(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2011년09월07일
(11) 등록번호 10-1063354
(24) 등록일자 2011년09월01일

(51) Int. Cl.
G06Q 20/00 (2006.01) *G06F 21/00* (2006.01)
(21) 출원번호 10-2009-0069639
(22) 출원일자 2009년07월29일
심사청구일자 2009년07월29일
(65) 공개번호 10-2011-0012085
(43) 공개일자 2011년02월09일
(56) 선행기술조사문헌
국내학회논문 한국차세대컴퓨팅학회 논문지, Vol.5, No.4, pp.4-17(2009.12.)
JP2004272669 A*
KR1020070044473 A
JP2009507427 A
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
한국과학기술원
대전 유성구 구성동 373-1
(72) 발명자
박규호
대전시 유성구 구성동 KAIST 전자과
박기웅
서울 노원구 월계4동 500-11번지 8/5
박성규
대전 유성구 구성동 KAIST 동측기숙사 6118
(74) 대리인
이원희

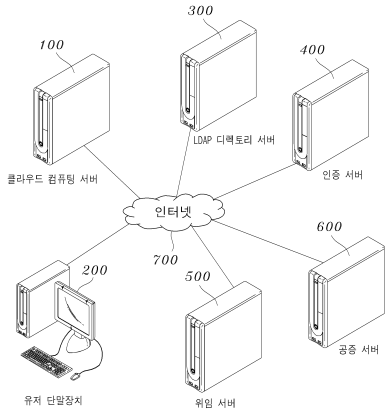
전체 청구항 수 : 총 12 항 심사관 : 곽중환

(54) 공개 키 기반의 프로토콜을 이용한 과금 시스템 및 그 방법

(57) 요 약

공개 키 기반의 프로토콜을 이용한 과금 시스템이 개시된다. 본 발명에 따른 과금 시스템은, 외부로부터 자원 요청이 있을 경우 요청된 자원의 과금에 대한 제1 마이크로 계약서를 생성하는 클라우드 컴퓨팅 서버, 제1 마이크로 계약서를 수신하고 요청한 자원의 과금에 대한 제2 마이크로 계약서를 생성하여 암호화하는 유저 단말장치, 유저 단말장치에서 사용한 자원에 대한 과금 연산을 위임받은 대행 인증서를 포함하며 유저 단말장치로부터 암호화된 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 수신하고 과금을 연산하여 클라우드 컴퓨팅 서버에 자원 제공 신호를 전송하여 유저 단말장치에 해당 자원을 할당하도록 하는 위임 서버 및 암호화된 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 전달받아 저장하고, 검증 요청이 있을 경우 클라우드 컴퓨팅 서버에서 제공한 자원의 과금과, 유저 단말장치에서 사용한 자원에 대한 과금을 검증하는 공증 서버를 포함한다.

대 표 도 - 도3



특허청구의 범위

청구항 1

복수의 자원을 구비하며, 외부로부터 자원 요청이 있을 경우 요청된 자원의 과금에 대한 제1 마이크로 계약서를 생성하는 클라우드 컴퓨팅 서버;

상기 클라우드 컴퓨팅 서버에 접속하여 자원을 요청하여 상기 제1 마이크로 계약서를 수신하고, 요청한 자원의 과금에 대한 제2 마이크로 계약서를 생성하여 암호화하는 유저 단말장치;

상기 유저 단말장치에서 사용한 자원에 대한 과금 연산을 위임받은, 대행 인증서를 포함하며, 상기 유저 단말장치로부터 암호화된 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 수신하고 과금을 연산하여 상기 클라우드 컴퓨팅 서버에 자원 할당 신호를 전송하여 상기 유저 단말장치에 해당 자원을 할당하도록 하는 위임 서버; 및,

상기 위임 서버로부터 암호화된 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 전달받아 저장하고, 과금 검증 요청이 있을 경우 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 복호화하여 비교하고 상기 클라우드 컴퓨팅 서버에서 제공한 자원의 과금과, 상기 유저 단말장치에서 사용한 자원에 대한 과금을 검증하는 공증 서버;를 포함하며,

상기 유저 단말장치는, 상기 유저 단말장치와 상기 위임 서버만 인식할 수 있는 제1 공유키와, 상기 유저 단말장치와 상기 공증 서버만 인식할 수 있는 제2 공유키를 이용하여 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 암호화하는 것을 특징으로 하는 공개 키 기반의 프로토콜을 이용한 과금 시스템.

청구항 2

삭제

청구항 3

제1항에 있어서,

상기 제1 마이크로 계약서는,

상기 제1 마이크로 계약서의 일련 번호를 나타내는 제1 필드,

상기 제1 마이크로 계약서의 일련 번호, 상기 유저 단말장치에 할당한 자원 및 상기 클라우드 컴퓨팅 서버에서 생성된 무작위 데이터에 대한 해시(Hash) 값을 나타내는 제2 필드 및

상기 제1 마이크로 계약서의 일련 번호 및 상기 유저 단말장치에 할당한 자원에 대한 해시 값을 상기 클라우드 컴퓨팅 서버와 상기 공증 서버만이 인식할 수 있는 제3 공유키($K_{Notary,Cloud}$)로 암호화된 값을 나타내는 제3 필드로 이루어진 것을 특징으로 하는 공개 키 기반의 프로토콜을 이용한 과금 시스템.

청구항 4

제3항에 있어서,

상기 제2 마이크로 계약서는,

상기 제2 마이크로 계약서의 일련 번호를 나타내는 제1 필드,

상기 제2 마이크로 계약서의 일련 번호, 상기 유저 단말장치가 할당받은 자원 및 상기 유저 단말장치에서 생성된 무작위 데이터에 대한 해시(Hash) 값을 나타내는 제2 필드 및

상기 제2 마이크로 계약서의 일련 번호 및 상기 유저 단말장치가 할당받은 자원에 대한 해시 값을 나타내는 제3 필드로 이루어진 것을 특징으로 하는 공개 키 기반의 프로토콜을 이용한 과금 시스템.

청구항 5

제4항에 있어서,

상기 공증 서버는,

상기 검증 요청이 있을 경우 상기 위임 서버와 상기 공증 서버만 인식할 수 있는 제4 공유키를 이용하여 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 복호화하는 것을 특징으로 하는 공개 키 기반의 프로토콜을 이용한 과금 시스템.

청구항 6

제5항에 있어서,

상기 공증 서버는,

상기 검증 요청이 있을 경우, 상기 유저 단말장치에 할당한 것으로 예측되는 제1 예측 할당 자원 및 제1 무작위 데이터를 상기 클라우드 컴퓨팅 서버로부터 수신하고, 상기 클라우드 컴퓨팅 서버로부터 할당받은 것으로 예측되는 제2 예측 할당 자원 및 제2 무작위 데이터를 상기 유저 단말장치로부터 수신하여, 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서 각각에 포함된 상기 제2 필드 값과 비교하고 일치 여부를 확인하여 과금을 검증하는 것을 특징으로 하는 공개 키 기반의 프로토콜을 이용한 과금 시스템.

청구항 7

제1항에 있어서,

상기 유저 단말장치는,

상기 클라우드 컴퓨팅 서버, 상기 위임 서버 및 상기 공증 서버와 상호 인증된 상태인 것을 특징으로 하는 공개 키 기반의 프로토콜을 이용한 과금 시스템.

청구항 8

클라우드 컴퓨팅 서버에 자원을 요청하여 할당받는 유저 단말장치와, 상기 유저 단말장치의 자원 사용에 따른 과금 연산을 위임받은 위임 서버 및 상기 클라우드 컴퓨팅 서버에서 제공한 자원의 과금과, 상기 유저 단말장치에서 사용한 자원에 대한 과금을 검증하는 공증 서버를 포함하는 과금 시스템의 공개 키 기반의 프로토콜을 이용한 과금 검증 방법에 있어서,

상기 유저 단말장치가 클라우드 컴퓨팅 서버에 필요한 자원을 요청하는 제1 단계;

상기 클라우드 컴퓨팅 서버가 상기 유저 단말장치로부터 요청된 상기 자원의 과금에 대한 제1 마이크로 계약서를 생성하여 상기 유저 단말장치에 전송하는 제2 단계;

상기 유저 단말장치가 상기 요청한 자원의 과금에 대한 제2 마이크로 계약서를 생성하여, 상기 클라우드 컴퓨팅 서버로부터 수신된 제1 마이크로 계약서와 함께 암호화한 후 상기 위임 서버에 전송하는 제3 단계;

상기 위임 서버가 상기 유저 단말장치로부터 암호화된 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 수신하여 과금을 연산하고, 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 상기 공증 서버에 전송하는 제4 단계; 및

상기 공증 서버가 상기 위임 서버로부터 암호화된 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 전달받아 저장하고, 검증 요청이 있을 경우 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 복호화하고 비교하여 과금을 검증하는 제5 단계;를 포함하며,

상기 제1 단계는, 상기 유저 단말장치가 클라우드 컴퓨팅 서버에 필요한 자원을 요청하면, 상기 유저 단말장치의 과금 연산을 위임받은 위임 서버의 유무 및 상기 유저 단말장치와 상기 클라우드 컴퓨팅 서버 간의 상호 인증 여부를 확인하는 제1 과정;

상기 과금 연산을 위임받은 위임 서버가 존재하지 않으면 일 위임 서버로부터 공개키 및 개인키를 수신하고, 상기 공개키를 상기 유저 단말장치의 개인키로 서명하여 대행 인증서를 생성한 후 상기 위임 서버에 전송하여 과금 연산을 위임시키는 제2 과정; 및

상기 유저 단말장치와 상기 클라우드 컴퓨팅 서버 간의 상호 인증이 되어 있지 않으면 상기 클라우드 컴퓨팅 서버로부터 인증 캡슐을 수신하고, 상기 유저 단말장치와 상기 위임 서버 간의 제1 공유키, 상기 유저 단말장치와 상기 공증 서버 간의 제2 공유키 및 상기 유저 단말장치와 상기 클라우드 컴퓨팅 서버 간의 제3 공유키를 분배하여 상호 인증하는 제3 과정을 포함하는 것을 특징으로 하는 공개 키 기반의 프로토콜을 이용한 과금 방법.

청구항 9

삭제

청구항 10

제8항에 있어서,

상기 제3 단계는,

상기 유저 단말장치와 상기 위임 서버 간의 제1 공유키와, 상기 유저 단말장치와 상기 공증 서버 간의 제2 공유키를 이용하여 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 암호화하는 것을 특징으로 하는 공개키 기반의 프로토콜을 이용한 과금 방법.

청구항 11

제10항에 있어서,

상기 제1 마이크로 계약서는,

상기 제1 마이크로 계약서의 일련 번호를 나타내는 제1 필드,

상기 제1 마이크로 계약서의 일련 번호, 상기 유저 단말장치에 할당한 자원 및 상기 클라우드 컴퓨팅 서버에서 생성된 무작위 데이터에 대한 해시(Hash) 값을 나타내는 제2 필드 및

상기 제1 마이크로 계약서의 일련 번호 및 상기 유저 단말장치에 할당한 자원에 대한 해시 값을 나타내는 제3 필드로 이루어진 것을 특징으로 하는 공개 키 기반의 프로토콜을 이용한 과금 방법.

