



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2018-0129623
(43) 공개일자 2018년12월05일

(51) 국제특허분류(Int. Cl.)

G06F 21/56 (2013.01)

(52) CPC특허분류

G06F 21/562 (2013.01)

(21) 출원번호 10-2018-0043918

(22) 출원일자 2018년04월16일

심사청구일자 2018년04월16일

(30) 우선권주장

1020170065636 2017년05월26일 대한민국(KR)

(71) 출원인

충남대학교산학협력단

대전광역시 유성구 대학로 99 (궁동, 충남대학교)

(72) 발명자

류재철

대전광역시 유성구 어은로 57, 132동 801호

김상후

대전광역시 유성구 도안동로 523, 215동 1102호

이우람

대전광역시 유성구 온천북로33번길 21-16, 404호

(74) 대리인

홍성욱, 심경식

전체 청구항 수 : 총 12 항

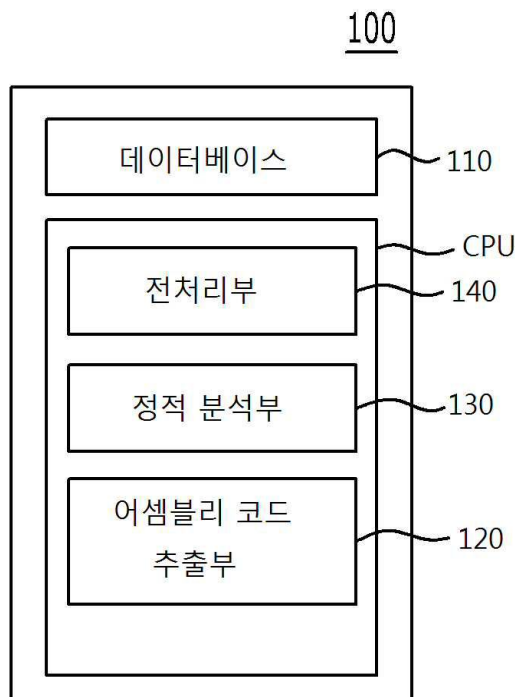
(54) 발명의 명칭 연관된 다중 파일 정적 분석 장치

(57) 요약

본 발명은 연관된 다중 파일 정적 분석 장치에 관한 것으로, 디스어셈블된 바이너리의 프로그램 이름, 함수의 이름, 어셈블리 코드, 분기 정보, 추적 정보, 추적 상태 및 레지스터 값 중 적어도 하나를 저장하는 데이터베이스, 상기 데이터베이스를 검색하여, 분석 대상 바이너리에서 사용되는 복수의 파일과, 각 파일에 포함되는 복수의 함

(뒷면에 계속)

대표도 - 도1



수 목록을 표시하고, 상기 표시된 복수의 함수 목록 중 제1 함수가 선택되면, 상기 제1 함수의 정방향 또는 역방향 콜체인을 표시하고, 상기 제1 함수, 상기 제1 함수의 정방향 함수인 제2 함수 또는 상기 제1 함수의 역방향 함수인 제3 함수의 어셈블리 코드를 표시하고, 상기 제1 함수의 선택이 해지되고 상기 제2 함수가 선택되면 상기 제1 함수, 상기 제2 함수, 상기 제3 함수의 어셈블리 코드 표시를 해제하고, 상기 제2 함수, 상기 제2 함수의 정방향 함수인 제4 함수 또는 역방향 함수인 제5 함수의 어셈블리 코드를 표시하는 어셈블리 코드 추출부, 및 상기 어셈블리 코드 추출부에서 상기 제1 함수 내지 제5 함수 중 하나에 대응하는 레지스터 값의 추적을 요청하는 경우, 상기 데이터베이스를 통해 대응되는 레지스터 정보를 추출하여 상기 어셈블리 코드 추출부로 제공하는 정적 분석부를 포함한다. 본 발명에 의하면, 기계어 수준에서도 서로 연관된 다중 파일의 함수의 흐름을 자동으로 표시하여 정적 분석이 가능하도록 제공하므로 복수의 실행 파일에서 함께 활용되는 특정 함수의 콜체인을 쉽게 추적할 수 있어 정적 분석 속도를 향상시킬 수 있다.

이 발명을 지원한 국가연구개발사업

과제고유번호	1711058948
부처명	과학기술정보통신부
연구관리전문기관	정보통신기술진흥센터(IITP)
연구사업명	정보보호핵심원천기술개발
연구과제명	(안전성 연구 2세부) 운영체제 안전성 연구
기 여 율	1/1
주관기관	충남대학교 산학협력단
연구기간	2017.07.01 ~ 2018.04.30

명세서

청구범위

청구항 1

디스어셈블된 바이너리의 프로그램 이름, 함수의 이름, 어셈블리 코드, 분기 정보, 추적 정보, 추적 상태 및 레지스터 값 중 적어도 하나를 저장하는 데이터베이스;

상기 데이터베이스를 검색하여, 분석 대상 바이너리에서 사용되는 복수의 파일과, 각 파일에 포함되는 복수의 함수 목록을 표시하고, 상기 표시된 복수의 함수 목록 중 제1 함수가 선택되면, 상기 제1 함수의 정방향 또는 역방향 콜체인을 표시하고, 상기 제1 함수, 상기 제1 함수의 정방향 함수인 제2 함수 또는 상기 제1 함수의 역방향 함수인 제3 함수의 어셈블리 코드를 표시하고, 상기 제1 함수의 선택이 해지되고 상기 제2 함수가 선택되면 상기 제1 함수, 상기 제2 함수, 상기 제3 함수의 어셈블리 코드 표시를 해제하고, 상기 제2 함수, 상기 제2 함수의 정방향 함수인 제4 함수 또는 역방향 함수인 제5 함수의 어셈블리 코드를 표시하는 어셈블리 코드 추출부; 및

상기 어셈블리 코드 추출부에서 상기 제1 함수 내지 제5 함수 중 하나에 대응하는 레지스터 값의 추적을 요청하는 경우, 상기 데이터베이스를 통해 대응되는 레지스터 정보를 추출하여 상기 어셈블리 코드 추출부로 제공하는 정적 분석부;

를 포함하는 연관된 다중 파일 정적 분석 장치.

