



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2020-0054728
(43) 공개일자 2020년05월20일

(51) 국제특허분류(Int. Cl.)
G06F 21/16 (2018.01) G06F 21/56 (2013.01)
(52) CPC특허분류
G06F 21/16 (2019.02)
G06F 21/56 (2013.01)
(21) 출원번호 10-2018-0138391
(22) 출원일자 2018년11월12일
심사청구일자 없음

(71) 출원인
충남대학교산학협력단
대전광역시 유성구 대학로 99 (궁동, 충남대학교)
(72) 발명자
류재철
대전광역시 유성구 어은로 57, 132동 801호
이우람
대전광역시 유성구 온천북로33번길 21-16, 404호
(74) 대리인
홍성욱, 심경식

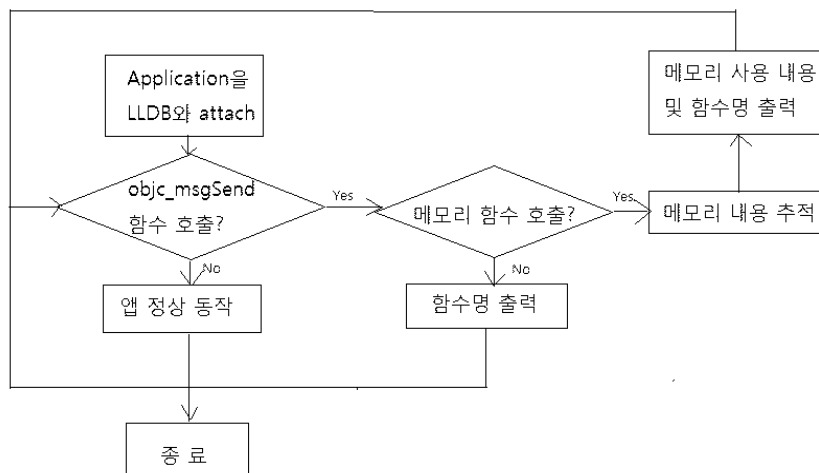
전체 청구항 수 : 총 1 항

(54) 발명의 명칭 LLDB 스크립트를 사용한 iOS 앱 행위 추적 방법

(57) 요약

본 발명은 LLDB 스크립트를 사용한 iOS 앱 행위 추적 방법으로, LLDB 디버거를 iOS 앱에 어태치(attach)시켜 앱이 동작하면서 발생하는 로그 정보를 LLDB 스크립트를 활용하여 앱 행위 분석에 필요한 정보를 추출해 보여주고, 메모리를 사용하는 함수를 추적하면 해당 함수가 사용한 메모리에 대한 주소와 쓰인 값을 확인하여 LLDB 스크립트를 사용하여 앱 행위를 추적할 수 있다.

대표도 - 도1



이 발명을 지원한 국가연구개발사업

과제고유번호	2015-0-00930
부처명	과학기술정보통신부
연구관리전문기관	정보통신기술진흥센터
연구사업명	SW전문인력양성사업/정보통신창의인재양성
연구과제명	2018 SW 중심대학 지원사업_충남대
기 여 율	1/1
주관기관	충남대학교 산학협력단
연구기간	2018.01.01 ~ 2018.12.31

명세서

청구범위

청구항 1

LDB 디버거를 iOS 앱에 어태치(attach)시켜 앱이 동작하면서 발생하는 로그 정보를 LLDB 스크립트를 활용하여 앱 행위 분석에 필요한 정보를 추출해 보여주고, 메모리를 사용하는 함수를 추적하면 해당 함수가 사용한 메모리에 대한 주소와 쓰인 값을 확인하여 LLDB 스크립트를 사용한 iOS 앱 행위 추적 방법.

발명의 설명

기술 분야

[0001] 본 발명은 LLDB 스크립트를 사용한 iOS 앱 행위 추적 방법에 관한 것이다.

