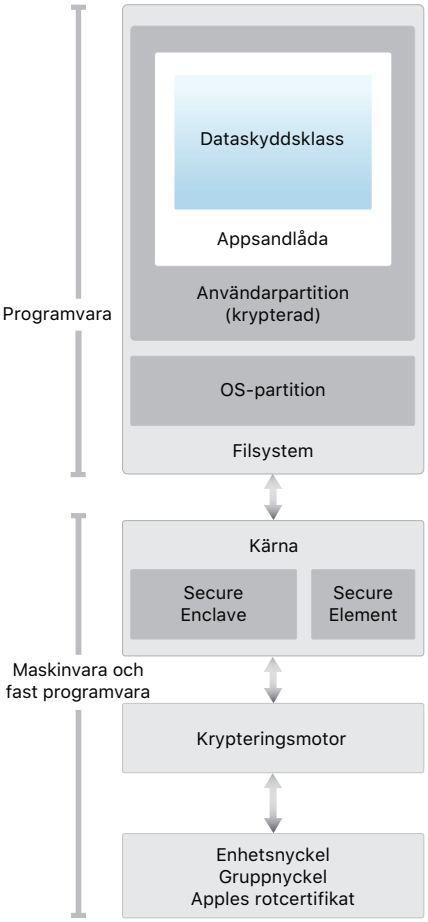
iOS-säkerhet

iOS 12.1

November 2018

Innehåll

Inledning



Den schematiska bilden av säkerhetsarkitekturen i iOS ger en överblick av de olika tekniker vi tar upp i det här dokumentet.

Apple utformade iOS från grunden med fokus på säkerhet. När vi föresatte oss att skapa den bästa möjliga mobilplattformen hade vi decennier av erfarenhet bakom oss och kunde bygga upp en helt ny arkitektur. Vi tänkte igenom de säkerhetsrisker som kan finnas i datoroperativsystem och tog oss an säkerhetsfrågorna på ett nytt sätt när vi utvecklade iOS. Vi utvecklade och använde innovativa funktioner som ökar mobilsäkerheten och skyddar hela systemet som förval. Resultatet är att iOS innebär ett stort steg framåt när det gäller säkerhet för mobila enheter.

Alla iOS-enheter är uppbyggda av programvara, maskinvara och tjänster som har utformats för att fungera tillsammans och ge maximal säkerhet och en transparent användarupplevelse. iOS skyddar inte bara enheten och de data som lagras på den, utan hela ekosystemet – det vill säga allt som användaren gör lokalt, via nätverk och med hjälp av internettjänster.

Säkerhetsfunktionerna i iOS och på iOS-enheterna är avancerade och samtidigt lätta att använda. Många av funktionerna är aktiverade som förval, så IT-avdelningen behöver inte utföra några omfattande konfigurationer. Viktiga säkerhetsfunktioner som enhetskryptering kan inte ens konfigureras, så det finns ingen risk att användarna stänger av dem av misstag. Andra funktioner, som Face ID, ger en bättre användarupplevelse eftersom det blir enklare och mer intuitivt att skydda enheten.

I det här dokumentet hittar du information om hur säkerhetstekniken och säkerhetsfunktionerna används i iOS. Här finns också praktisk information för företag och organisationer som vill kombinera säkerhetstekniken i iOS med egna policyer och rutiner som uppfyller deras säkerhetsbehov.

Dokumentet täcker in följande ämnesområden:

- **Systemsäkerhet:** Den integrerade och säkra programvara och maskinvara som utgör plattformen för iPhone, iPad och iPod touch.
- **Kryptering och dataskydd:** Den arkitektur och design som skyddar användarnas data om enheten tappas bort eller blir stulen eller om någon obehörig försöker använda eller ändra den.
- **Appsäkerhet:** De system som gör att appar kan köras säkert och utan att äventyra plattformens integritet.
- **Nätverkssäkerhet:** Nätverksprotokoll enligt branschstandarder som ger säker autentisering och kryptering av data vid överföring.
- **Apple Pay:** Apples lösning för säkra betalningar.
- **Internettjänster:** Apples nätverksbaserade infrastruktur för meddelanden, synkronisering och säkerhetskopiering.
- **Hantering av användarlösenord:** Lösenordsbegränsningar och tillgång till lösenord från andra auktoriserade källor.
- **Enhetshantering:** Metoder som gör det möjligt att hantera iOS-enheter, förhindra obehörig användning och aktivera fjärradering om en enhet försvinner eller blir stulen.
- **Integritetsinställningar:** Funktioner i iOS som kan användas till att reglera tillgången till platstjänster och användardata.
- **Säkerhetscertifieringar och program:** Information om ISO-certifieringar, kryptografisk verifiering, Common Criteria Certification och Commercial Solutions for Classified (CSfC).

Systemsäkerhet

Gå in i DFU-läge (Device Firmware Upgrade)

Återhämtning av en enhet efter att den går in i DFU-läge (kallas även för återhämtningsläge) återställer den till ett känt fungerande tillstånd som garanterat endast innehåller oförändrad Apple-signerad kod. DFU-läget kan aktiveras manuellt.

Börja med att ansluta enheten till en dator via en USB-kabel.

Beroende på enhet gör du följande:

iPhone X eller senare, iPhone 8 eller iPhone 8 Plus. Tryck på knappen som höjer volymen och släpp den sedan snabbt. Tryck på knappen som sänker volymen och släpp den sedan snabbt. Håll in sidoknappen och tryck sedan igen på knappen som sänker volymen. Släpp sidoknappen efter fem sekunder och fortsätt att hålla in knappen som sänker volymen tills du ser skärmen för återhämtningsläge.

iPhone 7 eller iPhone 7 Plus. Håll in sidoknappen och knappen som sänker volymen samtidigt. Släpp sidoknappen och fortsätt att hålla in knappen som sänker volymen tills du ser skärmen för återhämtningsläge.

iPhone 6s och tidigare, iPad eller iPod touch. Håll in både hemknappen och överknappen (eller sidoknappen) samtidigt. Släpp den övre knappen (eller sidoknappen) och fortsätt att hålla in hemknappen tills du ser skärmen för återhämtningsläge.

Obs! Ingenting visas på skärmen medan enheten är i DFU-läge. Om Apple-logotypen visas har sido- eller vilo-/väckningsknappen hållits in för länge.

Systemsäkerheten har utformats så att både programvara och maskinvara skyddas i alla kärnkomponenter på alla iOS-enheter. Det innefattar systemstartprocessen, programuppdateringar och Secure Enclave. Den här arkitekturen är central för säkerheten i iOS och stör aldrig användarupplevelsen.

iOS-enheternas täta integrering mellan maskinvara, programvara och tjänster gör att varje komponent i systemet är betrodd och validerar systemet som helhet. Varje steg – från systemstart till iOS-programuppdateringar och tredjepartsappar – analyseras och granskas så att maskinvaran och programvaran fungerar optimalt tillsammans i alla lägen och nyttjar resurserna på rätt sätt.

Säker startsekvens

Varje steg i startprocessen innehåller komponenter som är kryptografiskt signerade av Apple för att garantera säkerheten, och processen går inte vidare förrän varje tillförlitlighetssteg har verifierats. Här ingår bootloader-komponenterna, kärnan, tillägg till kärnan och den fasta programvaran för basbandet. Den här säkra startsekvensen hjälper till att garantera att den mest grundläggande programvaran inte har manipulerats.

När en iOS-enhet slås på körs omedelbart kod från ett skrivskyddat startminne som kallas **Boot ROM**. Den här statiska koden, som kallas betrodd rot för maskinvaran, skapas när kretsen tillverkas och är implicit betrodd. Startminneskoden innehåller den publika nyckeln för Apples rotcertifikatutfärdare som används för att verifiera att iBoot-bootloadern är signerad av Apple innan den läses in. Det här är det första steget i tillförlitlighetskedjan där varje steg garanterar att nästa är signerat av Apple. När iBoot är klar med sina uppgifter verifierar och kör den iOS-kärnan. För enheter med en A9- eller tidigare A-serieprocessor läses ytterligare
ett **LLB-steg (Low-Level Bootloader)** in och verifieras av startminnet (Boot ROM), och det läser i sin tur in och verifierar iBoot.

Om startminnet inte kan läsa in LLB (i äldre enheter) eller iBoot (i nyare enheter) försätts enheten i DFU-läge. Om LLB eller iBoot inte kan läsa in eller verifiera nästa steg stoppas startprocessen och ett meddelande om att enheten måste anslutas till iTunes visas på skärmen. Det här kallas för återhämtningsläge. I båda fallen måste enheten anslutas till iTunes via en USB-kabel och återställas till fabriksinställningarna.

BPR (Boot Progress Register) används av Secure Enclave för att begränsa tillgången till användardata i olika lägen och uppdateras innan enheten försätts i följande lägen:

- **Återhämtningsläge:** Ställs in av iBoot på enheter med Apple A10, S2 eller nyare med **SoC (system on chip)**.
- **DFU-läge:** Ställs in av Boot ROM på enheter med en A12-SoC.

Mer information finns i avsnittet Kryptering och dataskydd i det här dokumentet.

På enheter med möjlighet till mobilanslutning använder basbandets undersystem också en egen, liknande process för säker start med signerad programvara och nycklar som verifieras av basbandsprocessorn.

Secure Enclave-coprocessorn använder också en säker startprocess som garanterar att dess egen programvara är verifierad och signerad av Apple. Se avsnittet Secure Enclave i det här dokumentet.

Mer information om hur du aktiverar återhämtningsläget manuellt finns på https://support.apple.com/sv-se/HT1808

Auktorisering av systemprogram

Apple släpper regelbundet programuppdateringar för att åtgärda nya säkerhetsproblem och även för att erbjuda nya funktioner. Dessa uppdateringar släpps för alla enheter som stöds samtidigt. Användarna får meddelanden om iOS-uppdateringar direkt på enheten och via iTunes. Uppdateringarna skickas trådlöst, vilket uppmuntrar till snabb installation av de senaste säkerhetsfixarna.

Startprocessen som beskrevs tidigare hjälper till att garantera att endast kod som signerats av Apple kan installeras på enheten. För att förhindra att enheter nedgraderas till äldre versioner som inte har de senaste säkerhetsuppdateringarna använder iOS en process som kallas *auktorisering av systemprogram*. Om det vore möjligt att nedgradera skulle obehöriga personer som får tillgång till enheter kunna installera en äldre version av iOS och dra nytta av de sårbarheter som har åtgärdats i den nyare versionen.

På enheter med Secure Enclave använder Secure Enclave-coprocessorn också auktorisering av systemprogram för att garantera integriteten hos den egna programvaran och förhindra nedgraderingar. Se avsnittet Secure Enclave i det här dokumentet.

iOS-programuppdateringar kan installeras via iTunes eller trådlöst direkt på enheten. Om du använder iTunes hämtas och installeras en fullständig kopia av iOS. Vid trådlös programuppdatering hämtas inte hela operativsystemet, utan endast de komponenter som krävs för att slutföra uppdateringen, vilket ger en effektivare nätverksanvändning. Dessutom kan programuppdateringar cachelagras på en Mac med macOS High Sierra som har Innehållscachelagring aktiverad, så att iOS-enheterna inte behöver ansluta till Apples servrar för att komma åt nödvändiga uppdateringsdata. De måste fortfarande kontakta Apples servrar för att slutföra uppdateringsprocessen.

Under en iOS-uppgradering ansluter iTunes (eller enheten, om du uppdaterar trådlöst) till Apples auktoriseringsserver för installation och skickar en lista till den över kryptografiska åtgärder för de olika delarna av installationspaketet (exempelvis iBoot, kärnan och OS-avbilden), ett slumpmässigt anti-replay-värde (nonce) samt enhetens unika ID som kallas **ECID (Exclusive Chip Identification)**.

Auktoriseringsservern jämför listan över åtgärder med de versioner för vilka installation är tillåten, och om de överensstämmer lägger den till enhetens ECID och signerar resultatet. Servern överför en komplett uppsättning signerade data till enheten under uppgraderingsprocessen. Att lägga till

enhetens ECID är ett sätt att göra auktoriseringen unik för enheten. Genom att endast auktorisera och signera kända åtgärder försäkrar sig servern om att uppdateringen sker exakt som Apple har avsett.

Under startprocessens utvärdering av tillförlitlighetskedjan verifieras att signaturen kommer från Apple och att åtgärden för objektet som lästs in från skivan, i kombination med enhetens ECID, matchar det som signaturen avser. De här åtgärderna bekräftar att auktoriseringen gäller en specifik enhet och ser till att en gammal iOS-version inte kan kopieras från en enhet till en annan. Nonce-värdet förhindrar att obehöriga sparar serverns svar och använder det till att manipulera en enhet eller modifiera systemprogramvaran.

Secure Enclave

Secure Enclave är en coprocessor som finns inuti SoC. Den använder sig av krypterat minne och innehåller en fysisk slumptalsgenerator. Secure Enclave tillhandahåller alla kryptografiska uppgifter för hantering av **dataskyddsnycklar** och upprätthåller ett intakt dataskydd även om det uppstår säkerhetsproblem i kärnan. Kommunikationen mellan Secure Enclave och approcessorn sker endast via en avbrottsdriven brevlåda och delade minnesbuffertar för data.

Secure Enclave innehåller ett dedikerat Boot ROM för Secure Enclave. I likhet med approcessor-Boot ROM består Secure Enclaves Boot ROM av statisk kod som utgör det betrodda och maskinvarubaserade säkerhetssystemet för Secure Enclave.

Secure Enclave kör ett Secure Enclave-OS som bygger på en Apple-anpassad version av L4-mikrokärnan. Detta Secure Enclave-OS är signerat av Apple, verifieras av Secure Enclaves Boot ROM och uppdateras via en anpassad programuppdateringsprocess.

När enheten startas skapas en tillfällig minnesskyddsnyckel av Secure Enclaves Boot ROM. Nyckeln är kopplad till enhetens UID och används till att kryptera Secure Enclave-delen av enhetens minnesutrymme. Med undantag för Apple A7 autentiseras även Secure Enclave-minnet med minnesskyddsnyckeln. I A11 och nyare samt S4 med en SoC-enhet används ett integritetsträd för att förhindra återuppspelning av säkerhetskritiskt Secure Enclave-minne, autentiserat av minnesskyddsnyckeln och nonce-värden som lagras i processorns inbyggda SRAM-minne.

Data som sparas i filsystemet av Secure Enclave krypteras med en nyckel som är kopplad till UID:t samt en anti-replay-räknare. Anti-replay-räknaren lagras i ett dedikerat icke-flyktigt minne i en **integrerad krets (IC)**.

På enheter som har en A12- eller S4-SoC parkopplas Secure Enclave med en integrerad krets (IC) för säker lagring av anti-replay-räknare. IC:n för säker lagring är utformad med statisk ROM-kod, en maskinvarubaserad slumptalsgenerator, kryptografimotorer och detektering av fysisk manipulering. Secure Enclave och lagrings-IC:n läser och uppdaterar räknare genom ett säkert protokoll som endast ger tillgång till räknarna.

Antiåteruppspelningstjänster i Secure Enclave används för återkallande av data via event som markerar antiåteruppspelningsgränserna som inkluderar, men inte är begränsade till:

- Lösenkodsändring
- Aktivering/avaktivering av Touch ID eller Face ID
- Tillägg/borttagning av fingeravtryck
- Nollställning av Face ID

- Tillägg/borttagning av Apple Pay-kort
- Radering av allt innehåll och inställningar

Secure Enclave analyserar även fingeravtrycks- och ansiktsdata från Touch ID- och Face ID-sensorerna, för att bestämma om det finns en matchning, och godkänner därefter tillgång eller inköp för användarens räkning.

OS-integritetsskydd

Kärnintegritetsskydd

När iOS-kärnan slutför initiering aktiveras ett kärnintegritetsskydd (Kernel Integrity Protection, KIP) som förhindrar ändringar av kärn- och drivrutinskod. **Minnesstyrkretsen** tillhandahåller en skyddad fysisk minnesregion som **iBoot** använder till att läsa in kärnan och kärntillägg. När starten är klar nekar minnesstyrkretsen skrivningar till den skyddade fysiska minnesregionen. Approcessorns minneshanteringsenhet (Memory Management Unit, MMU) är dessutom konfigurerad så att den förhindrar mappning av privilegierad kod från fysiskt minne utanför den skyddade minnesregionen och förhindrar skrivbara mappningar av fysiskt minne inuti kärnminnesregionen.

Maskinvaran som används till att aktivera KIP blir låst när startprocessen är klar så att den inte går att omkonfigurera. KIP stöds på SoC-enheter från och med Apples A10 och S4.

Integritetsskydd för systemcoprocessor

Systemcoprocessorer är processorer som finns på samma SoC som approcessorn. Systemcoprocessorer är avsedda för ett särskilt syfte och iOS-kärnan delegerar många uppgifter till dem. Exempel omfattar:

- Secure Enclave
- Bildsensorprocessor
- Rörelsecoprocessor

Eftersom fast coprocessorprogramvara hanterar många kritiska systemuppgifter är dess säkerhet en viktig del av den övergripande systemsäkerheten.

Integritetsskydd för systemcoprocessor (System Coprocessor Integrity Protection, SCIP) använder en mekanism som liknar kärnintegritetsskydd för att förhindra ändring av fast coprocessorprogramvara. Vid start läser iBoot in de enskilda coprocessorernas fasta programvara i en skyddad minnesregion som är reserverad och separat från KIP-regionen. iBoot konfigurerar varje coprocessors minneshanteringsenheter för att förhindra:

- Körbara mappningar utanför dess del av den skyddade minnesregionen
- Skrivbara mappningar inuti dess del av den skyddade minnesregionen

Secure Enclaves operativsystem ansvarar för att konfigurera Secure Enclaves SCIP vid start.

Maskinvaran som används till att aktivera SCIP blir låst när startprocessen är klar så att den inte går att omkonfigurera. SCIP stöds på SoC-enheter från och med Apples A12 och S4.

Pekarautentiseringskoder

Pekarautentiseringskoder (Pointer authentication codes, PACs) används som skydd mot att minnesfelsbuggar utnyttjas. Systemprogramvara och inbyggda appar använder PAC till att förhindra ändring av funktionspekare

och returadresser (kodpekare). Detta innebär att många attacker blir mycket svårare att genomföra. Exempelvis försöker en ROP-attack (Return Oriented Programming) lura enheten att köra befintlig kod i skadligt syfte genom att manipulera funktionsreturadresser som lagras i stacken.

PAC stöds på enheter med A12 och S4-SoC.

Touch ID

Touch ID är ett system för fingeravtrycksläsning som ger snabb och säker tillgång till iPhone och iPad. Tekniken läser av fingeravtryck ur alla vinklar och lär sig mer om användarens fingeravtryck över tid. Sensorn kartlägger kontinuerligt fingeravtrycket i och med att den upptäcker nya, överlappande noder vid varje användning.

Face ID

Genom en enkel titt låser Face ID upp din Apple-enhet som har den funktionen. Det är en intuitiv och säker autentiseringsmetod som är möjlig tack vare TrueDepth-kamerasystemet som drar nytta av avancerad teknik för att kartlägga de geometriska punkterna i ditt ansikte. Face ID använder neuronnät för att bedöma uppmärksamhet, matchning och skydda mot bedrägeriförsök så att du kan låsa upp telefonen med ett ögonkast. Face ID anpassas automatiskt efter förändringar i ditt utseende och skyddar dina biometriska data så att de förblir privata.

Touch ID, Face ID och lösenkoder

För att kunna använda Touch ID eller Face ID måste du ställa in din enhet så att det krävs en lösenkod för att låsa upp den. När Touch ID eller Face ID känner igen en matchning låser den upp enheten utan att fråga efter lösenkoden. Det här gör det mindre besvärligt att använda en lång och komplex lösenkod eftersom du inte behöver ange koden lika ofta. Touch ID och Face ID ersätter inte en lösenkod, utan ger smidig tillgång till enheten inom genomtänkta gränser och tidsbegränsningar. Det här är viktigt eftersom ett starkt lösenord utgör grunden för hur din iOS-enhet kryptografiskt skyddar dina data.

Du kan när som helst använda lösenkoden istället för Touch ID eller Face ID, men följande åtgärder kräver alltid en lösenkod istället för en biometrisk metod:

- Uppdatering av programvara.
- Radering av enheten.
- Visning eller ändring av lösenkodsinställningar.
- Installation av iOS-konfigurationsprofiler.

En lösenkod krävs även om enheten befinner sig i något av följande lägen:

- Om du precis har slagit på eller startat om enheten.
- Om enheten inte har låsts upp på över 48 timmar.
- Om lösenkoden inte har använts för att låsa upp enheten under de senaste 156 timmarna (sex och en halv dag) och du inte har låst upp enheten under de senaste fyra timmarna med en biometrisk metod.
- Om enheten har mottagit ett fjärrlåsningskommando.
- Efter fem misslyckade biometriska matchningsförsök.
- Efter påbörjad avstängning/Nödsamtal SOS.

När Touch ID eller Face ID är aktiverat låses enheten direkt när du trycker på sidoknappen och varje gång enheten försätts i viloläge. Touch ID och Face ID kräver en lyckad matchning, eller att du anger lösenkoden, vid varje väckning.

Sannolikheten för att en slumpmässigt utvald person kan låsa upp din iPhone är 1 på 50 000 med Touch ID och 1 på 1 000 000 med Face ID. Sannolikheten ökar om du har flera registrerade fingeravtryck (upp till 1 på 10 000 med fem fingeravtryck) eller utseenden (upp till 1 på 500 000 med två utseenden). Som ett extra skydd tillåter både Touch ID och Face ID bara fem misslyckade matchningsförsök innan en lösenkod krävs för att få tillgång till enheten. Sannolikheten för en felaktig matchning med Face ID är annorlunda för tvillingar och syskon som liknar dig och för barn under 13 år (eftersom deras distinkta ansiktsdrag kanske inte har utvecklats helt ännu). Om detta berör dig rekommenderar Apple att du använder en lösenkod för autentisering.

Touch ID-säkerhet

Fingeravtryckssläsaren aktiveras när den beröringskänsliga stålringen som sitter runt hemknappen upptäcker beröring av ett finger. Det aktiverar den avancerade matrisen att skanna av fingret och skicka bilden till Secure Enclave. Kommunikationen mellan processorn och Touch ID-sensorn sker via en SPI-buss (Serial Peripheral Interface). Processorn vidarebefordrar data till Secure Enclave, men kan inte läsa dem. De krypteras och autentiseras med en sessionsnyckel som förhandlas med hjälp av enhetens delade nyckel för varje Touch ID-sensor och dess motsvarande Secure Enclave. Den delade nyckeln är stark, slumpmässig och unik för varje Touch ID-sensor. Vid utbytet av sessionsnycklar används AES-**nyckelpaketering** där båda sidor tillhandahåller en slumpmässig nyckel som upprättar sessionsnyckeln och använder AES-CCM-kryptering vid överföring.

Den inskannade rasterbilden sparas tillfälligt i Secure Enclave medan den vektoriseras för analys och raderas sedan. Vid analysen används så kallad subdermal **ridge flow angle mapping**, vilket är en reducerande process där de detaljerade data som skulle krävas för att återskapa användarens fingeravtryck kastas. Resultatet är en karta över noder i fingeravtrycket, och den sparas utan någon identitetsinformation i ett krypterat format som bara kan läsas av Secure Enclave. Dessa data lämnar aldrig enheten. De skickas inte till Apple och ingår inte i säkerhetskopior av enheten.

Face ID-säkerhet

Face ID är utformat för att bekräfta användarens uppmärksamhet, tillhandahålla robust autentisering med få falska matchningar och begränsa digitala och fysiska bedrägeriförsök.

TrueDepth-kameran letar automatiskt efter ditt ansikte när du väcker en Apple-enhet som har Face ID genom att höja den eller trycka på skärmen, liksom när enheten försöker autentisera dig för att visa en inkommande notis eller när en app som stöds begär Face ID-autentisering. När ett ansikte upptäcks bekräftar Face ID uppmärksamheten och avsikten att låsa upp genom att upptäcka att dina ögon är öppna och att din uppmärksamhet är riktad mot enheten. Det här avaktiveras när VoiceOver aktiveras och kan vid behov även avaktiveras separat.

När TrueDepth-kameran har bekräftat att det finns ett uppmärksamt ansikte projicerar den och läser av fler än 30 000 infraröda prickar för att skapa en djupkarta av ansiktet tillsammans med en infraröd 2D-bild. Dessa data används till att skapa en sekvens med 2D-bilder och djupkartor

som signeras digitalt och skickas till Secure Enclave. För att avvärja både digitala och fysiska bedrägeriförsök slumpar TrueDepth-kameran ordningen på 2D-bilder och sparade djupkartor samt projicerar ett enhetsspecifikt slumpmässigt mönster. En del av det neurala systemet i A11 och nyare SoC-enheter – skyddat i Secure Enclave – omvandlar dessa data till en matematisk representation och jämför sedan denna representation med de ansiktsdata som har registrerats. Dessa registrerade ansiktsdata utgör är en matematisk representation av ditt ansikte som har registrerats ur olika vinklar.

Ansiktsmatchningen utförs i Secure Enclave med hjälp av neuronnät som har specialtränats för det ändamålet. Vi utvecklade neuronnäten för ansiktsmatchning med hjälp av över en miljard bilder, inklusive IR- och djupbilder som har samlats in i studier som genomförts med deltagarnas informerade samtycke. Apple arbetade med deltagare från hela världen för att få med en representativ samling personer med tanke på kön, ålder, etnicitet och andra faktorer. Studierna utökades vid behov för att få en hög tillförlitlighet gällande olika typer av användare. Face ID är utformat för att fungera med hattar, halsdukar, glasögon, kontaktlinser och många solglasögon. Dessutom fungerar det inomhus, utomhus och till och med i totalt mörker. Ytterligare ett neuronnät är tränat för att upptäcka och avvärja försök att låsa upp iPhone X med hjälp av bilder eller masker.

Face ID-data, inklusive matematiska representationer av ditt ansikte, krypteras och är bara tillgängliga för Secure Enclave. Dessa data lämnar aldrig enheten. De skickas inte till Apple och ingår inte i säkerhetskopior av enheten. Följande Face ID-data sparas och krypteras, endast för Secure Enclave, vid normal användning:

- De matematiska representationerna av ditt ansikte som beräknats vid registreringen.
- De matematiska representationerna av ditt ansikte som beräknats under vissa upplåsningsförsök om Face ID bedömer att de kan användas till att förbättra kommande matchningar.

Ansiktsbilder som har tagits vid normal användning sparas inte, utan kasseras omedelbart när den matematiska representationen har beräknats för antingen registrering eller jämförelse med registrerade Face ID-data.

Hur Touch ID eller Face ID låser upp en iOS-enhet

Om Touch ID eller Face ID är avaktiverat när du låser enheten raderas nycklarna som förvaras i Secure Enclave för den högsta dataskyddsklassen. Filerna och **nyckelringsobjekten** i den klassen blir inte tillgängliga förrän du låser upp enheten med din lösenkod.

När Touch ID eller Face ID är aktiverat kastas inte nycklarna när enheten låses. Istället paketeras de och förvaras tillsammans med en nyckel som ges till Touch ID- eller Face ID-undersystemet inuti Secure Enclave. När du försöker låsa upp enheten, och den upptäcker en matchning, tillhandahåller den nyckeln för att packa upp dataskyddsnycklarna och enheten låses upp. Den här processen ger extra skydd eftersom den kräver att undersystemen för dataskydd och Touch ID eller Face ID samarbetar för att låsa upp enheten.

När enheten startas om förloras nycklarna som krävs för att låsa upp enheten med Touch ID eller Face ID. Nycklarna kasseras av Secure Enclave efter att något som kräver att du anger lösenkoden har inträffat (exempelvis att enheten inte har låsts upp på 48 timmar eller efter fem misslyckade matchningsförsök).

Face ID förbättrar upplåsningsprestanda och håller jämna steg med de naturliga förändringarna i ditt ansikte och utseende genom att utöka de lagrade matematiska representationerna över tid. Efter upplåsning kan Face ID använda den nyligen beräknade matematiska representationen (om kvaliteten är tillräckligt hög) för ett begränsat antal ytterligare upplåsningar innan dessa data kasseras. Om Face ID däremot inte känner igen dig, men kvaliteten på matchingen överstiger ett visst tröskelvärde och du omedelbart följer upp genom att ange din lösenkod, gör Face ID ytterligare en avläsning och utökar dess registrerade Face ID-data med den nyligen beräknade matematiska representationen. Dessa nya Face ID-data kasseras om du slutar matcha mot dem och efter ett begränsat antal upplåsningar. Tack vare de här utökningsprocesserna kan Face ID hålla jämna steg med stora förändringar av skäggväxt eller sminkning, samtidigt som falska matchningar minimeras.

Touch ID, Face ID och Apple Pay

Du kan också använda Touch ID och Face ID med Apple Pay för att smidigt och säkert betala för inköp i butiker, appar och på webben. Mer information om Touch ID och Apple Pay finns i avsnittet om Apple Pay i det här dokumentet.

När du vill auktorisera en butiksbetalning med Face ID måste du först bekräfta din avsikt att betala genom att dubbelklicka på sidoknappen. Sedan auktoriserar du med Face ID innan du placerar iPhone X i närheten av den kontaktlösa betalningsläsaren. Om du vill välja en annan betalningsmetod för Apple Pay när du har autentiserat med Face ID måste du autentisera igen, men du behöver inte dubbelklicka på sidoknappen igen.

När du vill auktorisera en betalning i en app eller på webben måste du först bekräfta din avsikt att betala genom att dubbelklicka på sidoknappen. Sedan autentiserar du betalningen med Face ID. Om Apple Pay-transaktionen inte slutförs inom 30 sekunder efter att du har dubbelklickat på sidoknappen måste bekräfta din avsikt att betala genom att dubbelklicka igen.

Face ID-diagnos

Face ID-data lämnar aldrig enheten och säkerhetskopieras aldrig till iCloud eller någon annanstans. Det är bara om du vill tillhandahålla Face ID-diagnosdata till AppleCare för support som den här informationen överförs från din enhet. Om du vill aktivera Face ID-diagnos krävs en digitalt signerad auktorisering från Apple som liknar den som används vid personanpassning av programuppdateringar. Efter auktoriseringen kan du aktivera Face ID-diagnos och påbörja inställningsprocessen i appen Inställningar på enheter som har stöd för Face ID.

När du ställer in Face ID-diagnos raderas din befintliga Face ID-registrering och du blir ombedd att registrera ditt ansikte på nytt i Face ID. På enheter som har stöd för Face ID börjar Face ID-bilder som tas vid autentiseringsförsök att registreras under de kommande tio dagarna och de upphör därefter automatiskt att spara bilder. Face ID-diagnos skickar inte data till Apple automatiskt. Du kan granska och godkänna registrering och låsa upp bilder (både lyckade och misslyckade) som finns i de Face ID-diagnosdata som samlas in under diagnosläget innan de skickas till Apple. Face ID-diagnos överför endast de Face ID-diagnosbilder som du har godkänt. Data krypteras innan de överförs och raderas omedelbart från enheten när överföringen är klar. Bilder som du avvisar blir omedelbart raderade.

Om du inte slutför Face ID-diagnossessionen genom att granska bilder och överföra godkända bilder avslutas Face ID-diagnos automatiskt efter 40 dagar, och alla diagnosbilder raderas från enheten. Du kan också avaktivera Face ID-diagnos när som helst. Om du gör detta raderas alla lokala bilder direkt, och inga Face ID-data delas med Apple.

Andra användningsområden för Touch ID och Face ID

Appar från tredje part kan dra nytta av systemomfattande API:er för att be användare att autentisera med Touch ID eller Face ID eller en lösenkod. Appar som redan stöder Touch ID stöder automatiskt Face ID utan några ändringar. När du använder Touch ID eller Face ID får appen bara ett meddelande om huruvida autentiseringen lyckades eller inte. Den får inte tillgång till Touch ID, Face ID eller till de data som är kopplade till den registrerade användaren. Nyckelringsobjekt kan också skyddas med Touch ID eller Face ID, så att de endast släpps från Secure Enclave när användaren gör en matchning eller anger lösenkoden till enheten. Apputvecklare har API:er som bekräftar att användaren har angett en lösenkod innan de kräver Touch ID, Face ID eller en lösenkod för att låsa upp nyckelringsobjekt. Apputvecklare kan göra följande:

- Kräva att API-anrop om autentisering inte faller tillbaka till ett applösenord eller enhetens lösenkod. Skicka en förfrågan om ifall en användare är registrerad och då tillåta att Touch ID eller Face ID används som en andra faktor i appar som kräver hög säkerhet.
- Generera och använda ECC-nycklar i Secure Enclave som kan skyddas med Touch ID eller Face ID. Åtgärder med de här nycklarna utförs alltid i Secure Enclave efter att Secure Enclave har auktoriserat användningen.

Du kan också konfigurera Touch ID eller Face ID för användning vid inköp på iTunes Store, App Store och Apple Books så att du slipper ange ditt Apple-ID-lösenord. Med iOS 11 eller senare används Touch ID- och Face ID-skyddade ECC-nycklar i Secure Enclave till att auktorisera ett köp genom att signera butiksbegäran.

Kryptering och dataskydd

Radera allt innehåll och inställningar
Alternativet ”Radera allt innehåll och inst.” i Inställningar utplånar alla nycklar i det raderingsbara lagringsutrymmet, vilket gör alla användardata på enheten kryptografiskt oåtkomliga. Därför är det ett idealiskt sätt att försäkra sig om att all personlig information tas bort från en enhet innan den byter ägare eller lämnas in på service.

