NIST Special Publication 800-57 Part 1 Revision 5

鍵管理における推奨事項

第一部：一般事項

Elaine Barker

This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-57pt1r5

コンピュータ セキュリティ

National Institute of Standards and Technology
U.S. Department of Commerce

この文書は以下の団体によって翻訳監修されています

IPA 独立行政法人 情報処理推進機構
INFORMATION-TECHNOLOGY PROMOTION AGENCY, JAPAN

NIST Special Publication 800-57 Part 1 Revision 5

鍵管理における推奨事項

第一部：一般事項

Elaine Barker
コンピュータセキュリティ部門
情報技術研究所

This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-57pt1r5

2020 年 5 月

米国商務省
Wilbur L. Ross, Jr.、長官

米国国立標準技術研究所
Walter Copan、NIST 標準技術局長兼商務次官

発行機関

本文書は、米国国立標準技術研究所（NIST：National Institute of Standards and Technology）によって、2014 年連邦情報セキュリティ近代化法（Federal Information Security Modernization Act of 2014）、合衆国法典（U.S. Code）第 44 編 第 3541 条等、公法（P.L.）113-283 に基づく法的責任を推進するために策定された。NIST は、連邦情報システムの最小限の要求事項を含め情報セキュリティ標準及びガイドラインを開発する責務があるが、これらの標準及びガイドラインは、国家安全保障システムについての政策的権限を有する適切な連邦機関の明示的な承認を得ることなしにはそれらのシステムに適用されてはならない。このガイドラインは、行政管理予算局（OMB：Office of Management and Budget）による通達（Circular）A-130 の要求事項に一致している。

本文書における一切は、商務長官が法的権威に基づき連邦政府に対して義務及び拘束力を与えた標準及びガイドラインを否定するものではない。また、これらのガイドラインは、商務長官、行政管理予算局長、又は他のすべての連邦政府当局者の既存の権威に変更を加えたり、これらに取って代わったりするものと解釈すべきではない。本文書は、非政府組織が自由意思で使用することもでき、米国における著作権の制約はないが、NIST に帰属する。

National Institute of Standards and Technology Special Publication 800-57 Part 1, Revision 5
Natl. Inst. Stand. Technol. Spec. Publ. 800-57 Part 1 Rev. 5, 170 pages (May 2020)
CODEN: NSPUE2

この公表文書は、以下から無料で利用可能である：
https://doi.org/10.6028/NIST.SP.800-57pt1r5

本文書中で特定される商業的組織、装置、又は資料は、実験手順や概念を適切に説明するためのものである。このような特定は、NIST による推奨又は同意を意味するものではなく、これらの組織、資料、又は装置が、その目的のために利用可能な最善なものであることを意味しているわけでもない。

与えられた法的責任に従い、NIST によって現在作成中のその他の文書への参照が本文書にあるかもしれない。本文書におけるその情報は、概念及び方法論を含め、このような関連文書の完成前であっても連邦政府機関によって利用されるかもしれない。したがって、それぞれの文書が完成されるまで、現在の要求事項、ガイドライン、及び手順が、存在する限り、運用の効力を有する。計画及び移行目的に関して、連邦政府機関は、NIST によるこれらの新しい文書の開発に密接に従うことを希望するかもしれない。

組織は、パブリックコメント期間中の全てのドラフト文書をレビューし、NIST へフィードバックを提供するよう奨励する。上記以外の多くの NIST サイバーセキュリティ文書は、http://csrc.nist.gov/publications から入手可能である。

本文書へのコメントは以下で受け付ける：

National Institute of Standards and Technology
Attn: Computer Security Division, Information Technology Laboratory
100 Bureau Drive（Mail Stop 8930）Gaithersburg, MD 20899-8930
Email: keymanagement@nist.gov

すべてのコメントは、連邦情報公開法（FOIA）の下での公開対象である。

コンピュータシステムの技術に関する報告書

米国国立標準技術研究所（NIST：National Institute of Standards and Technology）情報技術研究所（ITL：Information Technology Laboratory）は、国家の計測及び標準に関する基盤において技術的リーダーシップを提供することにより、米国の経済及び社会福祉に貢献している。ITL は、テスト、テスト技法、参照データ、概念実証及び技術的分析の開発を通じて、情報技術の開発と生産的利用の発展に努めている。ITL の責務は、連邦政府の情報システムにおいて、国家安全保障に関連する情報以外の情報に対する費用対効果の高いセキュリティとプライバシーを実現するための、管理面、運用面、技術面及び物理面での標準及びガイドラインを策定することを含んでいる。本 Special Publication 800 シリーズは、情報システムセキュリティに関する ITL の調査、ガイドライン及び普及活動、ならびに産業界、政府機関及び学術機関との共同活動について報告する。

要旨

本推奨事項は、暗号鍵管理に関するガイダンスを提供する。本文書は 3 部から構成される。パート 1 は、暗号鍵材料の管理に関する一般的なガイダンス及びベストプラクティスを提供する。これには、暗号を使用する際に提供されるセキュリティサービス及び採用されるアルゴリズムや鍵の種類の定義、各種類の鍵やその他の暗号情報が必要とする保護の仕様及びその保護を提供する方法、鍵管理に関わる機能についての説明、及び暗号を使用する際に対処すべき様々な鍵管理の問題についての説明などが含まれる。パート 2 は、米国政府機関向けの方針及びセキュリティ計画の要求事項に関するガイダンスを提供する。パート 3 は、現時点でのシステムの暗号機能を使用する際のガイダンスを提供する。

キーワード

アーカイブ； 認証； 認可； 可用性； バックアップ； 危殆化； 機密性； 暗号鍵； 暗号モジュール； デジタル署名； ハッシュ関数； 鍵合意； 鍵管理； 鍵復元； 鍵材料； 鍵配送；プライベート鍵； 公開鍵； 秘密鍵； トラストアンカー

謝辞

米国国立標準技術研究所（NIST）は、本推奨に関連する多くのセキュリティ課題に関して、本文書の以前の共著者による貢献に深く感謝の意を表する：NIST William Barker 氏、William Burr 氏、Timothy Polk 氏 ; Orion Security 社 Miles Smid 氏 ; 国家安全保障局（NSA）Lydia Zieglar 氏。 NIST はまた、公的機関及び民間の方々による、本刊行物の品質及び有用性を向上させる思慮深く建設的なコメントをいただいたことに感謝する。

特許公開の通知

通知：情報技術研究所（ITL）は、本書のガイダンスや要件に準拠するために使用が必要となる可能性のある特許請求項の所有者に対し、その特許請求項を ITL に開示するよう要請している。しかし、特許保有者は ITL の特許募集に応じる義務はなく、ITL は本書に適用される特許があるとしても、それを特定するための特許調査を行っていない。

本書を発行し、本書のガイダンスや要件に準拠するために使用が必要となる可能性のある特許請求項を特定するための呼びかけを行った時点で、ITL はそのような特許請求項を特定していない。

ITL は、本書の使用において特許侵害を回避するためのライセンスが必要ないことを表明又は暗示するものではない。

本文書は、原典に沿ってできるだけ忠実に翻訳するよう努めていますが、完全性、正確性を保証するものではありません。翻訳監修主体は、本文書に記載されている情報より生じる損失又は損害に対して、いかなる人物あるいは団体についても責任を負うものではありません。

エグゼクティブサマリ

暗号技術は、ネットワーク上の通信の安全性を確保したり、データベースに保存されている情報を保護したり、その他多くの重要なアプリケーションに使用される。暗号鍵は、暗号技術の運用において重要な役割を果たす。暗号鍵は、金庫（を開錠するための数字）の組み合わせに似ている。金庫の開錠のための番号が敵対者に知られている場合、最強の金庫であっても、侵入に対するなんらのセキュリティも提供しない。

暗号鍵の適切な管理は、セキュリティのために暗号技術を効果的に使用するために不可欠である。不適切な鍵管理は、強力なアルゴリズムを容易に危殆化させる可能性がある。本推奨は、暗号鍵の安全な生成、保管、配付、使用、及び破棄を含むライフサイクル全体を通じた暗号鍵の管理に関するガイダンスを提供する。

最終的に、暗号技術によって保護された情報のセキュリティは、鍵の強度、当該鍵に関連する暗号メカニズムやプロトコルの有効性、及び当該鍵に提供される保護に直接依存する。秘密鍵及びプライベート鍵は、不正な開示から保護される必要があり、全ての鍵は改ざんから保護される必要がある。

暗号鍵は企業の幅広いシステムやアプリケーションで使用されており、その多くは鍵管理の専門知識を持たない個人によって管理されている。そのため、組織は、鍵の適切な管理のために明確な指針と監視が提供され、その指針が適切に守られ、実施されていることを確認するための管理体制を確保しなければならない。

組織や開発者には、暗号メカニズムを使用する際に多くの選択肢が提示される。不適切な選択をした場合、安全であるような幻想とともに、プロトコルやアプリケーション上の実態としての安全性はほとんどないか、あるいは全く安全でないという結果をもたらすかもしれない。

本推奨（すなわち、特別刊行物（SP）800-57）は、暗号メカニズムを選択して使用する際のバックグラウンド情報を提供し、適切な決定を支援するためのフレームワークを確立する。

暗号モジュールは、これらの鍵を使用して暗号演算を実行するために使用される。本推奨は、そこで特定されたセキュリティ要件を達成するために使用される可能性のある暗号モジュールの、実装の詳細には触れない。これらの詳細は、連邦情報処理標準（FIPS）140［FIPS 140］とその関連する実装ガイダンス及び派生テスト要件（https://csrc.nist.gov/projects/cmvp/から入手可能）に記載されている。

本推奨は、複数の異なる対象者向けに書かれており、3 つの部分に分かれている：

- パート 1 *一般* には、鍵管理の基本的なガイダンスが含まれている。これは、鍵管理に関連する“ベストプラクティス”について、開発者やシステム管理者に助言することを目的としている。暗号モジュールの開発者は、この一般的なガイダンスから、特定のアプリケーションをサポートするために必要な鍵管理機能をより深く理解することでメリットを得ることができる。暗号モジュールの開発者は、この一般的なガイダンスから、特定のアプリケーションをサポートするために必要な鍵管理機能をより深く理解することでメリットを得ることができる。システム管理者は、どの構成設定が扱う情報に最も適しているかを判断するのに役立てるために、本文書を使用することができる。本推奨のパート 1 では、

    1. 提供される可能性のあるセキュリティサービスと、暗号メカニズムを使用する際に採用される可能性のある鍵の種類を定義する
    2. 暗号鍵材料を使用する暗号アルゴリズムに関するバックグラウンド情報を提供する
    3. 鍵及びその他の暗号技術に関する情報を機能別に分類し、各種類の情報が必要とする保護を規定し、その保護を提供する方法を特定する
    4. 暗号鍵がその寿命の間に存在する可能性のある状態を特定する

5. 鍵管理に関わる多くの機能を識別する
6. 鍵材料に関連する様々な鍵管理の課題について論じる;取り扱うトピックには、鍵の使用法、暗号期間の長さ、ドメインパラメータの検証、公開鍵の検証、鍵棚卸リスト管理、説明責任、監査、存続可能性、及び暗号アルゴリズムと鍵長の選択のためのガイダンスが含まれる。

- パート 2 *鍵管理組織のベストプラクティス* は、主にシステム所有者と管理者のニーズに対応することを意図している。組織内での暗号鍵管理の確立を支援するためのフレームワークと一般的なガイダンスを提供し、連邦政府組織の法的及び政策的なセキュリティ計画の要件における鍵管理の側面を満たすための基礎を提供する。
- パート 3 *アプリケーション固有の鍵管理ガイダンス* は、暗号を使用する現在利用可能な実装に関連する鍵管理の問題に対処することを目的としている。

目次

エグゼクティブサマリ vi

1 はじめに 1

1.1 目的 1
1.2 対象者 1
1.3 適用範囲 2
1.4 FIPS 及び NIST 推奨（NIST 標準）の目的 3
1.5 内容と構成 4

2 用語集と略語 5

2.1 用語集 5
2.2 頭字語 18

3 セキュリティサービス 21

3.1 機密性 21
3.2 データ完全性 21
3.3 認証 21
3.4 認可 22
3.5 否認防止 22
3.6 サポートサービス 22
3.7 サービスの組合せ 23

4 暗号アルゴリズム 25

4.1 暗号学的ハッシュ関数 25
4.2 対称鍵アルゴリズム 26
4.3 非対称鍵アルゴリズム 27
4.4 乱数ビット生成器 27

5 一般的な鍵管理ガイダンス 28

5.1 鍵タイプとその他の情報 28
5.1.1 暗号鍵 28
5.1.2 その他の関連情報 30
5.2 鍵の使用法 31
5.3 暗号利用期間 32
5.3.1 暗号利用期間に影響を与える要因 32
5.3.2 暗号利用期間に影響を与える結果要因 33
5.3.3 暗号利用期間に影響を与えるその他の要因 33

5.3.4 非対称鍵の使用期間と暗号利用期間 34
5.3.5 対称鍵の使用期間と暗号利用期間 35
5.3.6 特定の鍵タイプに対する暗号利用期間の推奨事項 36
5.3.7 その他の関連情報の推奨 43
5.4 保証 44
5.4.1 完全性の保証（完全性保護） 44
5.4.2 ドメインパラメータの有効性の保証 44
5.4.3 公開鍵の有効性の保証 44
5.4.4 プライベート鍵保有の保証 45
5.4.5 鍵確認 45
5.5 鍵及びその他の鍵材料の危殆化 45
5.5.1 影響 46
5.5.2 保護対策 47
5.6 暗号アルゴリズム及び鍵長選択のためのガイダンス 48
5.6.1 同等のアルゴリズム強度 48
5.6.2 アルゴリズムスイートの使用と有効なセキュリティ強度 53
5.6.3 計画されるセキュリティ強度の時間枠及び現在の承認状況 55
5.6.4 システムにおける新しいアルゴリズム及び鍵長への移行 56
5.6.5 セキュリティ強度の経年変化 58

6 鍵情報の保護要件 60
6.1 保護及び保証の要件 60
6.1.1 暗号鍵の保護及び保証要件の概要 61
6.1.2 その他の関連情報の保護要件の概要 65
6.2 保護メカニズム 66
6.2.1 配送中の鍵情報の保護メカニズム 67
6.2.2 保管中の鍵情報の保護メカニズム 70
6.2.3 鍵のメタデータ 72

7 鍵状態及び遷移 74
7.1 活性化前状態 75
7.2 活性化状態 76
7.3 一時停止状態 77
7.4 非活性化状態 80
7.5 危殆化状態 81
7.6 破棄状態 81

8 鍵管理のフェーズと機能 82
8.1 運用前フェーズ 83
8.1.1 エンティティ登録機能 83
8.1.2 システム初期化機能 84
8.1.3 初期化機能 85
8.1.4 鍵材料インストール機能 85
8.1.5 鍵確立機能 85
8.1.6 鍵登録機能 96
8.2 運用フェーズ 96
8.2.1 通常運用時のストレージ機能 97
8.2.2 運用継続性の機能 97
8.2.3 鍵変更機能 100
8.2.4 鍵導出方法 101
8.3 運用後フェーズ 102
8.3.1 鍵アーカイブ及び鍵復元機能 102
8.3.2 エンティティ登録解除機能 105
8.3.3 鍵登録解除機能 105
8.3.4 鍵破棄機能 105
8.3.5 鍵失効機能 106
8.4 破棄フェーズ 107

9 追加の考慮事項 108
9.1 アクセス制御と ID 認証 108
9.2 棚卸リスト管理 108
9.2.1 鍵の棚卸リスト 109
9.2.2 証明書の棚卸リスト 109
9.3 説明責任 110
9.4 監査 111
9.5 鍵管理システムの生存可能性 112
9.5.1 バックアップ及びアーカイブされた鍵 112
9.5.2 鍵の復元 112
9.5.3 システムの冗長性・継続性計画 114
9.5.4 危殆化からの回復 116

参考文献 118

付録 A - 暗号学的及び非暗号学的な完全性及びソース認証のメカニズム 124

付録 B - 鍵の復元 127
B.1 保管された鍵材料からの復元 127
B.2 鍵材料の再構築による復元 128
B.3 鍵材料の復元が必要な条件 128
B.3.1 署名鍵ペア 128
B.3.2 認証対称鍵 129
B.3.3 認証鍵ペア 130
B.3.4 データ暗号化対称鍵 131
B.3.5 鍵ラッピング対称鍵 131
B.3.6 乱数生成鍵 131
B.3.7 マスター対称鍵／鍵導出対称鍵 132
B.3.8 鍵配送鍵ペア 132
B.3.9 鍵合意対称鍵 133
B.3.10 静的鍵合意鍵ペア 133
B.3.11 一時的鍵ペア 134
B.3.12 認可対称鍵 134
B.3.13 認可鍵ペア 134
B.3.14 その他の関連情報 135
B.4 鍵復元システム 137
B.5 鍵復元ポリシー 138

付録 C - 改訂履歴 140
C.1 改訂 1 版（2006） 140
C.2 改訂 2 版（2007） 140
C.3 改訂 3 版（2011） 141
C.4 改訂 4 版（2015） 143
C.5 改訂 5 版（2020） 146

1 はじめに

暗号メカニズムの使用は、通信、データストレージ、及びその他のアプリケーションに対するセキュリティサービスを提供するための最も強力な方法の 1 つである。米国標準技術研究所（NIST）は、機密指定されていない機微な情報を保護するための暗号技術を規定する連邦情報処理標準（FIPS）と NIST 推奨（特別刊行物（SP）として発行される）を発行している。

NIST が 1977 年にデータ暗号化規格（DES）を公開して以来、承認された標準アルゴリズムの種類が増えてきている。安全なハッシュ関数やデジタル署名用の非対称鍵アルゴリズムなど、新しいクラスのアルゴリズムも追加された。今では、一連のアルゴリズムが、様々な鍵長により、異なるレベルの暗号強度を提供する。これらのアルゴリズムは、複雑化するプロトコルとアプリケーションをサポートするために、様々な方法で組み合わせることができる。本 NIST 推奨は、機密指定されていない機微な情報の保護に暗号を使用する米国政府機関に適用される。また、本推奨は、コンピュータシステムにおける健全なセキュリティ原則を導入したいと考えるその他の組織も、任意で適用することができる。

暗号鍵及びその他の鍵情報の適切な管理は、セキュリティのために暗号技術を効果的に使用するために不可欠である。鍵は金庫（を開錠するための数字）の組み合わせに似ている。その組合せを敵対者が知っているならば、最強の金庫でも侵入に対するセキュリティを提供しない。同様に、不十分な鍵管理は、強力なアルゴリズムを簡単に危殆化させる可能性がある。結局のところ、暗号技術によって保護される情報のセキュリティは、鍵の強度、その鍵に関連するメカニズムとプロトコルの有効性、及びその鍵に提供される保護に直接依存する。暗号技術は、脆弱な実装の使用、不適切なアルゴリズムの組み合わせ、脆弱な物理的セキュリティ、あるいは弱い（つまり、脆弱な）プロトコルを使用すると、効果を発揮できなくなる可能性がある。

鍵管理はライフサイクル全体を通じて鍵を管理するプロセスであり、鍵の安全な生成、保管、配付、使用及び破棄を含む。鍵は手動で管理されることもあるが、多くの場合、鍵管理プロセスを監視し、自動化し、保護するために自動化システムが必要となる。鍵管理を実行する自動化システムは一般的に（暗号）鍵管理システムとして知られている；SP 800-130[1]及び SP 800-152[2]を参照のこと。

1.1 目的

組織と開発者には、暗号メカニズムの使用において、多くの新しい選択肢が提示されている。不適切な選択をした場合は、プロトコルやアプリケーションが安全であるような幻想をもたらすかもしれないが、実際のセキュリティはほとんどないか、あるいは全く安全でない可能性がある。本推奨（つまり、SP 800-57）は、暗号メカニズムを選択して使用する場合のバックグランド情報を提供し、適切な決定をサポートするフレームワークを確立する。

1.2 対象者

鍵管理に関する本推奨 の対象者には、システム又はアプリケーションの所有者と管理者、暗号モジュール開発者、プロトコル開発者、及びシステム管理者が含まれる。本推奨は 3 つのパートで提供され、それらは特定の対象者に合わせて作られている。

---

[1] SP 800-130, *A Framework for Designing Cryptographic Key Management Systems*

[2] SP 800-152, *A Profile for U.S. Federal Cryptographic Key Management Systems (FCKMS).*

本推奨のパート 1（つまり、本ドキュメント）は、システム開発者とシステム管理者[3]の両方に役立つことを目的とした一般的な鍵管理のガイダンスを提供する。暗号モジュール開発者は、この一般的なガイダンスから、特定のアプリケーションをサポートするのに必要な鍵管理機能の理解を深めることでメリットを得ることができる。プロトコル開発者は、特定の種類のアルゴリズムに関連する鍵管理の特性を識別し、それらのアルゴリズムによって提供されるセキュリティサービスの理解を深めることができる。システム管理者は、パート 1 をパート 3 とともに使用して、システムに最も適切した構成設定を決定するのに役立てることができる。

本推奨のパート 2（すなわち、SP 800-57 パート 2[4]）は、適切な組織の鍵管理基盤を特定し、組織の鍵管理ポリシーを確立し、組織の鍵管理の実践と計画を規定するために使用するように、システム又はアプリケーションの所有者[5]に合わせて作成されている。

本推奨のパート 3（すなわち、SP 800-57 パート 3[6]）は、現在利用可能な暗号メカニズムに関連する鍵管理の課題に対処し、既存の鍵管理基盤やプロトコル、その他のアプリケーションのシステム導入者、システム管理者及びエンドユーザ、さらには現在利用可能な技術を使用した新しいシステムの購入決定を行う人に対してガイダンスを提供することを目的としている。

いくつかのバックグラウンド情報と根拠は、文脈や本推奨をサポートするために提供されるが、本ドキュメントは読者が暗号技術についての基本的な知識を持っていることを前提としている。バックグラウンド資料については、SP 800-175B[7]（暗号技術や NIST 暗号標準の利用法についてのガイダンス）や SP 800-32[8]（公開鍵基盤（PKI）への入門書）など、様々な NIST 及び市販の出版物を参照することができる。

1.3 適用範囲

本推奨には、暗号アルゴリズム、暗号基盤、プロトコル、実装、アプリケーション、及びその管理を包含する。機密指定されていない機微な情報の保護のための現在 NIST によって**承認されている**全ての暗号アルゴリズムは、本推奨の範囲内である。

本推奨は、暗号鍵の管理に関連する問題（暗号鍵の生成、使用、及び最終的な破棄）に焦点を当てる。アルゴリズムの選択、適切な鍵長、ならびに暗号化ポリシー及び暗号化モジュールの選択などの関連トピックについても、本推奨に含まれている。上記のトピックの一部は、他の NIST 標準とガイダンスでも扱われている。本推奨は、より焦点を絞った標準やガイドラインを補足する。

本推奨は、特定のセキュリティ要件を達成するために使用されうる暗号モジュールの実装の詳細については取り扱わない。これらの詳細については、FIPS 140[9]、FIPS 140 実装ガイダンス、及び派生テスト要件（https://csrc.nist.gov/projects/cmvp から入手可能）で取り扱う。

---

[3] システム管理者は、システムをセットアップするときに、追加の具体的な情報を必要とすることに注意されたい

[4] SP 800-57, Part 2, *Recommendation for Key Management: Part 2 – Best Practices for Key Management Organizations*.

[5] 例：組織内の情報セキュリティグループ

[6] SP 800-57, Part 3, *Recommendation for Key Management: Part 3 – Application-Specific Key Management Guidance*

[7] SP 800-175B, *Guideline for Using Cryptographic Standards in the Federal Government: Cryptographic Mechanisms*

[8] SP 800-32, *Introduction to Public Key Technology and the Federal PKI Infrastructure*

[9] FIPS 140, *Security Requirements for Cryptographic Modules.*

本推奨は、バックアップされる又はアーカイブに含まれるのに適切な鍵材料の種類、及びその鍵材料に提供される保護についての議論以外に、鍵アーカイブやバックアップ機能の運用についての要件や手順について取り扱わない。

本推奨では、“要件”という用語をよく使用する。この用語には、本ドキュメントでは次の意味がある：

1. **しなければならない（shall）**：この用語は、FIPS 要件又は本推奨への適合を主張するために満たされなければならない要求事項を示すために使用される。**しなければならない（shall）**は**否定（not）**と結びついて、**してはならない（shall not）**になる場合があることに留意すること
2. **すべきである（should）**：この用語は、重要な推奨を示すために使用される。本推奨を無視することは、望ましくない結果を招く可能性がある。**すべき（should）**は**否定（not）**と結びついて、**すべきでない（should not）**となる場合があることに留意すること

1.4 FIPS 及び NIST 推奨（NIST 標準）の目的

連邦情報処理標準（FIPS）と NIST 推奨（“NIST 標準と総称される）は、以下の理由から価値がある：

1. これらの標準は、米国政府システムに許容可能な最低レベルのセキュリティを確立する。これらの NIST 標準を実装するシステムは、機密指定さていない機微な政府情報の保護のために、**承認された**一貫したレベルのセキュリティを提供する。
2. NIST 標準を実装した異なるシステム間ではある程度の相互運用性が確立されていることが多い。例えば、Advanced Encryption Standard（AES）[10]暗号アルゴリズムを互いに実装する 2 つの製品では、それらの製品の他の機能に互換性がある場合、相互運用できる可能性がある。
3. これらの標準では、しばしばスケーラビリティを提供する。なぜなら、米国政府は効率よく数多くのものに適用できる製品と技術を要求するからである。
4. これらの標準は、高いレベルのセキュリティを提供することを保証するために、米国政府の専門家及び一般市民により精査されている。NIST 標準化プロセスは、採用前の正式な NIST パブリックレビュープロセスだけでなく、NIST ワークショップ、自主的な標準規格開発組織への参加、暗号研究会議への参加及び研究者との非公式の接触などを通じたオープンな暗号コミュニティとの対話により、幅広く民間の参加を呼びかけている。NIST は、NIST 標準に対する調査や暗号解読を奨励している。それらのセキュリティに関する情報提供は、初期の要件作成中、開発中及び採用後も含め、いつでも歓迎される。
5. NIST **承認**暗号技術は、継続的な有効性確認のために定期的に再評価される。何らかの技術が政府情報の継続的な保護に不適切であると判明した場合、NIST 標準は改訂又は廃止される。
6. NIST 標準に規定されたアルゴリズム（AES、SHA-2、ECDSA など）とそれらを搭載する暗号モジュールは、適合性試験が要求される。認定試験機関は、標準への適合を主張するベンダ実装に対して、これらの適合性試験を実施する。ベンダは、全ての適用可能な要件を満たすために、非適合とされた実装を改修することが要求される。認証済実装のユーザは、その認証済実装が標準に適合することの高い確信を得ることができる。

[10] FIPS 197, *Advanced Encryption Standard*.

1977 年以来、NIST は、**承認**された暗号の実装の根拠をなす NIST 標準の暗号“ツールキット[11]”を開発してきた。本推奨は、これら NIST 標準の多くを参照し、機微な情報を保護するための適切な使用方法についてのガイダンスを提供する。

1.5 内容と構成

パート 1 *一般的なガイダンス* は、基本的な鍵管理ガイダンスを含む。これは、鍵管理に関連する“ベストプラクティス”について、開発者とシステム管理者に助言することを目的としている。

- 1 節 *はじめに* は、鍵管理における推奨の目的、適用範囲、対象者を定める
- 2 節 *用語と略語の用語集* は、*鍵管理における推奨* の本パートで使用される用語と略語の定義を提供する。読者は、本推奨で使用される用語が他の文書では異なる定義がされている可能性があることに注意すべきである。
- 3 節 *セキュリティサービス* は、暗号メカニズムを使用して提供されるセキュリティサービスを定義する。
- 4 節 *暗号アルゴリズム* は、暗号鍵材料を生成及び使用する**承認された**暗号アルゴリズムに関するバックグラウンド情報を提供する。
- 5節 *一般的な鍵管理ガイダンス* は、用途に応じた異なるタイプの鍵とその他の鍵情報を分類し、暗号期間について検討して鍵のタイプごとに適切な暗号期間を推奨し、他の鍵材料に対する推奨や要件を提供し、ドメインパラメータと公開鍵の有効性保証の概念を紹介し、鍵危殆化の影響について説明し、暗号化アルゴリズムの強度選択や実装、置換えについてのガイダンスを提供する。
- 6 節 *鍵情報に対する保護要件* は、鍵情報の各々のタイプが要求する保護を規定し、この保護を提供する手段を特定する。これらの保護要件は、暗号モジュールベンダやアプリケーション実装者が特に関心を持つべきである。
- 7 節 *鍵の状態と遷移* は、暗号鍵がそのライフタイムの間に存在する可能性がある状態を特定する。
- 8 節 *鍵管理フェーズと機能* は、鍵管理に関わる 4 つのフェーズと多くの機能を特定する。この節は、暗号モジュールベンダや暗号基盤サービスの開発者が特に関心を持つべきである。
- 9 節 *その他の考慮事項* は、アクセス制御、ID 認証、棚卸リスト管理、説明責任、監査、及び存続可能性について説明する。
- 参考文献は、適切な参照文献のリストが記載されている。
- 付録 A *暗号学的及び非暗号学的な完全性認証及びソース認証のメカニズム* は、完全性認証及びソース認証のサービスに関する補足情報を提供する。
- 付録 B *鍵復元* は、鍵バックアップやアーカイブからの鍵復元に関する追加情報を提供する。
- 付録 C は、本ドキュメントの最初に公開されたバージョン以降の変更履歴が記載されている。

[11] ツールキットは、ソフトウェアコードではなく、アルゴリズムを規定する出版物とその使用ガイダンスの出版物とで構成されている。

2 用語集と略語

以下に示す定義は、本ドキュメントで使用されるものとして定義される。同じ用語が他の文書では異なる定義であるかもしれない。

本推奨では、暗号鍵情報の管理に関連するいくつかの用語を使用する。これらの用語はそれぞれ 2.1 節で定義されるが、本ドキュメント全体で使用されるため、これらの用語を比較したり関係を示したりするのに役立つ。

- *暗号鍵* は、暗号アルゴリズムと組み合わせて使用されるパラメータであり、鍵を知っているエンティティが再現、逆処理又は検証することができる一方、鍵を知らないエンティティはそれらができないような方法で暗号の操作を決定する。例としては、平文データを暗号化し暗号文データを復号するために AES で使用される対称鍵、デジタル署名を生成するためにデジタル署名アルゴリズムで使用される署名プライベート鍵、あるいはデジタル署名を検証するためにデジタル署名アルゴリズムで使用される署名検証公開鍵などがある。

  鍵は、他のエンティティと相互作用してビジネスを行うエンティティ（例えば、個人（人間）、組織、デバイス、又はプロセス）によって所有され、利用される。人間以外の所有者（例えば、組織、デバイス、又はプロセス）の場合、所有者は 1 人以上の人間によって代理されるか、又は保証される。例えば、所有者が組織の場合、複数の人間に鍵の使用が認可されている場合がある：その場合、そのビジネスを行うにあたって、その人間が組織を代表していると言ってもよい。デバイス又はプロセスの場合、デバイス又はプロセスが鍵を所有し使用するが、人間の保証人が鍵を管理する責任を負う（例えば、必要に応じて鍵を生成したり交換したりする）。

- *鍵材料* には、暗号アルゴリズムの実行中に使用される暗号鍵とその他の材料（例えば、初期ベクトル（IV）又はドメインパラメータ）が含まれる。

- *メタデータ* は、その特定の特性、制約、許容される用途、所有権などを説明する鍵に関連付けられた情報である。メタデータの一部は秘密にしてもよい（例えば、場合によっては鍵所有者の ID など）。

- *鍵情報* は特定の鍵に関する情報であり、それにはその鍵に関連する鍵材料、及び当該鍵の関連メタデータに関連する鍵材料の全てを含む。

対称鍵、及び非対称鍵（公開鍵）アルゴリズムのプライベート鍵には機密性保護が必要であり、一部のメタデータ要素にもこの保護が必要な場合がある。全ての鍵情報には、完全性保護が必要である。

2.1 用語集

Access control アクセス制御	リソースへのアクセスを認可されたエンティティだけに制限すること
Accountability 説明責任	1. 鍵管理の責任を個人に割り当て、その人にこれらの活動に責任を負わせること 2. あるエンティティの行為について、一意にそのエンティティまで追跡できることを保証する特性
Active state 活性化状態	鍵を使用して暗号的に情報を保護できる（例えば、平文の暗号化又はデジタル署名の生成）、以前に保護された情報に対して暗

	号的な処理が行える（例えば、暗号文の復号又はデジタル署名の検証）、又はその両方が行えるような鍵状態
Algorithm originator-usage period 作成者によるアルゴリズム利用期間	作成者がデータ保護（例えば、暗号化又はデジタル署名の生成）を行うために特定の暗号アルゴリズムを使用できる期間
Algorithm security lifetime アルゴリズム セキュリティライフタイム	鍵が危殆化しない限り、特定の暗号アルゴリズムによって保護されたデータが安全であると見積られる期間
Approved 承認された	FIPS 承認や NIST 推奨のこと。1）FIPS 又は NIST 推奨で規定されている、2）他のところで規定され、FIPS 又は NIST 推奨の参照先に採用されている、のいずれかに該当するアルゴリズム又は技術
Archive アーカイブ	1. 情報を長期保管すること 2. 長期保管に使用される場所又は媒体
Association 関連付け	特定の目的のための関係。例えば、鍵は、その鍵が使用されるアプリケーションやプロセスに関連付けられる
Assurance of (private key) possession （プライベート鍵）所持の保証	エンティティがプライベート鍵及び関連する鍵情報を所有し、そのプライベート鍵が所定の公開鍵に対応しているという信頼性
Assurance of validity 有効性の保証	公開鍵やドメインパラメータが数学的に正しいという信頼性
Asymmetric-key algorithm 非対称鍵アルゴリズム	公開鍵暗号アルゴリズムを参照
Authentication 認証	通信セッション、メッセージ、文書又は保管データにおける情報の情報源及び完全性の保証、並びにシステムとやり取りするエンティティの ID 保証を提供するプロセス ソース認証、ID 認証及び完全性認証を参照
Authentication code 認証コード	承認されたセキュリティ機能に基づく鍵付き暗号学的チェックサム。メッセージ認証コードとしても知られている
Authorization 認可	セキュリティ機能やアクティビティを実行するための“正式な”認可が与えられたエンティティに付与されるアクセス権
Availability 可用性	認可されたエンティティによる情報へのタイムリーかつ信頼できるアクセス
Backup バックアップ	必要な場合に、鍵の暗号有効期間中に回復を容易するための鍵情報のコピー
Block cipher (algorithm) ブロック暗号（アルゴリズム）	暗号鍵を使用して一度に 1 つのブロック情報を変換する対称鍵暗号アルゴリズム。ブロック暗号アルゴリズムの場合、入力ブロックの長さは出力ブロックの長さと同じである

Certificate 証明書	公開鍵証明書を参照
Certificate-inventory management 証明書棚卸リスト管理	鍵棚卸リスト管理を参照
Certification authority 認証局	証明書サブジェクトへ証明書を発行する公開鍵基盤（PKI）におけるエンティティ
Ciphertext 暗号文	暗号化された形式のデータ
Collision 衝突	2 つ以上の異なる入力が同じ出力を生成すること。ハッシュ関数も参照
Compromise 危殆化	機微な鍵情報（例えば、秘密鍵、プライベート鍵、秘密のメタデータなど）の不正な開示、改ざん、不正な置換又は不正利用
Compromised state 危殆化状態	鍵が危殆化した疑いがある、又は確認された場合に、鍵が遷移する鍵状態
Confidentiality 機密性	機微な情報が認可されていないエンティティに開示されない性質（鍵情報の秘密性が保たれているなど）
Contingency plan 緊急時対応計画（コンティンジェンシープラン）	緊急事態において、重要なリソースの可用性を確保し、業務継続を容易にするための、災害対応、バックアップ運用及び災害後復旧のためにメンテナンスされている計画
Contingency planning 緊急時対応計画策定（コンティンジェンシープランニング）	緊急時対応計画（コンティンジェンシープラン）を策定すること
Cryptanalysis 暗号解読	1. 保護の提供に寄与する鍵についての初期知識を持たずに、暗号の保護を破るために実行される操作 2. 暗号技術及び情報システムセキュリティを破ろうとするための数学的手法に関する研究。これには、アルゴリズム実装やアルゴリズム自体のエラーや弱点を探すプロセスを含む
Cryptographic algorithm 暗号アルゴリズム	暗号鍵を含む可変の入力を受け取って出力を生成する明確に定義された計算手順
Cryptographic boundary 暗号境界	明示的に定義された連続的な境界で、暗号モジュールの物理的境界を確定し、かつ暗号モジュールの全てのハードウェアやソフトウェア、ファームウェアの構成要素を含む
Cryptographic hash function 暗号学的ハッシュ関数	ハッシュ関数を参照
Cryptographic key (key) 暗号鍵（鍵）	暗号アルゴリズムと組み合わせて使用されるパラメータであって、鍵を知っているエンティティは再現、逆処理、又は検証す

	ることができる一方、鍵を知らないエンティティはそれらができないような方法でその操作を決定する 例として、以下のようなものがある： 1. 暗号文データへの平文データの変換 2. 平文データへの暗号文データの変換 3. データからのデジタル署名の計算 4. データ上のデジタル署名の検証 5. データからの認証コードの計算 6. データからの認証コード、及び受信又は取り出した認証コードの検証 7. 鍵材料の導出に使用される共有秘密の計算
Cryptographic module 暗号モジュール	**承認された**セキュリティ機能が実装され、暗号境界内に含まれているハードウェアやソフトウェア、ファームウェアの集合
Cryptoperiod 暗号利用期間	特定の鍵の使用が認可されている期間、又は特定のシステムやアプリケーションにおいて鍵の有効性が残存している期間
Data-encryption key データ暗号化鍵	鍵以外のデータを暗号化及び復号するために使用される鍵
Data integrity データ完全性	データが作成、配送、又は保管されてから、認可されていない方法で当該データが変更されていないことを示す性質
Deactivated state 非活性化状態	鍵が、暗号保護を適用（例えば暗号化）するためには使用されないが、場合によっては暗号保護された情報の処理（例えば復号）を行うために使用される鍵状態
Decryption 復号	暗号アルゴリズムと鍵を用いて、暗号文を平文に変換するプロセス
Destroyed state 破棄状態	鍵が破棄された時に遷移する鍵状態。鍵はもはや存在しないが、以前の存在が記録されている可能性がある（例えば、メタデータや監査ログのなか）
Deterministic random bit generator (DRBG) 決定論的乱数ビット生成器	DRBG アルゴリズムを含み、(少なくとも初期には) ランダム源へのアクセスを行う乱数ビット生成器。DRBG は、シードと呼ばれる秘密の初期値からビット列を生成する。暗号学的 DRBG は、シード値が知られていない場合、その出力が予測不能であるという付加特性がある。DRBG は、擬似乱数生成器（PRNG）又は決定論的乱数生成器と呼ばれることもある
Digital signature デジタル署名	データの暗号学的変換の結果であり、基盤及びポリシーをサポートして適切に実装されている時、以下のサービスを提供する。 1. ソース認証及び ID 認証

	2. データ完全性認証 3. 署名者の否認防止のサポート
Distribution 配付	鍵配付を参照
Domain parameter ドメインパラメータ	鍵ペアの生成や暗号処理（例えば、デジタル署名の生成や鍵材料の確立など）の実行のために、ある公開鍵暗号アルゴリズムと組み合わせて使用されるパラメータ
Encrypted key 暗号化された鍵	元の暗号化前の鍵の値を隠すために、**承認された**暗号アルゴリズムを使用して暗号化された暗号鍵
Encryption 暗号化	暗号アルゴリズムと鍵を用いて、平文を暗号文に変更するプロセス
Entity エンティティ	個人（人）、組織、デバイス又はプロセス
Entity registration エンティティ登録	暗号鍵のライフサイクルにおける機能。エンティティがセキュリティドメインのメンバーになるときのプロセス
Ephemeral key 一時的鍵	暗号プロセス（例えば、鍵確立）の実行ごとに生成され、鍵タイプのその他の要求事項（例えば、メッセージやセッションごとに固有であることなど）を満たす暗号鍵
Hash-based message authentication code (HMAC) ハッシュ関数ベースメッセージ認証コード	**承認された**鍵付きハッシュ関数を使用するメッセージ認証コード（FIPS 198[12]を参照）
Hash function ハッシュ関数	任意の（限度はあるが）長さのビット列を固定長のビット列に写像する関数。**承認された**ハッシュ関数は以下の特性を満たす: 1. （一方向性）あらかじめ指定されたいかなる出力に対してもその値に写像する入力を見つけることが計算量的に実行不可能である 2. （衝突困難性）同一の出力に写像される 2 つの別個の入力を見つけることが計算量的に実行不可能である
Hash value ハッシュ値	情報をハッシュ関数に適用した結果
Identifier 識別子	個人、デバイス又は組織に関連付けられたビット列。アプリケーションに応じて、識別名であったり、より抽象的なもの（例えば、IP アドレスとタイムスタンプからなる文字列）であったりする

[12] FIPS 198, *The Keyed-Hash Message Authentication Code (HMAC)*.

Identity ID、身元	エンティティを識別可能な特徴や個性
Identity authentication ID 認証	システムとやり取り（リソースへのアクセスなど）するエンティティの身元を保証するプロセス。エンティティ認証と呼ばれることもある
Initialization vector (IV) 初期ベクトル（IV	暗号プロセスの開始点を定義する際に使用するベクトル
Integrity (also, assurance of integrity) 完全性（完全性保証ともいう）	データ完全性を参照
Integrity authentication 完全性認証	データに対して認証コード又はデジタル署名が生成された時から、そのデータが改ざんされていないことを保証するプロセス
Integrity protection 完全性保護	データに対して計算された認証コード（例えば MAC）又はデジタル署名を使用して、送信又は保存されたデータが得られる保護のこと。完全性認証を参照
Key 鍵	暗号鍵を参照
Key agreement 鍵合意	二者以上の当事者が関与した情報から鍵材料を生成するようにし、どの当事者も他の当事者の関与から独立して鍵材料の値を定めることができないようにした鍵確立手続き
Key confirmation 鍵確認	ある当事者に対して、他の当事者が実際に同じ鍵材料や共有秘密を保有していることを保証する手続き
Key de-registration 鍵登録解除	暗号鍵のライフサイクル中の一機能。鍵又は関連する情報（例えばメタデータ）に、鍵がもはや使用されていないことを示す記録をすること
Key derivation 鍵導出	その他の情報と共に、事前共有鍵又は（鍵合意スキームからの）共有秘密のいずれかから、鍵材料が導出されるプロセス
Key-derivation function 鍵導出関数	暗号鍵又は共有秘密と、もしあれば他のデータを入力とする、鍵材料と呼ばれるバイナリ列を生成する関数
Key-derivation key 鍵導出鍵	追加の鍵を導出するために鍵導出方法で用いられる鍵。マスター鍵と呼ばれることがある
Key-derivation method 鍵導出法	鍵材料を導出するための、鍵導出関数又はその他の**承認された**手順
Key destruction 鍵破棄	物理的又は電子的手段によって鍵材料が復元できないように、暗号鍵の全ての痕跡を消去すること

Key distribution 鍵配付	鍵を所有、生成、あるいは別の方法で取得しているエンティティから、その鍵を使用しようとしている別のエンティティに、当該鍵及びその他の鍵材料を配送すること
Key-encrypting key 鍵暗号化鍵	他の鍵の暗号化又は復号のために用いられ、それらの鍵の機密性保護を提供する暗号鍵。鍵ラッピング鍵も参照のこと
Key establishment 鍵確立	暗号鍵のライフサイクル中の一機能。手動の配送手段（例えば、鍵ローダ）、自動化された手段（例えば、鍵配送プロトコルや鍵合意プロトコル）、又は自動化された手段と手動の手段との組み合わせを使用して、エンティティ間で暗号鍵を安全に確立するプロセス
Key information 鍵情報	鍵に関する情報であって、その鍵に関連する鍵材料及び関連メタデータを含むもの。鍵材料及びメタデータを参照
Key inventory 鍵棚卸リスト	鍵自体を含まない鍵ごとに関する情報（例えば、鍵の所有者、鍵タイプ、アルゴリズム、アプリケーション、及び有効期限）
Key-inventory (or certificate-inventory) management 鍵棚卸リスト（又は証明書棚卸リスト）管理	利用中の鍵や証明書の記録を確立し保持すること、所有者又は保証人を割り当て追跡すること、鍵や証明書の状態を監視すること、並びに、必要に応じて是正措置のために適切な担当者に状態を報告すること
Key length 鍵長	ビット数で表した鍵の長さ。“鍵サイズ”と同義に使用される
Key management 鍵管理	生成、保管、確立、入出力、使用及び破棄など、鍵のライフサイクル全体にわたり、暗号鍵及びその他の関連する鍵情報の取扱いに関わるアクティビティ
Key-Management Policy 鍵管理ポリシー	組織としての鍵管理ポリシーのハイレベルなステートメントであり、ハイレベルな構造、責任、管理標準、組織の依存性及びその他の関係、並びにセキュリティポリシーを明記したもの
Key management system 鍵管理システム	暗号鍵及びそのメタデータを管理（例えば、生成、配付、保管、バックアップ、アーカイブ、回復、使用、失効、及び破棄）するためのシステム。自動鍵管理システムは、鍵管理プロセスを監視、自動化、及び安全性を確保するために使用できる
Key-Management Practices Statement 鍵管理実践ステートメント	組織構成、責任権限、及び鍵管理ポリシーにおいて特定された機能に係わる組織のルールを詳細に記述した文書又は文書群
Key owner 鍵所有者	暗号鍵又は鍵ペアの使用を認可され、その識別子が当該暗号鍵又は鍵ペアに関連付けられているエンティティ
Key pair 鍵ペア	公開鍵とそれに対応するプライベート鍵：鍵ペアは公開鍵暗号アルゴリズムで用いられる

Key recovery 鍵復元	暗号鍵のライフサイクル中の機能:認可されたエンティティが、鍵バックアップ又はアーカイブから鍵を回復又は再構築することを可能にするメカニズム及びプロセス
Key registration 鍵登録	暗号鍵のライフサイクル中の機能：登録局が公式に鍵材料を記録するプロセス
Key revocation 鍵失効	暗号鍵のライフサイクル中で取り得る機能：ある鍵の確立された暗号有効期間の終了前に、当該鍵を運用上の使用から削除すべきことを、影響を受けるエンティティに対して通知することを可能にするプロセス
Key share 鍵シェア	n 個の鍵シェアのうち、任意の k 個（$k \leq n$）の鍵シェアを使って鍵の値を構成できるが、k-1 個以下の鍵シェアからは、その鍵の値に関する知識が得られないような、n 個のパラメータ （$n \geq 2$）のうちの一つ。暗号鍵コンポーネント又は分割鍵とも呼ばれることがある
Key size 鍵サイズ	ビット数で表した鍵の長さ。“鍵長”と同義に使用される
Key states 鍵状態	鍵の生成から破棄までの間に鍵が遷移する状態 活性化前状態、活性化状態、一時停止状態、非活性化状態、危殆化状態、及び破棄状態を参照
Key transport 鍵配送	ある当事者（送信者）が鍵を選択して暗号化（又はラッピング）し、別の当事者（受信者）にそれを配付する鍵確立手順。 公開鍵（非対称）アルゴリズムと組み合わせて使用するとき、鍵は受信者の公開鍵を使用して暗号化され、その後、受信者のプライベート鍵を使用して復号される。 対称鍵アルゴリズムと組み合わせて使用するとき、鍵は送信者と受信者で共有された鍵ラッピング鍵を使用して暗号化され、同じ鍵を使用して復号される
Key update 鍵更新	暗号鍵に対して実行される機能であって、古い鍵に関連しており、当該鍵を置き換えるために使う新しい鍵を計算する
Key wrapping 鍵ラッピング	対称鍵を用いて暗号学的に鍵を保護する方法であって、機密性及び完全性の両方の保護を提供する
Key-wrapping key 鍵ラッピング鍵	他の鍵の機密性と完全性の両方の保護を提供するために使用される対称鍵。鍵暗号化鍵も参照
Keying material 鍵材料	暗号アルゴリズムで使う暗号鍵及びその他のパラメータ（例えば、初期ベクトル（IV）やドメインパラメータなど）
Manual key transport 手動鍵配送	鍵や鍵シェアを含むデバイス又は文書を物理的に移動させることによって暗号鍵を配送する、自動化されていない手段

Master key マスター鍵	鍵導出鍵を参照
Message authentication code (MAC) メッセージ認証コード	**承認された**セキュリティ機能と対称鍵を使用する、データに対する暗号学的チェックサムであって、データの偶発的及び意図的の両方の改変を検出する
Metadata メタデータ	鍵に関連する情報であって、特定の性質、制約条件、受入れ可能な用途、所有権などが記述されたもの：鍵属性とも呼ばれる
NIST standards NIST 標準	連邦政府情報処理標準（FIPS）及び NIST 推奨
Non-repudiation 否認防止	あるメッセージが所与のエンティティにより実際に署名されたかどうかを第三者が判断するのをサポートするために使用されるデジタル署名を利用したサービス
Operational phase 運用フェーズ	暗号鍵のライフサイクル中における、標準的な暗号目的で鍵が使用されるフェーズ
Operational storage 運用ストレージ	鍵の暗号利用期間中における運用中の鍵材料の通常の保管
Owner (of a certificate) （証明書の）所有者	公開鍵証明書でサブジェクトとして識別される人間のエンティティ、又は証明書サブジェクトとして識別される人間以外のエンティティ（例えば、デバイス、アプリケーション、プロセスなど）の保証人となる人間のエンティティのこと
Owner (of a key or key pair) （鍵又は鍵ペアの）所有者	静的鍵ペアの場合、公開鍵に関連付けられ、かつプライベート鍵を使用する権限を持つエンティティのこと。一時的鍵ペアの場合、所有者は公開鍵とプライベート鍵の鍵ペアを生成したエンティティのことである。対称鍵の場合、所有者は、当該鍵の使用する権限を持つあらゆるエンティティのことである
Originator 作成者	情報を交換する処理、又は情報を保管する処理を開始するエンティティ
Originator-usage period 作成者使用期間	鍵の利用有効期間のうち、その鍵を用いてデータに暗号学的な保護が適用できる期間
Password パスワード	ID 認証、アクセス認可検証、又は暗号鍵の導出のために使用される文字列（文字、数字及びその他の記号）
Period of protection 保護期間	鍵の完全性又は機密性を維持する必要がある期間
Plaintext 平文	意味を持っており、復号の適用なしに理解可能な、分かりやすいデータ
Pre-activation state 活性化前状態	鍵が生成されたが、まだ利用の認可がされていない鍵状態

Private key プライベート鍵	公開鍵暗号アルゴリズムで使用される暗号鍵で、一意にエンティティに関連付けられ、公開されないもの。非対称（公開鍵）暗号系では、プライベート鍵は対応する公開鍵を持つ。アルゴリズムによっては、プライベート鍵は、以下の例のように利用できる： 1. 対応する公開鍵を算出する 2. 対応する公開鍵によって検証できるデジタル署名を作成する 3. 対応する公開鍵によって暗号化された鍵を復号する 4. 鍵合意処理における共有秘密を計算する
Proof of possession (POP) 所持証明（POP）	鍵ペアの所持者が実際に公開鍵に関連するプライベート鍵を所持している、という保証を得るための検証プロセス
Pseudorandom number generator (PRNG) 擬似乱数生成器（PRNG）	決定論的乱数ビット生成器（DRBG）を参照
Public key 公開鍵	公開鍵暗号アルゴリズムで使用される暗号鍵で、一意にエンティティに関連付けられ、公開できるもの。非対称（公開鍵）暗号系では、公開鍵は対応するプライベート鍵を持つ。公開鍵は誰でも知ることができ、アルゴリズムによっては、以下の例のように利用できる： 1. 対応するプライベート鍵によって生成されたデジタル署名を検証する 2. 対応するプライベート鍵によって復号できる鍵を暗号化する 3. 鍵合意処理における共有秘密を計算する
Public-key certificate 公開鍵証明書	エンティティを一意に識別し、エンティティの公開鍵や場合によっては他の情報を含み、信頼される組織によりデジタル署名された一連のデータであり、それによって公開鍵とエンティティを結びつける。証明書の追加情報には、鍵の利用用途及び有効期間を規定できる
Public-key (asymmetric-key) cryptographic algorithm 公開鍵（非対称鍵）暗号アルゴリズム	2 つの関連する鍵（公開鍵とプライベート鍵）を使用する暗号アルゴリズム。これら 2 つの鍵には、公開鍵からプライベート鍵を特定することは計算量的に困難であるという特性がある
Public Key Infrastructure (PKI) 公開鍵基盤（PKI）	公開鍵証明書の発行、維持、及び失効を行うために確立されているフレームワーク
Random bit generator (RBG) 乱数ビット生成器（RBG）	統計的に独立で偏りがないように見えるビット列を出力するデバイス又はアルゴリズム。乱数生成器も参照

Random number generator (RNG) 乱数生成器（RNG）	予測不能な一連の数列を生成するために使われるプロセス。乱数ビット生成器（RBG）とも呼ばれる
Recipient-usage period 受領者使用期間	保護された情報が処理（例えば復号）される期間
Registration authority 登録局	ユーザの識別子を確立し、保証する信頼されたエンティティ
Relying party 依拠当事者	暗号保護を適用することや、適用された保護を削除したり検証したりするために、鍵又は鍵ペアのセキュリティと信頼性に依拠する当事者。これには、公開鍵証明書の公開鍵に依拠する当事者、及び対称鍵を共有する当事者が含まれる
Representative (of a key owner) （鍵所有者の）代理人	（鍵の）保証人を参照
Retention period 保持期間	鍵又はその他の暗号的な関連情報が保持されるべき最短期間
RBG seed RBG シード	DRBG の初期化に用いられるビット列。単にシードとも呼ばれる
Secret key 秘密鍵	対称鍵暗号アルゴリズムで使用される単一の暗号鍵であり、1つ以上のエンティティに一意に関連付けられ、公開されない（つまり、鍵は秘密にされる）。秘密鍵は対称鍵とも呼ばれる。 この文脈での“秘密”という用語の使用は、機密レベルを意味するではなく、鍵を漏えいから保護する必要があるということを示す
Secret-key algorithm 秘密鍵アルゴリズム	対称鍵アルゴリズムを参照
Secret key information 秘密鍵情報	秘密にしておく必要がある鍵情報（つまり、対称鍵、プライベート鍵、鍵シェア、及び秘密メタデータ）
Secure communication protocol セキュア通信プロトコル	適切な機密性、ソース認証及び完全性保護を提供する通信プロトコル
Security domain セキュリティドメイン	単一の信頼機関の権限下にあるシステム又はサブシステム。セキュリティドメインはより大きなドメインを形成するために（例えば、階層的に）組織化されることもある
Security function セキュリティ機能	暗号アルゴリズムと暗号利用モード（適切な場合）の組み合わせ : 例えば、ブロック暗号、デジタル署名アルゴリズム、非対称鍵確立アルゴリズム、メッセージ認証コード、ハッシュ関数、又は乱数ビット生成器など。FIPS 140 を参照
Security life of data データのセキュリティ寿命	データのセキュリティ（例えば、機密性、完全性又は可用性）を保護する必要のある期間

Security services セキュリティサービス	情報の機密性、ID 認証、完全性認証、ソース認証及び否認防止のサポートを提供するために使用されるメカニズム
Security strength (Also "bits of security") セキュリティ強度("セキュリティビット"ともいう)	暗号アルゴリズムやシステムを破るために必要とされる計算量（つまり、操作数）に関連する数値。本推奨では、セキュリティ強度はビット数で規定され、{80、112、128、192、256}の組の中から規定される値である。80 ビットのセキュリティ強度はもはや十分に安全であるとは見なされないことに注意されたい
Seed シード	プロセス（DRBG など）を初期化するために使用される秘密の値。RBG シードも参照
Self-signed certificate 自己署名証明書	証明書内に含まれる公開鍵によって検証できるデジタル署名を持つ公開鍵証明書。自己署名証明書での署名は、証明書内の情報の完全性を保護するが、情報の真正性を保証しない。自己署名証明書の信頼性は、その配付に使用されるセキュアな手続きに基づく
Shall ～しなければならない	この用語は、連邦情報処理標準（FIPS）の要件、又は本推奨への適合を主張する際に満たされなければならない要求を示すために使用される。**～しなければならない（shall）**は **not** と結びついて、**～してはならない（shall not）**となることに注意されたい
Shared secret 共有秘密	鍵合意スキームを使用して計算される秘密の値であり、鍵導出方法への入力として使用される
Should ～すべきである	この用語は、非常に重要な推奨を表すために使用される。推奨を無視することは好ましくない結果を招く可能性がある。**～すべきである（should）**は **not** と結びついて、**～すべきでない（should not）**となることに注意されたい
Signature generation 署名生成	デジタル署名アルゴリズムとプライベート鍵を用いて、データに対するデジタル署名を作成すること
Signature verification 署名検証	デジタル署名アルゴリズムと公開鍵を用いて、データ上のデジタル署名を検証すること
Source authentication ソース認証	情報源についての保証を提供するプロセス。作成者認証と呼ばれることもある。ID 認証と比較すること
Split knowledge 知識分割	暗号鍵を *n* 個の鍵シェアに分割し、それぞれ単体では元の鍵に関する知識を明らかにすることがないようにするプロセス。その後、シェアを組み合わせて、暗号鍵を生成又は再生成したり、個々の鍵シェアを使って保護されたデータに対して独立した暗号処理を実行したりすることができる。元の鍵を構成するために *k*（ここで *k* は *n* 以下）個のシェアの知識が必要となる場合、任意の *k-1* 個の鍵シェアの知識からは、鍵長以外の元の鍵に関する情報は得られない

Sponsor (of a certificate) （証明書の）保証人	証明書サブジェクトとして識別された非人間のエンティティ（例えば、デバイス、アプリケーション、プロセスなど）の証明書を管理する責任がある人間のエンティティ。証明書の管理には、証明書の申請、鍵ペアの生成、必要に応じた証明書の交換、及び証明書の失効が含まれる。証明書の保証人は、証明書の公開鍵及び対応するプライベート鍵の保証人でもあることに注意されたい
Sponsor (of a key) （鍵の）保証人	鍵の使用が認可されている非人間のエンティティ（例えば、組織、デバイス、アプリケーション、プロセスなど）の鍵を管理する責任がある人間のエンティティ
Static key 静的鍵	比較的長い期間の使用を意図した鍵であって、一般的には暗号鍵確立スキームで多数回使用されることを想定している。一時的鍵と対比されたい
Suspended state 一時停止状態	鍵又は鍵ペアの使用が一定期間一時停止されうる鍵状態
Symmetric key 対称鍵	対称鍵暗号アルゴリズムで使用される単一の鍵であり、1 つ以上のエンティティに一意に関連付けられ、公開されない（つまり、鍵は秘密にされる）。対称鍵は秘密鍵とも呼ばれる。秘密鍵を参照
Symmetric-key algorithm 対称鍵アルゴリズム	ある処理とその逆処理（例えば、暗号化と復号）に同じ秘密鍵を使用する暗号アルゴリズム。秘密鍵アルゴリズムとも呼ばれる。SP 800-185[13]を参照
System システム	情報の収集、処理、保守、使用、共有、普及又は廃棄のために組織化された個別のリソースセット
System initialization システム初期化	暗号鍵のライフサイクル中の機能：安全な運用のためのセットアップ及びシステム構成を行うこと
Trust anchor トラストアンカー	1. 信頼が仮定されている権威エンティティ。PKI において、トラストアンカーは認証局であり、そのトラストアンカーが発行した証明書の署名を検証するために使用される証明書によって示される。検証プロセスのセキュリティは、トラストアンカーの証明書の真正性及び完全性に依存する。トラストアンカーの証明書は、自己署名証明書として配付されることが多い 2. 信頼された認証局の自己署名公開鍵証明書
Unauthorized disclosure 不正な開示	情報へのアクセスを認可されていないエンティティへの当該情報の開示に関するイベント

[13] SP 800-185, *SHA-3 Derived Functions: cSHAKE, KMAC, TupleHash, and ParallelHash*.

