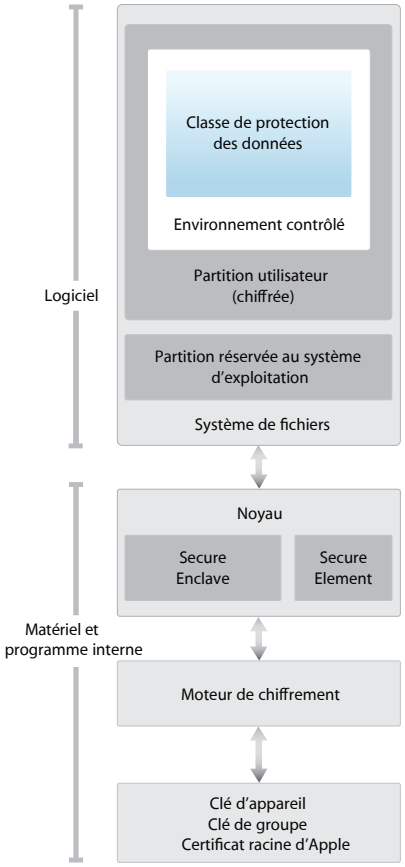
Sécurité iOS

iOS 10

Mars 2017

Table des matières

Introduction



Le schéma de l'architecture de sécurité d'iOS fournit une vue d'ensemble des différentes technologies présentées dans ce document.

Apple a conçu la plateforme iOS en mettant l'accent sur la sécurité. Quand nous avons entrepris de créer la meilleure plateforme mobile qui soit, nous avons mis à profit plusieurs décennies d'expérience pour mettre au point une architecture entièrement nouvelle. Nous avons pris en compte les risques de sécurité dans l'environnement de bureau et adopté une nouvelle approche de la sécurité lors de la conception d'iOS. Nous avons développé et intégré des fonctionnalités innovantes qui renforcent la sécurité mobile et protègent l'ensemble du système par défaut. iOS constitue donc une avancée majeure en termes de sécurité des appareils mobiles.

Chaque appareil iOS combine des technologies logicielles et matérielles et des services qui fonctionnent ensemble pour offrir une sécurité et une transparence maximales sans interférer avec l'expérience de l'utilisateur. iOS protège non seulement l'appareil et ses données, mais également l'ensemble de l'écosystème, notamment tout ce que les utilisateurs font localement, sur les réseaux et avec des services Internet clés.

Même si iOS et les appareils iOS offrent des fonctionnalités de sécurité avancées, ils n'en restent pas moins simples d'utilisation. Bon nombre de ces fonctionnalités étant activées par défaut, les services informatiques n'ont pas à réaliser de configurations importantes. En outre, les fonctionnalités de sécurité clés comme le chiffrement de l'appareil ne sont pas configurables et ne peuvent donc pas être désactivées par mégarde. D'autres fonctionnalités, comme Touch ID, améliorent l'expérience de l'utilisateur en lui permettant de sécuriser l'appareil plus simplement et intuitivement.

Le présent document fournit des informations détaillées sur l'implémentation des technologies et des fonctionnalités de sécurité au sein de la plateforme iOS. Il aide également les organisations à combiner les technologies et les fonctionnalités de sécurité de la plateforme iOS avec leurs propres stratégies et procédures pour répondre à leurs besoins spécifiques en matière de sécurité.

Ce document s'articule autour des thèmes suivants :

- **Sécurité du système :** les technologies logicielles et matérielles intégrées et sécurisées qui constituent la plateforme de l'iPhone, l'iPad et l'iPod touch.
- **Chiffrement et protection des données :** l'architecture et la conception qui protègent les données utilisateur en cas de perte ou de vol de l'appareil, ou en cas de tentative d'utilisation ou de modification de celui-ci par une personne non autorisée.
- **Sécurité des apps :** les systèmes qui permettent aux apps de s'exécuter en toute sécurité et sans compromettre l'intégrité de la plateforme.
- **Sécurité du réseau :** les protocoles de mise en réseau standard qui assurent la sécurisation de l'authentification et le chiffrement des données lors des transmissions.
- **Apple Pay :** l'implémentation des paiements sécurisés d'Apple.
- **Services Internet :** l'infrastructure du réseau d'Apple pour la messagerie, la synchronisation et la sauvegarde.
- **Contrôles de l'appareil :** méthodes permettant de gérer des appareils iOS, d'empêcher l'usage non autorisé et d'activer l'effacement à distance si un appareil est perdu ou volé.
- **Contrôles de confidentialité :** les fonctionnalités d'iOS qui permettent de contrôler l'accès au service de localisation et aux données utilisateur.

Sécurité du système

Accès au mode de mise à niveau du logiciel interne d'un appareil (DFU) La restauration d'un appareil placé en mode DFU permet de revenir à un état de bon fonctionnement connu de cet appareil en ayant la certitude qu'il ne contient que du code intact signé par Apple. Le mode DFU est accessible manuellement : connectez d'abord l'appareil à un ordinateur à l'aide d'un câble USB, puis maintenez simultanément enfoncés le bouton principal et le bouton Marche/Veille. Relâchez le bouton Marche/Veille après 8 secondes tout en continuant à maintenir le bouton principal enfoncé. Remarque : rien ne s'affiche à l'écran lorsque l'appareil est en mode DFU. Si le logo Apple apparaît, c'est que le bouton Marche/Veille a été maintenu enfoncé trop longtemps.

La sécurité du système est conçue de sorte que le logiciel et le matériel soient sécurisés dans tous les composants clés de chaque appareil iOS. Cela inclut le processus de démarrage, les mises à jour du logiciel et Secure Enclave. Cette architecture est au cœur de la sécurité d'iOS et n'interfère jamais avec la convivialité de l'appareil.

L'intégration étroite des technologies logicielles et matérielles sur les appareils iOS garantit la sécurisation de chaque composant du système et valide celui-ci dans son ensemble. Du démarrage aux apps tierces, en passant par les mises à jour du logiciel iOS, chaque étape est analysée et contrôlée pour s'assurer que le logiciel et le matériel interagissent de manière optimale et utilisent correctement les ressources.

Chaîne de démarrage sécurisé

Chaque étape du processus de démarrage contient des composants qui sont signés cryptographiquement par Apple pour garantir leur intégrité et qui ne s'exécutent qu'une fois la chaîne de confiance vérifiée. Cela concerne notamment les chargeurs de démarrage, le noyau, les extensions du noyau et le programme interne de bande de base. Cette chaîne de démarrage sécurisé permet de s'assurer que les niveaux les plus bas du logiciel ne sont pas altérés.

Lorsqu'un appareil iOS est mis sous tension, son processeur d'application exécute immédiatement un code stocké dans une mémoire en lecture seule appelée ROM de démarrage. Ce code immuable, appelé racine de confiance matérielle, est défini lors de la fabrication de la puce et implicitement considéré comme fiable. Le code de la ROM de démarrage contient la clé publique d'AC racine d'Apple, qui est utilisée pour vérifier que le chargeur de démarrage iBoot est signé par Apple avant d'autoriser son chargement. Il s'agit de la première étape de la chaîne de confiance, dans laquelle chaque étape vérifie que la suivante est signée par Apple. Une fois les tâches d'iBoot terminées, le chargeur vérifie et exécute le noyau iOS. Pour les appareils dotés d'un S1, d'un A9 ou d'un autre processeur antérieur de la série A, une phase incluant le chargeur de démarrage de bas niveau (LLB) est chargée et vérifiée par la ROM de démarrage qui à son tour charge et vérifie iBoot.

Si une étape de ce processus de démarrage ne parvient pas à charger ou à vérifier le processus suivant, le démarrage est interrompu et l'appareil affiche l'écran « Se connecter à iTunes ». C'est ce qu'on appelle le mode de récupération. Si la ROM de démarrage ne parvient pas à charger ou à vérifier le LLB, elle passe en mode de mise à niveau du programme interne de l'appareil (DFU). Dans les deux cas, l'appareil doit être connecté à iTunes par USB pour rétablir ses réglages par défaut d'origine. Pour en savoir plus sur le passage manuel en mode de récupération, consultez l'article https://support.apple.com/fr-fr/HT1808.

Sur les appareils avec accès cellulaire, le sous-système de bande de base fait également appel à un processus de démarrage sécurisé similaire, avec logiciel signé et clés vérifiés par le processeur de bande de base.

Pour les appareils dotés de Secure Enclave, le coprocesseur Secure Enclave emploie aussi un processus de démarrage sécurisé qui garantit que son logiciel indépendant est vérifié et signé par Apple.

Autorisation du logiciel système

Apple publie régulièrement des mises à jour logicielles pour traiter les nouveaux problèmes de sécurité et offrir de nouvelles fonctionnalités ; ces mises à jour sont disponibles en même temps pour tous les appareils pris en charge. Les utilisateurs reçoivent des notifications de mise à jour d'iOS sur leur appareil et dans iTunes, et les mises à jour sont transmises par le biais d'une connexion sans fil, ce qui favorise l'adoption rapide des derniers correctifs de sécurité.

Le processus de démarrage décrit ci-dessus permet de s'assurer que seul le code signé par Apple peut être installé sur un appareil. Pour empêcher le retour des appareils à une version antérieure ne disposant pas des dernières mises à jour de sécurité, iOS utilise un processus appelé Autorisation du logiciel système. Si le retour à une version antérieure était possible, une personne malintentionnée entrant en possession d'un appareil pourrait installer une ancienne version d'iOS et exploiter une vulnérabilité corrigée dans la version plus récente.

Sur un appareil doté d'une Secure Enclave, le coprocesseur Secure Enclave s'appuie également sur l'autorisation du logiciel système pour garantir l'intégrité de son logiciel et empêcher l'installation d'une version antérieure. Consultez la section « Secure Enclave » qui suit.

Les mises à jour du logiciel iOS peuvent être installées sur l'appareil à l'aide d'iTunes ou par le biais d'une connexion sans fil (mode OTA). Avec iTunes, une copie complète d'iOS est téléchargée et installée. Les mises à jour du logiciel en mode OTA ne téléchargent que les composants nécessaires à la mise à jour, ce qui améliore l'efficacité du réseau, au lieu de télécharger l'intégralité du système d'exploitation. En outre, les mises à jour du logiciel peuvent être mises en cache sur un serveur de réseau local exécutant le service de mise en cache de macOS Server, ce qui évite aux appareils iOS d'accéder aux serveurs d'Apple pour obtenir les données de mise à jour nécessaires.

Lors d'une mise à niveau d'iOS, iTunes (ou l'appareil lui-même, dans le cas d'une mise à jour du logiciel en mode OTA) se connecte au serveur d'autorisation d'installation d'Apple et lui envoie une liste de mesures cryptographiques pour chaque partie du paquet à installer (par exemple, iBoot, le noyau et l'image du système d'exploitation), une valeur anti-répétition aléatoire (nonce) et l'identifiant unique de l'appareil (ECID).

Le serveur d'autorisation compare alors la liste de mesures qui lui est fournie et les versions pour lesquelles l'installation est autorisée et, s'il trouve une correspondance, ajoute l'ECID à la mesure et signe le résultat. Le serveur transmet un jeu complet de données signées à l'appareil dans le cadre du processus de mise à niveau. L'ajout de l'ECID « personnalise » l'autorisation pour l'appareil émetteur de la requête. En n'accordant son autorisation et sa signature que pour des mesures connues, le serveur garantit que la mise à jour se déroule exactement comme prévu par Apple.

L'évaluation de la chaîne de confiance au démarrage vérifie que la signature provient d'Apple et que la mesure de l'élément chargé depuis le disque, associée à l'ECID de l'appareil, correspond à ce que couvre la signature.

Ces étapes garantissent que l'autorisation a bien été accordée pour l'appareil concerné et qu'une ancienne version d'iOS d'un appareil ne peut pas être copiée vers un autre. Le nonce empêche une personne malintentionnée d'enregistrer la réponse du serveur et de l'utiliser pour altérer un appareil ou modifier de quelconque façon le logiciel système.

Secure Enclave

Secure Enclave est un coprocesseur intégré aux processeurs Apple S2, Apple A7 et versions ultérieures de la série A d'Apple. Elle utilise une mémoire chiffrée et intègre un générateur de nombres aléatoires matériel. Secure Enclave assure toutes les opérations cryptographiques pour la gestion des clés de protection des données et préserve l'intégrité de cette dernière même si le noyau a été compromis. La communication entre Secure Enclave et le processeur d'application se limite à une boîte aux lettres déclenchée par interruption et à des tampons de données de mémoire partagée.

Secure Enclave exécute une version adaptée par Apple de la gamme de micronoyaux L4. Secure Enclave utilise sa propre séquence de démarrage sécurisé et peut être mis à jour à l'aide d'un processus de mise à jour logicielle personnalisé indépendant du processeur d'application. Sur les processeurs A9 et ultérieurs de la série A, la puce génère de façon sécurisée un UID (identifiant unique). Cet UID n'est connu ni d'Apple, ni des autres parties du système.

Au démarrage de l'appareil, une clé éphémère est créée, combinée à l'UID, et utilisée pour chiffrer la partie de l'espace mémoire de l'appareil réservée à Secure Enclave. Sauf sur l'Apple A7, la mémoire de Secure Enclave est également authentifiée à l'aide de la clé éphémère.

En outre, les données enregistrées par Secure Enclave sur le système de fichiers sont chiffrées à l'aide d'une clé combinée à l'UID et à un compteur anti-répétition.

Secure Enclave est chargé de traiter les données d'empreintes digitales transmises par le capteur Touch ID, de déterminer s'il y a une correspondance avec l'une des empreintes enregistrées, puis d'autoriser l'accès ou un achat pour le compte de l'utilisateur. La communication entre le processeur et le capteur Touch ID se fait à travers un bus d'interface périphérique série. Le processeur transmet les données à Secure Enclave mais ne peut pas les lire. Elles sont chiffrées et authentifiées avec une clé de session négociée à l'aide de la clé partagée de l'appareil prévue pour le capteur Touch ID et Secure Enclave. L'échange de la clé de session se fait à travers un enveloppement à clé AES, où les deux parties fournissent une clé aléatoire établissant la clé de session et utilisant le chiffrement AES-CCM pour le transport.

Touch ID

Touch ID est le système de lecture d'empreintes digitales qui permet de sécuriser rapidement et facilement l'accès à l'appareil. Cette technologie lit les données d'empreintes digitales sous n'importe quel angle et développe ses connaissances de l'empreinte d'un utilisateur au fil du temps, le capteur continuant d'étendre la carte de l'empreinte à chaque fois qu'un nœud commun supplémentaire est détecté.

Touch ID rend l'utilisation d'un code plus long et complexe beaucoup plus pratique, car les utilisateurs n'ont pas à le saisir aussi souvent. Touch ID élimine également les inconvénients liés à un verrouillage par code, non pas en le remplaçant, mais en permettant d'accéder en toute sécurité à l'appareil dans des limites et des contraintes de temps judicieuses.

Touch ID et les codes

Pour utiliser Touch ID, les utilisateurs doivent configurer leur appareil de sorte qu'un code soit nécessaire pour le déverrouiller. Lorsque Touch ID numérise et reconnaît une empreinte digitale enregistrée, l'appareil se déverrouille sans demander le code. Ce dernier peut toujours être utilisé à la place de Touch ID et reste indispensable dans les situations suivantes :

- L'appareil vient juste d'être allumé ou redémarré.
- L'appareil n'a pas été déverrouillé pendant plus de 48 heures.
- Le code n'a pas été utilisé pour déverrouiller l'appareil au cours des 156 dernières heures (six jours et demi) et Touch ID n'a pas déverrouillé l'appareil dans les quatre dernières heures.
- L'appareil a reçu une commande de verrouillage à distance.
- Après cinq tentatives infructueuses de reconnaissance d'une empreinte digitale.
- Lors de la configuration ou de l'enregistrement de nouvelles empreintes digitales avec Touch ID.

Lorsque Touch ID est activé, l'appareil se verrouille immédiatement lorsque l'utilisateur appuie sur le bouton Marche/Veille. Avec un verrouillage par code uniquement, bon nombre d'utilisateurs définissent une période de grâce de déverrouillage pour éviter d'avoir à saisir un code à chaque utilisation de l'appareil. Avec Touch ID, l'appareil se verrouille à chaque suspension d'activité et nécessite la lecture d'une empreinte digitale, ou éventuellement la saisie du code, à chaque réactivation.

Touch ID peut apprendre à reconnaitre jusqu'à cinq empreintes digitales différentes. Avec une seule empreinte digitale enregistrée, la probabilité de correspondance aléatoire avec une autre personne est de 1 sur 50 000. Toutefois, Touch ID n'autorise que cinq tentatives infructueuses de reconnaissance d'une empreinte digitale avant d'obliger l'utilisateur à saisir un code pour pouvoir accéder à l'appareil.

Autres utilisations de Touch ID

Touch ID peut également être utilisé avec Apple Pay, l'implémentation des paiements sécurisés d'Apple. Pour en savoir plus, consultez la section Apple Pay du présent document.

En outre, les apps tierces peuvent utiliser les API fournies par le système pour demander à l'utilisateur de s'authentifier à l'aide de Touch ID ou d'un code. L'app n'est informée que de la réussite ou de l'échec de l'authentification ; elle ne peut accéder ni à Touch ID, ni aux données associées à l'empreinte digitale enregistrée.

Les éléments du trousseau peuvent également être protégés avec Touch ID ; dans ce cas, Secure Enclave ne permet d'y accéder que si une empreinte est reconnue ou le code de l'appareil est saisi. Les développeurs d'apps ont aussi à leur disposition des API permettant de vérifier que l'utilisateur a défini un code et qu'il peut donc s'authentifier ou déverrouiller les éléments du trousseau à l'aide de Touch ID.

Depuis iOS 9 ou ultérieur, les développeurs peuvent :

- indiquer aux opérations d'API de Touch ID de ne pas obliger à saisir le mot de passe d'une application ou le code d'un appareil ; outre la possibilité de récupérer une représentation de l'état des empreintes digitales enregistrées, cela permet à Touch ID d'être employé comme deuxième facteur dans les apps où la sécurité est particulièrement sensible ;
- générer et utiliser des clés ECC au sein de Secure Enclave ; ces clés peuvent être protégées par Touch ID. Les opérations avec ces clés se font toujours à l'intérieur de Secure Enclave après que ce dernier autorise l'utilisation. Les apps peuvent accéder à ces clés à l'aide du trousseau à travers SecKey. Les SecKeys ne sont que des références aux clés de Secure Enclave ; celles-ci ne quittent jamais Secure Enclave.

Touch ID peut également être configuré pour autoriser des achats dans l'iTunes Store, l'App Store et l'iBooks Store, et ainsi éviter aux utilisateurs d'avoir à saisir le mot de passe de leur Identifiant Apple. Lorsqu'ils choisissent d'autoriser un achat, des jetons d'authentification sont échangés entre l'appareil et la boutique. Le jeton et le nonce cryptographique sont conservés dans Secure Enclave. Le nonce est signé avec une clé Secure Enclave partagée par tous les appareils et l'iTunes Store. Sous iOS 10, les clés ECC de Secure Enclave protégées par Touch ID permettent d'autoriser un achat en signant la requête du Store.

Sécurité de Touch ID

Le lecteur d'empreintes digitales n'est actif que lorsque l'anneau en acier capacitif qui entoure le bouton principal détecte le contact d'un doigt ; la matrice d'imagerie avancée numérise alors l'empreinte et envoie l'image à Secure Enclave.

L'image tramée est stockée temporairement dans la mémoire chiffrée de Secure Enclave le temps d'être vectorisée en vue de son analyse, puis elle est effacée. L'analyse fait appel à une cartographie des angles des crêtes papillaires, un processus avec perte qui élimine les données de minuties nécessaires à la reconstruction de l'empreinte digitale réelle de l'utilisateur. La carte de nœuds ainsi obtenue est stockée sans informations d'identification dans un format chiffré qui ne peut être lu que par Secure Enclave, et n'est jamais envoyée à Apple ni sauvegardée sur iCloud ou iTunes.

Comment Touch ID déverrouille un appareil iOS

Si Touch ID est désactivé, lorsqu'un appareil se verrouille, les clés de la classe de protection des données Complete, qui sont stockées dans Secure Enclave, sont effacées. Les fichiers et les éléments du trousseau appartenant à cette classe restent inaccessibles jusqu'à ce que l'utilisateur déverrouille l'appareil en saisissant son code.

Lorsque Touch ID est activé, les clés ne sont pas effacées lorsque l'appareil se verrouille ; par contre, elles sont enveloppées à l'aide d'une clé attribuée au sous-système Touch ID à l'intérieur de Secure Enclave. Lorsqu'un utilisateur tente de déverrouiller l'appareil, si Touch ID reconnaît son empreinte digitale, il fournit la clé permettant de désenvelopper les clés de protection des données, et l'appareil est déverrouillé. Ce processus apporte une protection supplémentaire en obligeant les sous-systèmes de protection des données et Touch ID à coopérer pour déverrouiller l'appareil.

Les clés dont Touch ID a besoin pour déverrouiller l'appareil sont perdues en cas de redémarrage de celui-ci et sont effacées par Secure Enclave après 48 heures ou cinq tentatives infructueuses de reconnaissance Touch ID.

Chiffrement et protection des données

Effacer contenu et réglages
L'option « Effacer contenu et réglages » de l'app Réglages efface toutes les clés présentes dans l'espace de stockage effaçable, rendant ainsi cryptographiquement inaccessibles toutes les données d'utilisateur sur l'appareil. Il s'agit donc d'un moyen idéal pour s'assurer que toutes les données personnelles sont supprimées d'un appareil avant de le transmettre à quelqu'un d'autre ou de le faire réparer. Important : n'utilisez jamais l'option « Effacer contenu et réglages » avant d'avoir effectué une sauvegarde de l'appareil, car il n'existe aucun moyen de récupérer les données effacées.

La chaîne de démarrage sécurisé, la signature du code et la sécurité des processus exécutés permettent de garantir que seuls le code et les apps fiables peuvent s'exécuter sur un appareil. iOS offre des fonctionnalités de chiffrement et de protection des données supplémentaires pour protéger les données utilisateur, même lorsque d'autres parties de l'infrastructure de sécurité ont été compromises (par exemple, sur un appareil sur lequel des modifications non autorisées ont été apportées). Cela apporte des avantages importants aussi bien pour les utilisateurs que pour les administrateurs informatiques qui sont ainsi assurés que les informations personnelles et d'entreprise sont protégées à tout moment et disposent de méthodes d'effacement instantané et complet à distance en cas de vol ou de perte de l'appareil.

Fonctionnalités de sécurité matérielles

Sur les appareils mobiles, la vitesse et l'efficacité énergétique sont essentielles. Les opérations cryptographiques sont complexes et peuvent engendrer des problèmes de performances ou d'autonomie de la batterie si elles ne sont pas conçues et implémentées en gardant ces priorités à l'esprit.

Chaque appareil iOS est doté d'un moteur de chiffrement AES 256 dédié intégré dans le chemin DMA entre le stockage Flash et la mémoire principale du système, ce qui rend le chiffrement des fichiers extrêmement efficace. Sur les processeurs A9 ou ultérieurs de la série A, le sous-système de stockage flash se trouve sur le bus isolé dont l'accès n'est accordé qu'à la mémoire contenant les données utilisateur à travers le moteur de chiffrement DMA.

L'identifiant unique (UID) et l'identifiant de groupe (GID) de l'appareil sont des clés AES 256 bits fusionnées (UID) ou compilées (GID) dans le processeur d'application et Secure Enclave lors de la fabrication. Aucun logiciel ni programme interne ne peut les lire directement ; ils ne peuvent voir que les résultats des opérations de chiffrement ou de déchiffrement réalisées par les moteurs AES dédiés implémentés dans le silicium à l'aide de l'UID ou du GID comme clé. En outre, l'UID et le GID de Secure Enclave ne peuvent être utilisés que par le moteur AES dédié à Secure Enclave. Les UID sont propres à chaque appareil et ne sont pas enregistrés ni par Apple, ni aucun de ses fournisseurs. Les GID sont communs à tous les processeurs d'une même classe d'appareils (par exemple, tous les appareils dotés du processeur A8 d'Apple) et sont utilisés pour les tâches ne présentant pas un risque majeur pour la sécurité, comme lors de la distribution du logiciel système pendant l'installation et la restauration. L'intégration de ces clés dans le silicium permet d'empêcher leur altération ou leur contournement et les rend inaccessibles hors du moteur AES. Les UID et les GID ne sont pas non plus accessibles via JTAG ou d'autres interfaces de débogage.

L'UID permet de lier cryptographiquement des données à un appareil précis. Par exemple, la hiérarchie des clés protégeant le système de fichiers inclut l'UID ; donc, si les puces de mémoire sont transférées d'un appareil à un autre, les fichiers sont inaccessibles. L'UID n'est lié à aucun autre identifiant sur l'appareil.

À part l'UID et le GID, toutes les autres clés cryptographiques sont créées par le générateur de nombres aléatoires (RNG, Random Number Generator) du système à l'aide d'un algorithme basé sur CTR_DRBG. L'entropie du système est générée à partir de variations de synchronisation lors du démarrage, et à partir d'une synchronisation

par interruption une fois l'appareil démarré. Les clés générées à l'intérieur de la puce Secure Enclave utilisent son générateur de nombres aléatoires matériel s'appuyant sur plusieurs oscillateurs en anneau post-traités par CTR_DRBG.

L'effacement sécurisé des clés enregistrées est tout aussi important que leur génération. Cela s'avère particulièrement délicat dans le stockage Flash, où le contrôle d'usure peut nécessiter l'effacement de plusieurs copies des données. Pour traiter ce problème, les appareils iOS intègrent une fonctionnalité dédiée à l'effacement sécurisé des données appelée Stockage effaçable. Cette fonctionnalité accède à la technologie de stockage sous-jacente (par exemple, NAND) pour effacer directement un petit nombre de blocs à un niveau très bas.

Protection des données des fichiers

En plus des fonctionnalités de chiffrement matérielles intégrées aux appareils iOS, Apple utilise une technologie appelée Protection des données pour accroître la protection des données stockées dans la mémoire Flash de l'appareil. La protection des données permet à l'appareil de répondre à des événements courants comme les appels téléphoniques entrants, tout en assurant un niveau de chiffrement élevé des données utilisateur. Les apps clés du système, comme Messages, Mail, Calendrier, Contacts, Photos et les données Santé, utilisent par défaut la protection des données, et les apps tierces installées sur iOS 7 ou ultérieur bénéficient automatiquement de cette protection.

La protection des données est implémentée en élaborant et en gérant une hiérarchie de clés, et repose sur les technologies de chiffrement matérielles intégrées à chaque appareil iOS. La protection des données est contrôlée fichier par fichier en attribuant une classe à chacun d'eux ; l'accessibilité est déterminée par le déverrouillage des clés de classe.

Vue d'ensemble de l'architecture

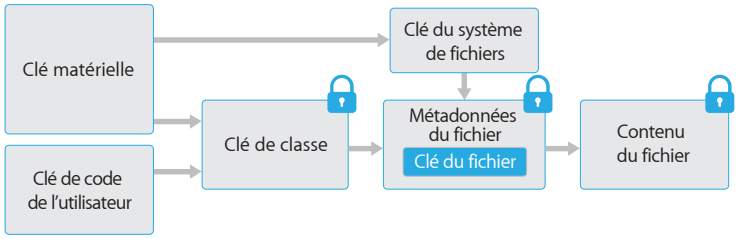
Chaque fois qu'un fichier est créé sur la partition de données, la protection des données crée une nouvelle clé 256 bits (la clé « par fichier ») et la transmet au moteur AES matériel, qui l'utilise alors pour chiffrer le fichier lors de son écriture dans la mémoire Flash en utilisant le mode AES CBC. (Sur les appareils dotés d'un processeur A8, le chiffrement AES-XTS est utilisé.) Le vecteur d'initialisation (IV, Initialization Vector) est calculé avec le décalage de bloc du fichier, chiffré avec le hachage SHA-1 de la clé par fichier.

La clé par fichier est enveloppée à l'aide d'une des clés de classe, selon les circonstances dans lesquelles le fichier doit être accessible. Comme tous les autres enveloppements, celui-ci est réalisé à l'aide de l'enveloppement à clé AES NIST, selon RFC 3394. La clé par fichier enveloppée est stockée dans les métadonnées du fichier.

Lorsqu'un fichier est ouvert, ses métadonnées sont déchiffrées à l'aide de la clé du système de fichiers, ce qui révèle la clé par fichier enveloppée ainsi que la classe qui la protège. La clé par fichier est désenveloppée à l'aide de la clé de classe, puis transmise au moteur AES matériel, qui déchiffre le fichier lors de sa lecture depuis la mémoire Flash. Toute la gestion des clés de fichier enveloppées se produit dans Secure Enclave ; la clé de fichier n'est jamais directement exposée au processeur d'application. Au démarrage, Secure Enclave négocie une clé éphémère avec le moteur AES. Lorsque Secure Enclave rétablit le format des clés d'un fichier, elle utilise la clé éphémère puis renvoie les clés au processeur d'application.

