(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2014년09월22일
(11) 등록번호 10-1434069
(24) 등록일자 2014년08월19일

(51) 국제특허분류(Int. Cl.)
H04L 9/32 (2006.01) *H04L 9/32* (2006.01)
(21) 출원번호 10-2012-7016069
(22) 출원일자(국제) 2010년11월29일
심사청구일자 2012년06월21일
(85) 번역문제출일자 2012년06월21일
(65) 공개번호 10-2012-0096019
(43) 공개일자 2012년08월29일
(86) 국제출원번호 PCT/US2010/058228
(87) 국제공개번호 WO 2011/084265
국제공개일자 2011년07월14일
(30) 우선권주장
12/653,796 2009년12월21일 미국(US)
(56) 선행기술조사문헌
US20030056051 A1*
US20050262361 A1*
US20060236127 A1*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
인텔 코포레이션
미국 캘리포니아주 95054 산타클라라 미션 칼리지 불바드 2200
(72) 발명자
스미스 네드 엠
미국 오레곤주 97124 힐스보로 노스이스트 엘람 영 파크웨이 5200
무어 빅토리아 씨
미국 아리조나주 85044 피닉스 사우스 솔라 캐년 드라이브 14244
그로브만 스티븐 엘
미국 캘리포니아주 95762 엘도라도 힐스 사우스릿지 코트 1511
(74) 대리인
제일특허법인

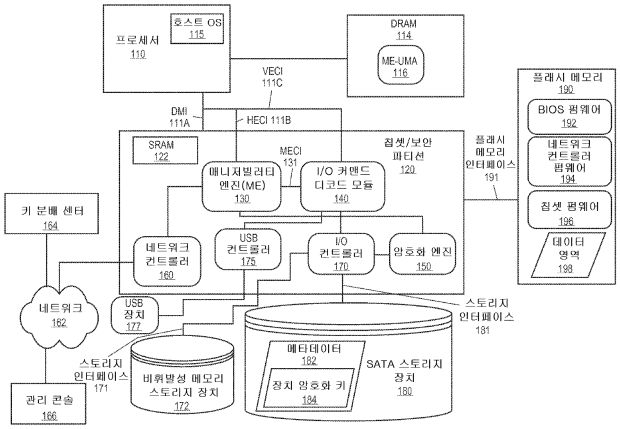
전체 청구항 수 : 총 20 항 심사관 : 심병로

(54) 발명의 명칭 **피보호 장치 관리**

(57) 요 약

암호화, 사용자 인증, 비밀번호 보호, 가상화 및 비 가상화 환경에서의 감사 방식에 의해 보호된 스토리지 장치의 관리를 위한 방법, 장치, 시스템 및 컴퓨터 프로그램 제품.

대 표 도



특허청구의 범위

청구항 1

시스템에 연결된 암호화 장치를 잠금해제하라는 요청을 수신하는 단계 -상기 요청은 신뢰성 있는 원격 콘솔과 상기 시스템의 보안 파티션 사이에 확립된 보안 통신 채널을 통해 상기 보안 파티션에 의해 수신되고, 상기 보안 파티션은 상기 시스템의 호스트 운영 시스템으로부터 분리됨- 와,

상기 보안 파티션이 상기 호스트 운영 시스템의 개입 없이 상기 요청에 응답하여 상기 암호화 장치를 잠금해제하는 단계와,

상기 보안 파티션에 의해 상기 신뢰성 있는 원격 콘솔로부터 토큰을 수신하는 단계와,

상기 암호화 장치의 블럭을 암호화하는 데 사용된 키를 언랩(unwrap)하기 위해 상기 토큰을 사용하는 단계를 포함하는

컴퓨터로 구현된 방법.

청구항 2

삭제

청구항 3

제 1 항에 있어서,

상기 암호화 장치의 보안 스토리지 영역으로부터 상기 키를 획득하는 단계를 더 포함하되,

상기 보안 스토리지 영역은 상기 호스트 운영 시스템으로부터 숨겨지는

컴퓨터로 구현된 방법.

청구항 4

제 1 항에 있어서,

상기 암호화 장치를 잠금해제하기 전에 상기 요청이 상기 신뢰성 있는 원격 콘솔에 의해 발생된 것임을 확인하는 단계를 더 포함하는

컴퓨터로 구현된 방법.

청구항 5

제 1 항에 있어서,

상기 암호화 장치가 잠금해제된 후에 관리 동작을 수행하는 단계 - 상기 요청은 수행될 상기 관리 동작을 더 지정함 - 와,

상기 관리 동작이 수행된 후에 상기 호스트 운영 시스템을 부팅하는 단계를 더 포함하는

컴퓨터로 구현된 방법.

청구항 6

제 1 항에 있어서,

상기 시스템의 상기 호스트 운영 시스템이 오작동하는 경우 상기 암호화 장치의 잠금해제가 수행되는

컴퓨터로 구현된 방법.

청구항 7

제 1 항에 있어서,

상기 시스템의 사용자의 개입 없이 상기 암호화 장치의 잠금해제가 수행되는

컴퓨터로 구현된 방법.

청구항 8

제 1 항에 있어서,

상기 요청은 상기 암호화 장치에 대한 비밀번호를 포함하고,

상기 암호화 장치의 잠금해제는 상기 암호화 장치를 잠금해제하도록 비밀번호를 사용하는 것을 포함하는

컴퓨터로 구현된 방법.

청구항 9

적어도 하나의 프로세서와,

상기 프로세서 상에서 실행되는 호스트 운영 시스템으로부터 분리된 보안 파티션과,

명령어를 포함하는 메모리를 포함하는 장치로서,

상기 명령어는,

상기 보안 파티션에서 실행되는 장치 매니저로 하여금,

상기 장치에 연결된 암호화 장치를 잠금해제하라는 요청을 수신 -상기 요청은 신뢰성 있는 원격 콘솔과 상기 보안 파티션 사이에 확립된 보안 통신 채널을 통해 상기 보안 파티션에 의해 수신되고, 상기 보안 파티션은 상기 호스트 운영 시스템으로부터 분리됨- 하고,

상기 호스트 운영 시스템의 개입 없이 상기 요청에 응답하여 상기 암호화 장치를 잠금해제하고,

상기 보안 파티션에 의해 상기 신뢰성 있는 원격 콘솔로부터 토큰을 수신하고,

상기 암호화 장치의 블럭을 암호화하는 데 사용된 키를 언랩(unwrap)하기 위해 상기 토큰을 사용하는 것을 수행하게 하는

장치.

청구항 10

명령어를 포함하는 컴퓨터 판독 가능한 저장 매체로서,

상기 명령어는, 처리 시스템의 보안 파티션에서 실행될 때, 상기 보안 파티션으로 하여금,

상기 처리 시스템에 연결된 암호화 장치를 잠금해제하라는 요청을 수신 -상기 요청은 신뢰성 있는 원격 콘솔과 상기 보안 파티션 사이에 확립된 보안 통신 채널을 통해 상기 보안 파티션에 의해 수신되고, 상기 보안 파티션은 상기 처리 시스템의 호스트 운영 시스템으로부터 분리됨- 하는 동작과,

호스트 운영 시스템의 개입 없이 상기 요청에 응답하여 상기 암호화 장치를 잠금해제하는 동작과,

상기 보안 파티션에 의해 상기 신뢰성 있는 원격 콘솔로부터 토큰을 수신하는 동작과,

상기 암호화 장치의 블럭을 암호화하는 데 사용된 키를 언랩(unwrap)하기 위해 상기 토큰을 사용하는 동작을 수행하게 하는

컴퓨터 판독 가능한 저장 매체.

청구항 11

신뢰성 있는 원격 콘솔과 시스템의 보안 파티션 사이에 보안 통신 채널을 확립하는 단계 -상기 보안 파티션은 상기 시스템의 호스트 운영 시스템으로부터 분리됨- 와,

상기 시스템에 연결된 암호화 장치를 잠금해제하라는 요청을 송신하는 단계 -상기 요청은 상기 보안 통신 채널을 통해 상기 보안 파티션으로 송신되고, 상기 암호화 장치는 상기 시스템의 호스트 운영 시스템의 개입 없이 상기 보안 파티션에 의해 잠금해제됨- 와,

상기 요청으로 상기 신뢰성 있는 원격 콘솔로부터의 토큰을 상기 보안 파티션에 제공하는 단계를 포함하며,

상기 보안 파티션은 상기 암호화 장치에 저장된 블럭을 복호화하는 데 사용된 키를 언랩하기 위해 상기 토큰을 사용하는

컴퓨터로 구현된 방법.

청구항 12

삭제

청구항 13

제 11 항에 있어서,

상기 요청으로 상기 암호화 장치에 대한 비밀번호를 제공하는 단계를 더 포함하되,

상기 보안 파티션은 상기 암호화 장치를 잠금해제하기 위해 상기 비밀번호를 사용하는

컴퓨터로 구현된 방법.

청구항 14

제 11 항에 있어서,

상기 요청으로 상기 암호화 장치가 잠금해제된 후에 수행될 관리 동작을 지정하는 단계를 더 포함하되,

상기 보안 파티션은 상기 암호화 장치가 잠금해제된 후에 상기 관리 동작을 수행하는

컴퓨터로 구현된 방법.

청구항 15

시스템에 부착된 복수의 장치 중 임의의 장치로의 액세스가 허가되기 전에 상기 시스템의 사용자의 제 1 크리덴셜(credential)을 인증하는 단계와,

상기 시스템에 대한 새로운 장치의 부착을 나타내는 이벤트를 차단하는 단계 -상기 차단하는 단계는 상기 시스템의 보안 파티션에 의해 수행되고, 상기 보안 파티션은 상기 시스템의 호스트 운영 시스템으로부터 분리됨- 와,

상기 새로운 장치에 액세스하기 위해 제 2 크리덴셜을 요청하는 단계 -상기 제 2 크리덴셜은 상기 시스템의 부팅 없이 요청됨- 와,

상기 제 2 크리덴셜을 인증하는 단계와,

상기 제 2 크리덴셜을 인증한 후에 상기 새로운 장치에 액세스할 수 있게 하는 단계와,

상기 새로운 장치에 대한 핫플러그 이벤트를 상기 호스트 운영 시스템으로 전달하는 단계를 포함하며,

상기 새로운 장치에 액세스하기 위해 제 2 크리덴셜을 요청하는 단계는 상기 제 2 크리덴셜에 대한 요청을 표시하는 디스플레이 장치 및 상기 제 2 크리덴셜을 수신하는 사용자 입력 장치에 대한 신뢰성 있는 경로 접속을 사용하는 단계를 포함하는

컴퓨터로 구현된 방법.

청구항 16

삭제

청구항 17

제 15 항에 있어서,

상기 새로운 장치에 액세스할 수 있게 하는 단계는 상기 새로운 장치의 복호화를 가능하게 하기 위해 상기 장치에 대한 자생 커맨드(native command)를 이용하는 단계를 포함하는

컴퓨터로 구현된 방법.

청구항 18

제 15 항에 있어서,

상기 제 2 크리덴셜은 상기 새로운 장치에 대한 비밀번호를 포함하고,

상기 새로운 장치에 액세스할 수 있게 하는 단계는 상기 새로운 장치를 잠금해제하기 위해 상기 비밀번호를 이용하는 단계를 포함하는

컴퓨터로 구현된 방법.

청구항 19

제 15 항에 있어서,

상기 제 2 크리덴셜은 사용자 식별자를 포함하고,

상기 새로운 장치에 액세스할 수 있게 하는 단계는, 신뢰성 있는 제3자가 상기 사용자 식별자를 인증하는 경우 상기 신뢰성 있는 제3자에게 상기 사용자 식별자를 제공하고 상기 새로운 장치에 액세스할 수 있게 하는 단계를 포함하는

컴퓨터로 구현된 방법.

청구항 20

적어도 하나의 프로세서와,

상기 프로세서 상에서 실행되는 호스트 운영 시스템으로부터 분리된 보안 파티션과,

명령어를 포함하는 메모리를 포함하는 장치로서,

상기 명령어는,

상기 보안 파티션에서 실행되는 펌웨어로 하여금,

시스템에 부착된 복수의 장치 중 임의의 장치로의 액세스가 허가되기 전에 상기 시스템의 사용자의 제 1 크리덴셜을 인증하고,

상기 시스템에 대한 새로운 장치의 부착을 나타내는 이벤트를 차단하고 -상기 차단은 상기 보안 파티션에 의해 수행됨- ,

상기 새로운 장치에 액세스하기 위해 제 2 크리덴셜을 요청하고 -상기 제 2 크리덴셜은 상기 시스템의 부팅 없이 요청됨- ,

상기 제 2 크리덴셜을 인증하고,

상기 제 2 크리덴셜을 인증한 후에 상기 새로운 장치에 액세스할 수 있게 하고,

상기 새로운 장치에 대한 핫플러그 이벤트를 상기 호스트 운영 시스템으로 전달하는 것을 수행하게 하며

상기 새로운 장치에 액세스하기 위한 제 2 크리덴셜의 요청은 상기 제 2 크리덴셜에 대한 요청을 표시하는 디스플레이 장치 및 상기 제 2 크리덴셜을 수신하는 사용자 입력 장치에 대한 신뢰성 있는 경로 접속을 사용하여 이루어지는

장치.

청구항 21

명령어를 포함하는 컴퓨터 판독 가능한 저장 매체로서,

상기 명령어는, 처리 시스템의 보안 파티션에서 실행될 때, 상기 보안 파티션으로 하여금,

시스템에 부착된 복수의 장치 중 임의의 장치로의 액세스가 허가되기 전에 상기 시스템의 사용자의 제 1 크리덴셜을 인증하고,

상기 시스템에 대한 새로운 장치의 부착을 나타내는 이벤트를 차단하고 -상기 차단은 상기 보안 파티션에 의해 수행되고, 상기 보안 파티션은 상기 시스템의 호스트 운영 시스템으로부터 분리됨-,

상기 새로운 장치에 액세스하기 위해 제 2 크리덴셜을 요청하고 -상기 제 2 크리덴셜은 상기 시스템의 부팅 없이 요청됨-,

상기 제 2 크리덴셜을 인증하고,

상기 제 2 크리덴셜을 인증한 후에 상기 새로운 장치에 액세스할 수 있게 하고,

상기 새로운 장치에 대한 핫플러그 이벤트를 상기 호스트 운영 시스템으로 전달하는 것을 포함하는 동작을 수행하게 하며,

상기 새로운 장치에 액세스하기 위한 제 2 크리덴셜의 요청은 상기 제 2 크리덴셜에 대한 요청을 표시하는 디스플레이 장치 및 상기 제 2 크리덴셜을 수신하는 사용자 입력 장치에 대한 신뢰성 있는 경로 접속을 사용하여 이루어지는

컴퓨터 판독 가능한 저장 매체.

청구항 22

시스템에 부착된 복수의 장치 중 임의의 장치로의 액세스가 허가되기 전에 상기 시스템의 사용자의 제 1 크리덴셜(credential)을 인증하는 단계와,

상기 시스템에 대한 새로운 장치의 부착을 나타내는 이벤트를 차단하는 단계 -상기 차단하는 단계는 상기 시스템의 보안 파티션에 의해 수행되고, 상기 보안 파티션은 상기 시스템의 호스트 운영 시스템으로부터 분리됨-와,

상기 새로운 장치에 액세스하기 위해 제 2 크리덴셜을 요청하는 단계 -상기 제 2 크리덴셜은 상기 시스템의 부

팅 없이 요청됨- 와,

상기 제 2 크리덴셜을 인증하는 단계와,

상기 제 2 크리덴셜을 인증한 후에 상기 새로운 장치에 액세스할 수 있게 하는 단계와,

상기 새로운 장치에 대한 핫플러그 이벤트를 상기 호스트 운영 시스템으로 전달하는 단계를 포함하며,

상기 새로운 장치에 액세스할 수 있게 하는 단계는 상기 새로운 장치의 복호화를 가능하게 하기 위해 상기 장치에 대한 자생 커맨드(native command)를 이용하는 단계를 포함하는

컴퓨터로 구현된 방법.

청구항 23

삭제

청구항 24

요청하는 시스템의 보안 파티션으로부터 감사 로그를 서비스하라는 요청을 수신하는 단계 -상기 보안 파티션은 상기 요청하는 시스템의 호스트 운영 시스템으로부터 분리되고, 상기 감사 로그는 상기 보안 파티션에서 수행된 감사가능한 이벤트의 감사 이벤트 기록을 포함하고, 상기 감사 로그는 상기 요청하는 시스템의 상기 호스트 운영 시스템으로부터 분리됨- 와,

상기 보안 파티션과의 보안 통신 채널을 확립하는 단계와,

상기 보안 통신 채널을 통해 상기 감사 로그를 서비스하는 단계를 포함하는

컴퓨터로 구현된 방법.

명 세 서

기 술 분 야

[0001] [저작권표시]

[0002] 저작권 보호 대상인 내용이 여기 포함된다. 저작권 소유자는, 본 특허 개시물의 복사가 특허청 파일 또는 기록으로 보인다면 임의의 개인에 의한 그 개시물의 복사에 이의가 없지만, 그렇지 않은 경우 무엇이든 저작권에 대한 모든 권리를 가진다.

[0003] 본 개시물은 일반적으로 암호화(encryption), 사용자 인증 및 비밀번호 보호 방식에 의해 보호된 장치의 관리에 관한 것이다.

배 경 기 술

[0004] 공동의 데이터는 점점 더 유동적이고, 광범위하며, 대량으로 된다. 데이터는 여행하거나 자유로운 작업 습관을 가진 근로자를 수용하기 위해 물리적으로 보증된 시설 밖으로 일상적으로 유출된다. 또한 기업의 사업 이익에 따라 데이터가 다른 도시, 주 및 국가로 이동함에 따라 데이터는 지리적으로 분포된다. 데이터는 생성되는 속도 및 제시될 수 있는 멀티미디어 포맷 모두의 측면에서 대량이다. 이들 모두는 데이터가 전달되는 동안 및 중단되는 동안 모두 보호될 것을 요구하는 새로운 스토리지 매체, 더 높은 대역폭 서브시스템 및 네트워크 접속 스토리지의 발전을 유도한다.

[0005] DAR(data-at-rest) 암호화 기술은 손실 또는 도난 스토리지 장치에 저장된 데이터가 허가없이 사용되는 것을 방지하고, 이에 따라 이들 데이터가 인터넷 또는 다른 네트워크에 확산되는 것을 방지한다. DAR 암호화는 자동화되고 신속한 응답 메커니즘으로 기능하여, 스토리지 장치의 불가피한 손실 및 절도가 그들 장치에 저장된 데이터의 손실 및 절도로 되는 것을 방지한다.

[0006] 컴퓨팅 플랫폼과 연관된 다양한 스토리지 장치에 저장된 데이터를 보호하는 과제 중 하나는 암호화 기술 및 키

관리 계획이 암호화를 수행하는 개체에 따라 다르다는 것이다. 스토리지 하드웨어는 스토리지 하드웨어 판매자 고유의 내장된 암호화 능력을 가질 수 있고, 그 때문에 데이터에 액세스하기 위해 스토리지 하드웨어 판매자의 툴을 사용하는 것이 요구된다. 소프트웨어 기반 암호화는 하드웨어 기반 암호화와 상이한 키 생성 및 관리 서비스를 필요로 하고, 따라서 소프트웨어로 암호화된 데이터에 액세스하기 위해 소프트웨어 판매자의 툴을 사용하는 것이 요구될 수 있다. 그러므로 절도 또는 분실의 경우 키 복구 및 데이터의 이송을 위한 계획은, 컴퓨팅 플랫폼과 연관된 데이터를 모두 보호 및/또는 복구하기 위해, 다수의 상이한 판매자의 툴의 사용을 필요로 할 수 있다.

