(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2016년04월04일
(11) 등록번호 10-1605766
(24) 등록일자 2016년03월17일

(51) 국제특허분류(Int. Cl.)
H04L 9/08 (2006.01)
(52) CPC특허분류
H04L 9/0816 (2013.01)
H04L 9/0869 (2013.01)
(21) 출원번호 10-2015-0009454
(22) 출원일자 2015년01월20일
심사청구일자 2015년01월20일
(56) 선행기술조사문헌
KR100789354 B1
KR1020040097016 A

(73) 특허권자
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
박경수
충남 아산시 둔포면 충무로 1803, 103동 407호 (경희주택)
엄지은
서울 서대문구 가재울로 45, 102동 102호 (남가좌동, 현대아파트)
이동훈
서울 종로구 사직로8길 34, 1404호 (내수동, 경희궁의아침3단지)
(74) 대리인
김등용, 김홍석, 이강녕

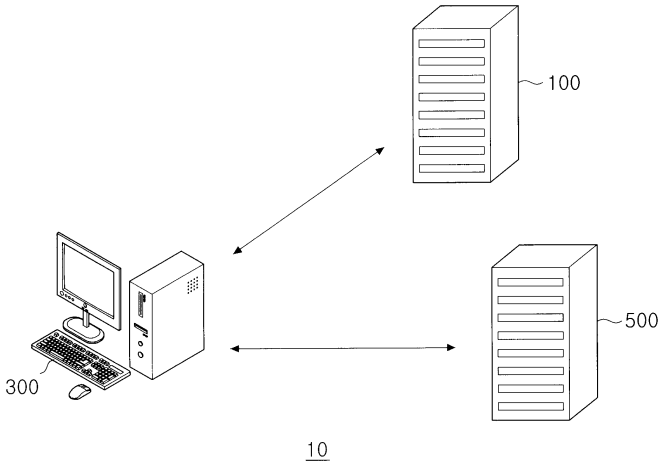
전체 청구항 수 : 총 9 항 심사관 : 문형섭

(54) 발명의 명칭 **비밀키 생성 방법 및 중복 제거 방법**

(57) 요 약

중복 제거 방법이 개시된다. 상기 중복 제거 방법은, 서버가 단말기로부터 태그와 제1 질의값을 수신하는 단계, 상기 서버가 제1 응답값과 제2 질의값을 상기 단말기로 송신하는 단계, 상기 서버가 상기 단말기로부터 제2 응답값과 식별자를 수신하는 단계, 상기 서버가 상기 제2 응답값을 검증하는 단계, 및 상기 서버가 상기 식별자를 저장하는 단계를 포함한다.

대 표 도 - 도1



(52) CPC특허분류
H04L 9/3249 (2013.01)

이 발명을 지원한 국가연구개발사업
과제고유번호 1711014861
부처명 미래창조과학부
연구관리전문기관 한국연구재단
연구사업명 도약연구지원사업(전략)
연구과제명 속성 기반 공개키 시스템에 관한 연구
기 여 율 1/1
주관기관 고려대학교
연구기간 2014.09.01 ~ 2015.08.31

명 세 서

청구범위

청구항 1

공개키($pk_{KS} = (e, N)$)와 비밀키($sk_{KS} = (N, d)$)가 설정된 RSA(Rivest Shamir Adleman) 블라인드 서명 기법을 이용한 비밀키 생성 방법에 있어서,

단말기가 데이터(f)를 입력으로 하는 해쉬 함수의 결과값($h = H(f)$)과 난수($r \in Z_N$)를 이용하여 생성된 질의값(x)을 제1 서버로 송신하는 단계;

상기 단말기가 상기 제1 서버로부터 응답값(y)을 수신하는 단계;

상기 단말기가 상기 응답값(y)을 검증하는 단계; 및

상기 단말기가 상기 데이터(f)에 대한 비밀키(sk_f)를 생성하는 단계를 포함하고,

상기 응답값(y)을 검증하는 단계는,

제2 수학식을 이용하여 연산값(z)을 계산하는 단계; 및

제3 수학식의 성립 여부를 판단하는 단계를 포함하고,

상기 제2 수학식은 $z = y \cdot r^{-1} \bmod N$ 이고,

상기 제3 수학식은 $z^e = h \bmod N$ 인,

비밀키 생성 방법.

청구항 2

제1항에 있어서,

상기 응답값(y)은 제1 수학식을 이용하여 생성되고,

상기 제1 수학식은 $y = x^d \bmod N$ 인, 비밀키 생성 방법.

청구항 3

삭제

청구항 4

제1항에 있어서,

상기 비밀키(sk_f)는 상기 연산값(z)을 입력으로 하는 해쉬 함수의 결과값인, 비밀키 생성 방법.

청구항 5

서버가 단말기로부터 태그(T)와 제1 질의값(s)을 수신하는 단계;

상기 서버가 제1 응답값(h)과 제2 질의값(k)을 상기 단말기로 송신하는 단계;

상기 서버가 상기 단말기로부터 제2 응답값(h')과 식별자(ID_c)를 수신하는 단계;

상기 서버가 상기 제2 응답값(h')을 검증하는 단계; 및

상기 서버가 상기 식별자(ID_c)를 저장하는 단계를 포함하고,

상기 태그(T)는 데이터(f)에 대한 비밀키(sk_f)를 입력으로 하는 해쉬 함수의 결과값이고,

상기 비밀키(sk_f)는 제1항에 기재된 비밀키 생성 방법에 의해 생성된,

중복 제거 방법.

