



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2013-0027158
(43) 공개일자 2013년03월15일

(51) 국제특허분류(Int. Cl.)
G06F 21/22 (2006.01) G06F 9/44 (2006.01)
(21) 출원번호 10-2011-0090551
(22) 출원일자 2011년09월07일
심사청구일자 2011년10월25일

(71) 출원인
주식회사 팬택
서울특별시 마포구 성암로 179 (상암동, 팬택계열 알앤디센터빌딩)
(72) 발명자
황계선
서울특별시 마포구 상암동 DMC I-2팬택빌딩
선왕석
서울특별시 마포구 상암동 DMC I-2 팬택빌딩
(74) 대리인
특허법인무한

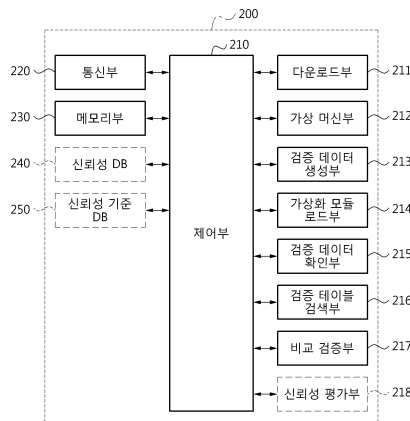
전체 청구항 수 : 총 25 항

(54) 발명의 명칭 임베디드 시스템에서 최적화된 가상화 모듈을 관리하는 장치 및 방법

(57) 요약

임베디드 시스템에서 최적화된 가상화 모듈을 관리하는 장치 및 방법을 제안한다. 실시 예에 따른 가상화 모듈을 관리하는 방법은 어플리케이션의 실행을 요청 받으면, 어플리케이션에 대응하고 장치에 최적화된 실행 과일인 가상화 모듈을 읽고, 가상화 모듈에 대응하는 검증 데이터를 검증 테이블에서 검색하고, 검증 테이블의 검증 데이터를 이용해서 가상화 모듈의 비정상적인 수정여부를 검증한다.

대표도 - 도2



특허청구의 범위

청구항 1

어플리케이션의 실행을 요청 받으면, 상기 어플리케이션에 대응하는 가상화 모듈인 ODEX 파일을 읽는 가상화 모듈 로드부;

상기 ODEX 파일에 포함된 검증 데이터를 확인하는 검증 데이터 확인부;

상기 ODEX 파일에 대응하는 검증 데이터를 검증 테이블에서 검색하는 검증 테이블 검색부; 및

상기 검증 테이블의 검증 데이터와 상기 ODEX 파일에 포함된 검증 데이터를 비교하고, 비교결과 상기 검증 테이블의 검증 데이터와 상기 ODEX 파일에 포함된 검증 데이터가 일치하면 상기 ODEX 파일이 비정상적으로 수정되지 않았다고 검증하는 비교 검증부를 포함하는

가상화 모듈을 관리하는 장치.

청구항 2

제1항에 있어서,

상기 비교 검증부는,

비교결과 상기 검증 테이블의 검증 데이터와 상기 ODEX 파일에 포함된 검증 데이터가 일치하지 않으면, 상기 ODEX 파일이 비정상적으로 수정되었다고 검증하는

가상화 모듈을 관리하는 장치.

청구항 3

제1항에 있어서,

상기 ODEX 파일을 검증할 수 있는 검증 데이터를 생성하는 검증 데이터 생성부를 더 포함하고,

상기 비교 검증부는,

상기 검증 데이터 생성부에서 생성한 검증 데이터, 상기 검증 테이블의 검증 데이터 및 상기 ODEX 파일에 포함된 검증 데이터를 모두 비교해서 모두 일치하면 상기 ODEX 파일이 비정상적으로 수정되지 않았다고 검증하고, 하나라도 일치하지 않으면 상기 ODEX 파일이 비정상적으로 수정되었다고 검증하는

가상화 모듈을 관리하는 장치.

청구항 4

제1항에 있어서,

상기 검증 데이터 확인부는,

상기 ODEX 파일에 검증 데이터가 포함되어 있지 않으면 상기 어플리케이션의 실행이 종료되도록 제어하고,

상기 검증 테이블 검색부는,

상기 ODEX 파일에 대응하는 상기 검증 데이터가 상기 검증 테이블에 존재하지 않으면, 상기 어플리케이션의 실행이 종료되도록 제어하는

가상화 모듈을 관리하는 장치.

청구항 5

제1항에 있어서,
상기 검증 데이터 확인부는,
상기 ODEX 파일의 헤더에서 상기 검증 데이터를 확인하는
가상화 모듈을 관리하는 장치.

청구항 6

제1항에 있어서,
상기 검증 데이터는,
상기 어플리케이션에 대응하는 상기 ODEX 파일의 변경 유무를 판단하기 위한 생성된 고유의 값인 검증정보를 포함하고,
추가로 상기 어플리케이션을 식별하는 식별정보를 더 포함할 수 있는
가상화 모듈을 관리하는 장치.

청구항 7

제6항에 있어서,
상기 검증정보는,
상기 ODEX 파일을 검사합(Checksum)해서 생성된 값이거나 또는 상기 ODEX 파일의 순환 중복 검사(CRC: Cyclic Redundancy Checks)해서 생성된 값인
가상화 모듈을 관리하는 장치.

청구항 8

제1항에 있어서,
상기 어플리케이션의 실행을 요청 받기 이전에, 상기 어플리케이션의 설치를 요청 받으면 상기 어플리케이션을 다운로드하는 다운로드부;
상기 어플리케이션에서 DEX 파일을 추출하고, 상기 DEX 파일을 최적화 해서 상기 ODEX 파일을 생성하는 가상 머신부; 및
상기 ODEX 파일을 검증할 수 있는 상기 검증 데이터를 생성하고, 상기 생성한 검증 데이터를 상기 ODEX 파일에 포함시키고, 상기 생성한 검증 데이터를 상기 검증 테이블에 저장하는 검증 데이터 생성부를 더 포함하는
가상화 모듈을 관리하는 장치.

청구항 9

제8항에 있어서,
상기 가상 머신부는,
상기 어플리케이션의 업데이트 요청을 수신하면 상기 다운로드부를 통해 업데이트 데이터를 수신하고 상기 업데이트 데이터를 설치하고,

상기 검증 데이터 생성부는,

상기 업데이트 데이터가 설치될 때 상기 ODEX 파일이 변경되면, 변경된 ODEX 파일을 검증할 수 있는 변경된 검증 데이터를 생성하고, 상기 변경된 ODEX 파일에 상기 변경된 검증 데이터를 포함시켜 저장하고, 상기 검증 테이블에 저장된 검증 데이터를 상기 변경된 검증 데이터로 갱신하는

가상화 모듈을 관리하는 장치.

청구항 10

어플리케이션의 실행을 요청 받으면, 상기 어플리케이션에 대응하는 가상화 모듈인 ODEX 파일을 읽는 가상화 모듈 로드부;

상기 ODEX 파일에 대응하는 검증 데이터를 검증 테이블에서 검색하는 검증 테이블 검색부;

상기 ODEX 파일을 검증할 수 있는 검증 데이터를 생성하는 검증 데이터 생성부; 및

생성한 검증 데이터와 상기 검증 테이블의 상기 검증 데이터를 비교해서 일치하면 상기 ODEX 파일이 비정상적으로 수정되지 않았다고 검증하고, 일치하지 않으면 상기 ODEX 파일이 비정상적으로 수정되었다고 검증하는 비교 검증부를 포함하는

가상화 모듈을 관리하는 장치.

청구항 11

제10항에 있어서,

상기 검증 테이블 검색부는,

상기 ODEX 파일에 대응하는 상기 검증 데이터가 존재하지 않으면, 상기 어플리케이션의 실행이 종료되도록 제어하는

가상화 모듈을 관리하는 장치.

청구항 12

제10항에 있어서,

상기 검증 데이터는,

상기 어플리케이션에 대응하는 상기 ODEX 파일의 변경 유무를 판단하기 위한 생성된 고유의 값인 검증정보를 포함하고,

추가로 상기 어플리케이션을 식별하는 식별정보를 더 포함할 수 있는

가상화 모듈을 관리하는 장치.

청구항 13

제12항에 있어서,

상기 검증정보는,

상기 ODEX 파일을 검사합(Checksum)해서 생성된 값이거나 또는 상기 ODEX 파일의 순환 중복 검사(CRC: Cyclic Redundancy Checks)해서 생성된 값인

가상화 모듈을 관리하는 장치.

청구항 14

제10항에 있어서,

상기 어플리케이션의 실행을 요청 받기 이전에, 상기 어플리케이션의 설치를 요청 받으면 상기 어플리케이션을 다운로드하는 다운로드부; 및

상기 어플리케이션에서 DEX 파일을 추출하고, 상기 DEX 파일을 최적화 해서 상기 ODEX 파일을 생성하는 가상 머신부를 더 포함하고,

상기 검증 데이터 생성부는,

상기 ODEX 파일을 검증할 수 있는 상기 검증 데이터를 생성하고, 상기 생성한 검증 데이터를 상기 검증 테이블에 저장하는

가상화 모듈을 관리하는 장치.

