(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2011-0124088
(43) 공개일자 2011년11월16일

(51) Int. Cl.
H04L 12/14 (2006.01) *H04L 9/14* (2006.01)
(21) 출원번호 10-2010-0043672
(22) 출원일자 2010년05월10일
심사청구일자 2010년05월10일

(71) 출원인
한국과학기술원
대전 유성구 구성동 373-1
(72) 발명자
박규호
대전광역시 유성구 구성동 한국과학기술원 6-3208
박기웅
서울특별시 노원구 월계4동 500-11 8/5
박성규
대전광역시 유성구 구성동 KAIST 동측기숙사 6118
(74) 대리인
김학제, 문혜정

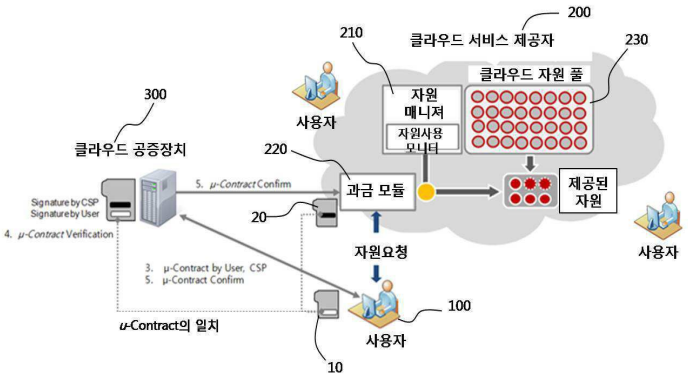
전체 청구항 수 : 총 20 항

(54) 클라우드 컴퓨팅 과금 공증장치, 과금 시스템 및 과금방법

(57) 요 약

본 발명은 클라우드 컴퓨팅 환경에서의 다수의 클라우드 서비스 사용자 단말장치들과 클라우드 서비스 제공자 장치 사이에 제3의 과금 공증장치를 두어 과금 처리를 검증하게 함으로써 과금처리의 정확성을 상호 검증 가능하게 한 클라우드 컴퓨팅 환경에서의 과금 공증장치, 과금 방법 및 장치에 관한 것이다.

대 표 도 - 도2



이 발명을 지원한 국가연구개발사업
과제고유번호 A1100-0901-1710
부처명 지식경제부
연구관리전문기관
연구사업명 독립형 컴포넌트 기반 서비스 지향형 페타급 컴퓨팅 플랫폼 기술개발
연구과제명 페타급 클라우드 컴퓨팅 플랫폼 개발
기여율
주관기관 한국클라우드컴퓨팅연구조합
연구기간 2009년 03월 ~ 2013년 02월

특허청구의 범위

청구항 1

네트워크를 통해서 연결된 클라우드 서비스 제공자 장치와 다수의 사용자 단말장치들 사이에서, 사용자가 사용한 컴퓨팅 자원에 대한 과금처리를 검증하는 과금 공증 장치로서,

상기 클라우드 서비스 제공자 장치 및 상기 다수의 사용자 단말장치들과 통신하는 통신부로서, 클라우드 서비스 제공자의 마이크로계약서 및 클라우드 서비스 사용자의 마이크로계약서를 수신하고, 과금 유효성 검증부에 의한 양자의 마이크로계약서의 유효성 검증 후, 클라우드 서비스 제공자 장치 및 클라우드 서비스 사용자 단말 장치에 과금 확인 메시지를 전송하는 통신부:

상기 사용자 단말장치로부터 수신한 사용자의 마이크로계약서 및 상기 클라우드 서비스 제공자의 마이크로계약서의 유효성을 검증하는 과금 유효성 검증부; 및

상기 마이크로계약서의 유효성 검증에 필요한 증거자료를 저장하고 사용자의 컴퓨팅 자원 사용 정보 및 과금 정보를 저장하는 과금 정보 데이터베이스를 포함하는 과금 공증 장치(billing notarizing apparatus).

청구항 2

제 1항에 있어서, 상기 클라우드 서비스 사용자의 마이크로계약서 및 클라우드 서비스 제공자의 마이크로계약서는 각각 해쉬 암호법에 따라 디지털 서명되고, 이러한 디지털 서명은 공유키로 암호화되어 상기 과금 공증장치로 전송되는 것을 특징으로 하는 과금 공증 장치.

청구항 3

제 1항에 있어서, 상기 과금 유효성 검증부는 클라우드 서비스 사용자의 마이크로계약서의 조항(stipulation)의 해쉬값과 클라우드 서비스 제공자의 마이크로계약서의 조항의 해쉬값 (H(S))을 비교하여 이들의 일치 여부로서 마이크로 계약서의 유효성을 검증하는 것을 특징으로 하는 과금 공증 장치.

청구항 4

제 1항에 있어서, 상기 과금 유효성 검증부는 단방향 해쉬 체인(oneway hash chain-based signature) 및 자원 사용 로그(resource usage log)에 기초해서 클라우드 서비스 제공자 및 클라우드 서비스 사용자를 기속할 수 있는 과금 정보를 생성하는 것을 특징으로 하는 과금 공증장치.

청구항 5

제 1항에 있어서, 상기 클라우드 서비스 제공자의 마이크로계약서는 자원 제공에 대한 계약 조항(S)에 대한 해쉬값, 그리고 해쉬 체인에서 아직까지 사용하지 않은 해쉬 요소(C_n) 및 랜덤 값(N)의 암호화된 값; 및 계약 조항(S)과 해쉬 요소를 두 개의 입력값으로 하는 이들의 해쉬값을 포함하는 것을 특징으로 하는 과금 공증 장치.

청구항 6

제 5항에 있어서, 상기 계약 조항(S)에는 타임 스탬프가 포함되는 것을 특징으로 하는 과금 공증장치.

청구항 7

제 1항에 있어서, 상기 과금 정보 데이터베이스는 클라우드 서비스 제공자의 마이크로계약서(*u*-Contract-CSP)와 사용자의 마이크로계약서 (*u*-Contract-User)가 동일한 해쉬값을 이용하여 작성되었다는 것을 나타내는 정보를 디지털 서명과 함께 공증 과금 리스트(notarized billing list, NBL)의 형태로 저장하는 것을 특징으로 하는 과금 공증 장치.

청구항 8

제 1항에 있어서, 상기 과금 공증 장치는 클라우드 서비스 제공자와 클라우드 서비스 사용자가 동일한 계약 조항(S)을 가질 경우, H(S)의 해쉬 값과 해쉬 요소 ($C_{n,n}$)를 포함하는 확인 메시지를 클라우드 서비스 제공자 및 클라우드 서비스 사용자에게 전송하는 것을 특징으로 하는 과금 공증 장치.

청구항 9

클라우드 컴퓨팅 환경에서 사용자가 사용한 컴퓨팅 자원에 대한 과금처리를 행하는 과금 시스템으로서, 상기 시스템이 클라우드 컴퓨팅 서비스를 제공하는 하나 이상의 서비스 제공자 장치; 상기 클라우드 서비스 제공자 장치의 컴퓨팅 자원을 사용하는 다수의 사용자 단말 장치들; 및 네트워크를 통해서 연결된 상기 클라우드 서비스 제공자 장치와 다수의 사용자 단말장치들 사이에서, 사용자가 사용한 컴퓨팅 자원에 대한 과금처리를 검증하는 제 1항 내지 제7항 중 어느 하나의 항의 과금 공증 장치를 포함하는 클라우드 컴퓨팅 과금 시스템.

청구항 10

클라우드 서비스 제공자 및 클라우드 과금 공증장치와 네트워크를 통하여 접속되고, 서비스 제공자의 컴퓨팅 자원의 제공에 대해 과금되는 사용자 단말 장치로서, 상기 장치는

사용자가 사용하고자 하는 컴퓨팅 자원에 대한 계약 조항을 기재한 마이크로계약서를 작성하는 마이크로계약서 작성부;

상기 마이크로계약서에 서명하기 위한 단방향 해쉬 체인 서명(oneway hash chain-based signature)을 생성하는 해쉬-체인 생성부; 및

상기 클라우드 서비스 제공자 장치 및 과금 공증장치와 통신하는 통신부를 포함하는 것을 특징으로 하는 클라우드 사용자 단말장치.

청구항 11

클라우드 컴퓨팅 환경에서 다수의 사용자에게 컴퓨팅 자원을 제공하고 그에 따른 대가를 지불받는 클라우드 서비스 제공자 장치로서,

제3의 과금 공증장치 및 클라우드 서비스 사용자 단말장치와 네트워크를 통하여 통신하는 통신부;

상기 통신부를 통해서 사용자의 자원 요청 메시지를 접수하여, 요청 받은 자원을 해당 사용자에게 제공하는 자원 매니저부(resource manager);

사용자에 대한 컴퓨팅 자원의 제공 이전에 자원 사용 로그를 생성하고, 마이크로계약서를 작성하는 클라우드 과금 모듈;

상기 마이크로계약서에 서명하기 위한 단방향 해쉬 체인 서명(oneway hash chain-based signature)을 생성 및 저장하는 수단을 포함하는 것을 특징으로 하는 클라우드 서비스 제공자 장치.

청구항 12

네트워크를 통해서 연결된 클라우드 서비스 제공자 장치와 다수의 사용자 단말장치들로 구성되는 클라우드 컴퓨

팅 환경에서 사용자가 사용한 컴퓨팅 자원에 대한 과금처리를 행하는 방법으로서, 상기 방법이

사용자의 최초 접속 시에 클라우드 서비스 제공자 장치 및 과금 공증 장치와의 인증과정을 통하여 인증키를 공유하는 단계;

클라우드 서비스 사용자 단말장치, 클라우드 서비스 제공자 장치, 및 클라우드 과금 공증장치에 각각 해쉬 체인을 생성하는 단계; 및

사용자의 컴퓨팅 자원 요청에 따라서 과금 처리를 행하는 단계로서, 상기 과금 처리 시에 과금 항목을 기재한 클라우드 서비스 제공자의 마이크로계약서 및 클라우드 서비스 사용자의 마이크로계약서의 해쉬값을 비교하여 과금 내용을 검증하는 단계를 포함하는 클라우드 컴퓨팅 환경에서의 과금방법.