[0007] 스토리지 장치에 저장된 데이터를 보호하는 다른 과제는 스토리지 장치 자체가 비밀번호 보호 방식을 이용하여 보호될 수 있다는 점이다. 예컨대, ATA(Advanced Technology Attachment) 사양에 따르면, 디스크 잠금(disk lock)은 하드디스크 드라이브의 내장된 보안 특징이다. ATA 사양은 디스크가 사용자 비밀번호 및 마스터 비밀번호의 두 가지 비밀번호를 가질 것을 요구한다. 디스크는 고 보안 모드(High security mode) 또는 최대 보안 모드(Maximum security mode)의 두 가지 모드로 잠길 수 있다. 고 보안 모드에서, 디스크는 "SECURITY UNLOCK DEVICE" ATA 커맨드를 사용하여 사용자 비밀번호 또는 마스터 비밀번호 중 하나에 의해 잠금해제될 수 있다. 디스크가 전력 순환되거나 하드 리셋되어야 한 후 잠금해제가 다시 시도되기 전에 보통 5로 설정된 시도 제한이 있다. 또한 고 보안 모드에서는 SECURITY ERASE UNIT 커맨드가 사용자 비밀번호 또는 마스터 비밀번호 중 하나와 함께 이용될 수 있다.

[0008] 최대 보안 모드에서, 디스크는 사용자 비밀번호 없이 잠금해제될 수 없다. 디스크가 사용 가능 상태로 다시 되게 하는 유일한 방법은 SECURITY ERASE PREPARE 커맨드를 발행하고 직후에 SECURITY ERASE UNIT 커맨드를 발행하는 것이다. 최대 보안 모드에서 SECURITY ERASE UNIT 커맨드는 사용자 비밀번호를 요구하고 디스크 상의 모든 데이터를 완전히 삭제할 것이다. 따라서, 디스크가 비밀번호로 보호되고, 최대 보안 모드로 설정되며 사용자 비밀번호가 비공개이면, 디스크 상의 데이터는 복구불가능하다.

[0009] 컴퓨팅 플랫폼과 연관된 스토리지 장치에 저장된 데이터를 보호하는 또 다른 과제는 연관된 스토리지 장치의 데이터에 대한 액세스가 허용되기 전에 플랫폼이 사용자 크리덴셜(credential)의 인증을 요구할 수 있다는 것이다. 예컨대, 일부 컴퓨팅 플랫폼은 커베로스(Kerberos) 사용자 인증을 이용하여 보호된다. 커베로스는 그 기본으로 대칭적 니드햄 쉬뢰더(Needham-Schroeder) 프로토콜을 사용한다. 이것은 인증 서버(AS) 및 티켓 허가 서버(TGS)의 두 개의 논리적으로 별개의 부분을 포함하는 키 분배 센터(KDC)라 불리는 신뢰성 있는 제3자를 이용하게 한다. 커베로스는 사용자의 신원을 증명하도록 기능하는 "티켓"에 기초하여 작동한다.

[0010] KDC는 비밀 키의 데이터베이스를 유지하고, 클라이언트든 서버든 네트워크의 각각의 개체가 그 자신과 KDC에만 공개된 비밀 키를 공유한다. 이 키를 알고 있는 것은 개체의 신원을 증명하도록 기능한다. 두 개체 사이의 통신을 위해, KDC는 그들 상호작용을 보장하는 데 사용될 수 있는 세션 키를 생성한다. 그 프로토콜의 보안은 느슨하게 동기화된 시간을 유지하는 참가자 및 커베로스 티켓이라 불리는 수명이 짧은 인증 주장에 매우 많이 의존한다.

[0011] 커베로스 프로토콜 하에서는, 클라이언트는 인증 서버에 대해 스스로 인증하고 티켓을 수신한다. (모든 티켓에는 타임스탬프가 찍힌다.) 그러면 클라이언트는 티켓 허가 서버에 접촉하고 티켓을 이용하여 그 신분을 증명하고 서비스를 요구한다. 클라이언트가 서비스에 대해 자격이 있으면, 티켓 허가 서버는 클라이언트에게 다른 티켓을 송신한다. 그 후 클라이언트는 서비스 서버에 접속하고 이 티켓을 이용해서 서비스를 받는 것이 승인되었다는 것을 증명한다.

도면의 간단한 설명

[0012] 도 1은 본 발명의 일 실시예에 따라 암호화, 사용자 신원 인증, 및 비밀번호 보호 방식에 의해 보호된 장치를 관리하도록 구성된 시스템의 블럭도이다.

도 2는 본 발명의 일 실시예에 따른 보호된 장치를 관리하는 경우의 도 1의 시스템을 더 세부적으로 나타내는 도면이다.

도 3은 본 발명의 일 실시예에 따라 암호화, 사용자 신원 인증, 및 비밀번호 보호 방식에 의해 보호된 장치를 갖는 시스템의 리셋에 따라 실행될 방법의 흐름도이다.

도 4는 본 발명의 일 실시예에 따라 암호화, 사용자 신원 인증, 및 비밀번호 보호 방식에 의해 보호된 장치를

잠금해제하는 명령의 수신에 따라 실행될 방법의 흐름도이다.

도 5a는 본 발명의 일 실시예에 따라 암호화, 사용자 신원 인증, 동적으로 부착될 비밀번호 보호 방식, 및 재부팅없이 동적으로 재확인될 사용자 인증 크리덴셜에 의해 장치를 보호 가능하게 하는 경우의 도 1의 시스템을 더 세부적으로 나타내는 도면이다.

도 5b는 장치의 핫플러그 이벤트를 인식하면 도 5a의 시스템에 의해 실행될 방법의 흐름도이다.

도 6은 본 발명의 일 실시예에 따라 보호된 장치를 관리하는 경우의 도 1의 시스템을 더 세부적으로 나타내는 도면이다.

도 7은 본 발명의 일 실시예에 따라 시스템의 보안 파티션 내에서 발생하는 잠재적으로 감사가능한(auditable) 이벤트를 검출하면 수행될 방법의 흐름도이다.

도 8은 본 발명의 일 실시예에 따라 암호화, 사용자 신원 인증 및 비밀번호 보호 방식을 이용하여 장치를 보호하는 등의 동작을 관리하기 위한 보안 파티션을 구현하는 가상머신 환경을 도시하는 도면이다.

발명을 실시하기 위한 구체적인 내용

[0013] 본 발명의 실시예는 암호화, 사용자 신원 인증 및 비밀번호 보호 방식에 의해 보호된 장치를 갖는 시스템을 관리하는 방법, 장치, 시스템 및 컴퓨터 프로그램 제품을 제공할 수 있다.

[0014] 본 명세서에서 본 발명의 "하나의 실시예" 또는 "일 실시예"라고 언급하는 것은 그 실시예와 관련되어 기술된 특정 특징, 구조 또는 특성이 본 발명의 적어도 하나의 실시예에 포함되는 것임을 의미한다. 따라서 명세서 전반에 걸쳐 나타나는 "일 실시예에서", "일 실시예에 따라" 등의 문구의 기재가 반드시 동일한 실시예를 가리키는 것은 아니다.

[0015] 설명을 목적으로 본 발명의 완전한 이해를 제공하기 위해 특정 구성 및 세부사항이 기술된다. 그러나, 본 발명의 실시예는 여기에 제시된 특정 세부사항 없이 실시될 수 있음은 당업자에게 자명할 것이다. 또한 본 발명을 불분명하게 하지 않도록 공지된 특징은 생략 또는 단순화될 수 있다. 다양한 예가 본 명세서 전체에 걸쳐 주어질 것이다. 이들은 단지 본 발명의 특정 실시예의 설명이다. 본 발명의 범위는 주어진 예에 한정되는 것이 아니다.

[0016] 일 실시예에서, 분리 및 제어된 환경을 제공하는 보안 파티션 내에 피보호 장치 관리가 제공된다. 보안 파티션은 신뢰성 있는 관리 애플리케이션으로부터 관리 동작을 수행하기 위한 명령을 수신할 수 있다. 보안 파티션은 피보호 장치를 관리하는 명령이 인증된 소스에 의해 유래되는 것으로 증명되는 것을 보장한다. 신뢰성 있는 관리 애플리케이션은 시스템과 원격일 수 있고, 보안 통신 채널을 통해 보안 파티션과 통신할 수 있다.

[0017] 피보호 장치 매니저의 분리 및 보안 환경은 전적으로 개별적인 하드웨어 파티션(예컨대, 인텔사의 매니저빌러티 엔진(Manageability Engine, "ME"), 활성 관리 기술(Active Management Technologies, "AMT"), 플랫폼 리소스 계층(Platform Recource Layer, "PRL") 및/또는 다른 비슷한 또는 유사한 기술을 이용함) 및/또는 가상화된 파티션(예컨대, 인텔사의 가상화 기술(Virtualization Technology, "VT") 방식의 가상 머신)을 포함하여 여러가지 상이한 타입의 파티션을 포함할 수 있다. 가상화 호스트는 또한 (도 8과 관련하여 더 자세히 설명되는 바와 같이) ME, AMT 및 PRL 기술을 구현하는 데에도 사용될 수 있음은 당업자에게 자명할 것이다.

[0018] 일 실시예에서, 피보호 장치 매니저는 시스템의 호스트 운영 시스템으로부터 분리되는 보안 파티션에서 실행한다. 보안 파티션은 시스템에 연결된 암호화된 장치를 잠금해제하기 위한 요청을 수신할 수 있다. 그 요청은 신뢰성 있는 원격 콘솔 및 보안 파티션 사이에 수립된 보안 통신 채널을 통해 보안 파티션에 의해 수신된다. 보안 파티션은 호스트 운영 시스템의 개입없이 요청에 응답하여 암호화된 장치를 잠금해제한다.

[0019] 보안 파티션은 신뢰성 있는 원격 콘솔로부터의 토큰을 수신하고 암호화된 장치의 블럭을 암호화하는 데 사용된 키를 언랩(unwrap)하기 위해 그 토큰을 사용할 수 있다. 보안 파티션은 암호화된 장치의 보안 스토리지 영역으로부터 키를 획득할 수 있고, 보안 스토리지 영역은 호스트 운영 시스템으로부터 숨겨진다. 보안 파티션은 그 요청이 암호화된 장치를 잠금해제하기 전에 신뢰성 있는 원격 콘솔에 의해 비롯되었음을 확인할 수 있다. 보안 파티션은 암호화된 장치가 잠금해제된 후에 관리 동작을 수행할 수 있고, 요청은 수행될 관리 동작을 더 특정하고 관리 동작이 수행된 후에 호스트 운영 시스템을 부팅할 수 있다. 시스템의 호스트 운영 시스템이 오작동하는 경우 시스템의 사용자의 개입없이 암호화된 장치의 잠금해제가 수행될 수 있다.

[0020] 도 1은 본 발명의 일 실시예에 따라 암호화, 사용자 신원 인증 및 비밀번호 보호 방식에 의해 보호된 장치를 관

리하도록 구성된 시스템을 도시한다. 플랫폼(100)은 데스크탑 관리 인터페이스(DMI)(111a)를 통해 칩셋/보안 파티션(120)에 접속된 프로세서(110)를 포함한다. 칩셋/보안 파티션(120)은 플랫폼(100)의 구성 및 동작을 관리하도록, 마이크로프로세서로서 구현될 수 있는 매니저빌러티 엔진(ME)(130)을 포함한다. 일 실시예에서, 매니저빌러티 엔진(ME)(130)은 감사(audit) 이벤트를 수집하고, 사용자를 인증하고, 주변 장치에 대한 액세스를 제어하고, 플랫폼(100)의 스토리지 장치에 저장된 데이터를 보호하기 위한 암호화 키를 관리하고, 네트워크 컨트롤러(160)를 통해 관리 콘솔(166)과 인터페이스한다. 관리 콘솔(166)을 사용하여, 매니저빌러티 엔진(ME)(130)은 플랫폼(100) 등의 플랫폼의 구성과 관리를 위해 전사적(enterprise-wide) 정책과의 일관성을 유지한다. 매니저빌러티 엔진(ME)(130)은 호스트 내장 컨트롤러 인터페이스(HECI)(111b)를 통해 프로세서(110)에 접속된다.

[0021] 가상화 엔진 컨트롤러 인터페이스(VECI)(111c)는 칩셋/보안 파티션(120)의 I/O 커맨드 디코드 모듈(140)에 프로세서(110)를 접속한다. 일 실시예에서, I/O 커맨드 디코드 모듈(140)은 스토리지 커맨드 디코딩 및 다른 가속화 동작을 수행하기 위해 특정 펌웨어를 이용하여 구성되는 범용 컨트롤러이다. I/O 커맨드 디코드 모듈(140)의 기능은 전적으로 특수 목적의 하드웨어로 구현될 수도 있다. I/O 커맨드 디코드 모듈(140)은 플랫폼(100)과 연관된 스토리지 장치에 기록된 데이터를 보호하는 관리 기능을 제공한다. 예컨대, I/O 커맨드 디코드 모듈(140)은 스토리지 장치를 암호화하고, 스토리지 장치를 보호하기 위해 사용된 메타데이터를 보호하고, 스토리지 장치에 관련된 하드웨어 인터럽트를 차단 및 처리하고, 스토리지 장치 상의 관리 동작을 용이하게 하기 위해 암호화 엔진(150)과 상호작용할 수 있다.

[0022] 매니저빌러티 엔진(ME)(130)은 정책 및 암호화 키를 구성함으로써 I/O 커맨드 디코드 모듈(140) 및 암호화 엔진(150)의 행동을 제어한다. 매니저빌러티 엔진(ME)(130), I/O 커맨드 디코드 모듈(140) 및 암호화 엔진(150)의 동작은 이하에 더욱 상세히 설명한다.

[0023] 플랫폼(100)은 동적 랜덤액세스 메모리(DRAM)(114), 칩셋/보안 파티션(120) 내의 정적 랜덤액세스 메모리(SRAM)(122) 및 플래시 메모리(190) 등의 메모리 소자를 더 포함한다. 플랫폼(100)이 최대로 가동되면, 상부 메모리 영역(upper memory area, UMA)으로 지칭되는 DRAM(114)의 일부인 ME-UMA(116)가 매니저빌러티 엔진(ME)(130)에 의해 사용 가능하다. 일반적으로 기본 입출력 시스템(Basic Input Output System, BIOS)에 의해 구성되는 메모리 분리 메커니즘 때문에, 플랫폼(100)용 호스트 운영 시스템(115)은 ME-UMA(116)에 액세스할 수 없다. 이 메모리 분리 메커니즘은 호스트 운영 시스템이 동작하기 전에 ME-UMA 메모리(116)에 대한 액세스를 못하게 한다. 매니저빌러티 엔진(130)에 의해 사용되는 DRAM(114)의 이 부분을 호스트 운영 시스템으로부터 분리시킴으로써, 매니저빌러티 엔진(130)의 무결성이 호스트 운영 시스템(115)을 감염시킬 수 있는 바이러스 또는 다른 악성 소프트웨어로부터 보호된다.

[0024] 플래시 메모리(190)는 플랫폼(100)을 초기화하는 데 사용된 펌웨어를 포함한다. 이 초기화 펌웨어는 BIOS 펌웨어(192), 네트워크 컨트롤러(160)를 구성하는 네트워크 컨트롤러 펌웨어(194), 칩셋/보안 파티션(120)을 구성하는 칩셋 펌웨어(196)를 포함한다. 매니저빌러티 엔진(130) 및 I/O 커맨드 디코드 모듈(140)을 위한 칩셋 펌웨어(196)의 무결성은 플래시 메모리(190)에 저장되기 전의 디지털 서명에 의해 보장된다. 사용자 인증 정보 등의 매니저빌러티 엔진(ME)(130)에 의해 사용되는 데이터는 매니저빌러티 엔진(ME)(130) 내의 암호화 펌웨어에 의해 암호화되고, 플래시 메모리(190)의 데이터 영역(198)에 저장될 수 있다.

[0025] 도 1에 도시된 플랫폼(100)의 실시예는 USB 장치(177)에 접속된 범용 직렬 버스(USB) 컨트롤러(175)를 더 포함한다. USB 장치는 (마우스 등의) 포인팅 장치, 키보드, 디지털 카메라, 프린터, 개인 미디어플레이어, 플래시 드라이브 및 외부 하드드라이브를 포함할 수 있다. USB 사양은 컴퓨터 케이스를 열지 않고(핫스와핑(hotswapping)), 또는 컴퓨터를 재시작하지 않고 장치의 설치 및 제거를 가능하게 하여, 이것을 다양한 종류의 드라이브를 포함한 모바일 주변 장치에 이용 가능하게 한다. 광학 스토리지 장치(CD-RW 드라이브, DVD 드라이브 등)에 대해 본래 고안되고 여전히 요즘도 사용되고 있지만, 다수의 제조업자는 내부 드라이브와 비슷한 성능을 제공하는 외부 휴대형 USB 하드드라이브 또는 디스크 드라이브용의 빈 인클로저를 제공하는데, 이는 부착된 USB 장치의 현재 커런트 넘버 및 타입, 그리고 USB 인터페이스의 상한(upper limit)(실제로 USB 2.0에 대해 대략 40MiB/s 및 대개 USB 3.0에 대해 잠재적으로 400MiB/s 이상)에 의해 제한된다. 이들 외부 드라이브는 일반적으로 드라이브의 인터페이스 사이(IDE, ATA, SATA, PATA, ATAPI 또는 심지어 SCSI)를 USB 인터페이스 포트에 연결하는 "해석 장치(translating device)"를 포함한다. 기능적으로, 드라이브는 사용자에게 대부분 내부 드라이브처럼 보인다. 외부 드라이브 접속을 위한 다른 경쟁할만한 표준은 eSATA, ExpressCard(지금은 버전 2.0) 및 FireWire(IEEE 1394)를 포함한다.

[0026] 도 1에 도시된 플랫폼(100)의 실시예는 스토리지 인터페이스(171)를 통해 액세스 가능한 비휘발성 메모리 스토리지 장치(172) 및 스토리지 인터페이스(181)를 통해 액세스 가능한 SATA(Serial Advanced Technology Attachment) 스토리지 장치(180)를 포함하여 I/O 컨트롤러(170)를 통해 액세스 가능한 상이한 타입의 스토리지 장치를 더 포함한다. 스토리지 인터페이스(171)는 비휘발성 메모리(NVM)를 위한 비휘발성 메모리 호스트 컨트롤러 인터페이스(HCI)로서 구형될 수 있고, 스토리지 인터페이스(181)는 SATA 스토리지 장치(180)를 위한 개선된 HCI(AHCI)로서 구현될 수 있다. I/O 컨트롤러(170)는 기능적으로 NVM 및 SATA 컨트롤러 모두를 포함한다.

[0027] 스토리지 장치(172, 180)에 저장된 데이터는 칩셋/보안 파티션(120)의 암호화 엔진(150)에 의해 암호화될 수 있다. SATA 스토리지 장치(180)는 칩셋 암호화 장치의 일례이고, 메타데이터(182)를 저장하는 예약 영역을 더 포함하며, 스토리지 장치(180)용의 적어도 하나의 장치 암호화 키(DEK) 및 매니저빌러티 엔진(ME)(130)에 의해 사용된 다른 메타데이터를 포함한다. 메타데이터(182)는 I/O 커맨드 디코드 모듈(140) 및 I/O 컨트롤러(170)에 의한 I/O 커맨드의 처리중에 프로세서(110)에서 실행되는 애플리케이션에 의해 재기록되는 것으로부터 보호된다.

[0028] 일 실시예에서, 데이터의 암호화 또는 복호화(decryption)가 칩셋/보안 파티션(120)의 암호화 엔진(150)에 의해 수행되기 전에, 매니저빌러티 엔진(ME)(130)은 입력/출력 동작에 관계된 스토리지 장치와 연관된, DEK(184) 등의 장치 암호화 키(DEK)를 암호화 엔진(150)과 연관된 메모리 레지스터에 삽입한다. 하나의 물리적 스토리지 장치가 다수의 상이한 논리적 장치 또는 파티션으로 논리적으로 분할되면, 각각의 논리적 장치 또는 파티션은 그 자신의 각각의 장치 암호화 키(DEK)를 가질 수 있고, 그 DEK 각각은 암호화 엔진(150)의 각각의 메모리 레지스터에 삽입될 수 있다.