청구항 12

제11항에 있어서,

상기 제2 마이크로 계약서는,

상기 제2 마이크로 계약서의 일련 번호를 나타내는 제1 필드,

상기 제2 마이크로 계약서의 일련 번호, 상기 유저 단말장치가 할당받은 자원 및 상기 유저 단말장치에서 생성된 무작위 데이터에 대한 해시(Hash) 값을 나타내는 제2 필드 및

상기 제2 마이크로 계약서의 일련 번호 및 상기 유저 단말장치가 할당받은 자원에 대한 해시 값을 나타내는 제3 필드로 이루어진 것을 특징으로 하는 공개 키 기반의 프로토콜을 이용한 과금 방법.

청구항 13

제12항에 있어서,

상기 제5 단계는,

상기 검증 요청이 있을 경우 상기 위임 서버와 상기 공증 서버 간의 제4 공유키를 이용하여 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 복호화하는 것을 특징으로 하는 공개 키 기반의 프로토콜을 이용한 과금 방법.

청구항 14

제13항에 있어서,

상기 제5 단계는,

상기 검증 요청이 있을 경우 상기 유저 단말장치에 할당한 것으로 예측되는 제1 예측 할당 자원 및 제1 무작위 데이터를 상기 클라우드 컴퓨팅 서버로부터 수신하고, 상기 클라우드 컴퓨팅 서버로부터 할당받은 것으로 예측되는 제2 예측 할당 자원 및 제2 무작위 데이터를 상기 유저 단말장치로부터 수신하여, 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서 각각에 포함된 상기 제2 필드 값과 비교하고 일치 여부를 확인하여 과금을 검증하는 것을 특징으로 하는 공개 키 기반의 프로토콜을 이용한 과금 방법.

명 세 서

발명의 상세한 설명

기 술 분 야

[0001] 본 발명은 공개 키 기반의 프로토콜을 이용한 과금 시스템 및 그 방법에 관한 것으로, 보다 상세하게는, 위임 서버를 통해 과금 연산을 수행하고 공증 서버를 통해 자원에 대한 과금을 검증하여 부인을 방지할 수 있는 공개 키 기반의 프로토콜을 이용한 과금 시스템 및 그 방법에 관한 것이다.

배 경 기 술

[0002] 클라우드 컴퓨팅(Cloud Computing)이란, 인터넷 기반(클라우드)의 컴퓨팅(computing) 기술을 의미한다. 클라우드 컴퓨팅은 컴퓨터 네트워크 구성도에서 인터넷을 구름으로 표현하는 것으로, 숨겨진 복잡한 인프라 구조를 가지며 IT 관련된 기능들이 서비스 형태로 제공되는 컴퓨팅 스타일을 갖는다. 사용자들은 인터넷을 이용하여 클라우드 컴퓨팅으로부터 제공되는 서비스를 이용할 수 있다.

[0003] 도 1은 종래 기술에 따른 클라우드 컴퓨팅 환경의 과금 시스템을 나타내는 도면이다. 도 1을 참조하면, 과금 시스템은 클라우드 컴퓨팅 서버(10), 유저 단말장치(20)를 포함하는 것으로, 이들은 인터넷(30)에 의해 연결되어 있다.

[0004] 유저 단말장치(20)가 클라우드 컴퓨팅 서버(10)에 필요한 자원(Source)을 요청하면, 클라우드 컴퓨팅 서버(10)는 요청에 따라 해당 자원을 유저 단말장치(20)로 전송하는 시스템이다. 이 과정에서, 클라우드 컴퓨팅 서버(10)는 자원을 전송하고, 이에 대한 요금을 결정하여 유저 단말장치(20)에 과금한다. 이 같은 구조의 클라우드 컴퓨팅 기반의 과금 방법은 클라우드 컴퓨팅 서버(10)에서 요금을 결정하여 유저 단말장치(20)에 통보하는 것으로, 일방적인 과금 통보 구조로 이루어진다. 따라서, 유저 단말장치(20)에 의한 전자 서명이나 부인 방지 등의 기능이 제공되지 않으므로 사용자는 유저 단말장치(20)를 이용하여 사용한 자원에 대해서 확인하기 어려워 부당한 요금이 결정되더라도 수용할 수 밖에 없었다.

[0005] 도 1에 도시된 클라우드 컴퓨팅 환경에서 과금에 따른 문제점을 해결하기 위하여 공개 키 기반(PKI:Public Key Infrastructure)에 의한 과금 시스템이 제안되었다.

[0006] 도 2는 종래 기술에 따른 공개 키 기반으로 한 클라우드 컴퓨팅 환경의 과금 시스템을 나타내는 도면이다. 도 2를 참조하면, 클라우드 컴퓨팅 환경의 과금 시스템은 클라우드 컴퓨팅 서버(10), 유저 단말장치(20), LDAP(Lightweight Directory Access Protocol) 디렉토리 서버(40) 및 인증 서버(50)를 포함하며, 이들은 인터넷(30)을 통해 연결된다.

[0007] 도 2에 도시된 클라우드 컴퓨팅 환경은 도 1과는 달리, LDAP 디렉토리 서버(40) 및 인증 서버(50)를 이용하여 인증 과정을 거치게 된다. 이 경우, LDAP 디렉토리 서버(40)는 클라우드 컴퓨팅 서버(10) 및 유저 단말장치(20)에 대한 인증서를 저장하고 있다. 그리고, 인증 서버(50)는 인증 인프라 내부에 있는 클라우드 컴퓨팅 서버(10) 및 유저 단말장치(20)의 인증서 생성 및 관리를 담당한다.

[0008] 구체적으로, 유저 단말장치(20)가 소정의 자원을 요청하기 위한 자원 요청 신호를 클라우드 컴퓨팅 서버(10)에 전송하면, 클라우드 컴퓨팅 서버(10)는 유저 단말장치(20)의 인증서를 LDAP 디렉토리 서버(40)에 요청하여 수신한다. 이 과정에서 클라우드 컴퓨팅 서버(10)는 유저 단말장치(20)가 요청한 자원을 유저 단말장치(20)에 할당하고, 개인키와 RSA 알고리즘 연산을 이용한 전자 서명을 생성하여 유저 단말장치(30)에 전송한다.

[0009] 유저 단말장치(20)를 사용하는 유저는 클라우드 컴퓨팅 서버(10)의 공개키를 이용하여 수신된 전자 서명에 대한 무결성을 확인하고, 클라우드 컴퓨팅 서버(10)로부터 제공받은 자원에 대한 사용 기록을 확인할 수 있다. 확인

결과, 사용 기록이 정확한 경우에는 유저 단말장치(20)의 개인키를 이용하여 사용 기록 확인에 대한 전자 서명을 생성하여 클라우드 컴퓨팅 서버(10)에 전송하고, 이 과정에 의해 클라우드 컴퓨팅 서버(10)는 과금하게 된다. 이 같은 과정은 유저가 유저 단말장치(20)를 통해 직접 사용 기록을 확인하여야 하는 것으로 사용자 편의성이 저하된다.

[0010] 또한, RSA 알고리즘 연산은 인터넷 암호화 및 인증시스템으로 두 개의 140자리 이상의 수인 소수를 이용하고, 이 수들의 곱 및 추가 연산을 통하여 공개키 및 개인키를 구성하고, 인터넷에서 사용하는 정보를 암호화하고 복호화할 수 있다. 유저 단말장치(20)가 이 같은 RSA 알고리즘 연산을 이용한 전자 서명을 확인하기 위해서는 RSA 알고리즘 연산을 수행할 수 있을 정도의 고속 프로세서를 내장하고 있어야 한다. 그러나, 클라우드 컴퓨팅 서버(10)를 이용하는 다수의 유저 단말장치가 모두 고속 프로세서를 내장하고 있지 않으므로, RSA 알고리즘 연산을 수행하는데 있어서 오류가 발생할 수 있다.

[0011] 그리고, 클라우드 컴퓨팅 서버(10) 역시 다수의 유저 단말장치 각각에 대한 전자 서명을 생성해야 하므로, 과금 연산 및 에 따른 오버헤드가 발생하게 된다는 문제점이 있다.

발명의 내용

해결 하고자하는 과제

[0012] 본 발명은 상술한 문제점을 해결하기 위한 것으로, 본 발명의 목적은, 유저 단말장치에 대한 과금을 위임받은 위임 서버를 이용함으로써 유저 단말장치에서의 과금 연산으로 인한 오버헤드를 감소시킬 수 있으며, 과금을 검증하기 위한 공증 서버를 이용하여 클라우드 컴퓨팅 서버와 유저 단말장치 간의 자원 사용에 대한 과금을 검증함으로써 과금에 대한 부인을 방지할 수 있는 공개 키 기반의 프로토콜을 이용한 과금 시스템 및 그 방법에 관한 것이다.

과제 해결수단

[0013] 이상과 같은 목적을 달성하기 위한 본 발명의 일 실시 예에 따른 공개 키 기반의 프로토콜을 이용한 과금 시스템은, 복수의 자원을 구비하며, 외부로부터 자원 요청이 있을 경우 요청된 자원의 과금에 대한 제1 마이크로 계약서를 생성하는 클라우드 컴퓨팅 서버, 상기 클라우드 컴퓨팅 서버에 접속하여 자원을 요청하여 상기 제1 마이크로 계약서를 수신하고, 요청한 자원의 과금에 대한 제2 마이크로 계약서를 생성하여 암호화하는 유저 단말장치, 상기 유저 단말장치에서 사용한 자원에 대한 과금 연산을 위임받은 대행 인증서를 포함하며, 상기 유저 단말장치로부터 암호화된 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 수신하고 과금을 연산하여 상기 클라우드 컴퓨팅 서버에 자원 할당 신호를 전송하여 상기 유저 단말장치에 해당 자원을 할당하도록 하는 위임 서버 및 상기 위임 서버로부터 암호화된 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 전달받아 저장하고, 과금 검증 요청이 있을 경우 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 복호화하여 비교하고 상기 클라우드 컴퓨팅 서버에서 제공한 자원의 과금과, 상기 유저 단말장치에서 사용한 자원에 대한 과금을 검증하는 공증 서버를 포함한다.

[0014] 한편, 상기 유저 단말장치는 상기 유저 단말장치와 상기 위임 서버만 인식할 수 있는 제1 공유키와, 상기 유저 단말장치와 상기 공증 서버만 인식할 수 있는 제2 공유키를 이용하여 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 암호화하는 것을 특징으로 한다.

