



# Sécurité iOS

Octobre 2014

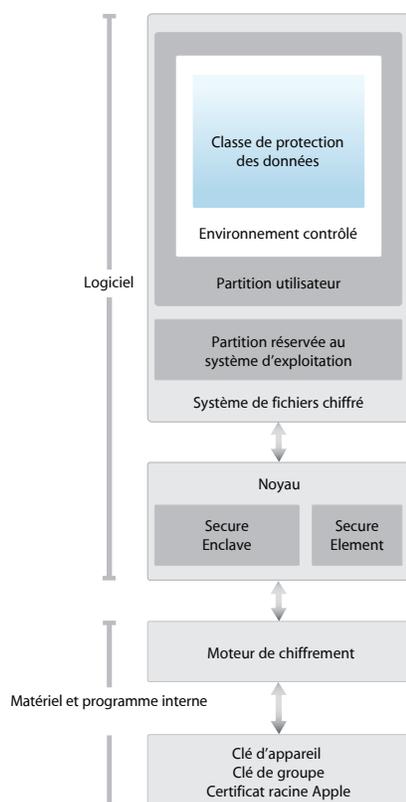
iOS 8.1 ou ultérieur

# Table des matières

<b>Page 4</b>	<b>Introduction</b>
<b>Page 6</b>	<b>Sécurité du système</b> Chaîne de démarrage sécurisé Autorisation du logiciel système Secure Enclave Touch ID
<b>Page 11</b>	<b>Chiffrement et protection des données</b> Fonctionnalités de sécurité matérielles Protection des données des fichiers Codes de déverrouillage Classes de protection des données Protection des données du trousseau Accès aux mots de passe enregistrés par Safari Conteneurs de clés FIPS 140-2
<b>Page 20</b>	<b>Sécurité des apps</b> Signature du code des apps Sécurité des processus exécutés Extensions Groupes d'apps Protection des données dans les apps Accessoires HomeKit HealthKit
<b>Page 28</b>	<b>Sécurité du réseau</b> SSL, TLS VPN Wi-Fi Bluetooth Authentification unique Sécurité AirDrop
<b>Page 32</b>	<b>Services Internet</b> Identifiant Apple iMessage FaceTime iCloud Trousseau iCloud Siri Continuité Suggestions Spotlight

<b>Page 45</b>	<b>Contrôle des appareils</b> Protection par code Modèle de jumelage iOS Application de la configuration Mobile Device Management (MDM) Programme d'inscription d'appareils Apple Configurator Restrictions d'appareils Restrictions spécifiques aux appareils supervisés Effacement à distance Localiser mon iPhone et verrouillage d'activation
<b>Page 52</b>	<b>Protection des données personnelles</b> Service de localisation Accès aux données personnelles Politique de confidentialité
<b>Page 54</b>	<b>Conclusion</b> Un engagement en faveur de la sécurité
<b>Page 56</b>	<b>Glossaire</b>

# 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 plate-forme iOS en mettant l'accent sur la sécurité. Quand nous avons entrepris de créer la meilleure plate-forme 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 de 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 plate-forme iOS. Il aide également les organisations à combiner les technologies et les fonctionnalités de sécurité de la plate-forme 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 plate-forme 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 plate-forme.
- **Sécurité du réseau** : les protocoles standards de mise en réseau qui assurent la sécurisation de l'authentification et le chiffrement des données lors des transmissions.

- **Services Internet** : l'infrastructure réseau d'Apple pour la messagerie, la synchronisation et la sauvegarde.
- **Contrôles de l'appareil** : les méthodes qui empêchent l'utilisation non autorisée de l'appareil et permettent l'effacement de son contenu à distance en cas de perte ou de vol.
- **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 de mise en veille/réactivation. Relâchez le bouton de mise en veille/réactivation après 8 secondes tout en continuant à maintenir le bouton principal enfoncé.

Remarque : rien n'est affiché à l'écran lorsque l'appareil est en mode DFU. Si le logo Apple apparaît, c'est que le bouton de mise en veille/réactivation 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.

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 de l'autorité de certification (AC) racine d'Apple, qui est utilisée pour vérifier que le chargeur de démarrage de bas niveau (LLB, Low-Level Bootloader) 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 que le LLB a terminé ses tâches, il vérifie et exécute le chargeur de démarrage de l'étape suivante, iBoot, qui à son tour vérifie et exécute le noyau iOS.

Cette chaîne de démarrage sécurisé permet de s'assurer que les niveaux les plus bas du logiciel ne sont pas altérés et n'autorise iOS à s'exécuter que sur des appareils Apple validés.

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

Pour les appareils dotés d'un processeur A7 ou d'une version ultérieure de la série A, le coprocesseur Secure Enclave emploie aussi un processus de démarrage sécurisé qui garantit que son logiciel indépendant est vérifié et signé par Apple.

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, Device Firmware Upgrade). 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/kb/HT1808?viewlocale=fr\\_FR](https://support.apple.com/kb/HT1808?viewlocale=fr_FR).

## Autorisation du logiciel système

Apple publie régulièrement des mises à jour du logiciel pour corriger les problèmes de sécurité qui surviennent et apporter de nouvelles fonctionnalités ; ces mises à jour sont fournies simultanément 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'un processeur A7 ou d'une version ultérieure de la série A, le coprocesseur Secure Enclave fait également appel à l'autorisation du logiciel système pour garantir l'intégrité de son logiciel et empêcher l'installation d'une version antérieure. Consultez la rubrique « Secure Enclave » ci-dessous.

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 d'OS X 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, le chargeur de démarrage de bas niveau, 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 aux 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 quelque façon le logiciel système.

## Secure Enclave

Secure Enclave est un coprocesseur intégré au processeur A7 ou une version ultérieure de la série A d'Apple. Il utilise ses propres processus de démarrage sécurisé et de mise à jour logicielle indépendamment du processeur d'application. Il 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.

Secure Enclave utilise une mémoire chiffrée et intègre un générateur de nombres aléatoires matériel. Son micronoyau est basé sur la gamme L4, à laquelle Apple a apporté des modifications. 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.