청구항 13

제 12항에 있어서, 상기 인증 단계는 클라우드 서비스 사용자 장치, 클라우드 서비스 제공자 장치 및 과금 공증장치 사이의 인증 과정을 통해서, 사용자와 클라우드 서비스 제공자 사이의 공유키, 사용자와 클라우드 과금 공증장치 사이의 공유키, 및 클라우드 서비스 제공자와 클라우드 과금 공증장치 사이에 인증키를 공유하는 단계인 것을 특징으로 하는 클라우드 컴퓨팅 환경에서의 과금방법.

청구항 14

제 12항에 있어서, 상기 해쉬값 생성 단계는 클라우드 서비스 사용자, 클라우드 서비스 제공자, 및 클라우드 과금 공증장치로 하여금 각각 해쉬 체인을 형성하게 하는 단계;

클라우드 서비스 사용자로부터 전 단계에서 생성된 해쉬 체인의 최종값을 개인키를 이용하여 서명하게 한 후에, 클라우드 과금 공증장치에 등록받는 단계;

클라우드 서비스 제공자로부터 전 단계에서 생성된 해쉬 체인의 최종값 을 개인키를 이용하여 서명하게 한 후에 클라우드 과금 공증장치에 등록받는 단계; 및

상기 클라우드 과금 공증장치가 생성한 해쉬 체인의 최종값을 공증 장치의 개인키를 이용하여 서명한 후에 클라우드 서비스 사용자 단말장치 및 클라우드 서비스 제공자 장치에 전송하는 단계를 포함하는 것을 특징으로 하는 클라우드 컴퓨팅 환경에서의 과금방법.

청구항 15

제 12항에 있어서, 상기 과금 처리 단계는

클라우드 서비스 사용자로부터 컴퓨팅 자원 요청 메시지를 수신하는 단계;

상기 사용자의 컴퓨팅 자원 요청에 대응해서 서비스 제공자가 요청받은 컴퓨팅 자원에 대한 계약항목을 생성하여 마이크로계약서(u-Contract-CSP) 형태로 만들어 사용자에게 전송하는 단계;

상기 사용자가 서비스 제공자의 마이크로계약서의 내용을 확인한 후 동의를 할 경우 사용자의 마이크로계약서(u-Contract-User)를 작성하여, 서비스 제공자로부터 받은 마이크로계약서(u-Contract-CSP)와 사용자의 마이크로계약서(u-Contract-User)를 과금 공증장치로 전송하는 단계;

과금 공증장치가 쌍방의 마이크로계약서를 비교하여 유효성을 검증하는 단계; 및

전단계의 검증에서 이상이 없을 경우에 클라우드 서비스 사용자 및 클라우드 서비스 제공자에게 확인 메시지를 전송하는 단계를 포함하는 것을 특징으로 하는 클라우드 컴퓨팅 환경에서의 과금방법.

청구항 16

제 12항에 있어서, 상기 클라우드 서비스 제공자의 마이크로계약서 및 클라우드 서비스 사용자의 마이크로계약서 작성은 AES 및 해쉬 함수를 이용하여 작성되고, 여기서 상기 마이크로계약서는 해쉬 암호법에 따라

서 디지털 서명되고, 암호화된 디지털 서명이 상기 과금 공증장치로 전송되는 것을 특징으로 하는 클라우드 컴퓨팅 환경에서의 과금방법.

청구항 17

제 16항에 있어서, 상기 마이크로계약서에 대한 디지털 서명은 단방향 해쉬 체인(oneway hash chain-based signature)을 이용하여 수행되는 것을 특징으로 하는 클라우드 컴퓨팅 환경에서의 과금방법.

청구항 18

제 11항에 있어서, 상기 클라우드 서비스 제공자의 마이크로계약서는

서비스 제공자에 의해서 작성된 계약서 조항(S)에 대한 해쉬값(H(S)), 해쉬 체인에서 아직 사용되지 않은 해쉬 요소(C_n), 랜덤 값(N_c)을 서비스 제공자와 공증장치 사이에 공유한 키로 암호화한 부분과

계약서의 계약 조항(S)과 해쉬 요소를 두 개의 입력값으로 하는 이들의 해쉬값을 포함하고 있는 것을 특징으로 하는 클라우드 컴퓨팅 환경에서의 과금방법.

청구항 19

제 18항에 있어서, 상기 계약 조항(S)에는 타임 스탬프가 포함되는 것을 특징으로 하는 과금방법.

청구항 20

제 12항에 있어서, 상기 방법이 사용자가 추가적인 자원을 요청하는 경우 과금 처리 단계로 리턴하는 단계를 추가로 포함하는 클라우드 컴퓨팅 환경에서의 과금방법.

명 세 서

기 술 분 야

[0001] 본 발명은 클라우드 컴퓨팅 과금 공증장치, 과금 시스템 및 과금방법에 관한 것으로, 더욱 상세하게는 클라우드 서비스 제공자 장치 및 클라우드 서비스 사용자 단말 장치 이외의 클라우드 과금 공증장치를 도입하여 클라우드 컴퓨팅 환경에서의 과금 처리의 공정성 및 정확성을 향상시킨 클라우드 컴퓨팅 과금 공증장치, 과금 시스템 및 과금방법에 관한 것이다.

배 경 기 술

[0002] 네트워크의 고도화와 가상화와 같은 소프트웨어 기술의 발전에 따라서 광범위한 분야의 소프트웨어와 IT 자원들이 인터넷을 통해 제공될 수 있는 환경이 조성되었다. 따라서 인터넷을 기반으로 다수의 고객들에게 높은 수준의 확장성을 가진 컴퓨팅 자원들을 서비스로 제공하는 클라우드 컴퓨팅(cloud computing)이 등장하였고, 이미 차세대 IT 패러다임으로 자리를 굳히고 있다. 클라우드 컴퓨팅 환경에서 사용자들은 인터넷이 연결된 단말장치를 통해서 클라우드 서비스 제공자의 대용량의 컴퓨터 집합에 접속하여 애플리케이션, 스토리지, OS, 보안 등 개인용 컴퓨터 기반으로 사용했던 각종 IT 자원을 인터넷 기반으로 원하는 시점에 필요한 만큼 사용하게 되며, 대신 해당 자원의 사용량(연산량, 스토리지 용량, 통신량 등) 만큼 비용을 지불하게 된다. 즉, 사용자가 컴퓨팅 자원을 사용하는데 있어서, 필요한 만큼만 사용하고, 사용한 만큼만 지불하는 Pay-as-you-go 모델을 따른다.

[0003] 따라서 클라우드 컴퓨팅 서비스의 광범위한 이용과 확산을 위해서는, 클라우드 컴퓨팅 환경에서 사용자

가 사용한 자원의 사용량의 기록 및 과금을 사용자 및 서비스 제공자가 모두 안전하게 검증할 수 있는 방법이 제공되어야 한다.

[0004] 현재 상용 클라우드 컴퓨팅 서비스를 제공하는 업체 (Amazon의 EC2, Google의 Google App. Engine, Microsoft의 Azure)의 경우, 클라우드 서비스 제공자가 사용량 측정 및 과금을 수행한 후, 사용자에게 청구하는 방식으로 과금되고 있다. 이와 같이, 기존의 과금 시스템은 과금 연산 및 관리를 서비스 제공자가 단방향적으로 수행하도록 되어 있어, 과금 기록에 대한 상호 검증성을 제공해 주지 못하며, 사용한 자원에 대한 무결성을 지원해 주지 못하는 문제점이 있다.

[0005] 현재 상호 검증이 가능한 과금 시스템을 구축을 하기 위해서 공개키 기반 구조(Public Key Infrastructure, "PKI")가 사용되고 있다. 도 1은 이러한 클라우드 컴퓨팅 과금 시스템의 개략도이다. 도 1을 참고하면, 클라우드 서비스 제공자의 클라우드 자원 풀(cloud resource pool) 중에서 사용자에게 제공된 CPU 싸이클, 저장량, 네트워크 대역폭과 같은 사용자의 컴퓨팅 자원 사용 정보는 자원 사용 모니터(resource usage monitor)에 의해 수집되고, 과금 모듈(billing agent)을 통해 과금되며, 사용자와 서비스 제공자는 사용자가 사용할 컴퓨팅 자원, 즉 과금 항목 마다 사용자와 서비스 제공자의 PKI 기반의 디지털 서명 연산을 삽입하여 상호 검증이 가능하도록 한다.

[0006] 그러나 PKI의 디지털 서명 연산은 연산 복잡도가 매우 높은 연산이다. 클라우드 컴퓨팅과 같이 수많은 사용자가 컴퓨팅 자원을 사용하는 시스템에서, 매 번의 과금 연산 마다, 사용자, 서비스 제공자는 최소 한번 이상의 PKI 기반의 디지털 서명 연산을 수행하여야 하며, 이로 인해 연산에 대한 오버헤드 (computation overhead)가 많아 과금 시스템 전체의 병목현상을 유발할 수 있다. 더욱이, 다양한 컴퓨팅 자원을 필요로 하는 사용자의 수가 늘어날수록 과금 처리(billing transactions)도 급속도로 증가하여, 이러한 문제가 더욱 심화될 수 있다. 사용자의 경우, 연산량이 비교적 적은 단말기를 가지고 사용하는 클라우드 컴퓨팅 환경을 감안할 때, 연산량이 적으면서도 과금 처리에 대해 서비스 사용자 및 서비스 제공자 양방이 상호검증할 수 있는 과금방법 또는 장치의 개발이 요구되고 있다.

[0007]

발명의 내용

해결하려는 과제

[0008] 본 발명은 상기와 같은 기술적 요구에 부응하기 위한 것으로, 본 발명의 하나의 목적은 기존 방식에 비하여 획기적으로 적은 연산량으로, 컴퓨팅 자원의 사용에 대한 무결성과 과금에 대한 상호 검증성을 지원하는 클라우드 컴퓨팅 환경에서의 과금 공증장치를 제공하는 것이다.

