УДК 004.622, 004.623, 004.632.4

СТІЙКІСТЬ ТА ЕФЕКТИВНІСТЬ КРИПТОГРАФІЧНИХ АЛГОРИТМІВ ЩО ВИКОРИСТОВУЮТЬСЯ У МОБІЛЬНИХ ПРИСТРОЯХ

DOI 10 36994 / 2788- 5518- 2022-02-04-21

Середа А.В. *Відкритий міжнародний університет розвитку людини «Україна», Київ, Україна. andrew.sereda@gmail.com*

Даценко І.П. *к.т.н. Відкритий міжнародний університет розвитку людини «Україна», Київ, Україна. docik_ivan@ukr.net*

Павленко В.І*. к.ф.-м.н. , доц. Відкритий міжнародний університет розвитку людини «Україна». Київ, Україна. pavlenko.v@i.ua*

Самарай В.П., *к.т.н., доц. Відкритий міжнародний університет розвитку людини, Київ, «Україна», Україна. samaraj@ukr.net*

Анотація: *у статті розглядаються новітні типи криптографічних алгоритмів у різних мобільних операційних системах(IOS, Android). Пропонується дослідити різні методи для оцінки стійкості та ефективності представлених алгоритмів. Криптоаналіз математичних моделей та фізичних приладів для захисту конфіденційної чи чутливої інформації.*

Ключові слова: *мобільні пристрої, криптографія, захист, безпека, стійкість, криптоаналіз, кібербезпека.*

STABILITY AND EFFICIENCY OF CRYPTOGRAPHIC ALGORITHMS USED IN MOBILE DEVICES

Andrey Sereda. *Open International University of Human Development “Ukraine”, Kyiv, Ukraine. andrew.sereda@gmail.com.*

Ivan Datsenko, *Ph.D. Open International University of Human Development “Ukraine”, Kyiv, Ukraine. docik_ivan@ukr.net.*

Volodymyr Pavlenko, *Ph.D., Ass Prof. Open International University of Human Development “Ukraine”, Kyiv, Ukraine. pavlenko.v@i.ua.*

Valerii Samarai, *Ph.D.,Ass Prof. pen Internationfl University of Human Development “Ukraine”, Kyiv, Ukraine. samaraj@ukr.net*

Abstrfct*: the article examines the latest types of cryptographic algorithms in various mobile operating systems (IOS, Android). Research describes how different operating systems protect user data; what vulnerabilities exist for now. Determine what is sensitive information for the user that should be carefully secured. Recommendations how to protect and increase security for mobile devices that are everywhere solving different tasks including digital actives. A lot of attention is paid to different technologies of protection, general principles of security of confidential data. Exploring existing levels of abstraction for design and analysis the stability or efficiency for cryptographic systems. Give a try to compare different environments with opposite ideology but one roots. Partial or complete breakdown, prompt response to vulnerability. The problem of disk space encryption and the choice of appropriate cryptographic algorithms by one or another platform are considered. It is proposed to investigate various methods for evaluating the stability and efficiency of the presented algorithms. Cryptanalysis of mathematical models and physical devices to protect confidential or sensitive information. An overview of end-to-end encryption and why it is important. Thoughts on further research on finding a universal method for checking the stability and efficiency, time or labor costs for the possibility of interfering with the security of the device under the condition of active floating or with the help of a hardware and software product are presented. At the end, conclusions and suggestions for improving the efficiency of the security system on any mobile device under the guidance of the operating system are presented. Everything invented by man or nature is subject to study and research in the possibility of obtaining benefits for humanity.*

Вступ

Однією із найважливіших характеристик криптографічного захисту інформації (КЗІ) вважається стійкість алгоритму до криптоаналізу. Тобто стійким вважається алгоритм, який для успішної атаки вимагає від противника недосяжних обчислювальних ресурсів, недосяжного обсягу перехоплених відкритих і зашифрованих повідомлень чи ж такого часу розкриття, що по його закінченню захищена інформація буде вже не актуальна, і т. д. У більшості випадків крипостійкість не можна математично довести, можна тільки довести уразливість криптографічного алгоритму[1].

Відомо [2], що існує декілька рівнів абстракції при проектуванні та аналізі стійкості криптографічних систем: абстрактна математична модель криптографічного перетворення (алгебраїчні та ймовірнісні моделі шифру), модель засобу криптографічного захисту інформації та криптосистеми реалізованій на конкретній програмно-апаратної платформі. Перші моделі переважно аналізуються методами класичного криптоаналізу, останні – криптоаналізу з використанням інформації з побічних каналів (англ. - side-channel cryptanalysis). Таким чином, у криптоаналізі побічними каналами досліджується вплив непрямої інформації, отриманої під час роботи криптосистеми, на ослаблення стійкості до часткового або повному зламу[2]. Очевидно, що методи криптоаналізу по побічним каналам залежить від архітектури та особливостей реалізації обчислювальної системи, у межах якої реалізовано засіб КЗІ.

На даний момент світовий ринок операційних систем для мобільних пристроїв з великим відривом очолюють двоє лідерів: Google Android та Apple iOS (**Рис. 1**)**.** У процесі еволюції обидві платформи сформували підходи до організації безпеки даних користувача. Кожна має свої переваги та недоліки. Закрите середовище розробки iOS та оперативне реагування на вразливість створюють довіру до Apple. Але досі Apple повністю не захищає користувачів від зламу та компрометації даних різними каналами.

Google за останні кілька років значно підвищила безпеку ОС Android за рахунок використання криптографічної функціональності. Але фахівці з кіберкриміналістики все ще можуть обійти захисні механізми Android та отримати доступ до даних.

Розглянемо детальніше кожну з операційних систем, розберемо методи захисту, позначимо сильні та слабкі сторони безпеки. У результаті сформуємо рекомендації як для користувачів, так і для розробників з обох платформ.

