できるか。 設定できない場合は「設定不可」をチェックする（④-5-1のチェック不要）		□設定可	□設定不可
	④-5-1) 【推奨項目】表30記載の暗号スイートの優先順位で設定したか。優先順位どおりに設定できない／しない場合には「設定せず」をチェックする	6.3節	□済	□設定せず
⑤附録	⑤) Appendix C：暗号スイートの設定例の最新版のドキュメントを参照して設定したか。参照していない場合には「参照せず」をチェックする	Appendix C	□参照済	□参照せず

【表30】（TLS暗号設定ガイドライン 6.3節）
※下記の暗号アルゴリズムのいずれかを含む暗号スイートは
「全種類」設定無効化（利用不可）とすること

【遵守項目】利用禁止暗号アルゴリズム一覧（2024年4月1日時点）		
署名		GOST R 34.10-2012
		SM2（署名）
暗号化	ブロック暗号	RC2
		EXPORT-RC2
		IDEA
		DES
		EXPORT-DES
		GOST 28147-89
		Magma
		3-key Triple DES
		Kuznyechik
		SM4
		ARIA
		SEED
	暗号利用モード	CTR_OMAC
	ストリーム暗号	RC4
		EXPORT-RC4
ハッシュ関数		MD5
		GOST R 34.11-2012
		SM3

【表31】（TLS暗号設定ガイドライン 6.3節）
※下記の暗号アルゴリズムだけを組み合わせた暗号スイートのみで
設定（利用可）されている

【推奨項目】利用推奨暗号アルゴリズム一覧		
鍵交換		DHE
		ECDHE
		RSAES-PKCS1-v1_5
		DH
		ECDH
署名		RSASSA-PKCS1-v1_5
		RSASSA-PSS（TLS1.3のみ）
		ECDSA
暗号化	ブロック暗号	AES
		Camellia（TLS1.2まで）
	暗号利用モード	GCM
		CCM
		CCM_8
		CBC
	ストリーム暗号	ChaCha20-Poly1305
ハッシュ関数		SHA-256
		SHA-384
		SHA-1

【表33】（TLS暗号設定ガイドライン 6.3節）

【推奨項目】TLS1.2を利用する場合の優先順位		
優先順位グループ	暗号スイート名	スイート番号
グループX	TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	(0x00,0x9E)
	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	(0xC0,0x2B)
	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	(0xC0,0x2F)
	TLS_DHE_RSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x7C)
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x86)
	TLS_ECDHE_RSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x8A)
	TLS_DHE_RSA_WITH_AES_128_CCM	(0xC0,0x9E)
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM	(0xC0,0xAC)
	TLS_DHE_RSA_WITH_AES_128_CCM_8	(0xC0,0xA2)
	TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8	(0xC0,0xAE)
	TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	(0x00,0x9F)
	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	(0xC0,0x2C)
	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	(0xC0,0x30)
	TLS_DHE_RSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x7D)
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x87)
	TLS_ECDHE_RSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x8B)
	TLS_DHE_RSA_WITH_AES_256_CCM	(0xC0,0x9F)
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM	(0xC0,0xAD)
	TLS_DHE_RSA_WITH_AES_256_CCM_8	(0xC0,0xA3)
	TLS_ECDHE_ECDSA_WITH_AES_256_CCM_8	(0xC0,0xAF)
	TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256	(0xCC,0xAA)
	TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256	(0xCC,0xA9)
	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	(0xCC,0xA8)
グループY	TLS_RSA_WITH_AES_128_GCM_SHA256	(0x00,0x9C)
	TLS_DH_RSA_WITH_AES_128_GCM_SHA256	(0x00,0xA0)
	TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256	(0xC0,0x2D)
	TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256	(0xC0,0x31)
	TLS_RSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x7A)
	TLS_DH_RSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x7E)
	TLS_ECDH_ECDSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x88)
	TLS_ECDH_RSA_WITH_CAMELLIA_128_GCM_SHA256	(0xC0,0x8C)
	TLS_RSA_WITH_AES_128_CCM	(0xC0,0x9C)
	TLS_RSA_WITH_AES_128_CCM_8	(0xC0,0xA0)
	TLS_RSA_WITH_AES_256_GCM_SHA384	(0x00,0x9D)
	TLS_DH_RSA_WITH_AES_256_GCM_SHA384	(0x00,0xA1)
	TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384	(0xC0,0x2E)
	TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384	(0xC0,0x32)
	TLS_RSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x7B)
	TLS_DH_RSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x7F)
	TLS_ECDH_ECDSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x89)
	TLS_ECDH_RSA_WITH_CAMELLIA_256_GCM_SHA384	(0xC0,0x8D)
	TLS_RSA_WITH_AES_256_CCM	(0xC0,0x9D)
	TLS_RSA_WITH_AES_256_CCM_8	(0xC0,0xA1)

（グループZに続く）

【表33（続）】（TLS暗号設定ガイドライン 6.3節）

【推奨項目】TLS1.2を利用する場合の優先順位（続）		
優先順位グループ	暗号スイート名	スイート番号
グループZ	TLS_DHE_RSA_WITH_AES_128_CBC_SHA256	(0x00,0x67)
	TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA256	(0x00,0xBE)
	TLS_DHE_RSA_WITH_AES_128_CBC_SHA	(0x00,0x33)
	TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA	(0x00,0x45)
	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	(0xC0,0x23)
	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	(0xC0,0x27)
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_CBC_SHA256	(0xC0,0x72)
	TLS_ECDHE_RSA_WITH_CAMELLIA_128_CBC_SHA256	(0xC0,0x76)
	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA	(0xC0,0x09)
	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	(0xC0,0x13)
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA256	(0x00,0x6B)
	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA256	(0x00,0xC4)
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA	(0x00,0x39)
	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA	(0x00,0x88)
	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	(0xC0,0x24)
	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	(0xC0,0x28)
	TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_CBC_SHA384	(0xC0,0x73)
	TLS_ECDHE_RSA_WITH_CAMELLIA_256_CBC_SHA384	(0xC0,0x77)
	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA	(0xC0,0x0A)
	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA	(0xC0,0x14)
	TLS_RSA_WITH_AES_128_CBC_SHA256	(0x00,0x3C)
	TLS_DH_RSA_WITH_AES_128_CBC_SHA256	(0x00,0x3F)
	TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256	(0xC0,0x25)
	TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256	(0xC0,0x29)
	TLS_RSA_WITH_CAMELLIA_128_CBC_SHA256	(0x00,0xBA)
	TLS_DH_RSA_WITH_CAMELLIA_128_CBC_SHA256	(0x00,0xBC)
	TLS_ECDH_ECDSA_WITH_CAMELLIA_128_CBC_SHA256	(0xC0,0x74)
	TLS_ECDH_RSA_WITH_CAMELLIA_128_CBC_SHA256	(0xC0,0x78)
	TLS_RSA_WITH_AES_128_CBC_SHA	(0x00,0x2F)
	TLS_DH_RSA_WITH_AES_128_CBC_SHA	(0x00,0x31)
	TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA	(0xC0,0x04)
	TLS_ECDH_RSA_WITH_AES_128_CBC_SHA	(0xC0,0x0E)
	TLS_RSA_WITH_CAMELLIA_128_CBC_SHA	(0x00,0x41)
	TLS_DH_RSA_WITH_CAMELLIA_128_CBC_SHA	(0x00,0x43)
	TLS_RSA_WITH_AES_256_CBC_SHA256	(0x00,0x3D)
	TLS_DH_RSA_WITH_AES_256_CBC_SHA256	(0x00,0x69)
	TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384	(0xC0,0x26)
	TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384	(0xC0,0x2A)
	TLS_RSA_WITH_CAMELLIA_256_CBC_SHA256	(0x00,0xC0)
	TLS_DH_RSA_WITH_CAMELLIA_256_CBC_SHA256	(0x00,0xC2)
	TLS_ECDH_ECDSA_WITH_CAMELLIA_256_CBC_SHA384	(0xC0,0x75)
	TLS_ECDH_RSA_WITH_CAMELLIA_256_CBC_SHA384	(0xC0,0x79)
	TLS_RSA_WITH_AES_256_CBC_SHA	(0x00,0x35)
	TLS_DH_RSA_WITH_AES_256_CBC_SHA	(0x00,0x37)
	TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA	(0xC0,0x05)
	TLS_ECDH_RSA_WITH_AES_256_CBC_SHA	(0xC0,0x0F)
	TLS_RSA_WITH_CAMELLIA_256_CBC_SHA	(0x00,0x84)
	TLS_DH_RSA_WITH_CAMELLIA_256_CBC_SHA	(0x00,0x86)

【推奨項目】TLS1.3を利用する場合の優先順位		
優先順位グループ	暗号スイート名	スイート番号
グループA	TLS_AES_128_GCM_SHA256	(0x13,0x01)
	TLS_AES_128_CCM_SHA256	(0x13,0x04)
	TLS_AES_128_CCM_8_SHA256	(0x13,0x05)
	TLS_AES_256_GCM_SHA384	(0x13,0x02)
	TLS_CHACHA20_POLY1305_SHA256	(0x13,0x03)
鍵交換	DHE	RFC8446に規定
	ECDHE	RFC8446に規定
署名	RSASSA-PKCS1-v1_5	RFC8446に規定
	RSA-PSS	RFC8446に規定
	ECDSA	RFC8446に規定

Appendix B：サーバ設定編

サーバ設定を行ううえでの参考情報として､設定方法例を記載した参考ガイドを以下の URL にて公開している。

なお、利用するバージョンやディストリビューションの違いにより、設定方法が異なったり、設定ができなかったりする場合があることに留意すること。正式な取扱説明書やマニュアルを参照するとともに、一参考資料として利用されたい。

URL: https://www.ipa.go.jp/security/vuln/ssl_crypt_config.html

Appendix C：暗号スイートの設定例

暗号スイートの設定を行ううえでの参考情報として、設定方法例を記載した参考ガイドを以下の URL にて公開している。

なお、利用するバージョンやディストリビューションの違いにより、設定方法が異なったり、設定ができなかったりする場合があることに留意すること。正式な取扱説明書やマニュアルを参照するとともに、一参考資料として利用されたい。

URL: https://www.ipa.go.jp/security/vuln/ssl_crypt_config.html

Appendix D：ルート CA 証明書の取り扱い

D.1.　ルート CA 証明書の暗号アルゴリズム及び鍵長の確認方法

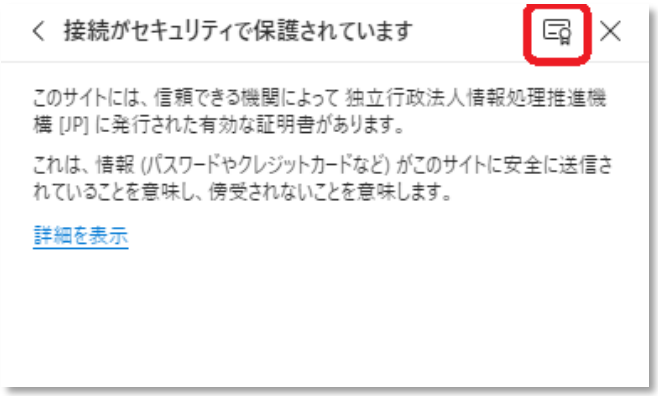
サーバ証明書を発行するサービスから発行された既存のサーバ証明書を利用したサイト、あるいはテストサイトなどの URL がわかっている場合には、当該 URL にアクセスして、以下のような手順を経ることで、ルート CA の公開鍵暗号アルゴリズム及び鍵長を確認することが可能である。

【Microsoft Edge（Chromium ベース）でアクセスする場合】

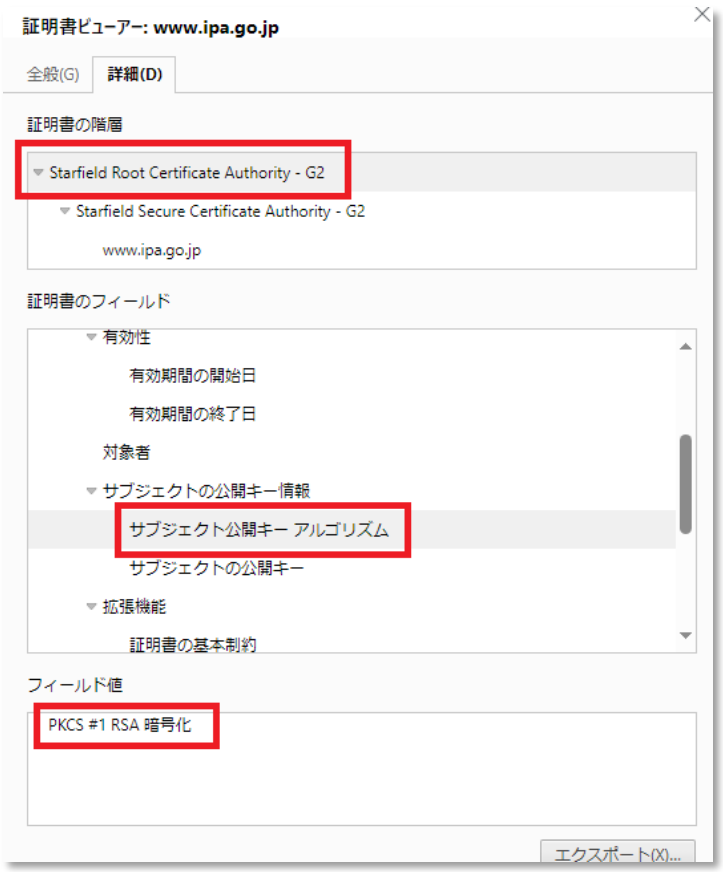
① 錠前マークをクリックする。

② 「接続がセキュリティで保護されています」をクリックする。

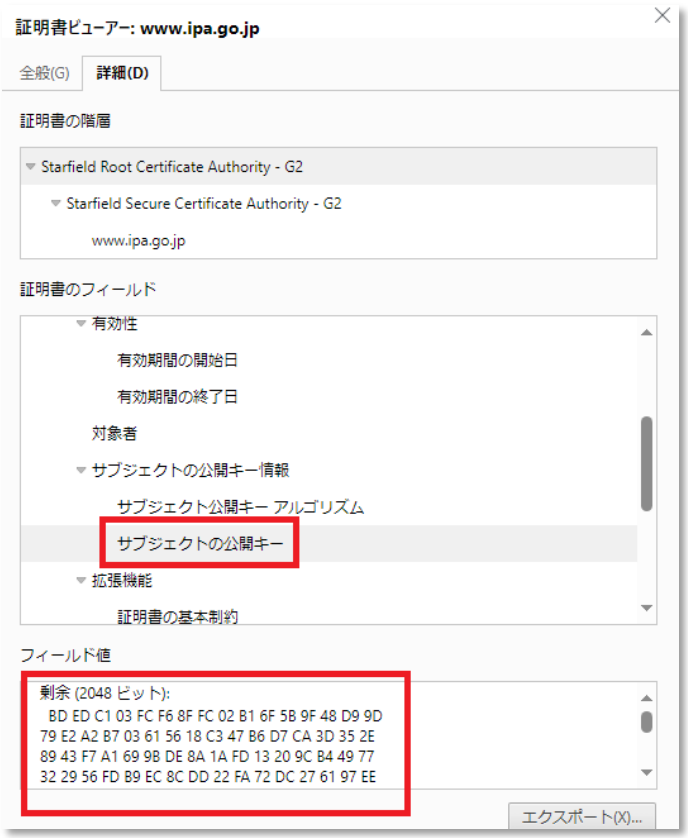
③　「証明書」アイコンをクリックする。
なお、EV 証明書の場合は、発行先情報が表示される。



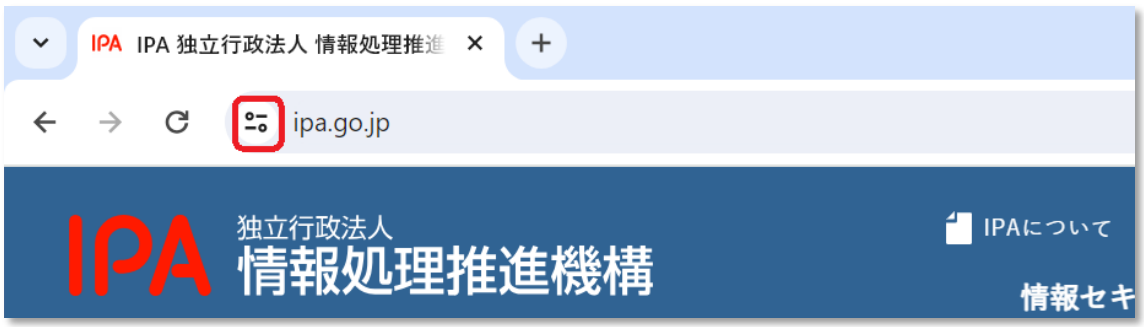
④　詳細タブをクリックし、証明書の階層の一番上に表示されている部分（これがルート CA 証明書に当たる）を選択する。証明書のフィールドをスクロールして「サブジェクト公開キー アルゴリズム」を選択する。署名アルゴリズムが「PKCS#1 RSA 暗号化（=RSASSA-PKCS1-v1_5 に相当）」であることが分かる。



