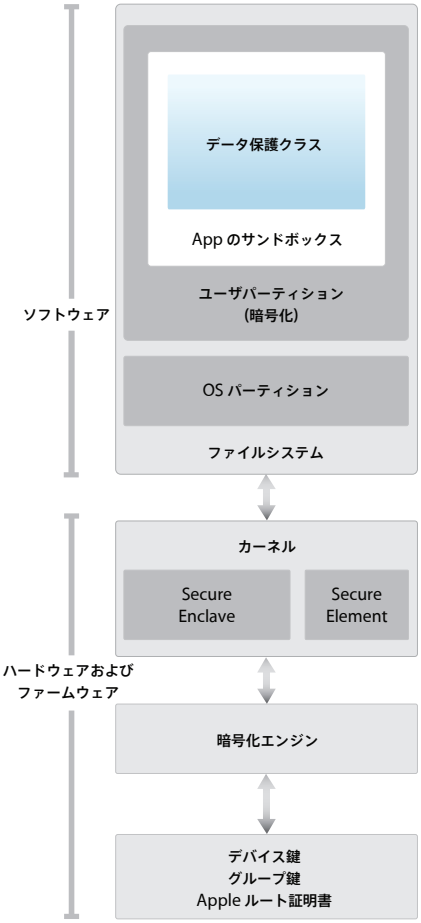
iOS のセキュリティ

iOS 10

2017 年 3 月

目次

概要



iOS のセキュリティアーキテクチャの図。この文書で説明する各種のテクノロジーの概要を視覚的に表したものです。

Apple の iOS プラットフォームは、セキュリティを核に据えて設計されています。数十年に及ぶ経験を出発点に最高のモバイルプラットフォームの開発に取り掛かり、まったく新しいアーキテクチャを構築しました。デスクトップ環境のセキュリティハザードを念頭に置き、iOS の設計では、セキュリティに対する新たなアプローチを確立しました。モバイルセキュリティを強化し、デフォルトでシステム全体を保護する革新的な機能を開発、統合しました。その結果、iOS ではモバイルデバイスのセキュリティが大きく進歩しています。

すべての iOS デバイスでは、ソフトウェア、ハードウェア、およびサービスを連携して機能するように統合することで、最高のセキュリティと透過的なユーザエクスペリエンスを実現しています。iOS はデバイスやデータのみを保護するのではなく、エコシステム全体を保護します。これには、ユーザがローカル上、ネットワーク上および主なインターネットサービス上で行うすべての操作を含みます。

iOS および iOS デバイスには、高度でありながら使いやすいセキュリティ機能が搭載されています。これらの多くの機能はデフォルトで有効になっており、IT 部門で何から何まで構成する必要はありません。また、デバイスの暗号化などの重要なセキュリティ機能は設定が変えられないので、ユーザがそうした機能を誤って無効にすることもありません。Touch ID などのその他の機能によってデバイスの保護がさらに簡単で直観的になり、ユーザエクスペリエンスが向上しています。

この文書では、iOS プラットフォームにおけるセキュリティ技術とセキュリティ機能の実装構造について詳しく説明しています。また、組織特有のセキュリティのニーズを満たすために、iOS プラットフォームにおけるセキュリティ技術とセキュリティ機能を組織独自のポリシーや手続きと統合する場合に役立てることもできます。

この文書は、以下のトピックに分かれています：

- **システムのセキュリティ**：iPhone、iPad、および iPod touch のプラットフォームとして統合された安全なソフトウェアおよびハードウェア。
- **暗号化とデータ保護**：デバイスを紛失したり盗まれたりした場合や、不正なユーザが使用したり変更したりしようとした場合でもユーザデータを保護するアーキテクチャと設計。
- **App のセキュリティ**：安全かつプラットフォームの完全性を損ねることなく App を実行するシステム。
- **ネットワークのセキュリティ**：安全な認証と転送データの暗号化を可能にする業界標準のネットワークプロトコル。
- **Apple Pay**：Apple が導入した安全な支払いのための機能。
- **インターネットサービス**：メッセージング、同期、およびバックアップを支える Apple のネットワークベースのインフラストラクチャ。
- **デバイスの制御**：iOS デバイスを管理し、不正使用を防ぎ、紛失または盗難時にリモートワイプを可能にする方法。
- **プライバシーの制御**：位置情報サービスおよびユーザデータへのアクセスを制御するために使用される iOS の機能。

システムのセキュリティ

デバイス・ファームウェア・アップグレード（DFU）モードにする
デバイスを DFU モードにした後にデバイスを復元すると、Apple が署名した未変更のコードしか存在しない、既知の正常な状態に戻ります。手動で DFU モードにできます：まず USB ケーブルを使ってデバイスをコンピュータに接続してから、ホームボタンとスリープ／スリープ解除ボタンの両方を押したままにします。8 秒経ったら、ホームボタンは押し続けたまま、スリープ／スリープ解除ボタンだけ離します。注記：デバイスが DFU モードのときは、画面には何も表示されません。スリープ／スリープ解除ボタンが長く押されすぎた場合、Apple ロゴが表示されます。

システムのセキュリティは、すべての iOS デバイスのあらゆるコアコンポーネントでソフトウェアとハードウェアの両方のセキュリティが保たれるように設計されています。システムのセキュリティには、ブートアップ・プロセス、ソフトウェア・アップデート、および Secure Enclave などがあります。このアーキテクチャは iOS のセキュリティの中核をなすものであり、これによってデバイスのユーザビリティが損なわれることはありません。

iOS デバイスではソフトウェアとハードウェアが密接に統合されているため、各システムコンポーネントの信頼性の確保や、システム全体の検証が可能になっています。iOS ソフトウェア・アップデートへの初回のブートアップから他社製の App に至るまで、ハードウェアとソフトウェアが最適な形で連動し、リソースが適切に使用されるように、各ステップが解析および検証されます。

セキュアブートチェーン

起動プロセスの各ステップに含まれるコンポーネントには、完全性を保証するために Apple による暗号学的な署名が付いており、信頼チェーンの検証後にのみ実行されます。これらのコンポーネントには、ブートローダ、カーネル、カーネル拡張機能、およびベースバンドファームウェアなどがあります。このセキュアブートチェーンにより、最下位レベルのソフトウェアが改ざんされていないことが保証されます。

iOS デバイスの電源を入れると、デバイスのアプリケーションプロセッサによって、Boot ROM という読み出し専用メモリから即座にコードが実行されます。ハードウェアの信頼の起点となるこの変更不可のコードは、チップ製造時に書き込まれたものであり、無条件に信頼されます。Boot ROM コードには Apple ルート CA の公開鍵が含まれており、この公開鍵は、iBoot ブートローダの読み込みを許可する前に iBoot ブートローダが Apple によって署名されていることを確認するために使用されます。これが信頼チェーンの最初のステップです。信頼チェーンの各ステップでは、その次のステップが Apple によって署名されていることを保証します。iBoot のタスクが終了すると、iOS カーネルが検証および実行されます。S1、A9、またはこれ以前の A シリーズプロセッサを搭載したデバイスではもう 1 つ段階が加わり、Boot ROM によって Low-Level Bootloader（LLB）が読み込まれて検証された後に、iBoot が読み込まれて検証されます。

ブートプロセスのステップが 1 つでも次のプロセスの読み込みや検証をできない場合は、起動が停止され、デバイスの画面に「iTunes に接続」と表示されます。これはリカバリモードと呼ばれます。Boot ROM で LLB の読み込みおよび検証ができない場合は、DFU（デバイス・ファームウェア・アップグレード）モードになります。いずれの場合でも、USB を使ってデバイスを「iTunes」に接続し、工場出荷時の設定に復元する必要があります。手動でリカバリモードにする方法について詳しくは、support.apple.com/ja-jp/HT1808 を参照してください。

モバイルデータ通信ネットワークにアクセスできるデバイスでは、ベースバンドサブシステムでも、署名されたソフトウェアとベースバンドプロセッサによって検証された鍵を使って、これと似た固有のセキュアブートプロセスが実行されます。

Secure Enclave を搭載したデバイスでは、Secure Enclave の独立したソフトウェアが Apple によって検証および署名されていることを保証するために、Secure Enclave コプロセッサでもセキュアブートプロセスが実行されます。

システムソフトウェア認証

Apple は、新たなセキュリティ上の懸念に対処したり、新しい機能を提供したりするために、定期的にソフトウェア・アップデートをリリースしています。これらのアップデートは、サポートされているすべてのデバイスに同時に提供されます。ユーザは、iOS アップデートの通知をデバイスまたは「iTunes」で受け取ります。アップデートはワイヤレスで配信されるので、最新のセキュリティ修正の迅速な導入を促すことができます。

上記の起動プロセスにより、Apple が署名したコードのみデバイスにインストールされることが確実になります。最新のセキュリティアップデートを含まない古いバーションにデバイスがダウングレードされるのを防ぐため、iOS はシステムソフトウェア認証というプロセスを使用します。ダウングレードが可能になってしまうと、デバイスを乗っ取った攻撃者に古いバーションの iOS をインストールされ、新しいバーションで修正された脆弱性を悪用されてしまいます。

Secure Enclave を搭載したデバイスでは、ソフトウェアの完全性を保証し、ダウングレード目的のインストールを防止するために、Secure Enclave コプロセッサでもシステムソフトウェア認証が利用されます。後述の「Secure Enclave」セクションを参照してください。

iOS ソフトウェア・アップデートは、「iTunes」または OTA（Over The Air、ワイヤレス通信経由）でデバイスにインストールできます。「iTunes」を使用する場合は、iOS の完全なコピーがダウンロードおよびインストールされます。OTA でソフトウェア・アップデートする場合は、アップデートの完了に必要なコンポーネントのみがダウンロードされるため、iOS 全体をダウンロードする場合よりもネットワーク効率が向上します。さらに、macOS Server でキャッシュサービスを実行しているローカル・ネットワーク・サーバにソフトウェア・アップデートをキャッシュすれば、必要なアップデートデータを入手するために iOS デバイスから Apple のサーバにアクセスする必要がなくなります。

iOS のアップデート中は、「iTunes」（OTA でのソフトウェア・アップデートの場合はデバイス自体）が Apple のインストール認証サーバに接続して、インストールされるバンドルの各コンポーネント（iBoot、カーネル、および OS イメージなど）の暗号処理による計算値（cryptographic measurements）のリスト、アンチリプレイの乱数（ノンス）、およびデバイスの固有 ID（ECID）を送信します。

認証サーバは、提示された暗号計算値リストとインストールが許可されているバージョンを照合し、一致が見つかった場合は、ECID を計算値に追加して結果に署名します。署名されたデーター式は、アップグレードプロセスの一部としてサーバからデバイスに送信されます。ECID を追加することで、リクエストしたデバイスの認証を「パーソナライズ」することができます。既知の計算値に対してのみ認証および署名することにより、Apple によって提供された通りにアップデートが完了することが保証されます。

起動時に信頼チェーンで評価することで、署名が Apple のものであるかどうかと、ディスクから読み込まれた項目の計算値とデバイスの ECID の組み合わせが署名されたものと一致するかどうかが検証されます。

これらのステップにより、認証がそのデバイスに対するものであることと、あるデバイスの古いバージョンの iOS が別のデバイスにコピーされないことが保証されます。ノンスが使用されるため、攻撃者はサーバの応答を保存し、それを使ってデバイスを不正に解析したり、あるいはシステムソフトウェアを改ざんしたりすることはできません。

Secure Enclave

Secure Enclave は、Apple S2、Apple A7 以降の A シリーズプロセッサに組み込まれたコプロセッサです。暗号化されたメモリを使用するほか、ハードウェア乱数生成器を備えています。Secure Enclave は、データ保護における鍵管理のすべての暗号演算を担い、カーネルが危殆化した場合でもデータ保護の完全性が維持されます。Secure Enclave とアプリケーションプロセッサ間の通信は、割り込み方式のメールボックスと共有メモリのデータバッファから隔離されています。

Secure Enclave は、Apple がカスタマイズした L4 マイクロカーネルファミリーを実行します。Secure Enclave は独自のセキュアブートを使用し、アプリケーションプロセッサとは異なるパーソナライズされたソフトウェア・アップデート・プロセスでアップデートできます。A9 以降の A シリーズプロセッサでは、チップによって UID（固有 ID）が安全に生成されます。この UID は、Apple もシステムのほかの部分も知ることはできません。

デバイスが起動すると、一時鍵が作成されて UID と関連付けられ、デバイスのメモリ領域の Secure Enclave 部分の暗号化に使用されます。Apple A7 を除き、Secure Enclave のメモリは一時鍵でも認証されます。

また、Secure Enclave によりファイルシステムに保存されたデータは、UID とエンタングルされた鍵とアンチリプレイカウンタを使用して暗号化されます。

Secure Enclave は、Touch ID センサーからの指紋データの処理を担当します。登録済みの指紋と一致しているかどうかを確認し、一致が見つかった場合はユーザに代わってアクセスや購入を許可します。プロセッサと Touch ID 間の通信は、シリアル・ペリフェラル・インターフェイス経由で実行されます。プロセッサは Secure Enclave にデータを渡すことはできますが、Secure Enclave から読み出すことはできません。通信はセッション鍵によって暗号化および認証されます。この鍵は、Touch ID センサーと Secure Enclave に書き込まれたデバイスの共有鍵を使って生成されます。セッション鍵の交換には AES 鍵ラッピングが使用されます。Touch ID センサーと Secure Enclave の両方がランダムな鍵を提供してセッション鍵を確立し、通信が AES-CCM によって暗号化されます。

Touch ID

Touch ID は、デバイスへの安全なアクセスをより速くより簡単にする指紋認証システムです。あらゆる角度から指紋を読み取り、継続的にユーザの指紋について学習を進めるテクノロジーです。使用するたびに新たなノードの重複をセンサーで検出することにより、指紋マップを拡張し続けます。

Touch ID によって、ユーザがパスコードを入力する頻度が減るため、長くて複雑なパスコードも実用的になります。また、パスコードに取って代わるのではなく、適度な使用範囲と時間の制約の中でデバイスに安全にアクセスできるようにすることで、パスコードによるロックの不便さが解消されます。

Touch ID とパスコード

Touch ID を使用するには、パスコードでロック解除するようにデバイスを設定する必要があります。Touch ID が登録済みの指紋をスキャンして認識すると、デバイスのパスコードを入力しなくても、デバイスがロック解除されます。いつでも Touch ID の代わりにパスコードを使用できます。また、以下の場合にはパスコードが必要になります：

- デバイスの電源を入れた直後、または再起動した直後。
- 48 時間以上デバイスのロックが解除されていない場合。
- パスコードが過去 156 時間（6 日半）以内にデバイスのロック解除に使用されておらず、かつ Touch ID が過去 4 時間以内にデバイスのロック解除に使用されていない場合。
- デバイスがリモート・ロック・コマンドを受け取ったとき。
- 指紋の認証に 5 回失敗した後。
- Touch ID を設定するとき、または Touch ID に新しい指を登録するとき。

Touch ID が有効な場合、スリープ／スリープ解除ボタンを押すとデバイスはすぐにロックされます。パスコードしか使えないデバイスでは、ユーザの多くがロックされるまでの猶予時間を設定し、デバイスを使うたびにパスコードを入力しなくても済むようにしています。Touch ID を搭載したデバイスでは、スリープ時に必ずロックされ、スリープを解除するときは毎回ユーザの指紋が必要です（パスコードにすることも可能）。

Touch ID は、最大 5 本の異なる指を認識するように学習させることができます。1 本の指を登録した場合、ほかの人の指紋と偶然に一致する確率は 50,000 分の 1 です。ただし、Touch ID では、指紋の一致に 5 回失敗しただけで、デバイスにアクセスするためにパスコードの入力が必要になります。

Touch ID のその他の用途

Touch ID は、安全な支払いのために Apple が導入した Apple Pay でも利用できます。詳しくは、この文書の「Apple Pay」セクションを参照してください。

さらに、他社製 App でもシステムが提供する API を使用して、ユーザに Touch ID またはパスコードでの認証を求めることができます。App には認証の成否しか通知されず、Touch ID や登録済みの指紋に関連付けられたデータにはアクセスできません。

キーチェーン項目も Touch ID で保護することで、Secure Enclave での指紋の一致やデバイスのパスコードでのみロック解除されるようにすることができます。App のデベロッパが使用する API には、パスコードがユーザによって設定されていることを検証して、それにより Touch ID を使って認証したりキーチェーン項目をロック解除したりできるものもあります。

iOS 9 以降では、デベロッパは以下を行うことができます：

- Touch ID API 操作がアプリケーションのパスワードまたはデバイスのパスコードを代替として使用することを禁止する。これにより、登録済みの指紋情報を取得できる機能を組み合わせることで、セキュリティが重視される App で Touch ID を第 2 要素として使用できます。
- Secure Enclave 内で ECC 鍵を生成および使用する。これらの鍵は Touch ID で保護できます。これらの鍵を使用する処理は、常に Secure Enclave による承認の後、Secure Enclave 内で実行されます。App はキーチェーンを使用して SecKey 経由でこれらの鍵にアクセスできます。SecKey は Secure Enclave 鍵への参照なので、鍵情報が Secure Enclave 外に漏れることはありません。

Touch ID は、iTunes Store、App Store、および iBooks Store での購入を承認するように構成することもできます。そうすれば、Apple ID パスワードの入力が不要になります。ユーザが購入の承認を選択すると、デバイスと Store 間で認証トークンが交換されます。トークンと暗号ノンスは Secure Enclave で保持されます。ノンスは、すべてのデバイスと iTunes Store で共有される Secure Enclave 鍵で署名されます。iOS 10 では、Store からのリクエストに署名することで、Touch ID で保護された Secure Enclave ECC 鍵を使用して購入が承認されます。

Touch ID のセキュリティ

Touch ID の指紋センサーは、ホームボタンを囲む静電容量式の金属リングが指の接触を検出したときにのみ起動します。指の接触が検出されると、高度なイメージングアレイが起動して指紋がスキャンされ、スキャン結果が Secure Enclave に送信されます。

ラスター形式のこのスキャン結果は、解析用にベクタ形式に変換されている間 Secure Enclave 内の暗号化されたメモリに一時的に保存され、その後破棄されます。解析は皮下の隆線角度のマッピングを利用して実行されます。これは不可逆的なプロセスで、ユーザの実際の指紋を再構築するために必要なマニューシャ（指紋の特徴点）のデータは破棄されます。結果として得られたノードのマップは、個人を特定する情報を含まずに、暗号化された形式で保存されます。これは Secure Enclave のみしか読み出すことができず、Apple に送信されたり、iCloud や「iTunes」にバックアップされたりすることは決してありません。

Touch ID が iOS デバイスをロック解除する仕組み

Touch ID をオフにすると、デバイスがロックされたときに、Secure Enclave に保持されているデータ保護クラス Complete の鍵が破棄されます。このクラスのファイルおよびキーチェーン項目は、ユーザがパスコードを入力してデバイスをロック解除しない限りアクセスできません。

Touch ID がオンになっている場合は、デバイスがロックされたときに鍵は破棄されず、代わりに Secure Enclave 内の Touch ID サブシステムに与えられている鍵でラップされます。ユーザがデバイスをロック解除しようとするときに Touch ID がユーザの指紋を認識した場合は、データ保護鍵をアンラップするための鍵が提供され、デバイスがロック解除されます。このプロセスでは、デバイスのロック解除のためにデータ保護と Touch ID のサブシステムを連携させることによって、保護を強化しています。

デバイスのロック解除に必要な Touch ID の鍵は、デバイスの再起動時には消去されます。また、48 時間が経過するか Touch ID の認証に 5 回失敗した場合は、Secure Enclave によって破棄されます。

暗号化とデータ保護

すべてのコンテンツと設定を消去する
「設定」の「すべてのコンテンツと設定を消去」オプションでは、Effaceable Storage のすべての鍵が完全に消去され、デバイス上のユーザデータは暗号論的にアクセス不可になります。そのため、ほかの人に譲渡したり修理に出したりする前にすべての個人情報をデバイスから確実に削除する場合に最適なオプションです。重要 :「すべてのコンテンツと設定を消去」を使用する前に、必ずデバイスをバックアップしてください。消去されたデータはどのような方法でも復元することはできません。

セキュアブートチェーン、コード署名、およびランタイムプロセスのセキュリティはすべて、信頼されたコードと App のみがデバイスで実行されることを保証するためのものです。iOS にはこのほかにも暗号化とデータ保護の機能が搭載されており、セキュリティインフラストラクチャのほかの部分が危殆化した場合でも（たとえば不正に改ざんされたデバイスでも）、ユーザデータが保護されます。これにより、個人や企業の情報が常時保護されるほか、デバイスの盗難または紛失時に迅速かつ完全にリモートワイプを実行できる手段が提供されるため、ユーザと IT 管理者の双方が重要なメリットを得ることができます。

ハードウェアのセキュリティ機能

モバイルデバイスにおいては、スピードと電力効率が極めて重要です。暗号演算は複雑であり、こうした優先事項を念頭に置かずに設計および実装してしまうと、パフォーマンスやバッテリー駆動時間の問題が発生する場合があります。

すべての iOS デバイスには、フラッシュストレージとシステムのメインメモリ間の DMA パスに AES 256 の暗号化専用エンジンが搭載されているので、ファイルの暗号化が非常に効率良く実行されるようになっています。A9 以降の A シリーズプロセッサでは、フラッシュ・ストレージ・サブシステムは隔離されたバス上にあり、ユーザデータが含まれるメモリへのアクセスは DMA 暗号化専用エンジン経由でのみ許可されます。

デバイスの固有 ID（UID）とデバイスグループ ID（GID）は、製造時にアプリケーションプロセッサと Secure Enclave に焼き付け（UID の場合）または組み込まれた（GID の場合）AES 256 ビット鍵です。ソフトウェアやファームウェアはそれらを直接読み出せず、シリコンに埋め込まれた AES 専用エンジンが UID または GID を鍵として実行した暗号化演算や復号演算の結果しか見ることができません。加えて、Secure Enclave の UID と GID は、Secure Enclave 専用の AES エンジンでしか使用できません。UID は各デバイスに一意であり、Apple や Apple のどのサプライヤーにも記録されません。GID はデバイスの特定クラス（たとえば Apple A8 プロセッサ搭載のすべてのデバイス）のすべてのプロセッサに共通であり、インストール時または復元時にシステムソフトウェアを送信する場合など、セキュリティが重要でないタスクに使用されます。これらの鍵をチップに埋め込むことで、改ざんやバイパス、また AES エンジン外でのアクセスを防止しています。UID と GID には、JTAG などのデバッグインターフェイス経由でもアクセスすることはできません。

UID により、データは特定のデバイスに暗号処理によって関連付けられます。たとえば、ファイルシステムを保護する鍵階層には UID が含まれているので、メモリチップをあるデバイスから別のデバイスに物理的に移動した場合、そのファイルにはアクセスできなくなります。UID はデバイスのその他の識別子には関連付けられていません。

UID と GID 以外のその他すべての暗号鍵は、CTR_DRBG に基づくアルゴリズムを使ってシステムの乱数生成器（RNG）により作成されます。システムのエントロピーは、起動時のタイミングおよびデバイス起動完了後の割り込みタイミングから生成されます。Secure Enclave 内で生成される鍵には、マルチリングオシレータで生成した後に CTR_DRBG で処理する真のハードウェア乱数生成器が使用されます。

保存された鍵を安全に消去することは、鍵の生成と同様に重要です。フラッシュストレージはウェアレベリングされているので、データの複数のコピーを消去しなければならない可能性があり、これが特に安全な消去を難しくしています。この問題に対処するため、iOS デバイスには、Effaceable Storage というデータ消去専用の安全な機能が搭載されています。この機能は、基盤となっているストレージテクノロジー（NAND など）にアクセスして、ごく下位にあるわずかなブロックを直接操作して消去します。

ファイルデータ保護

Apple は、iOS デバイスに内蔵されているハードウェア暗号化機能に加えて、データ保護という技術を採用して、デバイスのフラッシュメモリに保存されるデータの保護を強化しています。データ保護により、デバイスで電話の着信などの一般的なイベントに応答するだけでなく、ユーザデータを高いレベルで暗号化することが可能になっています。「メッセージ」、「メール」、「カレンダー」、「連絡先」、「写真」、および「ヘルスケア」などの重要なシステム App のデータ値では、デフォルトでデータ保護が使用されます。iOS 7 以降にインストールされた他社製 App は、自動的にこの保護が適用されます。

データ保護は鍵階層を構成および管理することで実装され、すべての iOS デバイスに内蔵されたハードウェア暗号化技術を基に構築されています。データ保護は、各ファイルをクラスに割り当てることでファイルごとに制御されます。ファイルにアクセスできるかどうかは、そのクラス鍵がロック解除されているかどうかによって決定されます。

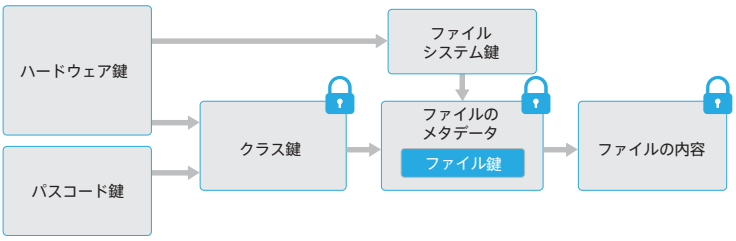
アーキテクチャの概要

データパーティション上にファイルが作成されるたびに、データ保護によって新しい 256 ビット鍵（「Per File」キー）が作成され、ハードウェア AES エンジンに渡されます。そしてハードウェア AES エンジンによりその鍵が使われて、ファイルがフラッシュメモリに書き込まれるときに AES CBC モードでファイルが暗号化されます。（A8 プロセッサを搭載したデバイスでは、AES-XTS モードが使用されます。）初期化ベクトル（IV）はファイルのブロックオフセットを使って計算され、Per File キーの SHA-1 ハッシュを使用して暗号化されます。

