(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2013년07월30일
(11) 등록번호 10-1275773
(24) 등록일자 2013년06월11일

(51) 국제특허분류(Int. Cl.)
G06F 21/60 (2013.01) *G06F 11/34* (2006.01)
G06F 11/30 (2006.01) *G06F 15/16* (2006.01)
(21) 출원번호 10-2011-0100649
(22) 출원일자 2011년10월04일
심사청구일자 2011년10월04일
(65) 공개번호 10-2013-0036522
(43) 공개일자 2013년04월12일
(56) 선행기술조사문헌
KR1020050054665 A*
논문(한국정보과학회_2010.06)
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
한국과학기술원
대전광역시 유성구 대학로 291(구성동)
(72) 발명자
박규호
대전 유성구 구성동 한국과학기술원 6-3208
박기웅
서울특별시 노원구 광운로15길 48 (월계동)
(74) 대리인
이원희

전체 청구항 수 : 총 15 항 심사관 : 박진아

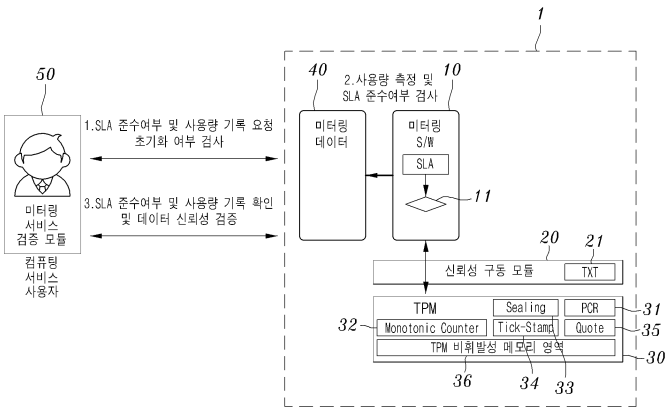
(54) 발명의 명칭 **클라우드 컴퓨팅 서비스에서 생성된 미터링 데이터에 대한 위조 및 변조를 방지하는 시스템**

(57) 요 약

본 발명은 클라우드 컴퓨팅과 같이 사용한 만큼만 돈을 지불을 하는 Pay-as-you-go pricing model 기반의 컴퓨팅 서비스 시스템에 있어서 위조 및 변조가 불가능한 미터링 데이터의 생성 및 미터링 데이터를 검증하는 방법에 관한 것으로서, 컴퓨터의 관리자조차도 컴퓨터 내부에서 발생한 미터링 데이터에 대한 내용을 함부로 고칠 수 없도록 하는 미터링 데이터 생성 방법 및 검증 방법을 제공한다.

본 발명의 구현을 위한 미터링 시스템은 Trusted Platform Module (TPM)이라 불리는 하드웨어 보안 모듈; Trusted Execution Technology (TXT)라 불리는 하드웨어 보안 모듈; 컴퓨팅 시스템의 미터링 연산을 수행하는 프로그램 모듈; 검증 연산을 수행하는 프로그램 모듈;을 포함하여 구성된다.

대 표 도 - 도4



이 발명을 지원한 국가연구개발사업

과제고유번호	KI002153
부처명	지식경제부
연구사업명	산업원천기술개발사업
연구과제명	독립형 컴포넌트 기반 서비스 지향형 페타급 컴퓨팅 플랫폼 기술개발
주관기관	한국클라우드컴퓨팅연구조합
연구기간	2009.03.02 ~ 2012.02.29

특허청구의 범위

청구항 1

클라우드 컴퓨팅 서비스에 있어서,

상기 클라우드 컴퓨팅 시스템에 설치된 미터링 소프트웨어 유닛이 초기화되어 구동되는 제 1 단계;

상기 클라우드 컴퓨팅 시스템에서 클라우드 컴퓨팅 서비스가 실행되는 제 2 단계;

상기 제 2 단계에서 구동되는 상기 클라우드 컴퓨팅 서비스에 대한 SLA(Service Level Agreement)의 준수 여부를 모니터링하는 제 3 단계;

상기 제 3 단계에서 상기 SLA의 위반 사항이 검출되면, 검출된 SLA 위반 메시지를 상기 미터링 소프트웨어 유닛에 기록하는 제 4 단계; 및

상기 클라우드 컴퓨팅 서비스를 종료하면, 기록된 상기 SLA 위반 메시지를 포함하는 미터링 데이터를 생성하는 제 5 단계;를 포함하며,

상기 생성된 미터링 데이터에 대한 위조 및 변조를 방지하기 위해 상기 미터링 데이터는 신뢰 플랫폼 모듈(Trusted Platform Module, TPM)에 의해 보안 기능이 제공되는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 방법.

청구항 2

제 1 항에 있어서,

상기 제 1 단계에서 구동되는 상기 미터링 소프트웨어 유닛은, TXT(Trusted Execution Technology) 유닛을 포함하는 신뢰성 구동모듈에 의해 실행되는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 방법.

청구항 3

삭제

청구항 4

제 1 항에 있어서,

상기 미터링 데이터에 대해 보안 기능을 제공하는 상기 신뢰 플랫폼 모듈은, 실행되는 상기 미터링 소프트웨어 유닛에 대한 해쉬(Hash)값을 저장하는 적어도 하나 이상의 플랫폼 구성 레지스터(Platform Configuration Register, PCR), 상기 미터링 데이터에 대해 단방향으로 하나씩만 카운터를 증가하는 모노토닉 카운터(Monotonic Counter), 상기 미터링 데이터를 안전하게 저장하고 저장된 미터링 데이터에 대한 무결성을 보장하는 실링(Sealing) 유닛 및 틱(Tick) 정보를 생성하여 상기 미터링 데이터에 디지털 서명을 수행하는 틱 스탬프(Tick Stamp) 유닛을 포함하는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 방법.

청구항 5

제 1 항에 있어서,

상기 제 3 단계에서 모니터링 되는 상기 SLA는 서비스 실패 허용시간, 사용시간, 스토리지 및 네트워크의 대역폭, 소프트웨어 라이센스 개수의 군으로 이루어지는 서비스 퀄리티로 정의되는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 방법.

청구항 6

제 4 항에 있어서,

상기 제 5 단계에서 기록되는 미터링 데이터는 시드(Seed) 값을 기반으로 모니터링되는 각 메시지마다 종속성을 부여하고, 최종적으로 상기 TPM을 통한 디지털 서명을 포함하는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 방법.

청구항 7

제 6 항에 있어서,

상기 미터링 데이터는 상기 미터링 소프트웨어의 초기화 시 기록되는 서비스 초기화 기록 메시지, 검출된 SLA 위반 메시지 및 클라우드 컴퓨팅 서비스 종료시 기록되는 서비스 종료 기록 메시지를 포함하는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 방법.

청구항 8

제 1 항에 있어서,

상기 제 5 단계 이후에 생성된 상기 미터링 데이터를 사용자측 컴퓨팅 시스템의 검증 모듈에서 검증하는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 방법.

청구항 9

클라우드 컴퓨팅 시스템에 있어서,

상기 클라우드 컴퓨팅 시스템에 설치된 미터링 소프트웨어를 실행하는 신뢰성 구동모듈; 및

상기 클라우드 컴퓨팅 서비스 시 상기 신뢰성 구동모듈을 통해 실행된 상기 미터링 소프트웨어에 의해 생성된 미터링 데이터에 대해 보안 기능을 제공하는 신뢰 플랫폼 모듈(Trusted Platform Module, TPM);을 포함하는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템.

청구항 10

제 9 항에 있어서,

상기 신뢰성 구동모듈은 TXT(Trusted Execution Technology) 유닛을 더 포함하고,

상기 TXT 유닛은 명시된 프로그램 리스트에 따라 구동될 프로그램의 해쉬값이 상기 프로그램 리스트에 존재하는 경우에만 프로그램을 실행시키는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템.

청구항 11

제 9 항에 있어서,

상기 신뢰 플랫폼 모듈은,

실행되는 상기 미터링 소프트웨어에 대한 해쉬(Hash)값을 저장하는 적어도 하나 이상의 플랫폼 구성 레지스터(Platform Configuration Register, PCR);

상기 미터링 데이터에 대해 단방향으로 하나씩만 카운터를 증가하는 모노토닉 카운터(Monotonic Counter);

상기 미터링 데이터를 안전하게 저장하고 저장된 미터링 데이터에 대한 무결성을 보장하는 실링(Sealing) 유닛; 및

틱(Tick) 정보를 생성하여 상기 미터링 데이터에 디지털 서명을 수행하는 틱 스탬프(Tick Stamp) 유닛;을 포함하는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템.