[0029] 일 실시예에서, 매니저빌러티 엔진(ME)(130)은, 칩셋에 의해 수행되는 것이 아니라 대신에 프로세서(110) 상에서 실행되는 소프트웨어 또는 스토리지 하드웨어 자체에 의해 수행되는 데이터의 암호화 뿐만 아니라, 칩셋/보안 파티션(120) 내의 암호화 엔진(150)에 의해 수행되는 암호화를 포함하여 플랫폼(100)과 연관된 모든 데이터의 암호화를 관리한다. 매니저빌러티 엔진(ME)(130)에 의해 제공된 서비스 중 하나는, 데이터의 암호화를 수행하는 플랫폼(100)의 구성요소에 관계없이, 공통 프레임워크 및 사용자 인터페이스의 암호화 키의 관리이다. 데이터의 암호화를 관리함에 있어 칩셋/보안 파티션(120) 및 매니저빌러티 엔진(ME)(130)의 프레임워크 및 동작에 대한 보다 상세한 사항은, 발명자가 네드 스미스(Ned Smith)인 "Enforcing Use of Chipset Key Management Services for Encrypted Storage Devices"라는 제목의 미국 특허출원 제12/319,210에서 제공되며, 그 전체가 참조를 위해 여기에 포함된다.

[0030] 도 2는 본 발명의 일 실시예에 따라 도 1의 칩셋/보안 파티션(120)의 매니저빌러티 엔진(ME)(130) 및 I/O 커맨드 디코드 모듈(140) 구성요소를 보다 상세히 도시한다. 칩셋/보안 파티션(120) 내에서, 매니저빌러티 엔진(ME)(130)은 ME 커널(231), ME 공통 서비스(233), 피보호 장치 매니저(235), 보안/키 관리 펌웨어(237) 및 신원 관리 펌웨어(239)를 포함한다. 이들 구성요소의 각각은 이하에 더 상세히 논의된다.

[0031] ME 커널(231)은 SRAM(122) 및 DRAM(112)(ME-UMA(114) 등)의 일부의 메모리 사용, 플래시 메모리(190)로의 지속적인 데이터 저장 및 액세스 제어를 포함하여 기본적인 기능을 제공한다. ME 커널(231)은 I/O 커맨드 디코드 모듈(140) 및 암호화 엔진(150)의 동작을 제어한다.

[0032] ME 공통 서비스(233)는 상이한 펌웨어 모듈에 의해 공통으로 필요한 서비스를 포함하고, 보안 서비스, 네트워킹 서비스 및 권한설정(provisioning) 서비스를 포함한다. ME 공통 서비스(233)에 의해 제공된 보안 서비스는 일반적으로 HTTP 다이제스트 및 커베로스 인증으로 이루어지는 사용자 인증, 마이크로소프트 액티브 디렉토리 및/또는 다른 서비스를 이용하는 도메인 인증, 클라이언트와 서버 클럭을 동기화하는 클럭 동기화 서비스, 및 보안 감사 서비스를 포함한다.

[0033] ME 공통 서비스(233)에 의해 제공된 네트워킹 서비스는 TCP/IP(Transmission Transport Protocol/Internet Protocol) 스택, TLS(Transport Layer Security), HTTP(Hypertext Transport Protocol), SOAP(Simple Object Access Protocol), WS-MAN(Web Services for Manageability) 및 LMS(Local Manageability Service)라 불리는 호스트 기반 TLS 인터페이스를 포함한다.

[0034] ME 공통 서비스(233)에 의해 제공된 권한설정 서비스는 플랫폼(100)에 기업 소프트웨어를 제공하기 위해 도 1의 관리 콘솔(166)과 연관되어 사용된다. 이들 권한 설정 서비스는 제로터치(zero touch) 및 원터치(one touch)의 두 가지 배치 모드를 지원한다. 제로터치 모드에서, 배치 인증 앵커 키(deployment certificate anchor key)는 도 1의 플래시 메모리(190)의 데이터 영역(198) 등의 데이터 스토리지 영역에 저장되며, 공지된 인증 기관

키가 플랫폼의 소유권을 획득하는 데 사용될 수 있는 IT 크리덴셜을 유효하게 하기 위해 사용되게 한다. 원터치 모드는 유기적 인증, 대칭 키 및 설정과 배치 임무를 원격으로 완료하는 데 사용될 수 있는 신뢰성 있는 호스트를 구성한다.

[0035] 매니저빌러티 엔진(130)은 또한 아웃오브밴드(OOB) 통신 모듈(230)을 포함한다. OOB 통신 모듈(230)은 플랫폼(100)의 구성요소 사이에서 네트워크 컨트롤러(160)를 통해 관리 콘솔(166)의 대응하는 구성요소와의 통신을 용이하게 한다. OOB 통신 모듈(230)은 칩셋/보안 파티션(120)과 관리 콘솔(166) 사이의 보안 OOB 통신 채널(168)을 확립한다.

[0036] 매니저빌러티 엔진(130)은 또한 신원 관리 펌웨어(239)를 포함한다. 신원 관리 펌웨어(239)는, 예컨대, 플래시 메모리(190)의 데이터 영역(198)에 저장된 사용자 계정 메타데이터와 사용자의 인증 정보를 비교할 수 있다. 신원 관리 펌웨어(239)는 또한 사용자의 정보가 SATA 스토리지 장치(180) 등의 스토리지 장치 내의 수용부에도 저장되어 있는 것을 확인하기 위해 매니저빌러티 엔진(130)의 보안/키 관리 펌웨어(237)와도 상호작용할 수 있다. SATA 스토리지 장치(180) 등의 특정 스토리지 장치에 대한 사용자의 액세스의 이러한 확인은 SATA 스토리지 장치(180)에 저장된 데이터의 보호의 추가적인 계층을 제공한다.

[0037] 보안/키 관리 펌웨어(237)는 암호화 엔진(150)에 의해 생성된 암호화 키 등의 키를 관리한다. 보안/키 관리 펌웨어(237)는 또한 플랫폼(100)과 연관된 스토리지 장치에 저장된 데이터에 대한 액세스가 허용되기 전에 사용자를 인증할 수 있다. 보안/키 관리 펌웨어(237)는 키 관리 정보를 관리하고, 이 키 관리 정보를 플래시 메모리(190) 또는 SATA 스토리지 장치(180) 등의 플랫폼과 연관된 메모리 또는 스토리지 장치에 저장한다. 키 관리 정보가 저장되는 위치는 이용가능한 스토리지 공간 및 저장될 데이터의 양에 따라 달라지고, 본 발명은 키 관리 정보를 저장하는 특정 구성에 한정되지 않는다. 일 실시예에서, 보안/키 관리 펌웨어(237)는 플랫폼(100)과 결합되는 플랫폼 컨테이너 키(PCK)를 이용하여 키 관리 정보를 암호화한다.

[0038] 보안/키 관리 펌웨어(237)에 의해 관리된 키 관리 정보는 칩셋에 의해(즉, 칩셋/보안 파티션(120) 내의 암호화 엔진(150)에 의해) 생성되고 메타데이터(182)에 저장된, 장치 암호화 키(DEK)(184)라고 불리는 암호화 키를 포함한다.

[0039] 매니저빌러티 엔진(130)은 또한 피보호 장치 매니저(235)를 포함하는 것으로 더 도시된다. 일 실시예에서, 피보호 장치 매니저(235)는 SATA 스토리지 장치(180) 등의 장치를 잠금해제하는 데 사용된 장치 비밀번호를 공급하기 위해 I/O 커맨드 디코드 모듈(140)과 통신한다. 피보호 장치 매니저(235)의 동작은 도 3 및 도 4와 관련하여 이하에 더 상세히 설명한다.

[0040] I/O 커맨드 디코드 모듈(140)은 I/O 모듈 커널(241) 및 SATA 가상화 펌웨어(243)를 포함하는 것으로 도시된다. I/O 모듈 커널(241)은 I/O 커맨드 디코드 모듈(140)에 기본 기능을 제공하고 ME 커널(231)로부터 커맨드를 수신한다. SATA 가상화 펌웨어(243)가 본 실시예와 관련하여 펌웨어로서 설명되었지만, SATA 가상화 펌웨어(243)의 기능은 전용 하드웨어로 구현될 수도 있다. SATA 가상화 펌웨어(243)는 SATA 스토리지 장치(180) 등의 SATA 스토리지 장치를 액세스하는 데 사용되고, 매니저빌러티 엔진(ME)(130)이 장치 관리 기능을 수행할 수 있게 한다. 예컨대, SATA 가상화 펌웨어(243)는 호스트 운영 시스템(115) 또는 프로세서(110) 상에서 실행되는 다른 호스트 소프트웨어의 개입없이 SATA 제어 패킷을 I/O 데이터 스트림으로 투입함으로써 관리 콘솔(166)에 의해 피보호 장치의 원격 액세스를 가능하게 한다. 제어 패킷은, 예컨대, 관리 콘솔(166)로부터의 커맨드를 통해 SATA 스토리지 장치(180)를 잠금해제하는 데 사용될 수 있다.

[0041] SATA 가상화 펌웨어(243)는 또한 SATA 스토리지 장치(180) 상의 선형 블럭 어드레스의 범위를 호스트 운영 시스템(115)으로부터 숨기는 데 사용된다. 여기서 호스트 운영 시스템 액세스로부터 숨겨진 이 선형 블럭 어드레스의 범위는, 장치 메타데이터(182)가 드라이브에 저장될 수 있도록 보호되는 보안 스토리지 영역이라고 불린다. 장치 메타데이터(182)는 SATA 스토리지 장치(180)의 블럭의 암호화 및 복호화를 가능하게 하는 장치 암호화 키(184)를 포함할 수 있다.

[0042] SATA 가상화 펌웨어(243)는 또한 새로운 장치가 플랫폼(100)에 부착되는 경우 생성되는 핫플러그 인터럽트 등의, I/O 컨트롤러(170)에 의해 검출된 이벤트를 차단할 수 있다. SATA 가상화 펌웨어(243)는 또한 스토리지 장치로의 그리고 스토리지 장치로부터의 I/O 스트림을 모니터링하고 감사를 위한 이벤트를 검출할 수 있다.

[0043] 일 실시예에서, SATA 스토리지 장치(180)는 암호화 및 ATA 비밀번호 등의, 장치가 액세스되기 전에 사용자에 의해 입력되어야 하는 비밀번호 모두를 사용하여 보호된다. 비밀번호는 SATA 스토리지 장치(180)의 자생(native) 잠금 메커니즘을 잠금해제하는 데 사용된다. 암호화 엔진(150)은 SATA 스토리지 장치(180)의 블럭을 암호화하

는 데 사용된다. DEK(184) 등의 SATA 장치 암호화 키(DEK)는 호스트 운영 시스템으로부터 숨겨지는 위치의 SATA 장치에 저장된다. 장치는 DEK가 암호화된 블럭을 복호화하기 위해 액세스될 수 있기 전에 먼저 자생 잠금 메커니즘에 대한 비밀번호를 이용하여 잠금해제된다.

[0044] 플랫폼(100)이 리셋되면, I/O 커맨드 디코드 모듈(140) 및 매니저빌러티 엔진(ME)(130)은 장치로부터 암호화 키 및 사용자 인증 크리덴셜 등의 장치 메타데이터를 판독하고, 플래시 메모리(190)의 데이터 영역(198) 등의 보안 스토리지에 장치 메타데이터를 장치 메타데이터(298)로서 저장하도록 협력한다. 일 실시예에서, 관리 콘솔(166)에 대해 인증될 수 있는 각각의 사용자에 대해, 특정 장치에 대한 래핑 키(wrapping key)를 유도하는 데 사용되는, 도 2의 토큰-1(266A) 등의 토큰이 있다. 장치 래핑 키는 그 특정 장치의 암호화 키를 래핑하는 데 차례로 사용된다.

[0045] 사용자 래핑 키 및 장치 래핑 키는 사용자가 특정 장치에 액세스할 수 있는 지 여부를 결정하는데 함께 사용되고, 사용자 래핑 키/장치 래핑 키 쌍은 플래시 메모리(190)의 데이터 영역(198)에 장치 메타데이터(298)로서 저장된다. 반대로, DEK(184) 등의 장치 암호화 키는 스토리지 장치 자체에 저장된다. 장치가 액세스될 때, 토큰-1의 사본이 장치 메타데이터(298)로부터 적절한 사용자 래핑 키/장치 래핑 키 쌍을 결정하는 데 사용된다. 그 쌍의 장치 래핑 키는 장치 상의 메타데이터(182)를 복호화하는 데 사용되며 장치 암호화 키(184)를 노출시킨다. 토큰-1은 사용자가 존재하지 않는 경우 또는 사용자가 필요한 인증 크리덴셜을 생성할 수 없는 경우에 스토리지 장치에서 관리 동작을 수행하는 데 사용된다.

[0046] 일 실시예에서, 장치 메타데이터(182)는 USB 장치(177) 등의 USB 장치에 저장되는, 여기서 토큰-2라고 불리는 또 다른 토큰에 의해 유도된 또 다른 장치 래핑 키를 이용하여 암호화 엔진(150)에 의해 암호화된다. USB 장치(177)는 절도범이 접근할 수 있는 곳에서 물리적으로 떨어져 있는 위치에 안전하게 저장되도록 의도된다. 토큰-2는 스토리지 장치에 접속하기 위해 원격 관리 콘솔(166)에 대해 이용 가능한 네트워크 접속이 없을 때 스토리지 장치에서 관리 동작을 수행하는 데 사용된다. 토큰-2의 홀더가 스토리지 장치에서 관리 동작을 수행하도록 암시적으로 인증되도록, USB 장치(177)는 암호화되지 않은 평문 형식인 토큰-2를 포함한다. 토큰-2가 제공되면, 사용자 래핑 키의 제 2 세트는 토큰-2를 이용하여 유도되고, 사용자 래핑 키/장치 래핑 키 쌍의 제 2 세트 또한 장치 메타데이터(298)의 일부로서 플래시 메모리(190)에 저장될 수 있을 것이다. 인증된 사용자만이 장치 암호화 키를 접하게 할 수 있도록, 토큰-1 및 토큰-2 값 모두 신원 관리 펌웨어(239) 등의 사용자 인증 시스템에 의해 보호된다.

[0047] 일 실시예에서, 토큰-1(266A)은 장치(180) 비밀번호(266B)와 함께, 관리 콘솔(166)(또는 디렉토리 서비스)과 연관된 원격 스토리지(266)에서 안전하게 보관된다. 장치(180) 비밀번호(266B) 및 토큰-1(266A)은 SATA 장치(180)를 원격으로 잠금해제하기 위해 관리 콘솔(166)에 의해 사용된다. 장치(180) 비밀번호(266B)는 원격 관리 콘솔(166)에 의해 피보호 장치 매니저(235)에 제공되고, 이것은 장치를 잠금해제하기 위해 장치(180) 비밀번호(266B)를 이용한다. 피보호 장치 매니저(235)는 보안/키 관리 펌웨어(237)에 토큰-1(266A)을 제공하는데, 이것은 사용자 래핑 키를 언랩(unwrap)하기 위해 신원 관리 펌웨어(239)를 사용하게 할 수 있다. 사용자 래핑 키는 장치 래핑 키를 언랩하는 데 사용되고, 이는 메타데이터(182)를 복호화하는 데 사용되며, 이에 따라 SATA 스토리지 장치(180)의 블럭을 복호화하기 위해 암호화 엔진(150)에 의해 사용될 수 있는 장치 암호화 키(184)에 대한 액세스를 제공한다. 토큰-1(266A)은 관리 콘솔(166)과 칩셋/보안 파티션(120) 사이에서 OOB 통신 채널(168) 등의 보안 통신 채널에 의존함으로써 네트워크 공격으로부터 보호된다. OOB 통신 채널(168)은, 예컨대, 커베로스 세션 키를 이용하여 보호될 수 있다.

[0048] SATA 스토리지 장치(180) 상의 데이터가 암호화될 수 있기 때문에, 장치 암호화 키(DEK)(184)는 매니저빌러티 엔진(ME)(130)의 피보호 장치 매니저(235)에 의해 액세스 가능한 위치에 저장된다. DEK를 피보호 장치 매니저(235)에 이용 가능하게 함으로써, SATA 스토리지 장치(180) 상의 데이터가 암호화되는 경우에도 HECI/VECI 인터페이스(111b, 111c)를 통한 SATA 판독/기록 요청이 서비스될 수 있다. 피보호 장치 매니저(235)가 SATA 스토리지 장치(180)를 잠금해제하기 위해 비밀번호에 액세스하면, 피보호 장치 매니저(235)는 장치 메타데이터(298)에 저장되는 장치 래핑 키를 복사할 수 있다. 장치 래핑 키는 플랫폼에 부착된 각 SATA 장치에 대한 장치 메타데이터에 포함된 장치 암호화 키를 언랩하는 데 사용될 수 있다.

[0049] 도 3은 본 발명의 일 실시예에 따라 암호화, 사용자 신원 인증 및 비밀번호 보호 방식에 의해 보호된 장치를 갖는 시스템의 리셋에 따라 수행되는 방법의 흐름도이다. 본 방법이 그러한 구현에 한정되는 것이 아니지만, 도 3의 방법은 도 2의 시스템의 구성요소에 의해 수행되는 것으로 기술될 것이다. 시스템이 리셋되면, 제어는 단계 310의 "ME 피보호 장치 매니저는 MECI를 통한 I/O 커맨드 디코드 모듈을 통해 간접적으로 SATA 장치로부터

장치 메타데이터를 판독한다"로 진행한다. 단계 310에서, 매니저빌러티 엔진(130)의 피보호 장치 매니저(235)는 SATA 스토리지 장치(180) 등의 SATA 장치로부터 장치 메타데이터를 판독함으로써 부착된 스토리지 장치에 대한 정보를 획득한다. 매니저빌러티 엔진(130)이 스토리지 장치에 직접 접속되지 않기 때문에, 매니저빌러티 엔진(130) 피보호 장치 매니저(235)는 MECI(131)를 거쳐 I/O 커맨드 디코드 모듈(140)을 통해 간접적으로 장치 메타데이터에 액세스한다. 매니저빌러티 엔진(130) 피보호 장치 매니저(235)는 SATA 스토리지 장치(180) 등의 SATA 장치에 저장된 장치 메타데이터에 액세스하기 위해 SATA 가상화 펌웨어(243)를 사용한다. SATA 스토리지 장치(180)가 선형 블럭 어드레스의 블럭 스토리지 장치로서 나타나도록, SATA 가상화 펌웨어(243)는 피보호 장치 매니저(235)에 스토리지 인터페이스를 노출시킨다. SATA 가상화 펌웨어(243)는 호스트 운영 시스템으로부터 선형 블럭 어드레스의 일부를 숨기고 피보호 장치 매니저(235)에 그들을 노출시킨다. SATA 가상화 펌웨어(243)는 SATA I/O 프로토콜을 이용하여 SATA 스토리지 장치(180)와 상호작용한다.

[0050] 단계 310의 "ME 피보호 장치 매니저는 MECI를 통한 I/O 커맨드 디코드 모듈을 통해 간접적으로 SATA 장치로부터 장치 메타데이터를 판독한다"로부터, 제어는 단계 320의 "I/O 커맨드 디코드 모듈은 메타데이터 기술자 정보를 포함하는 가상 드라이브 정의 메타데이터를 판독한다"로 진행한다. 단계 320에서, I/O 커맨드 디코드 모듈(140)의 SATA 가상화 펌웨어(243)는 SATA 스토리지 장치(180)에 저장된 메타데이터(182)에 저장되는 메타데이터 기술자 정보를 포함하는 가상 드라이브 정의 메타데이터를 판독한다. 일 실시예에서, 다수의 가상 드라이브 파티션이 인식될 수 있도록, SATA 가상화 펌웨어(243)는 스토리지 장치를 가상화한다. 이들 가상 드라이브 파티션의 각각은 가상 드라이브 정의 데이터에 기술된다. 제 1 가상 하드디스크 드라이브(HDD) 정의 내에 포함된 것은 전통적인 드라이브 기하학적 구성요소일 수 있다. 예컨대, 선형 블럭 어드레스(LBA)가 제로로 시작하면, 마스터 부트 레코드(MBR)가 저장될 수 있고, 뒤이어 운영 시스템 파일 및 사용자 파일 등의 드라이브 데이터가 저장될 수 있다. 일부 시스템은 BIOS 또는 다른 시스템 유틸리티에 의해 사용될 수 있는 숨겨진 파티션을 갖는다. 호스트 피보호 영역(HPA)은 비상 복구 OS(ROS), 멀티미디어 유틸리티, 진단 유틸리티 또는 다른 프로그램을 저장하는 데 사용될 수 있다. RAID(Redundant Arrays of Inexpensive Disks)를 구현하는 시스템은 가상 드라이브의 끝에서 RAID 메타데이터를 배치할 수 있다. 가상 드라이브의 끝에 RAID 메타데이터를 배치함으로써, RAID 선택적 ROM은 시스템 초기화시 RAID 메타데이터를 쉽게 위치시킬 수 있다.