청구항 6

삭제

청구항 7

제5항에 있어서,

상기 제1 질의값(s)은 상기 단말기에 의해 선택된 임의의 난수이고,

상기 제1 응답값(h)은 상기 서버에 저장되어 있는 암호문(C')과 상기 제1 질의값(s)을 입력으로 하는 해쉬 함수의 결과값이고,

상기 제2 질의값(k)은 상기 서버에 의해 선택된 임의의 난수인, 중복 제거 방법.

청구항 8

제5항에 있어서,

상기 단말기에 의해 생성되는 상기 제2 응답값(h')은 상기 단말기에 저장되어 있는 암호문(C)과 상기 제2 질의값(k)을 입력으로 하는 해쉬 함수의 결과값인, 중복 제거 방법.

청구항 9

제5항에 있어서,

상기 제2 응답값(h')을 검증하는 단계는,

상기 서버에 저장되어 있는 암호문(C')과 상기 제2 질의값(k)을 입력으로 하는 해쉬 함수의 결과값과 상기 단말기로부터 수신된 제2 응답값(h')을 비교하여 검증하는, 중복 제거 방법.

청구항 10

제5항에 있어서,

상기 중복 제거 방법은,

상기 서버가 상기 단말기로부터 데이터 다운로드 요청 메시지를 수신하는 단계;

상기 서버가 상기 단말기의 데이터 소유권을 검사하는 단계; 및

상기 서버가 상기 데이터 다운로드 요청 메시지에 대응하는 암호문과 암호화된 비밀키를 상기 단말기로 송신하는 단계를 더 포함하는, 중복 제거 방법.

청구항 11

삭제

청구항 12

제5항에 있어서,

상기 제2 응답값(h')과 상기 식별자(ID_c)를 수신하는 단계는 상기 단말기의 개인키(sk_c)를 이용하여 암호화된 비밀키($K_c = Enc_{sk_c}(sk_f)$)를 더 수신하고,

상기 식별자(ID_c)를 저장하는 단계는 상기 암호화된 비밀키(K_c)를 상기 식별자(ID_c)와 함께 저장하는, 중복 제거 방법.

발명의 설명

기 술 분 야

[0001] 본 발명의 개념에 따른 실시 예는 비밀키 생성 방법 및 중복 제거 방법에 관한 것으로, 특히 키 생성 서버와 클라이언트 단말기 간의 통신을 통하여 안전한 비밀키를 생성하고, 생성된 비밀키를 이용하여 데이터 중복 제거를 수행할 수 있는 비밀키 생성 방법 및 중복 제거 방법에 관한 것이다.

배 경 기 술

[0002] 빅 데이터(Big data) 시대를 맞아 클라이언트들은 클라우드 스토리지 서비스(Cloud Storage Service)를 이용하여 대용량의 데이터를 효율적으로 관리한다. 중복 제거 기술(deduplication)은 효율성 측면에서 클라우드 스토리지 환경에 가장 큰 영향을 미치고 있는 기술로 동일한 데이터를 하나만 저장하여 중복 저장을 방지하는 기술이다. 데이터와 연관성을 가지는 작은 태그(Tag) 값을 이용하여 클라우드 서버는 원본 데이터 하나만을 스토리지에 저장하고, 동일한 데이터에 대해서는 태그 값과 함께 클라이언트의 포인터(Pointer)를 모아 저장한다. 이를 통해 클라우드 서버는 90% 이상의 저장 공간을 확보할 수 있으며, 통신 대역폭 효율성을 얻을 수 있다. 따라서, 중복 제거 기술은 현재 대부분의 클라우드 스토리지 기술에 사용되고 있다.

[0003] 클라이언트 사이드 중복 제거 기술(client-side deduplication)은 데이터를 전송하기 전에 클라이언트가 작은 태그 값을 생성하고, 질의-응답(Challenge-Response) 메커니즘을 통해 데이터가 클라우드 서버에 저장되어 있는지 여부를 먼저 확인하는 기법이다. 만약, 클라우드 서버에 데이터가 이미 저장되어 있다면, 데이터 전송은 일어나지 않고 클라우드 서버는 기존에 저장되어 있는 데이터에 클라이언트의 포인터를 저장한다. 따라서, 클라이언트 사이드 중복 제거 기법은 클라우드 서버의 저장 공간 이용뿐만 아니라 통신 대역폭의 효율성도 얻을 수 있는 기법이다.

[0004] 2013년에 Bellare 등은 엔트로피가 낮거나 작은 도메인을 가지는 데이터들에 대한 전수 조사 공격을 막기 위해 키 서버(key server)를 두고 RSA(Rivest Shamir Adleman) 블라인드 서명(Blind Signature) 기법을 이용해 안전하게 데이터의 비밀키를 생성하는 서버 사이드 중복 제거 기술을 제안하였다(M. Bellare, and S. Keelveedhi. : Server-aided encryption for deduplicated storage., USENIX Security Symposium 2013.).