청구항 12

제 9 항에 있어서,

상기 미터링 소프트웨어는 상기 클라우드 컴퓨팅 서비스 시 SLA(Service Level Agreement)의 준수 여부를 모니터링하여 상기 SLA의 위반 사항이 검출되면, 검출된 SLA 위반 메시지를 미터링 데이터에 기록하는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템.

청구항 13

제 12 항에 있어서,

상기 SLA는 서비스 실패 허용시간, 사용시간, 스토리지 및 네트워크의 대역폭, 소프트웨어 라이센스 개수의 군으로 이루어지는 서비스 퀄리티로 정의되는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템.

청구항 14

제 12 항에 있어서,

상기 미터링 데이터는 시드(Seed) 값을 기반으로 모니터링되는 각 메시지마다 종속성이 부여되고, 최종적으로 상기 TPM을 통한 디지털 서명이 포함되는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템.

청구항 15

제 12 항에 있어서,

상기 미터링 데이터는 상기 미터링 소프트웨어 유닛의 초기화 시 기록되는 서비스 초기화 기록 메시지, 검출된 SLA 위반 메시지 및 클라우드 컴퓨팅 서비스 종료시 기록되는 서비스 종료 기록 메시지를 포함하는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템.

청구항 16

제 9 항에 있어서,

상기 미터링 소프트웨어에 의해 생성된 상기 미터링 데이터를 검증하는 검증 모듈을 더 포함하는 것을 특징으로 하는 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템.

명 세 서

기 술 분 야

[0001] 본 발명은 클라우드 컴퓨팅과 같이 사용한 만큼만 돈을 지불을 하는(Pay-as-you-go) 가격 책정 모델(pricing model) 기반의 컴퓨팅 서비스 시스템에서 생성된 미터링 데이터에 대한 위조 및 변조를 방지하는 방법 및 시스

템에 관한 것으로서, 컴퓨터의 관리자조차도 컴퓨터 내부에서 발생한 미터링 데이터를 함부로 고칠 수 없도록 하여 클라우드 컴퓨팅 시스템에서 생성된 미터링 데이터에 대한 위조 및 변조를 방지하는 방법 및 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템을 제공하기 위한 것이다.

배 경 기 술

[0002] 최근 클라우드 컴퓨팅(cloud computing)과 같은 인터넷 기반(cloud)의 컴퓨팅(computing) 기술이 활성화 되고 있다. 이러한 클라우드 컴퓨팅 환경에서 사용자들은 지원하는 기술 인프라 스트럭처에 대한 전문 지식이 없거나 제어하는 방법을 모른다고 하더라도 인터넷으로부터 쉽게 서비스를 이용할 수 있다.

[0003] 공개특허 제2011-0040604호는 제한된 자원을 가지는 클라이언트 단말이 클라우드 컴퓨팅 서비스를 제공하는 클라우드 서버의 자원을 이용하는 클라우드 컴퓨팅 시스템에 관한 것으로, 클라우드 서버가 통신부, 가상 디바이스 드라이버 매니저, 및 컴퓨팅 서비스 제공부를 포함하여, 클라우드 서버가 클라이언트 단말로부터 원하는 정보가 실행될 디바이스를 클라이언트 단말이 이용하도록 지원하는 컴퓨팅 서비스 요청을 수신하면, 컴퓨팅 서비스 요청에 따라 디바이스에 대한 가상 디바이스 드라이버를 검색하여 검색된 가상 디바이스 드라이버를 이용하여 클라이언트 단말에 컴퓨팅 서비스를 제공할 수 있다.

[0004] 또한, 공개특허 제2011-0038909호는 가맹점 PC방의 사용자 단말로 클라우드 컴퓨팅 서비스를 제공하여 개별 사용자에게 이용 요금을 과금하는 PC방 컨텐츠 제공 시스템에 관한 것이다. 특히 공개특허 제2011-0038909호는 클라우드 컴퓨팅 처리되어 수신된 컨텐츠의 화면 표시 기능을 갖는 PC방 사용자 단말과, 사용자 단말로 초기 화면을 제공하고 초기 화면에서 인증 번호 및 특정 컨텐츠의 이용 신청을 전송받아 인증을 수행하여 가상 서버의 접속 정보를 제공하는 게이트웨이 서버와, 복수 개 가상 서버의 생성, 기동 및 소멸을 관리하여 사용자 단말에 1 : 1로 할당되는 가상 서버를 배정하여 접속 정보를 전송하는 가상화 관리 서버와, 가상화 관리 서버의 제어를 받아 실행되고, 사용자 단말로부터 사용자 명령을 수신하여 클라우드 컴퓨팅 처리된 컨텐츠 제공 서비스를 제공하고, 개별 사용자 단말의 이용 시간 및 이용 컨텐츠에 대한 로그를 기록하는 가상 서버, 및 PC방 가맹점 별 이용 단가를 적용하여 과금 처리하고, 과금된 개별 사용자 단말의 이용 요금을 포탈 서버로 전송하는 과금 서버로 구성되어, 클라우드 컴퓨팅 시스템에 대한 사용자의 컨텐츠 이용 현황을 기준으로 개별 과금하도록 하고 있다.

[0005] 상기와 같이, 클라우드 컴퓨팅 시스템을 이용하여 사용한 만큼만 돈을 지불을 하는 가격 책정 모델 기반의 컴퓨팅 서비스가 보편화 되고 있어, 서비스 형태로 제공되는 컴퓨팅 자원의 사용 및 서비스 퀄리티(Quality)에 대한 기록을 믿을 수 있는 방법으로 미터링을 하고 이를 믿을 수 있는 방법으로 기록하기 위한 기술이 요구가 되고 있다. 이는 사용한 만큼 돈을 지불을 하는 서비스 모델에서는 각 사용자가 제공받은 서비스에 대한 사용 시간 및 약속받은 서비스의 수준으로 서비스가 제공이 되었는지의 여부를 믿을 수 있는 방법으로 미터링이 되어야 보다 믿을 수 있는 서비스가 되기 때문이다.

[0006] 일반적인 미터링 기술과 관련하여, 도 1은 전기 및 수도 서비스의 과금 서비스의 미터링 예를 도시화한 것으로, 사용한 만큼만 돈을 지불하는 서비스 모델인 수도 및 전기의 경우, 각각의 수도장치(100) 및 전기장치(110)에는 미터링 기기(101, 111)가 각각 구비되어 있어 사용자와 공급자는 얼마만큼의 서비스를 사용했는지 명확하게 알 수 있고, 각각의 미터링 기기(101, 111)의 내부는 쉴딩(102, 112)이 되어 있어, 사용자 및 공급자가 미터링 기기의 내부 값을 함부로 못 바꾸도록 설계가 되어 있기 때문에, 믿을 수 있는 방법으로 로깅이 될 수 있는 것이다.

[0007] 그러나, 수도 및 전기와 같이 하나의 단일 단위로, 예를 들어 전기의 경우 W, 수도의 경우 M^3로 측정될 수 있는 것과 달리, 클라우드 컴퓨팅 시스템은 서비스 실패 허용 시간, 사용 시간, 스토리지 및 네트워크의 대역폭, 소프트웨어 라이센스 개수 등 매우 다양한 메트릭으로 서비스의 퀄리티가 정의되고, 이는 일반적으로 SLA(Service

Level Agreement)라는 형식으로 정의가 된다.

[0008] 그러므로, 수도 및 전기 서비스와 비교하여 매우 다양한 메트릭으로 측정을 해야하며, 이모든 측정은 컴퓨터의 소프트웨어를 기반으로 이루어지기 때문에, 수도 및 전기 서비스의 미터링과 같이 상호적으로 믿을 수 있으면서도, 검증이 가능한 미터링 방법을 제공하는데 있어서 매우 어려운 문제를 야기시키게 된다.

[0009] 도 2는 위조 변조가 가능한 기존의 컴퓨팅 시스템의 미터링을 도시화한 것으로서, 컴퓨팅 시스템 내부에는 소프트웨어로 구성된 미터링 모듈(201)이 구비되고, 미터링 모듈(201)에서 SLA를 검사하여, SLA에 대한 위반 사항이 있을시 미터링 모듈(201)은 내부 데이터 공간에 위반 사항을 기록하고, 위반되었을 시 또는 서비스가 종료된 시점에 미터링 데이터(202)를 통해 사용자(200)에게 보고를 하는 방법을 취하게 된다. 하지만 이와 같은 로깅 방법은 보안적 측면에 있어서 여러 가지 제한점을 가지게 된다. 예를 들어, 서비스 제공자는 미터링된 데이터의 내용을 바꾸어 사용자에게 더 많은 과금이 나오게 할 수 있으며, 실제 만족해야 할 서비스의 퀄리티의 만족 여부를 기록한 데이터를 바꾸어, 비록 서비스의 퀄리티를 만족시키지 못했더라도, 마치 요건을 만족시킨 것처럼 정보를 바꿀 수 있는 등 여러 가지 보안에 있어 취약한 문제가 발생할 수 있다는 문제점이 있다.