User ユーザ	個人（人）。エンティティも参照
X.509 certificate X.509 証明書	ISO/ITU-T X.509 規格によって定義された、X.509 公開鍵証明書又は X.509 属性証明書。（本文書も含め）通常、X.509 証明書は X.509 公開鍵証明書のことを指す
X.509 public-key certificate X.509 公開鍵証明書	エンティティの公開鍵、及びその他の情報と組み合わせた当該エンティティの固有の名称を含むデジタル証明書であって、証明書を発行した認証局のデジタル署名により偽造不可能とされており、ISO/ITU-T X.509 規格で定義されたフォーマットで符号化されている

2.2 頭字語

以下の略語と頭字語が、本推奨で使用される：

2TDEA	SP 800-67[14]で規定された Two-key Triple Data Encryption Algorithm
3TDEA	SP 800-67 で規定された Three-key Triple Data Encryption Algorithm
AES	FIPS 197[15]で規定された Advanced Encryption Standard
ANS	American National Standard 米国規格
ANSI	American National Standards Institute 米国標準化機構
CA	Certification Authority 認証局
CRC	Cyclic Redundancy Check 巡回冗長検査符号
CRL	Certificate Revocation List 証明書失効リスト
DRBG	SP 800-90A[16]で規定された決定論的乱数ビット生成器（Deterministic Random Bit Generator）
DSA	FIPS 186[17]で規定されたデジタル署名アルゴリズム（Digital Signature Algorithm）

[14] SP 800-67, *Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher*.
[15] FIPS 197, *Advanced Encryption Standard (AES)*.
[16] SP 800-90A, *Recommendation for Random Number Generation Using Deterministic Random Bit Generators*.
[17] FIPS 186, *Digital Signature Standard (DSS)*.

ECC	Elliptic Curve Cryptography 楕円曲線暗号
ECDSA	FIPS 186 で規定された楕円曲線デジタル署名アルゴリズム（Elliptic Curve Digital Signature Algorithm）
EdDSA	RFC 8032[18]で規定され、FIPS 186 で承認された Edwards-Curve Digital Signature Algorithm
FFC	Finite Field Cryptography 有限体上の離散対数問題を安全性の根拠とする暗号
FIPS	Federal Information Processing Standard 連邦情報処理標準
HMAC	FIPS 198 で規定された鍵付きハッシュメッセージ認証コード（Keyed-Hash Message Authentication Code）
IFC	Integer Factorization Cryptography 素因数分解問題を安全性の根拠とする暗号
ISO/ITU-T	International Organization for Standardization/International Telecommunication Union – Telecommunication 国際標準化機構／国際電気通信連合－電気通信
IV	Initialization Vector 初期ベクトル
KMAC	SP 800-185 で規定された Keccak-based Message Authentication Code
MAC	Message Authentication Code メッセージ認証コード
MQV	Menezes-Qu-Vanstone。鍵確立用に SP 800-56A[19]で**承認された**アルゴリズム
NIST	National Institute of Standards and Technology 米国国立標準技術研究所
PKI	Public-Key Infrastructure 公開鍵基盤
POP	Proof of Possession 所持証明
RA	Registration Authority 登録局
RBG	Random Bit Generator 乱数ビット生成器

[18] RFC 8032, *Edwards-Curve Digital Signature Algorithm (EdDSA)*.

[19] SP 800-56A, *Recommendation for Pair-wise Key-Establishment Schems Using Discrete Logarithm Cryptography*.

RNG	Random Number Generator 乱数生成器
RSA	Rivest, Shamir, Adelman ; デジタル署名用として FIPS 186 で、鍵確立用として SP 800-56B[20]で**承認された**アルゴリズム
S/MIME	Secure Multipurpose Internet Mail Extensions
SHA-2	FIPS180[21]で規定された Secure Hash Algorithm
SSH	Secure Shell protocol
TDEA	Triple Data Encryption Algorithm。Triple DEA は SP 800-67 で規定された
TLS	Transport Layer Security

[20] SP 800-56B, *Recommendation for Pair-wise Key-Establishment Schemes Using Integer Factorization Cryptography*.

[21] FIPS 180, *Secure Hash Standard (SHS).*

3 セキュリティサービス

暗号技術は、いくつかの基本的なセキュリティサービス（すなわち、機密性、ID 認証、完全性認証、ソース認証、認可、及び否認防止）を提供又はサポートするために使用されてもよい。これらのサービスは、鍵及びその鍵に関連する他の鍵情報を保護するためにも必要とされる場合がある。さらに、これらのセキュリティサービスをサポートするために使用される他の暗号メカニズム及び非暗号メカニズムがある。一般的に、単一の暗号メカニズムは、複数のサービスを提供することができる（例えば、デジタル署名の使用は、完全性認証及びソース認証を提供することができる）が、全てのサービスを提供するわけではない。

3.1 機密性

機密性とは、情報が不正な者に開示されない特性のことである。秘密性が、しばしば機密性と同義の用語として使われる。機密性は、暗号化を使用して、暗号化された情報を復号するために適切な鍵を使用する権限のある者以外は情報を理解できないようにすることで達成できる。暗号化が機密性を提供するためには、暗号化に使用される暗号化アルゴリズムと暗号利用モードは、権限のない者が暗号化に関連する復号鍵を決定できず、また、その復号鍵を使用せずに平文を直接導き出すこともできないように設計され、実装されていなければならない。

3.2 データ完全性

データ完全性とは、データが作成、送信又は保存された後、認可されていない方法で改変されていないことを示す特性のことである。改変には、データの挿入、削除及び置換が含まれる。メッセージ認証コードやデジタル署名などの暗号メカニズムは、偶発的な改変（例えば、ノイズの多い送信中やハードウェアのメモリ障害によって時々発生する改変）と、敵対者による意図的な改ざんの両方を（高い確率で）検出するために使用することができる。非暗号メカニズムも偶発的な改変を検出するためによく使用されるが、意図的な改ざんを検出するためには信頼できない。この問題のより詳細な扱いは、付録 A で提供されている。

本推奨では、暗号アルゴリズムが“データ完全性を提供する”という記述は、そのアルゴリズムが認可されていない改変を検出するために使用できることを意味する。完全性認証については、次節で説明する。

3.3 認証

暗号を使用して提供できる認証サービスには、ID 認証、完全性認証及びソース認証の 3 種類がある。

- ID 認証サービスは、システムとやり取りするエンティティの ID を保証するために使用される
- 完全性認証サービスは、データが改変されていないことを検証するために使用される（すなわち、このサービスは完全性保護を提供する）
- ソース認証サービスは、情報を作成したり送信したりしたエンティティの身元を検証するために使用される

ソース認証と ID 認証は非常に似ているが、目的が異なる。例えば、ソース認証はメッセージの発信者に関係しているのに対し、ID 認証はサービスへのアクセスを得るために使用される。

いくつかの暗号メカニズムが、認証サービスを提供するために使用される場合がある。最も一般的には、デジタル署名又はメッセージ認証コードが認証を提供するために使用される。いくつかの鍵合意技術も認証を提供することがある。

複数の個人が同じ ID 又はソース認証情報（パスワードや暗号鍵など）を共有することが許可されている場合、これは役割ベース認証と呼ばれることがある。FIPS 140 を参照のこと。

3.4 認可

認可は、機能やアクティビティ（文書へのアクセスや部屋へのアクセスなど）を実行するための公式な制限や許可を提供することに関係している。認可は、しばしば暗号サービスによってサポートされるセキュリティサービスであると見なされる。通常、認可は、ID 認証サービスが実行され成功した場合にのみ付与される。ID 認証と認可の間の相互作用に対する、暗号を使わない類似行為として、個人の身元を確認（ID 認証プロセス）するための個人のクレデンシャルの検査がある。個人の身元を検証し、当該人が鍵のかかった部屋などのリソースへのアクセスを認可されていることを検証した後、当該人には、そのリソースへのアクセスを許可する鍵（例えば、認可鍵）やパスワードなどが提供される。

ID 認証は、個人を特定するのではなく、役割（システム管理者や監査人など）を認可するために使用することもできる。役割の認証を受けると、エンティティには、その役割に関連付けられた全ての権限が認可される。

3.5 否認防止

鍵管理において、否認防止は、デジタル署名鍵及び証明書サブジェクトの名前を公開鍵に結び付けるデジタル証明書に関連する用語である。デジタル署名鍵に対して否認防止が示されている場合、当該鍵によって生成される署名は、デジタル署名の通常の完全性認証サービスとソース認証サービスをサポートするだけでなく、（署名の文脈に応じて）文書上の手書き署名が契約へのコミットメントを示すのと同じ意味で、証明書サブジェクトによるコミットメントを示すことができることを意味する。

実際に否認防止が認められるかどうかの決定は、多くの側面を考慮したうえで法的に判断される。暗号メカニズムは、この判断の 1 つの要素としてのみ使用することができる（すなわち、デジタル署名は、否認防止の判断をサポートするためにのみ使用することができる）。

3.6 サポートサービス

上述した基本的な暗号セキュリティサービスは、しばしば、他のサポートサービスを必要とする。例えば、暗号サービスは、しばしば、鍵確立サービス及び乱数生成サービスの使用を必要とする。鍵確立は、手動の配送方法（例えば、鍵ローダ）、自動化された方法（例えば、鍵配送プロトコルや鍵合意プロトコル）、又は自動化された方法と手動の方法の組み合わせを使用して、エンティティ間で暗号鍵が安全に確立されるプロセスである。暗号鍵、チャレンジ値及びノンスの生成には、乱数が必要である（SP 800-175B を参照）。

3.7 サービスの組合せ

多くのアプリケーションでは、セキュリティサービス（機密性、完全性認証、ソース認証、及び否認防止のサポートなど）の組み合わせが求められる。安全なシステムの設計者は、システムが保存・処理する情報を保護するために、どのセキュリティサービスが必要かを検討することから始めることが多い。これらのサービスが決定された後、設計者は、どのようなメカニズムがこれらのサービスを提供するのに最適かを検討する。すべてのメカニズムが本質的に暗号学的であるわけではない。例えば、特定のタイプのデータの機密性を保護するために物理的なセキュリティが使用（例えば、データを金庫に入れるなど）されたり、ID バッジ又は生体認証デバイスが ID 認証に使用されたりする。しかしながら、アルゴリズム、鍵、及び他の鍵材料からなる暗号メカニズムは、しばしば、情報セキュリティを保護するための費用対効果の高い追加手段を提供する。これは、そうしなければ情報が認可されていないエンティティに開示されてしまうようなアプリケーションに特に当てはまる。

適切に実装されている場合、いくつかの暗号アルゴリズムは、複数のサービスを提供する。以下の例は、そのような場合を説明するものである：

1. メッセージ認証コード（4.2 節及び SP 800-175B[22]）は、対称鍵がエンティティの組ごとに一意である場合、完全性認証と同時にソース認証を提供することができる。
2. デジタル署名アルゴリズム（4.3 節及び SP 800-175B）は、ID 認証、完全性認証、及びソース認証を提供し、否認防止の判断をサポートすることができる。
3. 特定の暗号利用モードは、適切に実装されていれば、機密性、完全性認証及びソース認証を提供できる。これらのモードは、これらのサービスを提供するために特別に設計されるべきである。

望ましいサービスを全て提供するためには、異なるアルゴリズムや手順を採用する必要があることが多い。

例：

インターネット上のエンティティの組間で情報を安全に交換する必要があるシステムを考える。交換される情報のなかには完全性保護のみを必要とするものもあれば、完全性と機密性の両方の保護を必要とするものもある。また、情報交換に参加する各エンティティが他のエンティティの身元を知っていることも要求される。

このシステム例の設計者は、公開鍵基盤（PKI）を確立する必要があり、安全な通信を希望する各個人は、物理的に身元を証明した後、必要な公開鍵証明書を取得する必要があると判断した。PKI には、証明書の作成を担当する 1 つ以上の認証局（CA）と、通常は各 CA に関連付けられた少なくとも 1 つの登録局（RA）が含まれる：RA は、証明書を要求するエンティティの身元を確認する責任がある。ID 証明プロセスでは、適切なクレデンシャル（運転免許証、パスポート、出生証明書など）の提示が必要となる。

2 種類の公開鍵証明書が一般的に使用される：デジタル署名に使用される証明書、及び鍵確立（すなわち、鍵合意又は鍵配送）に使用される証明書。4.3 節及び SP 800-175B を参照。

[22] SP 800-175B, *Guideline for Using Cryptographic Standards in the Federal Government: Cryptographic Mechanisms*.

- デジタル署名証明書を取得するには、個人が特定のデジタル署名アルゴリズム（RSA や ECDSA など）の鍵ペアを生成する。ここで、当該人が鍵ペアの所有者とみなされる。鍵ペアは、互いに対応する公開鍵とプライベート鍵で構成され、特定のアルゴリズムでのみ使用可能である。鍵ペアの公開鍵は、当該鍵ペアの所有者が使用する識別子やその他の情報とともに証明書に含まれる。証明書は CA が所有するデジタル署名プライベート鍵を用いて CA によってデジタル署名され、証明書は鍵ペアの所有者に提供されるか、リポジトリに預けられるか、又はその両方である。プライベート鍵は所有者の単独管理下にある（すなわち、プライベート鍵は秘密にされている）。

  デジタル署名証明書を使用する場合、あるエンティティ（すなわち署名者）がプライベート鍵を使用してデータに署名し、署名されたデータを意図した受信者に送信する。受信者は、署名者の公開鍵証明書を（例えば、受信者又は何らかのリポジトリから）取得し、証明書の署名に使用されたプライベート鍵に対応する CA の公開鍵を使用して証明書を検証し、証明書内の公開鍵（すなわち、署名者が使用したプライベート鍵に対応する公開鍵）を使用して、受信したデータへの署名を検証する。このプロセスを使用することで、受信者は、デジタル署名アルゴリズムを使用して、受信データの完全性とソースの両方の保証を得る。

- 鍵確立のための証明書を取得するには、特定の鍵確立アルゴリズム（RSA や Diffie-Hellman アルゴリズムなど）用の鍵ペアを生成する必要がある。デジタル署名の場合と同様に、公開鍵は CA によって署名された証明書に格納され、プライベート鍵は鍵ペアの所有者によって秘密にされる。

  鍵確立には一般的に 2 つの方法が用いられる：鍵合意及び鍵配送。鍵合意では、2 つのエンティティが通信を希望する場合、両方のエンティティが同じ鍵を生成できるようにするための情報（鍵確立証明書やその他の情報など）を交換する必要がある。鍵配送では、一方のエンティティ（送信者）が他方のエンティティの認証済公開鍵を使用して、他方のエンティティに送信する鍵を暗号化する必要がある。どちらの場合も、合意された鍵又は配送された鍵を使用する前に、証明書内の CA の署名を検証することで、その鍵確立証明書が正当なものであるかチェックされる。

  合意された鍵及び配送された鍵は、その後暗号化又はメッセージ認証アルゴリズムなどで使用され、送信されたデータの機密性又は完全性の保護を提供する。対称鍵で保護されたデータの受信者は、データが公開鍵証明書で示されたエンティティから来たという保証を持つ（すなわち、対称鍵のソース認証が得られることとなる）。

上記の例は、複数のセキュリティサービスをサポートするために暗号アルゴリズムがどのように使用されるかの基本的な概要を提供する。しかし、このようなシステムのセキュリティは、以下のような多くの要因に依存することが容易に分かる：

a. 個人のクレデンシャル（例えば、運転免許証、パスポート、出生証明書など）の強度、及び ID 認証プロセス

b. 使用されている暗号アルゴリズムの強度

c. RA と CA に対する信頼度

d. 鍵確立プロトコルの強度

e. ユーザが鍵を生成し、当該鍵を不正使用から保護する際の配慮

したがって、暗号アルゴリズムと健全な鍵管理技術を活用して所望のセキュリティサービスを提供するセキュリティシステムの設計には、あらゆる要因とリスクを慎重に考慮する必要がある。このようなシステムの設計と実装は、これら全ての要因とリスクを効果的に検討し、対処するために必要なスキルと専門知識を有するアナリストによって行われる**べきである**。

4 暗号アルゴリズム

暗号サービスが必要とされる場合には、常に FIPS **承認**又は NIST 推奨の暗号アルゴリズムを使用**しなければならない**。これらの**承認された**アルゴリズムは、承認前に集中的なセキュリティ解析を受けており、そのアルゴリズムが適切なセキュリティを提供することを保証にするために継続的に検証されている。ほとんどの暗号アルゴリズムは、暗号鍵及びその他の鍵材料を必要とする。場合によっては、使用する鍵長を大きくすることでアルゴリズムが強化されることもある。本推奨は、アルゴリズムと鍵長の適切な選択について、暗号メカニズムのユーザに助言する。

> **重要な注意**：現在承認されている非対称アルゴリズムによって提供されるセキュリティは、将来の量子コンピュータ上で実行される Shor アルゴリズムのような暗号解読アルゴリズムにより破られると予測されている。将来的には、耐量子アルゴリズムへの移行が計画されている。この取り組み状況については、https://csrc.nist.gov/projects/post-quantum-cryptography を参照のこと。

本章では、セキュリティサービス（機密性、ID 認証、完全性認証、ソース認証など）を提供する**承認された**暗号アルゴリズムについて説明する。これらのサービスは、複数の異なるアルゴリズムを使用して実現することができ、多くの場合、同じアルゴリズムを使用して複数のサービスを提供することができる。暗号サービスの提供に関する追加情報については、SP 800-175B を参照のこと。

承認された暗号アルゴリズムには 3 つの基本的なクラスがある：ハッシュ関数（4.1 節）、対称鍵アルゴリズム（4.2 節）、及び非対称鍵アルゴリズム（4.3 節）。これらのクラスは、アルゴリズムと共に使用される暗号鍵の数によって定義される。これらのアルゴリズムを使用するために必要な鍵は、乱数ビット生成器を使用して生成されなければならない（4.4 節）。

4.1 暗号学的ハッシュ関数

暗号学的ハッシュ関数は、その基本動作に鍵を必要としない。暗号学的ハッシュ関数（ハッシュアルゴリズムとも呼ばれる）は、入力（メッセージや他のデータなど）の凝縮された表現を生成する暗号プリミティブである。ハッシュ関数の出力の一般的な名称には、ハッシュ値、ハッシュ、メッセージダイジェスト、デジタルフィンガープリントなどがある。入力と出力の最大ビット数は、ハッシュ関数の設計によって決定される。**承認された**ハッシュ関数はすべて暗号学的ハッシュ関数であり、FIPS 180、FIPS 202[23]及び SP 800-185 で規定されている。SP 800-175B[24]では、ハッシュ関数がどのように動作するかを簡単に説明している。

適切に設計された暗号学的ハッシュ関数では、与えられたハッシュ値を生成するメッセージを構築したり見つけたりすることは困難（現像困難性）であるだけでなく、同じハッシュ値を生成する 2 つのメッセージを見つけることも困難（衝突困難性）である。アルゴリズム標準では、ハッシュ関数の適切なサイズを指定するか、又はアルゴリズムが異なるハッシュ関数を使用するように構成できる場合にはハッシュ関数の選択基準を提供する必要がある。

---

[23] FIPS 202, *SHA-3 Standard: Permutation-based Hash and Extendable Output Functions*.

[24] SP 800-175B, *Guideline for Using Cryptographic Standards in the Federal Government: Cryptographic Mechanisms*.

セキュリティサービスを提供する多くのアルゴリズム及びスキームは、アルゴリズムのコンポーネントとしてハッシュ関数を使用する（すなわち、ハッシュ関数は構成要素として使用される）。例えば、以下のようなものがある：

1. ソース認証サービス及び完全性認証サービスを提供するために、ハッシュ関数は、メッセージ認証コード（MAC）を生成するために鍵とともに使用される（4.2 節の項目 2 を参照）
2. デジタル署名の生成と検証のためにメッセージを圧縮する（4.3 節の項目 1 を参照）
3. 事前共有鍵から鍵を導出する（4.2 節の項目 3 を参照）
4. 非対称鍵確立アルゴリズムを用いて鍵を導出する（4.3 節の項目 2 を参照）
5. 乱数を生成する（4.4 節参照）

4.2 対称鍵アルゴリズム

対称鍵アルゴリズム（秘密鍵アルゴリズムとも呼ばれる）は、秘密鍵を知らないと元に戻すことが根本的に難しい方法でデータを変換する。鍵が“対称”であるのは、暗号化処理とその逆処理（例えば、暗号化と復号の両方）に同じ鍵が使用されるからである。対称鍵は複数のエンティティが知っていることが多い：しかしながら、鍵はランダムなプロセスを用いて生成**されなければならず**、かつそのアルゴリズムと鍵によって保護されたデータへのアクセスを認可されていないエンティティに開示**されてはならない**。

対称鍵アルゴリズムには 2 つのクラスが**承認されている**：ブロック暗号アルゴリズム（例えば、FIPS 197 で規定されている AES）に基づくものと、ハッシュ関数の使用に基づくもの（例えば、FIPS 198 で規定されている鍵付きハッシュメッセージ認証コード）である。SP 800-175B は、ブロック暗号アルゴリズムで使用される暗号利用モードと一緒に、各アルゴリズムのタイプについて説明している。

対称鍵アルゴリズムは、例えば以下のように使用される：

1. データの機密性を提供する － データの暗号化と復号に同じ鍵が使用される [25]
2. メッセージ認証コード（MAC）[26]の形でソース認証サービス及び完全性認証サービスを提供する － MAC の生成及びそれの検証に同じ鍵を使用する（MAC は通常、暗号プリミティブとして対称鍵アルゴリズム又は暗号学的ハッシュ関数を使用する）
3. 事前共有鍵から、鍵導出方法 [27]を用いて鍵材料を導出する
4. 非対称鍵合意スキーム [28]の使用中に、共有秘密から鍵を導出する
5. 鍵ラッピングアルゴリズム [29]を使用して鍵をラッピングする
6. 乱数を生成する（4.4 節参照）

---

[25] 例えば、FIPS 197、SP 800-38A、SP 800-38C 及び SP 800-38D を参照。

[26] 例えば、SP 800-38B で規定された CMAC、ハッシュ関数を使った FIPS 198 で規定された HMAC、及び SP 800-185 で規定された KMAC を参照。

[27] SP 800-108 を参照。

[28] SP 800-56A 及び SP 800-56B を参照。

[29] 例えば、FIPS 197 及び SP 800-38F を参照。

4.3 非対称鍵アルゴリズム

非対称鍵アルゴリズム（一般的に公開鍵アルゴリズムとして知られている）は、機能を実行するために 2 つの関連する鍵（すなわち鍵ペア）を使用する：公開鍵とプライベート鍵。公開鍵は誰でも知ることができるが、プライベート鍵は当該鍵ペア[30]を“所有する”エンティティの単独管理下にあるべきである。鍵ペアの公開鍵とプライベート鍵が関連していても、公開鍵の知識を使用してプライベート鍵を決定することはできない。

非対称鍵アルゴリズムでは、鍵ペアの一方の鍵が暗号保護を適用するために使用され、他方の鍵がその保護を解除又は検証するために使用される。使用する鍵は、使用するアルゴリズムと提供するサービスに依存する。非対称アルゴリズムは、例えば以下のように使用される：

1. デジタル署名[31]の形で、ソース認証サービス、ID 認証サービス及び完全性認証サービスを提供する
2. 鍵合意アルゴリズム及び鍵配送アルゴリズム[32]を使用して、暗号鍵材料を確立する

SP 800-175B では、デジタル署名の生成及び鍵材料の確立に対する非対称鍵アルゴリズムの使用について説明している。

4.4 乱数ビット生成器

乱数ビット生成器（RBG）（乱数生成器（RNG）とも呼ばれる）は、鍵材料（鍵や IV など）の生成に必要である。RBG は、乱数ビット列（例えば、010011）を生成する：技術的には、RNG はそれらのビットを数字に変換する（例えば、010011 は数字の 19 に変換される）。しかし、“乱数生成器（RNG）”という用語は、両方の概念を参照するために一般的に使用される。RBG の使用については、SP 800-175B で説明しており、承認された RBG は SP 800-90 シリーズの文書で規定されている。

[30] 鍵ペアは、鍵の所有者ではなく、鍵の所有者が信頼している者によって生成され、鍵の所有者に提供されることもある。

[31] FIPS 186 を参照。

[32] SP 800-56A 及び SP 800-56B を参照。

5 一般的な鍵管理ガイダンス

本節では、用途に応じて様々な種類の鍵とその他の暗号情報を分類する：暗号利用期間について説明し、鍵のタイプごとに適切な暗号利用期間を提案する：その他の鍵材料に対する推奨と要件を提供する：ドメインパラメータの有効性、公開鍵の有効性及びプライベート鍵の所有についての保証を解説する：鍵材料の危殆化の影響について説明する：さらに、セキュリティ強度に応じた暗号アルゴリズム及び鍵長の選択、実装及び交換に関するガイダンスを提供する。

5.1 鍵タイプとその他の情報

暗号鍵にはいくつかの異なるタイプがあり、それぞれ異なる目的で使用される。さらに、暗号アルゴリズムと鍵に特に関連するその他の情報もある。これらの鍵生成については、SP 800-133[33]で説明される。

5.1.1 暗号鍵

いくつかの異なるタイプの鍵が定義される。鍵は、公開鍵、プライベート鍵、又は対称鍵（つまり秘密鍵）の分類に従って識別され、それらの利用法が示されている。鍵合意に利用される公開鍵及びプライベート鍵に対しては、静的鍵又は一時的鍵としての状態も指定される。鍵のタイプごとの必要な保護については、6.1.1 節の表 5 を参照。

1. *署名プライベート鍵*[34]：署名プライベート鍵は、長期使用を目的としているデジタル署名を生成するための公開鍵アルゴリズムで使用される非対称鍵（公開鍵）の鍵ペアのプライベート鍵である。適切に扱うことができれば、署名プライベート鍵は、メッセージやドキュメント、保存されたデータのソース認証と完全性認証を提供するほか、それらの否認防止をサポートするためにも、使用することができる。
2. *署名検証公開鍵*：署名検証公開鍵は、デジタル署名を検証するための公開鍵アルゴリズムで使用される非対称鍵（公開鍵）の鍵ペアの公開鍵であり、メッセージ、ドキュメント又は保存されたデータのソース認証と完全性認証を提供するほか、それらの否認防止をサポートすることを目的としたデジタル署名を検証するために使用される。
3. *認証対称鍵*[35]：認証対称鍵は、対称鍵アルゴリズムと共に使用され、通信セッション、メッセージ、文書又は保存されたデータの ID 認証と完全性認証を提供する。対称鍵アルゴリズムの認証暗号利用モードでは、一つの鍵が認証と暗号化の両方に使用されることに注意されたい（SP 800-175B を参照）。
4. *認証プライベート鍵*[36]：認証プライベート鍵は、エンティティの身元の保証（つまり、ID 認証）を提供するための公開鍵アルゴリズムで使用される非対称鍵（公開鍵）の鍵ペアのプライベート鍵であり、認証された通信セッション又は何らかのアクションを実行するための認可を確立するときに使用される[37]。

---

[33] SP 800-133, *Recommendation for Cryptographic Key Generation*.
[34] FIPS 186 を参照。
[35] SP 800-38B、FIPS 198 及び SP 800-185 を参照。
[36] FIPS 186 を参照。
[37] 完全性保護も提供されるが、それはこの鍵の主要な目的ではない。

5. *認証公開鍵*: 認証公開鍵は、エンティティの身元の保証（つまり、ID 認証）を提供するための公開鍵アルゴリズムで使用される非対称鍵（公開鍵）の鍵ペアの公開鍵であり、認証された通信セッション又は何らかのアクションを実行するための認可を確立するときに使用される [38]。

6. *データ暗号化対称鍵* [39]：これらの鍵は、対称鍵アルゴリズムを用いて、データの機密性保護に適用（つまり、平文データの暗号化）するために使用される。また、同じ鍵が、機密性保護を解除（つまり、暗号文データの復号）するためにも使用される。対称鍵アルゴリズムの認証付き秘匿モード [40]では、一つの鍵がソース認証と暗号化の両方に使用されることに注意されたい。

7. *鍵ラッピング対称鍵* [41]：鍵ラッピング対称鍵（鍵暗号化鍵と呼ばれることもある）は、対称鍵アルゴリズムを用いて、他の鍵を暗号化するために使用される。鍵の暗号化に使用された鍵ラッピング鍵は、暗号化処理を元に戻す（つまり、暗号化された鍵を復号する）ためにも使用される。鍵を使用するアルゴリズムによっては、完全性保護を提供するために鍵を使用することもできる。

8. *乱数生成対称鍵* [42]：これらの鍵は、乱数又は乱数ビットを生成するために使用される。

9. *マスター対称鍵／鍵導出対称鍵* [43]：マスター対称鍵は、対称暗号化方式を使用して他の対称鍵（データ暗号化鍵や鍵ラッピング鍵など）を導出するために使用される。マスター鍵は、鍵導出鍵とも呼ばれる。

10. *鍵配送プライベート鍵* [44]：鍵配送プライベート鍵は、その鍵に対応する公開鍵で公開鍵暗号アルゴリズムを使用して暗号化された鍵の復号に使用される、非対称鍵（公開鍵）の鍵ペアのプライベート鍵である。鍵配送鍵は、通常、対称鍵（例えば、鍵ラッピング鍵、データ暗号化鍵、MAC 鍵など）及び場合によっては他の鍵材料（例えば、初期ベクトル）を確立するために使用される。

11. *鍵配送公開鍵*: 鍵配送公開鍵は、公開鍵アルゴリズムを使用して鍵を暗号化するために使用される非対称鍵（公開鍵）の鍵ペアの公開鍵である。これらの鍵ペアは、対称鍵（例えば、鍵ラッピング鍵、データ暗号化鍵、MAC 鍵など）及び場合によっては他の鍵材料（例えば、初期ベクトル）を確立するために使用される。確立された鍵の暗号化形態は、後で鍵配送プライベート鍵を使用して復号するために保存できる。

12. *鍵合意対称鍵* [45]：これらの対称鍵は、対称鍵合意アルゴリズムを使用して、対称鍵（例えば、鍵ラッピング鍵、データ暗号化鍵、MAC 鍵など）及び場合によっては他の鍵材料（例えば、初期ベクトル）を確立するために使用される。

13. *静的鍵合意プライベート鍵* [46]：静的鍵合意プライベート鍵は、非対称鍵（公開鍵）の鍵ペアの長期的なプライベート鍵であり、対称鍵（例えば、鍵ラッピング鍵、データ暗号化鍵、MAC 鍵など）及び場合によってはで他の鍵材料（例えば、初期ベクトル）を確立するために使用される。

---

[38] 完全性の保護も提供されるが、それはこの鍵の主要な目的ではない。
[39] FIPS 197、SP 800-38A、SP 800-38C、SP 800-38D 及び SP 800-175B を参照。
[40] SP 800-38C 及び SP 800-38D を参照。
[41] SP 800-38F を参照。
[42] SP 800-90A を参照。
[43] SP 800-108 及び SP 800-56C と SP 800-135 での鍵導出手法を参照。
[44] SP 800-56B を参照。
[45] 現状は、承認された方法はない。
[46] SP 800-56A を参照。

14. *静的鍵合意公開鍵*: 静的鍵合意公開鍵は、非対称鍵（公開鍵）の鍵ペアの長期的な公開鍵であり、対称鍵（例えば、鍵ラッピング鍵、データ暗号化鍵、MAC 鍵など）及び場合によっては他の鍵材料（例えば、初期ベクトル）を確立するために使用される。
15. *一時的鍵合意プライベート鍵*[47]: 一時的鍵合意プライベート鍵は、非対称鍵（公開鍵）の鍵ペアの短期的なプライベート鍵であり、一つ以上の対称鍵（例えば、鍵ラッピング鍵、データ暗号化鍵、MAC 鍵など）及び場合によっては他の鍵材料（例えば、初期ベクトル）を確立するために一度だけ使用される。
16. *一時的鍵合意公開鍵*: 一時的鍵合意公開鍵は、非対称鍵の鍵ペアの短期的な公開鍵であり、一つ以上の対称鍵（例えば、鍵ラッピング鍵、データ暗号化鍵、MAC 鍵など）及び場合によっては他の鍵材料（例えば、初期ベクトル）を確立するために一回の鍵確立トランザクションで使用される。
17. *認可対称鍵*[48]: 認可対称鍵は、対称暗号方式を使用してエンティティに権限を付与するために使用される。認可鍵は、認可されたエンティティへのアクセス権限の監視と付与を担当する責任を負うエンティティ、及びリソースへのアクセスを求めるエンティティに知らされる。
18. *認可プライベート鍵*[49]: 認可プライベート鍵は、権限に対する所有者の権利を証明するために使用（例えば、デジタル署名を使用）される非対称鍵（公開鍵）の鍵ペアのプライベート鍵である。
19. *認可公開鍵*: 認可公開鍵は、関連する認可プライベート鍵を知っているエンティティに対する権限を検証するために使用される非対称鍵（公開鍵）の鍵ペアの公開鍵である。

5.1.2 その他の関連情報

暗号アルゴリズム及び鍵と組み合わせて使用されるその他の情報も保護する必要がある。各タイプの情報に必要な保護については、6.1.2 節の表 6 を参照のこと。

1. *ドメインパラメータ*: ドメインパラメータは、ある公開鍵アルゴリズム[50]と組み合わせて、鍵ペアの生成、デジタル署名の作成、又は鍵材料の確立のために使用される。
2. *初期ベクトル*: 初期ベクトル（IV）は、ブロック暗号アルゴリズムを使った暗号化や復号、及び MAC の計算のために、いくつかの暗号利用モードで使用される[51]。
3. *共有秘密*: 共有秘密は、鍵合意プロセス[52]中に生成される。
4. *RBG シード*: RBG シードは、決定論的乱数ビットの生成に使用される（例えば、秘密又はプライベートのままでなければならない鍵材料を生成するために利用される）[53]。
5. *その他の公開情報*: 公開情報（例えば、ノンス）は、鍵確立プロセスでしばしば使用される。

---

47 SP 800-56A を参照。
48 具体的に承認された方法はないが、対称アルゴリズムのどれでも使用することができる（例：AES、HMAC、KMAC）。
49 具体的に承認された方法はないが、この目的のためにデジタル署名を使用することができる。
50 FIPS 186 及び SP 800-56A を参照。
51 4.2 節を参照。
52 SP 800-56A 及び SP 800-56B を参照。
53 SP 800-90A を参照。

6. *その他の秘密情報*：秘密情報は、RBG シードや鍵材料の確立の際[54]に含まれる場合がある。
7. *中間結果*：暗号処理中の中間結果
8. *鍵制御情報／メタデータ*：鍵材料に関連する情報（識別子、目的、カウンターなど）は、関連する鍵材料を正しく使用できることを保証するために保護されなければならない。鍵制御情報は、鍵材料に関連付けられたメタデータに含まれる（6.2.3.1 節参照）。
9. *乱数*（又は乱数ビット）：乱数ビット生成器によって作成された乱数
10. *パスワード*: パスワードは、権限へのアクセスを取得するために使用され、ソース認証メカニズム又は ID 認証メカニズムの資格情報として使用できる。パスワードは、ストレージ内のデータを保護したりアクセスしたりするために使用される暗号鍵を導出するためにも使用できる[55]。
11. *監査情報*：監査情報には、鍵管理イベントの記録が含まれる。

5.2 鍵の使用法

一般に、一つの鍵は一つの目的（例えば、暗号化、完全性認証、鍵ラッピング、乱数ビット生成、デジタル署名など）にのみ使用**されなければならない**。これにはいくつかの理由がある：

1. 2 つの異なる暗号プロセスに同じ鍵を使用すると、一方又は両方のプロセスで提供されるセキュリティが低下する可能性がある
2. 鍵の使用を制限することで、鍵が危殆化した場合に発生する可能性のある被害を制限する
3. 鍵の用途によっては、互いに干渉することがある。例えば、鍵配送とデジタル署名の両方に使用される鍵ペアを考える。この場合、プライベート鍵は、暗号化された鍵を復号するための鍵配送プライベート鍵としてと、デジタル署名を生成するための署名プライベート鍵としての両方で使用される。暗号化されたデータにアクセスするために必要な暗号化された鍵を復号するために、対応する公開鍵の暗号利用期間を超えて配送鍵に使用されるプライベート鍵を保持する必要があるかもしれない。署名生成に使用されるプライベート鍵は、危殆化を防ぐために、暗号利用期間の満了時に破棄**されなければならない**（5.3.6 節参照）。この例では、鍵配送プライベート鍵の寿命要件とデジタル署名プライベート鍵の寿命要件とが矛盾する。

この原則は、同じプロセスが複数のサービスを提供できる場合に一つの鍵を使用することを排除しない。これには、例えば、一つのデジタル署名を使用して完全性認証とソース認証をデジタル署名が提供する場合や、一つの対称鍵を使用して一つの暗号処理でデータの暗号化と認証を行う（例えば、暗号化と認証の処理を別々に行うのではなく、認証暗号処理を利用する）ことができる場合などが当てはまる。3.7 節を参照のこと。

本推奨では、次の特別な場合のデジタル署名を生成するために、鍵配送プライベート鍵又は鍵合意プライベート鍵を使用することを容認する：

[54] SP 800-90A、SP 800-56A、SP 800-56B 及び SP 800-108 を参照。
[55] SP 800-132 を参照。

FIPS 186（SP 800-56A 及び SP 800-56B を参照）で規定されたように生成された静的な鍵確立鍵の（最初の）証明書を要求する場合、対応するプライベート鍵を使用して証明書要求に署名できる。8.1.5.1.1.2 節を参照。

5.3 暗号利用期間

暗号利用期間とは、特定の鍵を正規のエンティティが使用することが認可されている期間、又は特定のシステムの鍵が有効である期間のことである。適切に定義された暗号利用期間とするために、

1. 暗号解読で鍵を明らかにするために利用可能な情報量（例：当該鍵で暗号化された平文と暗号文のペア数）を制限する
2. 一つの鍵が漏洩した場合の暴露量を制限する
3. 特定のアルゴリズムの使用を制限する（例えば、そのアルゴリズムの推定有効期限に制限する）
4. 鍵を不正な開示から保護する物理的、手続き的、及び論理的なアクセスメカニズムが突破されようとした場合に、その試行が可能な時間を制限する
5. 不正なエンティティへの暗号鍵の不注意な開示によって情報が危殆化されうる期間を制限する
6. 集中的な計算を行う暗号解読に利用可能な時間を制限する

暗号期間は、任意の期間、又は鍵で保護されるデータの最大量によって定義されることがある。しかしながら、暗号期間の決定に伴うトレードオフは漏えいのリスクと結果に影響を与えるので、暗号期間を選択する際には慎重に考慮すべきである（5.6.4 節参照）。

鍵が危殆化した場合、その暗号利用期間はもはや有効とは**みなしてはならない**。危殆化した鍵の取り扱いについては、5.5 節を参照のこと。

5.3.1 暗号利用期間に影響を与える要因

暗号利用期間の長さに影響を与える要因の中には、以下のものがある：

1. 暗号メカニズムの強度（例えば、アルゴリズム、鍵長、ブロックサイズ、及び暗号利用モード）
2. メカニズムの実装形態（FIPS 140 レベル 4 実装か、パーソナルコンピュータ上のソフトウェア実装かなど）。
3. 動作環境（アクセスが限定された安全な施設か、オープンオフィス環境か、一般にアクセス可能な端末かなど）。
4. 要員の離職率（例えば、システム管理者及び CA システム要員の離職率）
5. データフロー量又はトランザクション数
6. データのセキュリティ寿命
7. アルゴリズム使用に必要な制限（例えば、ノンスの再利用を避けるための最大呼び出し回数）
8. セキュリティ機能（データ暗号化、デジタル署名、鍵導出、鍵保護など）

9. 鍵再設定方法（例えば、キーボード入力か、人間が鍵に直接アクセスできない鍵ロードデバイスを使用した鍵再設定か、PKI 内でのリモート鍵再設定か、など）
10. 使用される鍵再設定プロセス又は鍵導出プロセス
11. 共通鍵を共有するネットワーク内のノード数
12. 鍵のコピー数、及びそのコピーの配付先
13. 敵対者からの情報に対する脅威（例えば、攻撃を仕掛けるための想定される技術的能力や資金力など）
14. 新技術や破壊的な技術（量子コンピュータなど）による情報への脅威

一般的に、短い暗号利用期間はセキュリティを強化する。例えば、いくつかの暗号アルゴリズムでは、敵対者が単一の鍵の下で暗号化された限られた量の情報しか持っていない場合、暗号解読に対して脆弱ではないかもしれない。一方で、人為的なエラーや脆弱性を伴う手動鍵配付方法の場合には、鍵の変更をより頻繁に行うことで、実際には鍵の漏えいリスクが高まるかもしれない。このような場合、特に非常に強力な暗号技術がハードウェアで採用されている場合には、管理が貧弱な手動鍵配付をより頻繁に行うよりも、よく管理された手動鍵配付をより少ない回数行う方が賢明な場合がある。

一般的に、強力な暗号方式が採用されている場合、物理的、手続き的及び論理的なアクセス保護への考慮事項は、アルゴリズムや鍵長の要因への考慮事項よりも、暗号利用期間の選択に大きな影響を与えることが多い。**承認された**アルゴリズム、暗号利用モード及び鍵長の場合、敵対者は、暗号攻撃を実装して実行するよりも、システムへの侵入や破壊によるほうが、より少ない必要な時間と資源の支出で鍵にアクセスすることができるかもしれない。

5.3.2 暗号利用期間に影響を与える結果要因

暴露の結果は、情報の機微性、暗号によって保護されたプロセスの臨界性、及び情報又はプロセスの危殆化からの復旧費用によって測定される。機微性には、保護されている情報の寿命（10 分、10 日、10 年など）、及びその情報の保護が失われた場合の潜在的な結果（例えば、不正なエンティティへの情報の開示）が影響する。一般的に、暗号技術によって保護された情報の機微性又はプロセスの臨界性が高まるにつれて、関連する暗号利用期間の長さは、それぞれの危殆化から生じる可能性のある被害を制限するために短縮**すべきである**。これは、鍵再設定プロセス又は鍵導出プロセスの安全性と完全性に関する注意事項に従うものとする (8.2.3 節及び 8.2.4 節参照)。しかしながら、より短い暗号利用期間は、特に DoS 攻撃が最も重要な関心事であり、鍵再設定プロセスや鍵導出プロセスでエラーが発生する可能性が大きい場合には、逆効果であるかもしれない。

5.3.3 暗号利用期間に影響を与えるその他の要因

5.3.3.1 通信対ストレージ

通信のやり取りの機密保護のために使用される鍵は、保存データの保護のために使用される鍵よりも暗号利用期間が短いことが多い。暗号利用期間が一般に保存データ用のほうが長くなるのは、新しい鍵を生成し、古い鍵で暗号化されたデータを全て再暗号化するためのオーバーヘッドが負担になるためである。

5.3.3.2 鍵の失効と交換のコスト

場合によっては、鍵の変更に関連するコストが痛々しいほど高い。例としては、非常に大規模なデータベースの復号とその後の再暗号化、分散型データベースの復号と再暗号化、非常に多数の鍵の失効と交換（例えば、非常に多数の地理的にも組織的にも分散した鍵保有者が存在する場合）などが挙げられる。そのような場合、より長い暗号利用期間をサポートするために必要なセキュリティ対策の費用が正当化されるかもしれない（例えば、費用がかかって不便な物理的、手続き的及び論理的アクセスセキュリティや、長い暗号利用期間をサポートするのに十分な強力であり大幅な追加のオーバーヘッド処理をもたらしうる暗号技術の使用など）。その他のケースでは、暗号利用期間を必要とされるよりも短くすることがある。例えば、鍵管理システムがステータス情報を維持する期間を制限するために、鍵を頻繁に変更する。

5.3.4 非対称鍵の使用期間と暗号利用期間

非対称鍵ペアの場合、ペアの各鍵はそれぞれが自身の暗号利用期間を持つ。鍵ペアの一方の鍵は暗号保護の適用（例えば、電子署名の作成）に使用され、その暗号利用期間は“作成者使用期間”と呼ばれる。鍵ペアのもう一方の鍵は、保護された情報の処理（例えば、電子署名の検証）に使用され、その暗号利用期間は“受領者使用期間”と呼ばれる。鍵ペアの作成者使用期間と受領者使用期間は通常同時に始まるが、受領者使用期間は作成者使用期間を超えて延長することができる。例えば：

- デジタル署名鍵ペアの場合、署名プライベート鍵はデータの署名（すなわち、暗号保護の適用）に使用されるため、その暗号利用期間は作成者使用期間と見なされる。署名検証公開鍵は、デジタル署名の検証（すなわち、既に保護された情報の処理）に使用されるため、その暗号利用期間は受領者使用期間と見なされる。

  作成者証明として（すなわち、ソース認証のため）のデジタル署名を生成するために使用される署名プライベート鍵の場合、作成者使用期間（すなわち、プライベート鍵が署名を生成するために使用される期間）は、受領者使用期間（すなわち、署名検証公開鍵によって署名が検証される期間）よりも短くなることが多い。この場合、プライベート鍵は一定期間での使用を意図しており、その後、鍵所有者は当該プライベート鍵を破棄**しなければならない** [56]。公開鍵は、署名を検証するために、より長い期間利用可能であってもよい。

  チャレンジ情報に署名するために使用されるソース認証プライベート鍵の暗号利用期間は、基本的に、関連する公開鍵（すなわち、ソース認証公開鍵）の暗号利用期間と同じである。すなわち、プライベート鍵がチャレンジ情報の署名に使用されなくなった場合には、公開鍵は不要となる。この場合、作成者使用期間と受領者使用期間は同じである。

- 鍵配送鍵の場合、鍵配送公開鍵は保護の適用（すなわち、データの暗号化）に使用されるため、その暗号利用期間は作成者使用期間と見なされる。鍵配送プライベート鍵は暗号化されたデータの復号に使用されるため、その暗号利用期間は受領者使用期間と見なされる。

  作成者使用期間（すなわち、公開鍵が暗号化に使用される期間）は、受領者使用期間（すなわち、暗号化された情報が復号される期間）よりも短いことが多い。

---

[56] 鍵材料を単純に削除しても、情報を完全には消去できない場合がある。例えば、情報を消去するためには、当該情報に、ランダムなビット値や全て 0 か 1 のビット値など、関連性のない他の情報を複数回上書きする必要があるかもしれない。メモリに長期間保存された鍵は、“焼き付け”られる可能性がある。鍵を鍵シェアに分割し頻繁に更新することで、この問題を軽減することができる（[DiCrescenzo]参照）。

- For key-agreement algorithms, the cryptoperiods of the two keys of the key pair are usually the same.
- 鍵合意アルゴリズムの場合、鍵ペアの 2 つの鍵の暗号利用期間は通常同じである。

公開鍵が公開鍵証明書で配付される場合、各証明書は証明書の *notBefore* と *notAfter* の日付で示される有効期間を持つ。証明書は更新することができる（すなわち、同じ公開鍵を含む新しい証明書が新たな有効期間で発行されることがある）。元の証明書及び同一の公開鍵に対する全ての更新された証明書の有効期間の範囲は、保護を適用するために使用された鍵ペアの鍵の暗号利用期間（すなわち、作成者使用期間にある鍵）の開始日及び終了日を**超えてはならない**。

特定の鍵タイプに関するガイダンスについては、5.3.6 節を参照のこと。

5.3.5 対称鍵の使用期間と暗号利用期間

対称鍵の場合、保護の適用（データの暗号化や MAC の計算など）と保護された情報の処理（暗号化されたデータの復号や MAC の検証など）の両方に単一の鍵が使用される。データに暗号保護が適用される期間を作成者使用期間といい、保護された情報が処理される期間を受領者使用期間という。対称鍵は、作成者使用期間終了後は、保護を行うために**使ってはならない**。受領者使用期間は、作成者使用期間を超えて延長することができる（図 1 参照）。これにより、作成者によって保護された全ての情報は、保護が適用された後、受領者が延長された期間にわたって処理することが可能になる。しかし、多くの場合、作成者使用期間と受領者使用期間は同じである。対称鍵の（全体の）“暗号利用期間”は、歴史的には鍵の暗号利用期間として作成者使用期間が使用されてきたが、作成者使用期間の開始から受領者使用期間の終了までの期間である。

場合によっては、予め定められた暗号利用期間が、保護されたデータのセキュリティ寿命に対して適切でない場合があることに注意されたい。必要なセキュリティ寿命が暗号利用期間を超える場合は、新しい鍵を用いて保護を再適用する必要がある。

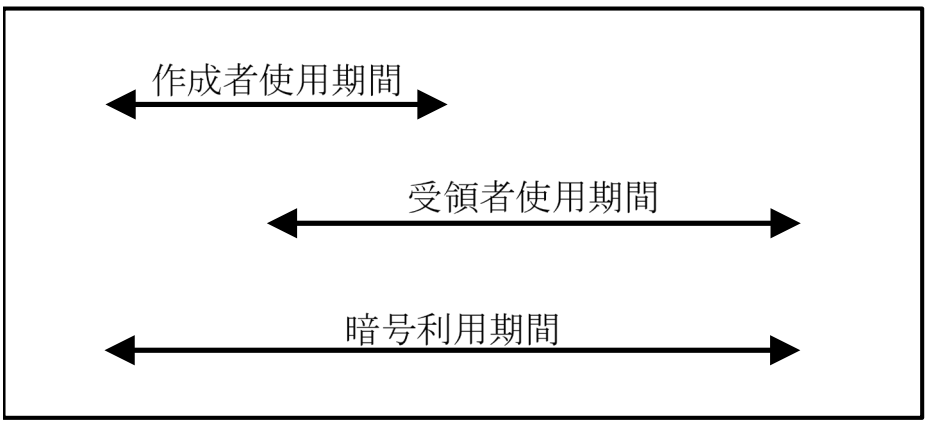

図 1：対称鍵の暗号利用期間

対称鍵の使用期間の例としては、以下のようなものがある：

a. 対称鍵が通信の安全を確保するためにのみ使用される場合、発信者が保護を適用してから受信者が処理するまでの期間は無視できるほど短いかもしれない。この場合、鍵は、暗号利用期間全体の間（すなわち、作成者使用期間と受領者使用期間は同じ）、どちらの目的でも認可される。

b. 対称鍵が保存された情報を保護するために使用される場合、作成者使用期間（作成者が保存された情報に暗号保護を適用する期間）は、受領者使用期間（保存された情報が処理される期間）よりも非常に早く終了することがある。この場合、暗号利用期間は、その鍵を用いた保護の適用が認可された最初の時刻から始まり、当該鍵を用いた処理が認可された最後の時刻で終了する。一般に、保存された情報の受領者使用期間は作成者使用期間を超えて継続され、保存された情報が後の時点で認証又は復号されてもよい。

c. 対称鍵が保存された情報を保護するために使用される場合、受領者使用期間は、図 1 に示すように、作成者使用期間の開始後に始まってもよい。例えば、情報は、ある記憶媒体に格納される前に暗号化されてもよい。後日、情報を復号して復元するために、鍵が配付されてもよい。

