정 보 보 호 학 회 지
제 32 권 제6호, 2022. 12

암호화된 스마트폰 앱 데이터에 대한 암호화 키 생성 및 암호 알고리즘 분류

이 신 영*, 김 한 결*, 박 명 서**

요 약

스마트폰 어플리케이션 데이터는 사용자와 밀접한 데이터를 포함하고 있기 때문에 디지털 포렌식 수사에서 매우 중요하게 수집되어야 할 대상이다. 하지만 어떤 어플리케이션은 암호화 기법을 활용하여 사용자 데이터를 포함한 어플리케이션 데이터를 보호한다. 데이터 암호화는 데이터 보호에 효율적이지만, 디지털 포렌식에서는 안티 포렌식으로 작용하여 어플리케이션 데이터 분석을 어렵게 한다. 따라서, 효율적인 디지털 포렌식 수사를 위해서는 암호화된 스마트폰 어플리케이션 데이터에 복호화에 대한 선제 연구가 필요하다. 본 논문에서는 최근 발표된 스마트폰 어플리케이션의 암호화된 데이터에 대한 분석 결과를 수집하여, 암호화 키 생성 및 데이터 암호화 방법에 대해 분류하였다. 이를 통해 스마트폰 어플리케이션 데이터의 암호화 방식을 습득하여 추후 또다른 어플리케이션 효율적인 분석에 활용할 수 있을 것으로 기대한다.

Ⅰ. 서 론

스마트폰의 어플리케이션은 다양한 서비스를 통해 사용자에게 편의성을 제공한다. 어플리케이션은 서비스 제공을 위해 사용자와 관련된 다수의 데이터를 어플리케이션이 관리하는 영역에 저장하여 관리한다. 하지만, 이러한 데이터들은 사용자와 밀접한 상당히 민감한 부분을 포함하고 있기 때문에 외부 노출 위험에서 보호될 필요가 있다. 어플리케이션은 민감한 데이터의 보호 수단 중 하나로 데이터 암호화를 사용하며, 이때 데이터 암호화 방식은 각 어플리케이션마다 고유한 방식을 사용한다. 데이터의 암호화는 정보보호 관점에서는 사용자의 데이터 보호이지만, 디지털 포렌식 관점에서는 안티 포렌식으로 작용한다. 따라서 이를 해결하기 위해 안티안티 포렌식 기술인 암호화된 데이터 복구 기술이 필수적이다. 어플리케이션에 대한 암호화 방식에 대해 최신 동향 파악은 효율적인 암호화된 데이터 분석에 도움을 줄 수 있다.

본 논문에서는 2018년부터 2022년 사이에 분석된 총 12개의 어플리케이션에서 사용 중인 암호화 방식에 동향에 대해 조사하였다. 암호화 방식은 암호화키 생성과 암호 알고리즘으로 나누었으며, 각 어플리케이션은 상이한 방법을 사용하고 있다.

Ⅱ. 관련 연구

스마트폰 어플리케이션 암호화 방식에 대한 분석 연구는 최근에도 활발히 진행되고 있다. 최용철 등 3명은 Valut 어플리케이션에서의 데이터 암호화 알고리즘 및 은닉 알고리즘을 분석하였다[1]. 박준성 등 2명은 크롬 브라우저에서 구글 패스워드 매니저 데이터 획득에 대한 연구를 진행하였다[2]. 박진성 등 4명은 포렌식 분석을 위한 LockMyPix의 미디어 파일 복호화 방안에 대해 연구하였다[3]. 김기윤 등 4명은 보안 메신저 SureSpot 애플리케이션에 대한 포렌식 분석을 진행하였다[4]. 강수진 등 4명은 macOS 환경에서의 Huawei 및 Apple 스마트폰 암호화 백업 데이터 복호화 및 아티팩트를 분석하였다[5]. 박귀은 등 3명은 안드로이드 환경에서의 지도 애플리케이션 아티팩트 분석 및 복호화 방안에 대한 연구를 진행하였다[6]. 김기윤 등 4명은 인스턴트 메신저 말랑말랑 톡카페 애플리케이션 데이터베이스 복호화 방안 및 분석을 진행하였다[7]. 신수민은 디지털 포렌식 관점에서 노트 및 다이어리 애플리케이션의 보안 프로세스 분석을 진행하였

