Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.28, NO.1, Feb. 2018 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2018.28.1.111

안드로이드 환경의 OAuth 프로토콜을 이용한 원격지 데이터 수집 방법 연구

남 기 훈,† 공 성 현, 석 병 진, 이 창 훈‡
서울과학기술대학교 컴퓨터공학과

Study on Remote Data Acquisition Methods Using OAuth Protocol of Android Operating System

Gi-hoon Nam,† Seong-hyeon Gong, Byoung-jin Seok, Changhoon Lee‡
Department of Computer Science and Engineering, Seoul National University of Science and Technology

요 약

안드로이드 운영체제의 서드파티 애플리케이션들은 사용자의 계정 및 개인정보에 대한 접근 권한을 획득하기 위해 OAuth 프로토콜을 이용하여 계정 정보 제공자들로부터 사용자의 크리덴셜 및 사용자의 자원에 접근할 수 있는 권한을 내포하는 접근 토큰을 전달받는다. 이러한 크리덴셜 및 토큰 정보는 안드로이드 운영체제가 제공하는 OAuth 관련 데이터 관리 방식에 의해 기기 내부에 저장되는데, 이 정보가 유출될 경우 공격자는 유출된 크리덴셜 및 토큰 정보를 이용하여 사용자의 계정에 대한 자유로운 로그인 및 데이터 접근이 가능하다. 이러한 특징은 디지털 포렌식 수사관이 증거 데이터를 수집하는 관점에서 피압수자가 사용하는 서비스의 원격지 서버로부터 직접 데이터를 수집하는 것을 가능하게 한다. 원격지에서 수집되는 증거 데이터는 2차적인 영장발부의 토대로 작용함으로써 수집한 디지털 증거의 증거능력을 확보할 수 있기 때문에 피압수자가 안드로이드 기기에서 애플리케이션 삭제 등의 증거 인멸을 시도한 경우 매우 중요한 증거가 될 수 있다. 본 논문에서는 다양한 안드로이드 운영체제 및 기기 환경에서 OAuth 토큰의 관리 현황을 분석하고, 이를 이용하여 다양한 서드파티 애플리케이션의 데이터를 수집하는 방법을 소개한다. 이를 통해 디지털 포렌식 관점에서 피압수자가 사용하는 서비스의 원격지 데이터를 수집함으로써 증거 수집의 영역을 확장할 수 있는 방안을 제안한다.

ABSTRACT

Using OAuth protocol, third-party applications on the Android operating system use user's credentials or access tokens that have access authority on user's resources to gain user's account and personal information from account information providers. These credentials and token information are stored in the device by the OAuth data management method provided by the Android operating system. If this information is leaked, the attacker can use the leaked credential and token data to get user's personal data without login. This feature enables the digital forensic investigator to collect data directly from the remote server of the services used by the target of investigation in terms of collecting evidence data. Evidence data collected at a remote location can be a basis for secondary warranties and provide evidence which can be very important evidence when an attacker attempts to destroy evidence, such as the removal of an application from an Android device. In this paper, we analyze the management status of OAuth tokens in various Android operating system and device environment,

Received(11. 23. 2017), Modified(01. 09. 2018), Accepted(01. 11. 2018)

† 주저자, namggi90@gmail.com
‡ 교신저자, chlee@seoultech.ac.kr(Corresponding author)

and show how to collect data of various third party applications using it. This paper introduces a method of expanding the scope of data acquisition by collecting remote data of the services used by the subject of investigation from the viewpoint of digital forensics.

Keywords: Credential, Data Acquisition, Digital Forensics, Login Bypass, OAuth Protocol

I. 서 론

인터넷 산업 인프라가 확장됨에 따라 사용자들의 편의 증대를 위해 웹을 통해 서비스를 제공하는 웹 서비스의 수가 급증하였다. 구글 및 네이버와 같은 대형 포털사이트 및 페이스북, 트위터, 인스타그램 등의 소셜 네트워크 서비스들은 대규모 사용자 데이터를 바탕으로 서비스를 제공하고 있다. 대량의 사용자 정보를 가지고 있는 대형 온라인 서비스들은 네트워크상에서 리소스(사용자 관련 정보) 서버의 역할을 하며 리소스를 이용한 부가가치를 창출한다. 이들은 사용자들이 허가할 경우 API(Application Programming Interface)의 형태로 사용자들의 개인정보 및 개인 데이터들을 타 서드파티 애플리케이션에 제공함으로써 부가 서비스를 제공한다[1]. OAuth (Open Authorization) 프로토콜은 서드파티 애플리케이션이 사용자의 데이터를 리소스 서버로부터 획득하기 위한 방법으로, OAuth 2.0 프로토콜은 IETF (Internet Engineering Task Force)에서 제정하여 RFC (Request For Comments) 6749로 발표된 표준기술이다[2]. 온라인 서비스를 이용하는 사용자들은 OAuth 프로토콜을 이용하여 별도의 회원가입 없이 자신의 계정 정보 및 개인정보, 개인데이터 등을 서드파티 애플리케이션에게 제공할 수 있으며, 그것을 통해 서드파티 애플리케이션이 제공하는 서비스를 사용할 수 있다.

OAuth 이전에 사용되었던 인증 프로토콜들은 사용자의 아이디 및 패스워드와 같은 크리덴셜 데이터 자체를 서드파티 애플리케이션에게 제공함으로써 사용자 인증을 수행하였는데, 이러한 방식은 크리덴셜 데이터가 서드파티 서비스를 통해 유출될 수 있다는 단점이 있었다. 또한 발급된 크리덴셜들의 효력을 개별적으로 중단시킬 수 없었기 때문에 한 크리덴셜의 권한을 해제하기 위해선 발급했던 모든 크리덴셜을 해제함으로써 크리덴셜의 효력을 정지시킬 수밖에 없다[2]. 이는 인증상의 편의성을 크게 저하시키는 단점이며, 이를 해결하기 위한 OAuth 프로토콜은 OAuth 토큰 형태의 인증 데이터를 발급함으로써 사용자의 개인정보에 대한 서드파티 서비스의 접근을 허용하는 방식을 채택하였다. 그러나 OAuth 프로토콜 상에서도 토큰 데이터의 유출로 인한 개인정보 유출이 발생할 수 있다. OAuth 토큰 역시 일종의 데이터이기 때문에 토큰 데이터 관리상의 취약점 및 사용자의 부주의로 인해 유출될 수 있으며, 토큰의 사용에 대한 부가적인 인증 과정이 없다면 개인정보가 유출될 수 있다는 문제점이 존재한다. 안드로이드 스마트폰과 같이 다량의 사용자 인증 데이터가 운용되면서 동시에 분실에 대한 위험이 존재하는 기기의 경우 이러한 문제점이 가장 크게 드러나는 환경이다. 이러한 문제점에 대응하기 위해 일반적으로 안드로이드 기기에는 적절한 수준의 보안 장치[3]가 마련되어 있지만, 인증 정보 탈취에 대한 대응을 위해서는 추가적인 보안 수단이 필요한 상황이다.