청구항 15

제14항에 있어서,

상기 가상 머신부는,

상기 어플리케이션의 업데이트 요청을 수신하면 상기 다운로드부를 통해 업데이트 데이터를 수신하고 상기 업데이트 데이터를 설치하고,

상기 검증 데이터 생성부는,

상기 업데이트 데이터가 설치될 때 상기 ODEX 파일이 변경되면, 변경된 ODEX 파일을 검증할 수 있는 변경된 검증 데이터를 생성하고, 상기 검증 테이블에 저장된 검증 데이터를 상기 변경된 검증 데이터로 갱신하는

가상화 모듈을 관리하는 장치.

청구항 16

어플리케이션의 실행을 요청 받으면, 상기 어플리케이션에 대응하는 가상화 모듈인 ODEX 파일을 읽는 단계;

상기 ODEX 파일에 포함된 검증 데이터를 확인하는 단계;

상기 ODEX 파일에 대응하는 검증 데이터를 검증 테이블에서 검색하는 단계;

상기 검증 테이블의 검증 데이터와 상기 ODEX 파일에 포함된 검증 데이터를 비교하는 단계; 및

비교결과 상기 검증 테이블의 검증 데이터와 상기 ODEX 파일에 포함된 검증 데이터가 일치하면, 상기 어플리케이션에 대응하는 가상 머신 객체를 생성해서 상기 ODEX 파일을 실행하는 단계를 포함하는

가상화 모듈을 관리하는 방법.

청구항 17

어플리케이션의 실행을 요청 받으면, 상기 어플리케이션에 대응하는 가상화 모듈인 ODEX 파일을 읽는 단계;

상기 ODEX 파일에 대응하는 검증 데이터를 검증 테이블에서 검색하는 단계;

상기 ODEX 파일을 이용해서 검증 데이터를 생성하는 단계;

생성한 검증 데이터와 상기 검증 테이블의 상기 검증 데이터를 비교하는 단계;

비교결과 일치하면 상기 ODEX 파일이 비정상적으로 수정되지 않았다고 검증하고, 비교결과 일치하지 않으면 상기 ODEX 파일이 비정상적으로 수정되었다고 검증하는 단계; 및

검증결과 상기 ODEX 파일이 비정상적으로 수정되지 않았으면 상기 어플리케이션에 대응하는 가상 머신 객체를 생성해서 상기 ODEX 파일을 실행하는 단계를 포함하는

가상화 모듈을 관리하는 방법.

청구항 18

어플리케이션의 실행을 요청 받으면, 상기 어플리케이션에 대응하고 장치에 최적화된 실행 파일인 가상화 모듈을 읽는 가상화 모듈 로드부;

상기 가상화 모듈에 대응하는 검증 데이터를 검증 테이블에서 검색하는 검증 테이블 검색부; 및

상기 검증 테이블의 상기 검증 데이터를 이용해서 상기 가상화 모듈의 비정상적인 수정여부를 검증하는 비교 검증부를 포함하는

가상화 모듈을 관리하는 장치.

청구항 19

제18항에 있어서,

상기 가상화 모듈을 검증할 수 있는 검증 데이터를 생성하는 검증 데이터 생성부를 더 포함하고,

상기 비교 검증부는,

생성한 검증 데이터와 상기 검증 테이블의 상기 검증 데이터를 비교해서 일치하면 상기 가상화 모듈이 비정상적으로 수정되지 않았다고 검증하고, 일치하지 않으면 상기 가상화 모듈이 비정상적으로 수정되었다고 검증하는

가상화 모듈을 관리하는 장치.

청구항 20

제18항에 있어서,

상기 가상화 모듈에 포함된 검증 데이터를 확인하는 검증 데이터 확인부;

상기 비교 검증부는,

상기 검증 테이블의 검증 데이터와 상기 가상화 모듈에 포함된 검증 데이터를 비교해서 일치하면 상기 가상화 모듈이 비정상적으로 수정되지 않았다고 검증하고, 일치하지 않으면 상기 가상화 모듈이 비정상적으로 수정되었다고 검증하는

가상화 모듈을 관리하는 장치.

청구항 21

제18항에 있어서,

상기 가상화 모듈을 검증할 수 있는 검증 데이터를 생성하는 검증 데이터 생성부; 및

상기 가상화 모듈에 포함된 검증 데이터를 확인하는 검증 데이터 확인부를 더 포함하고,

상기 비교 검증부는,

상기 검증 데이터 생성부에서 생성한 검증 데이터, 상기 검증 테이블의 검증 데이터 및 상기 가상화 모듈에 포

함된 검증 데이터를 모두 비교해서 모두 일치하면 상기 가상화 모듈이 비정상적으로 수정되지 않았다고 검증하고, 하나라도 일치하지 않으면 상기 가상화 모듈이 비정상적으로 수정되었다고 검증하는 가상화 모듈을 관리하는 장치.

청구항 22

제18항에 있어서,
상기 어플리케이션의 실행을 요청 받기 이전에, 상기 어플리케이션의 설치를 요청 받으면 상기 어플리케이션을 다운로드하는 다운로드부;
다운로드한 상기 어플리케이션을 이용해서 장치에 최적화된 상기 가상화 모듈을 생성하는 가상 머신부; 및
상기 가상화 모듈을 검증할 수 있는 상기 검증 데이터를 생성하고, 상기 검증 데이터를 상기 검증 테이블에 저장하는 검증 데이터 생성부를 더 포함하는
가상화 모듈을 관리하는 장치.

청구항 23

제22항에 있어서,
상기 검증 데이터 생성부는,
상기 생성한 검증 데이터를 상기 가상화 모듈에 포함시키는
가상화 모듈을 관리하는 장치.

청구항 24

제22항에 있어서,
상기 가상 머신부는,
상기 어플리케이션의 업데이트 요청을 수신하면 상기 다운로드부를 통해 업데이트 데이터를 수신하고 상기 업데이트 데이터를 설치하고,
상기 검증 데이터 생성부는,
상기 업데이트 데이터가 설치될 때 상기 가상화 모듈이 변경되면, 변경된 가상화 모듈을 검증할 수 있는 변경된 검증 데이터를 생성하고, 상기 검증 테이블에 저장된 검증 데이터를 상기 변경된 검증 데이터로 갱신하는
가상화 모듈을 관리하는 장치.

청구항 25

제24항에 있어서,
상기 검증 데이터 생성부는,
상기 변경된 가상화 모듈에 상기 변경된 검증 데이터를 포함시켜 저장하고,
가상화 모듈을 관리하는 장치.

명세서

기술분야

[0001] 본 발명은 임베디드 시스템에서 사용되는 가상 머신(VM: Virtual Machine)에서 어플리케이션을 위해 이용되는 장치 별로 최적화된 가상화 모듈에 관한 것으로서, 가상화 모듈에 대한 임의의 수정 여부를 검증하여 관리하는 장치 및 방법에 관한 것이다.

배경 기술

[0002] 가상 머신은 실제 하드웨어와 직접적인 통신이 없는 가상 컴퓨터를 나타낸다. 현재 모바일 장치에서 사용되고 있는 가상 머신으로는 자바 가상 머신을 비롯해 Android의 Dalvik, Apple IOS에서 사용되는 LLVM 등이 있다. 가상 머신은 아무 플랫폼에서나 같은 방식으로 실행하는 프로그램을 허용하고 기초가 되는 하드웨어나 운영체제의 상세한 부분을 가져오는 독립 프로그래밍 환경을 제공하기 위함을 목적으로 한다. 가상 머신은 하드웨어나 운영체제의 제약으로부터 벗어나기 위하여 바이트 코드로 컴파일하고 실제로 어플리케이션 동작 시 중간 코드인 바이트 코드를 해석해서 어플리케이션을 실행한다. 안드로이드 계열의 스마트 폰에서 어플리케이션은 APK 파일의 형태이지만 최초 스마트 폰에 설치된 이후에는 APK 안의 DEX 를 최적화한 ODEX(Optimizing DEX)로 수행이 된다. 여기서, ODEX는 DEX를 하드웨어 사양을 고려해서 최적화 시킨 가상화 모듈로써 바이트 코드에 일종이다. 하지만 스마트 폰의 백신 검사 영역은 APK 파일에 한정되고 실제 수행이 되는 ODEX가 임의의 사용자에게 의해 변경이 될 가능성이 발생한다.

발명의 내용

해결하려는 과제

[0003] 본 발명의 실시예는 어플리케이션을 다운로드하고, 어플리케이션을 이용해서 장치에 최적화된 상기 가상화 모듈을 생성하고, 가상화 모듈을 검증할 수 있는 검증 데이터를 생성해서 관리하는 가상화 모듈을 관리하는 장치 및 방법을 제공한다.

[0004] 본 발명의 실시예는 어플리케이션의 업데이트 요청을 수신하면 업데이트 데이터를 수신해서 설치하고, 업데이트 데이터가 설치될 때 상기 가상화 모듈이 변경되면 변경된 가상화 모듈을 검증할 수 있는 변경된 검증 데이터를 생성하고, 기존의 검증 데이터를 변경된 검증 데이터로 갱신하는 가상화 모듈을 관리하는 장치 및 방법을 제공한다.