청구항 2

제1항에 있어서,

상기 복수의 파일 각각에 대응하는 프로그램 이름, 함수의 이름, 어셈블리 코드, 분기 정보 및 레지스터 값을 생성하는 전처리부;를 더 포함하는, 연관된 다중 파일 정적 분석 장치.

청구항 3

제1항에 있어서,

상기 정적 분석 장치는, 클라이언트 장치와 연결되고,

상기 클라이언트 장치는, 상기 복수의 파일 각각에 대응하는 프로그램 이름, 함수의 이름, 어셈블리 코드, 분기 정보 및 레지스터 값을 생성하여 상기 정적 분석 장치로 전달하는, 연관된 다중 파일 정적 분석 장치.

청구항 4

제1항에 있어서,

상기 정적 분석부는, 특정 레지스터 값이 추적 대상으로 선택되면 상기 추적 대상의 추적 정보와 추적 상태를 상기 데이터베이스에 저장한 후 추적을 시작하는, 연관된 다중 파일 정적 분석 장치.

청구항 5

제4항에 있어서,

상기 정적 분석부는, 정추적이 요청되면 추적 대상을 분석하여 분기 명령이 있는지 판단하고, 분기 명령이 있다고 판단되면 추적 대상이 존재하는지 판단하고, 추적 대상이 존재하면 역추적 결과에 따라 추적 완료 또는 메모

리 추적을 수행하는, 연관된 다중 파일 정적 분석 장치.

청구항 6

제5항에 있어서,

상기 정적 분석부는, 상기 추적 대상이 존재하지 않으면 다음 명령어를 확인하여 상기 추적 대상 분석 과정부터 다시 수행하는, 연관된 다중 파일 정적 분석 장치.

청구항 7

제5항에 있어서,

상기 정적 분석부는, 상기 분기 명령이 없다고 판단되면 다른 함수에게 영향을 주는지 또는 다른 함수로부터 영향을 받는지에 따라 추적 레지스터 값을 변경하여 다음 명령어를 확인한 후 상기 추적 대상 분석 과정을 다시 수행하거나, 또는 추적을 완료하는, 연관된 다중 파일 정적 분석 장치.

청구항 8

제4항에 있어서,

상기 정적 분석부는, 역추적이 요청되면 추적 대상을 확인하고, 추적 대상이 존재하는지 판단하고, 추적 대상이 존재하면 다른 함수로부터 영향을 받는지에 따라 레지스터 값을 확인하는 단계로 넘어가거나 또는 참조 주소를 확인하는, 연관된 다중 파일 정적 분석 장치.

청구항 9

제8항에 있어서,

상기 정적 분석부는, 상기 추적 대상이 존재하지 않으면 참조 주소를 확인하는, 연관된 다중 파일 정적 분석 장치.

청구항 10

제9항에 있어서,

상기 정적 분석부는, 상기 참조 주소를 확인하여 내부 주소인지 외부 파일인지 판단하고, 상기 참조 주소가 내부 주소 또는 외부 파일인 경우, 추적 대상을 수정한 후 상기 추적 대상 확인 단계부터 다시 수행하고, 상기 참조 주소가 내부 주소도 아니고 외부 파일도 아닌 경우, 추적을 완료하는, 연관된 다중 파일 정적 분석 장치.

청구항 11

제8항에 있어서,

상기 정적 분석부는, 상기 레지스터 값을 확인하여, 상기 레지스터 값이 존재하면 명령어에 따른 추적 대상을 추가 또는 변경하고, 상기 레지스터 값이 존재하지 않으면 메모리 추적을 수행하는, 연관된 다중 파일 정적 분석 장치.

청구항 12

제1항에 있어서,

상기 어셈블리 코드 추출부는, 상기 복수의 함수 목록 중 특정 함수의 정방향 또는 역방향 콜체인을 표시하고, 상기 콜체인에 포함된 다른 파일의 함수를 선택한 경우, 상기 정적 분석부에서 상기 선택된 다른 파일의 함수를 호출하여 표시하는 방법으로 연관된 다중 파일의 실행 흐름을 분석하는, 연관된 다중 파일 정적 분석 장치.

발명의 설명

기술 분야

[0001] 본 발명은 바이너리 분석 중 디스어셈블된 코드를 따라가며 실행흐름을 파악하는 정적 분석에 관한 것이다.

배경 기술

- [0002] 스마트폰, PC 등을 비롯하여 사용자의 주변에 도움을 주는 각종 기기들은 점차 널리 보급되고 있다. 이러한 기기들은 사용자들에게 각종 생활 편의를 제공하고 있다.
- [0003] 이러한 기기들이 사용자에게 생활 편의를 제공하는 과정에서 필수적으로 개인정보가 사용되게 되며, 이에 따라 개인 정보 탈취 등을 목적으로 하는 악성 행위를 유발하는 프로그램들이 늘어나고 있다. 이러한 악성 행위를 하는 프로그램에 대처하기 위하여 해당되는 프로그램의 바이너리를 분석하는 것이 필수적이다.
- [0004] 대표적인 바이너리 분석 방법에는 크게 정적 분석 방법과 동적 분석 방법이 있고, 이들 중 정적 분석 방법은 디스어셈블된 코드를 따라가며 실행 흐름을 파악하는 방법이고, 동적 분석 방법은 디버거를 붙여서 코드의 줄 단위로 실행하며 분석하는 방법이다.
- [0005] 일반적으로, 바이너리 분석을 수행하기 위하여 상술한 두 가지 분석 방법을 동시에 사용하여 분석을 진행한다. 그러나, 악성 행위를 하는 프로그램 중 일부는 동적 분석 대상이 되는 파일에 대해 디버거를 활용하여 할 때 해당 파일을 종료시키거나 에러를 발생시켜 동적 분석을 방해하는 경우가 있다.
- [0006] 이러한 경우, 정적 분석 방법만으로 바이너리 분석을 진행해야 하기 때문에 분석자가 수동으로 디스어셈블된 코드를 따라가며 실행흐름을 파악하는데 소요되는 시간이 크게 늘어나는 문제가 있다.
- [0007] 또한, 정적 분석 방법으로 대표적으로 사용되는 상용 프로그램인 IDA pro는 하나의 바이너리를 대상으로 디스어셈블을 진행하고 함수 콜체인을 보여줘서 분석을 용이하게 한다. 정적 분석을 진행함에 있어서 사용되는 종래의 IDA 프로그램은 하나의 바이너리를 디스어셈블하여 리버싱 할 수 있도록 하는 프로그램으로, 분석 중 다른 프로그램에서 정의한 함수가 발견되는 경우 импорт 테이블을 확인하여 해당 함수가 어떤 프로그램에 정의되어있는지 사용자가 직접 확인하고, 확인된 프로그램을 다시 IDA로 열어 분석해야하는 번거로움이 있다.
- [0008] 또한, 종래의 IDA 프로그램은 바이너리 분석시에 레지스터의 값이 어디로부터 영향을 받았는지(역방향 분석), 어디로 영향을 끼치는지(정방향 분석) 각각에 대한 분석을 진행하는 경우 줄 단위로 확인하여 분석을 진행하여야 하므로 분석에 오랜 시간이 걸리며, 올바른 분석이 진행되었는지 확인하는데 어려움이 있다.