Виконання досліджень

Майже всі дані користувача мобільних пристроїв є конфіденційними: електронна пошта, повідомлення, біометрична інформація, фото- і відеоконтент, документи, місце розташування, паролі, історія роботи у браузері та багато іншого. Які саме дані найбільш конфіденційні і які техніка захисту використовуються зараз? Також розглянемо загальну модель загроз.

Можна виділити декілька ключових технік контролю доступу до даних:

- *Безпека та ізоляція ПО.* Сучасні операційні системи мобільних пристроїв чітко розмежовують доступ додатків і доступ користувача. Наприклад, сумнівні додатки не можуть отримати доступ до даних, на які не мають відповідних прав, дозволів. Для запуску довільного коду на пристрої необхідно виконати обхід встановленого обмеження: «джейлбрейк» для iOS або отримати рут-доступ для Android.
- *Доступ за допомогою кодів та біометрії.* Контроль доступу до інтерфейсуОС здійснюється переважно через пароль довільної сили, найчастіше він цифровий. Можна активувати паттерн (графічний ключ — послідовність) на екрані блокування. Все частіше застосовується біометричний доступ : через розпізнавання рис обличчя або відбиток пальця.

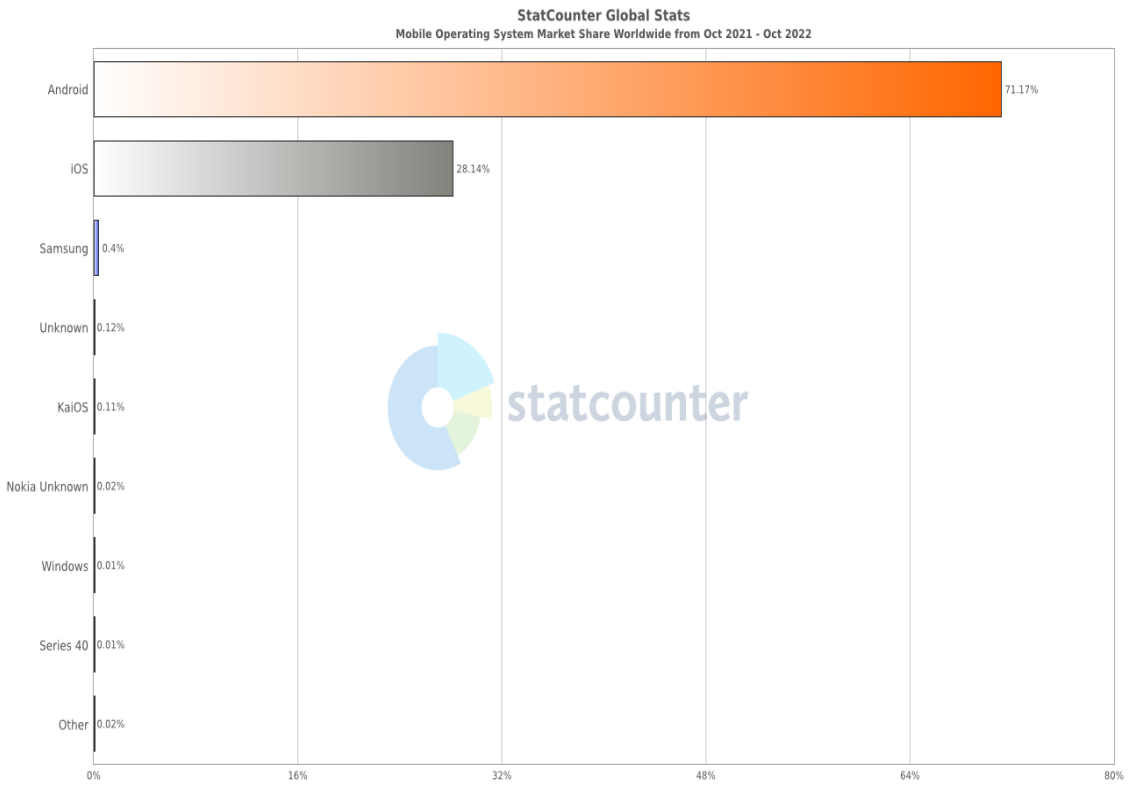

Рис. 1. Частка мобільних платформ у світі згідно statcounter.com

- *Шифрування файлів та дисків.* В разі обходу зловмисником програмних методів захисту він зіткнеться з зашифрованими даними. Це можуть бути, як окремі файли, так і розділи диска. Шифрування даних, як правило, організоване асиметрично: в обладнання вбудована одна частина ключа, а інша – генерується на основі вибраного користувачем паролю.
- *Безпечне апаратне забезпечення пристроїв.* Останнім часом виробники впроваджують співпроцесори безпеки та їх віртуалізовані аналоги. Це захищає від програмних атак і атак на обладнання пристрою, підсилюючи механізми шифрування конфіденційних даних, зокрема біометричних шаблонів.
- *Безпечне резервне копіювання та робота з хмаровими системами.* Зовсім недавно постачальники послуг почали пропонувати безпечну модель хмарного зберігання резервних копій даних з пристроїв. Сучасна система зашифрованого резервного копіювання унеможливлює прямий доступ хмарового провайдера, зловмисників чи правоохоронних органів до даних користувача

Модель загроз: Сучасні мобільні пристрої досить добре захищені від традиційних дистанційних атак. Але прямий фізичний доступ до пристрою (або його компонентів) протягом тривалого часу, значно спрощує роботу шахраїв або фахівців з кібербезпеки, що підвищує ризик зчитування інформації.