Viktigt: Använd inte alternativet ”Radera allt innehåll och inst.” förrän enheten är säkerhetskopierad, eftersom det inte finns något sätt att återskapa de data som raderas.

Den säkra startsekvensen, kodsigneringen och säkerheten vid körning bidrar till att endast betrodd kod och betrodda appar kan köras på enheten. iOS har ytterligare funktioner för kryptering och dataskydd som lagrar användarens data säkert, även i fall där andra delar av säkerhetsinfrastrukturen inte fungerar till fullo (till exempel på en enhet som har modifierats på ett otillåtet sätt). Det innebär viktiga fördelar för både användare och IT-administratörer, eftersom både personlig information och företagsdata alltid skyddas och det finns metoder för snabb och fullständig fjärradering av stulna eller borttappade enheter.

Säkerhetsfunktioner i maskinvaran

Snabbhet och energieffektivitet är avgörande när det gäller mobilenheter. Kryptografiska åtgärder är komplexa och kan orsaka problem med prestanda eller batteritid om de inte prioriteras vid utformningen av systemet.

Alla iOS-enheter har en dedikerad 256-bitars AES-krypteringsmotor inbyggd i DMA-sökvägen mellan flashminnet och systemets huvudminne, vilket gör filkrypteringen mycket effektiv. På A9- och senare A-serieprocessorer ligger undersystemet för flashminne på en isolerad buss som endast tillåts få tillgång till minne som innehåller användardata via DMA-krypteringsmotorn.

Enhetens **unika ID (UID)** och **enhetsgrupps-ID (GID)** är 256-bitars AES-nycklar som byggs in (UID) eller kompileras in (GID) i approcessorn och Secure Enclave vid tillverkningen. Ingen programvara eller fast programvara kan läsa dem direkt, utan det går bara att se resultatet av krypteringar och avkrypteringar som har utförts av dedikerade AES-motorer som är inbyggda i processorn, med UID eller GID som nyckel. Approcessorn och Secure Enclave har var sitt UID och GID. Secure Enclaves UID och GID kan endast användas av den AES-motor som är dedikerad för Secure Enclave. UID och GID är heller inte tillgängliga via **JTAG (Joint Test Action Group)** eller något annat felsökningsgränssnitt.

Med undantag för Apple A8 och tidigare SoC-enheter genererar varje Secure Enclave sitt eget UID (Unika ID) under tillverkningsprocessen. Eftersom UID:t är unikt för varje enhet, och eftersom det genereras helt och hållet inuti Secure Enclave istället för i ett tillverkningssystem utanför enheten, kan UID:t inte användas för anslutning eller lagring av Apple eller någon av dess underleverantörer.

Programvara som körs med Secure Enclave drar nytta av UID:t för att skydda enhetsspecifika hemligheter. Detta UID gör att data kan knytas till en specifik enhet kryptografiskt. Nyckelhierarkin som skyddar filsystemet innehåller till exempel detta UID, vilket innebär att det inte går att komma åt filerna om minneskretsarna flyttas fysiskt från en enhet till en annan. UID:t har inget samband med någon annan identitetsmärkning på enheten.

GID:t är gemensamt för alla processorer i en enhetsklass (exempelvis alla enheter som använder Apples A8-processor).

Alla kryptografiska nycklar, förutom UID och GID, skapas via systemets slumptalsgenerator med en algoritm baserad på CTR_DRBG-källkod. Systementropi genereras av tidsvariationer under startsekvensen och av avbrottsintervaller när enheten har startat. Nycklar som genereras inuti Secure Enclave använder en fysisk slumptalsgenerator (TRNG) baserad på flera ringoscillatorer som efterbehandlas med CTR_DRBG.

Det är lika viktigt att kunna radera sparade nycklar på ett säkert sätt som det är att generera dem. Det är särskilt svårt att göra det i exempelvis flashlagring där slitageutjämningsfunktionen kan innebära att flera datakopior måste raderas. I iOS-enheter finns därför en funktion för säker radering av data som kallas för det raderingsbara lagringsutrymmet. Den fungerar på så sätt att den raderar ett litet antal block på mycket låg nivå i den underliggande lagringstekniken (till exempel NAND).

Expresskort med strömsparläge

Om iOS inte är igång eftersom iPhone måste laddas kan det fortfarande finnas tillräckligt mycket ström i batteriet för att genomföra transaktioner med expresskort.

Vissa iPhone-modeller stöder automatiskt den här funktionen med:

- Ett kollektivtrafikkort som är valt som expresskollektivtrafikskort.
- Studentkort med expressläge aktiverat

När du trycker på sidoknappen viss symbolen för svagt batteri tillsammans med text som talar om att du har expresskort som kan användas. NFC-styrenheten genomför transaktioner med expresskort under likadana förhållanden som när iOS är igång, med undantag för att transaktionerna endast bekräftas med en haptisk notis. Du får inga synliga notiser.

Den här funktionen är inte tillgänglig när användaren väljer att stänga av enheten på vanligt sätt.

Dataskydd på filnivå

Utöver de inbyggda funktionerna för maskinvarukryptering i iOS-enheterna har Apple en speciell teknik för att skydda data som lagras i enhetens flashminne. Dataskyddet gör att enheten kan svara på vanliga händelser som inkommande telefonsamtal samtidigt som man upprätthåller en hög nivå när det gäller kryptering av användardata. Vanliga systemappar som Meddelanden, Mail, Kalender, Kontakter, Bilder och datavärden från Hälsa använder dataskydd som förval, och tredjepartsappar installerade i iOS 7 eller senare får det här skyddet automatiskt.

Dataskyddet implementeras genom att skapa och hantera en hierarki av nycklar. Det bygger på den teknik för maskinvarukryptering som finns inbyggd i alla iOS-enheter. Dataskyddet styrs på filnivå genom att tilldela varje fil en klass. Tillgängligheten avgörs sedan av om klassnycklarna har låsts upp eller ej. I och med introduktionen av APFS-formatet (Apple File System) kan filsystemet nu dela nycklarna i mindre delar på extentgrund (delar av en fil kan ha olika nycklar).

Översikt av arkitekturen

Varje gång en fil skapas på partitionen skapas en ny 256-bitarsnyckel (filnyckeln) i maskinvarans AES-motor som krypterar filen med nyckeln när filen sparas i flashminnet i AES-XTS-läget. På enheter med en A7, S2 eller

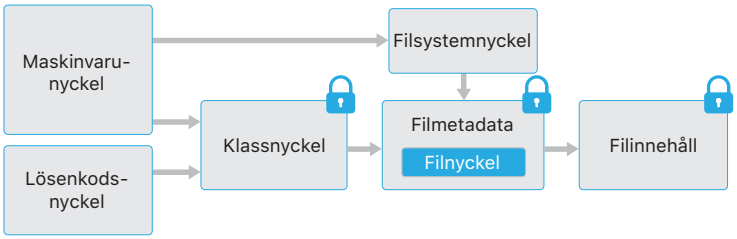
S3 med SoC används AES-CBC. Initieringsvektorn räknas ut med hjälp av blockets förskjutning in i filen och krypteras med hashfunktionen SHA-1 för **filnyckeln**.

Filnyckeln (eller filextentnyckeln) paketeras med någon av klassnycklarna, beroende på under vilka omständigheter filen ska vara tillgänglig. Som alla andra paketeringar utförs denna med hjälp av NIST AES-nyckelpaketering, enligt RFC 3394. Nyckelpaketet sparas i filens metadata.

Enheter som har APFS-format kan stöda filkloning (enkla kopior med copy-on-write-teknik). Om en fil klonas får varje halva av klonen en ny nyckel för att godkänna inkommande skrivförfrågningar så att nya data skrivs i mediet med en ny nyckel. Över tid kan filen komma att bestå av olika extent (fragment) som är kopplade till olika nycklar. Alla extent som utgör en fil vaktas dock av samma klassnyckel.

När en fil öppnas avkrypteras dess metadata med **filsystemnyckeln**, vilket öppnar nyckelpaketet och en notering om vilken klass som skyddar filen. Filnyckeln (eller filextentnyckeln) öppnas tillsammans med klassnyckeln och skickas sedan till maskinvarans AES-motor som avkrypterar filen när den läses från flashminnet. All hantering av paketerade filnycklar sker i Secure Enclave. Filnyckeln hanteras aldrig direkt av approcessorn. Vid start förhandlar Secure Enclave fram en tillfällig nyckel med AES-motorn. När Secure Enclave öppnar filens nycklar paketeras de om med den tillfälliga nyckeln och skickas tillbaka till approcessorn.

Metadata för alla filer i filsystemet krypteras med en slumpmässig nyckel som skapas när iOS installeras första gången eller när enheten raderas av en användare. På enheter som stöder APFS är filsystemets metadatanyckel paketerad av Secure Enclaves UID-nyckel för långsiktig lagring. I likhet med filnycklar eller filextentnycklar är metadatanyckeln aldrig exponerad mot approcessorn eftersom Secure Enclave tillhandahåller en tillfällig version varje gång enheten startas. Vid lagring är den krypterade systemfilsnyckeln dessutom paketerad med en tillfällig nyckel som lagras i det raderingsbara lagringsutrymmet. Den här nyckeln ger inget extra dataskydd. Den är istället utformad för att kunna raderas snabbt, antingen av användaren med hjälp av alternativet ”Radera allt innehåll och inst.”, eller av en användare eller administratör med hjälp av ett fjärraderingskommando från en MDM-lösning, Exchange ActiveSync eller iCloud. Om du tar bort nyckeln på det sättet blir alla filer kryptografiskt oåtkomliga.



Innehållet i en fil kan krypteras med en eller flera filnycklar (eller filextentnycklar) som sparas tillsammans med klassnyckeln i filens metadata, vilka i sin tur krypteras med filsystemnyckeln. Klassnyckeln skyddas av maskinvarans UID och, för vissa klasser, av användarens lösenkod. Den här hierarkin ger både flexibilitet och prestanda. Att till exempel ändra klass för en fil kräver bara att filnyckeln ompaketeras, och ett byte av lösenkod ompaketerar bara klassnyckeln.

Lösenkoder

Att tänka på när det gäller lösenkoder
Om du anger ett långt lösenord som bara innehåller siffror visas ett numeriskt tangentbord på upplåsningsskärmen istället för det fullständiga tangentbordet. En längre numerisk lösenkod kan vara enklare att ange än en kortare alfanumerisk och ändå ge samma säkerhet.

Genom att skapa en lösenkod för enheten aktiverar användaren dataskydd automatiskt. iOS stöder lösenkoder med sex eller fyra siffror och alfanumeriska lösenkoder med valfri längd. Förutom att låsa enheten tillhandahåller lösenkoden entropi för vissa krypteringsnycklar. Det innebär att en angripare som har enheten i sin ägo inte kan komma åt data i specifika skyddsklasser utan lösenkoden.

Lösenkoden är knuten till enhetens UID, så automatiserade intrångsförsök måste utföras på själva enheten. Ett högt antal beräkningsiterationer gör att varje försök tar lång tid. Iterationsantalet har kalibrerats så att ett försök tar ungefär 80 millisekunder. Det innebär att det skulle ta mer än fem och ett halvt år att testa alla kombinationer av en sex tecken lång alfanumerisk lösenkod.

Fördröjningar mellan lösenkodsförsök

Försök	Framtvingad fördröjning
1–4	ingen
5	1 minut
6	5 minuter
7–8	15 minuter
9	1 timme

Ju starkare lösenkod användaren har, desto starkare blir krypteringsnyckeln. Touch ID och Face ID kan användas för att förbättra säkerheten ytterligare genom att användaren kan ange en mycket starkare lösenkod än vad som annars vore praktiskt möjligt. Det ökar den effektiva mängden entropi som skyddar krypteringsnycklarna för dataskydd utan att inverka negativt på användarupplevelsen av att låsa upp iOS-enheten många gånger varje dag.

För att ytterligare avvärja automatiserade försök att knäcka lösenkoder ökar fördröjningen när en felaktig lösenkod anges på upplåsningsskärmen. Om Inställningar > Touch ID och Lösenkod > Radera data är aktiverat raderas enheten automatiskt efter tio på varandra följande felaktiga försök att ange lösenkoden. Flera på varandra följande försök att använda samma felaktiga lösenkod räknas inte i den gränsen. Den här inställningen kan användas som en administrativ policy via en MDM-lösning som stöder den här funktionen och Exchange ActiveSync, och det går även att ange ett lägre tröskelvärde.

På enheter med Secure Enclave framtvingas fördröjningarna av Secure Enclave-coprocessorn. Fördröjningen gäller även om enheten startas om under en fördröjning. Timern startas om för den aktuella perioden.

Som ett led i att förbättra säkerheten, utan att offra användarvänligheten, kräver iOS 11.4.1 eller senare Touch ID, Face ID eller en lösenkod för att aktivera USB-gränssnittet om USB inte har använts nyligen. Det här undanröjer risken för attacker från fysiskt anslutna enheter, som bedrägliga laddare, medan du fortfarande kan använda USB-tillbehör inom rimliga tidsgränser. Om det är mer än en timme sedan iOS-enheten låstes eller sedan en USB-anslutning kopplades från kommer enheten inte att tillåta några nya anslutningar förrän enheten blir upplåst. Den timslånga perioden:

- Ser till att användare som ofta ansluter enheten till en Mac eller PC, USB-tillbehör eller med en kabel till CarPlay inte behöver ange sin lösenkod varje gång de ansluter enheten.
- Är nödvändig eftersom ekosystemet för USB-tillbehör saknar ett tillförlitligt sätt att identifiera tillbehör innan en dataanslutning upprättas.

Dessutom kommer enheter med iOS 12 att neka nya USB-anslutningar omedelbart efter låsning om det är mer än tre dagar sedan en USB-anslutning upprättades. Det här ökar skyddet för användare som sällan ansluter något via USB. USB-anslutningar avaktiveras även så fort enheten befinner sig i ett läge där den kräver en lösenkod för att återaktivera biometrisk autentisering.

Användaren kan välja att återaktivera alltid på-USB i Inställningar, och om användaren ställer in hjälpmedelsenheter görs det automatiskt.

DFU- och återhämtningsläge

På enheter med Apple A10, A11 och S3-SoC-enheter går det inte att komma åt klassnycklar som skyddas av användarens lösenkod i återhämtningsläge. A12 och S4-SoC-enheter utökar det här skyddet till DFU-läge.

AES-motorn i Secure Enclave är utrustad med låsbara programvarustartbitar. När nycklar skapas från UID:t ingår de här startbitarna i härledningsfunktionen för nycklar som används till att skapa ytterligare nyckelhierarkier.

Med början i Apple A10 och S3-SoC-enheter är en startbit dedikerad till att särskilja nycklar som skyddas av användarens lösenkod. Startbiten används för nycklar som kräver användarens lösenkod (inklusive nycklar med dataskyddsklass A, klass B och klass C) och rensas för nycklar som inte kräver användarens lösenkod (inklusive filsystemets metadatanyckel och klass D-nycklar).

I A12:s SoC-enheter låser Secure Enclaves Boot ROM lösenkodens startbit om approcessorn försätts i DFU-läge eller återhämtningsläge. När lösenkodens startbit är låst tillåts inga åtgärder som ändrar den, vilket förhindrar tillgång till data som skyddas med användarens lösenkod.

I Apple A10, A11, S3 och S4-SoC-enheter blir lösenkodens startbit låst av Secure Enclaves OS om enheten försätts i återhämtningsläge. Både Secure Enclaves Boot ROM och OS kontrollerar BPR (Boot Progress Register) för att säkert avgöra det aktuella läget.

Dataskyddsklasser

När en ny fil skapas på en iOS-enhet får den en klass av den app som skapar den. Varje klass har olika regler för att avgöra när informationen i filen är tillgänglig. De grundläggande klasserna och reglerna beskrivs nedan.

Complete Protection

(`NSFileProtectionComplete`): Klassnyckeln skyddas av en nyckel som härleds ur användarens lösenkod och enhetens UID. Strax efter att användaren har låst enheten (tio sekunder om inställningen Kräv lösenkod är Direkt) kastas den avkrypterade klassnyckeln. Det innebär att inga data i klassen går att komma åt förrän användaren anger lösenkoden igen eller låser upp enheten med Touch ID eller Face ID.

Protected Unless Open

(`NSFileProtectionCompleteUnlessOpen`): En del filer kanske måste skrivas medan enheten är låst. Ett bra exempel är en e-postbilaga som hämtas i bakgrunden. Detta beteende uppnås genom att använda asymmetrisk kryptering med elliptiska kurvor (ECDH över Curve25519). Den vanliga filnyckeln skyddas av en nyckel som härleds genom One-Pass Diffie-Hellman Key Agreement enligt beskrivningen i NIST SP 800-56A.

Den tillfälliga publika nyckeln för överenskommelsen lagras tillsammans med den paketerade filnyckeln. KDF är Concatenation Key Derivation Function (Approved Alternative 1) enligt beskrivningen i 5.8.1 i NIST SP 800-56A. AlgorithmID utelämnas. PartyUInfo och PartyVInfo är den tillfälliga respektive den statiska publika nyckeln. SHA-256 används som hashfunktion. Så fort filen stängs raderas filnyckeln från minnet. När filen ska öppnas igen återskapas den delade hemligheten med hjälp av

den privata nyckeln till klassen Protected Unless Open och filens tillfälliga publika nyckel. Dessa används till att packa upp filnyckeln, som sedan används till att avkryptera filen.

Protected Until First User Authentication

(`NSFileProtectionCompleteUntilFirstUserAuthentication`): Den här klassen uppträder på samma sätt som Complete Protection, förutom att den avkrypterade klassnyckeln inte raderas från minnet när enheten låses. Skyddet i den här klassen har egenskaper som liknar kryptering av hela enheter på en dator, och skyddar mot attacker som innefattar en omstart. Det här är förvald klass för alla data tillhörande tredjepartsappar som inte har tilldelats någon annan dataskyddsklass.

No Protection

(`NSFileProtectionNone`): Den här klassnyckeln skyddas endast med UID:t och förvaras i det raderingsbara lagringsutrymmet. Eftersom alla nycklar som behövs för att avkryptera filer i den här klassen förvaras på enheten ger kryptering bara den fördelen att fjärradering går snabbt. Om en fil inte har tilldelats någon dataskyddsklass sparas den ändå i krypterad form (precis som alla andra data på en iOS-enhet).

Dataskyddsklassnyckel

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

De olika delarna i ett nyckelringsobjekt

Utöver åtkomstgruppen innehåller varje nyckelringsobjekt administrativa metadata (som tidsstämplar för ”skapad” och ”senast uppdaterad”).

Det innehåller också SHA-1-hashvärden för de attribut som används vid sökning efter objektet (som konto- och servernamn), vilket gör att det går att söka efter objekt utan att avkryptera dem. Slutligen innehåller det krypteringsdata, däribland följande:

- Versionsnummer
- Data om behörighetslistor (ACL)
- Ett värde som talar om vilken skyddsklass objektet tillhör
- Objektnyckel som paketeras med nyckeln för dataskyddsklassen
- Ordlista över attribut som beskriver objektet (som det överförs till SecItemAdd), kodad som en binär plist och krypterad med objektnyckeln

Krypteringen är AES-256 i GCM (Galois/Counter Mode). Åtkomstgruppen inkluderas i attributen och skyddas av den GMAC-tagg som beräknas vid krypteringen.

Dataskydd genom nyckelring

Många appar måste hantera lösenord och andra små men viktiga data, till exempel nycklar och inloggningstokens. Nyckelringen i iOS gör att du kan lagra dessa objekt på ett säkert sätt.

Nyckelringsobjekt krypteras med två olika AES-256-GCM-nycklar, en tabellnyckel (metadata) och per-row-nyckel (secret-key). Nyckelringsmetadata (alla attribut utöver kSecValue) krypteras med metadatanyckeln för att snabba upp sökning, medan det hemliga värdet (kSecValueData) krypteras med secret-key. Metadatanyckeln skyddas av Secure Enclave-processorn, men cachelagras i approcessorn så att det snabbt går att skicka förfrågningar till nyckelringen. Den hemliga nyckeln kräver alltid att förfrågningar görs via Secure Enclave-processorn.

Nyckelringen implementeras som en SQLite-databas som lagras i filsystemet. Det finns bara en databas. Bakgrundsprogrammet securityd avgör vilka nyckelringsobjekt som varje process eller appar kan komma åt. Tillgångs-API:er för nyckelringen genererar anrop till bakgrundsprogrammet som sedan söker efter appens rättigheter för ”Keychain-access-groups”, ”application-identifier” och ”application-group”. Med hjälp av tillgångsgrupper kan nyckelringsobjekt delas mellan appar istället för att tillgången begränsas till en enda process.

Nyckelringsobjekt kan endast delas mellan appar från samma utvecklare. För appar från andra tillverkare måste tillgångsgrupper användas med ett prefix som tilldelas via Apple Developer Program genom appgrupper. Kravet på prefix och på att appgruppen ska vara unik upprätthålls genom kodsignering, **tillhandahållandeprofiler** och Apple Developer Program.

Nyckelringsdata skyddas med hjälp av en klasstruktur som liknar den som används för att skydda data på filnivå. Klasserna fungerar på samma sätt som dataskyddsklasserna för filer, men har andra nycklar och är en del av API:er med andra namn.

Tillgänglighet	Dataskydd på filnivå	Dataskydd genom nyckelring
När den är upplåst	NSFileProtectionComplete	kSecAttrAccessibleWhenUnlocked
När den är låst	NSFileProtectionCompleteUnlessOpen	Ej tillämpligt
Efter första upplåsningen	NSFileProtectionCompleteUntilFirstUserAuthentication	kSecAttrAccessibleAfterFirstUnlock
Alltid	NSFileProtectionNone	kSecAttrAccessibleAlways
Lösenkodsaktiverad	Ej tillämpligt	kSecAttrAccessible WhenPasscodeSetThisDeviceOnly

Appar som kör uppdateringstjänster i bakgrunden kan använda `kSecAttrAccessibleAfterFirstUnlock` för nyckelringsobjekt som behöver vara tillgängliga under bakgrundsuppdateringarna.

Klassen `kSecAttrAccessibleWhenPasscodeSetThisDeviceOnly` beter sig likadant som `kSecAttrAccessibleWhenUnlocked`, men den är bara tillgänglig när enheten har konfigurerats med en lösenkod. Den här klassen finns bara i systemets **nyckelsamling** och den:

- Synkroniseras inte med iCloud-nyckelring
- Säkerhetskopieras inte
- Ingår inte i deponerade nyckelsamlingar

Om lösenkoden tas bort eller nollställs görs objekten oanvändbara genom att klassnycklarna kastas.

Andra nyckelringsklasser har en motsvarighet till funktionen ”endast denna enhet”, som alltid skyddas med UID:t när den kopieras från enheten vid säkerhetskopiering, vilket gör att den blir oanvändbar om den återställs till en annan enhet. För att hitta en bra balans mellan säkerhet och användbarhet väljer Apple nyckelringsklasser baserat på vilken typ av information som ska skyddas och när iOS behöver tillgång till den. Exempelvis måste ett VPN-certifikat alltid vara tillgängligt så att anslutningen inte bryts, men det klassas som ”ej flyttbart” så att det inte kan flyttas till en annan enhet.

Nyckelringsobjekt som skapas av iOS skyddas enligt följande klasser:

Objekt	Tillgängligt
Wi-Fi-lösenord	Efter första upplåsningen
E-postkonton	Efter första upplåsningen
Exchange-konton	Efter första upplåsningen
VPN-lösenord	Efter första upplåsningen
LDAP, CalDAV, CardDAV	Efter första upplåsningen
Tokens för sociala nätverkskonton	Efter första upplåsningen
Krypteringsnycklar för Handoff-annonsering	Efter första upplåsningen
iCloud-token	Efter första upplåsningen
Lösenord för Hemmadelning	När den är upplåst
Token för Hitta min iPhone	Alltid
Röstbrevlåda	Alltid

Säkerhetskopiering med iTunes	När den är upplåst, ej flyttbart
Safari-lösenord	När den är upplåst
Safari-bokmärken	När den är upplåst
VPN-certifikat	Alltid, ej flyttbart
Bluetooth-nycklar	Alltid, ej flyttbart
Token för APNs (Apples pushnotistjänst)	Alltid, ej flyttbart
iCloud-certifikat och privata nycklar	Alltid, ej flyttbart
iMessage-nycklar	Alltid, ej flyttbart
Certifikat och privata nycklar som installerats av en konfigurationsprofil	Alltid, ej flyttbart
SIM-kortets PIN-kod	Alltid, ej flyttbart

Behörighetsstyrning av nyckelringen

Nyckelringar kan använda behörighetslistor (ACL:er) för att ange policyer för tillgång samt autentiseringskrav. Objekten kan ställa upp villkor som kräver användarens närvaro genom att begära autentisering med Touch ID, Face ID eller att enhetens lösenkod måste anges. Du kan begränsa tillgången till objekt genom att ange att registreringen av Touch ID eller Face ID inte har ändrats sedan objektet lades till. Denna begränsning förhindrar att obehöriga personer lägger till egna fingeravtryck för att kunna få tillgång till ett nyckelringsobjekt. ACL:erna granskas inuti Secure Enclave och lämnas endast ut till kärnan om de angivna villkoren uppfylls.

Nyckelsamlingar

Nycklarna till både filens och nyckelringens dataskyddsklasser samlas in och hanteras i nyckelsamlingar. iOS använder följande nyckelsamlingar: användare, enhet, säkerhetskopiering, deponering och iCloud-säkerhetskopiering.

Användarnyckelsamlingen är den nyckelsamling där de paketerade klassnycklarna som används vid normal körning av enheten förvaras. Exempel: när användaren anger en lösenkod läses nyckeln `NSFileProtectionComplete` in från användarnyckelsamlingen och packas upp. Det är en binär egenskapslistefil (.plist) som lagras i klassen No Protection och vars innehåll krypteras med en nyckel som förvaras i det raderingsbara lagringsutrymmet. För att skydda nyckelsamlingarna raderas och omskapas den här nyckeln varje gång en användare ändrar sin lösenkod. Kärntillägget `AppleKeyStore` hanterar användarnyckelsamlingen och kan svara på sökningar angående enhetens låsningsstatus. Det rapporterar att enheten är upplåst endast om alla klassnycklarna i användarnyckelsamlingen är tillgängliga och har packats upp.

Enhetsnyckelsamlingen används till att förvara de paketerade klass-nycklarna som används vid åtgärder som innehåller enhetsspecifika data. iOS-enheter som har konfigurerats för delad användning behöver ibland få tillgång till inloggningsuppgifter innan en användare har loggat in. Därför krävs det en nyckelsamling som inte är skyddad med användarens lösenkod. iOS har inte stöd för kryptografisk delning av filsystemsinnehåll per användare, vilket betyder att systemet använder klassnycklar från enhetsnyckelsamlingen för att förvara nycklar per fil. Nyckelringen använder däremot klassnycklar från användarnyckelsamlingen till att skydda objekt i användarens nyckelring. På iOS-enheter

som har konfigurerats för att användas av endast en användare (standardkonfigurationen) är enhetsnyckelsamlingen densamma som användarnyckelsamlingen, och båda är skyddade av användarens lösenkod.

Nyckelsamlingen för säkerhetskopiering skapas när en krypterad säkerhetskopia görs i iTunes och sparas på den dator som enheten säkerhetskopieras till. I samband med detta skapas en ny nyckelsamling med en ny uppsättning nycklar, och säkerhetskopierade data krypteras om med de nya nycklarna. Som beskrevs tidigare förblir icke-flyttbara nyckelringsobjekt paketerade tillsammans med den UID-härledda nyckeln. Det innebär att de kan återskapas till enheten de ursprungligen säkerhetskopierades från, men inte är tillgängliga på andra enheter.

Nyckelsamlingen skyddas med det lösenord som har ställts in i iTunes, kört genom 10 miljoner iterationer av PBKDF2. Trots det höga iterationsantalet finns det ingen koppling till en specifik enhet, och därför skulle ett automatiserat intrångsförsök som körs parallellt på många datorer teoretiskt sett kunna utföras på nyckelsamlingen för säkerhetskopiering. Den här typen av hot kan undvikas med ett tillräckligt starkt lösenord.

Om användaren väljer att inte kryptera en iTunes-säkerhetskopia krypteras inga av filerna, oavsett deras dataskyddsklass, men nyckelringen skyddas fortfarande med en UID-härledd nyckel. Det är därför nyckelringsobjekt bara flyttas till en ny enhet om användaren har ställt in ett lösenord för säkerhetskopiering.

En deponerad nyckelsamling används för iTunes-synkronisering och MDM (mobile device management). Med den här nyckelsamlingen kan iTunes säkerhetskopiera och synkronisera utan att användaren behöver ange någon lösenkod. Den tillåter också att en MDM-lösning kan fjärradera en användares lösenkod. Samlingen förvaras på datorn som används till att synkronisera med iTunes eller på MDM-lösningen som fjärradministrerar enheten.

En deponerad nyckelsamling ger en bättre användarupplevelse vid synkronisering av enheter, då det kan krävas tillgång till alla dataklasser. När en enhet med lösenkodslås ansluts till iTunes för första gången blir användaren ombedd att ange lösenkoden. Enheten skapar sedan en deponerad nyckelsamling som innehåller samma klassnycklar som de som används på enheten, skyddade av en ny nyckel. Den deponerade nyckelsamlingen och nyckeln som skyddar den delas upp mellan enheten och värden eller servern. Data lagras på enheten i klassen Protected Until First User Authentication. Det är därför användaren måste ange lösenkoden till enheten innan den säkerhetskopieras med iTunes första gången efter en omstart.

Om det rör sig om en trådlös programuppdatering uppmanas användaren att ange sin lösenkod när uppdateringen startas. Denna används till att skapa en engångsupplåsningstoken som låser upp användarnyckelsamlingen efter uppdateringen. Denna token går inte att generera utan att användarens lösenkod anges, och en eventuellt tidigare genererad token blir ogiltig om användarens lösenkod ändras.

En engångsupplåsningstoken är avsedd för antingen övervakad eller oövervakad installation av en programuppdatering. Den krypteras med en nyckel som härleds från det aktuella värdet av en monoton räknare i Secure Enclave, nyckelsamlingens UUID och Secure Enclaves UID.

När engångsupplåsningstokens räknare räknas upp i Secure Enclave blir en eventuell befintlig token ogiltig. Räknaren räknas upp när en token utfärdas, efter den första upplåsningen av en omstartad enhet, när en programuppdatering avbryts (av användaren eller av systemet) eller när policytimern för en token har löpt ut.

En engångsupplåsningstoken för övervakade programuppdateringar upphör att gälla efter 20 minuter. Denna token exporteras från Secure Enclave och sparas i det raderingsbara lagringsutrymmet. En policytimer räknar upp räknaren om enheten inte har startats om inom 20 minuter.

Oövervakade programuppdateringar sker när systemet upptäcker att det finns en uppdatering och:

- Automatiska uppdateringar är konfigurerade i iOS 12.

  eller

- Användaren väljer Installera senare när det visas ett meddelande om uppdateringen.

När användaren anger sin lösenkod skapas en engångsupplåsningstoken som är giltig i upp till åtta timmar i Secure Enclave. Så länge uppdateringen inte har installerats förstörs denna engångsupplåsningstoken vid varje låsning och återskapas vid varje efterföljande upplåsning. Varje upplåsning startar om nedräkningen på åtta timmar.

Efter åtta timmar kommer en policytimer att göra denna engångsupplåsnings-token ogiltig.

Nyckelsamlingen för iCloud-säkerhetskopiering liknar nyckelsamlingen för säkerhetskopiering. Alla klassnycklar i denna nyckelsamling är asymmetriska (de använder Curve25519, precis som dataskyddsklassen Protected Unless Open), så iCloud-säkerhetskopieringar kan utföras i bakgrunden. För alla dataskyddsklasser utom No Protection läses krypterade data från enheten och skickas till iCloud. De motsvarande klassnycklarna skyddas av iCloud-nycklar. Nyckelringens klassnycklar paketeras med en UID-härledd nyckel på samma sätt som vid en okrypterad säkerhetskopiering i iTunes. En asymmetrisk nyckelsamling används också till säkerhetskopiering för återskapande av nyckelringen i iCloud-nyckelring.

Appsäkerhet

Appar tillhör de mest sårbara delarna av en modern säkerhetsarkitektur för mobilenheter. Appar kan innebära enorma produktivitetsfördelar för användarna, men de kan också ha negativa effekter på systemets säkerhet, stabilitet och användardata om de inte hanteras korrekt.

På grund av detta har iOS försetts med flera lager av skydd som kontrollerar att alla appar är signerade, verifierade och körs i en sandlåda för att skydda användarens data. De här funktionerna gör iOS till en stabil och säker plattform för appar, med tusentals utvecklare som levererar hundratusentals appar för iOS utan att det påverkar systemets integritet. Och användarna kan öppna apparna på sina iOS-enheter utan att oroa sig för virus, skadeprogram eller attacker från obehöriga.

Kodsignering av appar

När iOS-kärnan har startat styr den vilka användarprocesser och appar som kan köras. För att garantera att alla appar kommer från en känd och godkänd källa, och inte har manipulerats, kräver iOS att all körbar kod ska vara signerad med ett certifikat utfärdat av Apple. De appar som följer med enheten, som Mail och Safari, är signerade av Apple. Tredjepartsappar måste också valideras och signeras med ett certifikat utfärdat av Apple Med obligatorisk kodsignering utökas tillförlitlighetskedjan från operativsystemet till apparna. Det förhindrar att tredjepartsappar läser in osignerade kodresurser eller använder självmodifierande kod.