배경 기술

[0002] 스마트폰이 대중화됨에 따라 다양한 개인 정보가 스마트폰에 저장되어 있다. 이러한 개인 정보를 탈취하려고 하는 악성 앱들이 증가하고 있으며, 피해사례가 발생하고 있다.

[0003] 스마트폰 OS는 안드로이드(Android)와 iOS가 99% 이상을 차지하고 있으며, 안드로이드는 오픈 소스 프로젝트와 여러 플랫폼에서 앱을 다운로드 받을 수 있는 정책의 특성상 악성 앱 피해사례의 대부분을 차지하지만 그 만큼 악성 앱 분석 기술도 발전되어 있다.

[0004] 반면, iOS는 클로즈 소스와 앱 스토어에서만 앱을 다운로드 받을 수 있는 정책으로 인하여 악성 앱에 대해 안전하다는 인식으로 인해 분석 기술이 상대적으로 적다.

[0005] 그러나, iOS에서도 악성 앱이 발견되고 있어 iOS의 앱을 분석하여 악성 행위를 파악하는 기술이 필요하다.

발명의 내용

해결하려는 과제

[0006] 본 발명은 LLDB 디버거를 iOS 앱에 어태치(attach)시켜 앱이 동작하면서 발생하는 로그 정보를 LLDB 스크립트를 활용하여 앱 행위 분석에 필요한 정보를 추출해 보여주고, 메모리를 사용하는 함수를 추적하면 해당 함수가 사용한 메모리에 대한 주소와 쓰인 값을 확인하여 LLDB 스크립트를 사용한 iOS 앱 행위 추적 방법을 제공하는데 그 목적이 있다.

과제의 해결 수단

[0007] 상기의 목적을 달성하기 위해 본 발명의 일 실시예에 따른 LDB 디버거를 iOS 앱에 어태치(attach)시켜 앱이 동작하면서 발생하는 로그 정보를 LLDB 스크립트를 활용하여 앱 행위 분석에 필요한 정보를 추출해 보여주고, 메모리를 사용하는 함수를 추적하면 해당 함수가 사용한 메모리에 대한 주소와 쓰인 값을 확인하여 LLDB 스크립트를 사용한 iOS 앱 행위 추적 방법을 포함한다.

발명의 효과

[0008] 본 발명에 의하면, iOS 앱에 대한 동작 행위를 유추할 수 있고, iOS 앱의 메모리를 추적함으로써 앱의 악성 행위 여부를 파악할 수 있다. 또한, 악성 행위 기능을 추출하여 예방할 수 있다는 효과가 있다.

[0009] 또한, iOS 앱에 대하여 LLDB 디버거 스크립트를 활용해 로그를 분석하는 방법은 여러 가지가 존재하지만 앱이 사용하는 메모리에 집중하여 기능을 분석하지 않기 때문에 메모리 분석을 통해 앱의 행위를 분석할 수 있다.

도면의 간단한 설명

[0010] 도 1은 본 발명의 일 실시예에 따른 LLDB 스크립트를 사용한 iOS 앱 행위 추적 방법의 흐름도이다.