Per File キーは複数あるクラス鍵のうちのいずれかでラップされます。このときのクラス鍵は、ファイルへのアクセス条件によって異なります。その他すべての鍵ラッピングと同様に、これも RFC 3394 に基づく NIST AES 鍵ラッピングで実行されます。ラップされた Per File キーは、ファイルのメタデータに保存されます。

ファイルが開かれると、そのファイルのメタデータがファイルシステム鍵で復号され、ラップされた Per File キーとファイルを保護しているクラスの方式が明らかになります。Per File キーは、クラス鍵によってラップ解除されてから、ハードウェア AES エンジンに渡されます。そしてフラッシュメモリからファイルを読み出すときに、ハードウェア AES エンジンがファイルを復号します。ラップされたファイル鍵の処理はすべて Secure Enclave 内で実行されます。そのため、ファイル鍵がアプリケーションプロセッサに直接公開されることはありません。起動時に、Secure Enclave は AES エンジンと一時鍵のネゴシエーションを行います。Secure Enclave がファイル鍵をラップ解除した場合、ファイル鍵は一時鍵で再度ラップされてからアプリケーションプロセッサに戻されます。

ファイルシステム内のすべてのファイルのメタデータは、ランダムな鍵で暗号化されます。この鍵は、iOS がはじめてインストールされたとき、またはユーザによってデバイスがワイプされたときに作成されます。ファイルシステム鍵は Effaceable Storage に保存されます。デバイス上に保存されているので、この鍵はデータの機密性を維持するためには使用されず、要求に応じてすばやく消去されるように設計されています（ユーザが「すべてのコンテンツと設定を消去」オプションを選択するか、ユーザまたは管理者がモバイルデバイス管理サーバ、Exchange ActiveSync、または iCloud からリモートワイプ・コマンドを発行すると消去されます）。このようにして鍵を消去すると、すべてのファイルは暗号論的にアクセス不可になります。



ファイルの内容は Per File キーで暗号化され、Per File キーはクラス鍵でラップされてファイルのメタデータに保存されます。そしてメタデータはファイルシステム鍵で暗号化されます。クラス鍵はハードウェア UID で保護されますが、ユーザのパスコードで保護されるクラスもあります。この階層構造により、柔軟性とパフォーマンスの両方を達成することができます。たとえば、ファイルのクラスを変更する場合はそのファイルの Per File キーをラップし直すだけでよく、パスコードを変更した場合はクラス鍵のラップだけが変更されます。

パスコード

パスコードの検討事項
数字のみを含む長いパスワードを入力する場合は、ロック画面にフルキーボードではなくテンキーが表示されます。数字のみの長いパスコードは英数字を含む短いパスコードよりも簡単に入力できますが、同水準のセキュリティを確保できます。

デバイスパスコードを設定することで、ユーザはデータ保護を自動的に有効にできます。iOS は、6 桁の数字、4 桁の数字、および英数字を含む任意の長さのパスコードをサポートしています。パスコードを設定すると、デバイスをロック解除するだけでなく、一部の暗号化鍵にエントロピーを付加することができます。これによって、デバイスを乗っ取った攻撃者は、パスコードがない限り特定の保護クラスのデータにアクセスできなくなります。

パスコードはデバイスの UID とエンタングルされるので、デバイスを攻撃するには総当たり（ブルートフォース）攻撃以外に方法はありません。各試行にかかる時間を長くするために、反復間隔が大きく設定されています。反復間隔は、試行 1 回につき約 80 ミリ秒かかるように調整されています。このため、小文字のアルファベットと数字を含む英数字 6 文字のパスコードの場合、すべての組み合わせを試すには 5 年半超もの時間がかかることになります。

ユーザパスコードが強力であれば、それだけ暗号化鍵も強力になります。Touch ID を使用すれば、Touch ID を使用しない場合の現実的な長さのパスコードよりもはるかに長いパスコードを設定することで、この点をさらに強化できます。これにより、1 日に何度も実行する iOS デバイスのロック解除のユーザエクスペリエンスを損なうことなく、データ保護用の暗号化鍵を保護するエントロピーの有効性を増大させることができます。

パスコード入力間の待ち時間

入力回数	強制される待ち時間
1 ～ 4	なし
5	1 分
6	5 分
7 ～ 8	15 分
9	1 時間

パスコードに対する総当たり（ブルートフォース）攻撃をさらに抑制するために、ロック画面で無効なパスコードが入力された場合、次の入力までの待ち時間が延長されます。「設定」＞「Touch ID とパスコード」＞「データを消去」がオンの場合、パスコードの入力を 10 回連続で間違えるとデバイスが自動的にワイプされます。この設定は、モバイルデバイス管理（MDM）および Exchange ActiveSync の管理ポリシーとしても利用可能で、回数の上限を下げることもできます。

Secure Enclave を搭載したデバイスでは、Secure Enclave コプロセッサによって待ち時間が強制的に適用されます。遅延が適用されているデバイスが再起動された場合、遅延は適用されたままになり、再起動後にカウントが再開されます。

データ保護クラス

iOS デバイス上に新しいファイルが作成されると、ファイルを作成した App によってクラスが割り当てられます。データへのアクセス条件を決定するポリシーはクラスごとに異なります。基本のクラスとポリシーについて、以下のセクションで説明します。

Complete Protection

（`NSFileProtectionComplete`）：クラス鍵は、ユーザのパスコードとデバイスの UID から生成される鍵によって保護されます。ユーザがデバイスをロックした直後（「パスコードを要求」が「即時」に設定されている場合は 10 秒）、復号されたクラス鍵が破棄され、このクラスのすべてのデータは、ユーザがパスコードを再度入力するか Touch ID でデバイスをロック解除しない限りアクセスできなくなります。

Protected Unless Open

（NSFileProtectionCompleteUnlessOpen）：一部のファイルは、デバイスのロック中に書き込まれる必要がある場合があります。バックグラウンドでダウンロードされるメールの添付ファイルが良い例です。この動作は、楕円曲線に基づく非対称暗号方式（Curve25519 を使用する ECDH）により可能になっています。通常の Per File キーは、NIST SP 800-56A に記述されたワンパス Diffie-Hellman 鍵共有を使って保護されます。

この共有に使用する一時公開鍵は、ラップされた Per File キーと共に保存されます。鍵導出関数は、NIST SP 800-56A の 5.8.1 に記述された Concatenation Key Derivation Function（Approved Alternative 1）です。AlgorithmID は省略されています。PartyUInfo と PartyVInfo はそれぞれ一時的および静的な公開鍵です。SHA-256 がハッシュ関数として使用されます。ファイルが閉じられると、Per File キーはすぐにメモリからワイプされます。再度ファイルを開く場合は、Protected Unless Open クラスの秘密鍵とファイルの一時公開鍵を使って共有シークレットが再度作成されます。これらは Per File キーをアンラップするために使用され、Per File キーはファイルの復号に使用されます。

Protected Until First User Authentication

（NSFileProtectionCompleteUntilFirstUserAuthentication）：このクラスの動作は、Complete Protection と同じです。ただし、復号されたクラス鍵は、デバイスのロック時にメモリから削除されません。このクラスでの保護は、デスクトップでのボリューム全体の暗号化と似ており、デバイスを再起動させる攻撃からデータを保護します。すべての他社製 App では、データをほかのデータ保護クラスに割り当てない限り、これがデータのデフォルトのクラスになります。

No Protection

（NSFileProtectionNone）：このクラス鍵は UID でのみ保護され、Effaceable Storage に保存されます。このクラスのファイルの復号に必要な鍵はすべてデバイスに保存されるため、この暗号化から得られるメリットは、迅速なリモートワイプができるということだけです。ファイルにデータ保護クラスが割り当てられていない場合でも、（iOS デバイス上のすべてのデータと同様に）ファイルは暗号化された形式で保存されます。

データ保護クラス鍵

Class A	Complete Protection	(NSFileProtectionComplete)
Class B	Protected Unless Open	(NSFileProtectionCompleteUnlessOpen)
Class C	Protected Until First User Authentication	(NSFileProtectionCompleteUntilFirstUserAuthentication)
Class D	No Protection	(NSFileProtectionNone)

キーチェーンデータ保護

多くの App はパスワードだけでなく、その他の短くも機密性の高いデータ片（鍵やログイントークンなど）を扱う必要があります。iOS キーチェーンには、これらの項目を安全に保存する方法が用意されています。

キーチェーンは SQLite データベース形式で実装され、ファイルシステムに保存されています。データベースは 1 つしかありません。securityd デーモンによって、各プロセスや App がどのキーチェーン項目にアクセスできるかが決定されます。キーチェーンアクセス API の結果によりデーモンが呼び出され、デーモンによって App の「Keychain-access-groups」、「application-identifier」、および「application-group」の各エンタイトルメントが照会されます。アクセスは 1 つのプロセスには限定されず、アクセスグループを利用してキーチェーン項目を App 間で共有することができます。

キーチェーン項目は、同じデベロッパの App 間でのみ共有できます。この機構は、Apple Developer Program のアプリケーショングループを通じて割り当てられたプレフィックスに基づくアクセスグループの使用を他社製 App に義務付けることで管理されています。プレフィックス要件とアプリケーショングループの一意性は、コード署名、プロビジョニングプロファイル、および Apple Developer Program によって実現されます。

キーチェーン項目のコンポーネント

アクセスグループのほかに、各キーチェーン項目には管理メタデータ（「作成日」や「前回のアップデート」のタイムスタンプなど）が含まれます。

また、項目（アカウントやサーバ名など）を照会するための属性の SHA-1 ハッシュも含まれているので、各項目を復号せずに検索することができます。さらに、以下を含む暗号化データが含まれています：

- バージョン番号
- アクセス制御リスト（ACL）データ
- 項目が属する保護クラスを示す値
- 保護クラス鍵でラップされた Per Item キー
- バイナリ形式の plist にエンコードされ Per Item キーで暗号化された、項目を説明する属性辞書（SecItemAdd に渡される）

暗号化方式は、AES 128 GCM（Galois/Counter Mode）です。アクセスグループは属性に含まれ、暗号化中に計算される GMAC タグで保護されます。

キーチェーンデータは、ファイルデータ保護で使用されるものに似たクラス構造を使って保護されます。これらのクラスの動作は、ファイルデータ保護の各クラスと同等です。ただし、固有の鍵が使用され、API の名前が異なります。

利用できるタイミング	ファイルデータ保護	キーチェーンデータ保護
ロック解除時	NSFileProtectionComplete	kSecAttrAccessibleWhenUnlocked
ロック中	NSFileProtectionCompleteUnlessOpen	不可
初回ロック解除後	NSFileProtectionCompleteUntilFirstUserAuthentication	kSecAttrAccessibleAfterFirstUnlock
常時	NSFileProtectionNone	kSecAttrAccessibleAlways
パスコードが有効なとき	不可	kSecAttrAccessible WhenPasscodeSetThisDeviceOnly

バックグラウンド更新サービスを利用する App は、バックグラウンドでのアップデート中にアクセスする必要があるキーチェーン項目に `kSecAttrAccessibleAfterFirstUnlock` を使用できます。

クラス `kSecAttrAccessibleWhenPasscodeSetThisDeviceOnly` の動作は `kSecAttrAccessibleWhenUnlocked` と同じですが、利用できるのはデバイスにパスコードが構成されているときのみです。このクラスは、システムキーバッグにのみ存在し、iCloud キーチェーンに同期されたり、バックアップされたり、エスクローキーバッグに含まれたりすることはありません。パスコードが削除またはリセットされた場合、クラス鍵が破棄されることによって、これらの項目は使用できなくなります。

その他のキーチェーンクラスにも「このデバイスのみ」の保護クラスがあります。このクラスはバックアップ中にデバイスからコピーされるときに UID で常時保護されるので、別のデバイスに復元されると使用できなくなります。

Apple は、保護する情報のタイプや iOS で必要になるタイミングに応じてキーチェーンクラスを選択することで、セキュリティとユーザビリティのバランスに配慮しています。たとえば、VPN 証明書はデバイスで常時接続を維持するために常に利用できる状態になっている必要がありますが、「移行不可」に分類されているので別のデバイスに移動することはできません。

iOS で作成されたキーチェーン項目については、以下のクラス保護が強制的に適用されます：

項目	アクセスできるタイミング
Wi-Fi パスワード	初回ロック解除後
メールアカウント	初回ロック解除後
Exchange アカウント	初回ロック解除後
VPN パスワード	初回ロック解除後
LDAP、CalDAV、CardDAV	初回ロック解除後
ソーシャル・ネットワーク・アカウントのトークン	初回ロック解除後
Handoff アドバタイズメント暗号化鍵	初回ロック解除後
iCloud トークン	初回ロック解除後
ホームシェアリングパスワード	ロック解除時
「iPhone を探す」トークン	常時
留守番電話	常時
iTunes バックアップ	ロック解除時、移行不可
Safari パスワード	ロック解除時
Safari ブックマーク	ロック解除時
VPN 証明書	常時、移行不可
Bluetooth® 鍵	常時、移行不可
Apple Push Notification service トークン	常時、移行不可
iCloud の証明書と秘密鍵	常時、移行不可

iMessage 鍵	常時、移行不可
構成プロファイルによってインストールされる証明書と秘密鍵	常時、移行不可
SIM PIN	常時、移行不可

キーチェーンアクセス制御

キーチェーンでは、アクセス制御リスト（ACL）を使用して、アクセス権や認証要件のポリシーを設定できます。Touch ID の使用またはデバイスのパスコードの入力による認証がない限り項目にアクセスできないように設定することで、項目にユーザのプレゼンスを要求する条件を設定できます。また、項目の追加後に Touch ID の登録が変更されないように設定することで、項目へのアクセスを制限できます。この制限により、攻撃者が自分の指紋を追加してキーチェーン項目にアクセスすることを防止できます。ACL は Secure Enclave 内で評価され、指定した制限が満たされた場合にのみカーネルに渡されます。

「Safari」に保存されたパスワードへのアクセス

iOS App では、「Safari」に保存されたキーチェーン項目について、以下の 2 つの API を使ってパスワードを自動入力できます。

- `SecRequestSharedWebCredential`
- `SecAddSharedWebCredential`

App のデベロッパと Web サイトの管理者の両者の承認とユーザの同意がある場合にのみ、アクセスが許可されます。App のデベロッパは App にエンタイトルメントを含めることで、「Safari」に保存されたパスワードにアクセスする意思を表明できます。このエンタイトルメントには、関連する Web サイトの完全修飾ドメイン名がリストされます。Web サイトは、承認した App の一意の App 識別子をリストしたファイルをサーバに配置する必要があります。com.apple.developer.associated-domains エンタイトルメントを持つ App がインストールされると、iOS がリスト内の各 Web サイトに TLS リクエストを発行し、ファイル /apple-app-site-association を要求します。インストールされる App の App 識別子がファイルにリストされている場合は、その Web サイトと App が信頼関係にあるとマークされます。信頼関係がある場合にのみ、これら 2 つの API を呼び出したときにユーザにプロンプトが表示されます。ユーザがこれに同意しないと、パスワードを App に渡したり、アップデートまたは削除したりすることはできません。

キーバッグ

ファイルとキーチェーンのデータ保護クラスの鍵は、キーバッグに収集されて管理されます。iOS では、ユーザ、デバイス、バックアップ、エスクロー、iCloud バックアップのキーバッグが使用されます。

ユーザキーバッグには、デバイスの通常の操作に使用されるクラス鍵がラップされて保存されています。たとえば、パスコードが入力されると、`NSFileProtectionComplete` 鍵がユーザキーバッグから読み込まれ、アンラップされます。これは No Protection クラスに保存されているバイナリ形式の plist ですが、その内容は Effaceable Storage に保存されている鍵によって暗号化されています。キーバッグに前方秘匿性を追加するために、この鍵はユーザがパスコードを変更するたびにワイプされ再生成されます。AppleKeyStore カーネル拡張機能はユーザキーバッグを管理しており、デバイスのロック状態に関してはこの拡張機能に照会できます。ユーザキーバッグ内のすべてのクラス鍵がアクセスできる状態になっていて、正しくアンラップされている場合にのみ、AppleKeyStore はデバイスがロック解除されていると報告します。

デバイスキーバッグは、デバイス固有のデータに関わる操作に使用されるクラス鍵をラップして保存するために使用されます。共有して使用するように構成されている iOS デバイスは、ユーザのログイン前に資格情報へのアクセスが必要になる場合があるため、ユーザのパスコードで保護されていないキーバッグが必要になります。iOS は、各ユーザのファイルシステムコンテンツごとに暗号化を分離することをサポートしないため、システムはデバイスキーバッグからのクラス鍵を使用して Per File キーをラップします。ただし、キーチェーンはユーザキーバッグからのクラス鍵を使用して、ユーザキーチェーン内の項目を保護します。単一ユーザが使用するように構成されている（デフォルト構成の）iOS デバイスでは、デバイスキーバッグとユーザキーバッグはまったく同じで、ユーザのパスコードによって保護されます。

バックアップキーバッグは、暗号化されたバックアップが「iTunes」によって作成されたときに作成され、デバイスがバックアップされているコンピュータに保存されます。新しい鍵のセットを含む新しいキーバッグが作成され、バックアップされたデータはこれらの新しい鍵で再度暗号化されます。前述したように、移行不可のキーチェーン項目は UID 由来の鍵でラップされたままになっているため、これらはオリジナルのバックアップ元のデバイスには復元できますが、別のデバイス上に復元した場合はアクセスできなくなります。

バックアップキーバッグは「iTunes」で設定されたパスワードで保護され、PBKDF2 が 1000 万回反復実行されます。反復回数はこれだけ多く設定されていますが、特定のデバイスには関連付けられません。そのため理論上は、バックアップキーバッグは多くのコンピュータから同時並行的に総当たり（ブルートフォース）攻撃される可能性があります。こうした脅威は、十分に強いパスワードを使用することで軽減できます。

ユーザが iTunes バックアップを暗号化しないことを選択した場合、データ保護クラスにかかわらずバックアップファイルは暗号化されません。ただし、この場合でもキーチェーンは UID 由来の鍵で保護されます。このため、キーチェーン項目は、バックアップパスワードが設定されている場合にのみ新しいデバイスに移行されます。

エスクローキーバッグは、「iTunes」の同期と MDM に使用されます。このキーバッグにより、「iTunes」がバックアップや同期をするときにユーザによるパスコードの入力が不要になるほか、MDM サーバがユーザのパスコードをリモートで消去することが可能になります。エスクローキーバッグは、「iTunes」との同期に使用されるコンピュータか、デバイスを管理する MDM サーバに保存されます。

エスクローキーバッグにより、データのすべてのクラスへのアクセスが必要になる場合があるデバイス同期の際のユーザエクスペリエンスが向上します。パスコードでロックされたデバイスがはじめて「iTunes」に接続されると、ユーザはパスコードの入力を求められます。その後、デバイスで使用されているものと同じクラス鍵を含むエスクローキーバッグがデバイスによって作成され、新たに生成された鍵で保護されます。エスクローキーバッグとそれを保護する鍵は、デバイスとホストまたはデバイスとサーバに分けて保存され、デバイスに保存されているデータには Protected Until First User Authentication クラスが割り当てられます。このため、デバイスの再起動後にはじめて「iTunes」にバックアップするときに、デバイスのパスコードの入力が必要になります。

OTA でのソフトウェア・アップデートの場合、ユーザはアップデート開始時にパスコードの入力を求められます。このパスコードを使用して、アップデート後にユーザキーバッグをロック解除するためのワンタイムロック解除トークンが安全に作成されます。このトークンは、ユーザのパスコードを入力しないと生成できません。また、ユーザのパスコードが変更された場合、以前に生成されたトークンはすべて無効になります。

ワンタイムロック解除トークンは、ソフトウェア・アップデートの手動インストールおよび自動インストールの両方で使用されます。このトークンは、Secure Enclave のモノトニックカウンタの現在値、キーバッグの UUID、および Secure Enclave の UID から派生した鍵で暗号化されます。

Secure Enclave 内のワンタイムロック解除トークンのカウンタが増分されると、既存のトークンがすべて無効になります。カウンタが増分されるのは、トークンが使用されたとき、再起動したデバイスの初回のロック解除後、ソフトウェア・アップデートがユーザまたはシステムによってキャンセルされたとき、またはトークンのポリシータイマーが期限切れになったときです。

手動ソフトウェア・アップデートのワンタイムロック解除トークンは 20 分後に無効になります。このトークンは Secure Enclave から書き出され、Effaceable Storage に書き込まれます。デバイスが 20 分以内に再起動しなかった場合、ポリシータイマーによってカウンタが増分されます。

自動ソフトウェア・アップデートの場合（アップデートが通知されたときにユーザが「後でインストール」を選択すると設定されます）、アプリケーションプロセッサが Secure Enclave 内に保持するワンタイムロック解除トークンは最大 8 時間、有効な状態が保持されます。その時間が経過すると、ポリシータイマーによってカウンタが増分されます。

iCloud バックアップキーバッグは、バックアップキーバッグに似ています。このキーバッグ内のすべてのクラス鍵は、非対称鍵（Protected Unless Open データ保護クラスと同様に Curve25519 を使用）なので、iCloud バックアップはバックグラウンドで実行することができます。No Protection 以外のすべてのデータ保護クラスについては、暗号化されたデータがデバイスから読み出されて iCloud に送信されます。対応するクラス鍵は iCloud 鍵によって保護されます。キーチェーンクラス鍵は、暗号化されていない iTunes バックアップと同様に、UID 由来の鍵でラップされます。iCloud キーチェーンのキーチェーン復元内のバックアップには、非対称キーバッグも使用されます。

セキュリティ認定とプログラム

注記：iOS セキュリティの認証、認定、ガイダンスの最新情報については、support.apple.com/ja-jp/HT202739 を参照してください。

ISO 27001 認証

Apple は、Apple School Manager、iCloud、iMessage、FaceTime、管理対象 Apple ID、および iTunes U のインフラストラクチャ、開発、運用について、情報セキュリティ・マネジメント・システムに関する ISO 27001 認証を取得しました。これらは 2016 年 2 月 26 日付けの適用宣言書 v1.0 に基づきます。Apple の ISO 標準への準拠は英国規格協会によって認証されています。この認証を確認するには、www.bsigroup.com/en-GB/our-services/certification/certificate-and-client-directory/search-results/?searchkey=company=apple&licencenumber=IS+649475 を参照してください。

暗号認定（FIPS 140-2）

iOS の暗号モジュールは、iOS 6 以降、毎回のリリース後に米国連邦情報処理規格（FIPS）140-2 レベル 1 に準拠していることが認定されています。iOS 10 の暗号モジュールは iOS 9 および iOS 8 の暗号モジュールと同一ですが、Apple は iOS をリリースするごとに再認定のためにモジュールを提出しています。このプログラムは、Apple の App および iOS の暗号サービスを適切に使用する Apple の App および他社製 App の暗号演算の完全性を保証するものです。

コモンクライテリア認証（ISO 15408）

Apple は、すでに iOS についてコモンクライテリア認証（CCC）プログラムの認証を受けています。最初に完了した 3 つの認証は、Mobile Device Fundamental Protection Profile v2.0（MDFPP2）の VID10695（iOS 9 用）、VPN IPSecPP1.4 Client Protection Profile（VPNIPSecPP1.4）の VID10714、Extended Package for Mobile Device Management Agents v2.0（MDMAgentEP2）の VID10725 です。Apple は、これらと同じ認証の取得を iOS 10 でも進めており、iOS の今後のメジャーリリースでも継続していく予定です。Apple は International Technical Community（ITC）で、重要なモバイルセキュリティテクノロジーの評価に特化したコラボラティブ・プロテクション・プロファイル（cPP）（現在は利用不可）の開発において積極的な役割を果たしてきました。Apple は今後も、現在利用可能な PP の新しいバージョンやアップデートされたバージョンの評価と、それに基づいた認証を目指していきます。

Commercial Solutions for Classified（CSfC）

該当する場合、Apple は、Commercial Solutions for Classified（CSfC）プログラム・コンポーネント・リストへの追加のために iOS プラットフォームと各種サービスを提出しています。具体的には、モバイルプラットフォーム向けの iOS 9、IPSec VPN クライアント向けの IKEv2 クライアント（IKEv2 VPN 常時接続のみ）、および MDM Agent（MDMAgentEP2）です。Apple のプラットフォームとサービスは、コモンクライテリア認証の審査を受ける過程で、CSfC プログラムのコンポーネントリストへの追加を検討する対象としても提出されます。

セキュリティ構成ガイド

Apple は世界各国の政府と協力し、より安全な環境を維持する（「デバイスハードニング」する）ための手順や推奨事項を記載した各種ガイドを策定しています。これらのガイドには、保護を強化するための iOS の機能の構成方法および利用方法に関する検査済みの情報が明確に記載されています。

App のセキュリティ

App は現代のモバイル・セキュリティ・アーキテクチャにおいて最も重要な要素の 1 つです。App は生産性においてすばらしいメリットをもたらす一方で、適切に扱わないと、システムのセキュリティ、安定性、およびユーザデータに悪影響を及ぼす可能性があります。

このため、iOS には複数の保護レイヤーを構築し、App が署名され、検証され、サンドボックス化されていることを保証することでユーザデータを保護しています。これらの要素によって安定した安全な App プラットフォームが提供されているので、何千人ものデベロッパによる数十万もの App を、システムの完全性を損なうことなく iOS に配信することが可能になっています。そして、ユーザは、ウイルス、マルウェア、不正な攻撃などを過度に心配することなく、iOS デバイス上のこれらの App にアクセスできます。

App のコード署名

iOS のカーネルが起動すると、どのユーザプロセスと App の実行を許可するかがカーネルによって制御されます。すべての App について、既知の承認済みのソースから提供されていることと、改ざんされていないことを保証するために、iOS ではすべての実行可能コードが Apple 発行の証明書を使用して署名されている必要があります。「メール」や「Safari」といったデバイスに付属して提供される App は、Apple によって署名されています。他社製 App についても、Apple 発行の証明書を使用して検証および署名されている必要があります。こうしたコード署名の強制は、OS から App へとつながるトラストチェーンのコンセプトの延長であり、他社製 App によって未署名のコードリソースが読み込まれたり、自己書き換えコードが使用されたりするのを防ぐことができます。