5.3.6 特定の鍵タイプに対する暗号利用期間の推奨事項

上述の鍵タイプ、使用環境及びデータの特性は、与えられた鍵の必要な暗号利用期間に影響を与える可能性がある。様々な鍵タイプに対する推奨暗号利用期間を以下に示す。推奨されている暗号利用期間は、あくまでも大まかなガイドラインであり、当該鍵が使用されるアプリケーションや環境によっては、より長い暗号利用期間やより短い暗号利用期間が正当化される場合があることに注意されたい。しかし、以下の推奨暗号利用期間よりも長い暗号利用期間を設定する場合には、それに伴うリスクを十分に考慮**すべきである**（5.3.1 節参照）。推奨暗号利用期間の多くは、運用効率を最大化したいという要望と、使用環境の最低基準を想定したものを基準としている（FIPS 140 及び SP 800-37 参照）。5.3.1 節から 5.3.3 節に記載された要因を用いて、特定の使用環境における実際の暗号利用期間を決定**すべきである**。

1. *署名プライベート鍵*:

   a. タイプの考慮事項：一般に、署名プライベート鍵の暗号利用期間は、対応する署名検証公開鍵の暗号利用期間よりも短くてもよい。対応する公開鍵が CA によって認証されている場合、署名プライベート鍵の暗号利用期間は、当該公開鍵に対して最後に発行された証明書[57]の *notAfter* の日付に到達した時点で終了する。

   b. 暗号利用期間：**承認された**アルゴリズムと鍵長を使用すること、及び鍵の保管及び使用環境のセキュリティが、当該鍵が完全性保護を提供するプロセスの機微性や重要度が高まるにつれて、向上することが期待されることを考慮すると、最大で約 1 年から 3 年の暗号利用期間が推奨される。署名プライベート鍵は、その暗号利用期間の終了時に破棄**されなければならない**。

2. *署名検証公開鍵*:

   a. タイプの考慮事項：一般に、署名検証公開鍵の暗号利用期間は、対応する署名プライベート鍵の暗号利用期間よりも長くてもよい。暗号利用期間とは、実質的には、対応する署名プライベート鍵を使用して計算された署名を検証する必要がある期間のことである。署名

[57] 同じ公開鍵に複数回の継続的な証明書が発行される可能性があり、その場合、おそらく異なる *notBefore* 及び *notAfter* の有効期限が設定される。

検証公開鍵の暗号利用期間を（署名プライベート鍵の暗号利用期間よりも）長くしても、セキュリティ上の懸念は比較的少ない。

b. 暗号利用期間：暗号利用期間は数年のオーダである。しかし、保護メカニズムが敵対的な攻撃に長くさらされるため、署名の信頼性は時間の経過とともに低下する。つまり、与えられたアルゴリズムと鍵長にかかわらず、暗号解読に対する脆弱性は時間の経過とともに増大すると予想される。利用可能な最も強力なアルゴリズムと大きな鍵長を選択することで、暗号解読に対するこの脆弱性を最小化することができるが、プライベート鍵に対する物理的、手続き的、及び論理的なアクセス制御メカニズムに対する攻撃にさらされた時の結果には影響を受けない。

いくつかのシステムでは、暗号学的タイムスタンプ機能を使用して、各々の署名済メッセージに偽造不可能なタイムスタンプを打つ。署名プライベート鍵の暗号利用期間が満了した場合でも、対応する署名検証公開鍵を用いて、当該署名プライベート鍵の暗号利用期間内にタイムスタンプがあるメッセージの署名を検証することができる。この場合の検証者は、メッセージが署名プライベート鍵の作成者使用期間内に署名されたことの保証を提供する暗号学的タイムスタンプ機能に頼ることになる。

3. *認証対称鍵*：

a. タイプの考慮事項：認証対称鍵[58]の暗号利用期間は、保護される情報タイプの機微性、及び鍵と関連するアルゴリズムによって与えられる保護に依存する。非常に機微性の高い情報の場合、認証鍵は保護された情報ごとに変える必要があるかもしれない。機微性がより低い情報については、適切な暗号利用期間であれば、鍵を複数回使用してもよい。認証対称鍵の作成者使用期間は、情報に対して元の暗号保護を適用する（例えば、MAC を計算する）際に当該鍵を使用する場合に適用される。作成者使用期間の終了後に、情報に対する新しい MAC に当該鍵を使って計算**してはならない**。しかし、当該鍵は、作成者使用期間を超えて保護されたデータ上の MAC を検証するために利用可能である必要があるかもしれない（すなわち、受領者使用期間は、作成者使用期間を超えて延長される可能性がある）。受領者使用期間は、作成者使用期間中に生成された MAC を検証する必要がある期間である。MAC 鍵が危殆化した場合、敵対者がデータを修正して、その後に MAC を再計算する可能性があることに注意されたい。

b. 暗号利用期間：**承認された**アルゴリズムと鍵長を使用すること、及び鍵の保管及び使用環境のセキュリティが、当該鍵が完全性保護を提供するプロセスの機微性や重要度が高まるにつれて、向上することが期待されることを考慮すると、作成者使用期間は 2 年以下が推奨され、受領者使用期間は作成者使用期間の終了後 3 年を超えないようにすることが推奨される。

4. *認証プライベート鍵*：

a. タイプの考慮事項：認証鍵プライベートは、データの完全性認証及び ID 認証を可能にするために、複数回使用される可能性がある。例えば、認証局は、対応する公開鍵を認証することができる。ほとんどの場合、認証プライベート鍵の暗号利用期間は、対応する公開鍵の暗号利用期間と同じである。

b. 暗号利用期間：認証プライベート鍵の適切な暗号利用期間は、その利用環境や認証情報の機微性・重要性に依存するが、1 年から 2 年以内である。

[58] データの完全性及びソース認証を可能にするために使用する。

5. *認証公開鍵*:

   a. タイプの考慮事項：多くの場合、認証公開鍵の暗号利用期間は、対応する認証プライベート鍵の暗号利用期間と同じである。暗号利用期間は、実質的に、対応する認証プライベート鍵で保護された情報の作成者の身元を検証する必要がある（すなわち、身元を認証する必要がある）[59]期間である。

   b. 暗号利用期間：認証公開鍵の適切な暗号利用期間は、その利用環境や認証情報の機微性・重要度に依存するが、1 年から 2 年以内である。

6. *データ暗号化対称鍵*:

   a. タイプの考慮事項：データ暗号化対称鍵は、保存データ、メッセージ、又は通信セッションを保護するために使用される。危殆化した場合の結果を主に考慮して、大量のデータを短時間に暗号化（例えばリンク暗号化）するために使用されるデータ暗号化鍵は、比較的短い作成者使用期間と**すべきである**。時間をかけてより少ないデータを暗号化するために使用される暗号化鍵は、より長い作成者使用期間を持つ可能性がある。データ暗号化対称鍵の作成者使用期間は、情報を暗号化するために当該鍵を使用する場合に適用される（5.3.5 節参照）。

      作成者使用期間中、データの暗号化はデータ暗号化鍵を用いて実行してもよいが、この期間を超えて、当該鍵をデータの暗号化処理を実行するために使用**してはならない**。しかし、当該鍵は、作成者使用期間を超えて保護されたデータを復号するために利用可能である必要があるかもしれない（すなわち、受領者使用期間は、作成者使用期間を超えて延長する必要があるかもしれない）。

   b. 暗号利用期間：大量のデータを短期間に暗号化（例えば、リンク暗号化）するために推奨される作成者使用期間は、1 日から 1 週間のオーダである。より少量のデータを暗号化するために使用される暗号化鍵は、最大 2 年の作成者使用期間を持つかもしれない。受領者使用期間は、作成者使用期間の終了後 3 年を超えないようにすることが推奨される。

      メッセージ又は通信セッションを単独で暗号化するために使用されるデータ暗号化対称鍵の場合、暗号化されたメッセージは後で読むために保存される可能性があるため、保護されたデータの寿命は数ヶ月又は数年になる可能性がある。データが暗号化された形で維持されている場合、データ暗号化対称鍵は、データが新しい鍵で再暗号化されるか破棄されるまで、維持される必要がある。データの機密性の信頼性は、時間の経過とともに低下することに注意されたい。

7. *鍵ラッピング対称鍵*:

   a. タイプの考慮事項：短期間に非常に多くの鍵をラッピング（すなわち、暗号化と完全性保護）するために使用される鍵ラッピング対称鍵は、比較的短い作成者使用期間を**持つべきである**。少数の鍵がラッピングされる場合、鍵ラッピング鍵の作成者使用期間はより長くなる可能性がある。鍵ラッピング対称鍵の作成者使用期間は、鍵の鍵ラッピング保護を提供する際に当該鍵を使用する場合に適用される。作成者使用期間の終了後、ラッピング処理に当該鍵を使って**行ってはならない**。しかし、鍵ラッピング鍵は、作成者使用期間を超えて保護された鍵をアンラップする（すなわち、ラップされた鍵を復号して完全性検証する）ために利用可能である必要があるかもしれない（すなわち、受領者使用期間は、作成者使用期間を超えて延長する必要があるかもしれない）：受領者使用期間とは、鍵ラッピ

[59] 完全性保護も提供されるが、それはこの鍵の主要な目的ではない。

ング鍵の作成者使用期間中にラップされた鍵をアンラップする必要があるかもしれない期間である。

鍵ラッピング対称鍵の中には、1 回のメッセージ又は通信セッションにのみ使用されるものもある。これらの非常に短期の鍵ラッピング鍵の場合、適切な暗号利用期間（すなわち、作成者使用期間と受領者使用期間の両方を含む）は、1 回の通信セッションである。ラッピングされた鍵がラッピングされた状態で保持されないことを前提としているので、鍵ラッピング鍵の作成者使用期間と受領者使用期間は同じである。他のケースでは、ラッピングされた鍵によって暗号化されたファイルやメッセージを後から復元できるように、鍵ラッピング鍵を保持してもよい。この場合、受領者使用期間は、鍵ラッピング鍵の作成者使用期間よりも大幅に長く、何年にもわたる暗号利用期間を採用する場合がある。

b. 暗号利用期間：非常に多くの鍵を短時間でラッピングするために使用される鍵ラッピング対称鍵の場合、推奨される作成者使用期間は 1 日から 1 週間のオーダである。比較的少数の鍵を鍵ラッピング鍵でラッピングする場合、鍵ラッピング鍵の作成者使用期間は最大 2 年である。単独のメッセージや通信セッションにのみ使用される鍵ラッピング鍵の場合、暗号利用期間は当該の通信セッションに限定される。受領者使用期間は、作成者使用期間の終了後 3 年を超えないようにすることが推奨される。

8. *RBG 対称鍵*：

a. タイプの考慮事項：RBG 対称鍵は、決定論的乱数ビット生成関数で使用される。SP 800-90 で**承認された** RBG は、（例えば、再シード中の）鍵の変更を制御する。暗号利用期間は、作成者使用期間のみで構成される。

b. 暗号利用期間：**承認された** RBG を使用すると仮定して、RBG 対称鍵の最大暗号利用期間は、RBG の設計によって決定される（SP 800-90 参照）。

9. *マスター対称鍵／鍵導出対称鍵*：

a. タイプの考慮事項：マスター対称鍵（鍵導出鍵とも呼ばれる）は、（一方向の）鍵導出関数又は鍵導出方法（8.2.4 節参照）を使用して他の鍵を導出するために、複数回使用される可能性がある。したがって、暗号利用期間は、この鍵タイプの場合、作成者使用期間のみで構成される。適切な暗号利用期間は、マスター鍵から導出した鍵の性質と用途、及び 5.3 節のはじめに述べた考慮事項に依存する。マスター鍵から導出した鍵の暗号利用期間は比較的短くてもよい（例えば、一回の使用、通信セッショ ン、トランザクションなど）。あるいは、マスター鍵をより長い期間使用して、同じ目的又は異なる目的のために複数の鍵を導出（又は再導出）することも可能である。導出した鍵の暗号利用期間は、その用途（例えば、データ暗号化対称鍵用、完全性認証鍵用など）に依存する。

b. 暗号利用期間：マスター対称鍵の適切な暗号利用期間は、その利用環境、導出鍵で保護される情報の機微性・重要性、及びマスター鍵から導出した鍵の数に依存するが、1 年である。

10. *鍵配送プライベート鍵*：

a. タイプの考慮事項：鍵配送プライベート鍵は、鍵を復号するために複数回使用される可能性がある。配送のために暗号化された鍵を、その後に復号する必要が生じるため、鍵配送プライベート鍵の暗号利用期間は、関連する公開鍵の暗号利用期間よりも長くなる可能性がある。鍵配送プライベート鍵の暗号利用期間は、対応する鍵配送公開鍵で暗号化された鍵を復号する必要がある期間の長さである。

b. 暗号利用期間：1）**承認された**アルゴリズムと鍵長の使用、2）対応する鍵配送公開鍵で暗号化される鍵を用いて保護される可能性のある情報量、及び 3）鍵の保管及び使用環境のセキュリティが、当該鍵が保護を提供するプロセスの機微性や重要度が高まるにつれて、向上することが望まれることを考えると、鍵配送プライベート鍵の暗号利用期間は 2 年以下が推奨される。受信したメッセージを保存し、後で復号するような特定のアプリケーション（例えば、電子メール）では、鍵配送プライベート鍵の暗号利用期間は鍵配送公開鍵の暗号利用期間を超える可能性がある。

11. *鍵配送公開鍵*：

a. タイプの考慮事項：鍵配送公開鍵の暗号利用期間は、配送中に保護される鍵に暗号化処理を実際に適用するための公開鍵を使用することができる期間である。公開鍵が CA によって認証されている場合、公開鍵の暗号利用期間は、当該公開鍵に対して最後に発行された証明書の *notAfter* の日付に到達した時点で終了する。

鍵配送公開鍵は、公知にすることもできる。鍵配送プライベート鍵の議論で示したように、配送のために暗号化された鍵を、その後の時点で復号する必要が生じる可能性があるため、鍵配送公開鍵の暗号利用期間は、対応するプライベート鍵の暗号利用期間よりも短くてもよい。

b. 暗号利用期間：対応するプライベート鍵の暗号利用期間の前提に基づき、暗号利用期間は 1 年から 2 年以下を推奨する。

12. *鍵合意対称鍵*：

a. タイプの考慮事項：鍵合意対称鍵は複数回使用することができる。鍵合意対称鍵の暗号利用期間は、1）環境セキュリティ上の要因、2）確立される鍵の性質（タイプ、フォーマットなど）と量、及び 3）採用される鍵合意アルゴリズム及び鍵合意プロトコルの詳細に依存する。鍵合意対称鍵は、対称鍵（例えば、データ暗号化対称鍵）や他の鍵材料（例えば、IV）を確立するために使用されるかもしれないことに注意されたい。

b. 暗号利用期間：鍵合意対称鍵を使用する暗号技術が、1）**承認された**アルゴリズムと鍵スキームを使用し、2）暗号デバイスが FIPS 140 の要件を満たし、3）リスクレベルが FIPS 199[60]に準拠して確立されていると仮定すると、当該鍵の適切な暗号利用期間は、1 年から 2 年以下である。受信したメッセージが保存され、後で復号される特定のアプリケーション（例えば、電子メール）では、鍵の受領者使用期間は作成者使用期間を超える可能性がある。

13. *静的鍵合意プライベート鍵*：

a. タイプの考慮事項：静的（すなわち、長期的）な鍵合意プライベート鍵は、複数回使用することができる。CA が対応する公開鍵を認証する場合、静的鍵合意プライベート鍵の暗号利用期間は、対応する公開鍵に対して最後に発行された証明書の *notAfter* の日付に到達した時点で終了する。

鍵合意対称鍵の場合と同様に、この鍵の暗号利用期間は、1）環境セキュリティ上の要因、2）確立される鍵の性質（タイプ、フォーマットなど）と量、及び 3）採用される鍵合意アルゴリズム及び鍵合意プロトコルの詳細に依存する静的鍵合意プライベート鍵は、対称鍵（例えば、鍵ラッピング鍵）やその他の秘密の鍵材料を確立するために使用されるかもしれないことに注意されたい。

[60] FIPS 199, *Standard for Security Categorization of Federal Information Systems*.

b. 暗号利用期間：静的鍵合意プライベート鍵を使用する暗号技術が、1）承認されたアルゴリズムと鍵スキームを使用し、2）暗号デバイスが FIPS 140 の要件を満たし、3）リスクレベルが FIPS 199 に準拠して確立されていると仮定すると、当該鍵の適切な暗号利用期間は、1 年から 2 年以下である。静的鍵合意プライベート鍵と静的鍵合意公開鍵の暗号利用期間は通常同じであるが、受信したメッセージが保存され、後で復号される特定のアプリケーション（例えば、電子メール）では、静的鍵合意プライベート鍵の暗号利用期間が、対応する静的鍵合意公開鍵の暗号利用期間を超えてしまうことがある。

14. *静的鍵合意公開鍵*：

a. タイプの考慮事項：静的（すなわち、長期的）鍵合意公開鍵の暗号利用期間は、通常、対応する静的鍵合意プライベート鍵の暗号利用期間と同じである。

b. 暗号利用期間：静的鍵合意公開鍵の暗号利用期間は 1 年又は 2 年である。

15. *一時的鍵合意プライベート鍵*：

a. タイプの考慮事項：一時的（すなわち、短期的）鍵合意プライベート鍵は、1 つ以上の鍵を確立するために、1 回のトランザクションで使用される非対称鍵ペアのプライベート鍵要素である。一時的鍵合意プライベート鍵は、対称鍵（例えば、鍵ラッピング鍵）やその他の秘密の鍵材料を確立するために使用されてもよい。

b. 暗号利用期間：一時的鍵合意プライベート鍵は、1 回の鍵合意トランザクションに使用される。しかし、一時的プライベート鍵は、複数回使用して、同一のトランザクション（ブロードキャスト）中に複数の当事者と同一の対称鍵を確立することができる。一時的鍵合意プライベート鍵の暗号利用期間は、1 回の鍵合意トランザクションの期間である。

16. *一時的鍵合意公開鍵*：

a. タイプの考慮事項：一時的（すなわち、短期的）鍵合意公開鍵は、1 つ以上の鍵を確立するために一度だけ使用される非対称鍵ペアの公開鍵要素である。

b. 暗号利用期間：一時的鍵合意公開鍵は、1 回の鍵合意トランザクションに使用される。一時的鍵合意公開鍵の暗号利用期間は、共有秘密の生成に使用された直後に終了する。いくつかのケースでは、一時的鍵合意公開鍵の暗号利用期間は、鍵合意トランザクションの参加者によって異なるかもしれないことに注意されたい。例えば、暗号化された電子メールアプリケーションを考える。電子メールの送信者が一時的鍵合意の鍵ペアを生成し、その鍵ペアを使用して電子メールの内容を暗号化するために使用される暗号化鍵を生成する。送信者にとっては、共有秘密を生成し、暗号化鍵が導出された時点で、公開鍵の暗号利用期間は終了する。しかし、暗号化された電子メールの受信者にとっては、共有秘密が生成されて復号鍵が決定されるまでは、一時的公開鍵の暗号利用期間は終了しない。電子メールを受信してすぐに処理されない場合（例えば、電子メールの送信から 1 週間後に復号するなど）、当該公開鍵を用いて共有秘密が生成されるまでは、一時的公開鍵の暗号利用期間は（受信者側から見て）終了しない。

17. *認可対称鍵*：

a. タイプの考慮事項：認可対称鍵は、保護されるリソース及びアクセスが認可されたエンティティの役割に依存して、長期間使用される可能性がある。この鍵タイプの場合、作成者使用期間と受領者使用期間は同じである。認可対称鍵の暗号利用期間を設定する際の主な考慮事項には、鍵の堅牢性、暗号化方式の妥当性、及び鍵保護のメカニズムと手続きの妥当性が含まれる。

b. 暗号利用期間：**承認された**アルゴリズムと鍵長を使用し、鍵の保管及び使用環境のセキュリティが、認可プロセスの機微性や重要度が高まるにつれて、向上することが望まれることを考えると、暗号利用期間は 2 年以下が推奨される。

18. *認可プライベート鍵*：

a. タイプの考慮事項：認可プライベート鍵は、保護されるリソース及びアクセスが許可されたエンティティの役割に依存して、長期間使用される可能性がある。認可プライベート鍵の暗号利用期間を設定する際の主な考慮事項には、鍵の堅牢性、暗号化方式の妥当性、及び鍵保護のメカニズムと手順の妥当性が含まれる。認可プライベート鍵とそれに対応する公開鍵の暗号利用期間は同一で**なければならない**。

b. 暗号利用期間：**承認された**アルゴリズムと鍵長を使用し、鍵の保管及び使用環境のセキュリティが、認可プロセスの機微性や重要度が高まるにつれて、向上することが望まれることを考えると、認可プライベート鍵の暗号利用期間は 2 年以下が推奨される。

19. *認可公開鍵*：

a. タイプの考慮事項：認可公開鍵は、対応するプライベート鍵を保有するエンティティの権限を検証するために使用される非対称鍵ペアの公開鍵要素である。

b. 暗号利用期間：認可公開鍵の暗号利用期間は、認可プライベート鍵の暗号利用期間（2 年以下）と同じで**なければならない**。

以下の表 1 は、鍵タイプごとに推奨されている暗号利用期間の概要である。鍵が使用されるアプリケーションや環境によっては、より長い暗号利用期間やより短い暗号利用期間が正当化される場合がある。しかし、以下の推奨暗号利用期間よりも長い暗号利用期間を設定する場合には、それに伴うリスクを十分に考慮**すべきである**（5.3.1 節参照）。

表 1：鍵タイプごとの推奨暗号利用期間

鍵タイプ	暗号利用期間	
	作成者使用期間（OUP）	受領者使用期間
1. 署名プライベート鍵	1～3 年	–
2. 署名検証公開鍵	数年（鍵長に依存する）	
3. 認証対称鍵	2 年以下	OUP＋3 年以下
4. 認証プライベート鍵	1～2 年	
5. 認証公開鍵	1～2 年	
6. データ暗号化対称鍵	2 年以下	OUP＋3 年以下
7. 鍵ラッピング対称鍵	2 年以下	OUP＋3 年以下
8. RBG 対称鍵	SP 800-90 参照	–
9. マスター対称鍵／鍵導出対称鍵	約 1 年	–

10. 鍵配送プライベート鍵	2 年以下 [61]
11. 鍵配送公開鍵	1～2 年
12. 鍵合意対称鍵	1～2 年 [62]
13. 静的鍵合意プライベート鍵	1～2 年 [63]
14. 静的鍵合意公開鍵	1～2 年
15. 一時的鍵合意プライベート鍵	1 回の鍵合意トランザクション
16. 一時的鍵合意公開鍵	1 回の鍵合意トランザクション
17. 認可対称鍵	2 年以下
18. 認可プライベート鍵	2 年以下
19. 認可公開鍵	2 年以下

5.3.7 その他の関連情報の推奨

鍵以外の情報は、十分に確立された暗号利用期間となるものを持たない。そのため、その他の情報の処分について、以下のように推奨する。

1. ドメインパラメータは変更されるまで有効なままである。
2. IV は保護に役立つ情報に関連付けられており、暗号化された形で保護された情報が不要になるまで必要とされる。
3. 鍵合意スキームの実行中に生成された共有秘密は、鍵材料の導出に必要とされなくなった時点で破棄**されなければならない**。
4. RBG シードは、使用後直ちに破棄**されなければならない**。
5. その他の公開情報は、暗号処理に必要な期間を超えて保持**すべきではない**。
6. その他の秘密情報は、必要以上に長く保持**してはならない**。
7. 中間結果は、使用後直ちに破棄**しなければならない**。

[61] 受信したメッセージを保存し、後で復号するような特定の電子メールアプリケーションでは、鍵配送プライベート鍵の暗号利用期間が鍵配送公開鍵の暗号利用期間を超えることがある。

[62] 受信したメッセージを保存し、後で復号するような特定の電子メールアプリケーションでは、鍵の受領者使用期間が作成者使用期間を超えることがある。

[63] 受信したメッセージを保存し、後で復号するような特定の電子メールアプリケーションでは、静的鍵合意プライベート鍵の暗号利用期間が、対応する静的鍵合意公開鍵の暗号利用期間を超えることがある。

5.4 保証

鍵材料（鍵、IV、ドメインパラメータなど）が保存又は配付される場合、それが保護されていない環境を通過する可能性がある。このような場合、鍵材料を使用して通常の暗号処理を実行する前に、特定の保証が必要である。

5.4.1 完全性の保証（完全性保護）

全ての鍵材料を使用する前に、完全性の保証を得**なければならない**。

最低限、完全性の保証は、鍵材料が適切な形式であり、認可されたソースから入手したものであることを検証することによって、得**なければならない**。さらに、完全性の保証は、エラー検出コード、メッセージ認証コード及びデジタル署名を適切に使用して、鍵材料が変更されていないことを保証することによって、得る**べきである**。例えば、保存された鍵情報にメッセージ認証コードを生成して、鍵情報が保存中に変更されていないことを保証することができる。メッセージ認証コードは、簡単にアクセスできるように鍵情報とともに保存することができる。

5.4.2 ドメインパラメータの有効性の保証

ドメインパラメータは離散対数公開鍵アルゴリズムで使用され、鍵ペアの生成、デジタル署名の生成、署名検証、及び後に鍵材料を導出するために使用する共有秘密の生成（鍵合意スキームの実行中）のなかで使用される。ドメインパラメータの有効性の保証は、公開鍵暗号のアプリケーションにとって重要であり、それらを使用する前に得**なければならない**。

無効なドメインパラメータは、当該ドメインパラメータを使用している全てのエンティティに対して、意図したセキュリティを全て無効にする可能性がある。デジタル署名アルゴリズムのドメインパラメータの有効性の保証を得る方法は、SP 800-89[64]に規定されている。有限体上及び楕円曲線上の離散対数鍵合意アルゴリズムのドメインパラメータの有効性の保証を得る方法は、SP 800-56A に記載されている。

これらのアルゴリズムについて公開鍵が CA によって認証されている場合、CA は認証プロセス中にこの保証を得ることができることに留意されたい。そうでない場合は、鍵ペアの所有者と依拠当事者が、保証を得る責任を負う。

5.4.3 公開鍵の有効性の保証

公開鍵の有効性の保証は、全ての公開鍵を使用する前に得**なければならない**。

公開鍵の有効性の保証は、公開鍵が計算上正しいという確信をユーザに与える。これにより、脆弱な鍵や破損した鍵を使用する可能性が低くなる。無効な公開鍵は、意図したセキュリティを無効にする可能性がある。例えば、処理（例えば、デジタル署名の生成や鍵確立など）を行ったセキュリティが無効になる、所有者のプライベート鍵から一部又は全部の情報が漏えいする、無効な公開鍵と組み合わせたプライベート鍵についての一部又は全部の情報が漏えいする（鍵合意や公開鍵の暗号化を実行する際に行われる可能性がある）などがある。公開鍵の有効性の保証を得るためのいくつかの方法の一つは、エンティティが公開鍵の持つべき数学的な特性を検証することである。別の方法としては、信頼された第

[64] SP 800-89, *Recommendation for Obtaining Assurances for Digital Signature Applications*.

三者（例えば、CA）から、その信頼された第三者がそのプロパティを検証したという保証を得ることである。

DSA、ECDSA 及び RSA デジタル署名アルゴリズムの公開鍵の有効性の保証を得る方法は、SP 800-89 で提供されている。有限体上及び楕円曲線上の離散対数鍵確立スキームについての公開鍵の有効性の保証を得る方法は、SP 800-56A に記載されている。RSA 鍵確立スキームの（部分的な）公開鍵の有効性の保証を得る方法は、SP 800-56B に規定されている。

5.4.4 プライベート鍵保有の保証

静的（すなわち長期的）プライベート鍵の保有の保証は、対応する静的公開鍵を使用する前に得**なければならない**。公開鍵の有効性の保証は、常に、プライベート鍵の所持の保証に先立って又は同時に得**なければならない**。正しいプライベート鍵の保有の保証は、鍵ペアの所有者が得**なければならない**（例えば、利用可能であり、使用前に変更されていないことを確認するなど）。公開鍵を受領したエンティティは、鍵ペアの所有者が受領した公開鍵に対応するプライベート鍵を保有していることの保証を得**なければならない**。

鍵確立プライベート鍵の保有の保証についての具体的な詳細については、SP 800-56A 及び SP 800-56B を参照のこと。デジタル署名プライベート鍵の保有の保証についての具体的な詳細については、SP 800-89 を参照のこと。CA が認証する公開鍵については、CA は認証プロセス中にこの保証を得ることができることに留意されたい。そうでない場合は、所有者と依拠当事者が、保証を得る責任を負う。

5.4.5 鍵確認

鍵確立とは、その後の使用のためにエンティティ間で、通常はペアのエンティティ間で、鍵材料を安全に確立するプロセスである。鍵確認とは、これらのエンティティが実際に同じ鍵材料を共有していることを保証するために使用される手順である。この手順は強く推奨されており、鍵確立プロセス内で、又はプロセス終了後に実行できる。SP 800-56A 及び SP 800-56B は、自動鍵確立中に実行される鍵確認について記述している。

5.5 鍵及びその他の鍵材料の危殆化

暗号メカニズムによって保護された情報は、アルゴリズムが強力であり、鍵が危殆化していない場合にのみ安全である。プライベート鍵又は秘密鍵の所有者は、その鍵の機密性を保護する責任がある。鍵の危殆化は、鍵の保護メカニズムが機能しなくなり（例えば、鍵の機密性、完全性、又は鍵の所有者との関連付けが機能しなくなった場合。6 節参照）、当該鍵を信頼して必要なセキュリティを提供できなくなった場合に発生する。鍵の危殆化の可能性を報告するのは、鍵が危殆化したと疑われる者（例えば、鍵の所有者や、当該鍵で保護されたデータが危殆化したことを観察したエンティティなど）の責任である。

鍵が危殆化した場合、情報に暗号保護を適用するための鍵の使用（デジタル署名の計算や情報の暗号化など）は全て停止**させなければならず**、かつ危殆化した鍵は失効**させなければならない**（8.3.5 節参照）。ただし、保護を解除又は検証する（復号又はデジタル署名の検証などを行うなど）ために、管理された状況下で当該鍵を継続使用することは、継続使用のリスクと組織の鍵管理ポリシー（SP 800-57、パート 2 参照）に応じて、正当化される場合がある。危殆化した鍵の継続使用は、既に保護されている情報の処理に限定**されなければならない**。この場合、情報を使用するエンティティは、包含される危険

性を十分に認識**しなければならない**。鍵の暗号利用期間を制限することで、鍵が危殆化した場合に危殆化する（暴露される）材料の量を制限できる。異なる目的（例えば、異なるアプリケーションや異なる暗号化メカニズムなど）に異なる鍵を使用したり、一つの鍵で保護される情報の量を制限したりすることも、この目的を達成する（5.3 節参照）。

5.5.1 影響

鍵の危殆化には以下の影響がある：

1. 鍵が不正に開示されることは、別のエンティティ（不正なエンティティ）が当該鍵を知り、その鍵を使用して、その鍵の使用を必要とする計算を実行できる可能性があることを意味する。

   一般に、機密保護を提供する[65]（すなわち、暗号化を介して）ために使用される鍵の不正開示は、当該鍵によって暗号化される全ての情報が、不正なエンティティによって決定され得ることを意味する。例えば、データ暗号化対称鍵が危殆化した場合、不正なエンティティは、当該鍵を利用して、過去から将来の暗号化された情報を復号する可能性がある（すなわち、情報は、認可されたエンティティ間の機密ではもはやない）。さらに、危殆化した鍵は、敵対者によって、敵対者が選択した情報を暗号化し、結果として虚偽の情報を提供するために使用される可能性がある。

   署名プライベート鍵が不正に開示されることは、当該鍵で署名された全てのデータの完全性と否認防止の特性が疑われることを意味する。プライベート鍵を所持している不正な当事者は、虚偽の情報に署名し、それが有効であるように見せかけることができる。署名されたデータが危殆化する前の時点から他のメカニズム（例えば、物理的セキュリティ）で保護されていたことが証明できる場合には、署名にはまだ何らかの価値があるかもしれない。例えば、署名されたメッセージが 1 日目に受信され、後に署名プライベート鍵が 15 日目に危殆化したことが判明した場合、受信者は、15 日目以前に受信者が所有していたため、そのメッセージが有効であるという確信を持つことができるかもしれない。暗号タイムスタンプもまた、署名プライベート鍵が危殆化する前に署名されたメッセージの保護を提供することもあることに留意されたい。しかし、このような場合、これらの他のメカニズムによって提供されるセキュリティが、署名のセキュリティを保証するために非常に重要である。さらに、署名されたメッセージの信憑性が疑われるかもしれない。なぜなら、署名プライベート鍵がメッセージの受信者や他の組織に開示され、それらのエンティティによりメッセージが何らかの方法で改変される可能性があるからである。

   CA の署名プライベート鍵が開示されることは、敵対者が不正な証明書や証明書失効リスト（CRL）を作成できることを意味する。

2. 鍵の完全性の危殆化とは、鍵が不正であることを意味する。つまり、鍵が（意図的又は偶発的に）変更されたか、別の鍵が置き換えられたかのいずれかである。これには、当該鍵の削除（利用不可）を含む。完全性[66]を提供するために使用される鍵の置換や変更は、当該鍵で保護されている全ての情報の完全性に疑問を投げかけるものである。

3. 鍵の使用法やアプリケーションの関連性の危殆化とは、当該鍵が誤った目的（例えば、デジタル署名ではなく鍵確立）や誤ったアプリケーションに使用され、当該鍵で保護されている情報が危殆化している可能性があることを意味する。

---

65 例えば、署名のプライベート鍵として使用される鍵の機密性とは対照的である。

66 例えば、暗号化として使用される鍵の完全性とは対照的である。

4. 所有者又はその他のエンティティとの鍵の関連性の危殆化とは、エンティティの身元が保証されない（すなわち、エンティティが実際に誰であるかわからない）ことを意味する。
5. 鍵と他の情報との関連付けの危殆化とは、関連付けが全くないか、又は間違った情報との関連付けであることを意味する。これにより、暗号サービスが失敗したり、情報が失われたり、情報のセキュリティが危殆化したりする可能性がある。

5.5.2 保護対策

鍵の危殆化の可能性や影響を最小限に抑えるために、一定の保護対策がとられる場合がある。通常は、以下の手順が含まれる：

1. 対称秘密鍵又は非対称プライベート鍵が平文形式である時間を制限する。
2. 平文の対称秘密鍵や非対称プライベート鍵を人間が閲覧できないようにする。
3. 平文の秘密鍵とプライベート鍵を物理的に保護された“コンテナ”に制限する。これには、鍵生成器、鍵配送デバイス、鍵ローダ、暗号モジュール、ハードウェアセキュリティモジュール（HSM）、鍵保管デバイスなどがある。
4. 完全性チェックを使用して、鍵の完全性や他のデータとの関連性が危殆化していないことを確認する。例えば、鍵は、ラッピングされた鍵や鍵のメタデータへの不正な変更が検出されるような方法で、ラップされて（すなわち、暗号化され、完全性が保護されて）いてもよい。
5. 鍵確認（SP 800-175B、SP 800-56A、及び SP 800-56B 参照）を実施して、適切な鍵が実際に確立されたことの確認を支援する。
6. 平文形式の対称秘密鍵及び非対称プライベート鍵への各々のアクセス履歴を維持する説明責任システムを確立する。
7. 鍵の暗号学的完全性チェックを提供する（例：MAC やデジタル署名を使用する）。
8. 署名されたデータに信頼されるタイムスタンプを使用する。
9. 鍵が不要になったらすぐに鍵を破棄する。
10. 危殆化回復計画を作成する。特に CA 鍵の危殆化の場合。

鍵の危殆化の最悪の形態は、検出されないことである。とはいえ、この場合でも、一定の保護対策を講じることは可能である。鍵管理システムは、鍵の危殆化による負の影響を緩和するように設計される**べきである**。システムは、一つの鍵の危殆化によってデータが危殆化するのをできる可能な限り少なくなるように設計**すべきである**（SP 800-152 参照）。例えば、一つの暗号鍵を使用して、一人の人間のエンティティ又は限られた数のエンティティのデータを保護することができる。多くの場合、システムは、鍵の所有のみに依存しない通信対象のエンティティを認証するための代替方法を持っている。これは、壊滅的な弱点を持つシステムの構築を避けることを目的としている。

危殆化復旧計画は、鍵が危殆化した場合の暗号セキュリティサービスを復旧させるために、不可欠である。危殆化復旧計画は文書化され、容易にアクセスできるように**しなければならない**。この計画は、鍵管理実践ステートメント（SP 800-57 パート 2 参照）に含めることができる。含まれていない場合は、鍵管理実践ステートメントが、危殆化復旧計画を参照**すべきである**。

危殆化からの回復は、主に局所的な行動であるが、システムや機器を使用するコミュニティ全体がその影響を受ける。したがって、危殆化復旧手順には、コミュニティ全体を含める**べきである**。例えば、

ルート CA の署名プライベート鍵の危殆化からの回復には、インフラストラクチャを使用する全てのエンティティが新しいトラストアンカー証明書を取得し、インストールする必要がある。一般的に、これは物理的な手続きが必要であり、実装に費用がかかる。このような高価な手続きを回避するために、危殆化を回避するための精巧な予防措置が正当化される場合がある。

危殆化復旧計画には、以下の内容を含める**べきである**：

a. 通知すべき要員の特定、及び通知に**含むべき**内容（例えば、危殆化の範囲（特定の鍵が危殆化したのか、証明書生成プロセスが危殆化したのか、など））
b. 復旧行為を実行する要員の特定
c. 新しい鍵を取得する方法（すなわち、鍵再設定）
d. 全ての暗号鍵の棚卸リスト（例えば、システム内の全ての鍵と証明書の位置）
e. 危殆化復旧手順に関する全ての適切な要員への教育
f. 危殆化復旧手順をサポートするために必要な全ての要員の特定
g. 鍵の失効チェックの実施を要求するポリシー（危殆化の影響を最小限に抑えるために）
h. 鍵再設定作業の監視（影響を受けた全ての鍵に対して、必要な作業が全て実施されていることを確実にするために）
i. その他の危殆化復旧手順

その他の危殆化復旧手順としては、以下のものが含まれる：

j. 機器の物理的な検査
k. インシデントの結果として危殆化した可能性のある全ての情報の特定
l. 署名鍵の危殆化により無効となる可能性のある全ての署名の特定
m. 必要に応じて、新しい鍵材料の配付

5.6 暗号アルゴリズム及び鍵長選択のためのガイダンス

3 節で特定されたセキュリティサービスを提供する暗号アルゴリズムは、FIPS 及び NIST 推奨で規定又は採用されている。これらのアルゴリズムの中には、いくつかの鍵長について定義されているものがある。本節では、適切なアルゴリズム及び鍵長を選択するためのガイダンスを提供することで、1）システムの期待される寿命、及び 2）データの期待される寿命の間にシステムによって保護される機微データに対して適切な保護を提供する。

5.6.1 同等のアルゴリズム強度

暗号アルゴリズムは、使用されるアルゴリズムと鍵長（アルゴリズムによって鍵が必要とされる場合）に依存して、異なるセキュリティ“強度”を提供することができる。セキュリティ強度とは、暗号アルゴリズムやシステムを破るために必要な作業量（すなわち、処理数）に関連付けられた数値である。5.6.1.1 節及び 5.6.1.2 節では、一般的に使用されているセキュリティ強度（80 ビット、112 ビット、128 ビット、192 ビット及び 256 ビット）について、**承認された**暗号アルゴリズムの推定最大セキュリティ強度を示す。80 ビットのセキュリティ強度は、もはや適切ではないと見なされることに留意されたい。

以下に示す同等のセキュリティ強度は、現在知られている方法を用いた本推奨の発行時点で受け入れられている評価値に基づく。素因数分解アルゴリズム、一般的な離散対数攻撃、楕円曲線離散対数攻撃及びその他のアルゴリズムの進歩も量子コンピューティングも、将来的にこれらの同等性に影響を与える可能性がある。新しい又は改良された攻撃や技術が開発され、現在のアルゴリズムの一部は完全に安全でない状態になる可能性がある。

重要な注意：大規模な量子コンピュータが利用可能になった場合、**承認された**公開鍵アルゴリズムのセキュリティが脅かされる。特に、デジタル署名スキーム、Diffie-Hellman や MQV を用いた鍵合意スキーム [67]、及び RSA を用いた鍵合意スキームや鍵配送スキームは、安全な量子耐性（又は“ポスト量子”）のあるものに置き換える必要があるかもしれない。今回の推奨の改訂版が発行された時点では、NIST は耐量子計算機暗号アルゴリズムを標準に選択するプロセスを行っている。このプロセスは複数年にわたるプロジェクトであり、これらの新しい標準が利用可能になったとき、本推奨は適切なガイダンスとともに更新される。耐量子計算機暗号プロジェクトに関する情報は、https://csrc.nist.gov/projects/post-quantumcryptography で入手可能である。

強力な暗号アルゴリズムを使用することは、暗号技術のセキュリティにとって非常に重要である。しかし、それらの実装及び使用法もまた、極めて重要な関心事項である。なぜなら、アルゴリズムが意図せずに鍵に関する少量の情報を漏らすような方法で実装される可能性があるためである。この場合、より長い鍵を使用することで、この漏えいした情報が最終的に当該鍵を危殆化させる可能性が低くなるかもしれない。

5.6.1.1 対称ブロック暗号と非対称鍵アルゴリズムのセキュリティ強度

承認された対称鍵ブロック暗号（AES など）及び公開鍵（つまり、非対称鍵）アルゴリズムは、暗号鍵の使用を必要とする。セキュリティ強度の評価は、鍵が指定された長さであり、特定のルールに従って生成・処理される（例えば、鍵は十分なエントロピーでシードされた RBG を使用して生成され、特定の基準を満たす）という前提の下で行われる。しかし、これらのルールは守られないことも多く、それらの鍵によって保護されるデータに提供されるセキュリティは、提供されるセキュリティ強度の評価値よりも幾分か低いかもしれない（5.6.2 節参照）。

与えられた鍵長（X と Y）に対して 2 つのアルゴリズムが同等の強度を持つと見なされるのは、“アルゴリズムを破る”又は（与えられた鍵長と十分なエントロピーを持つ）鍵を決定するのに必要な作業量が、与えられたリソースを使用してほぼ同じである場合である。与えられた鍵長に対するアルゴリズムのセキュリティ強度は、伝統的に、鍵長が“X”であり、ショートカット攻撃を持たない（すなわち、最も効率的な攻撃は可能な全ての鍵を試すこと）対称鍵アルゴリズムに対して全ての鍵を試すのにかかる作業量を単位として記述される。この場合、最も効率的な攻撃は全数探索攻撃と言われる。Y ビットの鍵を持ち、X ビットの鍵を持つ対称鍵アルゴリズムと同等の推定最大セキュリティ強度を持つアルゴリズムは、“X ビットの推定最大セキュリティ強度”を持つ、又は“X ビットセキュリティ”を提供できると言われる。数個の平文ブロックと対応する暗号文が与えられた場合、X ビットセキュリティを提供できるアルゴリズムは、平均して攻撃に $2^{X-1}T$ 単位の時間を要する。T は、平文値の暗号化を 1 回行ってその結果を対応する暗号文値と比較するのに要する時間である。

アルゴリズムのセキュリティ強度を決定することは容易ではない。セキュリティ強度は、多くの要因に依存する。例えば、攻撃者の能力、鍵長、同じ鍵を使って処理されるデータ量、鍵間の関連性の高さ

[67] 有限体と楕円曲線の両方のバージョン

などに依存する。攻撃者の能力には、アルゴリズムを攻撃する時間を短縮するために使用される暗号解読技術、攻撃者が利用できる処理能力、新しいタイプのコンピューティングシステム（例えば、量子コンピュータ）の出現などが含まれる。

表 2 は、**承認された**対称ブロック暗号と非対称鍵アルゴリズムと鍵長について、評価された同等の最大セキュリティ強度を示す。

1. 1 列目は、特定の行に記載されているアルゴリズムと鍵長によって提供される推定最大セキュリティ強度（ビット単位）を示す。アルゴリズムに対する計算上の利点を有する攻撃のため、セキュリティ強度は必ずしも鍵長と同じとは限らないことに留意されたい。
2. 2 列目は、1 列目で示したセキュリティ強度を提供できる対称鍵アルゴリズムを特定する。2TDEA と 3TDEA は SP 800-67 で規定され、AES は FIPS 197 で規定される。2TDEA は 2 つの異なる鍵を持つ TDEA であり、3TDEA は 3 つの異なる鍵を持つ TDEA である。これらのブロック暗号を暗号プリミティブとして使用する暗号利用モードと RBG が**承認されている**ことに留意されたい（SP 800-38 シリーズ及び SP 800-90A を参照）；これらのアルゴリズムが提供するセキュリティ強度は、そのプリミティブが提供するセキュリティ強度と同じである。
3. 3 列目は、有限体上の暗号（FFC）を使用する標準規格に関連するパラメータの最小サイズを示す。このようなアルゴリズムの例としては、デジタル署名のための FIPS 186 で定義されている DSA、SP 800-56A で定義されている Diffie-Hellman (DH)や MQV 鍵合意などがある。L は公開鍵のサイズ、N はプライベート鍵のサイズである。
4. 4 列目は、整数素因数分解型暗号（IFC）に基づくアルゴリズムの k（モジュラス n のサイズ）の値を示す。このタイプの主流のアルゴリズムは、RSA アルゴリズムである。RSA は、デジタル署名については FIPS 186 で、鍵確立については SP 800-56B で**承認されている**。k の値は、一般的に鍵長と見なされる。
5. 5 列目は、FIPS 186 でデジタル署名用に規定され、SP 800-56A で鍵確立用に規定されている楕円曲線暗号（ECC）に基づくアルゴリズムの f（n のサイズ、n はベースポイント G の位数）の範囲を示す。f の値は、一般的に鍵長と見なされる。

表 2：対称ブロック暗号と非対称鍵アルゴリズムの同等のセキュリティ強度

セキュリティ強度	対称鍵アルゴリズム	FFC (DSA, DH, MQV)	IFC* (RSA)	ECC* (ECDSA, EdDSA, DH, MQV)
≤ 80	2TDEA	L = 1024 N = 160	k = 1024	f = 160-223
112	3TDEA[68]	L = 2048 N = 224	k = 2048	f = 224-255
128	AES-128	L = 3072	k = 3072	f = 256-383

[68] 3TDEA は 112 ビットのセキュリティ強度を持つと記載されているが、その使用は 2023 年まで非推奨であり、それ以降は暗号保護に適用することは認められない（SP 800-131A 参照）。非推奨のアルゴリズムを使用することは、そのアルゴリズムや鍵長を使用することのリスクを許容できる場合に使用してもよいということである。

		N= 256		
192	AES-192	L= 7680 N= 384	k= 7680	f= 384-511
256	AES-256	L= 15360 N= 512	k= 15360	f= 512+

* 量子コンピューティングが実用化された場合、セキュリティ強度の評価値は大きく影響を受ける。

FFC 及び IFC のアルゴリズムについては、記載されている鍵長は、引用文書（FIPS 186、SP 800-56A、SP 800-56B など）で**承認されている**鍵長と必ずしも一致しないことに注意されたい。つまり、いくつかの鍵長が記載されていない場合や、追加の鍵長が関連するセキュリティ強度とともに提供される場合がある。ただし、FFC、IFC、及び ECC のアルゴリズムの推定セキュリティ強度は、IG 7.5[69]の式を使用して計算することもできる。

また、最大セキュリティ強度が 112 ビット未満（すなわち、上のオレンジ色で示されている 80 以下のところ）と評価されているアルゴリズム／鍵長の組み合わせは、連邦政府の情報への暗号保護の適用（データの暗号化やデジタル署名の生成など）にはもはや承認されていないことにも留意されたい。しかし、ある程度の柔軟性が認められており、受信エンティティがそうすることに伴うリスクを受け入れる場合、既にこれらのセキュリティ強度で保護されている情報の処理（暗号化されたデータの復号やデジタル署名の検証など）が認められる。詳細については、SP 800-131A を参照のこと。

5.6.1.2 ハッシュ関数とハッシュベース関数のセキュリティ強度

暗号学的ハッシュ関数は、任意の（限度はあるが）長さのビット列を固定長のビット列に写像する。**承認された**ハッシュ関数は以下の特性を満たす：

1. （現像困難性）あらかじめ指定されたいかなる出力に対してもその値に写像する入力を見つけることが計算量的に実行不可能である
2. （衝突困難性）同一の出力に写像される 2 つの別個の入力を見つけることが計算量的に実行不可能である

ハッシュ関数のセキュリティ強度は、それが使用されるアプリケーションが要求する特性によって決まる。SHA-1 及び SHA-2 ハッシュ関数ファミリーについての説明は SP 800-107 を、SHA-3 ハッシュ関数についての説明は FIPS 202 を参照のこと。

採用される可能性のある適切なハッシュ関数は、ハッシュ関数が使用されるアルゴリズム、スキーム又はアプリケーションによって、及び提供されるべき最小のセキュリティ強度によって決定される。これらのアルゴリズムにおける推定セキュリティ強度は、ハッシュ関数名の下 3 桁で示されるハッシュ関数の出力ブロックの長さ（例えば、SHA-224 は 224 ビットの出力ブロック長を持つ）に依存する。

表 3 は、FIPS 186 及び FIPS 202 で規定されている**承認された**ハッシュ関数の一覧であり、様々なハッシュ関数アプリケーション（デジタル署名、HMAC、KMAC、鍵導出、乱数ビット生成など）において指定された各々のセキュリティ強度を提供するために使用できる最小の出力ブロック長を示している。より大きな出力ブロック長を有し、より高いセキュリティ強度を提供するハッシュ関数を使用し

[69] IG 7.5: Implementation Guidance for FIPS 140-2 and the Cryptographic Module Validation Program (CMVP), Section 7.5, Strength of Key Establishment Methods.