Les métadonnées de tous les fichiers présents dans le système de fichiers sont chiffrées avec une clé aléatoire qui est créée lors de l'installation initiale d'iOS ou lors de l'effacement du contenu de l'appareil par l'utilisateur. La clé du système de fichiers est conservée dans le stockage effaçable. Comme elle est stockée sur l'appareil, cette clé n'est pas utilisée pour préserver la confidentialité des données ; par contre, elle est conçue pour

être effacée rapidement à la demande (de l'utilisateur, à l'aide de l'option « Effacer données et réglages », ou d'un utilisateur ou administrateur par le biais d'une commande d'effacement à distance depuis un serveur de gestion des appareils mobiles, Exchange ActiveSync ou iCloud). Effacer la clé de cette manière rend impossible le déchiffrement des fichiers.



Le contenu d'un fichier est chiffré avec une clé par fichier, qui est enveloppée avec une clé de classe et stockée dans les métadonnées du fichier, qui sont à leur tour chiffrées avec la clé du système de fichiers. La clé de classe est protégée par l'UID du matériel et, pour certaines classes, le code de l'utilisateur. Cette hiérarchie offre à la fois souplesse et performances. Par exemple, changer la classe d'un fichier peut se faire simplement en réenveloppant sa clé par fichier, et la modification du code ne réenveloppe que la clé de classe.

Codes

En configurant un code d'appareil, l'utilisateur active automatiquement la protection des données. iOS prend en charge les codes à six chiffres, à quatre chiffres et alphanumériques de longueur arbitraire. En plus de déverrouiller l'appareil, un code fournit l'entropie pour certaines clés de chiffrement. Cela signifie qu'une personne malintentionnée en possession d'un appareil ne peut pas accéder aux données appartenant à des classes de protection spécifiques sans le code.

Remarques à propos du code
En cas de saisie d'un code de longueur importante ne contenant que des chiffres, un pavé numérique est affiché sur l'écran de verrouillage au lieu du clavier complet. Un code numérique de longueur importante peut être plus facile à saisir qu'un code alphanumérique court, tout en fournissant un niveau de sécurité identique.

Le code étant combiné à l'UID de l'appareil, des attaques en force sont nécessaires pour tenter d'accéder à celui-ci. Un grand nombre d'itérations est utilisé pour ralentir chaque tentative. Ce nombre d'itérations est étalonné de sorte qu'une tentative prenne environ 80 millisecondes. Cela signifie qu'il faudrait plus de 5 ans et demi pour essayer toutes les combinaisons d'un code alphanumérique à six caractères, composé de lettres minuscules et de chiffres.

Plus le code de l'utilisateur est complexe, plus la clé de chiffrement l'est également. Touch ID peut être utilisé pour renforcer cette équation en permettant à l'utilisateur de définir un code beaucoup plus compliqué qu'il ne le ferait en temps normal pour des raisons pratiques. Cela augmente le degré réel d'entropie protégeant les clés de chiffrement utilisées pour la protection des données sans avoir d'impact négatif sur l'expérience de l'utilisateur qui déverrouille son appareil iOS plusieurs fois par jour.

Délais entre tentatives de code

Tentatives	Délai imposé
1 à 4	aucun
5	1 minute
6	5 minutes
7 à 8	15 minutes
9	1 heure

Pour compliquer encore plus les attaques en force, des délais de plus en plus longs sont prévus après la saisie d'un code non valide sur l'écran de verrouillage. Si l'option Réglages > Touch ID et code > Effacer les données est activée, l'appareil efface automatiquement les données après 10 tentatives infructueuses consécutives de saisie du code. Ce réglage est également disponible sous forme de règle administrative via le serveur de gestion des appareils mobiles (MDM) et Exchange ActiveSync, et son seuil peut être défini sur une valeur plus basse.

Sur les appareils dotés de Secure Enclave, les délais sont imposés par ce dernier coprocesseur. Si l'appareil est redémarré au cours du décompte d'un délai, ce dernier reste imposé, la minuterie reprenant son cours.

Classes de protection des données

Lorsqu'un nouveau fichier est créé sur un appareil iOS, l'app qui le crée lui attribue une classe. Chaque classe utilise des règles différentes pour déterminer quand les données sont accessibles. Les classes et règles de base sont décrites dans les sections qui suivent.

Complete Protection

(`NSFileProtectionComplete`) : la clé de classe est protégée par une clé obtenue à partir du code de l'utilisateur et de l'UID de l'appareil. Peu après que l'utilisateur bloque son appareil (10 secondes si le réglage « Exiger le mot de passe » est défini sur Immédiatement), la clé de classe déchiffrée est abandonnée, rendant ainsi l'intégralité des données de la classe inaccessibles jusqu'à ce que l'utilisateur ressaisisse le code ou déverrouille l'appareil via la fonction Touch ID.

Protected Unless Open

(`NSFileProtectionCompleteUnlessOpen`) : il peut arriver que des fichiers doivent être écrits pendant le verrouillage de l'appareil. Cela est le cas, par exemple, lors du téléchargement d'une pièce jointe en arrière-plan. Ce comportement peut être obtenu grâce à la cryptographie asymétrique à courbes elliptiques (ECDH sur Curve25519). La clé par fichier habituelle est protégée par une clé obtenue par accord de clé Diffie-Hellman à une passe, comme décrit dans la norme NIST SP 800-56A.

La clé publique éphémère de l'accord est stockée avec la clé par fichier enveloppée. KDF est une fonction de dérivation de clé de concaténation (alternative agréée 1) comme décrit dans la section 5.8.1 de la norme NIST SP 800-56A. AlgorithmID est omis. PartyUInfo et PartyVInfo correspondent respectivement aux clés publiques éphémère et statique. SHA-256 est utilisée comme fonction de hachage. À la fermeture du fichier, la clé par fichier est effacée de la mémoire. Pour rouvrir le fichier, le secret partagé est recréé à l'aide de la clé privée de la classe Protected Unless Open et de la clé publique éphémère du fichier, lesquelles servent à désenvelopper la clé pour chaque fichier permettant de déchiffrer le fichier.

Protected Until First User Authentication

(`NSFileProtectionCompleteUntilFirstUserAuthentication`) : cette classe se comporte comme la classe Complete Protection, sauf que la clé de classe déchiffrée n'est pas supprimée de la mémoire lorsque l'appareil est verrouillé. Cette classe présente des propriétés de protection similaires à celles du chiffrement de volume complet de l'environnement de bureau et protège les données des attaques impliquant un redémarrage. Il s'agit de la classe par défaut pour toutes les données d'apps tierces auxquelles aucune classe de protection des données n'est spécifiquement affectée.

No Protection

(`NSFileProtectionNone`) : cette clé de classe n'est protégée que par l'UID et est conservée dans le stockage effaçable. Comme toutes les clés nécessaires au déchiffrement des fichiers appartenant à cette classe sont stockées sur l'appareil, le chiffrement n'apporte comme avantage que la possibilité d'effacement à distance rapide. Si aucune classe de protection des données n'est affectée à un fichier, celui-ci est tout de même stocké sous forme chiffrée (comme toutes les données d'un appareil iOS).

Clé de la classe de protection des données

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

Composants d'un élément de trousseau

Chaque élément de trousseau contient, en plus du groupe d'accès, des métadonnées administratives (comme des codes temporels de date de création et de date de dernière modification).

Il contient également des hachages SHA-1 des attributs utilisés pour demander l'élément (tels que le nom du compte et le nom du serveur) afin de permettre la recherche sans devoir déchiffrer chaque élément. Enfin, il contient les données de chiffrement qui incluent les éléments suivants :

- numéro de version ;
- données de liste de contrôle d'accès (ACL) ;
- valeur indiquant la classe de protection à laquelle appartient l'élément ;
- clé d'élément enveloppée avec la clé de classe de protection ;
- dictionnaire d'attributs décrivant l'élément (tel que transmis à SecItemAdd), codé en tant que fichier plist binaire et chiffré avec la clé d'élément.

Le chiffrement est de type AES 128 en mode GCM (Galois/Counter) ; le groupe d'accès est inclus dans les attributs et protégé par la balise GMAC calculée pendant le chiffrement.

Protection des données du trousseau

De nombreuses apps doivent traiter des mots de passe et d'autres petits fragments de données confidentielles, comme des clés et des jetons de session. Le trousseau iOS offre un moyen sûr de stocker ces éléments.

Le trousseau est implémenté sous la forme d'une base de données SQLite stockée sur le système de fichiers. Il n'existe qu'une seule base de données ; le daemon securityd détermine à quels éléments du trousseau peut accéder chaque processus ou app. Les API d'accès au trousseau envoient des appels au daemon, lequel interroge les droits « groupes d'accès au trousseau », « identifiant d'application » et « groupe d'application » de l'app. Au lieu de limiter l'accès à un seul processus, les groupes d'accès permettent de partager les éléments du trousseau entre les apps.

Les éléments du trousseau ne peuvent être partagés qu'entre les apps du même développeur. Cette restriction est implémentée en obligeant les apps tierces à utiliser des groupes d'accès portant un préfixe qui leur est alloué par le biais du programme Apple Developer Program via les groupes d'applications. L'obligation d'utiliser un préfixe et le caractère unique du groupe d'applications sont contrôlés par la signature du code, les profils d'approvisionnement et le programme Apple Developer Program.

Les données du trousseau sont protégées à l'aide d'une structure de classes similaire à celle utilisée pour la protection des données des fichiers. Ces classes présentent des comportements équivalents aux classes de protection des données des fichiers, mais les clés qu'elles utilisent sont différentes, tout comme les noms des API auxquelles elles sont intégrées.

Disponibilité	Protection des données des fichiers	Protection des données du trousseau
Lorsque l'appareil est déverrouillé	NSFileProtectionComplete	kSecAttrAccessibleWhenUnlocked
Lorsque l'appareil est verrouillé	NSFileProtectionCompleteUnlessOpen	N/D
Protected Until First User Authentication	NSFileProtectionCompleteUntilFirstUserAuthentication	kSecAttrAccessibleAfterFirstUnlock
Toujours	NSFileProtectionNone	kSecAttrAccessibleAlways
Code activé	N/D	kSecAttrAccessible WhenPasscodeSetThisDeviceOnly

Les apps qui font appel à des services d'actualisation en arrière-plan peuvent utiliser la classe `kSecAttrAccessibleAfterFirstUnlock` pour les éléments du trousseau qui doivent être accessibles lors des mises à jour en arrière-plan.

La classe `kSecAttrAccessibleWhenPasscodeSetThisDeviceOnly` se comporte comme la classe `kSecAttrAccessibleWhenUnlocked`, mais n'est disponible que si un code est configuré pour l'appareil. Cette classe n'existe que dans le conteneur de clés du système ; ils ne sont ni synchronisés avec le trousseau iCloud, ni sauvegardés, ni inclus dans les conteneurs de clés de dépôt. Si le code est supprimé ou réinitialisé, les éléments sont rendus inutilisables par effacement des clés de classe.

D'autres classes du trousseau ont un pendant « Cet appareil uniquement », qui est toujours protégé par l'UID quand il est copié depuis l'appareil lors d'une sauvegarde, ce qui le rend inutilisable s'il est restauré sur un autre appareil.

Apple a pris soin de trouver un juste équilibre entre sécurité et facilité d'utilisation en choisissant les classes du trousseau qui dépendent du type d'informations sécurisées et en définissant quand cela est requis par iOS. Par exemple, un certificat VPN doit toujours être disponible pour que l'appareil puisse rester connecté en permanence, mais il est classé comme élément « non itinérant » et ne peut donc pas être transféré vers un autre appareil.

Pour les éléments du trousseau créés par iOS, les protections de classe suivantes sont appliquées :

Élément	Accessible
Mots de passe Wi-Fi	Protected Until First User Authentication
Comptes Mail	Protected Until First User Authentication
Comptes Exchange	Protected Until First User Authentication
Mots de passe VPN	Protected Until First User Authentication
LDAP, CalDAV, CardDAV	Protected Until First User Authentication
Jetons des comptes de réseau social	Protected Until First User Authentication
Clés de chiffrement des annonces Handoff	Protected Until First User Authentication
Jeton iCloud	Protected Until First User Authentication
Mot de passe de partage à domicile	Lorsque l'appareil est déverrouillé
Jeton Localiser mon iPhone	Toujours
Messagerie	Toujours
Sauvegarde iTunes	Lorsque l'appareil est déverrouillé, non itinérant
Mots de passe Safari	Lorsque l'appareil est déverrouillé
Signets Safari	Lorsque l'appareil est déverrouillé
Certificats VPN	Toujours, non itinérant
Clés Bluetooth®	Toujours, non itinérant
Jeton du service Apple Push Notification	Toujours, non itinérant
Certificats et clé privée iCloud	Toujours, non itinérant
Clés iMessage	Toujours, non itinérant
Certificats et clés privées installés par le profil de configuration	Toujours, non itinérant
Code PIN de la carte SIM	Toujours, non itinérant

Contrôle de l'accès au trousseau

Les trousseaux peuvent utiliser des listes de contrôle d'accès (ACL, Access Control List) pour définir des règles précisant les conditions d'accessibilité et d'authentification. Les éléments peuvent établir des conditions nécessitant la présence de l'utilisateur en spécifiant qu'ils ne sont accessibles que si celui-ci s'authentifie à l'aide de Touch ID ou en saisissant le code de l'appareil. L'accès aux éléments peut également être limité en indiquant que l'inscription Touch ID n'a pas changé depuis l'ajout de l'élément. Cette limite contribue à empêcher une personne malintentionnée d'ajouter sa propre empreinte digitale dans le but d'accéder à un élément du trousseau. Les ACL sont évaluées à l'intérieur de Secure Enclave et ne sont transmises au noyau que si les conditions définies sont remplies.

Accès aux mots de passe enregistrés par Safari

Les apps iOS peuvent interagir avec les éléments du trousseau enregistrés par Safari pour le remplissage automatique des mots de passe à l'aide des deux API suivantes :

- `SecRequestSharedWebCredential`
- `SecAddSharedWebCredential`

L'accès n'est accordé que si le développeur de l'app et l'administrateur du site web donnent tous deux leur approbation, et si l'utilisateur donne son accord. Les développeurs d'apps expriment leur intention d'accéder aux mots de passe enregistrés par Safari en incluant un droit dans leur app. Ce droit répertorie les noms de domaine complets des sites web associés. Les administrateurs des sites web doivent placer sur leur serveur un fichier contenant la liste des identifiants uniques des apps qu'ils ont approuvées. Lorsqu'une app avec le droit com.apple.developer.associated-domains est installée, iOS envoie une requête TLS à chaque site web répertorié pour demander le fichier /apple-app-site-association. Si le fichier fait état de l'identifiant de l'app en cours d'installation, iOS marque alors le site web et l'app comme ayant une relation de confiance. L'établissement d'une relation de confiance est nécessaire pour que les appels à ces deux API entraînent la présentation d'une invite à l'utilisateur, qui doit alors donner son accord avant qu'un mot de passe ne soit transmis à l'app, mis à jour ou supprimé.

Conteneurs de clés

Les clés des classes de protection des données, pour les fichiers et le trousseau, sont rassemblées et gérées dans des conteneurs de clés. iOS utilise les conteneurs de clés suivants : utilisateur, appareil, sauvegarde, dépôt et sauvegarde iCloud.

Le conteneur de clés de l'utilisateur est l'endroit où les clés de classe enveloppées utilisées lors du fonctionnement normal de l'appareil sont stockées. Par exemple, lorsqu'un code est saisi, la clé `NSFileProtectionComplete` est chargée depuis le conteneur de clés du système et désenveloppée. Il s'agit d'un fichier plist binaire appartenant à la classe No Protection, mais dont le contenu est chiffré à l'aide d'une clé conservée dans le stockage effaçable. Afin d'assurer la sécurité à terme des conteneurs de clés, cette clé est effacée et régénérée chaque fois qu'un utilisateur modifie son code. L'extension du noyau AppleKeyStore gère le conteneur de clés de l'utilisateur et peut être interrogée sur l'état de verrouillage d'un appareil. Elle signale que l'appareil est déverrouillé uniquement si toutes les clés de classe du conteneur de clés de l'utilisateur sont accessibles et qu'elles sont correctement désenveloppées.

Le conteneur de clés de l'appareil sert à stocker les clés de classe enveloppées utilisées pour les opérations faisant appel à des données spécifiques à l'appareil. Les appareils iOS configurés pour un usage partagé ont parfois besoin d'accéder à des informations d'identification pour qu'un utilisateur puisse ouvrir sa session. Dès lors, un conteneur de clés qui n'est pas protégé par le code de l'utilisateur devient obligatoire. iOS ne prend pas en charge la distance cryptographique du contenu du système de fichiers propre à l'utilisateur, ce qui signifie que le système utilise les clés de classe tirées du conteneur de clés de l'appareil pour envelopper les clés pour chaque fichier. Le trousseau, cependant, fait appel à des clés de classe issues du conteneur de clés de l'utilisateur pour protéger les éléments inclus dans le trousseau de l'utilisateur. Sur les appareils iOS configurés pour un usage par un seul utilisateur (configuration par défaut), le conteneur de clés de l'appareil et celui de l'utilisateur sont un seul et même composant, protégé par le code de l'utilisateur.

Le conteneur de clés de sauvegarde est créé lorsqu'iTunes réalise une sauvegarde chiffrée et la stocke sur l'ordinateur sur lequel le contenu de l'appareil est sauvegardé. Un nouveau conteneur de clés est créé avec un nouveau jeu de clés, et les données sauvegardées sont rechiffrées avec ces nouvelles clés. Comme expliqué précédemment, les éléments du trousseau non itinérants restent enveloppés avec la clé dérivée de l'UID, ce qui permet de les restaurer sur l'appareil à partir duquel ils ont été initialement sauvegardés, mais les rend inaccessibles sur un autre appareil.

Le conteneur de clés est protégé par le mot de passe défini dans iTunes, soumis à 10 millions d'itérations de PBKDF2. Malgré ce grand nombre d'itérations, le conteneur de clés de sauvegarde n'est lié à aucun appareil précis et peut donc théoriquement faire l'objet d'une tentative d'attaque en force exécutée en parallèle sur plusieurs ordinateurs. Cette menace peut être atténuée en utilisant un mot de passe suffisamment complexe.

Si un utilisateur choisit de ne pas chiffrer une sauvegarde iTunes, les fichiers de sauvegarde ne sont pas chiffrés quelle que soit la classe de protection des données à laquelle ils appartiennent, mais le trousseau reste protégé par une clé dérivée de l'UID. C'est pourquoi les éléments du trousseau ne peuvent être transférés vers un nouvel appareil que si un mot de passe de sauvegarde est défini.

Le conteneur de clés de dépôt est utilisé pour la synchronisation iTunes et par les serveurs MDM. Ce conteneur de clés permet à iTunes de réaliser des sauvegardes et des synchronisations sans nécessiter la saisie d'un code par l'utilisateur, et à un serveur MDM d'effacer à distance le code d'un utilisateur. Il est stocké sur l'ordinateur utilisé pour effectuer la synchronisation avec iTunes, ou sur le serveur MDM qui gère l'appareil.

Le conteneur de clés de dépôt améliore l'expérience de l'utilisateur lors de la synchronisation de l'appareil, qui peut nécessiter l'accès à toutes les classes de données. Lors de la première connexion à iTunes d'un appareil verrouillé à l'aide d'un code, l'utilisateur est invité à saisir ce dernier. L'appareil crée ensuite un conteneur de clés de dépôt contenant les mêmes clés de classe que celles qu'il utilise et génère une nouvelle clé pour le protéger. Le conteneur de clés de dépôt et la clé qui le protège sont répartis entre l'appareil et l'hôte ou le serveur, les données stockées sur l'appareil étant affectées à la classe Protected Until First User Authentication. C'est pourquoi le code de l'appareil doit être saisi la première fois que l'utilisateur réalise une sauvegarde avec iTunes après un redémarrage.

Dans le cas d'une mise à jour logicielle en mode OTA, l'utilisateur est invité à saisir son code au lancement de la mise à jour. Cette technique sert à créer de façon sécurisée un jeton de déverrouillage à usage unique qui déverrouille le conteneur de clés de l'utilisateur après la mise à jour. Ce jeton ne peut pas être généré sans saisir le code de l'utilisateur, et tout jeton précédemment généré est invalidé si le code de l'utilisateur a changé entre temps.

Les jetons de déverrouillage à usage unique sont prévus aussi bien pour l'installation surveillée que pour celle sans surveillance d'une mise à jour logicielle. Ils sont chiffrés à l'aide d'une clé dérivée de la valeur active d'un compteur monotone dans Secure Enclave, de l'UUID du conteneur de clés et de l'UID de Secure Enclave.

L'incrémentation du compteur de jetons de déverrouillage à usage unique dans Secure Enclave invalide tout jeton existant. Le compteur est incrémenté lorsqu'un jeton est utilisé, après l'authentification initiale d'un appareil redémarré, lorsqu'une mise à jour logicielle est annulée (par l'utilisateur ou par le système) ou quand la minuterie du règlement d'un jeton a expiré.

Le jeton de déverrouillage à usage unique pour les mises à jour surveillées de logiciel expire au bout de 20 minutes. Ce jeton est exporté depuis Secure Enclave et est écrit sur un espace de stockage effaçable. La minuterie d'un règlement incrémente le compteur si l'appareil n'a pas redémarré dans les 20 minutes.

Pour les mises à jour logicielles sans surveillance, définies par l'utilisateur s'il choisit l'option « Installer plus tard » lorsqu'il est informé de la mise à jour, le processeur d'application peut conserver actif le jeton de déverrouillage à usage unique dans Secure Enclave jusqu'à 8 heures. Passé ce délai, une minuterie de règlement incrémente le compteur.

Le conteneur de clés de sauvegarde iCloud est similaire au conteneur de clés de sauvegarde. Toutes les clés de classe présentes dans ce conteneur de clés étant asymétriques (utilisation de Curve25519, comme la classe Protected Unless Open Data Protection), les sauvegardes iCloud peuvent être réalisées en arrière-plan. Pour toutes les classes de protection des données, à l'exception de No Protection, les données chiffrées sont lues sur l'appareil et envoyées à iCloud. Les clés de classe correspondantes sont protégées par des clés iCloud. Les clés de classe du trousseau sont enveloppées avec une clé dérivée de l'UID comme lors d'une sauvegarde iTunes non chiffrée. Un conteneur de clés asymétriques est également utilisé pour la sauvegarde dans la fonctionnalité de récupération du trousseau iCloud.

Certificats et programmes de sécurité

Remarque : pour obtenir les dernières informations sur les certifications de sécurité, les procédures de validation et les instructions touchant iOS, consultez https://support.apple.com/fr-fr/HT202739.

Certification ISO 27001

Apple a obtenu la certification ISO 27001 pour le système de gestion de sécurité des informations pour l'infrastructure, le développement et les opérations prenant en charge les produits et services : Apple School Manager, iCloud, iMessage, FaceTime, les identifiants Apple gérés et iTunes U, conformément à la Statement of Applicability v1.0 en date du 26 février 2016. La conformité d'Apple avec la norme ISO a été certifié par la British Standards Institution. Pour afficher ce certificat, consultez la page https://www.bsigroup.com/fr-FR/our-services/certification/certificate-and-client-directory/search-results/?searchkey=company=apple&licencenumber=IS+649475.

Validation cryptographique (FIPS 140-2)

La conformité des modules cryptographiques d'iOS a été validée selon la norme FIPS (Federal Information Processing Standards) 140-2 niveau 1 des États-Unis à chaque nouvelle version depuis iOS 6. Les modules cryptographiques d'iOS 10 sont identiques à ceux d'iOS 9 et d'iOS 8, mais comme pour chaque version, Apple procède à leur revalidation. Ce programme confirme l'intégrité des opérations cryptographiques pour les apps Apple et les apps de tierce partie qui exploitent correctement les services cryptographiques d'iOS.

Certification des critères communs (ISO 15408)

Apple a obtenu les certifications iOS dans le cadre du programme CCC (certification des critères communs). Les trois premières certifications obtenues sont la VID10695 pour iOS 9 qui touche au profil MDFPP2 (version 2.0 du profil de protection fondamentale des appareils portables), la VID10714 qui traite du profil VPNIPSecPP1.4 (profil de protection de client IPSecPP1.4 VPN) et la VID10725 pour le MDMAgentEP2 (version 2.0 du module étendu pour les agents de gestion des appareils portables). Apple explore déjà des voies pour ces mêmes certifications sur iOS 10 et compte poursuivre pour chaque version future d'iOS. Apple joue un rôle actif au sein de la communauté technique internationale (ITC) dans le développement de profils de protection collaborative (cPP) actuellement indisponibles se concentrant sur l'évaluation des technologies de sécurité mobile des clés. Apple continue d'évaluer et poursuit les certifications visant l'actualisation des PP disponibles à ce jour.

Solutions commerciales pour composants classifiés (CSfC)

Le cas échéant, Apple a également soumis la plateforme iOS et différents services pour leur ajout dans la liste des composants du programme des solutions commerciales pour composants classifiés (CSfC). Il s'agit plus précisément d'iOS 9 pour la plateforme mobile, du client IKEv2 pour le client IPSec VPN (IKEv2 VPN permanent uniquement) et de l'agent MDMAgentEP2. Dans la mesure où les plateformes et les services Apple font l'objet de certifications inhérentes aux critères communs, ils seront également soumis pour leur ajout dans la liste des composants du programme CSfC.

Guides de configuration de sécurité

Apple a collaboré avec les gouvernements du monde entier pour développer des guides donnant les instructions et recommandations nécessaires pour le maintien d'un environnement plus sécurisé, également appelé « durcissement des appareils ». Ces guides fournissent des informations bien définies et approfondies sur la configuration et l'utilisation de fonctionnalités d'iOS pour une protection améliorée.

Sécurité des apps

Les apps sont parmi les éléments les plus critiques d'une architecture de sécurité mobile moderne. Bien qu'elles apportent d'incroyables avantages aux utilisateurs en termes de productivité, elles sont aussi susceptibles d'avoir un impact négatif sur la sécurité du système, sa stabilité et les données utilisateur si elles ne sont pas correctement gérées.

C'est pourquoi iOS est doté de couches de protection chargées de s'assurer que les apps sont signées et vérifiées, et de les placer en environnement contrôlé pour protéger les données utilisateur. Ces éléments fournissent une plateforme stable et sécurisée pour les apps, ce qui permet à des milliers de développeurs de proposer des centaines de milliers d'apps sur iOS sans incidence sur l'intégrité du système. Et les utilisateurs peuvent accéder à ces apps sur leur appareil iOS sans craindre outre mesure les virus, les logiciels malveillants ou autres attaques non autorisées.

Signature du code des apps

Après son démarrage, le noyau iOS contrôle les apps et processus utilisateur autorisés à s'exécuter. Pour garantir que toutes les apps proviennent d'une source connue et approuvée et qu'elles n'ont pas été altérées, iOS exige que l'ensemble du code exécutable soit signé à l'aide d'un certificat émis par Apple. Les apps fournies avec l'appareil, comme Mail et Safari, sont signées par Apple. Les apps de tierce partie doivent également être validées et signées à l'aide d'un certificat émis par Apple. La signature obligatoire du code étend le concept de chaîne de confiance du système d'exploitation aux apps et empêche les apps de tierce partie de charger du code non signé ou d'utiliser du code automodifiant.

Pour pouvoir développer et installer des apps sur des appareils iOS, les développeurs doivent s'enregistrer auprès d'Apple et adhérer au programme Apple Developer Program. L'identité réelle de chaque développeur, qu'il s'agisse d'un particulier ou d'une entreprise, est vérifiée par Apple avant l'émission de son certificat. Ce certificat permet aux développeurs de signer des apps et de les soumettre à l'App Store en vue de leur distribution. Toutes les apps disponibles dans l'App Store sont donc proposées par une personne ou une entreprise identifiable, ce qui dissuade de créer des apps malveillantes. Elles sont également vérifiées par Apple afin de s'assurer qu'elles fonctionnent comme décrit et ne présentent pas de bogues évidents ou d'autres problèmes. En plus des technologies déjà abordées, ce processus de curation garantit aux clients la qualité des apps qu'ils achètent.

iOS permet aux développeurs d'incorporer des cadres d'application dans leurs apps ; ceux-ci peuvent être utilisés par l'app elle-même ou par des extensions intégrées à celle-ci. Pour empêcher le système et les autres apps de charger du code tiers dans leur espace d'adresse, le système procède à la validation de la signature du code de toutes les bibliothèques dynamiques auxquelles un processus se lie lors de son lancement. Cette vérification est réalisée par le biais de l'identifiant d'équipe (Team ID) issu d'un certificat émis par Apple. Un identifiant d'équipe est une chaîne alphanumérique comportant 10 caractères ; par exemple, 1A2B3C4D5F. Un programme peut se lier à n'importe quelle bibliothèque fournie avec le système ou à n'importe quelle bibliothèque comportant dans la signature de son code le même identifiant d'équipe que l'exécutable principal. Comme les exécutables préinstallés sur le système ne possèdent pas d'identifiant d'équipe, ils ne peuvent se lier qu'aux bibliothèques fournies avec le système.