[0005] 2011년에 Halevi 등은 실제 데이터에 대한 접근 권한이 없는 공격자로부터 클라우드 서버에 저장된 데이터의 정보 유출을 막기 위한 데이터 소유권 증명(Proofs of Ownership) 기법을 제안하였다(S. Halevi, D. Harnik, B. Pinkas, A. Shulman-Peleg, of Ownership in Remote Storage Systems, Proc. 18th ACM. Conf. Computer and Communications Security (CCS), ACM Press, 2011, pp. 491-500.)

[0006] 클라이언트 사이드 중복 제거 기법에 요구되는 조건으로 다음과 같은 4 가지를 조건을 제시할 수 있다.

[0007] 첫째, 데이터 프라이버시(Data privacy)가 보장되어야 한다. 클라이언트 사이드 중복 제거 기술은 클라이언트가 태그 값을 이용해 클라우드 서버로부터 다운로드 권한을 부여 받기 때문에 권한이 없는 클라이언트가 데이터를 다운로드하거나, 저장되어 있는 데이터가 암호화 되어 있지 않다면 데이터에 대한 정보가 노출되는 프라이버시 문제가 발생한다. 따라서, 공격자는 클라이언트와 클라우드 서버 간의 통신으로부터 받는 정보를 통해 저장된 데이터에 대한 어떠한 정보도 얻을 수 없어야 하고, 클라우드 서버 역시 저장된 데이터를 통해 데이터와 클라이언트의 어떠한 정보도 얻을 수 없어야 한다.

[0008] 둘째, 전수 조사 공격(Brute-force Attack)에 대한 안전성을 제공해야 한다. 구체적으로, 대부분의 클라이언트 사이드 중복 제거 기술의 경우, 데이터를 해쉬한 값을 비밀키로 이용하여 데이터를 암호화하는데, 이는 공격자가 엔트로피가 낮거나 작은 도메인을 가지는 데이터들의 전수 조사 공격을 통해 데이터에 대한 비밀키를 생성할 수 있는 문제가 발생한다.

[0009] 셋째, 온라인 추측 공격(Online-guessing Attack)에 대한 안전성을 제공해야 한다. 구체적으로, 클라이언트 사이드 중복 제거 기술은 클라우드 서버에 데이터의 저장 여부를 확인하는 태그가 크기가 작고 데이터로부터 결정적으로 생성되기 때문에 공격자가 클라이언트들이 저장한 데이터의 내용을 추측하여 알아낼 수 있는 문제점이 발생한다. 공격자가 엔트로피가 낮거나 작은 도메인을 가지는 목표 데이터에 대해, 데이터의 정보를 조금씩 변화시키면서 생성한 태그를 이용하여 클라우드 서버에 데이터가 저장되어 있는지의 여부를 알 수 있는데 이를 온라인 추측 공격이라 한다.

[0010] 넷째, 포이즌 공격(Poison Attack)에 대한 안전성을 제공해야 한다. 구체적으로, 클라이언트 사이드 중복 제거 기술의 경우 클라이언트는 태그 값을 이용하여 데이터가 클라우드 서버에 저장되어 있는지 여부를 확인한다. 만약 클라이언트가 클라우드 서버에 저장되어 있는 데이터에 대해 무결성(Integrity)을 확인할 수 있는 방법이 없다면, 악의적인 클라이언트에 의해 변형된 데이터가 저장되어 있을 때 이후 데이터 다운로드 시 원본 데이터를 잃어버리게 되는 문제가 발생하는데 이를 포이즌 공격이라 한다.

[0011] 그러나, 지금까지 제안된 클라이언트 사이드 중복 제거 기술에서는 상기 네 가지의 공격에 대해 모두 안전성을 제공하는 기법은 없다. 따라서, 본 발명은 상기 네 가지의 요구 조건을 만족시킬 수 있는 클라이언트 사이드 중복 제거 기법을 제안한다.

발명의 내용

해결하려는 과제

[0012] 본 발명이 이루고자 하는 기술적인 과제는 키 생성 서버로부터 수신된 비밀값을 이용해 비밀키를 생성하고, 클라이언트 사이드 중복 제거 기술에서 발생할 수 있는 다양한 공격에 대해 안전성을 제공할 수 있는 비밀키 생성 방법 및 중복 제거 방법을 제공하는 것이다.

과제의 해결 수단

[0013] 본 발명의 실시 예에 따른 비밀키 생성 방법은, 공개키와 비밀키가 설정된 RSA(Rivest Shamir Adleman) 블라인드 서명 기법을 이용한 비밀키 생성 방법으로서, 단말기가 데이터를 입력으로 하는 해쉬 함수의 결과값과 난수를 이용하여 생성된 질의값을 제1 서버로 송신하는 단계, 상기 단말기가 상기 제1 서버로부터 응답값을 수신하는 단계, 상기 단말기가 상기 응답값을 검증하는 단계, 및 상기 단말기가 상기 데이터에 대한 비밀키를 생성하는 단계를 포함한다.