App を開発して iOS デバイスにインストールするには、デベロッパは Apple に登録し、Apple Developer Program に参加する必要があります。各デベロッパの現実世界でのアイデンティティは、個人か企業かにかかわらず、Apple によって検証され、その後デベロッパの証明書が発行されます。この証明書を使用することで、デベロッパは App に署名したり、App を App Store に提出して配信したりできます。したがって、App Store のすべての App は、アイデンティティを特定できる個人や組織によって提出されたものであり、悪意のある App の作成が防止されています。また、説明の通りに動作することや、明らかな不具合や問題が含まれていないことを保証するために、すべての App は Apple によってレビューされています。前述のテクノロジーに加えて、このような選別プロセスを実施することで、ユーザは購入する App の品質に信頼を置くことができます。

iOS では、デベロッパは App 内にフレームワークを埋め込んで、そのフレームワークを App 自体や App に埋め込まれた拡張機能で使用することができます。システムやその他の App がそのアドレス空間内に他社製のコードを読み込むのを防止するため、プロセスがリンクするすべてのダイナミックライブラリについて、起動時にコード署名検証が実行されます。この検証は、Apple 発行の証明書から抽出されるチーム識別子（Team ID）を使用することで達成されます。チーム識別子は、英数字 10 文字の文字列 (例: 1A2B3C4D5F) です。プログラムは、システムに付属のプラットフォームライブラリや、コード署名内にメインの実行可能ファイルと同じチーム識別子を持つライブラリにリンクできます。システムの一部として提供される実行可能ファイルにはチーム識別子が含まれていないため、これらの実行可能ファイルはシステム自体に付属のライブラリにのみリンクできます。

企業は組織内で使用するための社内 App を開発して、従業員に配布することが可能です。企業や組織は D-U-N-S 番号を使って Apple Developer Enterprise Program（ADEP）に申請できます。Apple は識別情報と適格性を確認してから申請を承認します。組織は ADEP のメンバーになると、承認したデバイス上で社内 App の実行を許可するプロビジョニングプロファイルを登録および取得できます。ユーザが社内 App を実行するには、プロビジョニングプロファイルをインストールする必要があります。このため、組織が意図したユーザしか、組織の App を iOS デバイスに読み込めません。MDM でインストールされた App は、組織とデバイス間の信頼関係がすでに確立されているため、暗黙的に信頼されます。それ以外の App については、ユーザが「設定」で App のプロビジョニングプロファイルを承認する必要があります。組織は、不明なデベロッパの App をユーザが承認しないように制限できます。どのエンタープライズ App の初回起動時にも、App の実行を許可するという Apple からの許諾をデバイスで受信する必要があります。

ほかのモバイルプラットフォームとは異なり、iOS では、ユーザは悪意のある可能性のある未署名の App を Web サイトからインストールしたり、信頼されていないコードを実行したりすることはできません。実行時に、実行可能ファイルのメモリページが読み込まれるときに、そのすべてについてコード署名チェックが実行され、インストールまたは前回のアップデート以降に App が改ざんされていないことが確認されます。

ランタイムプロセスのセキュリティ

App が承認済みのソースからのものであることが確認されると、ほかの App やシステムのほかの部分の危殆化を防止するための iOS のセキュリティ対策が強制的に適用されます。

すべての他社製 App は「サンドボックス化」されるので、ほかの App によって保存されたファイルにアクセスしたり、デバイスに変更を加えたりすることはできません。これにより、ほかの App によって保存された情報が収集または変更されるのを防ぐことができます。各 App にはファイル保存用の一意のホームディレクトリが用意されますが、これは App がインストールされるときにランダムに割り当てられます。他社製 App が自身の情報以外の情報にアクセスする必要がある場合は、iOS によって明示的に提供されるサービスを使用したときのみアクセスできます。

システムファイルとリソースもユーザの App から保護されます。iOS の大部分は、他社製 App と同様に特権のないユーザ「mobile」として実行されます。OS のパーティション全体は、読み出し専用としてマウントされます。リモート・ログイン・サービスなどの不要なツールは、システムソフトウェアには含まれていません。また、App は API を使って自身の権限を昇格させてほかの App や iOS 自体を変更することはできません。

他社製 App によるユーザ情報および iCloud や拡張機能などの機能へのアクセスは、宣言されたエンタイトルメントにより制御されます。エンタイトルメントは、App に含まれる署名されたキー値ペアで、UNIX ユーザ ID のようなランタイム要素以外の認証を可能にします。エンタイトルメントはデジタル署名されているため変更できません。エンタイトルメントは、通常であればルート権限でプロセスを実行する必要がある特権的な操作を実行するために、システム App およびデーモンによってさまざまな状況で使用されます。これにより、危殆化されたシステムアプリケーションやデーモンによる権限昇格のリスクを大幅に低減できます。

それに加えて、App はシステムが提供する API 経由でしかバックグラウンド処理を実行できません。このため、App はパフォーマンスを低下させたりバッテリー駆動時間を大きく損ねたりすることなく、機能し続けることができます。

アドレス空間配置のランダム化（ASLR）は、メモリ破壊バグの悪用を防止します。内蔵 App では、ASLR により起動時にすべてのメモリ領域がランダム化されます。実行可能コード、システムライブラリ、および関連するプログラミング構成要素のメモリアドレスをランダムに配置することで、多くのセキュリティ上の弱点をつく精緻な攻撃の可能性を低減します。たとえば、return-to-libc 攻撃は、スタックとシステムライブラリのメモリアドレスを操作することによってデバイスを欺き、悪意のあるコードを実行させようとします。これらのメモリアドレスの配置をランダム化すれば、特に複数のデバイスを標的とした攻撃を実行することが極めて難しくなります。iOS の開発環境である「Xcode」は、自動的に ASLR サポートをオンにして他社製プログラムをコンパイルします。

iOS では、メモリページを実行不可能としてマークする ARM の Execute Never（XN）機能を使用することで保護をさらに強化しています。書き込み可能と実行可能の両方としてマークされたメモリページは、厳しく条件が管理された App のみが使用できます。カーネルによって Apple 独自の動的コード署名エンタイトルメントの有無が確認されます。この場合でも、ランダムなアドレスが与えられた実行可能かつ書き込み可能なページを要求するために、1 回の mmap 呼び出しか発行できません。「Safari」では、JavaScript JIT コンパイラでこの機能が使用されます。

拡張機能

iOS では、拡張機能を提供することで、App の機能をほかの App に提供できます。拡張機能は、特殊な目的の署名が付いた実行可能バイナリで、App 内にパッケージ化されています。App のインストール時に拡張機能が自動的に検出され、マッチングの仕組みを持つほかの App で利用できるようになります。

拡張機能をサポートするシステム領域は、拡張機能ポイントと呼ばれます。それぞれの拡張機能ポイントが API を提供し、その領域のポリシーを適用します。システムは拡張機能ポイントに特有のマッチングルールに基づいて、利用できる拡張機能を判断します。システムは必要に応じて拡張機能プロセスを自動的に起動し、そのライフタイムを管理します。拡張機能の利用を特定のシステムアプリケーションに制限するために、エンタイトルメントを使用できます。たとえば、「今日」表示ウィジェットは通知センターにだけ表示され、共有拡張機能は「共有」パネルからのみ利用できます。拡張機能ポイントは、「今日」ウィジェット、共有、カスタムアクション、写真編集、ドキュメントプロバイダ、カスタムキーボードです。

拡張機能は、自身のアドレス空間内で実行されます。拡張機能と拡張機能を起動した App 間の通信には、システムフレームワークが仲介するプロセス間通信が使用されます。互いのファイルやメモリ空間にはアクセスできません。拡張機能は、拡張機能同士、拡張機能を含む App 本体、および拡張機能を使用する App からは互いに隔離されるように設計されています。ほかの他社製 App と同様にサンドボックス化され、拡張機能を含む App 本体のコンテナとは別のコンテナを持ちます。ただし、プライバシー制御へのアクセスは、App 本体と同じものになります。そのため、ユーザが App に「連絡先」へのアクセス権を付与した場合、このアクセス権はその App に埋め込まれた拡張機能に対しては適用されますが、その App が起動する別の App の拡張機能には適用されません。

カスタムキーボードは、ユーザによってシステム全体で有効になる特殊なタイプの拡張機能です。有効にすると、パスコードの入力とテキストのセキュア表示以外のすべてのテキストフィールドでキーボード拡張機能が使用されます。ユーザデータの転送を制限するため、カスタムキーボードはデフォルトで厳しく制限されたサンドボックス内で実行されます。これにより、ネットワーク、プロセスに代わってネットワーク操作を実行するサービス、および入力データの漏えいが可能な API へのアクセスがブロックされます。カスタムキーボードのデベロッパは、拡張機能に Open Access を付与することを要求できます。これにより、拡張機能は、ユーザの同意を得た後にデフォルトのサンドボックス内で実行できるようになります。

モバイルデバイス管理に登録されたデバイスでは、書類とキーボードの拡張機能は Managed Open In ルールに従います。たとえば、MDM サーバは、ユーザが管理対象 App から管理対象外ドキュメントプロバイダに書類を書き出したり、管理対象 App 内で管理対象外キーボードを使用したりすることを禁止できます。また、App のデベロッパは App 内での他社製キーボード拡張機能の使用を禁止できます。

App グループ

特定のデベロッパアカウントが所有する App と拡張機能は、App グループのメンバーとして構成されると、コンテンツを共有できるようになります。デベロッパは任意で Apple Developer Portal 上で適切なグループを作成し、目的の App と拡張機能のセットをそのグループに追加できます。App グループのメンバーとして構成されると、App には以下の項目へのアクセス権が付与されます:

- データ保存用の共有オンディスクコンテナ（そのグループの App が 1 つ以上インストールされている限りデバイス上に残ります）
- 共有される環境設定
- 共有されるキーチェーン項目

Apple Developer Portal によって、App のエコシステム全体での App グループ ID の一意性が保証されます。

App 内のデータ保護

iOS の Software Development Kit（SDK）には、他社や社内のデベロッパがデータ保護を簡単に採用して、App 内で最高レベルの保護を達成できるようにする API がすべて揃っています。データ保護は、NSFileManager、CoreData、NSData、および SQLite などのファイル API とデータベース API で利用できます。

「メール」App（添付ファイルを含む）、管理対象のブック、Safari ブックマーク、App の起動イメージ、および位置情報データについても、ユーザのパスコードによって保護された鍵で暗号化されてデバイスに保存されます。カレンダー（添付ファイルを除く）、連絡先、リマインダー、メモ、メッセージ、および写真には、Protected Until First User Authentication が適用されます。

ユーザがインストールした App のうち、特定のデータ保護クラスに所属していない App には、デフォルトで Protected Until First User Authentication が割り当てられます。

アクセサリ

Made for iPhone/iPod touch/iPad（MFi）ライセンスプログラムでは、審査を通過したアクセサリメーカーは iPod Accessories Protocol（iAP）および必要な対応ハードウェアコンポーネントにアクセスできます。

MFi アクセサリが Lightning コネクタまたは Bluetooth 経由で iOS と通信するときは、デバイスがアクセサリに対して、Apple による認定を受けた証明として Apple 発行の証明書での応答を求め、その証明書を検証します。その後デバイスがチャレンジを送信し、アクセサリはそれに対して署名付きの応答で答える必要があります。このプロセスはすべて Apple が認定アクセサリメーカーに提供するカスタム集積回路で処理されるため、アクセサリ自体に対しては透過的なプロセスです。

アクセサリは、別の伝送方法や伝送機能へのアクセス（Lightning ケーブル経由でのデジタル・オーディオ・ストリームへのアクセスや、Bluetooth 経由での位置情報の提供など）を要求できます。デバイスへのフルアクセスは、認証 IC によって認定デバイスにのみ付与されます。アクセサリが認証情報を提供しない場合、アクセスはアナログオーディオおよび一部のシリアル（UART）オーディオ再生コントロールに限定されます。

AirPlay でも、レシーバが Apple によって認定済みであることを確認するために認証 IC が利用されます。AirPlay オーディオおよび CarPlay ビデオストリームでは、MFi-SAP（Secure Association Protocol）を利用して、アクセサリとデバイス間の通信が AES-128 の CTR モードで暗号化されます。Station-to-Station（STS）プロトコルの一部として、一時鍵が ECDH 鍵交換（Curve25519）を使って交換され、認証 IC の 1024 ビット RSA 鍵を使って署名されます。

HomeKit

HomeKit は、iCloud と iOS のセキュリティを利用してプライベートデータの保護と同期を行うことができるホームオートメーションのインフラストラクチャです。プライベートデータは Apple に開示されません。

HomeKit 識別情報

HomeKit の識別情報とセキュリティは、Ed25519 公開／秘密鍵ペアに基づいています。Ed25519 鍵ペアは、HomeKit のユーザごとに iOS デバイス上で生成され、それがそのユーザの HomeKit 識別情報となります。鍵ペアは、iOS デバイス間および iOS デバイスとアクセサリ間の通信の認証に使用されます。

これらの鍵はキーチェーンに保存され、暗号化されたキーチェーンのバックアップにのみ含められます。これらの鍵は iCloud キーチェーンを使ってデバイス間で同期されます。

HomeKit アクセサリとの通信

HomeKit アクセサリは、iOS デバイスとの通信に使用する固有の Ed25519 鍵ペアを生成します。アクセサリが工場出荷時の設定に復元されると、新しい鍵ペアが生成されます。

iOS デバイスと HomeKit アクセサリ間の接続を確立するため、Secure Remote Password（3072 ビット）プロトコルを使用して鍵の交換が行われます。ユーザが、アクセサリメーカーから提供された 8 桁のコードを iOS デバイスに入力すると、HKDF-SHA-512 から導出された鍵を用いる ChaCha20-Poly1305 AEAD によってそのコードが暗号化されます。アクセサリの MFi 証明書も設定中に検証されます。

使用時に iOS デバイスと HomeKit アクセサリが通信する場合は、上記のプロセスで交換された鍵を使用してそれぞれが相手方を認証します。各セッションは Station-to-Station プロトコルを使用して確立され、セッションごとの Curve25519 鍵に基づき、HKDF-SHA-512 から導出された鍵によって暗号化されます。これは、IP ベースと Bluetooth Low Energy 両方のアクセサリに適用されます。

ローカル・データ・ストレージ

HomeKit はユーザの iOS デバイスに、ホーム、アクセサリ、シーン、およびユーザに関するデータを保存します。保存されるこのデータは、ユーザの HomeKit 識別情報鍵から導出された鍵と乱数ノンスを使用して暗号化されます。さらに、HomeKit データは、データ保護クラス Protected Until First User Authentication を使用して保存されます。HomeKit データは暗号化されたバックアップにのみバックアップされます。たとえば、暗号化されていない iTunes バックアップに HomeKit データは含まれません。

デバイスとユーザ間のデータ同期

HomeKit データは、iCloud と iCloud キーチェーンを使って、1 人のユーザの iOS デバイス間で同期できます。HomeKit データは、ユーザの HomeKit 識別情報から導出された鍵と乱数ノンスを使用して、同期中に暗号化されます。このデータは、同期中は不透明な BLOB として処理されます。同期を有効にするため最新の BLOB が iCloud に保存されますが、それは他のいかなる目的のためにも用いられません。HomeKit データはユーザの iOS デバイスでのみ利用できる鍵を使って暗号化されるため、転送中や iCloud での保管中にその内容にアクセスすることはできません。

HomeKit データは、同じホームの複数のユーザ間でも同期されます。このプロセスでは、iOS デバイスと HomeKit アクセサリ間で使用されるのと同じ認証と暗号化が使用されます。この認証は、ユーザがホームに追加されたときにデバイス間で交換される Ed25519 公開鍵に基づいています。新しいユーザがホームに追加されると、それ以降のすべての通信が、Station-to-Station プロトコルとセッションごとの鍵を使用して認証および暗号化されます。

新しいユーザを追加できるのは、HomeKit でホームを最初に作成したユーザか、編集権限のある別のユーザです。所有者のデバイスは、アクセサリが新しいユーザを認証し、新しいユーザからのコマンドを受け付けることができるように、新しいユーザの公開鍵を使ってアクセサリを構成します。編集権限のあるユーザが新しいユーザを追加すると、このプロセスはホームハブに委任されて処理が完了します。

ユーザが iCloud にサインインすると、Apple TV を HomeKit で使用するためのプロビジョニングプロセスが自動的に実行されます。iCloud アカウントでは 2 ファクタ認証を有効にしておく必要があります。Apple TV と所有者のデバイスは、一時的な Ed25519 公開鍵を iCloud 経由で交換します。所有者のデバイスと Apple TV が同じローカルネットワーク上にあるとこの一時鍵が使用され、ローカルネットワークでの接続は Station-to-Station プロトコルとセッションごとの鍵によってセキュリティ保護されます。このプロセスでは、iOS デバイスと HomeKit アクセサリ間で使用されるのと同じ認証と暗号化が使用されます。このセキュリティ保護されたローカル接続を経由して、所有者のデバイスは Apple TV にユーザの Ed25519 公開／秘密鍵ペアを転送します。その後、これらの鍵を使用して Apple TV と HomeKit アクセサリとの通信がセキュリティ保護されます。また、Apple TV と HomeKit ホームの一部であるその他の iOS デバイスとの通信もセキュリティ保護されます。

複数のデバイスを使用していないユーザが、追加ユーザによる各自のホームへのアクセスを許可しない場合、HomeKit データは iCloud に同期されません。

ホームデータと App

App からのホームデータへのアクセスは、ユーザの「プライバシー」設定で制御されます。App がホームデータを要求すると、「連絡先」、「写真」、その他の iOS データソースの場合と同様、ユーザにアクセスの許可が求められます。ユーザが承認すると、部屋の名前のリスト、アクセサリの名前のリスト、各アクセサリが存在する部屋、その他の情報に App からアクセスできるようになります。詳しくは、developer.apple.com/homekit の HomeKit に関するデベロッパ向けマニュアルを参照してください。

HomeKit と Siri

Siri は、アクセサリに対するクエリと制御、およびシーンの起動に使用できます。Siri には、ホームの構成に関する最小限の情報が匿名で提供されます。コマンドを認識するには、部屋の名前のリスト、アクセサリ、およびシーンが必要だからです。Siri に送られた音声は特定のアクセサリまたはコマンドを示しますが、Siri のこうしたデータが HomeKit などの Apple のその他の機能に関連付けられることはありません。詳しくは、この文書の「インターネットサービス」セクションの「Siri」を参照してください。

HomeKit アクセサリへの iCloud リモートアクセス

HomeKit アクセサリは、Bluetooth や Wi-Fi が使用できない場合、iCloud に直接接続して iOS デバイスがアクセサリを制御できるようにします。

iCloud リモートアクセスはセキュリティを考慮して設計されているため、アクセサリを制御したりアクセサリから通知を送信したりするときに、アクセサリ自体の情報や送信中のコマンドおよび通知の内容が Apple に公開されることはありません。HomeKit は自宅に関する情報を iCloud リモートアクセス経由で送信しません。

ユーザが iCloud リモートアクセスを使ってコマンドを送信するときは、アクセサリと iOS デバイスが相互に認証し、ローカル接続で説明した手順と同じ方法でデータが暗号化されます。通信内容は暗号化されるため、Apple がその内容を見ることはできません。iCloud 経由でのアドレス指定は、設定プロセス中に登録された iCloud 識別子に基づきます。

iCloud リモートアクセスをサポートしているアクセサリは、アクセサリの設定プロセス時にプロビジョニングされます。プロビジョニングプロセスは、ユーザが iCloud にサインインすると開始されます。次に、iOS デバイスは、Built for HomeKit アクセサリに内蔵されている Apple 認証コプロセッサを使ってチャレンジに署名するようにアクセサリに要求します。アクセサリは prime256v1 楕円曲線鍵も生成し、署名されたチャレンジおよび認証コプロセッサの X.509 証明書と共に公開鍵が iOS デバイスに送信されます。これらは、iCloud プロビジョニングサーバからアクセサリの証明書を要求するために使用されます。証明書はアクセサリに保存されますが、

HomeKit iCloud リモートアクセスへのアクセス権が付与されているという情報を除き、アクセサリを特定する情報は含まれません。また、プロビジョニングを実行中の iOS デバイスはアクセサリにバッグも送信します。このバッグには、iCloud リモートアクセス・サーバへの接続に必要な URL などの情報が含まれています。この情報はユーザやアクセサリに特有のものではありません。

各アクセサリは、許可したユーザのリストを iCloud リモートアクセス・サーバに登録します。これらのユーザは、アクセサリを自宅に追加した人によってアクセサリを制御する権限が付与されたユーザです。ユーザには iCloud サーバによって識別子が付与されます。また、アクセサリからの通知メッセージおよび応答を配信する目的で iCloud アカウントにユーザをマップすることもできます。同様に、アクセサリには iCloud から発行された識別子が付与されますが、これらの識別子は不明瞭な情報となっているため、アクセサリ自体のことについては何も分かりません。

アクセサリは HomeKit iCloud リモートアクセス・サーバに接続するときに証明書とパスを提示します。このパスは別の iCloud サーバから取得されたもので、各アクセサリに固有のパスではありません。アクセサリがパスを要求するとき、その要求にはアクセサリのメーカー名、モデル名、およびファームウェアバージョンが含まれます。この要求では、ユーザまたは自宅を特定する情報は送信されません。プライバシーを保護するため、パスサーバへの接続は認証されません。

アクセサリが iCloud リモートアクセス・サーバに接続するときは、HTTP/2 が使用され、TLS 1.2（AES-128-GCM と SHA-256 を使用）によってセキュリティが確保されます。アクセサリから iCloud リモートアクセス・サーバへの接続は開いたままになるため、アクセサリは着信メッセージを受信したり、応答や発信メッセージを iOS デバイスに送信したりできます。

SiriKit

Siri は iOS 拡張機能メカニズムを利用して他社製 App と通信します。Siri は iOS 連絡先とデバイスの現在位置にアクセスできますが、拡張機能を含む App にそれらの情報を提供するときは、まず App のアクセス権を調べて、iOS が保護するユーザデータへのアクセスが許可されているかどうかを確認します。Siri は元のユーザ・クエリ・テキストの関連部分のみを拡張機能に渡します。たとえば、App に iOS 連絡先へのアクセス権がない場合、Siri は「PaymentApp を使って母さんに 10 ドルを支払って」などのユーザリクエスト内の関係を解決しません。この場合、拡張機能の App は、App に渡される生の音声部分でのみ「母さん」を認識します。一方、App に iOS 連絡先へのアクセス権がある場合、App はユーザの母についての iOS 連絡先情報を受け取ります。たとえば「兄さんはすごいよと MessageApp で母さんに伝えて」など、連絡先がメッセージ本文で言及されている場合、Siri は App の TCC に関係なく「兄さん」を解決しません。App によって提示されるコンテンツは、ユーザが App で使用する言葉を Siri が認識できるようにするため、サーバに送られる場合があります。

Siri は実行時に SiriKit 対応 App がアプリケーションインスタンスに固有のカスタムワードを提供することを許可します。これらのカスタムワードは「Siri」セクションで述べられているランダム識別子に関連付けられていて、同じライフタイムを持ちます。

HealthKit

HealthKit は、ユーザの許可を得てヘルスケアおよびフィットネス App のデータを保存および集計します。HealthKit は、互換性のある Bluetooth LE 心拍モニタのようなヘルスケアおよびフィットネスデバイスや、多くの iOS デバイスに内蔵されているモーションコプロセッサとも直接連係します。

ヘルスケアデータ

HealthKit は、身長、体重、歩いた距離、血圧などのユーザのヘルスケアデータを保存および集計します。このデータはデータ保護クラス Complete Protection に保存されます。つまり、このデータには、ユーザがパスコードを入力するか Touch ID を使用してデバイスをロック解除してはじめてアクセスできます。

HealthKit は、App 用アクセス許可、HealthKit に接続されているデバイスの名前のリスト、新しいデータが利用可能になった時点で App を起動するためのスケジュール情報などの管理データの集計も行います。これらのデータは、データ保護クラス Protected Until First User Authentication に保存されます。

ユーザが運動しているときなど、デバイスがロックされている間に生成されるヘルスケアレコードは、一時ジャーナルファイルに保存されます。これらのデータは、データ保護クラス Protected Unless Open に保存されます。デバイスがロック解除されると、これらのデータが、主要なヘルスケアデータベースに読み込まれ、結合の完了後に削除されます。

ヘルスケアデータがデバイス間で同期されることはありません。ヘルスケアデータは、iCloud へのデバイスバックアップや暗号化された iTunes バックアップに含められます。ヘルスケアデータは、暗号化されていない iTunes バックアップには含められません。

データの完全性

データベースに保存されるデータには、各データレコードの出自を追跡するためのメタデータが含まれます。このメタデータには、当該レコードを保存した App を特定するアプリケーション識別子が含まれます。加えて、オプションのメタデータ項目に当該レコードのデジタル署名されたコピーを含めることができます。これは、信頼できるデバイスによって生成されたレコードにデータの完全性を付与するためです。デジタル署名に使用されるフォーマットは、IETF RFC 5652 で定められている暗号メッセージ構文（CMS）です。

他社製 App からのアクセス

HealthKit API へのアクセスはエンタイトルメントで制御されます。App は、データの利用方法に関する制限に従う必要があります。たとえば、App でヘルスケアデータを広告に利用することはできません。ヘルスケアデータの利用について詳細に規定したプライバシーポリシーを App でユーザに提示することも要求されます。

App からのヘルスケアデータへのアクセスは、ユーザの「プライバシー」設定で制御されます。App がヘルスケアデータへのアクセスを要求すると、「連絡先」、「写真」、その他の iOS データソースの場合と同様、ユーザにアクセスの許可が求められます。ただし、ヘルスケアデータでは、データの種類ごとに別々にアクセスが許可されることに加え、データの読み取りと書き込みも別々にアクセスが許可されます。ユーザは、「ヘルスケア」App の「ソース」タブで、ヘルスケアデータのアクセスに関して付与した権限を確認および取り消すことができます。