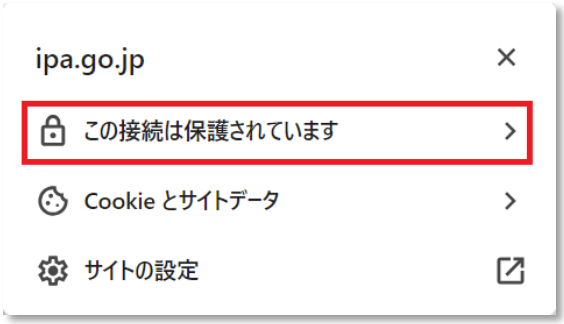
⑤　証明書フィールドをスクロールして、「サブジェクトの公開キー」を選択する。
この例では、暗号アルゴリズムが RSA、鍵長が 2048 ビットであることがわかる、



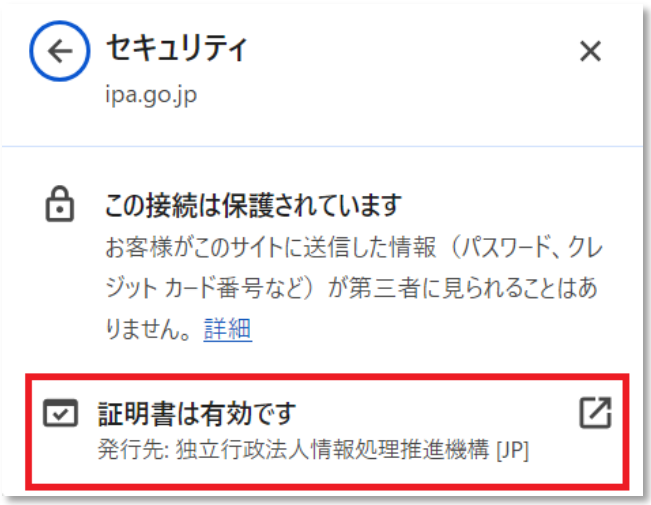
【Google Chrome でアクセスする場合】
①　調整アイコンをクリックする



②　「この情報は保護されています」をクリックする。

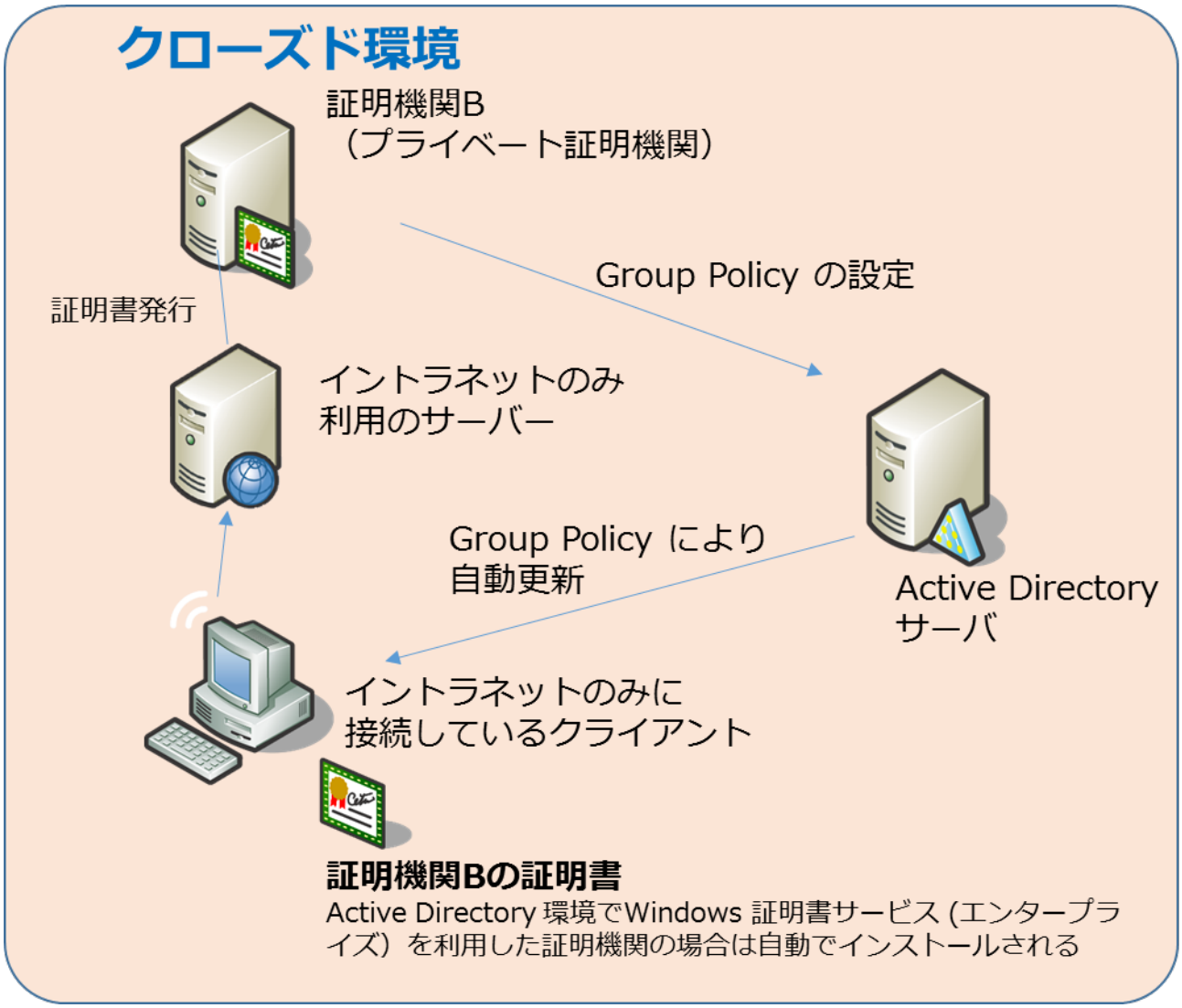


③ 「証明書は有効です」をクリックする。
なお、EV 証明書の場合は、発行先情報が表示される。



④ 「Microsoft Edge でアクセスする場合」と同様の手順（④～⑤）で、「サブジェクトの公開鍵」フィールドに表示されている値を確認する。

D.2. Active Directory を利用したプライベートルート CA 証明書の自動更新

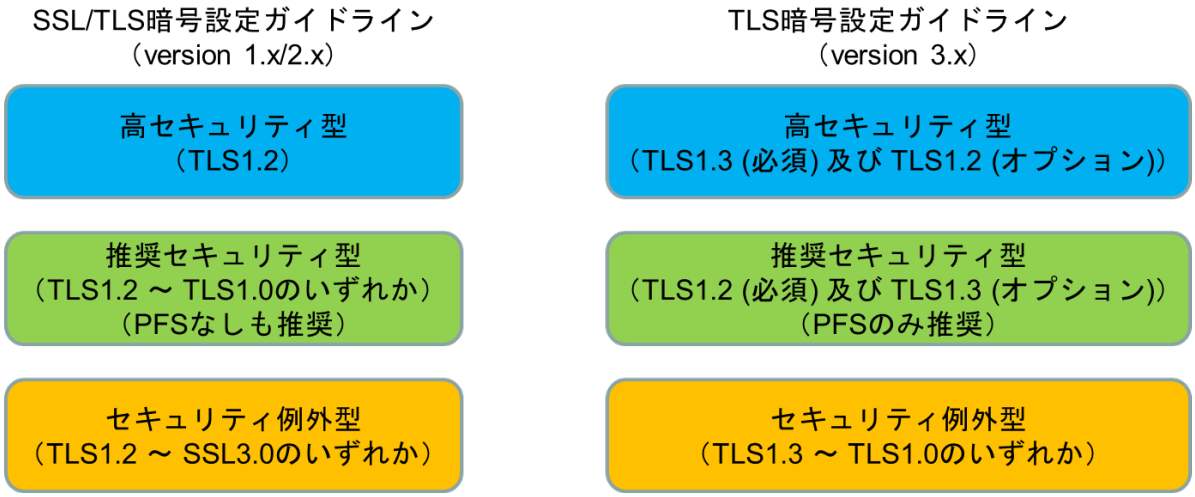


Appendix E：version 1.x/2.x と version 3.x の大きな差分

「SSL/TLS 暗号設定ガイドライン（version 1.x/2.x)」と「TLS 暗号設定ガイドライン（version 3.x)」との大きな差分は以下の通りである。

1. TLS1.3 の採用及び SSL3.0 の禁止に伴う各設定基準における要求設定の変更

TLS 暗号設定ガイドライン（version 3.x）では、プロトコルバージョンの要求設定において TLS1.3 の採用及び SSL3.0 の禁止が行われた。これに伴い、各設定基準における要求設定についても大幅な変更が行われており、SSL/TLS 暗号設定ガイドライン（version 1.x/2.x）における設定基準から一段階高い安全性を求めるようになった項目も多い。例えば、推奨セキュリティ型で利用が認められていた TLS1.0 や TLS1.1 は、本ガイドラインではセキュリティ例外型のみで利用可能となった。また、鍵交換では Perfect Forward Secrecy の特性をもつ ECDHE や DHE をさらに強く推奨するようにした。



2. 要求設定における「遵守項目」と「推奨項目」の区分けの新設

SSL/TLS 暗号設定ガイドライン（version 1.x/2.x）では全ての設定項目について一律に「要求設定」と位置付けていた。

今回は、設定項目のおける安全性への寄与度を考慮し、TLS 暗号設定ガイドライン（version 3.x）では、選択した設定基準としての最低限の安全性を確保するために必ず満たさなければならない「遵守項目」と当該設定基準としてよりよい安全性を実現するために満たすことが望ましい「推奨項目」とに分け、より現実的かつ実効性が高い要求設定とした。

3. 章構成の変更

SSL/TLS 暗号設定ガイドライン（version 1.x/2.x）では、

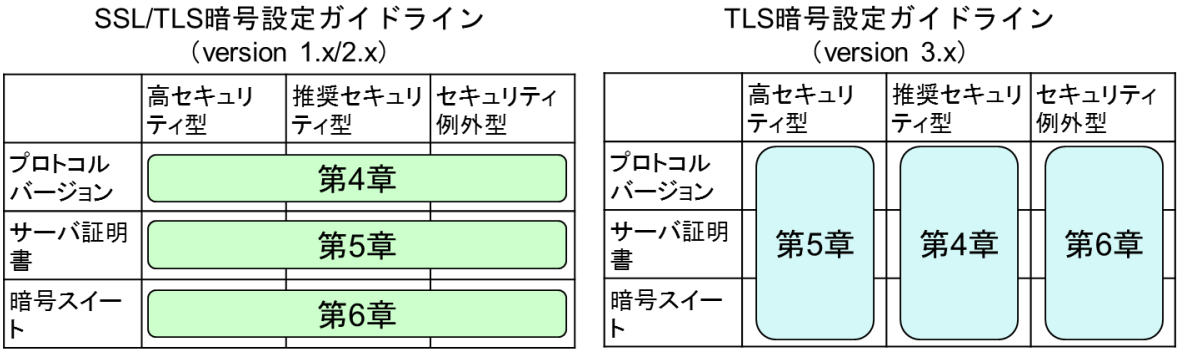
- プロトコルバージョンの設定（4 章）
- サーバ証明書の設定（5 章）
- 暗号スイートの設定（6 章）

の章構成とし、各章に「高セキュリティ型」「推奨セキュリティ型」「セキュリティ例外型」の設定項目を記載していた。

今回、章構成を見直し、TLS 暗号設定ガイドライン（version 3.x）では、

- 推奨セキュリティ型の要求設定（4 章）
- 高セキュリティ型の要求設定（5 章）
- セキュリティ例外型の要求設定（6 章）

の章構成とし、各章に「プロトコルバージョン」「サーバ証明書」「暗号スイート」の設定項目を記載した。これにより、選択した設定基準での該当章だけを参照すればよい構成とした。



発行日　2024 年 x 月 x 日　　第 3.1.0 版

発行者

・〒113-6591
東京都文京区本駒込二丁目 28 番 8 号
独立行政法人 情報処理推進機構
（セキュリティセンター　セキュリティ技術評価部　暗号グループ）
INFORMATION-TECHNOLOGY PROMOTION AGENCY, JAPAN
2-28-8 HONKOMAGOME, BUNKYO-KU
TOKYO, 113-6591 JAPAN

・〒184-8795
東京都小金井市貫井北町四丁目 2 番 1 号
国立研究開発法人 情報通信研究機構
（サイバーセキュリティ研究所　セキュリティ基盤研究室）
NATIONAL INSTITUTE OF
INFORMATION AND COMMUNICATIONS TECHNOLOGY
4-2-1 NUKUI-KITAMACHI, KOGANEI
TOKYO, 184-8795 JAPAN

資料４－４

暗号技術活用委員会からの Triple DES 等の取り扱いについての意見

1. 経緯

NIST は Triple DES を規定していた SP 800-67 Revision 2 を 2023 年 12 月 31 日に（予定通り）廃止したこと（下記）に伴い、暗号技術検討会事務局からの Triple DES の取り扱いについての意見聴取の依頼に対し、暗号技術活用委員会としては 2.のとおり回答する。

The scheduled withdrawal of SP 800-67 Rev. 2 will signify that TDEA is no longer an approved block cipher. TDEA will continue to be allowed for the decryption, key unwrapping, and verification of MACs of already-protected data, and SP 800-67 Rev. 2 will remain available online for historical purposes.
https://csrc.nist.gov/news/2023/nist-to-withdraw-sp-800-67-rev-2

2. 暗号技術活用委員会としての回答

暗号技術活用委員会としては、Triple DES の取り扱いについて、以下のとおり意見表明する。

なお、本件に関連し、運用監視暗号リスト掲載のアルゴリズムの取り扱いおよび DSA の取り扱いについても合わせて意見付記する。

【Triple DES の扱いに対する意見】

- 現時点では、「運用監視暗号リスト」からの削除を検討する必要性はない
- 現時点では、「運用監視暗号リスト」の条件である「互換性維持以外の目的での利用は推奨しない。」が実質的かつ十分な制約になっており、特段の利用制限を付加する必要性もない
- 「SP 800-67 Revision 2 が 2023 年 12 月に廃止されたが、それ以外は、運用監視暗号リストに移行した時点での状況とほとんど変わっていないため、Triple DES の位置づけに変更はない。」との注釈を付記する

【上記意見に至った理由】

① 廃止理由が、安全性が著しく低下したわけではなく、NIST のスケジュールに基づく動きであること
② 利用実績調査結果からは依然として極めて高い実装率であること
③ すでに運用監視暗号リストに掲載されており、互換性維持以外の目的での利用が推奨されていないこと
④ NIST も、Triple DES ですでに暗号化されたデータに対する処理は引き続き許容していること
⑤ 「電子政府推奨暗号リスト」に掲載されている DSA は、現在の FIPS PUB 186-5 では廃止されているが、FIPS PUB 186-5 になるときに削除すべきとの議論はなかったこと

【運用監視暗号リスト掲載のアルゴリズムの取り扱いについて】

- 運用監視暗号リスト掲載の暗号アルゴリズムは、新規に極力使用しないように促していく活動を積極的に進めるべきである。

【DSA の扱いに対する意見】

- 今回、Triple DES の取り扱いについて検討することになった理由が「SP 800-67 Revision 2 が廃止された」ことが契機になっていると承知している。その場合、上記⑤に記載の通り、DSA も「現在の FIPS PUB 186-5 では廃止されている」ことから、Triple DES との注釈と同様の注釈を追記すべきではないか。

以上

資料５－１

電子政府における調達のために参照すべき暗号のリスト（CRYPTREC 暗号リスト）の更新について（案）

電子政府における調達のために参照すべき暗号のリスト（CRYPTREC 暗号リスト）に関して、次のとおり更新する方針について御審議いただきたい。

１．DSA の取り扱いについて注釈を付記

NIST が DSA を規定していた FIPS PUB 186-4 を 2024 年 2 月 3 日に廃止したが、安全性・利用実績の状況に大きな変化がないため、電子政府推奨暗号リストの「DSA」について、注釈として「FIPS PUB 186-5 では廃止されているが、本リスト掲載時から安全性・利用実績の状況に大きな変化がないため、掲載を継続する。」を追記する。