[0014] 또한, 본 발명의 실시 예에 따른 중복 제거 방법은, 서버가 단말기로부터 태그와 제1 질의값을 수신하는 단계, 상기 서버가 제1 응답값과 제2 질의값을 상기 단말기로 송신하는 단계, 상기 서버가 상기 단말기로부터 제2 응답값과 식별자를 수신하는 단계, 상기 서버가 상기 제2 응답값을 검증하는 단계, 및 상기 서버가 상기 식별자를 저장하는 단계를 포함한다.

발명의 효과

[0015] 본 발명의 실시 예에 따른 비밀키 생성 방법 및 중복 제거 방법에 의하면, 일정 기간 동안 데이터의 비밀키 질의를 할 수 있는 횟수를 제한하는 키 생성 서버를 둠으로써 전수조사 공격 및 온라인 추측 공격에 대한 안전성

을 제공할 수 있다.

[0016] 또한, 데이터 업로드시 클라이언트가 클라우드 서버에 저장된 암호문의 무결성을 검증함으로써 데이터 손실을 가져오는 포이즌 공격에 대한 안전성을 제공할 수 있다.

[0017] 또한, 데이터 업로드 및 다운로드시 클라우드 서버가 클라이언트의 데이터 소유권을 검증함으로써 정당한 클라이언트에게만 접근 권한을 부여할 수 있다.

도면의 간단한 설명

[0018] 본 발명의 상세한 설명에서 인용되는 도면을 보다 충분히 이해하기 위하여 각 도면의 상세한 설명이 제공된다.

도 1은 본 발명의 일 실시 예에 의한 중복 제거 시스템을 도시한다.

도 2는 도 1에 도시된 제1 서버와 클라이언트 단말기 간의 비밀키 생성 과정을 설명하기 위한 흐름도이다.

도 3은 도 1에 도시된 클라이언트 단말기와 제2 서버 사이에 수행되는 중복 제거 방법을 설명하기 위한 흐름도이다.

도 4는 도 1에 도시된 제2 서버와 클라이언트 단말기 간의 데이터 다운로드 과정을 설명하기 위한 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0019] 본 명세서에 개시되어 있는 본 발명의 개념에 따른 실시 예들에 대해서 특정한 구조적 또는 기능적 설명은 단지 본 발명의 개념에 따른 실시 예들을 설명하기 위한 목적으로 예시된 것으로서, 본 발명의 개념에 따른 실시 예들은 다양한 형태들로 실시될 수 있으며 본 명세서에 설명된 실시 예들에 한정되지 않는다.

[0020] 본 발명의 개념에 따른 실시 예들은 다양한 변경들을 가할 수 있고 여러 가지 형태들을 가질 수 있으므로 실시 예들을 도면에 예시하고 본 명세서에서 상세하게 설명하고자 한다. 그러나, 이는 본 발명의 개념에 따른 실시 예들을 특정한 개시 형태들에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물, 또는 대체물을 포함한다.

[0021] 제1 또는 제2 등의 용어는 다양한 구성 요소들을 설명하는데 사용될 수 있지만, 상기 구성 요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성 요소를 다른 구성 요소로부터 구별하는 목적으로만, 예컨대 본 발명의 개념에 따른 권리 범위로부터 벗어나지 않은 채, 제1 구성 요소는 제2 구성 요소로 명명될 수 있고 유사하게 제2 구성 요소는 제1 구성 요소로도 명명될 수 있다.

[0022] 어떤 구성 요소가 다른 구성 요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성 요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성 요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성 요소가 다른 구성 요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는 중간에 다른 구성 요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성 요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.

[0023] 본 명세서에서 사용한 용어는 단지 특정한 실시 예를 설명하기 위해 사용된 것으로서, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 본 명세서에 기재된 특징, 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0024] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

[0025] 우선, 본 발명의 배경이 되는 RSA(Rivest Shamir Adleman) 블라인드 서명에 대한 간략한 설명은 다음과 같다.

[0026] RSA 블라인드 서명 기법은 서명자가 검증자로부터 전송받은 메시지를 RSA 기법을 이용하여 자신의 서명키로 서

명을 하여 전달하고, 검증자는 서명자로부터 받은 서명을 검증하는 개념이다. RSA 블라인드 서명 기법은 다음과 같은 단계로 구성된다.

[0027] 1. $KeyGen(\lambda)$ 알고리즘은 서명자가 보안상수 λ 를 입력받아 공개키($pk=(e,N)$)와 서명키($sk=(d,N)$)를 생성하는 알고리즘이다.

[0028] 2. Signature 프로토콜은 다음과 같이 구성된다.

[0029] - 검증자는 서명자의 공개키(pk)와 메시지(M)을 입력받아 난수(r)를 생성하고, $x=r^e \bullet M \bmod N$ 를 계산하여 서명자에게 전송한다.

[0030] - 서명자는 자신의 비밀키(sk)를 이용하여 $y=x^d \bmod N$ 를 계산한 후 검증자에게 전송한다.

[0031] - 검증자는 난수(r)의 역수를 계산하여 $y=x^d \bmod N$ 얻는다.