てもよい（例えば、256 ビットセキュリティ強度をサポートすることができる SHA-512 を使用して、128 ビットセキュリティ強度を提供することができる）。

鍵を必要とする HMAC や KMAC の場合、推定セキュリティ強度は、鍵の長さと鍵を生成するために使用されるエントロピーが、少なくともセキュリティ強度に等しいことを前提としていることに留意されたい。

表 3：ハッシュ関数とハッシュベース関数の最大セキュリティ強度

セキュリティ強度	デジタル署名及び衝突困難性を必要とするその他のアプリケーション	HMAC[70]、KMAC[71]、鍵導出関数[72]、乱数ビット生成[73]
≤ 80	SHA-1[74]	
112	SHA-224, SHA-512/224, SHA3-224	
128	SHA-256, SHA-512/256, SHA3-256	SHA-1, KMAC128
192	SHA-384, SHA3-384	SHA-224, SHA-512/224, SHA3-224
≥ 256	SHA-512, SHA3-512	SHA-256, SHA-512/256, SHA-384, SHA-512, SHA3-256, SHA3-384, SHA3-512, KMAC256

ハッシュベースのアプリケーションでは、暗号鍵はアプリケーションに関連付けられており、アプリケーションが実際に与えるセキュリティ強度を決定する際に考慮する必要がある。例えば、デジタル署名の生成については、与えられたセキュリティ強度に対する鍵の最小鍵長は、5.6.1.1 節の表 2 の FFC、IFC 及び ECC の列に記載されている一方、HMAC については SP 800-107 で鍵長が説明されている。

セキュリティ強度が 112 ビット未満（すなわち、上のオレンジ色で示す 80 以下）のハッシュ関数及びアプリケーションは、連邦政府の情報に暗号保護を適用すること（例えば、デジタル署名を生成すること）がもはや承認されていないことに留意されたい。しかし、ある程度の柔軟性が認められており、受信エンティティがそうすることに伴うリスクを受け入れる場合、既にこれらのセキュリティ強度で保護されている情報の処理（例えば、デジタル署名の検証）が認められる。詳細については、SP 800-131A を参照のこと。

---

70 衝突困難性ではなく、原像困難性が要求されている前提とする。

71 KMAC は、技術的には承認されたハッシュ関数に基づくものではないが、FIPS 202 で規定された関数に基づいている。

72 鍵導出のセキュリティ強度は、使用する共有秘密や鍵が、必要なセキュリティ強度を裏付けるのに十分なエントロピーを有していることを前提とする。

73 乱数ビット生成のセキュリティ強度は、乱数ビット生成器に、必要なセキュリティ強度を裏付けるのに十分なエントロピーが与えられていることを前提とする。

74 SHA-1 は、衝突困難性を必要とするデジタル署名に対して、80 ビット以下のセキュリティしか提供できないことが実証されている。本推奨の発行時点では、デジタル署名の衝突に対するセキュリティ強度は、まだ推測の域を出ていない。

5.6.2 アルゴリズムスイートの使用と有効なセキュリティ強度

多くのアプリケーションでは、複数の暗号サービス（鍵確立、機密性保護、完全性保護、ソース認証など）を必要とする。各々のサービスを提供するために、異なるアルゴリズムと鍵を使用することができる（例えば、データ暗号化に AES を使用し、完全性保護のためのデジタル署名を生成するために RSA を使用することができる）場合もあれば、同じアルゴリズムが同じ鍵又は異なる鍵を使用し複数のサービスを提供することができる（例えば、デジタル署名を生成する RSA を使用して、ソース認証と完全性保護を実行することができる）場合もある。また、多くのサービスが、複数のアルゴリズムによって提供されることが可能となっている（例えば、鍵確立は、RSA 又は Diffie-Hellman（DH）アルゴリズムのいずれかによって提供することができる）。

複数のアルゴリズムが同じサービスを実行するために利用可能である場合において、いくつかのアルゴリズムは、設計上の理由で本質的により効率的である（例えば、HMAC とデジタル署名のどちらの利用も完全性保護を提供することができるが、HMAC の方がより効率的に設計されている）。

多くの場合、様々な鍵長がアルゴリズムで利用可能である。アルゴリズムの中には（例えば、RSA などの公開鍵アルゴリズムでは）、必要以上に大きな鍵長を使用すると、処理に影響が出る場合がある（例えば、より大きな鍵の生成はより時間がかかり、より多くのメモリと送信帯域幅を必要とし、データの処理に時間がかかる）。しかし、小さすぎる鍵長の使用は十分なセキュリティを提供しない可能性がある。

ブロック暗号アルゴリズム（例えば、AES）を選択する際には、ブロック長も考慮すべき要素である可能性がある。なぜなら、SP 800-38 シリーズで定義されているいくつかのモードで提供されるセキュリティ量がブロック長に依存するためである。この話題に関する詳細は、SP 800-38 シリーズに記載されている。

保護すべきデータに十分な保護が提供できるのであれば、異なる強度と鍵長のアルゴリズムを、性能、可用性、又は相互運用性の理由から、一緒に使用することができる。一般に、暗号保護の強度は、保護を提供するために使用される最も弱いアルゴリズムと鍵長によって決定される。データに提供される保護の実際の強度の決定には、情報に暗号保護を適用するために使用されるアルゴリズム及び鍵長の分析だけでなく、当該鍵及びその元となった情報がどのように生成されたか（例えば、鍵生成中に使用される RBG がサポートするセキュリティ強度）及び生成後に当該鍵がどのように取り扱われるかの詳細の分析が含まれる。

鍵の取扱いには、鍵を操作した全てのプロセス（例えば、鍵が何らかの暗号処理の入力として使用される場合）が含まれる。s ビットセキュリティ強度を提供するように生成された鍵が、s よりも小さいセキュリティ強度を持つプロセスによって意図されたアルゴリズムを処理するときに使われた場合、当該鍵によって提供できるセキュリティ強度は、そのプロセスのセキュリティ強度まで低下する。例えば、鍵が 256 ビットセキュリティ強度を持つ RBG によって生成された場合、その鍵が AES-256 で使用されるとき、鍵とアルゴリズムの組み合わせは、256 ビットセキュリティ強度を提供することができる。しかし、当該鍵を AES-128（最大 128 ビットセキュリティ強度しか提供できない）でラッピングした場合、AES-256 で使用した場合でも、256 ビット鍵で提供できるセキュリティ強度は 128 ビットに低下する。

以下に、いくつかのアルゴリズムの組み合わせの一覧と、アルゴリズム／鍵長の組み合わせのセキュリティ上の意味合いについての話題を示す：

1. 鍵確立スキームを使用して、1 つ以上の対称鍵アルゴリズム（AES、HMAC など）で使用するための鍵材料を確立する場合、鍵材料がサポートできるセキュリティ強度は、使用される最も弱いアルゴリズムと鍵長によって決定される。例えば、128 ビットの AES 鍵を確立するために SP 800-56A で規定される 224 ビット ECC 鍵が使用される場合、その AES 鍵で保護され

た情報に対して 112 ビットを超えるセキュリティを提供することはできない。なぜなら、224 ビット ECC 鍵は最大 112 ビットセキュリティ強度しか提供できないためである（5.6.1.1 節表 2 参照）。

2. ハッシュ関数とデジタル署名アルゴリズムを組み合わせてデジタル署名を計算する場合、署名のセキュリティ強度は、2 つのアルゴリズムのうち、より弱い方のアルゴリズムによって決定される。例えば、SHA-256 を 2048 ビット鍵の RSA と組み合わせて使用する場合、112 ビットを超えるセキュリティを提供することはできない。なぜなら、デジタル署名生成に使ったのが 128 ビットセキュリティ強度をサポートできる SHA-256（表 2 及び表 3 参照）であったとしても、2048 ビット RSA 鍵では 112 ビットを超えるセキュリティ強度を提供できないためである。
3. 乱数ビット発生器（RBG）を使って、*X* ビットセキュリティの提供を意図した暗号アルゴリズムの鍵生成をする場合、少なくとも *X* ビットセキュリティをサポートする**承認された**乱数ビット発生器を使用**しなければならない**。例えば、AES-128 とその鍵が 128 ビットセキュリティ強度を提供することを意図している場合、RBG は少なくとも 128 ビットセキュリティをサポートする必要がある。

所定のセキュリティ強度をサポートするためには、アルゴリズムと鍵長の組み合わせを慎重に選択しなければならない。例えば、通信されるデータを保護するために 128 ビットセキュリティ強度が必要であり、機密性保護、完全性保護及びソース認証を提供する場合、以下のアルゴリズムと鍵長の選択が適切である：

a. 鍵を生成するために少なくとも 128 ビットセキュリティ強度をサポートする RBG を選択する。
b. 機密性：AES-128 と a.項の RBG によって生成された鍵を使って情報を暗号化する。
c. 完全性保護及びソース認証：1 つだけの暗号処理が望ましい場合は、デジタル署名を使用する。SHA-256 又はそれよりも大きなハッシュ関数を、署名生成前のデータハッシュに使用することができる。デジタル署名アルゴリズムは、アプリケーションで利用可能なものから選択する（例えば、少なくとも 256 ビット鍵を持つ ECDSA）。複数のアルゴリズムと鍵長が利用可能な場合、最低限の要件を満たしている限り、アルゴリズムの性能、メモリ要件などに基づいて選択してもよい。
d. 鍵確立：鍵確立スキームを、アプリケーション及び環境（SP 800-56A 又は SP 800-56B を参照）、実装におけるアルゴリズムの可用性及びその性能に基づいて選択する。少なくとも 128 ビットセキュリティを提供できるアルゴリズムと鍵長について、表 2 から鍵長を選択する。例えば、ECC 鍵合意スキームが利用可能な場合は、ECC スキームと少なくとも 256 ビット鍵（表 2 の *f* の値）の曲線を使用する。ただし、鍵合意に使用される鍵は、デジタル署名に使用される ECDSA 鍵とは異なるもので**なければならない**（上記 c.項参照）。

システムを調達する機関は、システムの潜在的な運用寿命を考慮**すべきである**。機関は、システム全体のライフタイム中は安全であると予想されるアルゴリズムと鍵長を選択**しなければならない**か、又はアルゴリズムと鍵長が容易に更新できることを保証**すべきである**。

5.6.3 計画されるセキュリティ強度の時間枠及び現在の承認状況

時間の経過とともに、暗号アルゴリズムとそれに関連する鍵長はより脆弱となり攻撃が成功する可能性があり、より強力なアルゴリズムやより長い鍵長への移行が必要となる場合がある。表 4 は、最低限のセキュリティ強度（例えば、2031 年に少なくとも 128 ビット）で暗号保護を適用するための計画される時間枠を示す。表 4 では：

1. 1 列目は、2 つのサブ列に分割されている。第 1 のサブ列は提供されるセキュリティ強度を示し、第 2 のサブ列はデータに暗号保護が適用されるか（例えば、暗号化されているか）又は暗号保護されたデータが処理されるか（例えば、復号されているか）を示す。
2. 2 列目及び 3 列目は、セキュリティ強度が受け入れ可能か、レガシーユースでは OK であるか、又は不許可であるかの時間枠を示す。
   - “受け入れ可能”とは、アルゴリズム又は鍵長が現在安全であると見なされていることを示す。
   - “レガシーユース”とは、アルゴリズム又は鍵長が、レガシーのアプリケーションでの使用のために、使われる可能性があることを意味する（すなわち、アルゴリズム又は鍵長が暗号保護されたデータを処理するために使用される可能性がある）。
   - “不許可”とは、アルゴリズム又は鍵長が、暗号保護の適用（例えば、暗号化）に使用されてはならないことを意味する。

表 4：セキュリティ強度の時間枠

<table>
<tr><th colspan="2">セキュリティ強度</th><th>2030 年まで</th><th>2031 年以降</th></tr>
<tr><td rowspan="2">< 112</td><td>保護の適用</td><td colspan="2">不許可</td></tr>
<tr><td>処理</td><td colspan="2">レガシーユース</td></tr>
<tr><td rowspan="2">112</td><td>保護の適用</td><td rowspan="2">受け入れ可能</td><td>不許可</td></tr>
<tr><td>処理</td><td>レガシーユース</td></tr>
<tr><td>128</td><td rowspan="3">保護の適用、及び既に保護されている情報の処理</td><td>受け入れ可能</td><td>受け入れ可能</td></tr>
<tr><td>192</td><td>受け入れ可能</td><td>受け入れ可能</td></tr>
<tr><td>256</td><td>受け入れ可能</td><td>受け入れ可能</td></tr>
</table>

SP 800-131A は、NIST **承認**の暗号アルゴリズムとその鍵長の承認状況を提供する。この状況は、上記で定義された**受け入れ可能**、**不許可**、**レガシーユース**という用語に加えて、**非推奨**（アルゴリズム及び鍵長を使用することはできるが、そうすることに伴うリスクを受け入れなければならないことを意味する）という追加されたカテゴリで示される。

特定のデータを保護しなければならない期間中に必要とされる最低限のセキュリティ強度と使用すべきアルゴリズムは、以下のようにして得**なければならない**。

- 保護期間全体でデータを保護するために必要なセキュリティ強度を決定する。

- 表 1 及び表 2 を使用して、必要とされる強度と同じかそれ以上の強度のアルゴリズムと鍵長を選択する。セキュリティ強度は、使用するアルゴリズムによってだけでなく、鍵長及び当該鍵をどのように生成し取り扱うかにも依存して決まる（5.6.2 節参照）。
- あるアルゴリズムに対して鍵が RBG によって生成される場合、当該鍵はそのアルゴリズムが要求するセキュリティ強度をサポートする RBG（すなわち、RBG にシードを与えた時にそのようなエントロピーを提供するように設計された RBG）を使用して生成**されなければならない。**

5.6.4 システムにおける新しいアルゴリズム及び鍵長への移行

5.6.1.1 節及び 5.6.1.2 項の表 2 及び表 3 は、それぞれ、所定の鍵長で承認された暗号アルゴリズムがサポートすることができるセキュリティ強度の現在の評価値を示している。5.6.3 節の表 4 は、より高いセキュリティ強度への移行スケジュールを提供し、SP 800-131A は、米国連邦政府の現在のアルゴリズム承認状況と移行計画を提供する。これらを合わせて、必要に応じて将来の暗号保護戦略を計画するために使用することができる（少なくとも、量子コンピュータや耐量子アルゴリズムが利用可能になるまでは）。**最も重要なアプローチは、柔軟性を持たせることである。最も簡単に暗号セキュリティ製品に適応できる実装やアプリケーションを使用し、それらへの移行計画を立てることが最良の解決策である。**本節では、移行のための論点のいくつかを述べる。

特定の暗号アルゴリズム（及び鍵長）によって保護されたデータが安全であり続けると評価された期間は、アルゴリズムのセキュリティ寿命と呼ばれる。この期間中、アルゴリズムは暗号保護の適用（例えば、データの暗号化）と保護された情報の処理（例えば、データの復号）の両方に使用される可能性があるが、保護を適用するために許容される期間（作成者使用期間）は、アルゴリズムのセキュリティ寿命よりも短くなる可能性がある（図 2 参照）。アルゴリズムは、アルゴリズムのセキュリティ寿命の間、保護されたデータに対して適切な保護を提供することが期待される。

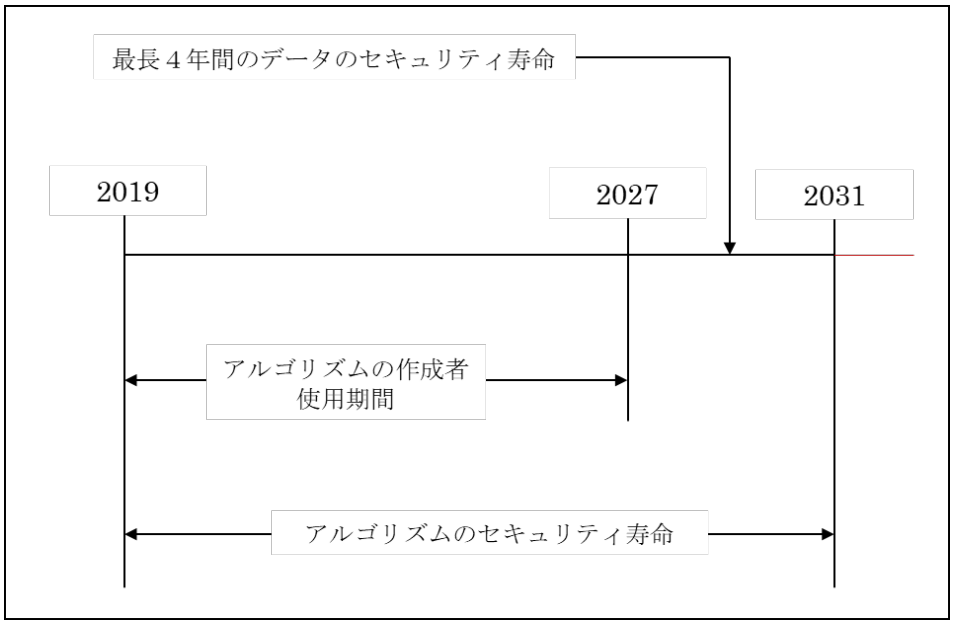

図 2：アルゴリズムの作成者使用期間の例

通常、組織は特定のアプリケーションに必要な暗号サービスを選択する。次に、アルゴリズムのセキュリティ寿命と保護すべきデータのセキュリティ寿命に基づいて、要件を満たすのに十分なアルゴリズ

ムと鍵長のスイートが選択される。次に、組織は、アプリケーションによって要求されるサービスを提供する認証済み暗号製品を含む鍵管理システムを確立する。アルゴリズムや鍵長のスイートがそのセキュリティ寿命の終わりに近づいてきたら、新しいアルゴリズム及び鍵長のスイートへの移行を計画する**べきである**。

アルゴリズムや鍵長が、もはや情報に対して望ましい保護を提供しないと判断された（例えば、アルゴリズムが“破られた”可能性がある）場合、そのアルゴリズムや鍵長によって保護されている情報は疑わしいと見なされる（例えば、そのデータがもはや機密ではないかもしれない、又は完全性が保証できない）。保護されたデータが保持されている場合は、そのセキュリティ寿命の残りの期間、情報を保護するために、**承認された**アルゴリズムと鍵長を使用して再保護される**べきである**。しかし、認可されていない当事者によって送信されたり、その他の方法でアクセスされたりした暗号化された情報は、後のある時点で復号するために収集され、保持されている可能性があると想定**すべきである**。さらに、回復された平文は、新しいアルゴリズムの平文－暗号文一致攻撃を試みるために使用される可能性がある。

表 2、表 3、表 4 を使用して適切なアルゴリズムと鍵長を選択する際には、データに対する期待されるセキュリティ寿命を考慮に入れることが非常に重要である。前述したように、アルゴリズム（及び鍵長）は、データに暗号保護を適用するためと、保護されたデータを処理するための両方に使用される可能性がある。データのセキュリティ寿命を考慮に入れた場合、データのセキュリティ寿命がアルゴリズムのセキュリティ寿命の終わりを超えているならば（すなわち、アルゴリズム又は鍵長が不許可となる時間枠内に延びているならば；表 4 参照）、与えられたアルゴリズム（及び鍵長）を使用してデータに暗号保護を適用**すべきではない**。

例えば、図 2 を使用して、暗号アルゴリズムのセキュリティ寿命が 2030 年 12 月 31 日に終了する場合、セキュリティ寿命が 4 年のデータは、2026 年 12 月 31 日以降に当該アルゴリズムを使用して暗号化されるべきではない。その代わりに、データのセキュリティ寿命をカバーする別のアルゴリズムを使用すべきである。データのセキュリティ寿命が当初の期待よりも長くなった場合、2030 年以降に提供される保護は必要とされるものよりも低いかもしれず、（2030 年以降に）データの機密性が危殆化するかもしれないリスクがある。危殆化の可能性に関連するリスクを受け入れることは、表 4 の“レガシーユース”の表示で示される。

データの暗号保護を開始する際には、保護を提供するのに適切な最強のアルゴリズムと鍵長を使用して、コストのかかる移行を最小限に抑える**べきである**。しかし、不必要に大きいアルゴリズムや鍵長を選択すると、パフォーマンスに悪影響を及ぼす可能性がある（例えば、アルゴリズムが許容できないほど遅くなる可能性がある）ことに留意すべきである。

新しいアルゴリズムや鍵長への移行プロセスは、現在のシステムが提供するセキュリティスイートの中からより安全性の高いオプションを選択するような単純なものから、全く新しいシステムを構築するような複雑なものまである。移行の影響は、移行の規模にも依存する。例えば、単一のストレージシステムで新しいアルゴリズムや鍵長に移行する際には一通りの考慮すべき点があるが、TLS や SSH の全ての実装でアルゴリズムや鍵長を変更する際にはより広い関連事項がある。

システムのために新しいアルゴリズムスイートを開発する必要がある場合、以下の論点を考慮すべきである：

1. **情報の機微性とシステムの寿命**：新しいアルゴリズムの寿命の間に、そのシステムが保護する必要のある情報の機微性を評価し、システムの最小のセキュリティ要件を決定**すべきである**。システムの必要な寿命、及びシステムが保護する必要のある情報の機微性を過小評価しないように注意**すべきである**。一時的又は暫定的な初期決定と考えられていたデータの機微性に関する多くの決定は、その後、不適切であることが証明された（例えば、データの機微性が初期に期待されていた寿命をはるかに超えて持続した）。

2. **アルゴリズムの選択**：新しいアルゴリズムは、システムのセキュリティ要件を満たすか、又はその要件を超えることを確実にするために慎重に選択されるべきである。一般的に、高いセキュリティを提供する暗号アルゴリズムや鍵長を選択することは比較的容易である。しかし、そのような決定を行う際、素人は暗号専門家に相談するのが賢明である。システムは、将来の成長に備えたアルゴリズムスイートのオプションを提供**すべきである**。

3. **システム設計**：新しいシステムは、最低限の性能とセキュリティ要件を満たすように設計され、暗号の更新にも柔軟に対応できるように**すべきである**。性能目標とセキュリティ目標が衝突する可能性があるため、これはしばしば困難な作業である。セキュリティの全ての側面（例えば、物理的セキュリティ、コンピュータセキュリティ、運用上のセキュリティ、要員セキュリティなど）が関与する。現在のシステムを変更して新しいアルゴリズムを組み込む場合、その結果を分析する必要がある。例えば、既存のシステムは、新しいアルゴリズムの“フットプリント”（例えば、鍵長、ブロック長など）に対応するために、大幅な変更が必要になるかもしれない。加えて、現在のシステムに保持されている（暗号アルゴリズム以外の）セキュリティ対策が、新システムでも有効であることを確認するためにレビューを行う**べきである**。

4. **導入前の評価**：強力な暗号は、実装が不十分な場合がある。したがって、新しい暗号技術への変更は、その技術がシステム内でどの程度有効で安全であるかを検討するための評価を行わずにする**べきではない**。

5. **テスト**：どのようなシステムであっても、導入前にテストを行う**べきである**。

6. **トレーニング**：新しいシステムで新しいタスクや異なるタスク（例えば、鍵管理手順）を実行する必要がある場合、それらのタスクを実行する個人に対して適切に訓練**すべきである**。改善を目的とした機能が、便利な機能や必要な機能ではなく、単に迷惑な機能とみなされるかもしれない。

7. **システムの実装と移行**：システムをできるだけ設計に近い形で実装するように注意を払う**べきである**。例外には注意を払う**べきである**。

8. **移行**：旧システムから新システムへの移行が可能な限りスムーズに行われるように、移行計画を策定し、それに従う**べきである**。

9. **導入後の評価**：システムを評価して、実装されたシステムがシステムのセキュリティ要件を満たしていることを検証**すべきである**。

5.6.5 セキュリティ強度の経年変化

ある時期には、アルゴリズムや鍵によって提供されるセキュリティ強度が低下したり、完全に失われたりすることがある。例えば、計算能力や暗号解読の向上により、使用されているアルゴリズムや鍵長が適切なセキュリティを提供しなくなる場合がある。この場合、“新しい”情報に対しては、より強力なアルゴリズム又は鍵を使って保護を適用することができる。しかし、その時期においては不十分なアルゴリズムや鍵を使用して以前に保護されていた情報は、もはや安全ではないかもしれない。この情報には、他の鍵や、当該鍵によって保護された機微情報が含まれる可能性がある。アルゴリズムや鍵によって提供されるセキュリティ強度の低下は、以下のような意味合いを持つ：

- 暗号化情報：暗号化された形の情報であっても、認可されていないエンティティが何れかのタイミングでアクセスできた暗号化情報の安全性は疑わしいと考える**べきである**。例えば、暗号化された形（例えば、鍵ラッピング鍵や鍵配送鍵を利用して、後に破られたアルゴリズムや鍵長で暗号化したもの）で送信された鍵は危殆化したとみなす必要があるかもしれない。なぜなら、敵対者が暗号化された形の鍵を保存しておくことで、いつかアルゴリズムを破る方法が見つかった場

合、後で復号できる可能性があるためである（鍵の危殆化についての話題は 5.5 節参照）。送信された暗号化情報が、その後、別の鍵やアルゴリズムを用いて保存のために再暗号化されたとしても、送信アルゴリズムや鍵の脆弱性のために、その情報はすでに危殆化している可能性がある。

このような“開示”がされなかった（例えば、送信されなかった）暗号化情報は、暗号アルゴリズムや鍵長が、最初に必要とされた保護量を提供しなくなったとしても、安全である可能性がある。例えば、暗号化された鍵の形式とそれらの鍵によって保護された情報が一度も送信されなかった場合、その情報は機密情報であり続ける可能性がある。

学ばなければならない教訓は、暗号化された形で（例えば、送信を介して）認可されていないエンティティがアクセスできる情報に使用される暗号メカニズムは高いレベルのセキュリティ保護を提供**すべきであり**、危殆化した鍵が非常に多くの情報を開示するのに使用することができないように、各鍵の使用は制限される**べきである**（すなわち、暗号利用期間は短くすべきである）。仮にアルゴリズム自体が破られた場合[75]においても、各鍵が非常に限られた量の情報を暗号化するために使われたときには、敵対者は全ての情報を復号するためにより多くの作業を行うことを余儀なくされる。暗号利用期間についての論点は、5.3.6 節を参照のこと。

- **最初に送信された保存データに対するデジタル署名**[76]：デジタル署名は、送信前やその後の保存前のデータに対して計算されることがある。この場合、署名されたデータとデジタル署名の両方が保存される。署名のセキュリティ強度が後に低下した場合（例えば、アルゴリズムが破れたため、敵対者が鍵を特定したため、など)、強度が低下する前の時点から保存データとそれに関連するデジタル署名が改ざんから（例えば、より強力なアルゴリズム又は鍵を使用したデジタル署名を適用することによって）十分に保護されていれば、当該署名はまだ有効であってもよい。さらなる議論については、5.5 節の項目 1 を参照のこと。元の署名メカニズム又はその鍵の通常のセキュリティ寿命を超えてデジタル署名されたデータを保存できる、暗号学的タイムスタンプを採用した保存機能が開発されている。
- **最初に送信された保存データに対する対称認証コード**[77]：デジタル署名と同様に、対称認証コード（すなわち、MAC）は、送信前やその後の保存前のデータに対して計算されることがある。受信したデータ及び認証コードが受信したまま保存され、その後認証アルゴリズム又は鍵のセキュリティ強度が低下した場合（例えば、アルゴリズムが破れたため）において、強度が低下する前の時点から保存データ及びそれに関連する認証コードが（例えば、より強力なアルゴリズム又は鍵を使用して別の認証コードを適用することによって）改ざんから十分に保護されていれば、当該認証コードはまだ有効であってもよい。さらなる議論については、5.5 節の項目 1 を参照のこと。元の認証メカニズム又はその鍵の通常のセキュリティ寿命を超えて認証されたデータを保存できる、暗号学的タイムスタンプを採用した保存機能が開発されている。

---

75 例えば、鍵の全数探索を行うよりも、鍵を復元する方が簡単である。

76 送信されたが保存されないデータに対するデジタル署名というのは、その価値が短期的であると考えられるため、対象としない（例えば、送信中にのみ発生するエラーを検出するために使用されることを目的としたデジタル署名）。

77 送信されたが保存されないデータに対する対称認証コードというのは、その価値が短期的であると考えられるため、対象としない。

6 鍵情報の保護要件

本節では、鍵情報に必要な保護の種類についてのガイダンスを提供する。鍵情報とは、鍵材料及び関連するメタデータと定義される。具体的な鍵情報は、鍵のタイプによって異なる。鍵情報は、セキュリティサービスが“意味のあるもの”となるために、保護されなければならない。FIPS 140 認証暗号モジュールは、そのセキュリティレベルに応じて、必要とされる保護の大部分を提供することができる。しかし、鍵情報が FIPS 140 暗号モジュールの外部に存在する場合はいつでも、追加の保護が必要である（例えば、オペレーティングシステム内のアクセス制御メカニズムや、外部データベース内の鍵情報の暗号化及び完全性保護など）。必要な保護のタイプは、鍵のタイプと、当該鍵が使用されるセキュリティサービスによって異なる。SP 800-152[78]は、連邦暗号鍵管理システム（FCKMS）に対して、FIPS 140 認証暗号モジュールの外部にある場合の鍵情報の保護、及び対処すべきその他の鍵管理要素についてのガイダンスを提供する。

6.1 保護及び保証の要件

鍵材料は、関連する暗号サービスが必要とされる限り、（運用上）利用可能であるべきである。鍵は、通常使用されている間は、暗号モジュール内に保持してもよいし、（適切な保護が施されている前提で）外部に保存され必要に応じて呼び出すこともできる。鍵によっては、鍵の作成者使用期間を超えて必要とされる場合には、アーカイブする必要があるかもしれない（作成者使用期間についての話題は 5.3.5 節参照）。

以下の保護及び保証が、鍵情報には必要となる場合がある。

機密性保護 は、全ての秘密鍵情報（すなわち、秘密にしておくことを意図した鍵情報。対称（秘密）鍵、非対称プライベート鍵、鍵シェア、秘密メタデータなど）に提供**されなければならない**。公開鍵、ドメインパラメータ、及びメタデータの多くは、一般的に機密保護を必要としない。秘密鍵情報が認証暗号モジュール内に存在する場合、その暗号モジュールが FIPS 140 に準拠し、鍵で保護されるデータに関連する FIPS 199 の影響レベルと一致するセキュリティレベルであれば、適切な機密性保護が提供される（SP 800-152 参照）。秘密鍵情報又はプライベート鍵情報が暗号モジュールの外部で利用可能な場合、その機密性保護は、適切なセキュリティ強度での暗号化（SP 800-152 参照）又は物理的手段を介した秘密鍵情報へのアクセス制御[79]によって、提供**されなければならない**。特定の機密性メカニズムのセキュリティと運用への影響は場合により異なる。適切な機密性メカニズムの選択のためのガイダンスは、6.2.1.3 節及び 6.2.2.3 節に記載されている。

完全性保護 は、全ての鍵情報に対して提供**されなければならない**。完全性保護は、常に、受信又は取得した鍵情報のソース及び形式を確認することを必要とする（5.4.1 節参照）。鍵情報が認証暗号モジュール内に存在する場合、その暗号モジュールが FIPS 140 に準拠し、鍵で保護されるデータに関連する FIPS 199 の影響レベルと一致するセキュリティレベルであれば、適切な完全性保護が提供される（SP 800-152 参照）。鍵情報が暗号モジュールの外部で利用可能な場合、完全性保護は、適切な暗号学的完全性メカニズム（暗号チェックサム、暗号学的ハッシュ関数、MAC、デジタル署名など）、非暗号学的完全性メカニズム（CRC、パリティチェックなど）（付録 A 参照）、

[78] SP 800-152, *A Profile for U.S. Federal Cryptographic Key Management Systems (FCKMS)*.

[79] 例えば、秘密鍵又はプライベート鍵の情報をアクセスが制限された金庫に保管し、その鍵への全てのアクセスをログに記録し、ログを定期的に確認して、不正なアクセスを検出した場合又は疑わしい場合は自動化されたアラートを使用する。

又は物理的保護メカニズムによって、提供**されなければならない**。適切な完全性メカニズムの選択のためのガイダンスは、6.2.1.2 節及び 6.2.2.2 節に記載されている。

関連性保護 は、正しい鍵材料が正しいアプリケーション又は装置において正しいデータを保護するために使用されることを保証するもので、暗号セキュリティサービスに対して提供**されなければならない**。適切な関連性保護の選択のためのガイダンスは、6.2.1.4 節及び 6.2.2.4 節に記載されている。

ドメインパラメータ及び公開鍵の有効性の保証 は、暗号アルゴリズムで使用されるパラメータ及び鍵が算術的に正しいことを保証するものである（5.4.2 節及び 5.4.3 節参照）。適切な保証メカニズムの選択のためのガイダンスは、SP 800-56A 及び SP 800-89、ならびに本文書に記載されている。

プライベート鍵保有の保証 は、公開鍵の所有者が対応するプライベート鍵を実際に保有していることを保証するものである（5.4.4 節参照）。

可用性保護 は、保護されたデータの即時使用（復号や鍵情報の継続的な完全性検証など）以外の局面でも利用可能である必要がある全ての鍵情報に対して提供**されなければならない**。これは、情報をバックアップ又はアーカイブすることで達成される（8.2.2.1 節及び 8.3.1 節参照）。

鍵情報の保護期間 は、鍵のタイプ、関連する暗号サービス、及び暗号サービスが必要とされる期間に依存する。保護期間には、鍵の暗号期間も含まれる（5.3 節参照）。保護期間は、完全性の場合と機密性の場合では必ずしも同じではない。完全性保護は鍵が使用されなくなる（まだ破棄されていない）までしか必要とされないが、機密性保護は鍵が実際に破棄されるまで必要とされることがある。

6.1.1 暗号鍵の保護及び保証要件の概要

表 5 は、配付及び保管中の鍵の保護要件をまとめたものである。必要な保護を提供するための方法については、6.2 節で説明する。

表 5 へのガイド：

a. 1 列目（鍵タイプ）は、鍵のタイプを特定する。

b. 2 列目（セキュリティサービス）は、暗号技術に関連して鍵が提供するセキュリティサービスのタイプを示す。この列では、“サポート”という用語が使用される場合がある。これは、関連する鍵が主要な暗号サービス（機密性、完全性認証、及びソース認証）をサポートするために使用されることを意味する。例えば、鍵合意鍵は、機密性を提供するために使用される鍵を確立することにより、機密性サービスをサポートすることができる。RBG 鍵は暗号技術の使用をサポートする。なぜなら、情報を暗号的に保護することに使用される鍵を生成する乱数値を提供することに使用されるためである。

c. 3 列目（セキュリティ保護）は、鍵に要求される保護の種類（すなわち、機密性や完全性、可用性）を示す。

d. 4 列目（関連性保護）は、鍵に対して保護する必要がある関連性の種類を示す。例えば、鍵と、使用法、アプリケーション、認可された通信参加者又はその他の指示された情報との関連付けなど。ドメインパラメータとの関連付けは、それらが使用されるアルゴリズムにのみ適用される。

e. 5 列目（必要な保証）は、SP 800-56A、SP 800-56B、SP 800-89 及び本推奨で定義されているように、公開鍵の有効性の保証やプライベート鍵の保有の保証を取得する必要があるかどうかを示す。公開鍵の有効性の保証は、鍵が算術的に正しいという信頼度を提供する。詳細は 5.4.3 節を参照のこと。プライベート鍵の保有の保証は、公開鍵を提供するエンティティが、ある時点で関連するプライベート鍵を実際に保有していたという信頼度を提供する。詳細は 5.4.4 節を参照のこと。

f. 6 列目（保護期間）は、鍵の完全性や機密性を維持する必要がある期間を示す（5.3 節参照）。対称鍵及びプライベート鍵は、保護期間の終了時に破棄**されなければならない**（8.3.4 節及び 9.4 節参照）。

表 5：暗号鍵の保護要件

鍵タイプ	セキュリティサービス	セキュリティ保護	関連性保護	保証の必要性	保護期間
署名プライベート鍵	ソース認証 完全性認証 否認防止のサポート	機密性 完全性 [80]	使用方法又はアプリケーション ドメインパラメータ（該当する場合） 署名検証公開鍵	保有	生成から暗号利用期間終了まで
署名検証公開鍵	ソース認証 完全性認証 否認防止のサポート	完全性 可用性	使用方法又はアプリケーション 鍵ペアの所有者 ドメインパラメータ（該当する場合） 署名プライベート鍵 署名データ	有効性	生成から保護データの検証が不要になるまで
認証対称鍵	ID 認証 完全性認証	機密性 完全性 可用性	使用方法又はアプリケーション その他の認可されたエンティティ 認証データ		生成から保護データの検証が不要になるまで
認証プライベート鍵	ID 認証 完全性認証	機密性 完全性	使用方法又はアプリケーション 認証公開鍵 ドメインパラメータ（該当する場合）	保有	生成から暗号利用期間終了まで
認証公開鍵	ID 認証 完全性認証	完全性 可用性	使用方法又はアプリケーション	有効性	生成から保護データの検証が不要に

[80] 完全性保護は、さまざまな方法で行うことができる。6.2.1.2 節及び 6.2.2.2 節参照。

			鍵ペアの所有者 認証データ 認証プライベート鍵 ドメインパラメータ（該当する場合）		なるまで
データ暗号化対称鍵／データ復号対称鍵	機密性	機密性 完全性 可用性	使用方法又はアプリケーション その他の認可されたエンティティ 平文／暗号化データ		生成からデータの寿命が尽きる又は暗号期間が終わるまでのいずれか遅い方の期間
鍵ラッピング対称鍵	サポート	機密性 完全性 可用性	使用方法又はアプリケーション その他の認可されたエンティティ 暗号化鍵		生成から暗号期間が終わる又はラップされた鍵が保護を必要としなくなるまでのいずれか遅い方の期間
RBG 対称鍵	サポート	機密性 完全性	使用方法又はアプリケーション		生成から交換まで
マスター対称鍵／鍵導出対称鍵	サポート	機密性 完全性	使用方法又はアプリケーション その他の認可されたエンティティ 導出鍵		生成から暗号期間が終わる又は導出鍵の寿命が尽きるまでのいずれか遅い方の期間
鍵配送プライベート鍵	サポート	機密性 完全性 可用性	使用方法又はアプリケーション 暗号化鍵 鍵配送公開鍵	保有	生成から全ての配送鍵の保護期間終了まで
鍵配送公開鍵	サポート	完全性	使用方法又はアプリケーション 鍵ペアの所有者 鍵配送プライベート鍵	有効性	生成から暗号利用期間終了まで
鍵合意対称鍵	サポート	機密性 完全性	使用方法又はアプリケーション その他の認可されたエンティティ		生成から暗号期間が終わる又は鍵を特定する必要がなくなるまでのいずれか遅い方の期間
静的鍵合意プライベート鍵	サポート	機密性 完全性	使用方法又はアプリケーション ドメインパラメータ（該当する場合）	保有	生成から暗号期間が終わる又は鍵を特定する必要がなくなるまでのいずれか遅い方の期間

			静的鍵合意公開鍵		
静的鍵合意公開鍵	サポート	完全性	使用方法又はアプリケーション 鍵ペアの所有者 ドメインパラメータ（該当する場合） 静的鍵合意プライベート鍵	有効性	生成から暗号期間が終わる又は鍵を特定する必要がなくなるまでのいずれか遅い方の期間
一時的鍵合意プライベート鍵	サポート	機密性 完全性	使用方法又はアプリケーション 一時的鍵合意公開鍵 ドメインパラメータ（該当する場合）		生成から鍵合意プロセスの終了まで プロセス終了後、鍵を破棄**しなければならない**
一時的鍵合意公開鍵	サポート	完全性 [81]	鍵ペアの所有者 一時的鍵合意プライベート鍵 使用方法又はアプリケーション ドメインパラメータ（該当する場合）	有効性	生成から鍵合意プロセスが完了するまで
認可対称鍵	認可	機密性 完全性	使用方法又はアプリケーション その他の認可されたエンティティ		生成から鍵の暗号利用期間終了まで
認可プライベート鍵	認可	機密性 完全性	使用方法又はアプリケーション 認可公開鍵 ドメインパラメータ（該当する場合）	保有	生成から鍵の暗号利用期間終了まで
認可公開鍵	認可	完全性	使用方法又はアプリケーション 鍵ペアの所有者 認可プライベート鍵 ドメインパラメータ（該当する場合）	有効性	生成から鍵の暗号利用期間終了まで

[81] 一時的鍵合意公開鍵の機密性は、送信中には保護されないかもしれない。しかし、鍵合意プロトコルは、送信された一時的公開鍵の不正な置換や改ざんを検出するように設計されている場合がある。この場合、プロトコルはデータ完全性メカニズムを形成する。

6.1.2 その他の関連情報の保護要件の概要

表 6 は、配付及び保管中のその他の関連情報の保護要件をまとめたものである。必要な保護を提供するためのメカニズムは、6.2 節で説明する。

表 6 へのガイド：

a. 1 列目（情報のタイプ）は、情報のタイプを特定する。

b. 2 列目（セキュリティサービス）は、情報が提供するセキュリティサービスのタイプを示す。

c. 3 列目（セキュリティ保護）は、情報に要求されるセキュリティ保護の種類を示す。

d. 4 列目（関連性保護）は、情報のタイプごとに関連性についての関連タイプを示す。

e. 5 列目（ドメインパラメータの有効性の保証）は、SP 800-56A、SP 800-56B、及び本推奨の 5.4 節で定義されている保証を得**なければならない**情報を示す。ドメインパラメータの有効性の保証は、ドメインパラメータが算術的に正しいという信頼を与える。

f. 6 列目（保護期間）は、情報の完全性や機密性を維持する必要がある期間を示す。情報は、保護期間の終了時に破棄**されなければならない**（8.3.4 節参照）。

表 6：その他の関連情報の保護要件

情報のタイプ	セキュリティサービス	セキュリティ保護	関連性保護	ドメインパラメータの有効性保証	保護期間
ドメインパラメータ	パラメータに関連付けられた鍵に依存	完全性 可用性	使用方法又はアプリケーション プライベート鍵及び公開鍵	Yes	生成から鍵生成又は署名検証が不要になるまで
初期ベクトル	アルゴリズムに依存	完全性 [82] 可用性	保護データ		生成から保護データを処理する必要がなくなるまで
共有秘密	サポート	機密性 完全性			生成からトランザクション終了まで 共有秘密は、保護期間の終了時に破棄**しなければならない**
RBG シード	サポート	機密性 完全性	使用方法又はアプリケーション		一度だけ使用し、使用後すぐに破棄

[82] 初期ベクトルは一般的に送信中に保護されない。しかし、復号システムは、送信された初期ベクトルに対する不正な置換や改ざんを検出したり、その影響を最小限に抑えるように設計されている場合がある。この場合、復号システムはデータ完全性メカニズムになる。

その他の公開情報	サポート	完全性	使用方法又はアプリケーション その他の認可されたエンティティ nonce を使用して処理されたデータ		生成から公開情報を利用したデータ処理が不要になるまで
その他の秘密情報	サポート	機密性 完全性	使用方法又はアプリケーション その他の認可されたエンティティ 秘密情報を利用して処理されたデータ		生成から秘密情報を利用したデータ処理が不要になるまで
中間結果	サポート	機密性 完全性	使用方法又はアプリケーション		生成から不要となって中間の結果が破棄されるまで
鍵コントロール情報（ID、目的など）	サポート	完全性 可用性	鍵		生成から関連する鍵が破棄されるまで
乱数	サポート	機密性（使い方次第） 完全性			生成から不要となって乱数が破棄されるまで
パスワード	ID 認証 鍵導出	機密性 完全性 可用性	使用方法又はアプリケーション 所有者		生成から交換されるまで、又はエンティティ認証や鍵導出が不要になるまで
監査情報	サポート	完全性 アクセス認可 可用性	監査イベント 鍵コントロール情報／鍵コントロールメタデータ		生成から不要になるまで

6.2 保護メカニズム

鍵情報の有効期間中、鍵情報は“配送中”（例えば、鍵情報を使用する認可された通信参加者に、手動で配付されるプロセス中、又は自動化されたプロトコルを使用して配付されるプロセス中である）、“休止中”（例えば、鍵情報が保管中である）、又は“使用中”のいずれかである。全ての場合について、鍵情報は、6.1 節に従って保護**されなければならない**。

使用中の鍵について、当該鍵は適切な暗号モジュール内に存在（して使用されるように）**しなければならない**。鍵が使用中であるからといって、当該鍵が同時に配送中や保管中であることを排除するものではないことに注意されたい。

配送中又は保管中は、保護メカニズムの選択が異なる場合がある。いくつかの保護方法が以下の小節で提供されているが、全ての方法が同等のセキュリティを提供するわけではない。方法は慎重に選択されるべきである。また、規定されたメカニズムは、それ自体が保護を保証するものではない。実装及び関連する鍵管理が、実行可能な攻撃が成功しないように十分なセキュリティを提供する必要がある。

6.2.1 配送中の鍵情報の保護メカニズム

配送中の鍵情報には、以下の目的で配付される鍵材料が含まれる場合がある：

- 暗号サービスを取得するため（例えば、機密性を提供するために使用する鍵を確立するため）（8.1.5 節参照）
- 将来の使用又は回復の可能性を考慮して鍵情報をバックアップ又はアーカイブするため（8.2.2 節及び 8.3.1 節参照）、又はバックアップやアーカイブされた鍵情報を回復するため（8.2.2.2 節、8.3.1 節及び付録 B 参照）。

これは、手動で（すなわち、信頼できる配達人を介して）、自動化された方法で（すなわち、自動化された通信プロトコルを使用して）、又は手動と自動化された方法の組み合わせによって達成され得る。いくつかのプロトコルでは、保護はプロトコルによって提供され、他のケースでは、鍵情報の保護が鍵情報に直接提供される（例えば、受信側当事者のみが復号するように、送信前に鍵材料を暗号化する）。保護メカニズムを適用するのは作成者の責任であり、使用されたメカニズムを元に戻す又はチェックするのは受領者の責任である。

6.2.1.1 可用性

通信が文字化けしたり、意図的に改ざんされたり、破壊されたりする可能性があるため、配送後の鍵情報の可用性は、暗号方式だけを使って保証することはできない。しかし、可用性は、冗長化や複数チャネル化、ストア＆フォワード方式（受信確認後にだけ送信者が削除する方式）、誤り訂正符号、及びその他の非暗号学的メカニズムによって、サポートすることが可能である。

通信システムは、送信後の鍵情報の可用性を保証するために、元の送信者による再送に頼るのではなく、非暗号学的メカニズムを組み込むべきである。

6.2.1.2 完全性

完全性保護には、情報の改ざんの防止と検出の両方が含まれる。改変が検出された場合、情報を変更されていない形に復元するための対策が取られることがある。不正な変更を検出するために暗号メカニズムがしばしば使用される。配送中の鍵情報の完全性は、以下のメカニズムの 1 つ以上を使用して保護され、検証されなければならない：

1. 手動方式（物理的な保護が施されている）：
   (a) 完全性メカニズムを使用して、配付される鍵情報に“コード”（CRC、MAC、デジタル署名など）を生成し、その結果得られたコードを鍵情報とともに受信者に提供する。受信したコードが受信者によって正常に検証された場合、受信者は、鍵情報が正しく受信されたことを保証する。そうでない場合は、鍵情報が破損しているとみなす。鍵を使用する暗号アルゴリズムを使用してコードを生成する場合（MAC やデジタル署名など）、送信者と受

信者は、そのコードの生成と検証のための適切な鍵を知っていなければならない。物理的保護により意図的な変更に対しては保護が出来ている状況では、MAC 又はデジタル署名の代わりに、CRC を使用してコードを生成する場合があることに注意されたい。

又は、

(b) 配付される鍵情報中の鍵は、相互に既知のデータに対して、意図された暗号処理を実行するために使用される（例えば、送信者と受信者の両方が既知の平文データを暗号化するために使用される）；鍵情報と暗号的に保護されたデータの両方が受信者に送信される。受信者が受信した鍵を使用して、送信者によって実行された暗号処理を正常に反転又は検証する（例えば、鍵を使用して受信した暗号文データを復号し、結果として得られた平文データを相互に既知の平文データと比較し、その比較に成功する）ことができれば、受信者は、鍵情報が正しく受信されたことの保証を得る。そうでない場合は、鍵情報が破損しているとみなす。

2. 通信プロトコルによる自動配付（送信エンティティ又は通信プロトコルによる保護）：

(a) **承認された**暗号学的完全性メカニズム（MAC やデジタル署名アルゴリズムなど）が配付される鍵情報に使用され、結果として得られたコードが、その後の検証のために鍵情報とともに受信者に提供される。この目的では、CRC の利用は承認されていないことに留意されたい。完全性メカニズムは、鍵情報のみに適用されてもよいし、メッセージ全体に適用されてもよい。

又は、

(b) 鍵情報に含まれる鍵は、送信者がデータに対して意図した暗号処理を実行するために使用される（例えば、データの MAC を計算する）。鍵情報と暗号的に保護されたデータの両方が受信者に送信される。受信者が鍵を使用して暗号処理を反転又は検証する（例えば、受信した鍵を使用して受信データの MAC を検証する）ことに成功できれば、受信者は、鍵情報が正しく受信されたことの保証を得る。そうでない場合は、鍵情報が破損しているとみなす。

完全性の障害が検出された場合の対応は、個別の環境によって異なる。不適切なエラー処理は、（サイドチャネル攻撃などの）攻撃を許す可能性がある。セキュリティポリシー（SP 800-57 パート 2 参照）では、このようなイベントに対する対応を定義**すべきである**。例えば、受信した鍵情報にエラーが検出され、受信者は鍵情報が完全に正しいことを要求する（例えば、鍵情報にエラーがある場合には、受信者は処理を進めることができない）場合には、以下の対応をポリシーとして定義する：

a. 鍵情報を使用**すべきではない**

b. 受信者は、鍵情報の再送を要求する（再送は、あらかじめ決められた上限回数に制限**されるべきである**）

c. インシデントに関連する情報は、後にエラーの原因を特定するために、監査ログに保存**すべきである**

6.2.1.3 機密性

機密性保護は、配送中の秘密の対称鍵、非対称プライベート鍵、鍵シェア、及び秘密のメタデータ（すなわち、秘密の鍵情報）に対して、以下のメカニズムのうちの 1 つ以上を使用して提供**されなければならない**。

1. 手動方式：

   (a) 秘密の鍵情報は、鍵材料に要求されるセキュリティ強度（すなわち、当該鍵によって保護されるデータの保護に必要なセキュリティ強度）を満たすかそれ以上のセキュリティ強度での保護を提供する**承認された**技術を使用して暗号化（例えば、ラッピング）される。

   又は、

   (b) 鍵は鍵シェアに分割され、各鍵シェアは当該鍵材料に要求されるセキュリティ強度（すなわち、当該鍵によって保護されるデータの保護に必要なセキュリティ強度）を満たすかそれ以上のセキュリティ強度で生成される。各鍵シェアは、一個人が全ての鍵シェアへのアクセスを得ることができないように、知識分割手順（8.1.5.2.1 節及び 8.1.5.2.2.1 節参照）を用いて取り扱う。秘密にする必要のあるメタデータは全て暗号化される。

   又は、

   (c) 適切な物理的・手続き的保護が提供される（例えば、信頼できる配達人の利用）。

2. 通信プロトコルによる自動配付：秘密の鍵情報は、鍵材料に要求されるセキュリティ強度（すなわち、当該鍵によって保護されるデータの保護に必要なセキュリティ強度）を満たすかそれ以上のセキュリティ強度での保護を提供する**承認された**技術を使用して暗号化（例えば、ラッピング）される。

6.2.1.4 用途又はアプリケーションとの関連性

鍵材料とその使用用途又はアプリケーションとの関連付けは、配付プロセス中に具体的に特定される（例えば、送信するメタデータに含まれている）か、又は使用アプリケーションによって暗黙的に定義**されなければならない**。鍵に関連するメタデータについての話題は、6.2.3 節を参照のこと。

6.2.1.5 他のエンティティとの関連付け

鍵材料と全ての適切なエンティティ（例えば、鍵材料を共有する全てのエンティティ）との関連付けは、配付プロセス中に具体的に特定される（例えば、公開鍵証明書を使用する）か、又は使用アプリケーションによって暗黙的に定義**されなければならない**。鍵に関連するメタデータについての話題は、6.2.3 節を参照のこと。

6.2.1.6 その他の関連する鍵情報との関連性

その他の関連する鍵情報（例えば、ドメインパラメータ、暗号化鍵／復号鍵、IV）との関連付けは、配付プロセス中に具体的に特定されるか、使用アプリケーションによって暗黙的に定義**されなければならない**。その他の関連する鍵情報に関連するメタデータについての話題は、6.2.3 節を参照のこと。

6.2.2 保管中の鍵情報の保護メカニズム

鍵情報は、あるデバイス又は記憶媒体の中で休止している（すなわち、保管されている）場合がある。これには、配送中又は使用中でもある鍵情報のコピーも含まれる。休止状態の鍵情報（すなわち、保存された鍵情報であって、暗号モジュール内に含まれる鍵情報を含むもの）は、6.1 節に従って保護**されなければならない**。様々な保護メカニズムを使用してもよい。

鍵情報は、アプリケーションがすぐに利用できるように（例えば、ローカルのハードディスクやサーバ上に）保存されてもよい。このような保存は、暗号モジュール内又はすぐにアクセス可能なストレージ（例えば、ローカルハードドライブ）に保存される鍵情報では一般的である。鍵情報は、リムーバブルメディア（例えば、CD-ROM）に電子的な形式で保存されたり、リモートアクセス可能な場所に保存されたり、ハードコピー形式で金庫に保管されたりすることもある。このような保存は、バックアップ又はアーカイブストレージでは一般的である。

6.2.2.1 可用性

鍵情報は、データが当該鍵によって保護されている限り、すぐに利用できるようにしておく必要がある場合がある。この保護を提供するための一般的な方法は、鍵情報の 1 つ以上のコピーを作成し、別々の場所に保管することである。鍵の暗号利用期間中、長期的な可用性を必要とする鍵情報は、通常の運用ストレージ（8.2.1 節参照）とバックアップストレージ（8.2.2.1 節参照）の両方に保存**されるべきである**。鍵の暗号利用期間終了後に保持される鍵情報は、アーカイブストレージに保管**されるべきである**（8.3.1 節参照）。本推奨は、バックアップとアーカイブストレージの両方に同じストレージメディアを使用することを排除するものではない。

各鍵タイプの長期的な可用性の要件に関する詳細は、バックアップストレージについては 8.2.2.1 節で、アーカイブストレージについては 8.3.1 節で述べている。

失われた鍵情報を（例えば、通常の保管場所から）置き換える、又は当該鍵の暗号利用期間終了後に暗号処理を行う際に利用するための鍵情報の復旧については、8.2.2.2 節（通常運用中の復旧）、8.3.1 節（アーカイブストレージからの復旧）及び付録 B で論じる。

本節の主な焦点は鍵情報の可用性の保証を提供することであるが、この鍵情報の可用性を退けることが望まれる場合が少なくとも 1 つある。すなわち、暗号化された大量の鍵情報を消去する場合である。この場合、暗号化消去（すなわち、鍵情報の復号又はアンラップに使用する鍵を破棄する）が推奨される（SP 800-88[83]参照）。

6.2.2.2 完全性

完全性保護は、鍵情報が正しいことを保証することに関係する。改ざんに対する完璧な保護は不可能である。できる最善の方法は、合理的な手段を用いて改ざんを防止し、改ざんが発生した場合には（非常に高い確率で）それを検出する方法を用い、かつ不正な改ざんが検出された場合には鍵情報を元の内容に戻すことである。

全ての鍵情報には完全性保護が必要である。完全性保護は、物理的メカニズム、暗号メカニズム、又はその両方によって提供**されなければならない**。

物理的メカニズムには、以下のようなものがある：

[83] SP 800-88, *Guidelines for Media Sanitation*.