２．Triple DES の取り扱いについて注釈を付記

NIST が Triple DES を規定していた SP 800-67 Revision 2 を 2023 年 12 月 31 日に廃止したことに伴い、運用監視暗号リスト掲載の「3-key Triple DES」について、暗号技術活用委員会からの回答を踏まえ、注釈として「SP 800-67 Revision 2 では廃止されているが、本リスト掲載時から安全性・利用実績の状況に大きな変化がないため、掲載を継続する。」を追記する。

以上

資料５－２

CRYPTREC LS-0001-2022R1

電子政府における調達のために参照すべき暗号のリスト（CRYPTREC暗号リスト）（更新事務局案）

令和５年３月30日

デジタル庁・総務省・経済産業省

（最終更新：令和●年●月●日）

電子政府推奨暗号リスト

暗号技術検討会[1]及び関連委員会（以下、「CRYPTREC」という。）により安全性及び実装性能が確認された暗号技術[2]について、市場における利用実績が十分であるか今後の普及が見込まれると判断され、当該技術の利用を推奨するもののリスト。なお、利用する鍵長について、「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」[5]の規定に合致しない鍵長を用いた場合には、電子政府推奨暗号リストの暗号技術を利用しているとは見なされないことに留意すること。

技術分類		暗号技術
公開鍵暗号	署名	DSA（注18）
		ECDSA
		EdDSA
		RSA-PSS（注1）
		RSASSA-PKCS1-v1_5（注1）
	守秘	RSA-OAEP（注1）
	鍵共有	DH
		ECDH
共通鍵暗号	64ビットブロック暗号（注2）	該当なし
	128ビットブロック暗号	AES
		Camellia
	ストリーム暗号	KCipher-2
ハッシュ関数		SHA-256
		SHA-384
		SHA-512
		SHA-512/256
		SHA3-256
		SHA3-384
		SHA3-512
		SHAKE128（注12）
		SHAKE256（注12）
（次ページに続く）		

[1]デジタル庁統括官、総務省サイバーセキュリティ統括官及び経済産業省商務情報政策局長が有識者の参集を求め、暗号技術の普及による情報セキュリティ対策の推進を図る観点から、専門家による意見等を聴取することにより、デジタル庁、総務省及び経済産業省における施策の検討に資することを目的として開催。

[2] 暗号利用モード、メッセージ認証コード、エンティティ認証は、他の技術分類の暗号技術と組み合わせて利用することとされているが、その場合、CRYPTREC暗号リストに掲載されたいずれかの暗号技術と組み合わせること。

[5] CRYPTREC, 暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準, https://www.cryptrec.go.jp/list.html

技術分類		暗号技術
暗号利用モード	秘匿モード	CBC
		CFB
		CTR
		OFB
		XTS(注17)
	認証付き秘匿モード(注13)	CCM
		GCM(注4)
メッセージ認証コード		CMAC
		HMAC
認証暗号		ChaCha20-Poly1305
エンティティ認証		ISO/IEC 9798-2
		ISO/IEC 9798-3
		ISO/IEC 9798-4

（注1） 「政府機関の情報システムにおいて使用されている暗号アルゴリズムSHA-1及びRSA1024に係る移行指針」（平成20年４月情報セキュリティ政策会議決定、平成24年10月情報セキュリティ対策推進会議改定）を踏まえて利用すること。
https://www.nisc.go.jp/pdf/policy/general/angou_ikoushishin.pdf
（平成25年３月１日現在）

（注2） CRYPTREC暗号リストにおいて、64ビットブロック暗号により、同一の鍵を用いて暗号化する場合、2^{20}ブロックまで、同一の鍵を用いてCMACでメッセージ認証コードを生成する場合、2^{21}ブロックまでとする。

（注4） 初期化ベクトル長は96ビットを推奨する。

（注12） ハッシュ長は256ビット以上とすること。

（注13） CRYPTREC暗号リスト掲載のブロック暗号を、認証付き秘匿モードと組み合わせて、「認証暗号」として使うことができる。

（注17） ブロック暗号には、CRYPTREC暗号リスト掲載128ビットブロック暗号を使う。利用用途はストレージデバイスの暗号化に限り、実装方法はNIST SP800-38Eに従うこと。

（注18） FIPS PUB 186-5では廃止されているが、本リスト掲載時から安全性・利用実績の状況に大きな変化がないため、掲載を継続する。

推奨候補暗号リスト

CRYPTRECにより安全性及び実装性能が確認され、今後、電子政府推奨暗号リストに掲載される可能性のある暗号技術[3]のリスト。なお、本リストに記載されている暗号技術を利用する際は、「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」[6]の規定に合致する鍵長を用いることが求められることに留意すること。

技術分類		暗号技術
公開鍵暗号	署名	該当なし
	守秘	該当なし
	鍵共有	PSEC-KEM（注5）
共通鍵暗号	64ビットブロック暗号（注6）	CIPHERUNICORN-E
		Hierocrypt-L1
		MISTY1
	128ビットブロック暗号	CIPHERUNICORN-A
		CLEFIA
		Hierocrypt-3
	ストリーム暗号	Enocoro-128v2
		MUGI
		MULTI-S01（注7）
ハッシュ関数		該当なし
暗号利用モード	秘匿モード	該当なし
	認証付き秘匿モード（注14）	該当なし
メッセージ認証コード		PC-MAC-AES
認証暗号		該当なし
エンティティ認証		該当なし

（注5） KEM (Key Encapsulating Mechanism) – DEM (Data Encapsulating Mechanism) 構成における利用を前提とする。

（注6） CRYPTREC暗号リストにおいて、64ビットブロック暗号により、同一の鍵を用いて暗号化する場合、2^{20}ブロックまで、同一の鍵を用いてCMACでメッセージ認証コードを生成する場合、2^{21}ブロックまでとする。

（注7） 平文サイズは64ビットの倍数に限る。

（注14） CRYPTREC暗号リスト掲載のブロック暗号を、認証付き秘匿モードと組み合わせて、「認証暗号」として使うことができる。

[3] 暗号利用モード、メッセージ認証コード、エンティティ認証は、他の技術分類の暗号技術と組み合わせて利用することとされているが、その場合、CRYPTREC暗号リストに掲載されたいずれかの暗号技術と組み合わせること。

[6] CRYPTREC, 暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準, https://www.cryptrec.go.jp/list.html

運用監視暗号リスト

実際に解読されるリスクが高まるなど、推奨すべき状態ではなくなったとCRYPTRECにより確認された暗号技術[4]のうち、互換性維持のために継続利用を容認するもののリスト。互換性維持[7]以外の目的での利用は推奨しない。なお、本リストに記載されている暗号技術を利用する際は、「暗号強度要件(アルゴリズム及び鍵長選択)に関する設定基準」[8]の規定に合致する鍵長を用いることが求められることに留意すること。

技術分類		暗号技術
公開鍵暗号	署名	該当なし
	守秘	RSAES-PKCS1-v1_5(注8)(注9)
	鍵共有	該当なし
共通鍵暗号	64ビットブロック暗号(注15)	3-key Triple DES(注19)
	128ビットブロック暗号	該当なし
	ストリーム暗号	該当なし
ハッシュ関数		RIPEMD-160
		SHA-1(注8)
暗号利用モード	秘匿モード	該当なし
	認証付き秘匿モード(注16)	該当なし
メッセージ認証コード		CBC-MAC(注11)
認証暗号		該当なし
エンティティ認証		該当なし

(注8) 「政府機関の情報システムにおいて使用されている暗号アルゴリズムSHA-1及びRSA1024に係る移行指針」(平成20年4月 情報セキュリティ政策会議決定、平成24年10月情報セキュリティ対策推進会議改定)を踏まえて利用すること。
https://www.nisc.go.jp/pdf/policy/general/angou_ikoushishin.pdf
(平成25年3月1日現在)

(注9) TLS 1.0, 1.1, 1.2で利用実績があることから当面の利用を認める。

(注11) 安全性の観点から、メッセージ長を固定して利用すべきである。

(注15) CRYPTREC暗号リストにおいて、64ビットブロック暗号により、同一の鍵を用いて暗号化する場合、2^{20}ブロックまで、同一の鍵を用いてCMACでメッセージ認証コードを生成する場合、2^{21}ブロックまでとする。

(注16) CRYPTREC暗号リスト掲載のブロック暗号を、認証付き秘匿モードと組み合わせて、「認証暗号」として使うことができる。

(注19) SP 800-67 Revision 2では廃止されているが、本リスト掲載時から安全性・利用実績の状況に大きな変化がないため、掲載を継続する。

[4] 暗号利用モード、メッセージ認証コード、エンティティ認証は、他の技術分類の暗号技術と組み合わせて利用することとされているが、その場合、CRYPTREC暗号リストに掲載されたいずれかの暗号技術と組み合わせること。
[7] 既に稼働中のシステムやアプリケーション等との間での相互運用を継続すること
[8] CRYPTREC, 暗号強度要件(アルゴリズム及び鍵長選択)に関する設定基準, https://www.cryptrec.go.jp/list.html

更新履歴情報

更新日付	更新箇所	更新前の記述	更新後の記述
令和６年 ●月●日	（注18）	[新規追加]	FIPS PUB 186-5では廃止されているが、本リスト掲載時から安全性・利用実績の状況に大きな変化がないため、掲載を継続する。
	（注19）	[新規追加]	SP 800-67 Revision 2では廃止されているが、本リスト掲載時から安全性・利用実績の状況に大きな変化がないため、掲載を継続する。

資料6-1

2023年度第1回 CRYPTREC暗号技術検討会

電子署名法特定認証業務の暗号基準の改正スケジュールについて

2024-03-26

デジタル庁　デジタル社会共通機能グループ　トラスト班

デジタル庁
Digital Agency

背景

- 電子署名及び認証業務に関する法律（平成12年法律第102号。以下「電子署名法」という。）における特定認証業務の認定の基準（第６条）の一つとして、利用する暗号の強度が施行規則に定められており、現在、112ビットセキュリティ強度以上の暗号が規定されている。

電子署名及び認証業務に関する法律施行規則
（平成13年総務省・法務省・経済産業省令第２号）

第二条　法第二条第三項の主務省令で定める基準は、電子署名の安全性が次のいずれかの有する困難性に基づくものであることとする。
一　ほぼ同じ大きさの二つの素数の積である二千四十八ビット以上の整数の素因数分解
二　大きさ二千四十八ビット以上の有限体の乗法群における離散対数の計算
三　楕円曲線上の点がなす大きさ二百二十四ビット以上の群における離散対数の計算
四　前三号に掲げるものに相当する困難性を有するものとして主務大臣が認めるもの

- また、2022年3月にデジタル庁・総務省・経済産業省が策定した「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準（CRYPTREC LS-0003-2022r1）」においては、2030年末までを112ビットセキュリティ強度の暗号の移行完遂期間（より上位の強度に移行する期間）としている。
- 電子署名法における特定認証業務の認定の基準についても、本基準を踏まえた改正等を実施するため、認定認証業務を営む事業者、認定認証業務が発行する電子証明書を受け入れる関係システムの運営者等に対して、アンケート及びヒヤリングを実施し、次の通り改正に係る方針をまとめた。

電子署名法特定認証業務の暗号基準改正に係る方針

- 電子署名法施行規則第２条及び電子署名法に基づく特定認証業務の認定に係る指針第３条に規定する特定認証業務の暗号アルゴリズム及び暗号強度に係る基準については、令和15年（2033年）末を目途に改正する。
  - 現行は112ビットセキュリティ強度以上を規定しているが、改正後は128ビットセキュリティ強度以上を規定す

理由

- 認定認証業務であることを民間認証局の相互認証基準の条件の一つとしているGPKIブリッジ認証局の暗号移行対応スケジュール（2028年秋に新環境の供用を開始する予定）
- 認定認証業務が発行する電子証明書の有効期限（最長5年）
- 認定認証事業者の事業の安定性及び利用者への影響（有効期限が短い証明書の発行に伴う影響）　等

- 本方針は、使用する暗号技術の解法アルゴリズムに係る進展、量子コンピュータの性能向上等の急速な危殆化等特別の事情が認められない場合に限り、令和15年末（2033年末）まで現行の暗号強度を特定認証業務に利用することを許容するものである。
- したがって、現行の暗号技術が急速に危殆化するおそれが生じた場合等、暗号技術の動向等を踏まえ、必要に応じて上記スケジュール及び基準の改正内容を見直すことがある。
- 本暗号移行方針について、次ページ以降の留意点を含めた上で、認定認証事業者に周知を実施する。

暗号移行に係る留意点 (1/3)

認定認証事業者に対し、暗号基準改正に係る方針とともに、認定認証業務の暗号移行対応にあたり以下の点に留意する旨周知。

＜暗号危殆化時の対応＞

- 112ビットセキュリティの安全性指標を持つ暗号技術が急速に危殆化するおそれが生じた場合には、これらを利用した電子証明書については、CRYPTRECによる注意喚起や主務省庁からの情報提供等を踏まえ、電子証明書の失効の請求や有効期限の短い電子証明書の発行等の対応を行い、電子署名に対する信頼性の低下を最小限とするように努めること。

＜利用者への周知＞

- 電子署名及び認証業務に関する法律に基づく特定認証業務の認定に係る指針（平成13年４月27日総務省・法務省・経済産業省告示第２号）第８条に定める利用申込者に対する説明に関する規定の通り、暗号危殆化時の取扱いについて利用申込者に説明を行うこと。
- 特に、令和13年（2031年）以降に有効期限を迎える電子証明書の利用申込者に対しては、電子証明書の失効の請求や有効期限の短い電子証明書の発行等の取扱いが発生する可能性を説明に含める等、その時点における暗号危殆化の状況を踏まえ、利用申込者への説明をより一層明確な形とすること。

暗号移行に係る留意点 (2/3)

**　認定認証事業者に対し、暗号基準改正に係る方針とともに、認定認証業務の暗号移行対応にあたり以下の点に留意する旨周知。**

＜新暗号への対応開始時期＞

- GPKIブリッジ認証局との相互認証を行う認定認証業務については、GPKIブリッジ認証局の新暗号対応予定時期である、令和10年（2028年）中を目途に新暗号に対応した認証局の運用を開始すること。

＜移行先の暗号アルゴリズム及び暗号強度＞

- 移行先の暗号技術の強度については、128ビットセキュリティ以上の安全性指標を持つ暗号技術を利用すること。
- なお、認定認証事業者であることを民間認証局の相互認証基準の一つ としている、GPKIブリッジ認証局においては、現時点において次の暗号アルゴリズム及び鍵長に対応予定である。（仕様の詳細等については、令和６年８月に公表予定である次期システムに係る相互運用性仕様書を参考にされたい。）
  - ECDSAWithSHA384（1.2.840.10045.4.3.3）
    ※ECDSAの曲線は鍵長が384ビットであるNIST P-384（secp384r1、OID： 1.3.132.0.34）とする。