Прямий доступ може здійснюватися незаконно, (крадіжка, обман), або працівниками органів юстиції, кіберполіції законно, котрі виконують законодавчі приписи, що передбачають розсекречування паролей. Слід бути дуже обережним, оскільки не виключене використання державними чиновниками своєї службової компетентності та невігластва користувача. Це нашкодить в подальшому хорошій репутації пристрою. Треба завчасно перевіряти особисті документи та дозволи, бо шахраї часто користуються соціальною довірою, видають себе за співробітників правоохоронних, державних установ, і обізнані в інженерних техніках, легко проникають в бази мобільних пристроїв. Таке проникнення через застосування поля кіберкриміналістики допоможе не тільки скопіювати необхідну інформацію, але й мати токени доступу до безпечних хмарних сховищ, із пам'яті. Інформативні файли з мобільних пристроїв і

хмарних сховищ завжди містять в собі особисті дані, заволодіння якими становитиме загрозу конфіденційності користувачів.

Конфіденційність даних мобільних пристроїв.

Полюючи на злочинця**,** фахівці з кіберполіції дістають найбільш важливі категорії закритої інформації, яка допоможе розкрити злочин. Приміром за основу візьмемо:

- IMEI, MEID / ESN та інші показники абонента мобільного зв'язку;
- всі вихідні і вхідні дзвінки, залишені в журналах;
- списки контактів, календарні дати, нагадування і т. п.;
- відео та аудіо матеріали, файли з документами;
- різні види повідомлень (миттєві, MMS.SMS);
- адреса і вміст електронної скриньки;
- історія та закладки в браузері;
- координати місця знаходження та пересування;
- діяльність в соціальних мережах;
- SIM / UICC, провайдер, IMSI, MSISDN та інше.

Даний перелік не обов'язково має незаперечний характер. Частина відомостей може мати тимчасову актуальність. Але треба пам'ятати, що тісні мережеві зв'язки з родичами, колегами, друзями (особисті листування, розмови, коментарі) містять приватну інформацію та під час розголосу загрожують безпеці великої групи людей.

Компанія Apple, операційна система iOS.

Кількість працюючих гаджеті у всьому світі за підрахунками корпорації Apple в 2022 році перевищила 1, 5 млрд. одиниць. Це майже 46 % мобільних телефонів в США та інших країнах, що знаходяться під контролем операційної системи iOS. Постійне зростання кількості новинок Apple на руках у світової спільноти приваблює, як шахраїв, так і багхантерів (фахівців, що шукають проблемні місця, недоліки в програмах), які співпрацюють з корпорацією. На оплату послуг з тестування щорічно витрачається понад 2 млн доларів США.

Захист даних користувача iOS на сьогоднішній день

Основні елементи захисту формуються на тісній взаємодії Ключові елементи захисту сформовані за допомогою взаємозв'язку пристрою з хмарними технологіями.

Автентифікація

Користувачі фізично взаємодіють з пристроями через: біометричну аутентифікацію або код, що складається з літер та цифр. Переважно це пароль із шести символів, активований по замовчуванню. Рекомендується застосування довгих паролів, що складаються з цілих фраз (комбінацій слів, літер). Apple не радить повністю вимикати аутентифікацію, хоча така можливість існує. Кількаразові спроби набору паролю блокуються. Повторно ввести код дозволяється тільки через певний час. Вищий рівень захисту надають TouchID (датчик відбитку пальця) і FaceID (ідентифікація рис обличчя з допомогою камери).

Підписи кодів програм

Код, що виконується на iOS , простіше всього, обмежити з допомогою цифрового підпису, вбудованого на нижчому рівні (Boot ROM). В момент, коли підпис проходить перевірку, відбувається безпечна загрузка, яка гарантує тривалий захист від ініціалізації підозрілого коду) і за рахунок підпису додатка, яка складається із частини, контрольованої Apple і сертифікату з відкритим ключем для масштабування системи. Існують також «сертифікати підпису установи», призначені для спеціалізованого запуску додатків на iOS.

Пісочниця та аналіз коду

Убезпечити користувацькі дані і API від проникнення в систему підозрілих додатків допомагає запровадження обмеженого доступу до простору пам'яті, файлів через встановлення ізольованого

середовища, так званої пісочниці. При підписанні маніфесту спеціально вказуються ресурси і служби широкого доступу. До них належить геолокація. Додатки із App Store підлягають автоматичній та ручній перевірці коду. Хоча навіть найстрогіші обмеження не дають стовідсоткової гарантії перепони проникненню чужих небажаних додатків. Вони можуть пройти перевірку і порушити винятковість прав користувача.

Шифрування

В Apple діє також надійна система шифрування даних пристрою Data Protection. Це на випадок коли шахраям якимось чином все ж вдасться обійти попередні захисні механізми, приміром, через логічні прорахунки чи недоліки обладнання. Використовуються криптографічні стандарти AES, ECDH over Curve25519 та інше, те що схвалене NIST(Національним інститутом стандартів і технологій) США. Всі дані прикріплені до пристрою і їх контролює сам користувач. Тільки йому відомий ключ шифрування — унікальна апаратна секретна комбінація UID (криптоключ) і пароль. Якщо пристрій перезапускається, то ключ шифрування треба встановлювати заново, тобто ввести раніше встановлений пароль і розблокувати ключі з допомогою біометрії. Заважає обійти шифрування метод обмеження пропозицій із збільшенням інтервалів часу та функцією отримання ключа на основі паролю. Apple застосовує такі «класів захисту» (Class key) шифрування на вибір розробників:

- Повний захист (CP) —прилади ключі шифрування видаляються через 10 секунд після блокування.
- Захищено тільки, коли закрито (PUO) — в пам’яті залишається відкритий ключ передачі зашифрованих файлів, коли пристрій заблокований. Повний захист включається (CP) після створення файлу, в момент його закриття.
- Захищено до першого розпізнавання користувача — початкового розблокування (AFU). Коли вводиться перший раз пароль, то ключі шифрування розблоковуються і залишаються в пам’яті пристрою навіть під час блокування.
- Захист відсутній (NP) —пристрій включений, шифрування здійснено тільки апаратними ключами UID і до них постійно є доступ в пам’яті, коли пристрій працює.

