관 인 생 략

출 원 번 호 통 지 서

출 원 일 자	2014.12.05
특 기 사 항	심사청구(무) 공개신청(무) 참조번호(14)
출 원 번 호	10-2014-0174293 (접수번호 1-1-2014-1188000-63)
출원인 명칭	삼성전자주식회사(1-1998-104271-3) 외 1명
대리인 성명	리앤목 특허법인(9-2005-100002-8)
발 명 자 성 명	박찬익 신재복 오영섭
발명의 명칭	응용 프로그램의 자원을 보호하는 방법 및 장치

특 허 청 장

<< 안내 >>

1. 귀하의 출원은 위와 같이 정상적으로 접수되었으며, 이후의 심사 진행상황은 출원번호를 통해 확인하실 수 있습니다.

2. 출원에 따른 수수료는 접수일로부터 다음날까지 동봉된 납입영수증에 성명, 납부자번호 등을 기재하여 가까운 우체국 또는 은행에 납부하여야 합니다.
※ 납부자번호 : 0131(기관코드) + 접수번호

3. 귀하의 주소, 연락처 등의 변경사항이 있을 경우, 즉시 [출원인코드 정보변경(경정), 정정신고서]를 제출하여야 출원 이후의 각종 통지서를 정상적으로 받을 수 있습니다.
※ 특허로(patent.go.kr) 접속 > 민원서식다운로드 > 특허법 시행규칙 별지 제5호 서식

4. 특허(실용신안등록)출원은 명세서 또는 도면의 보정이 필요한 경우, 등록결정 이전 또는 의견서 제출기간 이내에 출원서에 최초로 첨부된 명세서 또는 도면에 기재된 사항의 범위 안에서 보정할 수 있습니다.

5. 외국으로 출원하고자 하는 경우 PCT 제도(특허·실용신안)나 마드리드 제도(상표)를 이용할 수 있습니다. 국내출원일을 외국에서 인정받고자 하는 경우에는 국내출원일로부터 일정한 기간 내에 외국에 출원하여야 우선권을 인정받을 수 있습니다.
※ 제도 안내 : http://www.kipo.go.kr-특허마당-PCT/마드리드
※ 우선권 인정기간 : 특허·실용신안은 12개월, 상표·디자인은 6개월 이내
※ 미국특허상표청의 선출원을 기초로 우리나라에 우선권주장출원 시, 선출원이 미공개상태이면, 우선일로부터 16개월 이내에 미국특허상표청에 [전자적교환허가서(PTO/SB/39)를 제출하거나 우리나라에 우선권 증명서류를 제출하여야 합니다.

6. 본 출원사실을 외부에 표시하고자 하는 경우에는 아래와 같이 하여야 하며, 이를 위반할 경우 관련법령에 따라 처벌을 받을 수 있습니다.
※ 특허출원 10-2010-0000000, 상표등록출원 40-2010-0000000

7. 기타 심사 절차에 관한 사항은 동봉된 안내서를 참조하시기 바랍니다.

2014-12-05

【서지사항】

【서류명】 특허출원서

【참조번호】 14

【출원구분】 특허출원

【출원인】

【명칭】 삼성전자주식회사

【출원인코드】 1-1998-104271-3

【출원인】

【명칭】 포항공과대학교 산학협력단

【출원인코드】 2-2004-043336-1

【대리인】

【명칭】 리앤목 특허법인

【대리인코드】 9-2005-100002-8

【지정된변리사】 이영필,이해영,이호근,하수영,박이주,김지혜

【포괄위임등록번호】 2005-049725-9

【포괄위임등록번호】 2005-049722-7

【발명의 국문명칭】 응용 프로그램의 자원을 보호하는 방법 및 장치

【발명의 영문명칭】 Method and apparatus for protecting resource of application program

【발명자】

【성명】 박찬익

【성명의 영문표기】 PARK, Chan Ik

【주민등록번호】 610301-1XXXXXX

【우편번호】 790-784

【주소】 경상북도 포항시 남구 청암로 77, 정보통신연구소 311호 (지곡동)

【국적】 KR

【발명자】

【성명】 신재복

【성명의 영문표기】 SHIN, Jae Bok

【주민등록번호】 861118-1XXXXXX

【우편번호】 790-784

【주소】 경상북도 포항시 남구 청암로 77, 정보통신연구소 423호 (지곡동)

【국적】 KR

【발명자】

【성명】 오영섭

【성명의 영문표기】 OH, Young Sup

【주민등록번호】 880417-1XXXXXX

【우편번호】 790-784

【주소】 경상북도 포항시 남구 청암로 77, 정보통신연구소 423호 (지곡동)

【국적】 KR

2014-12-05

【취지】 위와 같이 특허청장에게 제출합니다.

대리인 리앤목 특허법인 (서명 또는 인)

【수수료】

【출원료】	0 면	46,000 원
【가산출원료】	50 면	0 원
【우선권주장료】	0 건	0 원
【심사청구료】	0 항	0 원
【합계】	46,000 원	

【명세서】

【발명의 명칭】

응용 프로그램의 자원을 보호하는 방법 및 장치{Method and apparatus for protecting resource of application program}

【기술분야】

【0001】 본 발명은 응용 프로그램의 자원을 운영 체제 또는 다른 프로그램으로부터 보호하는 방법 및 장치에 대한 것이다.

【발명의 배경이 되는 기술】

【0002】 운영 체제는 하드웨어와 응용 프로그램 간의 인터페이스 역할을 하면서, CPU, 주기억 장치, 입출력 장치 등의 컴퓨터 자원을 관리한다. 운영 체제는 하드웨어를 제어하고 컴퓨터 자원을 관리하기 위하여 응용 프로그램보다 높은 권한을 가지고 동작할 수 있다.

【0003】 그러나, 운영 체제의 보안 취약성을 이용하여, 악성 소프트웨어가 운영 체제에 침입하게 되면, 악성 소프트웨어는 운영 체제의 최상위 권한을 이용하여 응용 프로그램 및 컴퓨터의 자원에 침입할 수 있다.

【0004】 운영 체제의 기능이 확대되고 발전함에 따라 운영 체제의 크기가 커지고, 또한, 그에 따라 운영 체제의 보안 취약성이 증가하고 있다. 따라서, 해킹된 운영 체제가 응용 프로그램의 자원에 대해 접근하려고 하는 경우, 응용 프로그램의 자원을 보호할 수 있는 방법이 필요하다.

【발명의 내용】

【해결하고자 하는 과제】

【0005】 본 발명은 응용 프로그램의 자원을 보호하는 방법 및 장치에 관한 것으로, 구체적으로, 응용 프로그램의 자원에 대한 운영 체제 또는 다른 프로그램의 접근으로부터 응용 프로그램의 자원을 보호하는 방법 및 장치에 관한 것이다.

【과제의 해결 수단】

【0006】 일 실시 예에 의한 디바이스가 응용 프로그램의 자원을 보호하기 위한 방법은, 보호 대상으로 설정된 응용 프로그램이 실행되는 동안, 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지 여부를 판단하는 단계; 및 상기 판단 결과에 따라, 상기 응용 프로그램의 자원에 대한 접근 권한을 제어하는 단계를 포함한다.

【0007】 더하여, 상기 자원에 대한 접근 권한을 제어하는 단계는 상기 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인 경우, 상기 응용 프로그램의 자원으로 할당된 메모리 페이지에 대한 접근 권한 정보 및 상기 응용 프로그램의 레지스터 값을 백업하는 단계; 상기 메모리 페이지에 대한 접근 권한을 재설정하고, 상기 레지스터 값을 삭제하는 단계를 포함한다.

【0008】 더하여, 상기 메모리 페이지에 대한 접근 권한 정보 및 상기 레지스터 값 중 적어도 하나가 백업되어 있는 경우, 상기 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 없는 상태로 전환되면, 상기 백업된 값을 이용하여 상기 메

모리 페이지에 대한 접근 권한 정보 및 상기 레지스터 값 중 적어도 하나를 복구하는 단계를 더 포함한다.

【0009】더하여, 상기 판단하는 단계는 상기 응용 프로그램의 실행 모드가 커널 모드인지 유저 모드인지 여부에 기초하여, 상기 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지 여부를 판단하는 단계를 포함한다.

【0010】더하여, 상기 운영 체제가 시스템 호출을 처리하기 위하여, 상기 응용 프로그램의 자원에 대한 접근을 시도하는 단계; 상기 시스템 호출에 기초하여, 상기 자원에 대한 접근 권한을 제어하는 단계를 더 포함한다.

【0011】더하여, 상기 자원에 대한 접근 권한을 제어하는 단계는 상기 응용 프로그램의 자원으로 할당된 메모리 페이지에 저장된 데이터를 암호화할 지 여부를 결정하는 단계; 상기 결정된 결과에 따라, 상기 데이터를 암호화한 후, 상기 운영 체제의 상기 메모리 페이지에 대한 접근을 허용하거나, 상기 데이터를 상기 운영 체제에 제공하는 단계를 더 포함한다.

【0012】더하여, 상기 운영 체제가 시스템 호출을 처리하기 위해, 상기 응용 프로그램의 자원으로 할당된 메모리 페이지에 대한 접근을 시도함을 감지하는 단계; 상기 시스템 호출에 의해 처리될 데이터의 원본 데이터를 획득하는 단계; 상기 운영 체제에 의해 상기 시스템 호출에 대한 처리가 완료되면, 상기 시스템 호출에 따라 처리된 데이터를 획득하는 단계; 상기 원본 데이터의 해시값과 상기 시스템 호출에 따라 처리된 데이터의 해시값을 비교함으로써, 상기 시스템 호출에 따라 처리된 데이터의 무결성을 검증하는 단계를 더 포함한다.

【0013】더하여, 상기 응용 프로그램이 보호 대상으로 설정되면, 상기 응용 프로그램의 자원이 할당된 메모리 페이지는 보호 대상으로 등록된다.

【0014】더하여, 상기 운영 체제에 의해 상기 응용 프로그램의 자원으로 새로운 메모리 페이지가 할당됨을 감지하는 단계; 상기 할당된 새로운 메모리 페이지가 상기 보호 대상으로 등록된 메모리 페이지와 동일하다고 판단되면, 상기 새로운 메모리 페이지에 대한 할당을 해제하는 단계를 더 포함한다.

【0015】더하여, 상기 운영 체제가 상기 응용 프로그램의 페이지 테이블을 수정하는 경우, 상기 수정된 페이지 테이블에 따라 상기 응용 프로그램의 자원으로 할당된 메모리 페이지를 보호 대상으로 등록하거나 보호 대상에서 해제하는 단계를 더 포함한다.

【0016】더하여, 상기 운영 체제 대신 다른 응용 프로그램이 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지 여부를 판단하고, 상기 판단 결과에 따라 상기 응용 프로그램의 자원에 대한 접근 권한을 수행한다.

【0017】일 실시 예에 의한 응용 프로그램의 자원을 보호하기 위한 디바이스는 보호 대상으로 설정되어 실행 중인 응용 프로그램; 상기 응용 프로그램에서 발생된 시스템 호출을 처리하는 운영 체제; 상기 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지 여부를 판단하고, 상기 판단 결과에 따라, 상기 응용 프로그램의 자원에 대한 접근 권한을 제어하는 응용 프로그램 보호부; 상기 응용 프로그램의 자원을 포함하는 하드웨어를 포함한다.

【0018】일 실시 예에 의한 디바이스가 응용 프로그램의 자원을 보호하기 위한 방법을 구현하기 위한 프로그램이 기록된 컴퓨터로 판독 가능한 기록 매체는 보호 대상으로 설정된 응용 프로그램이 실행되는 동안, 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지 여부를 판단하는 단계; 및 상기 판단 결과에 따라, 상기 응용 프로그램의 자원에 대한 접근 권한을 제어하는 단계를 포함한다.

