



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2012-0059435
(43) 공개일자 2012년06월08일

(51) 국제특허분류(Int. Cl.)

G06F 17/15 (2006.01) G06F 21/00 (2006.01)

(21) 출원번호 10-2011-0127140

(22) 출원일자 2011년11월30일

심사청구일자 2011년11월30일

(30) 우선권주장

1020100120734 2010년11월30일 대한민국(KR)

(71) 출원인

고려대학교 산학협력단

서울 성북구 안암동5가 1

(72) 발명자

이동훈

서울특별시 종로구 사직로8길 34, 경희궁의아침
3단지 아파트 1404호 (내수동)

정익래

서울특별시 성동구 성수동1가 709번지 성수2차
대우아파트 101동 1704호

(뒷면에 계속)

(74) 대리인

특허법인충현

전체 청구항 수 : 총 6 항

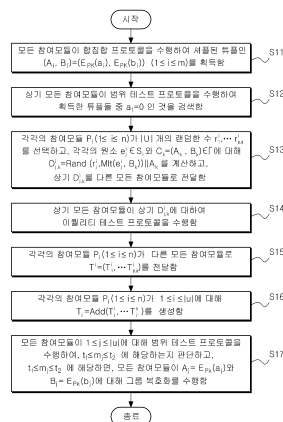
(54) 발명의 명칭 프라이버시를 보호할 수 있는 집합 연산 방법

(57) 요약

본 발명은 프라이버시(privacy)를 보호할 수 있는 집합 연산 방법에 관한 것으로, 보다 상세하게는 집합 연산(set operation)이 적용되는 암호화 환경에 속하는 적어도 하나의 참여모듈의 프라이버시를 보호할 수 있는 집합 연산 방법에 있어서, 각각의 참여모듈이 합집합 프로토콜(Privacy Preserving Set Union)을 수행하여 셔플링(shuffling)된 튜플(tuple)을 획득하는 튜플획득단계; 상기 참여모듈이 획득한 튜플 중 평문(plain text)이 0인 것을 제거하여 상기 튜플의 합집합을 확정하는 합집합확정단계; 상기 참여모듈이 확정된 합집합을 다른 참여모듈로 전달하는 합집합전달단계; 및 참여모듈이 상기 합집합에 속하는 원소들 중 자신에 속하는 원소들만을 복호화하는 복호화단계;를 포함하는 것을 특징으로 한다.

본 명세서에서 개시하는 집합 연산 방법은 집합 연산(set operation)이 적용되는 암호화 환경에 참여한 모듈(이하 '참여모듈')들 각자가 가진 비밀 집합의 원소들 중 상기 참여모듈들 중 소정 개수 범위의 참여모듈들이 갖는 원소들을 제외한 나머지 원소들에 대해서는 노출을 방지하여 상기 참여모듈들의 프라이버시(privacy)를 보호한다.

대 표 도 - 도1



(72) 발명자

천지영

서울특별시 성북구 하월곡4동 두산위브 아파트
118동 104호

노건태

서울특별시 강남구 개포3동 주공아파트 707동
1304호

이 발명을 지원한 국가연구개발사업

과제고유번호 ITAA1100090100480001000100100

부처명 지식경제부

연구사업명 IT원천기술개발

연구과제명 Car-헬스케어 보안 기술개발

주관기관 고려대학교 산학협력단

연구기간 2010.03.01 ~ 2011.02.28

특허청구의 범위

청구항 1

집합 연산(set operation)이 적용되는 암호화 환경에 속하는 적어도 하나의 참여모듈의 프라이버시를 보호할 수 있는 집합 연산 방법에 있어서,

각각의 참여모듈이 합집합 프로토콜(Privacy Preserving Set Union)을 수행하여 셔플링(shuffling)된 튜플(tuple)을 획득하는 튜플획득단계;

상기 참여모듈이 획득한 튜플 중 평문(plain text)이 0인 것을 제거하여 상기 튜플의 합집합을 확정하는 합집합확정단계;

상기 참여모듈이 확정된 합집합을 다른 참여모듈로 전달하는 합집합전달단계; 및

참여모듈이 상기 합집합에 속하는 원소들 중 자신에 속하는 원소들만을 복호화하는 복호화단계;

를 포함하는 것을 특징으로 하는 프라이버시를 보호할 수 있는 집합 연산 방법.