[0005] 본 발명의 실시예는 어플리케이션의 실행을 요청 받으면, 어플리케이션에 대응하고 장치에 최적화된 실행 파일인 가상화 모듈을 읽고, 가상화 모듈에 대응하는 검증 데이터를 검증 테이블에서 검색하고, 검증 테이블의 검증 데이터를 이용해서 가상화 모듈의 비정상적인 수정여부를 검증하는 가상화 모듈을 관리하는 장치 및 방법을 제공한다.

과제의 해결 수단

[0006] 본 발명의 실시예에 따른 가상화 모듈을 관리하는 장치는, 어플리케이션의 실행을 요청 받으면, 상기 어플리케이션에 대응하는 가상화 모듈인 ODEX 파일을 읽는 가상화 모듈 로드부와, 상기 ODEX 파일에 포함된 검증 데이터를 확인하는 검증 데이터 확인부와, 상기 ODEX 파일에 대응하는 검증 데이터를 검증 테이블에서 검색하는 검증 테이블 검색부 및 상기 검증 테이블의 검증 데이터와 상기 ODEX 파일에 포함된 검증 데이터를 비교하고, 비교결과 상기 검증 테이블의 검증 데이터와 상기 ODEX 파일에 포함된 검증 데이터가 일치하면 상기 ODEX 파일이 비정상적으로 수정되지 않았다고 검증하는 비교 검증부를 포함한다.

[0007] 본 발명의 실시예에 따른 가상화 모듈을 관리하는 장치는, 어플리케이션의 실행을 요청 받으면, 상기 어플리케이션에 대응하는 가상화 모듈인 ODEX 파일을 읽는 가상화 모듈 로드부와, 상기 ODEX 파일에 대응하는 검증 데이터를 검증 테이블에서 검색하는 검증 테이블 검색부와, 상기 ODEX 파일을 검증할 수 있는 검증 데이터를 생성하는 검증 데이터 생성부 및 생성한 검증 데이터와 상기 검증 테이블의 상기 검증 데이터를 비교해서 일치하면 상기 ODEX 파일이 비정상적으로 수정되지 않았다고 검증하고, 일치하지 않으면 상기 ODEX 파일이 비정상적으로 수정되었다고 검증하는 비교 검증부를 포함한다.

[0008] 본 발명의 실시예에 따른 가상화 모듈을 관리하는 방법은, 어플리케이션의 실행을 요청 받으면, 상기 어플리케이션에 대응하는 가상화 모듈인 ODEX 파일을 읽는 단계와, 상기 ODEX 파일에 포함된 검증 데이터를 확인하는 단

계와, 상기 ODEX 파일에 대응하는 검증 데이터를 검증 테이블에서 검색하는 단계와, 상기 검증 테이블의 검증 데이터와 상기 ODEX 파일에 포함된 검증 데이터를 비교하는 단계 및 비교결과 상기 검증 테이블의 검증 데이터와 상기 ODEX 파일에 포함된 검증 데이터가 일치하면, 상기 어플리케이션에 대응하는 가상 머신 객체를 생성해서 상기 ODEX 파일을 실행하는 단계를 포함한다.

[0009] 본 발명의 실시예에 따른 가상화 모듈을 관리하는 방법은, 어플리케이션의 실행을 요청 받으면, 상기 어플리케이션에 대응하는 가상화 모듈인 ODEX 파일을 읽는 단계와, 상기 ODEX 파일에 대응하는 검증 데이터를 검증 테이블에서 검색하는 단계와, 상기 ODEX 파일을 이용해서 검증 데이터를 생성하는 단계와, 생성한 검증 데이터와 상기 검증 테이블의 상기 검증 데이터를 비교하는 단계와, 비교결과 일치하면 상기 ODEX 파일이 비정상적으로 수정되지 않았다고 검증하고, 비교결과 일치하지 않으면 상기 ODEX 파일이 비정상적으로 수정되었다고 검증하는 단계 및 검증결과 상기 ODEX 파일이 비정상적으로 수정되지 않았으면 상기 어플리케이션에 대응하는 가상 머신 객체를 생성해서 상기 ODEX 파일을 실행하는 단계를 포함한다.

[0010] 본 발명의 실시예에 따른 가상화 모듈을 관리하는 장치는, 어플리케이션의 실행을 요청 받으면, 상기 어플리케이션에 대응하고 장치에 최적화된 실행 파일인 가상화 모듈을 읽는 가상화 모듈 로드부와, 상기 가상화 모듈에 대응하는 검증 데이터를 검증 테이블에서 검색하는 검증 테이블 검색부 및 상기 검증 테이블의 상기 검증 데이터를 이용해서 상기 가상화 모듈의 비정상적인 수정여부를 검증하는 비교 검증부를 포함한다.

발명의 효과

[0011] 본 발명은 임베디드 시스템에서 최적화된 가상화 모듈을 관리하는 장치 및 방법에 관한 것으로, 본 발명의 가상화 모듈을 관리하는 장치는 비정상적인 수정이 발생했는지 여부를 확인할 수 있는 검증 데이터를 가상화 모듈 별로 생성하여 관리함으로써, 임의의 사용자에 의해 가상화 모듈이 변경되면 실행 전에 확인해서 실행차단 할 수 있다.

도면의 간단한 설명

[0012] 도 1은 본 발명의 실시예에 따른 안드로이드 플랫폼의 구조를 나타낸다.
 도 1을 참조하면, 안드로이드 플랫폼은
 도 2는 본 발명의 일 실시예에 따른 가상화 모듈을 관리하는 단말 장치의 구조를 나타낸다.
 도 3은 본 발명의 다른 실시예에 따른 가상화 모듈을 관리하는 단말 장치의 구조를 나타낸다.
 도 4는 본 발명의 실시예에 따라 가상화 모듈을 검증하기 위한 검증 정보를 생성할 때 사용할 수 있는 검증합을 생성하는 예를 나타낸다.
 도 5는 본 발명의 실시예에 따른 검증 테이블의 예를 나타낸다.
 도 6은 본 발명의 실시예에 따라 도 2의 단말 장치에서 어플리케이션을 설정하는 과정을 도시한 흐름도이다.
 도 7은 본 발명의 실시예에 따라 도 2의 단말 장치에서 어플리케이션을 업데이트하는 과정을 도시한 흐름도이다.
 도 8은 본 발명의 실시예에 따라 도 2의 단말 장치에서 어플리케이션을 실행하는 과정을 도시한 흐름도이다.
 도 9는 본 발명의 실시예에 따라 도 3의 단말 장치에서 어플리케이션을 설정하는 과정을 도시한 흐름도이다.
 도 10은 본 발명의 실시예에 따라 도 3의 단말 장치에서 어플리케이션을 업데이트하는 과정을 도시한 흐름도이다.
 도 11은 본 발명의 실시예에 따라 도 3의 단말 장치에서 어플리케이션을 실행하는 과정을 도시한 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0013] 이하, 본 발명의 실시예를 첨부된 도면을 참조하여 상세하게 설명한다.