발명의 내용

해결하려는 과제

[0010] 따라서, 본 발명의 기술적 과제는 상기한 바와 같은 종래 기술에 나타낸 문제점을 해결하기 위한 것으로, 컴퓨팅 시스템 내부에 탑재된 보안 모듈을 활용하여 클라우드 컴퓨팅 시 생성된 미터링 데이터에 대한 위조 및 변조를 방지하는 방법 및 시스템의 제공을 목적으로 한다.

[0011] 그러나 본 발명의 목적은 상기에 언급된 목적으로 제한되지 않으며, 언급되지 않은 또 다른 목적들은 아래의 기재로부터 당업자에게 명확하게 이해될 수 있을 것이다.

과제의 해결 수단

[0012] 상기 목적을 달성하기 위하여, 클라우드 컴퓨팅 서비스에서 생성된 미터링 데이터에 대한 위조 및 변조를 방지하는 방법은, 상기 클라우드 컴퓨팅 시스템에 설치된 미터링 소프트웨어 유닛이 초기화되어 구동되는 제 1 단계; 상기 클라우드 컴퓨팅 시스템에서 클라우드 컴퓨팅 서비스가 실행되는 제 2 단계; 상기 제 2 단계에서 구동되는 상기 클라우드 컴퓨팅 서비스에 대한 SLA(Service Level Agreement)의 준수 여부를 모니터링하는 제 3 단계; 상기 제 3 단계에서 상기 SLA의 위반 사항이 검출되면, 검출된 SLA 위반 메시지를 상기 미터링 소프트웨어 유닛에 기록하는 제 4 단계; 및 상기 클라우드 컴퓨팅 서비스를 종료하면, 기록된 상기 SLA 위반 메시지를 포함하는 미터링 데이터를 생성하는 제 5 단계;를 포함하는 것을 특징으로 한다.

[0013] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 방법은, 상기 제 1 단계에서 구동되는 상기 미터링 소프트웨어 유닛은, TXT(Trusted Execution Technology) 유닛을 포함하는 신뢰성 구동모듈에 의해 실행되는 것을 특징으로 한다.

[0014] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 방법은, 상기 제 5 단계에서 생성되는 상기 미터링 데이터는 신뢰 플랫폼 모듈(Trusted Platform Module, TPM)에 의해 보안 기능이 제공되는 것을 특징으로 한다.

[0015] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 방법은, 상기 미터링 데이터에 대해 보안 기능

을 제공하는 상기 신뢰 플랫폼 모듈은, 실행되는 상기 미터링 소프트웨어 유닛에 대한 해쉬(Hash)값을 저장하는 적어도 하나 이상의 플랫폼 구성 레지스터(Platform Configuration Register, PCR), 상기 미터링 데이터에 대해 단방향으로 하나씩만 카운터를 증가하는 모노토닉 카운터(Monotonic Counter), 상기 미터링 데이터를 안전하게 저장하고 저장된 미터링 데이터에 대한 무결성을 보장하는 실링(Sealing) 유닛 및 틱(Tick) 정보를 생성하여 상기 미터링 데이터에 디지털 서명을 수행하는 틱 스탬프(Tick Stamp) 유닛을 포함하는 것을 특징으로 한다.

[0016] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 방법은, 상기 제 3 단계에서 모니터링 되는 상기 SLA는 서비스 실패 허용시간, 사용시간, 스토리지 및 네트워크의 대역폭, 소프트웨어 라이센스 개수의 군으로 이루어지는 서비스 퀄리티로 정의되는 것을 특징으로 한다.

[0017] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 방법은, 상기 제 5 단계에서 기록되는 미터링 데이터는 시드(Seed) 값을 기반으로 모니터링되는 각 메시지마다 종속성을 부여하고, 최종적으로 상기 TPM을 통한 디지털 서명을 포함하는 것을 특징으로 한다.

[0018] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 방법은, 상기 미터링 데이터가 상기 미터링 소프트웨어의 초기화 시 기록되는 서비스 초기화 기록 메시지, 검출된 SLA 위반 메시지 및 클라우드 컴퓨팅 서비스 종료시 기록되는 서비스 종료 기록 메시지를 포함하는 것을 특징으로 한다.

[0019] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 방법은, 상기 제 5 단계 이후에 생성된 상기 미터링 데이터를 사용자측 컴퓨팅 시스템의 검증 모듈에서 검증하는 것을 특징으로 한다.

[0020] 본 발명에 따른 클라우드 컴퓨팅 서비스에서 생성된 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템은, 상기 클라우드 컴퓨팅 시스템에 설치된 미터링 소프트웨어를 실행하는 신뢰성 구동모듈; 및 상기 클라우드 컴퓨팅 서비스 시 상기 신뢰성 구동모듈을 통해 실행된 상기 미터링 소프트웨어에 의해 생성된 미터링 데이터에 대해 보안 기능을 제공하는 신뢰 플랫폼 모듈(Trusted Platform Module, TPM);을 포함하는 것을 특징으로 한다.

[0021] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템은, 상기 신뢰성 구동모듈이 TXT(Trusted Execution Technology) 유닛을 더 포함하고, 상기 TXT 유닛은 명시된 프로그램 리스트에 따라 구동될 프로그램의 해쉬값이 상기 프로그램 리스트에 존재하는 경우에만 프로그램을 실행시키는 것을 특징으로 한다.

[0022] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템은, 상기 신뢰 플랫폼 모듈이, 실행되는 상기 미터링 소프트웨어 유닛에 대한 해쉬(Hash)값을 저장하는 적어도 하나 이상의 플랫폼 구성 레지스터(Platform Configuration Register, PCR); 상기 미터링 데이터에 대해 단방향으로 하나씩만 카운터를 증가하는 모노토닉 카운터(Monotonic Counter); 상기 미터링 데이터를 안전하게 저장하고 저장된 미터링 데이터에 대한 무결성을 보장하는 실링(Sealing) 유닛; 및 틱(Tick) 정보를 생성하여 상기 미터링 데이터에 디지털 서명을 수행하는 틱 스탬프(Tick Stamp) 유닛;을 포함하는 것을 특징으로 한다.

[0023] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템은, 상기 미터링 소프트웨어 유닛이 상기 클라우드 컴퓨팅 서비스 시 SLA(Service Level Agreement)의 준수 여부를 모니터링하여 상기 SLA의 위반 사항이 검출되면, 검출된 SLA 위반 메시지를 미터링 데이터에 기록하는 것을 특징으로 한다.

[0024] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템은, 상기 SLA이 서비스 실패 허용시간, 사용시간, 스토리지 및 네트워크의 대역폭, 소프트웨어 라이센스 개수의 군으로 이루어지는 서비스 퀄리티로 정의되는 것을 특징으로 한다.

[0025] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템은, 상기 미터링 데이터가 시드(Seed) 값을 기반으로 모니터링되는 각 메시지마다 종속성을 부여하고, 최종적으로 상기 TPM을 통한 디지털 서명이 포함되는 것을 특징으로 한다.

[0026] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템은, 상기 미터링 데이터가 상기 미터링 소프트웨어 유닛의 초기화 시 기록되는 서비스 초기화 기록 메시지, 검출된 SLA 위반 메시지 및 클라우드 컴퓨팅 서비스 종료시 기록되는 서비스 종료 기록 메시지를 포함하는 것을 특징으로 한다.

[0027] 본 발명에 따른 미터링 데이터에 대한 위조 및 변조를 방지하는 클라우드 컴퓨팅 시스템은, 상기 미터링 소프트웨어 유닛에 의해 생성된 상기 미터링 데이터를 검증하는 검증 모듈을 포함하는 것을 특징으로 한다.

발명의 효과

[0028] 이상에서 살펴본 바와 같이, 본 발명은 종래의 Pay-as-you-go pricing model 방식에서 수반되는 미터링 방식의 단점들, 즉 미터링된 데이터에 대한 위/변조를 방지하게 되어, 종래 시스템에서 발생되는 보안의 취약성을 해결할 수 있어, 안전하고 신뢰성이 보장된 방법으로 미터링 데이터를 생성하고, 또한, 생성된 미터링 데이터를 검증할 수 있는 이점이 있다.