Les entreprises ont également la possibilité de développer des apps réservées à un usage interne et de les distribuer à leurs employés. Les entreprises et les organismes peuvent déposer une candidature au programme Apple Developer Enterprise Program (ADEP) avec un numéro D-U-N-S. Apple accepte les candidats après vérification de leur identité et de leur admissibilité. Une fois qu'un organisme est membre de l'ADEP, il peut s'enregistrer pour obtenir un profil d'approvisionnement permettant d'exécuter des apps internes sur des appareils autorisés. Les utilisateurs doivent installer le profil d'approvisionnement pour pouvoir exécuter les apps internes. Cela garantit que seuls les utilisateurs prévus de l'organisme peuvent charger les apps sur leurs appareils iOS. Les apps installées via MDM sont considérées implicitement comme fiables, car la relation entre l'entreprise et l'appareil est déjà établie. À défaut, les utilisateurs doivent approuver le profil d'approvisionnement de l'app dans Réglages. Les entreprises peuvent empêcher les utilisateurs d'approuver des apps issues de développeurs inconnus. Au premier lancement d'une app d'entreprise quelconque, l'appareil doit recevoir la confirmation d'Apple que l'app est autorisée à s'exécuter.

Contrairement aux autres plates-formes mobiles, iOS n'autorise les utilisateurs ni à installer des apps non signées potentiellement malveillantes depuis des sites web, ni à exécuter du code non approuvé. Lors de l'exécution, des contrôles de signature du code des pages mémoire de tous les exécutables sont réalisés au fil de leur chargement pour s'assurer qu'une app n'a pas été modifiée depuis son installation ou sa dernière mise à jour.

Sécurité des processus exécutés

Après avoir vérifié qu'une app provient d'une source approuvée, iOS applique des mesures de sécurité destinées à l'empêcher de compromettre les autres apps ou le reste du système.

Toutes les apps de tierce partie sont placées en environnement contrôlé afin qu'elles ne puissent ni accéder aux fichiers stockés par les autres apps, ni apporter de modifications à l'appareil. Cela empêche les apps de collecter ou de modifier les informations stockées par les autres apps. Chaque app se voit attribuer de façon aléatoire un répertoire de départ unique pour ses fichiers lors de son installation. Si une app de tierce partie doit accéder à des informations autres que les siennes, elle ne peut le faire qu'en utilisant les services explicitement fournis par iOS.

Les fichiers et ressources du système sont également protégés des apps de l'utilisateur. La majeure partie d'iOS s'exécute en tant qu'utilisateur non privilégié « mobile », comme toutes les apps de tierce partie. L'ensemble de la partition du système d'exploitation est monté en lecture seule. Les outils qui ne sont pas indispensables, comme les services d'ouverture de session à distance, ne sont pas inclus dans le logiciel système et les API ne permettent pas aux apps d'augmenter leurs propres privilèges afin de modifier les autres apps ou l'iOS.

L'accès aux informations de l'utilisateur et à des fonctionnalités comme iCloud et l'extensibilité par les apps de tierce partie est contrôlé à l'aide de droits déclarés. Les droits sont des paires clé-valeur signées intégrées à une app et permettent l'authentification au-delà des facteurs d'exécution, comme l'identifiant utilisateur Unix. Comme les droits sont signés numériquement, ils ne peuvent pas être modifiés. Les droits sont très utilisés par les apps et les daemons système pour réaliser des opérations nécessitant des privilèges spécifiques pour lesquelles le processus devrait normalement s'exécuter en tant qu'utilisateur root. Cela réduit considérablement le risque d'augmentation des privilèges par une application ou un daemon système compromis.

En outre, les apps ne peuvent réaliser un traitement en arrière-plan que par le biais des API fournies par le système. Cela leur permet de continuer à s'exécuter sans affecter les performances ni réduire de façon importante l'autonomie de la batterie.

La randomisation du format d'espace d'adresse (ASLR, Address Space Layout Randomization) empêche l'exploitation des bogues d'altération de mémoire. Les apps intégrées utilisent l'ASLR pour garantir la randomisation de toutes les régions de la mémoire au lancement. L'organisation aléatoire des adresses mémoire du code exécutable, des bibliothèques système et des structures de programmation associées réduit la probabilité de nombreuses exploitations sophistiquées. Par exemple, une attaque de type return-to-libc tente d'amener un appareil à exécuter un code malveillant en manipulant les adresses mémoire des bibliothèques de pile et système. La randomisation de l'organisation de celles-ci rend l'attaque beaucoup plus difficile à exécuter, en particulier sur plusieurs appareils. Xcode, l'environnement de développement d'iOS, compile automatiquement les programmes tiers avec la prise en charge de l'ASLR activée.

Une protection supplémentaire est apportée par iOS à l'aide du bit XN (Execute Never) du processeur ARM, qui permet de marquer des pages mémoire comme non exécutables. Les pages mémoire marquées à la fois comme accessibles en écriture et exécutables ne peuvent être utilisées par les apps que dans des conditions étroitement contrôlées : le noyau vérifie la présence du droit de signature de code dynamique réservé à Apple. Même dans ce cas, un seul appel mmap est autorisé pour demander une page exécutable et accessible en écriture, qui se voit attribuer une adresse randomisée. Safari utilise cette fonctionnalité pour son compilateur JavaScript JIT.

Extensions

iOS permet à des apps d'étendre les fonctionnalités d'autres apps par le biais d'extensions. Les extensions sont des exécutables binaires signés ayant une fonction spéciale incorporés dans une app. Le système détecte automatiquement les extensions lors de l'installation et les rend accessibles aux autres apps à l'aide d'un système de mise en correspondance.

Une zone système prenant en charge les extensions est appelée point d'extension. Chaque point d'extension fournit des API et applique des règles pour cette zone. Le système détermine quelles extensions sont disponibles d'après des règles de mise en correspondance propres au point d'extension. Le système lance automatiquement les processus d'extension lorsque cela est nécessaire et gère leur durée de vie. Des droits peuvent être utilisés pour limiter la disponibilité des extensions à des applications système précises. Par exemple, un widget d'affichage Aujourd'hui n'apparaît que dans le Centre de notifications et une extension de partage n'est disponible que dans la sous-fenêtre Partage. Les points d'extension sont les widgets Aujourd'hui, Partager, les actions Personnalisé, Édition photo, Fournisseur de documents et Clavier personnalisé.

Les extensions s'exécutent dans leur propre espace d'adresse. La communication entre une extension et l'app à partir de laquelle elle a été activée se fait par le biais des communications interprocessus assistées par le cadre d'application système. Elles n'ont accès ni aux fichiers, ni aux espaces mémoire de l'autre. Les extensions sont conçues pour être isolées l'une de l'autre, de l'app contenante et des apps qui les utilisent. Elles sont placées en environnement contrôlé comme toute autre app de tierce partie et possèdent un conteneur distinct de celui de l'app contenante. Toutefois, elles partagent le même accès aux contrôles de confidentialité que cette dernière. Ainsi, si un utilisateur accorde l'accès Contacts à une app, cette autorisation est étendue aux extensions intégrées à celle-ci, mais pas aux extensions activées par celle-ci.

Les claviers personnalisés sont un type d'extension particulier dans la mesure où ces extensions sont activées par l'utilisateur pour l'ensemble du système. Une fois activée, une extension de clavier est utilisée pour toute saisie de texte, à l'exception des codes et de tout autre texte saisi dans une présentation sécurisée. Pour limiter le transfert de données de l'utilisateur, les claviers personnalisés s'exécutent par défaut dans un environnement contrôlé très restrictif bloquant l'accès au réseau, aux services réalisant des opérations réseau pour le compte d'un processus et aux API qui permettraient à l'extension d'envoyer les données saisies. Les développeurs de claviers personnalisés peuvent demander à ce que leur extension bénéficie d'un accès libre, ce qui permet au système de l'exécuter dans l'environnement contrôlé par défaut après obtention du consentement de l'utilisateur.

Pour les appareils inscrits auprès d'un serveur de gestion des appareils mobiles, les extensions de document et de clavier obéissent aux règles de gestion d'ouverture de fichier (Managed Open In). Par exemple, le serveur MDM peut empêcher un utilisateur d'exporter un document d'une app gérée vers un fournisseur de documents non géré, ou d'utiliser un clavier non géré avec une app gérée. En outre, les développeurs d'apps peuvent empêcher l'utilisation d'extensions de clavier tierces avec leur app.

Groupes d'apps

Les apps et les extensions appartenant à un compte de développeur donné peuvent partager du contenu lorsqu'elles sont intégrées à un même groupe d'apps. Il appartient au développeur de créer les groupes appropriés sur le portail Apple Developer et d'y inclure les apps et les extensions souhaitées. Une fois intégrées à un groupe d'apps, les apps ont accès aux éléments suivants :

- un conteneur sur disque partagé pour le stockage, qui reste sur l'appareil tant qu'au moins une app du groupe est installée ;
- des préférences partagées ;
- des éléments de trousseau partagés.

Le portail Apple Developer garantit que les identifiants de groupes d'apps sont uniques dans l'ensemble de l'écosystème d'apps.

Protection des données dans les apps

Le kit de développement de logiciels (SDK, Software Development Kit) pour iOS offre une suite complète d'API permettant aux développeurs de tierce partie et internes d'adopter facilement la protection des données et d'assurer un niveau de protection maximal dans leurs apps. La protection des données est disponible pour les API de fichiers et de bases de données, notamment NSFileManager, CoreData, NSData et SQLite.

L'app Mail (y compris les pièces jointes), les livres gérés, les signets Safari, les images de lancement d'app et les données de localisation sont également stockées sous forme chiffrée avec des clés protégées par le code de l'utilisateur sur son appareil. Les apps Calendrier (à l'exception des pièces jointes), Contacts, Rappels, Notes, Messages et Photos implémentent la classe Protected Until First User Authentication.

Les apps installées par l'utilisateur qui n'optent pas pour une classe de protection des données spécifique reçoivent par défaut la classe Protected Until First User Authentication.

Accessoires

Le programme d'homologation Made for iPhone, iPod touch et iPad (MFi) permet aux fabricants d'accessoires approuvés d'accéder au protocole d'accessoires iPod (iAP, iPod Accessories Protocol) et aux composants matériels de prise en charge nécessaires.

Lorsqu'un accessoire MFi communique avec un appareil iOS par le biais d'un connecteur Lightning ou via Bluetooth, l'appareil demande à l'accessoire de prouver qu'il a été autorisé par Apple en répondant avec un certificat fourni par Apple, qui est vérifié par l'appareil. L'appareil envoie ensuite un challenge auquel l'accessoire doit répondre à l'aide d'une réponse signée. Ce processus est entièrement géré par un circuit intégré personnalisé qu'Apple fournit aux fabricants d'accessoires approuvés et se fait en toute transparence pour l'accessoire.

Les accessoires peuvent demander l'accès à différentes fonctionnalités et méthodes de transport ; par exemple, l'accès à des flux audio numériques sur le câble Lightning ou à des informations de localisation fournies par Bluetooth. Un CI d'authentification garantit que seuls les accessoires approuvés se voient accorder l'accès complet à l'appareil. Si un accessoire ne s'authentifie pas, son accès est limité au flux audio analogique et à un sous-ensemble restreint de commandes de lecture audio série (UART).

AirPlay utilise également le CI d'authentification pour vérifier que les récepteurs ont été approuvés par Apple. Les flux audio AirPlay et vidéo CarPlay emploient le protocole MFi-SAP (Secure Association Protocol), qui chiffre les communications entre l'accessoire et l'appareil à l'aide du protocole AES-128 en mode CTR. Des clés éphémères sont échangées à l'aide du protocole d'échange de clés ECDH (Curve25519) et signées à l'aide de la clé RSA 1 024 bits du CI d'authentification dans le cadre du protocole Station-to-Station (STS).

HomeKit

HomeKit fournit une infrastructure d'automatisation à domicile qui fait appel aux fonctionnalités de sécurité d'iCloud et d'iOS pour protéger et synchroniser les données personnelles sans les exposer à Apple.

Identité HomeKit

La sécurité et l'identité HomeKit reposent sur des paires de clés publique-privée Ed25519. Une paire de clés Ed25519 est générée pour HomeKit sur l'appareil iOS pour chaque utilisateur et devient son identité HomeKit. Elle est utilisée pour authentifier la communication entre les appareils iOS, et entre les appareils iOS et les accessoires.

Les clés sont stockées dans le trousseau et incluses uniquement dans les sauvegardes chiffrées de ce dernier. Elles sont synchronisées entre les appareils à l'aide du trousseau iCloud.

Communication avec les accessoires HomeKit

Les accessoires HomeKit génèrent leur propre paire de clés Ed25519 pour communiquer avec les appareils iOS. Si les réglages d'origine de l'accessoire sont rétablis, une nouvelle paire de clés est générée.

Pour établir une relation entre un appareil iOS et un accessoire HomeKit, les clés sont échangées à l'aide du protocole Secure Remote Password (3 072 bits), en utilisant un code à huit chiffres fourni par le fabricant de l'accessoire et saisi sur l'appareil iOS par l'utilisateur, puis chiffrées avec l'algorithme AEAD ChaCha20-Poly1305 avec des clés obtenues à l'aide de la fonction de dérivation HKDF-SHA-512. La certification MFi de l'accessoire est également vérifiée lors de la configuration.

Lorsque l'appareil iOS et l'accessoire HomeKit communiquent, chacun authentifie l'autre en utilisant les clés échangées comme décrit ci-dessus. Chaque session est établie à l'aide du protocole Station-to-Station et chiffrée avec les clés obtenues à l'aide de la fonction de dérivation HKDF-SHA-512 à partir des clés Curve25519 par session. Cela s'applique aux accessoires IP et aux accessoires Bluetooth économes en énergie.

Stockage local des données

HomeKit stocke les données concernant les domiciles, les accessoires, les scènes et les utilisateurs sur l'appareil iOS d'un utilisateur. Ces données stockées sont chiffrées à l'aide de clés obtenues à partir des clés de l'identité HomeKit de l'utilisateur et d'un nonce aléatoire. En outre, les données HomeKit sont stockées avec la classe Protected Until First User Authentication. Les données HomeKit ne sont sauvegardées que sous forme chiffrée ; ainsi, les sauvegardes iTunes non chiffrées, par exemple, ne contiennent pas les données HomeKit.

Synchronisation des données entre les appareils et les utilisateurs

Les données HomeKit peuvent être synchronisées entre les appareils iOS d'un utilisateur à l'aide d'iCloud et du trousseau iCloud. Les données HomeKit sont chiffrées pendant la synchronisation à l'aide de clés obtenues à partir de l'identité HomeKit de l'utilisateur et d'un nonce aléatoire. Ces données sont traitées sous la forme d'un blob opaque pendant la synchronisation. Le blob le plus récent est stocké dans iCloud pour permettre la synchronisation, mais il n'est pas utilisé à d'autres fins. Comme il est chiffré à l'aide de clés disponibles uniquement sur les appareils iOS de l'utilisateur, son contenu est inaccessible pendant la transmission vers iCloud et pendant son stockage.

Les données HomeKit sont également synchronisées entre plusieurs utilisateurs du même domicile. Ce processus fait appel à une authentification et un chiffrement identiques à ceux utilisés entre un appareil iOS et un accessoire HomeKit. L'authentification est basée sur des clés publiques Ed25519 échangées entre les appareils lorsqu'un utilisateur est ajouté à un domicile. Après l'ajout d'un utilisateur à un domicile, chaque communication ultérieure est authentifiée et chiffrée à l'aide du protocole Station-to-Station et des clés par session.

L'utilisateur ayant initialement créé le domicile dans HomeKit ou tout autre utilisateur bénéficiant d'autorisation de modification peut ajouter des utilisateurs. L'appareil du propriétaire configure les accessoires avec la clé publique du nouvel utilisateur afin qu'ils puissent authentifier et accepter les commandes de ce dernier. Lorsqu'un utilisateur bénéficiant d'autorisations de modification ajoute un nouvel utilisateur, le processus est délégué à un concentrateur domestique pour terminer l'opération.

La procédure de transfert de l'Apple TV en vue d'une utilisation avec HomeKit est réalisée automatiquement si l'utilisateur se connecte à iCloud. Le compte iCloud nécessite l'activation de l'identification à deux facteurs. Apple TV et les clés publiques Ed25519 temporaires d'échange d'appareil du propriétaire sur iCloud. Lorsque l'appareil du propriétaire et l'Apple TV se trouvent sur le même réseau local, les clés temporaires servent à sécuriser une connexion à travers le réseau local en utilisant le protocole Station-to-Station et des clés de session. Ce processus fait appel à une authentification et un chiffrement identiques à ceux utilisés entre un appareil iOS et un accessoire HomeKit. Par le biais de cette connexion locale sécurisée, l'appareil du propriétaire transfère à l'Apple TV la paire de clés Ed25519 publique-privée de l'utilisateur. Ces clés sont ensuite utilisées pour sécuriser les transmissions entre l'Apple TV et les accessoires HomeKit mais aussi entre l'Apple TV et les autres appareils iOS intégrant le domicile HomeKit.

Si un utilisateur n'a qu'un seul appareil et qu'il refuse l'accès à son domicile à d'autres utilisateurs, aucune donnée HomeKit n'est synchronisée avec iCloud.

Données du domicile et apps

L'accès aux données du domicile par les apps est contrôlé par les réglages de confidentialité de l'utilisateur. Lorsque des apps demandent à accéder aux données du domicile, l'utilisateur est invité à indiquer son choix, comme pour Contacts, Photos et d'autres sources de données iOS. S'il donne son accord, les apps peuvent connaître le nom des pièces et des accessoires, savoir dans quelle pièce se trouve chaque accessoire et accéder à d'autres informations, comme décrit en détail dans la documentation du développeur HomeKit à l'adresse https://developer.apple.com/homekit/.

HomeKit et Siri

Siri peut être utilisé pour interroger et commander les accessoires, et pour activer des scènes. Un minimum d'informations concernant la configuration du domicile est fourni de façon anonyme à Siri, afin de communiquer le nom des pièces, des accessoires et des endroits nécessaires à la reconnaissance des commandes. Il se peut que le son envoyé à Siri fasse état d'accessoires ou de commandes spécifiques, mais ces données Siri ne sont pas associées aux autres fonctionnalités Apple comme HomeKit. Pour en savoir plus, reportez-vous à « Siri » dans la section Services Internet de ce document.

Accès distant à iCloud pour les accessoires HomeKit

Un accessoire HomeKit peut se connecter directement à iCloud pour permettre aux appareils iOS de le contrôler si les transmissions par Bluetooth ou par Wi-Fi ne sont pas disponibles.

L'accès distant à iCloud a été soigneusement conçu pour que les accessoires puissent être contrôlés et des notifications envoyées sans révéler à Apple l'identité des accessoires ou les commandes et notifications envoyées. HomeKit n'envoie pas d'informations relatives au domicile à travers l'accès distant à iCloud.

Lorsqu'un utilisateur envoie une commande par le biais de l'accès distant à iCloud, l'accessoire et l'appareil iOS sont mutuellement authentifiés et les données sont chiffrées en utilisant la même procédure décrite pour les connexions locales. Le contenu des transmissions est chiffré et n'est pas visible par Apple. L'adressage à travers iCloud s'articule autour d'identifiants iCloud inscrits au cours du processus de configuration.

Les accessoires prenant en charge l'accès distant à iCloud sont attribués pendant le processus de configuration de l'accessoire. Le processus d'attribution commence par l'ouverture d'une session de l'utilisateur sur iCloud. L'appareil iOS demande ensuite à l'accessoire de signer un défi en utilisant le coprocesseur d'authentification d'Apple, intégré dans tous les accessoires conçus pour HomeKit. L'accessoire génère également des clés elliptiques de type prime256v1, et la clé publique est envoyée à l'appareil iOS accompagné du défi signé et du certificat X.509 du coprocesseur d'authentification. Ceux-ci servent à demander un certificat pour l'accessoire depuis le serveur d'attribution iCloud. Le certificat est stocké par l'accessoire, mais il ne contient aucune information d'identification sur l'accessoire, hormis la mention que l'accès distant à iCloud pour HomeKit lui a été accordé. L'appareil iOS conduisant l'attribution envoie également un conteneur à l'accessoire, incluant les URL et autres informations nécessaires pour la connexion au serveur d'accès distant à iCloud. Ces informations ne sont pas spécifiques à un utilisateur ou un accessoire particulier.

Chaque accessoire inscrit une liste d'utilisateurs autorisés auprès du serveur d'accès distant à iCloud. Ces utilisateurs se voient accorder le droit de contrôler l'accessoire par la personne ayant ajouté l'accessoire au domicile. Le serveur iCloud affecte un identifiant aux utilisateurs, qu'il est possible d'associer à un compte iCloud dans le but de distribuer les messages de notification et les réponses des accessoires. De même, les accessoires possèdent un identifiant émis par iCloud, mais qui est opaque et ne révèle aucune information relative à l'accessoire même.

Lorsqu'un accessoire se connecte au serveur d'accès distant à iCloud pour HomeKit, il présente son certificat et un billet. Ce billet est obtenu auprès d'un serveur iCloud différent ; celui-ci n'est pas unique à chaque accessoire. Lorsqu'un accessoire demande un billet, il indique dans sa requête son fabricant, son modèle et la version de son programme interne. Aucune information d'identification de l'utilisateur ou du domicile n'est envoyée dans cette requête. La connexion au serveur de billet n'est pas authentifiée, afin de contribuer à protéger la confidentialité.

Les accessoires se connectent au serveur d'accès distant à iCloud par HTTP/2, dont la liaison est sécurisée par TLS 1.2 avec AES-128-GCM et SHA-256. L'accessoire garde sa connexion au serveur d'accès distant à iCloud ouverte afin de pouvoir recevoir les messages entrants et envoyer les réponses et les notifications sortantes aux appareils iOS.

SiriKit

Siri utilise le mécanisme d'extension iOS pour communiquer avec les applications tierces. Bien que Siri ait accès aux contacts iOS et à la géolocalisation de l'appareil, Siri vérifie l'autorisation d'accès aux données utilisateur protégées par iOS qui contiennent l'extension pour savoir si l'app a accès avant de lui fournir ces informations. Siri ne passe que le fragment approprié du texte de la requête utilisateur d'origine à l'extension. Par exemple, si l'app n'a pas accès aux contacts iOS, Siri n'interprète pas la relation personnelle dans la requête d'un utilisateur telle que « Envoie 10 € à ma mère avec <app de paiement> ». Dans ce cas, l'app de l'extension ne voit que « mère » à travers le fragment d'occurrence brute qui lui est passé. Cependant, si l'app a effectivement accès aux contacts iOS, elle reçoit les données de contact iOS pour la mère de l'utilisateur. Si un contact a été mentionné dans le corps d'un message, par exemple « Dis à ma mère sur <app de messagerie> que mon frère est génial », Siri n'interprète alors pas « mon frère » indépendamment des TCC de l'app. Il se peut que le contenu présenté par l'app soit envoyé au serveur pour permettre à Siri de comprendre le vocabulaire qu'un utilisateur est susceptible d'employer dans l'app.

Lors de l'exécution, Siri permet à l'app compatible SiriKit de fournir un ensemble de mots personnalisés propres à l'instance d'application. Ces mots personnalisés sont liés à l'identifiant aléatoire, abordé dans la section sur Siri, et possèdent la même durée de vie.

HealthKit

HealthKit stocke et recueille les données provenant des apps de santé et de condition physique avec la permission de l'utilisateur. HealthKit fonctionne également directement avec les appareils de santé et de condition physique, comme les ceintures cardiaques Bluetooth faible énergie compatibles et le coprocesseur de mouvement intégré à de nombreux appareils iOS.

Données de santé

HealthKit stocke et recueille les données de santé de l'utilisateur, comme la taille, le poids, la distance parcourue, la tension artérielle, et plus encore. Ces données sont conservées dans la classe de protection des données Complete Protection, ce qui signifie qu'elles ne sont accessibles qu'après que l'utilisateur a déverrouillé l'appareil en saisissant son code ou en utilisant Touch ID.

HealthKit recueille les données de gestion, comme les autorisations d'accès des apps, les noms des appareils connectés à HealthKit et les informations de programmation utilisées pour lancer les apps lorsque de nouvelles données sont disponibles. Ces données sont stockées avec la classe Protected Until First User Authentication.

Des fichiers journaux temporaires stockent les informations de santé générées pendant que l'appareil est verrouillé, comme lorsque l'utilisateur pratique une activité physique. Ils sont associés à la classe de protection des données Protected Unless Open. Lorsque l'appareil est déverrouillé, ils sont importés dans les bases de données de santé principales, puis supprimés une fois la fusion terminée.

Les données de santé ne sont pas synchronisées entre les appareils. Les données de santé sont incluses dans les sauvegardes de l'appareil conservées sur iCloud et dans les sauvegardes iTunes chiffrées. Les données de santé ne sont pas incluses dans les sauvegardes iTunes non chiffrées.

Intégrité des données

Les données stockées dans la base de données comprennent des métadonnées permettant de connaître la provenance de chaque enregistrement. Ces métadonnées incluent un identifiant d'application qui identifie l'app ayant stocké l'enregistrement. En outre, un élément de métadonnées facultatif peut contenir une copie signée numériquement de l'enregistrement, afin d'assurer l'intégrité des données des enregistrements générés par un appareil de confiance. Le format utilisé pour la signature numérique est la syntaxe de message cryptographique (CMS, Cryptographic Message Syntax) spécifiée dans le RFC 5652 de l'IETF.

Accès par des apps de tierce partie

L'accès à l'API HealthKit est contrôlé par des droits, et les apps doivent se conformer aux restrictions concernant l'utilisation des données. Par exemple, elles ne sont pas autorisées à utiliser les données de santé pour afficher des publicités. Elles doivent également fournir aux utilisateurs une politique de confidentialité indiquant en détail comment elles utilisent les données de santé.

L'accès aux données de santé par les apps est contrôlé par les réglages de confidentialité de l'utilisateur. Lorsque des apps demandent à accéder aux données de santé, l'utilisateur est invité à indiquer son choix, comme pour Contacts, Photos et d'autres sources de données iOS. Toutefois, pour les données de santé, les apps se voient accorder des accès distincts pour la lecture et l'écriture, ainsi que pour chaque type de données de santé. Les utilisateurs peuvent consulter, et révoquer, les autorisations d'accès aux données de santé qu'ils ont accordées dans l'onglet Sources de l'app Santé.

Si des apps sont autorisées à écrire des données, elles peuvent également lire celles-ci. Si elles sont autorisées à lire des données, elles peuvent lire les données écrites par toutes les sources. Toutefois, les apps ne peuvent pas déterminer les autorisations d'accès accordées aux autres apps. En outre, elles ne peuvent pas savoir avec certitude si elles sont autorisées à lire les données de santé. Quand une app ne dispose pas d'une autorisation de lecture, les requêtes ne renvoient aucune donnée, comme lorsque la base de données est vide. Cela évite que les apps interfèrent avec l'état de santé de l'utilisateur en s'adaptant aux types de données qui l'intéressent.

Fiche médicale

L'app Santé offre aux utilisateurs la possibilité de renseigner une fiche médicale contenant des informations qui pourraient s'avérer importantes en cas d'urgence. Celles-ci sont saisies ou actualisées manuellement et ne sont pas synchronisées avec les informations contenues dans les bases de données de santé.

Les informations de la fiche médicale peuvent être consultées en touchant le bouton Urgence sur l'écran de verrouillage. Elles sont stockées sur l'appareil avec la classe de protection des données No Protection afin d'être accessibles sans avoir à saisir le code de l'appareil. La fiche médicale est une fonctionnalité facultative qui permet aux utilisateurs de trouver un juste équilibre entre sécurité et confidentialité.

ReplayKit

ReplayKit constitue un cadre d'application qui permet aux développeurs d'ajouter des fonctionnalités d'enregistrement et de diffusion en direct à leurs apps. De plus, il permet aux utilisateurs d'annoter leurs enregistrements et leurs diffusions à l'aide de la caméra frontale et du micro de l'appareil.