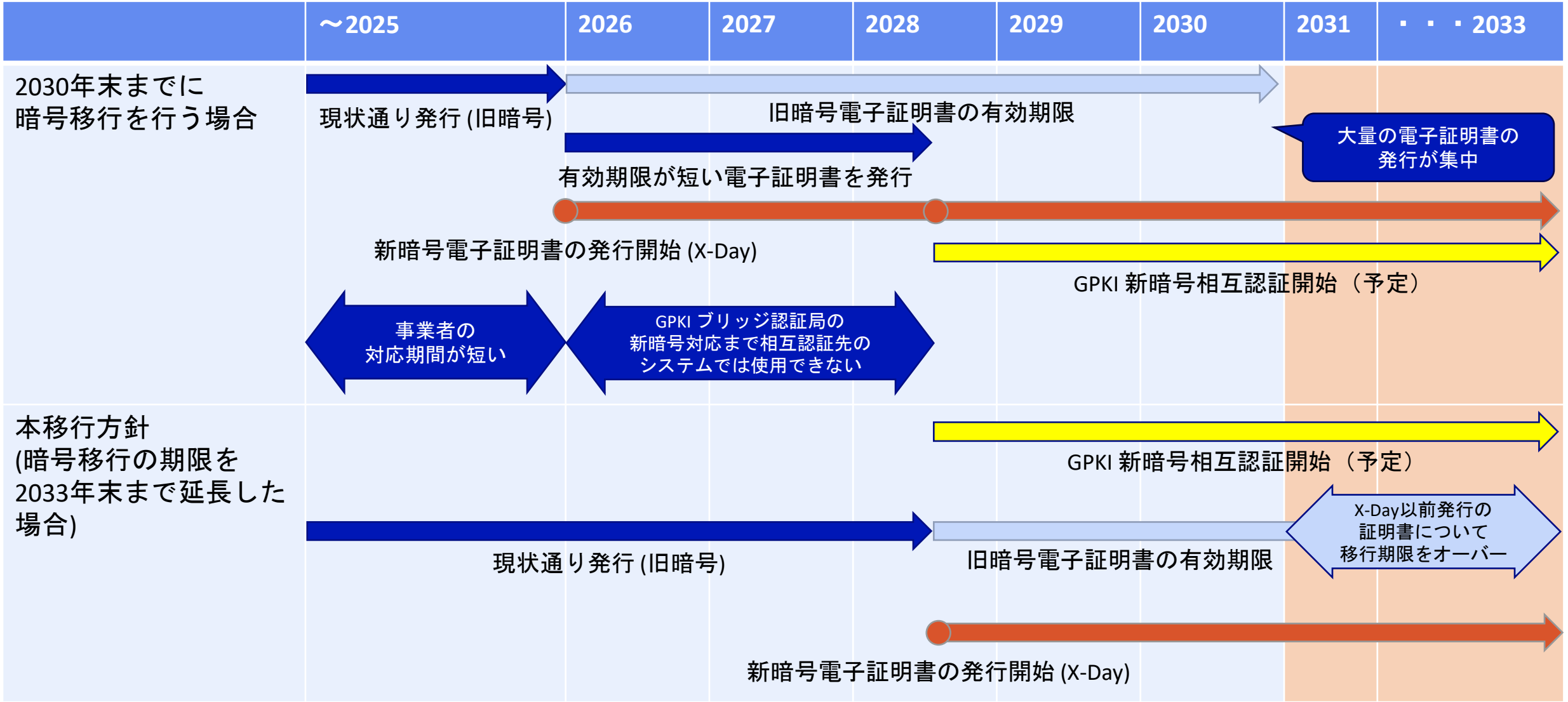
暗号移行に係る留意点 (3/3)

**　認定認証事業者に対し、暗号基準改正に係る方針とともに、認定認証業務の暗号移行対応にあたり以下の点に留意する旨周知。**

＜暗号移行に係る調整について＞

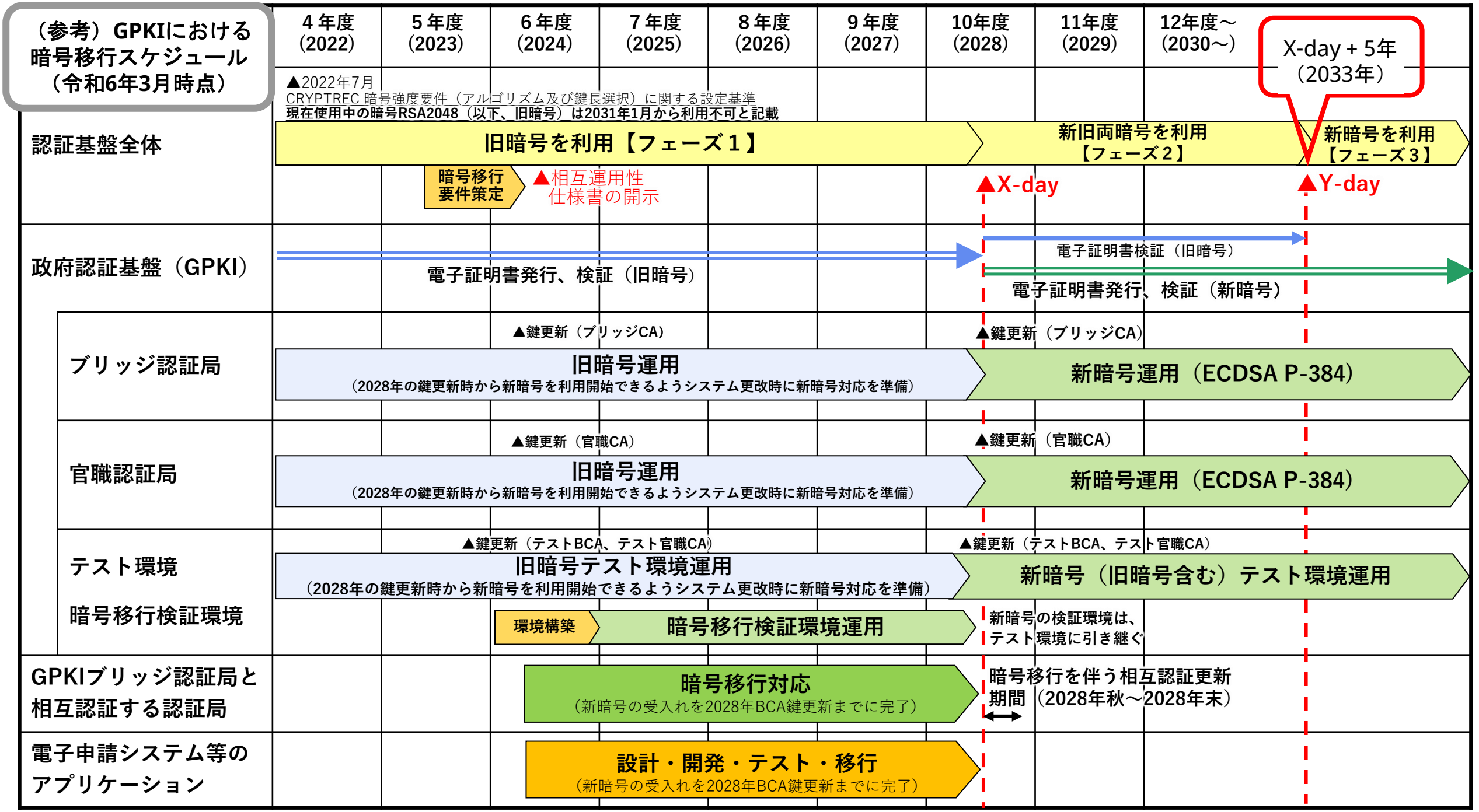
- GPKIブリッジ認証局の暗号移行時期の周辺においては、多数の認証局の暗号移行対応時期が重なると考えられる。そのため、認定に係る調査のスケジュールについては、主務省庁及び指定調査機関による調整に協力すること。
- また、GPKIブリッジ認証局についても、相互認証を実施している認定認証業務以外の認証局を含め、暗号移行対応の時期が重なるため、円滑な相互認証の実施のため、相互認証の審査に係るスケジュールについてデジタル庁による調整に協力すること。
- 加えて、電子証明書の暗号移行においては、認定認証業務のみならず、認定認証業務の電子証明書を受け入れている電子申請システム等署名検証側における対応も同時に行う必要があるため、関係システムを運営している行政機関及び民間組織との調整及び相互運用性の確保に努めること。

（参考）対応スケジュールの例



※電子署名法の認定認証事業における電子証明書の有効期限は**最長**5年

（参考）GPKIにおける暗号移行スケジュール（令和6年3月時点）



（参考）各種基準と本周知の関係整理

- NISC（内閣サイバーセキュリティセンター）が策定している「政府機関等のサイバーセキュリティ対策のための統一基準群（以下「NISC統一基準群」という。）」においては、政府情報システムにおいて、CRYPTREC電子政府推奨暗号リストに掲載されている暗号を利用することが定められている。
- また、CRYPTREC電子推奨暗号リストにおいては、「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準（CRYPTREC LS-0003-2022r1）（以下「鍵長要件」という。）」 に合致しない鍵長を用いた場合は、電子政府推奨暗号リストの暗号技術を利用していないとみなされる。
- 本周知に係る電子署名法特定認証業務の認定基準の対象である、民間事業者が運営する認証局については、政府情報システムではないため、NISC統一基準群（及び同基準群を踏まえて策定された各政府機関における情報セキュリティポリシー等）、CRYPTREC電子政府推奨暗号リスト、鍵長要件のいずれについても、直接の対象ではない。
- 本周知については、電子署名法の認定基準の観点から112ビットセキュリティ強度暗号の移行に係る方針及びスケジュール等を整理したものである。
- 政府情報システムにおいて、2030年末以降も112ビットセキュリティ強度暗号の利用を容認する必要がある場合については、NISC統一基準群等における例外措置の適用等、各システム運営者における対処が必要である。

（参考）他システムにおける暗号移行関係

- **個人番号カード（マイナンバーカード）について**

次期個人番号カードタスクフォース最終とりまとめ

２．その他重要論点

(1)次期カード発行直前に発行されるカードの電子証明書の扱い

現行カードの電子証明書に用いられる暗号アルゴリズム（RSA2048）は、CRYPTRECの「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」において 2031 年１月１日以降利用不可とされているが、このスケジュールでの次期カードへの移行完了は困難と考えられる。CRYPTREC では当該基準の見直しを少なくとも５年毎に行うことになっており、次の見直しのタイミングは 2026～2027 年頃と予想されるため、状況によっては利用延長に向けた相談等の検討を行う必要がある。

なお、その場合においても、可能な限り速やかに新暗号への移行を図るため、次期カード導入時期以降、現行カードの電子証明書の更新の際には、電子証明書の更新ではなく、次期カードの取得を推奨する（電子証明書の更新を案内する時期に、電子証明書の更新を案内するのではなく、交付申請書を送付して、次期カードの取得を勧奨する等（※））。

※現行の RSA2048 による電子証明書の利用がどの程度の期間許容されるかを判断し、具体的な対応案を検討する。

https://www.digital.go.jp/assets/contents/node/basic_page/field_ref_resources/58b82d5b-338d-4f5b-be7e-7b771135e2c3/0d7a6b39/20240318_meeting_mynumber-card-renewal_outline_06.pdf

デジタル庁

Digital Agency

資料6-2

事　務　連　絡
令和６年●月●日

認定認証事業者　各位

特定認証業務の基準の改正スケジュール等の周知について

デジタル庁　デジタル社会共通機能グループ　参事官（トラスト担当）
法務省　民事局商事課

平素からデジタル行政にご理解とご協力を賜り、厚く御礼申し上げます。

先般、デジタル庁・総務省・経済産業省において「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」を策定したところですが、電子署名及び認証業務に関する法律施行規則（平成 13 年総務省・法務省・経済産業省令第２号）第２条及び電子署名及び認証業務に関する法律に基づく特定認証業務の認定に係る指針（平成 13 年４月 27 日総務省・法務省・経済産業省告示第２号）第３条に規定する特定認証業務の基準につきまして、別添の通り「電子署名法特定認証業務の基準における暗号移行方針」を策定しましたので、電子署名法第 33 条の規定に基づき情報提供します。認定認証事業者各位におかれましては、本方針を踏まえ、円滑な暗号移行へのご協力をお願いいたします。

資料6-2

（別添）

令和６年●月●日

電子署名法特定認証業務の基準における暗号移行方針

デジタル庁　デジタル社会共通機能グループ　参事官（トラスト担当）
法務省　民事局商事課

１．特定認証業務の基準の改正スケジュール及び改正内容

電子署名及び認証業務に関する法律施行規則（平成 13 年総務省・法務省・経済産業省令第２号）第２条及び電子署名及び認証業務に関する法律に基づく特定認証業務の認定に係る指針（平成 13 年４月 27 日総務省・法務省・経済産業省告示第２号）第３条に規定する特定認証業務の暗号アルゴリズム及び暗号強度に係る基準については、令和 15 年（2033 年）末を目途に改正する。

現行は 112 ビットセキュリティ強度以上を規定しているが、改正後は 128 ビットセキュリティ強度[1]以上を規定することを想定している。

これは、認定認証業務が発行する電子証明書の有効期限、認定認証事業者の事業の安定性及び利用者への影響、認定認証業務であることを民間認証局の相互認証基準の条件の一つとしている GPKI ブリッジ認証局の暗号移行対応スケジュール等を踏まえ、使用する暗号技術の解法アルゴリズムに係る進展、量子コンピュータの性能向上等の急速な危殆化等特別の事情が認められない場合に限り、令和 15 年末（2033 年末）まで現行の暗号強度を特定認証業務に利用することを許容するものである。

したがって、現行の暗号技術が急速に危殆化するおそれが生じた場合等、暗号技術の動向等を踏まえ、必要に応じて上記スケジュール及び改正内容を見直すことがある。

２．暗号移行に係る留意点

認定認証業務の暗号移行の対応にあたっては、以下の点に留意することとする。

＜暗号危殆化時の対応＞

112 ビットセキュリティの安全性指標を持つ暗号技術が急速に危殆化するおそれが生じた場合には、これらを利用した電子証明書については、CRYPTREC による注意喚起や主務省庁からの情報提供等を踏まえ、電子証明書の失効の請求や有効期限の短い電子証明書の発行等の対応を行い、電子署名に対する信頼性の低下を最小限とするように努めること。

---

[1] 112 ビットセキュリティ強度を持つ暗号技術の例としては、2048 ビットの RSA 暗号及び 224 ビットの楕円曲線暗号（P-224）、128 ビットセキュリティ強度を持つ暗号技術の例としては、3072 ビットの RSA 暗号及び 256 ビットの楕円曲線暗号（P-256）が挙げられる。

＜利用者への周知＞

電子署名及び認証業務に関する法律に基づく特定認証業務の認定に係る指針（平成13年４月27日総務省・法務省・経済産業省告示第２号）第８条に定める利用申込者に対する説明に関する規定の通り、暗号危殆化時の取扱いについて利用申込者に説明を行うこと。

特に、令和13年（2031年）以降に有効期限を迎える電子証明書の利用申込者に対しては、電子証明書の失効の請求や有効期限の短い電子証明書の発行等の取扱いが発生する可能性を説明に含める等、その時点における暗号危殆化の状況を踏まえ、利用申込者への説明をより一層明確な形とすること。

＜新暗号への対応開始時期＞

GPKIブリッジ認証局との相互認証を行う認定認証業務については、GPKIブリッジ認証局の新暗号対応予定時期である、令和10年（2028年）中を目途に新暗号に対応した認証局の運用を開始すること。

＜移行先の暗号アルゴリズム及び暗号強度＞

移行先の暗号技術の強度については、128ビットセキュリティ以上の安全性指標を持つ暗号技術を利用すること。

なお、認定認証事業者であることを民間認証局の相互認証基準の一つ[2]としているGPKIブリッジ認証局においては、現時点において次の暗号アルゴリズム及び鍵長に対応予定である。[3]

・ECDSAWithSHA384（1.2.840.10045.4.3.3）

※ECDSAの曲線は鍵長が384ビットであるNIST P-384（secp384r1、OID：1.3.132.0.34）とする。

＜暗号移行に係る調整について＞

GPKIブリッジ認証局の暗号移行時期の周辺においては、多数の認証局の暗号移行対応時期が重なると考えられる。そのため、認定に係る調査のスケジュールについては、主務省庁及び指定調査機関による調整に協力すること。

また、GPKIブリッジ認証局についても、相互認証を実施している認定認証業務以外の認証局を含め、暗号移行対応の時期が重なるため、円滑な相互認証の実施のため、相互認証の審査に係るスケジュールについてデジタル庁による調整に協力すること。

---

[2] 政府認証基盤におけるブリッジ認証局の相互認証基準について（平成13年4月25日行政情報化推進各省庁連絡会議幹事会了承。最終改定令和3年12月9日デジタル社会推進会議関係課長等連絡会議了承。）

https://www.gpki.go.jp/cross/cross.pdf

[3] 仕様の詳細等については、令和６年８月に公表予定である次期システムに係る相互運用性仕様書を参考にされたい。

（別添）

加えて、電子証明書の暗号移行においては、認定認証業務のみならず、認定認証業務の電子証明書を受け入れている電子申請システム等署名検証側における対応も同時に行う必要があるため、関係システムを運営している行政機関及び民間組織との調整及び相互運用性の確保に努めること。

資料6-2

（別添）

＜参考１＞電子署名法施行規則（抄）
第２条　法第２条第３項の主務省令で定める基準は、電子署名の安全性が次のいずれかの有する困難性に基づくものであることとする。
一　ほぼ同じ大きさの二つの素数の積である 2048 ビット以上の整数の素因数分解
二　大きさ 2048 ビット以上の有限体の乗法群における離散対数の計算
三　楕円曲線上の点がなす大きさ 224 ビット以上の群における離散対数の計算
四　前三号に掲げるものに相当する困難性を有するものとして主務大臣が認めるもの