도면의 간단한 설명

[0029] 도 1은 일반적인 전기 및 수도 서비스의 과금 서비스의 미터링 시스템을 나타내는 예시도이다.

도 2는 위조 및 변조가 가능한 기존의 컴퓨팅 시스템의 미터링 시스템의 동작을 나타내는 도면이다.

도 3은 본 발명에 따른 클라우드 컴퓨팅 서비스에서 생성된 미터링 데이터에 대한 위조 및 변조를 방지하는 방법을 나타내는 흐름도이다.

도 4은 본 발명에 따른 미터링 소프트웨어 유닛이 적재된 클라우드 컴퓨팅 시스템에서 미터링을 수행하는 연산 과정을 나타내는 도면이다.

도 5는 본 발명에서 사용하는 하드웨어 기반 보안 모듈인 TPM 및 TXT 기반의 수행 동작을 나타내기 위한 도면이다.

도 6은 본 발명에 따른 미터링 데이터 생성을 수행하기 위한 메시지 트랜잭션의 과정을 나타내는 도면이다.

도 7은 본 발명에 따른 미터링 데이터 생성을 수행하기 위한 메시지 트랜잭션 연산의 과정을 나타내는 도면이다.

발명을 실시하기 위한 구체적인 내용

[0030] 이하, 본 발명의 바람직한 실시 예의 상세한 설명은 첨부된 도면들을 참조하여 설명할 것이다. 하기에서 본 발명을 설명함에 있어서, 관련된 공지 기능 또는 구성에 대한 구체적인 설명이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명을 생략할 것이다.

[0031] 본 발명의 개념에 따른 실시 예는 다양한 변경을 가할 수 있고 여러 가지 형태를 가질 수 있으므로 특정 실시 예들을 도면에 예시하고 본 명세서 또는 출원에 상세하게 설명하고자 한다. 그러나, 이는 본 발명의 개념에

따른 실시 예를 특정한 개시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.

[0032] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.

[0033] 본 명세서에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 설시된 특징, 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0034] 도 3은, 본 발명에 따른 클라우드 컴퓨팅 서비스에서 생성된 미터링 데이터에 대한 위조 및 변조를 방지하는 방법을 나타내는 흐름도이다. 도면을 참조하면, 사용자 측 컴퓨팅 시스템에서 클라우드 컴퓨팅 시스템에 서비스를 요청하면, 먼저, 클라우드 컴퓨팅 시스템에 설치된 미터링 소프트웨어 유닛이 초기화되어 구동된다(S10). 이후, 클라우드 컴퓨팅 시스템에서는 사용자가 요청한 클라우드 컴퓨팅 서비스가 실행되고(S20), 이때, 클라우드 컴퓨팅 시스템에서는 클라우드 컴퓨팅 서비스에 대한 SLA(Service Level Agreement)의 준수 여부를 모니터링한다(S30). 이 SLA는 서비스 실패 허용시간, 사용시간, 스토리지 및 네트워크의 대역폭, 소프트웨어 라이센스 개수의 군으로 이루어지는 서비스 퀄리티로 정의될 수 있다.

[0035] 만약, 단계(S30)에서, SLA의 위반 사항이 검출되면(S40), 검출된 SLA 위반 메시지를 상기 미터링 소프트웨어 유닛에 기록하고, 사용자가 클라우드 컴퓨팅 서비스에 대한 종료를 요청하면, 기록된 SLA 위반 메시지를 포함하는 미터링 데이터를 생성한다(S50). 이후에, 사용자는 사용자측 컴퓨팅 시스템에서 생성된 미터링 데이터를 수신하여 이를 검증할 수 있다(S60).

[0036] 이하, 도 4 내지 도 8을 참조하여 미터링 데이터에 대한 위조 및 변조를 방지하는 방법 및 그 시스템에 대해 설명한다. 도 4에 도시한 바와 같이, 클라우드 컴퓨팅 시스템(1)은 하드웨어적인 방법으로 데이터에 대한 안전한 저장 및 프로그램 무결성 검증 기능을 제공하는 신뢰 플랫폼 모듈(TPM)(30), 검증된 프로그램만을 안전한 방법으로 실행시키기 위한 TXT(Trusted Execution Technology) 유닛(21)을 포함하고, TPM(30)과 TXT 유닛(21)에 기반하여 미터링 프로그램을 신뢰성 있는 방법으로 실행시키기 위한 신뢰성 구동 모듈(20), 신뢰성 구동 모듈(20)이 실행된 후, 시스템(1) 내부의 상태 및 SLA를 측정하여 기록하는 미터링 소프트웨어 유닛(10), 미터링된 데이터의 검증을 위한 검증 모듈(50)을 포함할 수 있다.

[0037] 신뢰성 구동 모듈(20)의 TXT(Trusted Execution Technology) 유닛(21)은 사용자가 명시한 프로그램 리스트에 한해서 실행을 시키도록 한다. 사용자는 자신이 구동할 프로그램 리스트의 해쉬값을 명시하면, 새로운 프로그램이 실행이 될 때마다, 리스트를 검사하여, 구동될 프로그램의 해쉬 값이 프로그램 리스트에 있을 경우에만 실행을 시킬 수 있도록 함으로써, 신뢰된 프로그램만을 안전하게 실행할 수 있게 되며, 미터링 소프트웨어 유닛(10)은 TXT 유닛(21)을 기반으로 실행이 되는 것을 특징으로 한다.

[0038] TPM(30)은 일종의 보안 하드웨어의 장치로서 암호화 관련 기능과 중요한 데이터를 저장할 수 있는 보안 기능을 제공하며, 현재 많은 컴퓨팅 장치에 TPM이 장착되어 있다. 본 발명은, 도 5에 도시된 바와 같이 TPM(30)에서 제공하는 보안 기술을 활용하게 된다.

[0039] 플랫폼 구성 레지스터(Platform Configuration Register, PCR)(31)는 수행할 프로그램에 대한 무결성 검사 기능을 수행한다. TPM(30) 내부에는 PCR(31)이라 불리는 레지스터가 다수 존재하며, PCR(31)은 현재의 컴퓨팅 시스템(1)의 플랫폼 상태를 저장하게 된다. 예를 들어, 하나의 컴퓨팅 시스템이 구동이 되기 위해서는, (A)CMOS 롬 내부의 코드 --> (B) 부트로더 --> (C) 운영체제 --> (D) 프로그램 등의 일련의 순서로 실행이 되며 이를 부팅이라 부르는데, PCR(31)에는 해쉬 함수를 이용하여 구동할 프로그램에 대한 상태를 저장하게 된다. 예를 들어, 부팅 과정중 (A)CMOS 롬 내부의 코드를 실행하기 전에, A' = Hash (A)를 수행하여 A' 값을 PCR0에 저장시키며, (B)부트로더를 실행하기 전에 상기 A'값과 부트로더 코드 값을 해쉬 함수에 입력하여 얻은 값을 다음과 같이, PCR1에 B' = Hash (B + 부트로더 코드)를 저장시키며, (C)운영체제를 실행시키기 전에 C'=Hash (B' + 운영체제 코드)를 PCR2에 저장시키고, (D)프로그램을 실행시키기 전에 D' = hash (C' + 프로글매 코드)를 PCR3에 저장시키게 되는 것이다. 이와 같은 일련의 연산과정을 통해 PRC0, PCR1, PCR2, PCR3 내부에 저장된 값을 확인하여, 현재의 컴퓨팅 시스템이 "어떤 프로그램"이 "어떠한 순서"로 실행되었는지의 여부를 검증할 수 있게 된다. 이러한 부팅 과정에서 사용한 해쉬 함수는 단방향적인 특성을 가지는 것으로서 결과값을 알고 있다해도 해쉬에 입력된 값을 알기는 매우 힘들기 때문에, 변조를 가한 프로그램을 중간에 삽입을 하여도, 변조가 되지 않은 프로그램과 동일한 Hash값을 나오게 하는 것은 매우 힘들게 된다.