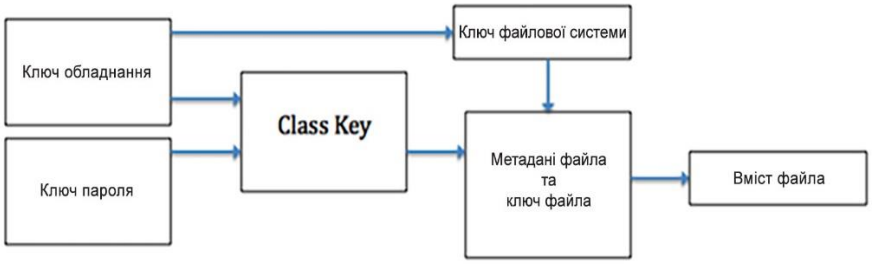

Рис. 2. Схема ієрархії ключів iOS Data Protection

Keychain

Це — захищене апаратними ключами та паролем користувача зашифроване сховище ключів та секретної інформації (логіни / паролі). Воно має загальнодоступний API. Опція Non-Migratory (NM) дає можливість провести дешифрування тільки на пристрої, де дані були зашифровані. В икористовується унікальний ідентифікатор UID.

Резервні копії

Резервне копіювання можна виконати на ПК або в iCloud. Локальний бекап шифрується вибраним паролем, так формується структура Keybag. Ненадійний пароль не зв'язаний з апаратними ключами і тому має слабке місце, а сааме, не стійкий до брутфорсу (перебору варіантів). Зазвичай iOS використовує 10 млн. ітерацій PBKDF2, щоб підвищити стійкість до

переборів. При копіюванні в Apple iCloud дані шифруються Curve25519, це дозволяє зробити резервне копіювання навіть при заблокованому пристрої.

Зашифровані ключі за допомогою iCloud Keychain, стають відомими Apple, тому виробники чи шахраї можуть отримати доступ до бекапу. Додатково Keychain шифрує обидва типи резервної копії ключем UID, щоб запобігти передачі даних на сторонній пристрій. Резервному копіюванню в iCloud підлягають: дані додатків; резервні копії Apple Watch; настройки пристроїв; SMS і MMS, фото і відео; історія покупок в сервісах Apple; домашній екран і iMessage; пароль голосової пошти.

SEP: TouchID і FaceID

Secure Enclave Processor (SEP) — співпроцесор, який використовує зашифровану пам'ять і організує роботу з автентифікації, керуванню ключами, шифруванню и генерації випадкових чисел. Якщо ядро iOS було зламано, то SEP забезпечить шифрування за рахунок власної автономної архітектури.

Зменшення фронту атаки

При заблокованому пристрої обмежуються шляхи вводів і виводів даних і сааме це зменшує фронт атаки. Щоб доставити експлойт на пристрій Apple , треба пройти одним із таких шляхів:

- синхронізацію даних в фоновому режимі;
- пуш-повідомлення;
- деякі повідомлення з мережі (Wi-Fi, Bluetooth, стільниковий зв'язок);
- порт Lightning на пристрої.

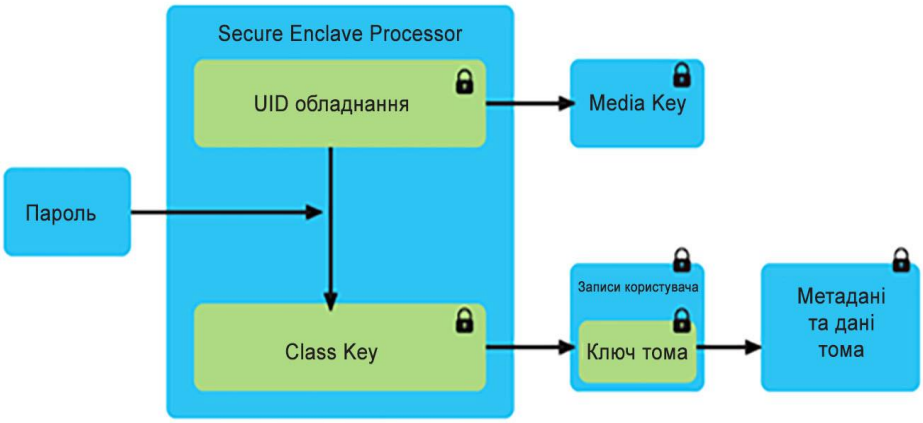

Рис. 3. Схема обробки ключа Secure Enclave Processor

Робота автентифікації TouchID / FaceID узгоджується через SEP.

Після введення USB Restricted операційна система почала обмежувати доступ несанкціонованих підключень через USB/Lightning, наприклад, інструментів комп’ютерної криміналістики. iOS зупиняє роботу USB після першого підключення, якщо пристрій не був зазначений користувачем, як довірений. Довірені підключення з використанням USB/Lightning залишаються в пам'яті на місяць.

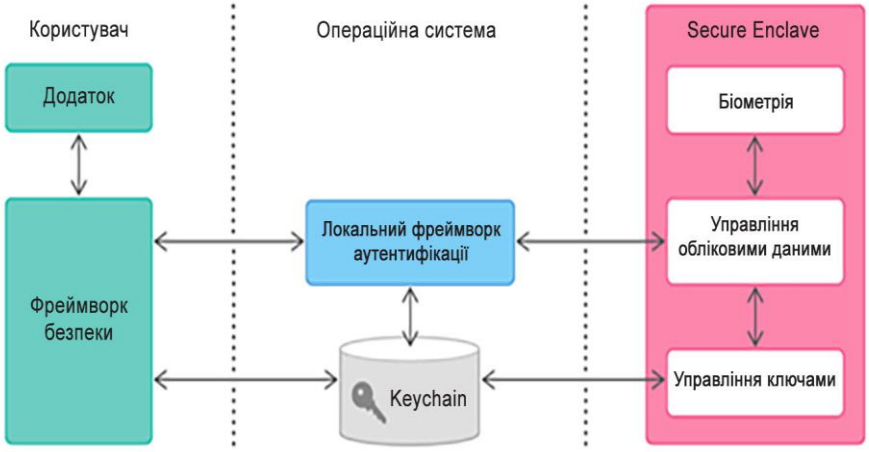

Рис. 4. Схема роботи Apple TouchID і FaceID

Message і FaceTime

За допомогою FaceTime, Apple підтримує наскрізне (end-to-end) шифрування відео- / аудіозв'язку між своїми пристроями. Дане шифрування базується на основі безпечного транспортного протоколу в режим реального часу(SRTP). В iMessage використовується схема шифрування «signcryption».