Chaque Secure Enclave se voit attribuer à la fabrication son propre UID (identifiant unique), qui n'est ni accessible aux autres parties du système ni connu d'Apple.

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.

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 via 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 le Secure Enclave. L'échange de la clé de session se fait via 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.
- 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 de mise en veille/réactivation. 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 reconnaître 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 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.

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 pas accéder 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.

### **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 de 88 x 88 pixels et 500 ppi 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 du flux des crêtes, qui est un processus sans pertes 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 ; elles sont enveloppées avec 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

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.

L'identifiant unique (UID) et l'identifiant de groupe (GID) de l'appareil sont des clés AES 256 bits fusionnée (UID) ou compilée (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 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 ; ainsi, 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

### 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.

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

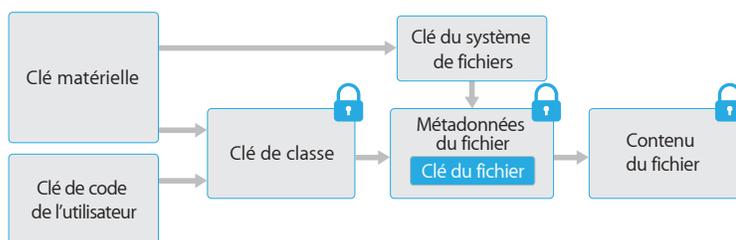
La clé par fichier est enveloppée avec l'une des clés de classe, selon les circonstances dans lesquelles le fichier doit être accessible. Comme tous les autres enveloppements, celui-ci est réalisé à l'aide de l'enveloppement à clé AES NIST, selon RFC 3394. La clé de fichier enveloppée est stockée dans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

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 stockée dans le stockage effaçable. Comme elle est stockée sur l'appareil, cette clé n'est pas utilisée pour préserver la confidentialité des données ; elle est conçue pour être effacée rapidement à la demande (par l'utilisateur, à l'aide de l'option « Effacer données et réglages », ou par un utilisateur ou administrateur émettant une commande d'effacement à distance depuis un serveur de gestion des appareils mobiles (MDM, Mobile Device Management), Exchange ActiveSync ou iCloud). Effacer la clé de cette manière rend impossible le déchiffrement des fichiers.

### Remarques à propos du code de déverrouillage

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 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 de déverrouillage

En configurant un code de déverrouillage pour un appareil, l'utilisateur active automatiquement la protection des données. iOS prend en charge les codes alphanumériques à quatre chiffres ou 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.

Le code étant combiné à l'UID de l'appareil, des attaques en force brute 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.

Pour compliquer encore plus les attaques en force brute, l'interface iOS impose des délais de plus en plus longs après la saisie d'un code non valide sur l'écran de verrouillage. Les utilisateurs peuvent configurer leur appareil de sorte que son contenu soit automatiquement effacé si un code erroné est saisi 10 fois de suite. 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 un appareil doté d'un processeur A7 ou d'une version ultérieure de la série A, les opérations cryptographiques sont réalisées par Secure Enclave, qui impose également un délai de 5 secondes entre deux tentatives de déverrouillage infructueuses. Cela vient renforcer les mesures de sécurité appliquées par iOS contre les attaques en force brute.

## Classes de protection des données

Lorsqu'un 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 de temps après le verrouillage d'un appareil (10 secondes, si le réglage « Exiger le mot de passe » est défini sur Immédiatement), la clé de classe déchiffrée est effacée, rendant toutes les données appartenant à cette classe inaccessibles jusqu'à ce que l'utilisateur saisisse à nouveau le code ou déverrouille l'appareil à l'aide de Touch ID.

### Protected Unless Open

(`NSFileProtectionCompleteUnlessOpen`) : il se peut que certains fichiers doivent être écrits pendant que l'appareil est verrouillé. 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. La KDF est la fonction de dérivation de clé par concaténation (alternative approuvée 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 ; son hachage est utilisé pour désenvelopper la clé par fichier, qui est ensuite utilisée pour 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'applications 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).

## 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'y a qu'une seule base de données ; le démon *securityd* détermine à quels éléments du trousseau peut accéder chaque processus ou app. Les API d'accès au trousseau envoient des appels au démon, qui interroge les « groupes d'accès au trousseau » de l'app et le droit « identifiant d'application ». 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 via le programme iOS Developer Program ou, dans iOS 8, 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 iOS 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.

### Composants d'un élément du 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.

Disponibilité	Protection des données des fichiers	Protection des données du trousseau
Lorsque l'appareil est déverrouillé	<code>NSFileProtectionComplete</code>	<code>kSecAttrAccessibleWhenUnlocked</code>
Lorsque l'appareil est verrouillé	<code>NSFileProtectionCompleteUnlessOpen</code>	N/D
Après le premier déverrouillage	<code>NSFileProtectionCompleteUntilFirstUserAuthentication</code>	<code>kSecAttrAccessibleAfterFirstUnlock</code>
Toujours	<code>NSFileProtectionNone</code>	<code>kSecAttrAccessibleAlways</code>
Code activé	N/D	<code>kSecAttrAccessible-WhenPasscodeSetThisDeviceOnly</code>

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` n'est disponible que si un code est configuré pour l'appareil. Les éléments appartenant à cette classe ne figurent 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 convivialité 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	Après le premier déverrouillage
Comptes Mail	Après le premier déverrouillage
Comptes Exchange	Après le premier déverrouillage
Mots de passe VPN	Après le premier déverrouillage
LDAP, CalDAV, CardDAV	Après le premier déverrouillage
Jetons des comptes de réseau social	Après le premier déverrouillage
Clés de chiffrement des annonces Handoff	Après le premier déverrouillage
Jeton iCloud	Après le premier déverrouillage
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
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. 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 CMS signé 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 8 envoie une requête TLS à chaque site Web répertorié pour demander le fichier `/apple-app-site-association`. Si la signature provient d'une identité valide pour le domaine et considérée comme fiable par iOS et que le fichier comporte l'identifiant de l'app installée, iOS marque 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 des fichiers et du trousseau sont rassemblées et gérées dans des conteneurs de clés. iOS utilise les quatre conteneurs de clés suivants : système, sauvegarde, dépôt et sauvegarde iCloud.