[0040] 이와 같은 원리로, 미터링 소프트웨어 유닛(10)의 미터링 프로그램을 구동하려고 하는 사용자는 각 PCR 값에 저장되어 있는 값을 확인하여, 자신의 미터링 프로그램((D)프로그램)이 자신이 알고 있는 운영체제 및 그 이하의 소프트웨어 (A, B, C)와 함께 수행했는지 여부를 판단하기 위해서는 PCR0 - PCR3값을 자신이 알고 있는 올바른 값과 비교를 통해 검증을 할 수 있게 되는 것이다. 각각의 PCR 값(PCR0 - PCR3)은 TPM(30) 내부에서 연산이 되고 기록이 되므로, TPM(30)은 PCR(31)의 값을 임의의 값으로 변경할 수 없도록 하여, 미터링 소프트웨어 유닛(10)을 구동하려는 사용자는 PCR값의 검증을 통해 자신의 미터링 프로그램이 안전하게 초기화 및 구동이 되는지를 검사할 수 있게 되는 것이다.

[0041] 보다 상세하게는, TPM(30)의 Quote 연산 유닛(35)은, PCR(31)의 각 값에 대하여, TPM(30)만이 생성할 수 있는 디지털 서명 정보를 삽입하여 미터링 소프트웨어 유닛(10)을 구동한 사용자에게 전달하고, 사용자는 해당하는 디지털 서명연산이 TPM(30)에 의해서 생성된 것인지를 해당 TPM(30)의 AIK(Attestation Identity Key)를 이용하여 검증을 할 수 있게 된다.

[0042] 모노토닉 카운터(Monotonic Counter)(32)는 단방향으로 하나씩만 증가를 하는 카운터로서, 특정 프로그램만이 카운터 값을 변경할 수 있도록 만들 수 있다. 예를 들어 사용자가 구동하려는 미터링 소프트웨어 유닛(10)의 프로그램만이 특정 모노토닉 카운터 값을 변경할 수 있도록 만들 수 있는 기능을 제공해 준다. 이와 같은 모노토닉 카운터의 기능을 활용하여 미터링 소프트웨어 유닛(10)에 대한 실행의 무결성 여부를 검사한다.

[0043] 실링(Sealing) 유닛(33)은 실링 연산 기능을 제공하는데, 이는 특정 데이터를 안전하게 저장하면서도 저장된 데이터에 대한 무결성을 보장해 주기 위한 기능을 제공해 준다. 즉 상기 미터링 소프트웨어 유닛(10)이 실링 연산을 이용하여 데이터를 암호화하면, 미터링 소프트웨어 유닛(10)만이 해당하는 데이터를 해독할 수 있도록 하드웨어적으로 보호해 주는 기능을 제공한다. 이는 PCR(31)의 특성을 활용한 것으로서, 미터링 소프트웨어 유닛(10)이 실행될 때, PCR(31)의 값은 미터링 소프트웨어 유닛(10)의 해쉬 값으로 변경되고, 미터링 소프트웨어 유닛(10)은 PCR값과 함께 특정 데이터를 암호화시키고, 해독을 할 시에는 미터링 소프트웨어 유닛(10)에 상응하는 PCR값을 가질 때에만 데이터를 해독할 수 있도록 함으로써, 미터링 소프트웨어 유닛(10)에서 생성한 데이터를 보호할 수 있는 기능을 제공해 주는 것이다. 또한 TPM(30) 내부에는 비휘발성 메모리 영역(36)이 존재하여, 데이터를 실링 연산을 활용하여 TPM(30) 내부에 안전하게 저장할 수 있다. 미터링 소프트웨어 유닛(10)이 실링 연산을 이용하여 TPM(30) 내부의 비휘발성 메모리 영역(36)에 데이터를 저장을 하였다면, 미터링 소프트웨어 유닛(10)만이 이 저장된 데이터를 해독할 수 있으며, 비휘발성 메모리 영역(36) 뿐만 아니라, 미터링 소프트웨어 유닛(10)이 적재된 컴퓨팅 시스템(1)의 별도의 스토리지 영역에도 실링 유닛(33)에 의한 연산을 통하여 데이터를 저장할 수 있게 된다.

[0044] TPM(30)은 하드웨어 내부적으로 발생한 틱(Tick) 정보를 생성하고 이를 TPM(30) 내부에서 디지털 서명을 하여 출력하는 기능을 제공하는데, 틱 스탬프(Tick Stamp) 유닛(34)은 틱(Tick) 정보에 TPM(30)에 의해 생성된 디지털 서명을 포함시킨다. 따라서, 미터링 소프트웨어 유닛(10)을 구동한 사용자는 틱 스탬프 유닛(34)에 의한 틱 스탬프가 TPM(30)에 의해서 생성된 것임을 검증할 수 있게 된다. 따라서, 본 발명은 이러한 특징을 이용하여 미터링된 데이터의 기록시마다 틱 스탬프를 활용하여 해당 로깅이 정확히 언제 기록이 되었는지를 기록할 수 있게 된다.

[0045] 이러한 TPM(30) 및 TXT 유닛(21)의 기능을 활용한 시스템은 도 4에 나타낸 바와 같이, 클라우드 컴퓨팅 서비스를 사용하는 사용자 측 컴퓨팅 시스템(50), 사용자에게 서비스를 제공하는 클라우드 컴퓨팅 시스템(1) 및 클라우드 컴퓨팅 시스템(1)을 미터링 하는 미터링 소프트웨어 유닛(10)으로 구성된다. 미터링 소프트웨어 유닛(10)을 구동하여 자신이 제공받는 서비스의 SLA를 미터링 하려고 하는 사용자가 미터링 소프트웨어 유닛(10)을 TXT 유닛(21)을 활용하여 신뢰성 있는 방법으로 실행시키고, 제대로 초기화 및 구동이 되었는지를 확인하는 과정(제 1 과정)을 거치고, 사용자가 클라우드 컴퓨팅 서비스를 시작함과 동시에, 미터링 소프트웨어 유닛(10)이 SLA 준수여부를 모니터링하고, SLA 위반 사항이 있다면, 위반한 내용을 안전하게 기록부(11)에 기록한 후(제 2 과정), 사용자가 컴퓨팅 시스템의 사용을 종료함과 동시에 미터링 소프트웨어 유닛(10)에 의해 기록된 데이터를 미터링 데이터(40)로 생성하여 수신하고, 사용자 측 컴퓨팅 시스템(50)의 검증 모듈을 통해 검증할 수 있다(제 3 과정).

[0046] 따라서, 사용자는 신뢰성 구동 모듈(20)에 의해 안전한 방법으로 미터링 소프트웨어 유닛(10)의 구동을 제 1 과정을 통해 수행하여 미터링 소프트웨어 유닛(10)을 안전하게 구동시킬 수 있으며, 제 2 과정을 수행하여 사용자가 제공받은 미터링 소프트웨어 유닛(10)이 생성한 로깅 데이터를 변경 및 삭제를 할 수 없는 특징을 갖게 된다. 마지막으로 서비스를 종료할 시에는 제 3 과정을 수행하여 미터링 소프트웨어 유닛(10)에 의해 생성된 미터링 데이터(40)를 검증하여, 최종적으로 사용자는 자신이 제공받은 서비스에 대한 SLA 준수 여부를 검증할 수 있게 된다.

[0047] 이하, 상술한 제 1 과정 내지 제 3 과정의 일련의 연산 흐름은, 본 발명의 기술적 특징인 내부적인 메시지 교환 방법 및 연산 방법을 통하여 실행되고, 다음의 메시지 표기법에 따라 메시지 교환 방법 및 연산을 하게 된다. 또한 본 발명에 따른 바람직한 실시예를 도 6 및 도 7을 참조하여 보다 상세하게 설명한다.

[0048] <메시지 표기법>

[0049] 1. {A | B} : A와 B로 구성된 단일 메시지

[0050] 2. H(A) : 메시지 A를 입력 값으로 한 도출한 해쉬 함수 결과값

[0051] 3. Tick Stamp : TPM에 의해서 생성된 디지털 서명 정보가 포함되어 있는 Tick 값

[0052] 4. SK : 디지털 서명을 위한 개인키

[0053] 5. PK : 디지털 서명의 검증을 위한 공개키

[0054] 6. N : Nonce의 약자로서, 랜덤하게 생성된 일련의 랜덤값으로서, 데이터의 무결성 및 Replay 공격을 방지하기 위한 데이터

[0055] 7. NV_Write() : TPM 내부의 실링 연산을 수행하여, TPM 내부 비휘발성 메모리 영역에 저장하는 연산

[0056] 8. Extend() : PCR을 이용한 수행할 프로그램 무결성 검사를 위한 함수

[0057] 9. Key_Generation() : 사용자와 미터링 소프트웨어 유닛 간의 디지털 서명 및 검증을 위한 개인키(SK) 및 공개키(PK)를 생성

[0058] 10. Quote() : 현재의 PCR값에 대하여 디지털 서명을 생성하는 연산

[0059] 본 발명의 기술 (Description)을 위하여 메시지 표기는 '{'와 '}'를 이용하여 시작과 끝을 나타내며, {A | B}는

하나의 메시지로서 A와 B의 데이터로 구성된 하나의 메시지를 의미한다. 또한 {A}K는 A라는 데이터를 K라는 키로 암호화된 데이터를 의미한다.