Публічні ключі використовують Apple Identity Service — як довірений орган, що доводить справжність користувачів.

Актуальні техніки обходу захисту користувацьких даних iOS

Як тільки в продажу з'являється нова серія пристроїв Apple з підвищеним рівнем безпеки, відразу ж вона стає цікавою для зловмисників, і фахівців з кіберкриміналістики і багхантерів, які отримують достойну платню за виявлення недоліків. Отже, назвемо кілька сучасних технік обходу захисту для iOS.

Джейлбрейк і програмні експлойти

Для усунення обмежень iOS найчастіше використовується джейлбрейк випускається під конкретний пристрій і версію iOS. Більшість інструментів криптоаналізу застосовує *Jailbreak* для доставки «корисного навантаження» в проблемний компонент програми, через USB/Lightning. Доставка здійснюється з допомогою повідомлень або спеціальних веб-сторінок. Ще користуються ланцюжком експлойтів. З початком контролю над ядром iOS , можна дістати зашифровані дані, Наступна модифікація (патчинг) програмного ядра дозволить запустити будь-який код в системі. Apple дбає про власну репутацію, тому вкрай швидко реагує на виявлення недоліків, які призводять до джейлбрейку, таким чином вдається тримати високу ціну сучасних робочих варіантів. Джейлбрейки checkm8 / checkra1n стали дуже ефективними в 2020 році в процесі кіберкриміналістичних досліджень, дозволяють працювати з пристроями до iPhone X в різних версіях iOS. Експлойти Cellebrite UFED Touch и 4PC – це можливість запуску резервного копіювання без авторизації користувача або активації загрузчика з кодом Cellebrite для вилучення частин файлової системи прист

Підбір паролю

Раніше зловмисники могли проводити більш успішні атаки за рахунок анулювання показників лічильника неправильно введених паролів, бо перевірка паролю контролювалася процесором самого пристрою. З появою архітектури SEP з'явилися обмеження на вгадування комбінації, і ймовірність зламу зменшилася В раніших версіях iOS для збою лічильника відключали живлення пристрою, а в 2016 році була застосована техніка віддзеркалювання флеш-пам'яті NVRAM на iPhone 5C і число вводу паролів стало необмеженим. (Табл.1) Обмеження на підбір в 80 мс робить

атаки з перебором безглуздими для складних паролів, що складаються з літері цифр. Для PIN-коду з 6 цифр, що використовується за замовчуванням, обмеження на вгадування є основною перешкодою.

Таблиця 1.
Оцінка часу перебору цифрових паролів в iOS

Довжина паролю	4 цифри	6 цифр	10 цифр
Всього варіантів	10^4 = 10 000	10^6 = 1 000 000	10^10
Всього використовується	9276	997090	-
80 мс/спроби очікування	12,37 хвилини 6,19 хвилини	22,16 години 11,08 години	~25 років
10 хв./спроба очікування	~70 днів ~35 днів	~20 років ~10 років	~200 000 років

Злам SEP

Для взаємодії с SEP необхідні привілеї ядра. Отже, *Jailbreak* – це складова частина ланцюжка для злому SEP. Якщо атака на SEP вдалася, то шахрай чи фахівець з кібераналізу мають необмежений доступ не тільки до ключів шифрування, а й до багатьох інших функцій захисту та можливість дістатися до інформації з будь якого файлу. Компанія Grayshift з 2018 р. винесла на ринок цікавий інструмент GrayKey, який ніби то зміг обходити обмеження SEP. Як доказ, було продемонстровано низку витоків інформації (вдалий злам пароля блокування екрану iPhone X). На початку 2020 року дізналися про витік з ФБР. За не підтвердженими даними GrayKey отримав доступ до заблокованого iPhone 11 Pro Max (iOS 13), невразливого для checkm8. Цей факт викликає появу багатьох питань, наприклад: чи було це робочим эксплойтом? Чи, може, це звичайна компрометація SEP?

Вилучення даних

Стратегічні плани вилучення даних бувають різні і залежать від того, включеним чи виключеним є пристрій на момент захоплення. Часто, коли відбувається конфіскація пристрою Apple, правоохоронці забирають також і суміжні пристрої MacBook чи Apple TV, щоб проаналізувати їх вміст, використовуючи клітку Фарадея або автономне джерело живлення для запобігання вимкнення приладів. У випадках, коли атаку на iOS, здійснюють зловмисники, останнім рубіжем захисту залишається шифрування від Data Protection. Тому витяг даних без згоди чи при відсутності користувача проходитиме по одному із трьох сценаріїв.

1. ***Доступ до приладу за допомогою ключів***. Ключі шифрування, що загрузилися в пам'ять, можна вилучити. Скажімо, інструменти криптоаналізу Cellebrite UFED і XRY Logical дають дозвіл підключитися до пристроїв iOS через порт USB/Lightning або через Bluetooth, а також здійснити резервне копіювання чи переглянути файли.
2. ***Обхід захисних механізмів***. Навіть недоступні для вилучення дані, в окремих випадках все ж вилучаються, з допомогою інструменту GrayKey. Він з успіхом справляється з завданням дешифрування кодів доступу користувача. Вдала високочастотна атака шляхом перебору паролю користувача дозволить вилучити всі файли iOS, Keychain і вміст iCloud. І результат не залежить від того в якому стані (AFU чи BFU) знаходився пристрій до початку атаки.
3. ***Альтернативні джерела даних.*** Застосовується атака на резервну копію iOS, що зберігається в локальному комп'ютері і має слабкий захист від брутфорсу. Також тут вилучаються дані користувача із хмарових сервісів.