발명의 내용

해결하려는 과제

[0009] 본 발명은 연관된 복수의 파일들을 분석하여 복수의 파일들 각각에 포함된 함수들의 관계를 데이터베이스화함으로써 서로 다른 파일에 포함된 함수들의 관계를 추적할 수 있는 연관된 다중 파일 정적 분석 장치를 제공하는데 그 목적이 있다.

과제의 해결 수단

[0010] 상기의 목적을 달성하기 위해 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치는 디스어셈블된 바이너리의 프로그램 이름, 함수의 이름, 어셈블리 코드, 분기 정보, 추적 정보, 추적 상태 및 레지스터 값 중 적어도 하나를 저장하는 데이터베이스, 상기 데이터베이스를 검색하여, 분석 대상 바이너리에서 사용되는 복수의 파일과, 각 파일에 포함되는 복수의 함수 목록을 표시하고, 상기 표시된 복수의 함수 목록 중 제1 함수가 선택되면, 상기 제1 함수의 정방향 또는 역방향 콜체인을 표시하고, 상기 제1 함수, 상기 제1 함수의 정방향 함수인 제2 함수 또는 상기 제1 함수의 역방향 함수인 제3 함수의 어셈블리 코드를 표시하고, 상기 제1 함수의 선택이

해지되고 상기 제2 함수가 선택되면 상기 제1 함수, 상기 제2 함수, 상기 제3 함수의 어셈블리 코드 표시를 해제하고, 상기 제2 함수, 상기 제2 함수의 정방향 함수인 제4 함수 또는 역방향 함수인 제5 함수의 어셈블리 코드를 표시하는 어셈블리 코드 추출부, 및 상기 어셈블리 코드 추출부에서 상기 제1 함수 내지 제5 함수 중 하나에 대응하는 레지스터 값의 추적을 요청하는 경우, 상기 데이터베이스를 통해 대응되는 레지스터 정보를 추출하여 상기 어셈블리 코드 추출부로 제공하는 정적 분석부를 포함한다.

- [0011] 연관된 다중 파일 정적 분석 장치는 상기 복수의 파일 각각에 대응하는 프로그램 이름, 함수의 이름, 어셈블리 코드, 분기 정보 및 레지스터 값을 생성하는 전처리부를 더 포함할 수 있다.
- [0012] 정적 분석 장치는, 클라이언트 장치와 연결되고, 상기 클라이언트 장치는, 상기 복수의 파일 각각에 대응하는 프로그램 이름, 함수의 이름, 어셈블리 코드, 분기 정보 및 레지스터 값을 생성하여 상기 정적 분석 장치로 전달할 수 있다.
- [0013] 또한, 정적 분석부는 특정 레지스터 값이 추적 대상으로 선택되면 상기 추적 대상의 추적 정보와 추적 상태를 상기 데이터베이스에 저장한 후 추적을 시작할 수 있다.
- [0014] 또한, 정적 분석부는 정주적이 요청되면 추적 대상을 분석하여 분기 명령이 있는지 판단하고, 분기 명령이 있다고 판단되면 추적 대상이 존재하는지 판단하고, 추적 대상이 존재하면 역추적 결과에 따라 추적 완료 또는 메모리 추적을 수행할 수 있다.
- [0015] 또한, 정적 분석부는 상기 추적 대상이 존재하지 않으면 다음 명령어를 확인하여 상기 추적 대상 분석 과정부터 다시 수행할 수 있다.
- [0016] 또한, 정적 분석부는 상기 분기 명령이 없다고 판단되면 다른 함수에게 영향을 주는지 또는 다른 함수로부터 영향을 받는지에 따라 추적 레지스터 값을 변경하여 다음 명령어를 확인한 후 상기 추적 대상 분석 과정을 다시 수행하거나, 또는 추적을 완료할 수 있다.
- [0017] 또한, 정적 분석부는 역주적이 요청되면 추적 대상을 확인하고, 추적 대상이 존재하는지 판단하고, 추적 대상이 존재하면 다른 함수로부터 영향을 받는지에 따라 레지스터 값을 확인하는 단계로 넘어가거나 또는 참조 주소를 확인할 수 있다.
- [0018] 또한, 정적 분석부는 상기 추적 대상이 존재하지 않으면 참조 주소를 확인할 수 있다.
- [0019] 또한, 정적 분석부는 상기 참조 주소를 확인하여 내부 주소인지 외부 파일인지 판단하고, 상기 참조 주소가 내부 주소 또는 외부 파일인 경우, 추적 대상을 수정한 후 상기 추적 대상 확인 단계부터 다시 수행하고, 상기 참조 주소가 내부 주소도 아니고 외부 파일도 아닌 경우, 추적을 완료할 수 있다.
- [0020] 또한, 정적 분석부는 상기 레지스터 값을 확인하여, 상기 레지스터 값이 존재하면 명령어에 따른 추적 대상을 추가 또는 변경하고, 상기 레지스터 값이 존재하지 않으면 메모리 추적을 수행할 수 있다.
- [0021] 또한, 어셈블리 코드 추출부는 상기 복수의 함수 목록 중 특정 함수의 정방향 또는 역방향 콜체인을 표시하고, 상기 콜체인에 포함된 다른 파일의 함수를 선택한 경우, 상기 정적 분석부에서 상기 선택된 다른 파일의 함수를 호출하여 표시하는 방법으로 연관된 다중 파일의 실행 흐름을 분석할 수 있다.