* 강남대학교 소프트웨어응용학부 (학부생, sin99010@kangnam.ac.kr, 학부생, biequal@kangnam.ac.kr)
** 강남대학교 ICT융합공학부 (조교수, pms91@kangnam.ac.kr)

다[8]. 김기윤 등 3명은 LG 갤러리 애플리케이션 잠금 파일 복호화 연구를 진행하였다[9].

Ⅲ. 암호화 키 생성 방법

본 장에서는 데이터 암호화 시 사용되는 암호화 키 생성 방법에 대해 설명한다. 특정 어플리케이션에서 여러 개의 암호화 키 생성 방법을 사용하는 경우도 존재했으며, 어플리케이션 별 공통된 암호화 키 생성 방법을 사용하는 경우도 존재했다.

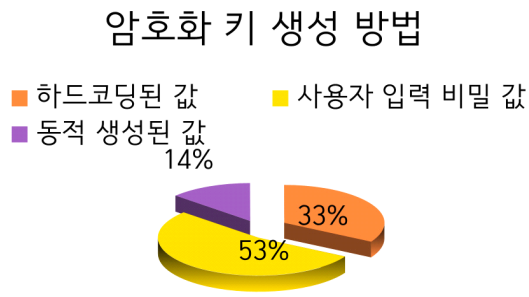
암호화 키 생성 방법은 총 4가지 유형으로 구분될 수 있으며, 그 유형은 다음과 같이 정리할 수 있다.

첫 번째는 하드 코딩된 값을 이용하는 것이다. 단순히 소스코드 상에서 존재하는 문자열 또는 바이트 값을 그대로 사용하거나 여러 위치에 존재하는 값을 조합하여 사용한다. 보통, 하드 코딩된 값은 XOR 또는 base64를 통해 인코딩한 결과를 해시하여 암호화 키로 사용한다.

두 번째는 동적으로 값을 생성하는 것이다. 어플리케이션 초기 실행 시 임의의 값을 생성하여 암호화 키로 사용하거나 전용 라이브러리 함수를 사용하여 생성한다.

세 번째는 사용자 입력 비밀 값을 이용하는 방법이다. 어플리케이션은 별도로 사용자 입력 비밀 값인 PIN 또는 패스워드 입력을 받아 이 값을 기반으로 암호화 키를 생성하게 된다. 이때 암호화 키를 생성하는 방법으로는 해시 함수 또는 PBKDF2를 사용할 수 있다. 해시 함수를 이용하는 방법은 MD5, SHA1, SHA256, SHA512에 대한 입력으로 사용자 입력 비밀 값을 받고, 그에 대한 출력인 해시 값을 암호화 키로 사용한다. Password-Based Key Derivation Function 2 (PBKDF2)를 이용하는 방법은 사용자 입력 비밀 값, salt 및 반복횟수를 입력으로 하여 출력된 값을 암호화 키로 사용한다. 이때, PBKDF2의 PRF(PseudoRandom Function)은 HMAC-MD 5, HMAC-SHA1, HMAC-SHA256 등을 사용할 수 있다.

[그림 1]은 12종의 어플리케이션에서 사용되는 암호화 키 생성 방법의 비율을 나타낸 것이다. 사용자 입력 비밀 값을 기반으로 암호화 키를 생성하는 어플리케이션이 53% 비율로 가장 많았고, 동적 생성된 값을 이용하여 암호화 키를 생성하는 어플리케이션이 14% 비율로 가장 적었다.



〔그림 1〕 암호화 키 생성 방법에 대한 분류

3.1. 하드 코딩된 값을 이용한 암호화 키 생성

하드코딩된 값을 이용하여 암호화 키를 생성하는 어플리케이션은 다음과 같다.

Calculator는 파일 및 동적 라이브러리에 분산되어 저장된 16-byte의 문자열을 조합하여 암호화 키를 생성한다.