안드로이드 기기에서의 이러한 문제점은 포렌식 관점에서 새로운 증거 수집의 방법이 될 수 있다. OAuth 토큰 정보를 이용해 서드파티 애플리케이션에 대한 로그인 우회를 시도할 수 있다는 점을 이용하여 증거 인멸 시도가 진행된 스마트폰 및 태블릿 PC 등의 기기에서 서드파티 애플리케이션의 크리덴셜을 획득하고, 이것을 활용하여 서버에 저장된 사용자의 행위가 저장된 데이터를 확보할 수 있으며, 이는 범죄자의 범죄 행위를 증명하기 위한 결정적인 증거가 될 수 있다. 2016년 10월 기준으로 스마트폰의 인터넷 이용률(51.3%)이 PC의 인터넷 이용률(48.7%)을 추월한 만큼[4], 스마트폰을 이용한 증거수집이 용이해질 경우 이는 매우 현실적인 디지털 증거 수집의 매개체가 될 수 있다. 안드로이드 기기에 대한 디지털 포렌식 절차 중 일차적인 기기 압수 및 이미징 과정을 거치는 동안 OAuth 토큰 및 크리덴셜 정보를 입수하고, 이를 바탕으로 서드파티 애플리케이션 데이터의 압수를 위한 2차 영장 발부가 진행되면 전자기기 및 전자 데이터에 대한 법제적인 한계[5]를 완화시킬 수 있다. 본 논문에서는 OAuth 토큰이 관리되는 방식 및 토큰이 저장되는 위치를 다양한 안드로이드 기기에서 분석하고 이들을 활용하여 로그인 과정을 우회하는 방법 및 서드파티 애플리케이션으로부터 서버에 저장되어있는 사용자 데이터를 수집하는 방법을 통해 증거 데이터를 수집

할 수 있는 방안을 제시한다.

본 논문의 2장에서는 OAuth 및 안드로이드 기기에서의 개인정보 유출, 로그인 우회와 관련된 연구들을 소개한다. 3장에서는 다양한 안드로이드 기기들에서 OAuth 토큰이 어떠한 방식으로 저장되어있는지에 대하여 분석한다. 4장에서는 획득한 OAuth 토큰 데이터를 이용하여 타 안드로이드 기기 및 안드로이드 가상환경에서 로그인 우회를 시도하는 과정을 설명하고, 5장에서 결론 및 향후 연구 방향을 논한다.

II. 관련 연구

2.1 OAuth 2.0 프레임워크

OAuth 2.0 프레임워크는 자원 소유자와 HTTP 서비스 사이의 인증 과정을 조율함으로써 서드파티 애플리케이션이 HTTP 서비스 및 자원 소유자의 행동에 대한 제한된 접근 권한을 얻도록 하는 사용자 인증 프레임워크다[2]. OAuth 2.0 프로토콜은 서드파티 애플리케이션에게 제한된 자원에 대한 접근을 허가하기 위해 인증코드(Authorization Code), Implicit, 리소스 소유자 암호 자격 증명(Resource Owner Password Credentials), 클라이언트 자격 증명(Client Credentials) 네 가지 형태의 인가 방식을 제공한다[2, 6].

인증 코드(Authorization Code)를 이용한 방식은 기밀성이 필요한 클라이언트 서비스에서 주로 이용된다. 이 방식은 리소스 서버가 사용자의 인증 요청을 허가한 후, 클라이언트에게 인증 코드를 제공하며, 클라이언트는 인가받은 인증 코드 및 client id, client secret 정보를 이용하여 리소스 서버에 대한 접근 권한을 획득한다.

Implicit 방식은 클라이언트가 사용하는 권한인가 방식으로써, 브라우저 기반의 서비스 및 모바일 애플리케이션 기반의 서비스에 활용되며 URL을 이용하여 접근 토큰을 제공한다.

리소스 소유자 암호 자격 증명(Resource Owner Password Credentials)방식은 클라이언트 내에 직접 사용자의 아이디 및 패스워드와 같은 크리덴셜 정보를 저장하고, 이를 이용하여 클라이언트가 직접 리소스 서버에게 접근 토큰을 요청하는 방식이다.

클라이언트 자격 증명(Client Credentials)방식은 클라이언트가 client_id, client_secret, return URL 등의 인증 정보를 이용하여 클라이언트의 인증을 수행하고 접근 토큰을 발급받는 형태의 인증 방식이다[6].