- [0014] 본 발명은 임베디드 시스템에서 가상화 모듈을 관리하는 장치 및 방법에 관한 것이나, 설명의 편의를 위해 이하의 설명에서는 안드로이드 플랫폼을 탑재한 임베디드 시스템의 경우를 예로 설명한다. 여기서 가상화 모듈이란 어플리케이션을 해당 시스템에 최적화한 실행 파일을 나타낸다.
- [0015] 도 1은 본 발명의 실시예에 따른 안드로이드 플랫폼의 구조를 나타낸다.
- [0016] 도 1을 참조하면, 안드로이드 플랫폼은 어플리케이션(110), 어플리케이션 프레임워크(120), 라이브러리(130), 안드로이드 런타임(140), 리눅스 커널(150)으로 구분된다.
- [0017] 리눅스 커널(150)은 메모리, 네트워크, 보안, 드라이버와 관련된 코어 시스템 서비스를 관리한다. 라이브러리(130)는 어플리케이션(110)과 어플리케이션 프레임워크(120)에서 사용하는 다양한 컴포넌트들을 제공한다. 이때, 라이브러리(130)에서 제공하는 컴포넌트로는 Surface Manager, Media Framework, SQLite, OpenGL ES, FreeType, Webkit, SGL, SSL, libc 등이 있다.
- [0018] 어플리케이션 프레임워크(120)는 어플리케이션 구성 및 동작을 위한 컴포넌트들을 제공한다. 이때, 어플리케이션 프레임워크(120)에서 제공하는 컴포넌트로는 Activity Manager, Window Manager, Contents Providers, View System, Notification Manager, Package Manager, Telephony Manager, Resource Manager, Location Manager, XMPP Service 등이 있다.
- [0019] 어플리케이션(100)은 Home, Contacts, Phone 등의 일반적인 어플리케이션을 제공한다.
- [0020] 안드로이드 런타임(140)은 자바 프로그래밍의 기능을 사용해서 어플리케이션 동작을 처리한다. 안드로이드 런타임(140)은 가상 머신부(142)와 가상화 모듈 관리부(144)를 포함할 수 있다.
- [0021] 가상화 모듈 관리부(144)는 어플리케이션에 대응하는 검증 데이터를 생성해서 관리하고, 검증 데이터를 이용해서 어플리케이션의 가상화 모듈이 임의의 사용자에게 의해 변경되었는지 검증한다. 가상화 모듈 관리부(144)는 어플리케이션이 실행될 때 어플리케이션 별로 가상화 모듈 관리 객체를 생성해서 가상화 모듈을 검증할 수도 있다.
- [0022] 가상 머신부(142)는 어플리케이션의 설치와 실행시 하드웨어가 인식할 수 있도록 컴파일과 해석을 담당하기 위한 가상 머신이다. 가상 머신부(142)는 APK 파일에 포함된 DEX 파일을 최적화해서 ODEX 파일을 생성할 수 있고, DEX 파일과 ODEX 파일에 들어있는 정보를 해석할 수 있다. 또한, 가상 머신부(142)는 어플리케이션의 실행 이벤트 발생시 어플리케이션 별로 가상 머신 객체를 생성해서 어플리케이션에 대응하는 ODEX 파일을 실행할 수 있다.
- [0023] 가상 머신부(142)과 가상화 모듈 관리부(144)에 대한 상세한 설명은 이도 도 2와 도 3을 통해 상세히 후술한다.
- [0024] 도 2는 본 발명의 일 실시예에 따른 가상화 모듈을 관리하는 단말 장치의 구조를 나타낸다.
- [0025] 도 2를 참조하면, 단말 장치(200)는 제어부(210), 다운로드부(211), 가상 머신부(212), 검증 데이터 생성부(213), 가상화 모듈 로드부(214), 검증 데이터 확인부(215), 검증 테이블 검색부(216), 비교 검증부(217), 통신부(220) 및 메모리부(230)를 포함한다. 또한, 단말 장치(200)는 추가로 신뢰성 평가부(218), 신뢰성 데이터베이스(240) 및 신뢰성 기준 데이터베이스(250) 중 하나 내지는 모두를 더 포함할 수도 있다.
- [0026] 통신부(220)는 유/무선으로 데이터를 송수신하는 장치로서 어플리케이션, 어플리케이션의 업데이트 데이터, 어플리케이션들의 신뢰성에 관한 정보 및 어플리케이션들의 신뢰성을 판단할 수 있는 신뢰성 기준 정보 중에서 적어도 하나를 수신할 수 있다.
- [0027] 통신부(220)는 FDMA(Frequency Division Multiple Access), TDMA(Time Division Multiple Access), SDMA(Space-Division Multiple Access), CDMA(Code Division Multiple Access), WCDMA(Wideband Code Division Multiple Access), OFDM, Wif, 와 이브로(Wipro), 블루투스, 적외선 통신 등을 기반으로 하는 무선 통신 기법을 통해 무선으로 데이터를 송수신할 수 있다. 무선으로 통신하는 통신부(220)는 안테나를 통해 입출력 되는 데이터의 무선신호를 송수신 처리하는 기능을 수행할 수 있다. 예를 들어, 송신인 경우, 송신할 데이터를 채널 코딩(Channel coding) 및 확산(Spreading)한 후, RF처리하여 송신하는 기능을 수행하고, 수신인 경우, 수신된 RF신호를 기저대역신호로 변환하고 기저대역신호를 역 확산(De-spreading) 및 채널 복호(Channel decoding)하여 데이터를 복원하는 기능을 수행한다.

- [0028] 메모리부(230)는 단말 장치(200)의 전반적인 동작을 제어하기 위한 운영체제에 속하는 시스템 데이터, 어플리케이션, 어플리케이션의 변경이력 정보 및 사용자 데이터(전화번호, SMS 메시지, 압축된 이미지 파일, 동영상 등) 등을 저장한다. 그리고, 메모리부(230)는 APK 파일, DEX 파일, ODEX 파일 및 검증 테이블을 저장한다. 여기서, APK 파일은 어플리케이션을 설치하기 위한 설치파일을 나타낸다. DEX 파일은 APK 파일에 포함된 정보로서 어플리케이션을 실행하기 위한 일종의 바이트 코드이다. ODEX 파일은 DEX 파일을 해당 장치에 최적화 해서 생성된 가상화 모듈을 나타낸다. 그리고, 검증 테이블은 ODEX 파일들 각각의 변경여부를 검증할 수 있는 검증 데이터들을 저장한 테이블 이다. 검증 테이블은 아래 도 5와 같이 구성될 수 있다.
- [0029] 도 5는 본 발명의 실시예에 따른 검증 테이블의 예를 나타낸다.
- [0030] 도 5를 참조하면, 검증 테이블은 식별정보, 이름 및 검증정보를 포함할 수 있다. 여기서, 식별정보를 어플리케이션을 식별할 수 있는 정보로써 어플리케이션의 설치 순서에 따라 순차적으로 부여될 수 있다. 이름은 DEX 파일을 ODEX 파일로 변환하는 과정에서 부여되는 정보로써 단말 장치(200) 내에 실제 APK 파일이 저장된 경로를 나타낸다. 검증정보는 ODEX 파일의 변경 유무를 확인할 수 있는 정보이다. 검증정보에 관한 보다 상세한 설명은 검증 데이터 생성부(213)를 통해 후술한다.
- [0031] 신뢰성 데이터베이스(240)는 어플리케이션들 각각에 대한 신뢰성에 관한 정보들을 저장한다. 이때, 신뢰성 데이터베이스(240)에 저장된 어플리케이션들 각각에 대한 신뢰성에 관한 정보들은 신뢰성을 평가하는 별도의 서버로부터 제공받을 수 있다.
- [0032] 신뢰성 기준 데이터베이스(250)는 어플리케이션의 신뢰도를 평가하는 기준이 되는 정보인 기준정보를 저장한다. 이때, 신뢰성 기준 데이터베이스(250)에 저장된 기준정보는 기준정보를 생성하는 별도의 서버로부터 제공받을 수 있다. 여기서, 기준정보는 신뢰성이 있는 카테고리, 신뢰성이 있는 생산자, 신뢰성이 있는 판매 서버, 신뢰성이 있는 기준 출시일자 및 신뢰성이 있는 권한 중에서 적어도 하나 또는 하나 이상의 조합으로 구성된 정보일 수 있다.
- [0033] 기준정보는 특정 생산자가 생성한 어플리케이션, 특정 판매 서버를 통해 판매되는 특정 생산자의 특정 출시일자 이전에 생성된 어플리케이션, 특정 판매 서버를 통해 판매되는 특정 카테고리의 어플리케이션, 특정 판매 서버를 통해 판매되는 어플리케이션들 중에서 특정 권한만을 가진 어플리케이션 등이 될 수 있다.
- [0034] 예를 들어, 기준정보가 판매 서버 P인 경우, 판매 서버 P를 통해 판매되는 어플리케이션들은 신뢰할 수 있다. 다른 예로 기준정보가 생산자 A인 경우, 생산자 A가 생성한 어플리케이션은 신뢰할 수 있다. 또 다른 예로 기준정보가 판매 서버 P와 생산자 A의 조합인 경우, 판매 서버 P를 통해 판매되는 생산자 A가 생성하는 어플리케이션은 신뢰할 수 있다.
- [0035] 기준정보를 생성하는 별도의 서버에서 기준정보를 생성하는 예를 살펴보면, 기준 정보를 생성하는 서버는 판매 서버 P를 통해 판매되는 생산자 A가 이전에 생성한 어플리케이션의 수가 50개(기설정 기준 개수)를 초과하고, 판매 서버 P를 통해 판매되는 생산자 A가 이전에 생성한 모든 어플리케이션에 바이러스가 존재하지 않은 경우, 판매 서버 P를 통해 판매되는 생산자 A가 생성하는 어플리케이션이 기준정보로 생성 될 수 있다. 이 경우, 추후 설치하고자 하는 어플리케이션이 판매 서버 P를 통해 판매되는 생산자 A가 생성하는 어플리케이션에 부합되면 설치하고자 하는 어플리케이션은 신뢰성이 있다고 판단될 수 있다.
- [0036] 다운로드부(211)는 제어부(210)의 제어에 따라 판매 서버로부터 어플리케이션 또는 업데이트 데이터를 다운로드 한다.
- [0037] 가상 머신부(212)는 다운로드부(211)를 통해 다운로드된 어플리케이션인 APK 파일에서 DEX 파일을 추출하고, DEX 파일을 최적화 해서 ODEX 파일을 생성하고, 생성한 ODEX 파일을 메모리부(230)에 저장한다.
- [0038] 가상 머신부(212)는 다운로드부(211)를 통해 다운로드된 업데이트 데이터를 설치하고, 업데이트 데이터 설치시 대응하는 ODEX 파일이 변경되면, ODEX 파일의 변경되었음을 검증 데이터 생성부(213)로 제공한다.
- [0039] 가상 머신부(212)는 어플리케이션의 실행을 요청 받으면, 비교 검증부(217)를 통해 어플리케이션에 대응하는 ODEX 파일에 의도되지 않은 수정이 발생했는지 확인하고, ODEX 파일이 수정되지 않았으면 가상 머신 객체를 생성해서 ODEX 파일을 실행한다.
- [0040] 도 2에서 검증 데이터 생성부(213), 가상화 모듈 로드부(214), 검증 데이터 확인부(215), 검증 테이블 검색부(216), 비교 검증부(217)는 도 1의 가상화 모듈 관리부(144)에 포함되는 구성이다.