[0009] 본 발명의 다른 목적은 클라우드 서비스 제공자 장치와 클라우드 서비스 사용자 단말장치 사이에 제 3의 클라우드 공증장치를 도입하여, 보안 연산을 경감시키면서도 상호검증이 가능하도록 한 클라우드 컴퓨팅 과금 시스템 및 방법을 제공하는 것이다.

과제의 해결 수단

[0010] 본 발명의 하나의 구현예는 네트워크를 통해서 연결된 클라우드 서비스 제공자 장치와 다수의 사용자 단말장치들 사이에서, 사용자가 사용한 컴퓨팅 자원에 대한 과금처리를 검증하는 클라우드 컴퓨팅 과금 공증 장치를 위치시키는 것, 이러한 과금 공증 장치는

[0011] 상기 클라우드 서비스 제공자 장치 및 상기 다수의 사용자 단말장치들과 통신하는 통신부로서, 클라우드 서비스 제공자의 마이크로계약서 및 클라우드 서비스 사용자의 마이크로계약서를 수신하고, 과금 유효성 검증부에 의한 양자의 마이크로계약서의 유효성 검증 후, 클라우드 서비스 제공자 장치 및 클라우드 서비스 사용자 단말장치에 과금 확인 메시지(billing confirmation message)를 전송하는 통신부:

[0012] 상기 사용자 단말장치로부터 수신한 사용자의 마이크로계약서 및 상기 클라우드 서비스 제공자의 마이크로계약서의 유효성을 검증하는 과금 유효성 검증부; 및

[0013] 상기 마이크로계약서의 유효성 검증에 필요한 증거자료를 저장하고 사용자의 컴퓨팅 자원 사용 정보(resource usage log) 및 과금 정보를 저장하는 과금 정보 데이터베이스를 포함하는 과금 공증 장치(billing

notarizing apparatus)에 관한 것이다.

[0014] 본 발명의 다른 구현예는 클라우드 서비스 제공자 장치 및 클라우드 과금 공증장치와 네트워크를 통하여 접속되고, 서비스 제공자의 컴퓨팅 자원의 제공에 대해 과금되는 사용자 단말 장치로서, 상기 장치는

[0015] 사용자가 사용하고자 하는 컴퓨팅 자원에 대한 계약 조항을 기재한 마이크로계약서를 작성하는 마이크로계약서 작성부;

[0016] 상기 마이크로계약서에 서명하기 위한 단방향 해쉬 체인 서명(oneway hash chain-based signature)을 생성하는 해쉬-체인 생성부; 및

[0017] 상기 클라우드 서비스 제공자 장치 및 상기 과금 공증장치와 통신하는 통신부를 포함하는 것을 특징으로 하는 클라우드 사용자 단말장치에 관한 것이다.

[0018] 본 발명의 또 다른 구현예는 클라우드 컴퓨팅 환경에서 다수의 사용자에게 컴퓨팅 자원을 제공하고 그에 따른 대가를 지불받는 클라우드 서비스 제공자 장치로서,

[0019] 과금 공증장치 및 클라우드 서비스 사용자 단말장치와 네트워크를 통하여 통신하는 통신부;

[0020] 상기 통신부를 통해서 사용자의 자원 요청 메시지를 접수하여, 요청 받은 자원을 해당 사용자에게 제공하는 자원 매니저부(resource manager);

[0021] 사용자에 대한 컴퓨팅 자원의 제공 이전에 자원 사용 로그를 생성하고, 마이크로계약서를 작성하는 클라우드 과금 모듈;

[0022] 상기 마이크로계약서에 서명하기 위한 단방향 해쉬 체인 서명(oneway hash chain-based signature)을 생성 및 저장하는 수단을 포함하는 것을 특징으로 하는 클라우드 서비스 제공자 장치에 관한 것이다.

[0023] 본 발명의 또 다른 구현예는 네트워크를 통해서 연결된 클라우드 서비스 제공자 장치와 다수의 사용자 단말장치들로 구성되는 클라우드 컴퓨팅 환경에서 사용자가 사용한 컴퓨팅 자원에 대한 과금처리를 행하는 방법으로서, 상기 방법이

[0024] 사용자의 최초 접속 시에 클라우드 서비스 제공자 장치 및 과금 공증 장치와의 인증과정을 통하여 인증키를 공유하는 단계;

[0025] 클라우드 서비스 사용자 단말장치, 클라우드 서비스 제공자 장치, 및 클라우드 과금 공증장치에 각각 해쉬 체인을 생성하는 단계; 및

[0026] 사용자의 컴퓨팅 자원 요청에 따라서 과금 처리를 행하는 단계로서, 상기 과금 처리 시에 과금 항목을 기재한 클라우드 서비스 제공자의 마이크로계약서 및 클라우드 서비스 사용자의 마이크로계약서의 해쉬값을 비교하여 과금 내용을 검증하는 단계를 포함하는 클라우드 컴퓨팅 환경에서의 과금방법에 관한 것이다.

발명의 효과

[0027] 본 발명에 따른 과금 공증장치(billing notarizing apparatus) 기반의 과금 방법 및 장치에 의하면, 서비스 제공자 및 사용자가 과금 내용을 상호 검증할 수 있고, 기존 방식 대비 매우 적은 연산량으로 과금처리가 가능하고 과금 연산시 발생되는 시간도 획기적으로 단축되어 매우 효율적인 과금 연산이 가능하다.

[0028] 또한 본 발명에서는 클라우드 서비스 제공자 장치 및 클라우드 서비스 사용자 단말 장치 이외의 신뢰되는 제3의 과금 엔터티, 즉 클라우드 과금 공증장치가 과금 처리의 공정성 및 정확성을 검증하므로, 사용자 또는 공증장치, 또는 서비스 제공자가 과금 내용을 위조 또는 변조하더라도 이러한 사실을 검출가능하여 과금 처리에 대한 신뢰성을 제고할 수 있다.

도면의 간단한 설명

[0029] 도 1은 기존의 클라우드 컴퓨팅 과금 시스템의 개략도이다.

도 2는 본 발명의 일구현예에 의한 클라우드 컴퓨팅 과금 시스템의 개략도이다.

도 3은 본 발명의 일구현예에 의한 클라우드 과금 공증장치의 블록도이다. 도 4는 본 발명의 일구현예의 과금

에 대한 상호검증 기능을 갖는 사용자 단말장치의 블록도이다.

도 5는 본 발명의 일구현예에 의한 클라우드 컴퓨팅 과금 방법의 흐름도이다.

도 6은 본원에서 사용되는 과금 엔터티 및 메시지 심볼을 설명한 도면이다.

도 7은 본 발명의 일구현예에 의한 클라우드 컴퓨팅 과금 방법의 시그널링 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0030] 이하에서 첨부 도면을 참조하여 본 발명의 다양한 구현예에 대하여 상세하게 설명한다. 본 발명의 목적, 작용, 효과를 포함하여 기타 다른 목적들, 이점들 및 동작상의 이점들은 이하의 상세한 설명에 의해 더욱 자명해질 것이다.

[0031] 본 발명에서 “클라우드 컴퓨팅(cloud computing)”이라는 용어는

[0032] 인터넷 기술을 활용하여 다수의 고객들에게 높은 수준의 확장성을 가진 컴퓨팅 자원들을 전기, 가스 등과 같은 유틸리티 서비스 처럼 제공하는 서비스로서, 퍼블릭 클라우드, 프라이빗 클라우드, 모바일 클라우드 등 상기와 같은 서비스를 제공하는 모든 형태의 서비스를 의미한다.

[0033] 본 발명에서 "클라우드 서비스 제공자(cloud service provider, CSP)“라는 용어는 네트워크를 통해서 다수의 사용자들에게 대용량의 IT 자원을 서비스로 제공하는 서비스 제공자를 의미한다.

[0034] 본 발명에서 “사용자 단말장치(user equipment)”라는 용어는 "클라우드 컴퓨팅 서비스를 이용하는 사용자의 데스크탑 컴퓨터, 노트북 컴퓨터, 벽걸이 컴퓨터, 이동통신 단말기, PDA, 기타 휴대용 컴퓨팅 기기 등 클라우드 컴퓨팅 서비스를 이용할 수 있는 현재 및 장래의 모든 컴퓨팅 장치를 의미하다.

[0035] 본 발명에서 “마이크로계약서(u-Contract)”란 클라우드 서비스 제공자 및 클라우드 서비스 사용자에 의해 작성되는 것으로, 계약서 내에는 컴퓨팅 자원에 대한 계약 조항(컴퓨팅 자원의 종류, 사용시간, 가격 등의 과금 플랜 정보)에 대한 정보가 포함된다.

[0036] 도 2는 본 발명의 일구현예에 의한 클라우드 컴퓨팅 과금 시스템의 개략도이고, 도 3은 본 발명의 일구현예에 의한 클라우드 과금 공증장치의 블록도이며, 도 4는 본 발명의 일구현예의 과금에 대한 상호검증 기능을 갖는 사용자 단말장치의 블록도이다.

[0037] 도 2 및 도 3을 참고하면, 본 발명의 하나의 구현예는 네트워크를 통해서 연결된 클라우드 서비스 제공자 장치(200)와 다수의 사용자 단말장치들(100)로 구성되는 클라우드 컴퓨팅 환경에서, 사용자가 사용한 컴퓨팅 자원에 대한 과금처리를 검증하는 과금 공증 장치(300)에 관한 것이다. 이러한 구현예에서는 매 자원 요청시마다 클라우드 서비스 제공자 및 사용자 사이에 계약서를 작성하여 보관하고 컴퓨팅 자원을 할당받아 사용하고, 특정 간격 (주간, 월간)으로 사용대금을 일시에 결재하는 후불제 모델을 예로 들어 설명한다.