2.2 OAuth 프로토콜을 이용한 서드파티 인증

OAuth 토큰을 이용하면 서드파티 애플리케이션이 특정 사용자의 계정 정보 및 개인 정보를 리소스 서버로부터 획득하여 서비스 제공에 활용할 수 있다. 사용자가 애플리케이션에 서비스를 요청하면, 애플리케이션은 사용자의 리소스를 사용하기 위해 사용자의 리소스를 저장 및 관리하고 있는 외부 리소스 서버에 대한 접근 권한을 사용자에게 요청한다. 사용자는 애플리케이션으로부터 접근 권한 허가 요청이 오면 해당 리소스를 보유하고 있는 리소스 서버에게 우선적으로 자신의 크리덴셜 정보를 제공함으로써 사용자 자신에 대한 인증을 수행한다. 사용자 자신에 대한 인증에 성공하면 리소스 서버는 사용자에 대한 인증이 성공했다는 메시지가 담긴 일회성 인증 코드를 사용자에게 반환하며, 사용자는 이 일회성 인증 코드를 애플리케이션에게 제공한다. 애플리케이션은 클라이언트의 아이디 및 크리덴셜 정보와 함께 이 일회성 인증 코드를 리소스서버로 전송함으로써 리소스 서버에게 자신이 정상적으로 사용자의 리소스에 대한 접근 권한을 요청하고 있음을 증명하며, 리소스 서버는 자신이 발급한 일회성 인증 코드가 담긴 접근 허가 요청을 확인하고, 해당 애플리케이션에게 향후 특정 리소스에게 자유롭게 접근할 수 있는 권한을 내포하고 있는 접근 토큰을 제공한다. 애플리케이션은 리소스 서버로부터 인가받은 접근 토큰을 이용하여 제한된 리소스 영역에 자유롭게 접근할 수 있으며, 이를 이용하여 사용자에게 서비스를 제공할 수 있다. Fig. 1은 서드파티 애플리케이션이 사용자를 통하여 리소스 서버에게 접근 토큰을 인가받는 전체 과정에 대한 프로세스를 나타낸다[2].

① 서드파티 애플리케이션 서비스 요청
② 사용자 리소스 서버 접근 토큰 요청
③ 사용자는 아이디/패스워드를 인증하여 리소스 서버에 크리덴셜 공급
④ 사용자에게 인증 코드 발급
⑤ 발급 받은 인증 코드를 서드파티 애플리케이션 서버에 전송

⑥ 사용자의 리소스 정보를 권한을 얻기 위해 사용자 인증 코드를 리소스 서버에 전송
⑦ 사용자 리소스 접근 권한을 위한 접근 코드 발급
⑧ 접근 코드를 리소스 서버에 전송
⑨ 사용자 리소스 권한 획득
⑩ 사용자에게 애플리케이션 서비스 제공

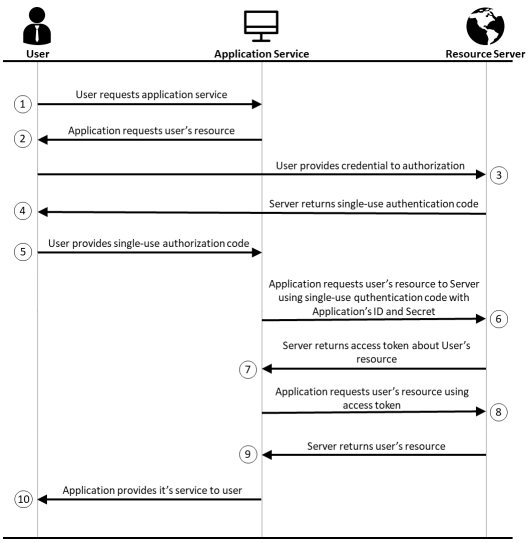

Fig. 1. Third party application authorizati on process using OAuth protocol

2.3 구글 계정 애플리케이션의 취약점

안드로이드의 지메일, 구글 드라이브 등 구글(Google) 애플리케이션에는 크리덴셜 유출 가능성으로 인한 개인정보 유출에 대한 취약점이 존재한다. 김진욱 외 3인은 안드로이드 스마트폰 내부에 저장된 구글 계정 서비스의 크리덴셜 정보를 획득 및 이를 이용하여 타 기기에서의 자동 로그인을 시도하였다[6]. 구글 계정 애플리케이션은 구글이 제공하는 드라이브, 메일 서비스와 같은 구글 자체 서드파티 애플리케이션에 대한 접근 권한을 안드로이드 기기 내부의 특정 데이터베이스 파일에 저장한다. 해당 크리덴셜 정보를 타 기기에 주입하면, 크리덴셜 중 토큰 데이터에 포함된 서드파티 항목들에 대하여 해당 사용자의 계정과 관련된 개인 데이터들을 동기화할 수 있다.

본 연구는 구글계정 뿐만 아니라 네이버, QQ 등의 애플리케이션에서 사용되는 크리덴셜 관리 방식의 취약점을 활용한 로그인 우회 방법과 해당 크리덴셜을 OAuth로 활용하는 다른 서드파티 애플리케이션들에 대한 로그인 우회를 연구했다. 우리는 이것을 디지털 포렌식 데이터 수집 관점에서 피압수자의 원격지 데이터 수집 방법으로 제안한다

III. 분석 대상 및 크리덴셜 분석

3.1 분석 대상 및 환경

안드로이드 기기의 OAuth 크리덴셜에 대한 분석은 국내 시장점유율이 높은 안드로이드 스마트폰 제조사 두 곳의 제품을 대상으로 진행하였다. Table. 1은 분석에 사용된 안드로이드 기기들의 정보다.

안드로이드 기기상에서의 분석은 특정 기기에서 크리덴셜 및 OAuth 토큰을 획득하여 동일 기기에 재 주입, 동일 기종의 기기에 주입, 타 기종의 기기에 주입을 각각 수행하는 방식으로 진행되었다. 스마트폰의 기기 정보를 변경해야 하는 경우 안드로이드 가상환경을 이용하여 분석하였다. 가상환경은 디바이스 루팅 과정이 간단하며 모바일 기기의 IMEI (International Mobile Equipment Identity) 값을 임의로 설정할 수 있다는 장점이 있다. Fig. 2는 가상 안드로이드 환경의 모습이며[7] Fig. 3은 가상 안드로이드 환경에서 ROOT권한과 디바이스 설정 및 모델, IMEI를 변경하여 분석에 적절한 환경을 구성하는 모습을 나타낸다.

Table. 2는 OAuth 토큰 주입을 이용한 로그인 우회 분석 대상 애플리케이션들의 목록으로, 각 안드로이드 디바이스 환경에서 분석에 사용된 구글, 네이버, QQ 애플리케이션들의 apk 명과 버전을 나타낸

Table. 1. List of analyzed Android based devices

Android device	Device Model	Android version	Kernel version	Build number
Samsung Galaxy S4	SHV-E330K	4.4.2	3.4.0-2933244	KOT49H.E330KKKUCNJ1
Samsung Galaxy S6	SM-G920S	7.0	3.10.61-11594515	NRD90M.G920LKLU3EQF3
Samsung Galaxy S7	SM-G930S	7.0	3.18.14-11698140	NRD90M.G930SKIS1DAG1
LG G4	LG-F500S	5.1	3.10.49	LAY47D

Fig. 2. Android Virtual Environment (Mumu version 3.3.1)

Fig. 3. Settings of virtual environment

다. 명시되어 있는 버전들은 모두 로그인 우회가 가능함을 확인하였다.

