УДК 004.056

ІНФОРМАЦІЙНА ЗАХИЩЕНІСТЬ МОБІЛЬНИХ ЗАСТОСУНКІВ

Р. Ю. Клоченок[1], С. О. Носок[1]

[1] *Національний технічний університет України «Київський політехнічний інститут»*

Анотація

Необхідним етапом розробки мобільних застосунків як програмних компонентів інформаційних систем, що взаємодіють з критичними ресурсами, є дослідження розроблених програмних рішень в контексті інформаційної безпеки. У статті представлений підхід до дослідження інформаційної захищеності мобільних застосунків і систематизація типових вразливостей застосунків, вироблена в процесі аналізу набору мобільних застосунків, до яких пред'являються підвищені вимоги безпеки.

Ключові слова: безпека програм, мобільні застосунки, декомпіляція, зворотна розробка.

Вступ

Невід'ємною частиною процесу розробки інформаційних систем, що взаємодіють з критичними ресурсами, є етап аналізу інформаційної захищеності таких систем. Безліч компонентів інформаційних систем, що взаємодіють з критичними ресурсами, включає програмні рішення для мобільних платформ. Актуальність таких рішень визначається значним розвитком мобільних екосистем і, в сукупності з попитом аудиту інформаційної безпеки як етапу розробки захищених систем обробки даних, обумовлює нагальність дослідження мобільних додатків в контексті інформаційної безпеки. Предметом дослідження даної роботи являється аналіз інформаційної захищеності мобільних застосунків, до яких застосовуються підвищені вимоги інформаційної безпеки.

1. Систематизація типових вразливостей мобільних додатків

У даному розділі систематизовано інформацію про мобільні застосунки на предмет відповідності вимогам інформаційної безпеки. В розділі розглядаються типові уразливості мобільних застосунків та їх властивості.

1.1. Використання незахищених протоколів передачі інформації

У контексті сервісів, до яких пред'являються підвищені вимоги безпеки, необхідно використовувати захищені протоколи передачі даних для здійснення міжмережевих запитів. В іншому випадку, зловмисник може здійснити атаку на канал зв'язку типу «людина посередині», що може призвести до втрати конфіденційність даних, які передаються.[1]

1.2. Небезпечна конфігурація захищеного з'єднання

У разі, якщо настройка пакета реалізації захищеного протоколу проведена некоректно, зловмисник може здійснити атаку на захищене з'єднання, що може привести до повної втрати конфіденційності даних, які передаються. До потенційних вразливостей захищених з'єднань слід віднести:

- Використання застарілих протоколів. Зокрема, протокол SSLv2 (Secure Sockets Layer) є застарілим і має ряд відомих вразливостей
- Використання вразливих протоколів обміну криптографічними ключами, слабких алгоритмів перевірки цілісності повідомлень, уразливих алгоритмів шифрування або алгоритмів шифрування з малою ефективною стійкістю ключа шифрування.
- Використання небезпечного протоколу повторного рукостискання.
- Використання алгоритмів стиснення
- Відомі вразливості пакетів реалізації протоколу SSL

1.3. Використання небезпечних криптографічних методів

У разі, якщо мобільним застосунком застосовуються криптографічні методи, такі як шифрування або криптографічні функції згортки, мобільним застосунком можуть застосовуватись методи, що володіють відомими вразливостями. До таких методів слід віднести:

- Слабкі і застарілі алгоритми симетричного шифрування, такі як DES (Data encryption standard), Triple DES з двома ключами шифрування
- Алгоритми шифрування з малою ефективною стійкістю ключа шифрування.

- Алгоритми формування ключа шифрування на підставі парольної фрази з малою кількістю ітерацій.
- Слабкі і вразливі режими алгоритмів шифрування, наприклад ECB (Electronic code book) або OFB (Output feed back)
- Слабкі і застарілі криптографічні функції згортки, такі як MD4 (Message Digest 4) і SHA-1 (Secure Hash Algorithm 1)

У разі, якщо мобільний застосунок використовує вразливі криптографічні методи або при компрометації ключів шифрування, зловмисник може відновити вихідні дані – це може призвести до компрометації конфіденційних даних користувача. Однією з поширених помилок реалізації механізмів шифрування є використання режимів шифрування за умовчанням.[2]

1.4. Небезпечне зберігання конфіденційних даних

Мобільні операційні системи надають системні інтерфейси для взаємодії мобільних додатків з файловою системою. У разі якщо політики безпеки мобільної операційної системи не були змінені, мобільні застосунки мають дозвіл на читання і запис даних в домашній директорії застосунку і, в окремих випадках, до зовнішнього пристрою допоміжної пам'яті. Проте, в тому випадку якщо до даних застосунку не застосовуються алгоритми шифрування, доступ до даних мобільних додатків може бути отриманий зловмисником при наявності у нього доступу до файлової системи пристрою. Наприклад, доступ до файлової системи може бути отриманий шкідливими програмами на пристроях зі зміненими політиками безпеки операційної системи, в результаті атаки типу «обхід файлового шляху», а також при наявності у зловмисника фізичного доступу до пристрою.