För att utveckla och installera appar på iOS-enheter måste utvecklare registrera sig hos Apple och gå med i Apple Developer Program. Varje utvecklares verkliga identitet – oavsett om det är en individ eller ett företag – kontrolleras av Apple innan ett certifikat utfärdas. Med det här certifikatet kan utvecklarna signera appar och skicka in dem till App Store för distribution. Det innebär att alla appar i App Store har skickats in av en identifierbar person eller organisation, vilket avskräcker från att skicka in skadliga appar. Apparna har också granskats av Apple, som kontrollerar att de fungerar enligt beskrivningen och inte innehåller några uppenbara buggar eller andra problem. Utöver de tekniska aspekterna gör den här urvalsprocessen att kunderna får förtroende för att de appar de köper håller hög kvalitet.

Med iOS kan utvecklare bädda in ramverk inuti apparna, som kan användas av appen själv eller av tillägg som är inbäddade i appen. För att skydda systemappar och andra appar mot inläsning av kod från tredje part inom deras adressutrymme kontrollerar systemet kodsignaturerna för alla dynamiska bibliotek som någon process länkar till under start. Den här kontrollen genomförs med hjälp av teamidentifieraren (Team ID) som utvinns ur ett certifikat utfärdat av Apple. En teamidentifierare är en alfanumerisk sträng på tio tecken, till exempel 1A2B3C4D5F. Ett program kan länka till vilket plattformsbibliotek som helst som följer med systemet, eller vilket annat bibliotek som helst som har samma teamidentifierare som primär körbar kod i sin kodsignatur. Eftersom den körbara kod som ingår i systemet inte har någon teamidentifierare kan den endast länka till de bibliotek som är inbyggda i systemet.

Företag kan också utveckla interna appar och distribuera dem till sina anställda. Företag och organisationer kan ansöka till Apple Developer Enterprise Program (ADEP) med ett D-U-N-S-nummer. Apple godkänner ansökningarna efter att ha kontrollerat den sökandes identitet och behörighet. När en organisation blir medlem i ADEP kan den registrera sig och erhålla en tillhandahållandeprofil som gör det möjligt att köra interna appar på enheter som organisationen godkänner. Användarna måste ha tillhandahållandeprofilen installerad för att kunna köra interna appar. Detta garanterar att endast behöriga användare inom organisationen kan läsa in apparna på sina iOS-enheter. Appar som installeras via MDM är implicit betrodda eftersom relationen mellan organisationen och enheten redan är upprättad. I annat fall måste användarna godkänna appens tillhandahållandeprofil i Inställningar. Organisationerna kan förhindra att användare godkänner appar från okända utvecklare. Första gången en företagsapp startas måste enheten få en positiv bekräftelse från Apple om att appen tillåts att köras.

Till skillnad från andra mobilplattformar tillåter inte iOS att användarna installerar potentiellt skadliga, osignerade appar från webbplatser eller kör kod som inte är betrodd. Vid körning kontrolleras kodsignaturen för alla körbara minnessidor när de läses in – detta för att bekräfta att appen inte har modifierats sedan installationen eller den senaste uppdateringen.

Säkerhet vid körning

När en app har kontrollerats och bekräftats komma från en godkänd källa använder iOS tvingande säkerhetsåtgärder som förhindrar att den påverkar andra appar eller resten av systemet.

Alla tredjepartsappar körs i en ”sandlåda”, vilket innebär att de inte har tillgång till filer som sparats av andra appar och inte kan göra några ändringar på enheten. Detta förhindrar att appar samlar in eller ändrar information som sparats av andra appar. Varje apps filer sparas i en unik hemkatalog som tilldelas slumpmässigt när appen installeras. Om en tredjepartsapp behöver tillgång till information utifrån kan den endast få det via tjänster som uttryckligen tillhandahålls av iOS.

Systemfiler och systemresurser är också skyddade från användarens appar. Större delen av iOS, liksom alla tredjepartsappar, körs som användaren ”mobile” som saknar privilegier. Hela OS-partitionen är skrivskyddad. Verktyg som inte behövs, till exempel tjänster för fjärrinloggning, inkluderas inte i systemprogramvaran, och API:er tillåter inte att appar utökar sina egna behörigheter så att de kan modifiera andra appar eller själva iOS.

Tredjepartsappars tillgång till användarinformation och funktioner som iCloud och utökning genom tillägg styrs genom fastställda rättigheter. Rättigheter är par av nyckelvärden som skrivs in i en app och tillåter autentisering utöver faktorer vid körning, som UNIX-användar-ID. Eftersom rättigheter signeras digitalt kan de inte ändras. Rättigheter används ofta av systemappar och bakgrundsprocesser till att utföra specifika uppgifter som kräver behörighet och som annars skulle kräva att processen kördes som rotanvändare. Detta minskar risken för att eventuella systemappar eller bakgrundsprocesser som manipulerats ska utöka sina behörigheter.

Dessutom kan appar bara utföra bakgrundsbearbetningar genom system-API:er. Det gör att appar kan fortsätta köras utan att prestanda eller batteritid påverkas nämnvärt.

ASLR (Address Space Layout Randomization) skyddar mot att minnesfelsbuggar utnyttjas. Inbyggda appar använder ASLR för att se till att alla minnesregioner tilldelas en slumpmässig plats vid start. Genom att minnesadresserna för körbar kod, systembibliotek och sammanhörande programmeringsstrukturer ordnas slumpmässigt minskar sannolikheten för många avancerade angrepp. En return-to-libc-attack försöker till exempel lura enheter att köra skadlig kod genom att manipulera stackens och systembibliotekens minnesadresser. Om placeringen av minnesadresserna är slumpmässig är den här typen av angrepp mycket svårare att genomföra, särskilt mot flera enheter samtidigt. Xcode, utvecklingsmiljön för iOS, aktiverar ASLR automatiskt vid kompilering av tredjepartsprogram.

Ytterligare skydd tillhandahålls av iOS genom ARM:s Execute Never-funktion (XN), som markerar minnessidor som icke-körbara. Minnessidor som både markerats som skrivbara och körbara kan endast användas av appar under strängt kontrollerade former: Kärnan söker efter Apples särskilda rättighet till dynamisk kodsignering. Även om den hittas kan endast ett mmap-anrop göras för att begära en körbar och skrivbar sida, vilken tilldelas en slumpgenererad adress. Safari använder den här funktionen i sin JIT-kompilator för JavaScript.

Tillägg

Appar kan utöka funktionerna hos andra appar i iOS genom *tillägg*. Tillägg är signerade körbara binärfiler för speciella ändamål som paketeras inuti en app. Systemet upptäcker tillägg automatiskt under installationen och gör dem tillgängliga för andra appar genom ett matchningssystem.

Ett systemområde som har stöd för tillägg kallas en *tilläggspunkt*. Varje tilläggspunkt tillhandahåller API:er och ser till att policyer följs för det området. Systemet avgör vilka tillägg som är tillgängliga utifrån specifika matchningsregler för respektive tilläggspunkt. Systemet startar tilläggsprocesser automatiskt vid behov och ser till att de är aktiva under den tid som behövs. Rättigheter kan användas till att begränsa tillgängligheten för tillägg till specifika systemappar. Exempelvis visas en widget för vyn Idag bara i Notiscenter, medan ett tillägg för delning bara visas på panelen Delning. Tilläggspunkterna är Idag-widgetar, delning, anpassade åtgärder, bildredigering, dokumentappar och anpassat tangentbord.

Tilläggen körs i sitt eget adressutrymme. Vid kommunikation mellan tillägget och appen det aktiverats från används interprocesskommunikation som förmedlas via systemramverket. De har inte tillgång till varandras filer eller minnesutrymmen. Tillägg har utformats för att vara isolerade från varandra, från apparna som innehåller dem och från apparna som använder dem. De körs i en sandlåda, precis som andra tredjepartsappar, och har en behållare som är åtskild från behållaren för den app de finns i. De har dock samma tillgång till integritetsinställningar som appen som innehåller dem. Så om en användare ger Kontakter tillgång till en app omfattar tillgången också de tillägg som är inbäddade i appen, men inte de tillägg som aktiveras av appen.

Anpassade tangentbord är en speciell typ av tillägg eftersom användaren aktiverar dem för hela systemet. När de har aktiverats används ett tangentbordstillägg för alla textfält, med undantag för lösenkodsinmatningen och eventuella säkra textfält. För att begränsa överföringen av användardata körs anpassade tangentbord normalt i en

mycket restriktiv sandlåda som blockerar tillgång till nätverket, till tjänster som utför nätverksåtgärder för processers räkning och till API:er som skulle kunna ge tillägget obehörig tillgång till data som skrivs på tangentbordet. Utvecklare av anpassade tangentbord kan begära att deras tillägg får så kallad öppen tillgång (Open Access), vilket innebär att systemet får köra tillägget i den vanliga sandlådan efter användarens medgivande.

För enheter som är registrerade i en MDM-lösning styrs dokument- och tangentbordstillägg av reglerna för Managed Open In. Till exempel kan MDM-lösningen förhindra att användaren exporterar ett dokument från en hanterad app till en ohanterad dokumentapp eller använder ett ohanterat tangentbord med en hanterad app. Dessutom kan apputvecklare förhindra att tangentbordstillägg från tredje part används i deras app.

Appgrupper

Appar och tillägg som ägs av ett och samma utvecklarkonto kan dela innehåll om de konfigureras så att de ingår i samma appgrupp. Det är upp till utvecklaren att skapa rätt grupper på Apples utvecklarportal och inkludera den önskade uppsättningen appar och tillägg. En app som har konfigurerats för att ingå i en viss appgrupp har tillgång till följande:

- En delad lagringsbehållare på volymen. Behållaren sparas på enheten så länge som minst en app från gruppen är installerad.
- Delade inställningar.
- Delade nyckelringsobjekt.

Apples utvecklarportal garanterar att ID:n för appgrupper är unika i hela appekosystemet.

Dataskydd i appar

iOS Software Development Kit (SDK) innehåller en komplett uppsättning API:er som gör det enkelt för tredjepartsutvecklare och internutvecklare på företag att använda dataskydd och göra sina appar så säkra som möjligt. Dataskydd finns för fil- och databas-API:er, som NSFileManager, CoreData, NSData och SQLite.

Mail-appens databas (inklusive bilagor), hanterade böcker, Safari-bokmärken, appstartbilder och platsdata lagras också i krypterad form med nycklar som skyddas av användarens lösenkod på enheten. Kalender (exklusive bilagor), Kontakter, Påminnelser, Anteckningar, Meddelanden och Bilder använder dataskyddsrättigheten Protected Until First User Authentication.

Appar som installeras av användaren, utan att de har en angiven dataskyddsklass, använder Protected Until First User Authentication som förval.

Tillbehör

Licensprogrammet ”Made for iPhone, iPad and iPod touch” (MFi) ger granskade och godkända tillbehörstillverkare tillgång till iAP-protokollet (iPod Accessories Protocol) och nödvändiga maskinvarukomponenter.

När ett MFi-tillbehör kommunicerar med en iOS-enhet via Lightning-kontakten eller via Bluetooth ber enheten tillbehöret att bevisa att det är godkänt av Apple genom att svara med ett certifikat från Apple, som sedan

verifieras av enheten. Enheten skickar då en utmaning som tillbehöret måste besvara med ett signerat svar. Den här processen hanteras helt av en anpassad integrerad krets (IC) som Apple tillhandahåller till godkända tillbehörstillverkare, och den är osynlig för tillbehöret självt.

Tillbehör kan begära tillgång till andra överföringsmetoder och funktioner, till exempel tillgång till digitala ljudströmmar via Lightning-kabeln eller platsinformation som skickas via Bluetooth. En integrerad krets för autentisering garanterar att endast godkända tillbehör ges full tillgång till enheten. Om ett tillbehör inte kan autentiseras begränsas dess tillgång till analogt ljud och ett litet urval av seriella (UART) reglage för ljuduppspelning.

AirPlay använder också integrerade autentiseringskretsar för att verifiera att mottagarna har godkänts av Apple. Strömmat AirPlay-ljud och CarPlay-video använder MFi-SAP (Secure Association Protocol) som krypterar kommunikationen mellan tillbehöret och enheten med hjälp av AES-128 i CTR-läge. Tillfälliga nycklar utväxlas under ECDH-nyckelutbytet (Curve25519) och signeras med den integrerade autentiseringskretsens 1 024-bitars RSA-nyckel som en del av protokollet STS (Station-to-Station).

HomeKit

HomeKit ger tillgång till en infrastruktur för automatisering av hemmet som drar nytta av iCloud och iOS-säkerhet för att skydda och synkronisera privata data utan att avslöja dem för Apple.

HomeKit-identitet

HomeKit-identiteter och -säkerhet bygger på publika/privata Ed25519-nyckelpar. För varje HomeKit-användare genereras ett Ed25519-nyckelpar på iOS-enheten. Detta blir den användarens HomeKit-identitet. Den används till att autentisera kommunikationen mellan flera iOS-enheter samt mellan iOS-enheter och tillbehör.

Nycklarna förvaras i nyckelringen och inkluderas bara i krypterade säkerhetskopieringar av nyckelringen. Nycklarna synkroniseras mellan enheter med hjälp av iCloud-nyckelringen där den finns. HomePod och Apple TV får nycklar via tryck för att installera eller installationsläget som beskrivs nedan. Nycklar delas från en iPhone till en parkopplad Apple Watch via IDS (Apple Identity Service).

Kommunikation med HomeKit-tillbehör

HomeKit-tillbehör genererar sina egna Ed25519-nyckelpar för kommunikation med iOS-enheter. Om tillbehöret återställs till fabriksinställningarna genereras ett nytt nyckelpar.

För att upprätta ett förhållande mellan en iOS-enhet och ett HomeKit-tillbehör utväxlas nycklar med protokollet Secure Remote Password (3072 bitar) med hjälp av en åttasiffrig kod som tillhandahålls av tillbehörets tillverkare, anges på iOS-enheten av användaren och sedan krypteras med CHACHA20-POLY1305 AEAD med HKDF-SHA-512-härledda nycklar. Tillbehörets MFi-certifiering kontrolleras också under installationen. Tillbehör som saknar en MFi-krets kan bygga in stöd för programvaruautentisering med iOS 11.3 eller senare.

När iOS-enheten och HomeKit-tillbehöret kommunicerar under användning autentiserar de varandra med hjälp av nycklarna som utväxlades i den ovanstående processen. Varje session upprättas med STS-protokollet

(Station-to-Station) och krypteras med HKDF-SHA-512-härledda nycklar baserade på sessionsgenererade Curve25519-nycklar. Det gäller både IP-baserade tillbehör och Bluetooth LE-tillbehör.

Bluetooth LE-tillbehör som har stöd för utsända notiser får en krypteringsnyckel skickad till sig av en parkopplad iOS-enhet via en säker session. Nyckeln används till att kryptera data gällande statusförändringar på enheten som meddelas via Bluetooth LE-annonseringar. Den utsända krypteringsnyckeln är en HKDF-SHA-512-härledd nyckel och data krypteras med algoritmen CHACHA20-POLY1305 AEAD (Authenticated Encryption with Associated Data). Den utsända krypteringsnyckeln ändras då och då av iOS-enheten och synkroniseras till andra enheter som använder iCloud enligt beskrivningen i avsnittet Datasynkronisering mellan enheter och användare nedan.

Lokal datalagring

HomeKit lagrar data om hemmet, tillbehör, scenarier och användare i användarens iOS-enhet. Dessa lagrade data krypteras med nycklar som härleds från användarens HomeKit-identitetsnycklar, plus ett slumpgenererat nonce-värde. Dessutom lagras HomeKit-data med hjälp av dataskyddsklassen Protected Until First User Authentication. HomeKit-data säkerhetskopieras bara i krypterade säkerhetskopior, så exempelvis okrypterade iTunes-säkerhetskopior innehåller inte några HomeKit-data.

Datasynkronisering mellan enheter och användare

HomeKit-data kan synkroniseras mellan en användares iOS-enheter med hjälp av iCloud och iCloud-nyckelringen. Användarens HomeKit-data krypteras under synkroniseringen med nycklar som härleds ur användarens HomeKit-identitet och ett slumpgenererat nonce-värde. Dessa data hanteras som en OBB-fil (Opaque Binary Blob) under synkroniseringen. Den senaste OBB-filen lagras i iCloud och används till synkronisering, men inte till någonting annat. Eftersom den är krypterad med nycklar som endast är tillgängliga på användarens iOS-enheter är dess innehåll inte tillgängligt vid överföring och lagring på iCloud.

HomeKit-data synkroniseras också mellan flera användare i samma bostad. Vid den här processen används samma autentisering och kryptering som mellan en iOS-enhet och ett HomeKit-tillbehör. Autentiseringen bygger på de publika Ed25519-nycklarna som utväxlas mellan enheterna när en användare läggs till i ett hem. När en ny användare har lagts till i ett hem autentiseras och krypteras all vidare kommunikation med STS-protokollet och sessionsgenererade nycklar.

Nya användare kan bara läggas till av den användare som skapade hemmet i HomeKit, eller av en annan användare med redigeringsbehörighet. Ägarens enhet konfigurerar tillbehören med den nya användarens publika nyckel, så att tillbehöret kan autentisera och ta emot kommandon från den nya användaren. När en användare med redigeringsbehörighet lägger till en ny användare delegeras processen till en hemhubb för att slutföra åtgärden.

Processen för att tillgängliggöra Apple TV för användning med HomeKit utförs automatiskt när användaren loggar in till iCloud. Tvåfaktorsautentisering måste vara aktiverat för iCloud-kontot. Apple TV och ägarens enhet utbyter tillfälliga publika Ed25519-nycklar via iCloud. När ägarens enhet och Apple TV finns i samma nätverk används de tillfälliga nycklarna till att skapa en säker anslutning via det lokala nätverket

med STS-protokoll och sessionsgenererade nycklar. Vid den här processen används samma autentisering och kryptering som mellan en iOS-enhet och ett HomeKit-tillbehör. Via den här säkra lokala anslutningen överför ägarens enhet användarens publika/privata Ed25519-nyckelpar till Apple TV. Dessa nycklar används sedan till att säkra kommunikationen mellan Apple TV och HomeKit-tillbehören, och även mellan Apple TV och andra iOS-enheter som ingår i HomeKit-hemmet.

Om användaren bara har en enhet, och inte lägger till några andra användare i hemmet, synkroniseras inga HomeKit-data till iCloud.

Hemdata och appar

Appars tillgång till hemdata regleras av användarens integritets-inställningar. Användaren tillfrågas när appar begär tillgång till hemdata, på samma sätt som med Kontakter, Bilder och andra datakällor i iOS. Om användaren ger sitt godkännande får apparna tillgång till antalet rum, namnen på tillbehören och vilka rum de finns i, samt annan information som anges i dokumentationen för HomeKit-utvecklare på https://developer.apple.com/homekit/.

HomeKit och Siri

Siri kan användas till att söka efter och styra tillbehör samt för att aktivera scenarier. Mycket begränsad information om hemmets konfiguration överförs anonymt till Siri. Det gäller namnen på rum, tillbehör och scenarier som är nödvändiga för att förstå kommandon. Ljud som skickas till Siri kan gälla specifika tillbehör eller kommandon, men sådana Siri-data kopplas inte till andra Apple-funktioner som HomeKit. Mer information finns i avsnittet om Siri i avsnittet Internettjänster i det här dokumentet.

HomeKit-IP-kameror

IP-kameror i HomeKit skickar video- och ljudströmmar direkt till iOS-enheten i det lokala nätverket som tar emot strömmarna. Strömmarna krypteras med slumpmässigt genererade nycklar i iOS-enheten och IP-kameran som utväxlas via den säkra HomeKit-sessionen till kameran. När iOS-enheten inte finns i det lokala nätverket vidarebefordras de krypterade strömmarna via hemhubben till iOS-enheten. Hemhubben krypterar inte strömmarna, utan fungerar bara som ett relä mellan iOS-enheten och IP-kameran. När en app visar HomeKit-IP-kamerans videovy för användaren renderar HomeKit videobildrutorna säkert från en separat systemprocess så att appen varken kan komma åt eller lagra videoströmmen. Utöver detta saknar appar behörighet att göra skärmavbilder från den här strömmen.

iCloud-fjärråtkomst för HomeKit-tillbehör

HomeKit-tillbehör kan ansluta direkt till iCloud och ge iOS-enheter kontroll över tillbehöret när Bluetooth- eller Wi-Fi-kommunikation inte är möjlig.

iCloud-fjärråtkomst har omsorgsfullt utformats så att det går att kontrollera och skicka notiser till tillbehören utan att avslöja för Apple vilka tillbehören är, eller vilka kommandon och notiser som skickas. HomeKit skickar inte informationen om hemmet via iCloud-fjärråtkomst.

När en användare skickar ett kommando med hjälp av iCloud-fjärråtkomst autentiseras både tillbehöret och iOS-enheten inför varandra, och alla data krypteras med samma metod som har beskrivits för lokala anslutningar. Innehållet i kommunikationen krypteras och kan inte läsas av Apple. Adresseringen via iCloud bygger på de iCloud-identifierare som har registrerats under konfigurationsprocessen.

Tillbehör som stöder iCloud-fjärråtkomst etableras under tillbehörets konfigurationsprocess. Etableringsprocessen inleds med att användaren loggar in till iCloud. Därefter ber iOS-enheten tillbehöret att signera en utmaning med hjälp av Apples autentiseringscoprocessor som är inbyggd i alla Built for HomeKit-tillbehör. Tillbehöret genererar dessutom elliptiska prime256v1-kurvnycklar, och den offentliga nyckeln skickas till iOS-enheten tillsammans med den undertecknade utmaningen och autentiseringscoprocessorns X.509-certifikat. Dessa används för att begära ett certifikat för tillbehöret från iClouds etableringsserver. Certifikatet lagras av tillbehöret, men det innehåller ingen information som identifierar tillbehöret, annat än att det har fått tillgång till HomeKits iCloud-fjärråtkomst. iOS-enheten som genomför etableringen skickar även ett informationspaket till tillbehöret med de webbadresser och den övriga information som krävs för att ansluta till iCloud-fjärråtkomstservern. Denna information är inte specifik för någon användare eller något tillbehör.

Varje tillbehör registrerar en lista med tillåtna användare på iCloud-fjärråtkomstservern. Dessa användare har fått tillstånd att styra tillbehöret av den person som har lagt till tillbehöret i hemmet. Användarna tilldelas ett ID av iCloud-servern och kan kopplas till ett iCloud-konto i syfte att skicka och ta emot notismeddelanden och svar från tillbehöret. På samma sätt får tillbehören ID-nummer från iCloud, men dessa ID-nummer ger ingen information om själva tillbehöret.

När ett tillbehör ansluter till HomeKits iCloud-fjärråtkomstserver lämnar den över sitt certifikat och en passersedel. Passersedeln kommer från en annan iCloud-server och är inte unik för ett tillbehör. När ett tillbehör ber om en passersedel skickar den information om tillverkaren, modellen och versionen på den fasta programvaran i sin begäran. Ingen information som gör det möjligt att identifiera användaren eller hemmet skickas i denna begäran. Anslutningen till passersedelsservern autentiseras inte. Detta skyddar användarens integritet.

Tillbehören ansluter till iCloud-fjärråtkomstservern via HTTP/2. Anslutningen skyddas med TLS v1.2 med AES-128-GCM och SHA-256. Tillbehöret håller anslutningen till iCloud-fjärråtkomstservern öppen så att den kan ta emot inkommande meddelanden och skicka svar och utgående notiser till iOS-enheter.

HomeKit TV-fjärrkontrollstillbehör

HomeKit TV-fjärrkontrollstillbehör från tredje part tillhandahåller HID-händelser och Siri-ljud till en associerad Apple TV som har lagts till via appen Hem. HID-händelserna skickas via den säkra sessionen mellan Apple TV och fjärrkontrollen. En Siri-kompatibel TV-fjärrkontroll skickar ljuddata till Apple TV när användaren uttryckligen aktiverar mikrofonen på fjärrkontrollen med en dedikerad Siri-knapp. Ljudrutorna skickas direkt till Apple TV med en dedikerad lokal nätverksanslutning mellan Apple TV och fjärrkontrollen. Den lokala nätverksanslutningen krypteras med ett sessionsgenererat HKDF-SHA-512-härlett nyckelpar som förhandlas via HomeKit-sessionen mellan Apple TV:n och TV-fjärrkontrollen. HomeKit avkrypterar ljudrutorna på Apple TV och vidarebefordrar dem till Siri-appen där de behandlas med samma integritetsskydd som all Siri-ljudinmatning.

SiriKit

Siri använder iOS-tilläggsmekanismen till att kommunicera med appar från tredje part. Trots att Siri har tillgång till iOS-kontakter och enhetens nuvarande plats kontrollerar Siri att appen med tillägget har åtkomstbehörighet för iOS-skyddade användardata och att appen har behörighet innan information tillhandahålls till den. Siri skickar endast det relevanta fragmentet av den ursprungliga användarfrågetexten till tillägget. Om appen exempelvis inte har tillgång till iOS-kontakter kommer Siri inte att lösa relationen i en användarfråga av slaget ”Skicka 100 kr till mamma med <betalningsapp>”. I det här fallet kommer tilläggsappen endast att se ”mamma” via det fragment som skickas till den. Om appen däremot har tillgång till iOS-kontakter mottar den iOS-kontaktinformationen för användarens mamma. Om en referens till en kontakt görs i brödtexten för ett meddelande, t.ex. ”Skicka ett meddelande till min mamma i <meddelandeapp> att min bror är fantastisk” löser Siri inte ”min bror” oavsett appens TCC. Innehåll som presenteras av appen kan skickas till servern för att göra det möjligt för Siri att förstå de ord en användare kanske använder i appen. I fall som ”Skaffa skjuts hem till mamma med <appnamn>”, där användarens förfrågan kräver att platsinformation hämtas från användarens kontakter, tillhandahåller Siri den platsinformationen till appens tillägg, endast för denna förfrågan, oavsett vilken tillgång till plats- eller kontaktinformation som appen har.

Vid körning tillåter Siri att den SiriKit-aktiverade appen tillhandahåller en uppsättning anpassade ord som är specifika för applikationsinstansen. Dessa anpassade ord är knutna till den slumpmässiga identifierare som beskrivs i Siri-avsnittet i det här dokumentet och har samma livslängd.

HealthKit

HealthKit lagrar och slår ihop data från hälso- och träningsappar med användarens tillåtelse. HealthKit fungerar också direkt med hälso- och träningstillbehör som Bluetooth LE-kompatibla anslutna pulsmätare och rörelsecoprocessorn som är inbyggd i många iOS-enheter.

Hälsodata

Med HealthKit kan användare lagra och sammanställa sina hälsodata från källor som appar, enheter och vårdgivare. Dessa data lagras i dataskyddsklassen Protected Unless Open. Tillgången till dessa data upphör tio minuter efter att enheten låses och de blir tillgängliga igen nästa gång användaren anger sin lösenkod eller använder Touch ID eller Face ID till att låsa upp enheten.

HealthKit slår också ihop hanteringsdata, som tillgångsbehörigheter för appar, namn på enheter som är anslutna till HealthKit samt schemaläggningsinformation som används för att starta appar när nya data är tillgängliga. Dessa data lagras i dataskyddsklassen Protected Until First User Authentication.

Tillfälliga journalfiler sparar hälsodata som genereras när enheten är låst, till exempel när användaren tränar. Dessa lagras i dataskyddsklassen Protected Unless Open. När enheten låses upp importeras de tillfälliga journalfilerna till den primära hälsodatabasen och raderas när sammanslagningen är klar.

Hälsodata kan lagras på iCloud. När iCloud-lagring används synkroniseras hälsodata mellan enheter och säkras av kryptering som skyddar data både vid överföring och i vila. Hälsodata inkluderas endast i krypterade iTunes-säkerhetskopior. De inkluderas varken i okrypterade iTunes-säkerhetskopieringar eller iCloud-säkerhetskopiering.

Medicinska journaler

Användare kan logga in på hälsosystem hos kompatibla vårdgivare i appen Hälsa. När en användare ansluter till ett hälsosystem autentiserar användaren med OAuth 2-klientautentiseringsuppgifter. När användaren är ansluten hämtas medicinska journaldata direkt från vårdgivaren via en anslutning som skyddas med TLS v1.2. När de har hämtats lagras medicinska journaler tryggt och säkert tillsammans med andra hälsodata.

Dataintegritet

Bland de data som lagras i databasen finns också metadata som kan användas till att spåra posternas ursprung. Dessa metadata innehåller en appidentifierare som identifierar den app som lagrar respektive post. Dessutom finns ett valfritt metadataobjekt som kan innehålla en digitalt signerad kopia av posten. Syftet med det är att erbjuda dataintegritet för poster som genererats av en betrodd enhet. Formatet som används för den digitala signaturen är CMS (Cryptographic Message Syntax) som specificeras i IETF RFC 5652.

Tillgång från tredjepartsappar

Tillgången till API:et för HealthKit styrs med hjälp av rättigheter, och appar måste anpassa sig efter begränsningar för hur hälsodata får användas. Exempelvis får appar inte använda hälsodata i reklamsyften. Alla appar måste också innehålla en integritetspolicy som specificerar appens användning av hälsodata.

Appars tillgång till hälsodata regleras av användarens integritets-inställningar. Användaren tillfrågas när appar begär tillgång till hälsodata, på samma sätt som med Kontakter, Bilder och andra datakällor i iOS. När det gäller hälsodata ges dock apparna separat tillgång för läsning och skrivning av data, liksom separat tillgång för de olika typerna av hälsodata. Användarna kan visa och återkalla behörigheter för tillgång till hälsodata på fliken Källor i appen Hälsa.

Appar med behörighet att skriva data kan också läsa de data de skriver. Appar med behörighet att läsa data kan läsa data från alla källor. Däremot kan ingen app avgöra vilka behörigheter andra appar har. Dessutom kan appar inte avgöra helt säkert om de har fått behörighet att läsa hälsodata eller inte. När en app inte har läsbehörighet returnerar alla förfrågningar ”inga data” – samma svar som en tom databas skulle ge. Detta förhindrar att appar drar slutsatser om användarens hälsa genom att ta reda på vilka typer av hälsodata användaren registrerar.

Medicinskt ID

I appen Hälsa har användarna möjlighet att fylla i ett formulär för ett medicinskt ID med information som kan vara viktig i medicinska nödsituationer. Informationen anges eller uppdateras manuellt och synkroniseras inte med informationen i hälsodatabaserna.

Den medicinska ID-informationen visas med ett tryck på nödknappen på låsskärmen. Informationen lagras på enheten i dataskyddsklassen *No Protection* så att den kan visas utan att enhetens lösenkod anges. Medicinskt ID är en valfri funktion där användarna själva kan bestämma

hur de vill balansera säkerhet mot integritet. Dessa data säkerhetskopieras med iCloud-säkerhetskopiering och synkroniseras inte mellan enheter med hjälp av CloudKit.

ReplayKit

ReplayKit är ett ramverk som gör det möjligt för utvecklare att lägga till funktioner för inspelning och direktsändningar i sina appar. Dessutom får användare möjlighet att kommentera sina inspelningar och sändningar via enhetens framåtvända kamera och mikrofon.

Filminspelning

Flera säkerhetslager är inbyggda vid inspelningen av en film:

- **Tillståndsdialogruta:** Innan inspelningen börjar visar ReplayKit en varningsdialogruta där användaren blir ombedd att bekräfta sin avsikt att spela in skärmen och spela in med mikrofonen och kameran på framsidan. Varningen visas en gång per approcess och visas igen om appen lämnas i bakgrunden längre tid än 8 minuter.
- **Skärm- och ljudinsamling:** Skärm- och ljudinsamling sker via appens process i ReplayKits bakgrundsprogram *replayd*. Det säkerställer att det inspelade innehållet aldrig är tillgängligt för approcessen.
- **Skapa och lagra film:** Filmfilen skrivs till en katalog som bara är tillgänglig för ReplayKits undersystem. Den är aldrig tillgänglig för några appar. Detta förhindrar att inspelningar används av tredje part utan användarens godkännande.
- **Förhandsvisning och delning:** Användaren kan förhandsvisa och dela filmen med gränssnitt som tillhandahålls av ReplayKit. Detta gränssnitt presenteras vid sidan av processen via iOS-tilläggsinfrastrukturen och har tillgång till den skapade filmfilen.

Sändning

- **Skärm- och ljudinsamling:** Mekanismen för skärm- och ljudinsamling under sändning är identisk med filminspelning och sker i *replayd*.
- **Sändningstillägg:** För att tjänster från tredje part ska kunna delta i ReplayKit-sändningar måste de skapa två nya tillägg som konfigureras med slutpunkten com.apple.broadcast-services:
  - Ett gränssnittstillägg där användaren kan ställa in sin sändning.
  - Ett överföringstillägg som hanterar överföring av video- och ljuddata till tjänstens back-end-servrar.

Arkitekturen säkerställer att värdapparna inte har någon behörighet för det video- och ljudinnehåll som sänds – endast ReplayKit och sändningstilläggen från tredje part har tillgång.