[0015] 이 경우, 상기 제1 마이크로 계약서는 상기 제1 마이크로 계약서의 일련 번호를 나타내는 제1 필드, 상기 제1 마이크로 계약서의 일련 번호, 상기 유저 단말장치에 할당한 자원 및 상기 클라우드 컴퓨팅 서버에서 생성된 무작위 데이터에 대한 해시(Hash) 값을 나타내는 제2 필드 및 상기 제1 마이크로 계약서의 일련 번호 및 상기 유저 단말장치에 할당한 자원에 대한 해시 값을 상기 클라우드 컴퓨팅 서버와 상기 공증 서버만이 인식할 수 있는 제3 공유키($K_{Notary,Cloud}$)로 암호화된 값을 나타내는 제3 필드로 이루어질 수 있다.

[0016] 또한, 상기 제2 마이크로 계약서는 상기 제2 마이크로 계약서의 일련 번호를 나타내는 제1 필드, 상기 제2 마이크로 계약서의 일련 번호, 상기 유저 단말장치가 할당받은 자원 및 상기 유저 단말장치에서 생성된 무작위 데이터에 대한 해시(Hash) 값을 나타내는 제2 필드 및 상기 제2 마이크로 계약서의 일련 번호 및 상기 유저 단말장

치가 할당받은 자원에 대한 해시 값을 나타내는 제3 필드로 이루어질 수 있다.

[0017] 한편, 상기 공증 서버는 상기 검증 요청이 있을 경우 상기 위임 서버와 상기 공증 서버만 인식할 수 있는 제4 공유키를 이용하여 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 복호화할 수 있다.

[0018] 또한, 상기 공증 서버는, 상기 검증 요청이 있을 경우, 상기 유저 단말장치에 할당한 것으로 예측되는 제1 예측 할당 자원 및 제1 무작위 데이터를 상기 클라우드 컴퓨팅 서버로부터 수신하고, 상기 클라우드 컴퓨팅 서버로부터 할당받은 것으로 예측되는 제2 예측 할당 자원 및 제2 무작위 데이터를 상기 유저 단말장치로부터 수신하여, 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서 각각에 포함된 상기 제2 필드 값과 비교하고 일치 여부를 확인하여 과금을 검증할 수 있다.

[0019] 본 과금 시스템에서 상기 유저 단말장치는 상기 클라우드 컴퓨팅 서버, 상기 위임 서버 및 상기 공증 서버와 상호 인증된 상태일 수 있다.

[0020] 한편, 클라우드 컴퓨팅 서버에 자원을 요청하여 할당받는 유저 단말장치와, 상기 유저 단말장치의 자원 사용에 따른 과금 연산을 위임받은 위임 서버 및 상기 클라우드 컴퓨팅 서버에서 제공한 자원의 과금과, 상기 유저 단말장치에서 사용한 자원에 대한 과금을 검증하는 공증 서버를 포함하는 과금 시스템의 공개 키 기반의 프로토콜을 이용한 과금 검증 방법은 상기 유저 단말장치가 클라우드 컴퓨팅 서버에 필요한 자원을 요청하는 제1 단계, 상기 클라우드 컴퓨팅 서버가 상기 유저 단말장치로부터 요청된 상기 자원의 과금에 대한 제1 마이크로 계약서를 생성하여 상기 유저 단말장치에 전송하는 제2 단계, 상기 유저 단말장치가 상기 요청한 자원의 과금에 대한 제2 마이크로 계약서를 생성하여, 상기 클라우드 컴퓨팅 서버로부터 수신된 제1 마이크로 계약서와 함께 암호화한 후 상기 위임 서버에 전송하는 제3 단계, 상기 위임 서버가 상기 유저 단말장치로부터 암호화된 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 수신하여 과금을 연산하고, 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 상기 공증 서버에 전송하는 제4 단계 및 상기 공증 서버가 상기 위임 서버로부터 암호화된 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 전달받아 저장하고, 검증 요청이 있을 경우 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 복호화하고 비교하여 과금을 검증하는 제5 단계를 포함한다.

[0021] 이 경우, 상기 제1 단계는 상기 유저 단말장치가 클라우드 컴퓨팅 서버에 필요한 자원을 요청하면, 상기 유저 단말장치의 과금 연산을 위임받은 위임 서버의 유무 및 상기 유저 단말장치와 상기 클라우드 컴퓨팅 서버 간의 상호 인증 여부를 확인하는 제1 과정, 상기 과금 연산을 위임받은 위임 서버가 존재하지 않으면 일 위임 서버로부터 공개키 및 개인키를 수신하고, 상기 공개키를 상기 유저 단말장치의 개인키로 서명하여 대행 인증서를 생성한 후 상기 위임 서버에 전송하여 과금 연산을 위임시키는 제2 과정 및, 상기 유저 단말장치와 상기 클라우드 컴퓨팅 서버 간의 상호 인증이 되어 있지 않으면 상기 클라우드 컴퓨팅 서버로부터 인증 캡슐을 수신하고, 상기 유저 단말장치와 상기 위임 서버 간의 제1 공유키, 상기 유저 단말장치와 상기 공증 서버 간의 제2 공유키 및 상기 유저 단말장치와 상기 클라우드 컴퓨팅 서버 간의 제3 공유키를 분배하여 상호 인증하는 제3 과정을 포함할 수 있다.

[0022] 한편, 상기 제3 단계는 상기 유저 단말장치와 상기 위임 서버 간의 제1 공유키와, 상기 유저 단말장치와 상기 공증 서버 간의 제2 공유키를 이용하여 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 암호화할 수 있다.

[0023] 이 경우, 상기 제1 마이크로 계약서는 상기 제1 마이크로 계약서의 일련 번호를 나타내는 제1 필드, 상기 제1 마이크로 계약서의 일련 번호, 상기 유저 단말장치에 할당한 자원 및 상기 클라우드 컴퓨팅 서버에서 생성된 무작위 데이터에 대한 해시(Hash) 값을 나타내는 제2 필드 및 상기 제1 마이크로 계약서의 일련 번호 및 상기 유저 단말장치에 할당한 자원에 대한 해시 값을 나타내는 제3 필드로 이루어진 것일 수 있다.

[0024] 또한, 상기 제2 마이크로 계약서는 상기 제2 마이크로 계약서의 일련 번호를 나타내는 제1 필드, 상기 제2 마이크로 계약서의 일련 번호, 상기 유저 단말장치가 할당받은 자원 및 상기 유저 단말장치에서 생성된 무작위 데이터에 대한 해시(Hash) 값을 나타내는 제2 필드 및, 상기 제2 마이크로 계약서의 일련 번호 및 상기 유저 단말장치가 할당받은 자원에 대한 해시 값을 나타내는 제3 필드로 이루진 것일 수 있다.

[0025] 한편, 상기 제5 단계는 상기 검증 요청이 있을 경우 상기 위임 서버와 상기 공증 서버 간의 제4 공유키를 이용하여 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서를 복호화할 수 있다.

[0026] 또한, 상기 제5 단계는 상기 검증 요청이 있을 경우 상기 유저 단말장치에 할당한 것으로 예측되는 제1 예측 할당 자원 및 제1 무작위 데이터를 상기 클라우드 컴퓨팅 서버로부터 수신하고, 상기 클라우드 컴퓨팅 서버로부터 할당받은 것으로 예측되는 제2 예측 할당 자원 및 제2 무작위 데이터를 상기 유저 단말장치로부터 수신하여, 상기 제1 마이크로 계약서 및 상기 제2 마이크로 계약서 각각에 포함된 상기 제2 필드 값과 비교하고 일치 여부를 확인하여 과금을 검증할 수 있다.

효 과

[0027] 본 발명에 따르면, 유저 단말장치가 클라우드 컴퓨팅 서버를 이용하여 필요한 자원을 제공받는 클라우드 컴퓨팅 환경에서, 유저 단말장치에 대한 과금을 위임받은 위임 서버를 이용함으로써 유저 단말장치는 RSA 알고리즘 연산을 수행할 필요가 없게 되어 오버헤드 발생이 감소되며 고속 프로세서를 내장할 필요가 없게 되어 사양 부담이 감소할 수 있다.

[0028] 또한, 클라우드 컴퓨팅 서버에서 생성된 제1 마이크로 계약서와 유저 단말장치에서 생성된 제2 마이크로 계약서를 공증 서버에 저장해둠으로써, 클라우드 컴퓨팅 서버에서 제공한 자원의 과금과, 상기 유저 단말장치에서 사용한 자원에 대한 과금을 검증하여 과금에 따른 부인을 방지할 수 있게 된다.

발명의 실시를 위한 구체적인 내용

[0029] 이하에서는 첨부된 도면을 참조하여 본 발명을 보다 자세하게 설명한다.

[0030] 도 3은 본 발명의 일 실시 예에 따른 공개 키 기반의 프로토콜을 이용한 과금 시스템을 나타내는 도면이다. 도 3을 참조하면, 과금 결제 시스템은 클라우드 컴퓨팅 서버(100), 유저 단말장치(200), LDAP 디렉토리 서버(300), 인증 서버(400), 위임 서버(500), 공증 서버(600) 및 인터넷망(700)을 포함한다.

[0031] 클라우드 컴퓨팅 서버(100)는 각종 프로그램이나 자료들 등과 같은 다양한 자원을 저장하고 있는 대형 컴퓨터로써, 인터넷망(700)을 통해 접속 가능한 외부 기기(예를 들어, PC 등)에 프로그램이나 자료 등을 제공하며, 프로그램이나 자료 제공에 따른 과금을 청구할 수 있다.

[0032] 유저 단말장치(200)는 일반 개인이 사용하는 PC가 될 수 있으며, 연산 성능이 낮은 저속의 프로세서를 구비한 저가형 PC로 이루어질 수 있다. 도면 상에는 설명의 편의를 위하여 하나의 PC만을 도시하였으나, 다수의 PC가 인터넷망(700)을 통해 클라우드 컴퓨팅 서버(100)에 접속할 수 있다.

[0033] 인증 서버(400)는 인증 인프라 내부에 있는 클라우드 컴퓨팅 서버(100) 및 유저 단말장치(200)의 인증서 관리 및 서명을 담당한다. 그리고, LDAP 디렉토리 서버(300)는 클라우드 컴퓨팅 서버(100) 및 유저 단말장치(200)에 대한 인증서를 저장하고 있으며, 클라우드 컴퓨팅 서버(100) 및 유저 단말장치(200)가 인증을 요구하면 그에 대응되는 인증서를 전송한다.

[0034] 한편, 위임 서버(500)는 유저 단말장치(200)의 과금 연산을 위임받아 유저 단말장치(200)에 대한 과금 연산을 대행하는 서버로, 유저 단말장치(200)로부터 수신한 대행 인증서를 포함한다.

[0035] 공증 서버(600)는 클라우드 컴퓨팅 서버(100)가 유저 단말장치(200)에 할당한 자원의 과금과 유저 단말장치

(200)가 클라우드 컴퓨팅 서버(100)로부터 할당받아 사용한 자원에 대한 과금을 검증한다. 따라서, 클라우드 컴퓨팅 서버(100) 또는 유저 단말장치(200)가 과금을 부인하여 과금 검증을 요청할 경우, 과금 검증을 거쳐 과금 사실을 증명할 수 있게 된다.

[0036] 도 3에 도시된 과금 시스템의 동작을 설명한다.