【0019】일 실시 예에 의한 하드웨어와 결합되어 디바이스가 응용 프로그램의 자원을 보호하기 위한 방법을 실행시키는 컴퓨터 프로그램은 보호 대상으로 설정된 응용 프로그램이 실행되는 동안, 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지 여부를 판단하는 단계; 및 상기 판단 결과에 따라, 상기 응용 프로그램의 자원에 대한 접근 권한을 제어하는 단계를 포함한다.

【도면의 간단한 설명】

【0020】도 1은 일 실시 예에 의한 응용 프로그램 보호부를 포함하는 디바이스의 내부 구성을 나타낸 블록도이다.

도 2는 일 실시 예에 의한 가상화 기술을 기반으로 구동되는 응용 프로그램 보호부를 포함하는 디바이스의 내부 구성을 나타낸 블록도이다.

도 3은 일 실시 예에 의한 응용 프로그램의 자원에 대한 접근 권한을 제어하는 방법을 나타낸 순서도이다.

도 4는 일 실시 예에 의한 응용 프로그램의 자원에 대한 접근 권한 및 레지

스터 값을 제어하는 방법을 나타낸 순서도이다.

도 5는 일 실시 예에 의한 운영 체제가 응용 프로그램의 자원에 접근하여 시스템 호출을 처리하는 방법을 나타낸 순서도이다.

【발명을 실시하기 위한 구체적인 내용】

【0021】 다만, 하기의 설명 및 첨부된 도면에서 본 발명의 요지를 흐릴 수 있는 공지 기능 또는 구성에 대한 상세한 설명은 생략한다. 또한, 도면 전체에 걸쳐 동일한 구성 요소들은 가능한 한 동일한 도면 부호로 나타내고 있음에 유의하여야 한다.

【0022】 이하에서 설명되는 본 명세서 및 청구범위에 사용된 용어나 단어는 통상적이거나 사전적인 의미로 한정해서 해석되어서는 아니 되며, 발명자는 그 자신의 발명을 가장 최선의 방법으로 설명하기 위한 용어로 적절하게 정의할 수 있다는 원칙에 입각하여 본 발명의 기술적 사상에 부합하는 의미와 개념으로 해석되어야만 한다. 따라서 본 명세서에 기재된 실시 예와 도면에 도시된 구성은 본 발명의 가장 바람직한 일 실시 예에 불과할 뿐이고, 본 발명의 기술적 사상을 모두 대변하는 것은 아니므로, 본 출원시점에 있어서 이들을 대체할 수 있는 다양한 균등물과 변형 예들이 있을 수 있음을 이해하여야 한다.

【0023】 첨부 도면에 있어서 일부 구성요소는 과장되거나 생략되거나 또는 개략적으로 도시되었으며, 각 구성요소의 크기는 실제 크기를 전적으로 반영하는 것이 아니다. 본 발명은 첨부한 도면에 그려진 상대적인 크기나 간격에 의해 제한

되어지지 않는다.

【0024】 명세서 전체에서 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있음을 의미한다. 또한, 명세서에서 사용되는 "부"라는 용어는 소프트웨어, FPGA 또는 ASIC과 같은 하드웨어 구성요소를 의미하며, "부"는 어떤 역할들을 수행한다. 그렇지만 "부"는 소프트웨어 또는 하드웨어에 한정되는 의미는 아니다. "부"는 어드레싱할 수 있는 저장 매체에 있도록 구성될 수도 있고 하나 또는 그 이상의 프로세서들을 재생시키도록 구성될 수도 있다. 따라서, 일 예로서 "부"는 소프트웨어 구성요소들, 객체지향 소프트웨어 구성요소들, 클래스 구성요소들 및 태스크 구성요소들과 같은 구성요소들과, 프로세스들, 함수들, 속성들, 프로시저들, 서브루틴들, 프로그램 코드의 세그먼트들, 드라이버들, 펌웨어, 마이크로 코드, 회로, 데이터, 데이터베이스, 데이터 구조들, 테이블들, 어레이들 및 변수들을 포함한다. 구성요소들과 "부"들 안에서 제공되는 기능은 더 작은 수의 구성요소들 및 "부"들로 결합되거나 추가적인 구성요소들과 "부"들로 더 분리될 수 있다.

【0025】 아래에서는 첨부한 도면을 참고하여 본 발명의 실시예에 대하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유

사한 부분에 대해서는 유사한 도면 부호를 붙였다.

【0026】 이하, 첨부된 도면을 참조하여 본 발명의 바람직한 실시예를 설명한다.

【0027】 도 1은 일 실시 예에 의한 응용 프로그램 보호부를 포함하는 디바이스의 내부 구성을 나타낸 블록도이다.

【0028】 도 1을 참조하면, 디바이스(100)는 응용 프로그램(110), 운영 체제(120), 응용 프로그램 보호부(130) 및 하드웨어(140)를 포함하여 구성될 수 있다. 도면들과 후술되어 있는 실시예에서, 디바이스(100)에 포함되어 있는 개개의 구성요소는 물리적 형태 또는 논리적 형태로 분산되어 배치될 수도 있고, 통합될 수도 있다.

【0029】 일 실시 예에 의한 디바이스(100)는 예를 들어, 휴대폰, 태블릿 PC, PDA, MP3 플레이어, 키오스크, 전자 액자, 네비게이션 장치, 디지털 TV, 손목 시계(Wrist watch), 스마트 글라스(smart glass)와 같은 웨어러블 기기(Wearable device) 등의 다양한 유형의 장치일 수 있다.

【0030】 응용 프로그램(110)은 특정한 작업을 처리하기 위해 디바이스(100)에서 실행될 수 있는 프로그램이다. 응용 프로그램(110)은 운영 체제(120)에 의해 할당된 하드웨어 자원을 기반으로 동작할 수 있다. 또한, 응용 프로그램(110)은 코드 영역, 텍스트 영역, 데이터 파일, 라이브러리 등을 포함하는 다양한 컴포넌트들로 구성될 수 있다. 예를 들면, 응용 프로그램(110)은 운영 체제(120)를 기반으로

수행되는 소프트웨어, 프로세스, 쓰레드(thread), 가상 머신 등일 수 있으나, 이에 한정되는 것은 아니다.

【0031】운영 체제(120)는 응용 프로그램(110)의 실행에 필요한 자원을 할당하거나 해제할 수 있다. 또한, 운영 체제(120)는 복수 개의 응용 프로그램(110)들이 원활히 동작할 수 있도록 스케줄링(scheduling)을 수행함으로써, 복수 개의 응용 프로그램(110)들이 동시에 동작할 수 있다. 운영 체제(120)는 응용 프로그램(110)보다 높은 권한을 가지고 하드웨어 및 하나 이상의 응용 프로그램(110)을 제어할 수 있다.

【0032】응용 프로그램(110)은 운영 체제(120)에 의해 할당된 하드웨어 영역, 예를 들면, 응용 프로그램(110)의 자원으로 할당된 일부 메모리 영역에만 직접 접근할 수 있고, 이외 자신에게 할당되지 않은 하드웨어 영역에는 직접 접근할 수 없을 수 있다. 반면에, 운영 체제(120)는 응용 프로그램(110)보다 높은 권한을 가지고 있으므로, 응용 프로그램(110)의 자원으로 할당되지 않은 영역에도 접근할 수 있다. 따라서, 응용 프로그램(110)은 운영 체제(120)에 의해 실행 가능한 작업을 운영 체제(120)에 요청하기 위한 시스템 호출(system call)을 이용하여, 운영 체제(120)를 통해 자신에게 할당되지 않은 하드웨어 영역에 접근할 수 있다. 응용 프로그램(110)은 운영 체제(120)를 통해 자신에게 할당되지 않은 하드웨어 영역에 접근하여, 데이터를 획득하거나, 자신의 자원으로 새롭게 할당할 수 있다.

【0033】그러나, 응용 프로그램(110)에서 발생된 시스템 호출을 처리하는 운영 체제(120)는 응용 프로그램(110)의 자원으로 할당되지 않은 하드웨어 영역뿐만

아니라 응용 프로그램(110)의 자원으로 할당된 하드웨어 영역에도 접근할 수 있다. 따라서, 운영 체제(120)가 악성 소프트웨어에 의해 감염되어 있는 경우, 시스템 호출에 따라 운영 체제(120)가 응용 프로그램(110)의 자원으로 할당된 하드웨어 영역에 접근하게 되면, 응용 프로그램(110)의 자원에 존재하는 데이터가 유출되거나 위변조될 수 있다.

【0034】응용 프로그램 보호부(130)는 디바이스(100)에서 실행될 수 있는 응용 프로그램(110)의 자원을 보호할 수 있다. 응용 프로그램(110)의 자원은 디바이스(100)의 리소스(예를 들면, 메모리, 저장 장치, CPU 등) 중에서 응용 프로그램(110)에게 할당된 영역을 의미할 수 있다. 응용 프로그램 보호부(130)는 보호 대상으로 설정된 응용 프로그램(110)의 자원에 대한 접근 권한을 제어함으로써 운영 체제(120)로부터 응용 프로그램(110)의 자원을 보호할 수 있다. 예를 들면, 응용 프로그램 보호부(130)는 보호 대상으로 설정된 응용 프로그램(110)의 자원에 대한 접근 권한 정보를 설정함으로써 자원에 대한 운영 체제(120)의 접근을 차단하거나 부분적으로 허용할 수 있다.

【0035】응용 프로그램 보호부(130)는 운영 체제(120)에 한하지 않고, 다른 응용 프로그램 또는 외부 장치로부터 보호 대상으로 설정된 응용 프로그램(110)의 자원을 보호할 수 있다. 따라서, 응용 프로그램 보호부(130)는 다른 응용 프로그램 또는 외부 장치로부터 응용 프로그램(110)의 자원에 접근할 수 있는 상태이면, 응용 프로그램(110)의 자원에 대한 접근 권한 정보를 설정할 수 있다. 이로써, 응용 프로그램 보호부(130)는 자원에 대한 다른 응용 프로그램 또는 외부 장치의 접근을

차단하거나 부분적으로 허용할 수 있다.

【0036】 응용 프로그램 보호부(130)는 운영 체제(120)와 독립적으로 구성되고, 독립적으로 동작함으로써, 악성 소프트웨어에 의해 해킹될 가능성이 있는 운영 체제(120)로부터 응용 프로그램(110)의 자원을 보호할 수 있다. 응용 프로그램 보호부(130)는 Intel VT-x, AMD-V, ARM TrustZone 등을 이용하여 운영 체제(120)와 독립적으로 동작할 수 있다. 예를 들면, 응용 프로그램 보호부(130)는 ARM TrustZone의 안전 영역(secure world)에서 운영 체제(120)와 독립적으로 동작할 수 있다.

【0037】 구체적으로, 운영 체제(120)가 보호 대상으로 설정된 응용 프로그램(110)의 자원에 접근할 수 있는 상태가 되면, 응용 프로그램 보호부(130)는 보호 대상으로 설정된 응용 프로그램(110)의 자원에 대한 접근 권한 정보를 재설정할 수 있다. 예를 들면, 응용 프로그램 보호부(130)는 운영 체제(120)가 접근할 수 없도록 자원에 대한 접근 권한 정보를 재설정할 수 있다. 더하여, 운영 체제(120)에 의해 응용 프로그램(110)이 사용하던 레지스터 값이 유출되거나 위변조되는 것을 방지하기 위하여, 응용 프로그램 보호부(130)는 응용 프로그램(110)이 사용하던 레지스터 값을 삭제할 수 있다.

【0038】 하드웨어(140)는 운영 체제(120) 및 응용 프로그램(110)이 실행되기 위해 필요한 메모리, 저장 장치, CPU(central processing unit), 레지스터, 입출력 기기 등을 포함할 수 있다. 이에 한하지 않고, 하드웨어(140)는 운영 체제(120) 및 응용 프로그램(110)의 실행을 위한 구성 요소를 더 포함할 수 있다.