- **Sändningsväljare:** För att välja vilken sändningstjänst som ska användas tillhandahåller ReplayKit en visningsstyrenhet (påminner om UIActivityViewController) som utvecklaren kan använda i sin app. Visningsstyrenheten implementeras med UIRemoteViewController SPI och är ett tillägg som finns i ReplayKit-ramverket. Det ligger inte i samma process som värdappen.
- **Systemsändningsväljare:** Gör det möjligt för användare att starta systemsändning direkt från app med samma systemdefinierade gränssnitt som är tillgängligt via Kontrollcenter. Gränssnittet implementeras med UIRemoteViewController SPI och är ett tillägg som finns i ReplayKit-ramverket. Det ligger inte i samma process som värdappen.

- **Överföringstillägg:** Överföringstillägget som sändningstjänster från tredje part implementerar för att hantera video- och ljudinnehåll under sändning kan välja att ta emot innehåll på två sätt:
  – Små kodade MP4-klipp.
  – Okodade raw-samplingsbuffertar.
    - **MP4-klipphantering:** I det här hanteringsläget genereras de små kodade MP4-klippen av *replayd* och lagras på en privat plats som endast är tillgänglig för ReplayKits undersystem. När ett filmklipp har genererats skickar *replayd* platsen för filmklippet till överföringstillägget från tredje part via NSExtension-begäran SPI (XPC-baserad). *replayd* genererar även en engångssandlådetoken som också skickas till överföringstillägget så att tillägget får tillgång till det aktuella filmklippet under tilläggsbegäran.
    - **Samplingsbufferthantering:** I det här hanteringsläget serialiseras video- och ljuddata och skickas till överföringstillägget från tredje part i realtid via en direkt-XPC-anslutning. Videodata kodas genom att extrahera IOSurface-objektet från videosamplingsbufferten, säkert kodat som ett XPC-objekt. Data skickas sedan via XPC till tillägget från tredje part och avkodas säkert tillbaka till ett IOSurface-objekt.

Säkra anteckningar

I Anteckningar finns en funktion för säkra anteckningar som användare kan använda till att skydda innehållet i särskilda anteckningar. Säkra anteckningar är krypterade med en lösenfras som användaren anger och som sedan måste anges för att visa anteckningarna i iOS och macOS samt på iCloud-webbplatsen.

När en användare skyddar en anteckning härleds en nyckel på 16 byte från användarens lösenfras med hjälp av PBKDF2 och SHA256. Anteckningens innehåll krypteras med AES-GCM. Nya poster skapas i Core Data och CloudKit för att kunna spara den krypterade anteckningen, taggen och initieringsvektorn, och de ursprungliga anteckningsposterna raderas. Krypterade data skrivs inte till de posterna. Bilagor krypteras på samma sätt. Bilagor som stöds är bland annat bilder, skisser, tabeller, kartor och webbplatser. Anteckningar som innehåller andra slags bilagor kan inte krypteras, och bilagor som inte stöds kan inte läggas till i säkra anteckningar.

När en användare anger rätt lösenfras, antingen för att visa eller skapa en säker anteckning, öppnar Anteckningar en säker session. Medan sessionen är öppen behöver inte användaren ange lösenfrasen, eller använda Touch ID eller Face ID, för att visa eller skydda andra anteckningar. Om du använder en annan lösenfras för vissa anteckningar gäller den säkra sessionen bara för de anteckningar som skyddas med den aktuella lösenfrasen. Den säkra sessionen stängs när:

- Användaren trycker på knappen Lås nu i Anteckningar.
- Anteckningar placeras i bakgrunden under mer än 3 minuter.
- Enheten låses.

En användare som glömmer sin lösenfras kan ändå visa säkra anteckningar eller skydda ytterligare anteckningar om denne har aktiverat Touch ID eller Face ID på enheten. Dessutom visas en ledtråd som användaren själv har

angett i Anteckningar efter tre misslyckade försök att ange lösenfrasen. Användaren måste känna till den nuvarande lösenfrasen för att kunna ändra den.

Användare kan skapa en ny lösenfras om de har glömt den nuvarande. Med den här funktionen kan användare skapa nya säkra anteckningar med en ny lösenfras, men de kan inte se de anteckningar som har skyddats tidigare. Det går fortfarande att visa de tidigare skyddade anteckningarna om du kommer ihåg den gamla lösenfrasen. Användarens lösenkod för iCloud-kontot krävs för att skapa en ny lösenfras.

Delade anteckningar

Anteckningar kan delas med andra. Delade anteckningar är inte E2E-krypterade. Apple använder den krypterade CloudKit-datatypen för text eller bilagor som användaren placerar i en anteckning. Resurser krypteras alltid med en nyckel som är krypterad i CKRecord. Metadata, som skapelse- och ändringsdatum, krypteras inte. CloudKit hanterar processen genom vilken deltagarna kan kryptera/avkryptera varandras data.

Apple Watch

Apple Watch använder säkerhetsfunktionerna och -teknikerna som utvecklats för iOS till att skydda data på enheten och vid kommunikation med dess parkopplade iPhone och internet. Detta omfattar tekniker som dataskydd och behörighetsstyrning av nyckelringen. Användarens lösenkod är även knuten till enhetens UID när krypteringsnycklar skapas.

Parkopplingen av Apple Watch med iPhone säkras med en direkt parkoppling vid utväxling av publika nycklar, följt av den Bluetooth LE-länkade delade hemligheten. Apple Watch visar ett animerat mönster som fångas av kameran på iPhone. Mönstret innehåller en kodad hemlighet som används för den direkta parkopplingen via BLE 4.1. En vanlig BLE-inmatning av kodnyckeln används som reservmetod vid parkopplingen om behov uppstår.

När Bluetooth LE-sessionen har upprättats och krypterats med det högsta säkerhetsprotokollet som finns i Bluetooth-kärnspecifikationen utväxlar Apple Watch och iPhone nycklar med en process som har anpassats från IDS (Apple Identity Service) enligt beskrivningen under iMessage i avsnittet Internettjänster i det här dokumentet. När nycklarna har utväxlats kasseras Bluetooth-sessionsnyckeln. All kommunikation mellan Apple Watch och iPhone krypteras sedan med IDS, med de krypterade Bluetooth-, Wi-Fi- och mobillänkarna som ett ytterligare krypteringslager. Bluetooth LE-adressen roteras var 15:e minut för att minska risken för att kommunikationen avlyssnas.

Appar som kräver strömmande data stöds genom att kryptering tillhandahålls enligt metoder som beskrivs under FaceTime i avsnittet Internettjänster i det här dokumentet. Metoderna drar antingen nytta av IDS-tjänsten som den parkopplade iPhone-enheten tillhandahåller eller en direkt internetanslutning.

Apple Watch drar nytta av maskinvarukrypterad lagring och klassbaserat skydd av filer och nyckelringsobjekt, vilket beskrivs i avsnittet Kryptering och dataskydd i det här dokumentet. Nyckelsamlingar med

behörighetsstyrning för nyckelringsobjekt används också. Nycklar som används för kommunikation mellan klockan och iPhone säkras dessutom med ett klassbaserat skydd.

Om Apple Watch inte är inom Bluetooth-räckvidd går det att använda Wi-Fi eller mobilnät istället. Apple Watch ansluter automatiskt till Wi-Fi-nätverk som dess parkopplade iPhone redan har anslutit till och vars inloggningsuppgifter har synkroniserats till Apple Watch medan båda enheterna var inom räckvidden. Det här automatiska anslutningsbeteendet kan konfigureras på enskild nätverksnivå i Wi-Fi-avsnittet i appen Inställningar på Apple Watch. Wi-Fi-nätverk som ingen av enheterna har anslutit till tidigare kan anslutas manuellt i Wi-Fi-avsnittet i appen Inställningar på Apple Watch.

När Apple Watch och iPhone är utom räckvidd ansluter Apple Watch direkt till iCloud- och Gmail-servrar för att hämta e-post i Mail istället för att synkronisera Mail-data med dess parkopplade iPhone via internet. För Gmail-konton måste användaren autentisera till Google i avsnittet Mail i Apple Watch-appen på iPhone. Den OAuth-token som fås från Google skickas över till Apple Watch i ett krypterat format via IDS (Apple Identity Service) så att den kan användas till att hämta e-post. Denna OAuth-token används aldrig för anslutning till Gmail-servern från den iPhone som är parkopplad.

Apple Watch kan låsas manuellt genom att hålla in sidoknappen. Så länge handledsavkänning inte är avaktiverad blir enheten dessutom automatiskt låst ett ögonblick efter att den tas bort från användarens handled. När Apple Watch är låst kan Apple Pay endast användas genom att ange klockans lösenkod. Handledsavkänning stängs av med Apple Watch-appen på iPhone. Den här inställningen kan även genomdrivas med en MDM-lösning.

Den iPhone som parkopplats kan också låsa upp klockan, förutsatt att klockan bärs. Det åstadkoms genom att upprätta en anslutning som autentiseras med nycklarna som genererades under parkopplingen. iPhone skickar nyckeln som klockan använder för att låsa upp sina dataskyddsnycklar. iPhone känner inte till klockans lösenkod och den överförs inte heller. Den här funktionen kan stängas av med Apple Watch-appen på iPhone.

Apple Watch kan endast parkopplas med en iPhone åt gången. iPhone visar anvisningar om att radera allt innehåll och alla data från Apple Watch när parkopplingen tas bort.

Apple Watch kan ställas in för en systemprogramuppdatering samma natt. Information om hur lösenkoden till Apple Watch lagras för användning under uppdateringen finns i avsnittet Nyckelsamlingar i det här dokumentet.

Om Hitta min iPhone aktiveras på den iPhone som är parkopplad kan aktiveringslåset också användas på Apple Watch. Aktiveringslåset gör det svårare för någon att använda eller sälja en Apple Watch som har tappats bort eller stulits. Aktiveringslåset kräver användarens Apple-ID och lösenord för att bryta parkopplingen, radera eller återaktivera en Apple Watch.

Nätverkssäkerhet

Förutom de inbyggda säkerhetsfunktionerna som Apple använder för att skydda data på iOS-enheter finns det många åtgärder för nätverkssäkerhet som företag och organisationer kan vidta för att skydda information som överförs till och från iOS-enheter.

Mobila användare behöver kunna ansluta till företagets nätverk oavsett var de befinner sig. Därför är det viktigt att se till att de är auktoriserade och att deras data skyddas under överföringen. iOS använder – och ger utvecklare tillgång till – vanliga nätverksprotokoll för autentiserad, behörighetsskyddad och krypterad kommunikation. För att uppnå den här höga säkerhetsnivån använder iOS beprövade tekniker och de senaste standarderna för både Wi-Fi och mobildatanät.

På andra plattformar behövs brandväggsprogramvara som skyddar öppna kommunikationsportar mot intrång. Eftersom iOS har en minskad attackyta tack vare att antalet lyssnande portar begränsas, och att onödiga nätverksfunktioner som telnet, skal eller en webbserver inte används, behövs ingen extra brandväggsprogramvara på iOS-enheter.

TLS

iOS stöder TLS-säkerhet (TLS v1.0, TLS v1.1, TLS v1.2) och DTLS. Det stöder både AES-128 och AES-256 och föredrar kodpaket med PFS (Perfect Forward Secrecy). Safari, Kalender, Mail och andra internetappar använder det här protokollet automatiskt för att skapa en krypterad kommunikationskanal mellan enheter och nätverkstjänster. API:er på hög nivå (som CFNetwork) gör det enkelt för utvecklare att använda TLS i sina appar, medan API:er på låg nivå (Network.framework) ger en mer finmaskig kontroll. SSLv3 tillåts inte av CFNetwork, och appar som använder WebKit (t.ex. Safari) får inte öppna SSLv3-anslutningar.

I iOS 11 eller senare och macOS High Sierra eller senare är SHA-1-certifikat inte längre tillåtna för TLS-anslutningar, med undantag för om de är betrodda av användaren. Certifikat med RSA-nycklar som är kortare än 2048 bitar är inte heller tillåtna. Den symmetriska kodningsgruppen RC4 är föråldrad i iOS 10 och macOS Sierra. Som förval är RC4 inte aktiverat för TLS-klienter eller -servrar som implementeras med SecureTransport API, och de kan inte ansluta när RC4 är enda tillgängliga kodningsgruppen. För att öka säkerheten bör tjänster och appar som kräver RC4 uppgraderas för användning med moderna, säkra kodningsgrupper. I iOS 12.1 måste certifikat som är utfärdade efter den 15 oktober 2018 från ett systembetrott rotcertifikat loggas i en betrodd Certificate Transparency-logg för att tillåtas för TLS-anslutningar.

App Transport Security

App Transport Security tillhandahåller förvalda anslutningskrav så att appar följer bästa praxis för säkra anslutningar när API:erna NSURLConnection, CFURL eller NSURLSession används. Som förval begränsar App Transport Security kodvalet till att endast inkludera grupper som tillhandahåller FS (forward secrecy), mer specifikt ECDHE_ECDSA_

AES och ECDHE_RSA_AES i GCM- eller CBC-läge. Appar kan avaktivera FS-kravet per domän. I sådana fall läggs RSA_AES till i uppsättningen tillgängliga koder.

Servrarna måste stöda TLS v1.2 och FS samt certifikaten vara giltiga och signerade med SHA-256 eller bättre och ha minst en 2 048-bitars RSA-nyckel eller en elliptisk 256-bitars kurvnyckel.

Nätverksanslutningar som inte uppfyller dessa krav kommer att misslyckas, förutsatt att appen inte förbigår App Transport Security. Ogiltiga certifikat leder obevekligen till fel och att ingen anslutning upprättas. App Transport Security används automatiskt i appar som kompileras för iOS 9 eller senare.

VPN

Säkra nätverkstjänster som VPN kräver oftast minimala insatser vad gäller installation och konfiguration för att fungera med iOS-enheter. iOS-enheter fungerar med VPN-servrar som stöder följande protokoll och autentiseringsmetoder:

- IKEv2/IPSec med autentisering med en delad hemlighet, RSA-certifikat, **ECDSA**-certifikat, EAP-MSCHAPv2 eller EAP-TLS.
- SSL-VPN med lämplig klientapp från App Store.
- Cisco IPSec med användarautentisering via lösenord och maskinautentisering med delad hemlighet och certifikat.
- L2TP/IPSec med användarautentisering via MS-CHAPV2-lösenord och maskinautentisering med delad hemlighet.

iOS stöder följande:

- **VPN On Demand** för nätverk som använder certifikatbaserad autentisering. IT-policyer anger vilka domäner som kräver en VPN-anslutning genom att använda en konfigurationsprofil.
- **VPN per app**, vilket möjliggör mycket mer detaljkontrollerade VPN-anslutningar. MDM kan ange en anslutning för varje hanterad app och specifika domäner i Safari. Detta bidrar till att säkra data alltid skickas till och från företagets nätverk – och att användarnas personliga data inte skickas.
- **Alltid på-VP**N som kan konfigureras för enheter som hanteras via en MDM-lösning och övervakas med Apple Configurator 2, Apple School Manager eller Apple Business Manager. Det gör att användarna slipper slå på VPN för att aktivera skyddet när de ansluter till mobilnät och Wi-Fi-nätverk. Alltid på-VPN ger organisationen full kontroll över trafiken till och från enheter genom att all IP-trafik dirigeras genom en tunnel tillbaka till organisationen. Det tunnelprotokoll som används som förval, IKEv2, skyddar trafiken genom datakryptering. Organisationen kan övervaka och filtrera trafiken till och från enheter, skydda data inom nätverket och begränsa enheters tillgång till internet.

Wi-Fi

iOS stöder Wi-Fi-protokoll av branschstandard, till exempel WPA2 Enterprise, som ger autentiserad tillgång till trådlösa företagsnätverk. WPA2 Enterprise använder 128-bitars AES-kryptering så att användarna kan känna sig trygga i vetskapen om att deras data förblir skyddade när de skickar och tar emot kommunikation via en Wi-Fi-anslutning. Tack vare stöd för 802.1X kan iOS-enheter integreras i en mängd olika RADIUS-

autentiseringsmiljöer. Bland de trådlösa autentiseringsmetoder med 802.1X som stöds av iPhone och iPad finns EAP-TLS, EAP-TTLS, EAP-FAST, EAP-SIM, PEAPv0, PEAPv1 och LEAP.

Utöver dataskydd utökar iOS WPA2-nivåskyddet till unicast- och multicast-hanteringsramar via tjänsten Protected Management Frame som refereras i 802.11w. Stöd för PMF är tillgängligt på iPhone 6 och iPad Air 2 och senare.

iOS använder en slumpmässig MAC-adress (Media Access Control) när Wi-Fi-sökning utförs då det inte är kopplat till något Wi-Fi-nätverk. Dessa sökningar kan utföras för att hitta och ansluta till ett förvalt Wi-Fi-nätverk eller för att hjälpa Platstjänster för appar som använder geostängsel, som platsbaserade påminnelser eller åtgärda en plats i Apples Kartor. Observera att Wi-Fi-sökningar som utförs när iOS försöker ansluta till ett förvalt Wi-Fi-nätverk inte är slumpmässiga.

iOS använder också en slumpgenererad MAC-adress vid ePNO-sökningar (enhanced Preferred Network Offload) när en enhet inte är kopplad till ett Wi-Fi-nätverk eller dess processor är i vila. ePNO-sökningar körs när en app på enheten använder platstjänster som drar nytta av så kallade geostängsel, till exempel platsbaserade påminnelser som känner av när enheten är i närheten av en specifik plats.

Eftersom en enhets MAC-adress nu ändras när den kopplas ned från ett Wi-Fi-nätverk kan en passiv observatör inte använda den till att spåra enheten kontinuerligt, även om enheten är ansluten till ett mobilnät. Apple informerar tillverkare av Wi-Fi-utrustning om att iOS Wi-Fi-sökningar använder en slumpgenererad MAC-adress, och att varken Apple eller tillverkarna kan förutsäga dessa slumpgenererade MAC-adresser. Slumpgenererade Wi-Fi-MAC-adresser stöds inte på iPhone 4s eller tidigare.

På iPhone 6s eller senare är den gömda egenskapen för ett känt Wi-Fi-nätverk känd och uppdateras automatiskt. Om SSID (Service Set Identifier) för ett Wi-Fi-nätverk sänds ut skickar iOS-enheten inte ut någon testare med den SSID som är inkluderad i begäran. Detta förhindrar enheten från att sända nätverksnamnet för icke-dolda nätverk.

För att skydda enheten från sårbarheter i nätverksprocessorns fasta programvara har nätverksgränssnitt som Wi-Fi och basband begränsad tillgång till approcessorminne. När USB eller SDIO används som gränssnitt till nätverksprocessorn kan nätverksprocessorn inte inleda DMA-transaktioner (Direct Memory Access) till approcessorn. När PCIe används är varje nätverksprocessor en egen isolerad PCIe-buss. En IOMMU på varje PCIe-buss begränsar nätverksprocessorns DMA-behörighet till minnessidor som innehåller dess nätverkspaket eller kontrollstrukturer.

Bluetooth

Bluetooth-stödet i iOS har utformats för att ge användbar funktionalitet utan att i onödan öka tillgången till personliga data. iOS-enheter stöder anslutningar med Encryption Mode 3, Security Mode 4 och Service Level 1. iOS fungerar med följande Bluetooth-profiler:

- Hands-Free Profile (HFP)
- Phone Book Access Profile (PBAP)
- Message Access Profile (MAP)
- Advanced Audio Distribution Profile (A2DP)

- Audio/Video Remote Control Profile (AVRCP)
- Personal Area Network Profile (PAN)
- Human Interface Device Profile (HID)

Stödet för profilerna varierar beroende på enhet.

Om du vill veta mer går du till:
https://support.apple.com/sv-se/ht3647

Enkel inloggning

iOS stöder autentisering till företagsnätverk genom enkel inloggning (Single sign-on, SSO). SSO fungerar med Kerberos-baserade nätverk och autentiserar användare till tjänster som de har behörighet till. SSO kan användas vid en rad olika nätverksaktiviteter, från säkra Safari-sessioner till tredjepartsappar. Certifikatbaserad autentisering (PKINIT) stöds också.

Vid enkel inloggning i iOS används SPNEGO-tokens och HTTP Negotiate-protokollet i kombination med Kerberos-baserade gateways för autentisering och IWA-system (Integrated Windows Authentication) med stöd för Kerberos-biljetter. SSO-stödet bygger på det öppna källkodsprojektet Heimdal.

Följande krypteringstyper hanteras:

- AES-128-CTS-HMAC-SHA1-96
- AES-256-CTS-HMAC-SHA1-96
- DES3-CBC-SHA1
- ARCFOUR-HMAC-MD5

Safari stöder SSO, och tredjepartsappar som använder standard-API:er för nätverk i iOS kan också konfigureras för det. För konfiguration av SSO har iOS en nyttolast för konfigurationsprofilen som gör det möjligt för MDM-lösningar att bestämma vissa inställningar. Exempel på sådana inställningar är användarnamnet (det vill säga Active Directory-användarkontot) och sfärinställningarna för Kerberos, samt konfiguration av vilka appar och Safari-webbadresser som får använda SSO.

Kontinuitet

Kontinuitetsfunktionen drar nytta av teknik som iCloud, Bluetooth och Wi-Fi och gör det möjligt för användare att påbörja en aktivitet på en enhet och fortsätta den på en annan, ringa och ta emot samtal, skicka och ta emot SMS och dela en mobil internetanslutning.

Handoff

När en användares Mac och iOS-enhet är i närheten av varandra ser Handoff till att användaren automatiskt kan övergå från att arbeta på en enhet till en annan. Med Handoff kan användaren växla mellan enheter och fortsätta jobba direkt.

När en användare loggar in på iCloud på en andra enhet som är Handoff-kompatibel upprättar de två enheterna en direkt parkoppling via Bluetooth LE 4.2 med hjälp av APNs. De enskilda meddelandena krypteras på ett liknande sätt som i iMessage. När enheterna har parkopplats genererar var och en av dem en symmetrisk 256-bitars AES-nyckel som lagras i enhetens nyckelring. Den här nyckeln kan kryptera och autentisera

Bluetooth LE-annonseringar som förmedlar enhetens aktuella aktivitet till andra iCloud-parkopplade enheter via AES-256 i GCM-läge, med skyddsåtgärder mot replay-attacker.

Första gången en enhet tar emot en annonsering från en ny nyckel upprättar den en Bluetooth LE-anslutning till den första enheten och utför en utväxling av krypteringsnycklar. Anslutningen skyddas med standardkryptering för Bluetooth LE 4.2 samt kryptering av de enskilda meddelandena – en metod som påminner om krypteringen i iMessage. I vissa situationer dirigeras dessa meddelanden via APNs istället för Bluetooth LE. Aktivitetens innehåll skyddas och överförs på samma sätt som ett iMessage-meddelande.

Handoff mellan inbyggda appar och webbplatser

Med Handoff kan inbyggda iOS-appar återuppta webbsidor i domäner som styrs och kontrolleras av apputvecklaren. Användarens aktivitet i en inbyggd app kan också återupptas i en webbläsare.

För att förhindra att inbyggda appar återupptar webbplatser som inte kontrolleras av utvecklaren måste appen bevisa att den har legitim kontroll över de webbdomäner den försöker återuppta. Kontroll över en webbplats avgörs via mekanismen för delade webbinloggningsuppgifter. Mer information finns under ”Apptillgång till sparade lösenord” i avsnittet Kryptering och dataskydd i det här dokumentet. Systemet måste kunna bekräfta att appen har kontroll över domännamnet innan appen får godkänna Handoff för användaraktiviteter.

Källan till en Handoff för en webbsida kan vara vilken webbläsare som helst som använder API:erna för Handoff. När användaren öppnar en webbsida tillkännager systemet webbsidans domännamn i form av en krypterad Handoff-annonsering. Det är bara användarens andra enheter som kan avkryptera annonseringen (vilket beskrivits tidigare i det här avsnittet).

På den mottagande enheten upptäcker systemet att en installerad Apple-app godkänner Handoff från det annonserade domännamnet och visar Apple-appens symbol som Handoff-alternativ. När Apple-appen startas tar den emot den fullständiga URL:en och titeln på webbsidan. Ingen annan information överförs från webbläsaren till appen.

I andra riktningen fungerar det så att en Apple-app kan ange en reserv-URL som används om enheten som tar emot Handoff inte har samma app installerad. I detta fall visar systemet användarens förvalda webbläsare som alternativ till Handoff-appen (förutsatt att den webbläsaren använder Handoff-API:er). När en begäran om Handoff görs startas webbläsaren och öppnas med den reserv-URL som källappen har tillhandahållit. Reserv-URL:en behöver inte vara begränsad till de domännamn som kontrolleras av apputvecklaren.

Handoff av större datamängder

Utöver den grundläggande funktionen i Handoff kan vissa appar välja att använda API:er som har stöd för att skicka större mängder data via Apples P2P-teknik för Wi-Fi-anslutning direkt mellan enheter (samma teknik som används av AirDrop). Ett exempel är Mail-appen som använder dessa API:er för Handoff vid överlämning av brevutkast som kan innehålla stora bilagor.

När en app använder den här funktionen startar utbytet mellan de två enheterna precis som i Handoff (se föregående avsnitt). När den mottagande enheten har tagit emot ett första informationsinnehåll via Bluetooth LE upprättar den däremot en ny anslutning via Wi-Fi. Anslutningen krypteras (TLS) och enheterna utbyter iCloud-identitetscertifikat. Identiteten i certifikaten kontrolleras mot användarens identitet. Därefter skickas efterföljande information via denna krypterade anslutning tills överföringen är klar.

Universella urklipp

Universella urklipp drar nytta av Handoff för att säkert överföra innehållet i en användares urklipp mellan enheter så att denne kan kopiera på en enhet och klistra in på en annan. Innehållet skyddas på samma sätt som andra Handoff-data. Om apputvecklaren inte väljer att förhindra delning delas innehållet med universella urklipp som förval.

Appar har tillgång till urklippsdata oavsett om användaren har klistrat in urklipp i appen inte. Med universella urklipp sträcker sig denna datatillgång till appar som körs på användarens andra enheter (som är inloggade på iCloud).

Autoupplåsning

Mac-datorer som har stöd för autoupplåsning använder Bluetooth LE och P2P-Wi-Fi till att tillåta säker upplåsning av Mac-datorn med användarens Apple Watch. Varje kompatibel Mac och Apple Watch som är kopplad till ett iCloud-konto måste ha tvåfaktorsautentisering.

När en Apple Watch aktiveras för upplåsning av en Mac upprättas en säker länk med autoupplåsningsidentiteter. Mac-datorn skapar en slumpmässig engångsupplåsningshemlighet och överför den till Apple Watch via länken. Hemligheten lagras på Apple Watch och går endast att komma åt när Apple Watch är upplåst (se avsnittet Dataskyddsklasser i avsnittet Kryptering och dataskydd). Den nya hemligheten kan inte vara användarens lösenord.

Vid en upplåsning använder Mac-datorn Bluetooth LE till att skapa en anslutning till Apple Watch. En säker länk upprättas sedan mellan de två enheterna med de delade nycklar som användes när den aktiverades första gången. Datorn och Apple Watch använder sedan P2P-Wi-Fi och en säker nyckel härledd från den säkra länken till att avgöra avståndet mellan de två enheterna. Om enheterna är inom räckvidden för varandra används den säkra länken till att överföra den tidigare delade hemligheten för att låsa upp datorn. Efter en lyckad upplåsning ersätter datorn den aktuella upplåsningshemligheten med en ny engångsupplåsningshemlighet och överför den nya upplåsningshemligheten till Apple Watch via länken.

Vidarekoppling av mobilsamtal från iPhone

När en användares Mac, iPad eller iPod touch finns i samma nätverk som dennes iPhone kan den ringa och besvara telefonsamtal via iPhones mobilabonnemang. För att det ska fungera måste alla enheterna vara inloggade på både iCloud och FaceTime med samma Apple-ID-konto.

När ett samtal kommer in meddelas alla konfigurerade enheter via **Apples pushnotistjänst (APNs)**. Alla enskilda notiser använder samma heltäckande kryptering som iMessage använder. Enheter som finns i samma nätverk visar en notis om inkommande samtal i gränssnittet. När användaren besvarar samtalet överförs ljudet direkt från dennes iPhone via en säker P2P-anslutning mellan enheterna.

När ett samtal besvaras på en enhet slutar det ringa på enheter som är iCloud-parkopplade i närheten genom att en kort annonsering visas via Bluetooth LE. Annonseringen krypteras på samma sätt som Handoff-annonseringar.

Utgående samtal vidarekopplas också till iPhone via APNs, och ljudet skickas på liknande sätt via den säkra P2P-länken mellan enheterna.

Användarna kan avaktivera vidarekoppling av samtal på en enhet genom att stänga av iPhone-mobilsamtal i FaceTime-inställningarna.

SMS-koppling från iPhone

Vid SMS-koppling skickas SMS som tas emot på användarens iPhone till användarens registrerade iPad, iPod touch och/eller Mac. Alla enheter måste vara inloggade till iMessage-tjänsten med samma Apple-ID. När SMS-koppling är påslagen registreras enheter inom en användares tillförlitlighetscirkel automatiskt om tvåfaktorsautentisering är aktiverad. I annat fall verifieras registreringen på enskilda enheter genom att skriva in en slumpmässig sexsiffrig kod som genereras av iPhone.

När enheterna är länkade till varandra krypterar och vidarebefordrar iPhone inkommande SMS till varje enhet med hjälp av de metoder som beskrivs under iMessage i det här dokumentet. Svaren skickas tillbaka till iPhone med samma metod, och iPhone skickar sedan svaret i form av ett SMS via operatörens överföringsmetod för SMS. SMS-koppling kan aktiveras och avaktiveras i inställningarna för Meddelanden.

Instant Hotspot

iOS-enheter som har stöd för Instant Hotspot använder Bluetooth LE för att upptäcka och kommunicera med enheter som är inloggade på samma iCloud-konto. Kompatibla Mac-datorer som kör OS X Yosemite eller senare använder samma teknik för att upptäcka och kommunicera med iOS-enheter som använder Instant Hotspot.

När en användare anger Wi-Fi-inställningar på en iOS-enhet sänder enheten ut en Bluetooth LE-annonsering med en identifierare som alla enheter som är inloggade på samma iCloud-konto har kommit överens om. Identifieraren genereras från ett DSID (Destination Signaling Identifier) som är kopplat till iCloud-kontot och byts ut med jämna mellanrum. När andra enheter som är inloggade på samma iCloud-konto finns i närheten och har stöd för Internetdelning känner de av signalen och svarar med att signalera sin tillgänglighet.

När en användare väljer en enhet som är tillgänglig för Internetdelning skickas en förfrågan till enheten om att aktivera Internetdelning. Förfrågan skickas via en länk som använder standardkryptering för Bluetooth LE och förfrågan krypteras på ett sätt som liknar det för iMessage-meddelanden. Enheten svarar sedan genom att skicka anslutningsinformation för Internetdelning via samma Bluetooth LE-länk med samma typ av kryptering för enskilda meddelanden.

AirDrop-säkerhet

iOS-enheter som stöder AirDrop använder Bluetooth LE och Apples egen teknik för P2P-Wi-Fi vid överföring av filer och information mellan enheter som befinner sig i närheten av varandra, vilket omfattar AirDrop-kompatibla Mac-datorer med OS X 10.11 eller senare. Enheterna kommunicerar direkt med varandra genom att sända och ta emot Wi-Fi-signaler, utan att gå via internet eller någon Wi-Fi-anslutningspunkt.

När en användare aktiverar AirDrop sparas en 2048-bitars RSA-identitet på enheten. Dessutom skapas en hashkod för AirDrop-identitet baserat på de e-postadresser och telefonnummer som är kopplade till användarens Apple-ID.

När en användare väljer AirDrop som metod för delning av ett objekt sänder enheten en AirDrop-signal via Bluetooth LE. Andra enheter som är aktiverade, befinner sig i närheten och har AirDrop på tar emot signalen och svarar med en förkortad version av sin ägares identitetshashkod.

AirDrop är inställt på att endast dela med kontakter som förval. Användaren kan också välja att låta AirDrop dela med alla, eller helt stänga av funktionen. I läget Endast kontakter jämförs de mottagna identitetshashkoderna med hashkoderna för personer som finns bland sändarens kontakter. Om de matchar varandra upprättar den sändande enheten ett P2P-nätverk via Wi-Fi och tillkännager en AirDrop-anslutning med hjälp av Bonjour. Via den här anslutningen skickar de mottagande enheterna sina fullständiga identitetshashkoder till sändaren. Om även den fullständiga hashkoden matchar informationen i Kontakter visas mottagarens namn och bild (om den finns i Kontakter) på AirDrop-delningsbladet.

När AirDrop används är det sändaren som väljer vem han/hon vill dela med. Den sändande enheten upprättar en krypterad (TLS) anslutning med den mottagande enheten, och enheterna utbyter iCloud-identitetscertifikat. Identiteten i certifikaten kontrolleras mot varje användares kontakter (i appen Kontakter). Den mottagande användaren får en förfrågan om att ta emot en inkommande överföring från den identifierade personen eller enheten. Om flera mottagare har valts upprepas processen för var och en av dem.

I läget Alla används samma process, men om ingen träff hittas i Kontakter visas de mottagande enheterna på AirDrop-sändningsbladet med en silhuett och med enhetens namn, så som det angetts i Inställningar > Allmänt > Om > Namn.

Organisationer kan begränsa användningen av AirDrop för enheter eller appar som hanteras med en MDM-lösning.

Wi-Fi-lösenordsdelning

iOS-enheter som stöder Wi-Fi-lösenordsdelning använder en mekanism som liknar AirDrop till att skicka ett Wi-Fi-lösenord mellan två enheter.