1.5. Ненавмисна компрометація даних

Компоненти мобільного додатка можуть здійснювати дії, які можуть привести до компрометації конфіденційної інформації. До таких дій слід віднести:

- Зберігання бази даних підсистеми кешування запитів і відповідей веб-сервера
- Зберігання бази даних підсистеми кешування, що вводиться користувачем;
- Зберігання конфіденційної інформації в системному буфері обміну даними.

1.6. Небезпечна обробка вхідних даних

Мобільні додатки, які не перевіряють або некоректно обробляють вхідні дані можуть бути схильні до непередбаченої зміни поведінки застосунку, наприклад до компроментації конфіденційних даних або зміну потоку управління програми, а також до атак типу «відмова в обслуговуванні». До вхідних даними, що використовуються мобільними застосунками, слід віднести дані, отримані в процесі міжмережевої взаємодії, в процесі взаємодії між процесами та інше. Некоректна обробка вхідних даних програмою може привести до неепердбачуваних змін поведінки програми. В залежності від контексту, в якому використовуються такі вхідні дані, техніки, які використовуються для атаки на застосунок, і пов'язані з такою атакою ризики можуть значно відрізнятися. До типових класів атак слід віднести такі атаки як «впровадження SQL запитів»,«обхід файлового шляху», помилки переповнення буфера.[3]

1.7. Недостатній захист пакету застосунку і його компонентів

До потенційних вразливостей мобільного застосунку, пов'язаних з недостатнім захистом його пакету і його компонентів, слід віднести:

- Недостатнє покриття програмного коду маскуючими перетвореннями;
- Відсутність перевірки цілісності пакету застосунку і його компонентів;
- Відсутність механізму виявлення налагоджувача програмного коду;
- Наявність налагоджувальної і символьної інформації про програмний код в пакеті застосунку

2. Методика аналізу програм на предмет відповідності вимогам інформаційної безпеки

Процес аналізу мобільного застосунку повинен включати в себе аналіз програмного коду цього застосунку. Програмний код мобільної програми повинен бути досліджений як із застосуванням інструментальних засобів статичного аналізу, так і вручну. При цьому методика аналізу програмного коду істотно залежить від доступності вихідного коду мобільного застосунку або його компонентів. Далі запропоновано послідовність дій при дослідженні мобільного застосунку з доступним і недоступним вихідним кодом програми.

2.1. Аналіз програмного коду з доступним вихідним кодом

Аналіз програмного коду з доступним вихідним кодом мобільного застосунку може бути здійснений з використанням інструментальних засобів статичного аналізу. Дослідження програмного коду таких мобільних програм повинно здійснюватись наступним чином:

- Проведення аналізу вихідного коду на наявність стороннього програмного коду з відкритих джерел. При виявленні стороннього програмного коду, такий код повинен бути досліджений на наявність описаних вище класів вразливостей.
- Проведення аналізу вихідного коду мобільного застосунку інструментальними засобами статичного аналізу.
- Проведення аналізу вихідного коду програми вручну

2.2. Аналіз програмного коду з частково або повністю недоступним вихідним кодом

Аналіз програмного коду з частково або повністю недоступним вихідним кодом мобільних застосунків повинен бути здійснений з використанням методів зворотньої розробки програмного коду. Дослідження програмного коду таких мобільних програм повинно здійснюватись наступним чином:

- Проведення аналізу програмного коду на наявність стороннього програмного коду з відкритих джерел. У контексті дослідження програмних компонентів, вихідний код яких недоступний, наявність стороннього программного коду повинна визначатись на підставі символьної інформації у двійковому образі таких компонентів. При наявності стороннього програмного коду, такий код має бути перевірений на наявність зазначених вище вразливостей.
- Застосування методів зворотньої розробки до програмного коду мобільного застосунку. У процесі дослідження програмного коду слід застосовувати як інструментальні засоби дизасемблерування програмного коду, так і засоби відновлення вихідного коду з використанням методів декомпіляції.
- У разі, якщо були застосовані методи відновлення вихідного коду, повинен відбуватися аналіз такого коду інструментальними засобами статичного аналізу.
- Проведення аналізу дизасемблейованого коду вручну

Також слід звернути увагу та провести аналіз на коректність роботи:

- потоків управління і даних мобільного застосунку;
- міжмережевих взаємодій,які здійснюються мобільним застосунком;
- викликів системних функцій, зокрема процедур читання і запису на пристрої допоміжної пам'яті і записів в системний журнал подій;
- використання сервісів, що надаються операційною системою;
- взаємодії між процесами.

Висновки

У роботі запропонована методика аналізу застосунків для мобільних платформ на предмет відповідності вимогам інформаційної безпеки. Також було проведено опис типових класів уразливостей мобільних застосунків, а також потенційні ризики, пов'язані з такими вразливостями.

Перелік використаних джерел

1. OWASP Mobile Security Project. — URL: `https://www.owasp.org/index.php/OWASP_Mobile_Security_Project`.
2. Turner S. Chen L. Updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithms. — 2011.
3. The Web Application Security Consortium / Thread Classifi cation. — URL: `http://projects.webappsec.org/w/page/13246978/Threat_20Classification`.