【0039】 운영 체제(120)는 하드웨어(140)에 포함된 구성 요소 중 일부를 응용 프로그램(110)의 자원으로 할당할 수 있다. 응용 프로그램(110)은 자신의 자원으로 할당된 하드웨어(140)의 일부 영역을 이용하여 필요한 작업을 수행할 수 있다.

【0040】 이하 도 2를 참조하여, 응용 프로그램 보호부에서 응용 프로그램의 자원을 보호하는 구체적인 방법에 대해 설명하기로 한다.

【0041】 도 2는 일 실시 예에 의한 가상화 기술을 기반으로 구동되는 응용 프로그램 보호부를 포함하는 디바이스의 내부 구성을 나타낸 블록도이다.

【0042】 도 2를 참조하면, 디바이스(100)는 응용 프로그램(110), 운영 체제(120), 응용 프로그램 보호부(130)가 구동되는 가상화 머신 모니터(150), 하드웨어(140)를 포함하여 구성될 수 있다. 도면들과 후술되어 있는 실시예에서, 디바이스(100)에 포함되어 있는 개개의 구성 요소는 물리적 형태 또는 논리적 형태로 분산되어 배치될 수도 있고, 통합될 수도 있다.

【0043】 운영 체제(120)는 페이지 테이블(121)과 시스템 호출 처리부(122)를 포함할 수 있다.

【0044】 페이지 테이블(121)은 실행 중인 응용 프로그램(110)의 자원으로 할당된 메모리 페이지에 관한 정보를 포함한다. 메모리 페이지란, 미리 설정된 크기로 나뉘어진 메모리 영역의 각 부분을 의미할 수 있다. 응용 프로그램(110)의 자원으로 할당된 메모리 페이지는 응용 프로그램(110)의 실행 중 발생된 데이터를 저장

할 수 있다. 운영 체제(120)는 메모리 페이지들을 응용 프로그램(110)의 자원으로 할당하거나 해제함에 따라 페이지 테이블(121)에 포함된 메모리 페이지에 관한 정보를 갱신할 수 있다. 페이지 테이블(121)은 각 메모리 페이지에 관한 식별 정보와, 게스트 물리 주소를 포함한다. 운영 체제(120)는 게스트 물리 주소를 이용하여 가상화 머신 모니터(150)를 통해 하드웨어(140)에 접근할 수 있다.

【0045】 시스템 호출 처리부(122)는 응용 프로그램(110)에서 발생된 시스템 호출(system call)을 처리한다. 시스템 호출 처리부(122)는 시스템 호출에 따라 페이지 테이블(121)의 게스트 물리 주소를 이용하여, 접근하고자 하는 응용 프로그램(110)의 자원으로 할당된 메모리 페이지에 접근할 수 있다. 시스템 호출 처리부(122)는 응용 프로그램(110)의 자원에 접근하여 시스템 호출에 따른 동작, 예를 들면, 읽기, 쓰기 등의 동작을 수행할 수 있다. 시스템 호출 처리부(122)는 시스템 호출에 따른 작업을 수행한 후, 작업 수행에 대한 결과 값을 응용 프로그램(110) 및 응용 프로그램 보호부(130) 중 적어도 하나로 전달할 수 있다.

【0046】 응용 프로그램 보호부(130)는 가상화 머신 모니터(150)의 가상화 기술을 이용하여 운영 체제(120)와 독립된 공간에서 운영 체제(120)의 권한 보다 높은 최상위 권한으로 동작할 수 있다. 가상화 머신 모니터(150)는 하드웨어(140)와 운영 체제(120) 사이에 위치하고, 운영 체제(120)는 가상화 머신 모니터(150)를 통하여 하드웨어(140)에 접근할 수 있다. 응용 프로그램 보호부(130)는 중첩 페이지 테이블(151)을 이용하여, 응용 프로그램(110)의 자원으로 할당된 영역에 대한 운영 체제(120)의 접근 권한을 제어할 수 있다.

【0047】중첩 페이지 테이블(151, nested page table)은 게스트 물리 주소와 호스트 물리 주소 간의 사상(mapping) 정보 및 호스트 물리 주소와 대응되는 메모리 페이지에 대한 접근 권한 정보를 포함한다. 운영 체제(120)는 접근하고자 하는 메모리 페이지의 게스트 물리 주소를 이용하여 호스트 물리 주소와 대응되는 메모리 페이지에 접근할 수 있다.

【0048】구체적으로, 응용 프로그램 보호부(130)는 운영 체제(120)가 접근하고자 하는 메모리 페이지의 게스트 물리 주소를 중첩 페이지 테이블(151)을 이용하여 호스트 물리 주소로 변환할 수 있다. 응용 프로그램 보호부(130)는 중첩 페이지 테이블(151)에 포함된 호스트 물리 주소와 대응되는 메모리 페이지에 대한 접근 권한 정보를 재설정 할 수 있다. 재설정된 접근 권한 정보에 따라 운영 체제(120)의 메모리 페이지에 대한 접근이 제어될 수 있다. 중첩 페이지 테이블(151)에 포함된 접근 권한 정보에 따라, 호스트 물리 주소와 대응되는 메모리 페이지에서 운영 체제(120)의 읽기, 쓰기 또는 실행 동작이 제어될 수 있다.

【0049】중첩 페이지 테이블(151)에 포함된 접근 권한 정보는 호스트 물리 주소와 대응되는 메모리 페이지에 대한 읽기, 쓰기 및 실행 동작에 대한 권한 비트(permission bits)를 포함한다. 읽기 동작은 메모리 페이지에 저장된 데이터를 읽는 동작을 의미한다. 쓰기 동작은 메모리 페이지에 소정의 데이터를 저장하는 동작을 의미한다. 실행 동작은 메모리 페이지에 저장된 데이터를 실행시키는 동작을 의미한다. 권한 비트는 각 동작 별로 존재할 수 있으며, 0 또는 1의 값을 가질 수 있다. 예를 들어, 읽기 동작에 대한 권한 비트 값이 0의 값을 가지는 경우, 해당 메

모리 페이지에서의 읽기 동작은 허여되지 않는다. 반면, 읽기 동작에 대하여 권한 비트 값이 1의 값을 가지는 경우, 해당 메모리 페이지에서의 읽기 동작은 허여된다. 메모리 페이지에 대한 접근 권한 정보는 접근하는 주체를 구분하지 않고, 메모리 페이지에 대한 동작 별로 다르게 설정될 수 있다. 다만, 이에 한하지 않고, 접근 권한 정보는 메모리 페이지에 접근하는 주체 별로 다르게 설정될 수도 있다.

【0050】중첩 페이지 테이블(151)은 가상화 머신 모니터(150) 내에 위치하여, 운영 체제(120)와 독립적으로 구성될 수 있고, 운영 체제(120)의 접근으로부터 보호될 수 있다.

【0051】응용 프로그램 보호부(130)는 응용 프로그램 관리부(132), 자원 보호부(133) 및 커널 수행 검증부(134)를 포함한다.

【0052】응용 프로그램 관리부(132)는 응용 프로그램(110)을 보호 대상으로 설정할 수 있다. 응용 프로그램 관리부(132)는 응용 프로그램(110)의 요청에 따라 응용 프로그램(110)을 보호 대상으로 등록할 수 있다. 이에 한하지 않고 응용 프로그램 관리부(132)는 응용 프로그램(110) 이외 다른 응용 프로그램의 요청 또는 미리 설정된 이벤트의 발생을 감지함에 따라 응용 프로그램(110)을 보호 대상으로 등록할 수 있다.

【0053】응용 프로그램 관리부(132)는 응용 프로그램(110)의 고유 식별 정보를 이용하여 응용 프로그램(110)을 보호하고자 하는 응용 프로그램으로 등록할 수 있다. 고유 식별 정보는 응용 프로그램(110)에 부여될 수 있는 프로세스 식별자

(PID, process identifier) 또는 응용 프로그램(110)과 대응되는 페이지 디렉토리(page directory) 주소를 포함할 수 있다. 또는, 고유 식별 정보는 응용 프로그램 관리부(132)에 의해 보호하고자 하는 응용 프로그램(110)에 대하여 부여된 식별 정보를 포함할 수 있다. 고유 식별 정보는 상술된 예에 한하지 않고 다양한 형태의 식별 정보를 포함할 수 있다.

【0054】응용 프로그램 관리부(132)는 보호 대상으로 설정된 응용 프로그램(110)을 감시하여, 운영 체제(120)가 응용 프로그램(110)의 자원에 접근하는 것을 제어할 수 있다. 구체적으로, 응용 프로그램 관리부(132)는 고유 식별 정보에 기초하여 현재 실행 중인 응용 프로그램(110)이 보호 대상인지 여부를 판단할 수 있다. 그리고, 응용 프로그램 관리부(132)는 보호 대상으로 설정된 응용 프로그램(110)의 실행 모드에 따라 운영 체제가 응용 프로그램(110)의 자원에 접근할 수 있는지 여부를 판단할 수 있다. 응용 프로그램 관리부(132)는 보호 대상으로 설정된 응용 프로그램(110)의 프로세스 또는 쓰레드 등을 보호 대상으로 등록하여, 등록된 보호 대상에 대한 운영 체제(120)의 접근을 제어할 수 있다.

【0055】응용 프로그램 관리부(132)는 보호 대상으로 설정된 응용 프로그램(110)에서 프로세스 또는 쓰레드가 새롭게 생성되면, 새롭게 생성된 프로세스 또는 쓰레드를 보호 대상으로 등록할 수 있다. 또한, 보호 대상으로 등록되어 있던 프로세스 또는 쓰레드가 소멸되면, 소멸된 프로세스 또는 쓰레드를 보호 대상에서 해제할 수 있다.

【0056】 응용 프로그램(110)이 응용 프로그램 관리부(132)에 의하여 보호 대상으로 설정되는 경우, 자원 보호부(133)는 응용 프로그램(110)의 페이지 테이블(121)에 기초하여, 응용 프로그램(110)의 자원으로 할당된 메모리 페이지의 게스트 물리 주소를 획득할 수 있다. 그리고, 자원 보호부(133)는 중첩 페이지 테이블(151)을 이용하여, 게스트 물리 주소를 호스트 물리 주소로 변환한다. 자원 보호부(133)는 호스트 물리 주소 및 응용 프로그램(110)의 고유 식별 정보에 기초하여, 응용 프로그램(110)의 자원으로 할당된 메모리 페이지를 보호 대상으로 등록할 수 있다.

【0057】 자원 보호부(133)는 중첩 페이지 테이블(151)에 포함된 보호 대상으로 등록된 메모리 페이지의 권한 정보를 재설정할 수 있다. 예를 들어, 운영 체제(120)가 응용 프로그램(110)의 자원에 접근할 수 있는 상태임을 응용 프로그램 관리부(132)가 감지하면, 자원 보호부(133)는, 보호 대상으로 등록된 메모리 페이지에 대한 접근 권한 정보를 재설정할 수 있다. 자원 보호부(133)는 읽기, 쓰기, 실행 동작에 대한 권한 비트를 0으로 설정함으로써 메모리 페이지에 대한 운영 체제(120)의 접근을 제어할 수 있다.

【0058】 운영 체제(120)가 응용 프로그램(110)의 자원에 접근할 수 있는 상태는 응용 프로그램(110)의 실행 모드가 유저(user) 모드에서 커널(kernel) 모드로 전환되어, 커널 모드로 응용 프로그램(110)이 실행되는 경우를 포함할 수 있다.