발명의 효과

- [0022] 본 발명에 의하면, 기계어 수준에서도 서로 연관된 다중 파일의 함수의 흐름을 자동으로 표시하여 정적 분석이 가능하도록 제공하므로 복수의 실행 파일에서 함께 활용되는 특정 함수의 콜체인을 쉽게 추적할 수 있어 정적 분석 속도를 향상시킬 수 있는 효과가 있다.

도면의 간단한 설명

- [0023] 도 1은 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치의 개략적인 구성도이다.
- 도 2는 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치를 포함하는 시스템의 개략적인 구성도이다.
- 도 3은 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치의 분석 결과를 설명하기 위한 도면이다.
- 도 4는 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치의 콜체인을 설명하기 위한 개념도이다.

도 5 내지 도 6은 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치의 정방향 분석을 설명하기 위한 도면이다.

도 7 내지 도 8은 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치의 역방향 분석을 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

- [0024] 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.
- [0025] 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.
- [0026] 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.
- [0027] 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.
- [0028] 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.
- [0029] 명세서 및 청구범위 전체에서, 어떤 부분이 어떤 구성 요소를 포함한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성 요소를 제외하는 것이 아니라 다른 구성 요소를 더 포함할 수 있다는 것을 의미한다.
- [0031] 이하, 본 발명에 따른 바람직한 실시예를 첨부된 도면을 참조하여 상세하게 설명한다.
- [0032] 도 1은 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치의 개략적인 구성도이다.
- [0033] 도 1을 참조하면, 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치(100)는 데이터베이스(110), 어셈블리 코드 추출부(120) 및 정적 분석부(130)를 포함한다. 또한, 연관된 다중 파일 정적 분석 장치(100)는 전처리부(140)를 더 포함할 수 있고, 도시하지는 않았지만, 통신부, 인터페이스부, 디스플레이부 등을 더 포함할 수 있다.
- [0034] 구체적으로, 데이터베이스(110)는 디스어셈블된 바이너리의 프로그램 이름, 함수의 이름 어셈블리 코드, 분기 정보, 추적 정보, 추적 상태 및 레지스터 값 중 적어도 하나를 저장할 수 있다. 이때, 데이터베이스(110)에 저장되는 정보는 전처리부(140)에서 생성된 정보일수도 있고, 외부의 클라이언트 장치로부터 수신한 정보일수도 있다.
- [0035] 어셈블리 코드 추출부(120)는 분석 대상 바이너리에서 사용되는 복수의 파일과 각 파일에 포함되는 복수의 함수 목록을 데이터베이스(110)를 검색하여 표시할 수 있다. 즉, 어셈블리 코드 추출부(120)는 분석 대상 바이너리가 입력되면 바이너리의 특성을 분석하여 대응하는 파일, 함수 목록을 데이터베이스(110)를 통해 검색하여 표시할

수 있다.

- [0036] 어셈블리 코드 추출부(120)는 표시된 복수의 함수 목록 중 제1 함수가 선택되면, 제1 함수의 정방향 함수인 제2 함수 또는 제1 함수의 역방향 함수인 제3 함수의 어셈블리 코드를 표시할 수 있다. 구체적으로, 제1 함수가 선택되면 제1 함수를 호출하는 적어도 하나의 정방향 함수와, 제1 함수로부터 호출된 적어도 하나의 역방향 함수를 표시할 수 있다. 설명의 편의를 위하여, 적어도 하나의 정방향 함수가 1개인 경우, 적어도 하나의 역방향 함수가 1개인 경우를 예로 들어 설명하지만, 제1 함수와 직접 연결된 복수의 정방향 함수와 복수의 역방향 함수가 존재할 수 있다.
- [0037] 어셈블리 코드 추출부(120)는 제1 함수의 선택이 해제되고 제2 함수가 선택되면 제1 함수, 제2 함수, 제3 함수의 어셈블리 코드 표시를 해제하고, 제2 함수, 제2 함수의 정방향 함수인 제4 함수 또는 역방향 함수인 제5 함수의 어셈블리 코드를 표시할 수 있다. 여기서, 어셈블리 코드 추출부(120)는 제2 함수가 선택되는 신호를 제1 함수의 선택 해제 신호로 판단할 수 있다. 제2 함수의 정방향 함수만 존재하는 경우에는 제4 함수만 표시하고, 역방향 함수만 존재하는 경우에는 제5 함수만 표시하고, 정방향 함수와 역방향 함수가 모두 존재하는 경우에는 제4 함수, 제5 함수를 모두 표시할 수 있다.
- [0038] 어셈블리 코드 추출부(120)는 복수의 함수 목록 중 특정 함수의 정방향 또는 역방향 콜체인을 표시하고, 콜체인에 포함된 다른 파일의 함수를 선택한 경우, 정적 분석부(130)에서 선택된 다른 파일의 함수를 호출하여 표시하는 방법의 연관된 다중 파일의 실행 흐름을 분석할 수 있다.
- [0039] 정적 분석부(130)는 어셈블리 코드 추출부(120)에서 제1 함수 내지 제5 함수 중 하나에 대응하는 레지스터 값의 추적을 요청하는 경우, 데이터베이스(110)를 통해 대응되는 레지스터 정보를 추출하여 어셈블리 코드 추출부(120)로 제공할 수 있다.
- [0040] 정적 분석부(130)는 특정 레지스터 값이 추적 대상으로 선택되면 추적 대상의 추적 정보와 추적 상태를 데이터베이스(110)에 저장한 후 추적을 시작할 수 있다. 정적 분석부(130)는 정주적이 요청되면 추적 대상을 분석하고, 분기 명령이 있는지, 추적 대상이 존재하는 지, 다른 함수와의 관계 등에 따라 구체적인 추적 방법을 결정할 수 있다. 또한, 정적 분석부(130)는 역주적이 요청되면 추적 대상을 확인하고, 추적 대상의 존재 여부, 다른 함수와의 관계 등에 따라 구체적인 추적 방법을 결정할 수 있다.
- [0041] 전처리부(140)는 분석 대상 바이너리를 분석하여 복수의 파일 각각에 대응하는 프로그램 이름, 함수의 이름, 어셈블리 코드, 분기 정보, 및 레지스터 값을 생성할 수 있다.
- [0042] 상기의 어셈블리 코드 추출부(120), 정적 분석부(130), 및 전처리부(140)는 각각 독립된 하드웨어를 의미할 수도 있고, 하나의 CPU(Computer Processing Unit) 내의 기능별로 구분된 모듈 내지 기능 단위를 의미할 수도 있다.
- [0043] 연관된 다중 파일 정적 분석 장치(100)는 전처리부(140)에서 분석 대상 바이너리를 분석하여 복수의 파일 각각에 대응하는 프로그램 이름, 함수의 이름, 어셈블리 코드, 분기 정보, 및 레지스터 값을 생성할 수 있지만, 전처리부(140)에서 이러한 기능을 수행하지 않고, 클라이언트 장치로부터 상기의 정보들을 수신하여 데이터베이스(110)에 저장할 수 있다. 여기서, 클라이언트 장치는 연관된 다중 파일 정적 분석 장치(100)와는 물리적 또는 개념적으로 분리된 별개의 장치를 의미한다.
- [0045] 도 2는 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치를 포함하는 시스템의 개략적인 구성도이다.
- [0046] 도 2를 참조하면, 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 시스템은 연관된 다중 파일 정적 분석 장치(100), 및 클라이언트 장치(200)로 구성될 수 있다. 또한, 연관된 다중 파일 정적 분석 시스템은 사용자 단말기(300)를 더 구비할 수 있다.
- [0047] 사용자 단말기(300)는 분석할 바이너리 임포트 테이블의 함수가 있는 바이너리를 입력하여 클라이언트 장치(200)에 제공한다.
- [0048] 클라이언트 장치(200)는 상술한 IDA PRO를 통해 바이너리를 분석함으로써 복수의 프로그램 각각의 이름, 복수의 프로그램 각각에 포함된 함수 정보, 복수의 프로그램 각각에 포함된 분기점 정보 및 복수의 프로그램 각각의 어셈블리코드 정보를 생성하여 연관된 다중 파일 정적 분석 장치(100)에 전송한다. 이때, 클라이언트 장치(200)는