- [0041] 검증 데이터 생성부(213)는 ODEX 파일을 검증할 수 있는 검증 데이터를 생성하고, 생성한 검증 데이터를 메모리부(230)의 검증 테이블에 저장하고, 생성한 검증 데이터를 ODEX 파일에 포함시킨다. 이때, 검증 데이터 생성부(213)는 생성한 검증 데이터를 ODEX 파일의 헤더에 삽입할 수 있다.
- [0042] 검증 데이터 생성부(213)는 어플리케이션의 실행을 요청 받으면, 가상화 모듈 로드부(214)를 통해 로드한 어플리케이션에 대응하는 ODEX 파일을 이용해서 검증 데이터를 생성하고, 생성한 검증 데이터를 비교 검증부(217)로 제공할 수도 있다.
- [0043] 검증 데이터 생성부(213)는 업데이트 데이터가 설치될 때 ODEX 파일이 변경되면, 변경된 ODEX 파일을 검증할 수 있는 변경된 검증 데이터를 생성하고, 검증 테이블에 저장된 검증 데이터를 변경된 검증 데이터로 갱신한다. 또한, 검증 데이터 생성부(213)는 변경된 ODEX 파일에 변경된 검증 데이터를 포함시켜 저장한다.
- [0044] 검증 데이터 생성부(213)에서 생성하는 검증 데이터는 ODEX 파일의 변경 유무를 판단하기 위한 생성된 고유의 값인 검증정보를 포함한다. 그리고, 검증 데이터는 추가로 어플리케이션을 식별하는 식별정보를 더 포함할 수 있다.
- [0045] 검증 데이터 생성부(213)는 검증정보를 ODEX 파일을 검사합(Checksum)해서 생성된 값을 검증정보를 사용할 수도 있고, ODEX 파일의 순환 중복 검사(CRC: Cyclic Redundancy Checks)해서 생성된 값을 검증정보로 사용할 수 있다. 이 외에도 데이터의 변경 유무를 확인할 수 있는 값이면 모두 검증정보로 사용될 수 있다.
- [0046] 검증 데이터 생성부(213)는 검증정보를 생성할 때, 기설정된 암호값을 추가해서 임의의 사용자가 검증정보를 유추할 수 없도록 할 수도 있다.
- [0047] 검증 정보를 생성하는 데 사용될 수 있는 검사합(Checksum)은 데이터를 이진수(binary number)의 연속으로 간주하여 그 이진수의 합을 계산하는 방법이다.
- [0048] 도 4는 본 발명의 실시예에 따라 가상화 모듈을 검증하기 위한 검증 정보를 생성할 때 사용할 수 있는 검증합을 생성하는 예를 나타낸다.
- [0049] 도 4는 참조하면, 검증 데이터 생성부(213)는 16 bits 검사합(checksum)을 사용해서 검증정보를 생성하는 경우, 데이터를 16 bits씩 나누어 16진수로 표현한 다음 그 합을 계산하여 16으로 나눈 나머지에 해당하는 값을 검증정보로 한다.
- [0050] 이때, 도 4에서 데이터를 16 bits씩 나누어 16진수로 표현한 다음 합할 때, 추가로 더한 carry는 임의의 사용자가 검증정보를 유추할 수 없도록 하는 기설정된 암호값이다.
- [0051] 가상화 모듈 로드부(214)는 어플리케이션의 실행을 요청 받으면, 어플리케이션에 대응하는 가상화 모듈인 ODEX 파일을 메모리부(230)에서 로드(load)한다.
- [0052] 검증 데이터 확인부(215)는 가상화 모듈 로드부(214)에서 로드한 ODEX 파일에 포함된 검증 데이터를 확인한다. 이때, 검증 데이터 확인부(215)는 ODEX 파일에 검증 데이터가 포함되어 있지 않으면 어플리케이션의 실행이 종료되도록 제어할 수도 있다.
- [0053] 검증 테이블 검색부(216)는 ODEX 파일에 대응하는 검증 데이터를 메모리부(230)에 저장된 검증 테이블에서 검색한다. 이때, 검증 테이블 검색부(216)는 ODEX 파일에 대응하는 검증 데이터가 검증 테이블에 존재하지 않으면, 어플리케이션의 실행이 종료되도록 제어할 수도 있다.
- [0054] 비교 검증부(217)는 검증 테이블 검색부(216)에서 검색한 검증 데이터와 검증 데이터 확인부(215)에서 확인한 검증 데이터를 비교하고, 비교결과 일치하면 ODEX 파일이 비정상적으로 수정되지 않았다고 검증하고, 일치하지 않으면 ODEX 파일이 비정상적으로 수정되었다고 검증한다.
- [0055] 다른 방법으로 비교 검증부(217)는 검증 테이블 검색부(216)에서 검색한 검증 데이터, 검증 데이터 확인부(215)에서 확인한 검증 데이터 및 검증 데이터 생성부(213)에서 생성한 검증 데이터를 모두 비교해서 모두 일치하면 ODEX 파일이 비정상적으로 수정되지 않았다고 검증하고, 하나라도 일치하지 않으면 ODEX 파일이 비정상적으로 수정되었다고 검증한다.
- [0056] 신뢰성 평가부(218)는 사용자가 설치하고자 하는 어플리케이션과 업데이트 데이터의 신뢰성을 평가하고, 평가결과 신뢰성이 낮으면 어플리케이션 또는 업데이트 데이터의 설치를 중단할 수 있다.
- [0057] 신뢰성 평가부(218)는 신뢰성 평가를 어플리케이션 또는 업데이트 데이터를 수신하기 이전에 평가할 수 있고,

어플리케이션 또는 업데이트 데이터를 수신하고 설치하기 이전에 평가할 수도 있다.

- [0058] 신뢰성 평가부(218)는 여러가지 방법을 통해 신뢰성을 평가할 수 있다. 신뢰성 평가부(218)가 신뢰성을 평가하는 예는 다음과 같다.
- [0059] 신뢰성 평가부(218)는 신뢰도를 평가하는 별도의 서버로 어플리케이션에 대한 신뢰성 평가를 요청하고, 신뢰도를 평가하는 별도의 서버로부터 어플리케이션에 대한 신뢰성 평가결과를 수신해서 어플리케이션의 신뢰성을 평가할 수 있다.
- [0060] 신뢰성 평가부(218)는 신뢰성 데이터베이스(240)에서 어플리케이션을 검색해서 어플리케이션의 신뢰성을 평가할 수도 있다.
- [0061] 신뢰성 평가부(218)는 어플리케이션을 공급하는 서버로부터 어플리케이션에 관한 기본정보를 수신하고, 기본정보가 신뢰성 기준 데이터베이스(250)에 저장되어 있는 기준을 만족하는지 여부를 판단해서 신뢰성을 평가할 수 있다.
- [0062] 예를 들어 기준정보가 판매 서버 P를 통해 판매되는 생산자 A가 생성하는 어플리케이션인 경우, 설치하고자 하는 어플리케이션이 판매 서버 P를 통해 판매되는 생산자 A가 생성하는 어플리케이션에 부합되면 설치하고자 하는 어플리케이션은 신뢰성이 있다고 판단한다.
- [0063] 신뢰성 평가부(218)는 업데이트 데이터의 경우 어플리케이션을 공급받은 판매서버와 생산자가 일치하는 경우 신뢰성이 있다고 판단할 수 있다.
- [0064] 제어부(210)는 단말 장치(200)의 전반적인 동작을 제어할 수 있다. 그리고, 제어부(210)는 다운로드부(211), 가상 머신부(212), 검증 데이터 생성부(213), 가상화 모듈 로드부(214), 검증 데이터 확인부(215), 검증 테이블 검색부(216), 비교 검증부(217) 및 신뢰성 평가부(218)의 기능을 수행할 수 있다. 제어부(210), 다운로드부(211), 가상 머신부(212), 검증 데이터 생성부(213), 가상화 모듈 로드부(214), 검증 데이터 확인부(215), 검증 테이블 검색부(216), 비교 검증부(217) 및 신뢰성 평가부(218)를 구분하여 도시한 것은 각 기능들을 구별하여 설명하기 위함이다. 따라서 제어부(210)는 다운로드부(211), 가상 머신부(212), 검증 데이터 생성부(213), 가상화 모듈 로드부(214), 검증 데이터 확인부(215), 검증 테이블 검색부(216), 비교 검증부(217) 및 신뢰성 평가부(218) 각각의 기능을 수행하도록 구성된(configured) 적어도 하나의 프로세서를 포함할 수 있다. 또한, 제어부(210)는 다운로드부(211), 가상 머신부(212), 검증 데이터 생성부(213), 가상화 모듈 로드부(214), 검증 데이터 확인부(215), 검증 테이블 검색부(216), 비교 검증부(217) 및 신뢰성 평가부(218) 각각의 기능 중 일부를 수행하도록 구성된(configured) 적어도 하나의 프로세서를 포함할 수 있다.
- [0065] 도 3은 본 발명의 다른 실시예에 따른 가상화 모듈을 관리하는 단말 장치의 구조를 나타낸다.
- [0066] 도 3를 참조하면, 단말 장치(300)는 제어부(310), 다운로드부(311), 가상 머신부(312), 검증 데이터 생성부(313), 가상화 모듈 로드부(314), 검증 테이블 검색부(316), 비교 검증부(317), 통신부(320) 및 메모리부(330)를 포함한다. 또한, 단말 장치(300)는 추가로 신뢰성 평가부(318), 신뢰성 데이터베이스(340) 및 신뢰성 기준 데이터베이스(350) 중 하나 내지는 모두를 더 포함할 수도 있다.
- [0067] 도 3에서 다운로드부(311), 가상 머신부(312), 가상화 모듈 로드부(314), 검증 테이블 검색부(316), 신뢰성 평가부(318), 통신부(320), 메모리부(330), 신뢰성 데이터베이스(340) 및 신뢰성 기준 데이터베이스(350)는 도 2와 구성과 동일한 동작을 수행함을 상세한 설명을 생략한다.
- [0068] 검증 데이터 생성부(313)는 ODEX 파일을 검증할 수 있는 검증 데이터를 생성하고, 생성한 검증 데이터를 메모리부(330)의 검증 테이블에 저장한다.
- [0069] 검증 데이터 생성부(313)는 어플리케이션의 실행을 요청 받으면, 가상화 모듈 로드부(314)를 통해 로드한 어플리케이션에 대응하는 ODEX 파일을 이용해서 검증 데이터를 생성하고, 생성한 검증 데이터를 비교 검증부(317)로 제공할 수도 있다.
- [0070] 검증 데이터 생성부(313)는 업데이트 데이터가 설치될 때 ODEX 파일이 변경되면, 변경된 ODEX 파일을 검증할 수 있는 변경된 검증 데이터를 생성하고, 검증 테이블에 저장된 검증 데이터를 변경된 검증 데이터로 갱신한다.
- [0071] 검증 데이터 생성부(313)는 가 도 2의 검증 데이터 생성부(213)와 다른 점은 생성한 검증 데이터를 ODEX 파일에