【0059】 유저 모드는 응용 프로그램(110)이 응용 프로그램(110)의 자원으로 할당된 하드웨어 영역에 접근할 수 있고, 운영 체제(120) 또는 다른 응용 프로그램

은 접근할 수 없는 응용 프로그램(110)의 실행 모드를 의미할 수 있다. 커널 모드는 응용 프로그램(110)에서 발생된 시스템 호출 또는 이벤트에 따라서, 응용 프로그램(110)의 자원으로 할당된 하드웨어(140) 영역에 운영 체제(120)가 접근할 수 있는 응용 프로그램(110)의 실행 모드를 의미할 수 있다.

【0060】응용 프로그램(110)이 유저 모드로 동작하는 경우, 응용 프로그램(110)의 자원으로 할당된 메모리 페이지에는 응용 프로그램(110)만 접근할 수 있고, 다른 응용 프로그램이나 운영 체제(120)는 접근할 수 없다. 그러나, 응용 프로그램(110)이 커널 모드로 동작하는 경우, 응용 프로그램(110)의 자원으로 할당된 메모리 페이지에 운영 체제(120) 또는 다른 응용 프로그램이 접근할 수 있다. 따라서, 일 실시 예에 의한 응용 프로그램 관리부(132)는 응용 프로그램(110)의 실행 모드가 유저 모드인지 커널 모드인지를 모니터링하고, 응용 프로그램(110)의 실행 모드가 커널 모드로 전환되면, 응용 프로그램의 자원에 대한 접근 권한을 제어할 수 있다.

【0061】또한, 응용 프로그램 관리부(132)가 운영 체제(120)가 응용 프로그램(110)의 자원에 접근할 수 있는 상태임을 감지하면, 자원 보호부(133)는 응용 프로그램(110)이 사용하던 범용 레지스터(general purpose registers)(144)에 저장된 값을 제거할 수 있다. 범용 레지스터(144)에는 응용 프로그램(110)이 실행 중일 때 사용된 데이터들이 저장될 수 있다. 응용 프로그램(110)의 자원으로 할당된 범용 레지스터(144) 영역에 응용 프로그램(110)과 관련된 데이터들이 저장될 수 있다. 예를 들면, 응용 프로그램(110)에서 발생된 이벤트, 프로세스, 쓰레드 등과 관련된

데이터들이 범용 레지스터(144)에 저장될 수 있다. 자원 보호부(133)는 운영 체제(120)가 범용 레지스터(144)에 저장된 값에 접근하는 것을 방지하기 위하여, 운영 체제(120)가 범용 레지스터(144)에 접근하기 전에 범용 레지스터의 값을 삭제할 수 있다.

【0062】 악성 소프트웨어, 악성 코드 등에 감염된 운영 체제(120)가 범용 레지스터(144)에 접근하는 경우, 응용 프로그램(110)이 사용하던 값이 운영 체제(120)에 의해 유출되거나 위변조될 수 있다. 따라서, 자원 보호부(133)는 운영 체제(120)로부터 범용 레지스터(144)에 저장된 값을 보호하기 위하여, 운영 체제(120)가 응용 프로그램(110)의 자원에 접근할 수 있는 상태가 되면, 범용 레지스터(144)에 저장된 값을 삭제할 수 있다.

【0063】 자원 보호부(133)는 운영 체제(120)가 응용 프로그램(110)의 접근할 수 있는 상태가 됨에 따라, 권한 정보를 재설정하고 범용 레지스터(144)에 저장된 값을 삭제하기 전에, 권한 정보와 범용 레지스터(144)에 저장된 값을 백업해둘 수 있다. 그리고 이후에, 자원 보호부(133)는 운영 체제(120)가 응용 프로그램(110)의 자원에 접근할 수 없는 상태가 되면, 미리 백업해둔 데이터를 이용하여, 권한 정보 및 범용 레지스터(144)에 저장된 값을 복구할 수 있다.

【0064】 운영 체제(120)가 응용 프로그램(110)의 자원에 접근할 수 없는 상태는 응용 프로그램(110)의 실행 모드가 커널(kernel) 모드에서 유저(user) 모드로 전환되어, 유저 모드로 응용 프로그램(110)이 실행되는 경우를 포함할 수 있다. 따라서, 응용 프로그램(110)의 실행 모드가 유저 모드가 되면, 운영 체제(120)는 응

용 프로그램(110)의 자원에 접근할 수 없으므로, 자원 보호부(133)는 백업된 데이터를 이용하여 권한 정보 및 범용 레지스터(144)에 저장된 값을 복구할 수 있다. 권한 정보 및 범용 레지스터(144)에 저장된 값이 복구됨에 따라, 응용 프로그램(110)은 복구된 데이터를 이용하여 정상적으로 실행될 수 있다.

【0065】 자원 보호부(133)에 의하여, 보호 대상으로 등록된 메모리 페이지에 대한 접근 권한 정보가 재설정되고, 범용 레지스터의 값이 제거된 이후, 운영 체제(120)의 시스템 호출 처리부(122)는 시스템 호출을 처리할 수 있다. 응용 프로그램(110)에서 발생된 시스템 호출을 처리하기 위하여, 시스템 호출 처리부(122)는 응용 프로그램(110)의 자원으로 할당된 메모리 페이지에 대한 접근을 시도할 수 있다.

【0066】 구체적으로, 시스템 호출 처리부(122)는 접근하고자 하는 메모리 페이지에 대한 게스트 물리 주소를 페이지 테이블(121)로부터 획득한다. 그리고, 시스템 호출 처리부(122)는 중첩 페이지 테이블(151)을 이용하여 게스트 물리 주소와 대응되는 호스트 물리 주소를 획득할 수 있다. 시스템 호출 처리부(122)는 중첩 페이지 테이블(151)로부터 획득된 호스트 물리 주소를 이용하여 접근하고자 하는 메모리 페이지에 대한 접근을 시도할 수 있다. 그러나, 시스템 호출 처리부(122)가 메모리 페이지에 접근할 수 없도록 접근 권한 정보가 자원 보호부(133)에 의해 재설정된 상태이므로, 시스템 호출 처리부(122)의 메모리 페이지에 대한 접근은 즉시 허용되지 않는다.

【0067】자원 보호부(133)는 시스템 호출 처리부(122)가 보호 대상으로 등록된 메모리 페이지에 접근하려고 하는 경우, 기약정된 보안 규칙에 따라 메모리 페이지에 저장된 데이터를 암호화한 후, 시스템 호출 처리부(122)에 암호화된 데이터를 제공할 수 있다. 자원 보호부(133)는 시스템 호출 처리부(122)가 처리하고자 하는 시스템 호출에 따라서 데이터를 암호화한 후, 시스템 호출 처리부(122)가 메모리 페이지에 접근할 수 있도록 접근 권한 정보를 다시 설정할 수 있다.

【0068】시스템 호출 처리부(122)가 보호 대상으로 등록된 메모리에 접근하는 시점에서, 보호 대상으로 등록된 메모리 페이지의 각 동작에 대한 권한 비트는 0으로 세팅된 상태이다. 따라서, 시스템 호출 처리부(122)는 보호 대상으로 등록된 메모리 페이지에 접근하여도, 접근 권한이 없으므로, 시스템 호출을 처리할 수 없다. 자원 보호부(133)는 기약정된 보안 규칙에 따라 메모리 페이지에 저장된 데이터를 암호화한 후, 시스템 호출 처리부(122)에 암호화된 데이터를 제공할 수 있다.

【0069】예를 들면, 시스템 호출 처리부(122)가 파일 열기 시스템 호출(file open system call)을 처리하는 경우, 열고자 하는 파일에 관한 정보(파일 이름 또는 경로)를 획득하는 것이 필요하다. 그러나, 파일 정보는 응용 프로그램(110)의 실행 중 사용된 정보이므로 응용 프로그램(110)의 자원으로 할당된 메모리 페이지에 저장되어 있다. 응용 프로그램(110)이 보호 대상으로 설정된 경우이면, 메모리 페이지도 보호 대상으로 등록되므로, 시스템 호출 처리부(122)가 메모리 페이지에 접근하는 시점에서, 메모리 페이지의 각 동작에 대한 권한 비트는 자원 보호부(133)에 의해 0으로 설정된 상태이다.

【0070】시스템 호출 처리부(122)가 파일 열기 시스템 호출의 처리를 위해 파일 이름 및 경로를 포함하는 파일 정보를 보호 대상으로 등록된 메모리 페이지로부터 획득하고자 하는 경우, 자원 보호부(133)는 메모리 페이지에 저장된 파일 정보를 암호화하지 않을 수 있다. 파일 이름이나 파일 경로는 보안상 중요한 정보에 해당되지 않으므로 자원 보호부(133)는 데이터를 암호화하지 않고 시스템 호출 처리부(122)에 제공할 수 있다. 자원 보호부(133)는 시스템 호출 처리부(122)가 파일 정보를 획득할 수 있도록 메모리 페이지에 대한 접근 권한 정보를 다시 설정하거나 암호화되지 않은 파일 정보를 시스템 호출 처리부(122)에 제공할 수 있다.

【0071】또 다른 예를 들면, 시스템 호출 처리부(122)가 파일 쓰기 시스템 호출(write system call)을 처리하는 경우, 소정의 저장 영역에서 쓰기 동작을 수행하기 위한 데이터를 획득하는 것이 필요하다. 그러나, 파일 쓰기 시스템 호출 처리에 필요한 데이터는 응용 프로그램(110)의 실행 중 사용된 정보이므로 응용 프로그램(110)의 자원으로 할당된 메모리 페이지에 저장되어 있다. 응용 프로그램(110)이 보호 대상으로 설정된 경우이면, 메모리 페이지도 보호 대상으로 등록되므로, 시스템 호출 처리부(122)가 메모리 페이지에 접근하는 시점에서, 메모리 페이지의 각 동작에 대한 권한 비트는 자원 보호부(133)에 의해 0으로 설정된 상태이다.

【0072】파일 쓰기 시스템 호출 처리에 필요한 데이터는 파일 이름 및 경로를 포함하는 파일 정보와는 달리, 응용 프로그램(110)의 데이터가 포함되어 있으므로, 보안상 중요한 정보를 포함할 수 있다. 따라서, 자원 보호부(133)는 파일 쓰기 시스템 호출 처리에 필요한 데이터를 암호화한 후, 시스템 호출 처리부(122)가 암

호화된 데이터를 획득할 수 있도록 메모리 페이지에 대한 읽기 권한을 허여할 수 있다. 또는, 자원 보호부(133)는 암호화된 데이터를 시스템 호출 처리부(122)에 제공하여 줄 수 있다.

【0073】 시스템 호출 처리부(122)는 암호화된 데이터에 대한 정당한 권리를 가지고 있지 않아 암호화 키를 획득할 수 없으므로, 암호화된 데이터를 복호화할 수 없다. 따라서, 시스템 호출 처리부(122) 및 운영 체제(120)에 의해 응용 프로그램(110)의 데이터가 유출되거나 위변조되는 것이 방지될 수 있다.

【0074】 더하여, 메모리 페이지가 응용 프로그램(110)의 자원으로 할당되거나 해제됨에 따라, 자원 보호부(133)는 메모리 페이지를 보호 대상으로 등록하거나 보호 대상에서 해제할 수 있다. 운영 체제(120)에 의해 메모리 페이지는 응용 프로그램(110)의 자원으로 할당되거나 해제될 수 있다. 운영 체제(120)는 메모리 페이지를 응용 프로그램(110)의 자원으로 할당하거나 해제함에 따라 페이지 테이블(121)을 갱신할 수 있다.

【0075】 자원 보호부(133)는 보호 대상으로 등록된 응용 프로그램(110)의 페이지 테이블(121)을 보호 대상으로 등록해 두고, 운영 체제(120)에 의해 페이지 테이블(121)이 수정되는지 여부를 감시할 수 있다.