[0037] 우선, 유저 단말장치(200)는 인터넷망(700)을 통해 클라우드 컴퓨팅 서버(100)에 접속하여 필요한 자원을 요청한다. 그리고, 유저 단말장치(200)는 자신의 과금 연산을 위임받은 위임 서버의 존재 유무와, 자신과 클라우드 컴퓨팅 서버(100) 간의 상호 인증 여부를 확인한다.

[0038] 만약, 위임 서버가 존재하지 않거나, 클라우드 컴퓨팅 서버와 상호 인증되어 있지 않은 경우, 위임 서버를 등록하고 클라우드 컴퓨팅 서버와 상호 인증하기 위한 절차를 거쳐야 한다. 위임 서버의 등록 절차, 및 클라우드 컴퓨팅 서버와 유저 단말장치 간의 상호 인증에 대한 구체적인 방법은 후술한다.

[0039] 한편, 위임 서버가 존재하거나 클라우드 컴퓨팅 서버와 상호 인증되어 있다면, 클라우드 컴퓨팅 서버(100)는 유저 단말장치(200)로부터 요청된 자원의 과금에 대한 제1 마이크로 계약서를 생성한다. 그리고, 이 제1 마이크로 계약서를 유저 단말장치(200)에 전송한다.

[0040] 유저 단말장치(200)는 제1 마이크로 계약서가 수신되면, 유저 단말장치(200)에서 사용한 자원의 과금에 대한 제2 마이크로 계약서를 생성한다. 그리고, 제1 마이크로 계약서와 제2 마이크로 계약서를 암호화시켜 위임 서버(500)에 전송한다. 이 경우, 제1 마이크로 계약서와 제2 마이크로 계약서의 암호화는 제1 및 제2 공유키를 이용하여 이루어질 수 있다. 제1 공유키는 유저 단말장치(200)와 위임 서버(500)만이 인식 가능한 것이며, 제2 공유키는 유저 단말장치(200)와 공증 서버(600)만이 인식 가능한 것이다.

[0041] 유저 단말장치(200)는 암호화된 제1 마이크로 계약서와 제2 마이크로 계약서를 위임 서버(500)에 전송한다. 위임 서버(600)는 암호화된 제1 마이크로 계약서와 제2 마이크로 계약서를 수신하여 과금을 연산하며, 클라우드 컴퓨팅 서버(100)에 자원 할당 신호를 전송하여 유저 단말장치(200)에 해당 자원을 할당하도록 한다.

[0042] 또한, 위임 서버(600)는 암호화된 제1 마이크로 계약서와 제2 마이크로 계약서를 공증 서버(600)에 전송한다.

[0043] 공증 서버(600)는 암호화된 제1 마이크로 계약서와 제2 마이크로 계약서를 전달받아 저장한다. 그리고, 추후 클라우드 컴퓨팅 서버(100) 또는 유저 단말장치(200)로부터 과금 부인에 따른 과금 요청이 있을 경우, 제1 마이크로 계약서와 제2 마이크로 계약서를 복호화하여 비교함으로써 클라우드 컴퓨팅 서버(100)가 할당한 자원의 과금과, 유저 단말장치(200)가 할당받은 자원에 대한 과금을 검증할 수 있게 된다.

[0044] 도 4는 본 발명의 일 실시 예에 따른 공개 키 기반의 프로토콜을 이용한 과금 방법을 설명하기 위한 흐름도이다. 도 4는 도 3에 도시된 공개 키 기반의 프로토콜을 이용한 과금 시스템의 과금 방법을 설명하기 위한 것이다. 도 4에 도시된 과금 방법을 설명함에 있어서, 특정 단계에 이용되는 프로토콜은 도 6 내지 도 8을 이용하여 설명한다. 이 경우, 도 6 내지 도 8에 표시된 각 기호들에 대한 설명은 도 5에 기재되어 있다.

[0045] 우선, 유저 단말장치(200)가 인터넷망(700)을 통해 클라우드 컴퓨팅 서버(100)에 접속하여 소정의 자원 사용을 요청하면(710 단계), 유저 단말장치(200)는 자원 요청과 동시에, 과금 연산을 위임받은 위임 서버가 존재하는지를 확인한다(715 단계). 확인 결과, 위임 서버가 존재하지 않을 경우 유저 단말장치(200)는 과금 연산을 위임하기 위한 위임 서버(500)를 등록한다(720 단계). 이 경우, 위임 서버(500)의 등록은 도 6에 도시된 제1 프로토콜을 이용하여 이루어질 수 있다.

[0046] 도 6에 도시된 제1 프로토콜을 참조하면, 메시지 1-1에서와 같이 유저 단말장치(200)는 특정 기간 동안 과금 결제에 대한 공증을 요청하기 위해, 유저 단말장치(200)와 공증 서버(600)만이 인식할 수 있는 공유키($K_{Notary,User}$)를 생성하고(①), 위임 요청 메시지(Delegation-Request)를 생성한다(②). 그리고, 생성된 위임 요청 메시지($E\{PU_{User,Delegation}$, Delegation-Request$\}$)를 위임 서버(500)에 전송한다(③).

[0047] 이후, 메시지 1-2에서와 같이, 위임 서버(500)는 복호화된 공유키($K_{User,Delegation}$)와, 유저 단말장치(200)로부터 수신한 위임 요청 메시지(Delegation-Request) 내에 있는 공유키($K_{User,Delegation}$)가 일치하면 유저 단말장치(200)의 인증 작업을 완료한다(①).

[0048] 위임 서버(500)는 유저 단말장치(200)의 과금 연산을 위임받기 위한 Key pair($PU_{User,Delegation}$, $PR_{User,Delegation}$)를 생성한다(②). 그리고, 이 Key pair 중 유저 단말장치(200)로부터 서명을 받기 위한 공개키($PU_{User,Delegation}$)를 위임 요청 메시지(Delegation Request) 내에 있는 공유키($K_{User,Delegation}$)로 암호화한 후, 무작위 데이터($Nonce_{USER}$)와 함께 유저 단말장치(200)에 전송한다(③).

[0049] 다음, 메시지 1-3에서와 같이, 유저 단말장치(200)는 위임 서버(500)로부터 수신한 메시지 1-2에서 무작위 데이터($Nonce_{User}$)를 이용하여 위임 서버(500)의 인증 작업을 완료한다(①). 그리고, 메시지 1-2에 포함된 공개키($PU_{User,Delegation}$), 위임 기한 및 위임 권한과, 위임 서버(500)로부터 수신한 메시지 내에 있는 공개키($PU_{User,Delegation}$)를 유저 단말장치(200)의 개인키(PR_{User})를 이용하여 서명한 후, 대행 인증서($CR_{Delegation,User}$)를 생성한다(②).

[0050] 유저 단말장치(200)는 이 대행 인증서($CR_{Delegation,User}$)를 공유키($K_{User,Delegation}$)로 암호화하여 위임 서버(500)로 전송한다(③). 이 같은 과정에 의해 위임 서버(500)는 위임 기간 동안 유저 단말장치(200)의 과금 연산에 대한 권한을 위임받을 수 있게 된다.

[0051] 한편, 메시지 1-4에서와 같이, 위임 서버(500)는 위임 요청 메시지(Delegation-Request) 내에 있는 $E\{PU_{Notary}$, $K_{Notary,User}\}$와, 유저 단말장치(200)로부터 수신한 대행 인증서($CR_{Delegation,User}$)를 위임 서버(500)의 개인키($PR_{Delegation}$)로 서명한다(①). 그리고, 암호화된 메시지와 공유키($K_{Delegation,Notary}$), 대행 인증서($CR_{Delegation,User}$) 및 무작위 데이터($Nonce_{Notary}$) 등을 공증 서버(600)의 공개키(PU_{Notary})로 암호화한다(②). 이렇게 생성된 메시지를 공증 서버(600)로 전송한다(③).

[0052] 이후, 메시지 1-5에서와 같이, 공증 서버(600)는 위임 서버(500)로부터 수신한 메시지 1-4를 개인키(PR_{Notary})을 이용하여 복호화한다(①). 그리고 복호화된 메시지를 위임 서버(500)의 공개키($PU_{Delegation}$)를 이용하여 복호화하고, 대행 인증서($CR_{Delegation,User}$)를 비교하여 위임 서버(500)에 대한 인증 작업을 완료한다(②).

[0053] 또한, 위임 서버(500)로부터 수신한 대행 인증서($CR_{Delegation,User}$)를 이용하여 유저 단말장치(200)에 대한 인증 작업을 완료한다(③). 그리고, 공증 서버(600)는 개인키(PR_{Notary})를 이용하여 $E\{PU_{Notary}$, $K_{Notary,User} \| ID_{User} \| ID_{Delegation}\}$를 복호화한 후, 유저 단말장치(200)에 시퀀스 넘버(Sequence Number)를 부여한다(④). 그리고, 시퀀스 넘버(Sequence Number) 및 무작위 데이터($Nonce_{Notary}$)를 공개키($K_{Delegation,Notary}$)로 암호화한 후, 위임 서버(500)에 전송하여(⑤) 유저 단말장치(200)에 대한 과금 공증 서비스를 제공하기 위한 준비를 완료한다(⑥).

[0054] 한편, 도 6에 나타낸 제1 프로토콜을 이용하여 위임 서버의 등록이 완료되거나, 715 단계의 확인 결과 기 등록된 위임 서버가 존재할 경우, 유저 단말장치(200)는 클라우드 컴퓨팅 서버와 상호 인증되어 있는지를 확인한다

(725 단계).

[0055] 만약, 확인 결과, 클라우드 컴퓨팅 서버와 상호 인증되어 있지 않을 경우 유저 단말장치(200)는 클라우드 컴퓨팅 서버와의 상호 인증 작업을 수행한다(730 단계).이 경우, 상호 인증 작업은 도 7에 도시된 제2 프로토콜을 이용하여 이루어질 수 있다.

[0056] 도 7에 도시된 제2 프로토콜을 참조하면, 메시지 2-1에서와 같이 클라우드 컴퓨팅 서버(100)는 유저 단말장치(200)와 클라우드 컴퓨팅 서버(100) 간의 상호 인증을 위해, 인증 캡슐(Authentication-Capsule)을 유저 단말장치(200)에 전송한다(①).

[0057] 이후, 메시지 2-2에서와 같이, 유저 단말장치(200)는 클라우드 컴퓨팅 서버(100)로부터 수신한 인증 캡슐(Authentication-Capsule)과, 유저 단말장치(200)의 아이디(ID_{User}) 및 세션-공유키(Session-$K_{User,Cloud}$)를 공유키($K_{Notary,User}$) 및 공유키($K_{User,Delegation}$)로 암호화한다(①). 이렇게 암호화된 메시지를 유저 단말장치(200)의 아이디(ID_{User})와 함께 위임 서버(500)로 전송한다(②).