App にデータの書き込み権限が付与されている場合は、書き込んだデータを読み取ることもできます。データの読み取り権限が付与されている場合は、すべてのソースによって書き込まれたデータを読み取ることができます。ただし、App から、ほかの App に付与されたアクセス権を調べることはできません。また、App 側でその App にヘルスケアデータの読み取りアクセス権が付与されたかどうかを確定的に知る方法はありません。App に読み取り権限がない場合は、どのクエリでもデータが返されません。これは、空のデータベースからの応答と同じです。これは、App がユーザの追跡しているデータの種類を知って、ユーザの健康状態を推測するのを防ぐためです。

メディカル ID

「ヘルスケア」App には、医療上の緊急事態時に重要となり得る情報をメディカル ID フォームに入力しておくオプションがあります。この情報は入力、更新とも手動で行い、ヘルスケアデータベースの情報との同期は行われません。

メディカル ID 情報は、ロック画面の緊急ボタンをタップすると表示されます。この情報はデータ保護クラス No Protection を使用してデバイスに保存されているため、デバイスのパスコードを入力しなくてもアクセスできます。メディカル ID は、安全とプライバシー両方の懸念のバランスをどのように取るかをユーザが自分で決定できるオプション機能です。

ReplayKit

ReplayKit はデベロッパが App に録画とライブ・ブロードキャストの機能を追加することを許可するフレームワークです。また、ユーザがデバイス前面側のカメラとマイクを使用して録画およびブロードキャストに注釈することを許可します。

ムービーの収録

ムービーの収録にはいくつかのセキュリティ層が埋め込まれています：

- **アクセス権ダイアログ**：収録が始まる前に、ReplayKit はユーザに同意を求める警告を表示して、ユーザが画面、マイク、および前面側カメラを収録する目的を確認されます。この警告は App プロセスごとに 1 回提示され、App がバックグラウンド内で 8 分超経過した場合にも提示されます。
- **画面および音声の取り込み**：画面および音声の取り込みは App のプロセス内ではなく ReplayKit デーモン replayd 内で発生します。これによって、収録されたコンテンツが App プロセスからアクセスすることがないことが保証されます。
- **ムービーの作成および保存**：ムービーファイルは ReplayKit のサブシステムからのみアクセスできるディレクトリに書き込まれるので、App からはアクセスできません。これによって、収録がユーザの同意を得ずに第三者によって使用されることが防止されます。
- **エンドユーザによるプレビューおよび共有**：ユーザは ReplayKit によって提供される UI を使用してムービーをプレビューおよび共有できます。UI は iOS 拡張機能インフラストラクチャを通じてアウトオブプロセスで提示され、生成されたムービーファイルにアクセスできます。

ブロードキャスト

- **画面および音声の取り込み**：ブロードキャスト中の画面および音声取り込みは、ムービーの収録と同様に replayd 内で実行されます。
- **ブロードキャスト拡張機能**：ReplayKit ブロードキャストに参加する他社製サービスの場合、com.apple.broadcast-services エンドポイントで構成される新しい 2 つの拡張機能を作成する必要があります：
  - ユーザがブロードキャストを設定することを許可する UI 拡張機能。
  - ビデオおよび音声データをサービスのバックエンドサーバにアップロードすることを扱うアップロード拡張機能。
  - アーキテクチャによって、ホスト側 App が、ブロードキャストされるビデオおよび音声コンテンツへのアクセス権を持たないことが保証されます。ReplayKit と他社製ブロードキャスト拡張機能のみがアクセス権を持ちます。
- **ブロードキャストピッカー**：ReplayKit には、使用するブロードキャストサービスを選択するために、デベロッパが App 内で提示できるビューコントローラ（UIActivityViewController と同様）が用意されています。ビューコントローラは UIRemoteViewController SPI を使用して実装され、ReplayKit フレームワーク内で動作する拡張機能です。ホスト側 App からはアウトオブプロセスです。

- **アップロード拡張機能**：ブロードキャスト中のビデオおよび音声コンテンツを扱うために他社製ブロードキャストサービスが実装するアップロード拡張機能は、2 つの方法でコンテンツを受信することを選択できます：
  - エンコードされた小さな MP4 クリップ。
  - エンコードされていない生のサンプルバッファ。
    - **MP4 クリップの処理**：この処理モードでは、エンコードされた小さな MP4 クリップが replayd によって生成され、ReplayKit のサブシステムからのみアクセス可能なプライバシーが保護された場所に保存されます。ムービークリップが生成されると、replayd は NSExtension リクエスト SPI（XPC ベース）を通じて他社製アップロード拡張機能にムービークリップの場所を渡します。replayd は、ワンタイム・サンドボックス・トークンも生成し、これもアップロード拡張機能に渡されます。このトークンは、拡張機能リクエスト中に特定のムービークリップへの拡張機能アクセスを許可します。
    - **サンプルバッファの処理**：この処理モードでは、ビデオおよび音声データはシリアライズされ、直接 XPC 接続を通じて他社製アップロード拡張機能にリアルタイムで渡されます。ビデオデータはビデオ・サンプル・バッファから IOSurface オブジェクトを抽出することで、エンコードされます。さらに、XPC オブジェクトとして安全にエンコードされ、XPC 経由で他社製拡張機能に送信され、IOSurface オブジェクトへ安全にデコードされます。

保護したメモ

「メモ」App にメモを保護する機能が搭載され、ユーザが特定のメモの内容を保護できるようになりました。保護したメモはユーザが設定したパスフレーズで暗号化され、iOS、macOS、iCloud Web サイトのメモを表示するにはこのパスフレーズが必要になります。

ユーザがメモを保護すると、16 バイト鍵が PBKDF2 および SHA256 を使用してユーザのパスフレーズから導出されます。メモの内容は AES-GCM を使用して暗号化されます。新しいレコードが Core Data および CloudKit 内に作成され、暗号化されたメモ、タグ、および初期化ベクトルが保存されます。元のメモのレコードは削除されるため、暗号化されたデータが書き込まれることはありません。また、添付ファイルも同じように暗号化されます。サポートされる添付ファイルは、イメージ、スケッチ、マップ、および Web サイトです。ほかの種類の添付ファイルを含むメモは暗号化できず、サポートされていない添付ファイルを、保護したメモに追加することはできません。

保護したメモを表示または作成するときに、ユーザがパスフレーズを正しく入力すると、「メモ」がセキュアなセッションを開始します。セッション中は、ほかのメモを表示したり保護したりするのに、パスフレーズの入力や Touch ID の利用を求められることはありません。ただし、異なるパスフレーズのメモがある場合、セキュアなセッションは現在のパスフレーズで保護されているメモにのみ適用されます。ユーザが「メモ」で「今すぐロック」ボタンをタップするか、「メモ」がバックグラウンドに切り替えられてから 3 分を超えるか、またはデバイスがロックされると、セキュアなセッションは終了します。

ユーザがパスフレーズを忘れても、Touch ID がデバイスで有効になっていれば、保護したメモを表示したりほかのメモを保護したりできます。また、パスフレーズの入力に 3 回失敗すると、ユーザが設定したヒントが表示されます。パスフレーズを変更するには、現在のパスフレーズを知っている必要があります。

現在のパスフレーズを忘れた場合は、そのパスフレーズをリセットできます。この機能では、新しいメモを新しいパスフレーズで保護することはできますが、以前に保護したメモを表示することはできません。以前に保護したメモを表示するには、古いパスフレーズを思い出す必要があります。パスフレーズをリセットするには、ユーザの iCloud アカウントのパスフレーズが必要です。

メモはほかのユーザと共有できます。メモデータは暗号化された状態で保存され、CloudKit は参加者が互いのデータを暗号化および復号するプロセスを管理します。

Apple Watch

Apple Watch は、iOS 用に構築されたセキュリティ機能とセキュリティ技術を使用して、デバイス上のデータを保護したり、ペアリングされた iPhone やインターネットと通信したりできるようにします。これには、データ保護やキーチェーンアクセス制御などの技術が含まれます。ユーザのパスコードも、暗号化鍵を作成するためにデバイス UID と関連付けられます。

Apple Watch と iPhone のペアリングは、アウトオブバンド（OOB）プロセスで公開鍵を交換し、その後、BTLE リンクの共有シークレットを使用して保護されます。Apple Watch には、iPhone のカメラで読み取るためのアニメーションパターンが表示されます。このパターンには、BTLE 4.1 のアウトオブバンドのペアリングに使用されるエンコードされたシークレットが含まれています。必要に応じて、代替ペアリング方式として標準の BTLE パスキー入力を使用できます。

BTLE セッションが確立されると、Apple Watch と iPhone は、この文書の「iMessage」セクションに説明されている IDS のプロセスを使用して鍵を交換します。鍵が交換されると、Bluetooth セッション鍵が破棄され、Apple Watch と iPhone 間のすべての通信が IDS を使用して暗号化されます。また、暗号化された BTLE と Wi-Fi のリンクも二次的な暗号化レイヤーを提供します。トラフィックが危殆化した場合に備えて、鍵は 15 分間隔で変更されるため、鍵の露出時間は制限されます。

ストリーミングデータを必要とする App をサポートするため、この文書の「インターネットサービス」セクションの「FaceTime」に説明されている方式で暗号化が提供されます。この方式では、ペアリングされた iPhone が提供する IDS サービスが使用されます。

Apple Watch には、この文書の「暗号化とデータ保護」セクションに説明されているように、ハードウェアで暗号化されたストレージとファイル／キーチェーン項目のクラスベースの保護が実装されています。また、キーチェーン項目用のアクセス制御されたキーバッグも使用されます。Apple Watch と iPhone 間の通信に使用される鍵も、クラスベースの保護を使用して保護されます。

Apple Watch が Bluetooth の通信範囲内にない場合は、代わりに Wi-Fi を使用できます。Apple Watch は、ペアリングされた iPhone 上にすでに資格情報（以前に Apple Watch に同期されている必要があります）が提示されている場合にのみ、Wi-Fi ネットワークに参加します。Apple Watch が iPhone 通信圏から外れると、iPhone 上の新しいネットワーク資格情報は Apple Watch 上にありません。

Apple Watch を手動でロックするには、サイドボタンを押したままにします。また、Apple Watch を手首から外すとすぐに、モーションヒューリスティックによって自動ロックが試みられます。ロックされている状態では、Apple Pay は使用できません。手首検出によって機能する自動ロックが設定でオフにされている場合、Apple Pay は無効になります。手首検出をオフにするには、iPhone で「Apple Watch」App を使用します。また、モバイルデバイス管理を使用してこの設定を強制的に適用することもできます。

Apple Watch を装着している場合は、ペアリングされた iPhone を使用して Apple Watch のロックを解除することもできます。これは、ペアリング中に確立された鍵を使用して認証される接続を確立することによって行われます。iPhone がこの鍵を送信すると、Apple Watch がこの鍵を使用してデータ保護鍵のロックを解除します。Apple Watch のパスコードは iPhone 側では把握しておらず、転送されることもありません。この機能をオフにするには、iPhone で「Apple Watch」App を使用します。

Apple Watch がペアリングできる iPhone は一度に 1 台のみです。ペアリングを解除すると、iPhone の指示により、Apple Watch からすべてのコンテンツとデータが消去されます。

ペアリングされた iPhone で「iPhone を探す」を有効にすると、Apple Watch 上でもアクティベーションロックの使用が許可されます。アクティベーションロックにより、Apple Watch の紛失または盗難時に、その Apple Watch を他人が使用または売却することが困難になります。アクティベーションロックが有効になっている場合、Apple Watch のペアリング解除、消去、再アクティベーションにはそのユーザの Apple ID とパスワードが必要になります。

ネットワークのセキュリティ

iOS デバイス上の保存データの保護に使用される内蔵の保護機能に加えて、iOS デバイスを行き来する情報のセキュリティを確保するために、組織は数多くのネットワークセキュリティ対策を実施できます。

モバイルユーザは世界のどこからでも企業ネットワークにアクセスできなければなりません。そのため、ユーザの認証と、データ転送時の保護が確実に行われることが重要です。iOS では、認証、承認、および暗号化された通信に対して標準的なネットワークプロトコルが使用され、デベロッパもこれらのプロトコルにアクセスすることができます。これらのセキュリティ上の目標を達成するために、iOS では、Wi-Fi とモバイルデータ通信ネットワークの両方の接続で、実績のあるテクノロジーと最新の標準規格が統合されています。

ほかのプラットフォームでは、オープンな通信ポートを侵害から保護するためにファイアウォールソフトウェアが必要になります。iOS では、リスニングポートを制限し、Telnet、シェル、Web サーバといった不要なネットワークユーティリティを省くことで、攻撃対象領域の削減に成功しているため、iOS デバイスには追加のファイアウォールソフトウェアは必要ありません。

TLS

iOS は、Transport Layer Security（TLS v1.0、TLS v1.1、TLS v1.2。AES 128 と SHA-2 の両方をサポートします）、および DTLS をサポートします。Safari、カレンダー、メールなどのインターネット App では、自動的にこれらのメカニズムを使用して、デバイスとネットワークサービスの間の通信チャンネルの暗号化を有効にしています。ハイレベル API（CFNetwork など）により、デベロッパは App に TLS を簡単に導入することができるほか、ローレベル API（SecureTransport）によってきめの細かい制御が可能になっています。CFNetwork は SSLv3 の使用を許可せず、Safari など WebKit を使用する App は SSLv3 接続の確立が禁止されます。

iOS 10 および macOS Sierra では RC4 対称暗号スイートが非推奨になっています。デフォルトでは、SecureTransport API を使って実装された TLS クライアントまたはサーバで、RC4 暗号スイートが無効になっています。また、利用できる暗号スイートが RC4 しかない場合は、接続できなくなります。安全性を高めるため、RC4 を必要とするサービスまたは App をアップグレードして、最新の安全な暗号スイートを使用してください。

App Transport Security

App Transport Security はデフォルトの接続要件を規定する機能です。これにより、CFURL または NSURLSession の各 API の使用時に、App が最良の方法で安全な接続を行うことが確実になります。デフォルトでは、App Transport Security は Forward Secrecy（前方秘匿性）を提供するスイートのみを含むように暗号選択を制限します。具体的には、GCM または CBC モードでの ECDHE_ECDSA_AES および ECDHE_RSA_AES です。App はドメインごとに Forward Secrecy 要件を無効にできます。この場合、利用可能な暗号のセットに RSA_AES が追加されます。

サーバは TLS 1.2 と Forward Secrecy（前方秘匿性）をサポートしている必要があり、2048 ビット以上の RSA 鍵または 256 ビット以上の楕円曲線鍵を用いた SHA-256 以上を使って署名された有効な証明書も必要です。

App で App Transport Security が無効になっている場合を除き、これらの要件を満たさないネットワーク接続は失敗します。証明書が無効な場合は必ず失敗し、接続は確立されません。App Transport Security は iOS 9 以降向けにコンパイルされた App に自動的に適用されます。

VPN

仮想プライベートネットワークなどの安全なネットワークサービスは、通常、最小限の設定と構成だけで、iOS デバイスで使用できるようになります。iOS デバイスは、以下のプロトコルと認証方法をサポートする VPN サーバと通信できます：

- IKEv2/IPSec（共有シークレット、RSA 証明書、ECDSA 証明書、EAP-MSCHAPv2、または EAP-TLS による認証）。
- SSL-VPN（App Store からの適切なクライアント App を使用）。
- Cisco IPSec（パスワード、RSA SecurID、または CRYPTOCard によるユーザ認証、および共有シークレットと証明書によるコンピュータ認証）。
- L2TP/IPSec（MS-CHAPV2 パスワード、RSA SecurID、または CRYPTOCard によるユーザ認証、および共有シークレットによるコンピュータ認証）。
- PPTP は iOS 9.3 以前でサポートされていますが、推奨されません。

iOS は、証明書ベースの認証を使用するネットワークでの VPN オンデマンドをサポートしています。IT ポリシーにより、VPN 接続が必要なドメインが構成プロファイルを使って指定されます。

iOS は Per App VPN もサポートしているので、これを利用すれば VPN 接続を非常に細かく設定することができます。モバイルデバイス管理（MDM）では、各管理対象 App や「Safari」の特定のドメインの接続を指定できます。これにより、セキュアなデータは常に企業ネットワークを経由し、ユーザの個人データは企業ネットワークを経由しないようにすることができます。

iOS は VPN 常時接続をサポートしています。これは、MDM で管理され、「Apple Configurator」または Device Enrollment Program で監視されているデバイスに構成できます。これにより、モバイルデータ通信ネットワークおよび Wi-Fi ネットワークに接続するときに、保護を有効にするためにユーザが VPN をオンにする必要がなくなります。VPN 常時接続では、組織に戻されるすべての IP トラフィックがトンネリングされるので、組織はデバイストラフィックを完全に制御できます。デフォルトのトンネリングプロトコルである IKEv2 は、データの暗号化によってトラフィックの転送を保護します。組織では、デバイスを行き来するトラフィックを監視およびフィルタリングしたり、ネットワーク内のデータをセキュリティ保護したり、デバイスからインターネットへのアクセスを制限したりできます。

Wi-Fi

iOS は、WPA2 エンタープライズなどの業界標準の Wi-Fi プロトコルをサポートしているので、企業のワイヤレスネットワークへの認証を用いたアクセスが可能になります。WPA2 エンタープライズは、128 ビットの AES 暗号化を採用しているため、ユーザは Wi-Fi ネットワーク接続での送受信時に最も確実にデータ保護を維持することができます。iOS デバイスは 802.1X に対応しているため、さまざまな RADIUS 認証環境に統合できます。iPhone および iPad がサポートしている 802.1X ワイヤレス認証方法には、EAP-TLS、EAP-TTLS、EAP-FAST、EAP-SIM、PEAPv0、PEAPv1、および LEAP があります。

データの保護に加え、iOS は 802.11w で言及されている「保護された管理フレーム（PMF）」サービスを通じて、WPA2 レベル保護をユニキャストおよびマルチキャスト管理フレームに拡張します。PMF サポートは iPhone 6s および iPad Air 2 以降で利用可能です。

iOS は、Wi-Fi ネットワークに関連付けられていない状態で Wi-Fi スキャンを実行するときに、ランダム化された Media Access Control（MAC）アドレスを使用します。これらのスキャンは、優先する Wi-Fi ネットワークを検索して接続するため、またはジオフェンスを使用する App の位置情報サービスを支援するため（位置情報に基づくリマインダーの使用時や「マップ」App での位置情報の修正時など）に実行されることがあります。優先する Wi-Fi ネットワークへの接続時に実行される Wi-Fi スキャンはランダム化されないことに注意してください。

デバイスが Wi-Fi ネットワークに関連付けられていないか、デバイスのプロセッサがスリープ状態にある場合、iOS は、拡張 Preferred Network Offload（ePNO）スキャンの実行時にもランダムな MAC アドレスを使用します。ePNO スキャンは、位置情報に基づくリマインダーでデバイスが特定の場所の近くにあるかどうかを判定する場合など、ジオフェンスを使用する App がデバイスの位置情報サービスを利用する際に実行されます。

Wi-Fi ネットワークとの接続が解除されているときにデバイスの MAC アドレスが変更されるようになったため、Wi-Fi トラフィックのパッシブなオブザーバは、MAC アドレスを使ってデバイスを継続的に追跡できません。これは、デバイスがモバイルデータ通信ネットワークに接続されている場合も同様です。Apple は、iOS Wi-Fi スキャンがランダムな MAC アドレスを使用すること、および Apple にもメーカーにもランダムな MAC アドレスの予測は不可能であることを Wi-Fi メーカーにお知らせしてきました。Wi-Fi MAC アドレスのランダム化のサポートは、iPhone 4s 以前では利用できません。

iPhone 6S 以降では、既知の Wi-Fi ネットワークの隠されているプロパティは自動的に既知になり更新されます。Wi-Fi ネットワークの SSID（Service Set Identifier）がブロードキャストされる場合、iOS デバイスは SSID がリクエストに含まれた状態のプローブを送信しません。これによって、隠されていないネットワークのネットワーク名をデバイスがブロードキャストすることが防止されます。

Bluetooth

iOS の Bluetooth サポートは、プライベートデータへの不要なアクセスを増大させることなく、便利な機能を提供するように設計されています。iOS デバイスは、Encryption Mode 3、Security Mode 4、および Service Level 1 の接続をサポートしています。iOS は以下の Bluetooth プロファイルをサポートしています：

- Hands-Free Profile（HFP 1.5）
- Phone Book Access Profile（PBAP）
- Advanced Audio Distribution Profile（A2DP）
- Audio/Video Remote Control Profile（AVRCP）
- Personal Area Network Profile（PAN）
- Human Interface Device Profile（HID）

これらのプロファイルのサポートは、デバイスによって異なります。詳しくは、support.apple.com/ja-jp/HT204387 を参照してください。

シングルサインオン

iOS では、企業ネットワークへの認証にシングルサインオン（SSO）を使用できます。SSO は Kerberos ベースのネットワークに対応しており、アクセスが承認されているサービスに対してユーザを認証します。SSO は、セキュアな Safari セッションから他社製 App まで、さまざまなネットワークアクティビティで使用できます。証明書ベースの認証（PKINIT）もサポートされています。

iOS の SSO は、SPNEGO トークンと HTTP Negotiate プロトコルを利用して、Kerberos ベースの認証ゲートウェイや、Kerberos チケットをサポートする Windows 統合認証システムで動作します。SSO サポートは、オープンソースの Heimdal プロジェクトに基づいています。

以下の暗号化タイプがサポートされています：

- AES128-CTS-HMAC-SHA1-96
- AES256-CTS-HMAC-SHA1-96
- DES3-CBC-SHA1
- ARCFOUR-HMAC-MD5

「Safari」は SSO をサポートしています。また、標準の iOS ネットワーク API を使用する他社製 App についても、SSO を使用するように構成できます。SSO を構成するために、iOS は、MDM サーバが必要な設定をプッシュできるようにする構成プロファイルペイロードをサポートしています。このペイロードは、ユーザのプリンシパル名（つまり Active Directory ユーザアカウント）や Kerberos 領域を設定できるだけでなく、SSO の使用を許可する必要のある App や「Safari」の Web URL を構成できます。

AirDrop のセキュリティ

AirDrop をサポートしている iOS デバイスは、Bluetooth Low Energy（BLE）と Apple 製のピアツーピア Wi-Fi テクノロジーを使用して、OS X 10.11 以降を搭載の AirDrop 対応 Mac コンピュータなどの近くのデバイスにファイルや情報を送信します。Wi-Fi 通信を使用して、インターネット接続や Wi-Fi アクセスポイントを使用せずにデバイス間で直接通信します。

ユーザが AirDrop を有効にすると、2048 ビットの RSA 識別情報がデバイスに保存されます。また、ユーザの Apple ID に関連付けられたメールアドレスと電話番号を基に、AirDrop 識別情報のハッシュが作成されます。

ユーザが項目の共有方法として AirDrop を選択すると、デバイスが Bluetooth Low Energy 経由で AirDrop 信号を発信します。スリープが解除され AirDrop がオンになっている別のデバイスが近くにあり、そのデバイスがこの信号を検出すると、所有者の識別情報のハッシュの短縮バージョンを使って応答します。

AirDrop は、デフォルトでは「連絡先のみ」と共有するように設定されています。AirDrop を使って「すべての人」と共有できるようにすることも、この機能を完全にオフにすることもできます。「連絡先のみ」モードでは、受信した識別情報のハッシュがイニシエータの「連絡先」内の人のハッシュと照合されます。一致が見つかると、送信デバイスがピアツーピア Wi-Fi ネットワークを作成して、Bonjour を使って AirDrop 接続をアドバタイズします。この接続を使って、受信デバイスは識別情報の完全なハッシュをイニシエータに送信します。完全なハッシュ値も「連絡先」内の情報と一致した場合は、受信者の下の名前と写真（「連絡先」にある場合）が AirDrop の共有シートに表示されます。

AirDrop を使用するときは、送信ユーザが共有したい相手を選択します。送信デバイスが、暗号化された（TLS）接続を受信デバイスと開始し、そこで iCloud 識別情報の証明書が交換されます。証明書内の識別情報は、各ユーザの「連絡先」と照合して検証されます。その後、受信ユーザは、識別情報が確認されたユーザまたはデバイスからの受信データの承諾を求められます。複数の受信者が選択された場合は、このプロセスが送信先ごとに繰り返されます。

「すべての人」モードでは、同じプロセスが使用されますが、「連絡先」での一致が見つからなかった場合は、AirDrop の送信シートに受信デバイスがシルエットとデバイス名（「設定」>「一般」>「情報」>「名前」で定義されているもの）で表示されます。

組織は、モバイルデバイス管理ソリューションで管理されているデバイスまたは App で AirDrop の使用を制限できます。

Apple Pay

サポートされている iOS デバイスおよび Apple Watch のユーザは、Apple Pay を利用して簡単、安全、プライベートな方法で、店舗や App、「Safari」の Web での支払いができます。Apple Pay は、ユーザにとってはシンプルであり、ハードウェアとソフトウェアの両方に統合されたセキュリティを使用して構築されています。

Apple Pay は、ユーザの個人情報が保護される設計にもなっています。ユーザが特定される可能性のある取引情報を一切収集しません。支払い取引は、ユーザ、加盟店、およびカード会社間で行われます。

Apple Pay のコンポーネント

Secure Element：Secure Element は、Java Card プラットフォームを実行する業界標準の認定チップで、電子決済に対する金融業界の要件に準拠しています。

NFC コントローラ：NFC コントローラは近距離無線通信プロトコルをサポートし、アプリケーションプロセッサと Secure Element 間、および Secure Element と POS 端末間の情報を送信します。