계산기는 “123456789” 문자열 8-bytes를 DES의 암호화 키로 사용한다. LockMyPix는 소스 코드상 존재하는 크기가 16인 byte 배열을 암호화 키로 사용한다. Hisuite는 백업 데이터 중 미디어 파일 및 해당 데이터가 저장된 데이터베이스를 암호화 시 16-byte의 고정 값을 암호화 키로 사용한다. Private Notepad는 apk 내에 저장된 두 개의 하드 코딩된 값을 base64로 디코딩한 후 XOR하여 암호화 키로 사용한다. iOS용 Diaro는 SQLCipher4의 Passphrase를 어플리케이션 내부 파일의 특정 항목에 저장되어 있는 고정된 값을 사용한다.

3.2. 동적 생성된 값을 이용한 암호화 키 생성

동적 생성된 값을 통해 암호화 키를 생성하는 어플리케이션은 다음과 같다. Chrome 브라우저는 처음 임의로 생성한 32-bytes 값을 암호화 키로 사용한다. LG 갤러리 어플리케이션은 liblgdrm 라이브러리와 구글 계정으로부터 암호화 키를 생성한다.

3.3. 사용자 입력 비밀 값을 이용한 암호화 키 생성

사용자 입력 비밀 값을 이용한 암호화 키 생성 방법은 해시 함수와 PBKDF2를 활용한 방법으로 분류된다.

해시 함수를 이용하여 암호화 키를 생성하는 어플리케이션은 다음과 같다.

LockMyPix는 PIN 또는 패스워드를 SHA-1을 이용하여 해시한 20-byte 해시 값의 상위 16-byte를 추출하여 암호화 키로 사용한다. SureSpot은 메시지 암호화 용도의 암호화 키 생성에 SHA256 해시 함수를 사용한다. Apple사의 암호화된 백업 데이터는 사용자 입력 패스워드를 기반으로 키를 생성한다. iOS 10 버전은 SHA256을 통해 암호화 키를 생성한다. 네이버 지도는 SQLCipher4의 default 파라미터를 사용하며, 해당 파라미터에서 Passphrase 생성은 SHA512를 통해 이루어진다. 안드로이드 용 Diaro는 android id를 MD5로 해싱한 후, hex string으로 변환하여 해당 값을 암호화 키로 사용한다. PBKDF2를 통한 키를 생성하는 어플리케이션과 PBKDF2를 사용할 시 사용된 PRF는 다음과 같다. Hisuite는 백업 데이터 중 미디어 파일 및 해당 데이터가 저장된 데이터베이스를 제외한 백업 파일을 암호화할 때 암호화 키를 사용자 입력 패스워드를 기반으로 PBKDF2을 통해 생성되며 이때 사용되는 PRF는 HMAC-SHA256 이다. SureSpot은 키 교환 프로토콜로 NIST 표준 타원곡선중 하나인 Secp521r1을 사용하며, 이를 위해 사용되는 공개키 쌍 데이터는 암호화되어 특정 파일에 저장된다. 암호화 키는 사용자 입력 비밀값에 대한 PBKDF2를 통해 생성되며, 사용되는 PRF는 HMAC-SHA256이다. Apple사의 암호화된 백업 데이터는 사용자 입력 패스워드를 기반으로 키를 생성한다. iOS 3, 4~9, 10.1 버전에서는 PBKDF2를 통해 생성하며 사용되는 PRF는 HMAC-SHA1이다. iOS 10.2 이상도 마찬가지로 PBKDF2를 사용하며 이때 PRF는 HMAC-SHA256과 HMAC-SHA1을 사용하여 생성한다. 말랑말랑 톡카페 어플리케이션은 암호화키를 PBKDF2를 이용하며 PRF는 HMAC-SHA1을 통해 생성한다.