[0058] 다음, 메시지 2-3에서와 같이, 위임 서버(500)는 유저 단말장치(200)로부터 수신한 메시지 중 E{$K_{Notary,User}$, Authentication-Capsule}을 개인키($PR_{User,Delegation}$)로 암호화한다(①). 그리고, 클라우드 컴퓨팅 서버(100)의 아이디(ID_{Cloud})와, 도 6의 메시지 1-5에 의해 생성되었던 시퀀스 넘버(Seq_{User}) 및 무작위 데이터($Nonce_{Notary}$)와, '①'에서 생성된 내용을 공유키($K_{Delegation,Notary}$)로 암호화하고, 이를 공증 서버(600)에 전송한다(②).

[0059] 한편, 메시지 2-4에서와 같이, 공증 서버(600)는 위임 서버(500)로부터 수신한 메시지 2-4를 공유키($K_{Delegation,Notary}$)를 이용하여 복호화한다(①). 그리고, 복호화된 메시지를 클라우드 컴퓨팅 서버(100)의 아이디(ID_{Cloud}) 및 시퀀스 넘버(Seq_{User})에 따라 분류하여 인증 캡슐(Authentication-Capsule)과 E{$PR_{User,Delegation}$, E{$K_{Notary,User}$, Authentication-Capsule}}를 대행 인증서($CR_{User,Delegation}$)에 저장되어 있는 공유키($PU_{User,Delegation}$)로 복호화한다(②). 그리고, 공유키($K_{Notary,User}$)를 이용하여 E{$K_{Notary,User}$, Authentication-Capsule}을 복호화한다(③).

[0060] 또한, 복호화된 인증 캡슐(Authentication-Capsule)을 비교하여, 동일한 경우에는 위임 서버(500)에 "허락(ACCEPT)" 메시지를 보내고, 상이한 경우에는 "거절(REJECT)" 메시지를 보낸다. "거절(REJECT)" 메시지를 보내는 경우, "거절(REJECT)" 메시지를 공증 서버(600)와 클라우드 컴퓨팅 서버(100) 간의 공유키($K_{Cloud,Notary}$)를 공개키(PU_{Cloud})로 암호화하고 무작위 데이터($Nonce_{Notary}$)와 함께 위임 서버(500)에 전송한다(④).

[0061] 한편, 메시지 2-5에서와 같이, 위임 서버(500)는 세션-공유키(Session- $K_{User,Cloud}$), 공유키($K_{Delegation,Cloud}$), 인증 캡슐(Authentication-Capsule) 및 유저 단말장치(200)의 아이디(ID_{User})를 공개키(PU_{Cloud})와 공유키($K_{Delegation,Cloud}$)로 암호화하여 이와 함께 공증 서버(600)로부터 수신한 공유키($K_{Cloud,Notary}$) 및 무작위 데이터($Nonce_{Notary}$)를 클라우드 컴퓨팅 서버(100)에 전송한다(①).

[0062] 이후, 메시지 2-6에서와 같이, 클라우드 컴퓨팅 서버(100)는 위임 서버(500)로부터 수신한 메시지 2-5를 복호화하여 인증 캡슐(Authentication-Capsule)을 확인하고, 무작위 데이터($Nonce_{Cloud}$)를 공개키($PU_{Delegation,User}$)로 암호화하여 유저 단말장치(200)에 전송한다(①). 그리고, 클라우드 컴퓨팅 서버(100)는 유저 단말장치(200)로부터 별도의 메시지가 수신될 때까지 대기 모드로 동작한다(②).

[0063] 다음, 메시지 2-8에서와 같이, 유저 단말장치(200)는 세션-공유키(Session-Key$_{User,Cloud}$)를 이용하여 인증 캡슐(Authentication-Capsule)을 암호화하여(①), 유저 단말장치(200)와 클라우드 컴퓨팅 서버(100) 간의 상호 인증을 완료한다(②).

[0064] 한편, 도 7에 나타낸 제2 프로토콜을 이용하여 유저 단말장치(200)와 클라우드 컴퓨팅 서버(100) 간의 상호 인증이 완료되거나, 715 단계의 확인 결과 이미 상호 인증이 완료되어 있을 경우, 740 단계 내지 760 단계의 과정을 통해 유저 단말장치(200)에 할당된 자원에 대한 과금 연산을 한다.

[0065] 구체적으로, 클라우드 컴퓨팅 서버(100)는 유저 단말장치(200)에 할당한 자원의 과금에 대한 제1 마이크로 계약서를 생성하여 유저 단말장치(200)에 전송한다(740 단계). 그리고, 유저 단말장치(200)는 제1 마이크로 계약서를 수신하고, 클라우드 컴퓨팅 서버(100)에서 할당받은 자원의 과금에 대한 제2 마이크로 계약서를 생성하여 제1 마이크로 계약서와 함께 암호화한 후 위임 서버(500)에 전송한다(745 단계).

[0066] 이후, 위임 서버(500)는 제1 마이크로 계약서 및 제2 마이크로 계약서를 이용하여 과금 연산을 수행한다(750 단계). 그리고, 위임 서버(500)는 클라우드 컴퓨팅 서버(100)에 자원 할당 신호를 전송하여(755 단계), 유저 단말장치(200)에 자원이 할당될 수 있도록 한다.

[0067] 또한, 위임 서버(500)는 공증 서버(600)에 제1 마이크로 계약서 및 제2 마이크로 계약서를 전송하여 저장될 수 있도록 한다(760 단계). 이에 따라, 공증 서버(600)는 추후, 클라우드 컴퓨팅 서버(100) 또는 유저 단말장치(200)가 과금을 부인하여 과금 검증을 요청하는 경우, 저장된 제1 마이크로 계약서 및 제2 마이크로 계약서를 비교 분석하여 과금을 검증할 수 있게 된다.

[0068] 그리고, 클라우드 컴퓨팅 서버(100)에 추가의 자원 요청이 있다면(765 단계), 740 단계 내지 760 단계의 과정을 반복한다.

[0069] 한편, 740 단계 내지 760 단계에 도시 및 설명된 과금 방법은 도 8에 도시된 제3 프로토콜을 이용하여 이루어질 수 있다. 구체적으로, 도 8에 도시된 제3 프로토콜을 참조하면, 메시지 3-1에서와 같이, 유저 단말장치(200)는 세션-공유키(Session-Key$_{User,Cloud}$)를 이용하여 필요한 자원 요청 메시지를 무작위 데이터(Nonce$_{User}$)와 함께 클라우드 컴퓨팅 서버(100)에 전송한다(①).

[0070] 이후, 메시지 3-2에서와 같이, 클라우드 컴퓨팅 서버(100)는 유저 단말장치(200)로부터 수신한 메시지 3-1을 수신하고, 클라우드 컴퓨팅 서버에서 할당한 자원 정보와, 추후 과금 연산 검증을 위한 제1 마이크로 계약서(micro-Contract-Cloud)를 생성하여 유저 단말장치(200)에 전송한다(①).

[0071] 다음, 메시지 3-3을 참조하면, 유저 단말장치(200)는 자신의 아이디(ID$_{User}$)와, 클라우드 컴퓨팅 서버(100)로부터 수신한 제1 마이크로 계약서(micro-Contract-Cloud), 그리고 유저 단말장치(200)에서 생성된 제2 마이크로 계약서(micro-Contract-User)를 이용하여 micro-Contract를 생성한 후, micro-Contract를 공유키($K_{Notary,User}$)와 공개키($K_{User,Delegation}$)로 암호화한다(①). 그리고, 암호화 메시지를 위임 서버(500)에 전송한다(②).

[0072] 한편, 메시지 3-4에서와 같이, 위임 서버(500)는 유저 단말장치(200)로부터 수신한 메시지 3-3 중 E{$K_{Notary,User}$, micro-Contract}를 개인키($PR_{User,Delegation}$)로 암호화한다(①). 그리고, 클라우드 컴퓨팅 서버(100)의 아이디(ID_{Cloud})와, 도 6의 메시지 1-5에서 생성된 시퀀스 넘버(Seq_{User}) 및 무작위 데이터($Nonce_{Notary}$)와 '①'에서 생성된 메시지를 공유키($K_{Delegation,Notary}$)로 암호화하여 공증 서버(600)에 전송한다(②).

[0073] 이후, 메시지 3-5에서와 같이, 공증 서버(600)는 위임 서버(500)로부터 수신한 메시지 3-4를 공유키($K_{Delegation,Notary}$)로 복호화한다(①). 그리고, 공증 서버(600)는 클라우드 컴퓨팅 서버(100)의 아이디(ID_{Cloud}), 유저 단말장치(200)의 시퀀스 넘버(Seq_{User}) 따라 분류하여 micro-Contract과 E{$PR_{User,Delegation}$, E{$K_{Notary,User}$, micro-Contract}}, 대행 인증서($CR_{User,Delegation}$)에 저장되어 있는 개인키($PU_{User,Delegation}$)로 복호화하고, 공유키($K_{Notary,User}$)로 E{$K_{Notary,User}$, micro-Contract}를 복호화한다(②).

[0074] 또한, 공증 서버(600)는 micro-Contract에서 제1 마이크로 계약서(micro-Contract-Cloud)와 제2 마이크로 계약서(micro-Contract-User)의 내용을 비교하여 동일한 경우에는 위임 서버(500)에 "허락(ACCEPT)" 메시지를 보내고 상이한 경우에는 "거절(REJECT" 메시지를 무작위 데이터($Nonce_{Notary}$) 함께 전송한다(③).

[0075] 다음, 메시지 3-6에서와 같이, 위임 서버(500)는 세션 공유키($Session\text{-}Key_{User,Cloud}$), 공유키($K_{Delegation,Cloud}$), micro-Contract, 유저 단말장치(200)의 아이디(ID_{User})를 공유키($K_{Delegation,Cloud}$)로 암호화하여 클라우드 컴퓨팅 서버(100)에 전송한다(①).

[0076] 한편, 메시지 3-7에서와 같이, 클라우드 컴퓨팅 서버(100)는 위임 서버(500)로부터 수신한 메시지 3-6을 복호화하여 micro-Contract을 확인하고 무작위 데이터($Nonce_{Cloud}$)를 공유키($K_{Delegation,Cloud}$)로 암호화시켜 위임 서버(500)에 전송한다(①). 그리고, 클라우드 컴퓨팅 서버(100)는 유저 단말장치(200)로부터 메시지가 수신될 때까지 대기 모드로 동작한다(②).

[0077] 이후, 메시지 3-8에서와 같이, 유저 단말장치(200)는 세션-공유키($Session\text{-}Key_{User,Cloud}$)로 micro-Contract를 암호화하여(①), 과금 연산을 완료한다(②).

[0078] 한편, 도 8에 나타낸 제3 프로토콜을 이용하여 공증 서버(600)가 클라우드 컴퓨팅 서버(100) 및 유저 단말장치(200)의 과금 검증 및 부인 방지 기능을 제공하는 메커니즘을 설명한다. 과금 시스템은 도 6 및 도 7에 나타낸 제1 및 제2 프로토콜을 의해 유저 단말장치(200)와 위임 서버(500)만이 인식할 수 있는 공유키($K_{User,Delegation}$), 유저 단말장치(200)와 공증 서버(600)만이 인식할 수 있는 공유키($K_{User,Notary}$), 유저 단말장치(200)와 클라우드 컴퓨팅 서버(100)만이 인식할 수 있는 공유키($Session\text{-}Key_{User,Cloud}$)가 공개 키 기반으로 분배되어 있는 상태이다.