1. 保存された鍵情報へのアクセスを制限する認証暗号モジュール又はオペレーティングシステム
2. 他のシステムに接続されていないコンピュータシステム又は媒体
3. コンピュータシステムの外部にある、適切なアクセス制御がなされた物理的に安全な環境（例えば、アクセスが制限された金庫の中）

暗号メカニズムには、以下のようなものがある：

a. 鍵情報に基づいて計算され、後に保存された鍵情報の完全性を検証するために使用される、**承認された**暗号完全性メカニズム（MAC、デジタル署名など）
b. 意図した暗号処理の実行。これは、正しい結果が容易に決定されることを前提としている。受信した鍵情報が正しくない場合、鍵材料が破損している可能性がある

エラーが検出された場合に鍵情報を復元できるように、鍵情報の 1 つ又は複数のコピーを物理的に別の場所（すなわち、バックアップ又はアーカイブストレージ ; 8.2.2.1 節及び 8.3.1 節参照)に保持**すべきである**。各コピーの完全性は、定期的にチェック**すべきである。**

6.2.2.3 機密性

保管されている秘密の鍵情報に機密性を提供するために、次のいずれかの仕組みを用い**なければならない**：

1. FIPS 140 認証暗号モジュール内での、**承認された**アルゴリズムを使った暗号化（又は鍵ラッピング)。暗号化では、秘密の鍵情報に要求されるセキュリティ強度を満たす、又はそれ以上のセキュリティ強度での保護を提供する**承認された**技術を使用**しなければならない**

   又は、

2. 鍵によって保護されるデータに関連付けられた FIPS 199 における影響レベルと整合するセキュリティレベルでの、FIPS 140 認証暗号モジュールによって提供される物理的保護（SP 800-152 参照）

   又は、

3. 管理されたアクセスが可能な安全な保管場所（例えば、金庫又は保護された場所）によって提供される物理的保護

6.2.2.4 用途又はアプリケーションとの関連性

鍵材料は、所定の暗号メカニズム（例えば、デジタル署名の生成、鍵確立など)、又は特定のアプリケーションで使用される。鍵材料が誤って使用されないことを確実にするための保護が提供**されなければならない**（例えば、用途又はアプリケーションが鍵材料と関連していなければならないだけでなく、この関連性の完全性が維持されなければならない)。この保護は、鍵材料を他のメカニズムやアプリケーションのそれと分離するか、あるいは鍵材料に関連付けられた適切なメタデータを使用することによって提供できる。6.2.3 節では、鍵に関連するメタデータを扱う。

6.2.2.5 他のエンティティとの関連性

鍵情報の中には、他のエンティティ（鍵ソース、鍵を所有又は使用するエンティティなど)と正しく関連付ける必要があるものがあり、この関連付けの完全性を維持**しなければならない**。例えば、鍵情報の暗号化や MAC の計算に使用される秘密の対称鍵は、その鍵を共有する他のエンティティと関連付ける必要がある。公開鍵は、（例えば、暗号化的な結び付きによる）関連付けにより、鍵ペアの所有者と正しく関連付ける（例えば、公開鍵証明書の使用による関連付け）必要がある。

鍵情報は、“エンティティ”、アプリケーション、又は必要に応じて鍵情報に適切なメタデータを使用して、鍵情報を分けて保存することにより、鍵情報の保存中の関連付けを保持**しなければならない**。6.2.3 節では、メタデータの使用について扱う。

6.2.2.6 その他の関連する鍵情報との関連性

保護された鍵情報と、その鍵情報を保護するために使用されている鍵材料との間での、関連付けは維持される必要がある場合がある。さらに、鍵は、その他の鍵情報との関連付けを必要とする場合がある（6.2.1.6 節参照)。

鍵情報を一緒に保存したり、情報間に何らかのリンクやポインタを提供したりすることは、関連付けの要件を満たすのに役立つ。多くの場合、鍵とそれが保護する情報との間のリンクは、鍵に識別子を提供し、その識別子を当該鍵のメタデータに格納し、その鍵の識別子を当該鍵によって保護されたデータと一緒に格納することによって、達成される。この関連付けは、保護データが処理される必要がある限り、維持**されなければならない**。

6.2.3 節では、メタデータの使用について扱う。

6.2.3 鍵のメタデータ

メタデータ [84]は、鍵に関する情報（その意図した用途を含む）を提供するために使用される。異なるアプリケーションでは同じ鍵タイプに対して異なるメタデータ要素が必要となる場合があり、また、異なる鍵タイプに対して異なるメタデータ要素が必要となる場合がある。各鍵に適したメタデータ要素を選択するのは実装者の責任である。メタデータが使用される場合、メタデータは鍵に付随**すべきである**（すなわち、メタデータは、通常、鍵とともに保存又は伝送される)。しかし、アプリケーションや実装によっては、あるメタデータが明示的に知られている場合もある（例えば、全ての情報が同じ機微度を持っている、全ての鍵が単一のアプリケーションによって使用されている、など)。

メタデータ要素の例としては、以下のようなものがある：

1. 鍵識別子
2. 鍵と一緒に使用されるアルゴリズム
3. 関連する鍵を識別する情報（例えば、公開鍵とプライベート鍵の関連性）
4. 鍵の所有者又は共有エンティティの身元
5. 所有者の保証人又は代理人の身元（所有者が人間以外のエンティティの場合）
6. 所有者がデバイス又はプロセスの場合は、デバイス又はプロセスの場所

[84] 鍵属性と呼ばれることもある。

7. 鍵の暗号利用期間（例えば、鍵の使用開始日と終了日）
8. 鍵のタイプ（署名プライベート鍵、暗号化鍵、マスター鍵など）
9. 鍵材料のソース（すなわち、鍵を提供したエンティティ）
10. 鍵が使用されるアプリケーション（購入、電子メールなど）
11. 鍵によって保護される情報の機微性
12. カウンタ [85]
13. 現在の鍵の状態（活性化前、活性化、破棄など）
14. 鍵の状態／履歴（配付、一時停止、失効（失効理由を含む）など）
15. 鍵ラッピングに使用する鍵ラッピング鍵の ID 及びラッピングに使用するアルゴリズムの ID
16. 使用される完全性保護メカニズム（例えば、鍵情報や保護コード（MAC 又はデジタル署名）の暗号保護を提供するために使用される、鍵とアルゴリズム）
17. その他の情報（鍵長、保護要件、鍵へのアクセス権を持つ者、追加の使用条件、など）

SP 800-152 は、メタデータの使用に関する追加情報を提供している。それには、メタデータの完全性の保護や関連する鍵との関連付けに関するガイダンスを含んでいる。

[85] 前に送信された鍵の再現を検出するために使用する。

7 鍵状態及び遷移

鍵は、生成から破棄までの間にいくつかの状態を通過する。図 3 は、鍵が想定しうる鍵状態とその間の遷移の例を示す。

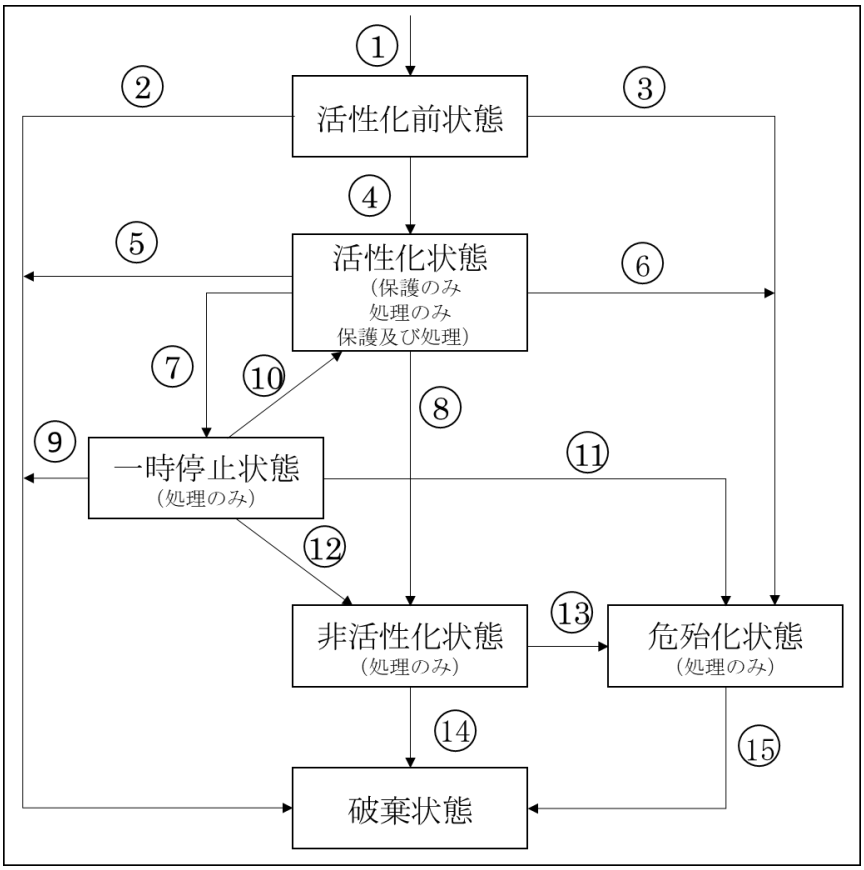

図 3：鍵状態と遷移の例

鍵は、鍵のライフサイクルにおける状態に応じて、異なる方法で使用される。単一の暗号モジュールの観点とは対照的に、複数の鍵状態がシステムの観点から定義される。以下の節では、運用中又はバックアップされた鍵の想定されうる状態を、図 3 に示す他の状態への遷移とともに説明する。追加の状態が、システムによっては適用可能な場合もある（例えば、本推奨の前バージョンで提供された例では、破棄された危殆化状態が示されている）。また、それらの状態の中には、あるシステムに必要ないものもあるかもしれない（例えば、一時停止状態は使用しないという決定が下される可能性がある）。

状態間の遷移は、しばしばイベントの記録を要求する。このような記録に適した場所は、監査ログと鍵のメタデータである（6.2.3 節参照）。SP 800-152 では、各イベントについて記録**しなければならない**最低限の情報を含め、これらのイベントの記録について説明している。

以下の節では、図 3 で示された状態と遷移の例について説明する。

7.1 活性化前状態

鍵は生成されたが、使用が認可されていない。棚卸リストに記載される鍵（長期鍵など）は、生成時に棚卸リストに記載**しなければならない**（9.2 節参照）。

この状態では、鍵は、所持証明（8.1.5.1.1.2 節）又は鍵確認（SP 800-175B 参照）の実行にのみ使用することができる。所持証明又は鍵確認の目的以外に、この状態の間、鍵を使用して、情報に暗号保護（送信や保存される情報への暗号化や署名など）を適用したり、暗号化された保護情報を処理（暗号文の復号やデジタル署名の検証など）**してはならない**。

状態遷移 1：

鍵は、生成された時点で直ちに活性化前状態に入る。

鍵の生成に関する情報は、記録**されなければならない**。

状態遷移 2：

鍵が活性化前状態にあり、将来的にその鍵が必要ないと判断された場合、当該鍵は、活性化前状態から破棄状態に直接遷移しなければならない。

非対称鍵の場合は、鍵ペアの両方の鍵が破棄状態に遷移**しなければならない**。

その遷移は記録**されなければならない**（SP 800-152 参照）。

状態遷移 3：

鍵が活性化前状態にあり、機密性保護が必要な鍵の機密性又は鍵の完全性が疑われる場合、当該鍵は、活性化前状態から危殆化状態に遷移**しなければならない**。

非対称鍵の場合、鍵ペアの両方の鍵が危殆化状態に遷移**しなければならない**。

その遷移は記録**されなければならない**（SP 800-152 参照）。その鍵が複数のエンティティに知られている場合は、失効通知が生成**されなければならない**。

状態遷移 4：

鍵は、その鍵が使用可能になると、活性化前状態から活性化状態に遷移**しなければならない**。この遷移は、活性化日に到達した時点で発生してもよいし、外部イベントにより発生してもよい。鍵がすぐに使用できるように生成された場合、遷移は活性化前状態に入った直後に行われる。

証明書に関連付けられた非対称鍵の場合、鍵ペアの両方の鍵が、当該鍵ペアの公開鍵に対して最初に発行された証明書内に記載される *notBefore* の日付に活性化される。

その遷移は記録**されなければならない**（SP 800-152 参照）。

この遷移は、対称鍵又は非対称鍵ペアの両鍵の暗号利用期間の開始を示す（5.3 節参照）。

7.2 活性化状態

活性化状態では、鍵を使って、情報を暗号化して保護（平文の暗号化やデジタル署名の生成など）したり、以前に保護された情報を暗号処理（暗号文の復号やデジタル署名の検証など）したり、又はその両方を行ったりする可能性がある。鍵が活性化状態の場合、当該鍵は、その鍵タイプに応じて、保護のみ、処理のみ、又は保護と処理の両方のためのものと指定されてもよい。例えば、非対称署名プライベート鍵と鍵配送公開鍵は、暗黙のうちに保護のみを適用するように指定され、署名検証公開鍵と鍵配送プライベート鍵は処理のみに指定される。データ暗号化対称鍵は、作成者使用期間中にデータを暗号化し、受領者使用期間中に暗号化データを復号するために使用される（5.3.5 節参照）。不許可（SP 800-131A 参照）である鍵は、処理にのみ使用**されなければならない**。

活性化状態での各鍵の使用状況は記録される**べきである**（SP 800-152 参照）。

状態遷移 5：

鍵タイプによっては、鍵が危殆化することなく、その鍵の暗号利用期間に達した又は鍵が交換された場合、活性化状態から破棄状態に直接遷移する。

非対称署名プライベート鍵及び認証プライベート鍵は、それぞれの鍵の作成者使用期間の終了時（例えば、対応する公開鍵に対して最後に発行された証明書の *notAfter* の日付に達した時）に破棄状態に遷移**しなければならない**。対応する公開鍵は、この時点で非活性化状態に遷移することに留意されたい。

RBG 対称鍵は、新しい鍵に置き換えられたとき、又は RBG が使用されなくなったときに、破棄状態に遷移**しなければならない**。

マスター対称鍵／鍵導出対称鍵、及び認可対称鍵は、それぞれの作成者使用期間の終了時に破棄状態に遷移**しなければならない** [86]。

非対称一時的鍵合意プライベート鍵は、使用後直ちに破棄状態に遷移**しなければならない**（SP 800-56A 参照）。対応する一時的鍵合意公開鍵は、対応するプライベート鍵が破棄されたときに、破棄状態に遷移**すべきである** [87]。

非対称認可プライベート鍵は、その暗号利用期間の終了時（例えば、対応する公開鍵に対して最後に発行された証明書の *notAfter* の日付に達した時）に破棄状態に遷移**しなければならない**。非対称認可公開鍵は、対応するプライベート鍵が破棄されたときに、破棄状態に遷移**すべきである** [88]。

その遷移は記録**されなければならない**（SP 800-152 参照）。

状態遷移 6：

対称鍵又は非対称鍵ペアは、対称鍵の完全性又は機密性保護を必要とする非対称鍵の機密性が疑われる場合、活性化状態から危殆化状態に遷移**しなければならない**。この場合、当該鍵又は鍵ペアは失効**されなければならない**。

---

86 鍵合意対称鍵及び認可対称鍵の受領者使用期間は、それらの作成者使用期間と同じであることを思い出されたい（5.6 節参照）。

87 認証プライベート鍵と認証公開鍵の暗号利用期間が同じであることを思い出されたい（5.6 節参照）。

88 認可プライベート鍵と認可公開鍵の暗号利用期間が同じであることを思い出されたい（5.6 節参照）。

非対称鍵ペアの場合、危殆化は鍵ペアのプライベート鍵に係わるのは明白だが、鍵ペアの両方の鍵を同時に危殆化状態に遷移**させなければならない**。例えば、署名プライベート鍵又は鍵配送プライベート鍵が危殆化した又は危殆化の疑いがある場合、対応する公開鍵も危殆化状態に遷移する必要がある。

その遷移は記録されなければならない（SP 800-152 参照）。その鍵が複数のエンティティに知られている場合は、失効通知が生成**されなければならない**。

状態遷移 7：

一時停止状態がアプリケーションによって使用されている場合（7.3 節参照）、対称鍵又は鍵ペアの両方の鍵は、何らかの理由により、当該鍵又は鍵ペアが一定期間（すなわち、一時停止期間）使用されないのであれば、活性化状態から一時停止状態に遷移**しなければならない**。例えば、署名プライベート鍵は、当該鍵に関連付けられたエンティティが休暇中であったり、その鍵が危殆化した疑いがあるといった理由で、一時停止されることがある。後者の場合、一時停止により、費用のかかる失効及び交換のプロセスを開始する前に、鍵の状態を調査することができる。

RBG 対称鍵は、一時停止状態のままではなく、危殆化状態に遷移**させなければならず**、かつ交換**されなければならない**。

その遷移は記録**されなければならない**（SP 800-152 参照）。その鍵又は鍵ペアが複数のエンティティに知られている場合、一時停止とその理由を示す通知が生成**されなければならない**。

状態遷移 8：

活性化状態の鍵又は鍵ペアは、データに暗号保護を適用するのに使用しなくなった場合、非活性化状態に遷移**しなければならない**。非活性化状態への遷移は、対称鍵が交換された（8.2.3 節参照）、作成者使用期間の終了日に達した（5.3.4 節及び 5.3.5 節参照）、又は鍵や鍵ペアが危殆化以外の理由で失効した（例えば、鍵の所有者がデータを暗号化するために鍵を使用する権限がなくなった）などの理由であってもよい。

認証対称鍵、データ暗号化対称鍵／データ復号対称鍵、鍵合意対称鍵、及び鍵ラッピング鍵は、その鍵の作成者利用期間の終了時に非活性化状態に遷移する。

署名検証公開鍵、認証公開鍵、及び静的鍵合意プライベート鍵／公開鍵の鍵ペアは、対応するプライベート鍵の作成者使用期間の終了時 （例えば、公開鍵に対して最後に発行された証明書の *notAfter* の日付に達した時）に非活性化状態に遷移する。

一時的鍵合意公開鍵及び認可公開鍵は、対応するプライベート鍵が破棄されたときに破棄されていなければ、非活性化状態に遷移する（遷移 5 参照）。

鍵配送プライベート鍵と公開鍵の鍵ペアは、公開鍵に対して最後に発行された証明書の *notAfter* の日付に到したときに、非活性化状態に遷移する。

その遷移は記録**されなければならない**（SP 800-152 参照）。

7.3 一時停止状態

鍵又は鍵ペアの使用は、いくつかの起こり得る理由により、一時停止されることがある。（非対称鍵ペアの場合、公開鍵とプライベート鍵の両方を同時に一時停止**しなければならない**ことに留意されたい）。

一時停止の理由の一つには、鍵危殆化の可能性がある。この場合、状況を調査するための時間を確保するために、一時停止が発せられるかもしれない。もう一つの理由は、デジタル署名の鍵ペアを所有するエンティティが利用できない場合（例えば、長期休暇中）である。一時停止期間中にサインしたと想定される署名は無効となる。

一時停止の理由に応じて、一時停止された鍵又は鍵ペアは、活性化状態、非活性化状態又は破棄状態に復元されるか、又は危殆化状態に遷移しうる。

一時停止されている鍵は、暗号保護（平文の暗号化やデジタル署名の生成など）を適用するために使用**してはならない**。これには、暗号保護の適用に使用されることのない鍵（すなわち、署名検証公開鍵、認証公開鍵、鍵配送プライベート鍵、認可公開鍵）を除き、5.1.1 節に記載されている全ての鍵タイプが含まれる。

一時停止の理由によっては、一時停止されている鍵は、一時停止期間前に暗号保護が適用された情報を処理するために使用される可能性がある。これには、デジタル署名を検証するための検証公開鍵の使用、暗号化情報を復号するためのデータ暗号化対称鍵の使用、鍵材料をアンラップするための鍵ラッピング対称鍵の使用などが含まれる。一時停止の理由が危殆化の疑いである場合、リスクが許容できるとしても、当該鍵ペアがその後再活性化される（すなわち、当該鍵が危殆化していなかった）までは、公開鍵を使用して署名を検証することは賢明ではないかもしれない。ただし、所有者が休暇中であることが理由の場合は、一時停止期間の開始前に生成したと判明している署名の検証を行うことが正当化される場合がある。

一時停止期間中にその情報に対して保護が適用されたと判明しているなら、当該情報に対して処理を**してはならない**。

状態遷移 9：

鍵タイプによっては、危殆化していないと判断された場合、一時停止状態から破棄状態に遷移する。

一時停止状態にある署名プライベート鍵及び認証プライベート鍵は、作成者使用期間の終了時（例えば、対応する公開鍵に対して最後に発行された証明書の *notAfter* の日付に達した時）に破棄状態に遷移**しなければならない**。この時点で、対応する公開鍵は、非活性化状態に遷移することに留意されたい（状態遷移 12 参照）。

一時停止状態にあるマスター対称鍵／鍵導出対称鍵、及び認可対称鍵は、作成者使用期間の終了時に破棄状態に遷移**しなければならない**[89]。

一時停止状態にある認可プライベート鍵は、その作成者使用期間の終了時（すなわち、対応する公開鍵に対して最後に発行された証明書の *notAfter* の日付に達した時）に破棄状態に遷移**しなければならない**。認可公開鍵は、対応するプライベート鍵が破棄された時点で破棄状態に遷移**すべきである**[90]。

その遷移は記録**されなければならない**（SP 800-152 参照）。

状態遷移 10：

---

89 鍵合意対称鍵及び認証対称鍵の受領者使用期間は、それらの作成者使用期間と同じであることを思い出されたい（5.3.6 節参照）。

90 認可プライベート鍵と認可公開鍵の暗号利用期間が同じであることを思い出されたい（5.6 節参照）。

一時停止状態にある鍵又は鍵ペアは、一時停止の理由が存在しなくなり、かつ作成者使用期間の終了に達していない場合、活性化状態に遷移**しなければならない**。

対称鍵の場合、当該鍵の作成者使用期間が終了する前にこの遷移は行われる必要がある。

非対称鍵の場合、例えば、公開鍵に対して最後に発行された証明書の *notAfter* の日付よりも前にこの遷移は行われる必要がある。この場合、プライベート鍵と公開鍵の両方が、同時に遷移**しなければならない**。

その遷移は記録**されなければならない**（SP 800-152 参照）。

状態遷移 11：

一時停止状態の鍵又は鍵ペアは、鍵の完全性又は機密性保護が必要な鍵の機密性が疑われた場合、又はそれが確認された場合、危殆化状態に遷移**しなければならない**。この場合、当該鍵又は鍵ペアは失効**されなければならない**。

非対称鍵ペアの場合、公開鍵とプライベート鍵の両方を同時に遷移**させなければならない**。

その遷移は記録**されなければならない**（SP 800-152 参照）。その鍵が複数のエンティティに知られている場合は、失効通知が生成**されなければならない**。

状態遷移 12：

鍵タイプによっては、危殆化していないことが判明し、一時停止の必要性がなくなった場合、一時停止状態から非活性化状態に遷移する。

認証対称鍵、データ暗号化対称鍵／データ復号対称鍵、鍵合意対称鍵、及び鍵ラッピング対称鍵は、作成者使用期間の終了には達したが、受領者使用期間の終了には達していない場合、非活動状態に遷移**しなければならない**。

署名検証公開鍵、認証公開鍵、及び静的鍵合意プライベート鍵／公開鍵の鍵ペアは、プライベート鍵の作成者使用期間の終了時（例えば、公開鍵に対して最後に発行された証明書の *notAfter* の日付に達した時）に非活性化状態に遷移する[91]。一時的鍵合意公開鍵及び認可公開鍵は、対応するプライベート鍵が破棄されたときに破棄されていなければ、非活性化状態に遷移する（状態遷移 9 参照）。

鍵配送プライベート鍵／公開鍵の鍵ペアは、鍵ペアの暗号利用期間の終了時（例えば、公開鍵に対して最後に発行された証明書の *notAfter* の日付に達した時）に非活性化状態に遷移する。

その遷移は記録**されなければならない**（SP 800-152 参照）。

[91] 一時的鍵合意公開鍵の場合、暗号利用期間は、対応する一時的鍵合意プライベート鍵（使用後に破棄状態に移行したもの。状態遷移 5 参照）と同じタイミングで終了する。しかし、公開鍵を直ちに破棄するという実際の要件はないので、ここでは破棄状態ではなく非活性化状態に遷移するものとして記載している。破棄状態に直接遷移することも許容される。

7.4 非活性化状態

非活性化状態の鍵は、暗号保護を適用するために使用**されてはならない**が、場合によっては、暗号化された保護情報を処理するために使用されうる。鍵が（危殆化以外の理由で）失効した場合、当該鍵は引き続き処理に使用されうる。

鍵アーカイブから取り出された鍵（例えば、アーカイブされたデータの復号のために）は、非活性化状態にあるかもしれないことに注意されたい。

- 署名検証公開鍵は、対応するプライベート鍵の作成者使用期間の終了前（例えば、公開鍵の最後の証明書の *notAfter* の日付よりも前）に生成されたデジタル署名を検証するために使用されうる。
- 認証対称鍵、データ暗号化対称鍵、及び鍵ラッピング対称鍵は、当該鍵の作成者使用期間中に保護が適用されたことを条件に、受領者使用期間が終了するまで暗号化された保護情報を処理するために使用されうる。
- 認証公開鍵は、対応するプライベート鍵の作成者使用期間の終了前（例えば、公開鍵の最後の証明書の *notAfter* の日付よりも前）に実行された処理を認証するために使用されうる。
- 鍵配送プライベート鍵は、公開鍵の作成者使用期間が終了する前（例えば、公開鍵の最後の証明書の *notAfter* の日付よりも前）に対応する公開鍵を使用して暗号化された鍵を復号するために使用されうる。
- 鍵合意対称鍵は、十分な情報が利用できる前提で、（既に）合意された鍵を決定するために使用されうる。ただし、それらを新しい鍵を決定するために使用**されてはならない**。
- 静的鍵合意プライベート鍵／公開鍵は、鍵ペアの暗号利用期間が終了する前（例えば、利用した鍵合意スキームについて十分な情報が利用できる前提で、公開鍵の最後の証明書の *notAfter* の日付よりも前）に作成された合意された鍵を再生成するために使用されうる。
- 一時的鍵合意公開鍵は、（利用した鍵合意スキームについて十分な情報が利用できる前提で）合意された鍵を再生成するために使用されうる。
- 認可公開鍵は使用**されてはならない**。

非活性化状態にある鍵は、ある時点で危殆化状態又は破棄状態に遷移してもよい。

状態遷移 13：

鍵は、その鍵の完全性又は機密性保護を必要とする鍵の機密性が疑われるようになった場合、非活性化状態から危殆化状態に遷移**しなければならない**。この場合、当該鍵又は鍵ペアは失効**されなければならない**。

その遷移は記録**されなければならない**（SP 800-152 参照）。その鍵が複数のエンティティに知られている場合は、失効通知が生成**されなければならない**。

状態遷移 14：

非活性化状態にある対称（秘密）鍵及び非対称プライベート鍵は、（例えば、データの復号に）不要になった場合には破棄状態に遷移**しなければならない**。

非活性化状態の公開鍵は、不要になった時点で速やかに破棄状態に遷移**すべきである**。公開鍵が破棄されるか否かに関わらず、メタデータは監査目的で保持される**べきである**（8.4 節参照）。

その遷移は記録**されなければならない**（SP 800-152 参照）。

7.5 危殆化状態

一般的に、鍵は、認可されていないエンティティに提供されたり、認可されていないエンティティによって特定されたりした場合、危殆化する。危殆化した鍵は、情報に対する暗号保護を適用するために使用**されてはならない**。しかし、場合によっては、危殆化した対称鍵、又は危殆化したプライベート鍵に対応する鍵ペアの公開鍵が、暗号化された保護情報を処理するために使用されることがある。例えば、署名は、その署名の危殆化が発生する前から物理的に保護されている場合、又は信頼できるタイムスタンプが署名データに含まれている場合であれば、署名データの完全性を判断するために検証されてもよい。この処理は、情報のユーザが起こりうる結果を十分に認識したうえで、高度に管理された条件下でのみ実行**されなければならない**。

アーカイブから取り出された鍵は、危殆化状態にある可能性があることに注意されたい。

状態遷移 15：

危殆化した対称（秘密）鍵又は非対称プライベート鍵は、破棄状態に遷移**しなければならない**。

危殆化した公開鍵は、その使用が許可されなくなったり、必要とされなくなったりした時点で、破棄状態に遷移**すべきである**。公開鍵が破棄されるか否かに関わらず、メタデータは監査目的で保持される**べきである**（8.4 節参照）。

その遷移は記録**されなければならない**（SP 800-152 参照）。

7.6 破棄状態

鍵は、8.3.4 節で規定されたように破棄される。この状態では鍵はもはや存在しないが、監査のために特定のメタデータ（鍵の状態遷移履歴、鍵名、タイプ、暗号利用期間など）は保持されるかもしれない（8.4 節参照）。

鍵が破棄された後、破棄された当該鍵の危殆化が判明する可能性がある。この場合、そのイベントは記録**されなければならない**（SP 800-152 参照）。

8 鍵管理のフェーズと機能

暗号鍵管理のライフサイクルは、4 つのフェーズに分けることができる。鍵は、各フェーズにおいて、7 節で説明されているある特定の鍵状態にある。さらに、各フェーズ内において、通常は特定の鍵管理機能が実行される。これらの機能は、鍵とそれに関連するメタデータの管理に必要なものである。

鍵管理情報はメタデータと呼ばれる。鍵管理に必要なメタデータには、その鍵に関連付けられた人やシステムの身元や、その人にアクセスが認可されている情報の種類が含まれる場合がある。アプリケーションはメタデータを使い、特定のサービスに適した暗号鍵を選択する。メタデータは暗号アルゴリズムには登場しないが、アプリケーションやアプリケーションプロトコルの実装には欠かせないものである。

鍵管理の 4 つのフェーズとは、以下の通りである：

1. **運用前フェーズ**：鍵材料は、通常の暗号処理にはまだ利用できない。鍵は活性化前状態にあるか、又はまだ生成されていないかもしれない。システム又は企業の属性は、このフェーズで確立される。
2. **運用フェーズ**：鍵材料が使用可能で、通常の使用状態である。鍵は、活性化状態又は一時停止状態にある。活性化状態の鍵は、保護のみ、処理のみ、又は保護と処理の両方として指定することができる。一時停止状態の鍵は、処理のみに使用することができる（7.3 節参照）。
3. **運用後フェーズ**：鍵材料はもはや通常の使用状態ではなくなるが、鍵材料へのアクセスは可能であり、鍵材料は保護された情報を処理するために使用することができる。鍵は、非活性化状態又は危殆化状態にある。運用後フェーズの鍵は、アーカイブの中にある可能性がある（8.3.1 節参照）。

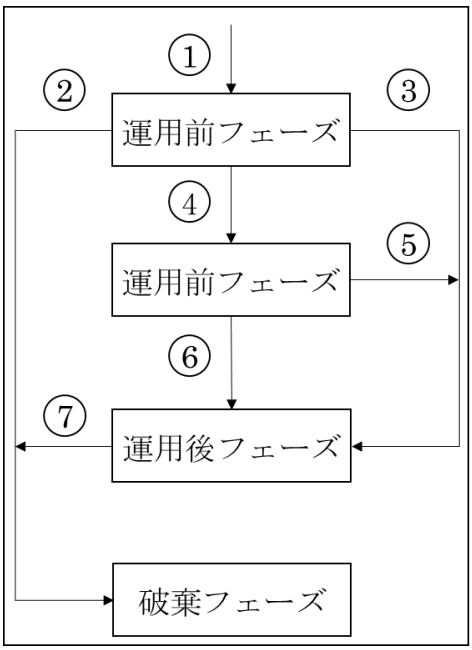

図 4：鍵管理フェーズ

4. **破棄フェーズ**：鍵はもはや利用できない。その存在の記録は削除されているかもしれないし、削除されていないかもしれない。鍵は破棄された状態にある。鍵自体は破棄されたかもしれないが、当該鍵のメタデータ（鍵名、タイプ、暗号利用期間、使用期間など）は保持されているかもしれない（8.4 節参照）。

鍵管理フェーズのフロー図を図 4 に示す。この図では、7 つのフェーズの遷移が示されている。鍵は、遷移前のどのフェーズにも戻ることはできないように**しなければならない**。

フェーズ遷移 1： 鍵は、生成時の運用前フェーズ（活性化前状態）にある。

フェーズ遷移 2： 鍵が生成されたが使用されなかった場合、運用前フェーズから直接破棄フェーズに遷移して破棄されることがある。

フェーズ遷移 3： 運用前フェーズの鍵が危殆化した場合、運用後フェーズ（危殆化状態）に遷移する。

フェーズ遷移 4： 必要なメタデータが確立され、鍵材料が生成され、運用前フェーズで鍵にメタデータが関連付けられた後、当該鍵はアプリケーションが使用できる状態になり、適切なタイミングで運用フェーズに遷移する。

フェーズ遷移 5： 運用フェーズの鍵が危殆化した場合、運用後フェーズ（危殆化状態）に遷移する。

フェーズ遷移 6： 鍵が通常使用に必要とされなくなった（すなわち、暗号利用期間が終了し、鍵が活性化状態でなくなった）が、当該鍵へのアクセスを維持する必要がある場合、当該鍵は運用後フェーズに遷移する。

フェーズ遷移 7： アプリケーションによっては、アクセスを一定期間保存し、その後、鍵材料を破棄することを要求する場合がある。運用後フェーズの鍵が不要になったことが明らかになった場合、破壊フェーズに遷移する場合がある。

鍵状態と鍵フェーズの組み合わせを図 5 に示す。

以下の小章では、鍵管理の各フェーズで実行される機能について説明する。本文中の“システム”とは、鍵を扱うシステムのことを指し、システムが何らかのプロトコルを使用するクライアントであるか、鍵管理システム全体であるかを問わない。多くの機能は、鍵管理システムによって実行されてもよい。鍵管理システムは、いくつかの機能が適切でない場合があり、特定された全ての機能を有していない場合がある。場合によっては、1 つ以上の機能を組み合わせてもよいし、機能を異なる手順で実行してもよい。例えば、鍵が暗号保護を適用するのに使用されなくなった場合、又は鍵が危殆化した場合に直ちに破壊されるならば、システムは運用後フェーズの機能を省略してもよい。この場合、鍵は運用フェーズから破棄フェーズに直接遷移する。

8.1 運用前フェーズ

鍵管理の運用前フェーズでは、鍵材料は、通常の暗号運用にまだ利用できない。

8.1.1 エンティティ登録機能

鍵管理システムへの登録中に、エンティティは、登録局とやり取りをして、セキュリティドメインの認可されたメンバーになる。このフェーズでは、将来の取引においてメンバーを識別するために、エンティティ識別子が確立されうる。特に、セキュリティインフラストラクチャが、識別情報をエンティティの鍵と関連付けを行いうる（8.1.5 節及び 8.1.6 節参照）。エンティティは、電子メールアドレスなどの様々な情報を提供してもよく、システム管理者によってエンティティの役割と認可情報が確立されてもよい。ID 情報と同様に、この情報は、インフラストラクチャによってエンティティの鍵と関連付けられ、セキュアなアプリケーションレベルのセキュリティサービスをサポートすることがある。

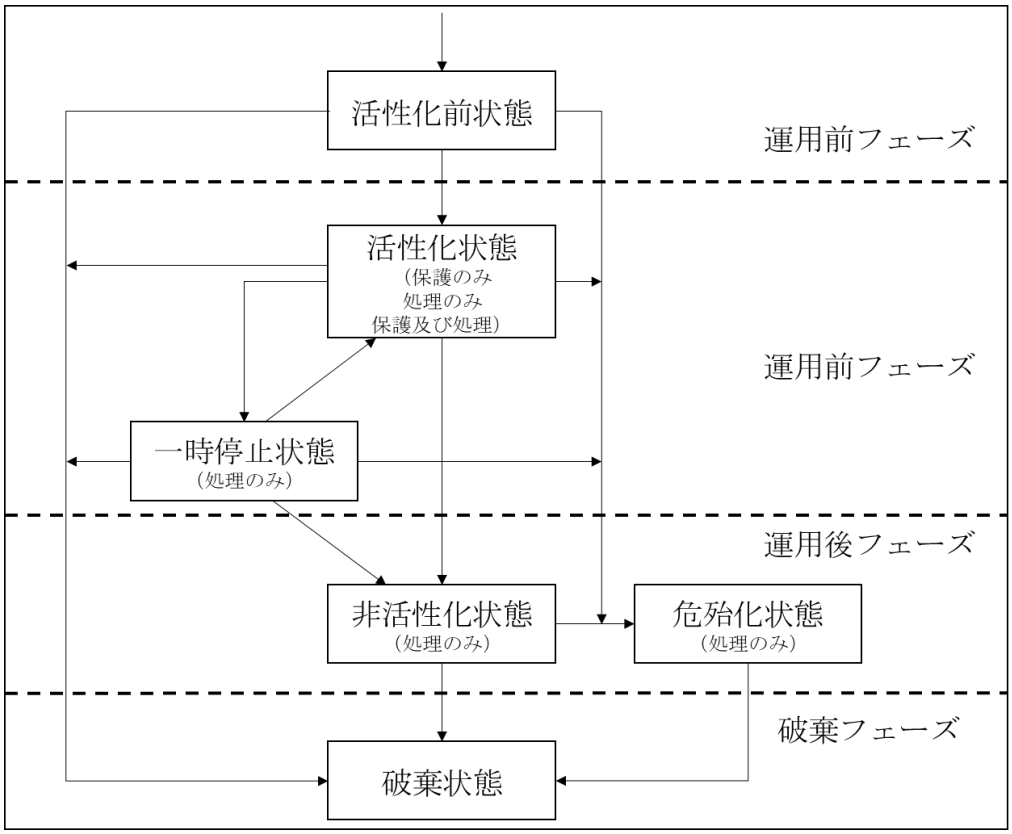

図 5：鍵管理の状態と段階

アプリケーションは、このプロセス中に確立された ID に依存するので、ID の検証（すなわち、ID 証明）のための適切な手順を確立して使用することが極めて重要である。ID 証明は、組織（例えば、組織のセキュリティオフィス）が行うことが多いが、鍵管理システムの登録局が行うこともある。セキュリティインフラストラクチャの強さ（又は弱さ）は、多くの場合、本人確認プロセスに依存する。FIPS 201[92]及び SP 800-63[93]は、ID を確立するための要件に対処している。

エンティティ登録と鍵登録（8.1.6 節参照）は、別々に実行されてもよいし、同時に実行されてもよい。別々に実行する場合、エンティティ登録プロセスは一般に秘密値（パスワード、PIN、HMAC 鍵など）を確立する。秘密値は、鍵登録ステップでエンティティの身元を認証するために使用することができる。同時に実行される場合、エンティティは同一のプロセスで身元を確立し、鍵登録を実行するので、秘密値は必要ない。

8.1.2 システム初期化機能

システムの初期化には、安全な運用のためにシステムをセットアップ又は構成設定することが含まれる。鍵を扱うシステムの場合、これには、アルゴリズムの設定、信頼される当事者の識別、及びドメインパラメータポリシーと信頼されるパラメータ（例えば、認識された証明書ポリシー）の定義が含まれる。

---

[92] FIPS 201, *Personal Identity Verification (PIV) of Federal Employees and Contractors*.

[93] SP 800-63, *Digital Identity Guidelines*.

8.1.3 初期化機能

初期化は、エンティティがその暗号アプリケーションを初期化すること（例えば、ソフトウェアやハードウェアのインストールや初期化）からなる。これには、エンティティ登録時に得られる可能性のある初期鍵材料の使用又はインストール（8.1.4 節参照）が含まれる。例としては、CA での鍵のインストール、信頼パラメータ、ポリシー、信頼された当事者、アルゴリズムの設定などが挙げられる。

8.1.4 鍵材料インストール機能

鍵材料のインストールにおけるセキュリティは、システムのセキュリティにとって非常に重要であり、鍵材料がシステム内（例えば、システムの暗号モジュール内）で生成されるか、別のシステム内で生成され、その後（ターゲットの）システムにインストールされるかに依存しない。この機能のために、鍵材料は、エンティティのソフトウェア、ハードウェア、システム、アプリケーション、暗号モジュール、又はデバイス内での運用上の使用のために、様々な技法を用いてインストールされる。鍵材料は、初期セットアップ中に、新しい鍵材料を既存の鍵材料に追加したり、既存の鍵材料を交換（例えば、鍵再設定又は鍵導出を介して；8.2.3 節及び 8.2.4 節参照）したりする場合に、インストールされる。

鍵材料の初期インストール（例えば、手動入力、電子的鍵ローダの使用、製造時のベンダによる）のプロセスには、ソフトウェア、ハードウェア、システム、アプリケーション、デバイス、又は暗号モジュールへの入力中の鍵材料の保護（保護レベルの違いに応じた FIPS 140 の要件及び個々に異なる要件を考慮に入れて）を含み、さらに必要とされる可能性のある追加の手順を含むものに**しなければならない**。

多くのアプリケーションやシステムには、新たにインストールされたアプリケーション／システムが適切に機能していることをテストするために使用される鍵材料がメーカから提供される。このテスト用鍵材料は、運用に使用**してはならない**。

8.1.5 鍵確立機能

鍵確立には、エンティティ間の通信のための鍵材料の生成、及び配付又は合意が含まれる。全ての鍵は、FIPS 140 認証暗号モジュール内で生成されるか、国家安全保障情報の保護のために米国政府が承認した別のソース源から入手**しなければならない**。長期鍵は、棚卸リスト管理**されなければならない**（9.2 節参照）。鍵確立プロセスの間、一部の鍵材料は配送中である場合がある（すなわち、鍵材料が手動配付中か、又は自動化されたプロトコルを使用して配付中の場合である）。他の鍵材料は、配付されるのではなく、ローカルに保持されているかもしれない。いずれの場合も、鍵材料は 6 節に従って保護**されなければならない**。

エンティティは、個人（人）、組織、デバイス、又はプロセスであってもよい。鍵材料をエンティティが自身の使用のために生成する場合、6.2.2 節の保管情報の適切な保護メカニズムのうち 1 つ以上を使用**しなければならない**。鍵の“所有者”は、鍵の使用を認可されたエンティティである。所有者が人間ではない（すなわち、所有者が組織、デバイス、又はプロセスである）場合、その所有者は鍵材料の取得及び管理を、多くの場合、認可された人間の代理人又は保証人によって支援される（例えば、鍵と証明書の取得とインストール）。

エンティティ間、又はエンティティとそのサブエンティティ（組織内の様々な個人、デバイス、プロセスなど）の間で配付される鍵材料は、配付中に 6.2.1 節に規定されている適切な保護メカニズムのうちの 1 つ以上を使用して保護**されなければならない**。配付されない鍵材料（鍵ペアのプライベート鍵、

対称鍵の自分自身のコピーなど)、又は受信した後に保存される鍵材料は、6.2.2 節に規定されている適切な保護メカニズムの 1 つ以上を使用して保護**されなければならない**。

8.1.5.1 節及び 8.1.5.2 節では、それぞれ非対称鍵及び対称鍵の生成と配付について述べる。SP 800-133 は、鍵材料の生成について述べる。

8.1.5.1 非対称鍵ペアの生成及び配付

鍵ペアは、適切な**承認された** FIPS 又は NIST 推奨の数学的仕様に従って生成**されなければならない**。

静的鍵ペアは、以下のいずれかによって生成**されなければならない**：1）鍵ペアを所有するエンティティ（すなわち、暗号計算でプライベート鍵を使用するエンティティ)、2）8.1.5.1.3 節に従って鍵ペアを配付する施設、又は 3）所有者と施設の協力プロセス、のいずれか。

デジタル署名の鍵ペア（署名検証公開鍵とそれに関連するプライベート鍵）の場合、静的鍵ペアの所有者が鍵材料を生成**すべきであり**、当該所有者のために他のエンティティが鍵材料を生成すべきではない。これにより、否認防止のサポートが容易になる。鍵ペアを所有するエンティティが生成した場合、署名プライベート鍵は他のエンティティに配付**してはならない**。しかし、所有者が組織である場合には、鍵材料を組織のサブエンティティ（従業員やデバイスなど）に配付することは許容される。この場合、組織が真の所有者であり、サブエンティティが所有者を代理する。

一時的鍵は、静的鍵の使用の代わりに又は静的鍵に加えて、鍵合意に使用されることが多い（SP 800-56A 参照)。一時的鍵のペアは、新しい鍵確立トランザクションごとに（例えば、メッセージごと又はセッションごとに固有)、所有者によって生成される。

生成された鍵ペアは、6.1.1 節に従って保護**されなければならない**。

8.1.5.1.1 公開鍵の配付

静的公開鍵は比較的寿命が長く、通常はアルゴリズムの数回の実行に使用される。一時的公開鍵は短命である。静的公開鍵又は一時的公開鍵の配付では、公開鍵の受信者に対して、当該鍵の真の所有者が既知であること（すなわち、主張された所有者が実際の所有者であること）が保証される**べきである**。匿名性が許容される場合は、この要件を無視しても構わない。しかし、全体的なアーキテクチャの強度と保護データの妥当性に対する信頼性は、大部分が公開鍵所有者の身元の保証に依存する。

また、公開鍵の配付は、受信者に対して以下のことを保証するもので**なければならない**：

1. その鍵の目的・用途が既知であること（RSA デジタル署名、楕円曲線の鍵合意など）
2. 公開鍵に関連する全てのパラメータが既知であること（ドメインパラメータなど）
3. 公開鍵が有効であること（例えば、公開鍵が必要とされる演算特性を満たしている）
4. 所有者が対応するプライベート鍵を実際に所有していること

8.1.5.1.1.1 PKI におけるトラストアンカーの公開鍵の配付

信頼された認証局（CA）の公開鍵は、全ての PKI ベースのセキュリティサービスの基盤である。信頼された CA は、トラストアンカーであると見なされる。トラストアンカーの公開鍵は秘密ではないが、その公開鍵の真正性は PKI の重要な前提である。トラストアンカーの公開鍵は、様々なレベルの保証を提供する、多様なメカニズムを介して取得されうる。提供されるメカニズムの種類は、インフラストラ

クチャ内のエンティティの役割に依存する。常に“依拠当事者”としてのみ行動するエンティティ（つまり、インフラストラクチャに登録された鍵を持たないエンティティ）は、インフラストラクチャに登録された鍵を保有するエンティティとは異なるメカニズムを使用する可能性がある。

トラストアンカー公開鍵は、ルート CA 証明書（すなわち、証明書内の公開鍵に対応するプライベート鍵によって署名された“自己署名された”X.509 証明書）として配付されることが多い。本書では、信頼された CA を“トラストアンカー”、その証明書を“トラストアンカー証明書”というが、他の多くの文書では、信頼された CA とその CA 証明書の両方を指すために“トラストアンカー”という用語を使用していることに留意されたい。

トラストアンカー証明書は、多くの場合、アプリケーション内に埋め込まれ、アプリケーションとともに配付される。例えば、新しい Web ブラウザのインストールには、通常、トラストアンカー証明書のエンティティリストのインストール又は交換が含まれる。オペレーティングシステムは、通常、その他の証明書の検証など、様々な理由のためにトラストアンカー証明書を搭載して出荷される。エンティティは、インストール時又は交換時に有効なトラストアンカー証明書のみがインストールされることを確実にするためのソフトウェア配付メカニズムの信頼性に依存する。しかし、場合によっては、他のアプリケーションが Web ブラウザにトラストアンカー証明書をインストールすることがある。

Web ブラウザ内のトラストアンカー証明書は、TLS プロトコルで提示される TLS 証明書の検証など、いくつかの目的で使用される。トラストアンカーとなる CA によって発行されていない証明書を持つ“安全な”ウェブサイトを訪問したエンティティには、1 回限りのセッション又は永続的なセッションのいずれかのために、当該証明書を受け入れる機会が与えられることがある。**依拠当事者は、未知の認証局からの証明書の受け入れに慎重になるべきである。なぜなら、不用意に、実際には信頼できないトラストアンカー証明書を新たに永続的に追加することがないようにするためである。**

> **警告**：ローミング中のユーザは、使用するホストシステム上の全てのソフトウェアを暗黙のうちに信頼していることを認識**すべきである**。“安全なウェブサイト”にアクセスするために、キオスク、図書館、インターネットカフェ、ホテルなどのシステムや、会議主催者が提供するシステムを使用する時、Web ブラウザが使用するトラストアンカー証明書について注意すべきである。ユーザは、ホストシステムにインストールされているトラストアンカー証明書を制御することができないため、ホストシステム管理者の健全な良識ある判断のもとでトラストアンカー証明書が許可されていることに依存する。依拠当事者は、トラストアンカー証明書がソフトウェア配付前にプリインストールされている場合、トラストアンカー証明書の選択に関与できず、さらにその後もどのトラストアンカー証明書がインストールされるかの決定に関与できないかもしれない。ユーザは、ソフトウェア配付メカニズムが悪意のあるコードをインストールすることがないと信頼していることを認識すべきである。この信頼を、特定のアプリケーションのトラストアンカー証明書もカバーするように拡張することは合理的であり、それにより依拠当事者は追加の手続きなしにそのアプリケーションのトラストアンカー証明書を得ることができる。

エンティティ（又はエンティティの代理人）は、インフラストラクチャと安全にやり取りして（例えば、証明書を取得するために）鍵を登録する。これらのインタラクションは、トラストアンカー情報をトラストアンカー証明書の形で提供するために拡張してもよい。これにより、インフラストラクチャがエンティティの鍵を登録しているのとほぼ同じ保証でトラストアンカー証明書を確立することができる。PKI の場合：

1. トラストアンカー証明書の最初の配付は、証明書要求プロセスにおいて、要求するエンティティの公開鍵を登録局（RA）又は CA に提示する際に併せて実施される**べきである**。一般的に、トラストアンカーの公開鍵、関連パラメータ、鍵の使用、及び所有の保証は、自己署名された

X.509 公開鍵証明書（ルート CA 証明書とも呼ばれる）として伝達される。この場合、その証明書は、当該証明書内の公開鍵に対応するプライベート鍵によってデジタル署名されている。自己署名された証明書ではパラメータや所有の保証は伝えられるかもしれないが、トラストアンカー証明書に関連付けられた身元やその他の情報は、自己署名された証明書自体からは検証できない（後述 2 項参照）。

2. 要求するエンティティの公開鍵、及び RA 又は CA の保証を伝達するために使用される信頼されたプロセスは、要求するエンティティに伝達されるトラストアンカー証明書を保護するためにも使用**されなければならない**。要求するエンティティ（又はエンティティの代理人）と直接会う場合には、その時点でトラストアンカー証明書を提供してもよい。エンティティ登録時に秘密値が設定される場合（8.1.1 節参照）、トラストアンカー証明書は、要求するエンティティの証明書と一緒に提供されてもよい。

8.1.5.1.1.2 登録局又は認証局への申請

公開鍵は、認証局（CA）又は登録局（RA）に提供され、CA による後続の認証を受けることができる。このプロセスにおいて、RA 又は CA は、8.1.5.1.1 節に記載されている保証、及び鍵を申請したエンティティ（すなわち、鍵の所有者又は権限を与えられた代理人）から鍵所有者の身元を得な**ければならない**。

一般に、鍵の所有者は、エンティティ登録時に確立された識別子で識別される（8.1.1 節参照）。エンティティ登録時には、鍵の適切な用途が、必要なパラメータとともに特定される。公開鍵の匿名所有が許容される場合、申請者又は登録局は、識別子として使用する仮名を決定する。識別子は、命名機関[94]にとって一意で**なければならない**。

所持証明（POP）は、鍵登録時にプライベート鍵の所有の保証を得るために、CA が一般的に使用するメカニズムである。この場合、証明は、評判の高い鍵ペア所有者によって提供**されなければならない**。所有の保証がなければ、CA は公開鍵を誤ったエンティティに関連付けることが可能となる。

（評判の高い）所有者は、指示された鍵の使用を満たすプライベート鍵を用いて処理を実行することで、POP を提供**すべきである**。例えば、鍵ペアが RSA デジタル署名生成のために意図されている場合、CA は、所有者のプライベート鍵を使用して署名するための情報を提供してもよい。CA が対応する公開鍵を使用して署名を正しく検証できれば、所有者は POP を確立したことになる。ただし、鍵ペアが鍵確立（すなわち、鍵合意又は鍵配送のどちらか）をサポートすることを意図しており、その鍵ペアが FIPS 186[95]で規定されているように生成された場合（SP 800-56A 及び SP 800-56B 参照）、デジタル署名するためにそのプライベート鍵を使用して証明書要求をすることで POP を提供することもできる（ただし、これは好ましい方法ではない）。鍵確立プライベート鍵（すなわち、鍵合意プライベート鍵又は鍵配送プライベート鍵）は、証明書発行後は署名処理を実行するために使用**してはならない**。

エンティティ登録の場合と同様、セキュリティインフラストラクチャの強度は、RA 又は CA に鍵を配付するために使用される方法に依存する。多くの異なる方法があり、それぞれがある範囲のアプリケーションに適している。一般的な方法の例をいくつか挙げる：

---

94 命名機関とは、ドメイン名の割り当てと配付に責任を持ち、ドメイン内で名前が一意であることを保証するエンティティである。命名機関は、.com、.net、.edu などの特定のドメインレベルに限定されることが多い。

95 SP 800-56A では、有限体の Diffie-Hellman 鍵合意のために、事前に定義されたグループの使用も認めている。これらのパラメータを使用した場合、デジタル署名を用いて POP を行うことはできない。

1. 公開鍵及び 8.1.5.1.1 節で特定された情報は、公開鍵所有者又は公開鍵所有者の権限のある代理人（組織、デバイス、プロセスなど）から直接提供される。

2. 公開鍵所有者又は公開鍵所有者の権限を有する代理人の身元が確立され、エンティティ登録時に RA 又は CA が直接その権限を検証する。一意で予測不可能な情報（認証コード又は暗号鍵）は、この時点で RA 又は CA が秘密値として所有者又は権限を与えられた代理人に提供される。8.1.5.1.1 節で特定された情報と公開鍵は、秘密値で保護された通信プロトコルを用いて、RA 又は CA に提供される。8.3.4 節に規定されているように、秘密値は、証明書が正常に生成されたことの確認を受けた後、鍵の所有者又は所有者の代理人によって破棄される**べきである**。RA 又は CA は、監査目的のためにこの秘密値を維持してもよいが、RA 又は CA は身元を証明するために当該秘密値のさらなる使用を認める**べきではない**。

   鍵を登録するための公開鍵所有者の具体的なリストが事前に認可されている場合、所有者が立ち会うことなく識別子が割り当てられることがある。この場合、秘密値を漏えいから保護することが重要であり、その手順は、保護の連鎖が維持されていたことを示すもので**なければならない**。秘密値の寿命は限定される**べきである**が、公開鍵所有者又は所有者の代理人が RA 又は CA を訪問し、鍵を生成し、公開鍵を（秘密値の保護の下で）RA 又は CA に提供できるように**しなければならない**。公開鍵所有者又は所有者の代理人が RA 又は CA を訪問するまでには時間がかかるので、秘密値の寿命は 2～3 週間が妥当であろう。

   公開鍵所有者が事前に認可されていない場合、RA 又は CA は、所有者又は代理人の立会いのもとで識別子を決定**しなければならない**。この場合、鍵ペアがオンサイトで生成され、当該公開鍵が CA 又は RA に提供されるならば、制限時間はより厳しく制限されてもよい。この場合、秘密値の寿命は 24 時間が妥当である。

3. 公開鍵所有者の身元は、公開鍵所有者の前回の身元確認を使用して RA 又は CA で確立される。証明書を受け取るための継続的な認可も検証される。再認証には、新しい公開鍵証明書の要求を、以前に認証されたデジタル署名鍵ペアに“結び付ける”ことが含まれる。例えば、新しい公開鍵証明書の要求は、認証される新しい公開鍵の所有者によって署名される。要求に署名するために使用される署名プライベート鍵は、新しい公開鍵を認証するのと同じ CA によって認証された署名検証公開鍵に対応**すべきである**。要求には、新しい公開鍵と鍵関連情報（鍵の使用方法や鍵のパラメータなど）が含まれる。さらに、CA は、公開鍵の有効性の保証と、所有者が対応するプライベート鍵を所有していることの保証を得**なければならない**。

4. 公開鍵、鍵の使用、パラメータ、有効性保証情報及び所有権の保証が、所有者の身元を主張し、証明書を受け取るための認可とともに、RA 又は CA に提供される。RA 又は CA は、公開鍵所有者の身元及び認可の検証を、別の信頼されたプロセスに委任する（例えば、米国郵政公社による公開鍵所有者への、要求された証明書を含む書留郵便による身元確認）。認証の要求を受け取ると、RA 又は CA は、一意で予測不可能な情報（認証子又は暗号鍵）を生成し、信頼されたプロセス（例えば、米国郵政公社経由で送付された書留郵便）を使用して要求者に送信する。この信頼されたプロセスは、RA 又は CA によって提供された情報を配達する前に、要求者の身元が確認されることを保証する。所有者または所有者の代理人は、この情報を使用して、信頼されたプロセスが成功したことを証明し、RA 又は CA は、その後、証明書を所有者又は所有者の代理人に配送する。8.3.4 節に規定されているように、一意で予測不可能な情報は、証明書が正常に生成されたことの確認を受けた後、鍵の所有者又は代理人によって破棄される**べきである**。（RA 又は CA は、監査目的でこの情報を保持してもよいが、身元を証明するために一意の識別子のさらなる使用を認める**べきではない**。）

5. 証明書の共通名（CN）又はサブジェクト代替名（SAN）に含まれる公開鍵及びドメインネームシステム（DNS）アドレスは、証明書署名要求を介してネットワーク接続により RA に提供

される。RA は、アドレスの DNS レコードを介したチャレンジの実行、DNS アドレスへの HTTP[96]を介したチャレンジの実行、又は DNS アドレスの認可を確認するためのその他の形式の検証を行うことにより、要求者がその DNS アドレスの証明書を要求することを認可されていることを確認する。

RA が関与する場合、要求するエンティティ（例えば、新しい公開鍵の所有者）から全ての情報を受け取った後、RA は、認証のために関連情報を CA に転送する。RA 及び CA は共同して、証明書を発行する前に、公開鍵を使用するアルゴリズムに必要な全ての検証又はその他のチェック（例えば、公開鍵の検証）を行わ**なければならない**。CA は、実行されうる又は実行されたチェック又は検証を（証明書、証明書ポリシー、認証局運用規程などに）示す**べきである**。生成後、証明書は、CA の認証局運用規程に基づいて、手動で又は自動化されたプロトコルを使用して RA へ配付されるか、公開鍵所有者又は所有者の代理人へ配付されるか、証明書リポジトリ（すなわち、ディレクトリ）に配付される。