＜参考２＞特定認証業務の認定に係る指針（抄）
第３条　規則第二条の基準を満たす電子署名の方式は、次の各号のいずれかとする。
一　RSA 方式であって、ハッシュ関数として SHA-256 を使用するもの（オブジェクト識別子 1 2 840 113549 1 1 11)、SHA-384 を使用するもの（オブジェクト識別子 1 2 840 113549 1 1 12）又は SHA-512 を使用するもの（オブジェクト識別子 1 2 840 113549 11 13）のうち、モジュラスとなる合成数が 2048 ビット以上のもの
二　RSA-PSS 方式（オブジェクト識別子 1 2 840 113549 1 1 10）であって、SHA-256（オブジェクト識別子 2 16 840 1 101 3 4 2 1)、SHA-384（オブジェクト識別子 2 16 840 1101 3 4 2 2）又は SHA-512（オブジェクト識別子 2 16 8401 101 3 4 2 3）を使用するもののうち、モジュラスとなる合成数が 2048 ビット以上のもの
三　ECDSA 方式であって、ハッシュ関数として SHA-256 を使用するもの（オブジェクト識別子 1 2 840 10045 4 3 2)、SHA-384 を使用するもの（オブジェクト識別子 1 2 840 10045 4 3 3）又は SHA-512 を使用するもの（オブジェクト識別子 1 2 840 10045 4 3 4）のうち、楕円曲線の定義体及び位数が 224 ビット以上のもの
四　DSA 方式であって、ハッシュ関数として SHA-256 を使用するもの（オブジェクト識別子 2 16 840 1 101 3 4 3 2）であり、かつ、モジュラスとなる素数が 2048 ビット以上のもの

（別添）

＜参考３＞CRYPTREC 暗号リスト（電子政府推奨暗号リスト）（抜粋）
https://www.cryptrec.go.jp/list/cryptrec-ls-0001-2022.pdf

暗号技術検討会[4]及び関連委員会（以下、「CRYPTREC」という。）により安全性及び実装性能が確認された暗号技術[5]について、市場における利用実績が十分であるか今後の普及が見込まれると判断され、当該技術の利用を推奨するもののリスト。

なお、利用する鍵長について、「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準」[6]の規定に合致しない鍵長を用いた場合には、電子政府推奨暗号リストの暗号技術を利用しているとは見なされないことに留意すること。

[4] デジタル庁統括官、総務省サイバーセキュリティ統括官及び経済産業省商務情報政策局長が有識者の参集を求め、暗号技術の普及による情報セキュリティ対策の推進を図る観点から、専門家による意見等を聴取することにより、デジタル庁、総務省及び経済産業省における施策の検討に資することを目的として開催。

[5] 暗号利用モード、メッセージ認証コード、エンティティ認証は、他の技術分類の暗号技術と組み合わせて利用することとされているが、その場合、CRYPTREC 暗号リストに掲載されたいずれかの暗号技術と組み合わせること。

[6] CRYPTREC, 暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準, https://www.cryptrec.go.jp/list.html

＜参考４＞暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準（抜粋）

https://www.cryptrec.go.jp/list/cryptrec-ls-0003-2022.pdf

2.2.1　公開鍵暗号の推定セキュリティ強度

表２　公開鍵暗号の推定セキュリティ強度[7]

セキュリティ強度（ビットセキュリティ）	IFC RSA-PSS RSASSA-PKCS1-v1.5 RSA-OAEP RSAES-PKCS1-v1_5	FFC DSA DH	ECC ECDSA ECDH PSEC-KEM
112	k = 2048	(L, N) = (2048, 224)	P-224 B-233 K-233
128	k = 3072	(L, N) = (3072, 256)	P-256 B-283 K-283 W-25519 Curve25519 Edwards25519
192	k = 7680	(L, N) = (7680, 384)	P-384 B-409 K-409 W-448 Curve448 Edwards448
256	k = 15360	(L, N) = (15360, 512)	P-521 B-571 K-571

[7] P: curve over Prime fields（素体曲線）、B: curve over Binary fields（拡大体（バイナリ）曲線）、K: Koblitz-curve（コブリッツ曲線）、W: Weierstrass-curve（ワイエルシュトラス曲線）、Curve: Montgomery-curve（モンゴメリ曲線）、Edwards: Edwards-curve（エドワード曲線）

3.2　セキュリティ強度要件の基本設定方針

表 5　セキュリティ強度要件の基本設定方針

<table>
<tr><th colspan="2">想定運用終了・廃棄年／
利用期間</th><th>2022～2030</th><th>2031～2040</th><th>2041～2050</th><th>2051～2060</th><th>2061～2070</th></tr>
<tr><td rowspan="2">112 ビット
セキュリティ</td><td>新規生成
((a)参照)</td><td rowspan="2">移行完遂
期間
((c)参照)</td><td>利用不可</td><td rowspan="2">利用不可</td><td rowspan="2">利用不可</td><td rowspan="2">利用不可</td></tr>
<tr><td>処理
((b)参照)</td><td>許容</td></tr>
<tr><td rowspan="2">128 ビット
セキュリティ</td><td>新規生成
((a)参照)</td><td rowspan="2">利用可</td><td rowspan="2">利用可</td><td rowspan="2">移行完遂
期間
((c)参照)</td><td>利用不可</td><td rowspan="2">利用不可</td></tr>
<tr><td>処理
((b)参照)</td><td>許容</td></tr>
<tr><td rowspan="2">192 ビット
セキュリティ</td><td>新規生成
((a)参照)</td><td rowspan="2">利用可</td><td rowspan="2">利用可</td><td rowspan="2">利用可</td><td rowspan="2">利用可</td><td rowspan="2">利用可</td></tr>
<tr><td>処理
((b)参照)</td></tr>
<tr><td rowspan="2">256 ビット
セキュリティ</td><td>新規生成
((a)参照)</td><td rowspan="2">利用可</td><td rowspan="2">利用可</td><td rowspan="2">利用可</td><td rowspan="2">利用可</td><td rowspan="2">利用可</td></tr>
<tr><td>処理
((b)参照)</td></tr>
</table>

(a)　　新規に暗号保護を適用する（例えば、暗号化や署名生成を実行する）際は、原則として、2040 年までは 128 ビット以上のセキュリティ強度のものを選択すべきである。2041 年以降は 192 ビット以上のセキュリティ強度のものを選択すべきである。

(b)　　保護済みのデータに対して処理を実行する（例えば、復号や署名検証を実行する）際は、2040 年までは 128 ビット以上、2041 年以降は 192 ビット以上のセキュリティ強度のものを選択すべきである。ただし、保護済みのデータに対する正当性を担保又は確認するための何らかの技術的又は運用的な対策やルール等（暗号技術によるものとは限らない）を併用している場合に、2031 年以降も 2040 年までの必要な範囲内で 112 ビットセキュリティ強度のものを選択することを許容する。同様に、2051 年以降も 2060 年までの必要な範囲内で 128 ビットセキュリティ強度のものを選択することを許容する。

(c)　　移行完遂期間内に、よりセキュリティ強度の高い暗号技術又は鍵長への移行を完遂させることを前提として、利用する暗号処理が短期間で完結する場合（例：エンティティ認証）、又は既存の電子政府システムの継続利用やそれらとの互換性・相互接続性維持などの必要がある場合には、2030 年までは 112 ビットセキュリティ強度のものを、2050 年までは 128 ビットセキュリティ強度のものを選択することを許容する。

資料6-2

（別添）

＜参考５＞GPKI ブリッジ認証局の暗号移行予定（令和６年３月時点）

資料７

2024 年度暗号技術評価委員会活動計画（案）

1. 活動目的

CRYPTREC 暗号リストに掲載されている暗号技術や電子政府システム等で利用される暗号技術の安全性維持及び信頼性確保のために安全性及び実装に係る監視及び評価を行う。

2. 活動概要

(1) 暗号技術の安全性及び実装に係る監視及び評価

下記の通り、暗号技術の安全性に係る監視・評価 及び 実装に係る技術の監視・評価を実施する。

① CRYPTREC 暗号等の監視

国際会議等で発表される CRYPTREC 暗号リストの安全性及び実装に係る技術（暗号モジュールに対する攻撃とその対策も含む）に関する監視を行い、会議や ML を通して報告する。

② 電子政府推奨暗号リストからの運用監視暗号リストへの降格、並びに、推奨候補暗号リスト及び運用監視暗号リストからの危殆化が進んだ暗号の削除に係る検討

CRYPTREC 暗号リストの安全性に係る監視活動を継続的に行い、急速に危殆化が進んだ暗号技術やその予兆のある暗号技術の安全性について評価を行う。また、リストからの降格や削除、注釈の改訂が必要か検討を行う。

③ CRYPTREC 注意喚起レポートの発行

CRYPTREC 暗号リストの安全性及び実装に係る技術の監視活動を通じて、国際会議等で発表された攻撃等の概要や想定される影響範囲、対処方法について早期に公開することが望ましいと判断された場合、注意喚起レポートを発行する。

④ 推奨候補暗号リストへの新規暗号（事務局選出）の追加に係る検討

標準化動向に鑑み電子政府システム等での利用が見込まれると判断される暗号技術の追加を検討する。

⑤ 新技術等に関する調査及び評価

将来的に有用になると考えられる技術やリストに関わる技術について、安全性・性能評価を行う。必要に応じて、暗号技術調査ワーキンググループによる調査・評価、または、外部評価による安全性・性能評価などを行う。

- NIST の PQC 標準化が進行中であることから、引き続き、暗号技術調査ワーキンググループ（耐量子計算機暗号）を設置して、耐量子計算機暗号に関する最新

動向を把握する。また、「素因数分解の困難性に関する計算量評価」、「楕円曲線上の離散対数計算の困難性に関する計算量評価」の予測図の更新についても当該ワーキンググループで検討し、更新を行う。

(2) 暗号技術の安全な利用方法に関する調査（技術ガイドラインの整備、学術的な安全性の調査・公表等）

暗号技術を利用する際の技術面での注意点に関する調査、新技術の安全性・性能に関する調査・評価を行う。

特に、大規模な量子コンピュータに対する共通鍵暗号系の安全性に関する動向調査を実施する。本調査結果は 2019 年度に行われた「量子コンピュータが共通鍵暗号の安全性に及ぼす影響の調査及び評価」の更新版として公表する。

理由：NIST の Lightweight Cryptography 等の外部状況では、量子コンピュータを利用した攻撃に対する共通鍵暗号系の安全性についても言及されている。また、国際会議においても非常に活発な研究が行われている。これらを考慮し、大規模な量子コンピュータに対する共通鍵暗号系の安全性に関する調査が必要と考える。

3. 活動スケジュール

暗号技術評価委員会は、2 回の開催を予定する。

回	開催日	議案
第 1 回	2024 年 6 月中旬 ～7 月上旬	● 暗号技術評価委員会活動計画の具体的な進め方に関する審議 ● 各暗号技術調査ワーキンググループの活動計画(案)の審議 ● 耐量子計算機暗号の実利用に向けた必要事項の調査の具体的な進め方に関する審議
第 2 回	2025 月 2 月中旬 ～3 月上旬	● 暗号技術評価委員会活動報告(案)についての審議 ● 暗号技術調査ワーキンググループ（耐量子計算機暗号）の活動報告(案)の審議、及び、ガイドライン(案)に関する審議 ● 耐量子計算機暗号の実利用に向けた必要事項の調査結果に関する審議

以上

暗号技術評価委員会委員名簿

（五十音順、敬称略）

委員	青木 和麻呂	文教大学　准教授
委員	岩田　哲	名古屋大学　教授
委員	上原　哲太郎	立命館大学　教授
委員	大東　俊博	東海大学　教授
委員	國廣　昇	筑波大学　教授
委員	四方　順司	横浜国立大学　教授
委員	高木　剛	東京大学　教授
委員	手塚　悟	慶應義塾大学　教授
委員	花岡　悟一郎	国立研究開発法人産業技術総合研究所 サイバーフィジカル研究センター　首席研究員
委員	藤崎　英一郎	北陸先端科学技術大学院大学　教授
委員	本間　尚文	東北大学　教授
委員	松本　勉	横浜国立大学　教授
委員	松本　泰	日本ネットワークセキュリティ協会　フェロー
委員	山村　明弘	秋田大学　教授

（委員長は互選により決定）

資料８

2024 年度 暗号技術活用委員会活動計画（案）

1. 活動目的

暗号技術活用委員会では、情報システム全般のセキュリティ確保に寄与することを目的として、暗号の取り扱いに関する観点から、運用ガイドライン／ガイダンスの作成を行う。

2. 活動概要

（1） 暗号鍵管理ガイダンスの拡充

暗号鍵管理ガイドラインの拡充を目的として進めていた暗号鍵管理ガイダンスについて、2023 年度に引き続いて暗号鍵管理ガイダンス WG を設置し、2022 年度発行版では記載を見送った部分の拡充を行う。2022 年度版の内容見直しも含め、2024 年度完成を目標とする。

（2） 暗号利活用のための新たなガイダンスの作成

以下のテーマを想定した新たなガイダンスの作成に着手する。おおむね 2 年程度での完成を想定して執筆作業を行う。

- クラウドにおける鍵管理ガイダンス（日本クラウドセキュリティアライアンス（CSA）と共同での作成も検討）
  ※ クラウド利用者が留意すべき鍵管理を解説することを目的とする

3. 活動スケジュール

活用委員会の開催日程・議題については、以下のとおり、年 2 回の委員会開催を予定する。また、必要に応じてメール審議を実施する。

回	開催日	議案（予定）
メール審議	2024 年 4 月	■ 暗号鍵管理ガイダンス WG 活動計画の審議
第 1 回	2024 年 7 月上旬	■ 2024 年度活用委員会活動計画の確認 ■ 暗号鍵管理ガイダンス WG 活動計画の審議 ■ 「暗号利活用のための新たなガイダンス」についての検討
第 2 回	2025 年 3 月上旬	■ 暗号鍵管理ガイダンス審議 ■ 「暗号利活用のための新たなガイダンス」についての中間とりまとめ ■ 2024 年度暗号技術活用委員会活動報告案について

以上

資料９

暗号技術検討会
２０２３年度　報告書（案）

２０２４年３月

１．目次

１．はじめに

情報通信技術の急速な発展により、自動車、家電、医療、農業、工場など様々な分野で、あらゆるモノがネットワークに繋がるIoT社会が到来し、サイバー空間と実空間の高度な融合により、多様なニーズにきめ細やかに対応したモノやサービスを提供できる社会への産業構造の変化が進みつつある。一方で、IoT機器の普及に伴うサイバー攻撃の起点の増加や、サイバー攻撃自体の巧妙化・複雑化が続く中で、サイバー攻撃の影響が実空間にまで到達するリスクも増していくと考えられる。このような産業構造、社会の変化に伴うサイバー攻撃の脅威の増大に対応したセキュリティ確保が求められる中、暗号技術は既に様々な製品やサービスに組み込まれ、セキュリティを支える基盤技術として欠かせないものであるが、IoT機器から得られる大量のデータの流通・連携を支える上でも、その重要性は一層増すと考えられる。

このような社会の変化に伴い、CRYPTRECにおける、暗号アルゴリズムの安全性の評価及び監視を通じたセキュリティ確保、そして情報システム全体のセキュリティ確保に向けた暗号技術の利活用のための情報提供等の取組が果たすべき役割も大きくなっている。

2023年度は、暗号技術検討会の活動として、CRYPTREC暗号リスト更新案の承認等を行った。そして、各委員会の活動として、暗号技術評価委員会では、軽量暗号技術に対する実装性能評価及び標準化動向調査を実施し、軽量暗号ガイドラインを更新した。また、同委員会の下に設置した暗号技術調査WG（耐量子計算機暗号）において、耐量子計算機暗号に関する調査報告書及びガイドライン(いずれも2024年度版)の作成に向けた研究技術動向調査を行うとともに、「素因数分解の困難性に関する計算量評価」及び「楕円曲線上の離散対数計算の困難性に関する計算量評価」の予測図を更新した。暗号技術活用委員会では、2020年に公開したTLS暗号設定ガイドラインの改訂を行った。また、同委員会の下に設置した暗号鍵管理ガイダンスWGにおいて、2022年度に発行した「暗号鍵管理ガイダンス」の拡充に向けた検討を行った。これらの2023年度の活動の詳細については、国立研究開発法人情報通信研究機構及び独立行政法人情報処理推進機構が取りまとめる「CRYPTREC Report 2023」を参照いただきたい。

今後も暗号技術を用いた情報システム及び情報社会全体のセキュリティ確保のために、成果物の検討や情報発信等を行っていく所存である。

末筆であるが、暗号技術検討会及び関係委員会等に御協力いただいた構成員、オブザーバの方々をはじめ、関係者の皆様に心から謝意を表する次第である。

2024年3月

暗号技術検討会
座長　松本　勉

２．暗号技術検討会開催の背景及び開催状況

２．１．暗号技術検討会開催の背景

暗号技術は情報セキュリティの基盤技術であり、その安全性を暗号技術の専門家により技術的、専門的な見地から客観的に評価することが重要である。電子政府のセキュリティ確保のためには、安全性に優れた暗号技術を利用することが不可欠である。