[0038] 본 발명의 하나의 구현예의 과금 공증 장치는 상기 클라우드 서비스 제공자 장치(200) 및 상기 다수의 사용자 단말장치들(100)과 통신하는 통신부(310)로서, 클라우드 서비스 제공자의 마이크로계약서(20) 및 클라우드 서비스 사용자의 클라우드 서비스 사용자 단말장치로부터 마이크로계약서(10)를 수신하고, 과금 유효성 검증부에 의한 양자의 마이크로계약서(10, 20)의 유효성 검증 후, 클라우드 서비스 제공자 장치 및 클라우드 서비스 사용자 단말장치에 과금 확인 메시지를 전송하는 통신부:

[0039] 상기 사용자 단말장치(100)로부터 수신한 사용자의 마이크로계약서(10) 및 상기 클라우드 서비스 제공자의 마이크로계약서(20)의 유효성을 검증하는 과금 유효성 검증부(330); 및

[0040] 상기 마이크로계약서의 유효성 검증에 필요한 해쉬값을 저장하고 사용자의 컴퓨팅 자원 사용 정보(resource usage log) 및 과금 정보를 저장하는 과금 정보 데이터베이스(320)를 포함한다.

[0041] 상기 과금 유효성 검증부(330)는 단방향 해쉬 체인(oneway hash chain-based signature) 및 자원 사용 로그(resource usage log)에 기초해서 클라우드 서비스 제공자 및 클라우드 서비스 사용자를 기속할 수 있는 과금 정보를 생성한다. 사용자의 컴퓨팅 자원의 사용을 모니터링 하기 위해서 정확한 컴퓨팅 자원 사용 로그(resource usage log)가 요구되는데, 이러한 자원 사용 로그는 클라우드 서비스 제공자의 서비스 제공에 대한 로그(log) 데이터의 해쉬 값을 과금 과금공증장치로 제공함으로써 추후에 서비스 제공에 대한 로그에 대한 무결성을 검증할 수 있다.

[0042] 클라우드 서비스 사용자의 마이크로계약서(10)와 클라우드 서비스 제공자의 마이크로계약서(20)는 각각 해쉬 암호법에 따라 디지털 서명되고, 이러한 디지털 서명은 공유키로 암호화되어 과금 공증장치(300)로 전송된다.

[0043] 해쉬 함수(hash function)란 임의의 크기의 입력값을 받아서 고정된 크기의 출력값을 내어주는 함수이다. 암호학적 해쉬 함수(cryptographic hash function)란 해쉬 함수 중, 약한 충돌 회피성(weak collision freeness)과 단방향 성(one-wayness)을 만족하는 것을 가리킨다. 해쉬 함수는 약한 충돌 회피성 때문에, 특정 값 a에 대해서 해쉬 값 (h(a))이 주어졌을 때, h(b)=h(a)를 만족하는 a와 다른 b를 찾기 어렵고, 또한 단방향 성질(onewayness) 때문에 해쉬값(h(a)) 을 알 때, a값을 알기 어렵다. 이와 같이 입력값의 미세한 변화에도 해쉬함수의 출력값은 완전히 달라지기 때문에, 정보의 무결성을 판단하는데 효과적으로 사용될 수 있다.

[0044] 해쉬 체인의 기본 구조는 L. Lamport(L. Lamport, Password Authentication with Insecure Communication, CACM, Vol. 24, No. 11. pp. 770-772, 1981)에 의해 제안되었다. 암호학적 해쉬 함수 h 를 사용한 해쉬 체인의 구조는 다음과 같다.

[0045] $Chain\ (n, Seed) = \{C_0, C_1, C_2\ .....\ C_n\}$

[0046] $C_i = h(C_{i+1}) = h^{n-i}(Seed)$

[0047] 본 발명에서 사용되는 해쉬 체인은 해쉬값의 해쉬, 그 해쉬 값의 해쉬 등으로 꼬리에 꼬리를 무는 해쉬 연산을 통하여 체인을 형성하는 것을 특징으로 한다. 해쉬 체인의 생성시 Seed값은 C_n에서 시작하여 체인에서 가장 마지막은 C_0가 되며, 등록시에 C_0를 등록하고 C_1, C_2, C_3 등의 순서로 사용하게 된다(도 6 참조).

[0048] 클라우드 컴퓨팅 환경에서의 매번의 과금 연산시 마다 각각의 과금 엔터티(사용자, 제공자, 공증장치)의 해쉬 체인에서 하나의 해쉬 요소(hash element)를 순차적으로 꺼내와 과금 연산을 실행하게 된다.

[0049] 상기 과금 유효성 검증부(330)는 클라우드 서비스 사용자로부터 수신된 마이크로계약서(10)의 조항(stipulation)의 해쉬값(H(S))과 클라우드 서비스 제공자로부터 수신된 마이크로계약서(20)의 조항의 해쉬값(H(S))을 비교하여 이들의 일치 여부로서 마이크로 계약서의 유효성을 검증한다. 상기 마이크로계약서의 해쉬값들이 일치할 경우 중간에 마이크로계약서의 내용이 위조 또는 변조되지 아니하여 계약서 상의 과금내용에 문제가 없음을 의미한다.

[0050] 상기 클라우드 서비스 제공자의 마이크로계약서(20)는 자원 제공에 대한 계약 조항(S)에 대한 해쉬값, 그리고 해쉬 체인에서 아직까지 사용하지 않은 해쉬 요소(C_n), 및 랜덤 값 N의 암호화된 값과 계약 조항(S)과 해쉬 요소($C_{c,n}$)를 두 개의 입력값으로 하는 해쉬값을 포함한다. 상기 계약 조항(S)에는 재시도 공격을 방지하기 위하여 타임 스탬프가 포함된다. 해쉬 요소(hash element)는 매번의 과금 연산 시 마다 chain-by-chain으로 업데이트되므로, 모든 해쉬 요소들은 서로 연결되어 해쉬-체인의 시드 값(seed value)($C_{c,0}$)에 대해서 순차적으로 검증될 수 있다.

[0051] 상기 과금 정보 데이터베이스(320)는 클라우드 서비스 제공자의 마이크로계약서(*u*-Contract-CSP)(20)와 사용자의 마이크로계약서 (*u*-Contract-User)(10)가 동일한 해쉬값을 이용하여 작성되었다는 것을 나타내는 정보를 디지털 서명과 함께 공증 과금 리스트(notarized billing list, NBL)의 형태로 저장한다.

[0052] 상기 과금 공증 장치(300)는 클라우드 서비스 제공자와 클라우드 서비스 사용자가 동일한 계약 조항(S)을 가질 경우, H(S)의 해쉬 값과 해쉬 요소 ($C_{n,n}$)를 포함하는 확인 메시지를 클라우드 서비스 제공자 장치(200) 및 클라우드 서비스 사용자 단말장치(100)로 전송한다. 이로써 클라우드 서비스 사용자 및 클라우드 서비스 제공자는 마이크로계약서의 내용이 중간 과정에서 아무런 변화가 없었다는 것을 확인할 수 있다.

[0053] 본 발명의 다른 구현예는 클라우드 컴퓨팅 환경에서 사용자가 사용한 컴퓨팅 자원에 대한 과금처리를 행하는 과금 시스템에 관계한다. 도 2를 참조하면, 이러한 과금시스템은 클라우드 컴퓨팅 서비스를 제공하는 하나 이상의 서비스 제공자 장치(200); 상기 클라우드 서비스 제공자 장치의 컴퓨팅 자원을 사용하는 다수의 사용자 단말 장치들(200); 및 네트워크를 통해서 연결된 상기 클라우드 서비스 제공자 장치와 다수의 사용자 단말 장치들 사이에서, 사용자가 사용한 컴퓨팅 자원에 대한 과금처리를 검증하는 과금 공증 장치(300)를 포함한다. 상기 과금 시스템의 상기 과금 공증장치는 이상에서 설명한 바와 같은 구성을 갖는다.

[0054] 본 발명의 또 다른 구현예는 클라우드 서비스 제공자 및 클라우드 과금 공증장치와 네트워크를 통하여 접속되고, 서비스 제공자의 컴퓨팅 자원의 제공에 대해 과금되는 클라우드 서비스 사용자 단말 장치에 관한 것이다. 사용자 단말 장치는 사용자가 사용하고자 하는 컴퓨팅 자원에 대한 계약 조항을 기재한 마이크로계약서를 작성하는 마이크로계약서 작성부(120); 상기 마이크로계약서에 서명하기 위한 단방향 해쉬 체인 서명(oneway hash chain-based signature)을 생성하는 해쉬-체인 생성부(130); 및 상기 클라우드 서비스 제공자 장치 및 과금 공증장치와 통신하는 통신부(110)를 포함한다. 상기 클라우드 서비스 사용자 단말 장치(100)는 클라우드 서비스 제공자 장치에 사용하고자 하는 자원을 요청하는데, 이 때 자원 요청 메시지를 클라우드 서비스 제공자 장치와 공유한 키($K_{u,c}$)로 암호화하여 전송한다. 클라우드 서비스 제공자 장치로부터 해쉬 암호법에 따라서 디지털 서명되고, 디지털 서명이 암호화된 마이크로계약서를 수신한 후에, 마이크로계약서 상의 조항(stipulation)을 확인하고 자신이 요청과 부합되는 경우에, 해쉬 암호법에 따라서 디지털 서명된 사용자의 마이크로 계약서(u-Contract-User)를 작성하여 사용자와 과금 공증장치 사이에 공유된 키($K_{u,n}$)를 이용하여 암호화하여 과금 공증장치로 전송한다. 과금 공증 장치에 의한 검증 후, 과금장치로부터 확인 메시지를 수신한다.