[0079] 유저 단말장치(200)는 제4 프로토콜의 메시지 3-2에서와 같이 클라우드 컴퓨팅 서버(100)로부터 제1 마이크로 계약서(micro-Contract-Cloud)를 전송받는다. 이 경우, 제1 마이크로 계약서(micro-Contract-Cloud)는 제1 내지 제3 필드로 구성되어 있다. 구체적으로, 제1 필드는 제1 마이크로 계약서(micro-Contract-Cloud)의 일련 번호(SN)를 나타내며, 제2 필드는 제1 마이크로 계약서(micro-Contract-Cloud)의 일련 번호(SN), 유저 단말장치(200)에 할당한 자원(Granted-Resource) 및 클라우드 컴퓨팅 서버(100)에서 생성된 무작위 데이터($Nonce_{Cloud}$)에 대한 해시(Hash) 값 "$Hash(SN \| Granted\text{-}Resource \| Nonce_{Cloud})$"을 나타내고, 제3 필드는 제1 마이크로 계약서(micro-Contract-Cloud)의 일련 번호(SN) 및 유저 단말장치(200)에 할당한 자원(Granted-Resource)에 대한 해시 값을 클라우드 컴퓨팅 서버(100)와 공증 서버(600)만이 인식할 수 있는 공유키($K_{Notary,Cloud}$)로 암호화한 값 "E{$K_{Notary,Cloud}$, $Hash(SN \| Granted\text{-}Resource)$}을 나타낸다. 이 경우, 해시 함수는 일 방향 함수이므로 유저 단말장치(200)는 해시 값의 원본을 알 수 없다.

[0080] 유저 단말장치(200)는 제4 프로토콜의 메시지 3-3에서와 같이 클라우드 컴퓨팅 서버(100)로부터 제1 마이크로 계약서(micro-Contract-User)를 전송받고, 제2 마이크로 계약서(micro-Contract-User)를 생성한다. 이 경우,

제2 마이크로 계약서(micro-Contract-User) 역시 제1 내지 제3 필드를 포함한다. 구체적으로, 제1 필드는 제2 마이크로 계약서(micro-Contract-User)의 일련 번호(SN)를 나타내며, 제2 필드는 제2 마이크로 계약서(micro-Contract-User)의 일련 번호(SN), 유저 단말장치(200)가 할당받은 자원(Granted-Resource) 및 유저 단말장치(200)에서 생성된 무작위 데이터($Nonce_{User}$)에 대한 해시(Hash) 값 ""$Hash(SN \| Granted\text{-}Resource \| Nonce_{User})$을 나타내고 제3 필드는 제2 마이크로 계약서(micro-Contract-User)의 일련 번호(SN) 및 유저 단말장치(200)가 할당받은 자원(Granted-Resource)에 대한 해시 값 "$Hash(SN \| Granted\text{-}Resource)$을 나타낸다.

[0081] 유저 단말장치(200) 또는 클라우드 컴퓨팅 서버(100)는 각각 예측 또는 주장하는 할당된 자원의 과금이 상이할 경우 과금 검증을 요청할 수 있다. 이 같은 과금 검증 요청이 있을 경우, 공증 서버(600)는 제1 마이크로 계약서와 제2 마이크로 계약서를 독출한 후, 동일한 일련번호(SN)에 대하여 검증을 수행한다. 단, 공증 서버(600)는 검증을 수행하기에 앞서, 클라우드 컴퓨팅 서버(100)가 유저 단말장치(200)에 할당한 것으로 예측하는 제1 예측 할당 자원(Granted-Resource)과 클라우드 컴퓨팅 서버(100)의 무작위 데이터($Nonce_{Cloud}$)를 수신하고, 유저 단말장치(200)가 클라우드 컴퓨팅 서버(100)로부터 할당받은 것으로 예측하는 제2 예측 할당 자원과 유저 단말장치(200)의 무작위 데이터($Nonce_{User}$)를 수신한다.

[0082] 그리고, 공증 서버(600)는 제1 예측 할당 자원(Granted-Resource)과 클라우드 컴퓨팅 서버(100)의 무작위 데이터($Nonce_{Cloud}$)를 제1 마이크로 계약서(micro-Contracte-Cloud)의 제2 필드 값들과 비교하여 일치되는지를 확인한다. 만약, 일치한다면 유저 단말장치(200)가 클라우드 컴퓨팅 서버(100)로부터 수신한 제1 마이크로 계약서(micro-Contrate-Cloud)를 위임 서버(500)에 전송한 것이 증명되므로, 유저 단말장치(200)는 클라우드 컴퓨팅 서버(100)에서 할당한 자원을 수락했음을 증명하게 된다.

[0083] 또한, 제2 예측 할당 자원(Granted-Resource)과 유저 단말장치(200)의 무작위 데이터($Nonce_{User}$)를 제2 마이크로 계약서(micro-Contracte-User)의 제2 필드 값들과 비교하여 일치되는지를 확인한다. 만약, 일치한다면 클라우드 컴퓨팅 서버(100)가 제1 마이크로 계약서(micro-Contracte-Cloud)를 유저 단말장치(200)에 전송한 것이 증명된다.

[0084] 이 같이 또한, 공증 서버(600)는 클라우드 컴퓨팅 서버(100)에서 생성된 제1 마이크로 계약서와 유저 단말장치(200)에서 생성된 제2 마이크로 계약서를 저장해둠으로써, 추후 과금 검증에 따른 요청이 있을 때 클라우드 컴퓨팅 서버(100)에서 제공한 자원의 과금과, 상기 유저 단말장치(200)에서 사용한 자원에 대한 과금을 검증하여 과금에 따른 부인을 방지할 수 있게 된다.

[0085] 이상에서는 본 발명의 바람직한 실시 예에 대하여 도시하고 설명하였지만 본 발명은 상술한 특정의 실시 예에 한정되지 아니하며, 청구범위에서 청구하는 본 발명의 요지를 벗어남이 없이 당해 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 다양한 변형실시가 가능한 것은 물론이고, 이러한 변형실시들은 본 발명의 기술적 사상이나 전망으로부터 개별적으로 이해되어서는 안 될 것이다.

도면의 간단한 설명

[0086] 도 1은 종래 기술에 따른 클라우드 컴퓨팅 시스템의 과금 방법을 설명하기 위한 도면,

[0087] 도 2는 종래 기술에 따른 공개 키 기반으로 한 클라우드 컴퓨팅 환경의 과금 시스템을 나타내는 도면,

[0088] 도 3은 본 발명의 일 실시 예에 따른 공개 키 기반의 프로토콜을 이용한 과금 시스템을 나타내는 도면,

[0089] 도 4는 본 발명의 일 실시 예에 따른 공개 키 기반의 프로토콜을 이용한 과금 방법을 설명하기 위한 흐름도, 그리고,

[0090] 도 5 내지 도 7은 도 4에 도시된 과금 방법에 이용되는 프로토콜을 나타내는 도면이다.