**Le conteneur de clés du système** 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 de déverrouillage. L'extension du noyau `AppleKeyStore` gère le conteneur de clés du système et peut être interrogée quant à 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 du système sont accessibles et qu'elles ont été correctement désenveloppées.

**Le conteneur de clés de sauvegarde** est créé lorsqu'une sauvegarde chiffrée est réalisée par iTunes et stockée 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 10 000 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 de déverrouillage 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.

Une instance spécifique d'un conteneur de clés de dépôt, appelée *conteneur de clés dissimulé*, est utilisée pendant une mise à jour logicielle pour permettre au processus de mise à jour d'accéder aux fichiers et aux éléments du trousseau quel que soit leur niveau de protection. Cet accès est requis après le redémarrage de l'appareil pendant la mise à jour afin d'exécuter les programmes de transfert de données, qui réalisent des tâches telles que la mise à jour des schémas de base de données, la génération des nouveaux aperçus des éléments ou même la mise à niveau des niveaux de protection des données.

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. Cela permet de définir de façon sécurisée dans le conteneur de clés du système un indicateur entraînant la création d'un conteneur de clés dissimulé dans la mémoire pendant que l'appareil est déverrouillé. Lorsque l'utilisateur est prêt à procéder à la mise à jour, qui ne peut avoir lieu que lorsque l'appareil est déverrouillé, le conteneur de clés dissimulé est écrit sur disque, protégé par une clé dans le stockage effaçable. Lors d'une mise à jour via iTunes depuis un hôte jumelé disposant d'un conteneur de clés de dépôt valide, les utilisateurs sont invités à déverrouiller leur appareil avant le lancement de la mise à jour pour permettre également l'écriture sur disque du conteneur de clés dissimulé pendant que l'appareil est déverrouillé.

Lors de l'exécution des programmes de transfert de données, le conteneur de clés dissimulé est chargé pour leur permettre d'accéder aux clés de classe de protection des données. À ce stade, le conteneur de clés dissimulé sur disque est effacé et la clé le protégeant est supprimée du stockage effaçable pour empêcher sa réutilisation. Une fois l'exécution du processus de transfert des données terminée, les clés généralement disponibles uniquement lorsque l'appareil est déverrouillé sont effacées, ce qui place l'appareil dans le même état qu'après le premier déverrouillage.

En cas d'échec de la création d'un conteneur de clés dissimulé avant la mise à jour, l'appareil affiche après le redémarrage l'invite « Mettre à niveau » et demande le code pour terminer le processus de mise à jour.

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.

## FIPS 140-2

La conformité des modules cryptographiques d'iOS 8 avec la norme FIPS (Federal Information Processing Standards) 140-2 niveau 1 des États-Unis est actuellement en cours de validation. Le but est de confirmer l'intégrité des opérations cryptographiques pour les apps Apple et les apps de tierce partie qui exploitent correctement les services cryptographiques d'iOS. Pour en savoir plus sur les précédentes validations et connaître le statut actuel d'iOS 8, consultez l'article [https://support.apple.com/kb/HT5808?viewlocale=fr\\_FR](https://support.apple.com/kb/HT5808?viewlocale=fr_FR).

# 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 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 plate-forme 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 iOS 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 8 permet aux développeurs d'incorporer des cadres d'application « (framework) » 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 une 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), qui est 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 iOS Developer Enterprise Program (iDEP) 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'iDEP, 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 internes vérifient également que la signature est valide à l'exécution. Les apps dont le certificat a expiré ou est révoqué ne s'exécutent pas.

Contrairement à certaines plates-formes mobiles, iOS n'autorise pas les utilisateurs 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 pas 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 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 de clés-valeurs 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 démons ou services 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 démon ou services 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 distribution aléatoire de l'espace d'adressage (ASLR, Address Space Layout Randomization) empêche l'exploitation de bogues de corruption de mémoire. Les apps intégrées utilisent l'ASLR pour garantir la distribution aléatoire 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 nombreux exploits sophistiqués. 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 distribution aléatoire 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 aléatoire. 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 les extensions 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. Les extensions n'ont pas accès 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 spécial d'extensions dans la mesure où elles sont activées par l'utilisateur pour l'ensemble du système. Une fois activée, l'extension est utilisée pour toute saisie de texte, à l'exception des codes de déverrouillage et de tout autre texte saisi dans une présentation sécurisée. Pour des raisons de confidentialité, 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 enregistrés 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 au sein de leur app.

## Groupes d'apps

Les apps et extensions provenant d'un même compte développeur peuvent partager du contenu si elles sont configurées pour appartenir à un groupe d'apps. Il appartient au développeur de créer les groupes appropriés sur le portail Apple Developer et d'y inclure les apps et extensions voulues. 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 groupe 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 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 test 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.

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 circuit intégré (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 circuit intégré 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 circuit intégré d'authentification dans le cadre du protocole Station-to-Station (STS).

## HomeKit

HomeKit propose une plate-forme de domotique faisant 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 8 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 reposant sur l'adressage IP ainsi que et la technologie Bluetooth faible 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 de protection des données Protected Until First User Authentication. Les données HomeKit ne sont sauvegardées que dans des sauvegardes chiffrées ; ainsi, les sauvegardes iTunes non chiffrées 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 et le stockage dans iCloud.

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.

Seul l'utilisateur ayant initialement créé le domicile dans HomeKit peut ajouter des utilisateurs. Son appareil configure les accessoires avec la clé publique du nouvel utilisateur afin qu'ils puissent authentifier et accepter les commandes de ce dernier. La procédure de configuration de l'Apple TV en vue d'une utilisation avec HomeKit fait appel à la même authentification et au même chiffrement que lors de l'ajout d'utilisateurs, mais elle est réalisée automatiquement si l'utilisateur ayant créé le domicile est connecté à iCloud sur l'Apple TV, et que l'Apple TV se trouve au domicile.

Si un utilisateur n'a qu'un seul appareil et qu'il n'accorde pas 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 les noms 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.

## 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, comme décrit dans la rubrique Siri du présent livre blanc, afin de communiquer les noms des pièces, des accessoires et des scènes nécessaires à la reconnaissance des commandes.