[0055] 본 발명의 또 다른 구현예는 클라우드 컴퓨팅 환경에서 다수의 사용자에게 컴퓨팅 자원을 제공하고 그에 따른 대가를 지불받는 클라우드 서비스 제공자 장치(200)에 관한 것이다. 클라우드 서비스 제공자 장치는 과금 공증장치(300) 및 클라우드 서비스 사용자 단말장치(100)와 네트워크를 통하여 통신하는 통신부(미도시); 상기 통신부를 통해서 사용자의 자원 요청 메시지를 접수하여, 요청 받은 자원을 해당 사용자에게 제공하는 자원 매니저부(resource manager)(210); 사용자에 대한 컴퓨팅 자원의 제공 이전에 자원 사용 로그를 생성하고, 마이크로계약서를 작성하는 클라우드 과금 모듈(220); 및 상기 마이크로계약서에 서명하기 위한 단방향 해쉬 체인 서명(oneway hash chain-based signature)을 생성 및 저장하는 수단을 포함한다. 클라우드 서비스 제공자 장치(200)는 사용자의 자원 요청에 따라서 해쉬 암호법에 따라서 디지털 서명된 클라우드 서비스제공자의 마이크로계약서 (u-Contract-CSP)를 사용자 단말장치로 전송하고, 과금 공증장치에 의한 유효성 검증 후 확인 메시지를 수신한다.

[0056] 클라우드 서비스 사용자 단말장치(100), 클라우드 서비스 제공자 장치(200) 및 과금 공증장치(300)의 동작에 대해서는 이하의 과금 방법과 관련하여 설명한다.

[0057] 본 발명의 또 다른 구현예는 이상에서 설명한 클라우드 컴퓨팅 과금 시스템에서의 과금 방법에 관한 것이다. 도 5는 본 발명의 하나의 구현예에 의한 클라우드 컴퓨팅 환경에서의 과금 방법의 흐름도이고, 도 6은 본원에서 설명에 사용되는 과금 엔터티 및 메시지 심볼을 설명한 표이다.

[0058] 도 5를 참조하면, C는 클라우드 서비스 제공자를 의미하며, u는 사용자, n은 클라우드 과금 공증장치를 의미한다. $K_{\alpha,\beta}$는 두 과금 엔터티 사이의 공유키를 의미하며, PK, SK는 비대칭키 기반 연산에서 각각 공개키(Public Key)와 개인키(Private Key)를 의미한다. H(M)은 해쉬 연산을 의미하며, Ts는 재시도 공격(Replay Attack)을 방지하기 위한 타임스탬프를 의미한다. N은 랜덤 값을 의미하는데, 랜덤 값은 무작위로 생성한 더미 데이터로서, 중간자 공격(Man in the middle Attack) 또는 재시도 공격을 방지하기 위하여 사용된다. {M}은 K, PK, SK 모두 상기 언급했던 키를 이용하여 암호화 연산을 취하는 것을 의미한다. SK(개인키)를 이용하여 암호화를 하는 경우 PKI-기반 디지털 서명이라고 한다.

[0059] 본 발명에 의한 과금 처리는 3 단계로 진행된다. 먼저 사용자의 최초 접속 시에 클라우드 서비스 제공자 장치 및 과금 공증 장치와의 인증과정을 통하여 인증키를 공유하고(State 1), 이어서 클라우드 서비스 사용자 단말장치, 클라우드 서비스 제공자 장치, 및 클라우드 과금 공증장치에 각각 해쉬 체인을 생성한다 (State 2). 이어서 사용자의 컴퓨팅 자원 요청에 따라서 과금 처리를 행하는 단계로서, 상기 과금 처리 시에 과금 항목을 기재한 클라우드 서비스 제공자의 마이크로계약서 및 클라우드 서비스 사용자의 마이크로계약서의 해쉬값을 비교하여 과금 내용을 검증한다(State 3).

[0060] 이하에서 각 단계에 대해서 더욱 상세하게 설명한다.

[0061] 클라우드 컴퓨팅 서비스를 이용하고자 하는 사용자는 최초 접속 시에 클라우드 서비스 제공자 및 과금 공증 기관과의 인증과정을 통하여 인증키를 공유한다(S10). 이러한 사용자, 서비스 제공자 및 과금 공증 기관 사이의 인증 과정을 통해서, 각각의 과금 엔터티는 아래와 같은 세 개의 인증키를 공유하게 된다.

[0062] -사용자 ↔ 클라우드 서비스 제공자: $K_{u,c}$

[0063] -사용자↔ 클라우드 과금 공증장치: $K_{u,n}$

[0064] -클라우드 서비스 제공자 ↔ 클라우드 과금 공증장치: $K_{c,n}$

[0065] 이미 인증이 완료된 상태라면 바로 해쉬-체인 생성 단계로 진행할 수 있다. 해쉬-체인 생성 단계(S20)에서는, 클라우드 서비스 사용자, 클라우드 서비스 제공자, 클라우드 과금 공증장치는 랜덤 값($C_{u,N}$, $C_{c,N}$)에 해쉬 함수를 N번 적용하여 길이 N의 해쉬-체인을 생성함으로써 최종값($C_{u,0}$, $C_{c,0}$)을 수득한다. 클라우드 서비스 사용자와 클라우드 서비스 제공자는 생성된 해쉬-체인의 최종값 ($C_{u,0}$, $C_{c,0}$)을 각자의 개인키($K_{u,n}$, $K_{c,n}$) 를 이용하여 서명한 후에, 클라우드 과금 공증장치로 전송하여 등록한다. 클라우드 과금 공증장치도 생성한 해쉬-체인의 최종값($C_{n,0}$)을 공증 장치의 개인키를 이용하여 서명한 후에 클라우드 서비스 사용자 단말장치 및 클라우드 서비스 제공자 장치에 전송한다. 도 6에 도시한 바와 같이, 해쉬 체인의 Seed값은 C_n에서 시작하여 해쉬-체인의 최종값은 C_0가 되며, 등록 시에는 C_0를 등록하고 C_1, C_2, C_3 등의 순서 대로 사용하게 된다.

[0066] 매번의 과금 연산시 마다 각 과금 엔터티의 해쉬 체인에서 하나의 해쉬 요소(hash element)를 순차적으로 꺼내와 과금 연산을 실행하게 된다. 그러므로 일단 단방향 해쉬 체인의 약정(commitment)이 완료되거나, 이후의 과금 과정에서 해쉬 체인에 있는 모든 해쉬 요소를 소모하였거나, 사용자 또는 공증장치, 또는, 서비스 제공자가 해쉬 체인을 취소한 경우를 제외하고는 모든 과금 연산은 바로 단계 3(state 3)에서 시작하게 된다.

[0067] 인증 및 해쉬-체인 생성 후에는 실제 과금 처리 및 검증이 진행된다. 단계 3(State 3)(S30)에서, 사용자는 클라우드 컴퓨팅을 이용하고자 하는 경우 클라우드 서비스 제공자에게 자신이 필요한 컴퓨팅 자원을 요청하는 메시지를 전송한다(S31). 사용자로부터 컴퓨팅 자원 요청 메시지를 수신한 서비스 제공자는 요청받은 자원에 대한 계약 조항(시간, 가격, 등)을 생성하여 마이크로계약서(u-Contract-CSP) 형태로 만들어 사용자에게 전송한다(S32). 이를 수신한 사용자는 클라우드 서비스제공자의 마이크로계약서(u-Contract-CSP)의 내용을 확인한 후 동의할 경우 사용자의 마이크로계약서(u-Contract-User)를 작성하여, 서비스 제공자로부터 받은 마이크로계약서(u-Contract-CSP)와 사용자의 마이크로계약서(u-Contract-User)를 공증장치로 전송한다(S33). 이를 수신한 공증장치는 쌍방의 마이크로계약서를 비교하여 검증한다(S34). 이상이 없을 시에는 사용자 및 서비스 제공자에게 확인 메시지를 전송한다(S35). 과금이 검증되면 클라우드 서비스 제공자는 사용자에게 사용자가 요청한 컴퓨팅 자원을 제공한다(S36).

[0068] 이와 같이 하나의 과금 연산이 완료된 후에 추가적인 과금이 필요한 경우(S37)에는 다시 단계 3(S30)의 첫 단계(S31)로 리턴한다. 이 때, 해쉬 체인의 모든 해쉬 요소(hash element)를 소모했거나, 사용자 또는 공증장치, 또는, 서비스 제공자가 해쉬 체인을 취소했는지 여부를 판단하여(S38), 그와 같은 경우에는 단계 2(S20)로 돌아가 해쉬 체인을 다시 생성하고, 해쉬 요소가 남아 있는 경우에는 단계 S31로 진행하여 과금처리를 계속한다.

[0069] 도 7은 본 발명의 일구현예의 과금 방법의 실시 중에 진행되는 각각의 과금 엔터티 사이의 시그널링 흐름을 도시한 것이다. 특히 도 7은 과금 처리 이전의 인증 및 해쉬 체인 생성 과정 및 과금 처리 중의 클라우드 서비스 제공자 장치, 클라우드 서비스 사용자 단말장치 및 클라우드 과금 공증장치 사이의 시그널링을 보여준다. 도 7을 참조하면, 단계 1(S10) 및 단계 2(S20)는 클라우드 컴퓨팅 사용자, 클라우드 서비스 제공자 및 클라우드 과금 공증 기간(장치) 사이의 상호 인증 및 키분배, 그리고 해쉬 체인 생성을 위한 것이다. 본 발명의 클라우드 컴퓨팅 과금 방법은 인증 단계 (authentication (State 1)) 및 해쉬 체인 등록 및 갱신 (hash chain register/renewal (State 2)) 단계 이후에 상호 검증가능한 과금방법을 제공한다.

[0070] 단계 1에서, 클라우드 서비스 제공자 장치는 해쉬 체인을 형성하고, 해쉬 체인의 가장 마지막 값인 C_0를 자신의 개인키로 서명한 후에 이를 공증장치로 전송하여 등록하게 되며, 이와 동시에 앞으로 서비스 제공자 장치 및 사용자 단말장치 사이의 통신 시에 사용할 대칭키를 공유하게 된다((Msg1)(11). 이를 수신한 클라우드 과금 공증장치 역시 서비스 제공자 장치의 공개키를 이용하여, 서비스 제공자 장치가 등록하려는 해쉬 체인의 유효성을 검사하며, 과금 공증장치가 생성한 해쉬 체인의 가장 마지막 값인 C_0를 자신의 개인키로 서명 후에 서비스 제공자에게 전송한다(Msg2)(12). 사용자와 과금 공증장치 사이에 있어서도, 이와 동일한 방식으로 사용자가 인증 및 해쉬 체인 등록을 요청하고(Msg3)(15), 이러한 요청을 접수한 과금 공증장치도 사용자 단말장치가 등록하려는 해쉬 체인의 유효성을 검사하며, 과금 공증장치가 생성한 해쉬 체인의 최종값인 C_0를 자신의 개인키로 서명한 후에 사용자에게 전송한다(Msg4)(17).