[0091] * 도면의 주요 부분에 대한 부호 설명 *

[0092] 100 : 클라우드 컴퓨팅 서버

[0093] 200 : 유저 단말장치

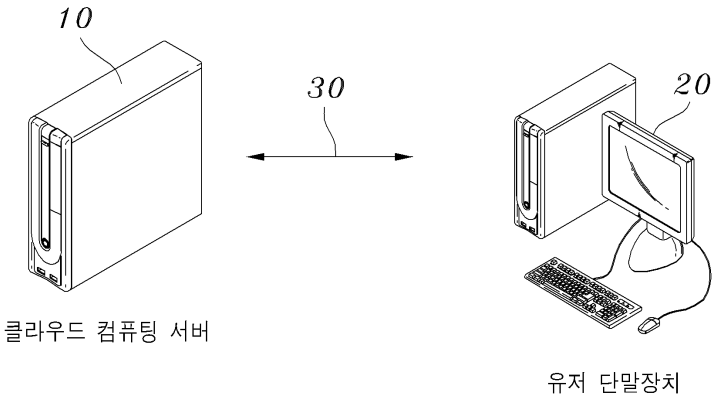
[0094] 300 : 인터넷망

[0095] 500 : 위임 서버

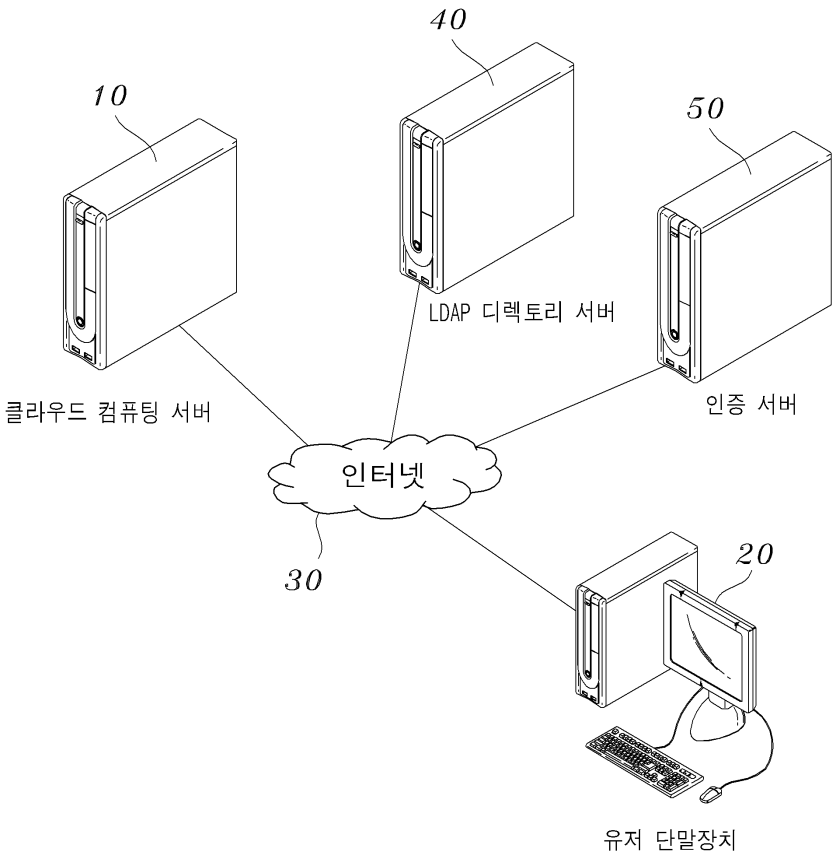
[0096] 600 : 공증 서버

도면

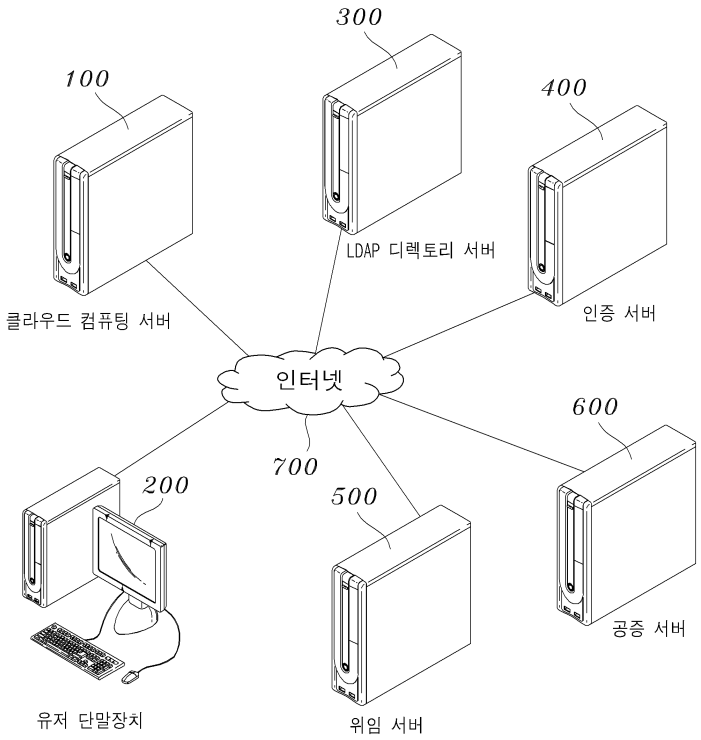
도면1



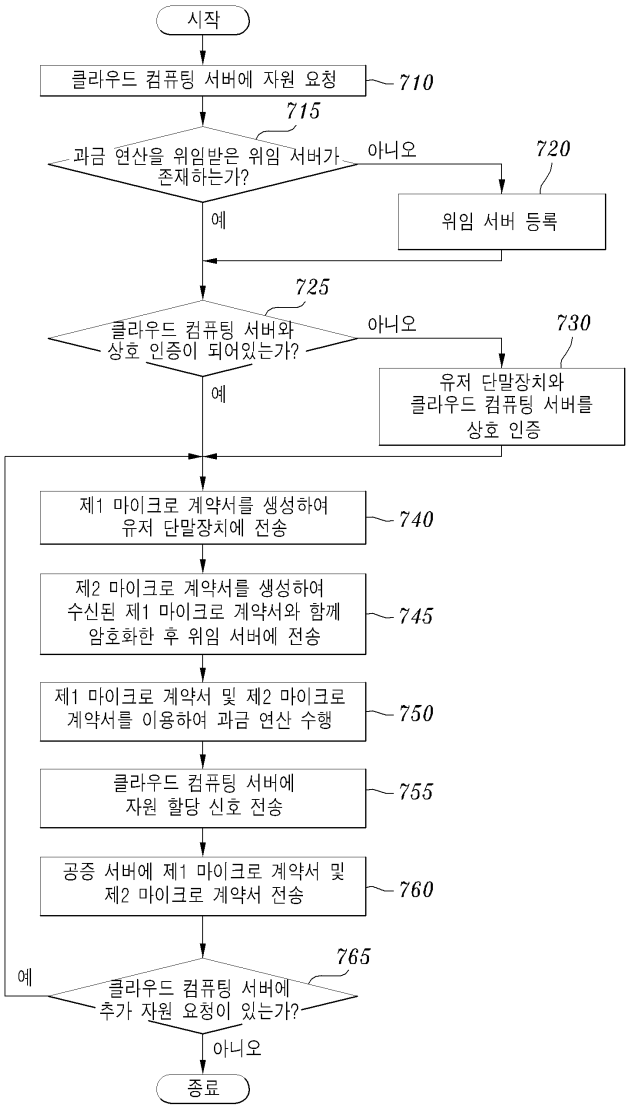
도면2



도면3


300
100
400
LDAP 디렉토리 서버
클라우드 컴퓨팅 서버
인증 서버
인터넷
600
700
200
500
공증 서버
유저 단말장치
위임 서버

도면4


시작
클라우드 컴퓨팅 서버에 자원 요청
710
715
과금 연산을 위임받은 위임 서버가 존재하는가?
아니오
720
위임 서버 등록
예
725
클라우드 컴퓨팅 서버와 상호 인증이 되어있는가?
아니오
730
유저 단말장치와 클라우드 컴퓨팅 서버를 상호 인증
예
제1 마이크로 계약서를 생성하여 유저 단말장치에 전송
740
제2 마이크로 계약서를 생성하여 수신된 제1 마이크로 계약서와 함께 암호화한 후 위임 서버에 전송
745
제1 마이크로 계약서 및 제2 마이크로 계약서를 이용하여 과금 연산 수행
750
클라우드 컴퓨팅 서버에 자원 할당 신호 전송
755
공증 서버에 제1 마이크로 계약서 및 제2 마이크로 계약서 전송
760
765
예
클라우드 컴퓨팅 서버에 추가 자원 요청이 있는가?
아니오
종료

도면5

◆ 개체 기호 정의

- 클라우드 컴퓨팅 서버 : Cloud
- 유저 단말장치 : User
- 위임 서버 : Delegation
- 공증 서버 : Notary

◆ 메시지 기호 정의

- IDx : X의 ID
- PUx : X의 공개키
- PRx : X의 개인키
- PUx,y : Y에 의해 생성된 X의 위임된 공개키
- CRx,y : Y에 의해 서명된 X의 인증서(대행 인증서)
- Noncex : Replay 공격에 대비한하여 X에 의해 생성된 무작위 데이타
- Kx,y : X와 Y사이에 공유하는 공유키(대칭키)
- SN : Serial Number

◆ 프로토콜 해석

A → B : Mssage1
Message1 = α || β

A: 송신자, B:수신자
Message1은 α 와 β 의 내용을 갖는다

도면6

• 메시지 1-1. User → Delegation : $E\{PU_{Delegation}, Delegation-Request\}$ //위임 요청 메시지
$Delegation-Request = ID_{Delegation} || ID_{User} || K_{User,Delegation} || Nonce_{User} || E\{PR_{User}, K_{User,Delegation} || E\{PU_{Notary}, K_{Notary,User} || ID_{User} || ID_{Delegation}\}\}$
• 메시지 1-2. Delegation → User : $E\{K_{User,Delegation}, PU_{User,Delegation} || Nonce_{User}\}$
• 메시지 1-3. User → Delegation : $E\{K_{User,Delegation}, CR_{Delegation,User}\}$
• 메시지 1-4. Delegation → Notary : $E\{PU_{Notary}, Notary\ Request\}$ //공증 등록 메시지
$Notary\ Request = K_{Delegation,Notary} || CR_{Delegation,User} || Nonce_{Notary} || E\{PR_{Delegation}, CR_{Delegation,User} || E\{PU_{Notary}, K_{Notary,User} || ID_{User} || ID_{Delegation}\}\}$
• 메시지 1-5. Notary → Delegation : $E\{PU_{Delegation}, Seq_{User} || Nonce_{Notary}\}$

※ (메시지 1-1) 송신자: 유저 단말장치(200), 수신자: 위임 서버(500)
① 유저 단말장치는 설정 시간동안 과금 연산에 대한 공증을 요청하기 위해 유저 단말장치와 공증 서버만이 인식할 수 있는 $K_{Notary,User}$를 생성
② 위임 요청(Delegation Request) 메시지를 생성
③ '②'에서 생성된 위임 요청 메시지($E\{PU_{Delegation}$, Delegation Request$\}$)를 위임 서버로 전송

※ (메시지 1-2) 송신자: 위임 서버(500), 수신자: 유저 단말장치(200)
① $K_{User,Delegation}$와 위임 요청 메시지 내에 있는 $K_{User,Delegation}$가 일치하면 유저 단말장치 인증 완료
② 유저 단말장치의 위임을 위한 Key pair($PU_{User,Delegation}$, $PR_{User,Delegation}$)를 생성
③ 유저 단말장치로부터 서명을 받기 위해 생성된 $PU_{User,Delegation}$을 유저 단말장치로부터 수신한 $K_{User,Delegation}$로 암호화 후 유저 단말장치에게 $Nonce_{User}$와 함께 전송

※ (메시지 1-3) 송신자: 유저 단말장치(200), 수신자: 위임 서버(500)
① 위임 서버로부터 수신한 메시지에 포함된 $Nonce_{User}$를 이용하여 Delegation 인증
② 위임 서버로부터 수신한 메시지에 포함된 $PU_{User,Delegation}$와, 대행 기간, 위임 권한을 자신의 개인키(PR_{User})를 이용하여 서명하여 대행 인증서($CR_{Delegation,User}$) 생성
③ 대행 인증서($CR_{Delegation,User}$)를 $K_{User,Delegation}$로 암호화하여 위임 서버로 전송
④ 대행 기간까지 위임 완료

※ (메시지 1-4) 송신자: 위임 서버(500), 수신자: 공증 서버(600)
① 위임 요청 메시지 중 $E\{PU_{Notary}, K_{Notary,User}\}$와, 유저 단말장치로부터 받은 대행 인증서($CR_{Delegation,User}$)를 위임 서버의 개인키($PR_{Delegation}$)로 서명
② 암호화된 메시지를 공유키($K_{Delegation,Notary}$), 대행 인증서($CR_{Delegation,User}$), 무작위 데이터($Nonce_{Notary}$)와 함께 공증 서버의 공개키로 암호화
③ '②'에서 생성된 메시지를 공증 서버로 전송

※ (메시지 1-5) 송신자 : 공증 서버(600), 수신자: 위임 서버(500)
① 위임 서버로부터 받은 메시지를 자신의 개인키로 복호화
② 복호화된 메시지를 위임 서버의 공개키를 이용하여 복호화하고 대행 인증서를 비교하여 위임 서버에 대한 인증 완료
③ 위임 서버로부터 수신한 대행 인증서를 이용하여 유저 단말장치 인증 완료
④ 공증 서버의 개인키를 이용하여 $E\{PU_{Notary}, K_{Notary,User} || ID_{User} || ID_{Delegation}\}$를 복호화 시킨 후 유저 단말장치에 대한 Sequence Number를 부여
⑤ Sequence Number, $Nonce_{Notary}$를 $K_{Delegation,Notary}$를 이용하여 암호화하여 위임 서버에 전송
⑥ 유저 단말장치에 대한 과금 공증 서비스 준비 완료