디스어셈블러 및 분석기로 구성이 구분될 수 있다.

- [0049] 그 결과, 본 발명의 실시예에 따른 연관된 다중 파일 정적 분석 장치(100)는 수신한 함수 정보, 분기점 정보 및 어셈블리코드 정보를 활용하여 복수의 프로그램에 대해 정적 분석을 수행할 수 있으며, 특히나 복수의 프로그램에서 연속적으로 활용되는 특정 함수에 대해서, 특정 함수 활용되는 프로그램의 콜체인을 분석할 수 있다. 여기서, 파일은 파일 이름, 파일 해쉬, 파일 함수 개수에 대한 정보를 포함하고, 함수는 함수 이름, 함수 해쉬, 함수 타입, 함수 콜 카운트, 함수 라인 등의 정보를 포함한다. 함수 콜은 연관된 함수, 호출명, 호출에 대한 정보 등을 포함하고, 함수 코드는 연관된 함수, 어셈블리 코드, 코드 정보 등을 포함하고, 기록은 추적 정보, 연관된 함수, 코드라인 정보 등의 정보를 포함한다. 추적은 함수 정보, 코드라인 정보, 레지스터, 추적 형태, 코드, 추적 체인, 추적 상태 등의 정보를 포함하며, 클라이언트 장치(200)에서 이러한 정보들은 추출되어 연관된 다중 파일 정적 분석 장치(100)의 데이터베이스(100)에 저장될 수 있다.
- [0051] 도 3은 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치의 분석 결과를 설명하기 위한 도면이고, 도 4는 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치의 콜체인을 설명하기 위한 개념도이다.
- [0052] 도 3을 참조하면, 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치는 분석 대상 바이너리에서 사용되는 복수의 파일 목록을 표시하고, 특정 파일을 선택하면 특정 파일에 포함된 복수의 함수 목록을 분석 대상 선택 화면에 표시할 수 있다.
- [0053] 또한, 특정 함수를 선택하면 특정 함수(Caller)의 정방향 함수 또는 역방향 함수(Callee)를 표시하고, 각각의 어셈블리 코드를 분석 대상 출력 화면에 표시한다. 분석 대상 선택 화면과 분석 대상 출력 화면은 디스플레이부의 구분된 화면 영역일 수 있다. 또한, 디스어셈블리 코드 출력 화면과 추적 대상 선택 화면도 분석 대상 선택 화면과 분석 대상 출력 화면과 구분할 수 있다.
- [0054] 도 4를 참조하면, 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치는 분석의 대상이 되는 특정 함수가 사용된 곳을 모든 바이너리에서 추출할 수 있으며, 이를 토대로 정방향 또는 역방향 분석을 수행할 수 있다.
- [0055] 더 나아가, 본 발명의 실시예에 따른 연관된 다중 파일 정적 분석 장치(100)는 각각의 함수의 어셈블리 코드를 추출하여 분석을 진행할 수 있으며, 이와 함께 서버에서 백 그라운드로 동작하여, 웹에서 해당 레지스터 값을 추적 요청 하면 DB에 저장된 데이터를 토대로 정방향 추적 또는 역방향 추적기능을 제공할 수 있다.
- [0056] 도시한 바와 같이, 함수 추적은 함수 간의 호출 관계에 따라 추적을 하게 되며, 파일 A의 함수가 같은 파일 A 내에 존재하는 함수 2, 3만 호출하는 것이 아니라, 파일 B에 있는 함수 5까지 추적해서 분석할 수 있다. 마찬가지로, 역으로 파일 A의 함수 1을 호출하는 것은 같은 파일 A의 함수 7뿐만 아니라 파일 C의 함수 8이 호출하는 것도 알고 분석을 이어갈 수 있다. 이때, 정방향 추적의 알고리즘은 도 5 내지 도 6에 도시된 바와 같으며, 역방향 추적의 알고리즘은 도 7 내지 도 8에 도시된 바와 같다.
- [0058] 도 5 내지 도 6은 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치의 정방향 분석을 설명하기 위한 도면이다.
- [0059] 도 5 내지 도 6을 참조하면, 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치의 정방향 분석은 정 추적 요청을 수신하는 경우 시작된다. 정추적은 향후 실행되는 방향으로 추적을 진행하는 것을 의미한다.
- [0060] 먼저, 정추적이 요청되면(A) 추적 대상을 분석하여(B) 분기 명령이 있는지 판단하고(C), 분기 명령이 있다고 판단되면(Yes) 추적 대상이 존재하는지 판단하고(D), 추적 대상이 존재하면(Yes) 역추적 결과에 따라(E) 추적 완료(J) 또는 메모리 추적을 수행(I)할 수 있다. 또한, 추적 대상을 분석하여(B) 추적 대상이 존재하지 않으면(No) 다음 명령어를 확인하여 추적 대상 분석 과정(B)부터 다시 수행할 수 있다. 또한, 분기 명령이 없다고 판단되면(C_No) 다른 함수에게 영향을 주는지(F) 또는 다른 함수로부터 영향을 받는지(G)에 따라 추적 레스트리 값을 변경하여(H) 다음 명령어를 확인한 후 추적 대상 분석 과정(B)을 다시 수행하거나, 또는 Crash로 판단(J)할 수 있다.
- [0061] 도 6의 구체적인 실시 예에서, 정추적이 요청되면 추적 대상 "LDR X2, [X8, #0x50]"을 분석하고, 추적 대상이 현재 인스트럭션이고, X8이며, 분기 명령은 'No', 영향을 주는지는 'Yes' 영향을 주는 레지스터는 X2이므로 추