[0032] 3. $Ver(pk,z,M)$ 알고리즘은 검증자가 $z^e = M \bmod N$ 이 성립하는지 검증하는 알고리즘이다.

[0033] 이하, 본 명세서에 첨부된 도면들을 참조하여 본 발명의 실시 예들을 상세히 설명한다.

[0034] 도 1은 본 발명의 일 실시 예에 의한 중복 제거 시스템을 도시한다. 도 1을 참조하면, 중복 제거 시스템(10)은 클라이언트 단말기(300)와 제2 서버(500)를 포함한다. 중복 제거 시스템(10)은 제1 서버(100)을 더 포함할 수 있다. 중복 제거 시스템(10)에서 제1 서버(100)는 유무선 통신망을 통하여 클라이언트 단말기(300)와 통신할 수 있고, 클라이언트 단말기(300)는 유무선 통신망을 통하여 제2 서버(500)와 통신할 수 있다.

[0035] 키 생성 기관(Key Generation Center), 키 생성 서버, 키 서버 등으로 불릴 수 있는 제1 서버(100)는 클라이언트 단말기(300)가 제2 서버(500), 예컨대 클라우드 서버에 저장할 데이터를 암호화하는 과정에서 필요한 비밀키에 대한 비밀값을 생성한다. 전수 조사 공격을 예방하기 위해 클라이언트마다 또는 클라이언트 단말기(300)마다 일정 기간 동안 데이터의 비밀키 질의를 할 수 있는 횟수를 제한하는 정책을 설정하여 제한된 횟수 이상 비밀키 질의를 했을 경우에는 응답을 멈추도록 설정된다. 추가적으로 제1 서버(100)는 클라이언트 단말기(300)와의 통신에서 필요한 연산을 정직하게 수행한다.

[0036] 클라이언트 단말기(300)는 PC(personal computer), 태블릿 PC, 노트북(notebook), 넷-북(net-book), e-리더(e-reader), PDA(personal digital assistant), PMP(portable multimedia player), MP3 플레이어, 또는 MP4 플레이어와 같은 데이터 처리 장치로 구현되거나, 모바일폰(mobile phone), 스마트폰(smart phone) 등과 같은 핸드헬드 장치(handheld device)로 구현될 수 있다.

[0037] 제1 서버(100)와 클라이언트 단말기(300) 간의 통신을 통한 비밀키 생성 방법과 클라이언트 단말기(300)와 제2 서버(500) 간의 통신을 통한 중복 제거 방법은 이하 도 2 내지 도 4를 통하여 상세하게 설명하기로 한다.

[0038] 도 2는 도 1에 도시된 제1 서버와 클라이언트 단말기 간의 비밀키 생성 과정을 설명하기 위한 흐름도이다.

[0039] 우선, 두 개의 해쉬 함수($H:\{0,1\}^* \rightarrow Z_N,\ G:Z_N \rightarrow \{0,1\}^k$)와 데이터(f)의 비밀키 공간($\{0,1\}^k$)이 주어져 있고, 제1 서버(100)의 공개키와 비밀키는 각각 RSA 블라인드 서명 기법의 공개키($pk_{KS}=(e,N)$)와 비밀키($sk_{KS}=(N,d)$)로 설정한다.

[0040] 클라이언트 단말기(300)는 제2 서버(500)에 업로드하려거나 제2 서버(500)로부터 다운로드하려는 데이터(f)의

해쉬 값($h = H(f)$)을 계산하고, 난수($r \in Z_N$)를 생성한다(S110).

[0041] 또한, 클라이언트 단말기(300)는 질의값(x)를 생성한다(S130). 질의값(x)는 아래의 수학식 1을 이용하여 생성될 수 있다.

수학식 1

[0042] $$x = r^e \cdot h \bmod N$$

[0043] 생성된 질의값(x)은 클라이언트 단말기(300)에 의해 제1 서버(100)로 송신될 수 있다(S150).

[0044] 질의값(x)을 수신한 제1 서버(100)는 제1 서버(100)의 비밀키($sk_{KS} = (N,d)$)를 이용하여 응답값(y)을 생성하고(S160), 생성된 응답값(y)을 클라이언트 단말기(300)로 송신한다(S170). 이때, 응답값(y)은 아래의 수학식 2를 이용하여 생성될 수 있다.

수학식 2

[0045] $$y = x^d \bmod N$$

[0046] 제1 서버(100)로부터 응답값(y)을 수신한 클라이언트 단말기(300)는 응답값(y)을 검증한다(S180). 구체적으로, 클라이언트 단말기(300)는 응답값(y)을 이용하여 연산값(z)을 계산한 후, 검증식을 이용하여 응답값(y)을 검증할 수 있다. 이때, 연산값(z)은 아래의 수학식 3을 이용하여 계산될 수 있고, 상기 검증식은 아래의 수학식 4일 수 있다.

수학식 3

[0047] $$z = y \cdot r^{-1} \bmod N$$

수학식 4

[0048] $$z^e = h \bmod N$$

[0049] 상기 수학식 4가 성립하는 경우, 클라이언트 단말기(300)는 데이터(f)의 비밀키($sk_f = G(z)$)를 생성한다(S190).