このため、総務省及び経済産業省は、客観的な評価により安全性及び実装性に優れると判断される暗号技術をリスト化し、各府省に対してその利用を推奨することにより、高度な安全性と信頼性に支えられ、国民が安心して利用できる電子政府の構築に貢献することを目指し、2001年5月に最初の暗号技術検討会を開催した。

その後、2021年のデジタル庁発足に伴いデジタル庁が加わり、デジタル庁、総務省及び経済産業省は、電子政府推奨暗号等の安全性を評価・監視し、暗号技術の更なる普及促進を行うため、継続的に暗号技術検討会を開催している。

暗号技術検討会での検討を経て、2003年2月に策定された電子政府推奨暗号リストは、2013年3月にCRYPTREC暗号リストとして改定され、2023年3月に再改定された。

２．２．CRYPTRECの体制

CRYPTRECとは、Cryptography Research and Evaluation Committeesの略であり、デジタル庁、総務省及び経済産業省が共同で開催する暗号技術検討会（座長：松本勉横浜国立大学教授）と、国立研究開発法人情報通信研究機構（NICT）及び独立行政法人情報処理推進機構（IPA）が共同で開催する委員会から構成される暗号技術評価プロジェクトをいう。

2023年度は、暗号技術検討会では、CRYPTREC暗号リスト更新案の承認等を行った。暗号技術評価委員会では、軽量暗号技術に対する実装性能評価及び標準化動向調査を実施し、軽量暗号ガイドラインを作成した。また、同委員会の下に設置された暗号技術調査WG（耐量子計算機暗号）において、耐量子計算機暗号に関する調査報告書とガイドライン（それぞれ2024年度版）の作成に向けた研究技術動向調査を行うとともに、「素因数分解の困難性に関する計算量評価」、「楕円曲線上の離散対数計算の困難性に関する計算量評価」の予測図を更新した。暗号技術活用委員会では、2020年に公開したTLS暗号設定ガイドラインの改訂を行った。また、同委員会の下に設置された暗号鍵管理ガイダンスWGにおいて、2022年度に発行した「暗号鍵管理ガイダンス」の拡充に向けた検討を行った。

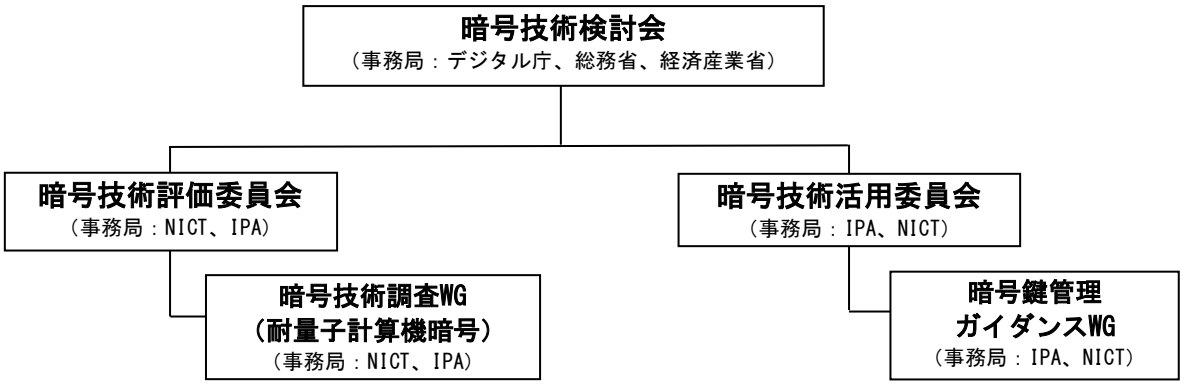


図2.2-1　CRYPTREC体制図（2023年度）

２．３．暗号技術検討会の開催実績

2023年度、暗号技術検討会は、下記内容について検討を行うため１回開催した。

【第 1 回】2024年3月26日（火）10:00～12:00

（主な議題）

・2023年度暗号技術評価委員会 活動報告について【報告】
・軽量暗号に関する外部評価報告書（案）CRYPTREC暗号技術ガイドライン（軽量暗号）（案）について【承認】
・2023年度暗号技術活用委員会 活動報告について【報告】
・TLS暗号設定ガイドライン（案）について【承認】
・Triple DES等の取り扱いに係る暗号技術活用委員会からの意見について【報告】
・CRYPTREC暗号リストの更新について【承認】
・電子署名法特定認証業務の暗号基準の改正スケジュールについて【報告】
・2024年度暗号技術評価委員会活動計画（案）について【承認】
・2024年度暗号技術活用委員会活動計画（案）について【承認】
・暗号技術検討会 2023年度 報告書（案）について【承認】

（概要）

・暗号技術評価委員会についてNICTより2023年度の活動報告が行われた。
・CRYPTREC暗号技術ガイドライン案（軽量暗号）（案）についてNICTより説明が行われ、原案のとおり承認された。
・暗号技術活用委員会についてIPAより2023年度の活動報告が行われた。
・TLS暗号設定ガイドライン（案）についてIPAより説明が行われ、原案のとおり承認された。
・Triple DES等の取り扱いに係る暗号技術活用委員会からの意見についてIPAより報告が行われた。
・CRYPTREC暗号リスト（更新事務局案）について事務局より説明が行われ、原案のとおり承認された。
・電子署名法特定認証業務の暗号基準の改正スケジュールについて、デジタル庁より報告が行われた。
・2024年度暗号技術評価委員会活動計画（案）についてNICTより説明が行われ、原案のとおり承認された。
・2024年度暗号技術活用委員会活動計画（案）についてIPAより説明が行われ、原案のとおり承認された。
・暗号技術検討会 2023年度 報告書（案）について事務局より説明が行われ、議論結果を追記することとした上で承認された。

３．各委員会の活動報告

３．１．暗号技術評価委員会

３．１．１．活動の概要

暗号技術評価委員会は、CRYPTREC暗号リストに掲載されている暗号技術や電子政府システム等で利用される暗号技術の安全性維持及び信頼性確保のために安全性及び実装に係る監視及び評価を行う。主要な検討課題は以下のとおりである。

- 暗号技術の安全性及び実装に係る監視及び評価
- 暗号技術の電子政府推奨暗号リストからの降格
- 暗号技術に関する注意喚起レポートのCRYPTRECホームページへの公表
- 推奨候補暗号リストへの新規暗号（事務局選出）の追加
- 新世代暗号に係る調査

また、CRYPTREC暗号リストとは別の文書として、耐量子計算機暗号、及び、軽量暗号に関するガイドラインを作成する。基本方針は以下のとおりである。

- 耐量子計算機暗号に関するガイドライン（2024年度版）を作成するため、2023-2024年度に、耐量子計算機暗号に関するワーキンググループを設置する。
- 軽量暗号に関するガイドラインについては、2016年度に作成した「CRYPTREC暗号技術ガイドライン（軽量暗号)」の更新のため、2023年度は、NIST Lightweight Cryptography Project で最終選考方式に選出されたAsconについて実装性能評価及び標準化動向調査を行った。そして、昨年度までに実施した調査と、今年度の調査を含め、2024年3月に現ガイドラインを更新した。

これらの課題について2023年度に行った具体的な検討内容を、以下のとおり報告する。

３．１．２．暗号技術の安全性及び実装に係る監視及び評価

学会等での情報収集に基づくCRYPTREC暗号等の監視活動を行った。監視報告の詳細については、CRYPTREC Report 2023（暗号技術評価委員会報告）に掲載する。

３．１．３．暗号技術調査ワーキンググループ（耐量子計算機暗号）

大規模な量子コンピュータが実用化され、その量子コンピュータを用いた攻撃に対しても安全性を保てると期待される暗号（耐量子計算機暗号:PQC）の研究開発及び標準化などが各国で進められている。そこで、2020年度第2回暗号技術検討会において、耐量子計算機暗号ガイドラインを作成するために暗号技術調査ワーキンググループ(耐量子計算機暗号)（以下：PQC WG）を設置することが承認された。また、「素因数分解の困難性に関する計算量評価」、「楕円曲線上の離散対数計算の困難性に関する計算量評価」の予測図の更新を PQC WG で実施することが承認された。

2022年度に耐量子計算機暗号に関する調査報告書とガイドライン（それぞれ2022年度版）を作成・公開したが、その後もNISTをはじめとする世界各国の機関において耐量子計算機暗号の選定・標準化活動が継続されており、情勢が流動的であることを鑑み、2023-2024年度の2年間で再度、耐量子計算機暗号に関する調査報告書とガイドライン（それぞれ2024年度版）を作成することが承認された。

2023年度のPQC WGの活動では調査報告書・ガイドライン作成の準備として、2022年度版が出版された以降の研究技術動向に関して調査を行った。ガイドライン執筆方針の基本的な部分は2022年度版調査報告書・ガイドラインを踏襲している。

- 耐量子計算機暗号のスコープ

  公開鍵暗号を中心にまとめる。

- 耐量子計算機暗号に関する現状調査

  ガイドライン及び調査報告書に記載する耐量子計算機暗号を5分類とする。2022年度版とは異なり、耐量子計算機暗号の活用方法を調査報告書・ガイドラインの両方に記載。導入の章の内容について2022年度版では同じものであったが、ガイドラインでは技術的な詳細を省き簡略化する。これらの項目に関する情報を調査した。

- ガイドライン及び調査報告書の目次案

  i. 導入

  ii. PQC の活用方法（ガイドラインにのみ記載）

  iii. 格子に基づく暗号技術

  iv. 符号に基づく暗号技術

  v. 多変数多項式に基づく暗号技術

  vi. 同種写像に基づく暗号技術

  vii. ハッシュ関数に基づく署名技術

- iii 章以降の構成（A 章の場合）

  A.1. 安全性の根拠となる問題（例：LWE問題、シンドローム復号問題）

  A.2. 代表的な暗号方式（例：Regev暗号、McEliece暗号）

  A.3. 主要な暗号方式

  A.3.1. 暗号方式1（例：CRYSTALS-KYBER, Classic McEliece）

  A.3.2. 暗号方式2

A.3.3. 暗号方式3

・・・

A.4. まとめ

そして、2023年度第一回暗号技術評価委員会において、2023年度のPQC WGの活動として下記2点について実施する活動計画が承認された。

- 耐量子計算機暗号に関し、NISTのPQC標準化において第4ラウンドが進行中であることをはじめ技術開発、標準化活動が引き続き世界的に活発であることから、動向を今後2年間かけて調査・把握し、調査報告書・ガイドラインの改定を行う。
- 「素因数分解の困難性に関する計算量評価」、「楕円曲線上の離散対数計算の困難性に関する計算量評価」の予測図を更新する。

それらの成果（３．１．３．１～３．１．３．２節）は2023年度第二回暗号技術評価委員会にて報告され、了承された。

３．１．３．１．耐量子計算機暗号に関する調査報告書・ガイドラインの作成方針

2024年度版の内容は、2022年度版の調査報告書・ガイドラインをベースとし、技術の進展に伴う部分を追記・修正する。なお、著者の著作権の関係から調査報告書・ガイドラインともに改定ではなく新規の扱いとし、過去の版と区別する必要がある際には（2024年度版）のように年度を明示する。

耐量子計算機暗号調査報告書・ガイドライン

耐量子計算機暗号ガイドラインは、暗号理論に精通していない利用者を対象とし、耐量子計算機暗号に関する調査報告書は、暗号理論の研究者や技術者を対象とする。基本的には耐量子計算機暗号ガイドラインは調査報告書から技術的詳細を省き、その一部を抜粋したものする。2022年度版とは異なり、耐量子計算機暗号の活用方法を調査報告書・ガイドラインの両方に記載する。導入の章の内容について2022年度版では同じものであったが、ガイドラインでは技術的な詳細を省き簡略化する。

耐量子計算機暗号ガイドライン及び調査報告書に記載する暗号方式の選定基準

公開鍵暗号方式である主要な耐量子計算機暗号（NIST PQC標準化への提案方式等）を記載するが、対象となる暗号方式は PQC WG によって承認されたものである。

ガイドラインの章立て

1　はじめに

2　PQC の活用方法

3　格子に基づく暗号技術

4　符号に基づく暗号技術

5　多変数多項式に基づく暗号技術

6　同種写像に基づく暗号技術

7　ハッシュ関数に基づく署名技術

３．１．３．２．「素因数分解の困難性に関する計算量評価」、「楕円曲線上の離散対数計算の困難性に関する計算量評価」の予測図の更新

「素因数分解の困難性に関する計算量評価」、「楕円曲線上の離散対数計算の困難性に関する計算量評価」の予測図（以下単に「予測図」という。）は公開鍵暗号方式のセキュリティパラメータの選択について検討を行うため、2006年度に設置された暗号技術調査WG（公開鍵暗号）において作成された。2019年度暗号技術評価委員会において、今後の予測図の取扱いについて審議し対応方針（「今後の予測図の取扱い」「今後の公開鍵暗号のパラメータ選択」）を決定した。2023年度において、対応方針の説明文をより一般の読者に読みやすくなるよう、以下のとおり修正した。

予測図の取扱い対応方針

〈今後の予測図の取扱い〉

(1) いわゆるムーアの法則を仮定して外挿線を年度末からプラス20年後まで従来通り直線で引き、評価に大きな変動がないと考えられる限りにおいては、安全サイドに倒した評価※として予測図を当面の間更新していく。

〈今後の公開鍵暗号のパラメータ選択〉

(2) 公開鍵暗号のパラメータ選択に関する対応方針については、安全性以外にも相互接続性など、運用上の観点もあるため、暗号技術評価委員会だけではなく、暗号技術検討会、暗号技術活用委員会や関係各所などを含めて検討する。

※予測図は各国・国際標準化機関等により示されている主要な暗号技術の安全性基準と比較すると、より現状に則した評価であり、危殆化時期がそれらよりも先に延びるものとなっている。

予測図の更新について

素因数分解問題の困難性および楕円曲線上の離散対数問題の困難性に関する計算量評価に大幅な進展はなかったため、TOP500.orgにおける2023年6月・11月のベンチマーク結果を追加して予測図の更新を行った（図3.2-1及び図3.2-2）。

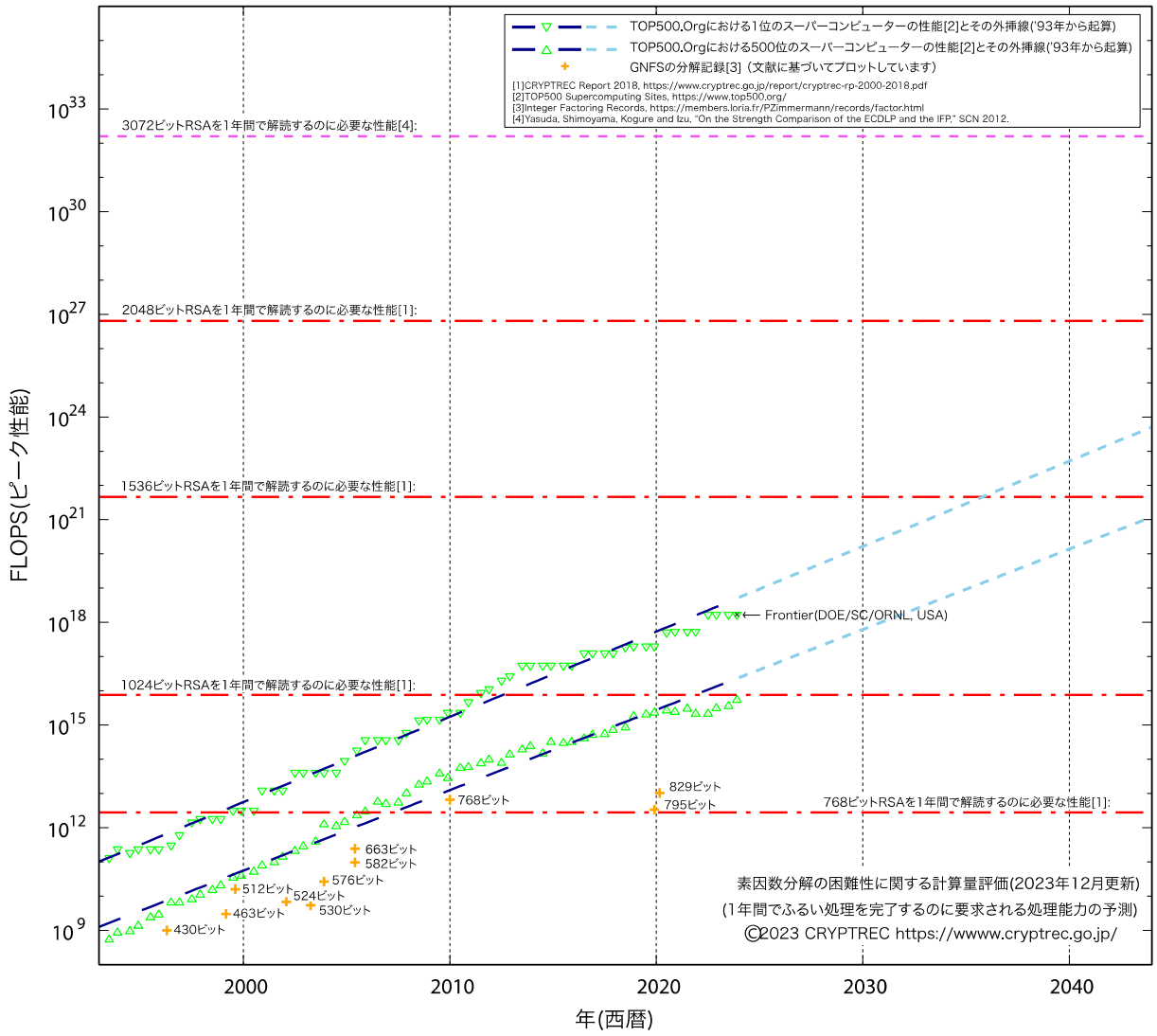


図3.2-1：素因数分解の困難性に関する計算量評価（2023年12月更新）[1]