Wallet：「Wallet」は、クレジットカード／デビットカード／ポイントカード／ショップカードの追加と管理、および Apple Pay による支払いに使用されます。ユーザは自分のカード、取引カード会社、取引カード会社のプライバシーポリシー、最近の取引明細、その他の情報を「Wallet」で確認できます。設定アシスタントと「設定」で、カードを Apple Pay に追加することもできます。

Secure Enclave：iPhone と iPad、Apple Watch Series 1 および Series 2 では、Secure Enclave がその認証プロセスを管理し、支払い取引の続行を許可します。Secure Enclave は、Touch ID 用の指紋データを保存します。

Apple Watch では、デバイスのロックを解除する必要があります。解除するにはサイドボタンをダブルクリックします。ダブルクリック動作が検出されると、そのダブルクリックはアプリケーションプロセッサを経由せず、Secure Element または利用可能な場合は Secure Enclave に直接渡されます。

Apple Pay Server：Apple Pay Server は、「Wallet」のクレジットカード／デビットカードの状態、および Secure Element に格納されているデバイスのアカウント番号を管理します。Apple Pay Server はデバイス、決済ネットワークのサーバの双方と通信します。Apple Pay Server は、App 内での支払いに使用する支払い資格情報の再暗号化も行います。

Apple Pay が Secure Element を利用する方法

Secure Element では、Apple Pay を管理するために特別に設計されたアプレットをホストします。Secure Element には、決済ネットワークによって認定された決済アプレットも含まれています。クレジットカード／デビットカード／プリペイドカードのデータは、決済ネットワークまたはカード会社からこれらの決済アプレットに送信されますが、その際、決済ネットワークと決済アプレットのセキュリティドメインしか知らない鍵によって暗号化されます。このデータは決済アプレット内に保存され、Secure Element のセキュリティ機能を使って保護されます。取引の間中、決済用端末は専用のハードウェアバスを使用して近距離無線通信（NFC）コントローラ経由で Secure Element と直接通信します。

Apple Pay が NFC コントローラを利用する方法

NFC コントローラは Secure Element へのゲートウェイとして、非接触型決済のすべての取引が、デバイスの近くにある POS 端末を使用して実行されることを確実にします。NFC コントローラは、フィールド内の端末から着信する支払い要求にのみ、非接触型取引のマークを付けます。

カード保持者が Touch ID またはパスコードを使用するか、あるいはロック解除された Apple Watch ではサイドボタンをダブルクリックして、支払いが承認されると、Secure Element 内の決済アプレットが作成した非接触型応答がコントローラによって排他的に NFC フィールドに配信されます。その結果、非接触型取引の支払い承認の詳細情報は、ローカルの NFC フィールドに格納され、アプリケーションプロセッサに開示されることは決してありません。これに対し、App 内および Web での支払い承認の詳細情報はアプリケーションプロセッサに配信されます。ただし、Apple Pay Server への配信前に必ず Secure Element によって暗号化されます。

クレジットカード、デビットカード、プリペイドカードのプロビジョニング

ユーザがクレジットカード／デビットカード／プリペイドカード（ショップカードを含む）を Apple Pay に追加すると、Apple は、そのカード情報をユーザのアカウントとデバイスについてのほかの情報と共に、該当するカード会社またはカード会社認定のサービスプロバイダにセキュリティで保護して送信します。カード会社はこの情報を使用して、そのカードの Apple Pay への追加を承認するかどうかを決定します。

Apple Pay は、カードのプロビジョニングプロセスの一部として、次の 3 つのサーバ側呼び出しを使用して、カード会社またはネットワークとデータの送受信を行います：Required Fields、Check Card、および Link and Provision。カード会社またはネットワークはこれらの呼び出しを使用して、カードの確認、承認、および Apple Pay への追加を行います。これらのクライアントサーバセッションは SSL を使って暗号化されます。

全体のカード番号は、デバイスにも Apple のサーバにも保存されません。その代わりに、デバイスのアカウント番号が一意に作成され、暗号化された後に Secure Element に保存されます。この固有のデバイスのアカウント番号は、Apple でもアクセスできないような方法で暗号化されます。デバイスのアカウント番号は一意で、通常のクレジットカード／デビットカード番号とは異なるため、カード会社側はクレジットカードやデビットカードの番号を磁気ストライプカード、電話での通話、Web サイトで使用しないようにすることができます。Secure Element 内のデバイスのアカウント番号は iOS および watchOS から切り離されており、Apple のサーバに保存されることは絶対にありません。また、iCloud にバックアップされることもありません。

Apple Watch で使用するカードを登録するには、iPhone で「Apple Watch」App を使用します。Apple Watch 用にカードを登録するには、その Apple Watch が Bluetooth の通信範囲内にあることが必要です。カードは Apple Watch で使用するために登録され、独自のデバイスのアカウント番号を持ちます。デバイスのアカウント番号は、Apple Watch の Secure Element 内に格納されます。

クレジットカード／デビットカード／プリペイドカードを Apple Pay にプロビジョニングする方法は、次の 3 通りです：

- カードを手動で Apple Pay に追加する
- iTunes Store アカウントに記録されているクレジットカード／デビットカードを Apple Pay に追加する
- カード会社の App からカードを追加する

クレジットカード／デビットカードを手動で Apple Pay に追加する

カード（ショップカードを含む）を手動で追加する場合は、名義、クレジットカード番号、有効期限、および CVV が、プロビジョニングプロセスの円滑な実行に使用されます。「設定」、「Wallet」App、または「Apple Watch」App で、それらの情報を手入力または iSight カメラを使用して入力できます。カメラでカード情報を撮影する際に、名義、カード番号、有効期限の取得が試みられ

ます。写真がデバイスに残ったり、フォトライブラリに格納されたりすることは絶対にありません。必要なフィールド値の取得が完了すると、Check Card プロセスが CVV 以外の各フィールドを確認します。情報は暗号化されて Apple Pay Server に送信されます。

Check Card プロセスから利用条件 ID が返されたら、Apple はカード会社の利用条件をダウンロードしてユーザに表示します。ユーザが利用条件に同意すると、Apple は同意を得た利用条件の ID および CVV を Link and Provision プロセスに送信します。このほか、Link and Provision プロセスの一部として Apple は、デバイスからの情報をカード会社またはネットワークと共有します。具体的には、iTunes と App Store のアカウント利用に関する情報（iTunes での長期間の取引記録があるかどうかなど）、デバイスに関する情報（電話番号、デバイスの名前とモデル、Apple Pay の設定に必要なペアリング相手の iOS デバイスなど）、カードを追加したときのおおよその位置（「位置情報サービス」を有効にしている場合）などです。カード会社はこの情報を使用して、そのカードの Apple Pay への追加を承認するかどうかを決定します。

Link and Provision プロセスの結果として以下の 2 つの処理が実行されます。

- デバイスが、クレジットカード／デビットカードを表す Wallet パスファイルのダウンロードを開始する。
- デバイスが、当該カードの Secure Element へのバインドを開始する。

パスファイルには、カードのデザイン、連絡先情報などカードに関するメタデータ、関連するカード会社 App、およびサポート機能をダウンロードするための URL が含まれています。このファイルにはパス状態、つまり、Secure Element のパーソナライズが完了したかどうか、カードが現在、カード会社によって差し止められているかどうか、当該カードを使用して Apple Pay で支払いを行うために追加の検証が必要かどうか、などの情報も含まれています。

iTunes Store アカウントからクレジットカード／デビットカードを Apple Pay に追加する

「iTunes」に登録されているクレジットカード／デビットカードの場合、ユーザは Apple ID パスワードの再入力を求められることがあります。カード番号が「iTunes」から取得され、Check Card プロセスが開始されます。そのカードが Apple Pay で使用できるなら、利用条件がダウンロードされてデバイスに表示されます。その後、利用条件の ID とカード・セキュリティ・コードが Link and Provision プロセスに送られます。登録されている iTunes アカウントのカードには追加の検証が必要になることがあります。

カード会社の App からクレジットカード／デビットカードを追加する

App を Apple Pay で使用するために登録すると、その App と加盟店のサーバ用の鍵が確立されます。これらの鍵は加盟店に送信されるカード情報の暗号化に使用されます。これによって、カード情報が iOS デバイスに読み取られることが防止されます。プロビジョニングの流れは、上で述べたような、手動で追加したカードの場合と同様です。ただし、CVV の代わりにワンタイムパスワードが使用されます。

追加の検証

カード会社は、クレジットカード／デビットカードに追加の検証が必要かどうかを決定できます。カード会社から提供されるサービスの内容によりますが、テキストメッセージ、メール、カスタマーサービスとの通話、承認された他社製 App 内で提供される方法など、追加の検証を行う方法をユーザがさまざまなオプションから選択できる場合があります。テキストメッセージやメールの場合は、カード会社に登録されている連絡先情報からユーザが選択します。その後、コードを受信します。ユーザはこのコードを、「Wallet」、「設定」、または「Apple Watch」App に入力する必要があります。カスタマーサービスや App を使用する検証の場合は、カード会社が独自の通信プロセスを実行します。

支払い承認

Secure Enclave が搭載されたデバイスでは、Secure Element は Secure Enclave から承認を受けた後にのみ支払いを許可します。iPhone または iPad ではさらに、ユーザが Touch ID またはデバイスパスコードで認証されていることを確認する必要があります。利用できる場合には Touch ID がデフォルトの方法ですが、パスコードも Touch ID の代わりにいつでも使用できます。指紋の認証に 3 回失敗すると自動的にパスコードが使用できるようになり、5 回失敗するとパスコードが必須になります。パスコードは、Touch ID が設定されていないか、Apple Pay 対応になっていないときにも要求されます。Apple Watch で支払いを実行するには、パスコードでデバイスのロックを解除し、サイドボタンをダブルクリックする必要があります。

Secure Enclave と Secure Element 間の通信はシリアルインターフェイスを介して行われます。Secure Element が NFC コントローラに接続され、それからアプリケーションプロセッサに接続されます。Secure Enclave と Secure Element は直接接続されてはいませんが、製造工程でプロビジョニングされた共有ペアリング鍵を使用して安全に通信できます。AES をベースにする通信の暗号化と認証は、通信の両側で使用される暗号ノンスでリプレイ攻撃から保護されます。ペアリング鍵は、Secure Enclave の UID 鍵と Secure Element の一意の識別子から Secure Enclave 内で生成されます。生成されたペアリング鍵は、工場で Secure Enclave からハードウェアセキュリティモジュール（HSM）に安全に保護されて転送されます。HSM には、次にペアリング鍵を Secure Element に導入するために必要な鍵マテリアルが用意されています。

ユーザが取引を承認すると、認証の種類に関する署名済みデータと取引の種類（非接触型か App 内か）についての詳細情報が Authorization Random（AR）値に付加されて、Secure Enclave から Secure Element に送信されます。AR は、ユーザがはじめてクレジットカードをプロビジョニングしたときに Secure Enclave 内で生成され、Apple Pay が有効になっている間存続し、Secure Enclave の暗号化およびロールバック防止メカニズムによって保護されます。AR は、ペアリング鍵を使って Secure Element に安全に配信されます。Secure Element は、新しい AR 値を受け取ると、以前に追加されたすべてのカードに削除済みのマークを付けます。

Secure Element に追加されたクレジットカード／デビットカード／プリペイドカードは、同じペアリング鍵とカードの追加時点以降の AR 値を使った承認が Secure Element に提示されない限り、使用できません。これにより、以下のような状況では AR のコピーに無効のマークを付けてカードを使用不能にするように、iOS から Secure Enclave に指示できます：

- パスコードがオフになった。
- ユーザが iCloud からサインアウトした。
- ユーザが「すべてのコンテンツと設定を消去」を選択した。
- デバイスがリカバリモードから復元された。

Apple Watch では、次の場合にカードが無効とマークされます：

- Apple Watch のパスコードが無効になっている。
- Apple Watch が iPhone からペアリング解除された。
- 手首検出がオフになっている。

Secure Element は、非接触型決済の決済アプレットを有効にする前に、ペアリング鍵と現在の AR 値のコピーを使用して、Secure Enclave から受け取った承認を検証します。このプロセスは、App 内取引用に、暗号化された支払いデータを決済アプレットから取得する場合にも適用されます。

取引固有の動的セキュリティコード

決済アプレットから送信されるすべての支払い取引には、デバイスのアカウント番号に加えて取引固有の動的セキュリティコードが含まれています。この 1 回限りのコードは、新しい取引が発生するたびに増分されるカウンタとパーソナライズ時に決済アプレットでプロビジョニングされる鍵を使って計算され、決済ネットワーク、カード会社の両方またはいずれかに通知されます。支払い方式によっては、これらのコードの計算に以下のようなデータも使用されます：

- 決済アプレットで生成される乱数
- 決済用端末で生成される別の乱数（NFC 取引の場合）
- サーバで生成される別の乱数（App 内取引の場合）

これらのセキュリティコードは決済ネットワークとカード会社に送信され、取引の検証に使用されます。セキュリティコードの長さは、実行中の取引の種類によって異なることがあります。

Apple Pay による非接触型決済

動作中の iPhone が NFC フィールドを検出すると、「設定」で管理されている、関連するクレジットカード／デビットカード／プリペイドカードまたはデフォルトカードがユーザに表示されます。ユーザは、「Wallet」App でクレジットカード／デビットカードを選択するか、デバイスのロック中にホームボタンをダブルクリックすることもできます。

次に、ユーザが Touch ID またはパスコードを使用して認証を行います。その後、支払い情報が伝送されます。Apple Watch のロックが解除されているときにサイドボタンをダブルクリックすると、デフォルトのカードが支払い用に有効になります。ユーザの認証がない限り、支払い情報は送信されません。

ユーザが認証すると、デバイスのアカウント番号と取引固有の動的セキュリティコードを使って支払いが処理されます。実際のクレジットカード／デビットカード番号全体が、Apple やユーザのデバイスから加盟店に送信されることはありません。Apple は取引のおおよその時間と場所などを匿名の取引情報として受け取る場合があります。これは、Apple Pay やその他の Apple の製品およびサービスの改善に役立ちます。

Apple Pay による App 内での支払い

Apple Pay は、iOS App 内、および watchOS 3 の Apple Watch App 内での支払いにも使用できます。ユーザが Apple Pay を利用して App 内で支払うと、Apple は、暗号化された取引情報を受信し、それをデベロッパ固有の鍵を使って再暗号化してからデベロッパまたは加盟店に送信します。Apple Pay には、おおよその購入金額などが匿名の取引情報として保持されます。この情報によってユーザを特定することはできず、ユーザの購買内容がこの情報に含まれることは決してありません。

App が Apple Pay の支払い取引を開始すると、デバイスからの暗号化された取引を、加盟店が受信するよりも前に Apple Pay Server が受信します。Apple Pay Server は次に、加盟店固有の鍵を使って取引を再暗号化した後、加盟店に伝達します。

App で支払いを要求する場合、App は API を呼び出して、デバイスが Apple Pay に対応しているかどうか、および加盟店が受け入れる決済ネットワーク上での支払いが可能なクレジットカード／デビットカードをユーザが持っているかどうかを調べます。App は、請求先住所、届け先住所、連絡先情報など、取引の処理および完了に必要となるすべての情報を要求します。次に App は、Apple Pay シートを表示するよう iOS に依頼します。Apple Pay シートは、App の情報と、使用するカードなど必要なその他の情報を要求します。

この時点で App には、最終的な送料を計算するための住所と郵便番号情報が通知されます。要求したすべての情報が App に提供されるのは、ユーザが Touch ID またはデバイスパスコードで支払いを承認した後です。支払いが承認されると、Apple Pay シートで提供された情報が加盟店に転送されます。

ユーザが支払いを承認すると、暗号ノンスを取得するための呼び出しが Apple Pay Server に対して行われます。暗号ノンスは、ストア内の取引に使用される NFC 端末から返される値に似ています。ノンスは、Apple の鍵を使って暗号化される支払い資格情報を生成するために、他の取引データと共に Secure Element に渡されます。暗号化された支払い資格情報は、Secure Element から Apple Pay Server に渡されます。そこでは、資格情報の復号、資格情報内のノンスと Secure Element から送信されたノンスとの照合、および加盟店 ID と関連付けられた加盟店鍵による支払い資格情報の再暗号化が行われます。その後、支払い資格情報はデバイスに返され、さらに API 経由で App に戻されます。次に App がその情報を加盟店のシステムに送信します。加盟店は、支払い資格情報を自分の秘密鍵で復号して処理できます。この仕組みと Apple のサーバからの署名との組み合わせにより、加盟店は、取引がこの特定の加盟店に向けられたものであることを確認できます。

API には、サポートされる加盟店 ID を指定するエンタイトルメントが必要です。取引を別の顧客が使用できないように、App で注文番号、顧客識別子などのデータを追加し、Secure Element に送信して署名を付加してもらうこともできます。この処理を行うのは App デベロッパの責任です。App デベロッパは、PKPaymentRequest に applicationData を指定できます。このデータのハッシュが、暗号化された支払いデータに含められます。その後、自分の applicationData ハッシュが、支払いデータに含まれている情報と一致することを確認するのは、加盟店の責任です。

Apple Pay による Web での支払い

Apple Pay は、Web サイトでの支払いに利用できます。iOS 10 では、iPhone および iPad から Web 上で Apple Pay 取引ができます。macOS Sierra では、Apple Pay 取引を Mac で開始し、同じ iCloud アカウントを使用して Apple Pay 対応 iPhone または Apple Watch で完了することもできます。

Web 上で Apple Pay に参加する Web サイトはすべて、Apple に登録する必要があります。Apple サーバがドメイン名検証を実行し、TLS クライアント証明書を発行します。Apple Pay をサポートする Web サイトでは、HTTPS 経由でコンテンツを提供する必要があります。支払い取引のたびに、Web サイトは Apple が発行した TLS クライアント証明書を使用して、Apple サーバとの安全な一意の加盟店セッションを取得する必要があります。加盟店セッションデータは Apple によって署名されます。加盟店セッションの署名が検証されると、Web サイトはユーザが Apple Pay 対応デバイスを持っているかどうか、またユーザがそのデバイスでクレジットカード／デビットカード／プリペイドカードを有効にしているかどうかを照会できます。そのほかの詳細情報は共有されません。ユーザがこの情報を共有したくない場合は、macOS と iOS 両方の「Safari」プライバシー設定で Apple Pay 照会を無効にできます。

加盟店セッションが検証されると、すべてのセキュリティおよびプライバシー対策は App 内での支払いの場合と同じになります。

Mac から iPhone または Apple Watch に引き継ぐ場合、Apple Pay はエンドツーエンドで暗号化された IDS プロトコルを使用して支払い関連情報をユーザの Mac から認証側デバイスに転送します。IDS はユーザのデバイス鍵を使用して暗号化を実行するため、ほかのデバイスはこの情報を復号できず、Apple は鍵を利用できません。Apple Pay を引き継ぐためのデバイス検出には、いくつかのメタデータと一緒にユーザのクレジットカードの種類と一意識別子が含まれます。ユーザのカードのデバイス固有アカウント番号は共有されず、ユーザの iPhone または Apple Watch に安全に保存されたままです。Apple は iCloud キーチェーンを通じて、ユーザが直近に使用した連絡先、届け先住所、請求先住所を安全に転送します。

ユーザが iPhone で Touch ID またはパスコードを使用するか Apple Watch のサイドボタンをダブルクリックすることで支払いを承認すると、各 Web サイトの加盟店証明書に一意に暗号化された支払いトークンがユーザの iPhone または Apple Watch から Mac に安全に転送されてから、加盟店の Web サイトに届きます。

互いの近くにあるデバイスのみが支払いを要求および完了できます。近接性は Bluetooth Low Energy アドバタイズメントを通じて判定されます。

ポイントカード

iOS 9 では、対応する NFC 端末に加盟店のポイントカード情報を送信するための付加価値サービス（VAS）プロトコルが Apple Pay でサポートされます。VAS プロトコルは加盟店の端末に実装でき、サポートされる Apple デバイスとの通信に NFC を使用します。VAS プロトコルは短距離で機能し、Apple Pay の取引の一環として、ポイントカード情報の送信などの補助サービスを提供します。

NFC 端末は、カードの要求を送信することで、カード情報の受信を開始します。ユーザが店舗の識別子を含むカードを持っている場合、ユーザはカード使用の承認を求められます。加盟店が暗号化をサポートしている場合、カード情報、タイムスタンプ、および 1 回限りのランダムな ECDH P-256 鍵が加盟店の公開鍵と一緒に使用されてカードデータの暗号化鍵が導出され、これが端末に送信されます。加盟店が暗号化をサポートしていない場合は、ポイントカード情報が送信される前に、ユーザが端末へのデバイスの再提示を求められます。

カードの差し止め、削除、消去

ユーザは、「iPhone を探す」でデバイスを紛失モードにすることにより、iPhone、iPad、および watchOS 3 以降を実行する Apple Watch で Apple Pay を差し止めることができます。ユーザには、「iPhone を探す」や iCloud.com を使用して、または「Wallet」を使ってデバイス上で直接、Apple Pay のカードを削除したり消去したりする選択肢もあります。Apple Watch では、iCloud の設定、iPhone の「Apple Watch」App、または Apple Watch で直接、カードを削除できます。デバイスがオフラインで、モバイルデータ通信ネットワークまたは Wi-Fi ネットワークに接続していない場合でも、デバイス上でカードを使って支払いができる機能は、カード会社または関連の決済ネットワークによって差し止められるか Apple Pay から削除されます。ユーザは、カード会社に電話をかけて、カードを差し止めたり Apple Pay から削除したりすることもできます。

加えて、ユーザが「すべてのコンテンツと設定を消去」または「iPhone を探す」を使用して、あるいは、リカバリモードでデバイスを復元して、デバイス全体を消去すると、iOS が、すべてのカードに削除済みのマークを付けるように Secure Element に指示します。これには、カードをただちに使用不能状態に変更して、Secure Element から Apple Pay Server に接続してカードを完全に消去できるまで安全を確保する効果があります。それとは別に、Secure Enclave は、以前に登録されたカードでそれ以上の支払い承認ができなくなるように AR に無効のマークを付けます。デバイスがオンラインのときは、Secure Element 内のすべてのカードが消去されていることを確実にするため、デバイスから Apple Pay Server への接続が試行されます。

インターネットサービス

強力な Apple ID パスワードを作成する
Apple ID は、iCloud、FaceTime、iMessage などの多くのサービスに接続するために使用されます。ユーザが強力なパスワードを作成できるように、すべての新規アカウントで以下のパスワード属性が必須になっています：

- 8 文字以上
- 小文字を含む
- 大文字を含む
- 数字を含む
- 同一文字を 3 文字以上連続して使用しない
- アカウント名と同一の文字列は使用できない

ユーザがデバイスから実用性と生産性をさらに引き出すことができるよう、Apple は iMessage、FaceTime、Siri、Spotlight の検索候補、iCloud、iCloud バックアップ、iCloud キーチェーンなどの堅牢なサービスを構築して支援しています。

これらのインターネットサービスが実現するセキュリティ上の設計目標は、iOS のプラットフォーム全体で推進するものと共通です。その目標には、デバイス内であってもワイヤレスネットワーク経由の転送時であっても安全が確保されたデータ処理、ユーザの個人情報の保護、情報とサービスへの悪意のあるアクセスや権限のないアクセスなどの脅威からの保護が含まれます。iOS の全体的な使いやすさを損なうことなく、各サービスで独自の強力なセキュリティアーキテクチャが採用されています。

Apple ID

Apple ID は、iCloud、iMessage、FaceTime、iTunes Store、iBooks Store、App Store などの Apple のサービスへのサインインに使うアカウントです。アカウントへの不正アクセスを防止するため、ユーザがそれぞれの Apple ID を安全に保持することが重要です。Apple はこれを支援するため、強力なパスワードを必須にしています。パスワードは、8 文字以上で英字と数字の両方を含んでいる必要があります。また、同一文字を 3 文字以上連続して使用したり、よく使用されるパスワードを設定したりすることはできません。多くの文字や英字句読点（ピリオドなど）を追加してパスワードをより強力にすることで、上記のガイドライン以上に安全にすることをお勧めします。また、Apple はユーザに対し、3 つのセキュリティ質問を設定することを必須にしています。これらの質問は、アカウント情報を変更したり忘れてしまったパスワードをリセットしたりするときに、所有者の識別情報の確認に使用できます。

Apple は、パスワードまたは請求先情報が変更されたときや、Apple ID が新しいデバイスでのサインインに使用されたときなど、アカウントに重要な変更が加えられた場合にメールやプッシュ通知の送信も行います。身に覚えのない変更が行われた場合、ユーザはただちに Apple ID のパスワードを変更するように指示されます。

また、Apple は、ユーザアカウントを保護するために設計されたさまざまなポリシーや手順を採用しています。これには、サインインの再試行回数やパスワードリセットの試行回数の制限、発生した攻撃の特定に役立つ不正行為の積極的な監視、お客様のセキュリティに影響する可能性がある新しい情報に対応するための定期的なポリシーの見直しなどがあります。

2 ファクタ認証

ユーザが自分のアカウントをさらに安全に保護できるようにするため、Apple は 2 ファクタ認証を提供しています。2 ファクタ認証によって、Apple ID のセキュリティがさらに 1 段階強化されます。これにより、ほかの人にパスワードを知られてしまった場合でも、アカウントの所有者だけが自分のアカウントにアクセスできるようになります。

2 ファクタ認証を使えば、自分で信頼した iPhone、iPad、または Mac などのデバイスでのみアカウントにアクセスできるようになります。新しいデバイスにはじめてサインインする場合は、Apple ID のパスワードと、信頼済みのデバイスに自動的に表示されるか、または信頼済みの電話番号に自動的に送信される 6 桁の確認コードという 2 つの情報の入力が必要になります。このコードを入力することが、新しいデバイスを本人が信頼し、安全にサインインできることの確認となります。パスワードだけではユーザのアカウントにアクセスできなくなるため、2 ファクタ認証のおかげで、ユーザの Apple ID のセキュリティと、Apple に保管されるすべての個人情報のセキュリティが向上します。