[0051] 일 실시예에서, DEK(184) 등의 하나의 장치 암호화 키는 장치 상의 각각의 가상 HDD를 확장시키고, 그 결과 모든 가상 HDD가 동일한 키로 암호화된다. 가상 드라이브 정의(VDD) 데이터는 마지막 선형 블럭 어드레스 LBA-n 등의 물리적 드라이브의 끝에 배치된다. VDD 데이터는 각 가상 HDD의 시작과 끝을 표시하는 드라이브 기하학 구조를 포함한다. VDD는 또한 메타데이터(182)의 시작과 끝의 위치와 같은 매니저빌러티 엔진 메타데이터 영역의 시작 및 끝의 위치를 식별한다. VDD 및 ME 메타데이터는 암호화 엔진(150)에 의해 암호화될 수 없지만 메타데이터(182)의 내용이 I/O 커맨드 디코드 모듈(140) 및 매니저빌러티 엔진(ME)(130)에 의해 보호된다.

[0052] 일 실시예에서, 메타데이터(182)는 AHCI 파일 시스템 블럭, 프리부트(pre-boot) 인증(PBA) 코드, PBA 메타데이터를 포함한다. AHCI 파일 시스템은 펌웨어 스토리지 드라이버에 의해 사용되는데, 이는 프로세서(110)에 의해 실행될 수 있다. 메타데이터(182)는 또한 래핑된 DEK(184), 장치 구성 데이터, 드라이브 변환 상태 정보 및 SATA 스토리지 장치(180) 등의 장치를 다른 플랫폼으로 이송하는 데 사용될 수 있는 드라이브 이송 패키지를 포함할 수 있다. 이송 패키지는 또한 플랫폼에 명확하게 연결되지 않은 복구 키로 래핑된 DEK(184)의 사본을 포함한다. 메타데이터(182)는 또한 호스트 운영 시스템이 로딩되기 전에 호스트 프로세서(110) 상의 프리부트동안 실행될 PBA 코드를 포함하는 스토리지 영역을 위한 식별자 및 PBA 엑스큐터블(executables)을 포함할 수 있다. 예컨대, PBA 코드를 포함하는 이 스토리지 영역은 플래시 메모리(190)의 일부일 수 있다. PBA 영역에 대한 액세스는 VECI(111c)를 이용하는 I/O 커맨드 디코드 모듈(140)을 통해 또는 HECI(111b)를 이용하는 호스트 커맨드 인터페이스를 통한 매니저빌러티 엔진(ME)(130)을 통해 호스트 프로세서(110) 상에서 실행되는 코드에 의해 허가된다. PBA 코드가 호스트 프로세서(110) 상에서 실행되기 때문에, I/O 커맨드 디코드 모듈(140)은 PBA 스토리지 영역을 액세스하기 위한 요청이 PBA 엑스큐터블이 저장되는 스토리지의 범위까지 제한되는 것을 보장한다.

[0053] I/O 커맨드 디코드 모듈(140)의 SATA 가상화 펌웨어(243)는 단계 320의 "I/O 커맨드 디코드 모듈은 메타데이터 기술자 정보를 포함하는 가상 드라이브 정의 메타데이터를 판독한다"에서 SATA 스토리지 장치(180)에 저장된 메타데이터(182)에 저장되는 메타데이터 기술자 정보를 포함하는 가상 드라이브 정의 메타데이터를 판독하는 경우, 메타데이터 기술자 정보는 장치 메타데이터(182) 내의 DEK(184) 등의 래핑된 장치 암호화 키의 다수의 인스턴스(instances)를 포함할 수 있다. 예컨대, DEK(184)는 플랫폼(100)에 연결되지 않는 복구 키에 의해 래핑될 뿐만 아니라 플랫폼이 특정되는 장치 래핑 키에 의해서도 래핑될 수 있다. 장치 암호화 키의 다수의 인스턴

스가 존재할 수 있기 때문에, 시스템 리셋을 수행하는 것과 관련된 사용자 인증 크리덴셜을 이용하여 언랩될 수 있는 특정 장치 암호화 키의 위치를 결정할 필요가 있다.

[0054] 상술한 바와 같이, 시스템 리셋을 수행하는 사용자는 장치 래핑 키를 래핑하는 데 사용되는 연관된 사용자 래핑 키를 가질 것이다. 사용자 래핑 키/장치 래핑 키는 플래시 메모리(190)의 장치 메타데이터(189)에 저장된다. 제어는 단계 330의 "I/O 커맨드 디코드 모듈은 사용자 인증 메타데이터 오프셋을 위치설정하고 장치 메타데이터를 판독한다"로 진행한다. 단계 330에서, I/O 커맨드 디코드 모듈(140)은 메타데이터 기술자 정보를 이용하여 시스템 리셋을 수행하는 데 사용되는 사용자 크리덴셜에 대해 플래시 메모리(190) 내에 사용자 인증 메타데이터 오프셋을 위치설정한다. 사용자 인증 메타데이터 및 다른 장치 메타데이터는 오프셋에 의해 식별되는 플래시 메모리(190)의 위치로부터 판독된다.

[0055] 단계 330의 "I/O 커맨드 디코드 모듈은 사용자 인증 메타데이터 오프셋을 위치설정하고 장치 메타데이터를 판독한다"에서 플래시 메모리(190)로부터 장치 메타데이터(298)를 판독한 후에, 제어는 단계 340의 "ME 피보호 장치 매니저는 보안 스토리지에 장치 메타데이터를 저장한다"로 진행한다. 예컨대, ME 피보호 장치 매니저(235)는 매니저빌러티 엔진(ME)(130)에 의한 이후의 액세스를 위해 SATA 스토리지 장치(180)를 위한 사용자 인증 크리덴셜을 포함하는 장치 메타데이터를 플래시 메모리(190)의 장치 메타데이터(298)에 저장할 수 있다. 그러면 제어는 단계 350의 "ME 피보호 장치 매니저는 매니저빌러티 동작 커맨드가 발행되기를 기다린다"로 진행한다. 예컨대, ME 피보호 장치 매니저(235)는 SATA 스토리지 장치(180)를 액세스하기 위해 잠금해제 커맨드 등의 매니저빌러티 동작 커맨드를 기다린다. 매니저빌러티 동작 커맨드가 수신되면, ME 피보호 장치 매니저(235)는 사용자 인증 크리덴셜 및/또는 SATA 스토리지 장치(180)를 액세스 하는 데 필요한 다른 정보를 획득하기 위해 저장된 메타데이터에 액세스할 수 있다.

[0056] 도 4는 본 발명의 일 실시예에 따라 암호화, 사용자 신원 인증 및 비밀번호 보호 방식에 의해 보호된 장치를 잠금해제하는 커맨드를 수신하면 수행될 방법의 흐름도이다. 도 4의 방법은, 그러한 구현에에 한정되지 않지만, 도 2의 시스템의 구성요소에 의해 수행되는 것으로 기술될 것이다. 피보호 장치를 잠금해제하라는 요청의 근원지에 따라 방법의 두가지 흐름의 예가 도 4에 제공된다. 원격 잠금해제 블럭(402)에 포함된 방법 단계는 OOB 통신 채널(168) 등의 보안 통신 채널을 통해 관리 콘솔(166)로부터 수신된 잠금해제 커맨드 등의 원격 잠금해제 커맨드의 처리를 포함한다. USB 잠금해제 블럭(418)에 포함된 방법 단계는 스토리지 장치를 잠금해제하기 위해 토큰을 저장하는 USB 장치와 함께 잠금해제 커맨드의 처리를 포함한다.

[0057] 원격 잠금해제 커맨드를 처리하는 원격 잠금해제 블럭(402)의 방법 단계는 단계 404의 "관리 콘솔은 SATA 장치에 대한 원격 잠금해제 요청을 트리거링한다"로 시작한다. 관리 콘솔(166)은 기업 관리 정책에 응답하여 요청을 개시할 수 있고, 또는 관리 콘솔(166)은 SATA 스토리지 장치(180) 등의 장치가 잠금해제되어야 한다는 매니저빌러티 엔진(ME)(130)으로부터의 통지에 응답하여 작동할 수 있다. 사용자가 키보드에 존재하지 않는 경우, 사용자가 존재하지만 플랫폼에 성공적으로 인증할 수 없는 경우, 플랫폼이 저전력 상태(ACPI(Advanced Configuration and Power Interface) Sx 전력 상태 S1 내지 S5 중 하나 등)인 경우, 시스템이 유선 또는 무선 네트워크를 통해 접속되거나 기업의 방화벽 바깥에 있지만 장치가 액세스 불가능한 경우, 및 시스템에 대한 호스트 운영 시스템이 오작동하는 경우에, 관리 콘솔(166)은 매니저빌러티 엔진(ME)(130) 및 칩셋/보안 파티션(120)을 통해 암호화된 SATA 장치를 잠금해제하라는 요청을 트리거링한다.

[0058] 제어는 단계 404의 "관리 콘솔은 SATA 장치에 대한 원격 잠금해제 요청을 트리거링한다"로부터 단계 406의 "관리 콘솔은 보안 파티션에 접속하고, 토큰-1 및 장치 비밀번호를 포함하는 디스크 잠금해제 커맨드를 송신한다"로 진행한다. 관리 콘솔(166)은 칩셋/보안 파티션(120)에 의해 제공된 내장 보안 서브시스템에 잠금해제 동작을 명령하는 보안 커맨드를 송신하기 위해 OOB 통신 채널(168) 등의 독립적으로 안전하고 암호화된 채널을 이용할 수 있다. OOB 통신 채널(168) 등의 보안 통신 채널이 관리 콘솔(166)과 칩셋/보안 파티션(120) 사이에 확립되는 경우, 관리 콘솔(166) 및 매니저빌러티 엔진(ME)(130)을 인증하기 위해 커베로스 인증이 신원 관리 펌웨어(239)에 의해 사용된다. 원격 잠금해제 요청이 매니저빌러티 엔진(ME)(130)에 의해 개시되면, 보안 통신 채널이 확립된 후에, 관리 콘솔(166)은 플랫폼(100)과 연관된 사용자에 대한 사용자명 및 비밀번호 등의 사용자 크리덴셜을 획득할 수 있다. 원격 잠금해제 요청이 관리 콘솔(166)에 의해 개시되면, 관리자 사용자 크리덴셜이 플랫폼(100)을 위해 사용될 수 있다. 이들 사용자 크리덴셜은 장치(180) 비밀번호(266B) 등의, 장치에 대한 연관된 비밀번호, 및 토큰-1(266A) 등의, 장치를 복호화하는 데 사용된 토큰을 관리 데이터 스토어(266)로부터 획득하기 위해 관리 콘솔(166)에 의해 사용된다.

[0059] 단계 406의 "관리 콘솔은 보안 파티션에 접속하고, 토큰-1 및 장치 비밀번호를 포함하는 디스크 잠금해제 커맨

드를 송신한다"에서, 장치 비밀번호는 장치가 장치 비밀번호를 이용하여 잠금해제될 수 있도록 잠금해제 커맨드에 포함된다. 커맨드에 토큰-1을 포함함으로써, 사용자 래핑 키/장치 래핑 키가 장치 메타데이터(298) 등에서 식별될 수 있다. 사용자 래핑 키/장치 래핑 키는 장치 암호화 키(184)를 포함하는 메타데이터(182)를 포함하여 암호화된 스토리지 장치의 블럭을 복호화하는 데 사용될 수 있다.

[0060] 단계 406의 "관리 콘솔은 보안 파티션에 접속하고, 토큰-1 및 장치 비밀번호를 포함하는 디스크 잠금해제 커맨드를 송신한다"로부터, 제어는 단계 408의 "ME 피보호 장치 매니저는 신뢰성 있는 콘솔로부터 발신된 커맨드를 입증한다"로 진행한다. 관리 콘솔(166)과 칩셋/보안 파티션(120) 사이의 보안 통신 채널을 확립하는 데 필요했던 커베로스 인증 크리덴셜은 커맨드가 신뢰성 있는 관리 콘솔(166)에 의해 발신된 것을 입증하는 데 사용될 수 있다. 그 후 제어는 단계 410의 "ME 피보호 장치 매니저는 장치 메타데이터에서 DEK의 사본을 언랩하기 위해 토큰-1을 사용한다"로 진행한다. 상술한 바와 같이, 피보호 장치 매니저(235)가 SATA 스토리지 장치(180)를 잠금해제 하기 위해 장치(180) 비밀번호(266B)에 액세스하면, 피보호 장치 매니저(235)는 플래시 메모리(190)의 장치 메타데이터(298)에 저장되는 장치 래핑 키를 복사할 수 있다. 장치 래핑 키는 플랫폼에 부착된 각 SATA 장치에 대한 장치 메타데이터에 포함된 장치 암호화 키를 언랩하는 데 사용될 수 있다.

[0061] ME 피보호 장치 매니저(235)가 암호화된 스토리지 장치에 대한 장치 암호화 키를 획득하면, 제어는 단계 412의 "ME 보안/키 관리 펌웨어는 암호화 엔진에 대한 DEK를 SATA 장치에 대응하는 키슬롯 레지스터에 기록한다"로 진행한다. 발명자 네드 스미스(Ned Smith)의 "Enforcing Use of Chipset Key Management Services for Encrypted Storage Devices"라는 제목의 미국 특허출원 제12/319,210호(참조로서 여기에 포함됨)에 기술된 바와 같이, 장치 암호화 키는 암호화 엔진(150) 내의 키슬롯 레지스터에 저장될 수 있다. 장치가 액세스되면, 암호화 엔진(150)는 대응하는 키슬롯 레지스터로부터의 저장된 장치 암호화 키를 사용하여 대응하는 장치에 저장된 데이터를 복호화한다.

[0062] 단계 412의 "ME 보안/키 관리 펌웨어는 암호화 엔진에 대한 DEK를 SATA 장치에 대응하는 키슬롯 레지스터에 기록한다"에서 장치 암호화 키가 암호화 엔진(150)에 액세스 가능하게 된 후에, 제어는 단계 414의 "ME 피보호 장치 매니저는 (OS를 부팅하는 것을 포함할 수 있는) 관리 동작을 수행한다"로 진행한다. 예컨대, 원격 잠금해제 커맨드에 응답하여, ME 피보호 장치 매니저(235)는 장치를 잠금해제하고, 이는 장치를 잠금해제하는 장치 비밀번호를 제공하는 것, 및, 암호화된 장치의 블럭을 복호화하기 위해 암호화 엔진(150)을 이용하는 것을 포함할 수 있다. 장치가 호스트 소프트웨어에 의해 더 암호화되면, 관리 동작은 관리 콘솔(166)이 신뢰성 있는 호스트 소프트웨어와 통신하여 장치를 더 복호화하도록 요구할 수 있다. USB 잠금해제 커맨드에 응답하여, ME 피보호 장치 매니저(235)는 장치를 잠금해제하는 장치 비밀번호 및 암호화된 장치의 블럭을 복호화하는 암호화 엔진(150)을 이용하는 것을 포함하여, 유사하게 장치를 잠금해제한다. 처리되는 특정 관리 동작 커맨드에 따라 플랫폼은 호스트 운영 시스템을 부팅하는 것을 포함하여 재부팅될 수 있다.

[0063] 단계 414의 "ME 피보호 장치 매니저는 (OS를 부팅하는 것을 포함할 수 있는) 관리 동작을 수행한다"에서 관리 동작이 수행된 후에, 제어는 단계 416의 "ME 피보호 장치 매니저는 플랫폼을 리셋하고 그 결과 SATA 장치가 다시 잠금 상태로 된다"로 진행한다. ME 피보호 장치 매니저(235)는 시스템을 리셋하기 위해 도 3과 관련하여 기술된 단계들을 수행하고 그 결과 스토리지 장치가 다시 잠금 상태로 된다.

[0064] 상술한 바와 같이, 도 4는 또한 USB 잠금해제 커맨드를 이용하여 장치를 수동으로 잠금해제하는 방법 단계를 포함한다. USB 잠금해제 커맨드를 처리하는 USB 잠금해제 블럭(418)의 방법 단계는 단계 420의 "BIOS 애플리케이션은 사용자가 토큰-2를 포함하는 USB 장치를 삽입하도록 프롬프팅한다"로 시작한다. 시스템이 시동하면, BIOS 애플리케이션은 사용자가 토큰-2를 포함하는 USB 장치를 삽입하도록 프롬프팅하여 사용자가 SATA 스토리지 장치(180) 등의 장치에 액세스할 수 있게 한다. 예컨대, 사용자가 장치에 액세스하기 위한 비밀번호를 제공할 수 없는 경우에 그러한 BIOS 프롬프트가 제공될 수 있다. 제어는 단계 422의 "BIOS는 토큰-2를 판독하고 그것을 HECI/DHCI를 통해 ME 피보호 장치 매니저로 송신한다"로 진행한다. BIOS 애플리케이션은 사용자에 의해 제공된 토큰-2를 판독하고 토큰-2를 HECI(111b)를 통해 ME 피보호 장치 매니저(235)로 송신한다. 그 후 제어는 단계 424의 "ME 피보호 장치 매니저는 토큰-2를 이용하여 장치 메타데이터의 DEK의 사본을 언랩한다"로 진행한다. 상술한 바와 같이, 피보호 장치 매니저(235)가 SATA 스토리지 장치(180)를 잠금해제하기 위해 잠금해제 토큰에 액세스하면, 피보호 장치 매니저(235)는 플래시 메모리(190)의 데이터 영역(198) 등의 장치 메타데이터에 저장되는 장치 랩 키(DWK)를 복사할 수 있다. DWK는 플랫폼에 부착된 각각의 SATA 장치에 대한 장치 메타데이터에 포함된 DEK를 복호화하는 데 사용될 수 있다. 단계 412의 "ME 보안/키 관리 펌웨어는 암호화 엔진에 대한 DEK를 SATA 장치에 대응하는 키슬롯 레지스터에 기록한다"에서 장치 암호화 키가 암호화 엔진(150)에 액세스 가능

하게 된 후에, 제어는 상술한 바와 같이 진행한다.

[0065] 도 2, 도 3 및 도 4의 시스템은 장치가 호스트 운영 시스템의 개입없이 잠금해제될 수 있게 한다. 액세스 전에 인증되는 시스템의 사용자의 크리덴셜이 시스템에 부착된 임의의 장치에 대해 허가되는 것이 바람직한 경우에는 특별한 고려사항이 필요하다. 시스템의 재부팅없이 동적으로 부착되는 장치가 사용자의 인증 크리덴셜의 확인을 건너뛰도록, 이 인증은 전형적으로 시스템이 부팅되면 발생한다. 인증되는 사용자에 의해 핫플러그된 장치에 대한 액세스가 바람직하지만, 사용자의 크리덴셜을 확인하기 위해 시스템을 재부팅하는 것은 지나치게 부담스럽다. 스토리지 장치의 동적 부착에 대한 인증을 가능하게 하는 것은, 예컨대, RAID 어레이의 일부인 스토리지 장치의 동적 교체가 인증된 사용자에 의해 수행되는 것을 보장하는 데 유용하다.

[0066] 유사한 문제가 ATA 커맨드를 사용하여 잠금 또는 암호화되는 장치에서 발생한다. ATA 잠금 또는 ATA 암호화 장치는 시스템 시동시에 BIOS에 의해 잠금해제되고 따라서 시스템에 핫플러그될 수 없다. 시스템의 재부팅은 장치 상의 데이터가 액세스될 수 있기 전에 장치를 잠금해제 또는 복호화하는 것을 필요로 한다.

[0067] 도 5a 및 도 5b의 시스템은 시스템의 사용자의 크리덴셜 인증에 따라 영향을 받게 될 핫플러그된 장치에 대한 액세스를 가능하게 한다. 장치가 ATA 커맨드를 이용하여 잠금 또는 암호화되는 경우라도, 핫플러그된 장치는 잠금해제 및/또는 복호화될 수 있고, 사용자의 크리덴셜은 시스템 재부팅 없이 확인될 수 있다.