[0060] 'H(A)'는 해쉬 연산의 결과값을 의미하며, A를 해쉬 함수에 입력하여 얻은 결과값을 나타내며, 본 발명의 해쉬 연산은 안전한 해쉬 함수, 예를 들어, SHA-1, SHA-256 등의 보안적으로 안정성이 증명되어 해쉬의 결과값으로 입력의 원본값을 유추해 내기 힘든 해쉬 함수를 사용한다.

[0061] 'tick-stamp' 표기는 TPM(30)에 의해서 생성된 틱 스탬프 정보를 의미하며 'tick-stamp' 내부에는 TPM(30)에 의해서 생성된 디지털 서명 정보가 포함되어 있는 것으로 약속한다.

[0062] 'SK' 기호는 Secret Key의 약자로서, 디지털 서명을 생성하는데 사용하는 개인키를 의미하며, 'PK' 기호는 Public Key의 약자로서, 디지털 서명 값을 검증하는데 사용하는 공개키를 의미한다.

[0063] 'N' 기호는 Nonce의 약자로서, 랜덤하게 생성된 일련의 랜덤값으로서, 데이터의 무결성 및 Replay 공격을 방지하는데 사용한다.

[0064] 'NV_Write()'는 TPM(30) 내부의 실링 연산을 수행하여, TPM(30) 내부 비휘발성 메모리 영역(36)에 저장하는 연산을 의미하며, 보다 상세하게는 NV_Write(PCR18, Message)의 경우 데이터 Message를 18번째의 PCR값을 이용하여 암호화시켜(Sealing), 비휘발성 메모리 영역(36)에 저장하는 것을 의미하므로, NV_Write()를 수행했을 때의 PCR18번 값이 해독을 할 때의 PCR18값과 일치를 해야만 Message를 얻을 수 있는 특징을 갖게 된다.

[0065] 'Extend()' 함수는 PCR(31)을 이용한 수행할 프로그램 무결성 검사 기능에 해당하는 것으로, PCR값을 해쉬 함수를 이용해 새로운 값을 유도하는 것을 의미하며, Extend(PCR18, Data)의 경우 현재의 PCR18값을 Data값과 함께 해쉬 함수에 넣어 새로운 PCR18값을 유도해내는 것을 의미한다.

[0066] 'Key_Generation()' 함수는 사용자와 미터링 소프트웨어 유닛(10) 간의 디지털 서명 및 검증을 위한 개인키(SK) 및 공개키 (PK)를 생성하는 것을 의미한다.

[0067] 'Quote()' 함수는 현재의 PCR값에 대하여 디지털 서명을 생성하는 연산으로서, 'Quote(PCR17, Data)'는 Data 및 PCR17번 값에 대한 디지털 서명을 생성하여, Data 및 PCR17에 저장된 값이 TPM(50)의 개인키를 이용하여 디지털 서명이 되었는지를 검증을 할 수 있도록 하는 기능을 하게 된다.

[0068] 도 6 및 도 7은 기술된 표기법과 상기 언급된 보안 기술을 활용하여 미터링 및 로깅을 하는 메시지 트랜잭션 및 연산의 일련의 과정을 나타낸다.

[0069] 본 발명에 따른 클라우드 컴퓨팅 서비스에 대한 미터링은 총 3 과정으로 이루어져 있으며, 초기화하는 제 1 과정, 실제 서비스를 사용하는 동안 미터링 및 로깅을 수행하는 제 2 과정, 서비스 종료 후 미터링 된 데이터를 검증하는 제 3 과정으로 구성될 수 있다.

[0070] 제 1 과정에서는 초기화를 하는 단계로서, 상기 언급한 TXT 유닛(21)으로 미터링 소프트웨어 유닛(10)을 구동할 때는 PCR17의 값이 미터링 소프트웨어 유닛(10)의 해쉬값으로 초기화가 된다. 미터링 소프트웨어 유닛(10)을 구동할 사용자는 S값과 랜덤한 값인 Nonce (N)을 보내게 되는데 S값은 SLA가 기술된 내용으로서, 미터링 소프트웨어 유닛(10)이 모니터링할 매트릭과 보장해야할 최소 수치가 기록되어 있다. 만약 SLA의 한 항목으로서 디스크 Bandwidth가 20MB/sec 이상으로 기술되었다면, 미터링 소프트웨어 유닛(10)은 현재의 디스크 Bandwidth가 20MB/sec 미만으로 측정되었을 경우에는 해당 위반사항을 기록하게 되는 것이다. 상기 메시지를 수신한 미터링 소프트웨어 유닛(10)은 상기 S 값의 해쉬 값인 H(S)값과, TPM(30)으로부터 현재의 Tick-Stamp값을 읽어와 Extend() 함수에 입력하여 PCR18값에 저장하게 되고, 이는 추후 미터링 소프트웨어 유닛(10)이 제대로 초기화가 되었는지를 검사하는데 사용되게 된다.

[0071] 그 후, 미터링 소프트웨어 유닛(10)은 사용자와 미터링 소프트웨어 유닛(10) 사이에 디지털 서명 및 검증을 위한 키인 SK과 PK를 생성하며, 사용자로부터 수신한 랜덤값 N을 Seed 값으로 초기화 하게 된다. 이때 Seed 값은 추후 모니터링 데이터에 삽입되어 로깅된 데이터의 무결성을 검증하는데 사용된다. 그 후에 상기 생성되

었던 키 중에 디지털 서명을 위한 키인 SK와 사용자로부터 수신한 SLA 데이터 및 TPM(30) 내부의 모노토닉 카운터 값인 Counter, 상기 Seed 값을 PCR17에 Extend 함수를 이용하여 저장하게 되고, 또한 위의 데이터를 미터링 된 데이터를 저장하는 공간인 Black-Box 형태의 데이터(BB)(60)에 실링 유닛(33)을 통해 저장하게 되며, 최종적으로 세번의 Extend 함수를 수행하여 PCR18번과 PCR17번을 업데이트 시키게 된다. 여기서 H(PCR18, H(PK))와 H(PCR18, 0xFF)에서 마지막에 0xFF값을 이용하여 Extend 연산을 하는 이유는, PCR값을 시스템 관리자가 추후에 읽더라도 중간값을 읽지 못하게 하기 위하여 마지막 Extend 연산에 있어서는 0xFF를 입력값으로 업데이트하는 것을 특징으로 한다. 또한 PCR17에 대해서도 Extend 연산을 0xFF를 입력값으로 업데이트 하는 것을 위와 같은 목적으로 수행하는 것을 특징으로 한다.

[0072] 마지막으로, Quote() 연산을 수행하는데 입력값은 PCR17번의 값과 PCR18번의 값 그리고 N을 입력값으로 하며 Quote()연산의 출력값은 q로 출력한다. 미터링 소프트웨어 유닛(10)을 구동한 사용자는 자신이 입력했던 S값과 Nonce를 이용하여 제대로 초기화가 되었는지를 검증할 수 있도록 하며, 이를 최종적으로 q값을 검증하는데 사용되는 AIK, PK, 초기화가 완료된 시간이 기록된 Tick Stamp를 사용자와 미터링 소프트웨어 사이에 공유된 키를 이용하여 암호화를 시킨 후 클라우드 컴퓨팅 시스템(1)에서 사용자 측 컴퓨팅 시스템(50)으로 전송을 하게 된다. 이를 수신한 사용자는 자신이 보낸 S값과 N값을 이용하여 초기화가 제대로 되었는지를 해쉬 함수와 AIK를 이용하여 검증해 볼 수 있게 되어 초기화 과정이 종료된다.