Ⅳ.암호화 알고리즘

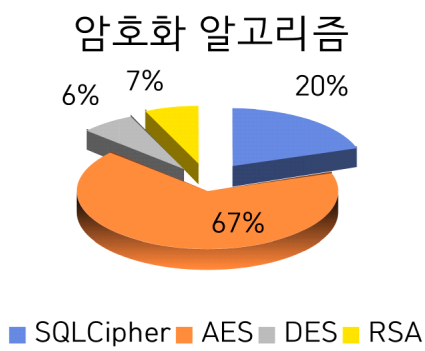
본 장에서는 데이터의 암호화에 사용된 암호화 알고리즘에 대해 설명한다. 총 12개의 어플리케이션들의 다양한 방식으로 데이터들의 암호화를 진행한다. 하나의 어플리케이션에서 여러 개의 암호화 알고리즘을 사용하는 경우도 있고 같은 방식의 암호화 알고리즘을 사용하는 경우도 있었다. 사용된 암호화 알고리즘은 크게 3가지 유형으로 나눌 수 있으며 다음과 같다.

첫 번째는 SQLCipher를 사용하는 것이다. SQL Cipher 사용 시 Pagesize, Iteration, HMAC Algorithm 매개변수 값이 필요하며 각 어플리케이션 마다 다른 파라미터값을 사용한다.

두 번째는 블록암호를 사용하는 것이다. 블록암호는 AES 또는 DES를 이용하며, 함께 사용된 운영모드는 ECB, CBC, CTR, GCM이다.

세 번째는 공개키 방식을 사용하는 것이다. 해당 방식은 RSA-ECB-PKCS1Padding을 암호화 알고리즘으로 사용한다.

[그림 2]는 데이터 암호에 사용된 암호화 알고리즘에 대한 비율을 나타낸 것이다. AES를 기반으로 데이터를 암호화하는 어플리케이션이 67% 비율로 가장 많았고, DES를 사용하는 비율이 6%로 가장 낮았다. 이는 실제 데이터를 암호화하기 위해 취약하다고 알려진 DES를 사용하지 않고 대부분 AES를 사용하여 암호화를 수행한다는 사실을 확인할 수 있다.



〔그림 2〕 데이터 암호에 사용된 암호화 알고리즘 분류

4.1. SQLCipher

SQLCipher는 오픈소스 형태로 제공되는SQLite 데이터베이스의 암호화 모듈이다. 즉, SQLCipher는 SQLite 데이터베이스 파일 자체를 암호화하는 방식이다. SQLCipher는 SQLite 데이터베이스를 AES256-CBC 방식을 사용하여 암호화를 하며 페이지 단위로 암호화한다. 따라서 페이지 크기를 설정하여야 하며 이 때문에 Pagesize 매개변수 설정이 필요하다. 암호화에 사용되는 키는 외부에서 입력된 Passphrase와 공개된 매개변수(salt, 반복횟수)를 사용한 PBKDF2 알고리즘으로 생성한다. 이때 PBKDF2의 필요한 반복횟수와 PRF값이 필요하다. 반복횟수는

Iteration, PRF값은 HMAC Algorithm 매개변수를 통해 설정한다. SQLCipher 방식을 사용하는 어플리케이션은 4종이며 4종의 어플리케이션의 매개변수 값은 다음과 같다.

Calculator의 Pagesize는 1024bytes, Iteration은 64000, HMAC Algorithm은 HMAC-SHA1이다. 네이버 지도의 Pagesize는 4096bytes, Iteration은 256000, HMAC Algorithm은 HMAC-SHA512이다. 말랑말랑 톡카페의 Pagesize는 1024bytes, Iteration은 64000, HMAC Algorithm은 HMAC-SHA1이다. Diaro의 Pagesize는 4096bytes, Iteration은 256000, HMAC Algorithm은 HMAC-SHA512이다.