[0068] 도 5a 및 도 5b의 시스템은 시스템에 부착된 복수의 장치 중 임의의 장치에 액세스가 허가되기 전에 시스템의 사용자의 제 1 크리덴셜을 인증하는 것을 요구한다. 시스템에 대한 새로운 장치의 부착을 나타내는 이벤트는 시스템의 호스트 운영 시스템으로부터 분리되는 시스템의 보안 파티션에 의해 차단된다. 새로운 장치에 액세스하기 위한 제 2 크리덴셜은 시스템을 부팅할 필요없이 요청되고, 제 2 크리덴셜이 인증되고, 새로운 장치에 대한 액세스는 제 2 크리덴셜 인증 후에 가능하게 된다. 새로운 장치에 대한 핫플러그 이벤트는 보안 파티션으로부터 호스트 운영 시스템에 전달된다.

[0069] 새로운 장치에 액세스하기 위한 제 2 크리덴셜의 요청은 제 2 크리덴셜에 대한 요청을 표시하는 디스플레이 장치 및 제 2 크리덴셜을 수신하는 사용자 입력 장치로의 신뢰성 있는 경로 접속을 이용하는 것을 포함할 수 있다. 제 1 및 제 2 크리덴셜의 인증은 신뢰성 있는 제3자에 의해 제 1 및 제 2 크리덴셜을 인증하는 것을 포함할 수 있다. 제 2 크리덴셜은 새로운 장치에 대한 비밀번호를 포함할 수 있고, 새로운 장치에 대한 액세스를 가능하게 하는 것은 새로운 장치를 잠금해제하는 비밀번호를 이용하는 것을 포함할 수 있다. 제 2 크리덴셜은 사용자 식별자를 포함할 수 있고, 새로운 장치에 대한 액세스를 가능하게 하는 것은 사용자 식별자를 신뢰성 있는 제3자에게 제공하고 신뢰성 있는 제3자가 사용자 식별자를 인증하면 새로운 장치에 대한 액세스를 가능하게 하는 것을 포함할 수 있다.

[0070] 도 5a는 본 발명의 일 실시예에 따라 암호화, 사용자 신원 인증, 동적으로 부착될 비밀번호 보호 방식, 및 재부팅없이 동적으로 재확인될 사용자 인증 크리덴셜에 의해 장치를 보호 가능하게 하는 경우의 도 1의 시스템을 더 세부적으로 나타낸다. 매니저빌러티 엔진 커널(531), 매니저빌러티 엔진 공통 서비스(533), 보안/키 관리 펌웨어(537), I/O 모듈 커널(541), SATA 가상화 펌웨어(543)는 도 2의 시스템의 대응하는 구성요소와 관련하여 기술한 바와 같이 동작한다.

[0071] 매니저빌러티 엔진(ME)(130) 내에, 신원 관리 펌웨어 커베로스 클라이언트(539A)는 사용자를 인증하기 위해 신원 관리 펌웨어 커베로스 서버(539B)와 상호작용한다. 커베로스 클라이언트(539A)는 도 1의 키 분배 센터(164) 등의 키 분배 센터에 대한 커베로스 프로토콜을 구현한다. 커베로스 클라이언트(539A)는 디스플레이 장치 및 시스템의 사용자로부터 크리덴셜을 획득하기 위한 사용자 입력 장치에 대한 신뢰성 있는 경로 접속을 이용하기 위해 신뢰성 있는 I/O 펌웨어(536)(가능하다면)를 이용할 수 있다. 커베로스 클라이언트(539A)는 사용자 크리덴셜을 키 분배 센터(164)에 제공하고 커베로스 서버(539B) 등의 커베로스 서비스에 액세스하기 위한 커베로스 티켓을 획득할 수 있다. 커베로스 서버(539B)는 장치에 액세스하는 사용자의 크리덴셜이 인증되었다는 것을 나타내는 커베로스 티켓을 수신하면 SATA 스토리지 장치(180)에 액세스할 수 있게 한다. 커베로스 티켓은, 도 2와 관련하여 상기에 설명한 장치 래핑 키 및 장치 암호화 키를 언랩하는 데 사용될 수 있는 사용자 래핑 키의 생성을 가능하게 하는, 도 2의 토큰-1(266A) 등의 사용자 토큰을 포함하는 익스텐션 필드(extention field)를 포함할 수 있다. 커베로스 티켓은, 장치를 잠금해제하는 데 사용될 수 있는, 도 2의 장치(180) 비밀번호(266B) 등의 장치 비밀번호를 포함하는 익스텐션 필드를 포함할 수 있다.

[0072] I/O 커맨드 디코드 모듈(140) 내에서 핫플러그 가상화 펌웨어(545)는 I/O 컨트롤러(170)에 의해 수신된 핫플러그 이벤트를 디코딩하고 호스트 운영 시스템(115)에 핫플러그 이벤트를 전달하기 전에 이들 이벤트를 처리한다.

핫플러그 가상화 펌웨어(545)의 동작은 도 5b와 관련하여 더 상세히 설명된다.

[0073] 도 5b는 장치의 핫플러그 이벤트를 인식하면 도 5a의 시스템에 의해 수행될 방법의 흐름도이다. 동작 5.1에서, I/O 컨트롤러(170)는 SATA 핫플러그 이벤트를 검출하고, 여기서 SATA 스토리지 장치(180)는 플랫폼(100)에 동적으로 부착되었다. 동작 5.2에서, 핫플러그 가상화 펌웨어(545)는 핫플러그 이벤트를 차단하고, 장치의 특성을 발견하기 위해 SATA 가상화 펌웨어(543)와 상호작용한다. 핫플러그 가상화 펌웨어(545)는 신원 관리 펌웨어(539)의 커베로스 클라이언트(539A)로부터 핫플러그된 장치에 대한 액세스를 요청한다. 장치가 ATA 비밀번호 방식, ATA 암호화, 및/또는 칩셋 기반 암호화를 이용하여 잠겨 있으면, 핫플러그 가상화 펌웨어(545)는 또한 SATA 스토리지 장치(180)가 장치를 액세스하라는 요청의 일부로 잠겨있음을 커베로스 클라이언트(539A)로 통지할 수 있다.

[0074] 동작 5.4에서, 커베로스 클라이언트(539A)는 사용자 인증 크리덴셜 등의 사용자 정보를 획득한다. 커베로스 클라이언트(539A)는 사용자의 크리덴셜이 SRAM(122) 등에서와 같이 매니저빌러티 엔진(ME)(130) 내에 국소적으로 저장되는지 여부를 판정할 수 있다. 사용자의 크리덴셜이 국소적으로 저장되면, 동작 5.4 및 5.5는 생략될 수 있다. 사용자의 크리덴셜이 국소적으로 저장되지 않은 경우, 사용자 인증 크리덴셜은, 플랫폼(100) 상에서 이용가능하다면, 신뢰성 있는 I/O 펌웨어(536)를 통해 획득될 수 있다. 신뢰성 있는 I/O 펌웨어(536)는 크리덴셜에 대한 요청을 표시하기 위한 디스플레이 장치로의 신뢰성 있는 경로 접속 등, 및 크리덴셜을 수신하기 위한 키보드 등의 사용자 입력 장치로의 신뢰성 있는 경로 접속 등의 신뢰성 있는 경로 접속들을 이용할 수 있다. 신뢰성 있는 I/O 펌웨어(536)가 플랫폼(100)에서 이용 가능하지 않은 일 실시예에서, 통지는 새로운 장치가 부착된 프로세서(110)에서 실행중인 호스트 에이전트(도시하지 않음)로 송신될 수 있다. 호스트 에이전트는 사용자의 크리덴셜을 수집하고, 장치를 잠금해제하고 장치를 호스트 운영 시스템(115)에 대해 가시적으로 되게 하기 위해 칩셋/보안 파티션(120)에 접속할 수 있다.

[0075] 동작 5.5에서, 커베로스 클라이언트(539A)는 키 분배 센터(164)로부터 커베로스 티켓을 획득한다. 일 실시예에서, 커베로스 티켓은 SATA 스토리지 장치(180)에 대해 도 2의 토큰-1(266A) 등의 사용자를 위한 잠금해제 토큰, 및 도 2의 장치(180) 비밀번호(266B) 등의, 사용자에 속한 ATA 비밀번호와 함께 제공된다. 사용자를 위한 이 잠금해제 토큰 및 ATA 비밀번호는 도 1 및 도 2의 관리 콘솔(166) 등의 디렉토리 서비스로부터 획득될 수 있다. 커베로스 티켓은 커베로스 서버(539B)로부터의 서비스를 수신하기 위해 사용자의 크리덴셜이 진실임을 확인한다. 일 실시예에서, 커베로스 서버(539B)는 커베로스 클라이언트(539A)가 보안/키 관리 펌웨어(537) 및 피보호 장치 매니저(539)로부터의 서비스 등의, 매니저빌러티 엔진(ME)(130) 내의 다른 서비스에 모두 액세스할 수 있게 한다. 다른 실시예에서, 별개의 커베로스 티켓이 보안/키 관리 펌웨어(537) 등의 다른 매니저빌러티 엔진(ME)(130)의 구성요소에 의해 제공된 서비스에 액세스하기 위해 획득될 수 있다. 일 실시예에서, SATA 스토리지 장치(180)에 대한 사용자를 위한 잠금해제 토큰 및 사용자에 속한 ATA 비밀번호가 커베로스 세션 키의 일부인 익스텐션 필드로서 전달된다.

[0076] 동작 5.6에서, 커베로스 클라이언트(539A)는 커베로스 서버(539B)에 의해 사용자의 크리덴셜을 확인한다. 대안적인 실시예에서, 커베로스 클라이언트(539A)는 커베로스 서버(539B) 등의 로컬 커베로스 서버를 통하지 않고 키 분배 센터(164)에 의해 직접 사용자의 크리덴셜을 확인할 수 있다. 예컨대, 커베로스 클라이언트(539A)는 후속 교환에서 토큰-1(266A) 및 장치(180) 비밀번호(266B)를 반환할 액티브 디렉토리 서비스 등의 상이한 사용자 인증 서비스에 액세스하기 위해 커베로스 티켓을 획득할 수 있다. 일 실시예에서, 도 1 및 도 2의 관리 콘솔(166)은 다른 사용자 인증 서비스에 대한 접속을 대리하고 및/또는 스스로 사용자 인증 서비스를 관리할 수 있다.

[0077] 동작 5.7 내지 5.10은 핫플러그된 SATA 스토리지 장치(180)가 ATA 비밀번호 또는 ATA 암호화 등의 자생 잠금 메커니즘에 의해 보호되는 경우 취해진 동작을 설명한다. 장치가 ATA 비밀번호 또는 ATA 암호화 등의 자생 잠금 메커니즘에 의해 보호되지 않는 경우 단계 5.7 내지 5.10은 생략될 것이다.

[0078] 동작 5.7에서, 핫플러그된 SATA 스토리지 장치(180)가 ATA 비밀번호를 이용하여 잠긴 상황에서, 커베로스 클라이언트(539A)는 피보호 장치 매니저(535)에 사용자의 ATA 비밀번호를 제공한다. 동작 5.8에서, 피보호 장치 매니저(535)는 I/O 커맨드 디코드 모듈(140)의 SATA 가상화 펌웨어(543)에 사용자의 ATA 비밀번호를 제공한다. 동작 5.9에서, SATA 가상화 펌웨어(543)는 SATA 스토리지 장치(180)를 잠금해제하기 위해 I/O 컨트롤러(170)에 ATA 커맨드를 송신한다. 동작 5.10에서, I/O 컨트롤러(170)는 SATA 스토리지 장치(180)를 잠금해제하기 위해 ATA 커맨드를 이용한다. 상술한 바와 같이, SATA 스토리지 장치(180)가 암호화 엔진(150)에 의해 암호화되었으면, 보안/키 관리 펌웨어/커베로스 서버(537)는 사용자 래핑 키를 유도하기 위해 커베로스 티켓의 익스텐션 필

드에 포함된 사용자의 잠금해제 토큰을 사용하도록 신원 관리 펌웨어/커베로스 클라이언트(539)와 협력하여 작동할 수 있다. 사용자 래핑 키는 SATA 스토리지 장치(180)로부터의 장치 래핑 키 및 장치 암호화 키에 액세스하는 데 사용될 수 있다.

[0079] 동작 5.11 및 5.12는 핫플러그된 SATA 스토리지 장치(180)가 암호화 엔진(150)에 의해 암호화되는 경우에 취해진 동작을 기술한다. 핫플러그된 SATA 스토리지 장치가 암호화 엔진(150)에 의해 암호화되지 않으면, 단계 5.11 및 5.12는 생략될 것이다. 핫플러그된 SATA 스토리지 장치(180)가 칩셋/보안 파티션(120)의 암호화 엔진(15)에 의해 암호화되면, 동작 5.11에서, 커베로스 클라이언트(539A)는 보안/키 관리 펌웨어(537)가 핫플러그된 SATA 스토리지 장치(180)에 대한 장치 복호화를 가능하게 하도록 요청할 수 있다. 사용자 크리덴셜은 도 2와 관련하여 상술한 바와 같이 장치 암호화 키를 획득하는 데 사용될 수 있다. 동작 5.12에서, 장치 암호화 키(184)는 암호화 엔진(150)에 제공된다. 상술한 바와 같이, 장치 암호화 키는 암호화 엔진(150)의 키슬롯 레지스터에 기입될 수 있고 핫플러그된 장치의 SATA 스토리지 장치(180)의 블럭을 복호화하기 위해 암호화 엔진(150)에 의해 사용될 수 있다. 핫플러그된 SATA 스토리지 장치(180)가 또한 ATA 비밀번호에 의해 보호되면, SATA 스토리지 장치(180)를 잠금해제하기 위해 동작 5.7 내지 5.10과 관련하여 상술한 단계는 장치에 저장된 장치 암호화 키가 액세스될 수 있기 전에 장치를 잠금해제하는 데 사용된다.

[0080] 동작 5.13에서, 커베로스 클라이언트(539A)는 SATA 스토리지 장치(180)에 대한 액세스가 승인되었음을 핫플러그 가상화 펌웨어(545)에 통지한다. 동작 5.7 내지 5.10과 관련하여 설명한 바와 같이, SATA 스토리지 장치(180)가 ATA 비밀번호로 잠겼으면, 장치는 잠금해제되었다. 동작 5.11 및 5.12와 관련하여 설명한 바와 같이, 장치가 암호화 엔진(150)에 의해 암호화되었으면, 복호화는 가능하게 되었다. 동작 5.14에서, 핫플러그 가상화 펌웨어(545)는 핫플러그 이벤트를 호스트 운영 시스템(115)에 전달한다. 그러면 호스트 운영 시스템(115)은 SATA 스토리지 장치(180)로부터의 잠금해제되고 복호화된 데이터에 액세스한다. 핫플러그 이벤트를 수신하는 것에 응답하여, 호스트 운영 시스템(115)은 SATA 스토리지 장치(180)를 RAID 어레이에 고정 및/또는 포함시키기 위해 파일 시스템을 호출할 수 있다.

[0081] 도 1 내지 도 5b와 관련하여 상술한 시스템에서, 스토리지 장치의 암호화는 칩셋/보안 파티션(120) 내에서 암호화 엔진(150)에 의해 수행된다. 또한, 위에서 도 1 내지 도 5b와 관련하여 설명한 시스템은 호스트 운영 시스템과 분리된 시스템의 보안 파티션 내에서 암호화 및 피보호 장치 관리 기능을 제공한다. 예컨대, 암호화 엔진(150)은 칩셋/보안 파티션(120) 내에 있고, 도 2의 피보호 장치 매니저(235)는 칩셋/보안 파티션(120) 내의 매니저빌러티 엔진(ME)(130) 내에 있으며, 도 5b의 SATA 가상화 펌웨어(543) 및 핫플러그 가상화 펌웨어(545)는 칩셋/보안 파티션(120)의 I/O 커맨드 디코드 모듈(140) 내에 있다.

[0082] 전형적으로, 감사가능한 이벤트는 호스트 운영 시스템 및/또는 BIOS의 제어하에서 실행중인 소프트웨어를 감사할 때 캡쳐된다. 여기서 기술된 관리 및 암호화 기능이 호스트 운영 시스템 및 BIOS와 분리되기 때문에, 보안 파티션 내에서 수행된 이벤트는 일반적인 감사 소프트웨어에 의해 캡쳐되지 않는다. 그러나, 피보호 장치의 관리 및 저장된 데이터의 암호화에 영향을 미치는 이벤트를 캡쳐 및 감사하는 것이 바람직하다. 또한 호스트 운영 시스템 및/또는 BIOS의 잠재적 변질로부터 보호되는 환경에서 감사 동작을 수행하는 것이 바람직하다. 또한 감사가능한 이벤트가 발생하는 시간 및 보안 환경에서 감사 정보를 캡쳐하는 것이 바람직하다.

[0083] 도 6은 본 발명의 일 실시예에 따라 피보호 장치를 관리하는 경우의 도 1의 시스템을 더 상세히 도시한다. 매니저빌러티 엔진 커널(631), 매니저빌러티 엔진 공통 서비스(633), 피보호 장치 매니저(635), 보안/키 관리 펌웨어(637) 및 신원 관리 펌웨어(639)는 도 2 및 도 5a의 대응하는 구성요소와 관련하여 설명한 것과 같이 동작한다.

[0084] 도 6에 도시된 실시예에서, 매니저빌러티 엔진(ME)(130)은 매니저빌러티 엔진 감사 서브시스템(638)을 포함하고, I/O 커맨드 디코드 모듈(140)은 I/O 모듈 감사 서브시스템(648)을 포함한다. 매니저빌러티 엔진 감사 서브시스템(638) 및 I/O 모듈 감사 서브시스템(648)은 칩셋/보안 파티션(120)의 그들 각각의 구성요소 내에서 발생하는 감사가능한 동작을 식별 및 처리한다. I/O 커맨드 디코드 모듈(140)이 스토리지 장치에 대한 I/O용 데이터를 준비하고, 스토리지 장치에 데이터가 기록되면 데이터를 암호화하기 위해 암호화 엔진(150)과 직접 작업하기 때문에, I/O 모듈 감사 서브시스템(648)은 I/O동안 발생하는 감사가능한 이벤트를 캡쳐한다. 반대로, 매니저빌러티 엔진(ME)(130)은 스토리지 장치에 대한 I/O에 직접 개입하지 않고, 따라서 매니저빌러티 엔진 감사 서브시스템(638)은 피보호 장치의 관리에 관한 감사가능한 이벤트를 캡쳐한다. 예컨대, 매니저빌러티 엔진 감사 서브시스템(638)은 암호화, 사용자 인증, 장치 초기화 및 실패, 암호화 키, 도난 검출 및 다른 기업 플랫폼 관리 정책을 관리하기 위해 시스템의 구성 및 설정에 관한 이벤트를 캡쳐한다.

[0085] 매니저빌러티 엔진 감사 서브시스템(638) 및 I/O 모듈 감사 서브시스템(648)은 매니저빌러티 엔진 컨트롤러 인터페이스(MECI)(131)를 통해 통신한다. 매니저빌러티 엔진 감사 서브시스템(638)은 또한 OOB 통신 채널(168), 네트워크 컨트롤러(160) 및 아웃오브밴드 통신 모듈(630)을 통해 관리 콘솔(166) 내의 원격 감사 관리 서비스(640)와 통신할 수 있다.

[0086] 일 실시예에서, 감사가능한 이벤트는 감사 정책에 정의된다. 감사 정책은 감사 이벤트 기록이 생성될 감사가능한 이벤트뿐만 아니라 무시될 수 있는 다른 비감사 이벤트도 정의할 수 있다. 시스템 내에서 발생하는 모든 이벤트를 감사하는 것은 시스템의 성능을 크게 저하시킬 수 있기 때문에, 감사 정책은 구조적 우선권 및 정책에 따라 특별히 관심있는 이벤트를 선택적으로 캡처하도록 사용된다. 일 실시예에서, 선택기가 감사가능한 이벤트를 검출할 수 있는 다른 하드웨어 및/또는 펌웨어 구성요소를 활성화 및 비활성화하는 경우, 감사 비트 마스크가 사용된다.