[0073] 제 2 과정은 실제 클라우드 컴퓨팅 서비스를 받는 동안 주기적으로 수행이 되는 일련의 과정으로서, 신뢰성 구동 모듈(20)을 통해 미터링 소프트웨어 유닛(10)을 실행하기 때문에, PCR17값을 미터링 소프트웨어 유닛(10)의 해쉬값으로 초기화가 되며, 제 1 과정에서 TPM(30)의 비휘발성 메모리 영역(36)에 저장되었던 데이터인 SLA, Monotonic Counter, Seed값을 읽어오는데, 저장할 때 PCR17값으로 실링이 되었기 때문에 미터링 소프트웨어 유닛(10)만이 이값을 읽어올 수 있게 된다. 이때, 미터링 소프트웨어 유닛(10)은 모노토닉 카운터값을 읽어 TPM(30)의 비휘발성 메모리 영역(36)에 저장되었던 카운터값과 일치를 할 경우에만 구동을 하게 된다. 일치를 할 경우에는 SLA_Monitoring 함수를 실행시켜, 현재의 SLA의 준수여부를 검사하게 되며, 만약 SLA를 위반한 사항이 발견될 경우에는 위반이 된 정보를 BB(블랙박스 형태의 데이터를 지칭하여 BB라 명명)(60)에 저장하게 되는데, 저장 정보는 위반사항이 기록된 Violation이라는 항목, 현재의 Tick Stamp 정보, Seed 값으로 구성이 되며, Seed 값은 제 1 과정 시에 초기화 되었던 값에, Violation 및 Tick Stamp 정보와 함께 업데이트 되는 것을 특징으로 하며, 매번 Violation이 발생할 시마다, Seed 값은 업데이트가 되어 이전 Seed값과 의존성이 있게 되는 것을 특징으로 한다.

[0074] 제 2 과정의 마지막 연산으로 모노토닉 카운터값을 1 증가시키고, Seed값을 PCR17과 함께 실링 유닛(33)을 통하여 TPM(30)의 비휘발성 메모리 영역(36)에 저장하게 되는데, 매번 카운터값을 증가시켜 저장을 하며, 제 2 과정의 시작지점에 있어서 카운터값의 일치 여부를 판단하도록 함으로써, 추후 미터링 소프트웨어 유닛(10)의 실행 여부에 대한 무결성을 검증할 수 있는 것을 특징으로 한다.

[0075] 제 3 과정은 서비스가 종료되는 시점에 수행되는 과정으로서, 사용자 측 컴퓨팅 시스템(50)이 미터링 소프트웨어 유닛(10)의 종료를 위한 요청 메시지를 보내게 되며, 이를 수신한 미터링 소프트웨어 유닛(10)은 TPM(30) 내부의 비휘발성 메모리 영역(36)으로부터 SK값과 counter, Seed값을 읽어와, PCR17값과 함께, 실링 유닛(33)을 통해 실링 연산을 수행하여 이를 BB(60)에 저장하게 되며, PCR17과 PCR18번에 대해서 Extend 연산을 통하여 업데이트 하게 된다. 이때 PCR17번을 0xFF로 업데이트하는 이유는 제 1 과정에서 0xFF로 업데이트하는 것과 같은 이유로서, PCR값을 시스템 관리자가 추후에 읽더라도 중간값을 읽지 못하게 하기 위하여 마지막 Extend 연산에 있어서는 0xFF를 입력값으로 업데이트하는 것을 특징으로 한다. 또한 PCR18에 대해서도 Extend 연산을 0xFF를 최종 입력값으로 업데이트하는 것을 위와 같은 목적으로 수행한다.

[0076] 이후, 미터링 소프트웨어 유닛(10)은 제 1 과정부터 제 2 과정까지 BB(60)에 기록된 데이터를 해독하기 위한 unsealing 연산을 수행하게 되며, 해독된 BB 데이터를 미터링 소프트웨어를 수행한 사용자로부터 수신한 N과 함께 Quote 연산을 수행하여, 사용자로 하여금, 생성된 BB데이터(60)가 미터링 소프트웨어 유닛(10)에 의해서 생

성된 데이터라는 것을 증명할 수 있게 되며, BB 데이터(60)에는 미터링 소프트웨어 유닛(10)의 초기화 시 기록되는 서비스 초기화 기록 메시지(61), 검출된 SLA 위반 메시지(62) 및 클라우드 컴퓨팅 서비스 종료시 기록되는 서비스 종료 기록 메시지(62)로 구성되고, 또한 BB데이터(60)는 제 1 과정에서 생성된 SK에 의해서 디지털 서명을 생성하여 Quote 연산의 결과인 q값과 함께 사용자에게 전송하게 되어, 사용자는 수신한 정보를 사용자 측 컴퓨팅 시스템(50)의 검증 모듈을 이용하여 검증을 할 수 있게 된다.

[0077] 상기 본 발명의 내용은 도면에 도시된 일 실시 예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시 예가 가능하다는 점을 이해할 것이다. 따라서 본 발명의 진정한 기술적 보호 범위는 첨부된 특허청구범위의 기술적 사상에 의해 정해져야 할 것이다.

부호의 설명

[0078] 1 : 클라우드 컴퓨팅 시스템

10 : 미터링 소프트웨어 유닛

20 : 신뢰성 구동 모듈

21 : TXT(Trusted Execution Technology) 유닛

30 : 신뢰 플랫폼 모듈(TPM)

31 : 플랫폼 구성 레지스터(PCR)