【0076】 예를 들어, 운영 체제(120)에 의해 메모리 페이지가 새롭게 할당되면, 페이지 테이블(121)에 새롭게 할당된 메모리 페이지에 대한 게스트 가상 주소와 게스트 물리 주소 간 사상 정보가 저장된다. 그리고, 중첩 페이지 테이블(151)에는 새롭게 할당된 메모리 페이지에 대한 게스트 물리 주소와 호스트 물리 주소간

사상 정보가 저장된다. 페이지 테이블(121) 및 중첩 페이지 테이블(151)에는 새롭게 할당된 메모리 페이지에 관한 정보가 저장될 수 있다.

【0077】 반면에, 운영 체제(120)에 의해 할당되어 있던 메모리 페이지가 해제되면, 페이지 테이블(121) 및 중첩 페이지 테이블(151)에 저장된 해제된 메모리 페이지에 대한 정보와 메모리 페이지에 저장되어 있던 정보가 삭제된다. 자원 보호부(133)는 페이지 테이블(121)에서 수정된 값을 확인하고, 새롭게 할당되거나 해제된 메모리 페이지에 관한 정보를 획득할 수 있다. 그리고, 자원 보호부(133)는 페이지 테이블(121)에서 수정된 값에 따라, 중첩 페이지 테이블(151)과 보호 대상으로 등록된 메모리 페이지에 관한 정보를 갱신할 수 있다.

【0078】 자원 보호부(133)는 보호 대상으로 설정된 응용 프로그램에 대한 자원으로 메모리 페이지가 새롭게 할당되면 새로 할당된 메모리 페이지를 보호 대상으로 등록할 수 있다. 또한, 보호 대상으로 등록된 메모리 페이지의 할당이 해제되면, 할당이 해제된 메모리 페이지에 저장된 데이터가 삭제된 이후, 할당이 해제된 메모리 페이지가 보호 대상에서 해제될 수 있다. 데이터 삭제가 보호 대상에서의 해제보다 선행됨으로써, 할당이 해제된 메모리 페이지가 보호 대상에서 해제됨에 따라 발생될 수 있는 데이터 유출 또는 위변조가 방지될 수 있다.

【0079】 커널 수행 검증부(134)는 시스템 호출 처리부(122)에 의해 시스템 호출이 처리된 이후 시스템 호출의 처리 결과에 대하여 검증한다. 운영 체제(120)가 악성 코드에 의해 감염된 상태인 경우, 운영 체제(120)의 시스템 호출 처리부(122)에 의해 수행된 시스템 호출에 따라 처리된 데이터는 위변조될 수 있는 가능

성이 있다. 따라서, 커널 수행 검증부(134)는 시스템 호출 처리부(122)에 의해 처리되는 데이터의 원본 데이터와, 시스템 호출에 따라 처리된 데이터를 비교하여 데이터의 무결성을 검증할 수 있다. 예를 들면, 커널 수행 검증부(134)는 원본 데이터의 해시값(hash value)과 시스템 호출에 따라 처리된 데이터의 해시값을 비교하여, 데이터의 무결성을 검증할 수 있다.

【0080】예를 들어, 시스템 호출 처리부(122)가 읽기 시스템 호출(read system call)을 수행하는 경우, 시스템 호출 처리부(122)는 다른 메모리 영역 또는 저장 장치에 저장된 데이터를 읽고, 읽은 데이터를 보호 대상으로 등록된 메모리 페이지에 저장할 수 있다. 커널 수행 검증부(134)는 읽기 시스템 호출이 처리되기 전에, 시스템 호출 처리부(122)가 시스템 호출 처리를 위해 읽으려는 데이터를 원본 데이터로써 획득할 수 있다. 그리고, 읽기 시스템 호출이 처리됨에 따라 시스템 호출 처리부(122)는 보호 대상으로 등록된 메모리 페이지에 데이터를 저장할 수 있다. 커널 수행 검증부(134)는 시스템 호출 처리부(122)에 의하여 저장된 데이터의 해시 값과, 원본 데이터를 해시 값을 비교함으로써, 데이터의 무결성을 검증할 수 있다.

【0081】또한, 시스템 호출 처리부(122)가 쓰기 시스템 호출을 수행하는 경우, 시스템 호출 처리부(122)는 보호 대상으로 등록된 메모리 페이지에 저장된 데이터를 읽고, 읽은 데이터를 다른 메모리 영역 또는 저장 장치에 저장할 수 있다. 커널 수행 검증부(134)는 쓰기 시스템 호출이 처리되기 전에, 보호 대상으로 등록된 메모리 페이지로부터 시스템 호출 처리부(122)가 시스템 호출 처리를 위해 읽으

려는 데이터를 원본 데이터로써 획득할 수 있다. 그리고, 쓰기 시스템 호출이 처리됨에 따라 시스템 호출 처리부(122)는 다른 메모리 영역 또는 저장 장치에 데이터를 저장할 수 있다. 쓰기 시스템 호출에 대한 처리가 완료된 이후, 커널 수행 검증부(134)는 시스템 호출 처리부(122)에 의하여 저장된 데이터의 해시 값과, 원본 데이터를 해시 값을 비교함으로써, 데이터의 무결성을 검증할 수 있다.

【0082】더하여, 시스템 호출 처리부(122)가 응용 프로그램의 자원으로 새로운 메모리 페이지를 할당하기 위한 시스템 호출을 수행하는 경우, 커널 수행 검증부(134)는 시스템 호출 처리부(122)에 의한 시스템 호출의 처리 결과를 검증할 수 있다. 응용 프로그램의 자원으로 새로운 메모리 페이지를 할당하기 위한 시스템 호출은 예를 들면, mmap 시스템 호출을 포함할 수 있다.

【0083】시스템 호출 처리부(122)에 의하여, 기존에 보호 대상으로 등록된 메모리 페이지가 새로운 메모리 페이지로서 할당되면, 응용 프로그램은 할당된 메모리 페이지에 대하여 읽기, 쓰기 등의 동작을 수행할 수 있다. 따라서, 이전에 저장되어 있던 데이터가 삭제되거나 변조될 수 있다.

【0084】커널 수행 검증부(134)는 새로운 메모리 페이지를 실행 중인 응용 프로그램의 자원으로 할당하기 위한 시스템 호출이 처리된 이후, 새롭게 할당된 메모리 페이지가 기존에 응용 프로그램의 자원으로 할당되어 있는 메모리 페이지와 동일한지 확인한다. 동일한 경우, 커널 수행 검증부(134)는 새롭게 할당된 메모리 페이지에 대한 할당이 해제되도록 처리할 수 있다. 기존에 할당된 메모리 페이지는 자원 보호부(133)에 의해 보호 대상으로 등록된 메모리 페이지를 포함할 수 있다.

【0085】 이하 도 3 및 도 4를 참조하여, 응용 프로그램의 자원에 대한 접근 권한을 제어하는 방법에 대하여 설명하기로 한다.

【0086】 도 3은 일 실시 예에 의한 응용 프로그램의 자원에 대한 접근 권한을 제어하는 방법을 나타낸 순서도이다.

【0087】 도 3을 참조하면, 단계 S301에서 디바이스(100)는 보호 대상으로 설정된 응용 프로그램이 실행되는 동안, 운영 체제가 보호 대상으로 설정된 응용 프로그램의 자원에 접근할 수 있는 상태인지 여부를 판단한다. 구체적으로, 디바이스(100)는 실행 중인 응용 프로그램 중에, 보호 대상으로 설정된 응용 프로그램을 판별하고, 보호 대상으로 설정된 응용 프로그램의 자원에 운영 체제가 접근할 수 있는 상태인지를 지속적으로 모니터링할 수 있다.

【0088】 단계 S303에서, 디바이스(100)는 보호 대상으로 설정된 응용 프로그램의 자원에 운영 체제가 접근 가능한 상태인지 판단할 수 있다. 예를 들면, 디바이스(100)는 응용 프로그램의 실행 모드가 유저 모드에서 커널 모드로 전환되는 경우, 운영 체제가 응용 프로그램의 자원에 접근할 수 있는 상태에 있는 것으로 판단할 수 있다. 응용 프로그램의 실행 모드가 커널 모드인 경우, 응용 프로그램에서 발생된 시스템 호출 또는 이벤트에 따라, 응용 프로그램의 자원으로 할당된 하드웨어 영역에 운영 체제가 접근할 수 있다.

【0089】 단계 S303에서, 보호 대상으로 설정된 응용 프로그램의 자원에 운영 체제가 접근 가능한 상태인 것으로 판단된 경우, 단계 S305에서, 디바이스(100)는

보호 대상으로 설정된 응용 프로그램의 자원에 대한 접근 권한을 제어할 수 있다. 보호 대상으로 설정된 응용 프로그램의 자원에 운영 체제가 접근하여, 응용 프로그램의 자원에 존재하는 데이터를 유출시키거나 위변조하는 것을 막기 위해, 디바이스(100)는 자원에 대한 접근 권한을 제어할 수 있다. 자원에 대한 접근 권한은 읽기, 쓰기, 실행 등의 동작 별로 허용 여부에 관해 설정될 수 있으며, 접근 주체를 구별하여 설정되지는 않을 수 있다. 이에 한하지 않고, 접근 주체 별로 자원에 대한 접근 권한이 설정될 수도 있다.

【0090】 디바이스(100)는 응용 프로그램의 자원으로 할당된 메모리 페이지를 보호하고자 하는 경우, 보호하고자 하는 메모리 페이지를 보호 대상으로 등록할 수 있다. 그리고, 디바이스(100)는 중첩 페이지 테이블(151)에 저장된 메모리 페이지에 대한 접근 권한을 재설정할 수 있다.

【0091】 이하 도 4를 참조하여, 응용 프로그램의 자원에 대한 접근 권한을 제어하는 일 예로, 메모리 페이지의 접근 권한과 레지스터의 데이터 값을 제거하는 방법에 대하여 더 자세히 설명하기로 한다.

【0092】 도 4는 일 실시 예에 의한 응용 프로그램의 자원에 대한 접근 권한 및 레지스터 값을 제어하는 방법을 나타낸 순서도이다.

【0093】 도 4를 참조하면, 단계 S401에서, 디바이스(100)는 보호 대상으로 설정된 응용 프로그램의 실행 모드를 판단할 수 있다. 응용 프로그램의 실행 모드에 따라서, 운영 체제가 보호 대상으로 설정된 응용 프로그램의 자원에 접근이 가능한지 여부가 결정될 수 있다. 디바이스(100)는 응용 프로그램의 실행 모드에 따

라 응용 프로그램의 자원에 대한 접근 권한을 제어할 수 있다.

【0094】 단계 S403에서, 디바이스(100)는 보호 대상으로 설정된 응용 프로그램이 커널 모드로 동작하는지 여부를 판단할 수 있다. 디바이스(100)는 보호 대상으로 설정된 응용 프로그램이 유저 모드로 동작하다가 커널 모드로 전환되어 동작하는지 여부를 판단할 수 있다.

【0095】 유저 모드에서 커널 모드로의 전환은 응용 프로그램에서 인터럽트(interrupt), 예외(exception), 시스템 호출(system call) 등의 이벤트가 발생됨에 따라 이루어질 수 있다. 또한, 커널 모드에서 유저 모드로의 전환은 응용 프로그램에서 스케줄링, 시스템 호출 반환(system call return) 등의 이벤트가 발생됨에 따라 이루어질 수 있다. 상술된 이벤트들은 예시에 불과하며, 이에 한정되지는 않는다.