## HealthKit

Le cadre d'application HealthKit fournit une base de données commune que les apps peuvent utiliser pour stocker et consulter des données 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 M7 ou M8 intégré à de nombreux appareils iOS.

### Données de santé

HealthKit utilise une base de données pour stocker les données de santé de l'utilisateur, comme la taille, le poids, la distance parcourue, la tension artérielle et plus encore. Cette base de données est associée à la classe de protection des données Complete Protection, ce qui signifie qu'elle n'est accessible qu'après que l'utilisateur ait déverrouillé l'appareil en saisissant son code de déverrouillage ou en utilisant Touch ID.

Une autre base de données stocke les données de fonctionnement, comme les tables 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. Cette base de données est associée à la classe de protection des données 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 consolidation terminée.

Les données de santé ne sont pas partagées via iCloud ni synchronisées entre les appareils. Les bases de données de santé sont incluses dans les sauvegardes chiffrées de l'appareil dans iCloud ou iTunes. 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 fournie avec iOS 8 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.

# 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 n'importe où 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.

## SSL, TLS

iOS prend en charge les protocoles Secure Socket Layer (SSL 3), Transport Layer Security (TLS 1.0, TLS 1.1, TLS 1.2) 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 niveau de contrôle plus fin.

## VPN

Les services réseau sécurisés comme les réseaux privés virtuels ne nécessitent généralement qu'une configuration minimale 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 :

- VPN SSL Juniper Networks, Cisco, Aruba Networks, SonicWALL, Check Point, Palo Alto Networks, Open VPN, AirWatch, MobileIron, NetMotion Wireless et F5 Networks 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. Le protocole IPSec Cisco prend en charge le VPN à la demande pour les domaines spécifiés lors de la configuration de l'appareil.
- L2TP/IPSec avec authentification utilisateur par mot de passe MS-CHAPV2, RSA SecurID ou CRYPTOCARD et authentification machine par secret partagé.
- PPTP avec authentification utilisateur par mot de passe MS-CHAPV2 et RSA SecurID ou CRYPTOCARD.

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 8 introduit 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'appareils (DEP, Device Enrollment Program). Cela évite aux utilisateurs d'avoir à activer le VPN pour être protégés lorsqu'ils se connectent à des réseaux Wi-Fi. Le VPN permanent offre à une entreprise un contrôle complet sur le trafic des appareils en tunnelisant tout le trafic IP jusqu'à elle. Le protocole de tunnelisation 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 iPhone et sur iPad comprennent EAP-TLS, EAP-TTLS, EAP-FAST, EAP-SIM, PEAPv0, PEAPv1 et LEAP.

iOS 8 utilise une adresse Media Access Control (MAC) aléatoire pour effectuer des recherches Preferred Network Offload (PNO) lorsqu'un appareil n'est pas associé à un réseau Wi-Fi et que son processeur est en veille. Le processeur d'un appareil se met en veille peu de temps après l'extinction de l'écran. Des recherches PNO sont alors exécutées pour déterminer si l'utilisateur peut se connecter à un réseau Wi-Fi préféré afin de réaliser des tâches comme une synchronisation sans fil avec iTunes.

iOS 8 utilise également une adresse MAC aléatoire 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 des périmètres virtuels (« geofences »), 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 n'est pas connecté à 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.

Nous avons travaillé avec les fabricants de cartes Wi-Fi pour les informer que les recherches en arrière-plan utilisent une adresse MAC aléatoire, mais ni Apple ni les fabricants ne peuvent prédire ces adresses MAC aléatoires.

Les adresses MAC Wi-Fi aléatoires sont supportées par l'iPhone 5c, 5s, 6 et 6 Plus, ainsi que l'iPad Air et l'iPad mini avec écran Retina.

## 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 l'article [https://support.apple.com/kb/ht3647?viewlocale=fr\\_FR](https://support.apple.com/kb/ht3647?viewlocale=fr_FR).

## 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.

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 intégrés à Windows prenant en charge les tickets Kerberos. L'authentification par certificat est également prise en charge. 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 Yosemite. 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 (TLS) chiffrée 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(e). Si plusieurs destinataires ont été sélectionnés, ce processus est répété pour chaque destination.

En mode Tout 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.

# 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 plate-forme. 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 (Apple ID) correspond à un nom d'utilisateur et à un mot de passe utilisés 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 couramment 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 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.

Apple propose aussi un système de vérification d'identifiant Apple en deux étapes qui ajoute un deuxième niveau de sécurité au compte de l'utilisateur. Lorsque la vérification en deux étapes 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 ou tout achat sur l'iTunes Store, l'iBooks Store ou l'App Store à partir d'un nouvel appareil. Cela permet d'interdire à quiconque d'accéder au compte d'un utilisateur même en connaissant son mot de passe. 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 vérification d'identifiant Apple en deux étapes, visitez [https://support.apple.com/kb/ht5570?viewlocale=fr\\_FR](https://support.apple.com/kb/ht5570?viewlocale=fr_FR).

## 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.

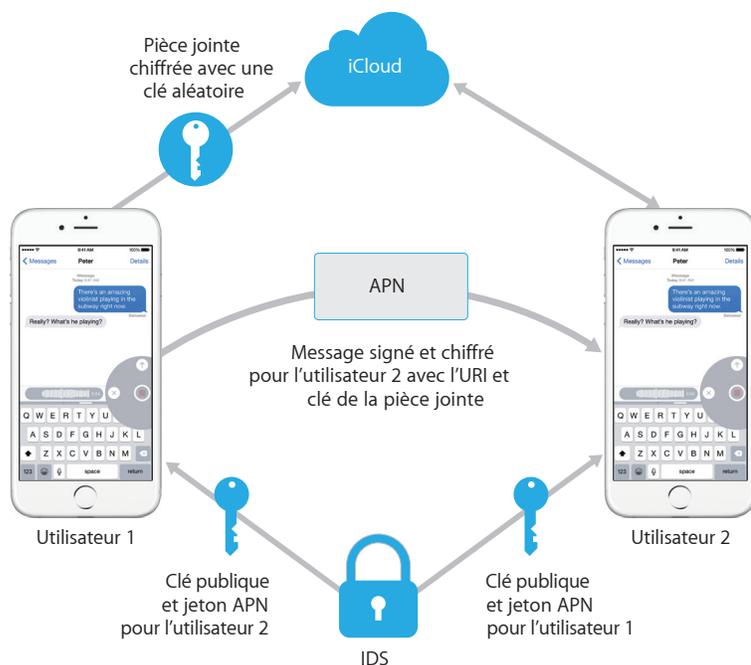
Au fur et à mesure que les utilisateurs activent des appareils supplémentaires à utiliser avec iMessage, leurs clés publiques de chiffrement et de signature, les adresses de service APN et les numéros de téléphone associés sont ajoutés au service de répertoire. Les utilisateurs ont également la possibilité 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 clé aléatoire 128 bits qu'il utilise pour chiffrer le message via AES en mode CTR. Cette clé AES propre à chaque message est chiffrée via RSA-OAEP avec 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 15 jours au maximum ; il est prévu d'étendre cette période à 30 jours dans un avenir proche.