När en användare väljer ett Wi-Fi-nätverk (sökare) och blir ombedd att ange Wi-Fi-lösenordet startar Apple-enheten en Bluetooth LE-annonsering som talar om att den vill ha Wi-Fi-lösenordet. Andra Apple-enheter som är vakna, i närheten och har lösenordet för det valda Wi-Fi-nätverket ansluter via Bluetooth LE till enheten som behöver lösenordet.

Enheten som har Wi-Fi-lösenordet (givare) kräver sökarens kontakt-information, och sökaren måste bevisa sin identitet med en mekanism som påminner om AirDrop. När identiteten är bekräftad skickar givaren en 64-teckens PSK som även kan användas till att ansluta till nätverket till sökaren.

Organisationer kan begränsa användningen av Wi-Fi-lösenordsdelning för enheter eller appar som hanteras med en MDM-lösning.

Apple Pay

Med Apple Pay går det att använda iOS-enheter som stöds, Apple Watch och Mac till att betala på ett enkelt, säkert och privat sätt i butiker, inuti appar och på webben i Safari. Användare kan också lägga till Apple Pay-kompatibla kollektivtrafikkort i Wallet. Det är enkelt för användarna och bygger på integrerad säkerhet i både maskin- och programvara.

Apple Pay är dessutom utformat för att skydda användarens personliga information. Apple Pay samlar inte in någon information om transaktioner som kan knytas till användaren. Alla transaktioner vid betalning sker mellan användaren, säljaren och kortutfärdaren.

Apple Pay-komponenter

Secure Element: Secure Element är en branschstandard för en certifierad krets som kör Java Card, en plattform som uppfyller finanssektorns krav för elektroniska betalningar.

NFC-styrenhet: NFC-styrenheten hanterar NFC-protokoll (Near Field Communication) och dirigerar kommunikation mellan approcessorn och Secure Element, samt mellan Secure Element och kassaterminalen.

Wallet: Wallet används till att lägga till och hantera kredit-, bank- och butikskort och till att betala med Apple Pay. Användarna kan se sina kort och eventuellt ytterligare information som tillhandahålls av kortutfärdaren, exempelvis kortutfärdarens integritetspolicy, de senaste transaktionerna med mera i Wallet. Användarna kan också lägga till kort för Apple Pay i:

- Inställningsassistenten och Inställningar för iOS
- Apple Watch-appen för Apple Watch
- Systeminställningspanelen Wallet och Apple Pay för Mac.

Dessutom kan användare lägga till och hantera kollektivtrafikkort, bonuskort, boardingkort, biljetter, presentkort, studentkort och annat i Wallet.

Secure Enclave: På iPhone, iPad och Apple Watch hanterar Secure Enclave autentiseringsprocessen och gör det möjligt att genomföra en betalningstransaktion.

På Apple Watch måste enheten låsas upp och användaren måste dubbelklicka på sidoknappen. Dubbelklicket identifieras och vidarebefordras direkt till Secure Element, eller Secure Enclave där det är tillgängligt, direkt utan att gå via approcessorn.

Apple Pay-servrar: Apple Pay-servrarna hanterar inställning av och tillhandahåller kredit-, bank- och kollektrafikkort samt studentlegitimationer i Wallet och enhetens kontonummer lagras i Secure Element. De kommunicerar både med enheten och med betalningsnätverkets eller kortutfärdarens servrar. Apple Pay-servrarna ansvarar också för att omkryptera betalningsuppgifterna vid betalningar inuti appar.

Hur Apple Pay använder Secure Element

Secure Element innehåller en specialutformad applet som hanterar Apple Pay. Det innehåller även appletar som är certifierade av betalningsnätverk eller kortutfärdare. Kredit- och bankkortsdata eller data från förbetalda kort skickas från betalningsnätverket eller kortutfärdaren till dessa appletar med nycklar som bara är kända av betalningsnätverket eller kortutfärdaren och appletens säkerhetsdomän. Dessa data lagras inom appletarna och skyddas med hjälp av säkerhetsfunktionerna i Secure Element. Under en transaktion kommunicerar terminalen direkt med Secure Element genom NFC-styrenheten (Near Field Communication) via en särskild maskinvarubuss.

Hur Apple Pay använder NFC-styrenheten

NFC-styrenheten fungerar som en gateway till Secure Element och ser till att alla kontaktfria betalningstransaktioner utförs med en kassaterminal som befinner sig nära enheten. Endast betalningsförfrågningar som kommer från en terminal som befinner sig inom fältet markeras av NFC-styrenheten som kontaktfria transaktioner.

När en betalning med ett kredit-, bank- eller förbetalt kort (inklusive butikskort) har auktoriserats av kortinnehavaren med Touch ID, Face ID eller lösenkod, eller på en upplåst Apple Watch genom att dubbelklicka på sidoknappen, dirigeras de kontaktlösa svaren som har förberetts av betalningsappletarna i Secure Element av styrenheten enbart till NFC-fältet. Det innebär att betalningsinformation för kontaktlösa betalningstransaktioner hålls inom det lokala NFC-fältet och aldrig exponeras för approcessorn. Betalningsinformation vid köp inuti appar och på webben dirigeras däremot till approcessorn, men skickas inte till Apple Pay-servern förrän den har krypterats av Secure Element.

Tillägg av kreditkort, bankkort och förbetalda kort

När en användare lägger till ett kredit- eller bankkort eller ett förbetalt kort (även butikskort) i Wallet skickar Apple kortinformationen tillsammans med annan information om användarens konto och enhet till kortutfärdaren, eller kortutfärdarens auktoriserade tjänsteleverantör. Med hjälp av den här informationen avgör kortutfärdaren om den ska godkänna att kortet läggs till i Wallet.

Apple Pay använder tre serverstyrda anrop till att skicka och ta emot information från kortutfärdaren eller nätverket som en del av kortbetalningsprocessen: *Required Fields, Check Card* och *Link and Provision*. Kortutfärdaren använder de här anropen till att verifiera, godkänna och lägga till kort i Wallet. Dessa klient-server-sessioner krypteras med TLS v1.2.

De fullständiga kortnumren lagras varken på enheten eller på Apples servrar. Istället skapas ett unikt kontonummer för enheten, som sedan krypteras och sparas i Secure Element. Enhetens unika kontonummer krypteras på ett sådant sätt att Apple inte har någon tillgång till det. Enhetens kontonummer är unikt och skiljer sig från vanliga kredit- och bankkortsnummer. Kortutfärdaren eller betalningsnätverket kan förhindra att det används vid betalning med magnetremsa, per telefon eller på webbplatser. Enhetens kontonummer finns i Secure Element och är isolerat från iOS och watchOS. Det lagras aldrig på Apples servrar och säkerhetskopieras aldrig på iCloud.

Kort som ska användas med Apple Watch tillhandahålls för Apple Pay med Apple Watch-appen på iPhone eller i en kortutfärdares iPhone-app. När du lägger till ett kort på Apple Watch måste klockan vara inom Bluetooth-räckvidden. Kort registreras specifikt för att användas med Apple Watch och har egna enhetskontonummer som lagras i Secure Element på Apple Watch.

När kredit-, bank- eller förbetalda kort (inklusive butikskort) läggs till visas de i en lista med kort medan inställningsassistenten körs på enheter som är inloggade med samma iCloud-konto. De här korten finns kvar i den här listan så länge de är aktiva på minst en enhet. Kort tas bort från den här listan efter att de har varit borttagna från alla enheter under sju dagar. Den här funktionen kräver att tvåfaktorsautentisering är aktiverad på respektive iCloud-konto.

Lägga till kredit- eller bankkort manuellt i Apple Pay

När du lägger till ett kort manuellt används namnet, kortnumret, sista giltighetsdatumet och CVV-koden för att förenkla tillhandahållandet. Användarna kan ange denna information i Inställningar, appen Wallet eller Apple Watch-appen genom att skriva eller genom att använda enhetens kamera. När kameran tar en bild av kortinformationen försöker Apple att fylla i namnet, kortnumret och sista giltighetsdatum. Bilden sparas aldrig på enheten eller lagras i bildbiblioteket. När alla fält är ifyllda verifieras alla fält utom CVV-fältet av processen Check Card. De krypteras och skickas till Apple Pay-servern.

Om ett villkors-ID returneras med Check Card-processen hämtar och visar Apple kortutfärdarens villkor för användaren. Om användaren godkänner villkoren skickar Apple ID:t för de godkända villkoren samt CVV-koden till processen Link and Provision. Under Link and Provision-processen delar Apple dessutom information från enheten med kortutfärdaren eller nätverket. Det kan röra sig om information om din kontoaktivitet i iTunes och App Store (t.ex. om du har genomfört transaktioner under lång tid i iTunes), information om din enhet (t.ex. telefonnummer, namn och enhetens modell, plus eventuella andra iOS-enheter som krävs för att konfigurera Apple Pay) samt ungefär var du befinner dig vid den tidpunkt då du lägger till kortet (om Platstjänster är aktiverat). Med hjälp av den här informationen avgör kortutfärdaren om den ska godkänna att kortet läggs till i Apple Pay.

Resultatet av processen Link and Provision är att två saker sker:

- Enheten börjar hämta den Wallet-passfil som representerar kredit- eller bankkortet.
- Enheten börjar koppla kortet till Secure Element.

Passfilen innehåller URL:er för hämtning av kortbilder, metadata om kortet som kontaktinformation, den relaterade utfärdarens app och funktioner som stöds. Den innehåller också passets status, det vill säga information om ifall anpassningen av Secure Element är klar, om kortet för närvarande är spärrat av kortutfärdaren samt om ytterligare verifiering krävs innan kortet kan användas för betalningar med Apple Pay.

Lägga till kredit- eller bankkort som har sparats i ett iTunes Store-konto i Apple Pay

Om användaren har ett kredit- eller bankkort som har sparats i iTunes kan han/hon behöva ange sitt Apple-ID-lösenord igen. Kortnumret hämtas från iTunes och processen Check Card startas. Om kortet godkänns för användning med Apple Pay hämtar och visar enheten villkor och skickar

sedan ID:t för villkoren tillsammans med kortets säkerhetskod till Link and Provision-processen. Ytterligare verifiering kan krävas för kort som sparats i ett iTunes-konto.

Lägga till kredit- eller bankkort från en kortutfärdares app

När appen registreras för användning med Apple Pay skapas nycklar för appen och kortutfärdarens server. Dessa nycklar används till att kryptera kortinformationen som skickas till kortutfärdaren, vilket förhindrar att information läses av iOS-enheten. Flödet liknar det som används för manuellt tillagda kort (vilket beskrivits tidigare), förutom att engångslösenord används i stället för CVV-numret.

Ytterligare verifiering

En kortutfärdare kan bestämma om ett kredit- eller bankkort behöver verifieras ytterligare. Beroende på vad kortutfärdaren erbjuder kan användaren ha möjlighet att välja mellan olika alternativ för ytterligare verifiering, till exempel SMS, e-post, samtal med kundtjänsten eller en metod i någon tredjepartsapp för att slutföra verifieringen. När det gäller SMS eller e-post kan användaren välja bland den kontaktinformation som finns registrerad hos utfärdaren. En kod skickas och måste anges i Wallet, Inställningar eller Apple Watch-appen. Utfärdaren hanterar kommunikationen vid samtal med kundtjänsten eller vid verifiering med hjälp av en app.

Betalningsauktorisering

På enheter med Secure Enclave tillåter Secure Element att en betalning genomförs endast efter att den har mottagit auktorisering från Secure Enclave. På iPhone och iPad omfattar detta att bekräfta att användaren har bevisat sin identitet med Touch ID, Face ID eller enhetslösenkoden. Touch ID eller Face ID är den förvalda metoden om den är tillgänglig, men lösenkoden kan alltid användas. Verifiering med lösenkod erbjuds automatiskt efter tre misslyckade försök att matcha ett fingeravtryck eller två misslyckade försök att matcha ett ansikte. Efter fem misslyckade försök måste lösenkoden anges. En lösenkod krävs också om Touch ID eller Face ID inte är konfigurerat eller aktiverat för Apple Pay. På Apple Watch måste enheten låsas upp med lösenkoden och användaren måste dubbelklicka på knappen på sidan för att en betalning ska genomföras.

Kommunikation mellan Secure Enclave och Secure Element sker via ett seriellt gränssnitt där Secure Element är anslutet till NFC-styrenheten, vilken i sin tur är ansluten till approcessorn. Trots att de inte är direkt anslutna till varandra kan Secure Enclave och Secure Element kommunicera säkert med en delad parkopplingsnyckel som tillhandahålls under tillverkningsprocessen. Krypteringen och autentiseringen av kommunikationen baseras på AES, med kryptografiska nonce-värden som används av bägge sidor för att skydda mot replay-attacker. Parkopplingsnyckeln genereras inuti Secure Enclave av dess UID-nyckel och Secure Elements unika identifierare. Parkopplingsnyckeln överförs sedan säkert från Secure Enclave till en **HSM-modul (Hardware Security Module)** i fabriken som har det nyckelmaterial som krävs för att infoga parkopplingsnyckeln i Secure Element.

När användaren auktoriserar en transaktion skickar Secure Enclave signerade data om typen av autentisering och detaljer om typen av transaktion (kontaktfri eller inuti appar) till Secure Element, kopplad till ett AR-värde (Authorization Random). AR-värdet genereras i Secure Enclave när en användare först lägger till ett kreditkort och sparas så länge

Apple Pay är aktiverat samt skyddas av Secure Enclaves kryptering och en bakåtspärr. Den levereras på ett säkert sätt till Secure Element med hjälp av parkopplingsnyckeln. När Secure Element tar emot ett nytt AR-värde markeras alla tidigare tillagda kort som raderade.

Kredit- och bankkort och förbetalda kort som lagts till i Secure Element kan bara användas om Secure Element tar emot en auktorisering med samma parkopplingsnyckel och AR-värde som när kortet lades till. Detta innebär att iOS kan instruera Secure Enclave att göra korten oanvändbara genom att markera deras kopia av AR-värdet som ogiltig under följande omständigheter:

- När lösenkoden avaktiveras.
- Användaren loggar ut från iCloud.
- Användaren väljer Radera allt innehåll och inst.
- Enheten återskapas från återhämtningsläget.

Med Apple Watch markeras kort som ogiltiga när:

- Klockans lösenkod avaktiveras.
- Klockans parkoppling till iPhone tas bort.

Med hjälp av parkopplingsnyckeln och dess kopia av det aktuella AR-värdet verifierar Secure Element auktoriseringen som mottagits från Secure Enclave innan den aktiverar betalningsappleten för en kontaktfri betalning. Den här processen gäller också vid hämtning av krypterade betalningsdata från en betalningsapplet för transaktioner inom appar.

Transaktionsspecifik dynamisk säkerhetskod

Betalningstransaktioner som har sitt ursprung i betalningsappletar innehåller ett betalningskryptogram tillsammans med ett kontonummer för enheten. Det här kryptogrammet är en engångskod som beräknas med en transaktionsräknare som ökar i värde för varje ny transaktion och en nyckel som tillhandahålls i betalningsappleten under anpassningen och är känd av betalningsnätverket och/eller kortutfärdaren. Beroende på betalningsschemat kan också andra data användas vid beräkningen, däribland följande:

- Ett terminalslumptal vid en NFC-transaktion.
- Ett nonce-värde för Apple Pay-servern vid transaktioner inuti appar.

Dessa säkerhetskoder skickas till betalningsnätverket och kortutfärdaren så att de kan verifiera alla transaktioner. Längden på dessa säkerhetskoder kan variera beroende på vilken typ av transaktion det rör sig om.

Betala med kredit- och bankkort i butiker

Om iPhone är på och den upptäcker ett NFC-fält visar den det begärda kortet för användaren (om automatiskt val är aktiverat för det kortet) eller det förvalda kortet som hanteras i Inställningar. Användaren kan även öppna appen Wallet och välja ett kort, eller göra följande när enheten är låst:

- Dubbelklicka på hemknappen på enheter med Touch ID.
- Dubbelklicka på sidoknappen på enheter med Face ID.

Sedan måste användaren autentisera med Touch ID, Face ID eller lösenkoden innan betalningsinformationen skickas. När Apple Watch är upplåst aktiveras det förvalda kortet för betalning när sidknappen dubbelklickas. Ingen betalningsinformation skickas utan användarens autentisering.

När användaren har autentiserat transaktionen används enhetens unika kontonummer och en transaktionsspecifik dynamisk säkerhetskod för att behandla betalningen. Varken Apple eller användarens enhet skickar det fullständiga kredit- eller bankkortsnumret till säljaren. Apple kan ta emot anonym transaktionsinformation, som ungefärlig tid och plats för transaktionen, vilket hjälper till att förbättra Apple Pay och andra Apple-produkter och -tjänster.

Betala med kredit- och bankkort inuti appar

Apple Pay kan även användas till att utföra betalningar inuti iOS-appar och Apple Watch-appar. När användaren betalar inuti en app med Apple Pay tar Apple emot krypterad transaktionsinformation och krypterar om den med utvecklarens specifika nyckel innan den skickas till utvecklaren eller säljaren. Apple Pay sparar anonym transaktionsinformation, t.ex. det ungefärliga beloppet. Den här informationen kan inte kopplas till användaren och talar aldrig om vad användaren köper.

När en app initierar en Apple Pay-betalning mottar Apple Pay-servrarna den krypterade transaktionen från enheten innan säljaren tar emot den. Apple Pay-servrarna krypterar sedan om den med säljarens specifika nyckel innan transaktionen skickas vidare till säljaren.

När appen begär en betalning anropar den ett API för att avgöra om enheten stöder Apple Pay och om användaren har ett kredit- eller bankkort som kan utföra betalningar i det betalningsnätverk som säljaren använder. Appen frågar efter den information den behöver för att behandla och fullfölja transaktionen, till exempel fakturerings- och leveransadressen samt kontaktinformation. Appen ber sedan iOS att visa Apple Pay-bladet som begär information åt appen, liksom annan nödvändig information, t.ex. vilket kort som ska användas.

Appen förses nu med information om ort och postnummer för slutlig beräkning av fraktkostnaden. All den information som begärts lämnas inte ut till appen förrän användaren godkänner betalningen med hjälp av Touch ID, Face ID eller enhetens lösenkod. När betalningen auktoriseras överförs informationen som visas på Apple Pay-bladet till säljaren.

När användaren godkänner en betalning skickas ett anrop till Apple Pay-servrarna för att erhålla ett kryptografiskt nonce-värde, vilket liknar det värde som returneras av NFC-terminalen vid transaktioner i butik. Nonce-värdet och andra transaktionsdata överförs till Secure Element som genererar betalningsuppgifter som krypteras med en Apple-nyckel. Från Secure Element skickas de krypterade betalningsuppgifterna vidare till Apple Pay-servrarna som avkrypterar dem, jämför nonce-värdet i uppgifterna med nonce-värdet som ursprungligen skickades från Apple Pay-servrar och krypterar om betalningsuppgifterna med säljarens nyckel som är kopplad till säljarens ID. De skickas sedan till enheten, som returnerar dem till appen via API:t. Appen skickar dem sedan vidare till säljarens system för bearbetning. Säljaren kan sedan avkryptera betalningsuppgifterna med sin privata nyckel för bearbetning. Detta, tillsammans med signaturen från Apples servrar, gör att säljaren kan verifiera att transaktionen var avsedd för just honom/henne.

API:erna kräver en rättighet som anger säljarens ID-uppgifter. En app kan också bifoga ytterligare data som skickas till Secure Element för signering, till exempel ett ordernummer eller kundens identitet, så att transaktionen inte kan dirigeras om till en annan kund. Det här åstadkoms av apputvecklaren som kan ange applicationData i PKPaymentRequest. En hashkod för dessa data inkluderas i den krypterade betalningsinformationen. Säljaren är sedan ansvarig för att verifiera att hans/hennes hashkod för applicationData matchar det som ingår i betalningsinformationen.

Betala med kredit- och bankkort på webben

Apple Pay kan användas till att utföra betalningar på webbplatser med iOS-enheter, Apple Watch och Mac. Apple Pay-transaktioner kan även inledas på en Mac och sedan slutföras på en Apple Pay-aktiverad iPhone eller Apple Watch som använder samma iCloud-konto.

Apple Pay på webben kräver att alla deltagande webbplatser registreras hos Apple. Apples servrar utför domännamnsvalidering och utfärdar ett TLS-klientcertifikat. Webbplatser som stöder Apple Pay måste visa sitt innehåll via HTTPS. För varje betalningstransaktion måste webbplatserna erhålla en säker och unik säljarsession via en Apple-server genom att använda det TLS-klientcertifikat som Apple har utfärdat. Säljarsessionsdata signeras av Apple. När en säljarsessionssignatur är verifierad kan webbplatsen skicka en förfrågan om användaren har en enhet förberedd för Apple Pay och om ett kredit- eller kontokort eller ett förbetalt kort är aktiverat på enheten. Inga andra detaljer delas. Om användaren inte vill dela den här informationen kan denna avaktivera Apple Pay-förfrågningar i integritetsinställningarna för Safari i både iOS och macOS.

När en säljarsession har validerats är alla säkerhets- och integritetsåtgärder samma som när en användare betalar inuti en app.

När det gäller överlämning från Mac till iPhone eller Apple Watch använder Apple Pay det E2E-krypterade IDS-protokollet (Apple Identity Service) till att överföra betalningsrelaterad information mellan användarens dator och den auktoriserande enheten. IDS använder användarens enhetsnycklar till att utföra krypteringen så att inga andra enheter kan avkryptera informationen, och nycklarna är inte tillgängliga för Apple. Enhetsupptäckt för Apple Pay-överlämning innehåller typ och unik identifierare för användarens kreditkort tillsammans med vissa metadata. Det enhetsspecifka kontonumret för användarens kort delas inte, och det fortsätter att lagras säkert på användarens iPhone eller Apple Watch. Apple överför också användarens senast använda kontaktuppgifter och leverans- och faktureringsadress säkert via iCloud-nyckelring.

När användaren har auktoriserat betalningen via Touch ID, Face ID, lösenkod eller genom att dubbelklicka på sidoknappen på Apple Watch, blir en betalningstoken som är unikt krypterad säkert överförd till varje webbplats säljcertifikat från användarens iPhone eller Apple Watch till datorn och levereras sedan till säljarens webbplats.

Endast enheter som finns i närheten av varandra kan begära och slutföra betalningar. Närheten bestäms genom Bluetooth LE-annonsering.

Kontaktlösa kuponger

Wallet stöder VAS-protokollet (Value Added Service) för överföring av data från kuponger som stöds till kompatibla NFC-terminaler. VAS-protokollet kan implementeras på kontaktlösa terminaler. NFC används för kommunikationen med de Apple-enheter som stöds. VAS-protokollet fungerar på korta avstånd och kan användas till att lösa in kontaktlösa kuponger fristående eller som en del av en Apple Pay-transaktion.

När enheten hålls nära NFC-terminalen inleder terminalen överföringen av kuponginformationen genom att skicka en kupongförfrågan. Om användaren har en kupong med butikens ID blir användaren ombedd att bekräfta att den ska användas med Touch ID, Face ID eller lösenkod. Kuponginformationen, en tidsstämpel och en slumpmässig ECDH P-256-engångsnyckel används med säljarens offentliga nyckel för att härleda en krypteringsnyckel för kupongens data som sedan skickas till terminalen.

Användare kan också välja en kupong manuellt och autentisera den med Touch ID, Face ID eller lösenkod innan den hålls mot säljarens NFC-terminal.

Apple Pay Cash

I iOS 11.2 eller senare och watchOS 4.2 eller senare kan Apple Pay användas på en iPhone, iPad eller Apple Watch till att skicka, ta emot och begära pengar från andra användare. När en användare tar emot pengar läggs de till i ett Apple Pay Cash-konto som kan nås i Wallet eller i Inställningar > Wallet och Apple Pay på valfri kompatibel enhet där användaren har loggat in med sitt Apple-ID.

För att göra betalningar mellan personer och använda Apple Pay Cash måste användaren vara inloggad på sitt iCloud-konto på en Apple Pay Cash-kompatibel enhet och ha ställt in tvåfaktorsautentisering för iCloud-kontot.

När du ställer in Apple Pay Cash kan samma information som när du lägger till ett kredit- eller bankkort delas med vår partnerbank Green Dot Bank och med Apple Payments Inc. som är ett helägt dotterbolag som skapades för att skydda din integritet genom att lagra och bearbeta information separat från övriga Apple och på ett sätt som övriga Apple inte känner till. Den här informationen används endast vid felsökning samt i syfte att förhindra bedrägerier och uppfylla lagstadgade krav.

Förfrågningar om pengar och överföringar mellan användare inleds i appen Meddelanden eller genom att fråga Siri. När en användare försöker skicka pengar visas Apple Pay-bladet av iMessage. Apple Pay Cash-saldot används alltid först. Om det behövs dras resterande belopp från ett annat kredit- eller bankkort som användaren har lagt till i Wallet.

Apple Pay Cash-kortet i Wallet kan användas med Apple Pay till att utföra betalningar i butiker, i appar och på webben. Pengarna på Apple Pay Cash-kontot kan även överföras till ett bankkonto. Utöver att ta emot pengar från en annan användare kan pengar läggas till på Apple Pay Cash-kontot från ett bank- eller förbetalt kort i Wallet.

Apple Payments Inc. lagrar och kan använda din transaktionsinformation vid felsökning samt i syfte att förhindra bedrägerier och uppfylla lagstadgade krav när en transaktion har genomförts. Övriga Apple vet inte vem eller vilka du skickar pengar till, tar emot pengar från eller var du har betalat med ditt Apple Pay Cash-kort.

När du skickar pengar med Apple Pay, lägger till pengar på ett Apple Pay Cash-konto eller överför pengar till ett bankkonto görs ett anrop till Apple Pay-servrarna för att få ett kryptografiskt nonce-värde som liknar värdet som returneras för Apple Pay inuti appar. Nonce-värdet och andra transaktionsdata överförs till Secure Element som genererar en betalningssignatur. När betalningssignaturen kommer ut från Secure Element överförs den till Apple Pay-servrarna. Transaktionens autenticitet, integritet och riktighet verifieras av Apple Pay-servrarna via betalningssignaturen och nonce-värdet. Därefter inleds pengaöverföringen och du får ett meddelande om den slutförda transaktionen.

Om transaktionen omfattar ett kredit- eller bankkort för att lägga till pengar på Apple Pay Cash-kontot, skicka pengar till en annan användare, eller skjuta till extra pengar om saldot på Apple Pay Cash-kontot är lågt, genereras även krypterade betalningsuppgifter som skickas till Apple Pay-servrarna. Det här liknar processen som används för Apple Pay inuti appar och på webbplatser.

När saldot på Apple Pay Cash-kontot överskrider ett visst belopp, eller om ovanlig aktivitet upptäcks, uppmanas användaren att verifiera sin identitet. Informationen som används till att verifiera användarens identitet, t.ex. personnummer eller svar på frågor (exempelvis namnet på en gata du har bott på) skickas säkert till Apples partner och krypteras med dennes nyckel. Apple kan inte avkryptera den här informationen.

Kollektivtrafikkort

I Kina och Japan kan användare lägga till kollektivtrafikkort som stöds i Wallet på kompatibla iPhone- och Apple Watch-modeller. Det görs genom att antingen överföra saldot och pendlingsbiljetten från ett fysiskt kort till dess digitala representation i Wallet, eller genom att lägga till ett nytt kollektivtrafikkort i Wallet från kortutfärdarens app. När kollektivtrafikkort läggs till i Wallet kan användare åka kollektivt genom att helt enkelt hålla iPhone eller Apple Watch nära kollektivtrafikkortsläsaren. I Japan kan Suica-kortet även användas till att betala med.

Tillagda kollektivtrafikkort associeras med en användares iCloud-konto. Om användaren lägger till fler än ett kort i Wallet kanske Apple eller kortutfärdaren kan länka användarens personliga uppgifter och den associerade kontoinformationen mellan korten. T.ex. kan MySuica-kort länkas till anonyma Suica-kort. Kollektivtrafikkort och transaktioner skyddas av en uppsättning hierarkiska kryptografiska nycklar.

När saldot från ett fysiskt kort överförs till Wallet måste användaren ange ID-siffror i kortets serienummer. Användare kan även behöva ange personuppgifter som bevis för kortinnehavet. Om kortet exempelvis är ett MySuica-kort eller ett Suica-kort som innehåller en pendlingsbiljett måste användaren också ange sin födelsedag. När biljetter överförs från iPhone till Apple Watch måste båda enheterna vara uppkopplade under överföringen.

Saldot kan fyllas på med pengar från kredit- och förbetalda kort via Wallet eller från kortutfärdarens app. Säkerheten vid påfyllning av saldot när Apple Pay används beskrivs i avsnittet Betala med kredit- och bankkort inuti appar i det här dokumentet.

Processen att tillhandahålla kollektivtrafikkortet från kortutfärdarens app beskrivs i avsnittet Lägga till kredit- eller bankkort från en kortutfärdares app i det här dokumentet.

Utfärdaren av kollektivtrafikkortet har de kryptografiska nycklar som behövs för att autentisera det fysiska kortet och verifiera användarens angivna data. När kortet har verifierats kan systemet skapa ett enhetskontonummer för Secure Element och aktivera den tillagda biljetten i Wallet tillsammans med det överförda saldot. I Japan avaktiveras det fysiska Suica-kortet när överföringsprocessen från det fysiska kortet är klar.

Vid slutet av båda processerna krypteras saldot på kollektivtrafikkortet och lagras i en utsedd applet i Secure Element. Kollektivtrafikföretaget har nycklarna som krävs för att utföra kryptografiska åtgärder gällande kortdata för saldotransaktioner.

Som förval drar användare nytta av expresskollektrafikupplevelsen som innebär att de kan betala och åka utan att använda Touch ID, Face ID eller en lösenkod. Information som senast besökta stationer, transaktionshistorik och ytterligare biljetter kan kontrolleras via en kontaktlös kortläsare i närheten med Expressläge aktiverat. Användare kan aktivera auktoriseringskrav för Touch ID, Face ID eller lösenkod i Wallet och Apple Pay-inställningarna genom att avaktivera Expresskollektivtrafik.

I likhet med andra Apple Pay-kort kan användare stänga av eller ta bort kollektivtrafikkort genom att:

- Fjärradera enheten med Hitta min iPhone.
- Aktivera förlorat läge med Hitta min iPhone.
- Fjärradera med ett MDM-kommando.
- Ta bort alla kort från deras Apple-ID-kontosida.
- Ta bort alla kort från iCloud.com.
- Ta bort alla kort från Wallet.
- Ta bort kortet i utfärdarens app.

Apple Pay-servrar meddelar kollektivtrafikföretaget om att stänga av eller avaktivera dessa kort. Om användare med Suica-kort har enheter som är nedkopplade när de försöker att radera korten kan deras Suica-kort fortfarande fungera vid en del terminaler tills 00.01 japansk tid påföljande dag. Om användarnas enheter är nedkopplade fortsätter kollektivtrafikkort i Kina att fungera.

Om användare tar bort sina kollektivtrafikkort kan de återvinna saldot genom att lägga tillbaka dem på en enhet som är inloggad med samma Apple-ID.

Studentkort

I iOS 12 kan studenter, lärare och personal på campus som deltar lägga till sina student- och personalkort i Wallet så att de kan öppna dörrar och betala överallt där kortet gäller.

En användare lägger till sitt kort i Wallet via en app som tillhandahålls av kortutfärdaren eller skolan som deltar. Den tekniska processen som används är samma som den som beskrivs i avsnittet Lägga till kredit- eller bankkort från en kortutfärdares app tidigare i det här dokumentet. Utfärdande appar måste dessutom stöda tvåfaktorsautentisering för de konton som bevakar tillgången till deras kort. Ett kort kan ställas in samtidigt på upp till två valfria kompatibla Apple-enheter som är inloggade med samma Apple-ID.

När ett studentkort läggs till i Wallet aktiveras expressläget som förval. Studentkort i expressläge interagerar med kompatibla terminaler utan autentisering med Touch ID, Face ID eller lösenkod. Användaren kan trycka på merknappen på framsidan av studentkortet i Wallet och stänga av expressläget för att avaktivera den här funktionen. Touch ID, Face ID eller lösenkod krävs för att återaktivera expressläget.

Studentkort kan avaktiveras eller tas bort genom att:

- Fjärradera enheten med Hitta min iPhone.
- Aktivera förlorat läge med Hitta min iPhone.
- Fjärradera med ett MDM-kommando.
- Ta bort alla kort från deras Apple-ID-kontosida.
- Ta bort alla kort från iCloud.com.
- Ta bort alla kort från Wallet.
- Ta bort kortet i utfärdarens app.

Stänga av, ta bort och radera kort

Användaren kan stänga av Apple Pay på iPhone, iPad och Apple Watch genom att försätta enheten i förlorat läge i Hitta min iPhone. Användaren har också möjlighet att ta bort och radera sina kort från Apple Pay via Hitta min iPhone, på iCloud.com eller direkt på enheten i Wallet. Kort kan tas bort i iCloud-inställningarna på Apple Watch, i Apple Watch-appen på iPhone eller direkt på klockan. Möjligheten att betala med kort från enheten stängs av eller tas bort från Apple Pay av kortutfärdaren eller respektive betalningsnätverk, även om enheten är offline och inte ansluten till ett mobilnät eller ett Wi-Fi-nätverk. Användaren kan även ringa kortutfärdaren för att stänga av eller ta bort kort från Apple Pay.