[0050] 도 3은 도 1에 도시된 클라이언트 단말기와 제2 서버 사이에 수행되는 중복 제거 방법을 설명하기 위한 흐름도이다.

[0051] 우선, 해쉬 함수($H_1 : \{0,1\}^* \rightarrow \{0,1\}^l$ 가 주어져 있고, 데이터 암호화 함수($Enc_{sk}(\cdot)$)는 안전한 대칭키 암호 기법을 사용한다고 가정한다.

[0052] 클라이언트 단말기(300)는 제1 서버(100)와의 통신을 통하여 생성한 데이터(f)의 비밀키(sk_f)를 해쉬하여 태그($T = H_1(sk_f)$)를 생성하고, 난수($s \in \{0,1\}^*$)를 생성한다(S210). 난수(s)는 제2 서버(500)에 저장된 암호문의 무결성을 질의하는 제1 질의값으로 사용될 수 있다.

[0053] 클라이언트 단말기(300)는 태그(T)와 제1 질의값(s)을 제2 서버(500)로 송신한다(S230). 이때, 클라이언트 단말기(300)는 태그(T)와 제1 질의값(s)을 데이터 업로드 요청 메시지와 함께 제2 서버(500)로 송신할 수 있다.

[0054] 제2 서버(500)는 수신된 태그(T)와 일치하는 태그가 제2 서버(500) 내에 저장되어 있는지 검색한다(S250). 태그(T)와 일치하는 태그가 제2 서버(500) 내에 저장되어 있지 않은 경우, 제2 서버(500)는 데이터 업로드 요청을 클라이언트 단말기(300)로 송신할 수 있다(S310).

[0055] 데이터 업로드 요청을 수신한 클라이언트 단말기(300)는 암호문($C = Enc_{sk_f}(f)$)과 암호화된 비밀키($K_c = Enc_{sk_c}(sk_f)$)를 생성하고, 생성된 암호문(C), 식별자(ID_c), 및 암호화된 비밀키(K_c)를 제2 서버(500)로 송신한다(S330). 여기서, 비밀키(sk_f)를 암호화할 때 사용되는 키(sk_c)는 클라이언트 단말기(300)가 비밀로 보관하고 있는 개인키일 수 있다.

[0056] 제2 서버(500)는 수신된 암호문(C), 식별자(ID_c), 및 암호화된 비밀키(K_c)를 태그(T)와 함께 저장한다(S350).

[0057] 태그(T)와 일치하는 태그가 제2 서버(500) 내에 저장되어 있는 경우, 제2 서버(500)는 제1 응답값($h = H_1(C' \| s)$)과 제2 질의값($k \in \{0,1\}^*$)을 생성한다(S410). 즉, 태그(T)와 일치하는 태그가 제2 서버(500)에 저장되어 있다는 것은 태그(T)에 대응하는 암호문(C')이 제2 서버(500)에 저장되어 있음을 의미한다. 따라서, 제2 서버(500)는 제2 서버(500) 내에 저장된 암호문(C')의 무결성을 증명하기 위한 제1 응답값(h)과 암호문(C')과 동일한 암호문(C)이 클라이언트 단말기(300)에 저장되어 있는지 유무(소유권 유무)를 질의하기 위한 제2 질의값(k)을 생성한다. 제2 질의값(k)은 제2 서버(500)에 의해 선택된 난수일 수 있다.

[0058] 생성된 제1 응답값(h)과 제2 질의값(k)은 제2 서버(500)에 의해 클라이언트 단말기(300)로 송신된다(S420).

[0059] 제1 응답값(h)을 수신한 클라이언트 단말기(300)는 제1 응답값(h)을 검증한다(S430). 제1 응답값(h)을 검증함으로써 제2 서버(500)에 저장되어 있는 암호문(C')의 무결성을 검증할 수 있다. 구체적으로, 클라이언트 단말기(300)는 데이터(f)의 암호문($C = Enc_{sk_f}(f)$)을 생성하고, 생성된 암호문(C)과 제1 질의값(s)을 이용하여 $H_1(C \| s)$ 을 계산한 후 $h = H_1(C \| s)$ 가 성립하는지 확인함으로써, 포이즌 공격을 검증할 수 있다.

[0060] 검증이 통과되면, 클라이언트 단말기(300)는 제2 응답값($h' = H_1(C \| k)$)을 생성하고, 생성된 제2 응답값(h'), 암호화된 비밀키($K_c = Enc_{sk_c}(sk_f)$), 및 식별자(ID_c)를 제2 서버(500)로 송신한다(S440). 식별자(ID_c)는 클라이언트 단말기(300)의 사용자를 식별할 수 있는 정보이거나 클라이언트 단말기(300)를 식별할 수 있는 정보일 수 있다. 이때, 클라이언트 단말기(300)는 승인 메시지와 함께 제2 응답값(h'), 암호화된 비밀키(K_c), 및 식별자(ID_c)를 제2 서버(500)로 송신할 수도 있다.

[0061] 이때, 검증이 통과되지 못하면, 클라이언트 단말기(300)는 오류 메시지를 제2 서버(500)로 송신할 수 있다.