## 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 utilise le protocole ICE (Internet Connectivity Establishment) pour établir une connexion poste-à-poste entre des appareils. Les appareils vérifient leurs certificats d'identification à l'aide de messages SIP (Session Initiation Protocol) et établissent un secret partagé pour chaque session. Les nonces cryptographiques fournis par chaque appareil sont combinées à des clés aléatoires (salage) pour chacun des canaux de support, qui sont ensuite diffusées via 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 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 IT 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

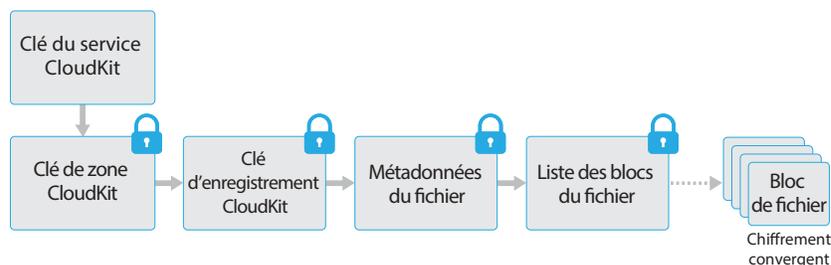
iCloud Drive ajoute des clés basées sur le compte pour protéger les documents enregistrés dans iCloud. À l'instar des autres services iCloud, il divise le contenu des fichiers en plusieurs blocs qu'il chiffre avant de stocker les blocs chiffrés par le biais de services tiers. Les clés de fichier sont 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 afficher les parties protégées du service de stockage iCloud Drive.

### CloudKit

CloudKit permet aux développeurs d'applications d'enregistrer des données de type clé-valeur, des données structurées et des ressources dans iCloud. L'accès à CloudKit est contrôlé au moyen de droits d'application. 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'application, 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 :

- informations relatives à la musique, aux vidéos, aux émissions de TV, aux apps et aux livres achetés, mais pas le contenu acheté lui-même ;
- photos et vidéos de la Pellicule ;
- contacts, événements de calendrier, rappels et notes ;
- réglages d'appareil ;
- données d'application ;
- fichiers PDF et livres ajoutés à iBooks, mais pas achetés ;
- historique des appels téléphoniques ;
- organisation de l'écran d'accueil et des apps ;
- messages texte (SMS) et MMS d'iMessage ;
- sonneries ;
- données HomeKit ;
- données HealthKit ;
- messagerie vocale visuelle.

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

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 partie tierce 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 (salage 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.

### Intégration de Safari et du trousseau iCloud

Safari peut générer automatiquement des chaînes de caractères aléatoires cryptographiquement complexes à utiliser comme des 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 Apple ni par d'autres appareils.

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 de 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 dans le cercle de signature et chacun possède la clé publique de son homologue. Ils commencent alors à s'échanger des éléments de trousseau individuels via l'espace de stockage des clé-valeurs 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 de crédit), 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 asymétriques et la place dans la zone de stockage des clé-valeurs 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 HSM (module de sécurité matériel) 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 code personnel ou simple code à 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. Des clusters de modules de sécurité matériels (HSM) positionnés derrière iCloud assurent la protection des enregistrements en 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 via 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 clé-valeurs 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 permettre la mise 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 rappel, 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.

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. De plus, certains enregistrements faisant référence à la musique, aux équipes sportives et à leurs joueurs, ainsi qu'au monde des affaires et aux points d'intérêt sont enregistrés de la même manière 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 via Siri.

## Continuité

Continuité, une nouveauté d'iOS 8 et d'OS X Yosemite, 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 de texte et de partager une connexion Internet mobile.

### Handoff

Avec Handoff, l'utilisateur peut placer son ordinateur Mac à côté d'un appareil 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 messages individuels (comme dans iMessage). Dans certains cas, ces messages sont transférés via le service APN plutôt que par Bluetooth Low Energy. 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 accreditations 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 dans les octets d'annonce Handoff chiffrés. Seuls les autres appareils de l'utilisateur sont capables de déchiffrer les octets d'annonce (voir description à la section précédente).

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'application 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.

#### **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 via 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.

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 de 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 découvrir, et communiquer avec, les appareils connectés au même compte iCloud. Les ordinateurs Mac exécutant OS X Yosemite utilisent la même technologie pour découvrir, et communiquer avec, les appareils iOS prenant en charge le partage de connexion instantané.

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 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. La requête est envoyée via une liaison chiffrée au moyen d'un algorithme standard Bluetooth Low Energy et chiffrée à l'aide de la méthode de chiffrement utilisée pour 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 Spotlight**

Les fonctionnalités de recherche de Safari et de Spotlight comprennent désormais des Suggestions Spotlight, c'est-à-dire des résultats de recherche provenant d'Internet, d'iTunes, de l'App Store, des programmes de cinéma, des lieux les plus proches, etc.

Pour faire des suggestions plus pertinentes aux utilisateurs, la fonctionnalité Suggestions Spotlight inclut le contexte de l'utilisateur et une évaluation des résultats de recherche avec les demandes de recherche envoyées à Apple.