청구항 2

제1항에 있어서,

상기 튜플획득단계는

상기 참여모듈이 자신의 비밀집합으로부터 셔플 프로토콜(Shuffle Protocol)을 수행하여 순서가 변경됨에 따라 셔플링된 튜플을 획득하는 것을 특징으로 하는 프라이버시를 보호할 수 있는 집합 연산 방법.

청구항 3

제1항에 있어서,

상기 합집합확정단계는

상기 참여모듈이 평문이 기설정된 범위 내 포함되는지 여부를 판단하는 범위 테스트 프로토콜(Range Test Protocol)을 수행하여 획득한 상기 튜플 중 평문이 0인 것을 제거하여 상기 튜플의 합집합을 확정하는 것을 특징으로 하는 프라이버시를 보호할 수 있는 집합 연산 방법.

청구항 4

제1항에 있어서,

상기 합집합전달단계는

하나의 참여모듈이 두 개의 암호문을 가지고 있을 때, 두 암호문의 평문이 동일한지 여부를 상기 하나의 참여모듈만이 확인하는 이퀄리티 테스트 프로토콜(Equality Test Protocol)을 수행하여 상기 참여모듈이 확정된 합집합을 다른 참여모듈로 전달하는 것을 특징으로 하는 프라이버시를 보호할 수 있는 집합 연산 방법.

청구항 5

제1항에 있어서,

상기 복호화단계는

범위 테스트 프로토콜(Range Test Protocol)을 수행하여 기설정된 개수 범위에 속하는 참여모듈이 검색되고, 검색된 참여모듈이 상기 합집합에 속하는 원소들 중 자신에 속하는 원소들만을 복호화하는 것을 특징으로 하

는 프라이버시를 보호할 수 있는 집합 연산 방법.

청구항 6

제1항 내지 제5항 중 어느 한 항의 방법을 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 판독 가능한 기록 매체.

명세서

기술 분야

[0001] 본 발명은 프라이버시(privacy)를 보호할 수 있는 집합 연산 방법에 관한 것으로, 보다 상세하게는 집합 연산(set operation)이 적용되는 암호화 환경에 참여한 모듈(이하 '참여모듈'로 약칭한다.)들이 가지고 있는 비밀 집합들에 존재하는 원소들 중 소정 개수 범위의 참여모듈이 가지고 있는 원소를 제외한 나머지 원소들에 대해서는 다른 참여모듈들에게 불필요한 노출을 방지하여 각 참여모듈의 프라이버시를 보호할 수 있는 집합 연산 방법에 관한 것이다.

배경 기술

[0002] 참여모듈의 프라이버시를 보호하기 위한 집합 연산은 데이터 마이닝(data mining) 등 암호화가 적용되는 많은 환경에서 유용하게 사용될 수 있는데, 참여모듈의 프라이버시를 보호하기 위해 합집합(Union set) 연산, 교집합(Intersection set) 연산, 부분 집합(Sub set) 연산 등의 집합 연산이 적용될 수 있으나 암호화 환경에 따라 자신이 가진 비밀 집합에 대한 정보의 불필요한 노출이 발생하여 참여모듈의 프라이버시 보호에 만전을 기할 수 없는 문제가 발생한다.

발명의 내용

해결하려는 과제

[0003] 본 발명은 상기와 같은 문제점을 해결하기 위해 창안된 것으로, 본 발명이 해결하려는 과제는 참여모듈의 프라이버시를 보호하기 위한 집합 연산을 수행하는 경우에 비밀 집합에 대한 정보의 불필요한 노출의 발생이 없도록 하여 참여모듈의 프라이버시 보호에 만전을 기할 수 있는 집합 연산 방법을 제시하는 것이다.

과제의 해결 수단

[0004] 위와 같은 과제를 해결하기 위한 본 발명의 한 실시 예에 따른 집합 연산(set operation)이 적용되는 암호화 환경에 속하는 적어도 하나의 참여모듈의 프라이버시를 보호할 수 있는 집합 연산 방법은 각각의 참여모듈이 합집합 프로토콜(Privacy Preserving Set Union)에 기초하여 셔플링(shuffling)된 튜플(tuple)을 획득하는 튜플획득단계; 상기 참여모듈이 획득한 튜플 중 평문(plain text)이 0인 것을 제거하여 상기 튜플의 합집합을 확정하는 합집합확정단계; 상기 참여모듈이 확정된 합집합을 다른 참여모듈로 전달하는 합집합전달단계; 및 참여모듈이 상기 합집합에 속하는 원소들 중 자신에 속하는 원소들만을 복호화하는 복호화단계;를 포함하는 것을 특징으로 한다.