적 레지스터를 변경한다. 다음으로, 추적 대상 "MOV X0, X20"을 분석하고, 추적 대상은 현재 인스트럭션 X2이고, 분기 명령은 'No', 영향을 주는지는 'No', 영향을 받는가는 'No'이므로 다음 추적 대상을 분석한다. 이러한 과정을 거쳐, 정방향으로 추적을 수행하면, 마지막 인스트럭션 "MOV X2, #0x0"과 같이 레지스터 X2를 '0'으로 설정하는 경우면 분기 명령 'No', 영향을 주는가 'No', 영향을 받는가 'Yes'로 분석이 완료된다. 결과적으로, X8을 추적하였으나, X2에는 영향을 주되 프로그램에 큰 영향은 없는 것으로 추적은 완료된다.

[0063] 도 7 내지 도 8은 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치의 역방향 분석을 설명하기 위한 도면이다.

[0064] 도 7 내지 도 8을 참조하면, 본 발명의 일 실시예에 따른 연관된 다중 파일 정적 분석 장치의 역방향 분석은 역추적 요청을 수신하는 경우 시작된다. 역추적은 분석 대상으로부터 프로그램이 실행되어 왔던 역방향을 추적하는 것을 의미한다.

[0065] 역추적이 요청되면(A) 추적 대상을 확인하고(B), 추적 대상이 존재하는지 판단하고(C), 추적 대상이 존재하면(Yes) 다른 함수로부터 영향을 받는지(D)에 따라 레지스터 값을 확인하는 단계(E)로 넘어가거나 또는 참조 주소를 확인(H)할 수 있다. 추적 대상이 존재하지 않으면(C_No) 참조 주소를 확인(H)할 수 있다.

[0066] 다음으로, 참조 주소를 확인하여(H) 내부 주소인지(I) 외부 파일인지 판단하고(J), 참조 주소가 내부 주소(Yes) 또는 외부 파일인 경우(Yes), 추적 대상을 수정한 후(K) 추적 대상 확인 단계(B)부터 다시 수행한다. 또한, 참조 주소가 내부 주소도 아니고(No) 외부 파일도 아닌 경우(N), Crash로 판단할 수 있다(L).

[0067] 레지스터 값을 확인하여(E), 레지스터 값이 존재하면(Yes) 명령어에 따른 추적 대상을 추가 또는 변경하고(F), 레지스터 값이 존재하지 않으면(No) 메모리 추적을 수행할 수 있다(G).

[0068] 도 8의 구체적인 실시 예에서, 역추적이 요청되면 추적 결과를 아래부터 위로 확인해야 하고, 가장 아래 라인을 보면 추적 대상은 이전 인스트럭션 X8이고, 상위 라인을 분석하면, 추적 대상은 현재 인스트럭션 X8이고, 추적 대상이 존재하고(Yes), 영향을 받는 함수가 있고(Yes), 레지스터가 존재하고(Yes), 추적 대상은 이전 인스트럭션 X20임을 확인할 수 있다. 이러한 방법으로, 역방향 추적을 수행할 수 있다.

[0069] 여기서, 이전 인스트럭션에서 X19와 같은 레지스터를 추적하는 경우, 도 8의 과정으로 추적된다. 이전 인스트럭션 X19값+0x5C0를 추적하는 경우, 해당 값은 특정 주소이므로 레지스터 값으로부터 온 것이 아니기 때문에 역추적을 종료하고 메모리 추적을 진행하게 된다.

[0070] 추적이 완료된 경우, X8을 추적한 결과 X8에 영향을 준 것은 X1으로 결정되고 추적은 완료된다. 결과적으로, 추적 대상 레지스터 X8을 중심으로 정추적과 역추적을 진행하게 된다.

[0071] 따라서, 본 발명에 의하면, 기계어 수준에서도 서로 연관된 다중 파일의 함수의 흐름을 자동으로 표시하여 정적 분석이 가능하도록 제공하므로 연관된 파일들 사이의 정적 분석도 용이하게 수행할 수 있다.