När en användare raderar hela enheten med ”Radera allt innehåll och inst.” i Hitta min iPhone, eller återskapar enheten i återhämtningsläget, ser iOS dessutom till att Secure Element markerar alla kort som raderade. Effekten av detta är att korten omedelbart blir oanvändbara, i väntan på att Apple Pay-servrarna kan kontaktas och helt radera korten från Secure Element. Oberoende av detta markerar Secure Element AR-värdet som ogiltigt så att inga betalningsgodkännanden för tidigare registrerade kort längre är möjliga. När enheten är online försöker den kontakta Apple Pay-servrarna för att försäkra sig om att alla kort i Secure Element har raderats.

Internettjänster

Skapa starka Apple-ID-lösenord
Ett Apple-ID används för att ansluta till olika tjänster, som iCloud, FaceTime och iMessage. För att hjälpa användarna att skapa starka lösenord måste alla nya konton innehålla följande lösenordsattribut:

- Minst åtta tecken långt
- Minst en bokstav
- Minst en stor bokstav
- Minst en siffra
- Högst tre på varandra följande likadana tecken
- Inte identiskt med ditt kontonamn

Apple har byggt upp en omfattande samling tjänster som hjälper användarna att få ut mer av sina enheter och göra dem ännu mer produktiva, till exempel iMessage, FaceTime, Siri-förslag, iCloud, iCloud-säkerhetskopiering och iCloud-nyckelring.

Dessa internettjänster har skapats med samma säkerhetsmål som för hela iOS-plattformen. Säkerhetsmålen är säker hantering av data, både på enheten och under överföring i trådlösa nätverk, skydd av användarnas personliga information samt skydd mot hot i form av potentiellt skadlig eller obehörig tillgång till information och tjänster. Varje tjänst har en egen, kraftfull säkerhetsarkitektur som inte inverkar negativt på användarvänligheten i iOS.

Apple-ID

Ett Apple-ID är det konto som används för att logga in på Apple-tjänster som iCloud, iMessage, FaceTime, iTunes Store, Apple Books, App Store med flera. Det är viktigt för användarna att hålla sina Apple-ID:n skyddade, så att ingen obehörig får tillgång till deras konton. För att åstadkomma detta kräver Apple starka lösenord som måste vara minst åtta tecken långa, innehålla både bokstäver och siffror, inte får innehålla mer än tre identiska tecken i rad och inte får vara vanligt förekommande lösenord. Användarna uppmuntras att gå ännu längre än riktlinjerna kräver genom att lägga till extra tecken och skiljetecken som gör lösenorden ännu starkare. Apple kräver också att användare ställer in tre säkerhetsfrågor som kan användas till att bekräfta ägarens identitet när ändringar görs i kontoinformationen eller när någon försöker skapa ett nytt lösenord.

Apple skickar också e-post och pushnotiser till användarna när viktiga ändringar sker i deras konton – exempelvis om lösenordet eller faktureringsinformationen ändras, eller om deras Apple-ID har använts för att logga in på en ny enhet. Om användaren inte känner igen ändringarna uppmanas denna att genast byta lösenord till sitt Apple-ID.

Dessutom har Apple en mängd olika policyer och rutiner som har utformats för att skydda användarkonton. Detta innebär bland annat att antalet inloggningsförsök och försök att skapa ett nytt lösenord begränsas, att bedrägeriförsök övervakas mer aktivt så att attacker kan identifieras när de inträffar och att policyer regelbundet granskas så att Apple kan anpassa sig efter eventuell ny information som kan påverka kundernas säkerhet.

Tvåfaktorsautentisering

För att hjälpa användare att skydda sina konton ännu bättre erbjuder Apple *tvåfaktorsautentisering* som är ett extra skyddslager för Apple-ID:n. Syftet med tvåfaktorsautentisering är att se till att bara ett kontos ägare kan komma åt kontot även om någon annan kan lösenordet.

Med tvåfaktorsautentisering går det bara att komma åt en användares konto på tillförlitliga enheter, till exempel användarens iPhone, iPad eller Mac. Första gången du ska logga in på en ny enhet måste du ange två saker: lösenordet till Apple-ID:t och en sexsiffrig verifieringskod som visas automatiskt på användarens tillförlitliga enheter eller skickas till ett

tillförlitligt telefonnummer. Genom att ange koden bekräftar användaren att den nya enheten är tillförlitlig och att det är säkert att logga in. Eftersom det inte längre räcker med ett lösenord för att komma åt en användares konto gör tvåfaktorsautentiseringen att användarens Apple-ID och all personlig information som användaren har sparat hos Apple är bättre skyddade. Funktionen är inbyggd i iOS, macOS, tvOS, watchOS och autentiseringssystemen som används på Apples webbplatser.

Mer information om tvåfaktorsautentisering hittar du på https://support.apple.com/sv-se/HT204915

Tvåstegsverifiering

Sedan 2013 har Apple dessutom erbjudit en liknande säkerhetsmetod som kallas *tvåstegsverifiering*. När tvåstegsverifiering är aktiverat måste användarens identitet verifieras via en tillfällig kod som skickas till en av användarens betrodda enheter innan några ändringar tillåts i användarens Apple-ID, före inloggning i iCloud, iMessage, FaceTime eller Game Center samt före inköp i iTunes Store, Apple Books eller App Store från en ny enhet. Användarna får även tillgång till en 14 tecken lång återställningsnyckel, som bör förvaras på en säker plats, ifall de skulle glömma sitt lösenord eller förlora tillgången till sina betrodda enheter. De flesta nya användarna uppmanas att använda tvåfaktorsautentisering, men i vissa fall rekommenderas fortfarande tvåstegsverifiering istället.

Mer information om tvåstegsverifiering för Apple-ID finns på https://support.apple.com/sv-se/HT204152

Hanterade Apple-ID:n

Hanterade Apple-ID:n fungerar ungefär på samma sätt som ett Apple-ID, men de ägs och hanteras av en utbildningsinstitution. Institutionen kan skapa nya lösenord, begränsa inköp och kommunikation via t.ex. FaceTime och Meddelanden och ställa in rollbaserade behörigheter för personal, lärare och elever.

Vissa Apple-tjänster, till exempel Apple Pay, iCloud-nyckelring, HomeKit och Hitta min iPhone, är avaktiverade för hanterade Apple-ID:n.

Mer information om hanterade Apple-ID:n finns på https://help.apple.com/schoolmanager/#/tes78b477c81

Granska hanterade Apple-ID:n

Hanterade Apple-ID:n har också stöd för granskning så att institutioner kan följa juridiska regler och integritetsregler. Administratörs- eller lärarkonton samt konton för ansvariga kan beviljas granskningsrättigheter för specifika hanterade Apple-ID:n. Granskare kan endast övervaka konton som ligger under dem i skolans hierarki. Det innebär att lärare kan övervaka elever, att ansvariga kan granska lärare och elever och att administratörer kan granska ansvariga, lärare och elever.

När inloggningsuppgifter för granskning begärs i Apple School Manager skapas ett särskilt konto som endast kan användas för att komma åt det hanterade Apple-ID som angavs när granskningen begärdes. Granskningsbehörigheten gäller i sju dagar. Under den perioden kan granskaren läsa och ändra användarens innehåll som har sparats på iCloud eller i program med CloudKit aktiverat. Varje begäran om granskningsåtkomst registreras i Apple School Manager. I loggarna

finns information om vem som var granskare, vilket hanterat Apple-ID granskaren begärde åtkomst till, tidpunkten för begäran och om granskningen genomfördes.

Mer information om hanterade Apple-ID:n finns på: https://help.apple.com/schoolmanager/#/tesd8fcbdd99

Hanterade Apple-ID:n och personliga enheter

Hanterade Apple-ID:n kan också användas på personligt ägda iOS-enheter och Mac-datorer. Eleverna loggar in på iCloud med ett hanterat Apple-ID som institutionen har skapat samt ytterligare ett lösenord för hemmabruk som fungerar som den andra faktorn i tvåfaktorsautentiseringen av Apple-ID:n. När ett hanterat Apple-ID används på en personlig enhet kan inte iCloud-nyckelringen användas, och institutionen kan välja att begränsa andra funktioner som FaceTime eller Meddelanden. Alla iCloud-dokument som elever skapar när de är inloggade kan granskas enligt beskrivningen tidigare i det här avsnittet.

iMessage

Apples iMessage är en meddelandetjänst för iOS- och Apple Watch-enheter samt Mac-datorer. iMessage stöder text och bilagor som bilder, kontakter och platser. Meddelanden visas på alla användarens registrerade enheter, så att konversationer kan fortsätta från en enhet till en annan. iMessage använder Apples tjänst för pushnotiser (APNs). Apple loggar inte innehållet i meddelanden eller bilagor, och de skyddas av E2E-kryptering så att ingen utom sändaren och mottagaren kan komma åt dem. Apple kan inte avkryptera informationen.

När en användare aktiverar iMessage på en enhet genererar enheten två nyckelpar för användning med tjänsten: en 1280-bitars RSA-nyckel för kryptering och en 256-bitars ECDSA-nyckel på NIST P-256-kurvan för signering. De privata nycklarna för båda nyckelparen sparas i enhetens nyckelring och de publika nycklarna skickas till Apples ID-tjänst (IDS). Där kopplas de till användarens telefonnummer eller e-postadress tillsammans med enhetens APNs-adress.

När användarna aktiverar ytterligare enheter för användning med iMessage läggs deras publika nycklar för kryptering och signering, APNs-adresser och kopplade telefonnummer till i katalogtjänsten. Användarna kan också lägga till fler e-postadresser som verifieras genom att skicka en bekräftelselänk. Telefonnummer verifieras av operatörsnätet och SIM-kortet. I en del operatörsnät måste SMS användas (en bekräftelsedialogruta visas för användaren om SMS:et inte är kostnadsfritt). Verifiering av telefonnumret kan krävas för flera systemtjänster utöver iMessage, exempelvis FaceTime och iCloud. Alla användarens registrerade enheter visar ett varningsmeddelande när en ny enhet, ett nytt telefonnummer eller en ny e-postadress läggs till.

I iOS 12 eller senare visas meddelanden som skickas från olika adresser, men är länkade till samma Apple-ID, som en enda konversation på enheterna som tar emot dem. Det här är möjligt tack vare ett konto-ID som hämtas från IDS tillsammans med de publika nycklarna och APNs-adresser för en e-postadress eller ett telefonnummer.

Hur iMessage skickar och tar emot meddelanden

Användarna startar en ny iMessage-konversation genom att ange en adress eller ett namn. Om de skriver in ett telefonnummer eller en e-postadress kontaktar enheten IDS och hämtar de publika nycklarna och APNs-

adresserna för alla enheter som är kopplade till mottagaren. Om användaren skriver in ett namn använder enheten först appen Kontakter för att samla in telefonnummer och e-postadresser som är kopplade till det namnet. Sedan hämtar den de publika nycklarna och APNs-adresserna från IDS.

Användarens utgående meddelande krypteras enskilt för var och en av mottagarens enheter. De publika RSA-krypteringsnycklarna för de mottagande enheterna hämtas från IDS. För varje mottagande enhet genererar den sändande enheten ett slumpmässigt 88-bitarsvärde och använder det som en HMAC-SHA256-nyckel för att konstruera ett 40-bitarsvärde härlett från avsändarens och mottagarens publika nyckel och texten. Hoplänkningen av 88-bitars- och 40-bitarsvärdet ger en 128-bitarsnyckel som krypterar meddelandet med AES i CTR-läge. 40-bitarsvärdet används av mottagarsidan till att verifiera integriteten för den avkrypterade texten. Denna AES-nyckel per meddelande krypteras med RSA-OAEP till den publika nyckeln för den mottagande enheten. Kombinationen av den krypterade meddelandetexten och den krypterade meddelandenyckeln bearbetas med hashfunktionen SHA-1, och hashvärdet signeras med ECDSA med hjälp av den avsändande enhetens privata signeringsnyckel. De resulterande meddelandena, ett för varje mottagande enhet, består av den krypterade meddelandetexten, den krypterade meddelandenyckeln och avsändarens digitala signatur. Dessa skickas sedan till APNs för leverans. Metadata som tidsstämpel och APNs-routinginformation krypteras inte. Kommunikation med APNs krypteras med en så kallad FS-TLS-kanal.

APNs kan endast vidarebefordra meddelanden på upp till 4KB eller 16KB, beroende på iOS-version. Om meddelandet är för långt, eller om en bild eller annan bilaga ingår, krypteras bilagan med hjälp av AES i CTR-läge med en slumpgenererad 256-bitarsnyckel och överförs till iCloud. Bilagans AES-nyckel, dess **URI (Uniform Resource Identifier)** och ett SHA-1-hashvärde för dess krypterade form skickas sedan till mottagaren som innehållet i ett iMessage. Deras konfidentialitet och integritet skyddas genom normal iMessage-kryptering (se diagrammet nedan).

När det gäller gruppkonversationer upprepas processen för varje mottagare och deras enheter.

På den mottagande sidan får varje enhet en kopia av meddelandet från APNs och hämtar bilagan från iCloud när det behövs. Avsändarens inkommande telefonnummer eller e-postadress matchas mot mottagarens kontakter, så att ett namn kan visas om möjligt.

I likhet med alla pushnotiser raderas meddelandet från APNs när det har levererats. Men till skillnad från andra pushnotiser ställs iMessage-meddelanden i kö för leverans till enheter som är offline. Meddelanden lagras för närvarande i upp till 30 dagar.

Kundchatt

Kundchatt är en meddelandetjänst som gör det möjligt för användare att kommunicera med företag i appen Meddelanden. Det är bara användare som kan inleda konversationen, och företaget får ett anonymt ID för användaren. Företaget får inte användarens telefonnummer, e-postadress eller iCloud-kontoinformation. När du chattar med Apple får Apple ett Kundchatt-ID som är associerat till ditt Apple-ID. Användare har full kontroll över om de vill kommunicera. När en Kundchatt-konversation raderas tas den bort från användarens app Meddelanden och företaget blockeras från att skicka fler meddelanden till användaren.

Meddelanden som skickas till företaget krypteras individuellt mellan användarens enhet och Apples meddelandeservrar där meddelandena först avkrypteras och sedan vidarebefordras till företaget via TLS. Företagets svar skickas på liknande sätt via TLS till Apples meddelandeservrar där meddelandena krypteras och sedan skickas till användarens enhet. I likhet med iMessage köas meddelanden för leverans till nedkopplade enheter i upp till 30 dagar.

FaceTime

FaceTime är Apples video- och röstsamtalstjänst. I likhet med iMessage använder också FaceTime Apples pushnotistjänst för att upprätta en första anslutning till användarens registrerade enheter. Ljud- och videoinnehållet i FaceTime-samtal skyddas av E2E-kryptering så att ingen utom sändaren och mottagaren kan komma åt det. Apple kan inte avkryptera informationen.

Den inledande FaceTime-anslutningen sker via Apples serverinfrastruktur som vidarebefordrar datapaket mellan användarnas registrerade enheter. Med hjälp av Apples pushnotiser och STUN-meddelanden (Session Traversal Utilities for NAT) via reläanslutningen verifierar enheterna sina identitetscertifikat och upprättar en delad hemlighet för varje session. Den delade hemligheten användas till att härleda sessionsnycklar för mediekanaler som strömmas via SRTP-protokollet (Secure Real-time Transport Protocol). SRTP-paket krypteras med AES-256 i Counter Mode och HMAC-SHA1. Efter den inledande anslutningen och säkerhetsinställningen använder FaceTime STUN och ICE (Internet Connectivity Establishment) till att upprätta en P2P-anslutning mellan enheter, om möjligt.

FaceTime-gruppsamtal utökar FaceTime så att det stöder upp till 33 samtidiga deltagare. I likhet med klassiska tvåvägssamtal via FaceTime är gruppsamtalen heltäckande krypterade mellan de inbjudna deltagarnas enheter. En stor del av infrastrukturen och utformningen hos tvåvägs-FaceTime återanvänds visserligen, men FaceTime-gruppsamtal har en ny mekanism för nyckelupprättning byggd ovanpå den autentisering som

tillhandahålls av IDS. Det här protokollet tillhandahåller forward secrecy, vilket innebär att en enhet vars säkerhet äventyras inte kommer att läcka innehåll från tidigare samtal. Sessionsnycklarna paketeras via AES-SIV och distribueras bland deltagarna genom att använda en ECIES-konstruktion med tillfälliga P-256 ECDH-nycklar.

När ett nytt telefonnummer eller en ny e-postadress läggs till i ett pågående FaceTime-gruppsamtal upprättar aktiva enheter nya medienycklar och kommer aldrig att dela de tidigare använda nycklarna med de nyligen inbjudna enheterna.

iCloud

iCloud lagrar användarens kontakter, kalendrar, bilder, dokument med mera och håller informationen uppdaterad på alla hans eller hennes enheter, helt automatiskt. iCloud kan också användas av tredjepartsappar för att lagra och synkronisera dokument och viktiga appdata som definieras av utvecklaren. Användarna ställer in iCloud genom att logga in med ett Apple-ID och välja vilka tjänster de vill använda. iCloud-tjänster, inklusive Min bildström, iCloud Drive och säkerhetskopiering, kan avaktiveras av IT-administratörer via MDM-konfigurationsprofiler. Tjänsten är ovetande om vad som lagras och hanterar allt filinnehåll på samma sätt, som en samling byte.

Varje fil delas upp i flera delar som krypteras av iCloud med AES-128 och en nyckel som härleds från respektive dels innehåll och som använder SHA-256. Nycklarna och filens metadata sparas av Apple i användarens iCloud-konto. De krypterade fildelarna lagras, utan nycklarna eller information som identifierar användaren, på Apples och tredje parts lagringstjänster.

iCloud Drive

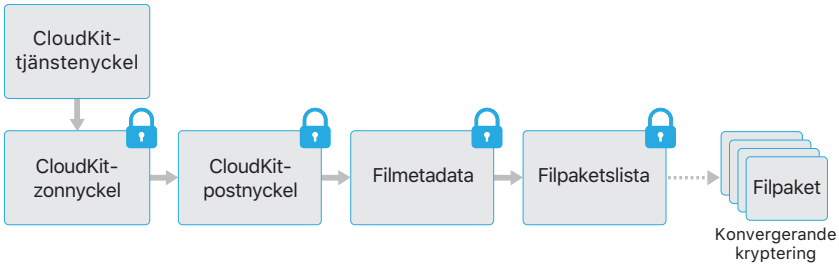
iCloud Drive lägger till kontobaserade nycklar för att skydda dokument som lagras i iCloud. Precis som andra iCloud-tjänster delar den upp och krypterar filinnehållet och lagrar de krypterade delarna med hjälp av tredjepartslösningar. Nycklarna till filinnehållet paketeras däremot av postnycklar som lagras i iCloud Drives metadata. Dessa postnycklar skyddas i sin tur av användarens iCloud Drive-tjänstenyckel, som sedan lagras i användarens iCloud-konto. Användarna får tillgång till metadata om sina iCloud-dokument genom att autentisera med iCloud, men de måste också ha iCloud Drive-tjänstenyckeln i sin ägo för att få tillgång till skyddade delar av iCloud Drive-lagringen.

CloudKit

Med CloudKit kan apputvecklare spara viktiga datavärden, strukturerade data och resurser på iCloud. Tillgången till CloudKit styrs med hjälp av apprättigheter. CloudKit hanterar både publika och privata databaser. Publika databaser används av alla kopior av appen, innehåller vanligen allmänna resurser och krypteras inte. I privata databaser lagras användarnas data.

Precis som iCloud Drive använder CloudKit kontobaserade nycklar för att skydda informationen som lagras i användarens privata databaser, och i likhet med andra iCloud-tjänster delar den upp filerna, krypterar dem och lagrar dem med hjälp av tredjepartstjänster. CloudKit använder en hierarki av nycklar som liknar dataskyddet. Filnycklarna paketeras av CloudKit-postnycklar. Postnycklarna skyddas i sin tur av en nyckel för hela

zonen som skyddas av användarens CloudKit-tjänstenyckel. CloudKit-tjänstenyckeln lagras i användarens iCloud-konto och är bara tillgänglig efter att användaren har autentiserat med iCloud.



CloudKit med E2E-kryptering

Apple Pay Cash, hälsodata, användarnyckelord, Siri-intelligens och Hej Siri använder CloudKits E2E-kryptering med en CloudKit-tjänstenyckel som skyddas av iCloud-nyckelringssynkronisering. För dessa CloudKit-behållare är nyckelhierarkin grundad i iCloud-nyckelring och delar därför säkerhetsfunktioner med iCloud-nyckelring. Nycklarna är bara tillgängliga på användarens betrodda enheter och inte för Apple eller någon tredje part. Om tillgången till iCloud-nyckelring förloras (se Säkerhet vid deponering senare i dokumentet) nollställs data i CloudKit, och om data finns på den betrodda lokala enheten överförs de på nytt till CloudKit.

Meddelanden på iCloud använder också CloudKit med E2E-kryptering med en CloudKit-tjänstenyckel som skyddas med iCloud-nyckelringssynkronisering. Om användaren har aktiverat iCloud-säkerhetskopiering blir CloudKit-tjänstenyckeln som används för Meddelanden på iCloud-behållaren säkerhetskopierad på iCloud så att användaren kan återskapa sina meddelanden även om han eller hon inte längre har tillgång till iCloud-nyckelring och sina betrodda enheter. Den här CloudKit-tjänstenyckeln byts ut varje gång användaren stänger av iCloud-säkerhetskopiering.

iCloud-säkerhetskopiering

iCloud säkerhetskopierar information dagligen via Wi-Fi. Den informationen omfattar sådant som enhetsinställningar, appdata, bilder och videor i kamerarullen samt konversationer i appen Meddelanden. iCloud skyddar innehållet genom att kryptera det när det skickas via internet, lagra det i ett krypterat format och använda säkra tokens för autentisering. iCloud-säkerhetskopiering görs bara när enheten är låst, ansluten till en strömkälla och är ansluten till internet via Wi-Fi. Genom krypteringen som används i iOS är systemet utformat för att skydda data samtidigt som det tillåter stegvis, oövervakad säkerhetskopiering och återställning.

Det här säkerhetskopieras på iCloud:

- Poster för inköpt musik, filmer, TV-program, appar och böcker. En användares iCloud-säkerhetskopia innehåller information om inköpt innehåll som finns på användarens iOS-enhet, men inte själva det inköpta innehållet. När användaren återskapar från en iCloud-säkerhetskopia hämtas det inköpta innehållet automatiskt från iTunes Store, Apple Books eller App Store. Vissa typer av innehåll hämtas inte automatiskt i alla länder eller regioner, och tidigare inköp kanske inte är tillgängliga om återköp har gjorts eller om objekten inte längre är tillgängliga i butiken. Den fullständiga historiken är kopplad till en användares Apple-ID.

Återställningsalternativ

Situation	Återställningsalternativ för användare med CloudKit med E2E-kryptering
Tillgång till betrodd enhet	Dataåterställning möjlig via en betrodd enhet eller iCloud-nyckelringsåterställning
Inga betrodda enheter	Dataåterställning endast möjlig via iCloud-nyckelringsåterställning

Situation	Återställningsalternativ för användare med Meddelanden på iCloud
iCloud-säkerhetskopiering aktiverad och tillgång till betrodd enhet	Dataåterställning möjlig via iCloud-säkerhetskopiering, tillgång till en betrodd enhet eller iCloud-nyckelringsåterställning
iCloud-säkerhetskopiering aktiverad och ingen tillgång till betrodd enhet	Dataåterställning möjlig via iCloud-säkerhetskopiering och iCloud-nyckelringsåterställning
iCloud-säkerhetskopiering avaktiverad och tillgång till betrodd enhet	Dataåterställning möjlig via en betrodd enhet eller iCloud-nyckelringsåterställning
Säkerhetskopiering avaktiverad och inga betrodda enheter	Dataåterställning endast möjlig via iCloud-nyckelringsåterställning

- Bilder och videor på en användares iOS-enheter. Lägg märke till att om en användare slår på iCloud-bildbibliotek på en iOS-enhet (iOS 8.1 eller senare) eller Mac (OS X 10.10.3 eller senare) lagras hans eller hennes bilder och videor redan i iCloud, så de ingår inte i iCloud-säkerhetskopian.
- Kontakter, kalenderaktiviteter, påminnelser och anteckningar
- Enhetsinställningar
- Appdata
- Samtalshistorik och ringsignaler
- Hemskärmen och ordningen på appar
- HomeKit-konfiguration
- Lösenord för visuell röstbrevlåda (kräver det SIM-kort som användes vid säkerhetskopieringen)
- Kundchatt, iMessage-, SMS- och MMS-meddelanden (kräver det SIM-kort som användes vid säkerhetskopieringen)

Obs! När Meddelanden i molnet är aktiverade tas iMessage- och Kunchatt-meddelanden samt SMS och MMS bort från användarens befintliga iCloud-säkerhetskopia och lagras istället i en E2E-krypterad CloudKit-behållare för Meddelanden. Användarens iCloud-säkerhetskopia behåller en nyckel till den behållaren. Om användaren senare stänger av iCloud-säkerhetskopiering byts nyckeln ut till den behållaren. Den nya nyckeln lagras endast i iCloud-nyckelring (där varken Apple eller tredje part kommer åt den) och nya data som skrivs i behållaren kan inte avkrypteras med den gamla behållarnyckeln.

När filer skapas i dataskyddsklasser som inte är tillgängliga när enheten är låst krypteras deras filnycklar med hjälp av klassnycklarna från nyckelsamlingen för iCloud-säkerhetskopiering. Filerna säkerhetskopieras till iCloud i sin ursprungliga, krypterade form. Filer i dataskyddsklassen No Protection krypteras under transport.

Nyckelsamlingen för iCloud-säkerhetskopiering innehåller asymmetriska nycklar (Curve25519) för varje dataskyddsklass, som används för att kryptera filnycklarna. Mer information om innehållet i nyckelsamlingen för säkerhetskopiering och nyckelsamlingen för iCloud-säkerhetskopiering finns under ”Dataskydd genom nyckelring” i avsnittet Kryptering och dataskydd i det här dokumentet.

Uppsättningen för säkerhetskopiering lagras i användarens iCloud-konto och består av en kopia av användarens filer samt nyckelsamlingen för iCloud-säkerhetskopiering. Nyckelsamlingen för iCloud-säkerhetskopiering skyddas av en slumpgenererad nyckel som också lagras tillsammans med säkerhetskopieringsuppsättningen. (Användarens iCloud-lösenord används inte för kryptering, så att byta iCloud-lösenord gör inte befintliga säkerhetskopior ogiltiga.)

Medan användarens nyckelringsdatabas säkerhetskopieras till iCloud fortsätter den att skyddas av en nyckel som är knuten till UID:t. Det gör att nyckelringen bara kan återskapas till samma enhet som den kom ifrån, och det betyder att ingen annan (inte heller Apple) kan läsa objekten i användarens nyckelring.

Vid återskapande hämtas de säkerhetskopierade filerna, nyckelsamlingen för iCloud-säkerhetskopiering och nyckeln till nyckelsamlingen från användarens iCloud-konto. Nyckelsamlingen för iCloud-säkerhetskopiering avkrypteras med dess egen nyckel, och sedan används filnycklarna

i nyckelsamlingen till att avkryptera filerna i uppsättningen för säkerhetskopiering. Dessa skrivs som nya filer till filsystemet och omkrypteras därmed enligt sin dataskyddsklass.

iCloud-nyckelring

Safari-integrering med iCloud-nyckelring
Safari kan skapa kryptografiskt starka slumpgenererade strängar att använda som lösenord till webbplatser. De sparas i nyckelringen och synkroniseras till andra enheter. Nyckelringsobjekt överförs från enhet till enhet och färdas då via Apples servrar, men de krypteras på ett sådant sätt att varken Apple eller någon annan enhet kan läsa deras innehåll.

Med iCloud-nyckelring kan användarna synkronisera sina lösenord säkert mellan iOS-enheter och Mac-datorer utan att lämna ut den informationen till Apple. Utöver stark integritet och säkerhet finns några andra faktorer som har haft stor inverkan på utformningen och arkitekturen hos iCloud-nyckelring: användarvänlighet och möjlighet att återskapa en nyckelring. iCloud-nyckelring innehåller två tjänster: synkronisering och återställning av nyckelringen.

Apple har utformat iCloud-nyckelring och återställning av nyckelringen så att användarens lösenord fortfarande skyddas under följande omständigheter:

- Om det uppstår säkerhetsproblem med användarens iCloud-konto.
- Om iCloud attackeras av obehöriga utifrån eller av en anställd.
- Om tredje part ansluter till användarkonton.

Synkronisering av nyckelringen

När en användare aktiverar iCloud-nyckelring för första gången upprättar enheten en tillförlitlighetscirkel och skapar en synkroniseringsidentitet till sig själv. Synkroniseringsidentiteten består av en privat nyckel och en publik nyckel. Den publika nyckeln till synkroniseringsidentiteten läggs till i cirkeln och cirkeln signeras två gånger: först av den privata nyckeln i synkroniseringsidentiteten, sedan igen med en asymmetrisk elliptisk nyckel (med P-256) som härleds ur lösenordet till användarens iCloud-konto. I cirkeln sparas också parametrarna (slumpmässigt utgångsvärde och iterationer) som används till att skapa nyckeln som baseras på användarens iCloud-lösenord.

Den signerade synkroniseringscirkeln placeras i lagringsutrymmet för användarens iCloud-nyckelvärde. Den kan inte läsas utan att känna till användarens iCloud-lösenord, och kan inte ändras på ett giltigt sätt utan tillgång till den privata nyckeln för medlemmens synkroniseringsidentitet.

När användaren aktiverar iCloud-nyckelring på en annan enhet märker den att användaren tidigare har upprättat en synkroniseringscirkel i iCloud som den inte är medlem i. Enheten skapar ett nyckelpar för sin synkroniseringsidentitet och skapar sedan en appbiljett för att begära medlemskap i cirkeln. Biljetten består av enhetens publika nyckel för dess synkroniseringsidentitet, och användaren uppmanas att autentisera med sitt iCloud-lösenord. Parametrarna för elliptisk nyckelgenerering hämtas från iCloud och genererar en nyckel som används till att signera appbiljetten. Slutligen placeras appbiljetten på iCloud.

När den första enheten ser att en appbiljett har kommit fram visar den en notis för användaren om att en ny enhet ber om att få bli medlem i synkroniseringscirkeln. Användaren skriver in sitt iCloud-lösenord och en matchande privat nyckel verifierar att appbiljetten är signerad. Det bevisar att personen som genererade förfrågan om att ingå i cirkeln angav användarens iCloud-lösenord vid det tillfälle då förfrågan gjordes.

När användaren godkänner att en ny enhet läggs till i cirkeln lägger den första enheten till den nya medlemmens publika nyckel i cirkeln och signerar den igen med dess synkroniseringsidentitet och med nyckeln som

härleds ur användarens iCloud-lösenord. Den nya synkroniseringscirkeln placeras i iCloud, där den signeras på ett liknande sätt av den nya cirkelmedlemmen.

Det finns nu två medlemmar i signeringscirkeln och varje medlem har den publika nyckeln till den andra medlemmen. De börjar nu att utväxla enskilda nyckelringsobjekt via lagringsutrymmet för nyckelvärden i iCloud eller lagrar dem i CloudKit vid behov. Om båda cirkelmedlemmarna har samma objekt synkroniseras det som senast har ändrats. Om den andra medlemmen har samma objekt och ändringsdatumet är detsamma hoppas objektet över. Varje objekt som synkroniseras krypteras så att det bara kan avkrypteras av en enhet som finns i användarens tillförlitlighetscirkel. Det kan inte avkrypteras av några andra enheter eller av Apple.

Processen upprepas när nya enheter går med i synkroniseringscirkeln. Om till exempel en tredje enhet går med visas en bekräftelse på båda de andra två enheterna. Användaren kan godkänna den nya medlemmen från vilken av dessa enheter han/hon vill. När nya enheter läggs till synkroniseras alla medlemmar med den nya enheten, så att alla har samma nyckelringsobjekt.

Däremot synkroniseras inte hela nyckelringen. Vissa objekt är enhetsspecifika, till exempel VPN-identiteter, och bör inte lämna enheten. Det är bara objekt med attributet `kSecAttrSynchronizable` som synkroniseras. Apple har ställt in det här attributet för Safari-användardata (inklusive användarnamn, lösenord och kreditkortsnummer), liksom för Wi-Fi-lösenord och HomeKit-krypteringsnycklar.

Som förval synkroniseras inte nyckelringsobjekt som lagts till av tredjepartsappar. Utvecklarna måste ställa in `kSecAttrSynchronizable` när de lägger till objekt i nyckelringen.

Återställning av nyckelringen

Återställning av nyckelringen gör det möjligt för användarna att deponera nyckelringen hos Apple utan att Apple får tillgång till lösenord eller andra data. Även om användaren bara har en enda enhet ger återställning av nyckelringen ett skydd mot dataförlust. Det är särskilt viktigt när Safari används till att skapa slumpmässiga, starka lösenord för webbkonton, eftersom dessa lösenord bara finns sparade i nyckelringen.

Sekundär autentisering och en säker deponeringstjänst som Apple har utvecklat särskilt för funktionen är centrala delar i tjänsten för återställning av nyckelringen. Användarens nyckelring krypteras med en stark lösenkod och deponeringstjänsten kräver att en strikt uppsättning villkor uppfylls för att en kopia av nyckelringen ska göras tillgänglig.