[0005] 보다 바람직하게는 상기 참여모듈이 자신의 비밀집합으로부터 셔플 프로토콜(Shuffle Protocol)을 통해 순서가 변경됨에 따라 셔플링된 튜플을 획득하는 튜플획득단계를 포함할 수 있다.

[0006] 보다 바람직하게는 상기 참여모듈이 평문이 기설정된 범위 내 포함되는지 여부를 판단하는 범위 테스트 프로토콜(Range Test Protocol)에 기초하여 획득한 상기 튜플 중 평문이 0인 것을 제거하여 상기 튜플의 합집합을 확정하는 합집합확정단계를 포함할 수 있다.

[0007] 보다 바람직하게는 하나의 참여모듈이 두 개의 암호문을 가지고 있을 때, 두 암호문의 평문이 동일한지 여부를 상기 하나의 참여모듈만이 확인하는 이퀄리티 테스트 프로토콜(Equality Test Protocol)에 기초하여 상기

참여모듈이 확정된 합집합을 다른 참여모듈로 전달하는 합집합전달단계를 포함할 수 있다.

[0008] 보다 바람직하게는 범위 테스트 프로토콜(Range Test Protocol)에 따라 기설정된 개수 범위에 속하는 참여모듈이 검색되고, 검색된 참여모듈이 상기 합집합에 속하는 원소들 중 자신에 속하는 원소들만을 복호화하는 복호화단계를 포함할 수 있다.

발명의 효과

[0009] 본 발명은 집합 연산이 적용되는 암호화 환경에서 n 개의 참여모듈이 각자의 비밀 집합을 가지고 있다고 할 때, 각자의 비밀 집합의 원소들 중 소정 개수 범위의 참여모듈이 가지고 있는 원소를 제외한 나머지 원소들에 대해서는 다른 참여모듈들에게 불필요한 노출을 방지할 수 있으므로 각 참여모듈의 프라이버시 보호에 만전을 기할 수 있다.

도면의 간단한 설명

[0010] 도 1은 본 발명의 흐름을 제시한 도면이다.
 도 2는 합집합 프로토콜의 수행에 따른 순서도이다.
 도 3은 범위 테스트 프로토콜의 수행에 따른 순서도이다.
 도 4는 이퀄리티 테스트 프로토콜의 수행에 따른 순서도이다.

발명을 실시하기 위한 구체적인 내용

[0011] 이하, 본 발명을 실시하기 위한 구체적인 내용을 본 발명의 바람직한 실시 예에 근거하여 첨부 도면을 참조하여 상세히 설명하되, 도면의 구성요소들에 참조번호를 부여함에 있어서 동일 구성요소에 대해서는 비록 다른 도면상에 있더라도 동일 참조번호를 부여하였으며 당해 도면에 대한 설명 시 필요한 경우 다른 도면의 구성요소를 인용할 수 있음을 미리 밝혀둔다. 아울러 본 발명과 관련된 공지 기능 혹은 구성에 대한 구체적인 설명 그리고 그 이외의 제반 사항이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우, 그 상세한 설명을 생략한다.

[0012] 집합 연산이 적용되는 암호화 환경에서 참여모듈들 각자의 프라이버시를 보호하기 위한 집합 연산의 가장 중요하고도 엄수되어야 할 사실은 각 참여모듈이 자신이 가진 비밀 집합의 원소들 중 소정의 원소 이외에 다른 원소는 불필요하게 노출되지 아니하여야 하는 것이다. 참여모듈들의 프라이버시를 보호하기 위한 기존 집합 연산의 가장 큰 문제점은 이러한 불필요한 노출이 발생할 위험이 매우 높은 것이다.

[0013] 기존 집합 연산의 이러한 문제점을 해결하기 위한(불필요한 노출을 방지하기 위한) 본 발명의 골자를 제시하면 다음과 같다.

[0014] 본 발명은 집합 연산이 적용되는 암호화 환경에서 n 개의 참여모듈이 각자의 비밀 집합을 가지고 있다고 할 때 각자의 비밀 집합의 원소들 중 t_1 개 보다는 많고 t_2 개 보다는 적은 참여모듈이 즉, 소정 개수 범위의 참여모듈이 가지고 있는 원소가 무엇인지 찾는 방안에 관한 것으로, 이러한 원소를 제외한 나머지 원소들에 대해서는 다른 참여모듈들에게 불필요한 노출을 방지하여 각 참여모듈의 프라이버시를 보호한다.