포함시키지 않는 것이다.

- [0072] 비교 검증부(317)는 검증 테이블 검색부(316)에서 검색한 검증 데이터와 검증 데이터 생성부(313)에서 생성한 검증 데이터를 비교해서 일치하면 ODEX 파일이 비정상적으로 수정되지 않았다고 검증하고, 일치하지 않으면 ODEX 파일이 비정상적으로 수정되었다고 검증한다.
- [0073] 제어부(310)는 단말 장치(300)의 전반적인 동작을 제어할 수 있다. 그리고, 제어부(310)는 다운로드부(311), 가상 머신부(312), 검증 데이터 생성부(313), 가상화 모듈 로드부(314), 검증 테이블 검색부(316), 비교 검증부(317) 및 신뢰성 평가부(318)의 기능을 수행할 수 있다. 제어부(310), 다운로드부(311), 가상 머신부(312), 검증 데이터 생성부(313), 가상화 모듈 로드부(314), 검증 테이블 검색부(316), 비교 검증부(317) 및 신뢰성 평가부(318)를 구분하여 도시한 것은 각 기능들을 구별하여 설명하기 위함이다. 따라서 제어부(310)는 다운로드부(311), 가상 머신부(312), 검증 데이터 생성부(313), 가상화 모듈 로드부(314), 검증 테이블 검색부(316), 비교 검증부(317) 및 신뢰성 평가부(318) 각각의 기능을 수행하도록 구성된(configured) 적어도 하나의 프로세서를 포함할 수 있다. 또한, 제어부(310)는 다운로드부(311), 가상 머신부(312), 검증 데이터 생성부(313), 가상화 모듈 로드부(314), 검증 테이블 검색부(316), 비교 검증부(317) 및 신뢰성 평가부(318) 각각의 기능 중 일부를 수행하도록 구성된(configured) 적어도 하나의 프로세서를 포함할 수 있다.
- [0074] 이하, 상기와 같이 구성된 본 발명에 따른 장치에서 가상화 모듈을 관리하는 방법을 아래에서 도면을 참조하여 설명한다.
- [0075] 도 6은 본 발명의 실시예에 따라 도 2의 단말 장치에서 어플리케이션을 설정하는 과정을 도시한 흐름도이다.
- [0076] 도 6을 참조하면, 단말 장치(200)는 610단계에서 설치하고자 하는 어플리케이션을 판매 서버로부터 다운로드한다. 그리고, 단말 장치(200)는 612단계에서 APK 파일로된 어플리케이션에서 DEX 파일을 추출한다. 그리고, 단말 장치(200)는 614단계에서 DEX 파일을 단말 장치(200)에 최적화 해서 ODEX 파일을 생성한다.
- [0077] 그리고, 단말 장치(200)는 616단계에서 ODEX 파일을 검증할 수 있는 검증 데이터를 생성한다. 그리고, 단말 장치(200)는 618단계에서 검증 데이터를 검증 테이블에 저장한다. 그리고, 단말 장치(200)는 620단계에서 ODEX 파일에 검증 데이터를 포함시켜 저장한다.
- [0078] 도 7은 본 발명의 실시예에 따라 도 2의 단말 장치에서 어플리케이션을 업데이트하는 과정을 도시한 흐름도이다.
- [0079] 도 7을 참조하면, 단말 장치(200)는 710단계에서 어플리케이션의 업데이트 이벤트의 발생을 감지하면, 712단계에서 어플리케이션 판매 서버로부터 업데이트 데이터를 수신한다.
- [0080] 그리고, 단말 장치(200)는 714단계에서 업데이트 데이터를 설치한다.
- [0081] 그리고, 단말 장치(200)는 716단계에서 업데이트 데이터의 설치로 인해서 ODEX 파일이 변경되었는지 여부를 확인한다.
- [0082] 716단계의 확인결과 업데이트 데이터의 설치로 인해서 ODEX 파일이 변경되지 않았으면, 단말 장치(200)는 본 알고리즘을 종료한다.
- [0083] 716단계의 확인결과 업데이트 데이터의 설치로 인해서 ODEX 파일이 변경되었으면, 단말 장치(200)는 718단계에서 변경된 ODEX 파일을 검증할 수 있는 변경된 검증 데이터를 생성한다.
- [0084] 그리고, 단말 장치(200)는 720단계에서 검증 테이블에 저장된 검증 데이터를 변경된 검증 데이터로 갱신한다. 그리고, 단말 장치(200)는 722단계에서 변경된 ODEX 파일에 변경된 검증 데이터를 포함시켜 저장한다.
- [0085] 도 8은 본 발명의 실시예에 따라 도 2의 단말 장치에서 어플리케이션을 실행하는 과정을 도시한 흐름도이다.
- [0086] 도 8을 참조하면, 단말 장치(200)는 810단계에서 어플리케이션을 실행하는 이벤트의 발생을 감지하면, 812단계에서 어플리케이션에 대응하는 가상화 모듈인 ODEX 파일을 로드한다.
- [0087] 그리고, 단말 장치(200)는 814단계에서 ODEX 파일에 검증 데이터가 포함되었는지 확인한다.
- [0088] 814단계의 확인결과 ODEX 파일에 검증 데이터가 포함되어 있으면, 단말 장치(200)는 816단계에서 ODEX 파일에 대응하는 검증 데이터를 검증 테이블에서 검색한다.