【0096】 또 다른 예로, 디바이스(100)는 응용 프로그램의 실행 모드를 전환시키는 이벤트에 의하여 발생될 수 있는 VM exit, EPT violation 등의 특정 이벤트들을 이용하여 응용 프로그램의 실행 모드가 전환됨을 감지할 수 있다. VM exit는 응용 프로그램의 실행 모드를 사용자 모드에서 커널 모드로 전환시키는 이벤트들에 의해 발생될 수 있는 이벤트이다. EPT violation은 커널 모드에서, 운영 체제(120) 또는 다른 응용 프로그램이 보호 대상으로 등록된 응용 프로그램의 메모리 페이지에 접근할 때 운영 체제(120) 또는 다른 응용 프로그램의 접근 권한이 없어 발생될 수 있는 이벤트이다.

【0097】 따라서, 디바이스(100)는 응용 프로그램에서 발생된 이벤트를 감지

함으로써 응용 프로그램의 실행 모드가 커널 모드로 전환되었는지 여부를 판단할 수 있다.

【0098】 예를 들면, 디바이스(100)는 응용 프로그램의 실행 모드를 전환 시키는 이벤트의 발생을 감지할 수 있도록 하는 수정된 라이브러리를 이용할 수 있다. 디바이스(100)는 응용 프로그램의 실행 모드를 전환시키는 이벤트를 처리하기 위한 기존의 진입점(entry)이 특정 진입점으로 교체된 라이브러리를 이용할 수 있다. 따라서, 디바이스(100)는 수정된 라이브러리를 이용함으로써 응용 프로그램의 실행 모드를 전환시키거나 실행 모드가 전환될 때 발생 가능한 이벤트가 발생되면, 특정 진입점으로 이벤트가 처리될 수 있다. 디바이스(100)는 특정 진입점으로 이벤트가 처리됨에 따라 이용함으로써 응용 프로그램의 실행 모드를 전환시키거나 실행 모드가 전환될 때 발생 가능한 이벤트의 발생을 감지할 수 있다. 예를 들면, 디바이스(100)가 수정된 라이브러리를 이용함에 따라, 발생된 이벤트를 감지할 수 있는 신호가 생성될 수 있다.

【0099】 단계 S403에서, 보호 대상으로 설정된 응용 프로그램의 실행 모드가 커널 모드인 것으로 판단된 경우, 단계 S405에서, 응용 프로그램 보호부(130)는 응용 프로그램의 자원으로 할당된 메모리 페이지의 접근 권한 정보와 레지스터 값을 백업해 둘 수 있다. 백업될 수 있는 레지스터 값은 응용 프로그램이 실행 중에 범용 레지스터에 저장해둔 데이터를 포함한다. 응용 프로그램 보호부(130)는 응용 프로그램의 실행 모드가 유저 모드로 전환되었을 때, 단계 S405에서 백업된 값을 이용하여 접근 권한 정보 및 레지스터 값을 복구할 수 있다.

【0100】 단계 S405에서 백업이 완료되면, 단계 S407에서, 응용 프로그램 보호부(130)는 응용 프로그램의 자원으로 할당된 메모리 페이지의 접근 권한 정보와 레지스터 값을 제거할 수 있다. 응용 프로그램 보호부(130)는 운영 체제(120)가 응용 프로그램의 메모리 페이지에 접근할 수 없도록 메모리 페이지의 접근 권한 정보를 재설정할 수 있다. 또한, 응용 프로그램 보호부(130)는 운영 체제(120)가 응용 프로그램의 레지스터 값을 유출할 수 없도록 응용 프로그램이 저장해둔 레지스터 값을 제거할 수 있다.

【0101】 단계 S411에서, 운영 체제(120)는 응용 프로그램(110)에서 발생된 시스템 호출을 처리할 수 있다. 일 실시 예에 의하면 운영 체제(120)가 시스템 호출을 처리하기 위해 응용 프로그램(110)의 자원에 접근하기 전에, 응용 프로그램 보호부(130)는 단계 S407에서 메모리 페이지의 접근 권한을 재설정하고, 레지스터의 데이터 값을 제거할 수 있다. 따라서, 디바이스(100)는 악성 코드, 악성 소프트웨어 등에 감염된 운영 체제(120)가 응용 프로그램(110)의 자원에 접근하여 데이터를 유출하거나 위변조하는 것을 미리 방지할 수 있다.

【0102】 더하여, 응용 프로그램 보호부(130)는 단계 S401에서 보호 대상으로 설정된 응용 프로그램의 실행 모드를 지속적으로 판단할 수 있다. 응용 프로그램 보호부(130)는 응용 프로그램의 실행 모드가 커널 모드에서 유저 모드로 전환되는지 여부를 지속적으로 모니터링할 수 있다.

【0103】 유저 모드에서는 운영 체제(120) 또는 다른 응용 프로그램에서 응용 프로그램의 자원에 접근하는 것이 가능하지 않다. 따라서, 유저 모드에서는 운영

체제(120) 다른 외부 응용 프로그램에서 응용 프로그램의 자원에 접근함에 따라 발생될 수 있는 데이터의 유출 및 위변조가 발생되지 않는다. 그러므로, 응용 프로그램의 실행 모드가 유저 모드로 전환되면, 응용 프로그램 보호부(130)는 응용 프로그램의 내부에서 응용 프로그램의 자원에 접근할 수 있도록 자원에 대한 접근 권한을 제어할 수 있다. 응용 프로그램이 자신의 작업을 수행하기 위해 자신의 자원에 접근할 수 있음이 바람직하다. 그러므로, 응용 프로그램 보호부(130)는 응용 프로그램의 실행 모드가 유저 모드로 전환되면, 응용 프로그램이 자원에 접근할 수 있도록 자원에 대한 접근 권한을 제어할 수 있다.

【0104】 단계 S403에서, 응용 프로그램의 실행 모드가 유저 모드인 것으로 판단되면, 단계 S413에서, 응용 프로그램 보호부(130)는 메모리 페이지에 대한 접근 권한 정보와 레지스터 값에 대하여 미리 백업된 값이 존재하는지 여부를 판단할 수 있다.

【0105】 응용 프로그램(110)의 실행 모드가 커널 모드일 때, 응용 프로그램 보호부(130)는 단계 S405에서 메모리 페이지에 대한 접근 권한 정보와 레지스터 값을 백업해 둘 수 있다. 그리고, 응용 프로그램(110)의 실행 모드가 커널 모드에서 유저 모드로 전환되면, 단계 S415에서, 응용 프로그램 보호부(130)는 단계 S405에서 백업해 둔 값을 이용하여 메모리 페이지에 대한 접근 권한 정보와 레지스터 값을 복구할 수 있다.

【0106】 단계 S413에서, 백업된 값이 존재하지 않는 경우, 데이터 복구를 수행할 수 없으므로, 응용 프로그램 보호부(130)는 단계 S415의 데이터 복구를 수행

하지 않는다.

【0107】 단계 S417에서, 응용 프로그램은 실행 모드가 커널 모드로 전환되기 전에 유저 모드에서 수행하던 작업을 단계 S415에서 복구된 값에 기초하여 계속 수행할 수 있다. 또한, 단계 S415에서 복구된 값에 기초하여, 응용 프로그램은 자신의 자원으로 할당된 메모리 페이지에 접근함으로써 응용 프로그램에서 작업을 수행할 수 있다.

【0108】 단계 S413에서 백업된 값이 존재하지 않아 단계 S415의 데이터 복구가 수행되지 않은 경우, 메모리 페이지에 대한 접근 권한 정보는 디폴트 값으로 설정되거나 응용 프로그램에서 수행되는 작업에 따라 새롭게 설정될 수 있다.

【0109】 이하 도 5를 참조하여, 운영 체제가 시스템 호출을 처리하는 방법에 대하여 상세히 설명하기로 한다.

【0110】 도 5는 일 실시 예에 의한 운영 체제가 응용 프로그램의 자원에 접근하여 시스템 호출을 처리하는 방법을 나타낸 순서도이다.

【0111】 도 5를 참조하면, 단계 S501에서, 응용 프로그램 보호부(130)는 운영 체제(120)가 보호 대상인 응용 프로그램의 자원에 접근함을 감지할 수 있다. 응용 프로그램의 실행 모드가 커널 모드인 경우, 응용 프로그램 보호부(130)는 운영 체제(120)가 응용 프로그램의 자원에 접근할 수 없도록 자원에 대한 접근 권한을 제어한다. 따라서, 운영 체제(120)의 응용 프로그램의 자원으로 할당된 메모리 페이지에 대한 접근은 허용되지 않는다. 운영 체제(120)에 의한 응용 프로그램의 자

원에 대한 접근 시도가 발생하면, 응용 프로그램 보호부(130)는 발생된 접근 시도를 감지하고, 분석하여, 접근 시도가 감지된 자원에 대한 접근 권한을 제어할 수 있다.

【0112】 단계 S503에서, 운영 체제(120)가 시스템 호출을 처리함에 따라 페이지 테이블에 접근한 것으로 판단한 경우, 단계 S505에서, 응용 프로그램 보호부(130)는 운영 체제(120)에 의해 수정된 페이지 테이블을 확인할 수 있다. 그리고, 수정된 페이지 테이블에 따라 응용 프로그램 보호부(130)는 보호 대상으로 등록된 메모리 페이지를 해제하거나 새로운 메모리 페이지를 보호 대상으로 등록할 수 있다. 운영 체제(120)가 새로운 메모리 페이지를 응용 프로그램의 자원으로 할당한 경우, 응용 프로그램 보호부(130)는 새로운 메모리 페이지를 보호 대상으로 등록할 수 있다. 운영 체제(120)가 메모리 페이지에 대한 할당을 해제하는 경우, 응용 프로그램 보호부(130)는 보호 대상으로 등록된 메모리 페이지를 보호 대상에서 해제할 수 있다.

【0113】 단계 S503에서, 운영 체제(120)가 페이지 테이블이 아닌 응용 프로그램의 자원으로 할당된 메모리 페이지에 접근하는 것으로 판단한 경우, 단계 S507에서 응용 프로그램 보호부(130)는 기 약정된 보안 규칙에 따라 운영 체제(120)가 접근한 메모리 페이지에 저장된 데이터를 암호화할지 여부를 결정할 수 있다.

【0114】 응용 프로그램 보호부(130)는 운영 체제(120)가 수행하는 시스템 호출 또는 운영 체제(120)가 접근한 메모리 페이지에 저장된 데이터의 종류에 따라 암호화 여부를 결정할 수 있다. 예를 들어, 운영 체제(120)가 파일 열기 시스템 호

출을 수행함에 따라 메모리 페이지에 접근하였거나, 메모리 페이지에 저장된 데이터가 보안상 중요하지 않은 경우, 응용 프로그램 보호부(130)는 데이터를 암호화하지 않을 수 있다. 반면에, 운영 체제(120)가 파일 쓰기 시스템 호출을 수행함에 따라 메모리 페이지에 접근하였거나, 메모리 페이지에 저장된 데이터가 보안상 중요한 데이터를 포함하는 경우, 응용 프로그램 보호부(130)는 데이터를 암호화할 수 있다.

【0115】 단계 S509에서 응용 프로그램 보호부(130)는 단계 S507에서의 결정에 따라, 운영 체제(120)에 의해 접근된 메모리 페이지에 저장된 데이터를 암호화할 수 있다. 그리고, 단계 S511에서, 응용 프로그램 보호부(130)는 암호화된 데이터 또는 암호화되지 않은 데이터에 접근할 수 있도록 메모리 페이지에 대한 접근 권한을 제어할 수 있다. 운영 체제(120)는 접근이 허용된 메모리 페이지에 접근하여 암호화된 데이터 또는 암호화되지 않은 데이터를 획득할 수 있다. 또는, 응용 프로그램 보호부(130)는 단계 S511에서 운영 체제(120)에 암호화된 데이터 또는 암호화되지 않은 데이터를 제공할 수도 있다.