Enregistrement vidéo

Il existe plusieurs couches de sécurité intégrées à l'enregistrement d'une séquence :

- **Zone de dialogue Autorisations :** avant que l'enregistrement ne démarre, ReplayKit demande à l'utilisateur de reconnaître leur intention d'enregistrer l'écran ou d'utiliser le micro ou la caméra frontale. Cette alerte s'affiche une fois par processus d'app et réapparaît si l'app s'exécute en arrière-plan plus de 8 minutes.
- **Capture d'écran et audio :** la capture écran et audio se produit hors du processus de l'app dans le replayd du daemon de ReplayKit. Cela permet de s'assurer que le contenu enregistré n'est jamais accessible au processus de l'app.
- **Création et stockage de film :** le fichier vidéo est écrit dans le répertoire uniquement accessible par les sous-systèmes ReplayKit et ne l'est jamais à n'importe quelle autre app. Ceci empêche des tierces parties d'exploiter les enregistrements sans le consentement de l'utilisateur.
- **Aperçu et partage par l'utilisateur final :** l'utilisateur a la possibilité de prévisualiser et de partager la séquence à travers l'interface utilisateur proposée par ReplayKit. L'interface utilisateur est présentée hors-processus à travers l'infrastructure d'Extension iOS et a accès au fichier de séquence généré.

Diffusion

- **Capture d'écran et audio :** le mécanisme de capture d'écran et audio lors de la diffusion est identique à l'enregistrement de séquence et se produit dans replayd.
- **Extensions de diffusion :** pour que les services de tierce partie participent à la diffusion ReplayKit, ils doivent créer deux extensions configurées à l'aide du point d'arrivée com.apple.broadcast-services :
  - une extension d'interface utilisateur qui permet à l'utilisateur de configurer sa diffusion ;
  - une extension de transfert qui gère le téléchargement des données vidéo et audio aux serveurs backend du service.

L'architecture permet de s'assurer que les apps d'hébergement ne possèdent aucun privilège vis-à-vis du contenu vidéo et audio diffusé — seuls les extensions de diffusion ReplayKit et de tierce partie y ont accès.

- **Sélecteur de diffusion :** pour sélectionner le service de diffusion à utiliser, ReplayKit propose un contrôleur de présentation (semblable à UIActivityViewController) que le développeur peut présenter dans son app. Le contrôleur de présentation est implémenté à l'aide du SPI UIRemoteViewController et constitue une extension qui intègre le cadre d'application ReplayKit. Il se trouve hors-processus par rapport à l'app hôte.
  Extension de transfert : l'extension de téléchargement, que les services de diffusion de tierce partie mettent en place pour gérer le contenu vidéo et audio lors de la diffusion, dispose de deux façons pour recevoir du contenu :
  - en petites séquences MP4 encodées ;
  - en tampons d'échantillons non encodés au format Raw.
    - **Gestion de séquences MP4 :** avec ce mode de gestion, les séquences MP4 encodées sont générées par replayd et stockées dans un emplacement privé uniquement accessible aux sous-systèmes de ReplayKit. Une fois un plan vidéo généré, replayd passe l'emplacement de celui-ci à l'extension de téléchargement de tierce partie à travers le SPI de requête NSExtension (s'appuyant sur XPC). replayd

génère aussi un jeton d'environnement contrôlé à usage unique également passé à l'extension de téléchargement, ce qui accorde l'accès de l'extension au plan vidéo particulier au cours de la requête d'extension.

- **Gestion de tampon d'échantillon :** dans ce mode de gestion, les données vidéo et audio sont sérialisées et transmises en temps réel à l'extension de téléchargement de tierce partie à travers une connexion XPC directe. Les données vidéo sont encodées en extrayant l'objet IOSurface du tampon d'échantillon vidéo, de façon sécurisée sous forme d'objet XPC, en les envoyant par XPC à l'extension de tierce partie, puis en les redécodant de façon sécurisée dans l'objet IOSurface.

Notes sécurisées

L'app Notes comprend une fonctionnalité de notes sécurisées permettant aux utilisateurs de protéger le contenu de notes spécifiques. Les notes sécurisées sont chiffrées à l'aide d'une phrase secrète fournie par l'utilisateur et requise pour afficher les notes sous iOS et OS X, ainsi que sur le site web iCloud.

Lorsqu'un utilisateur sécurise une note, une clé sur 16 octets est calculée d'après la phrase secrète de l'utilisateur grâce aux algorithmes PBKDF2 et SHA256. Le contenu de la note est chiffré par le biais de l'algorithme AES-GCM. Les nouvelles entrées sont créées dans Core Data et CloudKit pour stocker les note, mot-clé et vecteur d'initialisation chiffrés, puis les entrées de note d'origine sont supprimées ; les données chiffrées ne sont pas écrites en local. Les pièces jointes sont également chiffrées de cette façon. Parmi les pièces jointes prises en charge, l'on retrouve les images, les dessins, les plans et les sites web. Les notes contenant d'autres types de pièces jointes ne peuvent pas être chiffrées ; celles non prises en charge ne peuvent en outre pas être ajoutées aux notes sécurisées.

Lorsqu'un utilisateur saisit correctement la phrase secrète, que ce soit pour afficher ou pour créer une note sécurisée, Notes ouvre une session sécurisée. Tant que l'app est ouverte, l'utilisateur n'a pas à saisir la phrase secrète ou à passer par Touch ID pour afficher ou sécuriser d'autres notes. Cependant, si certaines possèdent une phrase secrète différente, la session sécurisée ne s'applique qu'aux notes protégées à l'aide de la phrase secrète active. La session sécurisée se ferme lorsque l'utilisateur touche le bouton Verrouiller dans Notes, lorsque l'app se trouve en arrière-plan pendant plus de trois minutes ou quand l'appareil se verrouille.

Les utilisateurs qui oublient leur phrase secrète peuvent néanmoins afficher leurs notes sécurisées ou sécuriser d'autres notes s'ils activent Touch ID sur leurs appareils. En outre, Notes affiche un indice fourni par l'utilisateur après trois tentatives infructueuses de saisie de la phrase secrète. L'utilisateur doit connaître la phrase secrète en vigueur afin de pouvoir la modifier.

Les utilisateurs peuvent réinitialiser la phrase secrète s'ils ont oublié celle active. Cette fonctionnalité permet aux utilisateurs de créer de nouvelles notes sécurisées à l'aide d'une nouvelle phrase secrète, mais celle-ci ne leur permet pas de consulter les notes précédemment sécurisées. Il est néanmoins possible d'afficher les notes précédemment sécurisées si l'utilisateur se souvient de l'ancienne phrase secrète. La réinitialisation de la phrase secrète nécessite la phrase secrète du compte iCloud de l'utilisateur.

Les notes peuvent être partagées avec d'autres utilisateurs. Les données vidéo sont encodées en extrayant l'objet IOSurface du tampon d'échantillon vidéo, de façon sécurisée sous forme d'objet XPC, en les envoyant par XPC à l'extension de tierce partie, puis en les redécodant de façon sécurisée dans l'objet IOSurface.

Apple Watch

L'Apple Watch fait appel aux fonctionnalités et aux technologies de sécurité conçues pour iOS afin de protéger les données sur l'appareil, ainsi que les communications avec l'iPhone jumelé et Internet. Cela inclut les technologies telles que la protection des données et le contrôle de l'accès au trousseau. Le code de l'utilisateur est également combiné à l'UID de l'appareil pour créer les clés de chiffrement.

Le jumelage de l'Apple Watch à l'iPhone est sécurisé à l'aide d'un processus hors bande pour l'échange des clés publiques, puis à l'aide du secret partagé de la liaison BTLE. L'Apple Watch affiche un motif animé, capturé par l'appareil photo de l'iPhone. Ce motif comporte un secret codé utilisé pour le jumelage hors bande BTLE 4.1. La saisie de code BTLE standard est employée comme méthode de jumelage de secours, si nécessaire.

Une fois la session BTLE établie, l'Apple Watch et l'iPhone échangent des clés par le biais d'un processus dérivé de l'IDS, comme décrit à la section iMessage de ce document. Une fois les clés échangées, la clé de session Bluetooth est effacée, et toutes les communications entre l'Apple Watch et l'iPhone sont chiffrées à l'aide de l'IDS, les liaisons BTLE et Wi-Fi chiffrées assurant une seconde couche de chiffrement. Les clés sont renouvelées toutes les 15 minutes afin de limiter l'exposition au cas où la transmission viendrait à être compromise.

Pour les apps devant diffuser des données, le chiffrement est assuré à l'aide des méthodes décrites dans « FaceTime » au sein de la section Services Internet de ce document, en faisant appel au service IDS fourni par l'iPhone jumelé.

L'Apple Watch implémente le chiffrement matériel du stockage et la protection des fichiers et des éléments de trousseau basée sur des classes, comme décrit dans la section « Chiffrement et protection des données » de ce document. Des conteneurs de clés dont l'accès est contrôlé sont également utilisés pour les éléments de trousseau. Les clés employées pour la communication entre l'Apple Watch et l'iPhone sont aussi sécurisées à l'aide d'une protection basée sur des classes.

Lorsque l'Apple Watch ne se trouve pas dans le champ Bluetooth, la connexion Wi-Fi peut être utilisée à la place. Apple Watch ne se connecte pas aux réseaux Wi-Fi tant que les informations d'identification — lesquelles doivent avoir été préalablement synchronisées avec l'Apple Watch — ne sont pas présentes sur l'iPhone jumelé. Si l'Apple Watch ne se trouve pas dans le champ de détection de l'iPhone, aucune nouvelle information d'identification réseau sur l'iPhone ne se trouve sur l'Apple Watch.

L'Apple Watch peut être verrouillée manuellement en maintenant enfoncé le bouton latéral. En outre, des heuristiques de mouvements sont utilisées pour tenter de verrouiller automatiquement l'appareil peu après son retrait du poignet. Lorsqu'il est verrouillé, Apple Pay ne peut pas être utilisé. Si le verrouillage automatique par la fonctionnalité de détection du poignet est désactivé dans les réglages, Apple Pay l'est également. La détection du poignet peut être désactivée à l'aide de l'app Apple Watch sur l'iPhone. Ce réglage peut également être appliqué à l'aide de la gestion des appareils mobiles.

L'iPhone jumelé peut aussi déverrouiller l'Apple Watch, à condition que celle-ci soit portée. Ce déverrouillage se fait en établissant une connexion authentifiée par les clés définies lors du jumelage. L'iPhone envoie la clé et l'Apple Watch l'utilise pour déverrouiller ses clés de protection des données. Le code de l'Apple Watch n'est ni connu de l'iPhone, ni transmis. Cette fonctionnalité peut être désactivée à l'aide de l'app Apple Watch sur l'iPhone.

L'Apple Watch peut être jumelée uniquement avec un iPhone à la fois. Ce dernier communique les instructions pour effacer tout le contenu et les données de l'Apple Watch lors du déjumelage.

L'activation de la fonctionnalité Localiser mon iPhone sur l'iPhone jumelé permet également l'utilisation du Verrouillage d'Activation sur l'Apple Watch. Le Verrouillage d'Activation complique l'usage ou la revente d'une Apple Watch en cas de perte ou de vol. Le Verrouillage d'Activation oblige l'utilisateur à saisir son identifiant et son mot de passe Apple pour déjumeler, effacer ou réactiver une Apple Watch.

Sécurité du réseau

En plus des dispositifs intégrés mis en place par Apple pour protéger les données stockées sur les appareils iOS, il existe de nombreuses mesures de sécurité réseau que les entreprises peuvent adopter pour sécuriser les informations lors de leur transfert vers et depuis un appareil iOS.

Les utilisateurs mobiles doivent pouvoir accéder aux réseaux d'entreprise depuis partout dans le monde, il est donc important de s'assurer qu'ils y sont autorisés et que leurs données sont protégées lors des transmissions. iOS utilise (et permet aux développeurs d'accéder à) des protocoles de mise en réseau standard pour établir des communications authentifiées, autorisées et chiffrées. Pour atteindre ces objectifs de sécurité, iOS intègre des technologies éprouvées et les dernières normes pour les connexions aux réseaux de données Wi-Fi et cellulaires.

Sur les autres plateformes, des logiciels coupe-feu sont nécessaires pour protéger les ports de communication ouverts de toute intrusion. Comme iOS réduit la surface d'attaque en limitant les ports d'écoute et en supprimant les utilitaires de réseau inutiles comme telnet, les shells ou un serveur web, aucun logiciel coupe-feu supplémentaire n'est nécessaire sur les appareils iOS.

TLS

iOS prend en charge les protocoles Transport Layer Security (TLS 1.0, TLS 1.1 et TLS 1.2, lesquels prennent en charge les méthodes AES 128 et SHA-2) et DTLS. Safari, Calendrier, Mail et d'autres apps Internet utilisent automatiquement ces mécanismes pour établir un canal de communication chiffré entre l'appareil et les services réseau. Les API de haut niveau (comme CFNetwork) permettent aux développeurs d'adopter facilement le protocole TLS dans leurs apps, tandis que les API de bas niveau (SecureTransport) apportent un contrôle fin. L'API CFNetwork n'autorise pas SSL 3. Les apps qui exploitent WebKit (comme Safari) se voient interdire d'établir une connexion SSL 3.

La suite de chiffrement symétrique RC4 est obsolète dans iOS 10 et macOS Sierra. Par défaut, les clients ou serveurs TLS implémentés à travers les API SecureTransport n'ont pas les suites de chiffrement RC4 activées et ne sont pas en mesure de se connecter lorsque RC4 est la seule suite de chiffrement disponible. Pour mieux se sécuriser, les services ou apps qui nécessitent RC4 doivent être mis à niveau pour exploiter les suites modernes de chiffrement sécurisé.

Sécurité du transport des apps

La sécurité du transport des apps impose des critères de connexion par défaut de sorte que les apps doivent se conformer aux « bonnes pratiques » pour les connexions sécurisées en cas d'usage des API NSURLConnection, CFURL ou NSURLSession. Par défaut, la sécurité du transport des apps limite la sélection de chiffrement pour n'inclure que les suites qui assurent la confidentialité persistante, particulièrement ECDHE_ECDSA_AES et ECDHE_RSA_AES en mode GCM ou CBC. Les apps sont en mesure de désactiver l'exigence de confidentialité persistante par domaine, auquel cas RSA_AES est ajouté à l'ensemble des chiffrements disponibles.

Les serveurs doivent prendre en charge TLS 1.2 et la confidentialité persistante, et les certificats doivent être valides et signés par l'algorithme SHA-256 ou plus évolué avec une clé RSA sur 2048 bits ou une clé elliptique de 256 bits minimum.

Les connexions réseau ne satisfaisant pas ces critères échouent, à moins que l'app outrepasse la sécurité du transport des apps. Les certificats non valides entraînent toujours un échec sec et n'obtiennent aucune connexion. La sécurité du transport des apps est automatiquement appliquée aux applications compilées pour iOS 9 ou ultérieur.

VPN

Les services réseau sécurisés comme les réseaux privés virtuels ne nécessitent généralement qu'une configuration minime pour fonctionner avec les appareils iOS. Ceux-ci sont compatibles avec les serveurs VPN prenant en charge les protocoles et méthodes d'authentification ci-dessous :

- IKEv2/IPSec avec authentification utilisateur par secret partagé, certificats RSA, certificats ECDSA, protocole EAP-MSCHAPv2 ou protocole EAP-TLS ;
- SSL-VPN à l'aide de l'app cliente appropriée disponible dans l'App Store ;
- IPSec Cisco avec authentification utilisateur par mot de passe, RSA SecurID ou CRYPTOCard, et authentification machine par secret partagé et certificats ;
- L2TP/IPSec avec authentification utilisateur par mot de passe MS-CHAPV2, RSA SecurID ou CRYPTOCard, et authentification machine par secret partagé ;
- PPTP est pris en charge dans iOS 9.3 et antérieur, mais déconseillé.

iOS prend en charge le VPN à la demande pour les réseaux qui utilisent une authentification par certificat. Les services informatiques spécifient les domaines qui nécessitent une connexion VPN à l'aide d'un profil de configuration.

iOS prend également en charge la fonctionnalité VPN par app, qui permet de définir beaucoup plus finement quand une connexion VPN doit être établie. Le serveur de gestion des appareils mobiles (MDM) peut spécifier une connexion pour chaque app gérée et/ou des domaines précis dans Safari. Cela permet de garantir que les données confidentielles sont toujours transmises vers et depuis le réseau de l'entreprise, mais pas les données personnelles d'un utilisateur.

iOS prend en charge le VPN permanent, qui peut être configuré pour les appareils gérés via MDM et supervisés à l'aide d'Apple Configurator ou du programme d'inscription d'appareil. Cela évite aux utilisateurs d'avoir à activer le VPN pour être protégés lorsqu'ils se connectent à des réseaux cellulaires et Wi-Fi. Le VPN permanent offre à une entreprise un contrôle complet sur le trafic des appareils en tunnellisant tout le trafic IP jusqu'à elle. Le protocole de tunnellisation par défaut, IKEv2, sécurise les transmissions en chiffrant les données. L'entreprise peut désormais surveiller et filtrer le trafic vers et depuis ses appareils, sécuriser les données au sein de son réseau et limiter l'accès des appareils à Internet.

Wi-Fi

iOS prend en charge les protocoles Wi-Fi standard, y compris WPA2 Enterprise, pour offrir un accès authentifié aux réseaux d'entreprise sans fil. WPA2 Enterprise utilise un chiffrement AES 128 bits, qui garantit aux utilisateurs une protection maximale de leurs données lors de l'envoi et de la réception par le biais d'une connexion réseau Wi-Fi. Grâce à la prise en charge de la norme 802.1X, les appareils iOS peuvent être intégrés dans un très grand nombre d'environnements d'authentification RADIUS. Les méthodes d'authentification sans fil 802.1X prises en charge sur l'iPhone et sur l'iPad comprennent EAP-TLS, EAP-TTLS, EAP-FAST, EAP-SIM, PEAPv0, PEAPv1 et LEAP.

Outre la protection des données, iOS étend la protection de niveau WPA2 aux images de la gestion monodiffusion et multidiffusion à travers le service image de gestion protégée mentionnée dans la norme 802.11w. La prise en charge PMF est disponible sur l'iPhone 6s et l'iPad Air 2 et ultérieurs.

iOS utilise une adresse Media Access Control (MAC) randomisée lors des recherches Wi-Fi s'il n'est pas déjà associé à un réseau Wi-Fi. Ces recherches sont possibles afin de pouvoir retrouver et connecter un réseau Wi-Fi préféré ou afin d'assister le service de localisation pour les apps exploitant des géorepérages, par exemple les rappels géodépendants ou la résolution d'un emplacement dans Plans Apple. Il est important de noter que les recherches Wi-Fi, lesquelles se produisent lors de la tentative de connexion à un réseau Wi-Fi préféré, ne sont pas aléatoires.

iOS utilise également une adresse MAC randomisée pour effectuer des recherches enhanced Preferred Network Offload (ePNO) lorsqu'un appareil n'est pas associé à un réseau Wi-Fi ou que son processeur est en veille. Ces recherches sont exécutées si un appareil fait appel au service de localisation pour des apps utilisant le géorepérage, comme les rappels en fonction du lieu qui interviennent si l'appareil se trouve près d'un lieu précis.

Comme l'adresse MAC d'un appareil change désormais lorsqu'il est déconnecté d'un réseau Wi-Fi, elle ne peut pas être utilisée par des observateurs passifs de trafic Wi-Fi pour suivre en permanence un appareil, même si celui-ci est connecté à un réseau cellulaire. Apple a informé les fabricants de cartes Wi-Fi que les recherches Wi-Fi d'iOS utilisent une adresse MAC randomisée, et que ni Apple, ni les fabricants ne peuvent prédire ces adresses MAC randomisées. La prise en charge de la randomisation d'adresse MAC Wi-Fi est indisponible sur l'iPhone 4s et téléphones antérieurs.

Sur l'iPhone 6s et ultérieur, la propriété masquée d'un réseau Wi-Fi est connue et actualisée automatiquement. Si l'identifiant d'ensemble de service (SSID) d'un réseau Wi-Fi est diffusé, l'appareil iOS n'envoie pas de sonde avec le SSID inclus dans la requête. Cela empêche l'appareil de diffuser le nom des réseaux non masqués.

Bluetooth

La prise en charge de Bluetooth dans iOS a été conçue pour offrir une fonctionnalité utile sans étendre inutilement l'accès aux données privées. Les appareils iOS prennent en charge les connexions avec mode de chiffrement 3, mode de sécurité 4 et niveau de service 1. iOS prend en charge les profils Bluetooth suivants :

- profil mains libres (HFP 1.5) ;
- profil d'accès à l'annuaire (PBAP) ;
- profil de distribution audio avancée (A2DP) ;
- profil de télécommande audio/vidéo (AVRCP) ;
- profil de réseau personnel (PAN) ;
- profil d'appareil à interface humaine (HID).

La prise en charge de ces profils dépend de l'appareil. Pour en savoir plus, consultez la page suivante : https://support.apple.com/fr-fr/ht3647.

Authentification unique

iOS prend en charge l'authentification aux réseaux d'entreprise par authentification unique (SSO, Single Sign-on). La SSO fonctionne avec les réseaux utilisant le protocole d'authentification Kerberos pour authentifier les utilisateurs auprès des services auxquels ils sont autorisés à accéder. La SSO peut être utilisée pour de nombreuses activités réseau, des sessions Safari sécurisées aux apps de tierce partie. L'authentification par certificat (PKINIT) est également prise en charge.

La SSO iOS fait appel à des jetons SPNEGO et au protocole HTTP Negotiate pour fonctionner avec les passerelles d'authentification basée sur Kerberos et les systèmes d'authentification intégrée Windows prenant en charge les tickets Kerberos. La prise en charge de la SSO repose sur le projet open-source Heimdal.

Les types de chiffrement suivants sont pris en charge :

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

Safari prend en charge la SSO et les applications tierces qui utilisent les API de mise en réseau standard d'iOS peuvent également être configurées pour l'utiliser. Pour configurer la SSO, iOS prend en charge une entité de profil de configuration permettant aux serveurs MDM de transmettre les réglages nécessaires. Cela comprend la définition du nom principal de l'utilisateur (c'est-à-dire, le compte utilisateur Active Directory) et des réglages du royaume Kerberos, ainsi que la configuration des apps et/ou des URL Safari autorisées à utiliser la SSO.

Sécurité AirDrop

Les appareils iOS qui prennent en charge AirDrop utilisent Bluetooth faible énergie (BLE, Bluetooth Low Energy) et une technologie Wi-Fi poste à poste créée par Apple pour envoyer des fichiers et des informations aux appareils se trouvant à proximité, notamment les ordinateurs Mac compatibles AirDrop exécutant OS X 10.11 ou ultérieur. Les appareils communiquent directement entre eux par Wi-Fi sans utiliser de connexion Internet ni de point d'accès Wi-Fi.

Lorsqu'un utilisateur active AirDrop, une identité RSA à 2 048 bits est stockée sur l'appareil. En outre un hachage d'identité AirDrop est créé à partir des adresses électroniques et des numéros de téléphone associés à l'Identifiant Apple de l'utilisateur.

Lorsqu'un utilisateur choisit AirDrop comme méthode de partage pour un élément, l'appareil émet un signal AirDrop via Bluetooth faible énergie. Les autres appareils allumés et situés à proximité sur lesquels AirDrop est activé détectent ce signal et répondent en envoyant une version courte du hachage d'identité de leur propriétaire.

Par défaut, AirDrop est configuré pour ne partager des données qu'avec les contacts. Les utilisateurs peuvent choisir d'utiliser AirDrop pour partager des données avec tout le monde ou de désactiver complètement cette fonctionnalité. En mode Contacts, les hachages d'identité reçus sont comparés à ceux des personnes présentes dans l'app Contacts de l'initiateur. Si une correspondance est trouvée, l'appareil émetteur crée un réseau Wi-Fi poste à poste et annonce une connexion AirDrop à l'aide de Bonjour. Les appareils récepteurs utilisent alors cette connexion pour envoyer leur hachage d'identité complet à l'initiateur. Si le hachage complet correspond toujours à celui présent dans Contacts, le prénom et la photo du destinataire (si disponibles dans Contacts) s'affichent dans la feuille de partage AirDrop.

Lors de l'utilisation d'AirDrop, l'expéditeur sélectionne les personnes avec lesquelles il veut partager des données. L'appareil émetteur établit avec l'appareil récepteur une connexion chiffrée (TLS) via laquelle sont échangés les certificats d'identité iCloud. L'identité figurant dans les certificats est vérifiée auprès de l'app Contacts de chaque utilisateur. Le destinataire est alors invité à accepter le transfert entrant en provenance de la personne ou de l'appareil identifié. Si plusieurs destinataires ont été sélectionnés, ce processus est répété pour chaque destination.

En mode Toute le monde, le même processus est utilisé, mais si aucune correspondance n'est trouvée dans Contacts, les appareils récepteurs apparaissent dans la feuille de partage AirDrop avec une silhouette et le nom de l'appareil défini dans Réglages > Général > Informations > Nom.

Les entreprises peuvent restreindre l'usage d'AirDrop pour les apps ou les appareils gérés par une solution de gestion d'appareils mobiles.

Apple Pay

Grâce à Apple Pay, les utilisateurs peuvent utiliser leurs appareils pris en charge par iOS et Apple Watch pour effectuer en toute simplicité des paiements de façon sûre et confidentielle dans les magasins, les apps et sur le web à travers le navigateur Safari. Ce système est simple pour les utilisateurs et sécurisé à la fois au niveau du matériel et des logiciels.

Apple Pay est aussi conçu pour protéger les informations personnelles de l'utilisateur. Il ne collecte aucune information relative aux transactions pouvant être associée à ce dernier. Les opérations de paiement sont réalisées entre l'utilisateur, le vendeur et l'émetteur de la carte.

Composants d'Apple Pay

Secure Element : Secure Element est une puce certifiée standard exécutant la plateforme Java Card qui est conforme aux exigences du secteur financier pour les paiements électroniques.

Contrôleur NFC : le contrôleur NFC gère les protocoles de communication en champ proche et achemine la communication entre le processeur d'application et Secure Element, et entre Secure Element et le terminal de point de vente.

Wallet : Wallet permet d'ajouter et de gérer des cartes bancaires et de fidélité, et de réaliser des paiements avec Apple Pay. Les utilisateurs peuvent consulter leurs cartes et des informations supplémentaires concernant l'émetteur de leur carte, la politique de confidentialité de celui-ci, les transactions récentes et plus encore dans Wallet. Ils peuvent également ajouter des cartes à Apple Pay dans Assistant de configuration et Réglages.

Secure Enclave : sur l'iPhone, l'iPad et l'Apple Watch série 1 et série 2, Secure Enclave gère le processus d'authentification et permet à une opération de paiement de se poursuivre. Elle stocke les données d'empreintes digitales pour Touch ID.

Sur l'Apple Watch, l'appareil doit être déverrouillé et l'utilisateur doit appuyer deux fois sur le bouton latéral. Le double-clic est détecté et transmis directement à Secure Element ou Secure Enclave le cas échéant, directement sans passer par le processeur d'application.

Serveurs Apple Pay : les serveurs Apple Pay gèrent l'état des cartes bancaires dans Wallet et les numéros de compte d'appareil stockés dans Secure Element. Ils communiquent à la fois avec l'appareil et avec les serveurs des réseaux de paiement. Les serveurs Apple Pay sont également chargés de rechiffrer les informations d'identification de paiement pour les paiements réalisés dans les apps.

Comment Apple Pay utilise Secure Element

Secure Element renferme un « applet » spécifiquement conçu pour gérer Apple Pay. Il comporte également des « applets » de paiement certifiés par les réseaux de paiement. Les données des cartes bancaires ou prépayées sont transmises par le réseau de paiement ou l'émetteur de la carte à ces « applets » de paiement chiffrées à l'aide de clés connues uniquement du réseau de paiement et du domaine de sécurité des « applets » de paiement. Ces données sont stockées dans les « applets » de paiement et protégées à l'aide des fonctionnalités de sécurité de Secure Element. Lors d'une transaction, le terminal communique directement avec Secure Element via le contrôleur de communication en champ proche (NFC) par le biais d'un bus matériel dédié.

Comment Apple Pay utilise le contrôleur NFC

En tant que passerelle vers Secure Element, le contrôleur NFC s'assure que toutes les opérations de paiement sans contact sont réalisées par le biais d'un terminal de point de vente situé à proximité immédiate de l'appareil. Seules les demandes de paiement émanant d'un terminal dans le champ sont considérées comme des transactions sans contact par le contrôleur NFC.

Une fois le paiement autorisé par le détenteur de la carte à l'aide de Touch ID ou d'un code, ou par un double appui sur le bouton latéral sur une Apple Watch déverrouillée, les réponses sans contact préparées par les « applets » de paiement dans Secure Element sont acheminées exclusivement vers le champ NFC par le contrôleur. Les détails de l'autorisation de paiement pour les transactions sans contact sont donc transmis uniquement au champ NFC local et ne sont jamais divulgués au processeur d'application. Par contre, pour les paiements réalisés dans les apps et sur le web, ils sont acheminés vers le processeur d'application puis, après chiffrement par Secure Element, vers le serveur Apple Pay.