Table. 2. List of bypassing login possible application

Device Model	Application name	Package name	Version name
SHV-E330K	Goggle	com.google.android.gsf.login	7.13.28.16.arm
	Naver	com.nhn.android.search	8.2.3
	Tencent QQ	com.tencent.mobileqqi	5.2.1
SM-G920S	Goggle	com.google.android.gsf.login	7.15.22.21.arm64
	Naver	com.nhn.android.search	8.3.1
	Tencent QQ	com.tencent.mobileqqi	5.2.1
SM-G930S	Goggle	com.google.android.gsf.login	7.13.28.21.arm64
	Naver	com.nhn.android.search	8.2.3
	Tencent QQ	com.tencent.mobileqqi	5.2.1
LG-F500S	Goggle	com.google.android.gsf.login	7.13.28.21.arm64
	Naver	com.nhn.android.search	8.3.1
	Tencent QQ	com.tencent.mobileqqi	5.2.1

3.2 안드로이드 폰의 크리덴셜 분석

3.2.1 네이버 계정 크리덴셜 분석

네이버는 국내의 대표적인 포털사이트로 검색 엔진 및 뉴스뿐만 아니라 블로그, 카페 등 다양한 서비스 및 서드파티 서비스를 제공한다. 네이버는 국내 64%의 검색엔진 시장점유율을 보유하고 있으며, 대량의 사용자 데이터양을 바탕으로 다양한 서드파티 서비스들에 OAuth 로그인 서비스 API를 제공하고 있다[8]. 네이버는 안드로이드에서 제공하는 OAuth 크리덴셜 관리 방법과 동일한 방법을 사용한다. 사용자가 네이버에 로그인을 수행하면, 네이버 서비스의 크리덴셜 정보가 /data/system_ce/0, /data/system_de/0 두 폴더에 각각 존재하는 accounts_ce.db, accounts_de.db 데이터베이스에 저장된다. Fig. 4, Fig. 5, Fig. 6은 실제 안드로이드 스마트폰 기기(갤럭시 S6: SM - G920S)에 저장된 네이버 로그인 크리덴셜의 내용을 DB Browser for Salite[9] 프로그램으로 확인한 결과다.

Fig. 4는 accounts_ce.db 파일의 accounts 테이블에 존재하는 데이터이다. 해당 테이블에서 네

	_id	name	type	password
	필터	필터	필터	필터
1	3	stcis1234@g…	com.google	aas_et/AKppINYoko…
2	5	stcis221	com.nhn.android.naveraccount	NULL

Fig. 4. accounts_ce.db of Naver login credentials (/data/system_ce/0/accounts_ce.db)

	_id	accounts_id	key	value
	필터	필터	필터	필터
1	178	5	key_tokenvalid	valid
2	177	5	key_tokensecret	zvje9DYi9vvG···
3	180	5	key_token_created	1511020520
4	179	5	key_token_changed	1511020520
5	176	5	key_token	AAAAjRixt/ AivPuAgILp7···
6	95	3	services	talk,friendvie···

Fig. 5. Extra credentials of Naver(/data/system_ce/0/accounts_ce.db)

이버 계정 명과 크리덴셜 ID를 확인할 수 있다.

Fig. 5는 accounts_ce.db 파일의 extras 테이블 안에 존재하는 크리덴셜 정보다. 네이버 크리덴셜 ID와 동일한 번호로 매칭된 accounts_id를 가진 크리덴셜 데이터들이 존재하며, 해당 데이터들은 네이버 웹 서비스에 로그인하기 위해 필요한 로그인 크리덴셜 및 로그인 후 이용할 수 있는 서비스와 관련된 정보들을 내포하고 있다.

안드로이드 버전 7.0 전까지는 OAuth 관련 크리덴셜 정보가 /data/system/0/accounts.db 파일에 일괄적으로 저장되었으나 안드로이드 버전 7.0부터 accounts_ce.db, accounts_de.db 두 개의 DB파일에 나뉘어 저장된다. Fig. 6은 분석 환경(Android 7.0)에서 별도로 관리되는 크리덴셜 데이터다.

	_id	name	type	previous_name	rd_entry_time_
	필터	필터	필터	필터	필터
1	5	stcis221	com.nhn.an···	NULL	1511020519824

Fig. 6. accounts_de.db of Naver login credentials (/data/system_de/0/accounts_de.db)

3.2.2 구글 계정 크리덴셜 분석

구글 크리덴셜의 위치는 선행연구에서 밝혀진 바 있다[10]. Fig. 7, Fig. 8은 안드로이드 환경에서 운용되는 구글 계정의 크리덴셜 및 OAuth 토큰 정보를 나타낸다. 그러나 크리덴셜 및 OAuth 토큰 데이터가 저장되는 경로는 안드로이드 버전별로 상이하며, 4장에서 후술한다.

	_id	name	type	password
	필터	필터	필터	필터
1	3	stcis1234@g···	com.google	aas_et/AKppINYokoe0pbeGI-PZuAo60JVgG3Fe8owJd6G2E0CRSEe7aVfDeL-ZX···

Fig. 7. account table of Google login credentials

	_id	accounts_id	type	authtoken
		필터	필터	필터
1	1	1	SID	GQWxvjxl4rvfbNAN···
2	2	1	LSID	GQWxvqr7OINOoK···
3	3	1	com.google.andro···	GQWxvjxl4rvfbNAN···
4	4	1	com.google.andro···	GQWxvqr7OINOoK···
5	10	1	com.google.andro···	ya29.Gmy1BKYSb···
6	14	1	com.google.andro···	ya29.GooBtQSaZ···
7	15	1	com.google.andro···	GQWxvuhZPrrVnk···
8	22	1	com.android.vendi···	GQWxvrvuD8iSSk···
9	24	1	com.google.andro···	GQWxvvK9Bhi7FN···
10	26	1	com.google.andro···	GQWxvm-VQim1sozhPjtjPjT···
11	27	1	com.google.andro···	GQWxvqbCRB4ho···
12	29	1	com.google.andro···	GQWxvj1M3K1wjp···
13	30	1	com.google.andro···	ya29.GowBtQShv···
14	31	1	com.google.andro···	INVALID_TOKEN
15	33	1	com.google.andro···	GQWxvmkuQvAUR···
16	135	1	com.instagram.an···	eyJhbGciOiJSUzI1···
17	157	1	com.google.andro···	ya29.GowBvASCR···