【0116】 단계 S513에서, 운영 체제(120)는 단계 S511에서 획득한 데이터를 이용하여 시스템 호출을 처리할 수 있다. 운영 체제(120)의 시스템 호출 처리에 따라 단계 S511에서 획득한 암호화된 데이터는 응용 프로그램, 다른 응용 프로그램, 외부 장치 등으로 전달될 수 있다. 암호화된 데이터는 암호화 키에 의해 복호화될 수 있다.

【0117】암호화된 데이터에 대하여 정당한 권한을 가지는 프로그램 또는 장치가 암호화 키를 보유할 수 있다. 예를 들면, 암호화 키는 암호화된 데이터에 대하여 정당한 권한을 가지는 응용 프로그램 또는 응용 프로그램 보호부(130)에 존재할 수 있다. 응용 프로그램 보호부(130)가 암호화 키를 가지는 경우, 암호화 키를 응용 프로그램에 전달하여 응용 프로그램이 암호화 키를 획득할 수 있다.

【0118】단계 S513에서 운영 체제(120)의 시스템 호출에 대한 처리가 완료되면, 단계 S515에서, 응용 프로그램 보호부(130)는 운영 체제(120)에 처리된 결과물에 대하여 검증할 수 있다. 응용 프로그램 보호부(130)는 운영 체제(120)에 의해 처리된 데이터에 대한 원본 데이터와 운영 체제(120)에 의해 처리된 데이터를 획득할 수 있다. 그리고, 응용 프로그램 보호부(130)는 운영 체제(120)에 의해 처리된 데이터의 해시값과 원본 데이터의 해시값을 비교함으로써 데이터의 무결성을 검증할 수 있다.

【0119】일 실시 예에 의하면, 악성 소프트웨어에 의해 해킹될 가능성이 있는 운영 체제 또는 다른 응용 프로그램으로부터 응용 프로그램의 자원을 보호할 수 있다.

【0120】일부 실시 예에 의한 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 본 발명을 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게

공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다.

【0121】비록 상기 설명이 다양한 실시예들에 적용되는 본 발명의 신규한 특징들에 초점을 맞추어 설명되었지만, 본 기술 분야에 숙달된 기술을 가진 사람은 본 발명의 범위를 벗어나지 않으면서도 상기 설명된 장치 및 방법의 형태 및 세부사항에서 다양한 삭제, 대체, 및 변경이 가능함을 이해할 것이다. 따라서, 본 발명의 범위는 상기 설명에서보다는 첨부된 특허청구범위에 의해 정의된다. 특허청구범위의 균등 범위 안의 모든 변형은 본 발명의 범위에 포섭된다.

【특허청구범위】

【청구항 1】

디바이스가 응용 프로그램의 자원을 보호하기 위한 방법에 있어서,

보호 대상으로 설정된 응용 프로그램이 실행되는 동안, 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지 여부를 판단하는 단계; 및

상기 판단 결과에 따라, 상기 응용 프로그램의 자원에 대한 접근 권한을 제어하는 단계를 포함하는, 방법.

【청구항 2】

제1항에 있어서, 상기 자원에 대한 접근 권한을 제어하는 단계는

상기 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인 경우, 상기 응용 프로그램의 자원으로 할당된 메모리 페이지에 대한 접근 권한 정보 및 상기 응용 프로그램의 레지스터 값을 백업하는 단계;

상기 메모리 페이지에 대한 접근 권한을 재설정하고, 상기 레지스터 값을 삭제하는 단계를 포함하는, 방법.

【청구항 3】

제2항에 있어서,

상기 메모리 페이지에 대한 접근 권한 정보 및 상기 레지스터 값 중 적어도 하나가 백업되어 있는 경우, 상기 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 없는 상태로 전환되면, 상기 백업된 값을 이용하여 상기 메모리 페이지에 대

한 접근 권한 정보 및 상기 레지스터 값 중 적어도 하나를 복구하는 단계를 더 포함하는, 방법.

【청구항 4】

제1항에 있어서, 상기 판단하는 단계는

상기 응용 프로그램의 실행 모드가 커널 모드인지 유저 모드인지 여부에 기초하여, 상기 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지 여부를 판단하는 단계를 포함하는, 방법.

【청구항 5】

제1항에 있어서,

상기 운영 체제가 시스템 호출을 처리하기 위하여, 상기 응용 프로그램의 자원에 대한 접근을 시도하는 단계;

상기 시스템 호출에 기초하여, 상기 자원에 대한 접근 권한을 제어하는 단계를 더 포함하는, 방법.

【청구항 6】

제5항에 있어서, 상기 자원에 대한 접근 권한을 제어하는 단계는

상기 응용 프로그램의 자원으로 할당된 메모리 페이지에 저장된 데이터를 암호화할 지 여부를 결정하는 단계;

상기 결정된 결과에 따라, 상기 데이터를 암호화한 후, 상기 운영 체제의 상기 메모리 페이지에 대한 접근을 허용하거나, 상기 암호화된 데이터를 상기 운영

체제에 제공하는 단계를 더 포함하는, 방법.

【청구항 7】

제1항에 있어서,

상기 운영 체제가 시스템 호출을 처리하기 위해, 상기 응용 프로그램의 자원으로 할당된 메모리 페이지에 대한 접근을 시도함을 감지하는 단계;

상기 시스템 호출에 의해 처리될 데이터의 원본 데이터를 획득하는 단계;

상기 운영 체제에 의해 상기 시스템 호출에 대한 처리가 완료되면, 상기 시스템 호출에 따라 처리된 데이터를 획득하는 단계;

상기 원본 데이터의 해시값과 상기 시스템 호출에 따라 처리된 데이터의 해시값을 비교함으로써, 상기 시스템 호출에 따라 처리된 데이터의 무결성을 검증하는 단계를 더 포함하는, 방법.

【청구항 8】

제1항에 있어서,

상기 응용 프로그램이 보호 대상으로 설정되면, 상기 응용 프로그램의 자원이 할당된 메모리 페이지는 보호 대상으로 등록되는, 방법.

【청구항 9】

제8항에 있어서,

상기 운영 체제에 의해 상기 응용 프로그램의 자원으로 새로운 메모리 페이지가 할당됨을 감지하는 단계;

상기 할당된 새로운 메모리 페이지가 상기 보호 대상으로 등록된 메모리 페이지와 동일하다고 판단되면, 상기 새로운 메모리 페이지에 대한 할당을 해제하는 단계를 더 포함하는, 방법.

【청구항 10】

제8항에 있어서,

상기 운영 체제가 상기 응용 프로그램의 페이지 테이블을 수정하는 경우, 상기 수정된 페이지 테이블에 따라 상기 응용 프로그램의 자원으로 할당된 메모리 페이지를 보호 대상으로 등록하거나 보호 대상에서 해제하는 단계를 더 포함하는, 방법.

【청구항 11】

제1항에 있어서,

상기 운영 체제 대신 다른 응용 프로그램이 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지 여부를 판단하고, 상기 판단 결과에 따라 상기 응용 프로그램의 자원에 대한 접근 권한을 수행하는, 방법.

【청구항 12】

응용 프로그램의 자원을 보호하기 위한 디바이스에 있어서,

보호 대상으로 설정되어 실행 중인 응용 프로그램;

상기 응용 프로그램에서 발생된 시스템 호출을 처리하는 운영 체제;

상기 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지 여

부를 판단하고, 상기 판단 결과에 따라, 상기 응용 프로그램의 자원에 대한 접근 권한을 제어하는 응용 프로그램 보호부;

상기 응용 프로그램의 자원을 포함하는 하드웨어를 포함하는 디바이스.

【청구항 13】

제12항에 있어서, 상기 응용 프로그램 보호부는

상기 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인 경우, 상기 응용 프로그램의 자원으로 할당된 메모리 페이지에 대한 접근 권한 정보 및 상기 응용 프로그램의 레지스터 값을 백업하고, 상기 메모리 페이지에 대한 접근 권한을 재설정하고, 상기 레지스터 값을 삭제하는, 디바이스.

【청구항 14】

제13항에 있어서, 상기 응용 프로그램 보호부는

상기 메모리 페이지에 대한 접근 권한 정보 및 상기 레지스터 값 중 적어도 하나가 백업되어 있는 경우, 상기 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 없는 상태로 전환되면, 상기 백업된 값을 이용하여 상기 메모리 페이지에 대한 접근 권한 정보 및 상기 레지스터 값 중 적어도 하나를 복구하는, 디바이스.

【청구항 15】

제12항에 있어서, 상기 응용 프로그램 보호부는

상기 응용 프로그램의 실행 모드가 커널 모드인지 유저 모드인지 여부에 기초하여, 상기 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지

여부를 판단하는, 디바이스.

【청구항 16】

제12항에 있어서, 상기 응용 프로그램 보호부는

상기 운영 체제가 시스템 호출을 처리하기 위하여, 상기 응용 프로그램의 자원에 대한 접근을 시도하면, 상기 시스템 호출에 기초하여, 상기 자원에 대한 접근 권한을 제어하는, 디바이스.

【청구항 17】

제16항에 있어서, 상기 응용 프로그램 보호부는

상기 응용 프로그램의 자원으로 할당된 메모리 페이지에 저장된 데이터를 암호화할 지 여부를 결정하고, 상기 결정된 결과에 따라, 상기 데이터를 암호화한 후, 상기 운영 체제의 상기 메모리 페이지에 대한 접근을 허용하거나, 상기 데이터를 상기 운영 체제에 제공하는, 디바이스.

【청구항 18】

제12항에 있어서, 상기 응용 프로그램 보호부는

상기 운영 체제가 시스템 호출을 처리하기 위해, 상기 응용 프로그램의 자원으로 할당된 메모리 페이지에 대한 접근을 시도함을 감지하면, 상기 시스템 호출에 의해 처리될 데이터의 원본 데이터를 획득하고, 상기 운영 체제에 의해 상기 시스템 호출에 대한 처리가 완료되면, 상기 시스템 호출에 따라 처리된 데이터를 획득하고, 상기 원본 데이터의 해시값과 상기 시스템 호출에 따라 처리된 데이터의 해

시값을 비교함으로써, 상기 시스템 호출에 따라 처리된 데이터의 무결성을 검증하는, 디바이스.

【청구항 19】

제12항에 있어서,

상기 응용 프로그램이 보호 대상으로 설정되면, 상기 응용 프로그램의 자원이 할당된 메모리 페이지는 보호 대상으로 등록되는, 디바이스.

【청구항 20】

제19항에 있어서, 상기 응용 프로그램 보호부는

상기 운영 체제에 의해 상기 응용 프로그램의 자원으로 새로운 메모리 페이지가 할당됨을 감지하고, 상기 할당된 새로운 메모리 페이지가 상기 보호 대상으로 등록된 메모리 페이지와 동일하다고 판단되면, 상기 새로운 메모리 페이지에 대한 할당을 해제하는, 디바이스.

【청구항 21】

제19항에 있어서, 상기 응용 프로그램 보호부는

상기 운영 체제가 상기 응용 프로그램의 페이지 테이블을 수정하는 경우, 상기 수정된 페이지 테이블에 따라 상기 응용 프로그램의 자원으로 할당된 메모리 페이지를 보호 대상으로 등록하거나 보호 대상에서 해제하는, 디바이스.

【청구항 22】

제17항에 있어서,

상기 운영 체제 대신 다른 응용 프로그램이 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지 여부를 판단하고, 상기 판단 결과에 따라 상기 응용 프로그램의 자원에 대한 접근 권한을 수행하는, 디바이스.

【청구항 23】

제1항 내지 제11항 중 어느 한 항에 있어서, 상기 방법을 구현하기 위한 프로그램이 기록된 컴퓨터로 판독 가능한 기록 매체.

【청구항 24】

제1항 내지 제11항 중 어느 한 항에 있어서, 하드웨어와 결합되어 상기 방법을 실행시키는 컴퓨터 프로그램.