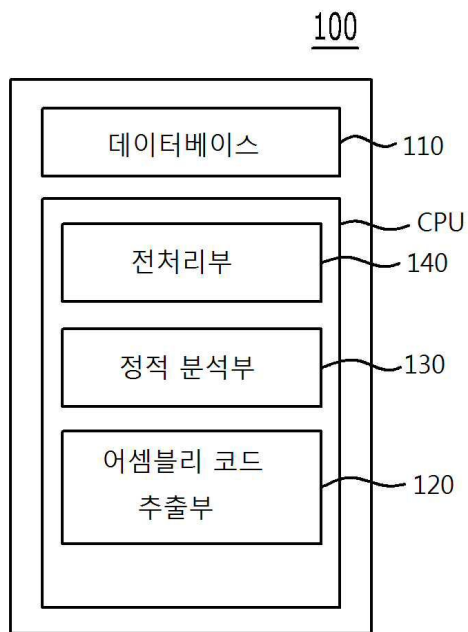
[0073] 이상의 설명은 본 발명의 기술 사상을 예시적으로 설명한 것에 불과한 것으로, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 사람이라면 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 다양한 수정 및 변형이 가능할 것이다. 따라서, 본 발명에 개시된 실시예들은 본 발명의 기술 사상을 한정하기 위한 것이 아니라 설명하기 위한 것이고, 이러한 실시예에 의하여 본 발명의 기술 사상의 범위가 한정되는 것은 아니다. 본 발명의 보호 범위는 아래의 청구범위에 의하여 해석되어야 하며, 그와 동등한 범위 내에 있는 모든 기술 사상은 본 발명의 권리범위에 포함되는 것으로 해석되어야 할 것이다.

부호의 설명

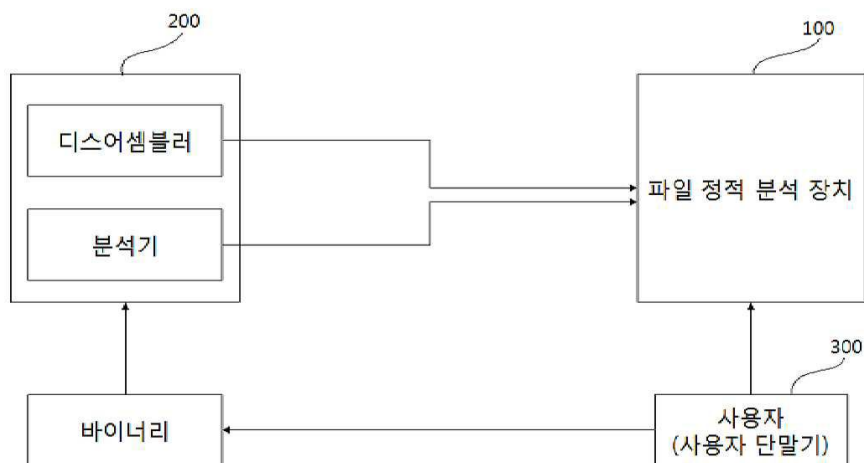
[0074] 100: 연관된 다중 파일 정적 분석 장치 110: 데이터베이스
120: 어셈블리 코드 추출부 130: 정적 분석부
140: 전처리부