〔표 1〕 암호화 키 생성 방법과 암호화 알고리즘에 대한 요약

어플리케이션	운영체제	버전 정보	암호화 키 생성 방법	데이터 암호 알고리즘
Calculator - Photo Vault & Video Vault hide Photos	Android	10.0.7	Hard-coded-String	AES128-CBC-PKCS#7Padding
계산기 - 사진 보관함(사진 숨김)	Android	2.6.1	Hard-coded-String	DES-ECB-PKCS#5Padding
Chrome	Android, iOS	78.0.3904	Dynamic Generation	AES-256-GCM
LockMyPix	Android	4.2.6-b	Hard-coded-String, SHA-1	AES128-CBC-PKCS#7Padding, AES128-CTR
SureSpot	Android, iOS	81, 21	PBKDF2-HMAC-SHA256, SHA256	AES-256-GCM
Hisuit	iOS	11.0.0.530	Hard-coded-String, PBKDF2-HMAC-SHA256	AES-128-CTR, AES-256-CTR
iTunes	iOS	12.9.5.5	PBKDF2-HMAC-SHA1, PBKDF2-HMAC-SHA256. SHA256	AES-256-CBC
네이버 지도	Android	5.15.2.7	SHA512	SQLCipher4 RSA-ECB-PKCS1Padding
말랑말랑 톡카페	Android	1.10.6	PBKDF2-HMAC-SHA1	SQLCipher, AES-256-CBC
Diaro	Android, iOS	3.91.2, 1.9.5	MD5, Hard-coded-String	AES256-CBC-PKCS#7Padding, SQLCipher4
Private Notepad	Android	6.4.0	Hard-coded-String	AES128-CBC-PKCS#7Padding
LG 갤러리	Android	10.2.5.301	Dynamic Generation	AES128-CBC-PKCS#7Padding

4.2. 블록 암호

블록 암호를 이용해 데이터가 암호화되는 경우 AES 또는 DES를 기반으로 한 ECB, CBC, CTR, GCM 운영모드가 사용되었다. 블록 암호 AES를 사용한 어플리케이션은 10종이다. 각 어플리케이션에서 사용된 AES의 종류와 운영모드 및 패딩 방법은 다음과 같다. Calculator는 AES128-CBC-PKCS#7Padding 방식을 사용하여 숨긴 사진, 동영상 파일을 암호화 하였다. Chrome 브라우저는 AES256-GCM 방식을 사용하여 패스워드를 암호화하였다. LockMyPix는 AES128-CBC-PKCS#7Padding 방식을 사용하여 파일명과 확장자를 암호화하며 AES128-CTR 방식을 사용하여 PIN 및 패스워드를 암호화하였다. SureSpot는 AES256-GCM 방식을 사용하여 공개키 쌍과 메시지 데이터를 암호화하였다. Hisuite는 미디어 파일 및 해당 데이터가 저장된 데이터베이스와 그 외 백업된 파일을 암호화한다. AES128-CTR 방식을 사용하여 미디어 파일 및 해당 데이터가 저장된 데이터베이스를 암호화하였고 AES256-CTR 방식을 사용하여 그 외 백업된 파일을 암호화하였다. iTunes는 AES256-CBC-PKCS#7Padding 방식을 사용하여 데이터베이스를 암호화하였다. 말랑말랑 톡카페는 AES256-CBCPKCS#7Padding 방식을 사용하여 데이터베이스를 암호화하였다. Diaro의 안드로이드 버전에서는 AES256-CBC-PKCS#7Padding 방식을 사용하여 데이터베이스를 암호화하였다. Private Notepad는 AES128-CBC-PKCS#7Padding 방식을 사용하여 데이터베이스를 암호화하였다. LG갤러리는 AES128-CBC-PKCS#7Padding 방식을 사용하여 미디어 파일을 암호화하였다. 블록 암호 DES를 사용한 어플리케이션은 계산기 1종이며 DES-ECB-PKCS#5Padding 방식을 사용한다.

4.3. 공개키 암호

공개키 방식을 사용한 어플리케이션은 네이버 지도 1종이며, RSA-ECB-PKCS#1Padding을 사용한다.