Посилення захисту даних. Apple створила якісне програмне середовище (фреймворк) для захисту даних користувача, але поки Data Protection використовується в не дуже строгому режимі. Скажімо, токени автентифікації для хмарових сервісів можуть бути вилучені із пам'яті. Хоча клас захисту в стані AFU досить надійний але ретельнішої організації його роботи.

Динамічний захист даних. Досягається завдяки взаємодії з користувачем на основі навчальних алгоритмів разом з покращенням захисту до класу CP. iOS передбачає, які ключі і коли саме повинні бути застосовані. Неактуальні своєчасно видаляються з пам'яті, натомість вводяться нові.

Наскрізне шифрування бекапів і iCloud. Apple зберігає ключі, які дозволяють розшифрувати дані бекапів в iCloud. Під час використання iCloud Keychain, якісно організується наскрізне резервне копіювання, недоступне для Apple, але доступне для будь якого довіреного пристрою користувача. Це використовується і для контейнера CloudKit, та підвищує безпеку зберігання даних користувача в хмарі.

Паролі на локальні бекапи. Захистити від переборів, зможуть паролі локальних бекапів, які мають підвищену складність. Саме такі рекомендують створювати шляхом навчання користувача і оптимізації інтерфейсу.

Підсилення iCloud Keychain. Є ще одна слабинка це — кластер HSM, тому що відсутня прозорість шифрування зі сторони серверу. Apple зазвичай *застосовує технологію Certificate Transparency, це коли вузли перевіряють* адреси и коректність HSM, роблять інформацію публічною і діляться нею.

Обмеження на USB-інтерфейс. Строгіші обмеження під час керування USB/Lightning на рівні ядра iOS можуть підсилити безпеку і запобігати роботі експлойтів, наприклад, checkm8.

Обмеження на режими DFU і JTAG. Користувачі коли завгодно мають змогу перевести свій пристрій в режим DFU, і експлойти здатні використати цю шпаринку. Організується автентифікація з криптографією. Подібні рекомендації підходять і до режиму JTAG, хоча ним користується менша кількість людей.

Прозорість і функціональні протиріччя. З питань захисту користувацьких даних в Apple э протиріччя, які відразу треба усувати. Наприклад, включення / виключення опції синхронності даних з iCloud, заплутує користувача і з'являється ризик витоку даних. Те саме можна сказати і про інтерфейс, і про налаштування, і про управління iCloud за замовчуванням.

Компанія Google, операційна система Android

Android, або Android Open Source Project (AOSP), — це набір програмного забезпечення з відкритим початковим кодом, розробленим Open Handset Alliance під крилом Google. Мобільні сервіси Google (GMS), достатньо добре розширюють і ускладнюють систему Android, бо містять в собі пропрієтарні API и програмні сервіси. Зайнявши основну позицію на базі Linux, Android має всі переваги та недоліки безпеки даної ОС. Вразливість мобільних пристроїв Android завжди гостро відгукується у світі з огляду на популярність і поширення цих гаджетів.

Захист даних користувача Android на сьогоднішній день.

Автентифікація

Графічний ключ (патерн), цифровий код чи фраза із літер і цифр. Патерн-автентифікацію по силі прирівнюють до цифрового паролю з 2-3 цифр. Також іноді задіяна і біометрична автентифікація за відбитком пальця чи розпізнавання обличчя. Вона вводиться виробниками пристроїв за бажанням, бо для Android є не обов'язковою. Підключається опція Smart Lock, в цьому випадку для розблокування системою використовується інформація про оточення (телефон перебуває в сумці, лежить вдома).

Пісочниця для додатків

Застосовується дискреційний (вибірковий) контроль за доступом (DAC) и SELinux. Файли додатка мають дозвіл, який базується на ідентифікаторах користувачів Linux. Спроба відкрити не свій файл приведе до помилки. Додаткову ізоляцію забезпечує SELinux. Дана політика SELinux жорстко контролює привілеї додатків. Обов'язковий контроль доступ (MAC) дає змогу гарантувати, захищеність системи, навіть коли процес запуску проходить на правах суперкористувача (root).

Шифрування

Повне шифрування диску (починаючи з Android 4.4), коли задіяно модуль «dm-crypt» ядра Linux. Використовується алгоритм AES-128 в режимі CBC с ESSIV. Із паролю користувача створюється спеціальний ключ для шифрування майстер-ключа. Саме він має бути доступним під час загрузки

ОС, щоб активувати основні функції системи. Майстер-ключ використовується для всіх операцій блокування доступу до даних, тому після проходження користувачем автентифікації, назавжди залишається в пам'яті пристрою.

Файлове шифування — це так званий гранульований контроль. Модуль «fscrypt» ядра Linux використовує алгоритм AES-256 в режимі XTS для файлів CBC і для метаданих файлів. Маємо дві категорії зашифрованих файлів: Device Encrypted (DE) і Credential Encrypted (CE). Дані DE ввібрали в себе секрети пристрою та стають доступними і під час загрузки, і при розблокуванні, вони зарезервовані для системних додатків. Наприклад, для клавіатури, калькулятора, годинника, викликів). Дані CE шифруються ключем з паролю користувача і стають доступними тільки після розблокування. Використовуються всіма додатками за замовчуванням.

Android Verified Boot (AVB)

Тут застосований модуль верифікації «dm-verity», щоб перевірити цілісність початкової загрузки системи на основі криптографічного хеш-дерева. Перевірка завершиться помилкою, якщо під час загрузки модулів операційної системи одне із значень буде введено не правильно, а пристрій залишиться в нефункціональному стані. Підтримується і опція повернення до попередньої безпечної версії, яка зберігається в спеціальному розділі, стійкому до проникнення. Загружчик в позиції «розблоковано» дозволяє вимкнути AVB і ввести довільний код під час загрузки пристрою. Такий хід може знадобитися розробникам чи користувачам, якщо вони хочуть мати рут-доступ.