[0087] 감사 정책의 이벤트 타입은 암호화 시스템 공급/제거 이벤트, 사용자 관리 이벤트, 장치 관리 이벤트, 키 관리 이벤트, 장치 초기화 이벤트, 도난 검출 이벤트 및 장치 고장 이벤트를 포함할 수 있다. 특정 감사가능한 이벤트는 보안 파티션 내의 활동을 유발하는 호스트 운영 시스템에 의해 트리거링되는 이벤트 등의, 시스템의 보안 파티션 외부의 동작, 및/또는 인터럽트 등의 보안 파티션 내에서 내부적으로 발생하는 동작에 의해 트리거링되는 이벤트를 포함할 수 있다.

[0088] 외부적으로 트리거링되는 이벤트는 도난 방지 서비스를 가능하게 또는 불가능하게 하는 것, 사용자 계정을 생성, 삭제 또는 수정하는 것, 사용자 로그온/로그오프 성공 또는 실패, 장치 암호화 키, 장치 래핑 키, 복구 키 등의 다양한 형태의 암호화 키에 대해 생성되거나 삭제되는 암호화 키, 암호화 또는 복호화를 위해 구성된 장치, 보안 관리 장치로서의 장치 변환 또는 역변환, PASS_THROUGH를 위한 장치 구성, 장치 이동 또는 장치 이동을 위한 준비, 암호화 엔진 레지스터에 대한 장치 암호화 키(DEK) 삽입 또는 제거, 감사 이벤트 정책 등록 또는 등록해제, 플랫폼 또는 장치 메타데이터의 복구, 로컬 플랫폼 토큰의 사용자, 암호화 정책의 키 강도, 키 리프레시 또는 원격 구성에 대한 변경 등의 암호화 정책의 변경, 인증 및 미인증 암호화간의 천이, 장치 잠금해제 동작, 장치 고장을 포함할 수 있다. 내부적으로 트리거링되는 감사가능한 이벤트는 매니저빌러티 엔진, I/O 커맨드 디코드 모듈, 암호화 엔진 및/또는 보안 파티션에 대한 인터페이스의 셀프테스트 실패, 연방정부 정보처리 표준(Federal Information Processing Standard) 셀프테스트 성공 또는 실패, 감사 초기화 실패, 및/또는 메모리 고장을 포함할 수 있다.

[0089] 이벤트가 매니저빌러티 엔진 감사 서브시스템(638) 또는 I/O 모듈 감사 서브시스템(648)에 의해 검출되면, 검출된 이벤트가 감사 정책에 정의된 감사가능한 이벤트 중 하나인지 판정될 수 있다. 검출된 이벤트가 감사 정책의 감사가능한 이벤트 중 하나이면, 이벤트는 감사가능한 이벤트로 식별된다.

[0090] 감사 정책은 각 감사가능한 이벤트에 대한 감사 이벤트 기록(audit event records)을 제공하기 위한 명령을 포함할 수 있다. 감사 정책은 감사가능한 이벤트가 기록될 수 없는 경우 취해질 동작을 더 지정할 수 있다. 예컨대, 매니저빌러티 엔진 감사 서브시스템(638) 또는 I/O 모듈 감사 서브시스템(648)은 감사 이벤트 기록이 감사 로그(audit log)에 기록될 수 없는 동작을 중단 또는 (그 효과를 반전시키는) 실행취소하도록 구성될 수 있다. 또한, 매니저빌러티 엔진 감사 서브시스템(638) 또는 I/O 모듈 감사 서브시스템(648)이 감사 로그를 재기입하거나 감사 로그에 감사 이벤트 기록을 기입하는 것을 중단하도록 구성될 수 있다.

[0091] 일 실시예에서, 매니저빌러티 엔진 감사 서브시스템(638) 및 I/O 모듈 감사 서브시스템(648)의 각각은 식별된 감사가능 이벤트에 대한 감사 이벤트 기록을 생성한다. 감사 이벤트 기록은 호스트 운영 시스템과 분리되는 감사 로그에 기입된다. 도 6에 도시된 실시예에서, 매니저빌러티 엔진 감사 서브시스템(638)은 감사 로그(610)에 감사가능한 이벤트를 기입하고, I/O 모듈 감사 서브시스템은 감사 로그(620)에 감사가능한 이벤트를 기입한다. 일 실시예에서, 감사 로그(610)는 도 1의 플래시 메모리(190)의 데이터 영역(198)의 분리 영역 등의 플래시 메모리의 분리 영역에 저장되고, 감사 로그(620)는 도 1의 비휘발성 메모리 스토리지 장치(172)의 분리 영역 등의 비휘발성 메모리의 분리 영역에 저장된다. 비휘발성 메모리는 플래시 메모리보다 빠르기 때문에, 일 실시예에서, 감사 이벤트 기록은, 비휘발성 메모리가 이용 가능하면, 비휘발성 메모리에 저장된 감사 로그(본 예에서는 감사 로그(620))에 우선 기입된다. I/O 커맨드 디코드 모듈(140)은 스토리지 장치에 대한 I/O를 위한 데이터를 준비하고, 스토리지 장치에 기입되는 데이터 등의 데이터를 암호화하기 위해 암호화 엔진(150)과 직접 작업하기 때문에, I/O 모듈 감사 서브시스템(648)은 I/O 이벤트를 처리할 때의 레이턴시를 감소시키기 위해 비휘발성 메모리에 저장된 더 빠른 감사 로그(620)에 연결된다. 매니저빌러티 엔진 감사 서브시스템(638)은 I/O에 직접 개입하기 않기 때문에, 매니저빌러티 엔진 감사 서브시스템(638)은 플래시 메모리(190) 등의 플래시 메모리에 저

장된 더 느린 감사 로그(610)에 감사 이벤트 기록을 기입한다.

[0092] 감사 로그(610) 및/또는 감사 로그(620)가 임계치에 도달하면, 매니저빌러티 엔진 감사 서브시스템(638)은 원격 감사 관리 서비스(640)에 감사 로그를 서비스할 것을 통지할 수 있다. 일 실시예에서, 감사 관리 서비스(640)는 원격 스토리지에 감사 로그(610, 620)의 내용을 복사하고 임계값을 리셋한다. 감사 관리 서비스(640)는 매니저빌러티 엔진 감사 서브시스템(638) 및 I/O 모듈 감사 서브시스템(648)의 동작을 인터럽트하지 않고, 감사가능한 이벤트가 식별되는 것과 동시에 그들 각각의 감사 로그(610, 620)에 감사 이벤트 기록을 계속해서 기입한다. 감사 로그(620)가 임계값에 접근하면, I/O 모듈 감사 서브시스템(648)은 MECI(131)를 통해 매니저빌러티 엔진 감사 서브시스템(638)에 통지하여 매니저빌러티 엔진 감사 서브시스템(638)이 감사 관리 서비스(640)에 대해 서비스하라는 요청을 송신할 수 있게 한다.

[0093] 일 실시예에서, 매니저빌러티 엔진 감사 서브시스템(638)은 보안 파티션 내에서 활성화되는 모든 감사 서브시스템을 관리하는 감사 관리 서비스(640)와 함께 작업한다. 매니저빌러티 엔진 감사 서브시스템(638)은, 다른 감사 서브시스템이 과부하가 걸려 감사가능한 이벤트를 처리할 수 없는 경우에, I/O 모듈 감사 서브시스템(648) 등의 다른 감사 서브시스템의 기능을 수행할 수 있다. 매니저빌러티 엔진 감사 서브시스템(638)은 또한 다른 감사 서브시스템에 대한 감사 로그를 서비스할 수 있다. 일 실시예에서, 매니저빌러티 엔진 감사 서브시스템(638)은 등록하기 위한 다른 감사 서브시스템을 필요로 한다. 등록은 다른 감사 서브시스템에 의해 유지되는, 감사 로그(620) 등의 로컬 감사 로그가 있는 것을 매니저빌러티 엔진 감사 서브시스템(638)에 통지하는 데 사용된다. 등록은 또한 이산적인 감사가능한 이벤트가 처리를 위해 다시 라우팅될 것인지 여부, 및/또는 감사 로그의 서비스가 요구되는지 여부를 매니저빌러티 엔진 감사 서브시스템(638)에 통지하는 데 사용될 수 있다.

[0094] 일 실시예에서, 매니저빌러티 엔진 감사 서브시스템(638) 및 I/O 모듈 감사 서브시스템(648)의 동작은 커베로스 티켓을 이용하는 기업 도메인 특전에 의해 제어된다. 일 실시예에서, 시스템의 보안 파티션 내에서 수행되는 감사가능한 이벤트가 식별되고, 보안 파티션은 시스템의 호스트 운영 시스템과 분리된다. 감사 이벤트 기록은 감사가능한 이벤트에 대해 생성되고, 호스트 운영 시스템으로부터 분리되는 감사 로그에 기입된다. 일 실시예에서, 복수의 감사가능한 이벤트는 감사 정책에 정의되고, 감사 정책은 복수의 감사가능한 이벤트의 각각의 감사가능한 이벤트를 서비스하는 명령을 포함하며, 감사가능한 이벤트를 식별하는 것은 검출된 이벤트가 감사 정책에 정의된 복수의 감사가능한 이벤트 중 하나인지 여부를 판정하는 것을 포함한다.

[0095] 감사 로그는 보안 파티션 내에서만 액세스 가능한 복수의 감사 로그 중 제 1 감사 로그일 수 있다. 복수의 감사 로그의 각각의 감사 로그는 호스트 운영 시스템으로부터 분리된다. 일 실시예에서, 제 1 감사 로그가 이용가능한지 여부가 판정된다. 감사 이벤트 기록은, 제 1 감사 로그가 이용 가능하면, 제 1 감사 로그와 연관된 제 1 감사 서브시스템에 송신되고, 제 1 감사 서브시스템은 제 1 감사 로그에 감사 이벤트 기록의 기입을 수행한다. 제 1 감사 로그가 이용 가능하지 않으면, 감사 이벤트 기록은 복수의 감사 로그 중 제 2 감사 로그와 연관된 제 2 감사 서브시스템에 송신되고, 제 2 감사 서브시스템은 제 2 감사 로그에 감사 이벤트 기록의 기입을 수행한다.

[0096] 일 실시예에서, 제 1 감사 로그에 대한 기입 동작의 레이턴시가 모니터링된다. 레이턴시가 사전결정된 임계값에 도달하면, 후속 기입 동작에 대한 후속 감사 이벤트 기록이 제 2 감사 로그에 기입될 수 있도록 후속 기입 동작이 제 2 감사 서브시스템으로 이송된다. 다른 실시예에서, 레이턴시가 사전결정된 임계값에 도달하면, 제 1 감사 로그를 서비스하라는 요청이 제 2 감사 서브시스템으로 송신된다. 제 2 감사 서브시스템은 감사 이벤트 기록을 제 1 감사 로그로부터 제 3 감사 로그 등의 다른 위치로 이동함으로써 제 1 감사 로그를 서비스한다. 일 실시예에서, 제 2 감사 서브시스템은 제 3 감사 로그를 서비스하는 원격 관리 애플리케이션을 스케줄링하고, 원격 관리 애플리케이션은 보안 파티션과의 보안 통신 채널을 확립하며, 원격 관리 애플리케이션은 보안 통신 채널을 통해 제 3 감사 로그를 서비스한다.

[0097] 일 실시예에서, 감사 로그를 서비스하라는 요청은 요청하는 시스템의 보안 파티션으로부터 수신되고, 보안 파티션은 요청하는 시스템의 호스트 운영 시스템과 분리되고, 감사 로그는 보안 파티션에서 수행된 감사가능한 이벤트의 감사 이벤트 기록을 포함하고, 감사 로그는 요청하는 시스템의 호스트 운영 시스템과 분리된다. 보안 통신 채널은 보안 파티션에 의해 확립되고, 감사 로그는 보안 통신 채널을 통해 서비스된다. 감사 로그를 서비스하는 것은 감사 정책에 따라 감사가능한 이벤트를 처리하는 것을 포함할 수 있다.

[0098] 도 7은 본 발명의 일 실시예에 따라 시스템의 보안 파티션 내에서 발생하는 잠재적으로 감사가능한 이벤트를 검출하면 수행될 방법의 흐름도이다. 단계 702의 "이벤트 검출"에서 칩셋/보안 파티션(120) 등의 보안 파티션 내에서 발생하는 이벤트를 검출하면, 제어는 판정 지점 704의 "감사가능한 이벤트인가?"로 진행한다. 판정 지점

704의 "감사가능한 이벤트인가?"에서, 하드웨어에서 인코딩된 각각의 로직 및/또는 각각의 감사 서브시스템(매니저빌러티 엔진 감사 서브시스템(638) 및 I/O 모듈 감사 서브시스템(648) 중 하나)이 이벤트가 감사가능한지 여부를 판정하기 위해 감사 정책을 확인할 수 있다. 일 실시예에서, 감사 비트 마스크는 감사가능한 이벤트를 검출할 수 있는 다른 하드웨어 및/또는 펌웨어 구성요소를 활성화하는 데 사용된다. 판정 지점 704의 "감사가능한 이벤트인가?"에서 감사 비트 마스크의 평가는 이벤트가 감사가능한지 여부를 판정한다.

[0099] 판정 지점 704의 "감사가능한 이벤트인가?"에서, 이벤트가 감사가능한 것이면, 제어는 단계 706의 "감사 이벤트 기록을 생성한다"로 진행한다. 감사 이벤트 기록은 하드웨어에서 인코딩된 각각의 로직 및/또는 각각의 감사 서브시스템(매니저빌러티 엔진 감사 서브시스템(638) 및 I/O 모듈 감사 서브시스템(648) 중 하나)에 의해 생성된다. 감사 이벤트 기록의 생성 후에, 제어는 판정 지점 708의 "NVM 로그가 이용 가능한가?"로 진행한다. 이미 논의된 바와 같이, 비휘발성 메모리 로그가 이용 가능하면, 이벤트의 처리와 연관된 레이턴시를 줄이기 위해 비휘발성 메모리에 감사 이벤트 기록을 기입하는 것이 바람직하다. 판정 지점 708의 "NVM 로그가 이용 가능한가?"에서, NVM 로그가 이용 가능하면 제어는 단계 710의 "I/O 모듈 감사 서브시스템으로 이벤트 기록을 송신한다"로 진행한다. 단계 710의 "I/O 모듈 감사 서브시스템으로 이벤트 기록을 송신한다"에서, 이벤트 기록은 I/O 모듈 감사 서브시스템(648)으로 송신된다.

[0100] 단계 710의 "I/O 모듈 감사 서브시스템으로 이벤트 기록을 송신한다"로부터 제어는 판정 지점 712의 "임계값 도달?"로 진행한다. 임계값에 도달하는 예는, I/O 모듈 이용이 평균 레벨 이하로 떨어진 경우 및/또는 감사 로그가 꽉찬 경우이다. 임계값에 도달하면, 제어는 단계 718의 "매니저빌러티 엔진 감사 서브시스템으로 임계 상태를 송신한다"로 진행한다. 예컨대, I/O 모듈 이용이 임계 레벨 이하로 떨어지면, 감사 활동은 매니저빌러티 엔진 감사 서브시스템(638)으로 넘겨질 필요가 있을 수 있고, 및/또는 감사 로그(620)가 서비스될 필요가 있을 수 있다. 단계 718의 "매니저빌러티 엔진 감사 서브시스템으로 임계 상태를 송신한다"가 실행되면, 매니저빌러티 엔진 감사 서브시스템(638)은 감사 정책에 따라 임계값 도달을 관리하는 적당한 동작을 취한다. 예컨대, 매니저빌러티 엔진 감사 서브시스템(638)은 감사 관리 서비스(640)가 다른 아카이브 스토리지에 대한 임계값에 도달한 로그를 서비스 및/또는 복사하도록 스케줄링할 수 있다. 단계 718의 "매니저빌러티 엔진 감사 서브시스템으로 임계 상태를 송신한다"로부터, 제어는 단계 715의 "감사 로그에 이벤트 기록을 기입한다"로 진행하고, 여기서 임계값이 도달되게 하는 감사 이벤트 기록은 매니저빌러티 엔진 감사 서브시스템(638)에 의해 로그에 기입된다.

[0101] 판정 지점 712의 "임계값 도달?"로부터, 임계값에 도달하지 않았다면, 제어는 단계 715의 "감사 로그에 이벤트 기록을 기입한다"로 진행하고, 여기서 각각의 감사 서브시스템은 그 각각의 로그에 감사 이벤트 기록을 기입한다. 그 후 제어는 단계 714의 "이벤트를 수행한다"로 진행하고, 여기서 이벤트가 수행되고, 감사가능한 이벤트의 처리가 완료된다.

[0102] 판정 지점 708의 "NVM 로그가 이용 가능한가?"에서, NVM 로그가 이용 가능하지 않으면 제어는 단계 716의 "매니저빌러티 엔진 감사 서브시스템에 이벤트 기록을 송신한다"로 진행한다. 감사 이벤트 기록은 매니저빌러티 엔진 감사 서브시스템(638)에 송신된다. 그 후 매니저빌러티 엔진 감사 서브시스템(638)은 단계 715의 "감사 로그에 이벤트 기록을 기입한다"에서 이벤트 기록을 감사 로그(610)에 기입한다. 제어는 단계 714의 "이벤트를 수행한다"로 진행하고, 여기서 이벤트가 수행되고, 감사가능한 이벤트의 처리가 완료된다.

[0103] 판정 지점 704의 "감사가능한 이벤트인가?"에서, 이벤트가 감사가능하지 않으면, 제어는 단계 714의 "이벤트를 수행한다"로 진행한다. 이벤트가 수행되고, 이벤트의 처리가 완료된다.

[0104] 감사가능한 이벤트의 처리는 하드웨어에서 인코딩된 로직 및/또는 펌웨어에 의해 수행될 수 있다. 매니저빌러티 엔진(ME)(130), I/O 커맨드 디코드 모듈(140) 및 암호화 엔진(150) 등의 칩셋/보안 파티션(120)의 구성요소의 초기화는 그들 각각의 구성요소에 대한 하드웨어로 인코딩될 수 있고, 및/또는 그들 각각의 구성요소에 대한 펌웨어에 포함될 수 있는 감사가능한 이벤트이다. 마찬가지로, HECI(111b), VECI(111c), 네트워크 컨트롤러(160), USB 컨트롤러(175), I/O 컨트롤러(170) 등의 컨트롤러 및 인터페이스에 대한 하드웨어 및/또는 펌웨어는 감사가능한 이벤트를 처리하는 로직을 포함할 수 있다.

[0105] 감사 이벤트 처리는, 도 2의 보안/키 관리 펌웨어(237)의 동작중 등의, 매니저빌러티 엔진(ME)(130)의 구성요소의 동작중 뿐만아니라 초기 구성의 매니저빌러티 엔진(ME)(130) 내에서 수행될 수 있다. 예컨대, 감사 이벤트는 보안/키 관리 펌웨어(237)가 암호화 엔진(150)의 대응하는 레지스터에 장치 암호화 키(DEK)를 기입하는 경우, 스토리지 장치가 암호화되거나 암호화 엔진(150)의 대응하는 레지스터로부터 DEK를 제거하는 경우, 암호화가 불가능한 경우에 트리거링될 수 있다.

[0106] 또한 감사 이벤트 처리는, 데이터가 (기입 동작을 위해) I/O 컨트롤러(170)로부터 암호화 엔진(150)으로 평문 형태로 이동되는 경우, 데이터가 암호문 형태로 암호화 엔진(150)에 의해 반환되는 경우에 수행될 수 있다. 감사 정책이 이들 이벤트를 감사하는 것을 주기적인 준수 테스트(compliance testing)로 제한할 수 있지만, I/O 컨트롤러(170) 및 암호화 엔진(150) 사이의 채널을 통해 발생하는 이벤트의 감사는 데이터가 암호화된다는 증명을 제공한다.

[0107] 감사 서브시스템간의 조정은 MECI(131)를 통해 통신될 수 있기 때문에, 감사 이벤트 처리는 매니저빌러티 엔진 컨트롤러 인터페이스(MECI)(131)에서 수행될 수 있다. I/O 커맨드 디코드 모듈(140)의 초기 구성은 또한 MECI(131)를 통해 수행되고 감사가능한 이벤트를 생성할 것이다.