[1] スーパーコンピューターの性能の伸びに関する外挿線は僅かであるが鈍化してきている。

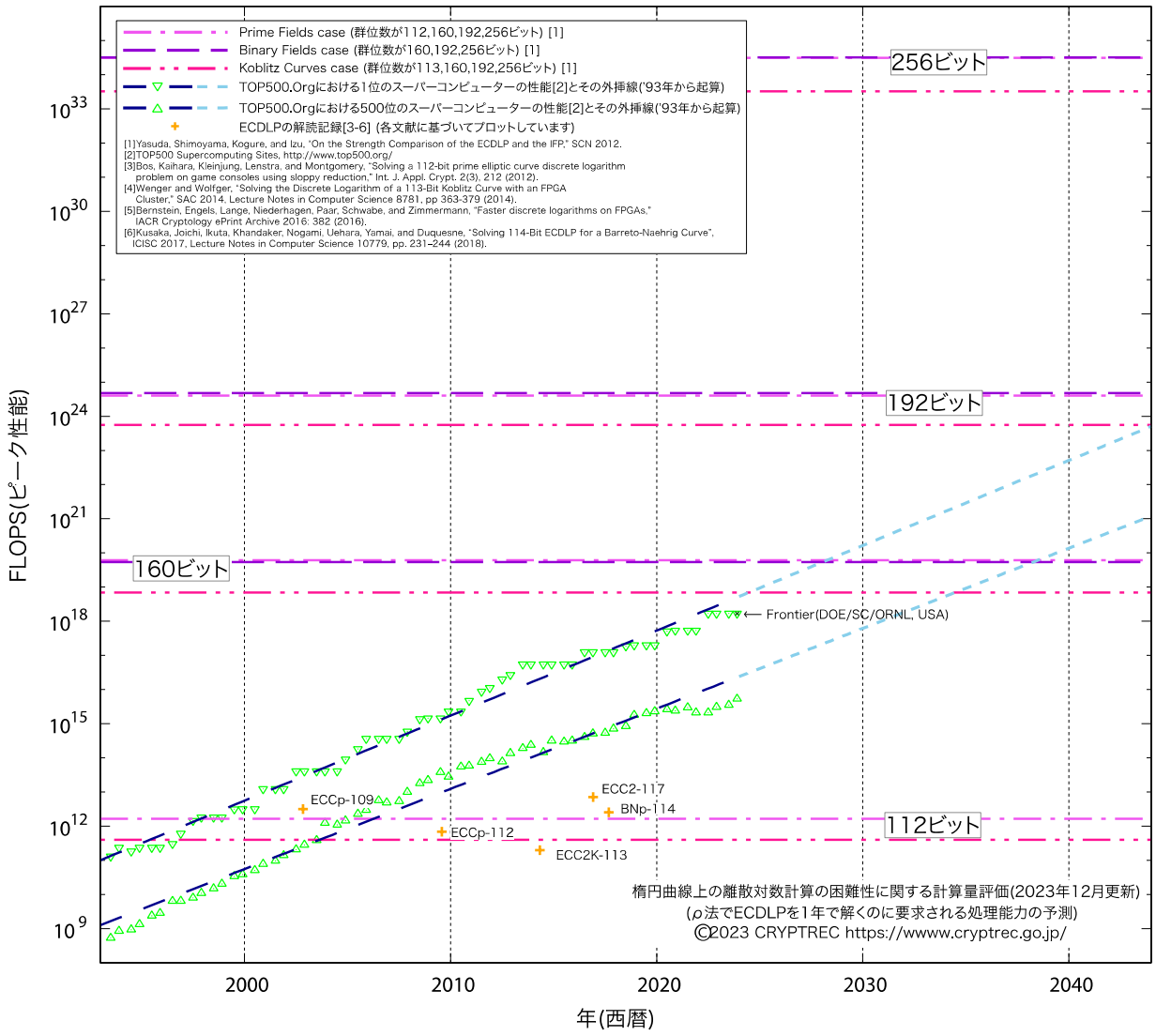


図3.2-2：楕円曲線上の離散対数計算の困難性に関する計算量評価（2023年12月更新）[2]

３．１．４．「CRYPTREC暗号技術ガイドライン（軽量暗号）」更新に関わる活動

３．１．４．１．背景

2019年度に設置された量子コンピュータ時代に向けた暗号の在り方検討タスクフォースにて、「CRYPTRECにおいて、軽量暗号はCRYPTREC暗号リストには組み込まず、別途ガイドラインという形で取り扱う」ことが決定され、2020年度第二暗号技術検討会にて、2016年度に作成した「CRYPTREC暗号技術ガイドライン（軽量暗号）」（以下、「2016年度版ガイドライン」という）を2023年度中を目処に更新することが承認された。2021年度第二回暗号技術評価委員会においてその更新方針が承認された。

[2] スーパーコンピューターの性能の伸びに関する外挿線は僅かであるが鈍化してきている。

当該更新方針に従い、今年度は、NIST軽量暗号プロジェクト（NIST Lightweight Cryptography Project． 以下、「NIST LWC」という）の選定方式Asconを対象とした実装性能評価を外部評価により実施した。また、軽量暗号に関わる NIST公開文書やISO/IECなどの標準化動向に関わる調査を外部評価により実施した。その外部評価実施内容を報告する。

そして、2021年度から2023年度にかけて実施した外部評価報告書を基に「CRYPTREC暗号技術ガイドライン（軽量暗号）」2023年度版を作成した。

３．１．４．２．評価・調査実施概要

Asconの実装性能評価と標準化動向調査について、以下のとおり外部評価を実施した。

- 実装性能評価：NIST LWCで選定されたAsconの実装性能（ハードウェア及びソフトウェア）に関する調査及び評価を実施した。
- 標準化動向調査：NIST LWCでAsconを選出するに至った選考過程や選考理由に関する調査、並びにAsconの標準化動向(IETF、W3C、ISO/IEC、ITU-T、Global Platformの5団体)に関する調査を実施した。

３．１．４．２．１．調査結果概要

[軽量暗号Asconの実装性能に関する調査及び評価結果概要]

Asconの実装性能について、物理攻撃耐性を持つ実装性能評価も含めた調査結果は以下の通りである。

- 実装面における特徴：Asconは認証暗号モードとハッシュモードに対応する軽量な暗号アルゴリズムであり、以下の特徴を有する。
  - 5ビットのコンパクトなS-boxを繰り返し使用
  - ラウンド処理を並列実装可能
  - 概ね全ての処理を同じラウンド処理の繰り返しで実現可能

  これらの特徴を鑑み、実装コストと処理性能のトレードオフの観点から高い柔軟性がある。
- 物理攻撃対策：代表的な物理攻撃対策技術として、Threshold Implementation (TI)とその発展的技術であるDomain Oriented Masking (DOM)がある。これらの物理攻撃対策を施したAsconの実装評価に関する調査結果をまとめた。
- 物理攻撃耐性評価：代表的な物理攻撃耐性評価技術として、相関電力解析（CPA)、Test Vector Leakage Assessment (TVLA)、テンプレート攻撃（TA)、などがある。 これらの評価技術を使用したAsconの物理攻撃耐性評価に関する調査結果をまとめた。

[標準化動向調査結果概要]

- 最終ラウンドにおける評価基準と選定プロセス：NISTはステータスレポートNISTIR 8454を発行し、最終ラウンドにおける評価基準や選定プロセスについて明らかにした。評価基準と選定プロセスの対応関係は次の表のとおり。

評価基準	選定プロセス
暗号学的安全性	第三者による安全性評価、耐量子安全性
制約のある環境下におけるソフトウェア及びハードウェアでの実装性能	ベンチマーク
サイドチャネル攻撃	ベンチマーク
知的財産	知的財産に関する声明
その他	バリエーション、設計の微調整

- Asconが選出されるに至った選定指標や評価の観点：次の表に示す選定指標においてAsconが評価された。

選定指標	評価
機能	● 認証暗号モードとハッシュモードに加え、XOF[3]機能を含む。 ● 暗号学的置換ベースの設計により、追加機能の実装コストが少ない。
成熟度	● CAESAR コンペティションのユースケース１（軽量アプリケーション） ● 最終ラウンドにおける設計の微調整なし
安全性	● 第三者による安全性評価が最も多いファイナリスト ● 他ファイナリストよりも先行的に安全性評価が行われているにも関わらず依然として高い安全性を維持
実装性能	● ソフトウェアとハードウェアの両面で非常に優れた性能を発揮 ● 実装コストと処理性能の様々なトレードオフをサポートする柔軟性 ● 物理攻撃対策にかかる追加コストが低い。

- 標準化動向：Asconに関するNIST以外の組織での標準化動向についてまとめる。調査対象の標準化団体は、IETF、W3C、ISO/IEC、ITU-T、Global Platformの5団体である。2024年3月現在、IETFを除く4団体においてAsconに関する標準化動向は確認できなかった。IETFでは以下でAsconが取り上げられている。
  - インターネットドラフト“Secure UAS Network RID and C2 Transport”
  - インターネットドラフト“Properties of AEAD algorithms”
  - IETF 117におけるTLS WGでの発表“New Post-Quantum Signatures on the Horizon”

  その他、産業界でもAsconを利用可能な環境を提供するような動向がある。

[3] eXtendable Output Function：可変長出力関数

３．１．４．２．２．外部評価報告書に対する暗号技術評価委員会の見解

2件の外部評価報告書は、今年度の調査対象であるAsconの実装性能及び標準化動向に関する技術動向調査として十分な内容を含んでいると考えられることから、本報告書をCRYPTRECの技術調査報告書とすることが了承された。

３．１．４．３．2023年度版ガイドラインの作成

以下の手順により2023年度版ガイドラインを作成した。

- 2021年度から2023年度にかけて実施した外部評価に基づき、事務局にて2016年度版ガイドラインの更新を行い、完成したものを2023年度版ガイドライン（ドラフト版）とする。
- 事務局が作成した2023年度版ガイドライン（ドラフト版）について、掲載内容の適切性や情報の過不足などを2名の外部有識者によりレビューいただくとともに、第二回暗号技術評価委員会にてレビュー結果を報告いただく。
- レビュー結果に基づき、事務局にて2023年度版ガイドライン（ドラフト版）の更新を行う。更新内容について外部有識者に了解頂いたものを最終的な2023 年度版ガイドライン（案）とする。

３．１．４．３．1. 2023年度版ガイドライン（ドラフト版）の作成

2023年度版ガイドライン（ドラフト版）の作成にあたり、以下の表に示す目次の赤字部分を新たに更新した。黒字部分は、2016年版ガイドラインと同一である。

章		章タイトル	概要
第1章		はじめに	導入、謝辞
第2章		軽量暗号とその活用法	
	2.1	軽量暗号とは	定義、代表的な軽量暗号
	2.2	軽量暗号の標準化動向	● CAESAR コンペティション ● NIST LWC ● 他標準化団体における Ascon の検討状況
	2.3	軽量暗号はどこに使えるのか	家電、スマートテレビ、スマート農業、医療、自動車、等での活用例
	2.4	どんな軽量暗号、パラメータを選べばいいか	一般的方針、鍵長・ブロック長の選択、利用シナリオ、等
	2.5	軽量暗号活用例と効果	家電、スマートテレビ、スマート農業、医療、自動車、等での効果
第3章		軽量暗号の実装性能	
	3.1	ブロック暗号の実装性能	12種類の軽量ブロック暗号に対するハードウェア・ソフトウェア実装評価
	3.2	認証暗号の実装性能	10種類の軽量認証暗号に対するソフトウェア実装評価

	3.3	ASCONの実装性能	● ハードウェア実装性能 ● ソフトウェア実装性能 ● 物理攻撃耐性評価
第4章		代表的な軽量暗号	
	4.1	ブロック暗号	各技術分野の各方式に関する仕様等の調査結果
	4.2	ストリーム暗号	
	4.3	ハッシュ関数	
	4.4	メッセージ認証コード	
	4.5	認証暗号	
付録A		Asconの物理攻撃耐性	
	A.1	サイドチャネル攻撃対策	● Threshold Implementation ● Domain Oriented Masking
	A.2	サイドチャネル解析・漏洩評価	● 相関電力解析 ● 故障利用攻撃 ● Test Vector Leakage Assessment ● テンプレート攻撃
付録B		CAESAR final portfolio: AEGIS, COLM	AEGIS、COLMに関する仕様等**エラー！ブックマークが定義されていません。エラー！ブックマークが定義されていません。**の調査結果
付録C		NIST LWCファイナリスト（Asconを除く）	Asconを除くNIST LWCファイナリスト9方式に関する仕様等**エラー！ブックマークが定義されていません。**の調査結果

３．１．４．３.2. 外部有識者によるレビュー概要

2023年度版ガイドライン（ドラフト版）に対し、本間尚文氏(東北大学)と峯松一彦氏（日本電気株式会社）にその掲載内容の適切性や情報の過不足などをレビューいただいた。レビュー内容の概要は以下のとおりである。

- 本間氏によるレビュー：主に第1章から第3章の更新内容に関するレビューを実施し、1箇所の構成変更（付録Aを新たに追加）といくつかの確認が必要と思われる箇所を除き、改定内容・構成が妥当である。

- 峯松氏によるレビュー：主に第4章と付録B・Cの更新内容に関するレビューを実施し、全般的に記載内容に関して大きな疑義を呈する箇所はなく、改定内容が妥当である。

３．１．４．３.3. ガイドラインの作成外部評価報告書に対する暗号技術評価委員会の見解

2023年度版ガイドラインは、軽量暗号に関する最新動向を踏まえて2016年度版ガイドラインを更新したものであり、暗号技術ガイドラインとして十分な内容を含んでいると考えられる。また、外部有識者によるレビューより更新内容の妥当性が評価されたことから、2023年度版ガイドライ

ン（案）が暗号技術ガイドラインとすることが了承された。

３．１．５．暗号技術評価委員会の開催実績

2023年度、暗号技術評価委員会は計2回開催した。各回会合の概要は表3.2-5のとおりである。

表3.2-5　暗号技術評価委員会の開催状況

回	開催日	議案
第1回	2023年7月3日	■　暗号技術評価委員会活動計画の具体的な進め方についての審議 ■　暗号技術調査ワーキンググループ（耐量子計算機暗号）の活動計画案の審議 ■　軽量暗号ガイドラインの更新について、公開スケジュールに関する報告 ■　ガイドライン作成にあたり、外部評価を行うこと、外部有識者によるレビューを行うことに関する審議 ■　監視状況報告
第2回	2024月2月27日	■　暗号技術調査ワーキンググループ（耐量子計算機暗号）の活動内容の報告 ■　軽量暗号ガイドラインに係る技術動向調査結果の報告 ■　軽量暗号ガイドラインに関するレビューの報告 ■　軽量暗号ガイドラインに関する審議 ■　監視状況報告 ■　CRYPTREC Report 2023作成について ■　CRYPTRECシンポジウム開催について

３．２．暗号技術活用委員会

３．２．１．活動の概要

2023年度の活動概要は以下の通りである。詳細については、CRYPTREC Report 2023暗号技術活用委員会報告[4]を参照されたい。

(1) TLS暗号設定ガイドラインの改訂

現在の「TLS暗号設定ガイドライン(Ver3.0.1)」の公開(2020年7月)以降、CRYPTRECではCRYPTREC暗号リストの改定、「暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準（以降「強度要件設定基準」と表記)」の策定を行っている。このため、これらCRYPTREC成果の取り込み及び３年間のTLSに関するRFC規格化や技術環境の変化なども踏まえ、本ガイドラインを改訂する。