8.1.5.1.1.3 静的公開鍵の一般的な配付

静的公開鍵は、いくつかの方法で、RA 又は CA 以外のエンティティに配付することができる。

配付方法は以下の通りである：

1. 公開鍵の所有者又は所有者の代理人による公開鍵自体の手動配付（対面での配送、補償つき宅配便などによる）：公開鍵の運用の使用前に、8.1.5.1.1 節に記載されている必須の保証を受領者に提供**しなければならない**。
2. 公開鍵の所有者、所有者の代理人、CA、又は証明書リポジトリ（すなわち、ディレクトリ）による公開鍵証明書の手動配付（対面での配送、受領メールなど）又は自動配付：CA が提供しない 8.1.5.1.1 節に記載されている必須の保証（例えば、公開鍵の検証）は、鍵を運用使用する前に、公開鍵の受領者に提供されるか、又は公開鍵の受領者が行わ**なければならない**。
3. 公開鍵の自動配付（例えば、認証及びコンテンツの完全性を有する通信プロトコルの使用）：公開鍵の運用の使用前に、8.1.5.1.1 節に記載されている必須の保証が、受領エンティティに提供さ**れなければならない**。

8.1.5.1.2 一時的公開鍵の配付

使用される場合、一時的公開鍵は安全な鍵合意プロトコルの一部として配付される。鍵合意プロセス（すなわち、鍵合意スキーム＋プロトコル＋鍵確認＋関連するネゴシエーション＋ローカル処理）は、8.1.5.1.1 節に記載されている保証を受領者に提供**すべきである**。一時的公開鍵の受領者は、鍵合意プロセスの後続のステップでその鍵を使用する前に、SP 800-56A に規定されている当該鍵の有効性の保証を得**なければならない**。

8.1.5.1.3 中央で生成された鍵ペアの配付

静的鍵ペアが中央で生成される場合、その鍵ペアは、FIPS 140 認証暗号モジュール内で生成されるか、又は国家安全保障情報の保護のために米国政府が承認した別のソース源から取得して、鍵ペアの意図する所有者に引き渡さ**れなければならない**。中央の鍵生成施設が加入者のために生成した署名鍵ペア

[96] HyperText Transfer Protocol

では、個々の加入者に対する否認防止の強力なサポートを提供することはできない。したがって、否認防止のサポートが必要な場合には、加入者は自分自身で署名鍵ペアを生成**すべきである**。しかし、中央の鍵生成施設が自組織のために署名鍵ペアを生成し、組織のメンバーに配付する場合、否認防止のサポートは組織レベルで提供されるかもしれない（しかし、個人レベルでは提供されない）。

中央の施設で生成された鍵ペアのプライベート鍵は、その鍵ペアの意図された所有者、又は後続のインストールのために所有者の代理人にのみ配付**されなければならない**。中央で生成されたプライベート鍵の機密性は保護**されなければならず**、配付の手順には、エンティティ登録時に確立された受領者の身元及び認可の認証を含め**なければならない**（8.1.1 節参照）。

鍵ペアは、意図された所有者又は所有者の代理人に、適切な手動の方法（宅配便、郵便、鍵生成施設が指定するその他の方法など）又は安全な自動化された方法（例えば、安全な通信プロトコル）を使用して、配付されてもよい。プライベート鍵は、対称鍵と同様の方法で配付**されなければならない**（8.1.5.2.2 節参照）。鍵ペアの公開鍵の配付については、8.1.5.1.1.3 節で説明する。配付プロセスの間、鍵ペアの各鍵は、その鍵に対して適切な保護を提供**しなければならない**（6.1 節参照）。

鍵ペアを受領した場合、所有者は公開鍵の有効性の保証を得**なければならない**（SP 800-56A、SP 800-56B、及び SP 800-89 参照）。所有者は、鍵ペアの公開鍵とプライベート鍵が正しく関連付けられていることの保証を得**なければならない**（すなわち、鍵配送公開鍵で暗号化された鍵が、対応する鍵配送プライベート鍵で復号できることを確認するなどして、一貫性のあるペアであることを確認する）。

8.1.5.2 対称鍵の生成と配付

データ又はその他の鍵の暗号化と復号、及び MAC の計算に使用される対称鍵（4.2.2 節及び 4.2.3 節参照）は、**承認された**方法で決定され**なければならず**、また 6 節に合致する保護の提供を受け**なければならない**。

対称鍵は、以下のいずれかで**なければならない**：

1. 生成された後、手動（8.1.5.2.2.1 節参照）、公開鍵配送メカニズム（8.1.5.2.2.2 節参照）の使用、又は事前に配付されたか合意された鍵ラッピング鍵の使用（8.1.5.2.2.2 節参照)のいずれかの方法で配付される（8.1.5.2.1 節及び 8.1.5.2.2 節参照）
2. 鍵合意スキームを使用して確立される（すなわち、生成と配付が 1 つのプロセスで行われる）（8.1.5.2.3 節参照）
3. マスター鍵／鍵導出鍵からの導出される（8.2.4 節参照）

8.1.5.2.1 鍵生成

対称鍵は、**承認された**方法（例えば、**承認された**乱数生成器の使用。SP 800-133 参照）で生成されるか、**承認された**鍵導出関数（SP 800-108 参照）を使用してマスター鍵／鍵導出鍵から導出**されなければならない**（8.2.4 節参照）。対称鍵は、鍵合意技術を使用して生成することもできる（8.1.5.2.3 節参照）。この場合、別個の鍵配付プロセス（例えば、SP 800-56A 及び SP 800-56B 参照）は必要ない。

知識分割手続きが使用される場合、鍵は、FIPS 140 暗号モジュールの外部では複数の鍵シェアとしてのみ存在するように**しなければならない**。鍵は、暗号モジュール内で作成された後にモジュールからエクスポートするためにシェアに分割されてもよいし、別々のシェアとして作成されてもよい。各鍵シェアは、鍵値の知識を提供**してはならない**（例えば、各鍵シェアはランダムに生成されたように見えなければならない）。*k* 個のシェアの知識が元の鍵を構築するために必要な場合、いかなる *k-1* 個の鍵シェ

アの知識も、その鍵長以外に、元の鍵に関する情報を提供**してはならない**。注意 : 適切な組合せ関数は、単純な連結では得られない（例えば、64 ビットの鍵シェアを 2 つ連結して 128 ビットの鍵を形成することは認められない）。

全ての鍵及び鍵シェアは、FIPS 140 認証暗号モジュール内で生成されるか、国家安全保障情報の保護のために米国政府が承認した別のソース源から入手**しなければならない**。

8.1.5.2.2 鍵配付

鍵ラッピング鍵（すなわち、鍵暗号化鍵）として、鍵導出に使用されるマスター鍵／鍵導出鍵として、又は通信情報の保護のために 8.1.5.2.1 節に従って生成された鍵は、手動（手動鍵配送）又は自動化された鍵配送プロトコル（自動鍵配送）を使用して配付される。

情報（すなわち、データ又は鍵材料）の保存にのみ使用される鍵は、バックアップのため、又は当該鍵で保護された保存情報へのアクセスを必要とする他の認可エンティティへの配付のためを除いて、配付**してはならない**。

8.1.5.2.2.1 手動鍵配付

手動で配付される（すなわち、自動化された鍵配送プロトコル以外の方法で配付される）鍵及び鍵シェアは、配付プロセス全体を通して保護**されなければならない**。手動で配付する場合、秘密鍵、プライベート鍵及び鍵シェアは、ラッピング（すなわち、完全性保護もできる暗号化）されるか、適切な物理的セキュリティ手順を用いて配付**されなければならない**。

鍵配付に知識分割手続きが使用される場合（すなわち、鍵が鍵シェアとして配付される場合、8.1.5.2.1 節参照）、各鍵シェアは、意図する受領者に個々に配付**されなければならない**。

秘密鍵、プライベート鍵及び鍵シェア（すなわち、秘密鍵材料）の手動配付プロセスは、以下のことを保証**しなければならない**：

1. 鍵材料は、認可されたソース源によって配付される
2. 平文の形で鍵材料を配付する全てのエンティティは、鍵材料を生成するエンティティ及び鍵材料を受け取るエンティティの両方から信頼される
3. 鍵材料は、6 節に従って保護される
4. 鍵材料は、認可された受領者によって受け取られる

暗号化された形で配付される場合、鍵又は鍵シェアは、**承認された**鍵ラッピングスキームと鍵ラッピングにのみ使用される鍵ラッピング鍵を使用して暗号化されるか、又は**承認された**鍵配送スキームと意図する受領者が所有する鍵配送公開鍵を使用して暗号化**されなければならない**。鍵ラッピング鍵又は鍵配送公開鍵は、本推奨に規定されているように配付されなければならない。

物理的セキュリティ手順は、あらゆる形式の手動鍵配付に使用してもよい。しかし、秘密鍵材料が平文形式で配付される場合に、これらの手順は特に重要となる。上記の保証に加えて、配付プロセスの説明責任と監査が行われる**べきである**（9.3 節及び 9.4 節参照）。

8.1.5.2.2.2 自動鍵配付／鍵配送／鍵ラッピング

自動鍵配付は、通信チャネル（例えば、インターネット）を介して、秘密鍵、プライベート鍵、及び鍵シェアを配付するために使用されてもよい。そのためには、以下のような鍵ラッピング鍵（すなわち、鍵暗号化鍵）や鍵配送公開鍵の配付／確立が必要となる：

1. 鍵ラッピング鍵は、8.1.5.2.1 節及び 8.1.5.2.2 節に従って生成・配付されるか、又は 8.1.5.2.3 節に規定されている鍵確立スキームを使用して確立**されなければならない**。
2. 鍵配送公開鍵は、8.1.5.1 節に規定されるように生成し、配付**されなければならない**。

承認された鍵ラッピング又は公開鍵配送スキームのみを使用**しなければならない**。**承認された**スキームは、以下のことを保証する：

a. 対称鍵ラッピング方式の場合：鍵ラッピング鍵及び配付された鍵材料が、開示又は変更されていないこと。機密性と完全性の両方の保護を提供する**承認された**鍵ラッピング方法は、SP 800-38F[97]に規定されている。

b. 非対称鍵配送方式の場合：鍵配送プライベート鍵及び配付された鍵材料は開示又は変更されておらず、鍵配送プライベート鍵と鍵配送公開鍵の間の正しい関連付けが維持されていること。非対称技術を使用した**承認された**鍵配送方式については、SP 800-56A 及び SP 800-56B で説明する。

c. 鍵材料が、6 節に従って保護されていること。

さらに、**承認された**スキームは、関連する鍵確立プロトコルとともに、以下の保証を提供**すべきである**：

d. 鍵配付プロセスの各エンティティは、他のエンティティに関連付けられた識別子を知っていること

e. 鍵材料が、鍵配付プロセスに関与するエンティティに正しく関連付けられていること

f. 鍵材料が正しく受領されていること（例えば、鍵確認方法を使って）

8.1.5.2.3 鍵合意

鍵合意は、通信環境において鍵材料を確立するために使用され、実際に鍵材料を送信することなく、通信中の全てのエンティティ（最も一般的には 2 つのエンティティのみ）が提供した情報を用いて行われる。**承認された**鍵合意スキームのみを使用**しなければならない**。非対称技術を用いた**承認された**鍵合意スキームは、SP 800-56A 及び SP 800-56B に規定されている。これらの鍵合意スキームは、非対称鍵ペアを使用して共有秘密を計算し、それを使用して対称鍵や他の鍵材料（例えば、IV）を導出する。

鍵合意スキームは、静的非対称鍵ペア又は一時的非対称鍵ペアのどちらか、又はその両方を使用する。非対称鍵ペアは、8.1.5.1 節で説明されているように生成され、配付**されるべきである**。鍵合意スキームを使用して導出された鍵情報は、6 節で規定されるように保護**されなければならない**。

鍵合意スキームとそれに関連する鍵確立プロトコルは、以下の保証を提供**すべきである**：

[97] SP 800-38F, *Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping.*

1. 鍵確立プロトコルに関与するエンティティの識別子が、それらのエンティティに正しく関連付けられていること。識別子のエンティティへの関連付けの保証は、鍵合意スキームによって達成される場合と、鍵合意が実行されるプロトコルによって達成される場合がある。識別子は、例えばエンティティの出生証明書に記載されている識別子などではなく、“擬似識別子（仮名）”であってもよいことに注意されたい。

   一般的なケースでは、鍵確立プロトコルに関与する各エンティティに識別子が関連付けられ、各エンティティは、他の全てのエンティティを適切な識別子で関連付けることができなければならない。ウェブサイト上での公開情報の安全な配付のような特別なケースでは、識別子との関連付けは、エンティティの一部分のみ（例えば、サーバのみ）に対して必要とされる場合がある。

2. 鍵合意スキームで使用される鍵が、鍵確立プロセスに関与するエンティティに正しく関連付けられていること。

3. 導出された鍵材料が正しいこと（例えば、鍵確認方法を使って）。

鍵合意及びその有効化プロトコルを通じて導出された鍵材料は、上記の 3 つの保証が得られるまでは、情報の保護及び送信のために使用**してはならない**。

8.1.5.3 その他の鍵材料の生成及び配付

鍵は、他の鍵材料と一緒に生成されたり、使用されたりすることが多い。このような鍵材料は、6.2 節に従って保護され**なければならない**。表 6 は、鍵以外の鍵材料に必要な保護のタイプを規定している。

8.1.5.3.1 ドメインパラメータ

ドメインパラメータは、鍵ペアの生成、デジタル署名の計算、又は鍵の確立のための公開鍵アルゴリズムで使用される。一般的に、ドメインパラメータの生成頻度は低く、あるエンティティのコミュニティでは長期間使用される。ドメインパラメータは、関連する公開鍵と同様の方法で配付される場合もあれば、他のアクセス可能な場所で利用できるようにする場合もある。ドメインパラメータの有効性の保証は、使用前に、パラメータを保証する信頼されたエンティティ（例えば、CA）又はパラメータを使用するエンティティのいずれかによって、得られ**なければならない**。ドメインパラメータ有効性の保証は、SP 800-56A（鍵確立スキーム用）及び SP 800-89（デジタル署名用）に記載されている。CA の認証局運用規程又は組織のセキュリティ計画の記載により、この保証が得られる**べきである**。

8.1.5.3.2 初期ベクトル

初期ベクトル（IV）は、暗号化と復号、認証、又はその両方のために、対称鍵アルゴリズムのいくつかの暗号利用モードによって使用される。IV の生成と使用の基準は、SP 800-38 シリーズの出版物に規定されている。IV は、6.1.2 節で規定されているように保護され**なければならない**。IV は、関連する鍵と同じ方法で配付されてもよいし、暗号メカニズムの一部として IV を使用する情報と一緒に配付されてもよい。

8.1.5.3.3 共有秘密

共有秘密は、非対称鍵合意スキームの実行中に計算され、その後、鍵材料を導出するために使用される。共有秘密は、適切な鍵合意スキーム（SP 800-56A 及び SP 800-56B 参照）で指定された通りに生成されるが、配付したり、鍵材料として直接使用**されてはならない**。

8.1.5.3.4 RBG シード

乱数ビット生成器（RBG）は、予測不可能なビット列を出力するデバイス又はアルゴリズムである。RBG はしばしば乱数生成器と呼ばれる。**承認された** RBG は、SP 800-90 シリーズの出版物に規定されている。RBG は、RBG の初期化に使用されるシードを生成するために使用される真正乱数ビットの初期値に依存する。これらのシードは秘密にされなければならない。初期化された RBG は、予測不可能性を必要とする鍵やその他の値を生成するために使用されることが多い。シード自体は，RBG への入力としての目的以外で使用**してはならない**。

8.1.5.3.5 その他の公開情報及び秘密情報

公開情報と秘密情報は、RBG のシーディング中（8.1.5.3.4 節参照）、又は鍵材料の生成又は確立中（SP 800-56A、SP 800-56B、及び SP 800-108[98]参照）に使用されてもよい。公開情報は配付してもよいが、秘密情報は配付中にプライベート鍵又は秘密鍵と同じ方法で保護され**なければならない**。

8.1.5.3.6 中間結果

中間結果は、暗号アルゴリズムを用いた計算中に発生する。これらの結果は、鍵材料として、又は鍵材料と一緒に配付**してはならない**。

8.1.5.3.7 乱数ビット／乱数

乱数ビット（又は乱数）は、多くの目的で使用される。その目的には、鍵やノンスの生成、通信プロトコルの実行中のチャレンジの発行が含まれる。乱数ビットは配付されてもよいが、機密性保護が必要かどうかは、乱数ビットが使用されるコンテキストに依存する。

8.1.5.3.8 パスワード

パスワードは、ID 認証、認可、及び場合によっては鍵材料の導出に使用される（SP 800-132 参照）。パスワードは配付されてもよいが、配付中の保護は、その使用で必要な保護と一致して**いなければならない**。例えば、データを保護する際に 128 ビットセキュリティ強度を提供するために使用する暗号鍵に対してアクセスするために、パスワードが使用される場合、パスワードにも少なくとも 128 ビットの保護を提供する必要がある。不適切に選択されたパスワードだけでは、鍵へのアクセスに必要な量の保護を提供できず、プロセスの弱点となる可能性があることに注意されたい（すなわち、パスワードを推測する方が、パスワードに使用されている暗号保護を“破る”ことを試みるよりも、はるかに容易である

[98] SP 800-108, *Recommendation for Key Derivation Using Pseudorandom Functions.*

かもしれない)。保護する鍵に必要な量の保護を提供するパスワードを選択することは、ユーザと組織の責任である。

8.1.6 鍵登録機能

鍵登録により、鍵材料と特定のエンティティに関連付けられた情報が結び付けられる。登録される鍵には、非対称鍵ペアの公開鍵や、エンティティをシステムに追加するブートストラップで使用される対称鍵が含まれる。通常、通信中に（例えば、鍵合意スキームや鍵導出関数を使用して）生成された鍵は登録されない。登録時に提供される情報は、典型的には、鍵材料に関連付けられたエンティティの識別子（所有者)、及び鍵材料の意図された用途（署名鍵、データ暗号化鍵など）を含む。追加情報には、認可情報を含んだり、信頼度のレベルを指定したり、鍵が格納されている場所（物理的な場所、インターネットアドレス、デバイス識別情報など）を示したりすることができる。結び付けは、エンティティの身元がシステムポリシーと一致する手段で認証された後に実行される（8.1.1 節参照)。結び付けは、鍵材料を正しいアプリケーションで正しいエンティティが使用していることを、コミュニティ全体に保証する。結び付けは、鍵材料とエンティティとの間に強い関連性を暗号学的に生成することが多い。信頼された第三者が結び付けを実行する。信頼された第三者の例としては、Kerberos レルムサーバや PKI 認証局（CA）などがある。信頼された第三者が発行する識別子は、その当事者に固有のもので**なければならない**。

Kerberos レルムサーバが結び付きを実行する場合、対称鍵が対応するメタデータとともにサーバに格納される。この場合、登録された鍵材料は、安全なストレージに維持される（すなわち、鍵には機密性と完全性の保護が提供される)。

CA が結び付きを実行する場合、公開鍵及び関連情報（例えば、ドメインパラメータ及び一部のメタデータ（しばしば属性と呼ばれる))は、CA によってデジタル署名された公開鍵証明書の中に格納される。この場合、登録された鍵材料は、公開されてもよい。

CA が公開鍵に対して証明書を提供する場合、その公開鍵が、公開鍵の所有者と称する者が知っているプライベート鍵と関連することを検証し確認**しなければならない**。このような手続きにより所持の保証が得られ、これは所持証明（POP）とも呼ばれる。所有の保証を得るために POP を使用する場合は、保証は、8.1.5.1.1.2 節に規定されている通りに行わ**なければならない**。

登録された鍵及び証明書は、棚卸リストに含め**なければならない**（9.2 節参照)。

8.2 運用フェーズ

鍵の暗号利用期間中に使用される鍵材料は、必要に応じてアクセスできるように保存されることが多い。保管中は、鍵材料及びその他の鍵情報を、6.2.2 節に規定されているように保護**しなければならない**。通常の使用時には、その情報を使用するデバイス又はモジュール内、もしくはすぐにアクセス可能な記憶媒体に、鍵情報は格納される。鍵材料は、運用上の必要な場合にストレージから取得され、デバイス又はモジュール内のアクティブメモリには置かれない。

鍵材料が、その暗号利用期間中に通常運用中のストレージから使用できなくなった場合（例えば、当該材料の紛失又は破損のため）に運用継続性を提供するために、鍵材料が復元可能であることが必要となる場合がある。システム運用の分析により、鍵情報を復元可能にする必要があることが示された場合、鍵情報のバックアップ（8.2.2.1 節参照）又はアーカイブ（8.3.1 節参照）のいずれかを行わ**なければならない**、もしくはシステムは鍵材料の再構築（例えば、再導出）を可能にするように設計**されなければ**

ならない。アーカイブ又はバックアップから鍵材料を回収又は再構築することは、一般的に鍵復元として知られている（8.2.2.2 節参照）。

鍵の暗号利用期間の終了時、処理を継続するためには、古い鍵に代わる新しい鍵が利用可能である必要がある。これは、鍵再設定（8.2.3.1 節参照）又は鍵導出（8.2.4 節参照）によって実現できる。鍵は、露出のリスクを減らすために、当該鍵が不要になったらすぐに破棄される**べきである**。破棄される場合、その鍵は 8.3.4 節に従って破棄**されなければならない**。

8.2.1 通常運用時のストレージ機能

鍵管理の目的の一つは、標準的な暗号目的のために鍵材料の運用可能性を高めることである。通常、鍵は、その鍵の暗号利用期間（すなわち有効期限）が終了するまで運用可能なままである。通常運用時の使用では、鍵材料はデバイス又はモジュール内（例えば、RAM 内）、もしくはすぐにアクセス可能な記憶媒体（例えば、ローカルハードディスク上）で利用可能である。

8.2.1.1 暗号モジュールのストレージ

鍵材料は、情報の暗号保護を追加、検査又は削除する暗号モジュールに格納されてもよい。鍵材料のストレージは、6.2.2 節及び FIPS 140 に準拠**しなければならない**。

8.2.1.2 即時アクセス可能な記憶媒体

鍵材料は、鍵の暗号利用期間中、通常の暗号処理のために、すぐにアクセス可能な記憶媒体（例えば、ローカルハードドライブ）に保管しておく必要があるかもしれない。6.2.2 節の保存要件は、この鍵材料に適用**されなければならない**。

8.2.2 運用継続性の機能

鍵材料は、ハードウェアの損傷、プログラムやデータファイルの破損や紛失、システムポリシー又は構成の変更によって、喪失したり使用不能になったりすることがある。継続性を維持するために、ユーザや管理者がバックアップストレージから鍵材料を復元できるようにすることが必要な場合が多い。しかし、鍵材料のバックアップなしに（例えば、鍵再設定によって）処理が継続できる場合、又は鍵材料の保存なしに当該鍵材料の復元又は再構築ができる場合、鍵材料を保存しない方が望ましい場合がある。なぜなら、鍵材料又は他の暗号関連情報の危殆化の可能性を低減するためである。

鍵材料の危殆化は、運用の継続性に影響を及ぼす（8.4 節参照）。鍵材料が危殆化した場合、運用の継続には、影響を受けた鍵材料の評価後に、全く新しい鍵材料（8.1.5 節参照）を確立する必要がある。つまり、影響を受けた全ての鍵材料を交換する必要がある。

8.2.2.1 バックアップストレージ

独立した安全なストレージ媒体への鍵材料のバックアップは、鍵の復元のためのソースを提供する（8.2.2.2 節参照）。バックアップストレージは、鍵の暗号利用期間中、通常運用中のストレージ（すなわち、暗号モジュール又はすぐにアクセス可能な記憶媒体。8.2.1.1 節参照）内で利用中でもある鍵材料のコピーを保存するために使用される。全ての鍵をバックアップする必要はない。6.2.2 節の保存要件

は、バックアップされた鍵材料に適用される。表 7 及び表 8 は、各タイプの鍵材料及びその他の関連情報のバックアップについてのガイダンスを提供する。“OK”は、保存が許容されているが、必ずしも必要ではないことを示している。バックアップに対する最終判断は、その鍵材料が使用されるアプリケーションに応じて行う**べきである**。各タイプの鍵及びその他の鍵情報のバックアップについての詳細は、付録 B.3 参照のこと。

バックアップストレージに保存されている鍵材料は、少なくとも、通常運用上の使用のために同じ鍵材料が維持されている間は、ストレージに維持される**べきである**（8.2.1 節参照）。通常運用上の使用に必要とされなくなった場合、鍵材料及びその他の関連情報は、バックアップストレージから削除される**べきである**。バックアップストレージから削除された場合、バックアップストレージ内の情報の全ての痕跡は、8.3.4 節に従って破棄**されなければならない**。

バックアップ及び復元に関する説明は、［ITL Bulletin］[99]に記載されている。

表 7：鍵のバックアップ

鍵タイプ	バックアップ？
署名プライベート鍵	No（一般的には）。否認防止のサポートに疑問が生じる。しかしながら、認証局の署名プライベート鍵など、いくつかのケースではバックアップが正当化されることがあるかもしれない。必要な場合、バックアップされた鍵は、全て所有者の管理下に保管されなければならない。
署名検証公開鍵	OK。他の場所で利用可能な公開鍵証明書の中に存在すれば十分である。
認証対称鍵	OK
認証プライベート鍵	OK。アプリケーションによって要求された場合。
認証公開鍵	OK。アプリケーションによって要求された場合。
データ暗号化対称鍵	OK
鍵ラッピング対称鍵	OK
乱数生成鍵	アプリケーションによっては、必須ではなく、望ましくない場合もある。
マスター対称鍵／鍵導出対称鍵	OK
鍵配送プライベート鍵	OK
鍵配送公開鍵	OK。他の場所で利用可能な公開鍵証明書の中に存在すれば十分である。
鍵合意対称鍵	OK
静的鍵合意プライベート鍵	OK

[99] ITL Bulletin: *Techniques for System and Data Recovery*.

静的鍵合意公開鍵	OK。他の場所で利用可能な公開鍵証明書の中に存在すれば十分である。
一時的鍵合意プライベート鍵	No
一時的鍵合意公開鍵	OK
認可対称鍵	OK
認可プライベート鍵	OK
認可公開鍵	OK。他の場所で利用可能な公開鍵証明書の中に存在すれば十分である。

表 8：その他の関連情報のバックアップ

鍵材料のタイプ	バックアップ？
ドメインパラメータ	OK
初期ベクトル	OK。必要に応じて。
共有秘密	No
RBG シード	No
その他の公開情報	OK
その他の秘密情報	OK
中間結果	No
鍵コントロール情報／メタデータ（ID、目的など）	OK
乱数	アプリケーションや乱数の用途に依存する。
パスワード	鍵導出のため、又はパスワードの再利用検知のために使用する場合は OK、それ以外の場合は No。
監査情報	OK

8.2.2.2 鍵復元機能

アクティブメモリにある、又は通常運用中のストレージに保存されている鍵材料は、時々紛失したり、破損したりすることがある（システムクラッシュ、電力変動などのため）。鍵材料の中には、処理を継続するために必要なものがあり、簡単には交換できないものもある。後になって復旧できるように、どの鍵材料を保存する必要があるかを評価する必要がある。

鍵の復元が必要かどうかの判断は、ケースバイケースで行われる**べきである**。決定は以下に基づいて行われる**べきである**：

1. 鍵タイプ（署名プライベート鍵、データ暗号化対称鍵など）

2. 鍵が使用されるアプリケーション（対話型通信、ファイル保存など）
3. 鍵を“所有”しているのが、ローカルエンティティ（例えば、プライベート鍵）なのか、他のエンティティ（例えば、他のエンティティの公開鍵）なのか、又は共有されているのか（例えば、2 つのエンティティによって共有されるデータ暗号化対称鍵）
4. 通信におけるエンティティの役割（送信者か、受信者かなど）
5. 鍵が使用されるアルゴリズム又は計算（例えば、鍵が復元されたとして、エンティティは所定の計算を実行するために必要な情報を持っているか）[100]

鍵復元の可否を決定する際の判断要素は、慎重に評価される**べきである**。運用の継続性と、鍵材料のコントロールが失われた場合に鍵材料とそれが保護する情報が漏えいするかもしれないリスクとのトレードオフに関係している。鍵を復元する必要があると判断され、かつ当該鍵がまだアクティブである（例えば、当該鍵の暗号利用期間が満了しておらず、当該鍵が危殆化していない）場合、当該鍵で保護されているデータの漏えいを制限するために、鍵を交換する必要があるかもしれない（8.2.3 節参照）。

鍵の復元に関連する課題と、異なるタイプの暗号材料が復元可能である必要があるかどうかの説明は、付録 B に記載されている。

8.2.3 鍵変更機能

鍵の変更とは、元の鍵と同じ機能を果たす別の鍵に鍵を交換することである：

1. 鍵が危殆化している可能性がある
2. 鍵の暗号利用期間が満了に近づいている
3. 与えられた鍵で保護されるデータ量を制限することが望ましい

8.2.3.1 鍵再設定

新しい鍵が、古い鍵の“値”から完全に独立した方法で生成される場合、そのプロセスは鍵再設定と呼ばれる。交換は、8.1.5 節で説明されている鍵確立方法のいずれかを用いて行われ**なければならない**。鍵再設定は、鍵が危殆化した場合（鍵再設定スキーム自体が危殆化していないことを条件とする）、又は暗号利用期間が満了した場合又は満了に近づいている場合に使用される。

8.2.3.2 鍵更新機能

新しい鍵の“値”が古い鍵の値に依存している場合、このプロセスは鍵更新と呼ばれる（すなわち、現在の鍵を修正して新しい鍵を生成する）。鍵更新は、導出された鍵にその導出に使用された鍵を交換するという鍵導出（8.2.4 節参照）の特殊なケースである。例えば、K_1 が暗号化鍵として使用されていると仮定する。K_1 を置き換える必要がある場合、K_1 は K_2 を導出するために使用される。その後、K_2 は、K_2 から導出された K_3 に置き換えられるまで、新しい暗号化鍵として使用される。

---

[100] これは、いくつかの鍵確立スキームで鍵確立処理を行う際に発生する可能性がある（SP 800-56A 及び SP 800-56B を参照）。

鍵更新は、敵対者が導出鍵の連鎖の中の鍵を入手し、かつ使用された更新プロセスを知っている場合、セキュリティ上の問題を引き起こす可能性がある。つまり、漏えいした鍵に続く鍵を簡単に特定することができる。

連邦政府のアプリケーションでは、鍵更新を使用**してはならない**（SP 800-152 も参照のこと）。

8.2.4 鍵導出方法

暗号鍵は、秘密値から導出されることがある。秘密値は、他の情報とともに、要求される鍵を出力する鍵導出方法（例えば、鍵導出関数）に入力される。導出方法は、導出した鍵から秘密値が特定できないように非可逆的（すなわち、一方向性関数）で**なければならない**。さらに、他の導出鍵から別の導出鍵を決定することができないように**しなければならない**。導出鍵の強度は、導出アルゴリズム及びその鍵の導出元である秘密値の強度以上にはならないことに留意すべきである。

以下では、一般的に使用される 3 つの鍵導出ケースについて説明する：

1. *2 つの当事者が共通の共有秘密から共通の鍵を導出する*。このアプローチは、SP 800-56A 及び SP 800-56B で規定されている鍵合意技術で使用される。このプロセスのセキュリティは、共有秘密及び使用される具体的な鍵導出方法のセキュリティに依存する。共有秘密が既知であれば、導出鍵を決定することができる。この目的のために、SP 800-56C[101]で規定又は許可された鍵導出方法を使用**しなければならない**。これらの導出鍵は、ランダムに生成された鍵と同様の機密性、ID 認証及びソース認証のサービスを提供するために使用してもよい。導出鍵のセキュリティ強度は、共有秘密の生成に使用されるスキームと鍵ペアによって決まる。
2. *鍵導出鍵（マスター鍵）から導出した鍵*。これは、秘密の鍵導出鍵とその他の既知の秘密情報や公開情報を、鍵を生成する関数への入力として使用することで実現されることが多い。この目的のために、SP 800-108 で規定された鍵導出関数の 1 つを使用**しなければならない**。このプロセスのセキュリティは、鍵導出鍵及び鍵導出方法のセキュリティに依存する。鍵導出鍵が敵対者に知られている場合、敵対者は任意の導出鍵を生成することができる。したがって、鍵導出鍵から導出された鍵は、鍵導出鍵自体のセキュリティと同程度にしかならない。鍵導出鍵が秘密にされている限り、導出鍵はランダムに生成された鍵と同じように使用できる。
3. *パスワードから導出された鍵*。その性質上、ユーザが生成したパスワードは、暗号鍵に必要とされるよりもランダム性が低い（すなわち、エントロピーが低い）。つまり、鍵を導出するために使用される可能性のあるパスワードの数は、与えられた鍵長に対して可能な鍵の数よりも著しく少ない。パスワードでよくある全数探索の困難性を高めるために、鍵導出関数を何度も繰り返す。鍵は、パスワードとその他の既知の秘密情報や公開情報を、鍵導出関数の入力として使用して生成される。導出鍵のセキュリティは、パスワード及び鍵生成プロセスのセキュリティに依存する。パスワードが既知であるか又は推測可能であれば、対応する導出鍵を生成することができる。したがって、このようにして導出された鍵は、ランダムに生成された鍵又は共有秘密や秘密の鍵導出鍵から導出された鍵よりもセキュリティが低い可能性が高い。ストレージアプリケーションでは、SP 800-132 に規定されている鍵導出関数の 1 つを使用して、パスワードから鍵を導出**しなければならない**。ストレージ以外のアプリケーションでは、このような方法で導出された鍵は、ID 認証及びソース認証の目的にのみ使用され、一般的な暗号化に使用**してはならない**。

[101] SP 800-56C, *Recommendation for Key-Derivation Methods in Key-Establishment Schemes*.

8.3 運用後フェーズ

運用後フェーズでは、鍵材料は運用上使用されなくなるが、当該鍵材料へのアクセスは可能なままである。

8.3.1 鍵アーカイブ及び鍵復元機能

鍵アーカイブとは、鍵及びその関連情報（すなわち、鍵情報）を格納したリポジトリのことであり、鍵の暗号利用期間を超えて復元するためのものである。全ての鍵をアーカイブする必要はない。組織のセキュリティ計画で、鍵アーカイブについて議論**すべきである**（SP 800-57 パート 2 参照）。

鍵アーカイブは、6.2.2 節で規定されているように、アーカイブ内の各鍵及びその他の全ての関連情報に対して、適切な保護を提供し続け**なければならない**。アーカイブは、認可されたエンティティのみへのアクセスを制限するための強力なアクセス制御メカニズムを必要とする。鍵情報がアーカイブに入力される際には、入力日を特定できるようにタイムスタンプが付けられることが多い。この日付自体は、検知されずに変更できないように、暗号保護されている場合がある。

鍵を復元可能にしなければならない場合（例えば、暗号利用期間の終了後に）、当該鍵をアーカイブ**しなければならない**か、又はアーカイブされた情報から当該鍵を再構築（例えば、再導出）できるようにシステムを設計**しなければならない**。アーカイブストレージから、又は再構築によって鍵を取り出すことは、一般に鍵復元と呼ばれる。アーカイブは、信頼できる当事者（鍵に関連する組織、信頼される第三者など）によって維持され**なければならない**。

アーカイブされた鍵情報は、運用データとは別に保管**しなければならない**。また、アーカイブされた鍵情報の複数のコピーを物理的に別々の場所に提供**すべきである**（すなわち、鍵アーカイブはバックアップされる**べきである**）。アーカイブされた鍵の下で暗号化された重要な情報については、アーカイブされた鍵をバックアップし、これらのアーカイブされた鍵の複数のコピーを別々の場所に保管することが必要な場合がある。

アーカイブする場合、鍵の暗号利用期間が終了する前にアーカイブ**すべきである**。例えば、鍵が活性化された時点でアーカイブするのが賢明である。不要になった場合、鍵は 8.3.4 節に従って破棄**されなければならない**。

アーカイブされた鍵情報の機密性は、アーカイブ暗号化鍵（アーカイブされた鍵情報の暗号化のためだけに使用される 1 つ以上の暗号化鍵）、アーカイブされた別の鍵、又はアーカイブされた鍵から導出された鍵によって提供される。アーカイブ暗号化鍵が使用されるアルゴリズムは、暗号化された情報の完全性保護を提供する場合もあることに注意されたい。

アーカイブ暗号化鍵及びその関連アルゴリズムが暗号化情報の完全性保護も提供しない場合、完全性保護は、別個のアーカイブ完全性鍵（すなわち、アーカイブのためだけに使用される 1 つ以上の認証鍵又はデジタル署名鍵）、又はアーカイブされた別の鍵によって提供され**なければならない**。

アーカイブされた鍵情報の機密性及び完全性の保護が別個のプロセスを用いて提供される場合、アーカイブ暗号化鍵及びアーカイブ完全性鍵は、互いに異なる（例えば、独立して生成される）もので**なければならず**、かつそれらの鍵タイプ（6.1.1 節参照）と同じ方法で保護され**なければならない**。これら 2 つのサービスは、単一の暗号アルゴリズム処理と一つの鍵を使用する認証済み暗号化を使用して提供されることも可能であることに留意されたい。

表 9 及び表 10 は、鍵及びその他の暗号関連情報をアーカイブすることの適切性を示している。2 列目（アーカイブ？）の“OK”は、アーカイブが許容されるが、必ずしも必要ではないことを示す。3 列

目（保持期間）は、鍵をアーカイブに保持**すべき**最小期間を示す。アーカイブストレージにおける鍵材料の保管に関する追加アドバイスは、付録 B.3 に記載されている。

表 9：鍵のアーカイブ

鍵タイプ	アーカイブ？	保持期間（最低）
署名プライベート鍵	No	
署名検証公開鍵	OK	関連するプライベート鍵で署名されたデータを検証する必要がなくなるまで
認証対称鍵	OK	データや ID の認証が不要になるまで
認証プライベート鍵	No	
認証公開鍵	OK	
データ暗号化対称鍵	OK	当該鍵で暗号化されたデータを復号する必要がなくなるまで
鍵ラッピング対称鍵	OK	当該鍵で暗号化された鍵を復号する必要がなくなるまで
乱数生成対称鍵	No	
マスター対称鍵／鍵導出対称鍵	アーカイブされたデータ用の他の鍵を導出する必要がある場合は、OK	他の鍵を導出する必要がなくなるまで
鍵配送プライベート鍵	OK	当該鍵で暗号化された鍵を復号する必要がなくなるまで
鍵配送公開鍵	OK	使用期間後は実際に使用しない
鍵合意対称鍵	OK	合意された鍵を決定するのに有用でなくなるまで
静的鍵合意プライベート鍵	OK	合意された鍵を決定するのに有用でなくなるまで
静的鍵合意公開鍵	OK	鍵材料の再構築が不要になるまで
一時的鍵合意プライベート鍵	No	
一時的鍵合意公開鍵	OK	合意された鍵を決定するのに有用でなくなるまで
認可対称鍵	No	
認可プライベート鍵	No	
認可公開鍵	OK	認証プライベート鍵の使用期間後は実際に使用しない

表 10：その他関連情報のアーカイブ

鍵タイプ	アーカイブ？	保持期間（最低）
ドメインパラメータ	OK	ドメインパラメータを使用した全ての鍵材料、署名、及び署名データがアーカイブから削除されるまで
初期ベクトル	OK。通常は保護情報と一緒に保存される	保護データを処理する必要がなくなるまで
共有秘密	No	
RBG シード	No	
その他の公開情報	OK	公開情報を利用したデータ処理が不要になるまで
その他の秘密情報	OK	秘密情報を利用したデータ処理が不要になるまで
中間結果	No	
鍵コントロール情報／メタデータ（ID、目的など）	OK	関連する鍵がアーカイブから削除されるまで
乱数		乱数のアプリケーションや使用方法に依存する
パスワード	鍵導出のため、又はパスワードの再利用検知のために使用する場合は OK、それ以外の場合は No。	鍵の（再）導出やパスワードの再利用検出をする必要がなくなるまで
監査情報	OK	不要になるまで

アーカイブされた鍵情報の復元は、他のアーカイブデータ上の暗号保護を削除（例えば、復号）又はチェック（例えば、デジタル署名や MAC の検証）するために必要とされる場合がある。復元した鍵は、当該鍵の暗号利用期間（または作成者利用期間）が経過している場合には、暗号保護の適用に使用**してはならない**。鍵復元プロセスでは、必要な暗号処理を実行するために、アーカイブストレージから目的の鍵材料を取り出すか、又は再構築する。この処理が完了した直後に、鍵材料は、当該暗号処理[102]から消去され**なければならない**（すなわち、通常の運用アクティビティに使用されてはならない）。ただし、その鍵は、必要とされる限り、アーカイブ（8.3.4 節参照）に保持されなければならない。鍵復元の課題に関する更なるアドバイスは、付録 B に記載されている。

[102] 例えば、アーカイブされた対称鍵は、単一のメッセージやファイルを復号するために復元されることも、複数のメッセージやファイルを復号するために使用されることもあるが、これらは全て、その鍵の作成者使用期間中にその鍵を使用して暗号化されたものである。

8.3.2 エンティティ登録解除機能

エンティティの登録解除機能は、セキュリティドメインに参加するエンティティの認可を削除する。エンティティがセキュリティドメインのメンバーでなくなった場合、当該エンティティは登録解除され**なければならない**。登録解除は、他のエンティティが、登録解除されたエンティティの鍵材料（例えば、登録解除されたエンティティと共有される対称鍵）に依存したり、使用したりすることを防ぐことを目的としている。

エンティティ及びそのエンティティに関連付けられた全ての記録には、エンティティがセキュリティドメインのメンバーではなくなったことを示すマークを付け**なければならない**が、その記録を削除**すべきではない**（鍵自体が破棄されていても）。混乱と回避できないヒューマンエラーを減らすために、登録解除されたエンティティに関連付けられた識別情報は、（少なくとも一定期間は）再利用**すべきではない**。例えば、“ジョン・ウィルソン”が退職して金曜日に登録解除された場合、翌週の月曜日に雇用される息子の“ジョン・ウィルソン”に割り当てられる識別情報は異なるように**すべきである**。

8.3.3 鍵登録解除機能

登録された鍵材料は、鍵所有者の身元、所有者情報（例えば、電子メールアドレス）、役割、又は認可情報に関連付けられている場合がある。鍵材料が不要になった場合、又は関連情報が無効になった場合、当該鍵材料は、適切な信頼された第三者により登録解除**されるべきである**（すなわち、鍵材料とその関連情報の全ての記録には、当該鍵がもはや使用されていないことを示すためのマークが付け**られるべきである**）。一般に、これを行うものは、鍵を登録した信頼される第三者である（8.1.6 節参照）。

鍵材料は、エンティティに関連する情報が変更された場合、登録解除**されるべきである**。例えば、エンティティの電子メールアドレスが公開鍵に関連付けられており、エンティティのアドレスが変更された場合、関連する情報が無効になったことを示すために、鍵材料は登録解除**すべきである**。鍵の危殆化の場合とは異なり、鍵の暗号利用期間が満了していない場合には、エンティティは、当該エンティティの情報を変更した後、エンティティ登録プロセス（8.1.1 節参照）を通じて安全に公開鍵を再登録することができる。

登録された暗号鍵が危殆化した場合、その鍵及び関連する鍵材料は全て登録解除**しなければならない**。危殆化した鍵が鍵ペアのプライベート鍵である場合、公開鍵も失効**させなければならない**（8.3.5 節参照）。登録解除された鍵は、再登録**してはならない**。

鍵ペアに関連する登録情報が変更されてはいるが、プライベート鍵が危殆化していない場合、公開鍵は適切な理由コード（8.3.5 節参照）を付して取り消す**べきである**。この場合、当該鍵は、暗号利用期間が満了していなければ、再登録できる。

8.3.4 鍵破棄機能

暗号鍵の複製を作成する場合、最終的な破棄に備えて注意を払うべきである（例えば、共有者の身元を鍵のメタデータに記録することができる）。プライベート鍵又は秘密鍵（対称鍵）の全てのコピーは、危殆化のリスクを最小化するために、（例えば、アーカイブ又は再構築アクティビティの）必要がなくなり次第、すぐに破棄され**なければならない**。秘密鍵及びプライベート鍵は、当該鍵の痕跡を全て除去す

る方法で破棄して、物理的又は電子的な手段では復元できないように**しなければならない**[103]。公開鍵は、必要に応じて保持又は破棄することができる。

8.3.5 鍵失効機能

鍵の失効は、1）確立された鍵の暗号利用期間が終了する前にその鍵の使用の認可を終了する必要がある場合、又は 2）作成者使用期間が満了した鍵が危殆化した場合、に使用される。鍵は、管理上の理由（鍵の所有者が組織を離れた、鍵を含むデバイスがサービスから削除された、など）で失効される場合もあるし、鍵が認可されていない組織に開示された又はその組織がアクセスしたと思われる理由がある場合には、緊急に失効される場合もある。いずれの場合も、暗号鍵は、失効の必要性が判断された後、可能な限り速やかに失効される**べきである**。

鍵を使用していた、使用している、又は使用する予定のエンティティ（例えば、依拠当事者）に、当該鍵が失効したことを通知する必要がある。

- （秘密の）対称鍵が失効した場合、当該鍵を共有する全てのエンティティに通知する必要がある（例えば、危殆化鍵リスト（CKL）の使用）
- 非対称鍵ペアの場合、失効はプライベート鍵を参照する。しかし、公開鍵証明書が使用される場合、プライベート鍵に対応する公開鍵を含む証明書は失効し、依拠当事者には、例えば、証明書失効リスト（CRL）又はオンライン証明書ステータスプロトコル（OCSP）を使用して通知される

通知は、失効した鍵を使用している可能性のある全てのエンティティに積極的に通知を送信するか、又はエンティティが鍵のステータスを要求できるようにすることで提供することができる（すなわち、ステータス情報の“プッシュ”又は“プル”）。通知には、鍵の完全な識別情報（鍵自体を除く）、失効の日時、及び適切な場合には失効の理由（例えば、鍵の危殆化）を含める**べきである**。提供された失効情報に基づいて、他のエンティティは、失効した鍵で保護された情報をどのように 取り扱うかを決定することができる。

例えば、あるエンティティが組織を離れたために署名検証公開鍵が失効した場合、失効日以前に作成された署名を全て尊重することが適切であるかもしれない（すなわち、それらの署名の検証を継続し、検証が成功した場合にはそれらの署名を有効なものとして受け入れる）。署名プライベート鍵が危殆化した結果として対応する公開鍵が失効した場合には、失効通知の日付より前に署名された情報が有効とみなされるかの評価を行う必要がある。

別の例として、MAC を生成するために使用する対称鍵を失効させ、その鍵が新しい情報に対する MAC を生成するために使用されないようにしてもよい。但し、当該鍵は、アーカイブされた文書を検証できるように、保持されてもよい。

鍵の失効の詳細は、個々の鍵のライフサイクルを反映**すべきである**。鍵がペアの状況で使用される場合（例えば、同じ暗号化対称鍵を使用して通信する 2 つのエンティティ）、当該鍵を失効したエンティティは、その他のエンティティに失効を通知し**なければならない**。鍵がインフラストラクチャに登録されている場合、鍵を失効したエンティティが、当該鍵に依拠するその他のエンティティに常に直接通知

[103] 鍵材料を単純に削除しても、情報を完全には消去できない場合がある。例えば、情報を消去するためには、当該情報に、ランダムなビット値や全て 0 か 1 のビット値など、関連性のない他の情報を複数回上書きする必要があるかもしれない。メモリに長期間保存された鍵は、“焼き付け”られる可能性がある。鍵を鍵コンポーネントに分割し頻繁に更新することで、この問題を軽減することができる（[DiCrescenzo]参照）。

することはできない。その代わりに、当該鍵を失効したエンティティは、その鍵を失効させる必要があることをインフラストラクチャに通知し**なければならない**（例えば、証明書失効要求を使用する）。インフラストラクチャは、当該鍵を失効して登録解除することで対応し**なければならない**（8.3.3 節参照）。

PKI では、鍵の失効は、一般的に、失効証明書のリスト（すなわち CRL）に当該証明書を含めることで達成される。PKI がオンラインステータスメカニズム（例えば、オンライン証明書ステータスプロトコル、RFC 2560[104]）を使用する場合、失効は、適切な証明書ステータスサーバに通知することで達成される。例えば、プライベート鍵が危殆化した場合、対応する公開鍵証明書は可能な限り速やかに失効**されなければならない**。鍵の危殆化を理由とする証明書の失効は、所有者と当該鍵の間の結び付きがもはや信頼できなくなったことを示している。依拠当事者は、リスクを真剣に検討し、このような状況についての組織の方針を参照することなしに、証明書を受け入れる**べきではない**。その他の失効理由においては、元の結び付きがまだ有効であり鍵が危殆化していないかもしれないが、証明書の公開鍵の使用を終了**すべきである**。繰り返しになるが、依拠当事者は、この問題について組織の方針を参照**すべきである**。

対称鍵システムでは、鍵の失効は、理論的には、サーバのストレージから当該鍵を削除するだけで可能である。対称鍵の失効は、ブラックリストや危殆化鍵リストに当該鍵を追加することで実現するのがより一般的である。これは、監査と管理の要件を満たすのに役立つ。

8.4 破棄フェーズ

その鍵はもう利用できない。これは必須ではないが、その存在に関する全ての記録が削除される可能性がある。組織によっては、監査目的で特定のメタデータ要素の保持を要求する場合がある。例えば、表向きは破棄されたと思われる鍵のコピーが制御されていない環境で発見された場合、又は後に危殆化していたと判断された場合、当該鍵の識別子、タイプ及び暗号利用期間の記録は、当該鍵によりどのような情報が保護されていたのか、及びその危殆化からどのように復元するのが最善かを判断するのに役立つ。

さらに、破棄された鍵と危殆化した鍵の両方のメタデータを記録して置いておけば、どの鍵が通常のライフサイクルを経て遷移したのか、どの鍵がライフサイクルのある時点で危殆化したのかを追跡することができる。したがって、通常のライフサイクルを経た鍵の名前にリンクされた保護情報は、アルゴリズムのセキュリティ強度が十分であれば、安全であると考えられる。しかし、危殆化した鍵の名前にリンクされている保護情報は全て、それ自体が危殆化している可能性がある。

[104] RFC 2560: *X.509 Internet Public Key Infrastructure, Online Certificate Status Protocol, OCSP.*

9 追加の考慮事項

鍵管理の多くの側面は、暗号鍵とそのメタデータを管理するために設計された鍵管理システムで処理できる。自動化された鍵管理システムは、鍵管理プロセスを監視、自動化及び安全にするために使用される。しかし、そのようなシステムを使用する場合、システムを設計し、運用するための追加の考慮事項がある。その中には、アクセスを要求する個人を認証し、その認可を検証することでシステムへのアクセスを制御すること（9.1 節参照）、鍵や証明書の棚卸リストを作成して維持し、鍵や証明書を交換する必要がある場合や、鍵や証明書の交換の責任者を監視すること（9.2 節参照）、責任の割り当てとシステム活動を監視すること（9.3 節参照）、システムの実装とパフォーマンスを監査し、不正がないか監査ログを調査すること（9.4 節参照）、及びシステムの生存性を確保すること（9.5 節参照）などがある。

9.1 アクセス制御と ID 認証

アクセス制御システムは、認可されたエンティティによる要求からの応答に限って、全ての鍵及びメタデータの管理機能が起動できることを保証するために必要である。エンティティによって鍵管理機能が起動される場合、アクセス制御システムは、そのエンティティが認証され、要求された機能のうち認可されたもののみを実行し、適用される全ての制約が満たされていることを保証しなければならない。

アクセス制御システムは、その全ての鍵及びメタデータの管理サービスや機能へのアクセスと開始を制御し**なければならず**、サービスや機能の実行を要求したエンティティの身元と認可を検証した後にのみ、要求されたサービスや機能を開始するためのアクセスと許可を与えなければならない。

アクセス制御を提供するためには、鍵にアクセスするエンティティを識別する手段を実装する必要がある。一般的に使用される方法には、二要素認証やデジタル署名証明書の使用などがある。これらの方法では、エンティティの身元を証明する必要がある。SP 800-63、SP 800-130 及び SP 800-152 は、アクセス制御と ID 認証に関する更なる説明と要件を記述している。

鍵への全てのアクセスは、定期的及び緊急時の検査のために、成功した試行と失敗した試行の両方を含めて、監査ログに記録されるべきである。これらのログの監査についての説明は、9.4 節参照のこと。

9.2 棚卸リスト管理

鍵を使用する暗号メカニズムを使用する場合は、全ての長期鍵を棚卸リストに記載し**なければならない**。対称鍵の場合、これには、配送中及び保管中の情報の保護に使用される鍵が含まれる。非対称鍵ペアの場合、これには、組織のエンティティ（すなわち、鍵ペアのプライベート鍵の使用を認可されている組織内のエンティティ）が所有する鍵ペアを含む。鍵ペアの公開鍵に対して証明書が発行された場合、その証明書の記録を維持し**なければならない**。これらの記録は、棚卸リスト管理システムによって維持される必要がある。

棚卸リスト管理は、使用中の鍵や証明書の記録を確立して維持すること、所有者又は保証人[105]を割り当て追跡すること（所有者又は保証人が誰かもしくは何者か、どこにいるか、及びどうやって連絡するか）、鍵及び証明書の状態を監視すること（有効期限、鍵が危殆化しているかどうか、など）、及び必要に応じて是正措置のために適切な責任者に状況を報告すること、が関係する。

[105] 所有者と保証人の説明は 2 節を参照のこと。

9.2.1 鍵の棚卸リスト

鍵の棚卸リストには、各鍵に関する情報（例えば、鍵に関連するメタデータの全部又は一部）を含めなければならない。棚卸リストが鍵のバックアップやアーカイブにも使用される場合を除き、棚卸リストに秘密鍵やプライベート鍵は含んではならないが、鍵への参照（鍵の識別子や鍵の場所へのポインタなど）は含まなければならない。棚卸リスト内の情報は、誰（何）が鍵を所有又は共有しているか、鍵の種類、鍵が使用されるアルゴリズム、鍵長、鍵の使用方法（例えば、アプリケーション）、及び有効期限を示すべきである。

鍵の棚卸リストは、中央のリポジトリ又は相互に信頼できるリポジトリのネットワークで管理され、鍵棚卸リスト管理ポリシーに従って運用されるべきである。追加情報については、SP 800-57 Part 2 を参照のこと。

鍵が危殆化した場合、危殆化した当該鍵と結びつく所有者又は保証人に通知することで、鍵の失効、危殆化の影響の分析の実施、相応しい場合は鍵の交換などの是正措置を取ることができる。棚卸リスト内の情報を利用して、誰に通知すべきか、またどのように連絡すべきかを特定することができる。