도면7

• 메시지 2-1. Cloud → User : ID_{Cloud} || Authentication-Capsule
Authentication-Capsule = SN || Hash (SN || Nonce $_{Cloud}$)
• 메시지 2-2. User → Delegation : ID_{User} || E { $K_{User,Delegation}$, ID_{User} || ID_{Cloud} || $Session\text{-}Key_{User,Cloud}$ || E { $K_{Notary,User}$, Authentication-Capsule } }
• 메시지 2-3. Delegation → Notary :
$ID_{Delegation}$ || E { $K_{Delegation,Notary}$, ID_{Cloud} || Seq_{User} || $Nonce_{Notary}$ || Authentication-Capsule || E { $PR_{User,Delegation}$, E { $K_{Notary,User}$, Authentication-Capsule } } }
• 메시지 2-4. Notary → Delegation : E { $K_{Delegation,Notary}$, ACCEPT or REJECT || E { PU_{Cloud} ,$K_{Cloud,Notary}$ || $Nonce_{Cloud}$} || $Nonce_{Notary}$}
• 메시지 2-5. Delegation → Cloud : E { PU_{Cloud} ,$K_{Delegation,Cloud}$ } || E { $K_{Delegation,Cloud}$, ID_{User} || $CR_{Delegation,User}$ || $Session\text{-}Key_{User,Cloud}$ || Authentication-Capsule || $Nonce_{Cloud-2}$ } || E { PU_{Cloud} ,$K_{Cloud,Notary}$ || $Nonce_{Cloud}$}
• 메시지 2-6. Cloud → Delegation : E { $PU_{Delegation,User}$, ID_{User} || $Nonce_{Cloud-2}$ || E { PU_{Cloud} , $Nonce_{Cloud}$} }
• 메시지 2-7. Delegation → Notary : E { $K_{Delegation,Notary}$, ID_{Cloud} || $Nonce_{Notary}$ || E { PU_{Cloud} , $Nonce_{Cloud}$} }
• 메시지 2-8. User → Cloud : E { $Session\text{-}Key_{User,Cloud}$, ID_{User} || Authentication-Capsule }

※ (메시지 2-1) 송신자: 클라우드 컴퓨팅 서버(100), 수신자: 유저 단말장치(200)

① 유저 단말장치와 클라우스 컴퓨팅 서버 간의 상호 인증을 위하여 Authentication-Capsule을 유저 단말 장치에 전송

※ (메시지 2-2) 송신자: 유저 단말장치(200), 수신자: 위임 서버(500)

① 유저 단말장치는 자신의 ID_{User}, 클라우드 컴퓨팅 서버로부터 수신한 Authentication-Capsule 및 $Session\text{-}Key_{User,Cloud}$를, $K_{Notary,User}$과 $K_{User,Delegation}$를 이용하여 암호화

② 생성된 메시지를 유저 단말장치의 ID_{User}와 함께 위임 서버로 전송

※ (메시지 2-3) 송신자: 위임 서버(500), 수신자: 공증 서버(600)

① 유저 단말장치로부터 수신된 메시지 중 E{$K_{Notary,User}$, Authentication-Capsule}를 $PR_{User,Delegation}$를 이용하여 암호화

② 클라우드 컴퓨팅 서버의 ID와 공증 서버 등록시 생성되었던 Seq_{User} 및 $Nonce_{Notary}$와 '①'에서 생성되었던 내용을, $K_{Delegation,Notary}$로 암호화

※ (메시지 2-4) 송신자: 공증 서버(600), 수신자: 위임 서버(500)

① 위임 서버로부터 수신한 메시지를 $K_{Delegation,Notary}$를 이용하여 복호화

② 공증 서버는 ID_{Cloud}, Seq_{User}에 따라 분류하여 Authentication-Capsule과 E{$PR_{User,Delegation}$, E{$K_{Notary,User}$, Authentication-Capsule}}를, 대행 인증서에 저장되어 있는 $PU_{User,Delegation}$를 이용하여 복호화

③ $K_{Notary,User}$를 이용하여 E{$K_{Notary,User}$, Authentication-Capsule}를 복호화시킨다.

④ Authentication-Capsule을 비교하여 같을 경우에는 ACCEPT 메시지를 보내고, 상이할 경우에는 REJECT 메시지를 보낸 후

추가적으로 공증 서버와 클라우드 컴퓨팅 서버 간의 $K_{Cloud,Notary}$를 PU_{Cloud}를 이용하여 암호화 시키고, $Nonce_{Notary}$와 함께 위임 서버에 전송

※ (메시지 2-5) 송신자: 위임 서버(500), 수신자: 클라우드 컴퓨팅 서버(100)

① $Session\text{-}Key_{User,Cloud}$,$K_{Delegation,Cloud}$, Authentication-Capsule, ID_{User}를 PU_{Cloue}와 $K_{Delegation,Cloud}$를 이용하여 암호화시킨 후 공증서버로부터 받은 내용($K_{Cloud,Notary}$, $Nonce_{Notary}$)을 클라우드 컴퓨팅 서버에 전송

※ (메시지 2-6) 송신자: 클라우드 컴퓨팅 서버(100), 수신자: 위임 서버(500)

① 수신한 2-5 메시지를 복호화 하여 Authentication-Capsule을 확인하고 $Nonce_{Cloud}$를 $PU_{Delegation,User}$를 이용하여 암호화시켜 전송

⑤ 클라우드 컴퓨팅 서버는 유저 단말장치로부터 메시지가 올 때 까지 대기

※ (메시지 2-7) 송신자: 위임 서버(500), 수신자: 공증 서버(600)

① 위임 서버는 클라우드 컴퓨팅 서버와 공증 서버간의 공유 키 기반의 상호 인증을 위하여 메시지 2-5를 통해 받은 $Nonce_{Notary}$를 공증 서버에 전송

※ (메시지 2-8) 송신자: 유저 단말장치(200), 수신자: 클라우드 컴퓨팅 서버(100)

① 유저 단말장치는 $Session\text{-}Key_{User,Cloud}$를 이용하여 Authentication-Capsule를 암호화

③ 상호 인증 완료

도면8

• 메시지 3-1. User → Cloud : ID_{User} || E { Session-$Key_{User,Cloud}$, Resources-Request || $Nonce_{User}$ }
• 메시지 3-2. Cloud → User : ID_{Cloud} || E { Session-$Key_{User,Cloud}$, $Nonce_{User}$ || Granted-Resources || micro-Contract-Cloud }
micro-Contract-Cloud = SN || Hash (SN || Granted-Resources || Nonce $_{Cloud}$) || E { $K_{Notary,Cloud}$, Hash (SN || Granted-Resources) }
• 메시지 3-3. User → Delegation : ID_{User} || E { $K_{User,Delegation}$, ID_{User} || ID_{Cloud} || micro-Contract || E { $K_{Notary,User}$, micro-Contract } }
micro-Contract = micro-Contract-User || micro-Contract-Cloud
micro-Contract-User = SN || Hash (SN || Granted-Resources || $Nonce_{User}$) || Hash (SN || Granted-Resources)
• 메시지 3-4. Delegation → Notary : $ID_{Delegation}$ || E { $K_{Delegation,Notary}$, ID_{Cloud} || Seq_{User} || $Nonce_{Notary}$ || micro-Contract ||
E { $PR_{User,Delegation}$, E { $K_{Notary,User}$, micro-Contract } } }
• 메시지 3-5. Notary → Delegation : E { $K_{Delegation,Notary}$, ACCEPT or REJECT || $Nonce_{Notary}$ }
• 메시지 3-6. Delegation → Cloud : ID_{Cloud} || E { $K_{Delegation,Cloud}$, ID_{User} || micro-Contract || $Nonce_{Cloud}$}
• 메시지 3-7. Cloud → Delegation : E { $K_{Delegation,Cloud}$,$Nonce_{Cloud}$ }
• 메시지 3-8. User → Cloud : ID_{User} || E { Session-$Key_{User,Cloud}$, micro-Contract }

※ (메시지 3-1) 송신자: 유저 단말장치(200), 수신자: 클라우드 컴퓨팅 서버(100)

① Session-$Key_{User,Cloud}$를 이용하여 필요한 자원 요청 메시지를 $Nonce_{User}$와 함께 클라우드 컴퓨팅 서버에 전송

※ (메시지 3-2) 송신자: 클라우드 컴퓨팅 서버(100), 수신자: 유저 단말장치(200)

① 클라우드 컴퓨팅 서버에서 할당한 자원 정보와 추후 과금 연산의 검증을 위한 제1 마이크로 계약서(micro-Contract-Cloud)를 작성하여 유저 단말장치에 전송

※ (메시지 3-3) 송신자: 유저 단말장치(200), 수신자: 위임 서버(500)

① 유저 단말장치는 자신의 ID_{User}와 함께, 클라우드 컴퓨팅 서버로부터 수신한 제1 마이크로 계약서와 유저 단말장치가 생성한 제2 마이크로 계약서(micro-Contract-User)를 이용하여 micro-Contract를 만든 후 $K_{Notary,User}$과 $K_{User,Delegation}$를 이용하여 암호화

② '①'에서 생성된 메시지를 유저 단말장치의 ID_{User}와 함께 위임 서버에 전송

※ (메시지 3-4) 송신자: 위임 서버(500), 수신자: 공증 서버(600)

① 유저 단말장치로부터 수신된 메시지 중 E{$K_{Notary,User}$, micro-Contract}를 $PR_{User,Delegation}$를 이용하여 암호화

② 클라우드 컴퓨팅 서버의 ID_{Cloud}와 공증 서버 등록시 생성되었던 Seq_{User}, $Nonce_{Notary}$와 '①'번에서 생성된 메시지를 $K_{Delegation,Notary}$로 암호화하여 공증 서버에 전송

※ (메시지 3-5) 송신자: 공증 서버(600), 수신자: 위임 서버(500)

① 위임 서버로부터 수신된 메시지를 $K_{Delegation,Notary}$를 이용하여 복호화

② 공증 서버는 ID_{Cloud}, Seq_{User}에 따라 분류하여 micro-Contract과 E{$PR_{User,Delegation}$, E{$K_{Notary,User}$, micro-Contract}}를, 대행 인증서에 저장되어 있는 $PU_{User,Delegation}$를 이용하여 복호화시키고 $K_{Notary,User}$를 이용하여 E{$K_{Notary,User}$, micro-Contract}를 복호화

③ micro-Contract에서 제1 마이크로 계약서와 제2 마이크로 계약서의 내용을 비교하여 같을 경우에는 ACCEPT 메시지를 보내고 그러지 않을 경우에는 REJECT 메시지를 $Nonce_{Notary}$와 함께 위임 서버에 전송

※ (메시지 3-6) 송신자: 위임 서버(500), 수신자: 클라우드 컴퓨팅 서버(100)

① Session-$Key_{User,Cloud}$,$K_{Delegation,Cloud}$, micro-Contract, ID_{User}를 $K_{Delegation,Cloud}$를 이용하여 암호화하여 클라우드 컴퓨팅 서버에 전송

※ (메시지 3-7) 송신자: 클라우드 컴퓨팅 서버(100), 수신자: 위임 서버(500)

① 수신한 메시지 3-5를 복호화하여 micro-Contract을 확인하고 $Nonce_{Cloud}$를 $K_{Delegation,Cloud}$를 이용하여 암호화시켜 위임 서버에 전송

⑤ 클라우드 컴퓨팅 서버는 유저 단말장치로부터 메시지가 올 때 까지 대기

※ (메시지 3-8) 송신자: 유저 단말장치(200), 수신자: 클라우드 컴퓨팅 서버(100)

① 유저 단말장치는 Session-$Key_{User,Cloud}$를 이용하여 micro-Contract를 암호화

③ 과금 연산 완료