[0015] 한편 기존 집합 연산에서 t_1 개 보다는 많고 t_2 개 보다는 적은 참여모듈이 즉, 소정 개수 범위의 참여모듈이 가지고 있는 원소가 무엇인지에 대해서 찾는 경우에는 소위 t over-threshold 합집합을 사용하여 다음과 같이 이루어지는데, 우선 t_1 over-threshold 합집합을 사용하여 t_1 개 보다 많은 참여모듈이 가지고 있는 원소를 찾고 t_2+1 over-threshold 합집합을 사용하여 t_2+1 개 보다 많은 참여모듈이 가지고 있는 원소를 찾는다. 그리고 첫 번째 계산한 집합에 있는 원소에서 두 번째 계산한 집합에 있는 원소들을 제거하면 t_1 개 보다는 많은 참여모듈이 가지고 있고, t_2 개 보다는 적은 참여모듈이 가지고 있는 원소가 무엇인지에 대해서 찾을 수 있다.

[0016] 하지만, 이 경우에는 반드시 알아야 하는 정보 이외의 정보들, 즉 t_2+1 개 보다 많은 사람이 가지고 있는 원소들에 대해서도 불필요한 노출이 발생하는 문제점이 있다. 본 발명에 의하면 이러한 노출을 방지하여 원하는

결과를 얻을 수 있도록 하는 것이다. 여기서 원하는 결과는 " t_1 개 보다 많은 참여모듈이 가지고 있고, t_2 개 보다는 적은 참여모듈이 가지고 있는 원소만을 얻고자 하는 것"이다.

[0017] 본 발명은 이러한 노출을 방지하기 위해 프라이버시를 보호하는 합집합(Privacy Preserving Set Union: PPSU) 프로토콜, 하나의 참여모듈이 두 개의 암호문을 가지고 있을 때 두 암호문의 평문이 같은지를 참여모듈 자신만이 알 수 있는 이퀄리티 테스트(Equality Test: ET) 프로토콜 및 암호문의 평문이 소정의 범위 안에 포함되는지를 모두가 알 수 있는 범위 테스트(Range Test: RT) 프로토콜로 구성된다. 여기서 소정의 범위는 해당 참여모듈 이외의 다른 참여모듈이 해당 참여모듈의 암호문에 대한 복호화를 수행하지 아니하여도 해당 참여모듈의 평문이 있는지를 알 수 있는 원소(값)의 범위를 의미하는 것으로 예를 들어, 특정 참여모듈이 가진 원소(평문) 5를 암호화한 암호문 $E_{pk}(5)$ 와 범위 (3,6)가 주어졌을 때, 특정 참여모듈 이외의 다른 참여모듈은 암호문 $E_{pk}(5)$ 에 대응되는 평문 5가 범위 (3,6)에 포함되는지를 암호문에 대한 복호화 없이도 알 수 있다.

[0018] 도 1은 본 발명의 흐름을 제시한 도면이다.

[0019] 기본 설정: 집합 연산이 적용되는 암호화 환경에 n 개의 참여모듈들이 있다. 각각의 참여모듈 P_i 는 k_i 개의 원소들을 가지는 비밀 집합 $S_i(1 \leq i \leq n)$ 를 가지며, S_i 는 k_i 개의 원소 $S_{i,j}(1 \leq j \leq k_i)$ 를 가진다.

[0020] 우선 n 개의 참여모듈들의 각 참여모듈 P_i 은 합집합(Privacy Preserving Set Union) 프로토콜을 이용하여 임의로 순서가 변경되어 셔플링(shuffling)된 튜플(tuple)들 $C_i=(A_i, B_i)=(E_{pk}(a_i), E_{pk}(b_i))$ ($1 \leq$

$i \leq m$)을 획득한다(S11). 여기서 각 튜플들은 $c_i=a_i/b_i \in U$ 이거나 $a_i=b_i=0$ 이며, m 은 셔플링된 튜플들의 개수를 의미한다. 그리고 U 는 합집합 프로토콜을 수행하여 계산된 합집합이며, $E_{pk}(a)$ 는 평문(plain text) a 를 암호화한 암호문을 의미한다. 즉 $E_{pk}(a_i)$ 와 $E_{pk}(b_i)$ 는 각각 참여모듈 P_i 가 가진 두 개의 평문 a_i 와 b_i 의 암호문을 의미한다.