Transfert sur cartes bancaires et prépayées

Lorsqu'un utilisateur ajoute une carte bancaire ou prépayée (y compris les cartes de fidélité) à Apple Pay, Apple envoie de façon sécurisée les données de celle-ci, ainsi que d'autres informations concernant le compte et l'appareil de l'utilisateur, à l'émetteur de la carte concerné ou à son fournisseur de service agréé. À l'aide de ces informations, l'émetteur de carte décide d'approuver ou non l'ajout de la carte à Apple Pay.

Apple Pay utilise trois appels côté serveur pour communiquer avec l'émetteur de carte ou le réseau dans le cadre du processus de transfert d'une carte : Champs obligatoires, Vérification de carte et Liaison et transfert. L'émetteur de la carte ou le réseau utilise ces appels pour vérifier, approuver et ajouter des cartes à Apple Pay. Ces sessions client-serveur sont chiffrées à l'aide du protocole SSL.

Les numéros de carte complets ne sont stockés ni sur l'appareil, ni sur les serveurs d'Apple. À la place, un numéro de compte d'appareil est créé, chiffré puis stocké dans Secure Element. Ce numéro unique est chiffré de sorte qu'Apple ne puisse pas y accéder. Le numéro de compte d'appareil étant unique et différent des numéros de carte bancaire habituels, l'émetteur de la carte peut empêcher son utilisation sur une carte à piste magnétique, par téléphone ou sur des sites web. Le numéro de compte d'appareil conservé dans Secure Element est isolé d'iOS et de watchOS, et n'est jamais stocké ni sur les serveurs Apple, ni sauvegardé dans iCloud.

Les cartes utilisées avec l'Apple Watch sont transférées pour Apple Pay à l'aide de l'app Apple Watch sur l'iPhone. Le transfert d'une carte pour l'Apple Watch nécessite que celle-ci se trouve à portée Bluetooth. Les cartes sont spécifiquement enregistrées pour une utilisation avec l'Apple Watch et possèdent leur propre numéro de compte d'appareil, stocké dans Secure Element sur l'Apple Watch.

Il existe trois moyens de transférer une carte bancaire ou prépayée dans Apple Pay :

- ajout manuel d'une carte à Apple Pay ;
- ajout de cartes bancaires enregistrées dans un compte iTunes Store à Apple Pay ;
- ajout d'une carte depuis l'app d'un émetteur de carte.

Ajout manuel d'une carte bancaire à Apple Pay

Lors de l'ajout manuel d'une carte, y compris une carte de fidélité, le nom, le numéro de carte, la date d'expiration et le code CVV sont utilisés pour faciliter le processus de transfert. Les utilisateurs peuvent saisir ces informations manuellement depuis Réglages, l'app Wallet ou l'app de l'Apple Watch, ou à l'aide de la caméra iSight. Lorsque la caméra capture les informations de la carte, Apple tente de renseigner le nom, le numéro de carte et la date d'expiration. La photo n'est jamais enregistrée sur l'appareil ni stockée dans la photothèque. Une fois tous les champs renseignés, le processus de Vérification de Carte vérifie les champs hormis le code CVV. Les données sont chiffrées puis envoyées au serveur Apple Pay.

Si le processus de vérification de carte renvoie un identifiant de conditions générales, Apple télécharge les conditions générales de l'émetteur de la carte concerné et les présente à l'utilisateur. Si ce dernier accepte les conditions générales, Apple envoie l'identifiant des conditions acceptées et le code CVV au processus de liaison et transfert. En outre, dans le cadre du processus de liaison et transfert, Apple partage des informations de l'appareil avec l'émetteur de la carte ou le réseau de paiement, comme des informations sur l'activité de vos comptes iTunes et App Store (par exemple, si vous effectuez souvent des transactions dans iTunes), des renseignements sur votre appareil (par exemple, le numéro de téléphone, le nom et le modèle de ce dernier, ainsi que ceux de tout appareil iOS complémentaire nécessaire à la configuration d'Apple Pay), et votre position approximative au moment de l'ajout de la carte (si le service de localisation est activé). À l'aide de ces informations, l'émetteur de carte décide d'approuver ou non l'ajout de la carte à Apple Pay.

À l'issue du processus de liaison et transfert, deux opérations ont lieu :

- L'appareil commence à télécharger le fichier Wallet représentant la carte bancaire.
- L'appareil commence à lier la carte à Secure Element.

Le fichier de billet contient des URL permettant de télécharger les illustrations de carte, ainsi que les métadonnées de carte telles que les coordonnées, l'app associée de l'émetteur de la carte et les fonctionnalités prises en charge. Il contient également l'état du billet qui comprend des informations indiquant par exemple si la personnalisation de Secure Element est terminée, si la carte est actuellement suspendue par l'organisme émetteur ou si une vérification supplémentaire est nécessaire pour que la carte puisse servir à effectuer des paiements avec Apple Pay.

Ajout d'une carte bancaire à Apple Pay à partir d'un compte iTunes Store

Pour les cartes bancaires sur fichier dans iTunes, l'utilisateur est parfois invité à saisir à nouveau son mot de passe d'Identifiant Apple. Le numéro de carte est obtenu via iTunes et le processus de vérification de carte est lancé. Si la carte est admissible pour Apple Pay, l'appareil télécharge et affiche les conditions d'utilisation, puis les envoie avec l'identifiant des conditions et le code de sécurité de la carte au processus de liaison et transfert. Une vérification supplémentaire peut être effectuée pour les cartes liées à un compte iTunes.

Ajout d'une carte bancaire depuis l'app d'un émetteur de carte

Lorsque l'app est inscrite pour être exploitée avec Apple Pay, les clés sont établies pour l'app et le serveur du vendeur. Ces clés servent à chiffrer les informations de la carte qui sont envoyées au vendeur, ce qui empêche l'appareil iOS de lire ces informations. Le flux de transfert ressemble à celui des cartes ajoutées manuellement, décrit ci-avant, hormis que des mots de passe à usage unique sont utilisés au lieu de codes CVV.

Vérification supplémentaire

L'émetteur de la carte peut décider si une carte nécessite une vérification supplémentaire. En fonction des services offerts par l'émetteur de la carte, l'utilisateur peut choisir entre différentes options de vérification supplémentaires, telles qu'un SMS, un e-mail, un appel au service client ou une procédure intégrée à une app de tierce partie agréée. Pour la vérification par SMS ou par courrier électronique, l'utilisateur choisit une adresse ou un numéro dans les coordonnées figurant dans les dossiers de l'émetteur. Un code est alors envoyé à l'utilisateur qui doit ensuite le saisir dans Wallet (disponible dans Réglages) ou dans l'app Apple Watch. Pour la vérification ou le service client à l'aide d'une app, l'émetteur utilise son propre processus de communication.

Autorisation du paiement

Sur les appareils dotés de Secure Enclave, Secure Element n'autorise le paiement qu'après avoir reçu l'autorisation de Secure Enclave. Sur l'iPhone ou l'iPad, cela suppose la confirmation que l'utilisateur s'est authentifié par Touch ID ou le code de l'appareil. Touch ID est la méthode par défaut si disponible, mais il est possible d'utiliser à tout moment la vérification par code à la place de Touch ID. La vérification par code est automatiquement proposée après trois tentatives infructueuses de reconnaissance d'empreinte digitale et exigée après la cinquième tentative infructueuse. Un code est également exigé si la fonctionnalité Touch ID n'est pas configurée ou n'a pas été activée pour Apple Pay. Sur l'Apple Watch, l'appareil doit être déverrouillé à l'aide du code et l'utilisateur doit double-cliquer sur le bouton latéral pour réaliser un paiement.

La communication entre Secure Enclave et Secure Element est effectuée via une interface série, Secure Element étant connecté au contrôleur NFC lui-même connecté au processeur d'application. Même s'ils ne sont pas directement connectés, Secure Enclave et Secure Element peuvent communiquer de manière sécurisée à l'aide d'une clé de jumelage partagée fournie durant le processus de fabrication. Le chiffrement et l'authentification de la communication reposent sur la norme standard AES, et des nonces de chiffrement sont utilisées des deux côtés pour assurer la protection contre les attaques par replay (« replay attacks »). La clé de jumelage est générée à l'intérieur de Secure Enclave, à partir de sa clé d'identification et de l'identifiant unique de Secure Element. Cette clé de jumelage est ensuite transférée de manière sécurisée depuis Secure Enclave en usine jusqu'à un module de sécurité matériel (HSM) qui dispose du matériel nécessaire pour injecter ensuite la clé de jumelage dans Secure Element.

Lorsque l'utilisateur autorise une transaction, Secure Enclave envoie à Secure Element des données signées relatives au type d'authentification ainsi que des détails concernant le type de transaction (sans contact ou au sein d'apps), le tout lié à une valeur d'autorisation aléatoire AR (Authorization Random). La valeur AR est générée dans Secure Enclave lorsqu'un utilisateur transfère pour la première fois une carte bancaire et est conservée tant qu'Apple Pay est activé ; elle est protégée par le chiffrement et le mécanisme anti-rollback de Secure Enclave. Elle est transmise de manière sécurisée à Secure Element par le biais de la clé de jumelage. À la réception d'une nouvelle valeur AR, Secure Element marque toutes les cartes précédemment ajoutées comme supprimées.

Les cartes bancaires et prépayées ajoutées à Secure Element ne peuvent être utilisées que si une autorisation est présentée à Secure Element au moyen de la clé de jumelage et de la valeur AR utilisées lors de l'ajout de la carte. Cela permet à iOS d'ordonner à Secure Enclave de rendre des cartes inutilisables en marquant la copie de la valeur AR en sa possession comme invalide dans les cas suivants :

- lorsque le code est désactivé ;
- l'utilisateur se déconnecte d'iCloud ;
- l'utilisateur sélectionne Effacer contenu et réglages ;
- l'appareil est restauré à partir du mode de récupération.

Avec l'Apple Watch, les cartes sont signalées comme non valides lorsque :

- le code de l'Apple Watch est désactivé ;
- l'Apple Watch n'est plus jumelée avec l'iPhone ;
- la détection du poignet est désactivée.

Avec la clé de jumelage et sa copie de la valeur AR actuelle, Secure Element vérifie l'autorisation envoyée par Secure Enclave avant d'activer l'« applet » de paiement pour effectuer un paiement sans contact. Ce processus est également utilisé pour récupérer des données de paiement chiffrées à partir d'une « applet » de paiement pour des transactions effectuées dans des apps.

Code de sécurité dynamique propre à la transaction

Toutes les transactions de paiement provenant d'« applets » de paiement comprennent un code de sécurité dynamique propre à la transaction ainsi qu'un numéro de compte d'appareil. Ce code à usage unique est calculé au moyen d'un compteur dont la valeur augmente d'une unité à chaque nouvelle transaction, ainsi que d'une clé fournie dans l'« applet » de paiement durant la personnalisation et connue du réseau de paiement et/ou de l'émetteur. En fonction du système de paiement, il est possible d'utiliser d'autres données pour calculer ces codes, y compris :

- un nombre aléatoire généré par l'« applet » de paiement ;
- un autre nombre aléatoire généré par le terminal (en cas de transaction NFC) ; ou
- un autre nombre aléatoire généré par le serveur (en cas de transaction effectuée dans des apps).

Ces codes de sécurité sont fournis au réseau de paiement et à l'émetteur de la carte, permettant à ces derniers de vérifier chaque transaction. La longueur des codes de sécurité peut varier en fonction du type de transaction.

Paiements sans contact avec Apple Pay

Lorsque l'iPhone est activé et qu'il détecte un champ NFC, il présente à l'utilisateur la carte bancaire ou prépayée adéquate ou la carte par défaut, ce qui est géré dans Réglages. L'utilisateur peut également accéder à l'app Wallet et choisir une carte bancaire ou, si l'appareil est verrouillé, appuyer deux fois sur le bouton principal.

L'utilisateur doit ensuite s'authentifier via Touch ID ou son code pour que les données de paiement soient transmises. Lorsque l'Apple Watch est déverrouillée, un double appui sur le bouton latéral permet d'activer la carte par défaut pour le paiement. Aucune donnée de paiement ne peut être envoyée sans authentification de l'utilisateur.

Une fois que l'utilisateur a été authentifié, le numéro de compte d'appareil et un code de sécurité dynamique propre à la transaction sont utilisés lors du traitement du paiement. Ni Apple, ni l'appareil de l'utilisateur n'envoie les numéros complets de carte bancaire aux vendeurs. Apple peut recevoir des données de transaction anonymes, telles que l'heure et le lieu approximatifs de la transaction, destinées à améliorer Apple Pay et d'autres produits et services Apple.

Utilisation d'Apple Pay pour effectuer des paiements dans les apps

Apple Pay permet aussi d'effectuer des paiements dans des apps iOS, ainsi que dans des apps Apple Watch depuis watchOS 3. Lorsque des utilisateurs effectuent des paiements depuis des apps à l'aide d'Apple Pay, Apple reçoit des données de transaction chiffrées qu'elle chiffre à nouveau au moyen d'une clé propre à chaque développeur ou vendeur avant de les envoyer à ce dernier. Apple Pay conserve des données de transaction anonymes comme le montant approximatif de l'achat. Ces données ne peuvent être associées à un utilisateur spécifique et n'incluent aucune information sur le contenu des achats.

Lorsqu'une app lance une transaction de paiement Apple Pay, les serveurs Apple Pay reçoivent la transaction chiffrée de l'appareil avant le vendeur. Les serveurs Apple Pay chiffrent à nouveau ces données au moyen d'une clé propre au vendeur avant de transmettre la transaction à ce dernier.

Lorsqu'une app demande un paiement, elle appelle une API pour déterminer si l'appareil prend en charge Apple Pay et si l'utilisateur possède des cartes bancaires capables d'effectuer des paiements sur les réseaux de paiement acceptés par le vendeur. L'app demande les éléments d'information dont elle a besoin pour traiter et terminer la transaction (coordonnées et adresses d'expédition et de facturation, par exemple). L'app demande ensuite à iOS de présenter la feuille Apple Pay qui demande les informations relatives à l'app, ainsi que d'autres informations nécessaires, telles que la carte à utiliser.

L'app reçoit alors les informations relatives à la ville, à la région et au code postal nécessaires pour calculer les frais d'expédition. L'ensemble des informations demandées n'est transmis à l'app que lorsque l'utilisateur a autorisé le paiement via Touch ID ou en saisissant le code de l'appareil. Une fois le paiement autorisé, les informations figurant sur le formulaire Apple Pay sont envoyées au vendeur.

Lorsque l'utilisateur autorise le paiement, un appel est effectué auprès des serveurs Apple Pay en vue d'obtenir un nonce cryptographique similaire à la valeur renvoyée par le terminal NFC utilisé pour les transactions en magasin. Le nonce ainsi que d'autres données de transaction sont transmises à Secure Element afin de générer une accréditation de paiement destinée à être chiffrée à l'aide d'une clé Apple. Une fois l'accréditation de paiement chiffrée générée par Secure Element, elle est transmise aux serveurs Apple Pay qui la déchiffrent, comparent le nonce inclus dans l'accréditation au nonce envoyé par Secure Element, puis effectuent un nouveau chiffrement de cette accréditation de paiement à l'aide de la clé de vendeur associée à l'identifiant

du vendeur. Elle est ensuite renvoyée à l'appareil qui la remet à l'app via l'API. L'app la transfère alors au système du vendeur en vue de son traitement. Le vendeur peut ainsi déchiffrer l'accréditation de paiement à l'aide de sa clé privée afin de la traiter. Grâce à ces informations et à la signature provenant des serveurs d'Apple, le vendeur peut vérifier que la transaction lui était bien destinée.

Les API requièrent un droit spécifiant les identifiants de vendeur pris en charge. Une app peut également inclure des données supplémentaires à envoyer à Secure Element en vue de leur signature, comme le numéro de commande ou l'identité du client, afin de veiller à ce que la transaction ne puisse pas être détournée vers un autre client. Cette tâche est effectuée par le développeur de l'app. Ce dernier est capable de spécifier les données d'application (applicationData) de la demande de paiement (PKPaymentRequest). Un hachage de cette donnée est inclus dans les données de paiement chiffrées. Le vendeur doit ensuite vérifier que le hachage de ses données d'application correspond à ce qui se trouve dans les données de paiement.

Utilisation d'Apple Pay pour effectuer des paiements sur le web

Apple Pay permet d'effectuer des paiements sur des sites web. Sous iOS 10, il est possible de réaliser des transactions Apple Pay sur le web depuis l'iPhone et l'iPad. En outre, dans macOS Sierra, les transactions Apple Pay peuvent commencer sur un Mac et se terminer sur un iPhone ou une Apple Watch compatible Apple Pay par le biais du même compte iCloud.

Apple Pay sur le web oblige tous les sites web participants à s'inscrire auprès d'Apple. Les serveurs d'Apple réalisent la validation du nom de domaine et émettent un certificat client TLS. Les sites web prenant en charge Apple Pay doivent servir leur contenu par HTTPS. Pour chaque transaction de paiement, les sites web doivent obtenir une session vendeur sécurisée et unique à travers un serveur Apple émettant le certificat client TLS. Les données de session vendeur sont signées par Apple. Une fois une signature de session vendeur est vérifiée, un site web peut demander si l'utilisateur possède un appareil compatible Apple Pay et si sa carte bancaire ou prépayée est activée sur celui-ci. Aucun autre détail n'est partagé. Si l'utilisateur ne veut pas partager ces informations, il peut désactiver les requêtes Apple Pay dans les réglages de confidentialité de Safari sous macOS et iOS.

Une fois une session vendeur validée, toutes les mesures de sécurité et de confidentialité sont identiques à celles du paiement dans une app.

En cas de Handoff du Mac à l'iPhone ou à l'Apple Watch, Apple Pay exploite le protocole IDS chiffré de bout-en-bout pour transmettre les informations en relation avec le paiement entre le Mac de l'utilisateur et l'appareil effectuant l'autorisation. IDS fait appel aux clés de l'appareil de l'utilisateur pour procéder au chiffrement de sorte qu'aucun autre appareil ne puisse déchiffrer ces informations, et les clés ne sont pas visibles par Apple. La découverte d'appareils pour le Handoff Apple Pay contient le type et l'identifiant unique des cartes bancaires de l'utilisateur accompagnés de certaines métadonnées. Le numéro de compte spécifique à l'appareil de la carte de l'utilisateur n'est pas partagé et reste stocké de façon sécurisée sur l'iPhone ou l'Apple Watch de l'utilisateur. Apple transfère également de façon sécurisée les adresses de contact, d'expédition et de facturation récemment employées par l'utilisateur à travers le trousseau iCloud.

Lorsque l'utilisateur autorise le paiement grâce à leur Touch ID ou code sur l'iPhone ou double-clique sur la touche latérale de l'Apple Watch, un jeton de paiement chiffré de façon unique pour chaque certificat de vendeur du site web est transmis de façon sécurisée de l'iPhone ou Apple Watch de l'utilisateur à son Mac puis livré au site web du vendeur.

Seuls les appareils à proximité l'un de l'autre peuvent demander et effectuer un paiement. La proximité est déterminée par publications Bluetooth Low Energy.

Cartes de fidélité

Depuis iOS 9, Apple Pay prend en charge le protocole VAS (service à valeur ajoutée) pour la transmission de cartes de fidélité de vendeurs à des terminaux NFC compatibles. Le protocole VAS peut être mis en œuvre sur les bornes de vendeurs, et exploite la technologie NFC pour communiquer avec les appareils Apple pris en charge. Le protocole VAS fonctionne sur une courte distance et sert à fournir des services complémentaires, tels que la transmission d'informations sur les cartes de fidélité, de façon intégrée à une transaction Apple Pay.

La borne NFC engage le processus de réception des informations sur la carte en envoyant la demande d'une carte. Si l'utilisateur possède une carte avec l'identifiant du magasin, il lui est demandé d'autoriser son usage. Si le vendeur prend en charge le chiffrement, les informations sur la carte, un horodatage et une clé ECDH P-256 aléatoire à usage unique est utilisée conjointement avec la clé publique du vendeur pour calculer une clé de chiffrement pour les données de la carte, lesquelles sont envoyées à la borne. Si le vendeur ne prend pas en charge le chiffrement, l'utilisateur doit représenter l'appareil à la borne pour que les informations de carte de fidélité soient envoyées.

Suspension, retrait et suppression de cartes

Les utilisateurs ont la possibilité de suspendre Apple Pay sur l'iPhone, l'iPad et l'Apple Watch fonctionnant sous watchOS 3 ou ultérieur, en plaçant leur appareil en mode Perdu à travers la fonctionnalité Localiser mon iPhone. Ils peuvent également retirer et supprimer leurs cartes d'Apple Pay à l'aide de la fonctionnalité Localiser mon iPhone, d'iCloud.com ou directement sur leur appareil à l'aide de Wallet. Sur l'Apple Watch, les cartes peuvent être supprimées à l'aide des réglages iCloud, via l'app Apple Watch sur l'iPhone ou directement sur la montre. L'émetteur de la carte ou le réseau de paiement concerné suspend ou supprime alors la possibilité d'effectuer des paiements avec les cartes par Apple Pay sur l'appareil, même si celui-ci est hors ligne et n'est pas connecté à un réseau cellulaire ou Wi-Fi. Les utilisateurs peuvent également appeler l'émetteur de leur carte pour suspendre ou retirer des cartes d'Apple Pay.

Par ailleurs, lorsqu'un utilisateur efface l'intégralité du contenu de son appareil à l'aide de la commande « Effacer contenu et réglages », via Localiser mon iPhone ou en restaurant son appareil en mode de récupération, iOS demande à Secure Element de marquer toutes les cartes comme supprimées. Cela rend immédiatement toutes les cartes inutilisables jusqu'à ce que les serveurs Apple Pay puissent être contactés afin de supprimer complètement les cartes dans Secure Element. Parallèlement, Secure Enclave marque la valeur AR comme étant invalide de sorte qu'il ne soit plus possible d'autoriser de paiement avec des cartes précédemment enregistrées. Une fois en ligne, l'appareil essaie de contacter les serveurs Apple Pay pour s'assurer que toutes les cartes présentes dans Secure Element sont effacées.

Services Internet

Création de mots de passe d'identifiant Apple complexes
Les Identifiants Apple sont utilisés pour se connecter à différents services comme iCloud, FaceTime et iMessage. Pour aider les utilisateurs à créer des mots de passe complexes, tous les nouveaux comptes doivent contenir les attributs de mot de passe suivants :

- au moins huit caractères ;
- au moins une lettre ;
- au moins une lettre majuscule ;
- au moins un chiffre ;
- pas plus de trois caractères identiques consécutifs ;
- différent du nom du compte.

Apple a créé une série de services fiables destinés à rendre les appareils de ses utilisateurs encore plus pratiques et productifs ; ces services comprennent notamment iMessage, FaceTime, Siri, Suggestions Spotlight, iCloud, Sauvegarde iCloud et Trousseau iCloud.

Ces services Internet ont été développés avec les mêmes objectifs de sécurité que ceux d'iOS mis en avant à travers toute la plateforme. Ces objectifs incluent la manipulation sécurisée des données, que ce soit au sein des appareils ou lors de leur transfert à travers des réseaux sans fil ; la protection des données personnelles des utilisateurs ; et la protection contre tout accès malveillant ou non autorisé aux informations et aux services. Chaque service utilise sa propre architecture de sécurité performante sans compromettre la facilité d'utilisation générale du système iOS.

Identifiant Apple

Un Identifiant Apple correspond à un compte utilisé pour se connecter à des services comme iCloud, iMessage, FaceTime, l'iTunes Store, l'iBooks Store, l'App Store et bien plus encore. Il est essentiel que chaque utilisateur protège son Identifiant Apple afin d'éviter tout accès non autorisé à ses comptes. Pour cela, Apple conseille d'utiliser des mots de passe complexes composés d'au moins huit caractères, comprenant à la fois des lettres et des chiffres, n'incluant pas plus de trois caractères identiques consécutifs et ne correspondant à aucun mot de passe actuellement utilisé. Les utilisateurs sont encouragés à aller au-delà de ces recommandations en ajoutant des caractères et des signes de ponctuation pour renforcer leurs mots de passe. Apple oblige également les utilisateurs à créer trois questions de sécurité permettant de vérifier l'identité du propriétaire en cas de modification de ses informations de compte ou de réinitialisation d'un mot de passe oublié.

Apple envoie également à ses utilisateurs des messages électroniques et des notifications lorsque des modifications importantes sont apportées à leur compte, par exemple en cas de changement de mot de passe ou de données de facturation, ou encore d'utilisation de l'Identifiant Apple pour se connecter à un nouvel appareil. Les utilisateurs sont de plus invités à changer le mot de passe de leur Identifiant Apple dès qu'ils remarquent quoi que ce soit d'inhabituel.

En outre, Apple emploie une gamme étendue de règlements et de procédures conçus pour protéger les comptes utilisateur. Parmi ces dispositions, l'on retrouve la limitation du nombre de tentatives d'ouverture de session et de réinitialisation du mot de passe, le contrôle actif antifraude pour aider à identifier les attaques dès qu'elles se produisent, ainsi que des passages en revue routiniers des règlements pour nous aider à nous adapter à toute nouvelle information susceptible d'affecter la sécurité des clients.

Identification à deux facteurs

Pour aider les utilisateurs à sécuriser davantage leur compte, Apple propose l'identification à deux facteurs. L'identification à deux facteurs est une couche de sécurité complémentaire pour les identifiants Apple. Elle est conçue dans le but de s'assurer que seul le propriétaire du compte peut accéder au compte, même si quelqu'un d'autre connaît le mot de passe.

Cette identification à deux facteurs permet à un utilisateur d'accéder à son compte uniquement sur des appareils de confiance, comme son iPhone, son iPad ou son Mac. Pour ouvrir une première session sur un nouvel appareil, deux informations sont obligatoires : le mot de passe de l'Identifiant Apple et un code de vérification à six chiffres automatiquement affiché sur les appareils de confiance de l'utilisateur ou envoyé à un numéro de téléphone de confiance. En saisissant le code, l'utilisateur confirme qu'il fait confiance au nouvel appareil et que celui-ci peut être utilisé pour ouvrir une session. Dans la mesure où un mot de passe seul n'est plus suffisant pour accéder au compte d'un utilisateur, l'identification à deux facteurs améliore la sécurité de l'Identifiant Apple de l'utilisateur et de l'intégralité des informations confidentielles qu'il confie à Apple pour les stocker.

L'identification à deux facteurs améliore la sécurité des identifiants Apple des utilisateurs et des informations confidentielles qu'ils confient à Apple pour les stocker. Cette technique est directement intégrée à iOS, à macOS, à tvOS, à watchOS et aux systèmes d'authentification employés par les sites web d'Apple.

Pour en savoir plus sur l'identification à deux facteurs, consultez https://support.apple.com/fr-fr/HT204915.

Validation en deux étapes

Apple propose en outre depuis 2013 un moyen sécurisé similaire appelé validation en deux étapes. Lorsque cette méthode est activée, l'identité de l'utilisateur doit être vérifiée au moyen d'un code temporaire envoyé à l'un de ses appareils de confiance, avant d'autoriser toute modification des informations de compte liées à son Identifiant Apple, toute connexion à iCloud, à iMessage, à FaceTime et au Game Center, ou avant tout achat sur l'iTunes Store, l'iBooks Store ou l'App Store à partir d'un nouvel appareil. Les utilisateurs disposent également d'une clé de récupération de 14 caractères à conserver en lieu sûr en cas d'oubli de leur mot de passe ou de perte d'accès à leurs appareils de confiance. Pour en savoir plus sur la validation d'Identifiant Apple en deux étapes, consultez https://support.apple.com/fr-fr/HT5570.

Identifiants Apple Gérés

Avec iOS 9.3 ou ultérieur, les Identifiants Apple Gérés fonctionnent de façon similaire aux identifiants Apple, mais sont détenus et contrôlés par un établissement d'enseignement. L'établissement peut réinitialiser les mots de passe, limiter les achats et les communications, par exemple par FaceTime et Messages, et configurer des autorisations s'appuyant sur des rôles pour les membres du personnel, les professeurs et les élèves.

Certains services Apple sont désactivés pour les Identifiants Apple Gérés, comme Touch ID, Apple Pay, le trousseau iCloud, HomeKit et Localiser mon iPhone.

Pour en savoir plus sur les identifiants Apple gérés, consultez la page https://help.apple.com/schoolmanager/.

Audit des Identifiants Apple Gérés

Les Identifiants Apple Gérés prennent également en charge l'audit, ce qui permet aux établissements de respecter la réglementation en vigueur et les règles de confidentialité. Les comptes administrateur, enseignant ou gestionnaire peuvent se voir accorder des privilèges d'audit pour certains Identifiants Apple Gérés. Les auditeurs sont en mesure de contrôler uniquement les comptes qui se trouvent sous eux dans la hiérarchie de l'école. En d'autres termes, les enseignants peuvent surveiller les élèves, les gestionnaires peuvent auditer les enseignants et les élèves, et les administrateurs peuvent effectuer l'audit des gestionnaires, des enseignants et des élèves.