2 ファクタ認証は、ユーザの Apple ID のセキュリティと、ユーザが Apple で保管した個人情報のセキュリティを強化します。iOS、macOS、tvOS、watchOS、および Apple の Web サイトで使用されている認証システムには、2 ファクタ認証が最初から統合されています。

2 ファクタ認証について詳しくは、support.apple.com/ja-jp/HT204915 を参照してください。

2 ステップ確認

2013 年より、Apple では 2 ステップ確認と呼ばれる同じようなセキュリティ機能を提供しています。2 ステップ確認を有効にした場合は、新しいデバイスから Apple ID アカウント情報の変更を許可したり、iCloud、iMessage、FaceTime、Game Center にサインインしたり、iTunes Store、iBooks Store、Apple Store で買い物をしたりする前に、ユーザの信頼できるいずれかのデバイスに送信される一時的なコードで識別情報を確認する必要があります。また、ユーザには 14 文字の復旧キーが発行されます。この復旧キーは、安全な場所に保管しておき、パスワードを忘れたか、信頼できるデバイスを使用できなくなった場合に使用します。Apple ID の 2 ステップ確認について詳しくは、support.apple.com/ja-jp/HT204152 を参照してください。

管理対象 Apple ID

iOS 9.3 以降で導入された管理対象 Apple ID は、Apple ID と同じように機能しますが、教育機関によって所有および管理されます。教育機関は、パスワードのリセット、購入の制限、「FaceTime」や「メッセージ」などの通信の制限、および職員、教師、生徒のための役割ベースのアクセス権の設定などを実行できます。

管理対象 Apple ID では、Touch ID、Apple Pay、iCloud キーチェーン、HomeKit、「iPhone を探す」など、一部の Apple サービスが利用できません。

管理対象 Apple ID について詳しくは、help.apple.com/schoolmanager を参照してください。

管理対象 Apple ID の監査

管理対象 Apple ID は、教育機関が法的規制やプライバシー規制を順守できるようにする監査機能もサポートしています。IT 管理者、教師、マネージャのアカウントなど、特定の管理対象 Apple ID に監査権限を付与できます。監査担当者が監視できるアカウントは、学校の組織構成で自分より下の階層にあるアカウントのみです。つまり、教師は生徒を監視できます。また、マネージャは教師と生徒を、管理者はマネージャと教師と生徒を監査できます。

Apple School Manager を使用して資格情報の監査を要求すると、監査を要求した管理対象 Apple ID のみにアクセスできる特別なアカウントが発行されます。監査用アカウントは 7 日後に無効になります。監査期間中、監査担当者は、iCloud または CloudKit 対応アプリケーションに保存されているユーザのコンテンツを表示および変更できます。監査用のアクセス要求はすべて Apple School Manager のログに記録されます。このログには、監査担当者、その担当者がアクセスを要求した管理対象 Apple ID、要求日時、監査の実行の有無が表示されます。

管理対象 Apple ID と個人用デバイス

管理対象 Apple ID を、個人所有の iOS デバイスで使用することもできます。生徒が iCloud にサインインするには、教育機関が発行した管理対象 Apple ID と、Apple ID の 2 ファクタ認証プロセスの第 2 要素として機能する追加の自作パスワードを使用します。管理対象 Apple ID を個人所有デバイスで使用する場合は、iCloud キーチェーンを使用できません。また、「FaceTime」や「メッセージ」など、ほかの機能が教育機関によって制限されることがあります。生徒がサインイン中に作成した iCloud の書類はすべて、前述の監査の対象になります。

iMessage

Apple の iMessage は、iOS デバイスと Mac コンピュータのメッセージサービスです。iMessage では、テキストに加え、写真、連絡先、位置情報などを添付することも可能です。メッセージは、ユーザが登録したすべてのデバイスに表示されるので、どのデバイスからも会話を続けることができます。iMessage では Apple Push Notification service（APNs）が多く使用されます。メッセージや添付ファイルは Apple に記録されず、そのコンテンツはエンドツーエンドの暗号化で保護されるため、送信者と受信者以外はだれもそれらにアクセスできません。Apple はデータを復号できません。

ユーザがデバイスで iMessage をオンにすると、そのサービスで使用される 2 つの鍵ペアが生成されます。暗号化用の鍵（RSA 1280 ビット）と、署名用の鍵（NIST P-256 曲線の ECDSA 256 ビット）です。両方の鍵ペアの秘密鍵がデバイスのキーチェーンに保存され、公開鍵が Apple のディレクトリサービス（IDS）に送信されます。公開鍵は、IDS でユーザの電話番号またはメールアドレス、およびデバイスの APNs アドレスに関連付けられます。

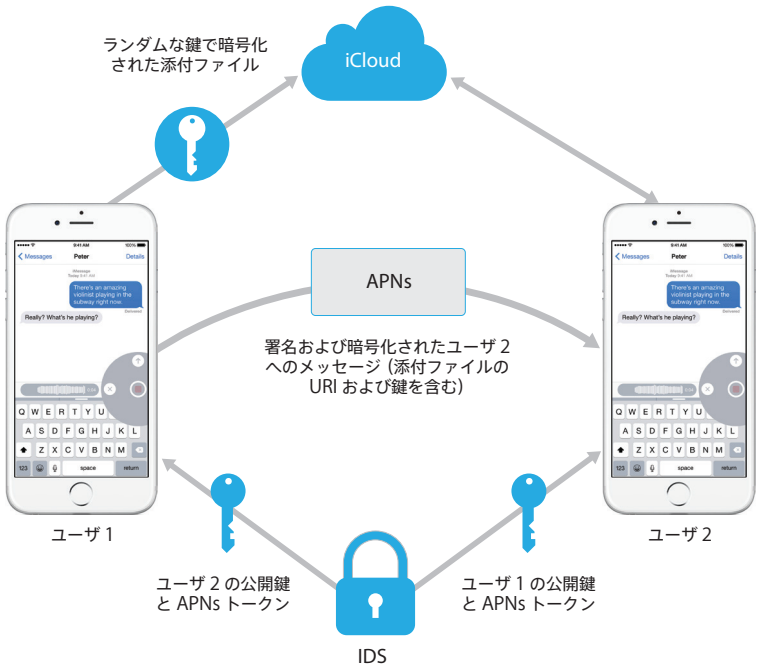
ユーザが iMessage で使用する追加のデバイスを有効にすると、デバイスの暗号化および署名用公開鍵、APNs アドレス、および関連付けられた電話番号がディレクトリサービスに追加されます。ユーザはメールアドレスを追加することもできます。追加したアドレスは Confirmation Link の送信によって確認されます。電話番号は、通信事業者のネットワークおよび SIM によって確認されます。さらに、新しいデバイス、電話番号、またはメールアドレスが追加されると、ユーザが登録したすべてのデバイスに通知メッセージが表示されます。

iMessage のメッセージの送受信方法

iMessage での会話を開始するには、相手のアドレスまたは名前を入力します。ユーザが電話番号またはメールアドレスを入力すると、デバイスは IDS と通信し、受信者に関連付けられたすべてのデバイスの公開鍵と APNs アドレスを取得します。ユーザが名前を入力すると、デバイスはまずユーザの「連絡先」を使用してその名前に関連付けられた電話番号およびメールアドレスを収集した後、IDS から公開鍵と APNs アドレスを取得します。

ユーザの送信メッセージは、受信者のデバイスごとに個別に暗号化されます。受信デバイスの公開 RSA 暗号化鍵は、IDS から取得されます。送信デバイスは受信デバイスごとにランダムな 88 ビット値を生成し、この値を HMAC-SHA256 鍵として使い、送信者と受信者の公開鍵とプレーンテキストから導出される 40 ビット値を構成します。88 ビット値と 40 ビット値を連結させると 128 ビット鍵となり、これが AES を使って CTR モードでメッセージを暗号化します。40 ビット値は復号されたプレーンテキストの完全性を検証するために受信側で使用されます。このメッセージごとの AES 鍵は、RSA-OAEP を使用して受信デバイスの公開鍵に対して暗号化されます。次に、暗号化されたメッセージテキストと暗号化されたメッセージ鍵の組み合わせが SHA-1 を使ってハッシュ化され、送信デバイスの署名用秘密鍵を用いてハッシュに ECDSA の署名が付加されます。その結果、メッセージは、暗号化されたメッセージテキスト、暗号化されたメッセージ鍵、および送信者のデジタル署名からなる、受信デバイスごとに異なるメッセージになります。次にメッセージは APNs に送られ、配信されます。タイムスタンプや APNs の経路情報などのメタデータは暗号化されません。APNs との通信は、前方秘匿 TLS チャンネルを使用して暗号化されます。

APNs がリレーできるメッセージのサイズは、iOS のバージョンにより最大 4 KB または 16 KB です。メッセージのテキストが長すぎる場合、または写真などの添付ファイルが含まれる場合は、添付ファイルが、ランダムに生成された 256 ビット鍵で AES の CTR モードを用いて暗号化され、iCloud にアップロードされます。次に、添付ファイルの AES 鍵、URI（Uniform Resource Identifier）、および暗号化結果の SHA-1 ハッシュが、iMessage の内容として受信者に送信されます。それらの機密性と完全性は、次の図に示す標準の iMessage 暗号化機能によって保護されます。



グループ会話の場合は、各受信者のデバイスごとにこのプロセスが繰り返されます。

受信側では、各デバイスが APNs からメッセージのコピーを受信し、必要に応じて iCloud から添付ファイルを取得します。可能な場合は名前を表示できるように、送信者の発信電話番号またはメールアドレスが受信者の連絡先と照合されます。

すべてのプッシュ通知と同様、メッセージは配信された時点で APNs から削除されます。ただし、ほかの APNs 通知と異なり、iMessage のメッセージはオフラインデバイスへの配信のためにキューに入れられます。メッセージは現在、最長 30 日間保存されます。

FaceTime

FaceTime は、Apple のビデオおよびオーディオ通話サービスです。FaceTime 通話では iMessage と同様、ユーザが登録したデバイスへの最初の接続を確立するために Apple Push Notification service（APNs）を使用します。FaceTime 通話のオーディオ／ビデオコンテンツはエンドツーエンドの暗号化によって保護されるため、送信者と受信者以外はだれもアクセスできません。Apple はデータを復号できません。

FaceTime では STUN（Session Traversal Utilities for NAT）および ICE（Internet Connectivity Establishment）を使用してデバイス間のピアツーピア接続を確立します。デバイスは APNs および STUN メッセージを使用して識別情報の証明書を確認し、各セッションの共有シークレットを確立します。各デバイスから送信された暗号ノンスが、各メディアチャンネルの Salt キーに結合されます。各メディアチャンネルでは、データは AES-256 で暗号化され、Secure Real Time Protocol（SRTP）経由でストリーミングされます。

iCloud

iCloud にユーザの連絡先、カレンダー、写真、書類などを保存すると、ユーザのすべてのデバイス間で情報を自動的に最新の状態に保つことができます。iCloud は他社製 App でも、書類や、デベロッパによって定義された App データのキー値を保存および同期するために使用できます。ユーザは、Apple ID でサインインし、使用したいサービスを選択して iCloud を設定します。「マイフォトストリーム」、iCloud Drive、バックアップなどの iCloud の機能は、IT 管理者が構成プロファイルによって無効にすることができます。このサービスでは保存されるデータの種類は認識されず、すべてのファイルコンテンツがバイトの集合として同様に扱われます。

iCloud によって各ファイルがチャンクに分割され、AES-128 と、各チャンクのコンテンツから導出される、SHA-256 を使用する鍵を使って暗号化されます。それらの鍵とファイルのメタデータは Apple によってユーザの iCloud アカウントに保存されます。暗号化されたファイルのチャンクは、Amazon S3 や Windows Azure のような他社のストレージサービスを利用して、ユーザを特定する情報を含めずに保存されます。

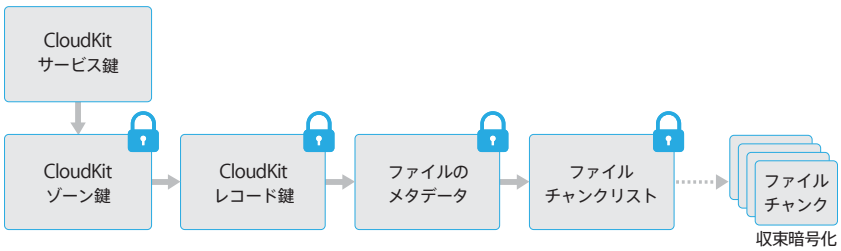
iCloud Drive

iCloud Drive では、iCloud に保存されている書類を保護するためにアカウントに基づく鍵が追加されます。既存の iCloud サービスと同様、ファイルコンテンツがチャンクに分割されて暗号化され、他社のサービスを利用して保存されます。ただし、ファイルコンテンツの鍵は、iCloud Drive メタデータと一緒に保存されるレコードキーでラップされます。これらのレコードキーも、ユーザの iCloud Drive サービスの鍵によって保護されます。iCloud Drive サービスの鍵はユーザの iCloud アカウントと一緒に保存されます。ユーザは iCloud への認証によって iCloud 書類のメタデータにアクセスできますが、iCloud Drive ストレージの保護されている部分を表示するには、iCloud Drive サービスの鍵を所有している必要があります。

CloudKit

App デベロッパは CloudKit を利用して、iCloud にキー値データ、構造化データ、および各種アセットを保存できます。CloudKit へのアクセスは App のエンタイトルメントを使用して制御されます。CloudKit は、パブリックデータベースとプライベートデータベースの両方に対応しています。パブリックデータベースは、App のすべてのコピーで使用され、通常は一般的なアセット用で、暗号化されません。プライベートデータベースには、ユーザのデータが格納されます。

CloudKit は iCloud Drive と同様に、アカウントに基づく鍵を使用して、ユーザのプライベートデータベースに格納されている情報を保護します。また、ほかの iCloud サービスと同様に、ファイルは他社のサービスを利用してチャンクに分割されて暗号化され、保存されます。CloudKit では、データ保護と同様の鍵階層を利用します。Per File キーは CloudKit レコード鍵でラップされます。これらのレコード鍵はさらにゾーン全体鍵で保護され、ゾーン全体鍵はユーザの CloudKit サービス鍵で保護されます。CloudKit サービス鍵はユーザの iCloud アカウントに保存され、ユーザが iCloud で認証を完了してはじめて利用可能になります。



iCloud バックアップ

iCloud では、デバイス設定、App データ、「カメラロール」の写真やビデオ、「メッセージ」App での会話などの情報を Wi-Fi 経由で毎日バックアップすることもできます。iCloud のコンテンツは、インターネット経由で送信される際に暗号化され、暗号化フォーマットで保存され、認証にセキュアトークンを使うことで、確実に保護されます。iCloud バックアップは、デバイスがロックされて電源に接続され、かつ Wi-Fi 経由でインターネットに接続できる場合にのみ実行されます。iOS では暗号化が使用されるため、システムはデータを安全に保護しながら、差分での自動的なバックアップと復元を実行できるように設計されています。

iCloud では以下の項目のバックアップが作成されます：

- 購入した音楽、映画、テレビ番組、App、およびブックについてのレコード。ユーザの iCloud バックアップにはユーザの iOS デバイスに表示される購入したコンテンツについての情報が含まれますが、購入したコンテンツ自体は含まれません。ユーザが iCloud バックアップから復元すると、購入したコンテンツが iTunes Store、App Store、または iBooks Store から自動的にダウンロードされます。一部の種類のコンテンツが自動的にダウンロードされない国もあります。また、コンテンツが払い戻されていたり、ストアで扱われなくなった場合、以前に購入したコンテンツを利用できなくなることがあります。全購入履歴はユーザの Apple ID に関連付けられています。
- ユーザの iOS デバイス上の写真とビデオ。ユーザが iOS デバイス（iOS 8.1 以降）または Mac（OS X v10.10.3 以降）の iCloud フォトライブラリをオンにしている場合、写真とビデオはすでに iCloud に保存されているため、ユーザの iCloud バックアップには含まれません。
- 連絡先、カレンダーイベント、リマインダー、メモ
- デバイス設定
- App データ
- 通話履歴
- ホーム画面および App の配置
- iMessage、テキスト（SMS）、および MMS メッセージ（バックアップ時に使用した SIM カードが必要）
- 着信音
- Visual Voicemail のパスワード（バックアップ時に使用した SIM カードが必要）
- HomeKit の構成
- HealthKit データ

デバイスのロック中にアクセスできないデータ保護クラスでファイルが作成されると、Per File キーは iCloud バックアップキーバッグにあるクラス鍵を使用して暗号化されます。ファイルは元の暗号化された状態で iCloud にバックアップされます。データ保護クラス No Protection のファイルは、転送中に暗号化されます。

iCloud バックアップキーバッグには、各データ保護クラス用の非対称（Curve25519）鍵が含まれます。これらは Per File キーを暗号化するために使用されます。バックアップキーバッグおよび iCloud バックアップキーバッグの内容について詳しくは、「暗号化とデータ保護」セクションの「キーチェーンデータ保護」を参照してください。

バックアップセットはユーザの iCloud アカウントに保存され、ユーザのファイルのコピーと、iCloud バックアップキーバッグで構成されます。iCloud バックアップキーバッグはランダムな鍵によって保護されます。この鍵もバックアップセットと一緒に保存されます。（ユーザの iCloud パスワードは暗号化に使用されないため、iCloud パスワードを変更しても既存のバックアップは無効になりません。）

ユーザのキーチェーンデータベースは iCloud にバックアップされますが、UID とタングルされた鍵によって常に保護されます。そのため、キーチェーンはバックアップの作成元と同じデバイスにのみ復元できます。つまり、Apple を含めてほかのだれも、ユーザのキーチェーン項目を読み出すことはできません。

復元時には、バックアップされたファイル、iCloud バックアップキーバッグ、およびキーバッグの鍵がユーザの iCloud アカウントから取得されます。iCloud バックアップキーバッグがその鍵を使って復号された後、キーバッグにある Per File キーを使ってバックアップセット内のファイルが復号されます。それらのファイルは新しいファイルとしてファイルシステムに書き込まれるため、それぞれのデータ保護クラスに従って再暗号化されます。

iCloud キーチェーン

Safari の iCloud キーチェーンとの統合
「Safari」では、Web サイトのパスワード用に、暗号論的に強力なランダムな文字列を自動的に生成できます。この文字列はキーチェーンに保存され、ほかのデバイスと同期されます。キーチェーン項目は Apple のサーバを経由してデバイス間で転送されますが、Apple もほかのデバイスも内容を読み出せないように暗号化されます。

iCloud キーチェーンを使うと、Apple に情報を開示することなく、iOS デバイスや Mac コンピュータの間でパスワードを安全に同期することができます。iCloud キーチェーンの設計とアーキテクチャに大きく影響を与えた目標としては、強力なプライバシーおよびセキュリティに加え、使いやすさと、キーチェーンの復元を可能にすることがありました。iCloud キーチェーンは、キーチェーン同期およびキーチェーン復元という 2 つのサービスで構成されます。

Apple は、ユーザのパスワードが以下の状況下でも保護されるように iCloud キーチェーンとキーチェーン復元を設計しました：

- ユーザの iCloud アカウントが危殆化した。
- 外部の攻撃者または従業員によって iCloud が危殆化した。
- ユーザアカウントに第三者がアクセスした。

キーチェーン同期

ユーザが iCloud キーチェーンをはじめて有効にすると、デバイスがトラストサークルを確立し、そのデバイス自体の同期用識別情報を作成します。同期用識別情報は秘密鍵と公開鍵で構成されます。同期用識別情報の公開鍵はサークルの中に置かれ、サークルは 2 回署名されます。まず同期用識別情報の秘密鍵で署名され、次にユーザの iCloud アカウントパスワードから導出される楕円曲線暗号の非対称鍵（P256 を使用）で署名されます。サークルと共に、ユーザの iCloud パスワードに基づく鍵の作成に使用されるパラメータ（ランダム Salt および反復回数）も保存されます。

署名された同期サークルはユーザの iCloud のキー値ストレージ領域に配置されます。これはユーザの iCloud パスワードを知っていないと読み出すことができず、サークルメンバーの同期用識別情報の秘密鍵がないと正規に変更を加えられません。

ユーザが別のデバイス上で iCloud キーチェーンをオンにすると、そのデバイスがメンバーになっていない同期サークルをユーザがすでに確立していることを、新しいデバイスが iCloud で認識します。新しいデバイスはその同期用識別情報の鍵ペアを作成してから、サークルのメンバーシップを要求する申請チケットを作成します。このチケットはデバイスの同期用識別情報の公開鍵で構成され、ユーザは iCloud パスワードでの認証を求められます。楕円曲線暗号鍵の生成パラメータは iCloud から取得され、これによって申請チケットへの署名に使用される鍵が生成されます。最後に、申請チケットが iCloud に配置されます。

申請チケットの受信が最初のデバイスに認識されると、新しいデバイスの同期サークルへの参加要求を承認するようユーザに求める通知が最初のデバイスに表示されます。ユーザが iCloud パスワードを入力すると、一致する秘密鍵で署名された申請チケットであることが確認されます。これによって、サークルへの参加要求を行った本人が、要求後にユーザの iCloud パスワードを入力したことが確認されます。

新しいデバイスをサークルに追加することをユーザが承認すると、最初のデバイスが新しいメンバーの公開鍵を同期サークルに追加し、自らの同期用識別情報と、ユーザの iCloud パスワードから導出された鍵の両方を使って再度公開鍵に署名します。新しい同期サークルが iCloud に配置されます。その同期サークルには、サークルの新しいメンバーも同様に署名しています。

これで同期サークルのメンバーが 2 つになり、各メンバーがお互いの公開鍵を持つことになります。メンバー同士で、iCloud のキー値ストレージを経由して個別のキーチェーン項目のやりとりが開始されます。両方のサークルメンバーに同じ項目がある場合、変更日が最新の項目が同期されます。他方のメンバーに同じ項目があり、変更日も同一の場合はスキップされます。同期される各項目は、送信先のデバイス専用に暗号化されます。ほかのデバイスにも Apple にも復号できません。また、暗号化された項目は一時的に iCloud に存在するだけで、同期される新しい項目で上書きされます。

新しいデバイスが同期サークルに追加されると、このプロセスが繰り返されます。たとえば、デバイスがもう 1 つ参加した場合、ユーザの残りのデバイスの両方に確認メッセージが表示されます。ユーザはそのどちらかのデバイスで新しいメンバーを承認できます。新しいピアが追加されると、各ピアが新しいピアと同期されて、すべてのメンバーのキーチェーン項目が同じになります。

ただし、キーチェーン全体は同期されません。VPN ID などの一部の項目はデバイス固有のものであり、そのデバイス以外には送信されません。属性が `kSecAttrSynchronizable` の項目のみが同期されます。Apple は、Safari ユーザデータ（ユーザ名、パスワード、およびクレジットカード番号を含む）と、Wi-Fi パスワードおよび HomeKit の暗号化鍵にこの属性を設定しています。

また、デフォルトでは、他社製 App によって追加されたキーチェーン項目は同期されません。デベロッパは、キーチェーンに項目を追加する際に `kSecAttrSynchronizable` を設定する必要があります。

キーチェーン復元

キーチェーン復元では、ユーザは Apple がパスワードおよびその他のデータを読み取れるようにすることなく、必要に応じてキーチェーンを Apple に預託することができます。ユーザは、デバイスを 1 つしか持っていない場合でも、キーチェーン復元によってデータの損失を防止できます。これは、Safari を使って Web のアカウント用にランダムで強力なパスワードを生成する場合に特に重要です。これらのパスワードの記録はキーチェーンにしか残らないためです。

キーチェーン復元は、この機能をサポートするために Apple が開発した二次認証と安全なエスクローサービスによって実現されます。ユーザのキーチェーンは強力なパスコードを使って暗号化され、条件が厳密に満たされた場合にのみ、エスクローサービスからキーチェーンのコピーが提供されます。

iCloud キーチェーンをオンにすると、ユーザは iCloud セキュリティコードの作成を求められます。このコードは預託したキーチェーンを復元するために必要です。デフォルトでは、ユーザはセキュリティコード用に単純な 4 桁の値を指定するよう求められます。ただし、ユーザが独自の長いコードを指定したり、暗号論的にランダムなコードをデバイスに作成させて、それを自分で記録して保管したりすることもできます。

次に、iOS デバイスがユーザのキーチェーンのコピーを書き出し、非対称キーバッグにある鍵でラップして暗号化し、iCloud のキー値ストレージ領域に保存します。キーバッグはユーザの iCloud セキュリティコードと、エスクローレコードが保存されるハードウェアセキュリティモジュール（HSM）クラスタの公開鍵でラップされます。これがユーザの iCloud エスクローレコードになります。

ユーザが、独自のセキュリティコードを指定したり、4 桁の値を使用したりするのではなく、暗号論的にランダムなセキュリティコードを受け入れることを決定した場合は、エスクローレコードは不要です。その代わりに、iCloud セキュリティコードを使用してランダムな鍵が直接ラップされます。

ユーザはセキュリティコードを確定するだけでなく、電話番号を登録する必要があります。これは、キーチェーン復元で二次レベルの認証を行うために使用されます。ユーザは SMS を受信します。復元を進めるにはそれに返信する必要があります。

エスクローのセキュリティ

iCloud には、認証されたユーザおよびデバイスのみが復元を実行できるようにするためのキーチェーンエスクローの安全なインフラストラクチャが用意されています。iCloud を背後で支えているのが、エスクローレコードを保護する HSM クラスタです。クラスタごとに鍵があり、前述したように、その鍵を使ってクラスタの監視下でエスクローレコードを暗号化します。

キーチェーンを復元するには、ユーザが iCloud アカウントとパスワードで認証し、登録済みの電話番号に送信される SMS に返信する必要があります。その後、ユーザは iCloud セキュリティコードを入力する必要があります。HSM クラスタは Secure Remote Password（SRP）プロトコルを使用して、ユーザが iCloud セキュリティコードを知っていることを確認します。コード自体は Apple に送信されません。クラスタの各メンバーは、ユーザがレコードを取得する際に許容される最大試行回数（後述）を超えていないことをそれぞれで確認します。超えていないという判断で過半数が一致した場合は、エスクローレコードがアンラップされ、レコードがユーザのデバイスに送信されます。