[0108] 감사 이벤트 처리는 인터페이스 HECI(111b) 및 VECI(111c)를 통해 프로세서(110)로부터의 통신에 의해 발생할 수 있다. 예컨대, 이들 인터페이스를 통해 I/O 컨트롤러(170) 또는 USB 컨트롤러(175)로 전파하는 커맨드뿐만 아니라, 장치의 잠금 상태와 관련된 ATA 보안 커맨드도 감사가능한 이벤트를 생성한다. 또한, 사용자 인증, 암호화, 보안, 키 관리 및 상태에 관한 HECI 커맨드는 감사가능한 이벤트이다. 또한 I/O 컨트롤러(170), USB 컨트롤러(175) 및 네트워크 컨트롤러(160) 등의 컨트롤러를 초기화하는 데 사용된 커맨드도 감사가능한 이벤트이다. 도 6의 원격 감사 관리 서비스(640)와 통신하는 감사 서브시스템과 마찬가지로, 감사 로그 스토리지 및 구성 커맨드도 감사가능하다. USB 컨트롤러(175) 및/또는 I/O 컨트롤러(170)를 통한 플랫폼(100)으로의 장치 부착은 감사가능한 이벤트이다.

[0109] 감사가능하게 될 또는 감사 정책에 없는 특정 이벤트를 구성함으로써, 감사 시스템은 성능, 스토리지 용량 및 보안의 균형을 잡기 위해 미세조정될 수 있다. 원격 관리 콘솔을 통해 감사 서브시스템을 관리하고 보안 통신 채널을 통해 감사 관리 서비스를 관리함으로서, 감사 정보의 무결성이 보호될 수 있다.

[0110] 도 8은 본 발명의 일 실시예에 따라 암호화, 사용자 신원 인증 및 비밀번호 보호 방식을 이용하여 장치를 보호하는 등의 동작을 관리하기 위한 보안 파티션을 구현하는 가상머신 환경을 도시한다. 플랫폼(800)이 가상화되면, 단일 프로세서만을 포함할 수 있지만, 호스트 상의 가상 머신 모니터("VMM(830)")는 호스트의 복수의 관념 및/또는 관점을 제시할 수 있어서, 호스트의 근본적인 하드웨어는 하나 이상의 독립적으로 동작하는 가상 머신("VM")으로 나타난다. VMM(830)은 (예컨대, 호스트 운영 시스템의 독립형 프로그램 및/또는 구성요소로서의) 소프트웨어, 하드웨어, 펌웨어 및/또는 그들의 임의의 조합으로 구현될 수 있다. VMM(830)은 호스트에서 리소스의 할당을 관리하고, 라운드로빈(round-robin) 또는 다른 사전결정된 방식에 따라 여러 VM 사이에서 순환하기 위해 필요한 경우 컨텍스트 전환을 수행한다. 하나의 프로세서만 도시되었지만("프로세서(805)") 본 발명의 실시예는 그와 같이 제한되는 것이 아니며 다수의 프로세서가 가상화된 환경 내에서 사용될 수도 있음을 당업자가 쉽게 알 수 있을 것이다.

[0111] 두 개의 VM 파티션만 도시되었지만("VM(810)" 및 "VM(820)", 이하에서는 총괄하여 "VM"이라고 부름), 이들 VM은 단지 예시이며 추가의 가상 머신이 호스트에 부가될 수 있다. VM(810) 및 VM(820)은 그들 자신의 "게스트 운영 시스템"(즉, "게스트 OS(811)" 및 "게스트 OS(821)"로서 도시된, VMM(830)에 의해 관리된 운영 시스템이며, 이하 총괄하여 "게스트 OS"라고 함) 및 다른 소프트웨어("게스트 소프트웨어(812)" 및 "게스트 소프트웨어(822)"로서 도시되며, 이하 총괄하여 "게스트 소프트웨어"라고 함)를 실행하는 자립적 플랫폼으로 각각 기능할 수 있다.

[0112] 각각의 게스트 OS 및/또는 게스트 소프트웨어는 가상 머신이 아니라 전용 컴퓨터 상에서 실행한 것처럼 동작한다. 즉, 각각의 게스트 OS 및/또는 게스트 소프트웨어는 다양한 이벤트를 제어하고 플랫폼(800)의 하드웨어 리소스에 액세스하는 것을 예상할 수 있다. 각 VM 내에서, 게스트 OS 및/또는 게스트 소프트웨어는 실제로 플랫폼(800)의 물리적 하드웨어(네트워크 컨트롤러(860)를 포함할 수 있는 "호스트 하드웨어(840)")에서 실행하는 것처럼 행동할 수 있다.

[0113] 도 1의 매니저빌러티 엔진(ME)(130) 등의 전용 프로세서가 있는 물리적 하드웨어 파티션은 (도 8에 도시된) 가상화 파티션보다 더 높은 보안 수준을 제공할 수 있지만, 본 발명의 실시예는 다양한 수준의 보안을 제공하는 환경 및/또는 이들 환경의 조합에서 실시될 수 있음은 당업자가 쉽게 알 것이다. 또한, ME, AMT 또는 PRL 플랫폼은 가상화 환경에서 구현될 수 있다는 것도 당업자가 쉽게 알 것이다. 예컨대, VM(810)이 호스트 상에서 전형적 애플리케이션을 실행하는 동안 VM(820)은 호스트 상에서 ME 파티션으로서 전용될(dedicated) 수 있다. 이 시나리오에서, 호스트는 다수의 프로세서를 포함할 수도 있고 그렇지 않을 수도 있다. 호스트가 두 개의 프로세서를 포함하면, 예컨대, VM(810)(및 호스트 상의 다른 VM)이 프로세서(805)의 리소스를 공유할 수 있는 반면 VM(820)은 다른 프로세서에 할당될 수 있다. 한편, 호스트가 하나의 프로세서만 포함하면, 프로세서는 양쪽 VM

에 모두 서비스할 수 있지만, VM(820)은 VMM(830)의 협조를 얻어 호스트의 다른 VM들과 여전히 분리될 수 있다. 간략화를 위해, 본 발명의 실시예는 매니저빌러티 엔진(ME) 환경에서 기술되었지만, 본 발명의 실시예는 그렇게 한정되는 것이 아니다. 대신에 매니저빌러티 엔진, ME, "파티션", "안전한 파티션", "보안 파티션" 및/또는 "관리 파티션"의 어떤 것도 (상술한 바와 같이) 임의의 물리적 및/또는 가상 파티션을 포함할 것이다.

[0114] 시동시 또는 새로운 장치가 플랫폼으로 핫플러그되는 경우, VMM(830)은 VM(810 또는 820)을 할당한다. 도 8에 기술된 바와 같은 가상화 환경에서 칩셋/보안 파티션(120) 내의 감사를 수행하기 위해, VMM(830)은 VM(810, 820)의 각각에 대한 감사 마스크 프로파일을 관리한다. 장치가 VM(810) 또는 VM(820)에 할당되면, VM에 대한 각각의 감사 마스크 프로파일이 칩셋/보안 파티션(120)과 연관된다. 칩셋/보안 파티션(120)과 연관된 VM 감사 마스크 프로파일이 변할 때마다, VMM(830)은 감사 이벤트 기록을 생성한다. 이런 식으로, 감사가능한 이벤트를 개시하는 VM(810 또는 820)은 감사 이벤트 기록에 표현된다. 예컨대, 장치에 스토리지 I/O 커맨드를 발행하는 VM(810 또는 820)은 감사 이벤트 기록에서 식별된다.

[0115] 장치가 플랫폼에 핫플러깅되면, 장치 할당을 수신한 VM(810 또는 820)은 감사 이벤트 기록에서 식별된다. 핫플러그 이벤트가 검출되면, I/O 커맨드 디코드 모듈(140)은 칩셋/보안 파티션(120)과 현재 연관된 VM(810 또는 820)이 장치 할당을 수신하도록 허가되는지 여부를 판정할 필요가 있을 수 있다. 장치가 할당되고, 칩셋/보안 파티션(120)과 연관될 정확한 감사 마스크 프로파일이 결정될 수 있을 때까지, 내부 감사 마스크 프로파일은 핫플러그 이벤트 후에 장치 할당이 발생할 때까지 이벤트를 감사하는 데 사용될 수 있다.

[0116] VMM(830)은 플래시 메모리(190)에 현재 활성화된 감사 마스크 프로파일을 기입함으로써 칩셋/보안 파티션에 대해 현재 활성화된 VM 감사 마스크 프로파일을 식별할 수 있다. 또한 플래시 메모리(190)는 각 VM과 연관된 사용자 계정 메타데이터를 유지하는 데 사용된다. 스토리지 장치가 장치 비밀번호 또는 장치 암호화 키를 이용하여 잠금해제되려고 하는 경우, 플래시 메모리(190)의 사용자 계정 메타데이터가 장치가 할당된 VM에 대응하는지를 보장하기 위해 추가 확인이 수행될 수 있다.

[0117] VMM(830)은 과도기적 VM 환경이 드라이브의 허가되지 않은 할당을 초래하지 않도록 보장한다. 일 실시예에서, VMM(830)은 각 VM(810, 820)에 대해 GUID(globally unique ID)를 생성한다. GUID는 플래시 메모리(190)의 메타데이터를 분할하는 데 사용된다.

[0118] 여기에 개시된 메커니즘의 실시예는 하드웨어, 소프트웨어, 펌웨어 또는 그러한 구현 방식의 조합으로 구현될 수 있다. 본 발명의 실시예는 적어도 하나의 프로세서, 데이터 스토리지 시스템(휘발성 및 비휘발성 및/또는 스토리지 요소를 포함함), 적어도 하나의 입력 장치 및 적어도 하나의 출력 장치를 포함하는 프로그래머블 시스템에서 실행하는 컴퓨터 프로그램으로서 구현될 수 있다.

[0119] 프로그램 코드는 여기에 기술된 기능을 수행하고 출력 정보를 생성하기 위해 입력 데이터에 적용될 수 있다. 본 발명의 실시예는 또한 본 발명의 동작을 수행하거나, 여기에 기술된 구조, 회로, 장치, 프로세서 및/또는 시스템 특징을 정의하는, HDL 등의 설계 데이터를 포함하는 명령을 포함하는, 기계에 의해 액세스 가능한 매체를 포함한다. 그러한 실시예는 또한 프로그램 제품으로 불릴 수 있다.

[0120] 그러한 기계에 의해 액세스 가능한 매체는 하드디스크 등의 스토리지 매체, 플로피 디스크, 광 디스크, 컴팩트 디스크 판독전용 메모리(CD-ROM), 컴팩트 디스크 재기록형(CD-RW), 자기 광 디스크(magneto-optical disk)를 포함하는 임의의 다른 형태의 디스크, 판독전용 메모리(ROM), 동적 랜덤 액세스 메모리(DRAM), 정적 랜덤액세스 메모리(SRAM) 등의 랜덤액세스 메모리(RAM), 소거가능한 프로그래머블 판독전용 메모리(EPROM), 플래시 프로그래머블 메모리(FLASH), 전기적으로 소거가능한 프로그래머블 판독전용 메모리(EEPROM) 등의 반도체 장치, 자기 또는 광 카드 또는 전자적 명령을 저장하기에 적당한 임의의 다른 타입의 매체를 포함하는, 기계 또는 장치에 의해 제조 또는 형성된 입자의 유형적 배치를 포함할 수 있지만 이것에 제한되지는 않는다.

[0121] 출력 정보는 공지된 방식으로 하나 이상의 출력 장치에 공급될 수 있다. 이 애플리케이션을 위해, 처리 시스템은, 예컨대, 디지털 신호 프로세서(DSP), 마이크로컨트롤러, ASIC(application specific integrated circuit) 또는 마이크로프로세서 등의 프로세서를 갖는 임의의 시스템을 포함한다.

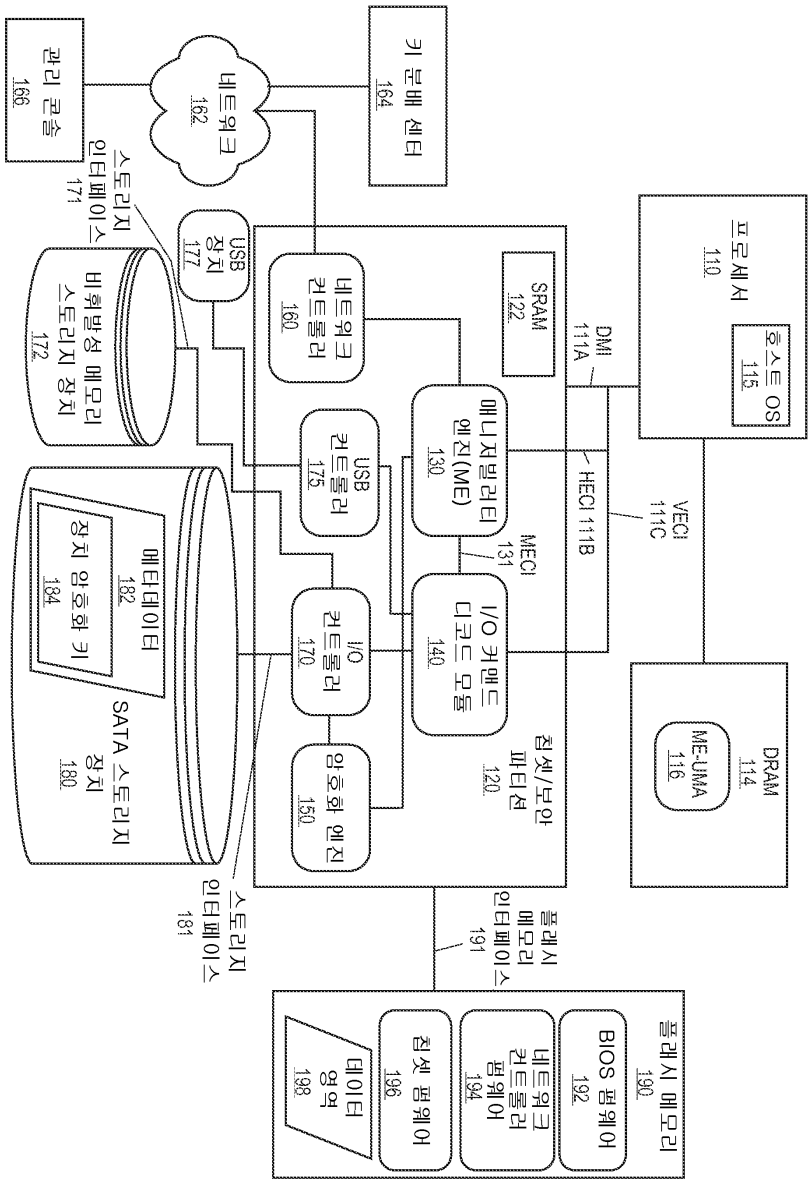
[0122] 프로그램은 처리 시스템과 통신하기 위해 하이레벨 절차언어(high-level procedural language) 또는 객체지향 프로그래밍 언어로 구현될 수 있다. 프로그램은 바람직하다면, 어셈블리 또는 기계 언어로 구현될 수도 있다. 실제로, 여기에 기술된 메커니즘은 임의의 특정한 프로그래밍 언어로 범위를 제한하는 것이 아니다. 임의의 경우에 언어는 컴파일 또는 해석된 언어일 수 있다.

[0123] 암호화, 사용자 인증, 비밀번호 보호 방식에 의해 보호된 장치의 관리를 위한 방법 및 시스템의 실시예가 여기

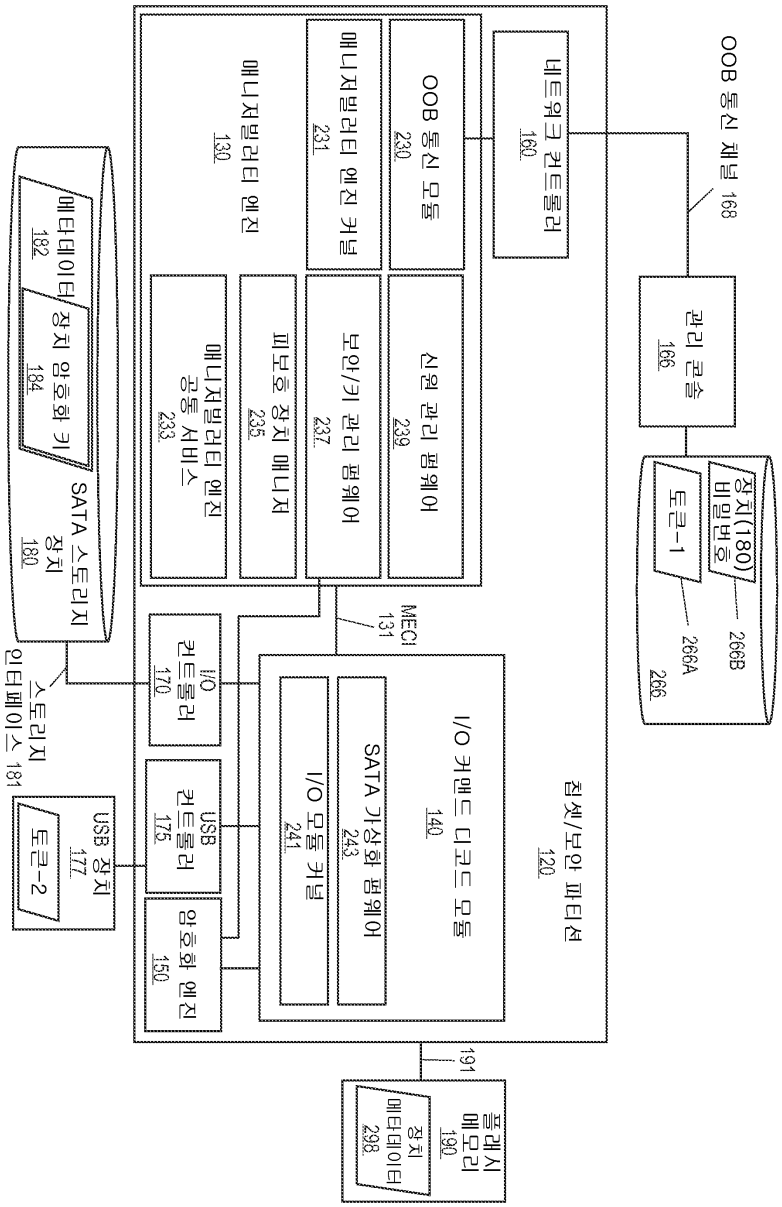
에 제시된다. 본 발명의 특정 실시예가 도시 및 기술되었지만, 첨부된 청구범위의 범위내에서 여러가지 변경, 변형 및 수정이 이루어질 수 있음은 당업자에게 명백할 것이다. 따라서, 당업자는 보다 넓은 관점에서 본 발명으로부터 벗어나지 않고 변경 및 수정이 이루어질 수 있음을 이해할 것이다. 첨부된 청구범위는 그 범위 내에서 본 발명의 진실한 범위 및 정신 내의 그러한 모든 변경, 변형 및 수정을 포함하는 것이다.