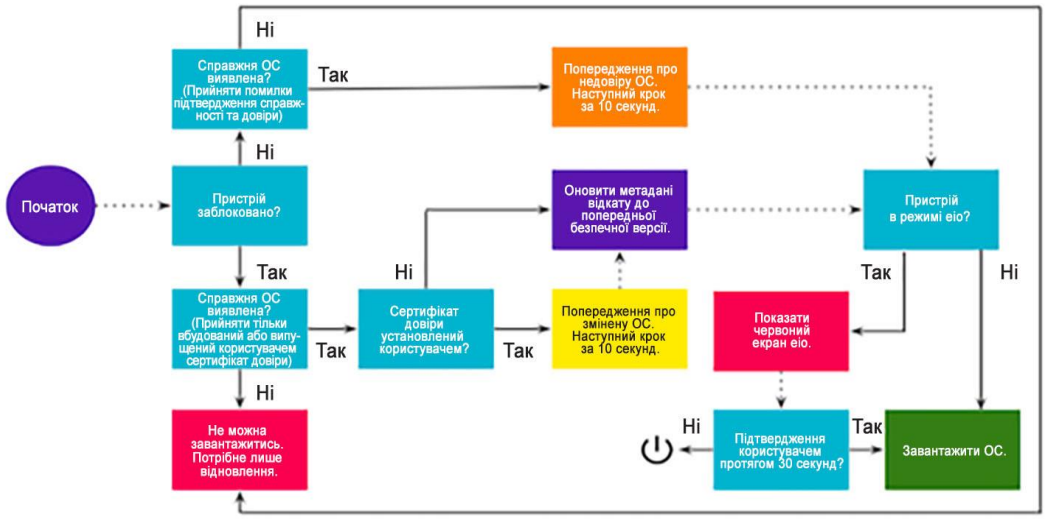

Рис. 5. Алгоритм роботи Android Verified Boot

Google Mobile Services (GMS)

Google користується Google Cloud для зберігання даних і протоколом TLS для зв'язку клієнтів з сервером. Дані із Google Cloud зашифровані з допомогою ключів, про які знає Google. Наскрізне шифрування підтримується тільки у програмі Google Meet. Розробники додатків співпрацюють з GMS через Play Services API. Є ще й API SafetyNet, така служба перевірки пристроїв Android на наявність рут-прав чи сумнівного ПО.

Підписування APK та перевірка коду

AOSP вимагає щоб розробник підписував створений ним додаток, який планується запустити. Підпис файлу APK відповідає відкритому ключ. Вони поширюються разом. До початку публікації додатку в Play Store відбувається автоматична і ручна перевірка коду співробітниками Google. В версії Android 11 уже задіяний модуль «fs-verity», який постійно перевіряє файли APK. При наявності сертифікату GMS телефон має опцію установки додатків з «невідомих джерел», відсутніх, наприклад, в Play Store або від інших розробників.

Резервне копіювання

З допомогою GMS, воно організоване 2 способами. Android Backup Service запроваджено в версії Android 2.2. Цей додаток створює копії пар ключів і значень відповідно до того, як їх розмістив користувач. Така опція не обов'язкова, вона закладається і налаштовується розробниками. В Android 6 вже працює механізм автоматичної синхронізації даних з Google Drive – Auto-Backup. Цей варіант обходиться без налаштування розробника, а користувач має змогу відмовитися від послуги. З 2018 року діє наскрізне шифрування для Android Backup Service. Ключ для шифрування бекапів створюється на пристрої, щоб дістатися до ключа, використовують класичну схему (PIN-код, пароль чи патерн). Дані додатків копіюються в сервіс Google Drive окремо, так як Android Auto-Backup. Вони шифруються паролем акаунту Google чи автентифікацією користувача на пристрої. У Google є доступ до цих ключів. Іноді виробники пропонують власні бекап сервіси. Android Debug Bridge та інші інструменти, що дозволяють робити резервне копіювання на комп'ютер через USB.

Техніку обходу захисту даних користувачів Android

Root-доступ і використання експлойтів**.** Отримання root-доступу в Android подібне до *Jailbreak* на iOS — це дозволяє модифікувати ОС відповідно до потреб користувача, оминати обмеження виробників чи запускати довільний код. Бувають пристрої, які користуються root-доступом для розблокування, тоді необхідно розблокувати загружчик. Всі користувацькі дані при цьому видаляються, бо система перестає довіряти загрузці і не може гарантувати безпеку. Але є такі експлойти, що дозволяють обходити ці обмеження. Згідно зі статистикою SafetyNet від 2017 року, 5,6 відсотків Android пройшли процедуру рутування. Починаючи з 2013 р. Samsung запровадив запрограмований електронний запобіжник Knox-bit. Він спрацьовує під час спроби рутування, виходить з ладу і може бути замінений тільки апаратно. Google SafetyNet також відстежує пристрої, де було застосовано root-доступ. Отже, техніка блокується виробниками. Різними способами отримується root-доступ за допомогою експлойтів без видалення даних користувача (атака на ядро ОС чи код OEM або особливості SoC). Android базується на ядрі Linux, проблемні місця у них спільні і дозволяють використати доступ суперкористувача. На вразливості Linux Dirty CoW базується експлойт під Android, а прогалина PingPongRoot, в ядрі Android, може використовуватися в Linux. Продовжують виявлятися ще і нові невідомі досі недоліки (0-day), що можуть забезпечити root-доступ. Атака на конкретний пристрій вузько спрямована. Враховуються особливості прошивки компонентів, обладнання, якщо проводять маніпуляції, то можлива загрузка інтерфейсу Qualcomm EDL, а не Android. Таким чином шахрай записує потрібний код в розділ Android і отримує права суперкористувача без втрати даних Рутування можливе і за допомогою інструменту SP-Flash-Tool без розблокування загружчика, якщо на пристрої присутні компоненти MediaTek. Список досить великий.