När iCloud-nyckelring slås på används enhetens lösenkod till att återskapa en deponerad nyckelring om tvåfaktorsautentisering är aktiverad för användarens konto. Om tvåfaktorsautentisering inte är inställd blir användaren ombedd att skapa en iCloud-säkerhetskod genom att ange en sexsiffrig lösenkod. Utan tvåfaktorsautentisering kan användaren välja att ange en egen längre kod, eller låta enheten skapa en kryptografiskt slumpmässig kod som denne kan registrera och spara på egen hand.

Nästa steg är att iOS-enheten exporterar en kopia av användarens nyckelring, krypterar och paketerar den med nycklarna i en asymmetrisk nyckelsamling och placerar den i användarens lagringsutrymme för nyckelvärden i iCloud. Nyckelsamlingen paketeras med användarens iCloud-säkerhetskod och den publika nyckeln för det HSM-modulkluster (Hardware Security Module) där deponeringsposten lagras. Detta blir användarens deponeringspost i iCloud.

Om användaren väljer att godkänna en kryptografiskt slumpmässig säkerhetskod istället för att ange en egen fyrsiffrig kod behövs inte någon deponeringspost. Istället används iCloud-säkerhetskoden för att paketera den slumpgenererade nyckeln direkt.

Förutom att välja en säkerhetskod måste användaren registrera ett telefonnummer. Det tillhandahåller en andra nivå av autentisering under återställningen av nyckelringen. Användaren får ett SMS som han/hon måste besvara för att återställningen ska fortsätta.

Säkerhet vid deponering

iCloud erbjuder en säker infrastruktur för deponering av nyckelringar där endast behöriga användare och enheter kan utföra en återställning. Bakom iCloud finns kluster av HSM-moduler som skyddar deponeringsposterna. Varje kluster har en nyckel som används för att kryptera deponeringsposterna som det ansvarar för (enligt vad som beskrivits tidigare i det här dokumentet).

För att återställa en nyckelring måste användaren ange användarnamn och lösenord till sitt iCloud-konto och svara på ett SMS som skickas till det telefonnummer som finns registrerat. När det är gjort måste användaren skriva in sin iCloud-säkerhetskod. HSM-klustret verifierar att användaren känner till sin iCloud-säkerhetskod med hjälp av SRP-protokollet (Secure Remote Password). Själva koden skickas inte till Apple. Varje medlem av klustret verifierar oberoende av de andra att användaren inte har överskridit antalet tillåtna försök att hämta posten (enligt beskrivningen nedan). Om majoriteten är överens packar klustret upp deponeringsposten och skickar den till användarens enhet.

Sedan använder enheten iCloud-säkerhetskoden för att packa upp den slumpmässiga nyckeln som användarens nyckelring krypterades med. Med den nyckeln avkrypteras nyckelringen – som hämtats från lagringsutrymmet för nyckelvärden i iCloud – och återskapas på enheten. Endast tio försök att autentisera och hämta en deponeringspost tillåts. Efter några misslyckade försök låses posten och användaren måste kontakta Apples support för att få fler försök. Efter det tionde misslyckade försöket förstör HSM-klustret deponeringsposten och nyckelringen går förlorad för gott. Detta skyddar mot automatiserade intrångsförsök i syfte att hämta deponeringsposten. Priset för den säkerheten är att informationen i nyckelringen kan behöva offras.

Dessa policyer är kodade i den fasta HSM-programvaran. De administrativa åtkomstkorten som tillåter att den fasta programvaran ändras har förstörts. Om någon försöker göra ändringar i den fasta programvaran eller komma åt den privata nyckeln raderar HSM-klustret den privata nyckeln. Om detta skulle inträffa kommer ägaren av varje nyckelring som skyddas av klustret att få ett meddelande om att deras deponeringspost har gått förlorad. De kan då välja att registrera sig igen.

Siri

Genom att bara prata som vanligt kan användarna låta Siri skicka meddelanden, schemalägga möten, ringa telefonsamtal med mera. Siri kan svara på en mängd olika förfrågningar med hjälp av röstigenkänning, talsyntes och en klient/server-modell. De uppgifter Siri utför har utformats för att använda så lite privat information som möjligt om användaren och skydda den informationen helt och fullt.

När Siri är på skapar enheten slumpmässiga identifierare som används tillsammans med röstigenkänning och Siris servrar. Dessa identifierare används endast inom Siri och till att förbättra tjänsten. Om Siri sedan stängs av kommer enheten att generera en ny slumpmässig identifierare som används när Siri aktiveras igen.

För att möjliggöra en del av Siris funktioner skickas viss information om användaren från enheten till servern. Det innefattar information om musikbiblioteket (låttitlar, artister och spellistor), namnen på påminnelselistor samt namn och relationer som anges i Kontakter. All kommunikation med servern sker via HTTPS.

När en Siri-session startas skickas användarens för- och efternamn (från Kontakter) till servern tillsammans med en ungefärlig geografisk position. Det gör det möjligt för Siri att svara användaren med dennes namn eller besvara förfrågningar som bara kräver en ungefärlig geografisk position, till exempel om vädret.

Om frågan kräver en mer exakt platsangivelse – exempelvis att fastställa adresserna till biografer i närheten – ber servern enheten om en mer exakt position. Detta är ett exempel på att information bara skickas till servern när det är absolut nödvändigt för att svara på användarens fråga. All information om sessionen raderas dock efter tio minuters overksamhet.

När en Siri-förfrågan görs från Apple Watch skapar klockan en egen slumpmässig, unik identifierare (vilket beskrivits tidigare). Men i stället för att skicka användarens uppgifter en gång till innehåller denna förfrågan även Siri-identifieraren för den iPhone som är parkopplad och tillhandahåller därmed en referens till denna information.

Inspelningen av användarens röst skickas till Apples röstigenkännings-server. Om uppgiften endast gäller diktering skickas den skrivna texten tillbaka till enheten. I annat fall analyserar Siri texten och kombinerar den, om det behövs, med information från profilen som är kopplad till enheten. Om en förfrågan till exempel är ”skicka ett meddelande till mamma” används de relationer och namn som har överförts från Kontakter. Kommandot för den identifierade åtgärden skickas sedan tillbaka till enheten och utförs av denna.

Många Siri-funktioner utförs av enheten på order från servern. Om användaren exempelvis ber Siri att läsa upp ett inkommande meddelande säger servern helt enkelt åt enheten att läsa upp innehållet i olästa meddelanden. Innehållet och avsändaren skickas inte till servern.

Användarens röstinspelningar sparas i sex månader, så att igenkännings-systemet kan använda dem för att bättre förstå användarens röst. Efter sex månader sparas en annan kopia, utan identifierare, som Apple kan använda i upp till två år för att förbättra och utveckla Siri. Apple kan fortsätta att använda ett litet urval av inspelningar, utskrifter och kopplade data utan identifierare till fortlöpande förbättring och kvalitetssäkring av Siri efter två år. Dessutom sparas vissa inspelningar med referenser till musik, idrottslag och spelare samt företag och sevärdheter i syfte att förbättra Siri.

Siri kan också anropas via röstaktivering. Den röststyrda aktiveringen utförs lokalt på enheten. I det här läget aktiveras Siri bara när det inkommande ljudet matchar ljudbilden i den angivna aktiveringsfrasen i tillräckligt hög grad. När aktiveringen upptäcks skickas motsvarande ljud tillsammans med det påföljande Siri-kommandot till Apples röstigenkänningsserver för vidare bearbetning, vilken sker enligt samma regler som andra röstinspelningar som görs via Siri.

Användare kan också anropa Siri på Apple Watch genom att hålla klockan nära munnen och säga en förfrågan till Siri. Siri aktiveras på det här sättet när både:

- En maskininlärningsmodell på enheten känner igen akustiken från mänskligt tal nära enheten.
- En andra maskininlärningsmodell på enheten identifierar en rörelseprofil och en enhetsvinkel som överensstämmer med gesten Håll upp för att tala.

När den här kombinationen av ljud och rörelse upptäcks skickas motsvarande ljud till Apples röstigenkänningsserver för vidare bearbetning, vilken sker enligt samma regler som andra röstinspelningar som görs via Siri.

Siri-förslag

Siri-förslag för appar och genvägar genereras med hjälp av maskininlärning på enheten. Inga data skickas till Apple förutom information (som inte kan användas till att identifiera användaren) om vilka signaler som var tydliga förutsägelser gällande genvägar eller appöppningar.

Genvägar i Siri

Genvägar som läggs till i Siri synkroniseras till alla Apple-enheter som använder iCloud och krypteras med hjälp av CloudKit med E2E-kryptering. Fraserna som är associerade med genvägar synkroniseras till Siri-servern för röstigenkänning och associeras med den slumpmässiga Siri-identifieraren som beskrivs i Siri-avsnittet. Apple får inte innehållet i genvägarna, utan de lagras lokalt i ett datavalv.

Appen Genvägar

Du kan välja att synkronisera anpassade genvägar i appen Genvägar med andra Apple-enheter med hjälp av iCloud. Genvägar kan också delas med andra användare via iCloud.

Anpassade genvägar är mångsidiga – de liknar skript eller program. Ett karantänsystem används till att isolera genvägar som har hämtats från internet. Användaren varnas första gången som en genväg ska användas och får möjlighet att först granska genvägen och information om sådant som dess ursprung.

Anpassade genvägar kan också köra användarspecificerade JavaScript på webbplatser i Safari när de anropas från delningsbladet. För att skydda mot bedrägliga JavaScript, som exempelvis lurar användaren att köra ett skript på en social mediewebbplats för att stjäla data, hämtas uppdaterade definitioner för skadeprogram så att bedrägliga skript kan identifieras vid körning. Den första gången en användare kör JavaScript på en domän blir användaren uppmanad att tillåta att genvägar som innehåller JavaScript körs på den aktuella webbsidan för den domänen.

Safari-förslag, Siri-förslag vid sökning, Slå upp, #images, appen News och widgeten News i länder utan News

Safari-förslag, Siri-förslag vid sökning, Slå upp, #images, appen News och widgeten News i länder utan News visar användarna förslag som sträcker sig bortom deras enheter, från källor som Wikipedia, iTunes Store, lokala nyheter, träffar från Kartor och App Store – och ger till och med förslag innan användaren börjar skriva.

När en användare börjar skriva i adressfältet i Safari, öppnar eller använder Siri-förslag vid sökning, använder Slå upp, öppnar #images, använder sökning i appen News eller använder widgeten News i länder utan News, skickas följande kontext krypterat via HTTPS till Apple för att tillhandahålla relevanta träffar åt användaren:

- En identifierare som roteras var 15:e minut för att skydda integriteten.
- Användarens sökfråga.
- Den troligaste frågekompletteringen baserad på sammanhang och lokalt cachelagrade tidigare sökningar.
- Enhetens ungefärliga plats, om Platstjänster är aktiverat för platsbaserade förslag. Hur stort ungefärligt område som anges beror på den uppskattade befolkningstätheten på den aktuella platsen. Om enheten befinner sig i en glesbygdsregion där användarna kan vara på geografiskt längre avstånd från varandra anges ett större område än om den är i en stadskärna, där användarna sannolikt är närmare varandra. Användarna kan avaktivera all överföring av platsinformation till Apple genom att öppna Inställningar och stänga av Platstjänster för platsbaserade förslag. Om Platstjänster avaktiveras kan Apple använda enhetens IP-adress till att ange en ungefärlig plats.
- Typ av enheter och om sökningen gjorts i Siri-förslag vid sökning, Safari, Slå upp, appen News eller Meddelanden.
- Typ av anslutning.
- Information om de tre senast använda apparna på enheten (för att ge ytterligare söksammanhang). Information skickas endast om appar som finns på Apples lista över populära appar och har använts inom de senaste tre timmarna.
- En lista över populära appar på enheten.
- Regionala inställningar för språk, plats och inmatning.
- Om användarens enhet har tillgång till prenumerationstjänster för musik eller video kan information som namn på prenumerationstjänsterna och typer av prenumerationer komma att skickas till Apple. Användarens kontonamn, nummer och lösenord skickas inte till Apple.
- Sammanfattad, sammanslagen representation av intresseområden

När en användare väljer en träff, eller avslutar appen utan att ha valt någon träff, skickas viss information till Apple för att förbättra kvaliteten på senare träffar. Den här informationen knyts endast till samma 15-minutersessionsidentifierare och inte till någon enskild användare. Denna feedback omfattar en del av den tidigare beskrivna sammanhangsberoende informationen, liksom information om interaktioner som:

- Tidsintervall mellan interaktion och söknätverksförfrågningar.
- Rankning och visningsordning för förslag.
- ID för den träff och den åtgärd som valdes om träffen är en icke-lokal träff, eller kategorin för den valda träffen om träffen är en lokal träff.
- En flagga som visar om användaren valde träffen.

Apple sparar loggar över förslag med förfrågningar, sammanhang och feedback i upp till 18 månader. Ett urval loggar med exempelvis förfrågningar, språk, domän, ungefärlig plats och sammanslagen statistik sparas i upp till fem år.

I vissa fall kan förslag vidarebefordra förfrågningar om vanliga ord och fraser till en berättigad partner för att kunna ta emot och visa sökträffar från partnern. Apple använder proxy för förfrågningarna, så dessa partners får inte tillgång till användares IP-adresser eller sökfeedback. All kommunikation med partners krypteras via HTTPS. För förfrågningar som ofta återkommer tillhandahåller Apple sökkontext i form av plats på stadsnivå, enhetstyp och klientspråk till partnern i syfte att förbättra sökprestanda.

För att förstå och förbättra förslagsprestanda geografiskt och i olika typer av nätverk loggas följande information utan sessionsidentifierare:

- Partiell IP-adress (utan den sista oktetten för IPv4-adressen, utan de sista 80 bitarna för IPv6-adressen)
- Ungefärlig plats
- Ungefärlig tidpunkt för begäran
- Latens-/överföringsförhållande
- Svarsstorlek
- Anslutningstyp
- Språk och region
- Enhetstyp och app som skickar begäran

Intelligent spårningsförebyggande i Safari

Intelligent spårningsförebyggande (Intelligent Tracking Prevention, ITP) är en del av Safaris integritetsfrämjande förvalda policy gällande cookies och webbplatsdata. Det bidrar till att förhindra spårning mellan webbplatser genom att begränsa tillgången till cookies och andra webbplatsdata.

ITP samlar in statistik om resursinläsningar (bilder, skript osv.) liksom användarinteraktioner som tryck och textinmatning. En maskininlärningsmodell används för klassificering på enheten av vilka domännamn som har förmåga att spåra användaren mellan webbplatser baserat på den insamlade statistiken.

När en domän klassificeras som att den har spårningsförmåga partitioneras dess cookies omedelbart av ITP om användaren tidigare har interagerat med den webbplatsen som första part. ITP börjar omedelbart att blockera cookies för klassificerade domäner som användaren inte har interagerat med. Exempelvis:

- video.example erbjuder en reklamfri abonnemangstjänst och många av dess videor är inbäddade på andra webbplatser.
- En användare loggar in på video.example och sedan på andra webbplatser som har inbäddat innehåll från video.example.
- ITP klassificerar video.example som att den har spårningsförmåga och partitionerar därför dess cookies.
- När en användare besöker newspaper.example, och den har inbäddat innehåll från video.example, är de cookies som tillhandahålls för video.example partitionerade cookies som är specifika för video.example på newspaper.example.

Inbäddat innehåll från tredje part kan be en användare om tillgång till dennas cookies från första part med hjälp av Storage Access-API:t. När en användare trycker eller klickar på inbäddat innehåll från tredje part som använder Storage Access-API:t visar Safari en dialogruta som frågar om användaren vill ge den tredje parten tillgång till dess cookies och webbplatsdata, vilket tillåter att den tredje parten spårar användaren på förstapartsdomänen. Om en användare väljer Tillåt får det inbäddade innehållet från tredje part tillgång till dennas cookies från första part så länge sidbesöket varar, och vid efterföljande besök får det inbäddade innehållet från tredje part tillgång till cookies från första part efter att en användare interagerar med det inbäddade innehållet och innehållet anropar Storage Access-API:t. Eftersom användaren har tillåtit denna tillgång tidigare visas ingen dialogruta. Användarens beslut sparas för kombinationen av första och tredje parter och rensas när användaren rensar Safari-historiken.

Befintliga cookies från domäner som klassificeras som att ha spårningsförmåga raderas om en användare inte har interagerat med domänen (direkt eller via Storage Access-API:t) under 30 dagars aktiv Safari-användning. Efter 30 dagar utan interaktion kan en domän som klassificeras som att ha spårningsförmåga inte heller skapa nya cookies. Safari ger aldrig tillgång till andra webbplatsdata från första part i sammanhang som rör tredje part.

Tack vare ITP:s isolering av förstaparts- och tredjepartsdata bidrar det till att förhindra användningen av cookies och webbplatsdata för spårning mellan webbplatser. Apple har inte tillgång till information om vilka domännamn som en viss enhet har samlat in statistik för eller klassificerat som att ha spårningsförmåga.

Utöver att blockera cookies från tredje part från domäner som är klassificerade som att ha spårningsförmåga putsar ITP även den HTTP Referer-information som skickas till tredjepartsdomäner som är klassificerade som att ha spårningsförmåga så att den bara innehåller sidans ursprung.

Hantering av användarlösenord

iOS innehåller en rad funktioner som gör det enkelt för användare att säkert och smidigt autentisera sig i tredjepartsappar och på webbplatser som använder lösenord för autentisering. Lösenord sparas i en särskild Autofyll lösenord-nyckelring som styrs och hanteras av användaren i Inställningar > Lösenord och konton > Webbplats- och applösenord. Appar kommer inte åt nyckelringen Autofyll lösenord utan användarens tillstånd. Inloggningsuppgifter som sparas i nyckelringen Autofyll lösenord synkroniseras mellan enheter med hjälp av iCloud-nyckelring när den är aktiverad.

Lösenordshanteraren för iCloud-nyckelring och Autofyll lösenord gör följande:

- Fyller i inloggningsuppgifter i appar och på webbplatser
- Genererar starka lösenord
- Sparar lösenord i både appar och på webbplatser i Safari
- Delar lösenord säkert till en användares kontakter
- Tillhandahåller lösenord till en Apple TV i närheten som efterfrågar inloggningsuppgifter

Apptillgång till sparade lösenord

API för delade webbinloggningsuppgifter

iOS-appar kan interagera med nyckelringen Autofyll lösenord med hjälp av följande två API:er:

`SecRequestSharedWebCredential`

`SecAddSharedWebCredential`

Tillgång till iOS-appar medges endast om både apputvecklaren och webbplatsadministratören har gett sitt godkännande och användaren samtycker. Apputvecklare meddelar sin avsikt att använda sparade lösenord i Safari genom att inkludera rättigheterna till det i appen. I rättigheterna anges de fullt berättigade domännamnen på associerade webbplatser. Webbplatserna måste även placera en fil på servern som innehåller en lista på de unika appidentifierarna hos appar som har godkänts av Apple.

När en app med rättigheten com.apple.developer.associated-domains installeras skickar iOS en TLS-begäran till alla webbplatser på listan och begär en av följande filer:

- apple-app-site-association
- .well-known/apple-app-site-association

Om filen innehåller appidentifieraren för appen som installeras kommer iOS att markera webbplatsen och appen som betrodda. Det är bara om relationen är betrodd som anrop till dessa två API:er resulterar i en fråga till användaren, som sedan måste samtycka innan några lösenord lämnas ut till appen, uppdateras eller raderas.

Autofyll lösenord för appar

iOS gör det möjligt för användare att mata in sparade användarnamn och lösenord i relaterade fält i appar genom att trycka på ett alternativ i iOS-tangentbordets QuickType-fält. Den drar nytta av samma app-website-associationsmekanism som drivs av apple-app-site-association-filen för att skapa en stark association mellan appar och webbplatser. Det här gränssnittet exponerar inte några inloggningsuppgifter för appen förrän en användare samtycker till att ge uppgifterna till appen. När iOS har markerat att en webbplats och app har en tillförlitlig relation kommer QuickType-fältet även att direkt föreslå inloggningsuppgifter som ska anges i appen. Det innebär att användare kan välja att lämna ut inloggningsuppgifter som har sparats i Safari till appar som har samma säkerhetsegenskaper, men utan att apparna måste använda en API.

När en app och webbplats har en betrodd relation, och en användare skickar inloggningsuppgifter inuti en app, kan iOS uppmana användaren att spara inloggningsuppgifterna i nyckelringen Autofyll lösenord för senare användning.

Automatiska starka lösenord

När iCloud-nyckelring är aktiverad skapar iOS starka, slumpmässiga, unika lösenord när användare registrerar sig eller ändrar lösenord i en app eller på en webbplats i Safari. Användare måste aktivt välja bort starka lösenord. Genererade lösenord sparas i nyckelringen och synkroniseras mellan enheter med iCloud-nyckelring när den är aktiverad.

Lösenord som genereras av iOS är 20 tecken långa som förval. De innehåller en siffra, ett versalt tecken, två bindestreck och 16 gemena tecken. De genererade lösenorden är starka och innehåller 71-bitars entropi.

iOS genererar lösenord i appar och Safari baserade på heuristik som bedömer att en password-field-upplevelse är avsedd för att skapa lösenord. Om heuristiken misslyckas med att upptäcka ett lösenordssammanhang där ett nytt lösenord bör skapas kan apputvecklare ange UITextContentType.newPassword i textfältet och webbutvecklare ange autocomplete=”new-password” i sina <input>-element.

Appar och webbplatser kan tillhandahålla regler för iOS som ser till att genererade lösenord är kompatibla med den relevanta tjänsten. iOS genererar starkast möjliga lösenord som uppfyller de angivna reglerna. Utvecklare kan tillhandahålla de här reglerna med UITextInputPasswordRules eller attributet passwordrules i sina <input>-element.

Skicka lösenord till andra personer eller enheter

AirDrop

När iCloud är aktiverat kan användare skicka en sparad inloggningsuppgift (med information om vilka webbplatser den är sparad för, användarnamnet och lösenordet) till en annan enhet via AirDrop När inloggningsuppgifter skickas via AirDrop används alltid läget Bara kontakter, oberoende av användarens inställningar. (Mer information finns i AirDrop-säkerhet.) När användaren har gett sitt medgivande på mottagarenheten sparas inloggningsuppgiften i användarens nyckelring Autofyll lösenord.

Apple TV

Autofyll lösenord är tillgängligt för att fylla i inloggningsuppgifter i appar på Apple TV. När användaren fokuserar på ett textfält för användarnamn eller lösenord i tvOS börjar Apple TV att annonsera en förfrågan om Autofyll lösenord via Bluetooth LE (BLE).

På alla iPhone-enheter i närheten visas ett meddelande som uppmanar användaren att dela en inloggningsuppgift med Apple TV. En iPhone och Apple TV som använder samma iCloud-konto krypterar kommunikationen mellan enheterna under den här processen. Om iPhone är inloggad på ett annat iCloud-konto än Apple TV:

- En PIN-kod används till att upprätta en krypterad anslutning.
- iPhone måste vara olåst och nära den Siri Remote som är parkopplad med Apple TV för att få det här meddelandet.

När den krypterade anslutningen har skapats med Bluetooth LE-länkkryptering skickas inloggningsuppgiften till Apple TV och anges automatiskt i relevanta textfält i appen.

Tillägg för inloggningsuppgifter

Användare kan utse en kompatibel app från tredje part som tillhandahåller inloggningsuppgifter för autofyll i inställningarna Lösenord och konton. Den här mekanismen bygger på tillägg. Tillägget för inloggningsuppgifter måste ha en vy för att välja inloggningsuppgifter och kan om möjligt tillhandahålla iOS-metadata om sparade inloggningsuppgifter så att de kan erbjudas direkt i QuickType-fältet. Dessa metadata innehåller webbplatsen för inloggningsuppgiften och det associerade användarnamnet, men inte lösenordet. iOS kommunicerar med tillägget för att få lösenordet när användaren väljer att fylla i det i en app eller på en webbplats i Safari. Metadata för inloggningsuppgifter sparas i sandlådan för inloggningsuppgiftstillhandahållaren och tas automatiskt bort när en app avinstalleras.

Enhetshantering

iOS har stöd för flexibla säkerhetspolicyer och konfigurationer som är enkla att genomdriva och hantera. Det hjälper företag och organisationer att skydda information och försäkra sig om att medarbetarna uppfyller företagets krav, även när de använder egna enheter – till exempel inom ramen för ett BYOD-program.

Organisationer kan använda resurser som lösenkodsskydd, konfigurationsprofiler, fjärradering och MDM-lösningar från tredje part för att hantera mängder av enheter. Därigenom kan företagsinformationen hållas skyddad även om de anställda har tillgång till den på sina personliga iOS-enheter.

Lösenkodsskydd

Den förvalda inställningen är att användaren får ange en numerisk PIN-kod som lösenkod. På enheter med Touch ID eller Face ID är den kortaste lösenkodslängden fyra siffror. Användaren kan ange längre alfanumeriska lösenkoder genom att öppna Inställningar > Touch ID och lösenkod, trycka på Lösenkodsalternativ och välja Alfanumerisk kod. Längre och mer komplexa lösenkoder är svårare att gissa eller knäcka och rekommenderas därför.

Administratörer kan se till att alla på företaget använder komplexa lösenkoder och andra policyer via MDM eller Exchange ActiveSync, eller genom att kräva att användarna installerar konfigurationsprofiler manuellt. Följande policyer för lösenkoder kan användas:

- Tillåt enkelt värde
- Kräv alfanumeriskt värde
- Minsta lösenkodslängd
- Minsta antal komplexa tecken
- Högsta lösenkodsålder
- Lösenkodshistorik
- Fördröjning för autolås
- Tidsfrist för enhetslås
- Högsta antal misslyckade försök
- Tillåt Touch ID eller Face ID

Administratörsinformation om de olika policyerna finns på https://help.apple.com/deployment/mdm/#/mdm4D6A472A

Utvecklarinformation om de olika policyerna finns på https://developer.apple.com/library/ios/featuredarticles/iPhoneConfigurationProfileRef/

iOS-parkopplingsmodell

iOS använder en parkopplingsmodell för att styra tillgången till en enhet från en värddator. Vid parkoppling upprättas en betrodd relation mellan en enhet och dess värd som bekräftas genom utväxling av publika nycklar. iOS använder detta bevis på tillit för att möjliggöra extra funktioner, som datasynkronisering, mellan enheten och den anslutna värden.

I iOS 9 går tjänster som kräver parkoppling inte att starta förrän enheten har låsts upp av användaren.

I iOS 10 eller senare kräver dessutom vissa tjänster, bland annat bildsynkronisering, att enheten är upplåst innan de går att starta.

I iOS 11 eller senare startas tjänster bara om enheten nyligen har låsts upp.

Parkopplingsprocessen kräver att användaren låser upp enheten och godkänner förfrågan om parkoppling från värden. I iOS 11 eller senare måste användaren även ange sin lösenkod. När användaren har gjort det utväxlar värden och enheten 2 048-bitars publika RSA-nycklar och sparar dem. Värden tilldelas sedan en 256-bitars nyckel som kan låsa upp en deponerad nyckelsamling som lagras på enheten (se beskrivningen av en deponerad nyckelsamling i avsnittet Nyckelsamlingar i det här dokumentet). De utväxlade nycklarna används till att starta en krypterad SSL-session som enheten kräver innan den skickar skyddade data till värden eller startar en tjänst (iTunes-synkronisering, filöverföringar, Xcode-utveckling och så vidare). Enheten kräver anslutningar från en värd via Wi-Fi för att använda den här krypterade sessionen till all kommunikation, så den måste ha parkopplats tidigare via USB. Parkoppling gör det också möjligt att använda flera olika diagnosfunktioner. Parkopplingsposter som inte har använts på mer än sex månader upphör att gälla i iOS 9. Den här tidsramen har förkortats till 30 dagar i iOS 11 eller senare.

Om du vill veta mer går du till:
https://support.apple.com/sv-se/HT6331

Vissa tjänster, som com.apple.pcapd, fungerar endast via USB. Tjänsten com.apple.file_relay kräver dessutom att en Apple-signerad konfigurationsprofil är installerad.

I iOS 11 eller senare kan Apple TV använda SRP-protokollet (Secure Remote Password) till att trådlöst upprätta en parkopplingsrelation.

Användaren kan rensa listan över betrodda värdar med alternativet Nollställ nätverk eller Nollställ integritetsskydd.

Om du vill veta mer går du till:
https://support.apple.com/sv-se/HT5868

Konfigurationskrav

En konfigurationsprofil är en XML-fil som gör det möjligt för administratörer att distribuera konfigurationsinformation till iOS-enheter. Inställningar som definieras av en installerad konfigurationsprofil kan inte ändras av användaren. Om användaren raderar en konfigurationsprofil försvinner också alla inställningar som definieras av profilen. Det här innebär att administratören kan genomdriva vissa inställningar genom att knyta policyer till Wi-Fi- och dataanslutning. Exempelvis kan en konfigurationsprofil som tillhandahåller e-postinställningar också ange enhetens lösenkodspolicy. För att få tillgång till e-posten måste alltså användarens lösenkod uppfylla administratörens krav.

Konfigurationsprofiler i iOS innehåller ett antal möjliga inställningar, däribland:

- Lösenkodspolicyer
- Begränsningar av funktioner (exempelvis avaktivering av kameran).
- Wi-Fi-inställningar
- VPN-inställningar
- E-postserverinställningar
- Exchange-inställningar
- Inställningar för LDAP-katalogtjänster
- Inställningar för CalDAV-kalendertjänster
- Webbklipp
- Inloggningsuppgifter och nycklar
- Avancerade mobilnätsinställningar

Du kan se en aktuell lista för administratörer på https://help.apple.com/deployment/mdm/#/mdm5370d089

Du kan se en aktuell lista för utvecklare på https://developer.apple.com/library/ios/featuredarticles/iPhoneConfigurationProfileRef/

Konfigurationsprofiler kan signeras och krypteras för att validera deras ursprung, garantera deras integritet och skydda deras innehåll. Konfigurationsprofiler krypteras med CMS (RFC 3852), med stöd för 3DES och AES-128.

Konfigurationsprofiler kan också låsas till en enhet så att de inte kan tas bort eller endast kan tas bort med en lösenkod. Eftersom många företagsanvändare äger sina egna iOS-enheter kan konfigurationsprofiler som kopplar enheten till en MDM-lösning tas bort – men då försvinner också all hanterad konfigurationsinformation, alla data och alla appar.

Användarna kan installera konfigurationsprofiler direkt på sina enheter med Apple Configurator 2 eller hämta dem via Safari, skicka dem via e-post eller överföra dem trådlöst från en MDM-lösning. När en användare ställer in en enhet i Apple School Manager eller Apple Business Manager hämtar enheten en profil för MDM-registrering och installerar den.

MDM (Mobile Device Management)

Stödet för MDM i iOS gör att företag säkert kan konfigurera och hantera små- eller storskalig driftsättning av iPhone, iPad, Apple TV och Mac-datorer i sina organisationer. MDM-funktionerna bygger på befintlig iOS-teknik, som konfigurationsprofiler, trådlös registrering och Apples tjänst för pushnotiser (APNs). Exempelvis används APNs till att väcka enheten så att den kan kommunicera direkt med MDM-lösningen via en säker anslutning. Ingen konfidentiell eller företagsägd information överförs via APNs.

Med hjälp av MDM kan IT-avdelningar registrera iOS-enheter i en företagsmiljö, konfigurera och uppdatera inställningar trådlöst, övervaka att företagspolicyer efterlevs, hantera programuppdateringspolicyer och till och med fjärrlåsa eller fjärradera hanterade enheter.

Mer information om MDM finns på

- https://www.apple.com/se/iphone/business/it/management.html
- https://help.apple.com/deployment/ios/#/ior07301dd60
- https://help.apple.com/deployment/mdm/#/mdmbf9e668

Delad iPad

Delad iPad är ett läge för flera användare som används vid driftsättning av iPad i utbildningsmiljöer. Funktionen gör det möjligt för elever att dela en iPad utan att dela dokument och data. Varje elev får en egen hemkatalog som skapas på en APFS-volym som skyddas av användarens inloggningsuppgifter. För att kunna använda Delad iPad krävs ett hanterat Apple-ID som har skapats och ägs av skolan. Med Delad iPad kan en elev logga in på alla organisationsägda enheter som har konfigurerats för att användas av flera elever. Elevernas data delas upp i separata hemkataloger, var och en i sin egen dataskyddsdomän, som skyddas genom både UNIX-behörigheter och körning i sandlåda.

Logga in på Delad iPad

När en elev loggar in autentiseras det hanterade Apple-ID:t med hjälp av SRP-protokollet från Apples identitetsservrar. Om autentiseringen godkänns beviljas en kortlivad åtkomsttoken som är specifik för enheten. Om eleven har använt enheten tidigare har han eller hon redan ett lokalt användarkonto som låses upp på samma sätt.

Om eleven inte har använt enheten tidigare får den ett nytt UNIX-användarnamn, en APFS-volym med användarens hemkatalog och en logisk nyckelring. Om enheten inte är ansluten till internet (t.ex. om eleven är på en studieresa) kan autentiseringen ske mot ett lokalt konto under ett begränsat antal dagar. I sådana fall kan endast användare med redan befintliga lokala konton logga in. När tidsgränsen har överskridits måste eleverna autentisera via internet, även om det redan finns ett lokalt konto.