발명을 실시하기 위한 구체적인 내용

- [0011] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면을 참조하여 상세하게 설명하도록 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다. 각 도면을 설명하면서 유사한 참조부호를 유사한 구성요소에 대해 사용하였다.
- [0012] 제1, 제2, A, B 등의 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 상기 구성요소들은 상기 용어들에 의해 한정되어서는 안된다. 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 본 발명의 권리범위를 벗어나지 않으면서 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소도 제1 구성요소로 명명될 수 있다. 및/또는 이라는 용어는 복수의 관련된 기재 항목들의 조합 또는 복수의 관련된 기재 항목들 중의 어느 항목을 포함한다.
- [0013] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급될 때에는 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급될 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다.
- [0014] 본 출원에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 출원에서, "포함하다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0015] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가지고 있다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥 상 가지는 의미와 일치하는 의미를 가지는 것으로 해석되어야 하며, 본 출원에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.
- [0016] 명세서 및 청구범위 전체에서, 어떤 부분이 어떤 구성 요소를 포함한다고 할때, 이는 특별히 반대되는 기재가 없는 한 다른 구성 요소를 제외하는 것이 아니라 다른 구성 요소를 더 포함할 수 있다는 것을 의미한다.
- [0018] 이하, 본 발명에 따른 바람직한 실시예를 첨부된 도면을 참조하여 상세하게 설명한다.
- [0019] 도 1은 본 발명의 일 실시예에 따른 LLDB 스크립트를 사용한 iOS 앱 행위 추적 방법의 흐름도이다.
- [0020] 도 1을 참조하면, 본 발명의 일 실시예에 따른 LLDB 스크립트를 사용한 iOS 앱 행위 추적 방법은 LLDB 디버거를 iOS 앱에 어태치시켜 앱이 동작하면서 발생시키는 로그 정보를 LLDB 스크립트를 활용하여 앱 행위 분석에 필요한 정보만 추출하여 보여줄 수 있다.
- [0021] 메모리를 사용하는 함수를 추적하면, 해당 함수가 사용한 메모리에 대한 주소와 쓰인 값을 알 수 있다. 악성 행위를 하기 위해서는 앱이 스마트폰의 저장된 데이터를 가져와서 사용해야 하므로 필수적으로 메모리가 사용된다. 따라서, 메모리에 어떠한 정보를 사용하는지 살펴 볼 수 있으면 사용된 함수와 연결지어 악성행위를 알 수 있다.
- [0022] 먼저, 본 발명의 일 실시예에 따른 LLDB 스크립트를 사용한 iOS 앱 행위 추적 방법 중 LLDB 스크립트 작동 절차를 설명하도록 한다. PC에서 LLDB를 동작시켜 아이폰(iPhone)에 어태치한다. 다음으로 PC에서 LLDB 스크립트를 업로드하고, 마지막으로 LLDB 스크립트에서 해당 앱을 분석한다.
- [0023] 다음으로, 메모리 사용 함수 확인 절차를 설명하도록 한다. 메모리 사용 함수 확인 절차는 iOS 앱이 다른 함수를 호출하기 위해서 objc_msgSend 함수를 사용하므로 사용되는 시점을 확인한다. 다음으로 메모리 관련 함수가 호출되는 시점을 확인하고, 메모리 관련 함수가 호출되고 종료되는 동안 메모리의 상태를 확인한다. 마지막으로 사용된 메모리의 주소와 값, 함수명을 출력한다.
- [0024] 이때, 메모리 관련 함수로는 init(), alloc(), free() 등이 포함된다.

[0025] 따라서, 본 발명에 의하면, iOS 앱에 대한 동작 행위를 유추할 수 있고, iOS 앱의 메모리를 추적함으로써 앱의 악성 행위 여부를 파악할 수 있다. 또한, 악성 행위 기능을 추출하여 예방할 수 있고, iOS 앱에 대하여 LLDB 디버거 스크립트를 활용해 로그를 분석하는 방법은 여러 가지가 존재하지만 앱이 사용하는 메모리에 집중하여 기능을 분석하지 않기 때문에 메모리 분석을 통해 앱의 행위를 분석할 수 있다.

[0027] 이상의 설명은 본 발명의 기술 사상을 예시적으로 설명한 것에 불과한 것으로, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 사람이라면 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 다양한 수정 및 변형이 가능할 것이다. 따라서, 본 발명에 개시된 실시예들은 본 발명의 기술 사상을 한정하기 위한 것이 아니라 설명하기 위한 것이고, 이러한 실시예에 의하여 본 발명의 기술 사상의 범위가 한정되는 것은 아니다. 본 발명의 보호 범위는 아래의 청구범위에 의하여 해석되어야 하며, 그와 동등한 범위 내에 있는 모든 기술 사상은 본 발명의 권리범위에 포함되는 것으로 해석되어야 할 것이다.

부호의 설명

도면

도면1