Le contexte envoyé avec les demandes de recherche permet à Apple de connaître :

- i) la position géographique approximative de l'appareil ;
- ii) le type d'appareil (Mac, iPhone, iPad ou iPod, par exemple) ;
- iii) l'app cliente, à savoir Spotlight ou Safari ;
- iv) les réglages d'appareil relatifs à la langue et à la région ;
- v) les trois dernières apps utilisées sur l'appareil ;
- vi) un identifiant de session anonyme.

Toutes les communications avec le serveur sont chiffrées via HTTPS.

Pour mieux respecter la vie privée de l'utilisateur, les Suggestions Spotlight n'envoient jamais la position exacte du client, mais rendent cette position plus approximative avant de l'envoyer. Le niveau d'approximation utilisé dépend de la densité de population estimée à l'endroit où se trouve l'appareil ; ce niveau est plus important dans les zones rurales que dans les zones urbaines où les utilisateurs sont généralement plus proches les uns des autres. Les utilisateurs ont de plus 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 Spotlight dans les Réglages. Si le Service de localisation est désactivé, Apple peut utiliser l'adresse IP du client pour inférer une position approximative.

L'identifiant de session anonyme permet à Apple de détecter des éléments récurrents dans les recherches effectuées au cours d'une période de 15 minutes. Ainsi, si les utilisateurs recherchent fréquemment l'expression « numéro de téléphone de restaurant » après avoir recherché « restaurant », Apple peut décider d'inclure le numéro de téléphone dans les résultats de recherche. Toutefois, contrairement à la plupart des moteurs de recherche, le service de recherche d'Apple n'utilise pas d'identifiant personnel permanent dans tout l'historique de recherche d'un utilisateur pour associer les recherches à cet utilisateur ou à son appareil ; au lieu de cela, les appareils Apple utilisent un identifiant de session anonyme temporaire qui est éliminé après 15 minutes.

L'information relative aux trois dernières apps utilisées sur l'appareil est incluse pour préciser le contexte de la recherche. Pour mieux respecter la vie privée des utilisateurs, seuls les apps reprises dans une liste blanche tenue à jour par Apple et contenant les apps populaires utilisées au cours des trois dernières heures sont incluses.

L'évaluation de recherche envoyé à Apple fournit à la société : i) la durée écoulée entre les actions de l'utilisateur (pressions sur les touches, par exemple et les sélections de résultat) ; ii) le résultat Suggestions Spotlight sélectionné éventuel ; et iii) le type de résultat local sélectionné (« Signet » ou « Contact » par exemple). Comme pour le contexte, l'évaluation de recherche n'est pas associée à un individu ou un appareil particulier.

Apple conserve pendant 18 mois au maximum des historiques de Suggestions Spotlight comprenant les requêtes, le contexte et le retour. Des historiques plus réduits, comprenant uniquement la requête, le pays, la langue, la date (heure incluse), ainsi que le type d'appareil, sont conservés pendant deux ans au maximum. Les adresses IP ne sont pas conservées avec les historiques de requêtes.

Dans certains cas, Suggestions Spotlight peut transférer des requêtes relatives à des mots et des phrases courants à un partenaire agréé, afin de recevoir et d'afficher les résultats de recherche de ce partenaire. Ces requêtes ne sont pas conservées par le partenaire agréé et ce dernier ne reçoit pas d'évaluation de recherche. Les partenaires ne reçoivent pas non plus les adresses IP des utilisateurs. La communication avec le partenaire est chiffrée via HTTPS. Apple fournit au partenaire un contexte de recherche comprenant la ville, le type d'appareil et la langue du client en fonction des lieux, des types d'appareils et des langues identifiés par Apple comme étant des éléments récurrents dans les recherches.

Il est possible de désactiver Suggestions Spotlight dans les Réglages pour Spotlight, pour Safari ou pour les deux. Si la fonctionnalité Suggestions Spotlight est désactivée pour Spotlight, ce dernier ne fonctionne plus alors que comme un client de recherche local limité à l'appareil et ne transmettant aucune information à Apple. Si la fonctionnalité est désactivée pour Safari, les requêtes de recherche de l'utilisateur, le contexte de recherche et le retour de recherche ne sont pas envoyés à Apple.

# Contrôle des appareils

La plate-forme 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 de déverrouillage, 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 de déverrouillage

Outre la protection cryptographique décrite précédemment dans ce document, les codes de déverrouillage offrent un moyen d'interdire tout accès non autorisé à l'interface utilisateur des appareils. L'interface iOS impose des délais progressifs après la saisie d'un code invalide, ce qui permet de réduire de manière significative l'efficacité des attaques en force brute au niveau de l'écran de verrouillage. Les utilisateurs peuvent opter pour l'effacement automatique de l'appareil après 10 tentatives manquées consécutives de saisie du code. Cette option, disponible en tant que règle administrative, peut être réglée sur un seuil inférieur à l'aide de profils de configuration, d'une solution MDM ou du protocole Exchange ActiveSync.

L'utilisateur peut choisir par défaut un code constitué de quatre chiffres. Il est possible de spécifier des codes alphanumériques plus longs via Réglages > Général > Code > Code complexe. 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 ;
- longueur de code minimum ;
- nombre minimum de caractères complexes ;
- âge de code maximum ;
- historique des codes ;
- délai de blocage automatique ;
- période de grâce pour le blocage de l'appareil ;
- nombre maximum de tentatives manquées ;
- autoriser Touch ID.

Pour obtenir plus de détails sur chaque politique, consultez l'article intitulé « Configuration Profile Key Reference » (en anglais), à l'adresse <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.

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 1024 bits. L'hôte reçoit ensuite une clé 256 bits capable de débloquent un conteneur de clés en dépôt sur l'appareil (voir la partie consacrée aux Conteneurs de clés en 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. Pour en savoir plus à ce sujet, voir [https://support.apple.com/kb/HT6331?viewlocale=fr\\_FR](https://support.apple.com/kb/HT6331?viewlocale=fr_FR).

Sous iOS 8, 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 sous iOS 8.