次に、デバイスが iCloud セキュリティコードを使用して、ユーザのキーチェーンの暗号化に使用したランダムな鍵をアンラップします。その鍵を使って、iCloud のキー値ストレージから取得されたキーチェーンが復号され、デバイス上に復元されます。認証およびエスクローレコード取得の試行は、最大 10 回のみ許容されます。試行に数回失敗するとレコードがロックされるため、それ以上試行するには、ユーザは Apple サポートに電話して承認を得る必要があります。10 回失敗すると、HSM クラスタによってエスクローレコードが破棄され、キーチェーンが完全に失われます。これは、キーチェーンデータを犠牲にする代わりに、レコードの取得を試みる総当たり（ブルートフォース）攻撃からレコードを守る手段になります。

これらのポリシーは HSM ファームウェアに組み込まれています。ファームウェアの変更を許可する管理アクセスカードは破棄されています。ファームウェアの改ざんまたは秘密鍵へのアクセスが試行されると、HSM クラスタによって秘密鍵が削除されます。万一この状況が発生した場合は、そのクラスタによって保護されているすべてのキーチェーンの所有者に、エスクローレコードが失われたことを通知するメッセージが送信されます。それらのユーザは、その後再登録することを選択できます。

Siri

自然に話しかけるだけで、メッセージを送信したり、会議のスケジュールを設定したり、電話をかけたりなど、さまざまなことを Siri に指示できます。Siri は、音声認識、テキスト読み上げ、クライアントサーバモデルを使ってさまざまなリクエストに答えます。Siri がサポートするタスクは、ごく最小限の個人情報のみが完全に保護された状態で利用されるように設計されています。

Siri をオンにすると、音声認識および Siri サーバで使用されるランダムな識別子が作成されます。これらの識別子は、サービスの向上のため Siri の内部でのみ使用されます。その後 Siri をオフにすると、再度 Siri をオンにしたときに使用されるランダムな識別子が新しく生成されます。

Siri の機能を向上させるため、ユーザの情報の一部がデバイスからサーバに送信されます。これには、ミュージックライブラリについての情報（曲のタイトル、アーティスト、プレイリスト）、「リマインダー」のリスト名、「連絡先」で定義されている名前と続柄などが含まれます。サーバとの通信はすべて HTTPS で行われます。

Siri セッションが開始されると、「連絡先」から取得されたユーザの名と姓が、地理上の大まかな位置と共にサーバに送信されます。これは、Siri が応答に名前を含めたり、天気に関する質問など、大まかな位置情報のみが必要な質問に答えたりできるようにするためです。

付近の映画館の場所を調べるときのように、位置を正確に特定する必要がある場合は、より正確な位置情報を送信するようにサーバがデバイスに要求します。これは、デフォルトではユーザのリクエストを処理するために本当に必要な場合にのみ情報がサーバに送信されるということを示す一例です。どのような場合でも、使用しない状態が 10 分間続くと、セッション情報が破棄されます。

Siri を Apple Watch で使用している場合、Apple Watch は、上で述べたように、独自のランダムな一意の識別子を作成します。ただし、ユーザの情報を再度送信するのではなく、ペアリングされた iPhone の Siri 識別子を送信して、その情報への参照を提供します。

ユーザが話した言葉の録音が Apple の音声認識サーバに送信されます。タスクの内容が音声入力のみの場合は、認識されたテキストがデバイスに送信されます。その他の場合は、Siri がテキストを分析し、必要に応じて、デバイスに関連付けられたプロフィールの情報と組み合わせます。たとえば、「お母さんにメッセージを送信して」というリクエストの場合は、「連絡先」から取得された続柄と名前が使用されます。認識されたアクションのコマンドは、実行のためにデバイスに送り返されます。

Siri の機能の多くは、サーバの指示の下でデバイスによって実行されます。たとえば、受信したメッセージを読むことをユーザが Siri に依頼した場合、サーバは未開封のメッセージの内容を読み上げるようにデバイスに指示します。メッセージの内容と送信者はサーバに送信されません。

ユーザの音声の録音は、認識システムが音声認識の精度を高める目的で利用できるように 6 か月間保存されます。6 か月経過した後は、Apple が Siri の改善および開発のために使用できるように、識別子を削除した別のコピーが最長 2 年間保存されます。録音、発音表記、関連データの中で、識別子を含まないごく一部のデータは、Siri の継続的な改善と品質保証のために 2 年を超えて使用されることがあります。また、音楽、スポーツのチームや選手、企業、お店や見どころに関する音声の録音の一部も、Siri を改善する目的で同様に保存されます。

Siri は音声アクティベーションによりハンズフリーで呼び出すこともできます。音声トリガーの検出は、デバイス上でローカルに行われます。このモードでは、入力されたオーディオパターンが、指定されたトリガーフレーズの音響に十分一致した場合にのみ Siri が起動します。トリガーが検出されると、対応するオーディオが後続の Siri コマンドと一緒に、その後の処理のために Apple の音声認識サーバに送信されます。これは Siri で行われるユーザのその他の音声録音と同じ規則に従います。

Continuity

iCloud、Bluetooth、Wi-Fi などのテクノロジーを利用する Continuity（連係機能）により、使用するデバイスを変更してもアクティビティを継続できるようになりました。Continuity は、通話の発信／着信、テキストメッセージの送受信、モバイルデータ通信インターネット接続の共有などに利用できます。

Handoff

Handoff を使用すると、ユーザの Mac と iOS デバイスがお互いの近くにあるとき、作業中のあらゆる項目を一方のデバイスから他方のデバイスに自動的に渡すことができます。Handoff でデバイスを切り替えて、すぐに作業を再開できます。

Handoff に対応する別のデバイスでユーザが iCloud にサインインすると、2 つのデバイスが Apple Push Notification service（APNs）を使用して Bluetooth Low Energy 4.0 の帯域外ペアリングを確立します。個別のメッセージは iMessage と同様の方法で暗号化されます。デバイスがペアリングされると、各デバイスで 256 ビットの AES 対称鍵が生成され、デバイスのキーチェーンに保存されます。この鍵は Bluetooth Low Energy アドバタイズメントの暗号化と認証に使用されます。このアドバタイズメントでは、GCM モードの AES-256 を使用して、iCloud でペアリングされたほかのデバイスにデバイスの現在のアクティビティを伝達します。このとき、リプレイ攻撃に対する防御策が講じられます。デバイスは、新しい鍵でのアドバタイズメントをはじめて受信すると、発信元のデバイスとの Bluetooth Low Energy 接続を確立し、アドバタイズメントの暗号化鍵の交換を実行します。この接続は、Bluetooth Low Energy 4.0 の標準の暗号化を使用すると共に、iMessage と同様に個別のメッセージを暗号化することで保護されます。特定の状況では、これらのメッセージが Bluetooth Low Energy ではなく APNs を介して送信されます。アクティビティのペイロードは、iMessage と同じ方法で保護および転送されます。

ネイティブ App と Web サイトの間での Handoff

Handoff を使用すると、iOS のネイティブ App で、その App のデベロッパが正当に制御しているドメインの Web ページの閲覧を再開できます。また、ネイティブ App でのユーザアクティビティを Web ブラウザで再開することもできます。

デベロッパが制御していない Web サイトの閲覧の再開をネイティブ App が要求することを防止するため、App は再開する Web ドメインを正当に制御していることを示す必要があります。Web サイトのドメインの制御は、共有 Web クレデンシャルで使用されるメカニズムによって確立されます。詳しくは、「暗号化とデータ保護」セクションの「「Safari」に保存されたパスワードへのアクセス」を参照してください。App がユーザアクティビティの Handoff の受け入れを許可されるには、App でのドメイン名の制御がシステムによって検証される必要があります。

Web ページの Handoff の発信元には、Handoff API を採用している任意のブラウザを使用できます。ユーザが Web ページを表示すると、その Web ページのドメイン名が、暗号化された Handoff アドバタイズメントバイトでアドバタイズされます。このアドバタイズメントバイトは、「Handoff」セクションで述べたように、同じユーザのほかのデバイスでのみ復号できます。

受信側のデバイスでは、アドバタイズされたドメイン名からの Handoff をインストール済みの App が受け入れたことが検知され、ネイティブ App のアイコンが Handoff のオプションとして表示されます。そのネイティブ App が起動すると、Web ページの完全な URL とタイトルを受け取ります。その他の情報はブラウザからネイティブ App に渡されません。

逆に、Handoff の受信側デバイスに同じネイティブ App がインストールされていないときのために、ネイティブ App はフォールバック URL を指定できます。その場合は、ユーザのデフォルトブラウザが Handoff の App のオプションとして表示されます（そのブラウザが Handoff API を採用している場合）。Handoff が要求されるとブラウザが起動し、発信元の App から提供されたフォールバック URL を開きます。このフォールバック URL には、ネイティブ App のデベロッパが制御しているドメイン名のみに制限されるという要件はありません。

サイズが大きいデータの Handoff

一部の App では、Handoff の基本機能に加え、Apple 製のピアツーピア Wi-Fi テクノロジーによるサイズが大きいデータの送信（AirDrop と同様の方法で行われます）をサポートする API の使用を選択できます。たとえば、「メール」App では、サイズが大きい添付ファイルが含まれる可能性があるメール下書きの Handoff をサポートするために、それらの API が使用されます。

App でこの機能が使用されると、2 つのデバイス間で通常の Handoff とまったく同じように受け渡しが開始されます（前のセクションを参照）。ただし、受信側のデバイスは、Bluetooth Low Energy を使用して最初のペイロードを受信した後で、Wi-Fi で新しい接続を開始します。この接続は暗号化され（TLS）、iCloud 識別情報の証明書が交換されます。証明書内の識別情報がユーザの識別情報と照合されて確認されます。それ以降のペイロードデータは、転送が完了するまで、この暗号化された接続で送信されます。

ユニバーサルクリップボード

ユニバーサルクリップボードを使用すると、Handoff を活用してクリップボードの内容をデバイス間で安全に転送できるため、1 台のデバイスでコピーして別のデバイスでペーストすることができます。クリップボードの内容はほかの Handoff データと同様に保護され、App のデベロッパが共有を禁止していない限り、デフォルトでユニバーサルクリップボードと共有されます。

App はユーザがクリップボードの内容をその App にペーストしたかどうかにかかわらず、クリップボードのデータにアクセスできます。ユニバーサルクリップボードを使用すると、このデータアクセスが、同じユーザのほかのデバイス（iCloud へのサインインによって確定します）で実行されている App にまで拡張されます。

自動ロック解除

自動ロック解除をサポートする Mac コンピュータでは、Bluetooth Low Energy とピアツーピア Wi-Fi を使用して、同じユーザの Apple Watch が Mac のロックを解除することを安全に許可できます。この機能に対応していて、同じ iCloud アカウントに関連付けられている各 Mac および Apple Watch は、2 ファクタ認証（TFA）を使用する必要があります。

Apple Watch による Mac のロック解除を有効にすると、自動ロック解除 ID を使用する安全なリンクが確立されます。Mac はランダムなワンタイムロック解除シークレットを作成し、安全なリンクを介して Apple Watch に送信します。このシークレットは Apple Watch 上に保存され、Apple Wach のロックが解除されているときにのみアクセスできます（「データ保護クラス」を参照）。マスターエントロピーとこの新しいシークレットは、どちらもユーザのパスワードとは異なります。

ロック解除操作時には、Mac が Bluetooth Low Energy を使用して Apple Watch への接続を作成します。その後、2 台のデバイス間に安全なリンクが確立されます。これには、安全なリンクが最初に有効になったときに使用された共有鍵が使用されます。Mac と Apple Watch が、ピアツーピア Wi-Fi と安全なリンクから導出された安全な鍵を使用して、2 台のデバイスの距離を特定します。デバイスが互いの通信圏内にある場合は、事前に共有されたシークレットが安全なリンクを使用して転送され、Mac のロックが解除されます。ロック解除が正常に行われると、Mac が現在のロック解除シークレットを新しいワンタイムロック解除シークレットに置き換え、安全なリンクを介して新しいロック解除シークレットを Apple Watch に送信します。

iPhone での通話リレー

Mac、iPad、または iPod が iPhone と同じ Wi-Fi ネットワークに接続されている場合、iPhone の携帯電話接続を利用して通話を発信／着信できます。構成の要件は、双方のデバイスが iCloud と FaceTime の両方に同じ Apple ID アカウントでサインインしていることです。

電話の着信があると、Apple Push Notification service（APNs）によって、構成済みのすべてのデバイスに通知されます。各通知には、iMessage で使用されるものと同じエンドツーエンドの暗号化が使用されます。同じネットワーク上にあるデバイスに、電話の着信通知の UI が表示されます。電話に出ると、2 つのデバイス間の安全なピアツーピア接続を使用して、オーディオが iPhone からシームレスに転送されます。

あるデバイスで着信に応答すると、Bluetooth Low Energy 4.0 経由で短くアドバタイズすることで、iCloud でペアリングされた近くにあるデバイスの着信音が停止します。アドバタイズバイトは、Handoff アドバタイズと同じ方法で暗号化されます。

電話の発信も Apple Push Notification service（APNs）によって iPhone にリレーされ、オーディオが同様にデバイス間の安全なピアツーピアリンクを介して転送されます。

ユーザは「FaceTime」設定の「iPhone での通話」をオフにして、デバイスでの通話のリレーを無効にすることができます。

iPhone メッセージ転送

メッセージ転送は、iPhone で受信した SMS テキストメッセージを、ユーザの登録した iPad、iPod touch、Mac に自動的に送信します。各デバイスは、同じ Apple ID アカウントを使用して iMessage サービスにサインインしている必要があります。SMS のメッセージ転送を有効にすると、iPhone で生成されたランダムな 6 桁の数値コードの入力によって、各デバイスで登録が検証されます。

デバイスがリンクされると、iPhone はこの文書の「iMessage」セクションで説明されている手法を利用して、着信した SMS テキストメッセージを暗号化し、各デバイスに転送します。返信は同じ方法で iPhone に送り返され、次に iPhone がその返信を通信事業者の SMS 伝送メカニズムを使用してテキストメッセージとして送信します。メッセージ転送は、「メッセージ」設定でオン／オフを切り替えられます。

Instant Hotspot

Instant Hotspot をサポートする iOS デバイスは、Bluetooth Low Energy を使用して、同じ iCloud アカウントにサインインしているデバイスを検出し、通信します。OS X Yosemite 以降を搭載していて互換性のある Mac も、同じテクノロジーを使用して Instant Hotspot 対応の iOS デバイスを検出し、通信します。

ユーザが iOS デバイスで「Wi-Fi」設定を開くと、そのデバイスは同じ iCloud アカウントにサインインしているすべてのデバイスが合意した識別子を含む Bluetooth Low Energy 信号を発信します。この識別子は iCloud アカウントに関連付けられた DSID（Destination Signaling Identifier）から生成され、定期的に入れ替えられます。同じ iCloud アカウントにサインインしているほかのデバイスがすぐ近くにあり、インターネット共有に対応している場合、それらのデバイスは信号を検出して応答し、デバイスが使用可能であることを示します。

ユーザがインターネット共有に使用できるデバイスを選択すると、インターネット共有をオンにするリクエストがそのデバイスに送信されます。このリクエストは、Bluetooth Low Energy の標準の暗号化を使用して暗号化されたリンクで送信され、リクエスト自体も iMessage と同様に暗号化されます。その後、デバイスは、同様に各メッセージを暗号化し、同じ Bluetooth Low Energy リンクを介して、インターネット共有の接続情報を含む応答を返します。

Safari 検索候補、Spotlight 検索候補、「調べる」、# イメージ、および「News」が提供されていない国での「News」ウィジェット

Safari 検索候補、Spotlight 検索候補、「調べる」、# イメージ、および「News」が提供されていない国での「News」ウィジェットには、Wikipedia、iTunes Store、ローカルニュース、「マップ」の検索結果、App Store などのソースから取得された、デバイスの対応範囲を超える検索候補がユーザに表示されます。検索候補はユーザが入力を開始する前から表示されることもあります。

ユーザが「Safari」のアドレスバーで入力を開始したり、Spotlight を開いたり使用したり、「調べる」を使用したり、# イメージを開いたり、「News」が提供されていない国での「News」ウィジェットを使用したりすると、以下のコンテキストが HTTPS で暗号化されて Apple に送信され、該当する結果がユーザに提供されます：

- プライバシーを保護するために 15 分おきに入れ替えられる識別子
- ユーザの検索クエリ
- デバイスの大まかな位置（「位置情報サービス」で「位置情報に基づく検索候補」を有効にした場合）。位置情報を「大まかにする」度合いは、デバイスの現在位置の推定人口密度に基づきます。たとえば、ユーザ同士が地理的に大きく離れている可能性がある地方ではより大まかになり、一般にユーザが密集する都市の中心部ではあまり大まかになりません。ユーザは「設定」で Apple へのあらゆる位置情報の送信を無効にできます。そのためには、「位置情報サービス」で「位置情報に基づく検索候補」をオフにします。位置情報サービスが無効な場合でも、Apple はデバイスの IP アドレスを使用しておおよその位置を推測することがあります。
- デバイスの種類と、検索が Spotlight、「Safari」、「調べる」、または「メッセージ」のいずれで行われているか
- 接続の種類
- デバイス上で最後に使用された 3 つの App の情報（補足的な検索コンテキストを提供するために使用）。Apple が維持する一般的な App の許可リストに含まれ、かつ過去 3 時間以内にアクセスされた App のみが含まれます。
- デバイス上で使用頻度が高いアプリケーションのリスト

- 地域の言語、ロケール、および入力環境設定
- ユーザのデバイスが音楽やビデオのサブスクリプションサービスにアクセスできる場合、サブスクリプションサービスの名前やサブスクリプションの種類などの情報が Apple に送信される場合があります。ユーザのアカウント名、番号、およびパスワードは Apple に送信されません。

ユーザがいずれかの検索結果を選択するか、結果を選択せずに Spotlight を終了すると、将来の検索結果の品質向上に役立てるため、一部の情報が Apple に送信されます。この情報は同じ 15 分間のセッション識別子に関連付けられ、特定のユーザには関連付けられません。このフィードバックには、上記のコンテキスト情報と以下の情報が含まれます：

- 操作と、検索ネットワーク要求の間のタイミング
- 検索候補のランキングと表示順序
- 検索結果の ID と、位置情報に基づかない結果の場合は選択されたアクション、位置情報に基づく結果の場合は選択された結果のカテゴリ

Apple は「検索候補」のログを、クエリ、コンテキスト、およびフィードバックと共に 18 か月間保持します。さらに、クエリ、国、言語、日時（時間単位）、およびデバイスの種類のみを含む限定的なログが 2 年間保持されます。

場合によっては、「検索候補」で、認定パートナー（Microsoft の Bing 検索エンジンなど）の検索結果を受信するために、一般的な語句のクエリが認定パートナーに転送されることがあります。パートナーはクエリの保存を許可されず、検索フィードバックも受信しません。Apple は、パートナーがクエリと共にユーザの IP アドレスを受信しないように、プロキシを介してクエリを送信します。パートナーとの通信は HTTPS で暗号化されます。Apple は頻繁に実行されるクエリについて、検索のパフォーマンスを改善するために、都市レベルの位置情報、デバイスの種類、クライアントの言語を検索コンテキストとしてパートナーに送信します。

さまざまな地理的位置およびネットワークの種類での「検索候補」のパフォーマンスを把握して改善するため、以下の情報がセッション識別子なしで記録されます：

- 部分的な IP アドレス（IPv4 アドレスの場合は末尾の 8 ビット、IPv6 アドレスの場合は末尾の 80 ビットを抜いたもの）
- 大まかな位置情報
- クエリの大まかな時刻
- レイテンシ／転送速度
- 応答のサイズ
- 接続の種類
- ロケール
- デバイスの種類とリクエスト元の App

デバイスの制御

iOS は、適用および管理しやすい柔軟なセキュリティポリシーと構成をサポートしています。そのため、「デバイス持参」（BYOD）プログラムの一環などで従業員が自ら用意したデバイスを使用していても、組織は企業情報を保護でき、従業員に社内要件を順守させることができます。

組織はパスコードによる保護、構成プロファイル、リモートワイプ、他社製 MDM ソリューションなどのリソースを使用してデバイス群を管理し、従業員が個人用の iOS デバイスで企業データにアクセスする場合でもそのデータを保護することができます。

パスコードによる保護

デフォルトでは、ユーザのパスコードは数字の PIN として定義できます。Touch ID に対応しているデバイスでは、最小のパスコード長は 6 桁です。その他のデバイスでは、最小長は 4 桁です。ユーザが「設定」>「パスコード」の「パスコードオプション」で「カスタムの英数字コード」を選択すると、より長い英数字のパスコードを指定できます。長く複雑なパスコードは推測や攻撃が困難なため、企業で使用する場合に推奨されます。

管理者は MDM または Exchange ActiveSync を使用して、あるいは構成プロファイルを手動でインストールするようユーザに求めることで、複雑なパスコードの要件を適用できます。以下のパスコードポリシーを適用できます：

- 単純値を許可
- 英数字の値が必要
- 最小のパスコード長
- 複合文字の最小数
- パスコードの有効期限
- パスコードの履歴
- 自動ロックのタイムアウト
- デバイスがロックされるまでの時間
- 入力失敗の許容回数
- Touch ID を許可

各ポリシーについて詳しくは、developer.apple.com/library/ios/featuredarticles/iPhoneConfigurationProfileRef を参照してください。

iOS のペアリングモデル

iOS では、ホストコンピュータからデバイスへのアクセスを制御するためにペアリングモデルが使用されます。ペアリングが実行されると、デバイスとそれに接続されたホストとの間に信頼関係が確立されます。これは公開鍵の交換によって行われます。iOS では、接続されたホストとの間でデータの同期などの追加機能を有効にするときに、この信頼の証が使用されます。iOS 9 では、ペアリングが必要なサービスはユーザがデバイスのロックを解除するまで起動できません。また、iOS 10 では、写真の同期などの一部のサービスを開始するためにデバイスのロックを解除する必要があります。

ペアリングプロセスでは、ユーザがデバイスのロックを解除し、ホストからのペアリング要求を受け入れる必要があります。ユーザがこれを行うと、ホストとデバイスが 2048 ビットの RSA 公開鍵を交換して保存します。次に、デバイス上に保存されているエスクローキーバッグのロックを解除できる 256 ビットの鍵がホストに提供されます（「キーバッグ」セクションの「エスクローキーバッグ」を参照）。交換された鍵は、暗号化された SSL セッションを開始するために使用されます。デバイスでは、保護されたデータをホストに送信したり、サービス（iTunes 同期、ファイル転送、Xcode 開発など）を開始したりする前に、このセッションを開始する必要があります。デバイスでこの暗号化されたセッションをすべての通信に使用するには、ホストからの Wi-Fi 経由での接続が必要なため、デバイスがあらかじめ USB でペアリングされている必要があります。ペアリングによって、いくつかの診断機能も有効になります。iOS 9 では、ペアリングの記録は 6 か月以上使用されないと期限切れになります。詳しくは、support.apple.com/ja-jp/HT6331 を参照してください。

com.apple.pcapd などの特定のサービスは、USB 経由でのみ機能するように制限されています。また、com.apple.file_relay サービスのために、Apple が署名した構成プロファイルをインストールする必要があります。

ユーザは「ネットワーク設定をリセット」または「位置情報とプライバシーをリセット」オプションを使用して、信頼するホストのリストを消去できます。詳しくは、support.apple.com/ja-jp/HT5868 を参照してください。

構成の適用

構成プロファイルは、管理者が構成情報を iOS デバイスに配布するために使用できる XML ファイルです。インストールされた構成プロファイルで定義されている設定は、ユーザが変更することはできません。ユーザが構成プロファイルを削除すると、そのプロファイルで定義されたすべての設定も削除されます。管理者はこの方法で、ポリシーをアクセス権に関連付けて設定を適用できます。たとえば、メールの構成を指定する構成プロファイルで、デバイスのパスコードポリシーを指定することもできます。ユーザは管理者が定めた要件に適合するパスコードを使用しない限り、メールにアクセスできません。

iOS 構成プロファイルには、以下のように、さまざまな指定可能な設定が含まれています：

- パスコードポリシー
- デバイスの機能制限（カメラを無効にするなど）
- Wi-Fi 設定
- VPN 設定
- メールサーバ設定
- Exchange 設定
- LDAP ディレクトリサービスの設定
- CalDAV カレンダーサービスの設定
- Web クリップ
- 資格情報および鍵
- モバイルデータ通信ネットワークの詳細設定

構成プロファイルは、送信元を検証し、整合性を確認し、内容を保護するために署名および暗号化できます。構成プロファイルは、CMS（RFC 3852）を使用して暗号化され、3DES および AES-128 をサポートします。

構成プロファイルをデバイス上にロックして、まったく削除できなくするか、パスコードを入力した場合にのみ削除できるようにすることもできます。多くの企業ユーザが自己所有の iOS デバイスを使用しているため、デバイスを MDM サーバにバインドする構成プロファイルは削除できます。ただし、削除すると、管理対象の構成情報、データ、および App もすべて削除されます。

ユーザは「Apple Configurator」を使って構成プロファイルをデバイスに直接インストールできます。また、構成プロファイルを「Safari」でダウンロードしたり、メールメッセージで受信したり、MDM サーバを使用してワイヤレスで受信したりすることもできます。

モバイルデバイス管理（MDM）

iOS は MDM をサポートしているため、企業は組織全体にわたる大規模な iPhone および iPad の導入を安全に構成して管理できます。MDM の機能は、構成プロファイル、ワイヤレスでの登録、Apple Push Notification service（APNs）などの既存の iOS テクノロジーを基礎としています。たとえば、APNs は、デバイスが MDM サーバとセキュリティ保護された接続で直接通信できるようにデバイスをスリープ解除する目的で使用されます。機密情報や専有情報が APNs 経由で伝送されることはありません。