Як можна було б підвищити захист Android

Шифрування даних користувача під час блокування екрану

Ключ шифрування видаляється з пам'яті після блокування екрану. Особисті дані будуть недоступні до того часу, поки користувач не ввімкне пристрій. Така функціональна особливість Android підвищить рівень безпеки на порядок але розробникам доведеться адаптувати роботу додатків в фоновому режимі із заблокованим екраном. Це також обмежить ефективність вилучення даних із рутованих через експлойти приладів.

Запровадження наскрізного шифрування для продуктів Google

Підвищить безпеку даних і довіру користувачів наскрізне шифрування в службах GMS. Хоча треба пам'ятати, що така надійна конфіденційність, може перешкоджати правоохоронцям отримувати цінну інформацію для боротьби з небезпечними злочинцями.

Безпечний доступ в роботі з прошивкою

Більшість експлойтів здатні обійти захисні механізми Android, користуючись недоліками в прошивці. Існує два шляхи вирішення питання: апаратний спосіб, коли розповсюджуються

програматори прошивки з компонентами, захищеними від проникнення і вони дозволять відслідковувати витік. Другий варіант — відкликати сертифікат скомпрометованого програматора.

Підсилення компонентів апаратної частини

Функціональність TEE TrustZone підсилюється апаратно, коли використовується співпроцесор безпеки. Виробники не завжди йдуть на такий крок, бо додатковий співпроцесор не дозволить зменшити вартість пристрою.

Підвищення частоти оновлення Android

Сьогодні пристрої мають дуже надійний захист даних користувача. Але остання версія ОС встановлена далеко не на всіх пристроях, тому ризик проникнення залишається досить високим. Особливо це стосується застарілих варіантів, коли просто неможливо провести оновлення операційної системи. Знизити відсоток ризику допоможе запровадження навчання користувачів і своєчасне оновлення ОС.

Переваги відкритого коду і екосистеми

Швидке та ефективне захоплення ринку та популярність Android у світі допоможуть підвищити захист. Дякуючи відкритому коду, розробники вводять покращені оновлення незалежно від Google. А Google може формувати стандарти безпеки та просувати різні протоколи. Цілісне ядро Linux і Android взаємовигідне для обох платформ. Оскільки, вони допомагають один одному вчасно виявляти і ліквідовувати проблеми, поступово підвищувати безпеку.

Особливості безпеки Apple iOS:

- Автентифікація користувача (цифровий PIN-код, пароль, біометрія) + SEP;
- підписування коду (Boot ROM і підписування додатків);
- пісочниця і аналіз коду додатків;
- шифрування (AES, ECDH over Curve25519, UID, Data Protection classes);
- Keychain (API для безпечного зберігання ключів);
- Резервне копіювання (Keybag і iCloud Backup Keybag);
- хмаровий сервіс iCloud і CloudKit (API для iCloud);
- iCloud Keychain (синхронізація і відновлення Keychain);
- апаратна безпека (SEP);
- контроль і обмеження фронту атаки (Wi-Fi, Bluetooth, пуш-повідомлення, USB/Lightning).

Особливості безпеки Google Android:

- Автентифікація користувача (цифровий PIN-код, пароль, патерн, біометрія) + Gatekeeper і Smart Lock;
- пісочниця додатків (DAC + SELinux);
- шифрування всього диска и файлів (AES-128/256, Credential Encrypted і Device Encrypted), а також SD-карти;
- Trusted Execution Environment (TEE) і додатковий апаратний модуль безпеки (secure element);
- Android Verified Boot;
- Резервне копіювання (Auto-Backup і Backup Service);
- Google Mobile Services;
- Підписування додатків і аналіз коду;
- Наскрізне шифрування в Google Meet.

Висновки

Зловмисник може проникнути систему і отримати конфіденційні дані, не залежно від використовуваної платформи, якщо докладе немалих зусиль. В кожній ОС є свої особливості безпеки.

Хоч Apple краще організує шифрування файлів на пристроях і в хмарі, ніж Google, виникають ситуації, коли шахраєві вдається отримати дані користувача. Зберігання токенів автентифікації для iCloud в режимі AFU треба переробити. Наскрізне шифрування всіх даних в хмарі, усуне

протиріччя в політиці конфіденційності Apple. Наскрізне шифрування рекомендують і для Android GMS. Серйозним недоліком Android є постійне зберігання ключів шифрування в пам'яті пристрою після першого розблокування. Аналог iOS Complete Protection збільшить захист даних користувача від шахрайських атак. Користувачам мобільних пристроїв рекомендують створювати дуже складні паролі разом з біометричною автентифікацією скрізь, де тільки можливо. Якщо є необхідність, то запроваджувати 2FA (двофакторну автентифікацію). Краще не закладати в хмару нічого зайвого і для цього треба гарно вивчити схему синхронізації даних з хмаровими сервісами. Рутирування/джейлбрейк приладів — це ризик запуску довільного коду (навіть в прошивці) і загроза виходу з ладу ОС пристрою. Сучасна безпека даних користувача на обох платформах організована досить якісно але і для Google Android, для Apple iOS є ще куди рости в цьому плані.

References

1. Wenbo Mao, “Modern Cryptography”, First Edition, Pearson Education, 2008 ISBN-. 13: 978-0132887410
2. Quisquater, J., Math Rizk, Side Channel Attack - State of the art, (2002).
3. Cetin Kaya Coc, Cryptographic Engineering/ Cetin Kaya Coc// Springer Science+Business Media, LLC 2009 – 171 p
4. Apple Platform Security. May 2022 available at: https://support.apple.com/guide/security/welcome/web (accessed 21 October 2022).
5. Secure Android devices. available at: https://source.android.com/docs/security (accessed 12 November 2022).