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, lisez l'article [https://support.apple.com/kb/HT5868?viewlocale=fr\\_FR](https://support.apple.com/kb/HT5868?viewlocale=fr_FR).

## Application de la configuration

Un profil de configuration est un fichier XML qui permet à un administrateur de distribuer des données de configuration à plusieurs 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 de courrier électronique, 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 politiques de code ;
- 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 courrier ;
- les réglages Exchange ;

- les réglages de service de répertoire LDAP ;
- les réglages de service de calendrier CalDAV ;
- les clips Web ;
- les accréditations et les clés ;
- les réglages avancés de réseau cellulaire.

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 (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 technologies iOS existantes intègrent des capacités MDM telles que les profils de configuration, l'inscription par connexion sans fil et le service APN. Le service APN, par exemple, est utilisé pour réveiller l'appareil afin qu'il puisse communiquer directement avec son serveur MDM via 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 via 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 des appareils mobiles, voir <https://www.apple.com/iphone/business/it/management.html>.

## Programme d'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 chez Apple. Cette organisation peut ainsi intégrer automatiquement ces appareils dans son système de gestion MDM sans avoir à les manipuler physiquement ni à les préparer avant de les remettre à leurs utilisateurs. 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 en outre autoriser ou non chaque utilisateur à supprimer le profil MDM de son appareil. Ils peuvent par exemple commander les appareils chez Apple, configurer tous les réglages de gestion et livrer directement les appareils au domicile des utilisateurs. 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 prêts pour l'utilisateur.

Le processus est simple : après l'inscription dans le programme, les administrateurs se connectent au site Web du programme, associent le programme à leur serveur MDM et « prennent possession » des appareils iOS achetés auprès d'Apple. Les appareils 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. Pour en savoir plus à ce sujet, voir <https://deploy.apple.com>.

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

## Apple Configurator

Outre le système de gestion MDM, Apple Configurator pour OS X simplifie le déploiement d'appareils iOS pour tout le monde. Apple Configurator peut être utilisé pour configurer rapidement les réglages, les apps et les données d'un grand nombre d'appareils. Les appareils initialement configurés au moyen d'Apple Configurator peuvent être « supervisés », afin qu'il soit possible d'y installer des réglages et des restrictions supplémentaires. Une fois qu'un appareil est supervisé à l'aide d'Apple Configurator, tous les réglages et restrictions disponibles peuvent également être installés sans fil via une solution MDM.

Pour en savoir plus sur la configuration et la gestion d'appareils à l'aide d'une solution MDM ou d'Apple Configurator, consultez la Référence pour le déploiement iOS, à l'adresse <https://help.apple.com/deployment/ios>.

## Restrictions d'appareils

Les administrateurs ont la possibilité de restreindre l'utilisation de certaines fonctionnalités sur les appareils en installant un profil de configuration. Ils disposent ainsi des restrictions suivantes :

- autoriser les installations d'application ;
- autoriser l'utilisation de la caméra ;
- autoriser FaceTime ;
- autoriser les captures d'écran ;
- autoriser la composition vocale ;
- 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 les jeux multijoueurs ;
- autoriser l'ajout d'amis dans le Game Center ;
- autoriser Siri ;
- autoriser Siri lorsque l'appareil est verrouillé ;
- autoriser l'utilisation de YouTube ;
- autoriser les notifications Passbook lorsque l'appareil est verrouillé ;
- autoriser l'utilisation de l'iTunes Store ;
- autoriser les médias à contenu explicite ;
- autoriser le contenu érotique dans l'iBooks 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 le 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 Safari ;
- activer les Suggestions Spotlight dans Spotlight ;
- autoriser Handoff ;
- 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 ;
- restreindre les classifications de films ;
- restreindre les classifications TV ;
- restreindre les classifications d’applications ;
- 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 ;
- autoriser les flux de photos ;
- autoriser les flux de photos partagés ;
- autoriser l’envoi de diagnostics à Apple ;
- autoriser l’utilisateur à accepter les certificats TLS non fiables ;
- forcer le chiffrement des sauvegardes ;
- restreindre les médias en fonction de la classification de leur contenu ;
- 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.

## Restrictions supplémentaires des appareils supervisés uniquement

- autoriser iMessage ;
- autoriser le Game Center ;
- autoriser l’iBooks Store ;
- autoriser la suppression des applications ;
- autoriser le filtre de grossièretés de Siri ;
- autoriser l’installation manuelle des profils de configuration ;
- 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 les podcasts ;
- 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 cellulaires ;
- autoriser le jumelage avec un hôte (iTunes) ;
- autoriser le verrouillage d'activation ;
- interdire Effacer contenu et réglages ;
- interdire l'activation des restrictions ;
- filtre de contenus de tierce partie ;
- mode App individuelle autonome ;
- VPN permanent.

## 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 du système de fichiers située dans la zone de stockage effaçable (Eraseable 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. Et, 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 de déverrouillage.

## Localiser mon iPhone et verrouillage d'activation

Il est important de pouvoir désactiver et effacer tout le contenu d'un appareil perdu ou volé. Avec iOS 7 ou ultérieur, il est impossible de réactiver un appareil sans saisir les accreditations d'identifiant Apple du propriétaire lorsque la fonctionnalité Localiser mon iPhone est activée. Il est conseillé aux organisations de superviser leurs appareils ou de mettre en place une politique permettant aux utilisateurs de désactiver la fonctionnalité Localiser mon iPhone, afin qu'elle n'empêche pas l'organisation d'attribuer l'appareil à un autre individu.

Avec iOS 7.1 ou ultérieur, les solutions MDM compatibles peuvent autoriser le verrouillage d'activation sur des appareils supervisés lorsque l'utilisateur active la fonctionnalité Localiser mon iPhone. Les administrateurs MDM peuvent gérer le verrouillage d'activation de la fonctionnalité Localiser mon iPhone en supervisant les appareils à l'aide d'Apple Configurator ou du programme d'inscription d'appareils (DEP). La solution MDM 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 plus de détails à ce sujet, consultez la documentation de votre solution MDM.

Important : 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 le 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, le verrouillage sera activé dès que l'utilisateur aura activé Localiser mon iPhone.