[0071] 실제 과금 연산을 수행하는 단계 3(S30)에서, 클라우드 컴퓨팅 서비스 사용자가 클라우드 서비스 제공자의 클라우드 자원을 이용하고자 하는 경우에, 먼저 사용자는 자신이 필요한 클라우드 컴퓨팅 자원(CPU, 메모리, 네트워크)등을 자원 요청(Resource-Request)에 담아 이를 랜덤 값 N과 함께 클라우드 서비스 제공자 장치와 공유한 키($K_{u,c}$)로 암호화한 자원 요청 메시지(Msg1)(31)를 전송한다. 사용자의 자원 요청 메시지(Msg1)를 수신한 클라우드 서비스 제공자는 해독하여 사용자가 필요로 하는 컴퓨팅 자원을 파악한 후 그에 맞는 마이크로계약서 (u-Contract-CSP)를 작성한다(32).

[0072] 상기 클라우드 서비스 제공자의 마이크로계약서 및 클라우드 서비스 사용자의 마이크로계약서 작성은 AES 및 해쉬 함수를 이용하여 작성되고, 여기서 상기 마이크로계약서는 해쉬 암호법에 따라서 디지털 서명되고, 암호화된 디지털 서명이 상기 과금 공증장치로 전송된다. 상기 마이크로계약서에 대한 디지털 서명은 단방향 해쉬 체인(oneway hash chain-based signature)을 이용하여 수행된다.

[0073] 클라우드 서비스 제공자의 마이크로계약서(u-Contract-CSP)는 크게 두 부분으로 구성된다. 하나의 부분은 서비스 제공자에 의해서 작성된 부분으로, 계약서 조항(S)에 대한 해쉬값(H(S)), 해쉬 체인에서 아직 사용되지 않은 해쉬 요소(C_n), 랜덤 값(N_c)을 서비스 제공자와 공증장치 사이에 공유한 키($K_{c,n}$)로 암호화한 부분이고, 나머지 하나의 부분은 계약서의 계약 조항(S)과 Cn값의 해쉬함수에 대입해서 구한 해쉬값이다. 클라우드 서비스 제공자는 이렇게 작성된 u-Contract-CSP와 실제 계약서 내용이 담긴 조항(S) 그리고, 사용자로부터 받았던 랜덤값 N을 이용하여 Msg2를 작성하여 사용자에게 전송한다(32).

[0074] 클라우드 서비스 제공자로부터 μ-contract-*CSP* (*Msg2*)를 수신한 사용자는 클라우드 서비스 제공자(CSP)와 클라우드 과금 공증장치 사이에 공유된 키 ($K_{c,n}$)를 모르고, Cn 값을 모르기 때문에 u-Contract-CSP 내부의 내용을 검증할 수 없다. 다만, Msg2 내부에는 S 값이 있기 때문에 사용자가 요청했던 부분과 S의 내용이 일치한다면, 사용자도 CSP로부터 받은 S를 이용하여 u-Contract-User를 만들고 이를 사용자와 공증장치 사이에 공유된 키($K_{u,n}$)를 이용하여 암호화하여 이를 클라우드 과금 공증장치로 전송한다 (Msg3)(33). 즉, 사용자는 과금 내용 상호 검증 및 무결성 확인을 위해서 *μ-contract-CSP*와 *μ-contract-User*를 결합하여 계약 검증 요청 메시지(*Msg3*)를 클라우드 과금 공증장치에 전송한다.

[0075] 클라우드 서비스 사용자로부터 계약 검증 요청 메시지(*Msg3*)를 수신한 클라우드 과금 공증장치는 사용자와 공유한 키($K_{u,n}$), CSP와 공유한 키($K_{c,n}$)가 모두 있기 때문에 u-Contract의 내부 내용을 모두 해독을 할 수 있다. u-Contract-CSP를 만드는데 사용이 되었던 H(S)값과, u-Contract-User를 만드는데 사용이 되었던 H(S) 값이 서로 일치하는지를 비교함으로써, 사용자와 서비스 제공자가 같은 계약 조항(stipulation)에 대해서 서명했다는 일치성을 검증할 수 있다. 또한 u-Contract-CSP를 만드는데 사용이 되었던 해쉬 체인의 C 값과, u-Contract-User를 만드는데 사용이 되었던 해쉬 체인의 C 값이 각각 사용자와 서비스 제공자의 해쉬 체인에 맞는 값인지를 검증함으로써, u-Contract의 유효성을 검증해 볼 수 있다. 만약 과금 공증장치가 검증하여 이상이 없을 시에는, 일치하는 H(S) 값과 공증장치의 Cn 값을 확인 메시지(Confirm Message)(Msg4)(34, 35)로서, 서비스 제공자 및 사용자로부터 받은 랜덤 값과 함께 서비스 제공자 장치 및 사용자 단말장치로 전송함으로써 과금 연산을 종료한다.

[0076] 본 발명에서 사용자 과금 기록 입증 방법에 대해서 더욱 상세하게 살펴본다. 본 발명에서 마이크로계약서는 클라우드 서비스 사용자, 클라우드 서비스 제공자, 및 클라우드 과금 공증 기관 사이에서 대칭인 해쉬 암호작성법(hash cryptography) 및 공유키(distributed keys ($K_{u,c}$, $K_{u,n}$, $K_{c,n}$))에 의해서 생성된다. 본 발명에서 과금 정보 상호 검증을 위해서 클라우드 과금 공증 기관은 단계 3(State 3)(S30)에서 과금 처리(billing transaction)에 대한 사용자의 메시지의 유효성을 검증한다. 이러한 과금처리는 사용자 및 클라우드 서비스 제공자에 대해서 어떠한 비대칭키 연산도 요구하지 않는다. 과금 검증 단계(Msg3, Msg4)에서, 클라우드 과금 공증장치는 Msg2 메시지(32)와 함께 전송된 u-Contract-CSP와 Msg3(33)과 함께 전송된 u-Contract-User가 동일한 H(S)값을 이용하여 작성되었다는 것을 나타내는 구속력 있는 정보를 공증 과금 리스트(notarized billing list, NBL)의 형태로 보관한다. 여기서 의미하는 구속력 있는 정보란 H(S), H(S || Cu,n), H(S || Cc,n)으로 구성되는 것을 특징으로 한다. 과금에 관련된 모든 상황은 공증 과금 리스트(NBL)의 무결성을 담보하기 위해서 클라우드 과금 공증장치에 디지털 서명과 함께 저장된다. 이러한 공증 과금 리스트는 장래의 분쟁을 대비해서 클라우드 과금 공증 기관이 과금 처리의 증거로서 보관한다.

[0077] 컴퓨팅 자원의 사용에 대한 과금과 관련된 모든 처리들(transaction)은 클라우드 서비스 사용자, 클라우드 서비스 제공자, 및 클라우드 과금 공증장치의 해쉬 체인들에 의해 서로 링크되어 있기 때문에, NBL은 상호

검증 및 무결성을 제공할 수 있다. 예를 들어, 클라우드 서비스 제공자가 특정 사용자가 서비스 제공자의 과금에 대한 대금 지급을 거부한다고 주장하는 경우를 상정하면, 클라우드 서비스 제공자는 클라우드 과금 공증장치에 서비스 제공자의 마이크로계약서(μ-contract-CSP)내에 포함된 해쉬 체인 및 계약의 조항(stipulation)(S)을 제출하면서 판단을 구할 수 있다. 이 경우 클라우드 과금 공증장치는 사용자에게 사용자의 마이크로계약서(μ-contract- User) 작성에 이용된 조항 (S) 과 해쉬 요소($C_{u,n}$)를 요청하여 받은 후에 이를 해쉬 함수에 입력할 수 있다. 해쉬 함수의 출력값이 클라우드 과금 공증장치에 저장되어 있는 값과 상이한 경우, 그것은 사용자가 문제가 되고 있는 과금 건에 대해서 조항을 변조했다는 것을 입증한다. 따라서 클라우드 과금 공증장치는 사용자의 지급거절이 부당하다고 판단할 수 있다.

[0078] 본 발명의 클라우드 과금 검증 방법에서는 클라우드 과금 공증장치 역시 S값이 아닌 H(S)값을 사용하여 검증하므로, 공증장치도 함부로, 계약 조항에 대한 변조가 불가능하도록 구성된다.

[0079] 본 발명의 클라우드 과금 검증 방법은 악의적인 서비스 제공자 및 사용자의 과금 정보 변조에 대해서 안전하다. 클라우드 서비스 사용자 또는 서비스 제공자는 악의적으로 u-Contract의 변조를 할 수 있다. 그러나 공증장치에서 u-Contract-User와 u-Contract-CSP의 일치성 및 유효성을 검증하여 일치하지 않을 경우 더 이상 진행을 하지 않게 되므로, 임의의 일방에 의한 과금 데이터 조작 시 공정한 판단을 행하여 클라우드 컴퓨팅 환경에서의 과금에 대한 신뢰성을 높일 수 있다.

[0080] 클라우드 서비스 사용자 또는 서비스 제공자는 악의적으로 과금 연산 후에 u-Contract의 변조를 할 수 있다. 그러나 과금 시에 공증장치는 공증장치의 해쉬 체인값과 H(S)값을 저장하게 되므로, 변조된 H(S)값과의 비교를 통하여 진위를 가릴 수 있다.