[0062] 제2 서버(500)는 수신된 제2 응답값을 검증함으로써, 소유권 검증을 수행할 수 있다(S450). 구체적으로, 제2 서버(500)는 제2 서버(500)에 저장되어 있는 암호문(C')과 제2 질의값(k)을 이용하여 $H_1(C' \| k)$를 계산한 후 $h' = H_1(C' \| k)$이 성립하는지 여부를 확인하여 소유권 검증을 한다. 소유권 검증이 완료되면, 제2 서버(500)는 승인 메시지를 클라이언트 단말기(300)로 송신하고(S460), 클라이언트 단말기(300)로부터 수신받은 식별자(ID_c)와 암호화된 비밀키(K_c)를 태그(T)와 함께 저장한다(S470).

[0063] 소유권 검증을 통과하지 못한 경우, 제2 서버(500)는 오류 메시지를 클라이언트 단말기(300)로 송신할 수 있다.

[0064] 도 4는 도 1에 도시된 제2 서버와 클라이언트 단말기 간의 데이터 다운로드 과정을 설명하기 위한 흐름도이다.

[0065] 클라이언트 단말기(300)는 태그(T)와 식별자(ID_c)를 제2 서버(500)로 송신한다(S510). 클라이언트 단말기(300)는 태그(T)와 식별자(ID_c)를 데이터 다운로드 요청 메시지와 함께 송신할 수도 있다.

[0066] 제2 서버(500)는 클라이언트 단말기(300) 또는 클라이언트에 대한 데이터 소유권 검사를 수행한다(S530). 즉, 제2 서버(500)는 수신된 태그(T)와 식별자(ID_c)가 제2 서버(500)에 저장되어 있는지 검색하고, 태그(T)와 동일한 태그와 태그(T)에 대응하는 식별자(ID_c)가 제2 서버(500)저장되어 있는 경우, 제2 서버(500)는 클라이언트 단말기(300) 또는 클라이언트가 데이터를 다운로드할 수 있는 권한이 있는 것으로 판단한다.

[0067] 데이터 소유권 검사가 완료되며, 제2 서버(500)는 태그(T)에 대응하는 암호문(C)과 암호화된 비밀키(K_c)를 클라이언트 단말기(300)로 송신한다(S550).

[0068] 클라이언트 단말기(300)는 제2 서버(500)로부터 수신한 암호화된 비밀키(K_c)를 복호화하여 데이터(f)의 비밀키(sk_f)를 생성하고, 생성된 비밀키(sk_f)를 이용하여 암호문(C)를 복호화하여 데이터(f)를 생성한다(S570).

[0069] 본 발명은 도면에 도시된 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 등록청구범위의 기술적 사상에 의해 정해져야 할 것이다.