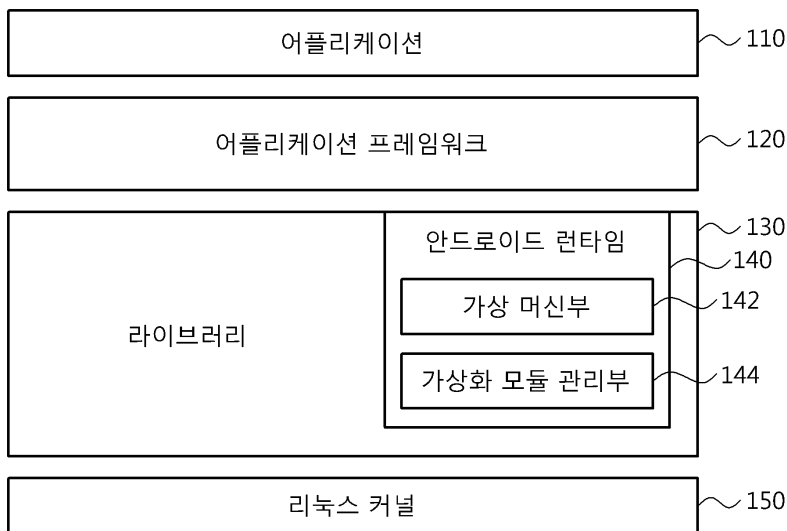
- [0089] 그리고, 단말 장치(200)는 818단계에서 검증 테이블의 검증 데이터와 ODEX 파일에 포함된 검증 데이터를 비교하고, 비교결과 일치하면 ODEX 파일이 비정상적으로 수정되지 않았다고 검증하고, 일치하지 않으면, ODEX 파일이 비정상적으로 수정되었다고 검증한다.
- [0090] 그리고, 단말 장치(200)는 820단계에서 비교 검증결과가 ODEX 파일의 비정상적으로 수정인지 확인한다.
- [0091] 820단계의 확인결과 ODEX 파일의 비정상적으로 수정되지 않은 경우, 단말 장치(200)는 822단계에서 어플리케이션에 대응하는 가상 머신 객체를 생성하고, 824단계에서 생성한 가상 머신 객체를 이용해서 ODEX 파일을 실행한다.
- [0092] 한편, 814단계의 확인결과 ODEX 파일에 검증 데이터가 포함되어 있지 않거나 또는 820단계의 확인결과 ODEX 파일의 비정상적으로 수정된 경우, 단말 장치(200)는 본 알고리즘을 종료한다.
- [0093] 도 9는 본 발명의 실시예에 따라 도 3의 단말 장치에서 어플리케이션을 설정하는 과정을 도시한 흐름도이다.
- [0094] 도 9를 참조하면, 단말 장치(300)는 910단계에서 설치하고자 하는 어플리케이션을 판매 서버로부터 다운로드한다. 그리고, 단말 장치(300)는 912단계에서 APK 파일로된 어플리케이션에서 DEX 파일을 추출한다. 그리고, 단말 장치(200)는 914단계에서 DEX 파일을 단말 장치(300)에 최적화 해서 ODEX 파일을 생성한다.
- [0095] 그리고, 단말 장치(300)는 916단계에서 ODEX 파일을 검증할 수 있는 검증 데이터를 생성한다. 그리고, 단말 장치(300)는 918단계에서 검증 데이터를 검증 테이블에 저장한다. 그리고, 단말 장치(300)는 920단계에서 ODEX 파일을 저장한다. 이때, 단말 장치(300)는 ODEX 파일에 검증 데이터를 포함시키지 않는다.
- [0096] 도 10은 본 발명의 실시예에 따라 도 3의 단말 장치에서 어플리케이션을 업데이트하는 과정을 도시한 흐름도이다.
- [0097] 도 10을 참조하면, 단말 장치(300)는 1010단계에서 어플리케이션의 업데이트 이벤트의 발생을 감지하면, 1012단계에서 어플리케이션 판매 서버로부터 업데이트 데이터를 수신한다.
- [0098] 그리고, 단말 장치(300)는 1014단계에서 업데이트 데이터를 설치한다.
- [0099] 그리고, 단말 장치(300)는 1016단계에서 업데이트 데이터의 설치로 인해서 ODEX 파일이 변경되었는지 여부를 확인한다.
- [0100] 1016단계의 확인결과 업데이트 데이터의 설치로 인해서 ODEX 파일이 변경되지 않았으면, 단말 장치(300)는 본 알고리즘을 종료한다.
- [0101] 1016단계의 확인결과 업데이트 데이터의 설치로 인해서 ODEX 파일이 변경되었으면, 단말 장치(300)는 1018단계에서 변경된 ODEX 파일을 검증할 수 있는 변경된 검증 데이터를 생성한다.
- [0102] 그리고, 단말 장치(300)는 1020단계에서 검증 테이블에 저장된 검증 데이터를 변경된 검증 데이터로 갱신한다. 그리고, 단말 장치(300)는 1022단계에서 변경된 ODEX 파일을 포함시켜 저장한다. 이때, 단말 장치(300)는 변경된 ODEX 파일에 변경된 검증 데이터를 포함시키지 않는다.
- [0103] 도 11은 본 발명의 실시예에 따라 도 3의 단말 장치에서 어플리케이션을 실행하는 과정을 도시한 흐름도이다.
- [0104] 도 11을 참조하면, 단말 장치(300)는 1110단계에서 어플리케이션을 실행하는 이벤트의 발생을 감지하면, 1112단계에서 어플리케이션에 대응하는 가상화 모듈인 ODEX 파일을 로드한다.
- [0105] 그리고, 단말 장치(300)는 1114단계에서 로드한 ODEX 파일을 이용해서 검증 데이터를 생성한다.
- [0106] 그리고, 단말 장치(300)는 1116단계에서 ODEX 파일에 대응하는 검증 데이터를 검증 테이블에서 검색한다.
- [0107] 그리고, 단말 장치(300)는 1118단계에서 검증 테이블의 검증 데이터와 ODEX 파일을 이용해 생성한 검증 데이터를 비교하고, 비교결과 일치하면 ODEX 파일이 비정상적으로 수정되지 않았다고 검증하고, 일치하지 않으면, ODEX 파일이 비정상적으로 수정되었다고 검증한다.
- [0108] 그리고, 단말 장치(300)는 1120단계에서 비교 검증결과가 ODEX 파일의 비정상적으로 수정인지 확인한다.
- [0109] 1120단계의 확인결과 ODEX 파일의 비정상적으로 수정되지 않은 경우, 단말 장치(300)는 1122단계에서 어플리케이션에 대응하는 가상 머신 객체를 생성하고, 1124단계에서 생성한 가상 머신 객체를 이용해서 ODEX 파일을 실행

행한다.

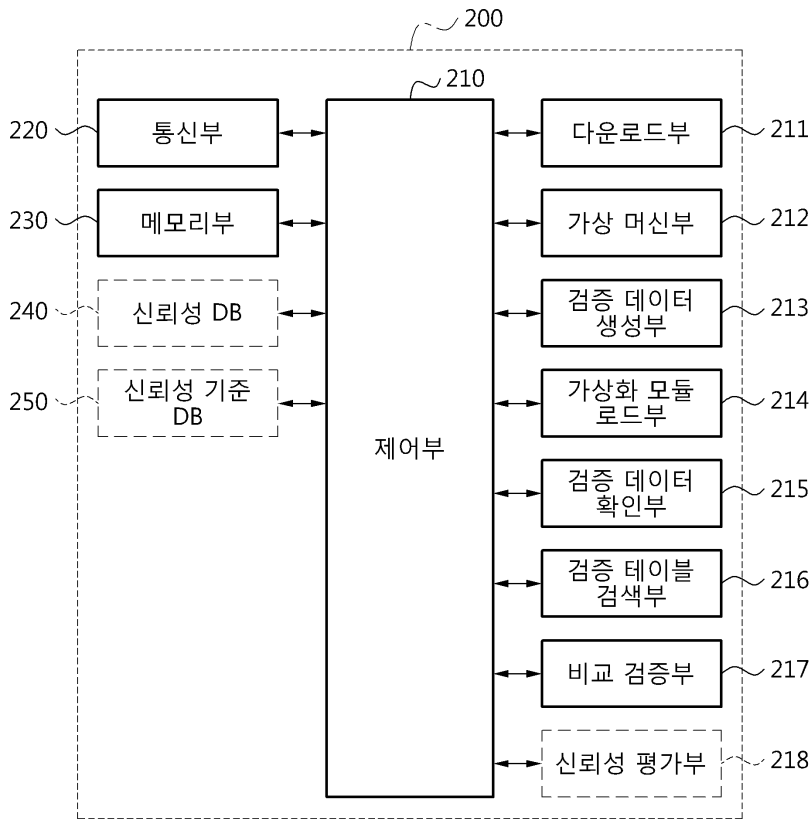
- [0110] 1120단계의 확인결과 ODEX 파일의 비정상적으로 수정된 경우, 단말 장치(300)는 본 알고리즘을 종료한다.
- [0111] 본 발명의 실시 예에 따른 방법들은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 매체에 기록되는 프로그램 명령은 본 발명을 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다.
- [0112] 이상과 같이 본 발명은 비록 한정된 실시예와 도면에 의해 설명되었으나, 본 발명은 상기의 실시예에 한정되는 것은 아니며, 본 발명이 속하는 분야에서 통상의 지식을 가진 자라면 이러한 기재로부터 다양한 수정 및 변형이 가능하다.
- [0113] 그러므로, 본 발명의 범위는 설명된 실시예에 국한되어 정해져서는 아니 되며, 후술하는 특허청구범위뿐 아니라 이 특허청구범위와 균등한 것들에 의해 정해져야 한다.