도면

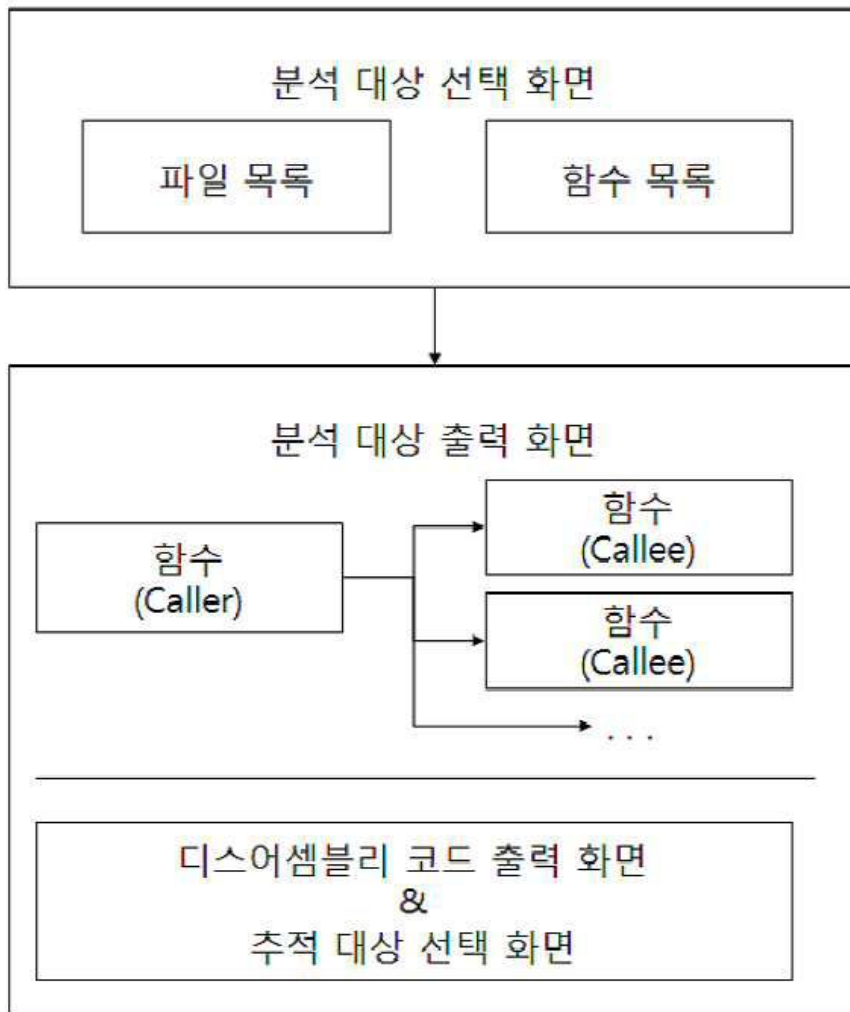
도면1



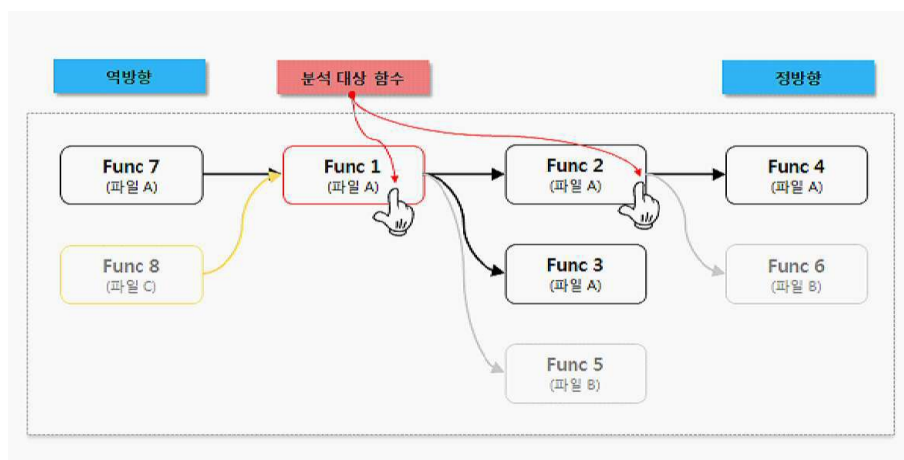
도면2



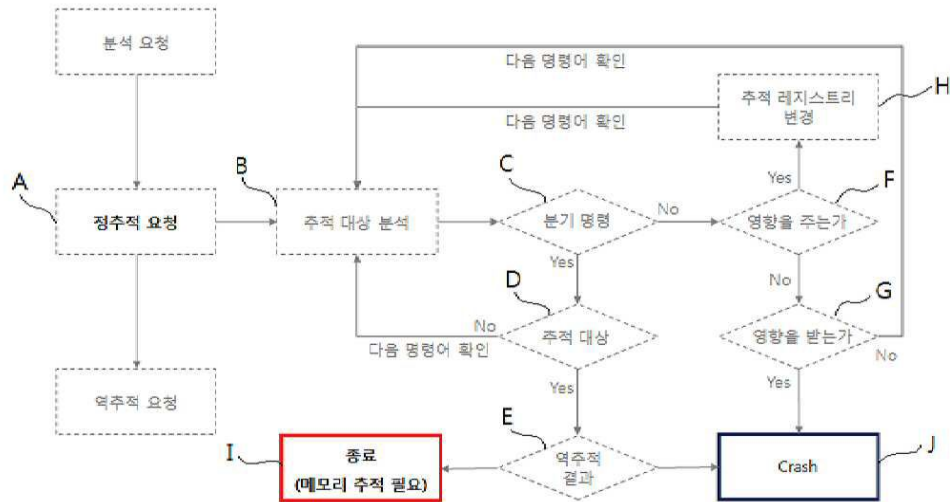
도면3



도면4



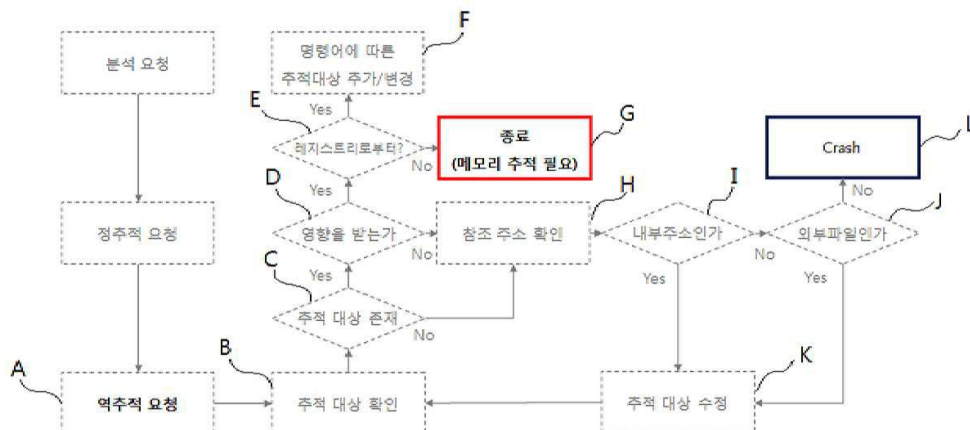
도면5



도면6

LDR X8, [X20]	
LDR X2, [X8, #0x50]	추적 대상은 현재 인스트럭션이고, X8임 분기 명령 : No → 영향 주는가 : YES 영향을 주는 레지스터는 X2 뿐임. 추적 레지스터 변경.
MOV X0, X20	추적 대상은 현재 인스트럭션이고, X2임 분기 명령 : NO → 영향 주는가 : NO → 영향 받는가 : NO
MOV X1, X19	추적 대상은 현재 인스트럭션이고, X2임 분기 명령 : NO → 영향 주는가 : NO → 영향 받는가 : NO
LDP X29, X30, [SP, #0x10]	추적 대상은 현재 인스트럭션이고, X2임 분기 명령 : NO → 영향 주는가 : NO → 영향 받는가 : NO
BR X2	추적 대상은 현재 인스트럭션이고, X2임 분기 명령 : YES → 추적 대상 : X2가 포함됨, YES → 역추적 결과에 따라 추적을 완료하거나, 종료시킴

도면7



도면8

...	...
LDR X1, [X19,#0x5C0]	추적 대상은 현재 인스트럭션이고, X1임 추적 대상 존재 : YES → 참조 주소 확인 : X19 값 + 0x5C0 → 내부 주소 상태 : YES → 추적 대상은 이전 인스트럭션, X19, X19 값+0x5C0임
MOV X20, X1	추적 대상은 현재 인스트럭션이고, X20임 추적 대상 존재 : YES → 영향을 받는가 : YES → 레지스터 ? : YES → 추적 대상은 이전 인스트럭션, X1임
LDR X8, [X20]	추적 대상은 현재 인스트럭션이고, X8임 추적 대상 존재 : YES → 영향을 받는가 : YES → 레지스터 ? : YES → 추적 대상은 이전 인스트럭션, X20임
LDR X2, [X8,#0x50]	추적 대상은 이전 인스트럭션이고, X8임