Lorsque des informations d'identification d'audit sont demandées par le biais d'Apple School Manager, un compte particulier est émis dont l'accès est limité à l'Identifiant Apple Géré pour lequel l'audit est demandé. L'autorisation d'audit expire au bout de sept jours. Au cours de cette période, l'auditeur peut lire et modifier le contenu de l'utilisateur stocké sur iCloud ou dans les applications compatibles CloudKit. Chaque demande d'accès à l'audit est consignée dans Apple School Manager. Les journaux indiquent qui est l'auditeur, l'Identifiant Apple Géré auquel l'auditeur a demandé l'accès, l'heure de la demande et si l'audit a été réalisé.

Identifiants Apple Gérés et appareils personnels

Les Identifiants Apple Gérés peuvent également être utilisés avec des appareils iOS personnels. Les élèves ouvrent une session iCloud avec l'Identifiant Apple Géré émis par l'établissement et un autre mot de passe à usage personnel faisant office de deuxième facteur lors du processus d'identification à deux facteurs pour l'identifiant Apple. Lors de l'utilisation d'un Identifiant Apple Géré sur un appareil personnel, le trousseau iCloud est indisponible et l'établissement peut restreindre d'autres fonctionnalités, comme FaceTime ou Messages. Tout document iCloud créé par des élèves lorsque leur session est active sont soumis à audit comme décrit précédemment

iMessage

L'application iMessage d'Apple est un service de messagerie pour appareils iOS et ordinateurs Mac. iMessage prend en charge aussi bien le texte que les pièces jointes telles que les photos, les contacts et les lieux. Les messages apparaissent sur tous les appareils enregistrés d'un utilisateur, de telle sorte qu'une conversation entamée sur un appareil puisse être poursuivie sur n'importe quel autre appareil. iMessage utilise le service de notification Push d'Apple (Apple Push Notification, ou APN) de manière intensive. Apple ne conserve ni les messages ni les pièces jointes, et leur contenu est protégé par un système de chiffrement de bout en bout, afin que personne d'autre que l'expéditeur et le destinataire ne puisse y accéder. Apple est incapable de déchiffrer ces données.

Lorsqu'un utilisateur active iMessage sur un appareil, ce dernier génère deux paires de clés à utiliser avec le service : une clé RSA 1280 bits pour le chiffrement et une clé ECDSA 256 bits sur la courbe NIST P-256 pour la signature. Les clés privées des deux paires de clés sont enregistrées dans le trousseau de l'appareil, tandis que les clés publiques sont envoyées au service de répertoire Apple (IDS) où elles sont associées au numéro de téléphone ou à l'adresse électronique de l'utilisateur, ainsi qu'à l'adresse du service APN de l'appareil.

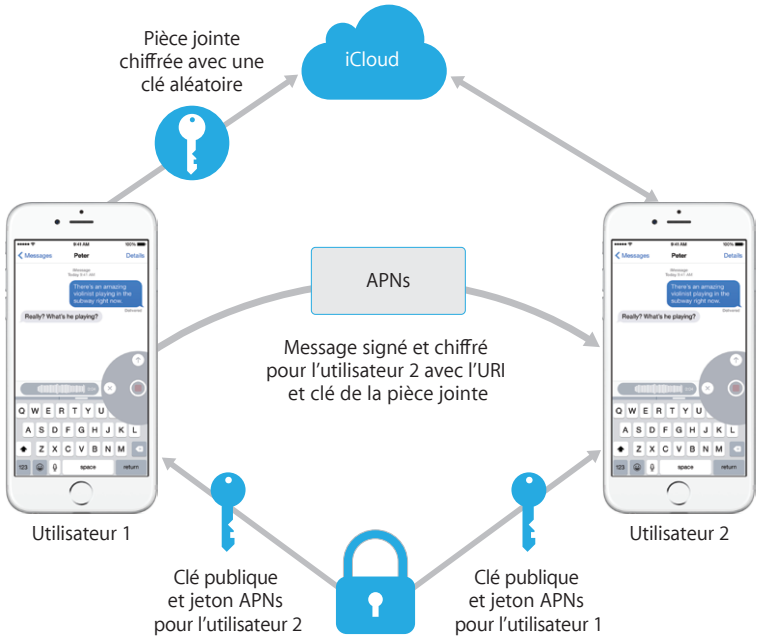
Au fur et à mesure que les utilisateurs activent des appareils supplémentaires à utiliser avec iMessage, leurs clés publiques de chiffrement et de signature, les adresses de service APN et les numéros de téléphone associés sont ajoutés au service de répertoire. Les utilisateurs ont également la possibilité d'ajouter des adresses électroniques qui seront vérifiées au moyen d'un lien de confirmation. Les numéros de téléphone sont vérifiés via la carte SIM et le réseau de l'opérateur. De plus, tous les appareils enregistrés de l'utilisateur affichent un message d'alerte dès qu'une nouvelle adresse électronique, un nouvel appareil ou un nouveau numéro de téléphone est ajouté.

Envoi et réception des messages par iMessage

L'utilisateur lance une nouvelle conversation iMessage en saisissant une adresse ou un nom. S'il saisit un numéro de téléphone ou une adresse électronique, l'appareil entre en contact avec le service IDS pour récupérer les clés publiques et les adresses de service APN de tous les appareils associés au destinataire. Si l'utilisateur saisit un nom, l'appareil utilise d'abord l'app Contacts de l'utilisateur pour récupérer les numéros de téléphone et les adresses électroniques associés à ce nom, puis récupère les clés publiques et les adresses de service APN via le service IDS.

Le message sortant envoyé par l'utilisateur est chiffré individuellement pour chacun des appareils du destinataire. Les clés de chiffrement RSA publiques des appareils destinataires sont récupérées via le service IDS. Pour chaque appareil destinataire, l'appareil expéditeur génère une valeur aléatoire sur 88 bits et l'utilise comme clé HMAC-SHA256 pour élaborer une valeur sur 40 bits dérivée de la clé publique de l'expéditeur et du destinataire et de texte au format simple. La concaténation des valeurs 88 bits et 40 bits produit une clé de 128 bits qui permet de chiffrer le message par la méthode AES en mode CTR. La valeur 40 bits est employée par le destinataire pour vérifier l'intégrité du texte au format simple déchiffré. Cette clé AES propre à chaque message est chiffrée via RSA-OAEP vers la clé publique de l'appareil destinataire. La combinaison constituée du texte du message chiffré et de la clé de message chiffrée est ensuite hachée avec l'algorithme SHA-1 et le hachage est signé avec l'algorithme ECDSA à l'aide de la clé de signature privée de l'appareil expéditeur. Les messages résultants (un pour chaque appareil destinataire) sont constitués du texte de message chiffré, de la clé de message chiffrée et de la signature numérique de l'expéditeur. Ils sont ensuite transmis au service APN en vue de leur livraison. Les métadonnées, comme le code temporel et les informations de routage du service APN, ne sont pas chiffrées. La communication avec le service APN est chiffrée par l'intermédiaire d'un canal TLS à confidentialité persistante.

Le service APN ne peut relayer que des messages de 4 ko ou 16 ko, selon la version d'iOS. Si le texte du message est trop long ou qu'une pièce jointe (telle qu'une photo) est incluse, la pièce jointe est chiffrée par AES en mode CTR à l'aide d'une clé 256 bits générée aléatoirement et téléchargée vers iCloud. La clé AES destinée à la pièce jointe, son URI (Uniform Resource Identifier) et un hachage SHA-1 de sa forme chiffrée sont ensuite envoyés au destinataire en tant que contenu de message iMessage, la confidentialité et l'intégrité de ces éléments étant protégées par un chiffrement iMessage normal (voir illustration ci-dessous).



Dans le cas d'une conversation de groupe, ce processus est répété pour chaque destinataire et ses appareils.

Du côté destinataire, chaque appareil reçoit sa copie du message par le biais du service APN et, si nécessaire, récupère la pièce jointe sur iCloud. L'adresse électronique de l'expéditeur ou le numéro de téléphone de l'appelant est comparé aux données figurant dans les contacts du destinataire, afin de pouvoir afficher un nom si possible.

Comme pour toutes les notifications de type Push, le message est supprimé du service APN dès sa livraison. Contrairement à d'autres notifications du service APN, les messages iMessage sont placés en file d'attente en attendant d'être livrés à des appareils déconnectés. Les messages sont actuellement conservés pendant une durée maximale de 30 jours.

FaceTime

FaceTime est le service d'appels audio et vidéo d'Apple. À l'instar d'iMessage, les appels FaceTime utilisent également le service de notifications Push d'Apple (APN) pour établir une première connexion aux appareils enregistrés de l'utilisateur. Le contenu audio/vidéo des appels FaceTime est protégé au moyen d'un chiffrement de bout en bout qui interdit l'accès à toute autre personne que l'expéditeur et le destinataire. Apple est incapable de déchiffrer ces données.

FaceTime exploite le NAT transversal (STUN) et la technique d'établissement de connexion Internet (ICE) pour établir les connexions de pair-à-pair entre les appareils. Par le biais de notifications APN et de messages STUN, les appareils vérifient leurs certificats d'identification et établissent un secret partagé pour chaque session. Les nonces cryptographiques fournis par chaque appareil sont combinés pour : les « salt keys » (clés salées) destinées à chacun des canaux de support, qui sont ensuite diffusées à travers le protocole SRTP (Secure Real Time Protocol) avec un chiffrement AES-256.

iCloud

iCloud est utilisé pour stocker les contacts, les calendriers, les photos, les documents et d'autres données d'un utilisateur, et tenir automatiquement ces informations à jour sur tous les appareils de cet utilisateur. iCloud peut également être utilisé par des apps de tierce partie pour stocker et synchroniser des documents ainsi que des valeurs de clé de données d'application définies par le développeur. Chaque utilisateur configure son espace iCloud en se connectant au moyen d'un identifiant Apple et en choisissant les services qu'il souhaite utiliser. Les fonctionnalités iCloud, telles que Mon flux de photos, iCloud Drive et Sauvegarde, peuvent être désactivées par des administrateurs informatiques via un profil de configuration. Le service ne tient pas compte du contenu stocké et traite le contenu de tous les fichiers de la même manière, comme s'il s'agissait de simples regroupements d'octets.

iCloud divise chaque fichier en plusieurs blocs qu'il chiffre à l'aide de l'algorithme AES-128 et d'une clé SHA-256 dérivée du contenu de chaque bloc. Les clés et les métadonnées du fichier sont stockées par Apple dans le compte iCloud de l'utilisateur. Les blocs chiffrés du fichier sont stockés, sans aucune information susceptible d'identifier l'utilisateur, dans des services de stockage tiers comme Amazon S3 et Windows Azure.

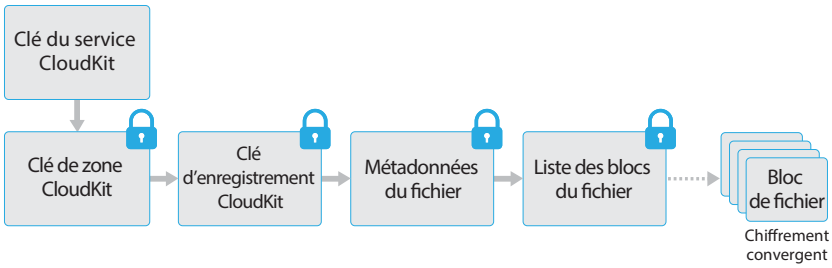
iCloud Drive

iCloud Drive ajoute des clés basées sur le compte pour protéger les documents enregistrés dans iCloud. À l'instar des autres services iCloud, il divise le contenu des fichiers en plusieurs blocs qu'il chiffre avant de stocker les blocs chiffrés par le biais de services tiers. Les clés de contenu de fichier sont toutefois enveloppées par des clés d'enregistrement stockées avec les métadonnées iCloud Drive. Ces clés d'enregistrement sont à leur tour protégées par la clé de service iCloud Drive de l'utilisateur qui est ensuite stockée avec le compte iCloud de l'utilisateur. Les utilisateurs ont accès aux métadonnées de leurs documents iCloud en s'authentifiant auprès du service iCloud, mais ils doivent également disposer de la clé de service iCloud Drive pour exposer les parties protégées du service de stockage iCloud Drive.

CloudKit

CloudKit permet aux développeurs d'apps d'enregistrer des données de valeur de clé, des données structurées et des ressources dans iCloud. L'accès à CloudKit est contrôlé au moyen de droits d'app. CloudKit prend en charge les bases de données publiques et les bases de données privées. Les bases de données publiques sont utilisées par toutes les copies de l'app, habituellement pour des ressources générales, et ne sont pas chiffrées. Les bases de données privées hébergent les données de l'utilisateur.

Comme avec iCloud Drive, CloudKit utilise des clés basées sur le compte pour protéger les informations stockées dans la base de données privée de l'utilisateur et, comme le font d'autres services iCloud, les fichiers sont divisés en blocs, chiffrés et stockés par le biais de services tiers. CloudKit utilise une hiérarchie de clés comme pour la protection des données. Les clés de fichier sont enveloppées par des clés d'enregistrement CloudKit. Ces dernières sont à leur tour protégées par une clé de zone, elle-même protégée par la clé de service CloudKit de l'utilisateur. La clé de service CloudKit est stockée dans le compte iCloud de l'utilisateur et n'est disponible qu'une fois que ce dernier s'est authentifié sur iCloud.



Sauvegarde iCloud

iCloud permet également de sauvegarder quotidiennement des informations (telles que les réglages d'appareil, les données d'app, les photos et vidéos de la Pellicule, ainsi que les conversations de l'app Messages) via Wi-Fi. iCloud protège le contenu en le chiffrant lorsqu'il est envoyé via Internet, en le stockant dans un format chiffré et en utilisant des jetons sécurisés pour l'authentification. La sauvegarde iCloud n'est effectuée que si l'appareil est verrouillé, branché sur une source d'alimentation et connecté à Internet via Wi-Fi. En raison du type de chiffrement utilisé dans iOS, le système est conçu pour protéger les données tout en autorisant des sauvegardes et des restaurations incrémentales et sans surveillance.

Données sauvegardées par iCloud :

- enregistrements des morceaux, des films, des programmes télévisés, des apps et des livres achetés : la copie de sauvegarde iCloud d'un utilisateur comprend des informations relatives au contenu acheté présent sur l'appareil iOS de l'utilisateur et non sur le contenu acheté même. Lorsque l'utilisateur restaure à partir d'une copie de sauvegarde iCloud, son contenu acheté est automatiquement téléchargé depuis l'iTunes Store, l'App Store ou l'iBooks Store. Certains types de contenu ne sont pas automatiquement téléchargés dans tous les pays, et les achats précédents peuvent être indisponibles s'ils ont été remboursés ou ne sont plus disponibles dans le store. L'historique complet des achats est associé à l'identifiant Apple d'un utilisateur ;
- photos et vidéos sur les appareils iOS d'un utilisateur (remarque : si un utilisateur active la photothèque iCloud sur son appareil iOS 8.1 ou ultérieur ou Mac OS X 10.10.3 ou ultérieur, ses photos et vidéos sont déjà stockées dans iCloud de sorte qu'elles ne sont pas incluses dans la sauvegarde iCloud de l'utilisateur) ;
- contacts, événements de calendrier, rappels et notes ;
- réglages d'appareil ;
- données d'app ;
- historique des appels téléphoniques ;
- organisation de l'écran d'accueil et des apps ;
- iMessage, SMS et MMS (requiert la carte SIM utilisée lors de la sauvegarde) ;
- sonneries ;
- code secret de messagerie visuelle (requiert la carte SIM utilisée lors de la sauvegarde) ;
- configuration HomeKit ;
- données HealthKit.

Si des fichiers sont créés dans des classes de protection de données (Data Protection) inaccessibles lorsque l'appareil est verrouillé, leurs clés de fichier sont chiffrées à l'aide des clés de classe provenant du conteneur de clés de Sauvegarde iCloud. Les fichiers sont sauvegardés sur iCloud dans leur état chiffré d'origine. Les fichiers de la classe de protection de données (Data Protection) sans protection (No Protection) sont chiffrés durant le transfert.

Le conteneur de clés de Sauvegarde iCloud contient des clés asymétriques (Curve25519) pour chaque classe Data Protection, utilisées pour chiffrer les clés de fichier. Pour en savoir plus sur le contenu du conteneur de clés de sauvegarde et sur le conteneur de clés de Sauvegarde iCloud, consultez la partie intitulée « Protection des données du trousseau » de la section « Chiffrement et protection des données ».

La sauvegarde est stockée dans le compte iCloud de l'utilisateur et est constituée d'une copie de ses fichiers, ainsi que du conteneur de clés de Sauvegarde iCloud. Le conteneur de clés de Sauvegarde iCloud est protégé par une clé aléatoire également stockée avec la sauvegarde. (Le mot de passe iCloud de l'utilisateur n'est pas utilisé pour le chiffrement, afin que toute modification du mot de passe iCloud n'ait aucune incidence sur la validité des sauvegardes existantes.)

Bien que la base de données du trousseau de l'utilisateur soit sauvegardée sur iCloud, elle demeure protégée par une clé entremêlée avec l'UID. Cela permet de restaurer le trousseau uniquement sur son appareil d'origine, afin qu'aucune autre personne (y compris Apple) n'ait accès aux éléments du trousseau de l'utilisateur.

Lors de la restauration, les fichiers sauvegardés, le conteneur de clés de Sauvegarde iCloud et la clé du conteneur de clés sont récupérés à partir du compte iCloud de l'utilisateur. Le conteneur de clés de Sauvegarde iCloud est déchiffré au moyen de sa clé, les clés de fichier du conteneur de clés sont ensuite utilisées pour déchiffrer les fichiers de la sauvegarde qui sont ensuite écrits en tant que nouveaux fichiers dans le système de fichiers, ce qui a pour conséquence de les chiffrer à nouveau en fonction de leur classe de protection de données (Data Protection).

Trousseau iCloud

Intégration de Safari et du trousseau iCloud
Safari peut générer automatiquement des chaînes de caractères aléatoires, cryptographiquement complexes, à utiliser comme mots de passe de site web, stocker ces mots de passe dans le trousseau et les synchroniser avec vos autres appareils. Les éléments de trousseau sont transférés d'un appareil à l'autre en passant par des serveurs Apple, mais ils sont chiffrés de telle manière que leur contenu ne peut être lu ni par des appareils Apple, ni par des appareils tiers.

Le trousseau iCloud donne aux utilisateurs la possibilité de synchroniser de manière sécurisée leurs mots de passe entre plusieurs appareils iOS et ordinateurs Mac sans divulguer ces informations à Apple. Outre la volonté de fournir un niveau de confidentialité et de sécurité supérieur, d'autres objectifs ont fortement influencé la conception et l'architecture du trousseau iCloud, comme la facilité d'utilisation et la possibilité de restaurer des trousseaux. Le trousseau iCloud consiste en deux services : la synchronisation de trousseaux et la récupération de trousseau.

Apple a conçu le trousseau iCloud et la récupération de trousseau de telle sorte que les mots de passe d'un utilisateur demeurent protégés dans les situations suivantes :

- le compte iCloud de l'utilisateur est compromis ;
- iCloud a été compromis par un employé ou une attaque externe ;
- une tierce partie a accédé aux comptes de l'utilisateur.

Synchronisation de trousseaux

Lorsqu'un utilisateur active le trousseau iCloud pour la première fois, l'appareil établit un cercle de confiance et crée une identité de synchronisation pour lui-même. L'identité de synchronisation est constituée d'une clé privée et d'une clé publique. La clé publique de l'identité de synchronisation est placée dans le cercle et ce dernier est signé deux fois : une première fois avec la clé privée de l'identité de synchronisation et une deuxième fois avec une clé asymétrique sur courbe elliptique (via P256) dérivée du mot de passe du compte iCloud de l'utilisateur. Les paramètres utilisés pour créer la clé basée sur le mot de passe iCloud de l'utilisateur (random salt et itérations aléatoires) sont également stockés avec le cercle.

Le cercle de synchronisation signé est ensuite placé dans la zone de stockage des valeurs de clé iCloud de l'utilisateur. Il est impossible de lire ce cercle sans connaître le mot de passe iCloud de l'utilisateur et il ne peut être modifié de manière valide sans la clé privée de l'identité de synchronisation de son membre.

Lorsque l'utilisateur active le trousseau iCloud sur un autre appareil, ce dernier signale à iCloud que l'utilisateur possède un cercle de synchronisation précédemment établi dont il ne fait pas partie. L'appareil crée sa paire de clés d'identification de synchronisation, puis crée un ticket de candidature pour demander à devenir membre du cercle. Le ticket est constitué de la clé publique de l'identité de synchronisation de l'appareil ; l'utilisateur est invité à s'authentifier à l'aide de son mot de passe iCloud. Les paramètres de génération de clé sur courbe elliptique sont récupérés à partir d'iCloud et permettent d'obtenir une clé destinée à signer le ticket de candidature. Pour terminer, le ticket de candidature est placé dans iCloud.

Lorsque le premier appareil constate qu'un ticket de candidature est arrivé, il affiche un message invitant l'utilisateur à confirmer qu'un nouvel appareil demande à faire partie du cercle de synchronisation. L'utilisateur saisit son mot de passe iCloud et le ticket de candidature est vérifié pour confirmer qu'il a été signé par la clé privée appropriée. Cela permet d'établir que la personne à l'origine de la demande d'entrée dans le cercle a saisi le mot de passe iCloud de l'utilisateur au moment de la requête.

Lorsque l'utilisateur accepte d'ajouter le nouvel appareil au cercle, le premier appareil ajoute la clé publique du nouveau membre au cercle de synchronisation et la signe à nouveau avec son identité de synchronisation et la clé dérivée du mot de passe iCloud de l'utilisateur. Le nouveau cercle de synchronisation est alors placé dans iCloud où il est signé de la même manière par le nouveau membre du cercle.

Il y a à présent deux membres du cercle de signature et chacun possède la clé publique de son homologue. Ils commencent alors à s'échanger des éléments de trousseau individuels à travers l'espace de stockage des valeurs de clé iCloud. Si les deux membres du cercle possèdent le même élément, c'est l'élément présentant la date de modification la plus récente qui est synchronisé. Les éléments détenus par les deux membres et dont les dates de modification sont identiques sont ignorés. Chaque élément synchronisé est chiffré de manière spécifique pour l'appareil auquel il est envoyé. Il ne peut pas être déchiffré par d'autres appareils ni par Apple. De plus, l'élément chiffré n'est conservé que temporairement dans iCloud ; il est écrasé chaque fois qu'un nouvel élément est synchronisé.

Cette procédure est répétée chaque fois que de nouveaux appareils se joignent au cercle de synchronisation. Ainsi, si un troisième appareil entre dans le cercle, le message de confirmation s'affiche sur les deux autres appareils de l'utilisateur. Il peut alors approuver le nouveau membre à partir de l'un de ces deux appareils. À mesure que de nouveaux appareils sont ajoutés, chaque appareil est synchronisé avec le nouveau pour s'assurer que tous les membres disposent des mêmes éléments de trousseau.

La totalité du trousseau n'est toutefois pas synchronisée. Certains éléments propres à chaque appareil, tels que les identités VPN, ne peuvent pas quitter leur appareil. Seuls les éléments possédant l'attribut `kSecAttrSynchronizable` sont synchronisés. Apple a défini cet attribut pour les données d'utilisateur Safari (notamment les noms d'utilisateur, les mots de passe et les numéros de carte bancaire), ainsi que pour les mots de passe Wi-Fi et les clés de chiffrement HomeKit.

De plus, les éléments de trousseau ajoutés par des apps de tierce partie ne sont pas synchronisés par défaut. Les développeurs doivent définir l'attribut `kSecAttrSynchronizable` lorsqu'ils ajoutent des éléments au trousseau.

Récupération de trousseau

La récupération de trousseau offre aux utilisateurs qui le souhaitent un moyen de confier leur trousseau à Apple sans permettre à Apple de lire les mots de passe et autres données qu'il contient. La récupération de trousseau fournit à l'utilisateur un filet de sécurité contre la perte de données, même s'il ne possède qu'un seul appareil. Cela s'avère particulièrement important lorsque Safari est utilisé pour générer des mots de passe complexes et aléatoires pour des comptes web, car le trousseau constitue le seul endroit où sont enregistrés ces mots de passe.

La récupération de trousseau repose sur un service d'authentification secondaire et de dépôt sécurisé créé spécifiquement par Apple pour prendre cette fonctionnalité en charge. Le trousseau de l'utilisateur est chiffré à l'aide d'un mot de passe complexe et le service de dépôt fournit une copie du trousseau uniquement si certaines conditions strictes sont remplies.

Lorsque le trousseau iCloud est activé, l'utilisateur est invité à créer un code de sécurité iCloud. Ce code est nécessaire pour récupérer tout trousseau confié en dépôt. Par défaut, l'utilisateur est invité à fournir une simple série de quatre chiffres pour le code de sécurité. Il peut toutefois spécifier son propre code plus long ou laisser son appareil créer automatiquement un code cryptographiquement aléatoire qu'il peut ensuite enregistrer et conserver en lieu sûr.

L'appareil iOS exporte ensuite une copie du trousseau de l'utilisateur, chiffre cette copie en l'enveloppant avec des clés dans un conteneur de clés asymétrique et la place dans la zone de stockage des valeurs de clé iCloud de l'utilisateur. Le conteneur de clés est enveloppé à l'aide du code de sécurité iCloud de l'utilisateur et de la clé publique du cluster de module de sécurité matériel (HSM) destiné à stocker l'enregistrement en dépôt. Ce dernier devient alors l'enregistrement en dépôt iCloud de l'utilisateur.

Si l'utilisateur a décidé d'accepter un code de sécurité cryptographiquement aléatoire au lieu d'indiquer son propre code constitué d'une série de quatre chiffres, aucun enregistrement en dépôt n'est nécessaire. Au lieu de cela, le code de sécurité iCloud est utilisé pour protéger directement la clé aléatoire.

En plus d'établir un code de sécurité, les utilisateurs doivent enregistrer un numéro de téléphone. Ce numéro est utilisé pour offrir un deuxième niveau d'authentification lors de la récupération d'un trousseau. L'utilisateur reçoit un SMS auquel il doit répondre pour que la récupération soit effectuée.

Sécurité du dépôt

iCloud offre une infrastructure de sécurité pour le dépôt de trousseau qui permet de garantir que seuls des utilisateurs et des appareils autorisés peuvent effectuer la récupération. Topographiquement placés derrière iCloud se trouvent les clusters HSM qui hébergent les fiches de dépôt. Chacun d'eux possède une clé utilisée pour chiffrer les enregistrements en dépôt placés sous sa garde (voir description précédente).

Pour récupérer un trousseau, les utilisateurs doivent s'authentifier avec leur nom d'utilisateur et leur mot de passe iCloud et répondre à un SMS envoyé à leur numéro de téléphone enregistré. Après avoir effectué cette opération, ils doivent également saisir leur code de sécurité iCloud. Le cluster HSM vérifie que l'utilisateur connaît son code de sécurité iCloud à travers le protocole SRP ; le code lui-même n'est pas envoyé à Apple. Chaque membre du cluster vérifie indépendamment que l'utilisateur n'a pas dépassé le nombre maximum de tentatives autorisées de récupération de son enregistrement (voir ci-dessous). Si cela est confirmé par une majorité de clusters, l'enregistrement en dépôt est débloqué et envoyé à l'appareil de l'utilisateur.

L'appareil utilise ensuite le code de sécurité iCloud pour débloquer la clé aléatoire utilisée pour chiffrer le trousseau de l'utilisateur. Grâce à cette clé, le trousseau (récupéré à partir de l'espace de stockage des valeurs de clé iCloud) est déchiffré et restauré sur l'appareil. Le nombre de tentatives d'authentification et de récupération d'un enregistrement en dépôt est fixé à 10. Après plusieurs tentatives manquées, l'enregistrement est verrouillé et l'utilisateur doit appeler l'assistance Apple pour pouvoir effectuer des tentatives supplémentaires. Après la 10e tentative manquée, le cluster HSM détruit l'enregistrement en dépôt et le trousseau est perdu définitivement. Cette règle constitue une protection efficace contre les tentatives de récupération de l'enregistrement en force, mais les données du trousseau sont sacrifiées.

Ces politiques sont codées dans le programme interne du cluster HSM. Les cartes d'accès administratif permettant de modifier le programme interne ont été détruites. Toute tentative de modifier le programme interne ou d'accéder à la clé privée entraîne la suppression de cette dernière par le cluster HSM. Dans ce cas, les propriétaires de tous les trousseaux protégés par le cluster reçoivent un message leur annonçant la perte de leur enregistrement placé en dépôt. Ils peuvent alors décider de se réinscrire au service.