Ⅴ. 결 론

본 논문에서는 2018년부터 2022년 사이의 연구된 12개의 어플리케이션들의 암호화 키 생성 방법과 암호화 알고리즘을 분석 및 분류하였다. 암호화 키 생성 방법과 암호화 알고리즘은 부록의 [표 1]로 정리한다. 암호화 키 생성 방법은 하드 코딩된 값의 사용한 암호화 키 생성, 동적 생성된 값을 이용한 암호화 키 생성, 사용자 입력 비밀을 이용한 암호화 키 생성이 사용되었다. 암호화 알고리즘은 SQLCipher, AES, DES, RSA 방식이 사용되었다.

스마트폰 기술의 발전으로 어플리케이션에 저장되는 정보들이 점점 늘어나고 있다. 이러한 정보들을 보호하기 위한 대안이 필요하며 암호화를 통해 해결할 수 있다. 많은 어플리케이션들 각자의 방법으로 암호화 키를 생성하고 다른 암호화 방식을 사용하여 정보들을 보호한다. 본 논문을 통해 최근 사용된 어플리케이션들의 암호화 키 생성 방법과 암호화 방식들의 동향을 확인함으로써 현재 어플리케이션의 보안수준을 측정이 가능하고, 향후 다른 어플리케이션 효율적인 분석을 위해 활용될 수 있을 것으로 기대한다.

참 고 문 헌

[1] 최용철, 김기윤, 김종성, “Vault 앱의 데이터 암호화 알고리즘 및 은닉 알고리즘 분석”, Journal of Digital Forensics, 15(4), pp. 27-38, 2021.

[2] 박준성, 이상진, “크롬 브라우저(Chrome)에서 구글 패스워드 매니저 데이터 획득에 대한 연구”, Journal of Digital Forensics, 15(2), pp. 12-23, 2021.

[3] 박진성, 서승희, 김역, 이창훈, “포렌식 분석을 위한 LockMyPix의 미디어 파일 복호화 방안 연구”, Journal of Digital Forensics, 14(3), pp. 269-278, 2020.

[4] 김기윤, 허욱, 이세훈, 김종성, “보안 메신저 SureSpot 애플리케이션에 대한 포렌식 분석”, Journal of Digital Forensics, 13(3), pp. 175-188, 2019.

[5] 강수진, 김기윤, 김소람, 김종성, “macOS 환경에서의 Huawei 및 Apple 스마트폰 암호화 백업 데이터 복호화 및 아티팩트 분석”, Journal of Digital Forensics, 15(4), pp. 112-137, 2021.

[6] 박귀은, 강수진, 김종성, “안드로이드 환경에서의 지도 애플리케이션 아티팩트 분석 및 복호화 방안 연구”, Journal of Digital Forensics, 16(2), pp. 163-184, 2022.

[7] 김기윤, 이종혁, 신수민, 김종성, “인스턴트 메신저 말랑말랑 톡카페 애플리케이션 데이터베이스 복호화 방안 및 분석”, Journal of The Korea Institute of Information Security & Cryptology, 19(3), pp. 541-547, 2019.

[8] 신수민, “디지털 포렌식 관점에서 노트 및 다이어리 애플리케이션의 보안 프로세스 분석”, 2021.

[9] 김기윤, 장성우, 김종성, “LG 갤러리 애플리케이션 잠금 파일 복호화 연구”, Journal of Digital Forensic, 12(2), pp. 31-38, 2018

〈저 자 소 개〉

이 신 영 (Sinyoung Lee)

2018년 3월~현재 : 강남대학교 소프트웨어응용학부 재학

<관심분야> 정보보호,디지털 포렌식

김 한 결 (Hangyeol Kim)

2019년 3월~현재 : 강남대학교 소프트웨어응용학부 재학

<관심분야> 정보보호,디지털 포렌식

박 명 서 (Myungseo Park)

정회원

2013년 2월 : 국민대학교 수학과 졸업

2015년 2월 : 국민대학교 금융정보보안학과 석사

2014년 12월~2017년 2월 : 국가보안기술연구소 연구원

2021년 8월 : 국민대학교 금융정보보안학과 박사

2021년 9월~2022년 2월 : 국민대학교 금융정보보안학과 박사후연구원

2022년 3월~현재 : 강남대학교 ICT융합공학부 조교수

<관심분야> 정보보호, 디지털포렌식