부호의 설명

[0070] 10 : 중복 제거 시스템

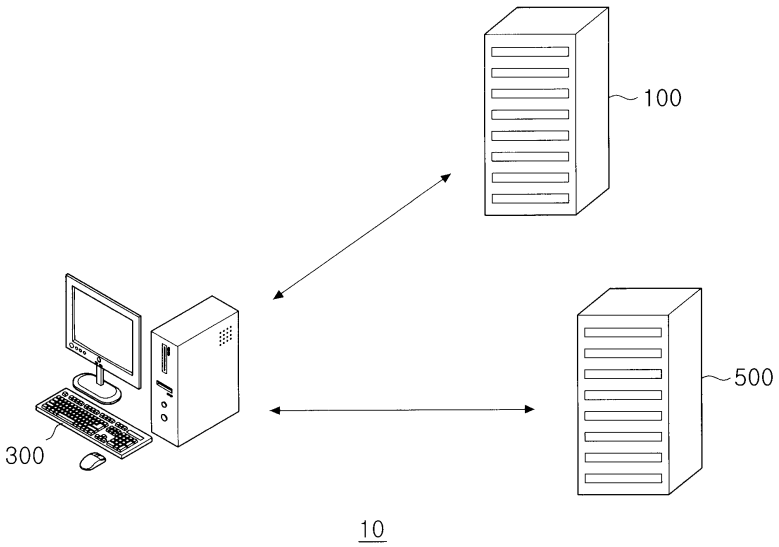
100 : 제1 서버

300 : 클라이언트 단말기

500 : 제2 서버

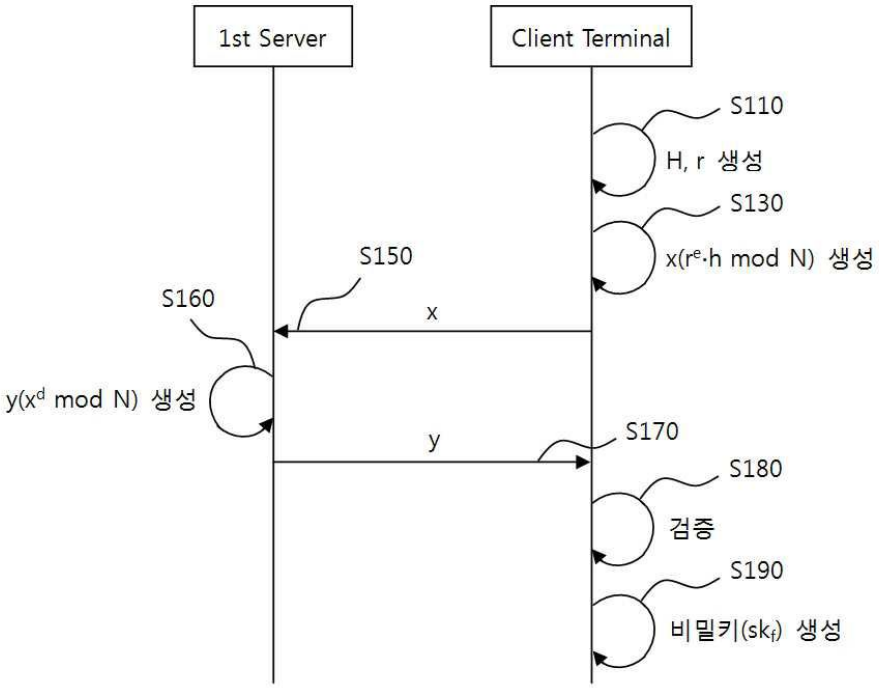
도면

도면1

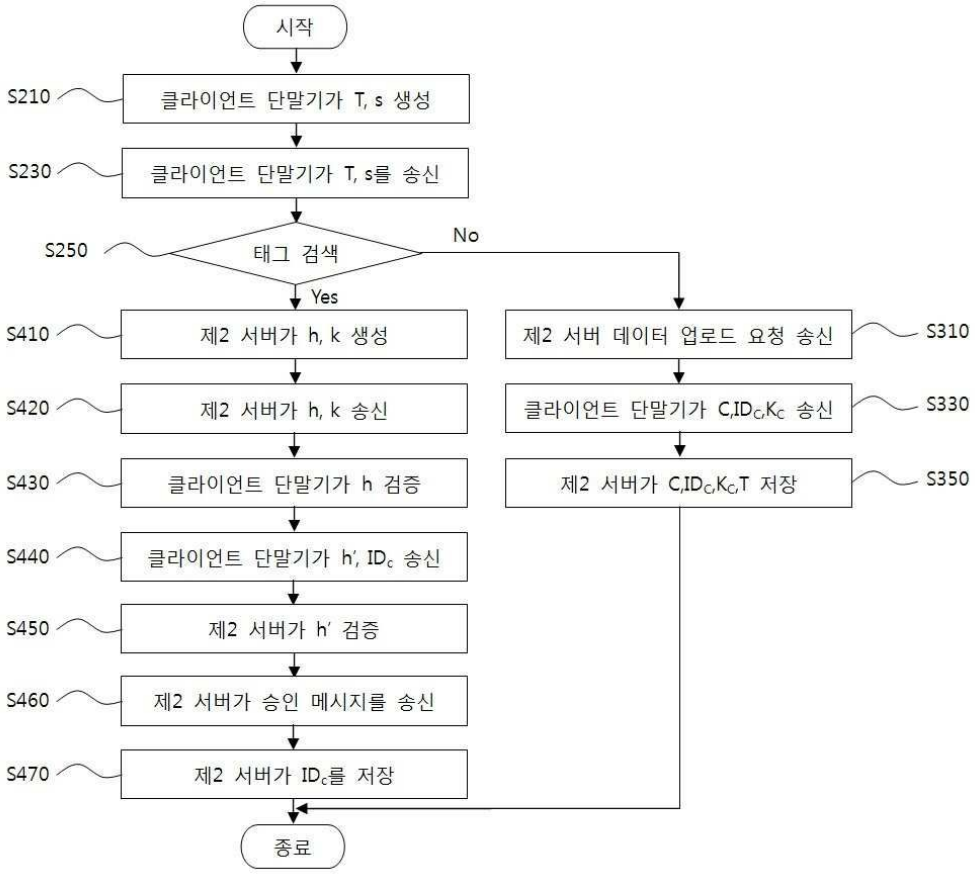


10

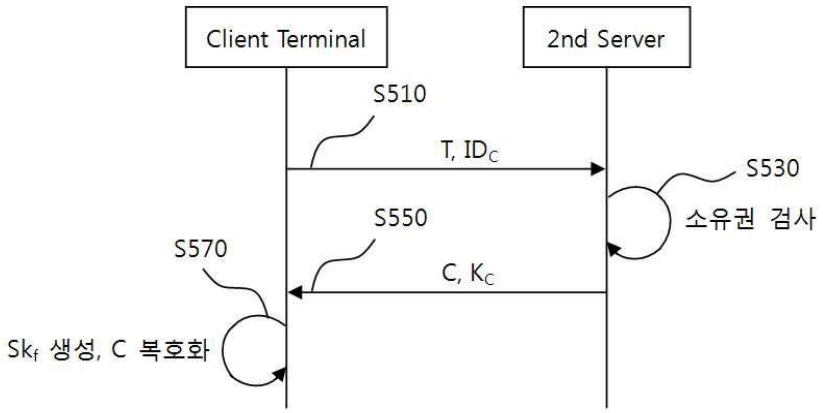
도면2



도면3



도면4



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 청구항 5

【변경전】

는 중복 제거 방법.

【변경후】

중복 제거 방법.