[0081] 본 발명의 상호검증 가능한 과금 방법은 재시도 공격(Replay Attack)에 대해서도 안전하다. 재시도 공격과 관련해서는 1) 사용자의 자원 요청 메시지 (Msg1) 및 마이크로계약서 전송 메시지(Msg2) 및 마이크로 계약서 검증 요청 메시지(Msg3)에 대한 재시도 공격과 2) 과금 확인 메시지(confirmation message, Msg4)에 대한 재시도 공격을 가정할 수 있다. 전자의 경우에, 침입자는 캡쳐한 메시지의 전송에 의해 공격을 시도할 것이다. 그러나 해당 메시지는 nonce data (Nu) 및 각각의 과금 처리에서 타임 스탬프에 의해 변화되는 마이크로계약서를 포함하기 때문에 침입자는 공격에 성공할 수 없다. 후자의 경우에, 침입자는 캡쳐한 메시지의 전송에 의해 공격을 시도할 것이다. 그러나 해당 메시지는 nonce data (Nu, Nc, 및 Nn)를 포함하기 때문에 침입자는 공격에 성공할 수 없다.

[0082] 본 발명의 프로토콜은 중간자 공격 (Man in the Middle Attack, MITM)에 안전하다. MITM 공격은 일반적으로 1) 중간자가 사용자 단말장치와 서비스 제공자 장치 사이에서 공격하거나, 2) 중간자가 클라우드 과금 공증장치와 사용자 단말장치 사이에서 공격하거나, 또는 3) 클라우드 과금 공증장치와 클라우드 서비스 제공자 장치 사이에서 공격한다. 본 발명에서는 상기 1)의 경우에는 서비스 제공자 및 사용자가 과금 계약 메시지(billing transaction messages)를 생성하고, 이러한 메시지는 매번의 과금 계약 시마다 변경되고 이전 단계에서 나눠가진 공유키를 이용하여 암호화하여 전송되기 때문에, 침입자는 사용자 또는 서비스 제공자로 가장할 수 없다. 한편 2) 및 3)의 경우에는 각 당사자들이 PKI를 통해서 상호인증하고 보안 연결을 위한 키를 공유하기 때문에, 침입자가 클라우드 과금 공증장치의 메시지를 해킹할 수 없다.

[0083] 이상에서 본 발명을 실시하기 위한 구현예를 들어 본 발명에 대하여 상세하게 설명하였으나, 본 발명의 정신 및 범위를 벗어나지 않는 범위 내에서 본 발명이 속하는 기술 분야의 당업자들은 본 발명을 다양하게 변형 또는 변경하여 실시할 수 있을 것이다. 따라서 이러한 변형 및 변경들도 본 발명의 보호범위에 포함되는 것으로 이해되어야 할 것이다. 예들 들어, 본원에서 클라우드 컴퓨팅 자원의 사용과 관련하여 후불제 모델에 기초하여 설명하였으나, 본 발명의 방법은 사용 후에 과금되는 선불제 모델에도 적용될 수 있다.

부호의 설명

[0084] 10: 클라우드 서비스 사용자의 마이크로계약서(*u*-Contract-User)

20: 클라우드 서비스 제공자의 마이크로계약서(*u*-Contract-CSP)

100: 클라우드 서비스 사용자 단말장치

200: 클라우드 서비스 제공자 장치

300: 클라우드 서비스 과금 공증장치

110: 통신부

120: 마이크로계약서 작성부

130: 해쉬체인 생성부

210: 자원 매니저(resource manager)

220: 과금모듈 (billing agent)

230: 과금 자원 풀(billing resource pool)

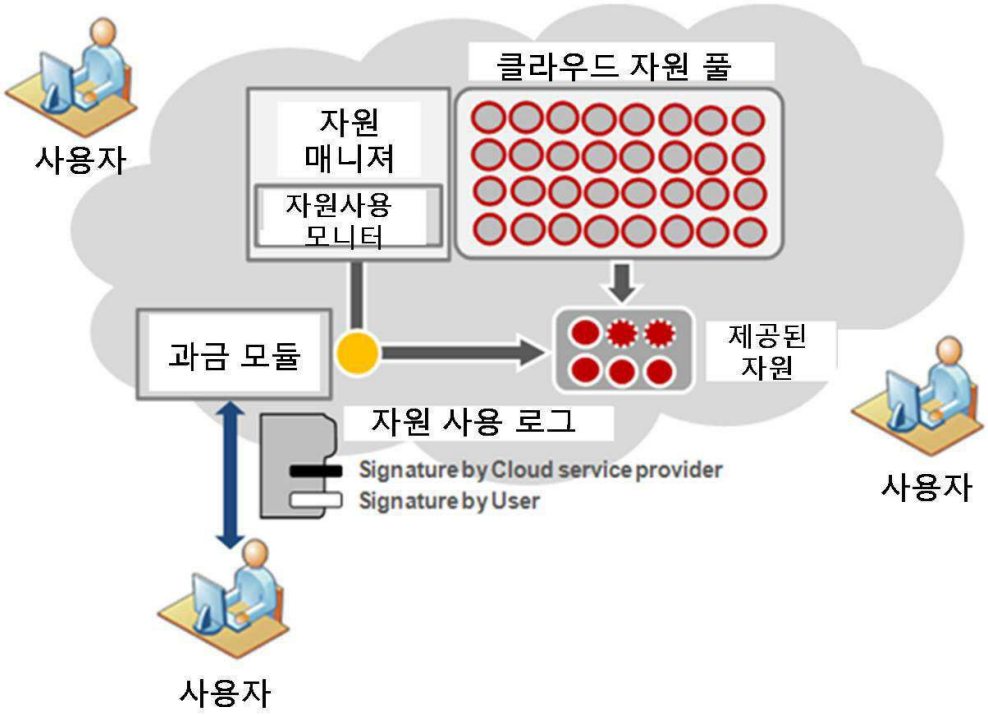
310: 통신부

330: 마이크로계약서 유효성 검증부

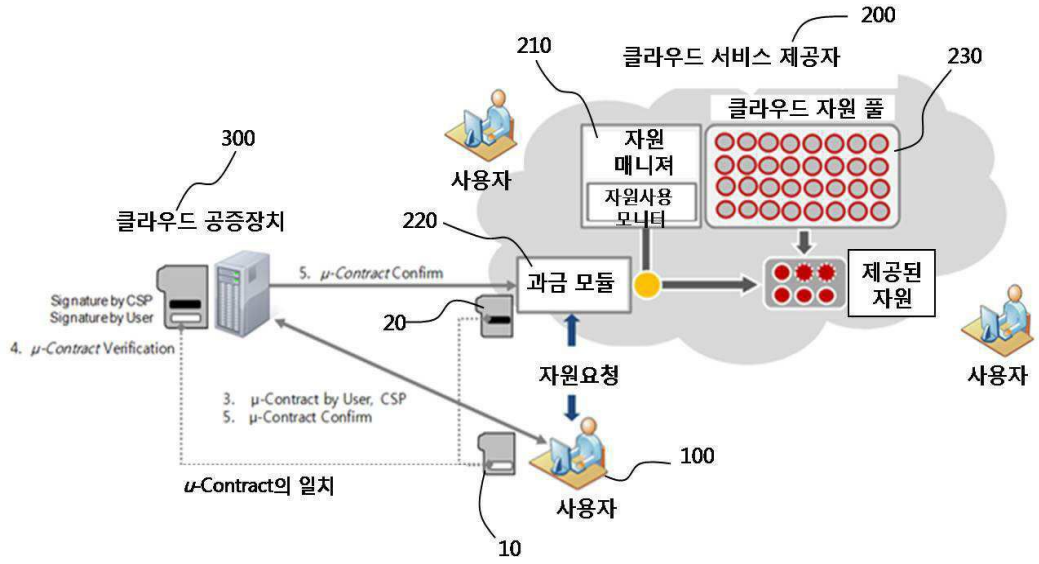
350: 과금정보 데이터베이스

도면

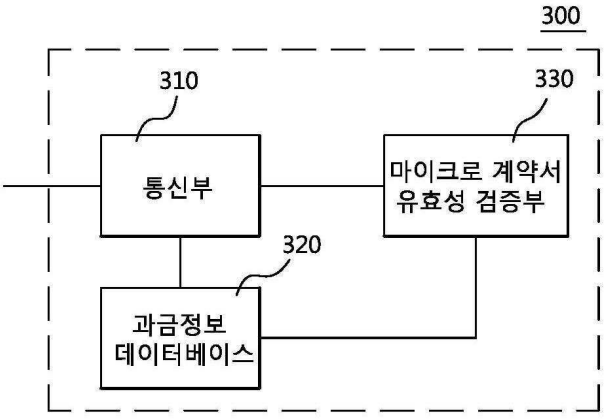
도면1



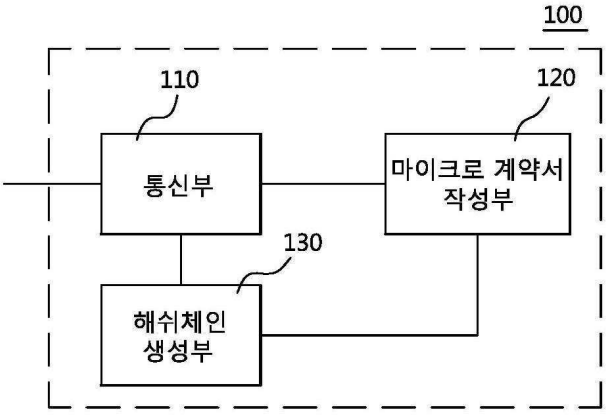
도면2


200
클라우드 서비스 제공자
210
230
클라우드 자원 풀
자원
매니저
자원사용
모니터
사용자
300
클라우드 공증장치
220
과금 모듈
제공된
자원
5. μ-Contract Confirm
Signature by CSP
Signature by User
20
4. μ-Contract Verification
사용자
자원요청
3. μ-Contract by User, CSP
5. μ-Contract Confirm
100
μ-Contract의 일치
사용자
10