Fig. 8. authtokens table of Google login credentials

3.2.3 텐센트 QQ(Tencent QQ)의 크리덴셜 분석

QQ는 중국 텐센트(Tencent)사에서 제공하는 무료 IM 서비스 어플리케이션으로 QQ은 모바일 환경과 PC 환경에서 모두 이용 가능하며 제공하는 기능은 문자, 이미지 및 동영상 전송과 더불어 음성 통화, 영상 통화, 파일 전송, 사용자 위치 기반 친구 추천 등 여러 다양한 기능을 제공한다. QQ는 전세계적으로 약 80개 국가에서 20억 명의 사용자가 등록되어 있으며 위챗(Wechat)과 더불어 가장 많이 사용되는 중국의 대표적 통신 애플리케이션이다[11][12]. 중국 텐센트(Tencent)사의 QQ는 OAuth 2.0 프로토콜을 이용하여 서드파티 애플리케이션에 대한 API를 제공하고 있다[13]. QQ는 OAuth 크리덴셜 정보를 운영체제 레벨에서 관리하지 않고 애플리케이션의 내부에서 관리하고 있기 때문에 크리덴셜 정보가 위에서 분석되었던 타 앱들과는 다른 경로에 존재한다. Fig. 9는 QQ 인터네셔널(QQ International) 애플리케이션의 경로 및 애플리케이션 경로에 존재하는 폴더 구조다.

QQ 애플리케이션의 사용자 정보를 내포하고 있는 폴더는 /data/data/com.tencent.mobileqqi/

databases, /data/data/com.tencent.mobileqqi/files, /data/data/com.tencent.mobileqqi/shared_prefs 들이며, 이중 files 폴더 내에 존재하는 ConfigStore2.dat 파일에 사용자 계정에 대한 크리덴셜이 저장된다. 파일 내부에 존재하는 사용자의 로그인 토큰 값을 산출하는 과정에는 사용자의 아이디, 패스워드에 해당하는 계정 정보와 함께 모바일 기기의 IMEI(International Mobile Equipment Identity) 값이 함께 사용된다.

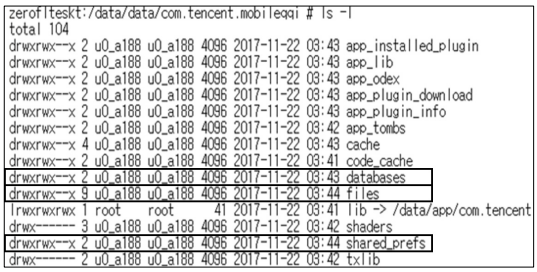

Fig. 9. directory of Mobile QQ International application

IV. 크리덴셜을 이용한 로그인 우회

4.1 로그인 우회 과정

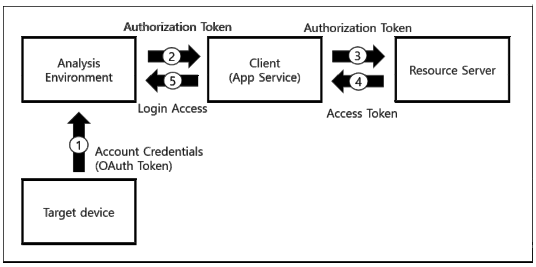

Fig. 10. Flow chart for detouring Login using OAuth Credentials

① 분석대상 모바일 기기에서 추출한 크리덴셜을 분석 환경에 주입
② 크리덴셜에 저장되어 있는 인증 토큰(Authorization Token)을 클라이언트에 전송
③ 리소스 서버에 인증 토큰(Authorization Token) 전송
④ 사용자 권한을 얻기 위한 접근 토큰(Access Token) 발급
⑤ 사용자에게 서비스 제공

본 절에서는 3장에서 분석한 로그인 크리덴셜을 이용하여 타 기기 및 타 환경에서 사용자의 계정으로 로그인을 수행하는 과정을 설명한다. 분석가가 특정 사용자의 안드로이드 기기로부터 로그인 크리덴셜이 저장되어있는 데이터베이스를 확보할 수 있다면, 분석가는 해당 데이터베이스 내에 존재하는 크리덴셜 정보를 타 안드로이드 기기 및 안드로이드 가상환경에 주입함으로써 분석 대상 사용자의 계정으로 로그인 우회를 할 수 있다. Fig. 10은 분석 대상 기기로부터 얻어진 크리덴셜 정보를 사용하여 로그인 우회하는 과정을 도식화 한 것이다.

분석가는 분석 대상이 되는 기기로부터 크리덴셜이 저장된 데이터베이스에 접근 및 해당 데이터를 추출한다. 일반적인 안드로이드 환경에서 기기 내부에 존재하는 데이터베이스에 접근하기 위해서는 루트(root) 권한이 요구되기 때문에, 이 과정에서 해당 기기에 대한 루팅(rooting) 과정이 선행되어야 한다. 애플리케이션은 구글, 네이버 등의 리소스 서버가 제공하는 사용자의 개인 리소스에 접근하기 위해 해당 리소스에 대한 접근 권한을 리소스 서버에 요청해야 하고, 접근 권한의 요청을 위해 주입된 크리덴셜 정보로 인증 토큰을 서버에 요청한다. 인증 토큰으로써 리소스 서버로 전송된 크리덴셜 정보는 서버의 인증과정에 사용되며, 서버는 해당 크리덴셜이 초기의 정상 사용자로부터 생성된 정보이기 때문에 정상적으로 인증 과정을 수행한다. 리소스 서버는 사용자의 크리덴셜 정보를 정상적으로 인증한 후, 애플리케이션에게 사용자의 리소스에 대한 접근 권한을 허가하기 위해 접근 권한이 담긴 접근 토큰을 전달한다. 애플리케이션은 이러한 접근 토큰을 이용하여 해당 사용자의 계정으로 서드파티 애플리케이션에 로그인을 수행할 수 있다.