(2) 暗号鍵管理ガイダンスの拡充

暗号鍵管理ガイドラインの拡充を目的として進めていた暗号鍵管理ガイダンスについて、2021年度・2022年度に引き続いて暗号鍵管理ガイダンスWGを設置し、2022年度発行版では記載を見送った部分の拡充を行う。2022年度版の内容見直しも含め、2024年度完成を目標とする。

３．２．２．TLS暗号設定ガイドラインの改訂

現行のTLS暗号設定ガイドライン（v3.0.1）からの一番大きな変更点は、強度要件設定基準の策定に伴い、安全性の基準として「鍵長」で表現されていた部分を「ビットセキュリティ」で表現するようにしたところである。これにより、「鍵長256ビットの楕円曲線」との要件に「X25519の楕円曲線」が許容されるか否かについて、明確に許容されることとなった。

また、３年間のTLS規格化や技術環境の変化なども踏まえ、主に以下の観点での議論を行い、必要な改訂を行うこととした。

① CRYPTREC暗号リスト改定等を踏まえたTLSでの利用を推奨／禁止する暗号アルゴリズムの改訂
② 「セキュリティ例外型」の取り扱い
③ DHEの強度設定について推奨要件の改訂要否
④ その他、改訂することが望ましい項目

作成したガイドラインv3.1ドラフト案に対する主な改訂内容を表にまとめる。なお、今回の改訂では、推奨の設定内容に大きな影響を与える項目がないことから、バージョン名はv3.1とすることとした。

[4] CRYPTREC Report 2023 暗号技術活用委員会報告, https://www.cryptrec.go.jp/promo_cmte.html

表　TLS暗号設定ガイドラインの主な改訂内容

項目	改訂内容概要
「鍵長」基準から「ビットセキュリティ」基準への変更	強度要件設定基準に従い、現行版（v3.0.1）の鍵長をそのままビットセキュリティ基準に置き換えた。なお、利用する楕円曲線は「強度要件設定基準」に記載のものから選択することを明記した。 また、セキュリティ例外型のDH／DHEの1024ビット鍵長は、対応するビットセキュリティ基準が存在しないため、鍵長表現のままとした。
CRYPTREC暗号リスト改定等を踏まえたTLSでの利用を推奨／禁止する暗号アルゴリズムの更新	改定されたCRYPTREC暗号リストによりリストの位置づけが変更されたアルゴリズム、及び3年間のTLSに関するRFC規格化や技術環境の変化などにより変更が必要と考えるアルゴリズムについて、以下のように改訂する。 ● サーバ証明書における DSA の利用推奨を削除する ● サーバ証明書における RSA-PSS の利用推奨を追加する ● EdDSA はサーバ証明書、暗号スイートとも利用推奨をしない ● 暗号スイートでの利用禁止暗号アルゴリズムに SM2（署名）、SM3、SM4 を追加する
「セキュリティ例外型」の取り扱い	移行を明確に促す観点から移行期限を明記した以下の表現に強化する。 「本ガイドラインで記載されているセキュリティ例外型の設定内容は、2029年度を目途とした改訂時に終了させる予定である。速やかに推奨セキュリティ型への移行を完了させるべきである。」
DHE の強度設定について推奨要件の改訂要否	以下のように改訂する。 ● 高セキュリティ型は 112 ビットセキュリティから 128 ビットセキュリティ以上に変更する。推奨セキュリティ型とセキュリティ例外型は変更しない。
その他	その他の主な改訂内容として以下のものがある。 ● 「Certificate Transparency」に関する節の追加 ● 「ブラウザを利用する際に注意すべきポイント」について、Microsoft、Google、Mozilla、Apple の各ブラウザの最新情報を反映 なお、IoT の普及という観点から組み込み系に向けた補足ドキュメントを検討してはどうかとの意見があったが、本ガイドラインの主たる読者層とは対象が異なると想定されることから、今後の新規ガイドラインの作成や拡充の候補として検討することになった。

３．２．３．暗号鍵管理ガイダンスの拡充

2022年度に発行した「暗号鍵管理ガイダンスVer.1.0」と今回作成中の「暗号鍵管理ガイダンス拡充分」は、「暗号鍵管理システム設計指針（基本編）」の章構成に対応して表のとおりである。なお、

下表はガイダンス拡充分を別冊とした場合の章構成であり、暗号鍵管理ガイダンスVer.1.0にマージするか別冊とするかは2024年度の執筆状況を踏まえて決定する。

表　暗号鍵管理ガイダンスの章構成

暗号鍵管理システム設計指針（基本編）	暗号鍵管理ガイダンスVer.1.0（2022年度発行）	暗号鍵管理ガイダンス拡充分（別冊時）
1. はじめに	1. はじめに	1. はじめに
2. 暗号鍵管理の在り方	（1章に集約）	（1章に集約）
3. 本設計指針の活用方法	（1章に集約）	（1章に集約）
4. 暗号鍵管理システム(CKMS)の設計原理と運用ポリシー		2. 暗号鍵管理システム(CKMS)の設計原理と運用ポリシー
5. 暗号アルゴリズム運用のための暗号鍵管理オペレーション対策	2. 暗号アルゴリズム運用のための暗号鍵管理オペレーション対策	
6. 暗号アルゴリズムの選択	3. 暗号アルゴリズムの選択	
7. 暗号アルゴリズム運用に必要な鍵情報の管理	4. 暗号アルゴリズム運用に必要な鍵情報の管理	
8. 暗号鍵管理デバイスへのセキュリティ対策		3. 暗号鍵管理デバイスへのセキュリティ対策
9. 暗号鍵管理システム(CKMS)のオペレーション対策		4. 暗号鍵管理システム(CKMS)のオペレーション対策

2023年度は、「暗号鍵管理システムの設計原理と運用ポリシー」及び「暗号鍵管理デバイスへのセキュリティ対策」について、記載すべき内容をダイジェスト形式で整理した。整理した主な概要は以下のとおりである。

【トイモデル】

暗号鍵管理システムのシンプルなモデル（トイモデル）を例示し、それに対する各検討項目への対応例を説明するためのモデルとして、「暗号鍵管理システムの設計原理と運用ポリシー」及び「暗号鍵管理デバイスへのセキュリティ対策」の節に記載するトイモデルを、IoT製品向けのプライベートCAシステムとすることに決定した。

【暗号鍵管理システムの設計原理と運用ポリシーでの解説・考慮点の主な概要】

節番号	FR番号	「解説・考慮点」の説明概要
4.1節 CKMSセキュリティポリシー	A.01-A.05	セキュリティポリシーとはCKMSが実現するセキュリティ機能や運用方針の概要を定めたもの。CKMSを利用するシステムやCKMSが構築されるIT環境のポリシーなどと矛盾がないことが前提

4.2節 情報管理ポリシー等からの要求事項	A.06	個人の説明責任が求められるケース（監査、リスクマネジメントの観点）を想定してCKMSでのサポートメカニズムを記載
	A.07-A.13	匿名性、連結不可能性、観測不可能性のサポート有無とサポートする場合のメカニズムを記載。一般に、匿名性、連結不可能性、観測不可能性を要求するのは特殊なケース
4.3節 ドメインのセキュリティポリシー	A.14-A.19	異なるセキュリティドメイン間での鍵情報の交換がなければ対象外。GPKIは異なるセキュリティドメイン間での鍵交換の事例
	A.22-A.26	マルチレベルのセキュリティドメインでの鍵情報の交換がなければ対象外。一般に、マルチレベルのセキュリティドメインでの鍵情報の交換は特殊なケース
4.4節 CKMSにおける役割と責任	A.27-A.28	CKMSの運用に関わるエンティティを定め、エンティティに割り当てる役割と実行できる鍵情報の管理機能へのアクセス権（権限）を定義する。
	A.29-A.31	不必要な権限の割り当てや権限の分離が不十分な場合、内部犯行を誘発するリスクがある
4.5節 CKMSの構築環境及び実現目標	A.32	CKMSを構成する主要なデバイスおよびコンポーネントの一式を定める
	A.33-A.36	CKMSが要求する時刻の精度や利用する権威時刻ソース、第三者タイムスタンプの要求有無を定める
	A.39-A.42	初期及び将来を想定してユーザ数やCKMS性能面の目標、負荷増大時の対応策を定める
	A.43-A.46	CKMS内デバイスやCKMS間の相互運用を可能とするため、インタフェース、プロトコル、コマンド仕様を定める
	A.47-A.50	使いやすいユーザインタフェースを検討し、ヒューマンエラーを防止する
	A.51-A.53	どのような商用既製品を利用してどのようなセキュリティ機能を実行するかを定める
4.6節 標準／規制に対する適合性	A.54-A.55	暗号アルゴリズム、暗号モジュール、セキュリティ認証などの標準への準拠性を明確にする
	A.57	CKMSが使用される国家・地域の法的規制を明確にする。欧州のサイバーセキュリティ法、中国のデータセキュリティ法、各国のデータ規制など
4.7節 将来的な移行対策の必要性	A.58-A.61	CKMSは暗号アルゴリズムのセキュリティライフタイムを超えたサービス提供や、危殆化により、暗号アルゴリズムの置き換えが必要になる。そのため、複数の暗号アルゴリズムや異なる鍵長をサポートするケースも多い
	A.62-A.69	技術の進歩をウォッチすると共に、予め潜在的な脅威に対する影響評価の実施を推奨する

【暗号鍵管理デバイスへのセキュリティ対策での解説・考慮点の主な概要】

節番号	FR番号	「解説・考慮点」の説明概要
8.1節 鍵情報へのアクセスコントロール	E.01-E.04	暗号モジュールの各機能の実行を認可されたエンティティに限定する。実行権を管理するアクセスコントロールシステム(ACS)は暗号モジュールと連動して動作する
	E.05	ACSによるエンティティ識別、認証、認可の粒度や機能を明確にする
	E.07-E.20	暗号モジュールとは､暗号境界内で暗号処理を実行するハードウェアもしくはソフトウェアの集合｡暗号境界内で利用される暗号鍵の保護機能を有する
	E.08-E.14	暗号モジュールへの鍵情報の入出力を平文形式で行うことは望ましくない｡出力は暗号化して行うことが望ましく､主に外部での保管(バックアップなど) 目的である
	E.21	鍵情報の入力を人間が行う場合､その正確さとセキュリティ面の問題がある。こうした入力がない場合は対象外
	E.22-E.25	マルチパーティコントロールを利用する機能を明確にする｡暗号鍵分割(K out of N秘密分散)やマルチパーティ機能をベンダに確認する
8.2節 セキュリティ評価・試験	E.26-E.34	いずれもシステムレベルの試験項目であるが、特に暗号モジュール(HSM)にも関連するものはベンダテスト、機能テスト、セキュリティテスト、環境テスト、セルフチェックテスト、第三者テストである
	E.26-E.34	FIPS140などの認証試験で上記テストをカバーするものが多い
8.3節 暗号モジュールの障害時の BCP 対策	E.35	暗号モジュールはセルフテスト機能を備えることが望ましい。FIPS140-2/-3の要件に動作前や条件付きのセルフテスト機能がある
	E.37	回復可能なエラー発生時のセルフテストを含む回復の手順､回復困難なエラー発生時の暗号モジュールの交換手順(鍵情報のバックアップや破壊を含む)を明確にする

３．２．４．Triple DES等の取り扱いについて

NISTがTriple DESを規定していたSP 800-67 Revision 2を2023 年12 月31 日に（予定通り）廃止したことに伴い、暗号技術検討会事務局からのTriple DESの取り扱いについての意見聴取の依頼に対し、暗号技術活用委員会としては検討の結果、以下のように回答した。

【Triple DESの扱いに対する意見】

- 現時点では､「運用監視暗号リスト」からの削除を検討する必要性はない
- 現時点では､「運用監視暗号リスト」の条件である「互換性維持以外の目的での利用は推奨しない。」が実質的かつ十分な制約になっており、特段の利用制限を付加する必要性もない

- 「SP 800-67 Revision 2が2023年12月に廃止されたが、それ以外は、運用監視暗号リストに移行した時点での状況とほとんど変わっていないため、Triple DESの位置づけに変更はない。」との注釈を付記する

【上記意見に至った理由】

① 廃止理由が、安全性が著しく低下したわけではなく、NISTのスケジュールに基づく動きであること
② 利用実績調査結果からは依然として極めて高い実装率であること
③ すでに運用監視暗号リストに掲載されており、互換性維持以外の目的での利用が推奨されていないこと
④ NISTも、Triple DESですでに暗号化されたデータに対する処理は引き続き許容していること
⑤ 「電子政府推奨暗号リスト」に掲載されているDSAは、現在のFIPS PUB 186-5では廃止されているが、FIPS PUB 186-5になるときに削除すべきとの議論はなかったこと

【運用監視暗号リスト掲載のアルゴリズムの取り扱いに対する意見】

運用監視暗号リスト掲載の暗号アルゴリズムは、新規に極力使用しないように促していく活動を積極的に進めるべきである。

【DSAの扱いに対する意見】

今回、Triple DESの取り扱いについて検討することになった理由が「SP 800-67 Revision 2が廃止された」ことが契機になっていると承知している。その場合、上記⑤に記載の通り、DSAも「現在のFIPS PUB 186-5では廃止されている」ことから、Triple DESとの注釈と同様の注釈を追記すべきではないか。

３．２．５．暗号技術活用委員会の開催状況

2023年度の暗号技術活用委員会での審議概要は表の通りである。

表　暗号技術活用委員会の開催状況

回	開催日	議案
第一回	2023年7月11日	● 2023 年度暗号技術活用委員会活動計画について ● 2023 年度暗号鍵管理ガイダンス WG 活動計画について ● TLS 暗号設定ガイドライン改訂について
メール	2023年1月12日～2月15日	● TLS 暗号設定ガイドライン改訂案 v3.1 のメール審議
第二回	2024年3月5日	● TLS 暗号設定ガイドライン改訂内容について ● 2023 年度暗号鍵管理ガイダンス WG 活動報告 ● Triple DES に関する扱いについて ● 2023 年度暗号技術活用委員会活動報告案について

４．今後のCRYPTRECの活動について

CRYPTRECでは、2024年度も、電子政府推奨暗号等の安全性を評価・監視するとともに、暗号技術の更なる普及促進を行うべく検討を進める。

暗号技術検討会においては、CRYPTREC暗号リストの更新等について必要に応じて検討を行う予定である。

暗号技術評価委員会においては、NISTをはじめとする世界各国の機関において耐量子計算機暗号の選定・標準化活動が継続されており、情勢が流動的であることを鑑み、引き続き耐量子計算機暗号に関する最新動向を把握し、耐量子計算機暗号ガイドラインの改定に向けた検討を進め、2024年度に改定案を審議する予定である。

暗号技術活用委員会においては、2022年度に公開した暗号鍵管理ガイダンスについて、引き続き、「暗号鍵管理システム設計指針（基本編）」に記載がありながら今回解説・考慮点の記載を見送った部分の拡充を行い、2024年度に拡充案を審議する予定である。また、暗号利活用に向けた新たな有用なガイダンス作成に着手する予定である。

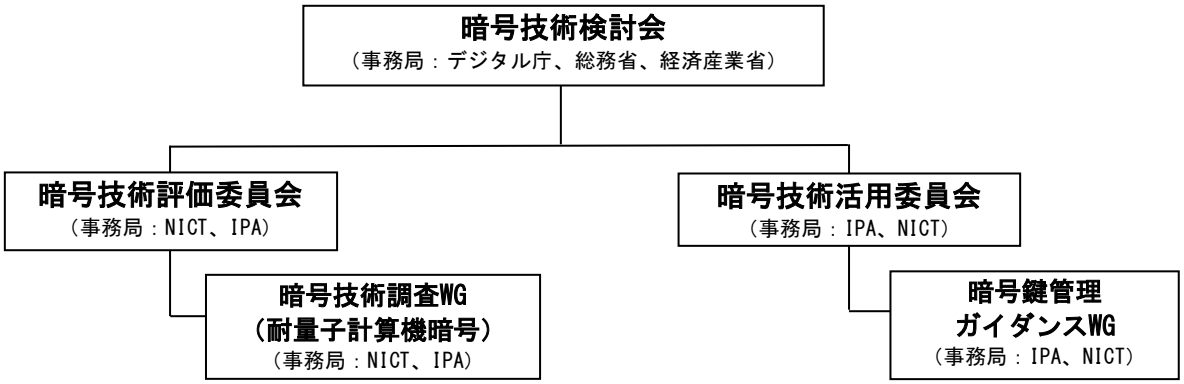


図4-1 CRYPTREC体制図（2024年度）（予定）