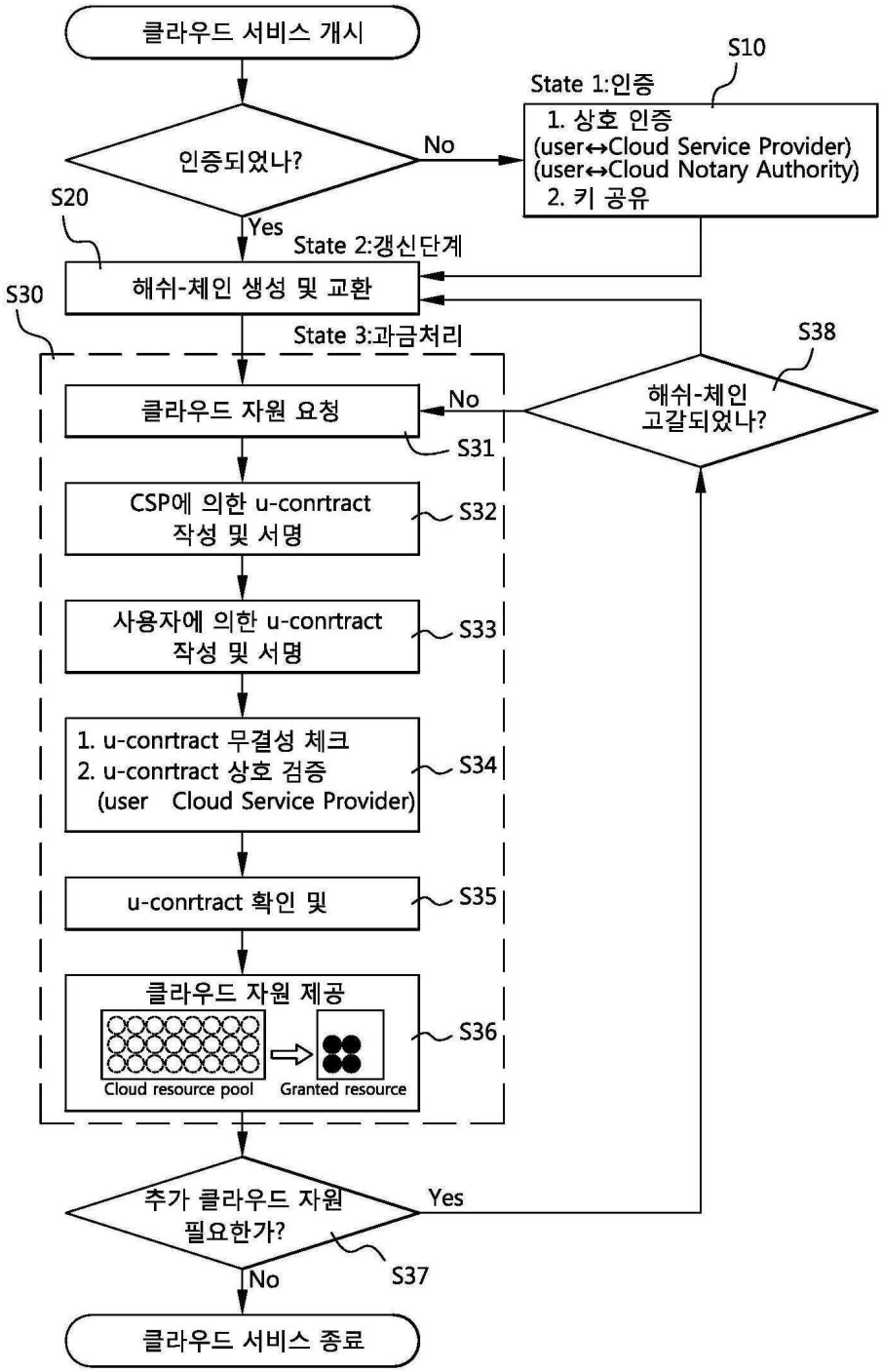
도면3


300
310
330
통신부
마이크로 계약서
유효성 검증부
320
과금정보
데이터베이스

도면4


100
110
120
통신부
마이크로 계약서
작성부
130
해쉬체인
생성부

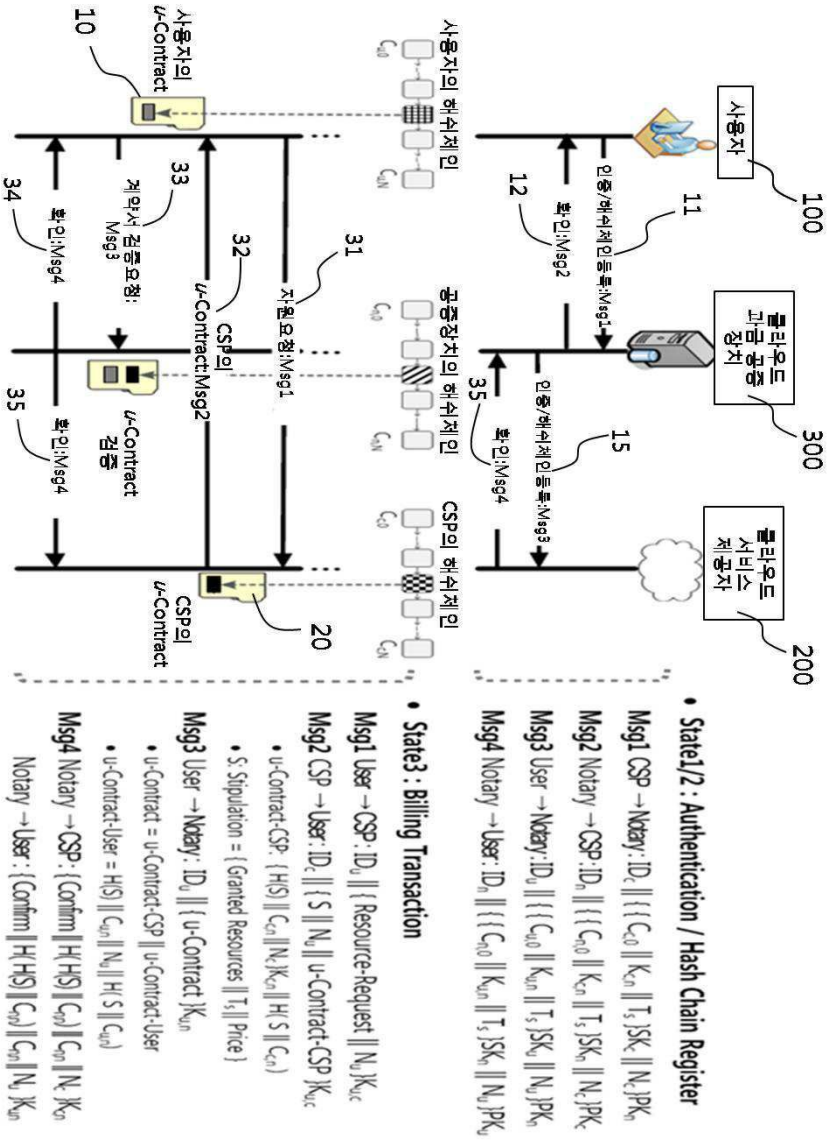
도면5


클라우드 서비스 개시
인증되었나?
No
Yes
S10
State 1:인증
1. 상호 인증
(user↔Cloud Service Provider)
(user↔Cloud Notary Authority)
2. 키 공유
S20
State 2:갱신단계
해쉬-체인 생성 및 교환
S30
State 3:과금처리
클라우드 자원 요청
No
S31
CSP에 의한 u-conrtract
작성 및 서명
S32
사용자에 의한 u-conrtract
작성 및 서명
S33
1. u-conrtract 무결성 체크
2. u-conrtract 상호 검증
(user Cloud Service Provider)
S34
u-conrtract 확인 및
S35
클라우드 자원 제공
Cloud resource pool
Granted resource
S36
S38
해쉬-체인
고갈되었나?
추가 클라우드 자원
필요한가?
Yes
No
S37
클라우드 서비스 종료

도면6

Definition of the entity symbols
c: Cloud Service Provider (CSP) u: User n: Cloud-Notary-Authority
Definition of the Message symbols
$K_{\alpha\beta}$: Shared Key between a and b PK_{α} : Public Key of a SK_{α} : Private Key of a H(M) : Hash result for message M Usage Sequence Hash Chain: $C_0 \rightarrow C_1 \rightarrow C_2 \rightarrow C_3 \rightarrow \ldots \rightarrow C_n$, $H(C_n) = C_{n-1}$ Hash Generation Sequence $C_{\alpha,n}$: n^{th} value of the α's hash chain T_s : Time-Stamp N_a: Random value for preventing replay attack by a { M }K : M encrypted by K { M }PK : M encrypted by public Key { M }SK : Digital signature for M by private key

도면7



- **State1/2 : Authentication / Hash Chain Register**

Msg1 CSP → Notary : $ID_c \| \{\{C_{c,0} \| K_{c,n} \| T_s\} SK_c \| N_c\} PK_n$

Msg2 Notary → CSP : $ID_n \| \{\{C_{n,0} \| K_{c,n} \| T_s\} SK_n \| N_c\} PK_c$

Msg3 User → Notary : $ID_u \| \{\{C_{u,0} \| K_{u,n} \| T_s\} SK_u \| N_u\} PK_n$

Msg4 Notary → User : $ID_n \| \{\{C_{n,0} \| K_{u,n} \| T_s\} SK_n \| N_u\} PK_u$

- **State3 : Billing Transaction**

Msg1 User → CSP : $ID_u \| \{$ Resource-Request $\| N_u\} K_{u,c}$

Msg2 CSP → User : $ID_c \| \{ S \| N_u \|$ u-Contract-CSP $\} K_{u,c}$

- u-Contract-CSP : $\{H(S) \| C_{c,n} \| N_c\} K_{c,n} \| H(S \| C_{c,n})$
- S: Stipulation = { Granted Resources $\| T_s \|$ Price }

Msg3 User → Notary : $ID_u \| \{$ u-Contract $\} K_{u,n}$

- u-Contract = u-Contract-CSP $\|$ u-Contract-User
- u-Contract-User = $H(S) \| C_{u,n} \| N_u \| H(S \| C_{u,n})$

Msg4 Notary → CSP : $\{$Confirm $\| H(H(S) \| C_{n,n}) \| C_{n,n} \| N_c\} K_{c,n}$

Notary → User : $\{$Confirm $\| H(H(S) \| C_{n,n}) \| C_{n,n} \| N_u\} K_{u,n}$