Fig. 4, Fig. 5, Fig. 6에서 추출한 사용자의 OAuth 토큰이 포함된 크리덴셜을 타 기기에 주입하면, Fig. 11과 같이 해당 사용자의 계정 정보로 로그인이 가능함을 확인할 수 있다.

OAuth 토큰이 담긴 크리덴셜을 통해 분석 대상

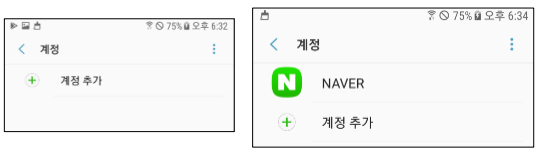

Fig. 11. Login Bypass on Naver application using OAuth Token Injection

계정으로 로그인할 경우, 사용자와 관련된 모든 개인 정보 및 사용자의 개인 데이터를 해당 애플리케이션의 서버로부터 동기화할 수 있다.

/data/system/0/accounts.db (또는 /data/system_ce/0/accounts_ce.db 및 /data/system_de/0/accounts_de.db) 파일 중 accounts 테이블 및 extras 테이블에 계정 크리덴셜 및 OAuth 접근 토큰 데이터가 존재하며, 해당 데이터를 로그인하고자 하는 기기의 동일 데이터베이스에 주입하는 것으로 크리덴셜 정보를 주입할 수 있다. Fig. 12는 OAuth 토큰 데이터를 포함한 크리덴셜을 주입함으로써 해당 리소스 서버의 계정정보 및 개인정보를 동기화한 결과다.

서드파티 애플리케이션들은 대량의 사용자 정보를 보유하고 있는 리소스 서버의 자원을 사용하기 위해 사용자에게 리소스 서버에 대한 접근 권한을 요청한다. 사용자는 자신의 크리덴셜을 이용한 인증 과정을 통해 서드파티 애플리케이션에게 리소스 서버에 저장된 자신의 데이터에 접근할 수 있는 권한을 부여할 수 있으며, 이것을 이용해 사용자는 서드파티 애플리케이션에 대한 별도의 회원가입 과정 없이 해당 서비스를 이용할 수 있다.

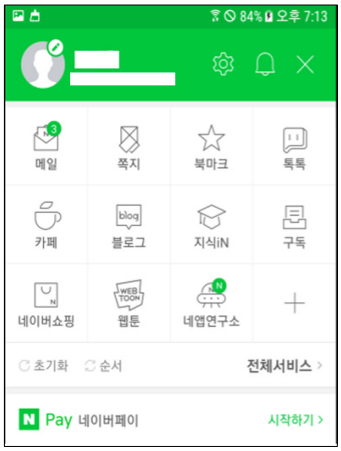

Fig. 12. Access to personal information using OAuth Token Injection

4.2 가상머신 환경에서의 로그인 우회

텐센트 QQ(Tencent QQ) 애플리케이션의 경우, 사용자의 크리덴셜을 이용하여 인증 토큰을 생성하는 과정에서 모바일 기기의 IMEI(International Mobile Equipment Identity)값이 사용되기 때문에 IMEI를 조작할 수 있는 환경 하에서만 로그인 우회가 가능하다. 분석하고자 하는 대상 모바일 기기에서 애플리케이션 경로 내의 /data/data/com.tencent.mobil-eqqi/files 폴더 내에 존재하는 ConfigStore2.dat 파일을 타 기기 및 가상환경에 주입하고 모바일 기기를 재부팅한 후 애플리케이션을 실행시키면 해당 사용자의 계정으로 자동 로그인이 되는 것을 확인할 수 있다. Fig. 13, Fig. 14는 OAuth 토큰 주입을 통해 QQ 인터네셔널(QQ International) 애플리케이션에 로그인을 성공한 화면이다.

QQ 인터네셔널(QQ International) 애플리케이션은 로그인 후 서버와 통신하여 정상적인 접근 토큰의 사용 여부를 확인한다. IMEI가 동일한 환경에서만 해당 과정을 통과할 수 있으며, IMEI가 일치하지 않는 환경에서는 자동로그인을 수행한 직후 에러메시지가 발생하면서 사용자의 계정이 로그아웃된다. 하지만 이것은 서버와의 통신이 필요한 절차이기 때문에 모바일 기기의 네트워크 기능을 중지시킨 상태에서 애플리케이션을 실행시키면 사용자의 계정으

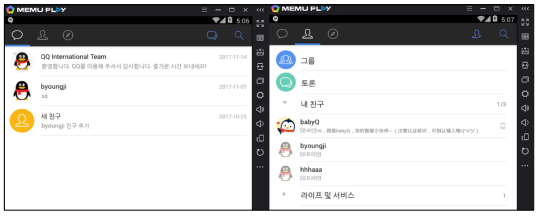

Fig. 13. Login success using OAuth credential injection (QQ International)

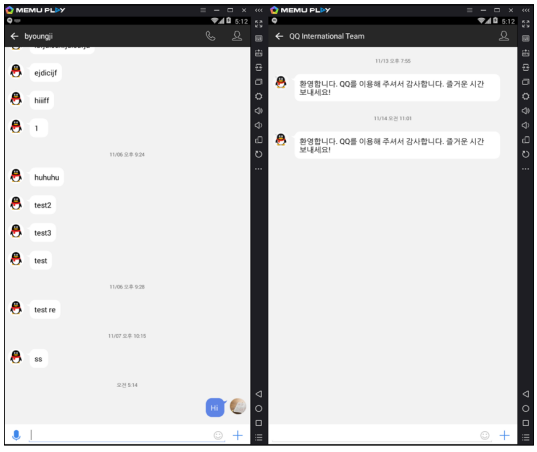

Fig. 14. Successful chatting history synchronization and chatting

로 로그인 된 상태를 유지할 수 있다. 만약 ConfigStore2.dat 파일 이외에도 사용자의 모바일 기기에서 QQ 애플리케이션 경로에 존재하는 폴더 중 databases, shared_pref 두 폴더를 함께 취득하여 주입한다면, 사용자가 가지고 있는 폰 내부에 저장된 대화 내역 및 사용자 정보들 또한 복구가 가능하다. 이 경우에도 모바일 기기의 네트워크 기능이 켜지는 즉시 서버와 통신하여 정상 로그인 여부를 판단하며, 그 즉시 로그아웃이 수행된다.