32 : 모노토닉 카운터

33: 실링(Sealing) 유닛

34 : 틱 스탬프 유닛

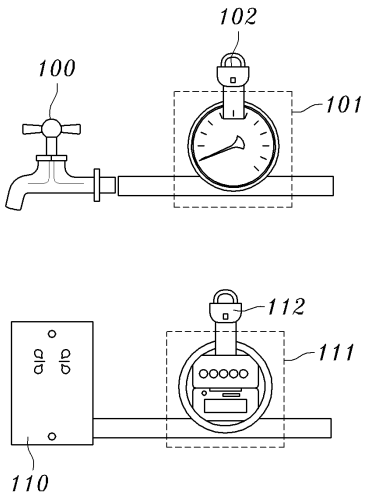
35 : Quote 연산 유닛

36 : 비휘발성 메모리 영역

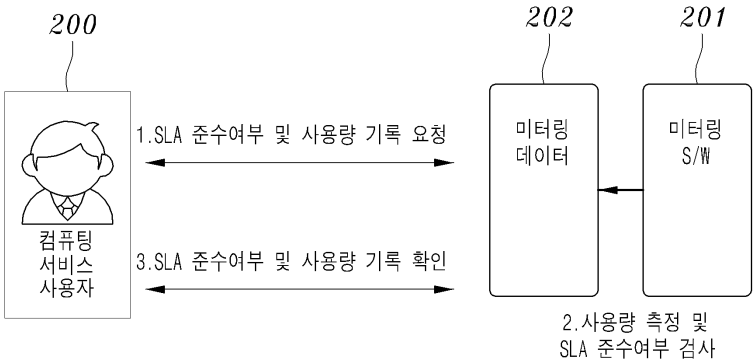
50 : 사용자 측 컴퓨팅 시스템

도면

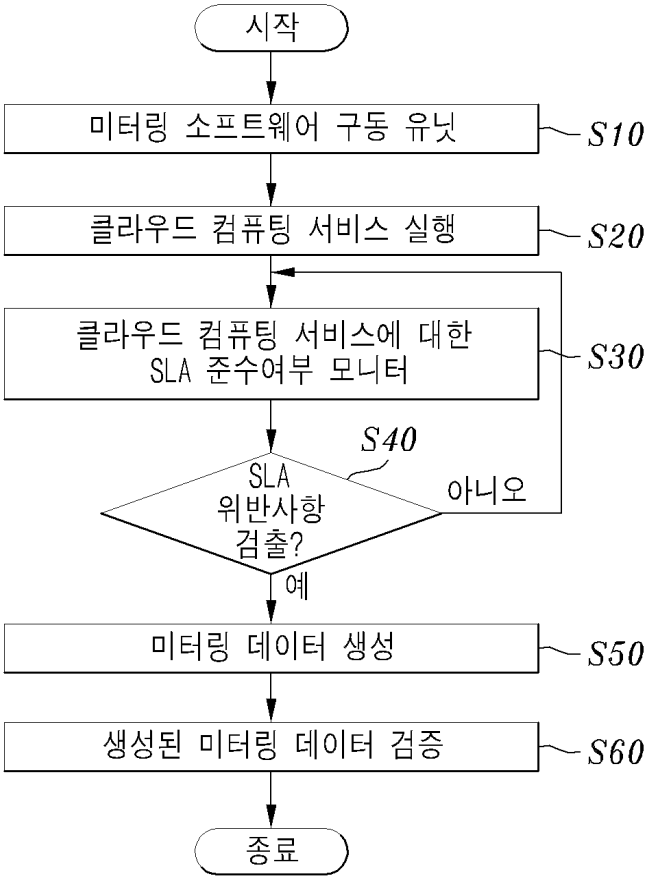
도면1



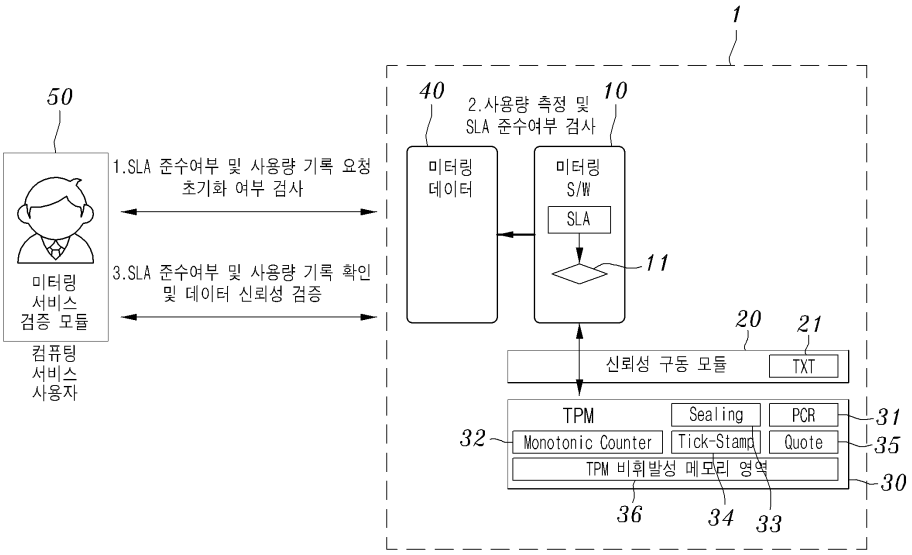
도면2


200
202
201
미터링
데이터
미터링
S/W
1.SLA 준수여부 및 사용량 기록 요청
3.SLA 준수여부 및 사용량 기록 확인
컴퓨팅
서비스
사용자
2.사용량 측정 및
SLA 준수여부 검사

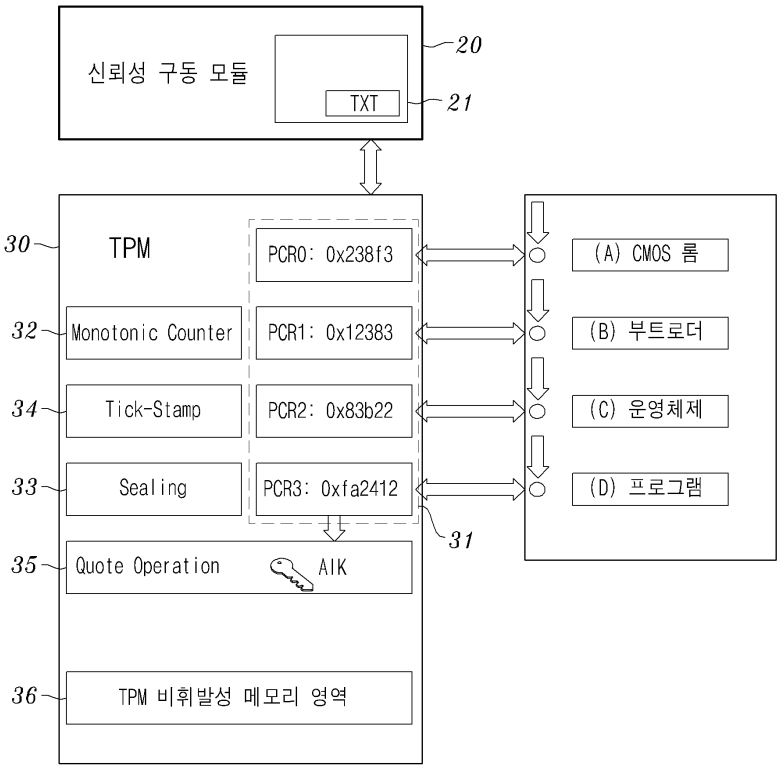
도면3


시작
미터링 소프트웨어 구동 유닛
S10
클라우드 컴퓨팅 서비스 실행
S20
클라우드 컴퓨팅 서비스에 대한
SLA 준수여부 모니터
S30
S40
SLA
위반사항
검출?
아니오
예
미터링 데이터 생성
S50
생성된 미터링 데이터 검증
S60
종료

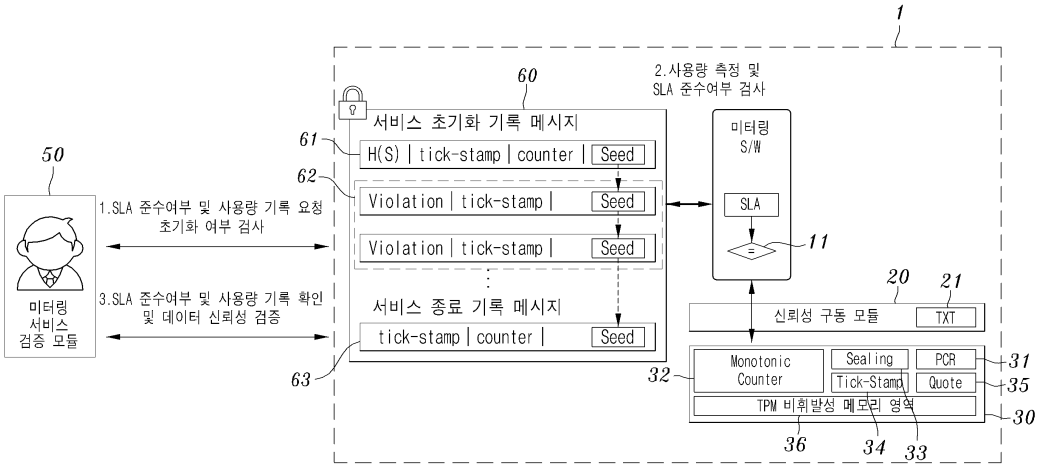
도면4


1
50
40
2.사용량 측정 및
SLA 준수여부 검사
10
1.SLA 준수여부 및 사용량 기록 요청
초기화 여부 검사
미터링
데이터
미터링
S/W
SLA
11
3.SLA 준수여부 및 사용량 기록 확인
및 데이터 신뢰성 검증
미터링
서비스
검증 모듈
컴퓨팅
서비스
사용자
20
21
신뢰성 구동 모듈
TXT
TPM
Sealing
PCR
31
32
Monotonic Counter
Tick-Stamp
Quote
35
TPM 비휘발성 메모리 영역
30
36
34
33

도면5


신뢰성 구동 모듈
20
TXT
21
30
TPM
PCR0: 0x238f3
(A) CMOS 롬
32
Monotonic Counter
PCR1: 0x12383
(B) 부트로더
34
Tick-Stamp
PCR2: 0x83b22
(C) 운영체제
33
Sealing
PCR3: 0xfa2412
(D) 프로그램
31
35
Quote Operation
AIK
36
TPM 비휘발성 메모리 영역

도면6



도면7

제1과정: S-Mon Initialization

1: 미터링 S/W←서비스 사용자: {S | N}
2: Extend(PCR18, H(S) | tick-stamp)
3: Key Generation (SK,PK)
4: Seed←N
5: NV_Write (PCR17, SK | SLA | counter | Seed)
6: BB←Seal(PCR17, H(S) | tick-stamp | counter | Seed)
7: Extend(PCR18, H(PK)), Extend(PCR18, 0xFF), Extend(PCR17, 0xFF)
8: q←Quote(PCR17-18, N)
9: 서비스 사용자←미터링S/W:{q | AIK | PK | tick-stamp}K

제2과정: SLA Monitoring

1: NV_Read(PCR17, SLA | counter | Seed)
2: If(counter==ReadCounter())
3: SLA_Monitoring (SLA, ReadStatus())
4: If(SLA Violation Exist)
5: Seed←H(Violation | tick-stamp | Seed)
6: BB←Seal(PCR17, Violation | tick-stamp | Seed)
7: NV_Write (PCR17, ++counter | Seed)

제3과정: SLA-Report

1: 미터링 S/W←서비스 사용자:{S | N}
2: NV_Read(PCR17, SK | counter | Seed)
3: BB←Seal(PCR17, tick-stamp | counter | Seed)
4: Extend(PCR17, 0xFF), Extend(PCR18, BB), Extend(PCR18, 0xFF)
5: BB←Unseal(PCR17, BB)
6: q←Quote(PCR17-18, BB | N)
7: 서비스 사용자←미터링S/W:{BB | {H(BB)}SK | q}k