도면

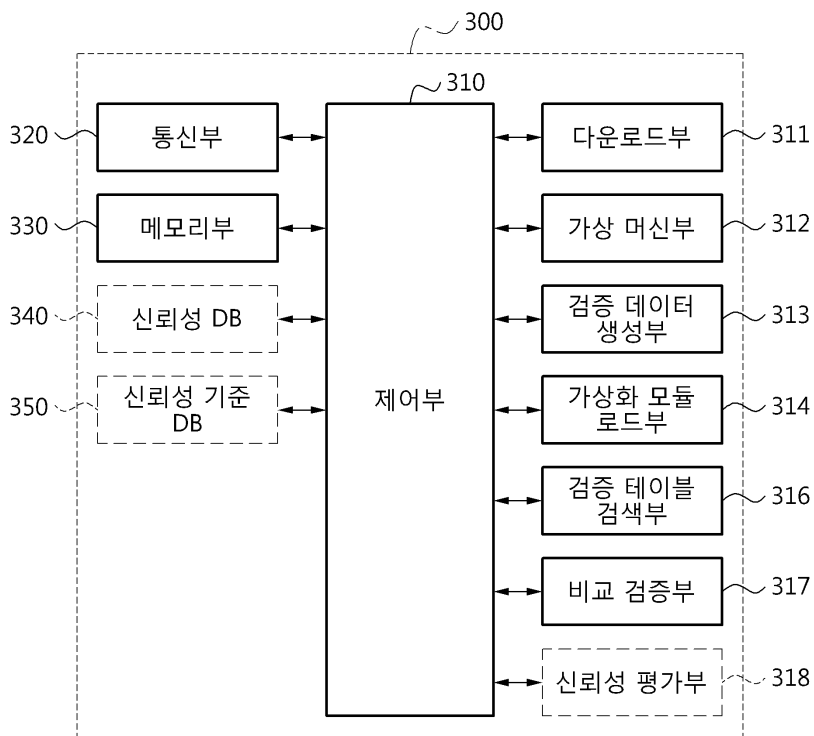
도면1



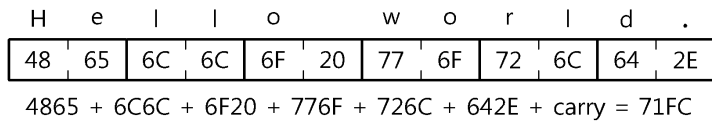
도면2



도면3



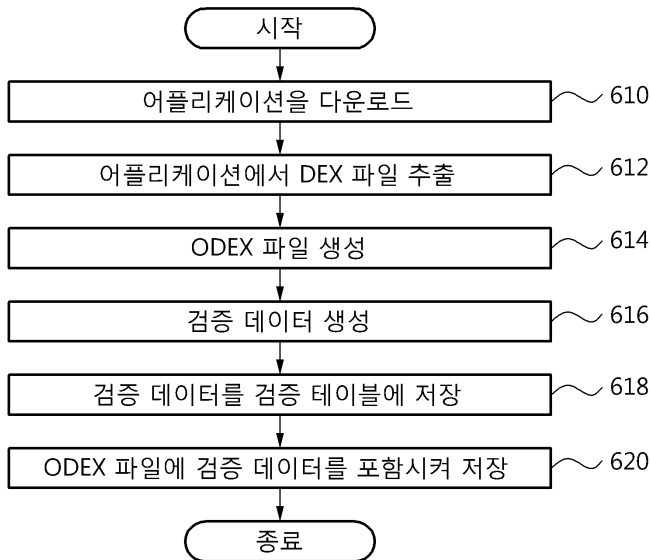
도면4



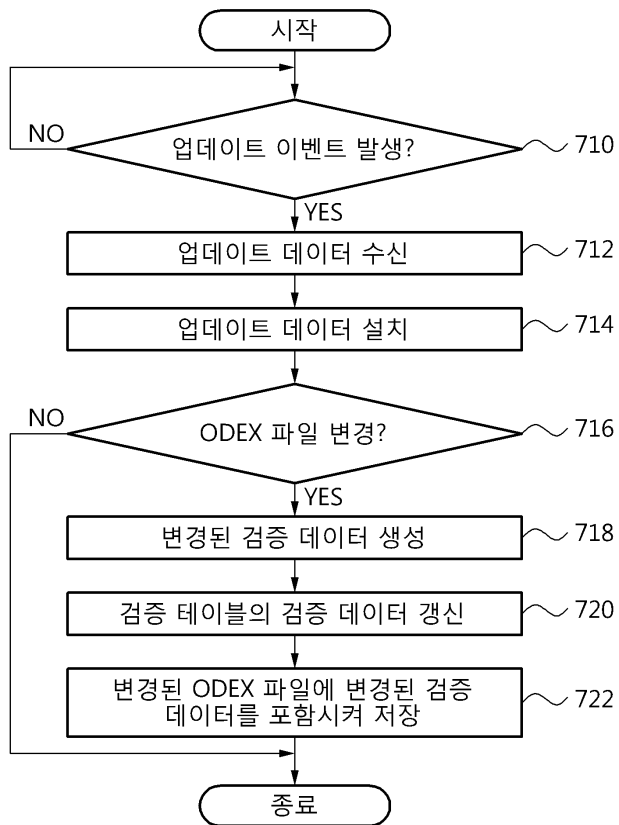
도면5

식별 정보	이름	검증 정보
...
0xE4	system@app@ Contacts.apk	0x6E58FD
0xE5	system@app@ Phone.apk	0x3B760C
0xE6	data@app@com. keyped-1.apk	0x0E761A
...

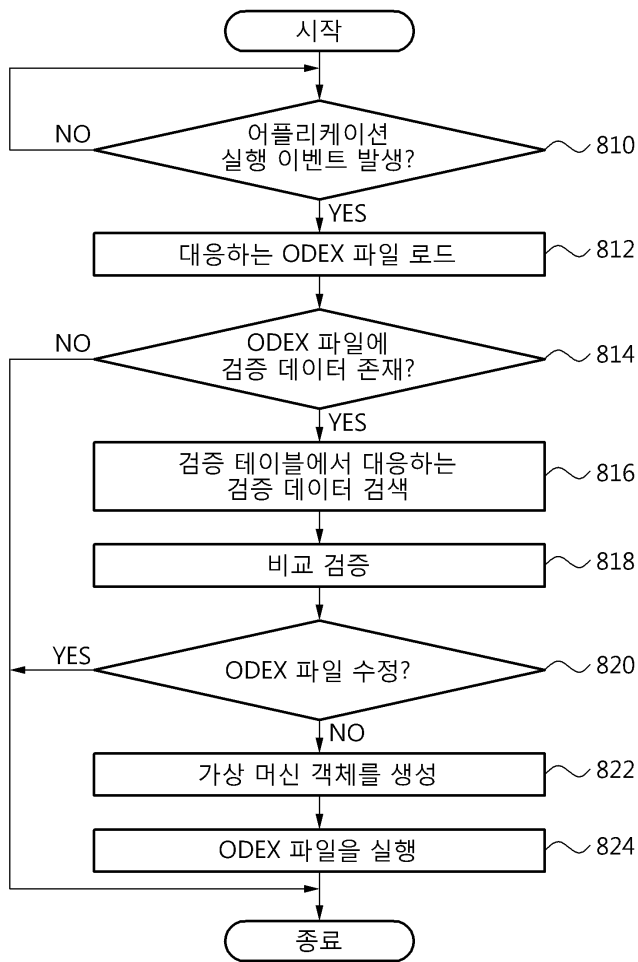
도면6



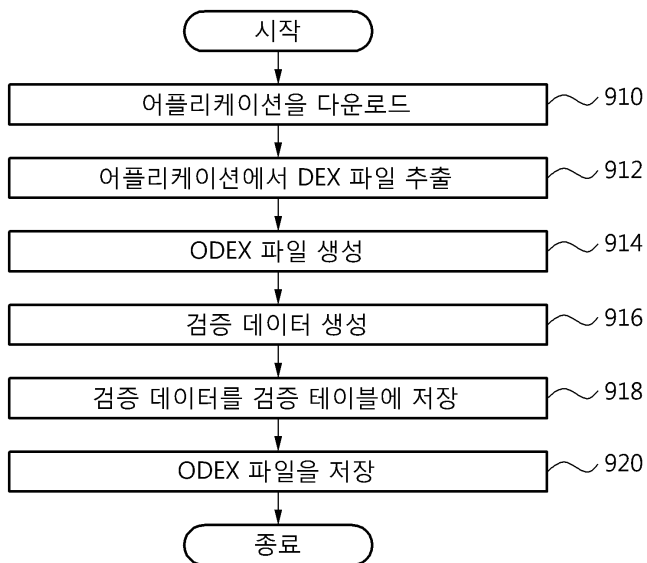
도면7



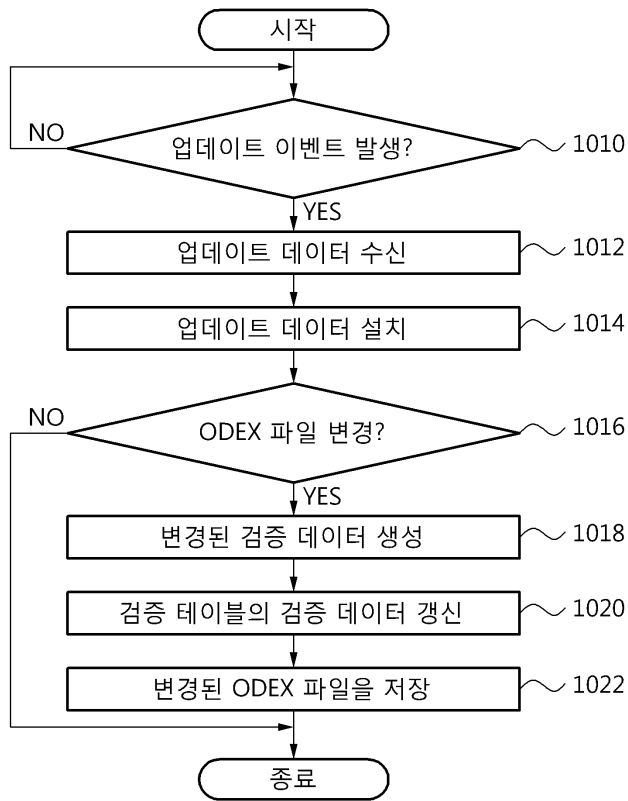
도면8



도면9



도면10



도면11