도면

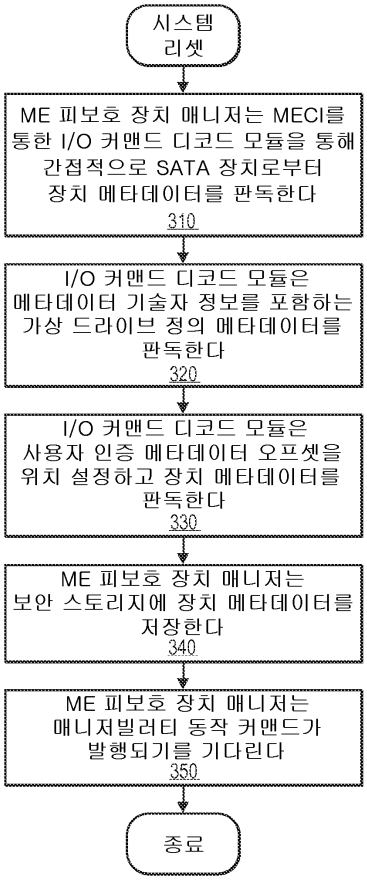
도면1



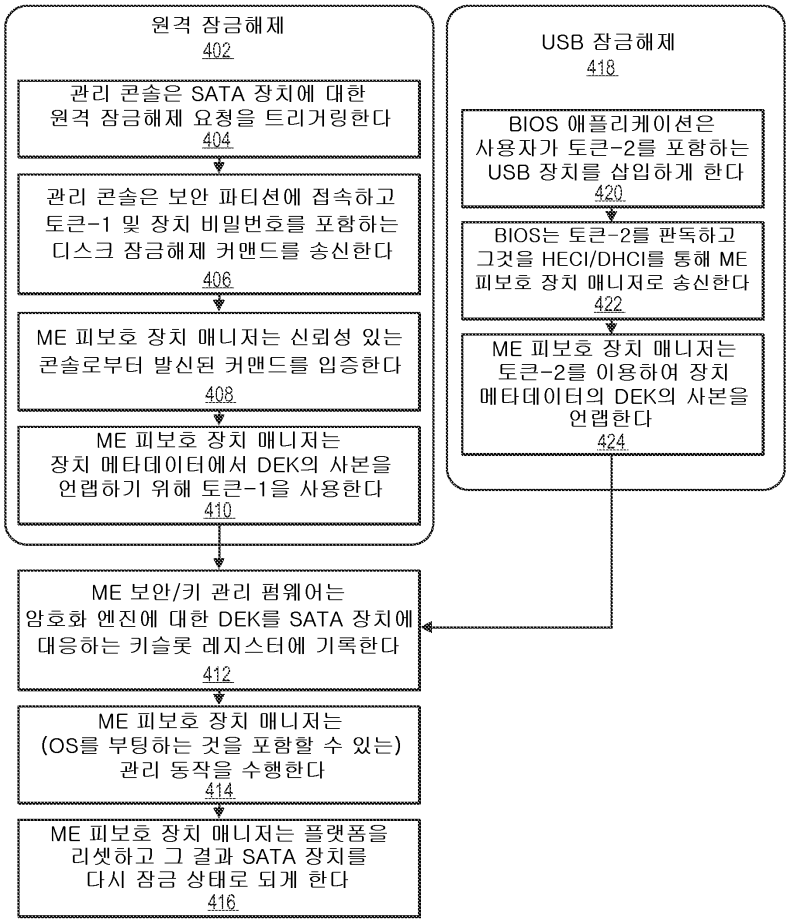
도면2


OOB 통신 채널 168
관리 콘솔 166
장치(180) 비밀번호
토큰-1
266B
266A
266
칩셋/보안 파티션 120
네트워크 컨트롤러 160
OOB 통신 모듈 239
매니저빌리티 엔진 커널 231
매니저빌리티 엔진 130
신원 관리 펌웨어 239
보안/키 관리 펌웨어 237
피보호 장치 매니저
매니저빌리티 엔진 공통 서비스 233
MECI 131
I/O 커맨드 디코드 모듈 140
SATA 가상화 펌웨어 243
I/O 모듈 커널 241
I/O 컨트롤러 170
USB 컨트롤러 175
암호화 엔진 150
191
플래시 메모리 190
장치 메타데이터 298
메타데이터/장치 암호화 키 182/184
SATA 스토리지 장치 180
스토리지 인터페이스 181
USB 장치 177
토큰-2

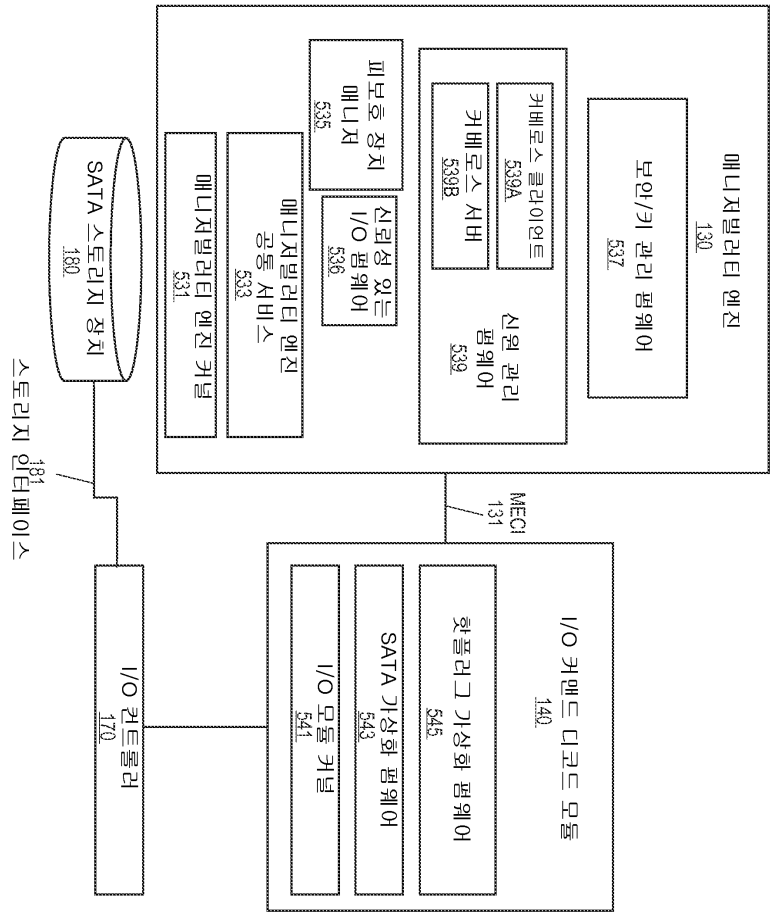
도면3


시스템
리셋
ME 피보호 장치 매니저는 MECI를
통한 I/O 커맨드 디코드 모듈을 통해
간접적으로 SATA 장치로부터
장치 메타데이터를 판독한다
310
I/O 커맨드 디코드 모듈은
메타데이터 기술자 정보를 포함하는
가상 드라이브 정의 메타데이터를
판독한다
320
I/O 커맨드 디코드 모듈은
사용자 인증 메타데이터 오프셋을
위치 설정하고 장치 메타데이터를
판독한다
330
ME 피보호 장치 매니저는
보안 스토리지에 장치 메타데이터를
저장한다
340
ME 피보호 장치 매니저는
매니저빌러티 동작 커맨드가
발행되기를 기다린다
350
종료

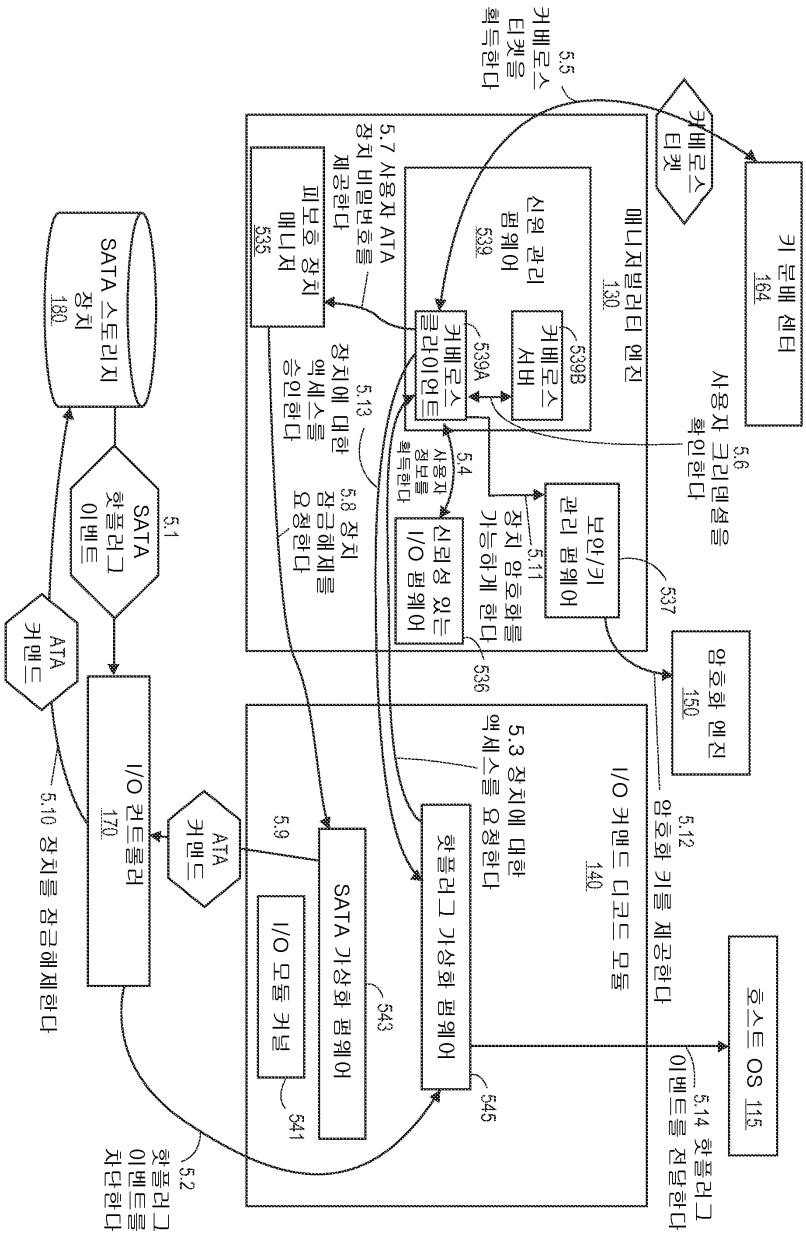
도면4


원격 잠금해제
402
관리 콘솔은 SATA 장치에 대한 원격 잠금해제 요청을 트리거링한다
404
관리 콘솔은 보안 파티션에 접속하고 토큰-1 및 장치 비밀번호를 포함하는 디스크 잠금해제 커맨드를 송신한다
406
ME 피보호 장치 매니저는 신뢰성 있는 콘솔로부터 발신된 커맨드를 입증한다
408
ME 피보호 장치 매니저는 장치 메타데이터에서 DEK의 사본을 언랩하기 위해 토큰-1을 사용한다
410
USB 잠금해제
418
BIOS 애플리케이션은 사용자가 토큰-2를 포함하는 USB 장치를 삽입하게 한다
420
BIOS는 토큰-2를 판독하고 그것을 HECI/DHCI를 통해 ME 피보호 장치 매니저로 송신한다
422
ME 피보호 장치 매니저는 토큰-2를 이용하여 장치 메타데이터의 DEK의 사본을 언랩한다
424
ME 보안/키 관리 펌웨어는 암호화 엔진에 대한 DEK를 SATA 장치에 대응하는 키슬롯 레지스터에 기록한다
412
ME 피보호 장치 매니저는 (OS를 부팅하는 것을 포함할 수 있는) 관리 동작을 수행한다
414
ME 피보호 장치 매니저는 플랫폼을 리셋하고 그 결과 SATA 장치를 다시 잠금 상태로 되게 한다
416

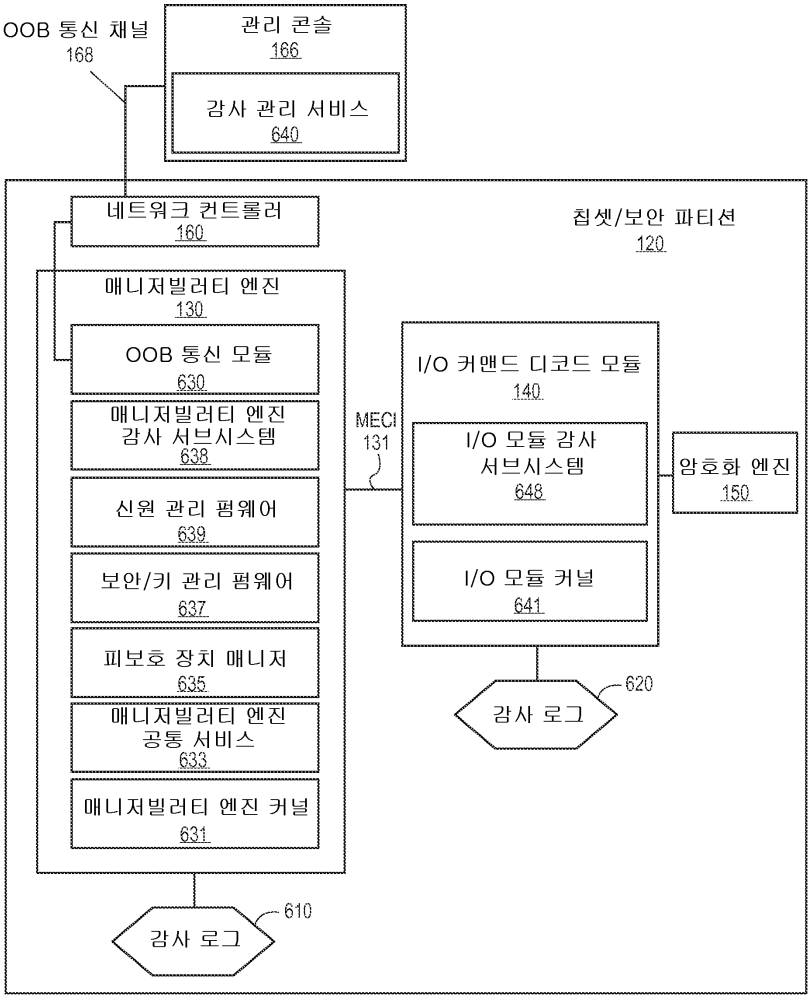
도면5a


매니저빌러티 엔진
130
보안/키 관리 펌웨어
537
신원 관리
펌웨어
539
커베로스 클라이언트
539A
커베로스 서버
539B
피보호 장치
매니저
535
신뢰성 있는
I/O 펌웨어
536
매니저빌러티 엔진
공통 서비스
533
매니저빌러티 엔진 커널
531
MECI
131
I/O 커맨드 디코드 모듈
140
핫플러그 가상화 펌웨어
545
SATA 가상화 펌웨어
543
I/O 모듈 커널
541
I/O 컨트롤러
170
181
스토리지 인터페이스
SATA 스토리지 장치
180

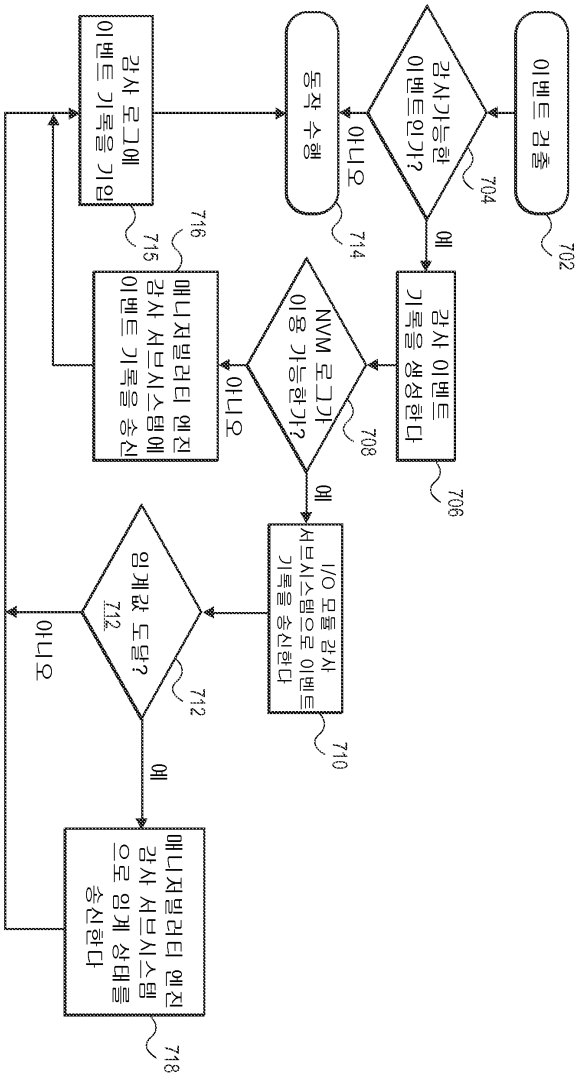
도면5b


키 분배 센터
164
케베로스 티켓
5.5 케베로스 티켓을 획득한다
매니저빌러티 엔진
130
신원 관리 펌웨어
539
케베로스 서버
539B
케베로스 클라이언트
539A
5.6 사용자 크리덴셜을 확인한다
5.7 사용자 ATA 장치 비밀번호를 제공한다
피보호 장치 매니저
535
5.4 사용자 정보를 획득한다
보안/키 관리 펌웨어
537
5.11 장치 암호화를 가능하게 한다
신뢰성 있는 I/O 펌웨어
536
5.13 장치에 대한 엑세스를 승인한다
5.8 장치 잠금해제를 요청한다
암호화 엔진
150
5.12 암호화 키를 제공한다
I/O 커맨드 디코드 모듈
140
5.3 장치에 대한 엑세스를 요청한다
핫플러그 가상화 펌웨어
545
SATA 가상화 펌웨어
543
I/O 모듈 커널
541
5.9
ATA 커맨드
5.1 SATA 핫플러그 이벤트
SATA 스토리지 장치
180
ATA 커맨드
I/O 컨트롤러
170
5.10 장치를 잠금해제한다
5.2 핫플러그 이벤트를 차단한다
5.14 핫플러그 이벤트를 전달한다
호스트 OS 115

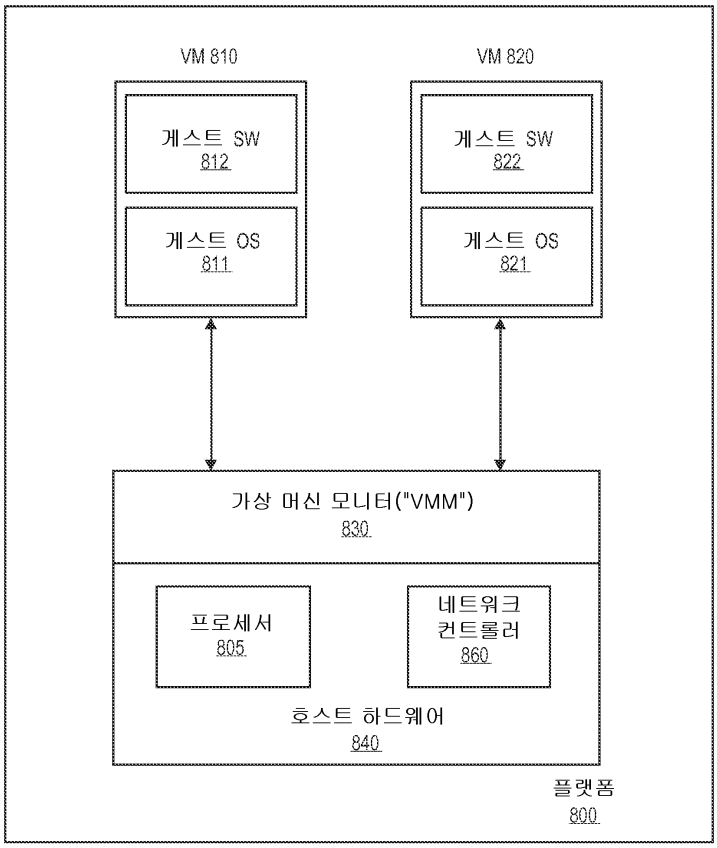
도면6


OOB 통신 채널
168
관리 콘솔
166
감사 관리 서비스
640
네트워크 컨트롤러
160
칩셋/보안 파티션
120
매니저빌러티 엔진
130
OOB 통신 모듈
630
매니저빌러티 엔진
감사 서브시스템
638
신원 관리 펌웨어
639
보안/키 관리 펌웨어
637
피보호 장치 매니저
635
매니저빌러티 엔진
공통 서비스
633
매니저빌러티 엔진 커널
631
감사 로그
610
MECI
131
I/O 커맨드 디코드 모듈
140
I/O 모듈 감사
서브시스템
648
I/O 모듈 커널
641
암호화 엔진
150
감사 로그
620

도면7


이벤트 감출
702
감시기능할 이벤트인가?
704
예
아니오
감시 이벤트 기록을 생성한다
706
NVM 로그가 이용 가능한가?
708
예
아니오
I/O 모듈 감시 서브시스템으로 이벤트 기록을 송신한다
710
임계값 도달?
712
예
아니오
매니저빌러티 엔진 감시 서브시스템으로 임계 상태를 송신한다
718
동작 수행
714
감시 로그에 이벤트 기록을 기입
715
716
매니저빌러티 엔진 감시 서브시스템에 이벤트 기록을 송신

도면8



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 청구항 10

【변경전】

'상기 처리시스템의 상기 호스트 운영 시스템으로부터'

【변경후】

'상기 처리 시스템의 호스트 운영 시스템으로부터'