【요약서】

【요약】

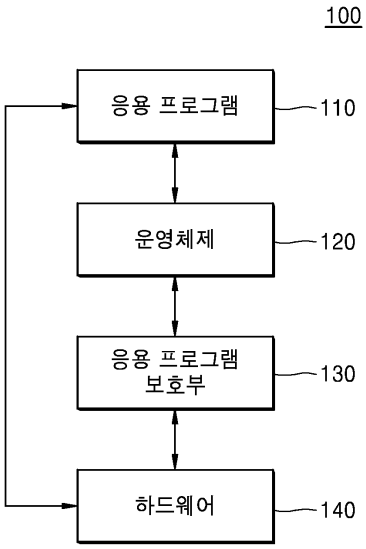
보호 대상으로 설정된 응용 프로그램이 실행되는 동안, 운영 체제가 상기 응용 프로그램의 자원에 접근할 수 있는 상태인지 여부를 판단하고, 상기 판단 결과에 따라, 상기 응용 프로그램의 자원에 대한 접근 권한을 제어하는, 디바이스가 응용 프로그램의 자원을 보호하기 위한 방법이 개시된다.

【대표도】

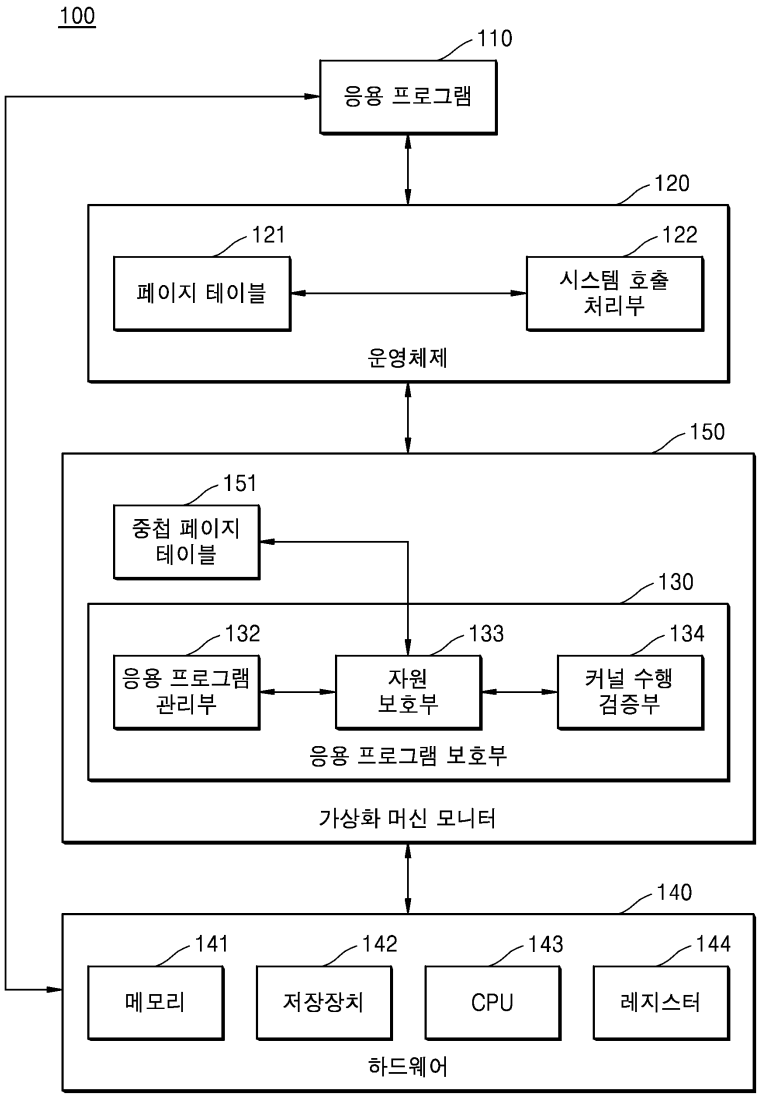
도 1

【도면】

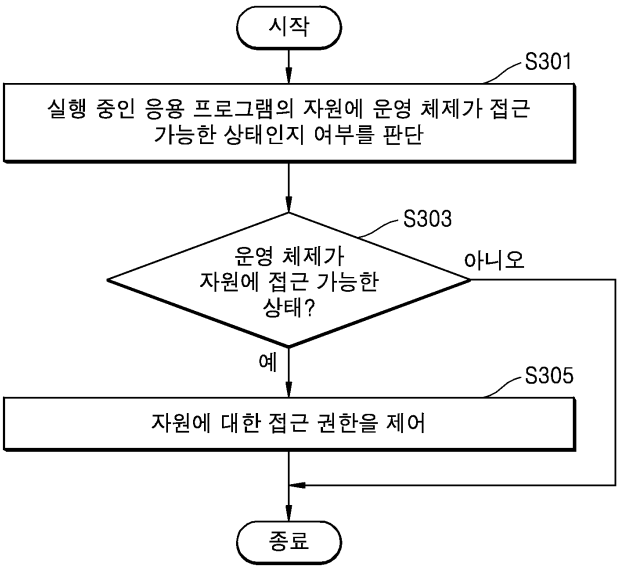
【도 1】



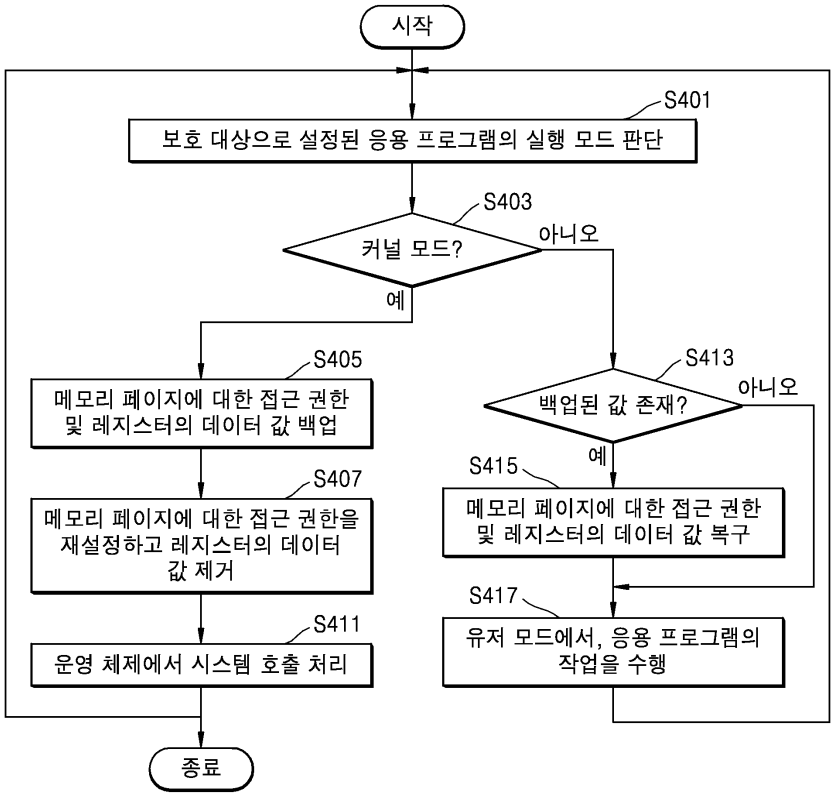
【도 2】


100
110
응용 프로그램
120
121
페이지 테이블
122
시스템 호출 처리부
운영체제
150
151
중첩 페이지 테이블
130
132
응용 프로그램 관리부
133
자원 보호부
134
커널 수행 검증부
응용 프로그램 보호부
가상화 머신 모니터
140
141
메모리
142
저장장치
143
CPU
144
레지스터
하드웨어

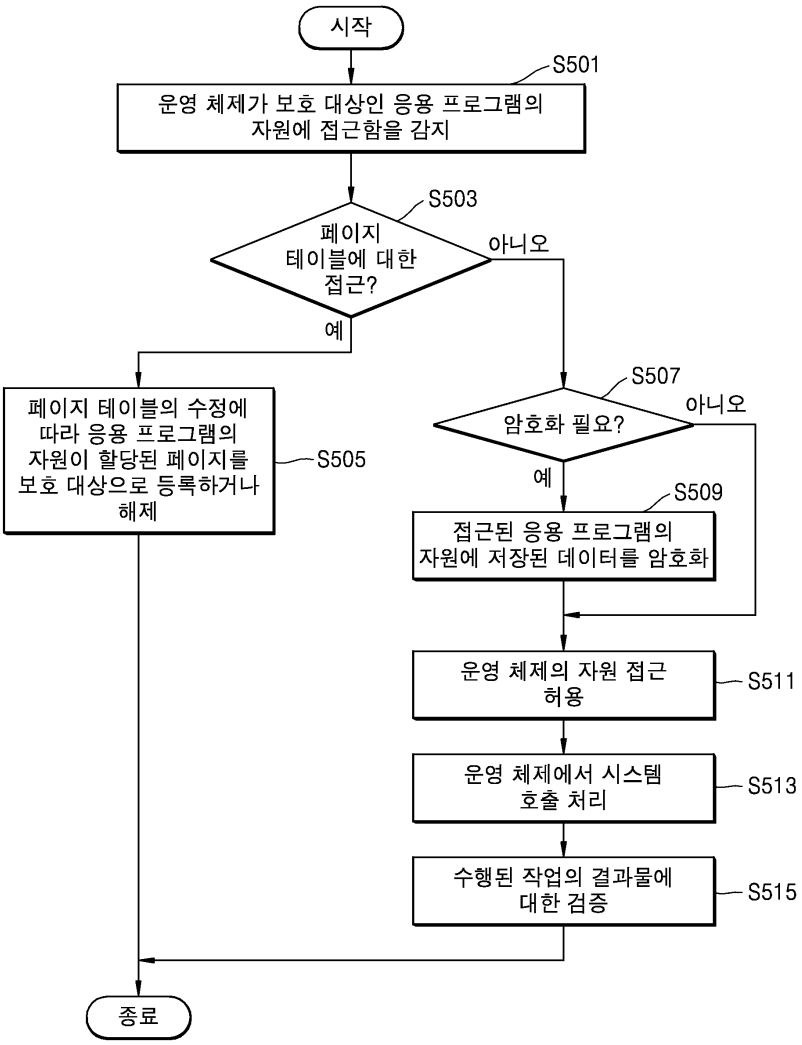
【도 3】


시작
S301
실행 중인 응용 프로그램의 자원에 운영 체제가 접근 가능한 상태인지 여부를 판단
S303
운영 체제가 자원에 접근 가능한 상태?
아니오
예
S305
자원에 대한 접근 권한을 제어
종료

【도 4】


시작
S401
보호 대상으로 설정된 응용 프로그램의 실행 모드 판단
S403
커널 모드?
아니오
예
S405
메모리 페이지에 대한 접근 권한 및 레지스터의 데이터 값 백업
S407
메모리 페이지에 대한 접근 권한을 재설정하고 레지스터의 데이터 값 제거
S411
운영 체제에서 시스템 호출 처리
S413
백업된 값 존재?
아니오
예
S415
메모리 페이지에 대한 접근 권한 및 레지스터의 데이터 값 복구
S417
유저 모드에서, 응용 프로그램의 작업을 수행
종료

【도 5】


시작
S501
운영 체제가 보호 대상인 응용 프로그램의 자원에 접근함을 감지
S503
페이지 테이블에 대한 접근?
아니오
예
S505
페이지 테이블의 수정에 따라 응용 프로그램의 자원이 할당된 페이지를 보호 대상으로 등록하거나 해제
S507
암호화 필요?
아니오
예
S509
접근된 응용 프로그램의 자원에 저장된 데이터를 암호화
S511
운영 체제의 자원 접근 허용
S513
운영 체제에서 시스템 호출 처리
S515
수행된 작업의 결과물에 대한 검증
종료