4.3 로그인 우회가 가능한 서드파티 애플리케이션

리소스 서버는 일반적으로 자신이 부여한 접근 토큰에 대한 2차적인 보안 과정을 거치지 않기 때문에 애플리케이션 자체적으로 접근 토큰에 대한 2차적인증 과정이 존재하지 않을 경우, 분석가는 주입된 접근 토큰을 획득하여 사용자의 계정정보 및 개인정보를 쉽게 취득할 수 있다. 이를 통해 유출될 수 있는 사용자의 개인정보는 일반적으로 정상 사용자가 정상적인 절차를 통하여 로그인을 수행하였을 경우에 제공되는 모든 데이터와 동일하다. 즉 이메일 내용 및 거래 내역과 같이 중요도가 높은 개인정보가 노출될 수 있다. 본 연구에서는 OAuth 프로토콜을 이용하여 서드파티 애플리케이션에 대한 접근이 가능함을 실험하기 위해 국내에서 일반적으로 사용되는 임의의 애플리케이션을 선정하고, 해당 애플리케이션에 대한 로그인 시도 실험을 진행하였다. 인지도 및 사용 빈도를 기준으로 10개의 서드파티 애플리케이션을 선정하였으며, 본 논문에서는 대표적으로 '티몬' 애플리케이션에 대한 로그인 실험의 결과를 제시한다. '티몬'은 월 거래액이 2000억원을 상회하는 대형 쇼핑몰 애플리케이션이다[14]. Fig. 15은 OAuth 프로토콜을 이용한 API를 제공하는 OAuth 프로바이더의 접근 토큰 정보를 '티몬' 애플리케이션에 로그인 한 결과다. OAuth 프로토콜을 이용하여 사용자의 개인정보 및 구매 내역등에 대한 정보를 모두 열람할 수 있음을 확인할 수 있다.

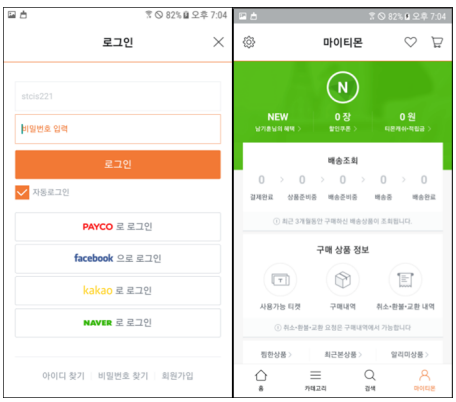

Fig. 15. Login Bypass on Third party application using resource server's OAuth injection

다음 Table. 3은 구글 및 네이버의 크리덴셜 정보를 이용하여 로그인 우회가 가능한 서드파티 애플리케이션과 해당 애플리케이션에서 획득할 수 있는 사용자의 개인 정보 항목이다.

Table. 3. Leakable personal information from third-party application using Oauth token

OAuth Provider	Application	App. function	Leakable Personal Information
Google	Gmail	email	email content
	GoogleDrive	cloud storage	file
	GoogleMap	map	location
	GooglePhoto	photo storage	file (image)
	GoogleCalendar	calender	personal schedule
	Google Hangout	chatting	chatting dialogue
	Dropbox	cloud storage	file
	SkyScanner	airline	purchase list
Naver	Naver mail	email	email content
	Interpark	shopping	purchase list
	JinAir	airline	purchase list
	Ndrive	cloud storage	file
	Timon	shopping	purchase list
	Onestore	app market	purchase list
	Coocha	shopping	purchase list
	Hotelscombined	accomomodation	purchase list
	Booking.com	accomomodation	purchase list
Tencent QQ	QQ International	chatting	chatting dialogue

텐센트사의 QQ의 경우 서드파티 애플리케이션에서 QQ의 OAuth 토큰정보를 이용하여 로그인을 시도하면, 정상 사용자 여부 판별을 위해 사용자의 아이디 및 패스워드를 다시 요구하는 과정이 수행된다. 따라서 QQ의 OAuth 토큰을 이용하면 QQ 애플리케이션의 원격지 데이터 자체는 수집이 가능하지만, QQ의 OAuth 기능을 이용하여 서비스를 제공하는 타 서드파티 애플리케이션에 대한 데이터 수집은 불가능하다.

4.4 OAuth 토큰의 경로

안드로이드 자체적으로 제공하는 OAuth 토큰 관리 방법을 사용하는 서비스의 경우 OAuth 토큰 데이터는 동일한 데이터베이스 파일에서 일괄적으로 관리된다. 이러한 방법을 준수하지 않는 애플리케이션의 경우(QQ 등) 애플리케이션 내부에 크리덴셜 및 OAuth 토큰 데이터를 보관한다. Table. 4는 각 안드로이드 기기 및 안드로이드 버전별 계정 크리덴셜 및 OAuth 토큰 데이터가 저장되는 위치를 정리한 표다.

Table. 4. Directory location of OAuth Token Data in various Android Versions and Devices

Mobile Device Model	Device Model	Android version	Credential Location	File name
Samsung Galaxy S4	SHV-E330K	4.2.2	/data/system/users/0	accounts.db
Samsung Galaxy S6	SM-G920S	7.0	/data/system_ce/0	accounts_ce.db
			/data/system_de/0	accounts_de.db
Samsung Galaxy S7	SM-G930S	7.0	/data/system_ce/0	accounts_ce.db
			/data/system_de/0	accounts_de.db
LG G4	LG-F500S	5.1	/data/system/users/0	accounts.db