所有者が鍵の使用を認可されなくなった（所有者が組織を離れた人間である、システムから削除されたデバイスである、などの）場合、他のエンティティに通知して、その鍵を使用したさらなるやり取りを終了するようにする必要がある。鍵が対称鍵の場合、棚卸リスト内の情報は、通知を受ける必要がある他のエンティティとその連絡方法を特定するために使用することができる。鍵が非対称鍵であり、PKI 証明書が使用される場合、通知は通常、CRL を使用して行われる。

鍵の暗号利用期間が満了した場合、又は満了しようとしている場合、当該鍵を使用するエンティティ間のやり取りを継続するためには、鍵を交換する必要がある（すなわち、交換鍵を使用する）。鍵棚卸リスト管理システムは、暗号利用期間を監視し、鍵の所有者や保証人に当該鍵が満了しようとしていることを警告するために使用できる。また棚卸リスト管理システムは、アルゴリズムや鍵の交換を手配するために、もはや安全であるとは見なされないアルゴリズムや鍵長の鍵を見つけるためにも使用できる。

9.2.2 証明書の棚卸リスト

長年にわたって、公開鍵証明書の利用は拡大している。通信サーバ（TLS や SSH など）が使用する証明書、ウェブアプリケーションやサービス（政府のサービス、オンラインバンキング、フライトオペレーション、組織内のミッションクリティカルなサービスなど）を提供するために使用する証明書、デバイス（ルータなど）が使用する証明書、それらの通信やサービスを利用する人間が使用するクライアントアプリケーション（ブラウザなど）が使用する証明書など。証明書は、身元を証明し、文書上の署名や通信情報の完全性を検証するための公開鍵を提供し、通信を保護するための鍵を確立するために使用される。

多くの場合、証明書は、証明書に関連する詳細（証明書に関連するデバイスやプロセスが誰か（どれか）、どのデバイスやプロセスがプライベート鍵を持っているか、デバイスやプロセスの場所、証明書の有効期間、など）を記録することなく作成され、インストールされる。その結果、1）業務を継続する前に証明書の有効期限が切れ、証明書を交換する必要がある場合や、2）プライベート鍵が危殆化し、安全な運用を確保する前に証明書を失効させ交換する必要がある場合に、重大な停止が発生する。証明書の有効期限が切れた場合の復元作業を容易にし、停止を回避するために、証明書は作成時に棚卸リストに記載されなければならない。

証明書の棚卸リストには、各エンティティの最新の証明書と、証明書所有者の身元や所有者の連絡先情報などの各証明書に関する情報が含まれる。証明書の公開鍵に関連付けられたプライベート鍵は、棚卸リストが鍵のバックアップ又はアーカイブにも使用され、かつプライベート鍵のバックアップ又はア

ーカイブが許可されている場合を除き、棚卸リストに含めてはならない（8.2.2.1 節及び 8.3.1 節参照）。証明書の棚卸リストは、中央のリポジトリ又は相互に信頼されたリポジトリのネットワーク内で維持され、証明書ポリシーに基づいて運用されるべきである。追加情報については、SP 800-57 パート 2 を参照のこと。

証明書棚卸リストアプリケーションは、証明書の棚卸リストへの入力、期限切れする前に証明書を交換するための証明書の有効期間の監視、もはや安全ではないアルゴリズムや鍵長の使用の検出、暗号インシデント（例えば、CA の危殆化）への対応、及び証明書の保守のための連絡先の変更のために使用されなければならない。

9.3 説明責任

説明責任には、鍵管理に関して、責任と追跡可能性という 2 つの異なる側面がある。

鍵管理に関わる各人は、鍵管理の責任について明確に知らされ、それを果たすための説明責任を負わなければならない。これには、人々に割り当てられた役割、それらの役割に対する鍵管理の責任、それらの役割に割り当てられた個人の役割履行の監視などが含まれる。鍵管理に関連する様々な役割と責任については、SP 800-130 を参照のこと。

説明責任／追跡可能性には、鍵の生成、アクセス、破棄、及びその他の鍵の使用を認可されているエンティティを識別することが含まれる。すなわち、これらの行為を実際に実行した人や内容を記録すること、及びこれらのログを監査してセキュリティ違反がないかどうかを確認すること（9.4 節も参照のこと）。

説明責任／追跡可能性は、鍵の危殆化を防止し、危殆化が検知された場合の危殆化の影響を軽減するのに役立つ有効なツールとなり得る。システムの追跡可能性を提供するためには、システムにアクセスするエンティティを識別する手段を実装し、アクセス制御メカニズムを採用する必要がある（9.1 節参照）。

人間が鍵を閲覧できないことが望ましいが、最低限、平文の暗号鍵へのアクセスは、当該鍵にアクセスしたエンティティまで（エンティティが人間、デバイス、アプリケーション、プロセスであるかどうかにかかわらず）追跡可能でなければならない。暗号化された鍵及び鍵シェアへのアクセスも、アクセスしたエンティティまで追跡可能であるべきである。例えば、洗練された説明責任／追跡可能性システムは、その全ライフサイクルに渡って、所与の鍵を管理していた各エンティティを特定することができるかもしれない。これには、鍵を生成したエンティティ、データを暗号保護するために鍵を使用したエンティティ、鍵にアクセスしたことが知られているその他のエンティティ、鍵が不要になったときに当該鍵を破棄する責任を負ったエンティティなどが含まれる。これら以外のエンティティは、平文形式の鍵に実際にアクセスしたことはないかもしれないが、鍵に対して行った行為、又は鍵を用いて行った行為は全てエンティティまで追跡可能であるべきである。

説明責任／追跡可能性には、3 つの大きな利点がある：

1. いつ危殆化が発生したのか、どのようなエンティティが関与しているのかを判断するのに役立つ。
2. 危殆化から保護される傾向がある。なぜなら、鍵にアクセスできる個人は、当該鍵にアクセスしたことが知られるということを知っており、さらに、鍵にアクセスするデバイス、アプリケーション及びプロセスの開発者は、これらのエンティティまでアクセスが追跡されることを知っているためである。

3. 検出された鍵の危殆化から復元する際、その鍵がどこで使用されたのか、及び危殆化した鍵によって保護されたデータやその他の鍵が何であるのかを知ることは非常に有用である。

暗号鍵の使用の追跡可能性を施行する際には、一定の原則が有用であることがわかっている。これらの原則は、全てのシステムや全ての鍵タイプに適用できるわけではない。原則には以下のようなものがある：

a. 鍵を一意に識別すること

b. 対称鍵又はプライベート鍵で保護されている他の鍵を特定すること

c. 鍵又は関連するメタデータに関連するアクティビティのログを取る（すなわち、記録する）こと。例えば、鍵又は関連するメタデータの生成、アクセス、変更、失効、破棄、それらへのその他のアクセス、など。SP 800-152 の 8.2.4 節に、鍵管理イベント（鍵の生成や破棄など）についての記録**しなければならない**適切な情報が列挙されている。また、どの鍵が使用されているかのデータも全て示される**べきである**

9.4 監査

鍵管理に関連するアクティビティの監査が求められる。鍵管理システムに対して、以下の 3 種類の監査を行う**べきである**：

1. 初期及び定期的なコンプライアンス監査は、鍵管理システムが鍵管理ポリシーと実践要件を遵守して運用する準備ができているか、又は運用を継続しているかを判断するために実施される**べきである**。これには、1）セキュリティ計画及びその計画を支援するために策定された手順の審査を行い、それらがそのポリシーを支援しているかどうかを判断すること（SP 800-57 パート 2 参照）、及び 2）役割と責任が定義されているか、全ての参加者が教育を受け、その役割を理解しているか、全ての鍵の正確な棚卸リストを維持するためのシステムが整備されているか、使用されるプロセスがアプリケーションとリスクに応じて適切であるか、アクセス制御が適切に実施されているか、及び要員の再配置が行われた場合にアクセスが削除されているかどうかを確認すること、などが含まれる。

2. 採用されている保護メカニズム（例えば、アクセス制御メカニズム）は、それが現在提供するセキュリティレベル及び将来的に提供されると期待されるセキュリティレベル、並びにそのメカニズムが適切なポリシーを正しくかつ効果的にサポートしているかどうかについて、定期的に再評価される**べきである**。新たな技術開発や攻撃を考慮に入れる**べきである**。

3. より頻繁に、システムを使用、運用及び保守するエンティティの行動を見直して、確立されたセキュリティ手順を継続して遵守しているかどうか、また、認可を与えられた鍵とメタデータのみにアクセスしているかどうかを検査する**べきである**。これは、通常、セキュリティ関連のイベントを記録するために作成されたログを調べることで達成される（9.3 節参照）。強力な暗号システムでも、甘く不適切な行動によって危険にさらされる可能性がある。極めて異常なイベントは、システムに対する攻撃の試みの可能性を示す指標として、注意しつつ検査**されるべきである**。

監査報告は、鍵管理ポリシーに規定されている通りに（例えば、システム責任者に）提供され**なければならない**。

9.5 鍵管理システムの生存可能性

鍵管理システムやその環境に障害が発生すると、組織の保存された情報へのアクセスが妨げられたり、停止したりする可能性がある。災害復旧には、施設の損傷、ユーティリティサービスの停止、通信や計算機の停止、ハードウェアやソフトウェアの障害、及び保管されている鍵情報や鍵管理システム自体の破損や損失につながるその他の障害から復旧するための手順と十分なバックアップ能力が必要である。

OMB11/01[106]は、暗号化は、機関の管理に委ねられた機微情報の機密性を保護するための重要なツールである一方、機関のデータを暗号化するとミッションの遂行に必要な情報の可用性にリスクももたらすと指摘している。機関は、暗号化を導入する際、機関の情報技術処理とサービスの継続性を保護する必要があることに気付かされる。そのガイダンスでは、情報を復号するために必要な暗号鍵へのアクセスがない場合、組織は当該情報へのアクセスを失うリスクがあることを特に指摘している。そのガイダンスでは、機関は、暗号鍵の復元などの適切なデータ復元メカニズムを通じて、情報の可用性と保証の要件に対処しなければならないことを特に強調している。したがって、保存されている暗号化情報を復号するために必要な鍵のバックアップ又はアーカイブコピーを保持することが賢明である。それらバックアップ又はアーカイブコピーには、マスター鍵、鍵ラッピング鍵、元となる平文情報へのアクセスの要求がなくなるまで暗号化された情報を復号するために必要な関連する鍵材料など（9.5.2 節及び 8.2.2.1 節の表 7、表 8 を参照のこと）が含まれる。

9.5.1 バックアップ及びアーカイブされた鍵

OMB11/01 は、暗号化された情報を復号するための復号鍵の復元の必要性に焦点を当てた（9.5.1 節参照）。しかし、8.2.2.1 節の表が示すように、復号に関連する鍵に加え、組織がバックアップ又はアーカイブする必要がある可能性のある運用鍵が存在する（署名検証公開鍵や認可鍵など）。鍵材料のバックアップ又はアーカイブされたコピーは、暗号化された情報の機密性、及びソース認証、ID 認証、完全性認証及び認可プロセスの完全性を保護するために、6 節の規定に従って保管され**なければならない**。

9.5.2 鍵の復元

鍵の復元は、暗号保護された情報を処理する（例えば、暗号化情報を復号したり、署名された情報の署名検証をしたりする）ために、鍵のバックアップやアーカイブから鍵を取得したり、再構築したりするプロセスである。鍵の復元には、以下のようないくつかの問題がある：

1. もしあるとしても、どの鍵情報が、後の復元のためにバックアップ又はアーカイブされる必要があるか
2. バックアップ又はアーカイブされた鍵情報はどこに保存されるか
3. アーカイブはいつ行うか（鍵活性化時、鍵の暗号利用期間の終了時、など）
4. バックアップ又はアーカイブされた鍵情報の保護は誰が責任を持つか
5. 鍵情報の保管と復元には、どのような手順を踏む必要があるか
6. 誰が、どのような条件で、鍵情報の復元を要求することができるか
7. 鍵情報の復元が行われた場合、誰がどのような条件で通知を受けるか

[106] OMB11/01, *OMB Guidance to Federal Agencies on Data Availability and Encryption.*

8. 鍵情報が認可されたエンティティにのみ提供されたことを確実にするために、どのような監査又は説明責任機能を実施する必要があるか

鍵の復元自体は、バックアップやアーカイブから鍵情報を削除することにはならないことに注意されたい。

復元後の鍵に許容される使用は、その暗号利用期間に依存する。5.3.4 節と 5.3.5 節で述べたように、鍵には作成者使用期間、受領者使用期間、又はその両方を割り当てることができる。鍵を復元して使用すべきかどうか、及びどこから復元すべきかは、多くの要因によって決まる。要因には、暗号利用期間、クラス（対称か非対称か）、用途や目的、危殆化したかどうか又は危殆化の疑いがあるかどうか、が含まれる。

作成者使用期間とは、鍵の暗号利用期間中に、その鍵を使用してデータに暗号保護が適用される（データが暗号化される、デジタル署名が生成されるなど）期間をいう。受領者使用期間とは、保護された情報が処理される（データが復号される、署名が検証されるなど）期間である。

鍵がバックアップ又はアーカイブされている場合、以下のように当該鍵を復元して使用することができる：

A. 鍵が危殆化していることが知られていない、又は危殆化している疑いがない場合：

- 秘密鍵（対称鍵）：

  復元した鍵は、当該鍵の作成者使用期間を超えていない**場合に限り**、保護の適用（例：暗号化）に使用できる。復元した鍵はできるだけ早く失効させて、作成者使用期間を終了させる**べきである**。失効後も機能を継続する必要がある場合、暗号保護を適用するために、新しい鍵を生成して復元した鍵を置き換え**なければならない**。

  復元した鍵は、当該鍵の受領者使用期間を超えていなければ、保護されたデータの処理（例：暗号文データの復号）に使用することができる。

- 非対称鍵ペアのプライベート鍵：

  復元した署名プライベート鍵は、当該鍵の作成者使用期間を超えていなければ、署名生成に使用することができる。ほとんどの場合、署名プライベート鍵のバックアップは推奨されておらず（8.2.2.1 節参照）、署名プライベート鍵をアーカイブすることは禁止されている（8.3.1 節参照）ことを思い出すこと。

  復元した鍵配送プライベート鍵は、受領者使用期間を超えていなければ、配送された鍵を復号するために使用することができる。

  復元した鍵合意プライベート鍵は、暗号利用期間を超えていなければ、新しい鍵の確立に使用することができる。

  復元した署名プライベート鍵や鍵合意プライベート鍵は、できるだけ早く失効させる**べきである**[107]。失効後も機能を継続する必要がある場合、新しい鍵ペアを生成して、復元したプライベート鍵とそれに対応する公開鍵を置き換え**なければならない**。

- 非対称鍵ペアの公開鍵：

[107] 対応する公開鍵が公開鍵証明書に含まれている場合は、証明書の公開鍵を失効させることで、鍵ペアを失効させることができる。

復元した署名検証公開鍵及び鍵配送公開鍵は、受領者使用期間を超えていなければ、割り当てられた目的に使用してもよい。

復元した鍵合意公開鍵は、当該鍵の使用期間（すなわち、暗号利用期間）を超えていなければ、使用することができる。

B. 鍵情報が危殆化**した**場合、復元した鍵は失効させ**なければならない**（又は失効していなければならない）（9.5.4 節参照）。

- 秘密の対称鍵：

  復元した鍵は、保護（例：暗号化）を適用するために使用**してはならない**。継続的な機能が必要な場合、新しい鍵を生成**しなければならない**。

  復号された鍵は、保護データの処理に使用（例：復号に使用）することができるが、ある程度のリスクは許容しなければならない。許容できるリスクのレベルを決定するのは、ユーザ及びユーザの組織の責任である。

- 非対称鍵ペアのプライベート鍵：

  復元した署名プライベート鍵は使用**してはならない**。

  復元された鍵配送プライベート鍵は、配送された鍵を復号することのリスクが許容できる場合に、その復号のために使用してもよい。

  復元した鍵合意プライベート鍵は、新しい鍵の確立には使用**してはならない**。既に確立された鍵の再構築することのリスクが許容できる場合には、その再構築のために使用してもよい。

- 非対称鍵ペアの公開鍵：

  復元した署名検証公開鍵は、署名検証することのリスクが許容できる**場合に限り**、署名検証のために使用してもよい。

  復元した鍵配送公開鍵は、使用**してはならない**。

  復元した鍵合意公開鍵は、新しい鍵の確立には使用**してはならない**。既に確立された鍵の再構築することのリスクが許容できる場合には、その再構築のために使用してもよい。

9.5.3 システムの冗長性・継続性計画

暗号技術は、データやリソースへの不正アクセスを防止するための有用なツールである一方、メカニズムが故障した場合、正当なエンティティによる重要な情報やプロセスへのアクセスが阻害される。暗号鍵の唯一のコピーの紛失や破損は、情報へのアクセスを拒否することになる。例えば、鍵屋は通常壊れた物理的なメカニズムを破ることができるが、強力なアルゴリズムで暗号化された情報へのアクセスは、正しい復号鍵がなければ現実的ではない。組織の運営の継続性は、鍵管理システムのコンティンジェンシープラン（緊急時対応計画）に大きく依存しており、その計画には鍵管理や暗号鍵などの重要な論理プロセスや要素の冗長性を含む。

9.5.3.1 一般原則

システム障害からの復旧計画は必須の管理機能である。重要なインフラサービスの中断を予測**すべきであり**、組織の主要なミッション要件をサポートするための業務継続性を維持するための計画を立て**なければならない**。鍵管理に関して、計画が必要とされる状況の代表的なものとして以下のようなものがある：

1. 鍵カード又は鍵トークンの紛失
2. 鍵へのアクセスを管理する認証コード（トークンやパスワードなど）を忘れた場合
3. 鍵入力装置（リーダ、PIV カードなど）の故障
4. 鍵や証明書が保存されているメモリ媒体の紛失又は破損
5. 鍵の危殆化
6. 交換用の証明書がインストールされないまま、証明書が終了した場合
7. 証明書失効リスト（CRL）又は危殆化鍵リスト（CKL）の破損
8. 鍵又は証明書の生成、登録、及び配付を行うシステム、サブシステム又はコンポーネントのハードウェア障害
9. 鍵又は証明書の生成、登録、及び鍵確立を行うシステム、サブシステム又はコンポーネントの再初期化を必要とする停電
10. 鍵又は証明書の生成、登録、及び鍵確立を行うシステム、サブシステム又はコンポーネントに必要なメモリ媒体の破損
11. 鍵又は証明書の配付記録や監査ログの紛失又は破損
12. 鍵と鍵所有者との関連付けの紛失又は破損
13. 鍵情報へのアクセス又は保護された情報の処理に必要な、古いソフトウェアやハードウェアが利用できない場合

復旧の議論では、最も一般的には暗号化されたデータの復旧及び暗号化された通信機能の復旧に焦点が当てられるが、計画はまた、次のようなことにも対応**すべきである**。1）アクセス制御メカニズムに暗号技術が使用されている場合のアクセスの復旧（アクセス保護の一時的な喪失なしに）、2）認可メカニズムに暗号技術が使用されている場合の重要なプロセスの復旧（認可制限の一時的な喪失なしに）、及び 3）デジタル署名やメッセージ認証アプリケーションにおける完全性保護の維持／復旧。

コンティンジェンシープランには、以下が含まれる**べきである**。1）重要な障害を迅速に認識して報告するための手段の提供及び責任の割り当て、2）障害が発生したシステム、サブシステム及びコンポーネントを迂回又は交換するための責任の割り当てとリソースの配置、及び 3）詳細な迂回手順や復旧手順の確立。

コンティンジェンシープランには、統合された物流支援機能の全範囲が含まれる。予備部品（重要なデバイス、ソフトウェアプログラム、マニュアル、及びデータファイルのコピーを含む）は、利用可能（取得済み又は手配済み）で、事前に配置（又は納入段階）されている**べきである**。緊急時メンテナンスや交換、迂回の指示は、事前に準備され、指定された個人及びアクセス可能な公表されたアクセスポイントの両方に配付される**べきである**。指定された個人は、割り当てられた復旧手順について訓練されている**べきであり**、全ての要員は、報告手順及び復旧手順について訓練されている**べきである**。

9.5.3.2 暗号化及び鍵管理に特化した復元の問題

鍵復元の特徴には、一般的に、鍵情報の何らかの冗長性又は複数のコピーが含まれる。重要な鍵情報の 1 つのコピーが失われたり、破損したりした場合、別のコピーは通常、データを回復したり、機能を復元したりするために利用可能である必要がある。同時に、鍵のコピーが多く存在し、異なる場所に配付されればされるほど、その鍵は通常、保管場所への侵入又は管理者（ユーザ、サービスエージェント、鍵の生産／配付の施設など）の背信による危殆化の影響を受けやすくなる。この意味で、鍵の機密性の要件は、運用継続性の要件と相反する。鍵情報（特に、対称鍵及び（非対称）プライベート鍵に関する情報）の全てのコピーを保護するために特別な注意を払う必要がある。コンティンジェンシープラン及び計画要件の詳細は、SP 800-57 パート 2 に記載されている。

9.5.4 危殆化からの回復

機微情報や重要なプロセスを保護するために使用されている秘密鍵又はプライベート鍵が不正なエンティティに開示された場合、その鍵で保護されている全ての情報やプロセスは、直ちに不正開示や改ざん、背信、サービス拒否の対象となる。危殆化した鍵は全て失効し**なければならない**。影響を受けた全ての鍵は必要に応じて交換され**なければならない**。さらに、機微性の高い又は重要な情報やプロセスが影響を受ける場合、直ちに損害評価を行う**べきである**。保護されたデータやプロセスへの不正アクセスが疑われる場合の結果を緩和し、将来の危殆化の可能性や頻度を低減するために必要な措置を講じる**べきである**。

秘密鍵（対称鍵）や（非対称）プライベート鍵を使用して、単一のエンティティのローカル情報、又は単一のペアのエンティティ間通信のみを保護する場合、危殆化からの回復プロセスは比較的単純かつ安価である。被害の評価や緩和策は、ローカルな問題であることが多い。

一方、鍵が多数のエンティティで共有されているか、多数のエンティティの影響を受けている場合は、被害が広範囲に及ぶ可能性があり、回復は複雑かつ高額になる。鍵の危殆化からの回復が特に困難であるときや費用がかかるような鍵の例としては、以下のようなものがある：

1. CA の署名プライベート鍵、特に公開鍵基盤のルート証明書に署名するために使用される場合
2. 多数のエンティティが共有する鍵ラッピング対称鍵
3. 多数のエンティティが鍵導出に使用するマスター鍵／鍵導出鍵
4. 大規模な分散データベースでデータを暗号化するために使用するデータ暗号化対称鍵
5. 多数の通信ネットワーク参加者が共有する対称鍵
6. 多数の保存された鍵を保護するために使用される鍵

これら全ての場合において、多数の鍵所有者又は依拠当事者（例えば、対称鍵アルゴリズムの秘密鍵、又は非対称鍵アルゴリズムの公開鍵を使用することを認可された全ての当事者）に、危殆化したことを直ちに通知する必要がある。鍵識別子を危殆化鍵リスト（CKL）に含める、又は後日公開される証明書失効リスト（CRL）に証明書のシリアル番号を含めるだけでは十分ではないかもしれない。このことは、影響を受ける（最も可能性の高い）エンティティのリストを維持する必要があり、危殆化のニュースを伝達する手段が必要であることを意味する。特に対称鍵の危殆化の場合、危殆化のニュース及び鍵の交換は、影響を受けたエンティティにのみ送られる**べきである。**これは、その他の人がこの状況を悪用できないようにするためである。

これら全ての場合において、危殆化した鍵を交換するための安全な経路が必要である。サービスの迅速な復旧を可能にするためには、自動化された（例：無線やネットワークベースの）交換経路が好ましい（8.2.3 節参照）。しかし、場合によっては、手動配付に代わる実用的な代替手段がない場合もある（例えば、ルート CA のプライベート鍵の危殆化）。不測の事態用の代替鍵配付は、状況によってはサービスの迅速な復旧に役立つかもしれない（例えば、広く保持されている対称鍵の危殆化）。しかし、運用中の鍵と不測の事態用代替鍵の同時危殆化の可能性を考慮する必要がある。

被害の評価は非常に複雑になる可能性がある。CA のプライベート鍵や広く使用されている配送鍵、大規模な分散データベースの多くのユーザが使用する鍵の危殆化及び交換のような場合は特にそうである。

参考文献

ANSX9.31 Accredited Standards Committee X9 (1998) *ANSI X9.31-1998— Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry (rDSA)* (American National Standards Institute) [Withdrawn].

DiCrescenzo Di Crescenzo G, Ferguson N, Impagliazzo R, Jakobsson M (1999) How to forget a secret. *STACS 99: 16th Annual Symposium on Theoretical Aspects of Computer Science* (Springer, Trier, Germany), pp 500-509. https://doi.org/10.1007/3-540-49116-3_47

FIPS140 National Institute of Standards and Technology (2002) Security

Requirements for Cryptographic Modules. (U.S. Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS) 140-2.
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.140-2.pdf

National Institute of Standards and Technology (2019) Security Requirements for Cryptographic Modules. (U.S. Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS) 140-3.
https://doi.org/10.6028/NIST.FIPS.140-3

FIPS180 National Institute of Standards and Technology (2015) Secure Hash Standard (SHS). (U.S. Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS) 180-4.
https://doi.org/10.6028/NIST.FIPS.180-4

FIPS186 National Institute of Standards and Technology (2013) Digital Signature Standard (DSS). (U.S. Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS) 186-4.
https://doi.org/10.6028/NIST.FIPS.186-4

National Institute of Standards and Technology (2013) Digital Signature Standard (DSS). (U.S. Department of Commerce, Washington, D.C.), Draft Federal Information Processing Standards Publication (FIPS) 186-5.
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-5-draft.pdf

FIPS197 National Institute of Standards and Technology (2001) Advanced Encryption Standard (AES). (U.S. Department of Commerce, Washington, DC), Federal Information Processing Standards Publication (FIPS) 197.
https://doi.org/10.6028/NIST.FIPS.197

FIPS198 National Institute of Standards and Technology (2008) The Keyed-Hash Message Authentication Code (HMAC). (U.S. Department of Commerce, Washington, DC), Federal Information Processing Standards Publication (FIPS) 198-1.
https://doi.org/10.6028/NIST.FIPS.198-1

FIPS199 National Institute of Standards and Technology (2004) Standards for Security Categorization of Federal Information and Information Systems. (U.S. Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS) 199.
https://doi.org/10.6028/NIST.FIPS.199

FIPS201 National Institute of Standards and Technology (2013) Personal Identity Verification (PIV) of Federal Employees and Contractors, (U.S. Department of Commerce,

Washington, D.C.), Federal Information Processing Standards Publication (FIPS) 201-2. https://doi.org/10.6028/NIST.FIPS.201-2.

FIPS202 National Institute of Standards and Technology (2015) SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions. (U.S. Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS) 202. https://doi.org/10.6028/NIST.FIPS.202

FPKI-KRP Federal Public Key Infrastructure Policy Authority (2017) Federal Public Key Infrastructure Key Recovery Policy, version 1.0. Available at https://www.idmanagement.gov/wp-content/uploads/sites/1171/uploads/fpki-krp-v1.0-10-6-2017.pdf

IG 7.5 National Institute of Standards and Technology, Canadian Centre for Cyber Security (2003) Strength of Key Establishment Methods. *Implementation Guidance for FIPS 140-2 and the Cryptographic Module Validation Program (CMVP)*. (National Institute of Standards and Technology, Gaithersburg, MD), Section 7.5 [Amended]. Available at https://csrc.nist.gov/csrc/media/projects/cryptographic-module-validationprogram/documents/fips140-2/fips1402ig.pdf

ITLBulletin Burr WE, Hash JS (2002) Techniques for System and Data Recovery. (National Institute of Standards and Technology, Gaithersburg, MD), ITL Bulletin, April 2002. Available at https://csrc.nist.gov/publications/detail/itl-bulletin/2002/04/techniques-forsystem-and-data-recovery/final

OMB11/01 Office of Management and Budget (2001) OMB Guidance to Federal Agencies on Data Availability and Encryption. (National Institute of Standards and Technology, Gaithersburg, MD), [November 26, 2001]. Available at https://csrc.nist.gov/csrc/media/projects/block-ciphertechniques/documents/ombencryption-guidance.pdf

RFC2560 Myers M, Ankney R, Malpani A, Galperin S, Adams C (1999) X.509 Internet Public Key Infrastructure, Online Certificate Status Protocol – OCSP. (Internet Engineering Task Force (IETF) Network Working Group), IETF Request for Comments (RFC) 2560. https://doi.org/10.17487/RFC2560

RFC 3647 Chokhani S, Ford W, Sabett R, Merrill C, Wu S (2003) Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework (Internet Engineering Task Force (IETF) Network Working Group), IETF Request for Comments (RFC) 3647. https://doi.org/10.17487/RFC3647

RFC 8032 Josefsson S, Liusvaara I (2017) Edwards-Curve Digital Signature Algorithm (EdDSA). (Internet Research Task Force (IRTF)), IRTF Request for Comments (RFC) 8032. https://doi.org/10.17487/RFC8032.

SP 800-32 Kuhn DR, Hu VC, Polk WT, Chang S-jH (2001) Introduction to Public Key Technology and the Federal PKI Infrastructure. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-32. https://doi.org/10.6028/NIST.SP.800-32

SP 800-37 Joint Task Force (2018) Risk Management Framework for Information Systems and Organizations: A System Life Cycle Approach for Security and Privacy. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-37, Rev. 2. https://doi.org/10.6028/NIST.SP.800-37r2

SP 800-38 Recommendation for Block Cipher Modes of Operation (all parts). Available at https://csrc.nist.gov/projects/block-cipher-techniques/bcm/current-modes

SP 800-38A Dworkin MJ (2001) Recommendation for Block Cipher Modes of Operation: Methods and Techniques. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-38A. https://doi.org/10.6028/NIST.SP.800-38A

SP 800-38B Dworkin MJ (2005) Recommendation for Block Cipher Modes of Operation: the CMAC Mode for Authentication. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-38B, Includes updates as of October 6, 2016. https://doi.org/10.6028/NIST.SP.800-38B

SP 800-38C Dworkin MJ (2004) Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-38C, Includes updates as of July 20, 2007. https://doi.org/10.6028/NIST.SP.800-38C

SP 800-38D Dworkin MJ (2007) Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-38D. https://doi.org/10.6028/NIST.SP.800-38D

SP 800-38F Dworkin MJ (2012) Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-38F. https://doi.org/10.6028/NIST.SP.800-38F

SP 800-52 Polk T, McKay KA, Chokhani S (2019) Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-52, Rev. 2. https://doi.org/10.6028/NIST.SP.800-52r2

SP 800-56A Barker EB, Chen L, Roginsky A, Vassilev A, Davis R (2018) Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-56A, Rev. 3. https://doi.org/10.6028/NIST.SP.800-56Ar3

SP 800-56B Barker EB, Chen L, Roginsky A, Vassilev A, Davis R, Simon S (2019) Recommendation for Pair-Wise Key-Establishment Using Integer Factorization Cryptography. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-56B, Rev. 2. https://doi.org/10.6028/NIST.SP.800-56Br2

SP 800-56C Barker EB, Chen L, Davis R (2018) Recommendation for Key-Derivation Methods in Key-Establishment Schemes. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-56C, Rev. 1. https://doi.org/10.6028/NIST.SP.800-56Cr1

SP 800-57, Part 2 Barker EB, Barker WC (2019) Recommendation for Key Management: Part 2 – Best Practices for Key Management Organizations. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-57 Part 2, Rev. 1. https://doi.org/10.6028/NIST.SP.800-57pt2r1

SP 800-57, Part 3 Barker EB, Dang QH (2015) Recommendation for Key Management, Part 3: Application-Specific Key Management Guidance. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-57 Part 3, Rev. 1. https://doi.org/10.6028/NIST.SP.800-57pt3r1

SP 800-63 Grassi PA, Garcia ME, Fenton JL (2017) Digital Identity Guidelines. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-63-3, Includes updates as of December 1, 2017. https://doi.org/10.6028/NIST.SP.800-63-3

SP 800-63A Grassi PA, Fenton JL, Lefkovitz NB, Danker JM, Choong YY, Greene KK, Theofanos MF (2017) Digital Identity Guidelines: Enrollment and Identity Proofing. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-63A, Includes updates as of December 1, 2017. https://doi.org/10.6028/NIST.SP.800-63A

SP 800-67 Barker EB, Mouha N (2017) Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-67, Rev. 2. https://doi.org/10.6028/NIST.SP.800-67r2

SP 800-88 Kissel R, Regenscheid A, Scholl M, Stine K (2014) Guidelines for Media Sanitization. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-88, Rev. 1. https://doi.org/10.6028/NIST.SP.800-88r1

SP 800-89 Barker EB (2006) Recommendation for Obtaining Assurances for Digital Signature Applications. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-89. https://doi.org/10.6028/NIST.SP.800-89

SP 800-90 Joint reference to SP 800-90A, SP 800-90B, and SP 800-90C.

SP 800-90A Barker EB, Kelsey JM (2015) Recommendation for Random Number Generation Using Deterministic Random Bit Generators. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-90A, Rev. 1. https://doi.org/10.6028/NIST.SP.800-90Ar1

SP 800-90B Sönmez Turan M, Barker EB, Kelsey JM, McKay KA, Baish ML, Boyle M (2018) Recommendation for the Entropy Sources Used for Random Bit Generation. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-90B. https://doi.org/10.6028/NIST.SP.800-90B

SP 800-90C Barker EB, Kelsey JM (2016), Recommendation for Random Bit Generator (RBG) Constructions. (National Institute of Standards and Technology, Gaithersburg, MD), Second Draft NIST Special Publication (SP) 800-90C. Available at https://csrc.nist.gov/publications/detail/sp/800-90c/draft

SP 800-107 Dang QH (2012) Recommendation for Applications Using Approved Hash Algorithms. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-107, Rev. 1. https://doi.org/10.6028/NIST.SP.800-107r1

SP 800-108 Chen L (2009) Recommendation for Key Derivation Using Pseudorandom Functions (Revised). (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-108, Revised. https://doi.org/10.6028/NIST.SP.800-108

SP 800-130 Barker EB, Smid ME, Branstad DK, Chokhani S (2013) A Framework for Designing Cryptographic Key Management Systems. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-130. https://doi.org/10.6028/NIST.SP.800-130

SP 800-131A Barker EB, Roginsky A (2019) Transitioning the Use of Cryptographic Algorithms and Key Lengths. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-131A, Rev. 2. https://doi.org/10.6028/NIST.SP.800-131Ar2

SP 800-132 Sönmez Turan M, Barker EB, Burr WE, Chen L (2010) Recommendation for Password-Based Key Derivation: Part 1: Storage Applications. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-132. https://doi.org/10.6028/NIST.SP.800-132

SP 800-133 Barker EB, Roginsky AL (2019) Recommendation for Cryptographic Key Generation. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-133, Rev. 1. https://doi.org/10.6028/NIST.SP.800-133r1

SP 800-135 Dang QH (2011) Recommendation for Existing Application-Specific Key Derivation Functions. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-135, Rev. 1. https://doi.org/10.6028/NIST.SP.800-135r1

SP 800-152 Barker EB, Branstad DK, Smid ME (2015) A Profile for U.S. Federal Cryptographic Key Management Systems (CKMS). (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-152. https://doi.org/10.6028/NIST.SP.800-152

SP 800-175B Barker EB (2020) Guideline for Using Cryptographic Standards in the Federal Government: Cryptographic Mechanisms. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-175B, Rev. 1. https://doi.org/10.6028/NIST.SP.800-175Br1

SP 800-185 Kelsey JM, Chang S-jH, Perlner RA (2016) SHA-3 Derived Functions: cSHAKE, KMAC, TupleHash, and ParallelHash. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-185. https://doi.org/10.6028/NIST.SP.800-185

付録 A - 暗号学的及び非暗号学的な完全性及びソース認証のメカニズム

完全性サービス及びソース認証サービスは、鍵管理を含むプロトコルにおいて特に重要である。本推奨で完全性サービス又はソース認証サービスが説明される場合、それらは暗号学的に“強力な”完全性又はソース認証のメカニズムの使用を前提としている。安全な通信及び鍵管理は、典型的には、完全性保護又は“信頼できる”配送サービス[108]などの特定のサービスを提供する通信プロトコルを使用して提供される。しかし、通信プロトコルの完全性保護や信頼できる配送サービスは、暗号アプリケーション、特に鍵管理には必ずしも適切ではなく、“完全性”などの用語の意味について混乱が生じる可能性がある。

全ての通信チャネルは、ある程度のノイズ（すなわち、伝送媒体によって挿入された意図しないエラー）を有しており、さらにネットワークの混雑などの他の要因によって、ネットワークパケット[109]が失われることがある。したがって、通信プロトコルのための完全性保護及び信頼できる配送サービスは、特定のワーストケースのノイズ特性を有するチャネル上で機能するように設計されている。送信ビットエラーは、典型的には、1）パケット内の送信エラーを検出するための非暗号学的チェックサム[110]、及び 2）失われたパケットを検出するために使用されるパケットカウンタを使用して検出される。破損したパケット（すなわち、ビットエラーを含むパケット）又は紛失したパケットを検出した受信エンティティは、送信者に再送信を要求することができる。非暗号学的チェックサムは、一般に、送信ノイズの検出に有効である。例えば、LAN アプリケーションで使用される一般的な CRC-32 チェックサムアルゴリズムは、32 ビット未満のスパンを有する全てのエラーバーストを検出し、2^{-32} の失敗確率でより長いランダムバーストを検出する。しかし、非暗号学的 CRC-32 チェックサムは、32 ビットのメッセージワードのスワップを検出せず、さらに、特定のメッセージビットにおける特定のエラーは、CRC-32 チェックサムの予測可能な変化を引き起こす。巧妙な攻撃者はこの特徴を利用して、メッセージが暗号化されている場合でも、場合によっては CRC-32 の完全性チェックを通過するような改ざんされたメッセージを作成することができる。

順方向エラー訂正コードは、再送なしで限られた数のエラーを訂正するために使用できる非暗号学的チェックサムのサブセットである。これらのコードは、通信チャネルのアプリケーションやノイズ特性に応じて、チェックサムとして使用されることがある。

一方、暗号学的な完全性認証メカニズム（MAC やデジタル署名など）は、攻撃をノイズとして偽装しようとする能動的で知的な攻撃者から保護する。通常、攻撃者によって改ざんされるビットはランダムではなく、システムの特性や脆弱性を対象としている。暗号学的な完全性認証メカニズムは、ランダムなノイズイベントを検出するのに有効だが、よりシステマティックな意図的な攻撃も検出する。SHA-256 などの暗号ハッシュ関数は、ハッシュ値の各ビットをメッセージテキストの各ビットの複雑で非線形な関数にして、同じ値にハッシュされた 2 つのメッセージを見つけることが非現実的になるように設計されている。平均して、同じ値にハッシュされた 2 つのメッセージを見つけるためには 2^{128} 回の SHA-

---

[108] 情報が正しく受信されることを保証するプロトコルを使用して、ネットワークで情報を送信する手段のこと。

[109] ネットワーク上でメッセージを送信するために使用されるフォーマットされたデータの単位のこと。メッセージを複数のパケットに分割して効率よく送信することがある。

[110] チェックサム：伝送するビットを使用してチェックサム値を作成するアルゴリズムのこと。チェックサム値は通常、送信時に送られる。受信者は、受信した送信内容のビットを使ってチェックサム値を再計算し、受信したチェックサム値と計算した値を比較して、送信内容が正しく受信されたかどうかを判断する。非暗号学的チェックサムアルゴリズムは、よく知られたアルゴリズムを秘密情報なしに（つまり、暗号鍵なしに）使用する。

256 ハッシュ処理を実行する必要があり、SHA-256 ハッシュが任意の与えられたメッセージのハッシュと同じ値である別のメッセージを見つけることはそれ以上にはるかに困難である。暗号学的メッセージ認証コード（MAC）アルゴリズムは、ハッシュ関数又は対称暗号アルゴリズムと鍵を使用して、メッセージのソースを認証し、メッセージの完全性を保護する（すなわち、エラーを検出する）。デジタル署名は、公開鍵アルゴリズムとハッシュ関数を使用して、完全性とソース認証の両方のサービスを提供する。非暗号学的な完全性やソース認証のメカニズムと比較して、これらの暗号サービスは通常計算量が多くなる。このことが避け難いのは、暗号保護は、相当なリソースを持つ知識豊富な敵対者による意図的な攻撃に対しても耐性を持たなければならないためである。

暗号学的及び非暗号学的な完全性認証メカニズムは一緒に使用することができる。例えば、TLS プロトコル（SP 800-52[111]参照）を想定する。TLS では、クライアントとサーバは互いの身元を認証し、共有の“マスター鍵”を確立し、暗号化されたペイロードデータを転送することができる。TLS プロトコル全体の実行中の各ステップは、暗号学的な完全性とソース認証のメカニズムによって保護されており、ペイロードは通常暗号化されている。ほとんどの暗号プロトコルと同様に、TLS は（与えられた確率で）プロトコルの実行中のいずれかの部分を変更するあらゆる攻撃やノイズイベントを検出する。しかし、TLS にはエラー回復プロトコルがない。エラーが検出された場合、プロトコルの実行は単に終了する。新しい TLS プロトコルの実行を開始するには、かなりのコストがかかる。そのため、TLS は、“信頼できる”配送サービス、通常はインターネットトランスポートコントロールプロトコル（TCP）を必要とし、それによって通常のネットワーク伝送エラーを処理して回復する。TLS は攻撃やノイズイベントによって引き起こされたエラーを検出するが、そこから回復するメカニズムはない。TCP は一般的にパケット単位でそのようなエラーを検出し、TLS にデータを配信する前に、個々のパケットを再送することでエラーから回復する。TLS と TCP の両方に完全性認証メカニズムがあるが、巧妙な攻撃者は、TCP のより弱い非暗号学的なチェックサムを簡単に誤魔化すことができる。しかし、TLS で提供されている暗号学的な完全性認証メカニズムのおかげで、攻撃は阻止される。

暗号学的及び非暗号学的な完全性やエラー訂正のメカニズムの間の相互作用のなかには、ユーザやプロトコル設計者が考慮しなければならないことがある。例えば、多くの暗号利用モードでは暗号文のエラーが拡張される。暗号文中の 1 ビットのエラーは、結果として得られる平文のブロック全体又はそれ以上を変化させる可能性がある。暗号化の前に前方誤り訂正[112]が適用され、送信中に暗号文にエラーが挿入された場合、復号中の誤り拡大が誤り訂正メカニズムを“圧倒”して誤りを訂正できなくなる可能性がある。そこで、暗号処理の後に前方誤り訂正メカニズムを適用することが好ましい。これにより、暗号文が復号される前に、受信側エンティティのシステムによる誤り訂正が可能となり、結果的に“正しい”平文が得られる。

暗号学的メカニズムと非暗号学的メカニズムの間の相互作用もまたセキュリティの脆弱性につながる可能性がある。これが発生する古典的な方法の一つは、非暗号学的チェックサム（例えば、CRC-32）を持つストリーム暗号[113]を使用するプロトコルであり、そのチェックサムは平文データ上で計算され、良好なパケットを認識する。攻撃者は、暗号化されたパケットをコピーし、個々の暗号文ビットを選択

---

[111] SP 800-52, *Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS).*

[112] 前方誤り訂正：データ送信時の誤りを制御するために用いられる技術のこと。中心となる考えは、送信者がエラー検出コードを使用して冗長な方法でメッセージを符号化することである。この冗長性により、受信者はメッセージの任意の場所で発生する可能性のある限られた数のエラーを検出することができ、多くの場合、再送信することなくこれらのエラーを修正することができる。

[113] ストリーム暗号は、一度に 1 つの要素（ビットやバイトなど）を暗号化や復号するものである。ストリーム暗号として特に指定された**承認済み**のアルゴリズムは無い。しかし、SP 800-38 で定義されている暗号利用モードのいくつかは、AES などの対称ブロック暗号アルゴリズムを使用することで、ストリーム暗号の機能を実行できる。

的に変更し、CRC のビットを選択的に変更してから、パケットを送信することができる。プロトコルの認識メカニズムを使用して、攻撃者は、CRC が正しい場合を決定することができ、その結果として元の平文の特定のビットを決定することができる。少なくとも 1 つの広く使われている無線暗号プロトコルは、このような攻撃で破られている。

付録 B - 鍵の復元

連邦政府機関は、情報技術システムに含まれる情報、情報技術システムで処理される情報、及び情報技術システム間で転送される情報を保護する責任を負う。暗号技術は、このプロセスの一部としてしばしば使用される。これらの技術は、機密性、完全性認証、ID 認証、ソース認証、否認防止のサポート、又はアクセス制御を提供するために使用される。暗号で保護された情報の保護と継続的なアクセス可能性に対処するためのポリシーが確立**されなければならず**、かつその情報の寿命中、当該情報が利用可能であることを確実にするための手順が確立**されなければならない**。暗号鍵材料が情報を保護するために使用されている場合、この同じ鍵材料は、それらの保護を除去（例えば、復号）又は検証（例えば、MAC を検証）するために利用可能である必要があるかもしれない。

多くの場合、暗号処理に使用される鍵材料は、容易に入手できないかもしれない。これには、以下のような理由による場合がある：

1. 鍵の暗号利用期間が満了し、鍵材料が運用中の保管場所にもはやない
2. 鍵材料が破損している（システムがクラッシュした、ウイルスが運用中のストレージに保存された鍵材料を変更した、など）
3. 鍵の所有者が対応不可であるときに、所有者の組織が平文情報を取得する必要がある

必要なときにこの鍵材料を利用できるようにするためには、鍵材料をどこかに保存するか、又は他の利用可能な鍵材料から構成可能な（例えば、導出可能な）ものにする必要がある。鍵材料を再取得するプロセスは、鍵復元と呼ばれる。鍵復元は、情報復元の一つの方法として、暗号化された情報から平文情報を復元する必要がある場合によく使用される。しかし、鍵材料又は他の関連情報は、別の理由で復元する必要がある場合がある。例えば、通常運用中のストレージの鍵材料の破損など（例：アーカイブされた文書の MAC の検証のため）。8.2.1 節参照のこと。また、鍵復元は、新しい鍵情報を生成して配付するよりも、鍵情報を復元する方が容易又は迅速である状況にも適していることもある。

しかし、鍵材料を長時間保存する必要がないアプリケーションもある。これは、鍵材料又は鍵材料によって保護された情報にアクセスできなくなった場合に、その他の手順で運用能力を回復するからである。この種のアプリケーションには、送信された情報を再送信できる通信や、配付用の新しい鍵材料を迅速に導出又は取得することができるアプリケーションが含まれる。

鍵材料の復元がそのアプリケーションに必要かどうかを判断するのは組織の責任である。鍵の復元が必要かどうかの決定は、ケースバイケースで行われるべきであり、この決定は、鍵管理ポリシー及び鍵管理実践ステートメントに反映される**べきである**（SP 800-57 パート 2 参照）。鍵復元を提供することを決定した場合、復元する鍵材料のタイプに基づいて、鍵復元の適切な方法を選択、設計、及び実装**すべきである**。適切なエンティティを選択して、バックアップやアーカイブのデータベースを維持し、鍵復元プロセスを管理する必要がある。

鍵の復元を提供することが決定された場合、その鍵に関連する全ての情報も復元可能で**なければならない**。

B.1 保管された鍵材料からの復元

鍵及びその他の鍵情報のバックアップやアーカイブの第一の目的は、当該鍵やその他の鍵情報が利用できなくなった場合に、それらを復元できるようにすることである。例えば、暗号化された情報は、復号鍵が紛失又は変更された場合、平文情報に戻すことができない。データの完全性を検証するために使用される鍵が利用できない場合、当該データの完全性は認証できない。鍵復元プロセスは、バックアッ

プやアーカイブのストレージから鍵材料を取り出し、デバイス、モジュール又はその他のすぐにアクセス可能なストレージに配置するプロセスで、多くの場合、人の手を借りて行われる（8.3.1 節参照）。

B.2 鍵材料の再構築による復元

一部の鍵材料は、他の利用可能な鍵材料である“ベースの”鍵材料（例えば、鍵導出法のための鍵導出鍵）から鍵材料を再構築又は再導出することによって、復元することができる。ベースの鍵材料は、通常運用のストレージ（8.2.1 節参照）、バックアップストレージ（8.2.2.1 節参照）、又はアーカイブストレージ（8.3.1 節参照）で利用可能で**なければならない**。

B.3 鍵材料の復元が必要な条件

鍵復元の可能性がある鍵資料をバックアップ又はアーカイブするかどうかの決定は、ケースバイケースで行う**べきであり**、8.2.2.2 節に記載されているリストに基づく**べきである**。

鍵の所有者から鍵復元処理が要求された場合、以下のアクションが取られ**なければならない**：

1. 紛失した鍵が危殆化した可能性がある場合は、復元した鍵とそれが保護するデータの漏えいを制限するために、当該鍵は復元後できるだけ早く交換し**なければならない**（8.2.3.1 節参照）。これには、新しい鍵を使用して保護されたデータに保護を再適用することも含まれる。
2. 鍵にアクセスできなくなったり、鍵が変更されたりしたものの、危殆化の疑いがない場合は、9.4.2 節で説明するように、当該鍵を復元して使用することができる。

以下の小節では、組織が鍵復元を必要とするか否かを判断する際に役立つ説明を提供する。以下の説明では、鍵の復元可能性のみを扱うが、全ての鍵情報（例えば、鍵に関連するメタデータ）も復元可能で**なければならない**。

B.3.1 署名鍵ペア

署名鍵ペアのプライベート鍵（署名プライベート鍵）は、鍵ペアの所有者が情報にデジタル署名を適用するために使用される。対応する公開鍵（署名検証公開鍵）は、依拠エンティティがデジタル署名を検証するために使用される。

B.3.1.1 署名プライベート鍵

一般的に、署名プライベート鍵はアーカイブ**されてはならない**（8.3.1 節表 9 参照）。署名鍵ペアのプライベート鍵については、通常、鍵バックアップは望ましくない。署名の否認防止のサポートが疑問視されるためである。しかし、例外が存在するかもしれない。例えば、署名プライベート鍵を交換し、それに対応する署名検証公開鍵を（8.1.5.1 節に従って）タイムリーに配付することは、状況によっては不可能な場合があるので、その時は、バックアップストレージから署名プライベート鍵を復元することが正当化されることもある。

これは、例えば、CA の署名プライベート鍵の場合に当てはまるかもしれない。

署名プライベート鍵のバックアップを検討する場合、その重要性及び鍵の復元に必要な時間を評価し、新しい鍵ペアを生成し、新しい署名検証公開鍵を認証して配付するために必要な時間と比較**すべきである**。署名プライベート鍵がバックアップされている場合、当該署名プライベート鍵は、安全性の高い方法で復元**されなければならない**。状況に応じて、その鍵は当面の使用に限って復元される**べきであり**、復元処理後は可能な限り速やかに交換**されなければならない**。

署名プライベート鍵をバックアップする代わりに、第 2 の署名プライベート鍵とそれに対応する公開鍵を生成し、第 1 の署名プライベート鍵が使用できなくなった場合に使用するために、8.1.5.1 節に従ってその公開鍵を配付することも可能である。

B.3.1.2 署名検証公開鍵

対応する署名プライベート鍵によって署名された情報を検証するために、必要な期間に限り、署名検証公開鍵をバックアップ又はアーカイブすることが適切である。(例えば、認証局によって) 認証済みの公開鍵の場合、公開鍵証明書を保存することが公開鍵を保存する適切な形式となる。バックアップ又はアーカイブのストレージは、(例えば、証明書リポジトリなどの) インフラとして提供される場合がある。公開鍵は、プライベート鍵の暗号利用期間が終了するまでバックアップストレージに保存**されるべきであり**、また署名されたデータの検証に必要な期間に限り、アーカイブストレージに保存**されるべきである**。

B.3.2 認証対称鍵

認証対称鍵は、情報の完全性とソースを保証するために使用される。認証対称鍵は、以下の場合に使用することができる：

1. 作成者が、後から検証することができるメッセージ認証コード (MAC) を作成し、その検証により認証された情報の完全性 (及び場合によってはソース) を判断することができる。認証された情報とその MAC は、後に取得するために保存されたり、別のエンティティに送信されたりすることができる
2. エンティティが、認証された情報と MAC をストレージから取得し、保存された情報の完全性を判断する (注：この場合、通信アプリケーションではない)
3. 受信エンティティが受信した場合、直ちに送信された情報の完全性とその情報のソースを判断する (受信した MAC 及び関連する認証された情報は、その後保存される場合もあれば、保存されない場合もある)
4. 受信エンティティ及び取得エンティティが、受領した後に格納した情報の完全性及びソースを、同じ MAC (及び同じ認証鍵) を使用して決定する。MAC のチェックは、格納の前に実行されない場合がある

上記の各ケースについて、鍵復元機能を提供するかどうかの決定は、以下の考慮事項に基づいて行われる**べきである**：

ケース 1 では、作成者は MAC を計算する前に新しい認証鍵を確立でき、後に情報を認証するためにその新しい鍵を使用して検証する必要がある全てのエンティティがその鍵を利用できるならば、認証対称鍵のバックアップ又はアーカイブは不要である。新しい認証鍵をタイムリーに取得できないならば、認証鍵はバックアップ又はアーカイブされる**べきである**。

ケース 2 では、認証対称鍵は、情報の完全性とソースを判断する必要がある期間に限り、バックアップ又はアーカイブされる**べきである**。

ケース 3 では、認証対称鍵は、当該認証鍵を受信者に再送可能であれば、バックアップ又はアーカイブは不要である。この場合、“失われた”鍵を再利用するのではなく、新しい認証対称鍵を確立して配付することも許容される。ただし、新しい MAC は、新しい認証鍵をその情報に対して使用して計算される必要がある。そうでなければ、認証対称鍵はバックアップされる**べきである**。認証鍵をアーカイブすることは、MAC と認証された情報が後に保存されない場合、適切ではない。なぜなら、認証鍵の暗号利用期間が終了すると、MAC の適用と確認の両方で当該鍵の使用が中止されるためである。MAC と認証された情報が後に保存される場合、認証対称鍵は、情報の完全性とソースを判断する必要がある期間に限り、バックアップ又はアーカイブされる**べきである**。

ケース 4 では、認証対称鍵は、情報の完全性とソースを判断する必要がある期間に限り、バックアップ又はアーカイブされる**べきである**。

認証対称鍵は、その鍵の暗号利用期間中、バックアップストレージに格納されてもよいし、不要になるまでアーカイブストレージに格納されてもよい。認証鍵が再構築によって復元される場合、“ベースの”鍵（例えば、鍵導出手法用のマスター鍵／鍵導出鍵）は、当該ベース鍵の暗号利用期間中、通常運用のストレージ又はバックアップストレージに格納されてもよいし、不要になるまでアーカイブストレージに格納されてもよい。

B.3.3 認証鍵ペア

認証公開鍵は、受信エンティティが送信エンティティの身元保証を得るために使用される。対応する認証プライベート鍵は、送信エンティティが、情報にデジタル署名を計算することにより、受信エンティティにこの保証を提供するために使用される。この鍵ペアは、否認防止のサポートを提供しない場合がある。