Siri

Les utilisateurs peuvent, en parlant naturellement, demander à Siri d'envoyer des messages, de programmer des rendez-vous, d'effectuer des appels téléphoniques et bien plus encore. Siri fait appel à la reconnaissance vocale, à la synthèse vocale et à un modèle client-serveur pour répondre à un large éventail de questions. Les tâches prises en charge par Siri ont été conçues en veillant à n'utiliser qu'une quantité absolument minimale de données personnelles et à assurer une protection totale de ces données.

Lorsque Siri est activé, l'appareil crée des identifiants aléatoires qui sont utilisés avec les serveurs Siri et les serveurs de reconnaissance vocale. Ces identifiants sont utilisés exclusivement dans Siri et servent à améliorer le service. Lorsque Siri est ensuite désactivé, l'appareil génère un nouvel identifiant aléatoire à utiliser à la réactivation de Siri.

Pour mettre en œuvre des fonctionnalités de Siri, certaines données d'utilisateur présentes sur l'appareil sont envoyées au serveur. Cela comprend notamment les données de la bibliothèque musicale (titres des morceaux, artistes et listes de lecture), les noms des listes de rappels, ainsi que les noms et les relations définies dans Contacts. Toutes les communications avec le serveur sont effectuées via HTTPS.

Lorsqu'une session Siri est lancée, le nom et le prénom de l'utilisateur (provenant de Contacts), ainsi que sa position géographique approximative sont envoyés au serveur. Cela permet à Siri de s'adresser à l'utilisateur par son nom ou de répondre à des questions ne nécessitant qu'une position approximative, comme la météo par exemple.

Si une position plus précise est nécessaire, pour indiquer les salles de cinéma les plus proches par exemple, le serveur demande à l'appareil de lui fournir une position plus précise. Cela montre comment, par défaut, l'information est envoyée au serveur uniquement lorsque cela s'avère strictement nécessaire pour traiter la demande de l'utilisateur. Les informations de session sont éliminées après 10 minutes d'inactivité, quoi qu'il arrive.

Lorsque Siri est utilisé sur l'Apple Watch, celle-ci crée son propre identifiant unique aléatoire, comme décrit ci-dessus. Toutefois, au lieu d'envoyer à nouveau les informations de l'utilisateur, elle intègre également à ses demandes l'identifiant Siri de l'iPhone jumelé pour fournir une référence à ces informations.

L'enregistrement des phrases prononcées par l'utilisateur est envoyé au serveur de reconnaissance vocale d'Apple. Si la tâche n'implique qu'une simple dictée, le texte reconnu est renvoyé à l'appareil. Sinon, Siri analyse le texte et, si nécessaire, le combine avec des informations provenant du profil associé à l'appareil. Si la demande est « envoyer un message à ma mère », par exemple, les relations et les noms téléchargés depuis Contacts sont utilisés. La commande de l'action identifiée est ensuite renvoyée à l'appareil afin d'être exécutée.

De nombreuses fonctionnalités de Siri sont effectuées par l'appareil sous les instructions du serveur. Par exemple, si l'utilisateur demande à Siri de lire un message entrant, le serveur demande simplement à l'appareil de lire le contenu de ses messages non lus à voix haute. Le contenu et l'expéditeur du message ne sont pas envoyés au serveur.

Les enregistrements vocaux de l'utilisateur sont conservés pendant six mois, afin que le système de reconnaissance vocale puisse les utiliser pour mieux comprendre la voix de l'utilisateur. Une autre copie est enregistrée après six mois, sans son identifiant, afin qu'Apple puisse l'utiliser pour améliorer et développer Siri, et ce, pendant deux ans au total. Un petit sous-ensemble de fiches, transcriptions et données associées sans identifiant sont susceptibles de continuer à être employées par Apple pour l'amélioration continue et le contrôle de qualité de Siri au-delà de deux ans. De plus, certains enregistrements faisant référence à la musique, aux équipes sportives et leurs joueurs, ainsi qu'au monde des affaires et divers points d'intérêt sont enregistrés de façon similaire en vue d'améliorer le service Siri.

Il est également possible d'utiliser Siri en mode mains libres au moyen de l'activation vocale. La détection de commande vocale est effectuée localement sur l'appareil. Avec ce mode, Siri n'est activé que si les mots entendus correspondent de manière satisfaisante au profil acoustique de la commande vocale spécifiée. Lorsque la commande est détectée, le son correspondant incluant la commande Siri est envoyé au serveur de reconnaissance vocale d'Apple en vue d'un traitement supplémentaire, selon les mêmes règles que celles appliquées aux autres enregistrements vocaux effectués par Siri.

Continuité

Continuité utilise des technologies comme iCloud, Bluetooth et Wi-Fi pour permettre aux utilisateurs de poursuivre sur un deuxième appareil une activité entamée sur un premier, d'effectuer et de recevoir des appels téléphoniques, d'envoyer et de recevoir des messages texte et de partager une connexion Internet mobile.

Handoff

Avec Handoff, l'utilisateur peut placer son ordinateur Mac à côté d'appareils iOS pour transférer automatiquement ce sur quoi il est en train de travailler d'un appareil à l'autre. Handoff permet ainsi à l'utilisateur de passer d'un appareil à l'autre tout en poursuivant son travail instantanément.

Lorsqu'un utilisateur se connecte à iCloud sur un deuxième appareil compatible Handoff, les deux appareils établissent un jumelage hors bande Bluetooth Low Energy 4.0 via le service de notification Push d'Apple (APN). Les messages individuels sont chiffrés de la même manière qu'avec iMessage. Une fois les appareils jumelés, chacun génère une clé symétrique AES 256 bits stockée dans le trousseau de chacun d'eux. Cette clé est utilisée pour chiffrer et authentifier les annonces Bluetooth Low Energy qui communiquent l'activité actuelle de l'appareil aux autres appareils jumelés via iCloud à l'aide d'un algorithme AES-256 en mode GCM, avec des mesures de protection contre la réexécution. La première fois qu'un appareil reçoit une annonce provenant d'une nouvelle clé, il établit une connexion Bluetooth Low Energy à l'appareil émetteur et exécute un échange de clés de chiffrement d'annonce. Cette connexion est sécurisée par chiffrement Bluetooth Low Energy 4.0 standard ainsi que par un chiffrement des différents messages (comme dans iMessage). Dans certains cas, ces messages sont transférés à travers les APN plutôt que par Bluetooth Low Energy (BLE). Le contenu de l'activité est protégé et transféré de la même manière qu'un message iMessage.

Handoff entre sites web et apps natives

Handoff permet à une app iOS native de reprendre des pages web appartenant à des domaines contrôlés de manière légitime par le développeur de l'app. Il autorise également la reprise dans un navigateur Web de l'activité de l'utilisateur de l'app native.

Pour éviter que des apps natives ne reprennent des sites Web non contrôlés par le développeur, l'app concernée doit prouver qu'elle contrôle légitimement les domaines qu'elle souhaite reprendre. Le contrôle d'un domaine de site Web est établi par le biais du mécanisme utilisé pour les accréditations Web partagées. Pour en savoir plus à ce sujet, reportez-vous à « Accès aux mots de passe enregistrés par Safari » dans la section

« Chiffrement et protection des données ». Le système doit valider le contrôle de l'app sur le nom de domaine avant que l'app ne soit autorisée à accepter la transmission de l'activité de l'utilisateur.

N'importe quel navigateur ayant adopté les API Handoff peut servir de source de transmission de page web. Lorsque l'utilisateur consulte une page web, le système diffuse le nom de domaine de cette page web dans les octets d'annonce Handoff chiffrés. Seuls les autres appareils de l'utilisateur sont capables de déchiffrer les octets d'annonce (tel que décrit précédemment à la section Handoff).

Sur l'appareil destinataire, le système détecte qu'une app native installée accepte la transmission du nom de domaine annoncé et affiche l'icône de cette app native comme option de transmission Handoff. Une fois ouverte, l'app native reçoit l'URL complète et le titre de la page web. Aucune autre information n'est transmise du navigateur à l'app native.

En sens inverse, une app native peut spécifier une URL de reprise lorsqu'un appareil destinataire Handoff ne possède pas la même app native installée. Dans ce cas, le système affiche le navigateur par défaut de l'utilisateur en tant que possibilité d'app Handoff (si le navigateur a adopté les API Handoff). Lorsque la transmission est demandée, le navigateur s'ouvre et reçoit l'URL de reprise fournie par l'app source. L'URL de reprise ne doit pas nécessairement être limitée aux noms de domaine contrôlés par le développeur de l'app native.

Transmission de volumes de données plus importants

Outre les fonctionnalités de base de Handoff, certaines apps peuvent choisir d'utiliser des API prenant en charge l'envoi de volumes de données plus importants par l'intermédiaire d'une technologie Wi-Fi poste-à-poste créée par Apple (de la même manière qu'avec AirDrop). L'app Mail, par exemple, utilise ces API pour prendre en charge la transmission de brouillons de message susceptibles d'inclure des pièces jointes volumineuses.

Lorsqu'une app exploite cette possibilité, l'échange entre les deux appareils démarre comme une transmission Handoff normale (voir sections précédentes). Toutefois, après la réception du contenu initial via Bluetooth Low Energy, l'appareil destinataire ouvre une nouvelle connexion via Wi-Fi. Cette connexion est chiffrée (TLS), ce qui implique l'échange de leurs certificats d'identité iCloud. L'identité des certificats est comparée à l'identité de l'utilisateur. Le reste des données de contenu est envoyé à travers cette connexion chiffrée jusqu'à ce que le transfert soit terminé.

Presse-papiers universel

Le presse-papiers universel s'appuie sur Handoff pour transférer de façon sécurisée le contenu de votre presse-papiers entre appareils afin de pouvoir copier sur un appareil et coller sur un autre. Le contenu est protégé de façon identique à toute autre donnée Handoff et partagé par défaut avec le presse-papiers universel, à moins que le développeur de l'app ne choisisse de ne pas autoriser le partage.

Les apps ont accès aux données du presse-papiers indépendamment du collage du presse-papiers dans l'app par l'utilisateur. Avec le presse-papiers universel, l'accès à ces données s'étend aux apps en cours d'exécution sur vos autres appareils (comme établi par votre session iCloud).

Verrouillage automatique

Les ordinateurs Mac prenant en charge le déverrouillage automatique font appel au Bluetooth Low Energy et aux réseaux Wi-Fi pair-à-pair pour autoriser de façon sécurisée l'Apple Watch de l'utilisateur à déverrouiller le Mac. Chaque Mac compatible et Apple Watch associés à un compte iCloud doivent utiliser l'identification à deux facteurs (TFA).

Lors de l'activation d'une Apple Watch pour déverrouiller un Mac, une liaison sécurisée faisant appel aux identités de déverrouillage automatique est établie. Le Mac crée un

secret de déverrouillage à usage aléatoire et unique puis le transmet à l'Apple Watch à travers la liaison. Le secret est stocké sur l'Apple Watch et ne peut être accédé que lorsque l'Apple Watch est déverrouillée (reportez-vous à la rubrique sur les classes de protection des données). Ni l'entropie maître, ni le nouveau secret ne correspond au mot de passe de l'utilisateur.

Au cours d'une opération de déverrouillage, le Mac utilise le Bluetooth Low Energy pour créer une connexion avec l'Apple Watch. Une liaison sécurisée est ensuite établie entre les deux appareils en utilisant les clés partagées générées au moment de la première activation. Le Mac et l'Apple Watch exploitent ensuite un réseau Wi-Fi pair-à-pair et une clé sécurisée dérivée de la liaison sécurisée pour déterminer la distance entre les deux appareils. Si les appareils se trouvent dans le champ de détection, la liaison sécurisée est alors utilisée pour transférer le secret prépartagé pour déverrouiller le Mac. Une fois le déverrouillage correctement établi, le Mac remplace le secret de déverrouillage en vigueur par un nouveau secret de déverrouillage à usage unique et le transmet à l'Apple Watch à travers la liaison.

Relais des appels cellulaires de l'iPhone

Si votre Mac, votre iPad ou votre iPod se trouve sur le même réseau Wi-Fi que votre iPhone, il peut effectuer et recevoir des appels téléphoniques à travers la connexion cellulaire de votre iPhone. Vos appareils doivent pour cela être connectés à la fois à iCloud et à FaceTime à l'aide du même Identifiant Apple.

À la réception d'un appel, tous les appareils configurés sont notifiés par l'intermédiaire du service APN, chaque notification utilisant le même chiffrement de bout en bout qu'iMessage. Les appareils qui se trouvent sur le même réseau affichent alors l'interface utilisateur de notification d'appel entrant. Lors de la prise de l'appel, le son est correctement transmis depuis votre iPhone via une connexion poste-à-poste sécurisée entre les deux appareils.

Si un appel est pris sur un appareil, la sonnerie sur les appareils à proximité et jumelés via iCloud est coupée par le biais d'une brève publication Bluetooth Low Energy 4.0. Les octets de cette publication sont chiffrés par le biais de la même méthode que les publications de type Handoff.

Les appels sortants sont également relayés vers l'iPhone par l'intermédiaire du service APN et le son est diffusé de la même manière via la liaison poste-à-poste sécurisée entre les appareils.

Il est possible de désactiver le relais d'appels téléphoniques sur un appareil en désactivant l'option « Appels cellulaires sur iPhone » dans les réglages FaceTime.

Transfert des SMS de l'iPhone

Le transfert des SMS envoie automatiquement les SMS reçus sur un iPhone à l'iPad, l'iPod touch ou le Mac enregistré d'un utilisateur. Chaque appareil doit être connecté au service iMessage à l'aide du même identifiant Apple. Lorsque le transfert des SMS est activé, l'enregistrement est vérifié sur chaque appareil en saisissant un code numérique aléatoire à six chiffres généré par l'iPhone.

Une fois que les appareils sont reliés, l'iPhone chiffre et transfère les SMS entrants à chaque appareil en faisant appel aux méthodes décrites dans la section iMessage de ce document. Les réponses sont renvoyées à l'iPhone à l'aide des mêmes méthodes, puis l'iPhone envoie la réponse sous forme de message texte en utilisant le mécanisme de transmission de SMS de l'opérateur. Le transfert des SMS peut être activé ou désactivé dans les réglages Messages.

Partage de connexion instantané

Les appareils iOS prenant en charge le partage de connexion instantané utilisent Bluetooth Low Energy pour détecter, et communiquer avec, les appareils connectés au même compte iCloud. Les ordinateurs Mac compatibles, exécutant OS X Yosemite et ultérieur, utilisent la même technologie pour détecter, et communiquer avec, les appareils iOS prenant en charge Instant Hotspot.

Lorsqu'un utilisateur accède aux réglages Wi-Fi de l'appareil iOS, ce dernier émet un signal Bluetooth Low Energy contenant un identifiant reconnu par tous les autres appareils connectés au même compte iCloud. L'identifiant est généré à partir d'un identifiant DSID (Destination Signaling Identifier) lié au compte iCloud et remplacé périodiquement. Lorsque d'autres appareils, connectés au même compte iCloud et prenant en charge le partage de connexion, se trouvent à proximité, ils détectent le signal et y répondent en indiquant leur disponibilité.

Lorsqu'un utilisateur choisit un appareil disponible pour le partage de connexion, une requête d'activation du partage de connexion est envoyée à cet appareil. Celle-ci est envoyée par le biais d'une liaison chiffrée au moyen de l'algorithme de chiffrement Bluetooth Low Energy standard et d'une méthode de chiffrement similaire à celle d'iMessage. L'appareil répond ensuite via la même liaison Bluetooth Low Energy, en utilisant la même méthode de chiffrement par message avec les informations du partage de connexion.

Suggestions Safari, Suggestions Spotlight, Recherche, #images et Widget Actualités dans les pays sans actualités

Suggestions Safari, Suggestions Spotlight, Recherche, #images et Widget Actualités dans les pays sans actualités affichent des suggestions dépassant le cadre de l'appareil des utilisateurs, provenant de sources telles que Wikipédia, l'iTunes Store, l'actualité locale, les résultats de cartes et l'App Store, mais aussi des suggestions avant même la saisie.

Lorsqu'un utilisateur commence à saisir dans la barre d'adresse de Safari, qu'il ouvre ou utilise Spotlight, qu'il utilise Recherche, qu'il ouvre #images ou fait appel au Widget Actualités dans les pays sans actualités, le contexte suivant est envoyé sous forme chiffrée par HTTPS à Apple pour offrir à l'utilisateur des résultats appropriés :

- un identifiant qui change toutes les 15 minutes afin de préserver la confidentialité ;
- la requête de recherche de l'utilisateur ;
- la localisation approximative de son appareil, si le Service de localisation pour les suggestions géodépendantes est activé. Le niveau d'« approximation » de la localisation dépend de la densité de population estimée à l'endroit où se trouve l'appareil ; ce niveau est plus important dans les zones rurales où les utilisateurs tendent à être géographiquement plus éloignés que dans les zones urbaines où les utilisateurs sont généralement plus proches les uns des autres. Les utilisateurs ont la possibilité de désactiver l'envoi à Apple de toutes les données de position géographique en désactivant l'option Service de localisation pour les Suggestions géodépendantes dans Réglages. Si le Service de localisation est désactivé, Apple peut utiliser l'adresse IP de l'appareil pour déduire une position approximative ;
- le type d'appareil et si la recherche est effectuée dans Spotlight, Safari, Recherche ou Messages ;
- le type de connexion ;
- l'information relative aux trois dernières apps utilisées sur l'appareil, incluse pour préciser le contexte de la recherche (seules les apps reprises dans une liste d'autorisations tenue à jour par Apple et contenant les apps populaires utilisées au cours des 3 dernières heures sont incluses) ;

- la liste des applications les plus utilisées sur l'appareil ;
- les préférences de langue, de pays et d'entrée ;
- si l'appareil de l'utilisateur peut accéder à des services de musique ou vidéo avec abonnement, les données telles que le nom du service et le type d'abonnement peuvent être envoyées à Apple ; le nom, le numéro et le mot de passe du compte ne sont pas envoyés à Apple.

Lorsqu'un utilisateur sélectionne un résultat ou ferme Spotlight sans rien avoir sélectionné, des informations sont envoyées à Apple pour contribuer à améliorer la qualité des résultats suivants. Ces informations sont liées uniquement au même identifiant de session de 15 minutes et non à un utilisateur donné. Le retour d'informations comprend les informations de contexte ci-dessus et les suivantes :

- les temps entre les interactions et les requêtes réseau des recherches ;
- le classement et l'ordre d'affichage des suggestions ;
- l'identifiant du résultat et l'action sélectionnée si le résultat n'est pas local, ou la catégorie du résultat local sélectionné.

Apple conserve pendant 18 mois les historiques de Suggestions comprenant les requêtes, le contexte et les commentaires. Des historiques plus réduits, comprenant uniquement la requête, le pays, la langue, la date et l'heure, ainsi que le type d'appareil, sont conservés pendant deux ans.

Dans certains cas, Suggestions peut transférer des requêtes relatives à des mots et des expressions courantes à un partenaire agréé, y compris le moteur de recherche Bing de Microsoft, afin de recevoir et d'afficher les résultats de recherche de ce partenaire. Les partenaires ne sont pas autorisés à stocker les requêtes et ne reçoivent aucun retour sur les recherches. Apple filtre les requêtes afin que les partenaires ne reçoivent pas en plus les adresses IP utilisateur. La communication avec le partenaire est chiffrée via HTTPS. Pour les requêtes fréquentes, Apple fournit au partenaire un contexte de recherche comprenant la ville, le type d'appareil et la langue du client pour améliorer la pertinence des résultats de recherche.

Pour comprendre et améliorer la pertinence géographique des suggestions et les performances à travers différents types de réseaux, les informations suivantes sont consignées sans identifiant de session :

- l'adresse IP partielle (sans le dernier octet dans le cas des adresses IPv4 et sans les derniers 80 bits pour les adresses IPv6) ;
- le lieu approximatif ;
- l'heure approximative de la requête ;
- la latence/le taux de transfert ;
- la taille de la réponse ;
- le type de connexion ;
- la langue ;
- le type d'appareil et l'app formulant la requête.

Contrôle des appareils

La plateforme iOS prend en charge des politiques et des configurations de sécurité souples, faciles à appliquer et à gérer. Cela permet aux organisations de protéger leurs informations et de s'assurer que les employés respectent les exigences de l'entreprise même s'ils utilisent leurs propres appareils dans le cadre d'un programme d'utilisation de matériel personnel au travail (BYOD), par exemple.

Les organisations peuvent utiliser des moyens comme la protection par code, les profils de configuration, l'effacement à distance ou des solutions de gestion d'appareils mobiles (MDM) de tierce partie pour gérer leurs parcs d'appareils et garantir la sécurité de leurs données, même si leurs employés y accèdent par le biais de leurs propres appareils iOS.

Protection par code

L'utilisateur peut choisir par défaut un code PIN constitué de chiffres. Sur les appareils dotés de Touch ID, la longueur minimale du code est de six chiffres. Sur les autres appareils, la longueur minimale est de quatre chiffres. Les utilisateurs peuvent indiquer un code alphanumérique plus long en sélectionnant Code alphanumérique personnalisé dans les Options de code de Réglages > Code. Les codes plus complexes ou plus longs sont plus difficiles à deviner ou à attaquer et sont recommandés pour les entreprises.

Les administrateurs peuvent imposer l'utilisation de codes complexes et d'autres politiques soit à l'aide d'Exchange ActiveSync ou d'une solution MDM, soit en demandant aux utilisateurs d'installer manuellement des profils de configuration. Il existe plusieurs possibilités en matière de politiques de code :

- autoriser des valeurs simples ;
- exiger des valeurs alphanumériques ;
- imposer une longueur minimale de code ;
- imposer un nombre minimal de caractères complexes ;
- imposer une durée limite de validité du code ;
- conserver un degré d'historique des codes ;
- appliquer un délai de blocage automatique ;
- offrir une période de grâce pour le blocage de l'appareil ;
- limiter le nombre de tentatives manquées ;
- autoriser Touch ID.

Pour en savoir plus sur chaque règlement, consultez l'article https://developer.apple.com/library/ios/featuredarticles/iPhoneConfigurationProfileRef/.

Modèle de jumelage iOS

iOS utilise un modèle de jumelage pour contrôler l'accès à un appareil à partir d'un ordinateur hôte. Le jumelage établit une relation de confiance entre l'appareil et son hôte connecté, concrétisée par un échange de clés publiques. iOS utilise cette marque de confiance pour activer des fonctionnalités supplémentaires avec l'hôte connecté, telles que la synchronisation de données. Dans iOS 9, les services qui nécessitent un jumelage ne peuvent pas être démarrés tant que l'appareil n'a pas été déverrouillé par l'utilisateur. De plus, sous iOS 10, certains services, notamment la synchronisation de photos, nécessitent que l'appareil soit déverrouillé pour commencer.

Le processus de jumelage nécessite que l'utilisateur déverrouille l'appareil et accepte la demande de jumelage émise par l'hôte. Une fois que l'utilisateur a accepté cette demande, l'hôte et l'appareil échangent et enregistrent des clés publiques RSA 2048 bits. L'hôte reçoit ensuite une clé 256 bits capable de débloquer un conteneur de clés en dépôt sur l'appareil (reportez-vous à la partie consacrée au « Conteneur de clés de dépôt » dans la section « Conteneurs de clés »). Les clés échangées sont utilisées pour lancer une session SSL chiffrée nécessaire pour que l'appareil puisse envoyer des données protégées à l'hôte ou démarrer un service (synchronisation iTunes, transferts de fichiers, développement Xcode, etc.). Comme l'appareil nécessite des connexions via Wi-Fi à partir d'un hôte pour utiliser cette session chiffrée pour toutes les communications, il faut qu'il ait été précédemment jumelé via USB. Le jumelage permet aussi d'activer plusieurs capacités de diagnostic. Dans iOS 9, si la fiche d'un jumelage n'a pas été utilisée pendant plus de six mois, celle-ci expire. Pour en savoir plus à ce sujet, consultez https://support.apple.com/fr-fr/HT6331.

Certains services, comme com.apple.pcapd, ne peuvent fonctionner que via USB. De même, le service com.apple.file_relay requiert un profil de configuration signé par Apple pour être installé.

L'utilisateur peut effacer la liste des hôtes de confiance en utilisant les options « Réinitialiser les réglages réseau » ou « Réin. localisation et confidentialité ». Pour en savoir plus à ce sujet, consultez https://support.apple.com/fr-fr/HT202778.

Application de la configuration

Un profil de configuration est un fichier XML qui permet à un administrateur de distribuer des informations de configuration à des appareils iOS. Les réglages définis par un profil de configuration installé ne peuvent être modifiés par l'utilisateur. Si l'utilisateur supprime un profil de configuration, tous les réglages définis par le profil sont également supprimés. Les administrateurs peuvent ainsi appliquer des réglages en associant des politiques à l'accès. Un profil de configuration destiné à fournir une configuration d'e-mail, par exemple, peut également spécifier une politique de code pour un appareil. Les utilisateurs ne pourront accéder à leur courrier électronique que si leurs codes sont conformes aux exigences de l'administrateur.

Les profils de configuration iOS contiennent plusieurs réglages qu'il est possible de spécifier :

- les règlements inhérents aux codes ;
- les restrictions liées aux fonctionnalités de l'appareil (comme la désactivation de la caméra) ;
- les réglages Wi-Fi ;
- les réglages VPN ;
- les réglages de serveur de messagerie ;
- les réglages Exchange ;
- les réglages de service d'annuaire LDAP ;

- les réglages de service de calendrier CalDAV ;
- les clips web ;
- les informations d'identification et les clés ;
- les réglages avancés de réseau mobile.

Il est possible de signer et de chiffrer les profils de configuration, afin de valider leur origine, de garantir leur intégrité et de protéger leur contenu. Les profils de configuration sont chiffrés en utilisant la syntaxe CMS (RFC 3852) qui prend en charge les algorithmes 3DES et AES-128.

Il est également possible de verrouiller des profils de configuration sur un appareil, afin d'interdire complètement leur suppression ou de n'autoriser cette dernière qu'au moyen d'un code. Comme beaucoup d'utilisateurs professionnels possèdent leurs propres appareils iOS, les profils de configuration qui associent un appareil à un serveur MDM peuvent être supprimés, mais cela a pour conséquence de supprimer également toutes les applications, les données et les informations de configuration gérées.

Les utilisateurs peuvent installer des profils de configuration directement sur leurs appareils à l'aide d'Apple Configurator, en télécharger via Safari, se les faire envoyer par courrier électronique ou les recevoir via une connexion sans fil à partir d'un serveur MDM.

Mobile Device Management - Gestion des appareils mobiles (MDM)

La prise en charge de la gestion MDM par iOS permet aux entreprises de gérer et de configurer de manière sécurisée des déploiements d'iPhone et d'iPad à grande échelle à travers leur organisation. Les fonctionnalités MDM sont intégrées aux technologies iOS existantes comme les profils de configuration, l'inscription en mode OTA et le service Apple de notification Push (APN). Par exemple, le service APN sert à réactiver l'appareil afin qu'il puisse communiquer directement avec son serveur MDM à travers une connexion sécurisée. Aucune information confidentielle ou propriétaire n'est transmise par le service APN.

Grâce à la gestion MDM, les services IT peuvent intégrer des appareils iOS dans des environnements d'entreprise, configurer et mettre des réglages à jour à travers une connexion sans fil, contrôler la conformité aux règles de la société et même verrouiller ou supprimer à distance le contenu des appareils gérés Pour en savoir plus sur la gestion d'appareils mobiles, consultez la page : https://www.apple.com/iphone/business/it/management.html.

iPad partagé

iPad partagé est un mode multi-utilisateur que l'on peut retrouver dans les déploiements d'iPad dans un cadre éducatif. Il permet aux élèves de partager un iPad sans partager de documents et de données. iPad partagé nécessite l'usage d'un Identifiant Apple Géré délivré et détenu par l'établissement scolaire. iPad partagé permet à un élève d'ouvrir une session sur un appareil détenu par l'établissement, configuré pour un usage par plusieurs élèves.

Les données des élèves sont partitionnées en répertoires de départ distincts, chacun protégé par des autorisations UNIX et par un environnement contrôlé de type sandboxing. Lorsqu'un élève ouvre une session, l'Identifiant Apple Géré est authentifié par le biais des serveurs d'identité d'Apple en faisant appel au protocole SRP. Si l'ouverture de session aboutit, un jeton d'accès de courte vie, spécifique à l'appareil, est accordé. Si l'élève a employé l'appareil auparavant, il dispose alors déjà d'un compte utilisateur local qui est alors déverrouillé. Si l'élève n'a pas utilisé l'appareil auparavant,

un nouvel identifiant d'utilisateur UNIX, un répertoire de départ et un trousseau lui sont fournis. (Si l'appareil n'est pas connecté à Internet, seuls les utilisateurs disposant préalablement de comptes locaux sont en mesure d'ouvrir une session.)