När en elevs lokala konto har låsts upp eller skapats, och det har fjärrautentiserats, kommer den kortlivade token som utfärdats av Apples servrar att omvandlas till en iCloud-token som tillåter inloggning till iCloud. Efter detta återskapas elevens inställningar och dokument och data synkroniseras från iCloud.

Dokument och data lagras på iCloud allteftersom de skapas eller ändras när elevsessionen är aktiv och enheten är ansluten. Utöver detta finns det en funktion för synkronisering i bakgrunden som ser till att ändringar synkroniseras med iCloud direkt när eleven loggar ut. När användarens bakgrundssynkronisering är klar blir användarens APFS-volym avlänkad och kan inte länkas in igen utan att ange användarens inloggningsuppgifter.

Logga ut från Delad iPad

När en elev loggar ut från Delad iPad blir elevens användarnyckelsamling omedelbart låst och alla appar avslutas. För att snabba upp inloggningen för en ny elev blir vissa vanliga utloggningsåtgärder tillfälligt pausade av systemet och ett inloggningsfönster visas för den nya eleven. Om en elev loggar in under detta tidsintervall (ca 30 sekunder) slutför Delad iPad den pausade utloggningen som en del av inloggningen på det nya elevkontot. Om Delad iPad förblir overksam utlöser detta den pausade utloggningen. Under utloggningsfasen blir inloggningsfönstret omstartat som om en annan utloggning hade inträffat.

Uppgraderingar av Delad iPad

När en Delad iPad uppgraderas från en version som är äldre än iOS 10.3 till 10.3 eller nyare sker en filsystemskonvertering för att konvertera HFS+-datapartitionen till en APFS-volym. Om någon eller några användares hemkataloger finns i systemet vid konverteringen stannar de kvar på den primära datavolymen istället för att konverteras till enskilda APFS-volymer.

När fler elever loggar in placeras även deras hemkataloger på den primära datavolymen. Nya användarkonton skapas inte på egna APFS-volymer, på det sätt som beskrivits tidigare, förrän alla användarkonton på den primära datavolymen har raderats. För att se till att alla användare får den extra säkerhet och de lagringsutrymmen som blir möjliga med APFS bör iPad antingen uppgraderas till 10.3 eller senare via en radering och ominstallation, eller alla användarkonton på enheten raderas via MDM-kommandot Delete User.

Mer information om MDM finns på:
https://help.apple.com/deployment/mdm/#/cad7e2e0cf56

Apple School Manager

Apple School Manager är en tjänst för utbildningsinstitutioner som gör det möjligt att köpa innehåll, ställa in automatisk enhetsregistrering i MDM-lösningar, skapa konton för elever och personal och arbeta med iTunes U-kurser. Apple School Manager är tillgängligt via webben och är avsett för teknikansvariga och IT-administratörer, personal och lärare.

Du hittar mer information om Apple School Manager på
https://help.apple.com/schoolmanager/

Apple Business Manager

Apple Business Manager är en enkel, webbaserad portal som IT-administratörer kan använda till att driftsätta iOS-, macOS- och tvOS-enheter från ett ställe. När den används tillsammans med en MDM-lösning kan du konfigurera enhetsinställningar samt köpa och distribuera appar och böcker. Apple Business Manager är tillgänglig via webben och är avsedd för IT-administratörer.

Du hittar mer information om Apple Business Manager på
https://help.apple.com/businessmanager/

Enhetsregistrering

Apple School Manager och Apple Business Manager erbjuder ett snabbt och effektivt sätt att driftsätta iOS-enheter som en organisation har köpt direkt från Apple eller från auktoriserade Apple-återförsäljare och operatörer som deltar. Enheter med iOS 11 eller senare och tvOS 10.2 eller senare kan också läggas till i Apple School Manager och Apple Business Manager efter köpet med hjälp av Apple Configurator 2.

Organisationer kan automatiskt registrera enheterna i MDM utan att behöva förbereda eller ens röra vid själva enheterna innan användarna får dem. När organisationen har gått med i ett av programmen kan administratörer logga in på programmets webbplats och koppla programmet till den egna MDM-lösningen. Enheterna de köpt kan sedan

tilldelas till användarna via MDM. Under enhetskonfigurationsprocessen kan säkerheten för känsliga data ökas genom att se till att lämpliga säkerhetsåtgärder vidtas. Exempelvis:

- Låt användare autentisera som en del av det inledande inställningsflödet i Apple-enhetens inställningsassistent under aktivering.
- Tillhandahåll en preliminär konfiguration med begränsad tillgång och kräv ytterligare konfiguration av enheten innan den får tillgång till känsliga data.

När en användare har tilldelats en enhet installeras MDM-angivna konfigurationer, begränsningar och inställningar automatiskt. All kommunikation mellan enheter och Apple-servrar krypteras via HTTPS (SSL) under överföring.

Installationsprocessen för användarna kan förenklas ytterligare genom att ta bort vissa steg i inställningsassistenten för iOS, tvOS och macOS så att användarna kommer igång snabbt. Administratörer kan också bestämma om användaren ska kunna ta bort MDM-profilen från enheten eller inte och se till att enhetsbegränsningar används redan från början. När användarna har packat upp och aktiverat sina enheter kan de registreras i organisationens MDM-lösning så att alla inställningar, appar och böcker som krävs finns på plats och är installerade.

Apple Configurator 2

Utöver MDM finns Apple Configurator 2 för macOS som gör det enkelt att ställa in och förkonfigurera iOS- och Apple TV-enheter innan de delas ut till användare. Med Apple Configurator 2 kan enheter snabbt förkonfigureras med appar, data, begränsningar och inställningar.

Med Apple Configurator 2 kan du använda Apple School Manager eller Apple Business Manager till att registrera enheter i en MDM-lösning så att användare inte behöver använda inställningsassistenten. Apple Configurator 2 kan också användas till att lägga till iOS- och Apple TV-enheter i Apple School Manager eller Apple Business Manager efter köpet.

Mer information om Apple Configurator 2 finns på https://help.apple.com/configurator/mac/

Övervakning

När en enhet ställs in kan en organisation ange att enheten ska övervakas. Övervakning innebär att enheten ägs av en institution, vilket ger ytterligare kontroll över dess konfiguration och begränsningar. Med Apple School Manager eller Apple Business Manager kan övervakning aktiveras trådlöst på enheten som en del av MDM-registreringsprocessen eller aktiveras manuellt med Apple Configurator 2. Övervakning av en enhet kräver att enheten raderas och att operativsystemet installeras om.

Mer information om konfigurering och hantering av iOS-enheter och Apple TV med MDM eller Apple Configurator 2 finns på https://help.apple.com/deployment/ios/

Begränsningar

Begränsningar kan aktiveras, eller i vissa fall avaktiveras, av administratörer som vill förhindra att användare kommer åt enskilda appar, tjänster eller funktioner på enheten. Begränsningar skickas till enheter i en begränsningsnyttolast som bifogas i en konfigurationsprofil. Begränsningar kan användas på enheter med iOS, tvOS och macOS. En del begränsningar på en hanterad iPhone kan speglas på en parkopplad Apple Watch.

Du kan se en aktuell lista för IT-chefer på https://help.apple.com/deployment/mdm/#/mdm0F7DD3D8

Fjärradering

iOS-enheter kan fjärraderas av en administratör eller användare. Direkt fjärradering uppnås genom säker borttagning av krypteringsnyckeln för blocklagring från det raderingsbara lagringsutrymmet, vilket gör alla data oläsbara. Fjärraderingskommandot kan initieras av MDM, Exchange eller iCloud.

När ett fjärraderingskommando skickas från MDM eller iCloud svarar enheten med en bekräftelse och utför sedan raderingen. Om fjärraderingen utförs via Exchange stämmer enheten av med Exchange-servern innan den utför raderingen.

Användarna kan också radera enheter de har i sin ägo med hjälp av appen Inställningar. Som tidigare har nämnts kan enheterna ställas in på att raderas automatiskt efter en serie misslyckade lösenkodsförsök.

Förlorat läge

Om en enhet försvinner eller blir stulen kan en MDM-administratör fjärraktivera förlorat läge på en övervakad enhet som har iOS 9.3 eller senare. När förlorat läge är aktiverat loggas den aktuella användaren ut och enheten kan inte låsas upp. På skärmen visas ett meddelande som administratören kan anpassa. Det kan t.ex. visa ett telefonnummer att ringa till om enheten hittas. När en enhet försätts i förlorat läge kan administratören begära att enheten skickar information om sin aktuella plats och välja att spela upp ett ljud. När en administratör stänger av förlorat läge, som är det enda sättet som läget kan avaktiveras på, får användaren veta detta genom att ett meddelande visas på låsskärmen eller en notis på hemskärmen.

Aktiveringslås

När Hitta min iPhone är aktiverat kan enheten inte återaktiveras utan att ange ägarens Apple-ID och lösenord eller enhetens föregående lösenkod.

Vi rekommenderar att du övervakar organisationsägda enheter så att organisationen kan hantera aktiveringslåset. Då slipper du förlita dig på att en enskild användare ska ange inloggningsuppgifterna till sitt Apple-ID för att återaktivera enheterna.

En kompatibel MDM-lösning kan spara en förbigångskod på övervakade enheter när aktiveringslåset är på, och senare använda denna kod till att automatiskt rensa aktiveringslåset när enheten behöver raderas och tilldelas till en ny användare.

Som förval har övervakade enheter aldrig aktiveringslåset på, även om användaren slår på Hitta min iPhone. En MDM-lösning kan däremot hämta en förbigångskod och tillåta att aktiveringslås aktiveras på enheten. Om Hitta min iPhone är på när MDM-lösningen slår på aktiveringslåset aktiveras funktionen i samband med detta. Om Hitta min iPhone är av när MDM-servern slår på aktiveringslåset aktiveras funktionen nästa gång användaren slår på Hitta min iPhone.

På enheter som används inom utbildning med ett hanterat Apple-ID som skapats i Apple School Manager kan aktiveringslåset kopplas till en administratörs Apple-ID istället för till användarens Apple-ID. Det kan också avaktiveras med enhetens förbigångskod.

Skärmtid

Skärmtid är en funktion i iOS 12 som hjälper användare att förstå och ta kontroll över sin egen eller sina barns app- och webbanvändning. Användare kan:

- Visa användningsinformation.
- Ställa in begränsningar för app- eller webbanvändning.
- Konfigurera Skärmfri tid.
- Genomdriva ytterligare begränsningar.

En användare som hanterar sin egen enhetsanvändning kan synkronisera begränsningar och användningsdata för Skärmtid mellan enheter som är associerade till samma iCloud-konto med CloudKits E2E-kryptering. Det förutsätter att användarens konto har tvåfaktorsautentisering aktiverad (synkronisering är avstängd som förval). Skärmtid ersätter funktionen Begränsningar i tidigare versioner av iOS.

När en användare rensar historiken i Safari eller raderar en app tas motsvarande användningsdata bort från enheten och alla synkroniserade enheter.

Föräldrar och Skärmtid

Föräldrar kan även använda Skärmtid på iOS-enheter i syfte att förstå och ta kontroll över sina barns användning. En förälder som är en familjesamordnare (i iCloud-familjedelning) kan se användningsdata och hantera inställningar för Skärmtid för sina barn. Barn blir meddelade när föräldrar slår på Skärmtid och kan också övervaka sin egen användning. När föräldrar slår på Skärmtid för sina barn kan föräldrarna ställa in en lösenkod så att barnen inte kan göra ändringar. På sin artonårsdag (beroende på land eller region) kan barn själva stänga av den här övervakningen.

Användningsdata och konfigurationsinställningar överförs mellan förälderns och barnets enheter via en E2E-krypterad anslutning till IDS (Apple Identity Service). Krypterade data kan tillfälligt lagras på IDS-servrar tills de kan läsas av den mottagande enheten (exempelvis så fort en avslagen iPhone eller iPad slås på igen). Dessa data kan inte läsas av Apple.

Skärmtidsanalys

Om användaren slår på Dela iPhone- och Watch-analys blir endast följande anonymiserade data insamlade så att Apple får en bättre förståelse för hur Skärmtid används:

- Om Skärmtid aktiverades i inställningsassistenten eller senare i Inställningar
- Om Skärmtid är på
- Om Skärmfri tid är aktiverad
- Antal gånger som förfrågan Be om mer tid användes
- Antal appbegränsningar

Apple samlar inte in några specifika app- eller användningsdata. När en användare ser en lista med appar i Skärmtids användningsinformation hämtas appsymbolerna direkt från App Store som inte behåller några data gällande de här förfrågningarna.

Integritetsinställningar

Apple tar kundernas integritet på allvar och har många inbyggda inställningar och alternativ som gör att iOS-användarna kan bestämma hur och när appar använder deras information, samt vilken information som används.

Platstjänster

Platstjänster använder GPS, Bluetooth och crowdsource-insamlade platsdata för Wi-Fi-nätverk och mobiltelefonmaster till att fastställa din ungefärliga plats. Platstjänster kan stängas av med ett enkelt reglage i Inställningar, eller så kan användarna godkänna platstjänster för varje separat app. En app kan begära att få ta emot platsdata endast när appen används eller när som helst. Användarna kan välja att neka detta, och de kan ändra inställningen när de vill i Inställningar. I Inställningar kan tillgången ställas in som aldrig tillåten, tillåten vid användning eller alltid, beroende på vad appen ska använda platsinformationen till. Om appar som har fått ständig behörighet till platsinformation drar nytta av den behörigheten när de är i bakgrundsläge påminns användaren om att han/hon har godkänt detta och kan ändra inställningen.

Dessutom kan användarna finjustera hur systemtjänster använder platsinformation. Det kan till exempel handla om att kunna stänga av att platsinformation tas med i informationen som samlas in av analystjänster som Apple använder för att förbättra iOS, platsbaserad Siri-information, platsbaserade sammanhang för sökningar med Siri-förslag, lokala trafikförhållanden och viktiga platser som har besökts tidigare.

Tillgång till personliga data

iOS hjälper till att förhindra att appar kommer åt användarens personliga information utan behörighet. Användaren kan öppna Inställningar och se vilka appar han/hon har gett behörighet att komma åt viss information, samt godkänna fortsatt tillgång eller återkalla den. Det innefattar tillgång till:

- Kontakter
- Kalendrar
- Påminnelser
- Bilder
- Rörelseaktivitet och kondition
- Platstjänster
- Apple Music
- Din musik- och videoaktivitet
- Mikrofon
- Kamera
- HomeKit
- Hälsa
- Röstigenkänning
- Bluetooth-delning
- Ditt mediebibliotek

Om användaren loggar in på iCloud får apparna tillgång till iCloud Drive som förval. Användaren kan styra tillgången per app under iCloud i Inställningar. Dessutom ger iOS möjlighet att begränsa dataöverföring mellan appar och konton som har installerats av en MDM-lösning och de som har installerats av användaren.

Integritetspolicy

Du kan läsa Apples integritetspolicy på
https://www.apple.com/se/legal/privacy

Säkerhetscertifieringar och program

Obs! Den senaste informationen om iOS-säkerhetscertifieringar, verifieringar och råd finns på https://support.apple.com/sv-se/HT202739

ISO 27001- och 27018-certifieringar

Apple har erhållit ISO 27001- och ISO 27018-certifiering för säkerhetshanteringssystemet för information för infrastruktur, utveckling och åtgärder som stöder dessa produkter och tjänster: Apple School Manager, iTunes U, iCloud, iMessage, FaceTime, hanterade Apple-ID:n, Siri och Skolarbete i enlighet med Statement of Applicability v2.1, daterat 11 juli 2017. Att Apple efterlever ISO-standarder certifierades av British Standards Institution. BSI-webbplatsen har certifikat om överensstämmelse med ISO 27001 och ISO 27018. Du hittar certifikaten på

https://www.bsigroup.com/en-GB/our-services/certification/certificate-and-client-directory/search-results/?searchkey=company=apple&licencenumber=IS+649475

https://www.bsigroup.com/en-GB/our-services/certification/certificate-and-client-directory/search-results/?searchkey=company=Apple&licencenumber=PII%20673269

Kryptografisk verifiering (FIPS 140-2)

De kryptografiska modulerna i iOS har granskats upprepade gånger för att kontrollera att de uppfyller kraven i U.S. Federal Information Processing Standards (FIPS) 140-2 efter varje version sedan iOS 6. I likhet med varje större versionssläpp skickar Apple in modulerna till CMVP för återvalidering när iOS-operativsystemet släpps. Detta program validerar integriteten hos de kryptografiska funktionerna i appar från Apple och tredje part som använder de kryptografiska tjänsterna och godkända algoritmerna i iOS på rätt sätt.

Apple erhöll FIPS 140-2-validering för den inbäddade maskinvarumodulen med namnet Apple Secure Enclave Processor (SEP) Secure Key Store (SKS) Cryptographic Module som möjliggjorde godkänd användning av SEP-genererade och -hanterade nycklar. Apple arbetar vidare i sin strävan efter högre nivåer för maskinvarumodulen i samband med släppet av varje ny större version av iOS när så är lämpligt.

Common Criteria Certification (ISO 15408)

Sedan släppet av iOS 9 har Apple erhållit ISO-certifieringar för varje större iOS-släpp under Common Criteria Certification-programmet och har utökat omfattningen till att inkludera följande:

- Mobile Device Fundamental Protection Profile
  - Utökat paket för MDM-agenter
  - Utökat paket för trådlösa LAN-klienter
  - PP-modul för VPN-klient

- Protection Profile for Application Software
  - Utökat paket för webbläsare

iOS 12 förväntas innehålla ytterligare certifieringar för följande:

- Utökat paket för e-postklienter

Apple planerar att utöka omfattningen i samband med släppet av varje ny större version av iOS.

Apple deltar aktivt inom ITC (International Technical Community) för att ta fram ännu inte tillgängliga gemensamma skyddsprofiler (Collaborative Protection Profiles, cPPs) med fokus på att utvärdera viktig säkerhetsteknik för mobiler. Apple fortsätter att utvärdera och sikta mot certifieringar enligt de nya och uppdaterade versioner av cPPs som finns idag och som är under utveckling.

Commercial Solutions for Classified (CSfC)

I förekommande fall har Apple även skickat in iOS-plattformen och diverse tjänster till Commercial Solutions for Classified (CSfC) för att de ska tas med på CSfC:s programkomponentlista. När Apples plattformar och tjänster genomgår certifiering enligt Common Criteria Certifications skickas de även in för att tas med på CSfC:s programkomponentlista.

Du kan se de senast listade komponenterna på https://www.nsa.gov/resources/everyone/csfc/components-list/

Säkerhetskonfigurationsriktlinjer

Apple har samarbetat med myndigheter i hela världen för att ta fram riktlinjer som innehåller anvisningar och rekommendationer för att upprätthålla en säkrare miljö, så kallad enhetshärdning för högriskmiljöer. Dessa riktlinjer innehåller väldefinierad och granskad information om hur man konfigurerar och använder inbyggda funktioner i iOS för ökat skydd.

Apples säkerhetsbelöning

Apple belönar personer som meddelar Apple om allvarliga problem som de har hittat. För att kunna få en av Apples säkerhetsbelöningar måste personen i fråga presentera en tydlig rapport samt ett hållbart POC (Proof of Concept). Sårbarheten måste påverka den senast lanserade iOS-versionen och eventuellt den senaste maskinvaran. Den exakta summan fastställs när Apple har granskat all information. Villkoren innefattar bland annat problemets nyhetsvärde, sannolikheten för att problemet upptäcks samt hur mycket användaren måste göra för att utlösa problemet.

När Apple har fått tillgång till probleminformationen prioriterar Apple att lösa bekräftade problem så fort som möjligt och ger ett offentligt erkännande, när det är lämpligt, med undantag för om användaren vill vara anonym.

Kategori	Högsta utbetalning (USD)
Komponenter för fast programvara, säker startprocess	200 000 $
Extrahering av konfidentiellt material skyddat av Secure Enclave	100 000 $
Körning av valfri kod med kärnbehörigheter	50 000 $
Icke-auktoriserad åtkomst till iCloud-kontodata på Apple-servrar	50 000 $
Åtkomst från en sandlådeprocess till användardata utanför den aktuella sandlådan	25 000 $

Sammanfattning

Vårt engagemang för säkerhet

Vi på Apple arbetar för våra kunders säkerhet genom att använda ledande integritets- och säkerhetsteknik som har utformats för att skydda personlig information, samt heltäckande lösningar för att skydda affärsdata i företagsmiljöer.

Säkerheten är inbyggd i iOS. Allting ett företag behöver – från plattformen till nätverket och apparna – finns inbyggt i iOS. Tillsammans ger de här komponenterna branschledande säkerhet i iOS, utan att kompromissa med användarupplevelsen.

Apple använder en konsekvent, integrerad infrastruktur för säkerhet genomgående i hela iOS och i ekosystemet av iOS-appar. Maskinvarubaserad lagringskryptering ger möjlighet att fjärradera förlorade enheter och gör det dessutom möjligt för användarna att radera alla företagsdata och all personlig information helt och hållet från enheten om den ska byta ägare. Diagnosinformation samlas också in anonymt.

Apples iOS-appar har skapats med förbättrad säkerhet som en av grundpelarna. iMessage och FaceTime har exempelvis klient-till-klient-kryptering. När det gäller tredjepartsappar skyddas användaren av obligatorisk kodsignering, körning i sandlåda och begränsade rättigheter som tillsammans ger ett branschledande skydd mot virus, skadeprogram och andra attacker. Granskningen av appar som skickas in till App Store skyddar också användarna mot säkerhetsrisker eftersom alla iOS-appar kontrolleras innan de släpps till användare.

För att få ut det mesta av de omfattande inbyggda säkerhetsfunktionerna i iOS uppmuntras företag att granska sina IT- och säkerhetspolicyer och se till att använda sig av alla de lager av säkerhet som iOS-plattformen erbjuder.

Apple har ett särskilt säkerhetsteam som deltar i arbetet med alla Apple-produkter. Teamet granskar och testar säkerheten hos produkter under utveckling och även färdiga produkter. Apple-teamet tillhandahåller också säkerhetsverktyg och utbildning, samt söker aktivt efter rapporter om nya säkerhetsproblem och hot. Apple är medlem i FIRST – Forum of Incident Response and Security Teams.

Mer information om att rapportera problem till Apple och prenumerera på säkerhetsmeddelanden hittar du på:
https://www.apple.com/se/support/security

Ordlista

APNs (Apples tjänst för pushnotiser)	En världsomspännande tjänst från Apple som levererar pushnotiser till iOS-enheter.
ASLR (address space layout randomization)	En teknik som iOS använder för att göra det svårare att utnyttja eventuella buggar i programvaran. Genom att se till att minnesadresser och -förskjutningar är oförutsägbara förhindras skadlig kod från att hårdkoda dessa värden. I iOS 5 och senare är alla systemappar och bibliotek slumpmässigt placerade samt tredjepartsappar kompilerade som positionsoberoende körbara filer.
Boot ROM (startminne)	Den första kod som körs av enhetens processor när den startar. Eftersom den är integrerad i processorn kan den inte ändras, varken av Apple eller någon annan.
BPR (Boot Progress Register)	En samling SoC-maskinvaruflaggor som programvara kan använda till att spåra de startlägen som enheten befinner sig i, som DFU-läge och återhämtningsläge. När en BPR-flagga (Boot Progress Register) anges kan den inte rensas efteråt. Det här gör det möjligt för senare programvaror att få en tillförlitlig indikator för systemets status.
Dataskydd	Mekanismer för skydd av filer och nyckelringar i iOS. Det kan också syfta på de API:er som appar använder till att skydda filer och nyckelringsobjekt.
DFU-läge (Device Firmware Upgrade)	Ett läge där enhetens Boot ROM-kod väntar på att återskapas via USB. Skärmen är svart i DFU-läge, men vid anslutning till en dator med iTunes visas följande meddelande: ”iTunes har upptäckt en iPad i återhämtningsläge. Du måste återställa denna iPad innan den kan användas med iTunes.”
ECDHE (Elliptic Curve Diffie-Hellman Exchange)	ECDHE med tillfälliga nycklar. Med ECDHE kan två parter komma överens om en hemlig nyckel på ett sätt som förhindrar att nyckeln upptäcks av någon som avlyssnar meddelandena mellan de båda parterna.
ECDSA	En digital signaturalgoritm som baseras på kryptering med elliptiska kurvor.
ECID (Exclusive Chip Identification)	En 64-bitars identifierare som är unik för processorn i varje iOS-enhet. När ett samtal besvaras på en enhet slutar det ringa på enheter som är iCloud-parkopplade i närheten genom att en kort annonsering visas via Bluetooth LE 4.0. Annonseringen krypteras på samma sätt som Handoff-annonseringar. Den används under anpassningen av enheten och betraktas inte som en hemlighet.
filnyckel	256-bitars AES-nyckel som används till att kryptera en fil i filsystemet. Filnyckeln paketeras av en klassnyckel och sparas i filens metadata.
filsystemsnyckel	Den nyckel som krypterar varje fils metadata, inklusive dess klassnyckel. Den förvaras i det raderingsbara lagringsutrymmet för möjlighet till snabb radering snarare än konfidentialitet.
grupp-ID (GID)	Som UID, men gemensamt för alla processorer i en klass.
HSM-modul (Hardware Security Module)	En specialiserad, manipuleringssäker komponent som skyddar och hanterar digitala nycklar.
iBoot	Kod som läser in XNU som en del av den säkra startsekvensen. Beroende på SoC-generation kan iBoot läsas in av LLB eller direkt av startminnet.
IDS (Apple Identity Service)	Apples katalog över publika iMessage-nycklar, APNs-adresser samt telefonnummer och e-postadresser som används till att söka efter nycklar och enhetsadresser.
Integrerad krets (IC)	Kallas också mikrochip.
Integritetsskydd för systemcoprocessor (SCIP)	Systemcoprocessorer är processorer som finns på samma SoC som approcessorn.
JTAG (Joint Test Action Group)	Vanligt felsökningsverktyg för maskinvara som används av programmerare och kretsutvecklare.

LLB (Low-Level Bootloader)	I system med en startarkitektur i två steg är det här kod som anropas av Boot ROM, och som i sin tur läser in iBoot, som en del av den säkra startsekvensen.
minnesstyrkrets	256-bitars AES-nyckel som används till att kryptera en fil i filsystemet. Filnyckeln paketeras av en klassnyckel och sparas i filens metadata.
nyckelgenerering	Den process då en användares lösenkod görs om till en kryptografisk nyckel och förstärks med enhetens UID. Detta gör att automatiserade intrångsförsök måste utföras på en given enhet, vilket begränsar hur ofta attackerna kan utföras och förhindrar att de utförs parallellt. Algoritmen för nyckelgenerering är PBKDF2. Den använder AES krypterad med enhetens UID som PRF-funktion (pseudorandom function) för varje iteration.
nyckelpaketering	Kryptering av en nyckel med en annan. iOS använder NIST AES-nyckelpaketering, enligt RFC 3394.
Nyckelring	Den infrastruktur och den uppsättning API:er som används av iOS och tredjepartsappar till att lagra och hämta lösenord, nycklar och andra känsliga inloggningsuppgifter.
nyckelsamling	En datastruktur som används till att lagra en samling klassnycklar. Varje typ (användare, enhet, system, säkerhetskopiering, deponering samt iCloud-säkerhetskopiering) har samma format: • En rubrik som innehåller: – Version (satt till tre i iOS 5) – Typ (system, säkerhetskopiering, deponering eller iCloud-säkerhetskopiering) – Nyckelsamlingens UUID – En HMAC om nyckelsamlingen är signerad – Den metod som används för paketering av klassnycklarna: tillsammans med UID eller PBKDF2, eller tillsammans med salt och iterationsantal • En lista över klassnycklar: – Nyckelns UUID – Klass (vilken dataskyddsklass för filer eller nyckelringar detta är) – Typ av paketering (endast UID-härledd nyckel, UID-härledd nyckel och lösenkodshärledd nyckel) – Paketerad klassnyckel – Publik nyckel för asymmetriska klasser
programvarustartbitar	Dedikerade bitar i Secure Enclaves AES-motor som bifogas i UID:t när nycklar genereras från UID:t. Varje programvarustartbit har en motsvarande låsbit. Secure Enclaves Boot ROM och OS kan oberoende av varandra ändra värdet för varje programvarustartbit så länge som dess motsvarande låsbit inte har ställts in. När låsbiten har ställts in kan varken programvarustartbiten eller låsbiten ändras. Programvarustartbitar och deras lås nollställs när Secure Enclave startar om.
Raderingsbart lagringsutrymme	Ett dedikerat NAND-lagringsutrymme för lagring av kryptografiska nycklar, som kan anropas direkt och raderas säkert. Det ger inget skydd mot angripare som har fysisk tillgång till enheten. Däremot kan nycklarna i det raderingsbara lagringsutrymmet användas i nyckelhierarkier för snabb radering och förebyggande säkerhet.
ridge flow angle mapping	En matematisk representation av riktningen och bredden på de linjer som extraheras från en del av ett fingeravtryck.
SoC (System on Chip)	En integrerad krets (IC) där flera komponenter är samlade på en krets. Approcessorn, Secure Enclave och andra coprocessorer är komponenter i SoC.
Tillhandahållandeprofil	En plist som är signerad av Apple och innehåller en uppsättning entiteter och rättigheter som tillåter att appar installeras och testas på en iOS-enhet. En tillhandahållandeprofil för utveckling listar de enheter som utvecklaren har valt för specialdistribution, och en tillhandahållandeprofil för distribution innehåller app-ID:t för en företagsutvecklad app.

UID (unikt ID)	En 256-bitars AES-nyckel som bränns in i processorn vid tillverkningen. Den kan inte läsas av fast programvara eller programvara och den används endast av processorns maskinvarubaserade AES-motor. För att komma åt den faktiska nyckeln måste en angripare utföra en mycket avancerad och kostsam fysisk attack mot processorkretsen. UID:t har inget samband med någon annan identitetsmärkning på enheten (inklusive men inte begränsat till UDID:t).
URI (Uniform Resource Identifier)	En teckensträng som identifierar en webbaserad resurs.
XNU	Kärnan i operativsystemen iOS och macOS. Den förutsätts vara betrodd och kräver säkerhetsåtgärder som kodsignering, sandlåda, rättighetskontroller och ASLR.

Dokumentets versionshistorik

Datum	Sammanfattning
November 2018	**Uppdaterat för iOS 12.1** • FaceTime-grupper
September 2018	**Uppdaterat för iOS 12** • Secure Enclave • OS-integritetsskydd • Expresskort med strömsparläge • DFU- och återhämtningsläge • HomeKit TV-fjärrkontrollstillbehör • Kontaktlösa kuponger • Studentkort • Siri-förslag • Genvägar i Siri • Appen Genvägar • Hantering av användarlösenord • Skärmtid • Säkerhetscertifieringar och program
Juli 2018	**Uppdaterat för iOS 11.4** • Biometriska policyer • HomeKit • Apple Pay • Kundchatt • Meddelanden på iCloud • Apple Business Manager
December 2017	**Uppdaterat för iOS 11.2** • Apple Pay Cash **Uppdaterat för iOS 11.1** • Säkerhetscertifieringar och program • Touch ID/Face ID • Delade anteckningar • CloudKit med E2E-kryptering • TLS • Apple Pay, betalning med Apple Pay på webben • Siri-förslag • Delad iPad Mer information om säkerhetsinnehållet i iOS 11 finns på https://support.apple.com/sv-se/HT208112

Datum	Sammanfattning
Juli 2017	**Uppdaterat för iOS 10.3** • System Enclave • Dataskydd på filnivå • Nyckelsamlingar • Säkerhetscertifieringar och program • SiriKit • HealthKit • Nätverkssäkerhet • Bluetooth • Delad iPad • Förlorat läge • Aktiveringslås • Integritetsinställningar Mer information om säkerhetsinnehållet i iOS 10.3 finns på https://support.apple.com/sv-se/HT207617
Mars 2017	**Uppdaterat för iOS 10** • Systemsäkerhet • Dataskyddsklasser • Säkerhetscertifieringar och program • HomeKit, ReplayKit, SiriKit • Apple Watch • Wi-Fi, VPN • Enkel inloggning • Apple Pay, betalning med Apple Pay på webben • Tillägg av kreditkort, bankkort och förbetalda kort • Safari-förslag Mer information om säkerhetsinnehållet i iOS 10 finns på https://support.apple.com/sv-se/HT207143
Maj 2016	**Uppdaterat för iOS 9.3** • Hanterat Apple-ID • Tvåfaktorsautentisering för Apple-ID • Nyckelsamlingar • Säkerhetscertifieringar • Förlorat läge, aktiveringslås • Säkra anteckningar • Apple School Manager, delade iPad-enheter Mer information om säkerhetsinnehållet i iOS 9.3 finns på https://support.apple.com/sv-se/HT206166

Datum	Sammanfattning
September 2015	**Uppdaterat för iOS 9** • Apple Watchs aktiveringslås • Lösenkodspolicyer • API-stöd för Touch ID • Dataskyddet på A8 använder AES-XTS • Nyckelsamlingar för obevakad programuppdatering • Certifieringsuppdateringar • Förtroendemodell för företagsappar • Dataskydd för Safari-bokmärken • App Transport Security • VPN-specifikationer • iCloud-fjärråtkomst för HomeKit • Apple Pay-bonuskort, Apple Pay-kortutfärdares app • Spotlight-indexering på enheten • iOS-parkopplingsmodell • Apple Configurator 2 • Begränsningar Mer information om säkerhetsinnehållet i iOS 9 finns på https://support.apple.com/HT205212

November 2018