B.3.3.1 認証公開鍵

認証済み通信セッションに参加しているエンティティの身元検証が必要な期間に限り、認証公開鍵をバックアップ又はアーカイブのストレージのいずれかに保存することが適切である。

認証済み（例えば、認証局によって）の公開鍵の場合、公開鍵証明書を保存することが公開鍵を保存する適切な形式となる。バックアップ又はアーカイブのストレージは、インフラ（例えば、証明書リポジトリ）によって提供される場合がある。公開鍵は、プライベート鍵の暗号利用期間が終了するまでバックアップストレージに保存されてもよく、必要であれば、アーカイブストレージに保存されてもよい。

B.3.3.2 認証プライベート鍵

プライベート鍵は、認証された通信セッションに参加しているエンティティの身元を確立するために使用される。認証プライベート鍵は、8.1.5.1 節に従い新しい鍵ペアをタイムリーに生成して配付できるならば、バックアップは不要である。ただし、新しい鍵ペアを迅速に生成できないならば、当該プライベート鍵の暗号利用期間中、そのプライベート鍵はバックアップストレージに格納**されるべきである**。プライベート鍵は、アーカイブストレージに保存**されてはならない**。

B.3.4 データ暗号化対称鍵

データ暗号化対称鍵は、保存されたデータ、送信されたデータ、又はその両方の機密性を保護するために使用される。平文データを保護するための初めの暗号化と、その後の暗号化されたデータ（すなわち、暗号文）の復号で同じ鍵が使用され、その結果、元の平文が得られる。

その鍵は、当該鍵を使って暗号化されたデータのなかに復号が必要となりうるデータがある限り、利用可能である必要がある。したがって、この鍵はその期間中バックアップ又はアーカイブ**されておくべきである**。

鍵の復元を可能にするために、データ暗号化対称鍵は、当該鍵の暗号利用期間中はバックアップストレージに保存**されるべきであり**、必要に応じてアーカイブストレージに保存**されるべきである**。多くの場合、その鍵は、暗号化されたデータとともに保護されて保存される。アーカイブされる場合、その鍵は、アーカイブ暗号化鍵、又は保護されたアーカイブ暗号化鍵によってラッピングされた鍵ラッピング対称鍵によってラッピングされる（すなわち、暗号化される）べきである。

送信にのみ使用されるデータ暗号化対称鍵は、送信エンティティがデータを暗号化するために使用され、また受信エンティティが受信直後に暗号文データを復号するために使用される。データ暗号化鍵が紛失又は破損した場合であっても、新しいデータ暗号化鍵を送信エンティティと受信エンティティが容易に入手できる場合には、その鍵をバックアップする必要はない。しかし、その鍵を新しい鍵に容易に置き換えることができない場合、交換されるデータが十分に重要なものであるならば、当該鍵はバックアップ**されるべきである**。データ暗号化鍵は、送信のみに使用する場合には、アーカイブする必要はないかもしれない。

B.3.5 鍵ラッピング対称鍵

鍵ラッピング対称鍵は、保護されるべき鍵材料をラッピング（すなわち、暗号化及び完全性保護）するために使用され、複数セットの鍵材料を保護するために使用されうる。その後、保護された鍵材料は、送信、保存、又はその両方が行われる。

鍵ラッピング対称鍵が鍵材料の送信にのみ使用され、鍵ラッピング鍵が利用できなくなった場合（例えば、紛失又は破損した場合）、鍵ラッピング鍵を再送信するか、新しい鍵ラッピング鍵を確立してそれを使用して鍵材料を再送信するかのいずれかが可能であってもよい。合理的な時間枠内で可能であれば、鍵ラッピング鍵のバックアップは必要ない。鍵ラッピング鍵の再送ができない場合や、新しい鍵ラッピング鍵を容易に入手できない場合は、鍵ラッピング鍵のバックアップが検討**されるべきである**。また、鍵材料の送信のみに使用される鍵ラッピング鍵のアーカイブは必要でないかもしれない。

保管中の鍵材料を保護するために鍵ラッピング対称鍵が使用される場合、鍵ラッピング鍵は、保護された鍵材料にアクセスする必要がある期間に限り、バックアップ又はアーカイブ**されるべきである**。

B.3.6 乱数生成鍵

乱数ビット生成に使用した鍵は、バックアップ又はアーカイブを**されてはならない**。この鍵が紛失又は変更された場合は、新しい鍵に置き換え**られなければならない**。

B.3.7 マスター対称鍵／鍵導出対称鍵

マスター対称鍵／鍵導出対称鍵は、通常、1 つ以上の他の鍵を導出するために使用される。その鍵はそれ以外の目的で使用**されてはならない**。

マスター対称鍵／鍵導出対称鍵をバックアップ又はアーカイブする必要があるかどうかの判断は、多くの要因に依存する：

1. 新しい対称鍵を確立するのはどれくらい簡単か？ 鍵が手動で配付されている場合（例えば、送達確認郵便によって配付されるスマートカードやハードコピーの中に）においては、当該鍵はバックアップ又はアーカイブ**されるべきである**。自動化された鍵確立プロトコルを使用して新しい鍵を簡単かつ迅速に確立できる場合、アプリケーションによっては、鍵のバックアップやアーカイブは必要ないか、又は望ましいものではないかもしれない。
2. 導出鍵は対称鍵を使用しなくても復元可能か？ 導出鍵がバックアップ又はアーカイブされる必要がない場合（例えば、その用途が理由で）、又は導出鍵の復元がマスター鍵／鍵導出鍵からの再構築に依存しない場合（例えば、導出鍵が暗号化された形で保存されている場合）は、鍵のバックアップ又はアーカイブは望ましくないかもしれない。導出鍵をバックアップ又はアーカイブする必要があり、かつ鍵の復元方法がマスター鍵／鍵導出鍵から導出鍵の再構築を必要とする場合は、マスター鍵はバックアップ又はアーカイブ**されるべきである**。

B.3.8 鍵配送鍵ペア

鍵配送鍵ペアは、通信中に送信エンティティから受信エンティティに鍵材料を配送するために使用される。配送された鍵材料は、暗号化された形で保存され、後で復号することができる。通信の送信エンティティは公開鍵を使用して鍵材料を暗号化する。受信エンティティ（又は保存された鍵材料を取得するエンティティ）はプライベート鍵を使用して暗号化された鍵材料を復号する。

B.3.8.1 鍵配送プライベート鍵

鍵配送鍵ペアが、暗号化された鍵材料を格納することなく通信中に使用される場合、代替鍵ペアをタイムリーに生成して配付することができれば、鍵配送プライベート鍵のバックアップは不要である。あるいは、1 つ以上の追加の鍵ペアを利用可能にすることもできる（すなわち、既に生成されて配付されている）。そうでなければ、プライベート鍵はバックアップ**されるべきである**。鍵配送プライベート鍵は、アーカイブされてもよい。

配送された鍵材料が暗号化された状態で保存されている場合、保護された鍵材料にアクセスする必要がある期間に限り、鍵配送プライベート鍵はバックアップ又はアーカイブ**されるべきである**。

B.3.8.2 鍵配送公開鍵

公開鍵のバックアップ又はアーカイブは行ってもよいが、必要ない場合もある。

送信エンティティ（通信の送信エンティティ）が鍵配送公開鍵を紛失した場合、又は鍵が破損していると判断した場合、鍵ペアの所有者から、又は公開鍵を含む公開鍵証明書（公開鍵が認証されている場合）を取得することにより、鍵を再取得することができる。

エンティティが鍵材料を保存するために暗号保護を適用し、そのエンティティにより鍵配送公開鍵が紛失又は破損されていると判断した場合、当該エンティティは、以下のいずれかの方法で当該鍵を復元することができる：

1. 公開鍵が認証済みで、インフラ内の別の場所に保存されている場合、証明書を要求できる
2. 他のエンティティ（例えば、鍵ペアの所有者）が公開鍵を知っている場合、鍵を当該エンティティに要求できる
3. プライベート鍵を知っている場合、公開鍵は再計算できる
4. 新しい鍵ペアを生成できる

B.3.9 鍵合意対称鍵

鍵合意対称鍵は、鍵材料（例えば、鍵ラッピング対称鍵、データ暗号化対称鍵、認証対称鍵、IV）を確立するために使用される。各鍵合意鍵は、2 つ以上のエンティティ間で共有される。これらの鍵が手動で配付されている場合（例えば、鍵読み込みデバイス内や送達確認郵便によって)、当該鍵合意対称鍵はバックアップ**されるべきである**。自動化された手段が新しい鍵を迅速に確立するために利用可能である場合（例えば、鍵配送メカニズムを使用して新しい鍵合意対称鍵を確立することができる場合)、鍵合意対称鍵のバックアップは不要である。

鍵合意対称鍵はアーカイブされてもよい。

B.3.10 静的鍵合意鍵ペア

静的鍵合意鍵ペアは、エンティティ間の共有秘密を確立するために使用される（SP 800-56A 及び SP 800-56B 参照）が、時には一時的鍵ペア（SP 800-56A 参照）と一緒に使用されることもある。各エンティティは、自分の鍵合意プライベート鍵と他のエンティティの鍵合意公開鍵、そして場合によっては自身の鍵合意公開鍵を使用して、共有秘密を決定する。共有秘密は、その後、共有する鍵材料を導出するために使用される。鍵合意スキームのなかには、1 つ以上のエンティティが静的鍵合意鍵ペアを使用しない場合があることに注意されたい（SP 800-56A 及び SP 800-56B 参照)。

B.3.10.1 静的鍵合意プライベート鍵

静的鍵合意プライベート鍵をタイムリーに交換できない場合、又は暗号化された保存データを復元するために当該鍵を保持する必要がある場合、当該プライベート鍵は処理を継続するためにバックアップ**されるべきである**。当該プライベート鍵をアーカイブしてもよい。

B.3.10.2 静的鍵合意公開鍵

エンティティが、静的鍵合意公開鍵を紛失又は破損していると判断した場合、当該エンティティは、以下のいずれかの方法で当該鍵を復元することができる：

1. 公開鍵が認証済みで、インフラ内の別の場所に保存されている場合、証明書を要求できる
2. 他のエンティティが公開鍵を知っている場合（例えば、他のエンティティが鍵ペアの所有者である場合)、鍵を当該エンティティに要求できる

3. プライベート鍵を知っている場合、公開鍵は再計算できる
4. エンティティが鍵ペアの所有者である場合、新しい鍵ペアを生成して配付することができる

これらの選択肢のいずれも不可能な場合、静的鍵合意公開鍵はバックアップ**されるべきである**。当該公開鍵をアーカイブしてもよい。

B.3.11 一時的鍵ペア

一時的鍵合意鍵は、単一の鍵合意トランザクション（例えば、通信セッションの開始時）の間に生成・配付され、再利用**されるべきではない**。これらの鍵ペアは共有秘密を確立するために（多くの場合、静的鍵ペアと組み合わせて）使用される。共有秘密は、その後、共有する鍵材料を導出するために利用される。全ての鍵合意スキームが一時的鍵ペアを使用するわけではなく、使用される場合でも、全てのエンティティが一時的鍵ペアを持つわけではない（SP 800-56A 参照）。

B.3.11.1 一時的プライベート鍵

一時的プライベート鍵をバックアップ又はアーカイブ**してはならない** [114]。一時的プライベート鍵が紛失又は破損した場合、新しい鍵ペアが生成**されなければならず**、新しい一時的公開鍵は鍵合意プロセスに参加している他のエンティティに提供**されなければならない**。

B.3.11.2 一時的公開鍵

一時的公開鍵をバックアップ又はアーカイブしてもよい。これにより、鍵合意の計算で一時的プライベート鍵が必要とされない限り、確立された鍵材料の再構築が可能となる。

B.3.12 認可対称鍵

認可対称鍵は、あるエンティティに特権を与えるために使用される（特定のデータへのアクセス、特定の機能を実行するための認可など）。これらの鍵を紛失すると、特権が拒否される（アクセスの禁止、これらの機能の実行拒否など）。認可鍵が紛失又は破損しても、タイムリーに交換できる場合は、認可鍵のバックアップは不要である。認可対称鍵はアーカイブ**されてはならない**。

B.3.13 認可鍵ペア

認可鍵ペアは、エンティティが想定する特権を決定するために使用される。プライベート鍵は、特権に対する“権利”を確立するために使用される。公開鍵は、そのエンティティが実際にその特権に対する権利を持っているかを判断するために使用される。

---

[114] SP 800-56A では、一時的プライベート鍵は使用後直ちに破棄**しなければならない**としている。これは、一時的プライベート鍵をバックアップ又はアーカイブ**してはならない**ことを意味する。

B.3.13.1 認可プライベート鍵

認可プライベート鍵を紛失すると、特権が拒否される（アクセスの禁止、認可を必要とする特定の機能の実行拒否など）。プライベート鍵が紛失又は破損しても、タイムリーに交換できる場合は、プライベート鍵のバックアップは不要である。そうでない場合は、プライベート鍵はバックアップ**されるべきである**。プライベート鍵はアーカイブ**されてはならない**。

B.3.13.2 認可公開鍵

認可鍵ペアをタイムリーに交換できる場合（すなわち、鍵ペアの再生成、及び認可を求めるエンティティへのプライベート鍵の安全な配付によって）、認可公開鍵のバックアップは不要である。そうでなければ、公開鍵はバックアップ**されるべきである**。認可は、アーカイブされない関連するプライベート鍵を用いてのみ付与されるため、認可公開鍵をアーカイブする必要はない（付録 B.3.13.1 参照）。

B.3.14 その他の関連情報

鍵と同様に、その他の関連情報も、用途によっては、バックアップやアーカイブが必要になる場合がある。

B.3.14.1 ドメインパラメータ

ドメインパラメータは、いくつかの公開鍵アルゴリズムと組み合わせて、鍵ペアを生成するために使用される。また、デジタル署名を作成・検証したり、鍵材料を確立したりするためにも、鍵ペアと一緒に使用される。同じドメインパラメータのセットが、多数のエンティティによって使用されることが多いが、常にというわけではない。

エンティティ（エンティティ A）が新しいドメインパラメータを生成すると、これらのドメインパラメータは、その後のデジタル署名の生成又は鍵確立プロセスで使用される。ドメインパラメータは、デジタル署名を検証する必要がある他のエンティティや、鍵確立する相手に提供する必要がある。エンティティ（エンティティ A）が、ドメインパラメータのコピーが紛失又は破損していると判断した場合に、新しいドメインパラメータをタイムリーに安全な配付ができないならば、ドメインパラメータはバックアップ又はアーカイブ**されるべきである**。

同じドメインパラメータのセットを複数のエンティティが使用する場合、ドメインパラメータは、ドメインパラメータが別の方法で（例えば、信頼できるソースから）入手できない限り、必要なくなるまでバックアップ又はアーカイブ**されるべきである**。

B.3.14.2 初期ベクトル（IV）

IV は、ブロック暗号アルゴリズムを使用したデータの暗号化又は認証の際に、いくつかの暗号利用モードで使用される。IV は保護するデータと一緒に保存されることが多い。データと一緒に保存されない場合、保護されたデータが IV を使用して処理（復号、認証など）する必要がある限り、当該 IV はバックアップ又はアーカイブ**されるべきである**。

B.3.14.3 共有秘密

共有秘密は、鍵合意プロセスに参加する各エンティティによって生成される。共有秘密は、その後の暗号処理で使用される共有鍵材料を導出するために使用される。共有秘密は、双方向通信中（例えば、双方のエンティティがオンラインである場合）に生成されてもよいし、非双方向通信中（保管中、順方向アプリケーションなど）に生成されてもよい。

共有秘密はバックアップ又はアーカイブ**されてはならない**。

B.3.14.4 RBG シード

RBG シードは乱数ビットの生成に使用され、秘密にしておく必要がある。これらのシードは他のエンティティと共有**されてはならず**、またバックアップ又はアーカイブ**されてはならない**。

B.3.14.5 その他の公開情報及び秘密情報

公開情報や秘密情報は、鍵確立時にしばしば使用される。その情報は、暗号的に保護されたデータの処理（復号、認証など）で必要な鍵を決定するために利用可能である必要がある場合がある。そのため、当該情報は、保護されたデータを処理する必要がなくなるまで、バックアップ又はアーカイブ**されるべきである**。

B.3.14.6 中間結果

暗号処理の中間結果はバックアップ又はアーカイブ**されてはならない**。

B.3.14.7 鍵制御情報／メタデータ

鍵制御情報は、例えば、暗号的に保護されたデータを処理（復号、認証など）に使用される鍵やその他の情報を決定したり、鍵の目的を特定したり、鍵を共有するエンティティを特定したりするために使用される（6.2.3 節参照）。この情報は、鍵のメタデータに含まれる（6.2.3.1 節参照）。

鍵管理情報は、関連する鍵が利用可能である必要がある限り、バックアップ又はアーカイブ**されるべきである**。

B.3.14.8 乱数

乱数は乱数発生器によって生成される。乱数のバックアップ又はアーカイブは、その使用方法に依存する。

B.3.14.9 パスワード

パスワードは、エンティティによる特権へのアクセス権の取得、鍵の導出、又はパスワードの再利用の検出のために使用される。

パスワードが特権へのアクセスを取得するためにのみ使用され、タイムリーに交換できる場合は、パスワードをバックアップする必要はない。この場合、パスワードはアーカイブ**されてはならない**。

パスワードが暗号鍵の導出やパスワードの再利用を防止するために使用される場合、パスワードはバックアップ及びアーカイブ**されるべきである**。

B.3.14.10 監査情報

鍵管理イベントを含む監査情報は、バックアップ及びアーカイブ**されなければならない**。

B.4 鍵復元システム

鍵復元は、いくつかの異なる鍵復元技術に適用される可能性のある広義の用語である。各技術は、暗号鍵と、場合によってはその鍵に関連する他の情報（例えば、鍵のメタデータ）の復元を行う。その鍵を復元するために必要な情報は、アプリケーションごと、又は鍵復元技術ごとに異なる場合がある。以下では、“鍵復元情報”（KRI）という用語を、暗号的に保護されたデータを復元又は検証するために必要な鍵情報の集合体を指すために使用する。KRI として考慮され得る情報には、復元されるべき鍵材料又は鍵材料を再構築するのに十分な情報、その他の関連する鍵情報、鍵が作成された時刻、鍵の所有者（すなわち、鍵を作成した又はその鍵によって保護されたデータを所有する個人、アプリケーション、又は組織）に関連する識別子、及び鍵材料を復元できるようにするために要求者が満たさなければならない条件などが含まれる。

組織が、鍵材料の全部又は一部について鍵復元が必要であると判断した場合、十分に規定された鍵復元ポリシー（付録 B.5 参照）に従って、安全な鍵復元システム（KRS）を確立する必要がある。KRS は、鍵復元ポリシーをサポートし**なければならず**、さらに鍵材料の保存及び復元のための技術と設備、システムの管理手順、及びシステムに関連する要員から構成され**なければならない**。

鍵復元が必要であると判断された場合、KRI は、組織内（バックアップストレージ又はアーカイブストレージの中）、又は信頼できるエンティティによるリモートサイトのいずれかに保存されうる。復元を可能にするための多くの許容可能な方法がある。KRS は、金庫を使用した鍵材料の保管によって確立することもでき、単一のコンピュータを使用して平文データの最初の保護、関連する鍵材料の保管、及びその鍵材料の復元を提供することもでき、中央の鍵復元センターを持つコンピュータのネットワークを含むことがあり、又は他の構成を使用して設計することもできる。KRS は暗号鍵を復元するための手段を提供するものであり、KRS が組織の情報を適切に保護し、必要なときに確実に KRI を提供することを保証するための、リスク評価が実施**されるべきである**。鍵復元ポリシー、鍵復元方法、及び鍵復元システムが適切に KRI を保護することを保証するのは、鍵復元を提供する必要がある組織の責任である。

連邦政府が使用する KRS は、以下のことを満たさ**なければならない**：

1. 保護された情報が保存されている場合に、その情報の復元又は検証を可能にするために十分な KRI を生成又は提供する
2. 保存された鍵及びその他の KRI の有効性を保証する
3. KRI が、対応する暗号的に保護されたデータと同等の永続性と可用性を持って保存されていることを保証する
4. FIPS 140 に準拠した暗号モジュールを使用する
5. 暗号を使用する場合は、**承認された**アルゴリズムを使用する
6. KRI に関連する情報の機微性に見合ったセキュリティ強度を提供するアルゴリズム及び鍵長を使用する

7. 鍵復元ポリシー（付録 B.5 参照）を実施するように設計されている
8. KRI を不正な開示又は破棄から保護する。KRS は、要求されかつ認可された情報のみが要求者に提供されるように要求のソースを検証し**なければならない**
9. KRI を改ざんから保護する
10. 監査証跡を提供する機能を有する。監査証跡には、復元される鍵又はシステムで使用されうるあらゆるパスワードが**含まれていてはならない**。また、監査証跡には、監査対象のイベントの識別、イベントの発生時刻、イベントの原因となったエンティティに関連する識別子、及びイベントの成否を含む**べきである**
11. KRI、監査証跡、及び認証データへのアクセスを、認可された個人に限定する
12. 監査証跡の変更を禁止する

B.5 鍵復元ポリシー

使用されるシステム、アプリケーション、及び暗号技術ごとに、後にその鍵材料を復元して当該鍵材料が保護している情報を復号又は検証できるようにするために、当該鍵材料を保存する必要があるかどうかを考慮し**なければならない**。鍵材料の一部又は全部の鍵復元が必要であると判断した組織は、その情報の保護と継続的なアクセス可能性に対処するための鍵復元ポリシー[115]を策定**すべきである**（例として、FPKI-KRP[116]を参照のこと）。このポリシーは、（最低でも）以下の設問に答える**べきである**：

1. 与えられたアプリケーションに対して、どの鍵材料を保存する必要があるか？ 例えば、保存された情報の復号に使用される鍵及び IV を保存する必要があるかもしれない。また、保存された情報又は送信された情報の認証に使用される鍵も保存する必要があるかもしれない。
2. 鍵材料は、どのようにして、どこに保存されるのか？ 例えば、鍵材料は、データの保護（例えば、暗号化されたデータ）を開始した個人によって金庫に保存されてもよいし、保護されたデータが送信、受信又は保存されるときに、鍵材料が自動的に保存されてもよい。鍵材料は、ローカルに保存されてもよいし、リモートサイトに保存されてもよい。
3. 誰が KRI を保護する責任を負うか？ 例えば、各個人、組織又は下位組織が、それぞれの鍵材料に責任を持つこともできるし、外部組織がこの機能を実行することもできる。
4. 要求に応じて、誰がどのような条件で KRI を受け取る権限を持っているか？ 例えば、情報を保護した個人（すなわち、KRI を使用し、保存した個人）、又はその個人が割り当てられた組織は、鍵材料を復元しうる。法的要件を考慮する必要があるかもしれない。組織は、KRI を保存した本人が対応できない場合に、情報を要求することができるかもしれない。
5. どのような条件の下で、誰がポリシーを変更できるか？
6. どのような監査機能及び手順を KRS に含むのか？ ポリシーは、監査対象となるイベントを特定し**なければならない**。監査可能なイベントには、KRI 要求とそれに関連する応答が含まれる場合がある。例えば、誰がいつ要求を行ったか、監査機能の起動及び停止、監査データの読み取り、変更又は破棄のために実行された操作、エンティティ認証データへのアクセス要求、認証メカニズムの使用、などがある。

---

[115] PKI の場合、組織の鍵復元ポリシーは、PKI 証明書ポリシーに含まれることがある。

[116] FPKI-KRP, *Federal Public Key Infrastructure Key Recovery Policy*.

7. KRS は、セキュリティ強度が許容レベル以下に低下した経年劣化した鍵材料について、どのように対処するのか？
8. 鍵材料が復元された場合、誰に、どのような条件で通知されるか？　例えば、データを暗号化して KRI を格納した個人に対して、その人が不在だったために組織が復号鍵を復元した場合は通知される可能性があるが、組織がその人の活動を監視している場合には通知されない可能性がある。
9. KRS 又は KRS 内のデータの一部が危殆化した場合、どのような手順を踏む必要があるか？

付録 C - 改訂履歴

本文書の初版は、2005 年 8 月に発行された。以下の改訂版が、その後に発行されている：

- 改訂 1 版（2006 年）
- 改訂 2 版（2007 年）
- 改訂 3 版（2011 年）
- 改訂 4 版（2015 年）
- 改訂 5 版（2019 年）

SP 800-57 パート 1 の古いバージョンは、https://csrc.nist.gov/publications の“withdrawn publications”で検索することで入手可能である。

C.1 改訂 1 版（2006）

1. *セキュリティ強度* の定義を修正し、“またはセキュリティレベル”という用語が本文書で使用されていないため、最初の列から削除した。
2. 5.6.1 節の表 2 における 2TDEA の脚注において、“*保証*（guarantee）”の語句を“*評価*（assessment）”に変更した。
3. 5.6.1 節の表 2 の下の段落において、2006 年改訂版で最初に特定された変更は、後述する 2011 年改訂版によって置き換えられている。
4. 5.6.1 節の表 3 において、適切なハッシュ関数の一覧に、HMAC 及び鍵導出関数の行が挿入された。さらに、脚注が鍵導出関数の列に付けられた。
5. 表 3 の直下の段落の初期の文章が削除された。

C.2 改訂 2 版（2007）

2007 年 3 月に、以下の改訂がなされ、証明書要求中の鍵の二重使用を許容するようになった：

1. 5.2 節では、以下の文章が追加された：

   “本推奨は、以下の特別な場合のデジタル署名を生成するために、鍵配送プライベート鍵又は鍵合意プライベート鍵の使用も許可する：

   静的鍵確立鍵の（初期の）証明書を要求する場合、関連するプライベート鍵を使用して証明書要求に署名してもよい。8.1.5.1.1.2 節も参照のこと。”

2. 8.1.5.1.1.2 節の第 4 段落は、元々以下のとおりであった：

   “所有者は、示された鍵用途を満たすプライベート鍵を用いて処理を実行することによって、POP を提供する。例えば、鍵ペアが鍵配送をサポートするように意図されている場合、所有者は、当該所有者の公開鍵を使用して暗号化され、CA によって所有者に提供された鍵を復号してもよい。所有者が、対応するプライベート鍵を用いて暗号化された鍵を正しく復号することができ、かつその鍵が正しく復号されたという証拠を提供することができる場合（例えば、CA からのランダムなチャレンジを暗号化することによって）、その所有者は POP を

確立したことになる。鍵ペアが鍵確立をサポートするように意図されている場合、POP は、当該鍵ペアを用いたデジタル署名を生成及び検証することによって提供**されてはならない**。"

本段落は、以下のとおり変更され、変更された文章がイタリック体で示されている：

"*(評判がある)*所有者が、示された鍵用途を満たすプライベート鍵を用いて処理を実行することによって、POP を*提供する**べきである***。例えば、鍵ペアが *RSA* 鍵配送をサポートするように意図されている場合、*CA* は、所有者の公開鍵を使用して暗号化された*鍵を所有者に提供してもよい*。所有者が、対応するプライベート鍵を用いて暗号化された鍵を正しく復号することができ、かつその鍵が正しく復号されたという証拠を提供することができる場合（例えば、CA からのランダムなチャレンジを暗号化することによって）、その所有者は POP を確立したことになる。*しかし、鍵ペアが鍵確立をサポートするように意図されている場合、POP も、証明書要求にデジタル署名するためにプライベート鍵を使用することによって提供されることもある（これは推奨方法ではないが）。鍵確立プライベート鍵（即ち、鍵合意プライベート鍵又は鍵配送プライベート鍵）は、証明書発行後に署名処理の実行に使用**されてはならない***。"

C.3 改訂 3 版（2011）

1. オーソリティの節が更新された。
2. 1.2 節では、SP 800-57 パート 3 の記述が当該文書に従い修正された。
3. 2.1 節では、*鍵導出関数*、*鍵導出鍵*、*鍵長*、*鍵サイズ*、*乱数ビット生成器*、及び *ユーザ* の定義が追加された。

   アーカイブ、*鍵管理アーカイブ*、*鍵復元*、*ラベル*、*所有者*、*プライベート鍵*、*所持証明*、*公開鍵*、*データのセキュリティ寿命*、*シード*、*共有秘密*、及び"*should*"の定義が修正された。

   暗号モジュール の定義が削除された。
4. 2.2 節では、RBG の頭字語が追加された。
5. FIPS 180-3、FIPS 186-3、SP 800-38、SP 800-56A、SP 800-56B、SP 800-56C、SP 800-89、SP 800-90、SP 800-107、SP 800-108、SP 800-131A、SP 800-132 及び SP 800-133 への参照が訂正又は挿入された。
6. 4.2.4 節では、2 種類の一般的なデジタル署名及び本推奨の焦点についての脚注が追加された。
7. 4.2.5 節、4.2.5.3 節、4.2.5.5 節及び 5.3 節では、SP 800-56B についての説明が含まれた。
8. 5.1.1 節では、署名プライベート鍵、署名検証公開鍵、認証対称鍵、認証プライベート鍵及び認証公開鍵の定義が、システムやプロトコルでの現在の用途を反映するために修正された。この変更は、文書全般にわたっている。
9. 5.1.2 節の項目 3 では、共有秘密の記述が修正され、共有秘密は暗号鍵であるように保護され、取り扱われるべきであることを明記した。
10. 5.1.2 節、5.3.7 節、6.1.2 節（表 5）、8.1.5.3.4 節、8.1.5.3.5 節、8.2.2.1 節（表 7）及び 8.3.1 節（表 9）では、"その他の秘密情報"が、その他の暗号学的又は関連する情報の一覧に追加された。
11. 5.3.1 節では、追加のリスク要因に、要員の離職が挿入された。

12. 5.3.4 節では、公開鍵の暗号利用期間と証明書の有効期限の間の違いを明確化する記述が挿入された。
13. 5.3.6 節では、推奨された暗号利用期間よりも長い、又は短いものが正当化される場合があることを強調する記述が挿入された。また、一時的鍵合意公開鍵の暗号利用期間についてのさらなる説明が追加された。
14. 5.4.4 節では、プライベート鍵保有の所有者の保証についての説明が追加された。
15. 5.5 節では、CA の署名プライベート鍵の危殆化についての記述が追加され、そのようなイベントの対応についてのアドバイスが提供された。
16. 5.6.1 節では、表 3 及び表の前の文章を明確化するために修正した。追加の脚注を表項目に挿入し、SHA-1 が提供するセキュリティ強度に関する脚注を修正して、デジタル署名アプリケーションのセキュリティ強度が依然として推定の対象であることを示した。
17. 5.6.2 節～5.6.4 節では、表 4 及びその前の文章を SP 800-131A と一致するように修正した。また、例も修正した。
18. 5.6.5 節は、新しい節として、計算能力又は暗号解読の向上によるセキュリティ強度の低下に関連する影響に対処するために追加された。
19. 7 節、7.1 節、7.2 節及び 7.3 節では、状態及びそれらの遷移の記述を書き換え、具体的な動作を要求した（例えば、“is”や“are（である）”などを使った事実の記述を含むというより、“**shall（しなければならない）**”や“**shall not（してはならない）**”の記述を使用した）。
20. 7.3 節では、鍵配送プライベート鍵及び一時的鍵合意プライベート鍵の遷移の説明を追加した。鍵合意プライベート鍵及び鍵合意公開鍵についての以前の説明は、静的鍵合意プライベート鍵及び静的鍵合意公開鍵、並びに一時的鍵合意公開鍵についての説明に変更された。
21. 8.1.5.3.4 節は、SP 800-90A との整合性を高めるために改訂された。
22. 8.1.5.3.7 節及び 8.1.5.3.8 節では、乱数及びパスワードの配付についての説明が挿入された。
23. 8.1.6 節では、鍵が登録されるかされないかについて示す文章が挿入された。
24. 8.2.4 節は、SP 800-56A、SP 800-56B、SP 800-56C、SP 800-108 及び SP 800-132 と一致するように改訂された。
25. 8.3.1 節の表 9 において、静的鍵合意鍵をアーカイブすることは OK と示すように修正された。
26. 8.3.1 節、9.3.2 節、及び付録 B、B.1、B.3、B.3.1.2、B.3.2、B.3.4、B.3.5 及び B.3.10.2 を変更し、アーカイブが鍵の暗号利用期間の終了後にのみ実行されるような表現（例えば、鍵は活性化時に直ちにアーカイブされる可能性がある）、及びアーカイブにおける鍵が過去への影響力のためにのみあるような表現（例えば、鍵の暗号利用期間のずっと後にデータを復号するために当該鍵が必要とされるかもしれない）を削除した。
27. 8.3.3 節では、危殆化した鍵及び危殆化していない鍵の登録解除についての説明が修正された。
28. 8.3.5 節では、PKI システム及び対称鍵システムにおける失効の実現方法についての説明が追加された。
29. 付録 B.14.9 は、SP 800-132 と一致するように改訂された。
30. FIPS への参照のタグを修正し、バージョン番号を削除した。バージョン番号は付録 C に記載されている。

C.4 改訂 4 版（2015）

1. SP 800-21 の参照先を SP 800-175 に変更した。
2. ウェブサイトへのリンクを訂正した。
3. 1.4 節では、FIPS 及び NIST 推奨への参照を“NIST 標準”と新たに表記し、暗号ツールキットの概念を説明した（脚注で）。
4. 2.1 節では、以下の定義を修正した：*アルゴリズムの作成者使用期間*、*アーカイブ*、*認証*、*認証コード*、*認証局*、*DRBG*、*デジタル署名*、*鍵導出*、*鍵暗号化鍵*、*鍵管理ポリシー*、*鍵配送*、*鍵更新*、*鍵ラッピング*、*鍵ラッピング鍵*、*メッセージ認証コード*、*否認防止*、*所有者*、*受領者使用期間*、*RBG シード*、*セキュア通信プロトコル*、*セキュリティサービス*、*署名生成*、*署名検証*、*ソース認証*、及び *トラストアンカー*

   以下の定義を追加した：*データ暗号化鍵*、*ID 認証*、*完全性認証*、*完全性保護*、*鍵導出方法*、*鍵長*、*NIST 標準*、及び *ソース認証*

   以下の定義を削除した：*鍵属性*、*及び作業*
5. 2.2 節では、該当する刊行物を参照した。
6. “属性”に関する多くの記述を、SP 800-152 での説明に合わせて、“メタデータ”に変更した。
7. 3 節及び文書全体において、認証を完全性認証又はソース認証のいずれかとして、より明確に説明するように修正した。ID 認証は、ソース認証と見なされるようになった。
8. 3.3 節は、完全性認証又はソース認証についてより明確に説明するように書き直された。
9. 3.4 節は、認可の取得方法についてより明確に説明するように書き直された。
10. 3.5 節では、否認防止に関するより現実的な説明を提供するように書き直された（すなわち、実際に否認防止を提供するのではなく、否認防止のサポートについて説明する）。文書中の否認防止への言及は、否認防止のサポートについて説明するように書き直された。
11. 3.6 節では、鍵設定サービス及び乱数生成サービスの説明を追加した。
12. 3.7 節では、例を書き換えた。

    FIPS 202 及び FIPS 180 への言及を追加した。
13. 4.1 節では、SP 800-90A で規定されている Dual_EC_DRBG への言及を削除した。
14. 4.2.2.2 節では、2015 年以降の保護適用に 2-key TDEA が非承認になる（SP 800-131A で示されている）のに対応するように書き直した。
15. 4.2.2.3 節に、ECB モードを使用しない根拠を挿入した。
16. 4.2.3.1 節では、CMAC と GMAC の両方の暗号利用モードを参照するように改訂した。
17. 4.2.4 節では、FIPS 186 に関するより多くの情報を提供するように書き直された。
18. 4.2.5.1 節は、SP 800-56A のさらなる説明を含む。
19. 4.2.5.3 節では、鍵確立スキームのセキュリティ特性の説明のために、SP 800-56A 及び SP 800-56B への参照を追加した。
20. 4.2.5.4 節を書き換え、本文書での“鍵ラッピング”及び“鍵暗号化”の使用を明確化した。
21. 4.2.7 節を書き換え、SP 800-90A、SP 800-90B、及び SP 800-90C を記述した。

22. 5.1.1 節において、データ暗号化対称鍵、鍵ラッピング対称鍵、及び鍵配送公開鍵についての更なる詳細を追加した。

    認証プライベート鍵及び認証公開鍵への目的の注釈を追加した。

23. 5.2 節では、1 行目の“should（するべきである）”の使用を“shall（しなければならない）”に変更し、鍵を複数の目的で使用してはならないことをより強く示すようにした。“should”の使用は、本文書の後の説明と矛盾していた。

24. 5.3.1 節では、量子コンピュータへの言及をリストに追加した。

25. 5.3.2 節では、鍵更新に関する言及を削除した。

    5.3.4 節では、非対称鍵ペアの作成者使用期間及び受領者使用期間についての説明を書き換えた。

26. 5.3.6 節では、以下についての暗号利用期間を更に明確に追記した。署名プライベート鍵（脚注）、署名検証公開鍵、認証プライベート鍵（脚注）、認証公開鍵（脚注）、認証対称鍵、鍵合意対称鍵、鍵ラッピング対称鍵、RBG 対称鍵、鍵配送公開鍵、及び静的鍵合意プライベート鍵。

    データ暗号化対称鍵及び鍵ラッピング対称鍵についての説明を表 1 と一致するように修正した。

    表 1 のヘッダは、作成者使用期間及び受領者使用期間を参照するように修正された。明確化のため、鍵合意対称鍵への注釈を追加した。

27. 5.4.2 節では、ドメインパラメータの有効性の保証の取得に関する追加情報を挿入した。

28. 5.4.3 節では、公開鍵の有効性の保証の取得に関する追加情報を挿入した。

29. 5.4.4 節において、プライベート鍵所持の保証を得ることについての詳細は、SP 800-89 で説明されているため、削除した。この保証が CA によって取得される可能性があるという注釈が追加された。

30. 5.4.5 節では、鍵確認についての簡単な説明を追加した。

31. 5.5 節では、不要な文章を削除した。

32. 5.6.1 節では、セキュリティ強度の説明を改訂し、SP 800-158 への参照を挿入した。

    不要となったブロック長についての注記を削除した。

    表 2 は、暗号保護を適用するのに承認されなくなった鍵長、承認されている鍵長、及び FIPS 標準では承認されているが明確に言及されていない鍵長についてすぐにわかる形で提供するために改訂された。SHA-1 についての注釈が修正された。

    表 3 及び以下の文章は、SHA-1 がもはやデジタル署名の生成に承認されなくなったことを明確に示すために改訂された。SHA-3 ハッシュ関数が新たに表に含められた。HMAC のヘッダに注釈が追加された。

33. 5.6.2 節の表 4 が更新され、現在予測されているセキュリティ強度の時間枠を示すようになった。

34. 5.6.3 節では、鍵の実際のセキュリティ強度の決定についての説明のために、SP 800-158 への参照が挿入された。この説明は、当該鍵がどのように生成され、その後どのように処理されたかに基づく。

35. 6.1 節では、SP 800-152 と一致するように、完全性保護及び機密性保護のトピックに変更が加えられた。完全性保護のトピックでは、“完全性保護は暗号学的完全性メカニズムによって提

供できる”が、“完全性保護は暗号学的完全性メカニズムによって提供され**なければならない**”に変更された。

36. 6.2 節では、“使用中”の状態が導入され、同時に鍵が通信中や保管中である可能性もあると認識される。
37. 6.2.1.3 節では、鍵コンポーネントの生成についての追加ガイダンスを加えた。
38. 6.2.2.1 節では、鍵の利用可能性が望まれないケースについて言及し、暗号化消去を説明する刊行物への参照を提供するための段落を追加した。
39. 6.2.2.3 節では、SP 800-152 に準拠した FIPS 140-2 のセキュリティレベルに対応するための追加の文章が挿入された。
40. 6.2.3.1 節では、鍵の履歴が潜在的なメタデータ項目として挿入された。SP 800-158 への参照が、メタデータの取扱いについてのガイダンスを提供するために含まれた。
41. 7 節は完全に書き直され、一時停止状態の追加や異なる鍵タイプの遷移の明確化などが含まれた。一時停止状態が、図 3 及び説明に追加された。
42. 8 節では、一時停止状態が説明に追加され、図 5 に含まれた。
43. 8.1.5 節では、SP 800-133 への参照を追加した。
44. 8.1.5.1 節では、2 段落目の最後に、組織のサブエンティティへの鍵材料の配付についての一文を追加した。
45. 8.1.5.1.1.1 節は改訂され、トラストアンカーとは何か（すなわち、CA のことであり、その CA の証明書ではない）を明確かつより正確に記述するようにした。
46. 8.1.5.1.2 節では、SP 800-56B への言及が削除された。一時的鍵を使用するスキームを含まないためである。
47. 8.1.5.2 節、8.1.5.2.2 節及び 8.2.3.2 節では、鍵更新の使用を鍵変更のための承認された方法として言及している部分が削除又は修正された。
48. 8.1.5.2.2.2 節では、SP 800-38F、SP 800-56A、及び SP 800-56B への参照が追加された。認証付き暗号利用モードを記述するための注釈が追加された。
49. 8.1.5.2.3 節では、鍵ラッピングの記述が削除された。それは鍵合意スキームで使用されないためである。
50. 8.1.5.3.4 節を書き換えた。
51. 8.2.1.1 節及び 8.2.1.2 節では、適切な参照先は暗号モジュールであるため、“デバイス”の記述を削除した。
52. 8.2.3.2 節において、SP 800-152 に記載されているように、鍵更新は現在許可されていない。
53. 8.3.1 節では、アーカイブの使用について更なるガイダンスが提供された。
54. 8.3.4 節では、破棄された鍵を含むメディアの破壊ではなく、鍵の破棄について説明する文章に修正した。
55. 8.3.5 節第 6 段落において、“対応する公開鍵証明書は失効される**べきである**”を“対応する公開鍵証明書はできるだけ早く失効され**なければならない**。”に変更し、失効した証明書の使用について更なるガイダンスを提供した。
56. 10 節では、SP 800-130 及び SP 800-152 への参照を追加した。

57. 10.2.7 節では、ID ベースの特権に関する参照を追加した。
58. 付録 B.3 において、決定項目の最初のリストは、重複を避けるため、8.2.2.2 節への参照に置き換えられた。
59. 付録 B.3.3.1 において、最初の文は、“真正性を検証する”ではなく“エンティティの身元を検証する”を使用して書き直された。
60. 付録 B.3.3.2 は書き換えられた。
61. 付録 B.3.4 及び B.3.5 において、セキュリティ強度に関する文章は、本節には不適切であるため、削除された。
62. 付録 C では、参照先を更新した。例えば、FIPS 202、SP 800-38G、SP 800-90、SP 800-130 及び SP 800-152 の追加など。

C.5 改訂 5 版（2020）

1. 抄録を拡大し、キーワードの数を減らした。
2. 鍵材料に関連するメタデータをより強調した。
3. 必要に応じて、*鍵材料 を 鍵* に変更する例が多くあった（その逆もある）。
4. *情報* と現れた多くが *データ* に置き換えられた（*鍵情報* という用語との混同を避けるため）。
5. 文書全体で、“鍵コンポーネント”を“鍵シェア”に変更した。
6. 文書全体で、文章中の明確化を行った（例：3.7 節第 2 段落の例）。
7. 脚注に文書名を追加した。
8. 1.4 節の項目 6 は、検証プログラムの変更に伴い、修正された。
9. 2 節では、*暗号鍵*、*鍵材料*、*メタデータ* 及び *鍵情報* についての説明と比較を追加した。また、これらの用語の多くの用途を参照のこと。
10. 2.1 節での以下の定義を変更した：*アクセス制御*、*説明責任*、*アルゴリズムのセキュリティ寿命*、*（プライベート鍵の）所持の保証*、*認証*、*危殆化*、*機密性*、*暗号鍵管理システム*、*暗号モジュール*、*デジタル署名*、*ドメインパラメータ*、*一時的鍵*、*完全性認証*、*完全性保護*、*鍵合意*、*鍵登録解除*、*鍵導出*、*鍵導出機能*、*鍵破棄*、*鍵配付*、*鍵確立*、*鍵管理*、*鍵所有者*、*鍵復元*、*鍵登録*、*鍵失効*、*鍵配送*、*鍵更新*、*鍵ラッピング鍵*、*鍵材料*、*メタデータ*、*否認防止*、*運用段階*、*（鍵ペアの）所有者*、*公開鍵*、*秘密鍵*、*セキュリティサービス*、*自己署名証明書*、*ソース認証*、*知識分割*、*対称鍵*、*対称鍵アルゴリズム*、*システム初期化*

    以下の定義を追加した：*活性化状態*、*ブロック暗号*、*危殆化状態*、*非活性化状態*、*破棄状態*、*決定論的乱数ビット生成器*、*エンティティ登録*、*ID 認証*、*鍵情報*、*鍵棚卸リスト*、*鍵棚卸リスト管理*、*鍵シェア*、*鍵状態*、*KMAC*、*（証明書の）所有者*、*秘密鍵アルゴリズム*、*秘密鍵情報*、*セキュリティ機能*、*（証明書の）保証人*、*（鍵の）保証人*、*一時停止状態*、*システム*、*ユーザ初期化*、*ユーザ登録*

    以下の定義を削除した：*暗号鍵コンポーネント*、*鍵コンポーネント*
11. 2.2 節では、EdDSA を追加した。ECDSA を修正した。
12. 3 節では、*ID 認証* をサービスとして追加した。文書全体での使用を参照されたい。

13. 3.1 節では、ソース認証及び ID 認証について説明する文章を追加した。
14. 3.3 節では、ID 認証を追加し、ソース認証と比較した。
15. 3.4 節では、“認証サービス”の参照が“ID 認証”に変更されている。
16. 3.6 節では、鍵確立及び乱数生成の必要性を説明する文章を追加した。
17. 3.7 節において、SP 800-175B への参照を、本文書内のここ及びその他の場所に挿入した。鍵確立をリストの 4 番目の項目として追加した。例を書き換えた。
18. 4 節において、耐量子問題に対する将来の移行について警告する注記が挿入された。SP 800-175B が利用可能になったため、本節はアルゴリズム情報についてその文書を参照するように書き直された。4.1 節及び 4.2 節に追加のアルゴリズムの用途を追記した（本文書の以前のバージョンでは省略されていた）。
19. 4.2 節及び 4.3 節において、対称アルゴリズム及び非対称アルゴリズムを規定する標準を示す脚注を挿入した。鍵生成には乱数処理を用いることを要件に加えた。
20. 5.1 節において、鍵生成に関する SP 800-133 への参照を追加した。
21. 5.1.1 節では、以下の説明を修正した：認証対称鍵（ソース認証ではなく ID 認証に使用される）、認証プライベート鍵、認証公開鍵、マスター対称鍵（鍵導出鍵と呼ばれることも認める。この変更は文書全体を通して行われている）。
22. 5.1.1 節及び 5.1.2 節では、鍵タイプ又はその他の情報が使用される標準を示すための脚注を追加した。
23. 5.1.2 節では、“Cryptographic or”をタイトルから削除した。項目 3 の FIPS 140 に関する一文が削除された（本文書の後段で言及する）。項目 8 では、鍵制御情報がメタデータの一部であることを明確にした。ID 認証が項目 10 に追加された。本節の全ての要件（**shall** 文及び **should** 文）は削除された。表 6 で提供されているためである。
24. 5.2 節（最後の文）及び 8.1.5.1.1.2 節（第 3 段落）では、文章が修正され、証明書要求の署名に使用できる秘密鍵は FIPS 186 での方法を使用して生成されたものだけに限定された。例えば、SP 800-56A で現在許可されているグループは、署名アプリケーションで使用すべきではない。
25. 5.3 節では、鍵が危殆化した場合、暗号利用期間はもはや有効とは見な**してはならない**という文言を挿入した。
26. 5.3.1 節の項目 8 では、“鍵更新”を“鍵再設定”に変更した。
27. 5.3.2 節では、鍵更新への言及を削除した。この利用方法は、政府のアプリケーションでは認められていないためである（8.2.3.2 節参照）。
28. 5.3.6 節の認証プライベート鍵の脚注において、否認防止に関する言及を削除した。SP 800-14 への参照は、それが廃止されたので、削除された。
29. 5.4.4 節では、第 1 段落に、プライベート鍵の所有者及び公開鍵の受領者がプライベート鍵の所持証明を取得することについての追加の文章を加えた。
30. 5.5 節を 2 つの小節に分割した。5.5 節では、第 1 段落に、鍵を保護する責任があるのは誰か、及び鍵の危殆化を報告する責任があるのは誰かを示すための新しい文章を挿入した。5.5.2 節において、項目 4 には“保護された完全性”が新たに含まれ、項目 5 には鍵確認への言及が新たに含まれた。また、SP 800-152 への参照も追加された。5.5.2 節では、項目 a が修正され、通知の内容を指定することが要求された。

31. 5.6 節の 2 行目において、“又は、採用された”を追加した。

32. 5.6.1 節は大幅に改訂され、ブロック暗号及び非対称アルゴリズムのセキュリティ強度を説明する節と、ハッシュ関数及びハッシュ関数が関連する関数のセキュリティ強度を説明するための節とに分割された（5.6.1.1 節及び 5.6.1.2 節）。将来的に対処すべき耐量子問題についての注記が追加された。

    5.6.1.1 節の第 1 節では、鍵長及びそれが生成時に満たさなければならない基準について言及するように修正された 2 列目の記述では、ブロック暗号ベースの関数のセキュリティ強度を説明する文章を挿入した。また、3TDEA の非推奨状態についての脚注を挿入した（表 2 参照）。表の下に、表に記載されている鍵長についての注記を追加した。

    第 3 段落に、攻撃者の能力を記述する文章を追加した。

33. 5.6.1.2 節では、ハッシュ関数のセキュリティ特性についての文章を挿入した。KMAC128 及び KMAC256 を表 3 に追加した。表の理解を容易にするための追加の文章を加えた。

34. 5.6.2 節（旧 5.6.3 節）では、タイトルに“有効なセキュリティ強度（Effective Security Strength）”を含むように変更した。本節は書き換えられ、本節の冒頭に入門的な段落、及びブロック暗号アルゴリズムの使用や鍵の取り扱いを検討する際にブロック長を考慮することについての段落が含まれた。

35. 5.6.3 節（旧 5.6.2 節）の文章を書き換えられたが、表は基本的に同じである。

36. 5.6.4 節では、冒頭に遷移の考え方についての段落を挿入し、例示の段落を修正した。項目 3（システム設計）に柔軟性に関する文言を新たに追加した。

37. 6 節では、タイトルを変更し、*鍵情報* に関する説明に対応させた（より具体性の低い *暗号情報* という用語ではなく）。これは、鍵に関連するメタデータを保護する必要性に対する意識を高めることを意図している。

38. 6.1 節では、*秘密鍵情報* という用語を導入する。これは、鍵情報の秘密部分（すなわち、対称鍵、プライベート鍵、鍵コンポーネント、及び秘密のメタデータ）を意味し、機密性保護の説明で使用する。可用性の保証が、保証要求のリストに追加された。

39. 6.1.1 節の表 5 において、認証対称鍵、認証プライベート鍵及び認証公開鍵の第 2 列の記載を“ソース認証”から“ID 認証”に変更した。表の記載の一部に“可用性”を挿入した。

40. 6.1.2 節のパスワードの行において、“ソース認証”を“ID 認証”に変更した。表の記載の一部に“可用性”を挿入した。

41. 6.2.1.2 節では、完全性保護の方法を明確化するように書き換えた。

42. 6.2.1.3 節では、“鍵材料（対称鍵、プライベート鍵、鍵コンポーネント、及び秘密のメタデータ）”を、秘密にしておくべき項目のリストに変更した。

43. 6.2.3 節では、メタデータの小節を 1 つの節に統合した。追加のメタデータ要素をリストに追加した。

44. 7 節では、不許可の鍵の使用についての一文が追加された。遷移 7 において、一時停止状態への遷移の 2 つ目の可能性のある理由を、可能性のある理由とともに追加した。

45. 7 節では、第 3 段落から第 5 段落に追加の明確化を行った。

    遷移 12 において、“しかし、受領者使用期間の終了に達していない”を第 2 段落の最後に追加した。

46. 8.1.1 節を修正し、身元証明プロセスの説明について SP 800-63 及び FIPS 201 により依存するようにした。
47. 8.1.5 節では、鍵の所有者を定義する。
48. 8.1.5.1.1.2 節において、FIPS 186 で規定されているように鍵合意鍵が生成された場合にのみ、POP を新たに取得することができるようになった。
49. 8.1.5.1.1.3 節では、タイトルを修正し、当該節が静的公開鍵を説明することを示した。
50. 8.1.5.2.2.1 節では、鍵シェアの取り扱いに関して、知識分割手続きに言及する。
51. 8.1.5.2.2.2 節では、第 1 段落から“前（prior）”という文言を削除した。鍵ラッピングを解くために必要なときに利用可能である限り、鍵ラッピング鍵はいつでも送信できる。項目 1 では、“鍵合意スキーム”を“鍵確立スキーム”に変更した。鍵ラッピング鍵は、鍵合意スキーム又は鍵配送スキームのいずれかを使用して確立することができる。
52. 8.1.5.2 節では、SP 800-56A 及び SP 800-56B に合わせるため、“**すべきではない**（should not）”を“**してはならない**（shall not）”に変更した。
53. 8.1.5.3.3 節において、共有秘密を配付**してはならない**という要件を追加した。
54. 8.1.5.3.4 節において、シードの送信に関する最後の文を削除した。シードの取り扱いについては、SP 800-90 シリーズで説明されている。
55. 8.2.4 節の項目 3 の最後の行において、“完全性（integrity）”を“身元（identity）”に変更した。
56. 8.3.1 節の第 6 段落を書き換えた。

    表 9 において、3 列目に鍵配送公開鍵、鍵合意対称鍵、静的鍵合意プライベート鍵、一時的鍵合意公開鍵、及び認可公開鍵についての文言を追加した。
57. 8.3.3 節において、鍵の再登録が可能な場合及び再登録をしてはならない場合を示す条件を挿入した。
58. 8.3.4 節では、文章を修正し、より具体性の低い *鍵材料* という用語ではなく、*秘密鍵* 及び *プライベート鍵* を説明するようにした。
59. 8.3.5 節では、節の冒頭に、失効がどのように機能するかを説明するための文章を追加した。*失効* が、第 6 段落の最後の文に追加された。
60. 9 節を全面的に改訂した。
61. 10 節 Key-Management Specifications for Cryptographic Devices or Applications は削除された。これは SP 800-57 パート 2 に記載されている。
62. 付録 A では、順方向誤り訂正を定義する脚注を追加した。
63. 付録 B.3 の項目 2 を修正し、復元した鍵の使用方法について 9.4.2 節を参照するようにした。
64. 付録 B.3.13.2 において、認可公開鍵をアーカイブする必要はないという文章を追加した。
65. 付録 B.5 では、第 1 段落の例を変更した。
66. 参考文献を付録 C から付録 A の前の参考文献の節に移動した。参考文献を更新した。IG 7.5、RFC 3647、RFC 8032、SP 800-175B 及び SP 800-185 への参照を追加した。
67. 付録 D は、付録 C という名前に変更された。