# Protection des données personnelles

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 données, le moment où elles le font et la nature des données 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 les 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 les Réglages. Dans les 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 les 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 sur iPhone 5s ou ultérieur
- Comptes de réseaux sociaux tels que Twitter et Facebook
- Micro
- Caméra
- HomeKit
- HealthKit
- 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 les 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>.

# 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 plate-forme 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 (Extended Validation) 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, depuis iOS 8, 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 des autres plate-formes. 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 plate-forme.

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, rendez-vous sur [apple.com/fr/support/security](https://apple.com/fr/support/security).

# Glossaire

<b>APN (Apple Push Notification), service de notification Push d'Apple</b>	Service mondial offert par Apple pour fournir des notifications de type Push aux appareils iOS.
<b>ASLR (Address Space Layout Randomization), distribution aléatoire de l'espace d'adressage</b>	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 les versions ultérieures, 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.
<b>Carte intelligente</b>	Circuit intégré incorporé qui fournit un stockage de données, une authentification et une identification sécurisés.
<b>Cartographie angulaire de la direction des crêtes</b>	Représentation mathématique de la direction et de la largeur des crêtes extraites d'une partie d'empreinte digitale.
<b>Circuit intégré (CI)</b>	Également appelé microprocesseur.
<b>Clé de fichier</b>	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.
<b>Clé du système de fichiers</b>	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.
<b>Conteneur de clés</b>	Structure de données utilisée pour stocker une collection de clés de classe. Chaque type (système, sauvegarde, dépôt ou Sauvegarde iCloud) possède le même format : <ul style="list-style-type: none"><li>• Un en-tête contenant :<ul style="list-style-type: none"><li>– la version (réglée sur 3 dans iOS 5) ;</li><li>– le type (système, sauvegarde, dépôt ou Sauvegarde iCloud) ;</li><li>– l'UUID du conteneur de clés ;</li><li>– un code HMAC si le conteneur de clés est signé ;</li><li>– la méthode utilisée pour envelopper les clés de classe : emmêlées avec l'UUID ou PBKDF2, ainsi que le salage et le nombre d'itérations.</li></ul></li><li>• Une liste de clés de classe :<ul style="list-style-type: none"><li>– UUID de clé ;</li><li>– classe (classe de protection des données de trousseau ou de fichier) ;</li><li>– type d'enveloppe (clé dérivée de l'UUID uniquement ou clé dérivée de l'UUID et clé dérivée du code) ;</li><li>– clé de classe enveloppée ;</li><li>– clé publique pour classes asymétriques.</li></ul></li></ul>
<b>Data Protection (Protection des données)</b>	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.
<b>DFU (Device Firmware Upgrade), mise à niveau de logiciel interne d'appareil</b>	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 afin qu'il puisse être utilisé avec iTunes. »
<b>ECID</b>	Identifiant 64 bits propre au processeur de chaque appareil iOS. Il est utilisé dans le cadre du processus de personnalisation et n'est pas considéré comme un secret.

<b>Effaceable Storage (Stockage effaçable)</b>	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.
<b>Emmèlement (Tangling)</b>	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'emmèlement est le PBKDF2 qui utilise une clé AES avec l'UID de l'appareil comme fonction PRF pour chaque itération.
<b>Enveloppement de clé</b>	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.
<b>GID (identifiant de groupe)</b>	Semblable à l'UID, mais commun à tous les processeurs d'une classe.
<b>HSM (Hardware Security Module), module de sécurité matériel</b>	Ordinateur spécialisé, protégé contre toute manipulation, utilisé pour sauvegarder et gérer des clés numériques.
<b>iBoot</b>	Code chargé par LLB et qui charge à son tour XNU, dans le cadre d'une chaîne de démarrage sécurisée.
<b>IDS (Identity Service), service d'identité</b>	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.
<b>JTAG (Joint Test Action Group)</b>	Outil de débogage matériel standard utilisé par les programmeurs et les développeurs de circuits.
<b>LLB (Low-Level Bootloader), chargeur de démarrage de bas niveau</b>	Code invoqué par la ROM de démarrage, qui charge à son tour l'iBoot dans le cadre d'une chaîne de démarrage sécurisée.
<b>Profil d'approvisionnement</b>	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'application d'une app développée par une entreprise.
<b>ROM de démarrage (boot)</b>	Tout premier code exécuté par le processeur d'un appareil lors du démarrage de ce dernier. Comme ce code fait partie intégrante du processeur, il ne peut être modifié ni par Apple ni par un attaquant.
<b>SoC (System on a chip), puce-système</b>	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.
<b>Trousseau</b>	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 accreditations sensibles.
<b>URI (Uniform Resource Identifier), identifiant de ressource uniforme</b>	Chaîne de caractères permettant d'identifier une ressource Web.
<b>UID (Unique ID), identifiant unique</b>	Clé AES 256 bits gravée sur chaque processeur au moment de sa fabrication. Elle ne peut être lue 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.
<b>XNU</b>	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).

© 2014 Apple Inc. Tous droits réservés. Apple, le logo Apple, AirDrop, AirPlay, Bonjour, FaceTime, iBooks, iMessage, iPad, iPhone, iPod, iPod touch, iTunes, Keychain, Mac, OS X, Passbook, Safari, Siri, Spotlight et Xcode sont des marques d'Apple Inc., déposées aux États-Unis et dans d'autres pays. Lightning et Touch ID sont des marques d'Apple Inc. iCloud et iTunes Store sont des marques de service d'Apple Inc., déposées aux États-Unis et dans d'autres pays. App Store et iBooks Store sont des marques de service d'Apple Inc. iOS est une marque ou une marque déposée de Cisco aux États-Unis et dans d'autres pays et utilisée sous licence. L'appellation et le logo Bluetooth® sont des marques déposées détenues par Bluetooth SIG, Inc. Toute utilisation de ces marques par Apple est effectuée sous licence. Java est une marque déposée d'Oracle et/ou de ses filiales. Les autres noms de produit et de société mentionnés dans le présent document sont des marques de leurs détenteurs respectifs. Les caractéristiques des produits sont indiquées sous réserve de modification sans avis préalable. Octobre 2014