Après le déverrouillage ou la création du compte local de l'élève, si ce dernier s'authentifie à distance, le jeton de courte vie délivré par les serveurs d'Apple est converti en un jeton iCloud permettant d'ouvrir une session sur iCloud. Les réglages de l'étudiant sont ensuite restaurés, et ses documents et données sont synchronisés depuis iCloud.

Si la session de l'élève est active et que l'appareil reste en ligne, les documents et les données sont stockés sur iCloud au fur et à mesure de leur création ou de leur modification. En outre, un mécanisme de synchronisation en arrière-plan s'assure que les modifications sont envoyées à iCloud une fois que l'élève a fermé sa session.

Apple School Manager

Apple School Manager est un service qui s'adresse aux établissements d'enseignement et leur permet d'acheter du contenu, de configurer l'inscription automatique d'appareils dans des solutions de gestion d'appareils mobiles (MDM), de créer des comptes pour les élèves et le personnel, et de configurer des cours iTunes U. Apple School Manager est accessible sur le Web et s'adresse aux responsables des technologies, aux administrateurs informatiques, au personnel et aux professeurs.

Inscription d'appareils

Le programme d'inscription d'appareils (ou DEP, de l'anglais Device Enrollment Program) fournit un moyen rapide et optimisé de déployer des appareils iOS qu'une organisation a achetés directement auprès d'Apple ou de revendeurs et opérateurs agréés Apple. L'inscription d'appareils est en outre une fonctionnalité intégrée d'Apple School Manager pour les établissements d'enseignement.

Les organisations peuvent ainsi intégrer automatiquement ces appareils dans leur système de gestion MDM sans avoir à les manipuler physiquement ou à les préparer avant de les remettre à leurs utilisateurs. Après l'inscription dans le programme, les administrateurs se connectent au site web du programme et associent le programme à leur serveur MDM. Les appareils acquis peuvent ensuite être attribués à leurs utilisateurs via MDM. Une fois qu'un utilisateur a reçu son appareil, toutes les configurations, les restrictions ou les commandes MDM spécifiées sont automatiquement installées. Toutes les transmissions entre les appareils et les serveurs d'Apple sont chiffrées au cours du transfert par HTTPS (SSL).

Il est possible de simplifier encore davantage le processus de configuration en supprimant certaines étapes spécifiques dans l'assistant de configuration, afin que les utilisateurs puissent être rapidement opérationnels. Les administrateurs peuvent également contrôler si les utilisateurs peuvent supprimer le profil MDM de leur appareil et s'assurer que les restrictions sur ces appareils sont en place dès le départ. L'appareil, une fois déballé et activé, est automatiquement inscrit dans le système de gestion MDM de l'organisation et tous les livres, applications et réglages de gestion sont installés.

Pour en savoir plus, consultez la page https://help.apple.com/deployment/business/ et pour connaître les institutions de l'Enseignement, consultez https://help.apple.com/schoolmanager/.

Remarque : l'inscription d'appareils n'est pas disponible dans tous les pays ou toutes les régions.

Apple Configurator 2

Outre le système de gestion MDM, Apple Configurator 2 pour macOS simplifie la configuration et la préconfiguration d'appareils avant leur remise aux utilisateurs. Apple Configurator permet de préconfigurer rapidement les appareils avec leurs apps, données, restrictions et réglages.

Apple Configurator 2 vous permet d'utiliser Apple School Manager (pour l'Enseignement) ou le programme d'inscription d'appareil (pour les entreprises du privé) pour inscrire des appareils dans une solution de gestion d'appareils mobiles (MDM) sans que les utilisateurs aient à faire appel à l'Assistant réglages.

Supervision

Pendant la configuration d'un appareil, une entreprise peut configurer un appareil pour qu'il soit supervisé. La supervision désigne un appareil détenu par l'organisme dans le but d'assurer un contrôle supplémentaire sur sa configuration et ses restrictions. Il est possible de superviser des appareils au cours de leur configuration à travers Apple School Manager, le programme d'inscription d'appareil ou Apple Configurator.

Pour en savoir plus sur la configuration et la gestion d'appareils à l'aide d'une solution MDM ou d'Apple Configurator 2, consultez la page https://help.apple.com/deployment/ios/.

Restrictions

Les administrateurs ont la possibilité de restreindre l'utilisation de certaines fonctionnalités en installant un profil de configuration. Parmi les restrictions disponibles, l'on retrouve notamment :

- autoriser les installations d'apps ;
- autoriser la confiance dans les apps d'entreprise ;
- autoriser l'utilisation de la caméra ;
- autoriser FaceTime ;
- autoriser les captures d'écran ;
- autoriser la composition vocale en cas de verrouillage ;
- autoriser la synchronisation automatique des données à l'étranger ;
- autoriser les achats intégrés à partir d'une app ;
- autoriser la synchronisation du courrier récent ;
- forcer l'utilisateur à saisir son mot de passe pour tous ses achats ;
- autoriser Siri lorsque l'appareil est verrouillé ;
- autoriser l'utilisation de l'iTunes Store ;
- autoriser les documents provenant de sources gérées dans des destinations non gérées ;
- autoriser les documents provenant de sources non gérées dans des destinations gérées ;
- autoriser la synchronisation du trousseau iCloud ;
- autoriser la mise à jour sans fil de la base de données des certificats de confiance ;
- autoriser l'affichage des notifications sur l'écran de verrouillage ;
- forcer les connexions AirPlay à utiliser des mots de passe de jumelage ;
- autoriser Spotlight à afficher le contenu créé par des utilisateurs sur Internet ;
- activer les Suggestions Spotlight dans Spotlight ;
- autoriser Handoff ;
- considérer AirDrop comme une destination non gérée ;
- autoriser la sauvegarde des livres d'entreprise ;
- autoriser la synchronisation des notes et des signets des livres d'entreprise sur tous les appareils de l'utilisateur ;
- autoriser l'utilisation de Safari ;
- autoriser le remplissage automatique des formulaires dans Safari ;
- forcer les alertes de sites web frauduleux ;
- activer JavaScript ;

- limiter le suivi publicitaire dans Safari ;
- bloquer les fenêtres surgissantes ;
- accepter les cookies ;
- autoriser la sauvegarde iCloud ;
- autoriser la synchronisation des documents et des valeurs de clé iCloud ;
- activer le partage de photos iCloud ;
- autoriser l'envoi de diagnostics à Apple ;
- autoriser l'utilisateur à accepter les certificats TLS non fiables ;
- forcer le chiffrement des sauvegardes ;
- autoriser Touch ID ;
- autoriser l'accès au Centre de contrôle à partir de l'écran verrouillé ;
- autoriser l'affichage du jour sur l'écran verrouillé ;
- exiger l'utilisation de la détection du poignet de l'Apple Watch ;
- autoriser l'observation de l'écran par En classe (Classroom) ;
- utiliser AirDrop depuis une app gérée ;
- faire confiance à un nouveau développeur d'app d'entreprise ;
- utiliser la photothèque iCloud.

Restrictions supervisées uniquement

- autoriser iMessage ;
- autoriser la suppression des apps ;
- autoriser l'installation manuelle des profils de configuration ;
- utiliser un proxy réseau mondial pour HTTP ;
- autoriser le jumelage avec des ordinateurs pour la synchronisation de contenus ;
- restreindre les connexions AirPlay à l'aide d'une liste blanche et de codes de connexion facultatifs ;
- autoriser AirDrop ;
- autoriser la modification de Localiser mes amis ;
- autoriser le mode App individuelle autonome pour certaines apps gérées ;
- autoriser la modification du compte ;
- autoriser la modification des données mobiles ;
- autoriser le jumelage avec un hôte (iTunes) ;
- autoriser le Verrouillage d'Activation ;
- interdire l'effacement du contenu et des réglages ;
- interdire l'activation des restrictions ;
- utiliser le filtre de contenus de tierce partie ;
- utiliser le mode App individuelle autonome ;
- utiliser le mode VPN permanent ;
- autoriser la modification du code ;
- autoriser le jumelage de l'Apple Watch ;
- autoriser les téléchargements automatiques d'apps ;
- autoriser la prédiction de la saisie, l'autocorrection, le correcteur orthographique et les raccourcis clavier ;
- autoriser Apple Music ;
- autoriser la radio ;
- faire observer l'écran par En classe (Classroom) ;
- connaître les modifications apportées aux réglages Notifications ;
- afficher ou masquer des apps particulières dans l'écran d'accueil ;
- installer des apps à l'aide de l'App Store ;
- télécharger automatiquement les apps ;
- utiliser les raccourcis clavier ;
- autoriser Définir ;
- modifier le nom de l'appareil ;
- changer de fond d'écran ;
- masquer l'app News ;
- jumeler avec l'Apple Watch.

Pour en savoir plus sur les contrôles de développement MDM complémentaires pour les appareils supervisés, consultez https://developer.apple.com/library/ios/featuredarticles/iPhoneConfigurationProfileRef/.

Effacement à distance

Les appareils iOS peuvent être effacés à distance par un administrateur ou un utilisateur. L'effacement instantané à distance est effectué en éliminant de manière sécurisée la clé de chiffrement de stockage en blocs du stockage effaçable (Effaceable Storage), ce qui rend toutes les données illisibles. La commande d'effacement à distance peut être envoyée à partir de la gestion MDM, d'Exchange ou d'iCloud.

Lorsqu'une commande d'effacement à distance est déclenchée via MDM ou iCloud, l'appareil envoie une confirmation et effectue l'effacement des données. Pour l'effacement à distance via Exchange, l'appareil confirme la commande auprès du serveur Exchange avant d'effectuer l'effacement des données.

Les utilisateurs ont également la possibilité d'effacer le contenu des appareils en leur possession en utilisant l'app Réglages. Enfin, comme cela a été mentionné précédemment, il est possible de régler les appareils afin qu'ils effacent automatiquement leurs données après un certain nombre de tentatives manquées de saisie de code.

Mode Perdu

Si un appareil vient à être perdu ou volé, un administrateur MDM peut activer à distance le mode Perdu sur un appareil supervisé doté d'iOS 9.3 ou ultérieur. Lorsque le mode Perdu est activé, l'utilisateur actif est déconnecté et l'appareil ne peut pas être déverrouillé. L'écran affiche un message que l'administrateur peut personnaliser, par exemple un numéro de téléphone à appeler si l'appareil vient à être retrouvé. Quand l'appareil est placé en mode Perdu, l'administrateur peut demander à l'appareil d'envoyer sa position. Si un administrateur désactive le mode Perdu, ce qui constitue le seul moyen de quitter le mode, l'utilisateur est informé de cette opération au travers d'un message sur l'écran de verrouillage et d'une alerte dans l'écran d'accueil.

Verrouillage d'Activation

Lorsque la fonctionnalité Localiser mon iPhone est activée, il est impossible de réactiver un appareil sans saisir les informations d'identification de l'Identifiant Apple du propriétaire.

Pour les appareils détenus par une organisation, il peut s'avérer judicieux de les superviser de sorte que la fonction de Verrouillage d'Activation puisse être gérée par l'organisation plutôt que de demander à chaque utilisateur de saisir ses informations d'identification Apple pour réactiver son appareil.

Sur des appareils supervisés, une solution MDM compatible peut ensuite conserver un code de contournement lorsque le Verrouillage d'Activation est activé et utiliser ce code ultérieurement pour effacer automatiquement le Verrouillage d'Activation lorsqu'un appareil doit être effacé et attribué à un nouvel utilisateur. Pour en savoir plus à ce sujet, consultez la documentation de votre solution MDM.

Par défaut, le Verrouillage d'Activation n'est jamais possible sur des appareils supervisés, même si l'utilisateur active la fonctionnalité Localiser mon iPhone. Un serveur MDM peut toutefois récupérer un code de contournement et autoriser l'activation du Verrouillage d'Activation sur l'appareil. Si la fonctionnalité Localiser mon iPhone est activée lorsque le serveur MDM autorise le Verrouillage d'Activation, le verrouillage est activé à partir de ce moment. Si la fonctionnalité Localiser mon iPhone est désactivée lorsque le serveur MDM autorise le Verrouillage d'Activation, ce dernier est activé dès que l'utilisateur en fait de même pour Localiser mon iPhone.

Pour les appareils exploités dans un contexte académique par le biais d'un Identifiant Apple Géré créé à travers Apple School Manager, la fonction de Verrouillage d'Activation peut être liée à l'Identifiant Apple d'un administrateur plutôt qu'à celui des utilisateurs, ou être désactivée à l'aide du code de contournement de l'appareil.

Contrôles de confidentialité

Apple accorde une grande importance à la protection des données personnelles de ses clients et a conçu plusieurs options et commandes intégrées qui permettent aux utilisateurs iOS de déterminer la manière dont les applications utilisent leurs informations, le moment où elles le font et la nature des informations utilisées.

Service de localisation

Le service de localisation utilise les données GPS, la connexion Bluetooth et une base de données communautaire des emplacements des bornes d'accès Wi-Fi et des antennes-relais de téléphonie mobile pour déterminer la position approximative des utilisateurs. Le service de localisation peut être désactivé au moyen d'un commutateur unique dans Réglages. L'utilisateur a également la possibilité d'autoriser l'accès de chaque app à ce service. Chaque app peut demander l'autorisation de recevoir des données de localisation de manière permanente ou uniquement lorsqu'elle est utilisée. L'utilisateur peut décider de ne pas autoriser cet accès et peut modifier son choix à tout moment dans Réglages. Dans Réglages, l'utilisateur peut choisir de ne jamais autoriser l'accès, de l'autoriser ponctuellement en cas d'utilisation ou de l'autoriser en permanence, en fonction de l'usage de la localisation demandée par l'app. Par ailleurs, si une app autorisée à utiliser les données de localisation en permanence profite de cette autorisation alors qu'elle est exécutée en arrière-plan, un message est envoyé à l'utilisateur pour le prévenir et lui donner la possibilité de modifier l'autorisation d'accès de l'app.

L'utilisateur dispose en outre d'un contrôle précis sur la manière dont les services du système utilisent les données de localisation. Cela inclut la possibilité de désactiver l'inclusion des données de localisation dans les informations recueillies par les services de diagnostic et d'utilisation employés par Apple pour améliorer le système iOS, les informations de Siri basées sur la localisation, le contexte basé sur la localisation des recherches de Suggestions Spotlight, les conditions de circulation locales et les lieux fréquemment visités utilisés pour estimer les durées de déplacement.

Accès aux données personnelles

iOS aide à interdire l'accès non autorisé des apps aux données personnelles de l'utilisateur. Ce dernier peut en plus utiliser Réglages pour voir quelles sont les apps autorisées à accéder à certaines informations et pour accorder ou refuser toute autorisation d'accès ultérieure. Cela comprend l'accès aux éléments suivants :

- Contacts
- Calendriers
- Rappels
- Photos
- Mouvements et activités physiques
- Services de géolocalisation
- Bibliothèque multimédia
- Comptes de réseaux sociaux tels que Twitter et Facebook
- Micro
- Caméra
- HomeKit
- Santé
- Reconnaissance vocale
- Partage Bluetooth

Si l'utilisateur se connecte à iCloud, les apps sont autorisées par défaut à se connecter à iCloud Drive. L'utilisateur peut contrôler l'accès de chaque app sous iCloud dans Réglages. iOS fournit également des restrictions qui interdisent tout mouvement de données entre les apps et les comptes installés par une solution MDM et ceux installés par l'utilisateur.

Politique de confidentialité

La politique de confidentialité d'Apple est disponible en ligne à l'adresse https://www.apple.com/fr/legal/privacy.

Récompense de sécurité Apple

Apple récompense les chercheurs qui font part à Apple des problèmes critiques qu'ils rencontrent. Afin d'avoir droit à la récompense de sécurité Apple, les chercheurs doivent fournir un rapport clair et des preuves concrètes du concept. La vulnérabilité doit affecter la dernière livraison d'iOS et s'avérer applicable au dernier matériel. Le montant exact du paiement est déterminé après revue par Apple. Les critères utilisés comprennent la nouveauté, la probabilité d'exposition et le degré d'interaction de l'utilisateur requise.

Une fois correctement partagés, Apple se donne pour priorité de résoudre les problèmes confirmés le plus rapidement possible. Selon les cas, Apple en fait état publiquement, sauf demande contraire

Catégorie	**Paiement maximal (en USD)**
Composants du programme interne de démarrage sécurisé	200 000 $
Extraction d'éléments confidentiels protégés par le Secure Enclave	100 000 $
Exécution de code arbitraire avec privilèges de noyau	50 000 $
Accès interdit aux données de compte iCloud sur les serveurs d'Apple	50 000 $
Accès depuis un processus contrôlé aux données de l'utilisateur en dehors de l'environnement contrôlé	25 000 $

Conclusion

Un engagement en faveur de la sécurité

Apple s'engage à contribuer à la protection de ses clients en leur proposant des technologies avancées de sécurité et de confidentialité conçues pour protéger leurs données personnelles, ainsi que des méthodes complètes destinées à protéger les données professionnelles dans les environnements d'entreprise.

La sécurité fait partie intégrante du système iOS. De la plateforme au réseau, en passant par les apps, iOS possède tout ce dont une entreprise a besoin. Ensemble, ces composants confèrent à iOS les fonctionnalités de sécurité les plus performantes du marché, sans compromettre l'expérience d'utilisation.

Apple fait appel à une infrastructure de sécurité intégrée et cohérente à travers tout le système iOS et l'écosystème constitué par les apps iOS. Le chiffrement matériel des espaces de stockage fournit des capacités d'effacement à distance en cas de perte d'appareil et permet aux utilisateurs de supprimer complètement toutes leurs données personnelles ainsi que celles de l'entreprise en cas de revente de leur appareil ou de son transfert à une autre personne. Les informations utilisées pour le diagnostic sont également collectées de manière anonyme.

Les apps iOS conçues par Apple sont développées dans un souci de sécurité avancée. Safari offre une navigation en toute sécurité grâce à la prise en charge du protocole OCSP (Online Certificate Status Protocol), des certificats EV et des alertes de vérification de certificat. Mail utilise des certificats pour l'authentification et le chiffrement du courrier en prenant en charge la norme S/MIME qui autorise le chiffrement S/MIME individuel des messages, afin d'offrir aux utilisateurs S/MIME la possibilité de choisir entre la signature et le chiffrement permanents par défaut et le contrôle sélectif du mode de protection des messages individuels. iMessage et FaceTime fournissent également un chiffrement de client à client.

Pour les apps de tierce partie, la combinaison de la signature obligatoire du code, du sandboxing et des autorisations fournit aux utilisateurs une protection fiable contre les virus, les logiciels malveillants et d'autres programmes susceptibles de compromettre la sécurité d'autres plateformes. Le processus de soumission à l'App Store renforce la protection des utilisateurs contre ces risques, car chaque app iOS est examinée avant d'être mise sur le marché.

Pour tirer le meilleur parti des fonctionnalités de sécurité étendues intégrées dans iOS, les entreprises sont encouragées à revoir leurs politiques en matière de sécurité et de services informatiques, afin de s'assurer qu'elles exploitent au mieux les couches de technologies de sécurité offertes par cette plateforme.

Apple dispose d'une équipe de sécurité spécialisée, chargée de fournir une assistance pour tous les produits Apple. L'équipe propose des services d'audit et de test aussi bien pour les produits en développement, que pour les produits déjà commercialisés. L'équipe Apple fournit également des formations et des outils de sécurité et se tient activement informée de tous les rapports concernant les nouveaux problèmes et menaces de sécurité. Apple fait partie du forum FIRST (Forum of Incident Response and Security Teams) qui rassemble des équipes chargées de la sécurité et de la réponse aux incidents. Pour en savoir plus sur le signalement des problèmes à Apple et l'abonnement aux notifications de sécurité, consultez la page : https://www.apple.com/fr/support/security.

Glossaire

APN (Apple Push Notification), service de notification Push d'Apple	Service mondial offert par Apple pour fournir des notifications de type Push aux appareils iOS.
Carte intelligente	Circuit intégré incorporé qui fournit une identification, une authentification et un stockage de données sécurisés.
Cartographie des angles des crêtes papillaires	Représentation mathématique du sens et de la largeur des crêtes extraites d'une partie d'empreinte digitale.
Circuit intégré (CI)	Également appelé micropuce.
Clé de fichier	Clé AES 256 bits utilisée pour chiffrer un fichier du système de fichiers. La clé de fichier est enveloppée par une clé de classe et stockée dans les métadonnées du fichier.
Clé du système de fichiers	Clé permettant de chiffrer les métadonnées de chaque fichier, y compris la clé de classe. Elle est conservée dans l'espace de stockage effaçable pour permettre l'effacement à distance plutôt que pour des raisons de confidentialité.
Conteneur de clés	Structure de données utilisée pour stocker une collection de clés de classe. Chaque type (utilisateur, appareil, système, sauvegarde, dépôt ou Sauvegarde iCloud) possède le même format : • Un en-tête contenant : – la version (définie sur 3 dans iOS 5) ; – le type (système, sauvegarde, dépôt ou Sauvegarde iCloud) ; – l'UUID du conteneur de clés ; – un code HMAC si le conteneur de clés est signé ; – la méthode utilisée pour envelopper les clés de classe : en incorporant l'UID ou PBKDF2, le salage et le nombre d'itérations. • Une liste de clés de classe : – UUID de clé ; – une classe (classe de protection des données de trousseau ou de fichier) ; – type d'enveloppe (clé dérivée de l'UID uniquement ou clé dérivée de l'UID et clé dérivée du code) ; – clé de classe enveloppée ; – clé publique pour les classes asymétriques.
Data Protection (Protection des données)	Mécanisme de protection des fichiers et des trousseaux pour iOS. Cette expression peut également faire référence aux API utilisées par des apps pour protéger des fichiers et des éléments de trousseau.
DFU (Device Firmware Upgrade), mise à niveau de logiciel interne d'appareil	Mode d'attente adopté par le code de la ROM de démarrage d'un appareil avant une récupération via USB. L'écran de l'appareil est noir en mode DFU, mais l'invite ci-dessous est affichée dès la connexion à un ordinateur exécutant iTunes : « iTunes a détecté un iPad en mode de récupération. Vous devez restaurer cet iPad avant de pouvoir l'utiliser avec iTunes. »
ECID	Identifiant 64 bits propre au processeur de chaque appareil iOS. Si un appel est pris sur un appareil, la sonnerie sur les appareils à proximité et jumelés à travers iCloud est coupée par le biais d'une brève publication Bluetooth Low Energy 4.0. Les octets de cette publication sont chiffrés par le biais de la même méthode que les publications de type Handoff. Il est utilisé dans le cadre du processus de personnalisation et n'est pas considéré comme un secret.
Effaceable Storage (Stockage effaçable)	Zone dédiée de l'espace de stockage NAND, utilisée pour stocker des clés cryptographiques. Il est possible de l'adresser directement et de l'effacer de manière sécurisée. Bien qu'elle n'offre aucune protection si l'attaquant prend physiquement possession de l'appareil, les clés conservées dans l'espace de stockage effaçable peuvent être utilisées dans le cadre d'une hiérarchie de clés pour faciliter l'effacement à distance et renforcer la sécurité.

Emmêlement (Tangling)	Processus par lequel le code d'un utilisateur est transformé en clé cryptographique et renforcé à l'aide de l'UID de l'appareil. Grâce à cette technique, les attaques en force ne peuvent être exécutées que sur un appareil donné à la fois, ce qui empêche les attaques massives menées en parallèle. L'algorithme d'emmêlement est le PBKDF2 qui utilise une clé AES avec l'UID de l'appareil comme fonction PRF pour chaque itération.
Enveloppement de clé	Chiffrement d'une clé à l'aide d'une autre clé. iOS utilise l'enveloppement de clé NIST AES conforme à la norme RFC 3394.
GID (identifiant de groupe)	Semblable à l'UID, mais commun à tous les processeurs d'une classe.
HSM (Hardware Security Module), module de sécurité matériel	Ordinateur spécialisé, protégé contre toute manipulation, utilisé pour sauvegarder et gérer des clés numériques.
iBoot	Code chargé par LLB et qui charge à son tour XNU, dans le cadre d'une chaîne de démarrage sécurisée.
IDS (Identity Service), service d'identité	Répertoire Apple contenant les clés publiques d'iMessage, les adresses de service APN, les numéros de téléphone et les adresses électroniques utilisés pour la recherche d'adresses d'appareil et de clés.
JTAG (Joint Test Action Group)	Outil de débogage matériel standard utilisé par les programmeurs et les développeurs de circuits.
LLB (Low-Level Bootloader), chargeur de démarrage de bas niveau	Code invoqué par la ROM de démarrage, qui charge à son tour l'iBoot dans le cadre d'une chaîne de démarrage sécurisée.
Profil d'approvisionnement	Fichier plist signé par Apple, qui contient un ensemble d'entités et de droits qui autorisent des apps à être installées et testées sur un appareil iOS. Un profil d'approvisionnement de développement répertorie les appareils sélectionnés par un développeur en vue d'une distribution ad hoc. Un profil d'approvisionnement de distribution contient l'identifiant d'une app développée par une entreprise.
Randomisation du format d'espace d'adresse (ASLR)	Technique utilisée par iOS pour rendre plus difficile l'exploitation d'un bogue de logiciel. Comme les décalages et les adresses mémoire sont imprévisibles, le code d'exploit ne peut pas coder ces valeurs en dur. Sous iOS 5 et ultérieur, la position de toutes les bibliothèques et apps système est déterminée de manière aléatoire, de même que celle des apps de tierce partie compilées en tant qu'exécutables indépendamment de la position.
ROM de démarrage (boot)	Tout premier code exécuté par le processeur d'un appareil lors du démarrage de ce dernier. Ce code fait partie intégrante du processeur, il ne peut donc être modifié ni par Apple, ni par un attaquant.
SoC (System on a chip), puce-système	Circuit intégré (CI) incorporant plusieurs composants sur une seule puce. Secure Enclave est une puce-système utilisée dans le processeur central A7 ou ultérieur d'Apple.
Trousseau	L'infrastructure et un ensemble d'API utilisés par iOS et les apps de tierce partie pour stocker et récupérer des mots de passe, des clés et d'autres accréditations sensibles.
URI (Uniform Resource Identifier), identifiant de ressource uniforme	Chaîne de caractères permettant d'identifier une ressource web.
UID (Unique ID), identifiant unique	Clé AES 256 bits gravée sur chaque processeur au moment de sa fabrication. Elle ne peut être lue ni par le programme interne, ni par le logiciel, et n'est utilisée que par le moteur AES matériel du processeur. Pour trouver cette clé, un attaquant potentiel devrait lancer une attaque physique onéreuse et extrêmement sophistiquée contre le silicium du processeur. L'UID n'est lié à aucun autre identifiant présent sur l'appareil, tel que l'UDID par exemple.
XNU	Noyau au centre des systèmes d'exploitation iOS et OS X. Il est supposé fiable et permet d'appliquer des mesures de sécurité telles que la signature de code, le sandboxing, la vérification des droits et la distribution aléatoire de l'espace d'adressage (ASLR).

Historique des révisions du document

Date	Résumé
Mars 2017	**Actualisé pour iOS 10** • Sécurité du système • Classes de protection des données • Certificats et programmes de sécurité • HomeKit, ReplayKit, SiriKit • Apple Watch • Wi-Fi, VPN • Authentification unique • Apple Pay, effectuer des paiements avec Apple Pay sur le web • Transfert sur cartes bancaires et prépayées • Suggestions Safari • Pour en savoir plus sur les correctifs de sécurité d'iOS 10, consultez : https://support.apple.com/fr-fr/HT207143
Mai 2016	**Actualisé pour iOS 9.3** • Identifiant Apple Géré • Identification à deux facteurs pour l'identifiant Apple • Conteneurs de clés • Certifications de sécurité • Mode Perdu et Verrouillage d'Activation • Notes sécurisées • Apple School Manager, iPad partagé • Pour en savoir plus sur les correctifs de sécurité d'iOS 9.3, consultez : https://support.apple.com/fr-fr/HT206166
Septembre 2015	**Actualisé pour iOS 9** • Verrouillage d'Activation de l'Apple Watch • Règlements inhérents aux codes • Prise en charge de l'API Touch ID • Protection des données sur l'A8 avec AES-XTS • Conteneurs de clés pour la mise à jour logicielle sans surveillance • Mises à jour de certification • Modèle de confiance des apps d'entreprise • Protection des données pour les signets Safari • Sécurité du transport des apps • Spécifications VPN • Accès distant à iCloud pour HomeKit • Cartes de fidélité Apple Pay, app d'émetteur de carte Apple Pay • Indexation Spotlight sur l'appareil • Modèle de jumelage iOS • Apple Configurator 2 • Restrictions • Pour en savoir plus sur les correctifs de sécurité d'iOS 9, consultez : https://support.apple.com/fr-fr/HT205212