IT 部門は MDM を使用して、企業環境内に iOS デバイスを登録したり、設定をワイヤレスで構成およびアップデートしたり、企業ポリシーへの準拠を監視したりでき、管理対象デバイスをリモートでワイプまたはロックすることもできます。モバイルデバイス管理について詳しくは、www.apple.com/iphone/business/it/management.html を参照してください。

共有 iPad

共有 iPad とは、iPad を導入した教育機関が使用するマルチユーザモード構成の iPad のことです。複数の生徒が書類やデータを共有することなく 1 台の iPad を共有できます。共有 iPad では、学校が発行および所有する管理対象 Apple ID を使用する必要があります。共有 iPad は複数の生徒で使用できるように構成されているため、生徒は教育機関が所有するどの iPad にもサインインできます。

生徒のデータは別々のホームディレクトリに分割され、各ディレクトリは UNIX のアクセス権とサンドボックスの両方で保護されます。生徒がサインインすると、管理対象 Apple ID が Apple の認証サーバによって SRP プロトコル経由で認証されます。認証に成功すると、そのデバイス専用の一時的なアクセストークンが付与されます。生徒がそのデバイスを以前に使ったことがある場合は、ロック解除されたローカルユーザアカウントがすでに設定されています。生徒がそのデバイスを以前に使ったことがない場合は、新しい UNIX ユーザ ID、ホームディレクトリ、およびキーチェーンがプロビジョニングされます。（そのデバイスがインターネットに接続していない場合には、既存のローカルアカウントをすでに持っているユーザのみがサインインできます。）

生徒のローカルアカウントがロック解除または作成され、リモートで認証されると、Apple のサーバによって発行された一時的なトークンが、iCloud へのサインインを許可する iCloud トークンに変換されます。次に、生徒の設定が復元され、その生徒の書類やデータが iCloud から同期されます。

生徒のセッションが進行中でデバイスがオンラインになっている間は、書類やデータが作成または変更されると iCloud に保存されます。また、バックグラウンド同期メカニズムによって、生徒のサインアウト後も、変更された書類やデータが iCloud にプッシュされます。

Apple School Manager

Apple School Manager は、教育機関が、コンテンツの購入、モバイルデバイス管理（MDM）ソリューションでの自動デバイス登録の構成、生徒と職員用のアカウント作成、iTunes U コースの設定ができるサービスです。Apple School Manager は Web でアクセスでき、技術マネージャ、IT 管理者、職員、および教師が利用できるように設計されています。

デバイス登録

Device Enrollment Program（DEP）は、組織が Apple から直接購入したか、Apple 製品取扱店や通信事業者から購入した iOS デバイスを迅速かつ効率的に導入する方法を提供します。デバイスを登録する機能は、教育機関向けの Apple School Manager にも搭載されています。

組織はユーザに渡す前にデバイスに物理的に触れたり準備したりすることなく、デバイスを MDM に自動的に登録できます。プログラムで登録後に、管理者はプログラム Web サイトにログインし、プログラムを MDM サーバにリンクします。すると、購入したデバイスを MDM 経由でユーザに割り当てることができます。ユーザの割り当てが完了すると、MDM で指定された構成、制限、または制御が自動的にインストールされます。デバイスと Apple サーバ間のすべての通信は、HTTPS（SSL）経由で転送時に暗号化されます。

設定アシスタントの特定の手順を省略してユーザの設定プロセスをさらに簡素化できるため、ユーザがすぐにデバイスを使い始めることができます。ユーザがデバイスから MDM プロファイルを削除できるかどうかを管理者が制御して、デバイスの制限が最初から適用されるようにすることもできます。デバイスを箱から出してアクティベートすると、デバイスが組織の MDM に登録され、すべての管理設定、App、およびブックがインストールされます。

詳しくは、help.apple.com/deployment/business または help.apple.com/schoolmanager（教育機関向け）を参照してください。

注記：デバイス登録機能を利用できない国や地域があります。

Apple Configurator 2

MDM だけでなく macOS 用の「Apple Configurator 2」でも、ユーザに渡す前にデバイスの設定や事前構成を簡単に行えます。「Apple Configurator」を使用すると、App、データ、機能制限、および設定をあらかじめデバイスにすばやく構成できます。

「Apple Configurator 2」では、Apple School Manager（教育機関向け）または Device Enrollment Program（企業向け）を使用してデバイスをモバイルデバイス管理（MDM）ソリューションに登録できるため、ユーザが設定アシスタントを使う必要はありません。

監視

デバイスの設定中に、組織がデバイスを監視対象として構成できます。監視対象であるということは、デバイスが組織に所有されているということです。そのため、デバイスの構成および制限がより厳密に制御されます。デバイスは、Apple School Manager、Device Enrollment Program、または「Apple Configurator」による設定時に監視対象にできます。

MDM と「Apple Configurator 2」を使用するデバイスの構成および管理方法について詳しくは、help.apple.com/deployment/ios を参照してください。

機能制限

管理者は構成プロファイルをインストールすることによって機能を制限できます。以下の制限を設定できます：

- App のインストールを許可
- エンタープライズ App の信頼を許可
- カメラの使用を許可
- FaceTime を許可
- スクリーンショットを許可
- ロック中の音声ダイヤルを許可
- ローミング中の自動同期を許可
- App 内での購入を許可
- 最近のメールの同期を許可
- 購入時に常に Store パスワードの入力を強制
- デバイスのロック中も Siri を許可
- iTunes Store の使用を許可
- 管理対象外出力先で管理対象ソースからの書類を許可
- 管理対象出力先で管理対象外ソースからの書類を許可
- iCloud キーチェーンの同期を許可
- 証明書信頼データベースのワイヤレスでのアップデートを許可
- ロック画面での通知の表示を許可
- AirPlay 接続でのペアリングパスワードの使用を強制
- Spotlight でのインターネットからのユーザ生成コンテンツの表示を許可
- Spotlight で「Spotlight の検索候補」を許可
- Handoff を許可
- AirDrop を管理対象外出力先として処理
- 企業により配布されたブックのバックアップ作成を許可
- 企業により配布されたブックのメモとブックマークをユーザの複数のデバイス間で同期することを許可
- 「Safari」の使用を許可
- 「Safari」の自動入力を有効にする
- 詐欺 Web サイトの警告を強制
- JavaScript を有効にする
- 「Safari」で追跡型広告を制限
- ポップアップをブロック
- Cookie を受け入れる
- iCloud バックアップを許可
- iCloud 書類とキー値の同期を許可
- iCloud 写真共有を許可
- Apple への診断情報の送信を許可
- 信頼されていない TLS 証明書の受け入れをユーザに許可
- 強制的に暗号化バックアップ
- Touch ID を許可
- ロック画面からのコントロールセンターへのアクセスを許可
- ロック画面での「今日」表示を許可
- Apple Watch の手首検出を強制
- 「クラスルーム」による画面の監視を許可
- 管理対象 App からの AirDrop を許可
- 新しいエンタープライズ App デベロッパを信頼
- iCloud フォトライブラリを使用

監視対象のみの制限

- 「iMessage」を許可
- App の削除を許可
- 構成プロファイルの手動インストールを許可
- HTTP 用グローバルネットワークプロキシ
- コンテンツ同期用のコンピュータとのペアリングを許可
- ホワイトリストとオプションの接続パスコードで AirPlay 接続を制限
- AirDrop を許可

- 「友達を探す」設定の変更を許可
- 特定の管理対象 App で自律的シングル App モードを許可
- アカウントの変更を許可
- モバイルデータ通信設定の変更を許可
- ホストとのペアリングを許可（iTunes）
- アクティベーションロックを許可
- 「すべてのコンテンツと設定を消去」を禁止
- 制限の有効化を禁止
- 他社製のコンテンツフィルタ
- シングル App モード
- VPN 常時接続
- パスコードの変更を許可
- Apple Watch のペアリングを許可
- App の自動ダウンロードを許可
- キーボードでの予測変換、自動修正、スペルチェック、およびショートカットを許可
- Apple Music を許可
- Radio を許可
- 「クラスルーム」で画面を監視
- 通知の設定を変更
- ホーム画面で特定の App を表示／隠す
- App Store を使用して App をインストール
- App を自動ダウンロード
- キーボードショートカット
- 定義を許可
- デバイス名を変更
- 壁紙を変更
- 「News」App を隠す
- Apple Watch とペアリング

監視対象デバイスに対する MDM デベロッパのその他の制御について詳しくは、developer.apple.com/library/ios/featuredarticles/iPhoneConfigurationProfileRef を参照してください。

リモートワイプ

iOS デバイスは、管理者またはユーザがリモートでワイプできます。Effaceable Storage からブロックストレージの暗号化鍵を安全に破棄し、すべてのデータを読み取れない状態にすることによって、リモートワイプを瞬時に実行できます。リモートワイプ・コマンドは、MDM、Exchange、または iCloud によって開始できます。

MDM または iCloud によってリモートワイプ・コマンドが始動すると、デバイスから確認応答が送信されてワイプが実行されます。Exchange によるリモートワイプの場合は、ワイプの実行前に Exchange Server でデバイスがチェックインされます。

ユーザも、自分が所有するデバイスを「設定」App でワイプできます。前に述べたように、パスコードの誤入力が続いた場合に、デバイスが自動的にワイプされるように設定できます。

紛失モード

iOS 9.3 以降の監視対象デバイスでは、デバイスを紛失したり盗まれたりした場合に、MDM 管理者が紛失モードをリモートで有効にできます。紛失モードを有効にすると、現在のユーザはログアウトされ、デバイスはロック解除できない状態になります。画面には、このデバイスを見つけた場合の連絡先電話番号など、管理者がカスタマイズしたメッセージを表示できます。デバイスが紛失モードになると、管理者はそのデバイスに対し、現在位置を送信するよう要求できます。管理者が紛失モードをオフにすると（これが紛失モードを終了する唯一の方法です）、そのことがロック画面のメッセージとホーム画面の警告でユーザに通知されます。

アクティベーションロック

「iPhone を探す」がオンの場合、デバイスを再アクティベーションするには所有者の Apple ID のアカウント情報を必ず入力する必要があります。

組織が所有しているデバイスでは、再アクティベーションのために各ユーザが自分の Apple ID アカウント情報を入力しなくても済むように、デバイスを監視して組織がアクティベーションロックを管理できるようにすることをお勧めします。

監視対象デバイスでは、対応する MDM ソリューションを使用して、アクティベーションロックが有効になった時点でバイパスコードを保存しておき、後でデバイスを消去して別のユーザに割り当てる必要が生じたときにはこのコードを使ってアクティベーションロックを自動的に解除することができます。詳しくは、MDM ソリューションの資料を参照してください。

デフォルトでは、ユーザが「iPhone を探す」をオンにしたとしても、監視対象デバイスのアクティベーションロックは有効になりません。ただし、MDM サーバがバイパスコードを取得し、デバイスでのアクティベーションロックの有効化を許可する場合があります。MDM サーバがデバイスのアクティベーションロックを有効にしたときに「iPhone を探す」がオンになっていると、その時点でアクティベーションロックが有効になります。MDM サーバがアクティベーションロックを有効にしたときに「iPhone を探す」がオフになっていると、アクティベーションロックはユーザが次回「iPhone を探す」をオンにしたときに有効になります。

Apple School Manager で作成された管理対象 Apple ID を持つ教育機関用のデバイスでは、アクティベーションロックをユーザの Apple ID ではなく管理者の Apple ID に関連付けたり、デバイスのバイパスコードを使用して無効にしたりすることができます。

プライバシーの制御

Apple はお客様のプライバシーを非常に重視しており、iOS ユーザの個人情報が App でいつどのように使用されるか、およびどのような情報が使用されるかをユーザ自身が決定できる組み込みの制御機能やオプションを多数用意しています。

位置情報サービス

位置情報サービスでは、GPS、Bluetooth、クラウドソースの Wi-Fi ホットスポットや携帯電話基地局を使って、ユーザのおおよその位置が判断されます。位置情報サービスは「設定」にある 1 つのスイッチでオフにできます。または、ユーザがこのサービスを利用する App ごとにアクセスを承認することもできます。App が使用中にのみ位置情報データの受信を要求できるようにするか、いつでも要求できるようにするかを選択できます。このアクセスを許可しないという選択もでき、選択内容は「設定」でいつでも変更できます。「設定」で、App が要求する位置情報の使用方法に応じて、常に禁止する、App の使用中のみ許可する、または常に許可するというアクセスの設定ができます。また、位置情報サービスを常に使用できるアクセスを与えられた App がバックグラウンドモードのときにその権限を使用すると、ユーザにそれを承認したことを示す通知が表示され、ユーザは App のアクセスを変更できます。

さらに、システムサービスでの位置情報の利用も、ユーザが細かく制御できます。たとえば、Apple が iOS を改善するために利用する診断および使用状況のサービスで収集される情報や、位置情報に基づく Siri の情報、「Spotlight の検索候補」での検索に使用される位置情報に基づくコンテキスト、周辺の交通情報、および移動時間の推定に使用される頻繁に訪れる場所の情報に、位置情報を含めることを停止できます。

個人データへのアクセス

iOS では、App がユーザの個人情報に許可なくアクセスすることを防止できます。また、「設定」で、ユーザが特定の情報へのアクセスを許可した App の確認と、将来のアクセスの許可または取り消しもできます。これには、以下の項目へのアクセスが含まれます：

- 連絡先
- カレンダー
- リマインダー
- 写真
- モーションとフィットネス
- 位置情報サービス
- メディアライブラリ
- Twitter や Facebook などのソーシャルメディアのアカウント
- マイク
- カメラ
- HomeKit
- ヘルスケア
- 音声認識
- Bluetooth 共有

ユーザが iCloud にサインインしている場合は、デフォルトで App に iCloud Drive へのアクセスが与えられます。ユーザは「設定」の「iCloud」で各 App のアクセスを管理できます。また、iOS では、MDM によってインストールされた App およびアカウントと、ユーザがインストールした App およびアカウントの間でのデータ移動を禁止するという制限を設定できます。

プライバシーポリシー

Apple のプライバシーポリシーは、オンラインの www.apple.com/legal/privacy/jp で参照できます。

Apple セキュリティバウンティ

Apple は、重大な問題を Apple と共有した研究者に報奨金を支払います。Apple セキュリティバウンティに応募するには、明確な報告書と実用的な概念実証（PoC）を提出する必要があります。対象となる脆弱性は、最新のハードウェア上の最新リリースの iOS に影響するものに限られます。報奨金の正確な額は、Apple によるレビュー後に決まります。査定では、発見の難しさ、危険度、必要なユーザ操作の度合いなどが基準になります。

問題の存在が確認されると、Apple は問題をできる限り早期に解決することを優先事項とします。また、適切な場合は、情報提供者を表彰します。（そうしないことを要請された場合を除く）。

カテゴリ	**最高支払額（米ドル）**
セキュア・ブート・ファームウェア・コンポーネント	$200,000
Secure Enclave で保護されている機密資料の抽出	$100,000
カーネル権限による任意のコードの実行	$50,000
Apple サーバ上の iCloud アカウントデータへの不正アクセス	$50,000
サンドボックス化されたプロセスから、そのサンドボックスの外部にあるユーザデータへのアクセス	$25,000

総括

セキュリティへの取り組み

Apple は、個人情報を保護するために設計されたプライバシーおよびセキュリティに関する先進的な技術と、企業環境内での企業データの保護に役立つ包括的な手法により、お客様を守ることに力を注いでいます。

iOS にはセキュリティが組み込まれています。プラットフォームからネットワーク、さらには App まで、企業に必要なあらゆるものが iOS プラットフォームで利用可能です。iOS はこれらの要素の組み合わせにより、ユーザエクスペリエンスを犠牲にすることなく、業界をリードするセキュリティを実現しています。

Apple は、iOS および iOS App のエコシステム全体を通じて、一貫した統合セキュリティインフラストラクチャを採用しています。ハードウェアベースのストレージ暗号化により、デバイスを紛失した場合にリモートワイプ機能を使用でき、デバイスを他者に売却または譲渡する場合にもユーザがすべての企業情報と個人情報を完全に削除できます。診断情報も匿名で収集されます。

Apple が設計した iOS App は、高度なセキュリティを念頭に置いて構築されています。「Safari」は、オンライン証明書状況プロトコル（OCSP）、EV 証明書、および証明書検証の警告のサポートにより、安全なブラウジングを提供します。「メール」では、S/MIME のサポートにより、「メール」の認証および暗号化のために証明書が活用されます。メッセージ単位の S/MIME が許可されるため、S/MIME ユーザはデフォルトで常に署名および暗号化するか、個別のメッセージの保護方法を選択的に制御するかを選択できます。iMessage と FaceTime でも、クライアント間での暗号化が行われます。

他社製 App については、必須のコード署名、サンドボックス化、およびエンタイトルメントの組み合わせによって、ほかのプラットフォームのセキュリティを侵害するウイルス、マルウェアやその他の悪用からユーザが確実に保護されます。App Store の提出プロセスは、すべての iOS App を発売前にレビューすることによって、さらにユーザを保護する役割を果たします。

iOS に組み込まれた幅広いセキュリティ機能を最大限に活用するため、企業には自社の IT ポリシーとセキュリティポリシーを見直し、このプラットフォームで提供されている何重ものセキュリティを十分活かせるものにすることをお勧めします。

Apple にはすべての Apple 製品をサポートする専任のセキュリティチームがあります。このチームは、開発中の製品だけでなく、リリース済みの製品についても、セキュリティの監査とテストを実施します。また、この Apple のチームはセキュリティツールやトレーニングを提供すると共に、セキュリティに関する新しい問題や脅威の報告がないか積極的に監視しています。Apple は Forum of Incident Response and Security Teams（FIRST）のメンバーです。Apple への問題の報告およびセキュリティ通知の購読について詳しくは、www.apple.com/jp/support/security を参照してください。

用語集

アドレス空間配置のランダム化（ASLR）	iOS に採用されている、ソフトウェアのバグの悪用をはるかに困難にする技術。メモリアドレスとオフセットが予測不能になるため、悪意のあるコードでそれらの値をハードコーディングできなくなります。iOS 5 以降では、すべてのシステム App およびライブラリの位置がランダム化されると共に、すべての他社製 App が位置に依存しない実行可能ファイルとしてコンパイルされます。
Apple Push Notification service（APNs）	iOS デバイスにプッシュ通知を配信する、Apple が世界中で提供しているサービス。
Boot ROM	デバイスが起動したときに最初に実行されるコード。プロセッサに不可欠な部分であるため、Apple にも攻撃者にも変更できません。
データ保護	iOS 用のファイルおよびキーチェーン保護メカニズム。App で使用される API を参照して、ファイルおよびキーチェーン項目を保護することもできます。
デバイス・ファームウェア・アップグレード（DFU）	デバイスの Boot ROM のコードが USB 経由で復元されるまで待機するモード。DFU モードのときは画面が真っ暗になりますが、「iTunes」を実行しているコンピュータに接続すると、「iTunes はリカバリモードの iPad を見つけました。iTunes でご利用になる前に、この iPad を復元する必要があります。」というメッセージが表示されます。
ECID	各 iOS デバイスのプロセッサに固有の 64 ビットの識別子。あるデバイスで着信に応答すると、Bluetooth Low Energy 4.0 経由で短くアドバタイズすることで、iCloud でペアリングされた近くにあるデバイスの着信音が停止します。アドバタイズバイトは、Handoff アドバタイズと同じ方法で暗号化されます。パーソナライズプロセスの一部として使用され、秘密とは見なされません。
Effaceable Storage	暗号鍵を保存するために使用される NAND ストレージの専用領域。直接アドレス指定でき、安全にワイプできます。攻撃者がデバイスを物理的に入手した場合は保護手段となりませんが、Effaceable Storage に保存されている鍵を鍵階層の一部として使用することで、高速のワイプと前方秘匿性を実現できます。
ファイルシステム鍵	各ファイルのクラス鍵などのメタデータを暗号化する鍵です。これは、機密保持ではなく高速のワイプを可能にするために、Effaceable Storage に保管されます。
グループ ID（GID）	UID と同じようなものですが、クラス内のすべてのプロセッサで共通です。
ハードウェア・セキュリティ・モジュール（HSM）	デジタル鍵の保護および管理に特化した、改ざん耐性を持つコンピュータ。
iBoot	LLB によって読み込まれるコード。セキュアブートチェーンの一部として XNU を読み込みます。
Identity Service（IDS）	iMessage の公開鍵、APNs アドレス、および電話番号とメールアドレスを含む Apple のディレクトリ。鍵およびデバイスのアドレスの検索に使用されます。
集積回路（IC）	マイクロチップとも呼ばれます。
Joint Test Action Group（JTAG）	プログラマや回路デベロッパが使用するハードウェアの標準デバッグツール。
キーバッグ	クラス鍵のコレクションを保存するために使用されるデータ構造。各タイプ（ユーザ、デバイス、システム、バックアップ、エスクロー、または iCloud バックアップ）のフォーマットは同じです： • 以下を含むヘッダ：バージョン（iOS 5 では 3 に設定されます） – タイプ（システム、バックアップ、エスクロー、または iCloud バックアップ） – キーバッグの UUID – HMAC（キーバッグが署名されている場合） – クラス鍵のラッピングに使用される方式：UID または PBKDF2 とのタングルと、Salt および反復回数 • クラス鍵のリスト： – 鍵の UUID – クラス（ファイルまたはキーチェーンのデータ保護クラス） – ラッピングのタイプ（UID から導出された鍵のみ／ UID から導出された鍵とパスコードから導出された鍵） – ラップされたクラス鍵 – 非対称クラスの公開鍵
キーチェーン	パスワードや鍵、機密性の高いその他の資格情報を保存したり取得したりするために iOS および他社製 App で使用されるインフラストラクチャおよび API セット。
鍵ラッピング	1 つの鍵を別の鍵で暗号化すること。iOS では RFC 3394 準拠の NIST AES 鍵ラッピングが使用されます。
Low-Level Bootloader（LLB）	Boot ROM によって呼び出され、セキュアブートチェーンの一部として iBoot を読み込むコード。

Per File キー	ファイルシステム上のファイルの暗号化に使用される AES 256 ビット鍵。Per File キーはクラス鍵でラップされ、ファイルのメタデータに保存されます。
プロビジョニングプロファイル	App を iOS デバイスにインストールしてテストできるようにする一連のエンティティおよびエンタイトルメントを含む、Apple によって署名されたプロパティリスト。開発プロビジョニングプロファイルにはデベロッパがアドホック配信用に選択したデバイスのリストが含まれ、配信プロビジョニングプロファイルには企業によって開発された App の App ID が含まれます。
皮下の隆線角度のマッピング	指紋の一部から抽出されたリッジ（隆起部）の向きと幅を数学的に表現したもの。
スマートカード	安全な識別、認証、およびデータ保存を可能にする組み込み集積回路。
System on a chip（SoC）	複数のコンポーネントを 1 つのチップに組み込んだ集積回路。Secure Enclave は、Apple の A7 以降の中央プロセッサ内にある SoC です。
タングル	ユーザのパスコードが暗号鍵に変換され、デバイスの UID と組み合わせて強化されるプロセス。これによって、どのデバイスを侵害するにも総当たり（ブルートフォース）攻撃が必要になるため、攻撃の速度が制限され、攻撃を並列的に実行できなくなります。タングルに使用されるアルゴリズムは PBKDF2 です。このアルゴリズムの各反復では、デバイス UID を鍵とする AES が擬似ランダム関数（PRF）として使用されます。
Uniform Resource Identifier（URI）	Web ベースのリソースを識別する文字列。
固有 ID（UID）	製造時に各プロセッサに焼き付けられる AES 256 ビット鍵。ファームウェアまたはソフトウェアによって読み出すことはできず、プロセッサのハードウェア AES エンジンによってのみ使用されます。攻撃者が実際の鍵を取得するには、プロセッサのシリコンに対して非常に高度でコストのかかる攻撃を仕掛ける必要があります。UID は、デバイス上にある UDID などのほかの識別子に関連しません。
XNU	iOS および macOS オペレーティングシステムの核心部にあるカーネル。前提として信頼され、コード署名、サンドボックス化、エンタイトルメントの確認、ASLR などのセキュリティ対策を実行します。

本書の変更履歴

日付	概要
2017 年 3 月	iOS 10 向けにアップデート • システムのセキュリティ • データ保護クラス • セキュリティ認定とプログラム • HomeKit、ReplayKit、SiriKit • Apple Watch • Wi-Fi、VPN • シングルサインオン • Apple Pay、Apple Pay による Web 上での支払い • クレジットカード、デビットカード、プリペイドカードのプロビジョニング • Safari 検索候補 • iOS 10 のセキュリティコンテンツについて詳しくは、次の Web サイトを参照してください：support.apple.com/ja-jp/HT207143
2016 年 5 月	iOS 9.3 向けにアップデート • 管理対象 Apple ID • Apple ID の 2 ファクタ認証 • キーバッグ • セキュリティの認証 • 紛失モード、アクティベーションロック • 保護したメモ • Apple School Manager、共有 iPad • iOS 9.3 のセキュリティコンテンツについて詳しくは、次の Web サイトを参照してください：support.apple.com/ja-jp/HT206166
2015 年 9 月	iOS 9 向けにアップデート • Apple Watch のアクティベーションロック • パスコードポリシー • Touch ID API のサポート • A8 でのデータ保護に AES-XTS を使用 • 自動ソフトウェア・アップデート用のキーバッグ • 証明書のアップデート • エンタープライズ App の信頼モデル • Safari ブックマークのデータ保護 • App Transport Security • VPN 仕様 • HomeKit 用の iCloud リモートアクセス • Apple Pay のポイントカード、Apple Pay のカード会社の App • Spotlight のデバイス上でのインデックス付け • iOS のペアリングモデル • Apple Configurator 2 • 機能制限 • iOS 9 のセキュリティコンテンツについて詳しくは、次の Web サイトを参照してください：support.apple.com/ja-jp/HT205212