V. 결론 및 향후연구

OAuth 프로토콜은 사용자의 리소스를 타 서드파티 애플리케이션에게 제공하는 과정을 용이하게 함으로써 통합된 온라인 서비스 환경을 구축하기 위한 프로토콜이다. 이는 사용자에게 회원가입 및 로그인 등의 절차를 간소화할 수 있다는 편의성을 제공하며 서비스 제공자의 입장에서는 대규모 사용자들의 데이터를 공유 및 사용하기 용이해진다는 장점이 있지만 사용자와 관련된 증거데이터를 수집하는 관점에서도 유용한 방법으로 사용할 수 있다. 모바일 기기의 사용량이 PC의 사용량을 넘어선 만큼 모바일 기기 내부에 존재하는 데이터는 사용자의 행위 및 범죄의 입증에 대한 결정적인 역할을 수행할 가능성이 높다. 하지만 피압수자가 애플리케이션을 삭제하는 등의 증거인멸을 시도하면, 최신 안드로이드 버전의 모바일 기기에서는 삭제된 데이터의 복구가 어렵기 때문에 서버에 저장된 피압수자의 데이터를 수집할 필요가 있다. 본 연구는 OAuth 프로토콜의 토큰 관리 방식을 분석하여 원격지 서버에 저장된 사용자의 데이터를 복구 및 동기화할 수 있는 방안을 제시하였다. 연구 결과, 하나의 OAuth 크리덴셜만 확보하더라도 다수의 서드파티 애플리케이션에 대한 접근이 가능하기 때문에 다양한 형태의 사용자 데이터를 수집할 수 있다는 것이 확인되었다. 1차적인 모바일 데이터 수집 및 분석을 통해 확보한 구글, 네이버, QQ 등의 크리덴셜과 OAuth 정보를 통해 해당 서비스뿐만 아니라 연계된 다른 서비스들의 서버에 저장된 피압수자의 데이터들도 수집할 수 있다. 이러한 원격지 데이터 수집은 추가적인 영장 발부가 필요하며 우리가 제안한 방법은 구글 등 서버가 외국에 존재하는 서비스의 데이터 수집에 유용하게 활용될 수 있을 것이다.

본 논문에서 제안한 OAuth를 이용한 데이터 수집 방법은 원격지에 존재하는 피압수자의 데이터를 수집할 수 있다는 장점이 있다. 하지만 실제 기기 환경 및 안드로이드 가상환경을 필요로 하기 때문에 기존의 포렌식 도구와 직접적인 연동이 쉽지 않다는 단점이 있다. 향후 연구에서는 분석을 위한 안드로이드 기기 및 에뮬레이터 없이, HTTP 프로토콜을 사용하여 PC에서 OAuth 프로토콜을 통한 원격지 서버 데이터 수집 방법을 연구하여 기존의 포렌식 도구에서 활용될 수 있도록 할 것이다.

References

[1] Login Authorization Documentation, Available on: http://www.oauthlogin.com

[2] D. Hardt, Ed., "OAuth 2.0 Autorization Framework", Internet Engineering Task

Force (IEFT) RFC 6749, Oct 2012.
〔3〕 Enck, W., Ongtang, M., & McDaniel, P. "Understanding android security." IEEE security & privacy, 7(1), pp. 50-57. Feb. 2009.
〔4〕 Trend Spectrum, "2017 Mobile Trend Prospect", DigiEco, Jan 2017.
〔5〕 Choi Yoonjung, "Legal Study of the Warrant in Principle and the Exception about Seizure and Search of Electronically Stored Information", Justice (154), pp. 110-144, Apr 2016.
〔6〕 Kim Jinouk, Jungsoo Park, Long Nguyen-Vu, Souhwan Jung, "A Study on Vulnerability Prevention Mechanism Due to Logout Problem Using OAuth", Jounal of The Korea Institute of Information Security & Cryptography, 27(1), pp. 5-14, Feb 2017.
〔7〕 Android emulator for PC, better than Bluestacks, Available on: http://www.memuplay.com
〔8〕 Naver Developers Documents, Available on: https://developers.naver.com/docs/login/api
〔9〕 DB Browser for SQLite, Available on: http://sqlitebrowser.org
〔10〕 Choi Jongwon, Yi Jeonghyun, "Analysis on Personal Information Leakage of Google Account App on Android", Journal of Digital Forensics, 8(2), pp. 65-81, Dec 2014.
〔11〕 Product & Service of Tencent company, Tencent, Available on: https://www.tencent.com/en-us/system.html
〔12〕 QQ International Application for Android, Tencent, Available on: https://play.google.com/store/apps/details?id=com.tencent.mobileqqi
〔13〕 OAuth 2.0 Introduction, Tencent, Available on: http://wiki.open.qq.com/wiki/mobile/OAuth2.0%E7%AE%80%E4%BB%8B
〔14〕 Growth Story, Tmon, Available on: http://corp.ticketmonster.co.kr

〈저자소개〉

남 기 훈 (Gi-hoon Nam) 학생회원
2011년 3월: 목원대학교 컴퓨터교육과 학사
2017년 3월~ 현재: 서울과학기술대학교 컴퓨터공학과 석사과정
〈관심분야〉 정보보호, 디지털포렌식, 암호학 등

공 성 현 (Seong-hyeon Gong) 학생회원
2012년 3월: 서울과학기술대학교 컴퓨터공학과 학사
2016년 3월: 서울과학기술대학교 컴퓨터공학과 석사
2018년 3월~ 현재: 서울과학기술대학교 컴퓨터공학과 박사과정
〈관심분야〉 정보보호, CTI, 네트워크 보안, 디지털포렌식 등

석 병 진 (Byoung-jin Seok) 학생회원
2012년 3월~2017년 8월: 서울과학기술대학교 컴퓨터공학과 학사
2017년 9월~현재: 서울과학기술대학교 일반대학원 컴퓨터공학과 석사과정
〈관심분야〉 정보보호, 암호학, 디지털포렌식 등

이 창 훈 (Changhoon Lee) 종신회원
2001년 3월: 한양대학교 자연과학부 수학전공 학사
2003년 3월: 고려대학교 정보보호대학원 석사
2008년 3월: 고려대학교 정보경영전문대학원 정보보호전공 박사
2008년 4월~2008년 12월: 고려대학교 정보보호연구원 연구교수
2009년 3월~2012년 2월: 한신대학교 컴퓨터공학부 조교수
2012년 3월~2015년 3월: 서울과학기술대학교 컴퓨터공학과 조교수
2015년 4월~현재: 서울과학기술대학교 컴퓨터공학과 부교수
〈관심분야〉 정보보호, 블록체인, CTI, IoT보안, 디지털포렌식, 암호학 등