[0021] 합집합 프로토콜은 기존에 공지된 프로토콜로 본 발명의 이해를 위해 도 2에 참고적으로 제시하였다.

[0022] 도 2는 합집합 프로토콜의 수행에 따른 순서도이다.

[0023] 상기 합집합 프로토콜을 수행하기 전에 선행되어야 하는 기본설정을 살펴보면 다음과 같다. 집합 연산이 적용되는 암호화 환경에 n 개의 참여모듈들이 있고, 각각의 참여모듈 P_i 는 k_i 개의 원소들을 가지는 비밀 집합 $S_i(1 \leq j \leq k_i)$ 를 가지며, S_i 는 k_i 개의 원소 $S_{i,j}(1 \leq j \leq k_i)$ 를 가지며, 준동형 암호화의 공개키 pk 에 대응되는 비밀키 sk 를 공유한다.

[0024] 도 2에 도시된 바와 같이, 각각의 참여모듈 P_i ($1 \leq i \leq n$)가 다항식

$$f_{S_i}(x) = (x - S_{i,1}) \dots (x - S_{i,k_i})$$

를 계산하고, 상기 다항식을 암호화하여 $E_{pk}(f_{S_i})$ 를 연산한다(S21).

$$E_{pk}\left(\prod_{j=1}^i f_{S_j}\right)$$

[0025] 이후, 각각의 참여모듈 $P_i(2 \leq i \leq n)$ 가 $E_{pk}\left(\prod_{j=1}^i f_{S_j}\right)$ 를 계산한다(S22).

[0026] 이어서, 상기 참여모듈 P_i 가 튜플들 $(E_{pk}(S_{1,j}), E_{pk}(1))$ ($1 \leq j \leq k_1$)을 게시하고, 각각의 참여모듈

P_i ($2 \leq i \leq n$)가 튜플들 $(E_{pk}\left(\left(\prod_{j=1}^{i-1} f_{S_j}(S_{i,k})\right) \cdot S_{i,k}\right), E_{pk}\left(\prod_{j=1}^{i-1} f_{S_j}(S_{i,k})\right))$ ($1 \leq k \leq k_i$)을

게시한다(S23).

[0027] 이후, 각각의 참여모듈 P_i ($1 \leq i \leq n$) 가 각각의 튜플 $(E_{pk}(x), E_{pk}(y))$ 에 대해 차례대로 랜덤한 수 r_i 를 선택하고, 튜플들값에 r_i 를 곱한 후, $(E_{pk}(x \cdot \prod_{j=1}^i r_j), E_{pk}(y \cdot \prod_{j=1}^i r_j))$ 를 게시한다(S24).

[0028] 이후, 모든 참여모듈 P_i ($1 \leq i \leq n$) 가 모든 튜플들에 대해 셔플 프로토콜을 수행한다(S25).

[0029] 다시 도 1로 돌아가서, 다음으로 모든 참여모듈은 상기 획득된 튜플들(A_i, B_i)에 대해 범위 테스트 프로토콜을 수행하여 $a_i=0$ 인 것을 찾는다(S12). 즉, S12 단계는 n 개의 모든 참여모듈의 비밀 집합에 대한 합집합을 확정하는 단계이다. 상기 합집합은 모든 원소가 고스란히 노출되면 안되기 때문에 Γ 와 같은 형태로 드러냅니다. 여기서 Γ 는 합집합의 원소를 각각 암호화한 형태이다. 만약 a_i 가 0이 아니면 합집합 Γ 에 (A_i, B_i) 를 추가하며, a_i 가 0이면 a_i 는 버린다. 결론적으로 $\Gamma=(C_1, \dots, C_{|\Gamma|})$ 가 된다. Γ 는 각각의 원소가 암호화된 상태로 되어 있는 합집합을 의미하며, $C_i = (A_i, B_i)$ 이고, $|\Gamma|$ 는 합집합 원소의 개수이다. 범위 테스트 프로토콜의 수행을 통해 $a_i=0$ 인 튜플들을 제거한다. 이 경우 상기한 소정의 범위는 $(0,0)$ 이며 $a_i=0$ 인 것들을 제거해야 정확한 합집합의 원소를 찾을 수 있다. 한편 $a_i=0$ 이면 반드시 $b_i=0$ 이다.

[0030] 이해의 편의를 위해 범위 테스트 프로토콜을 도 3에 제시하였다.

[0031] 도 3은 범위 테스트 프로토콜의 수행에 따른 순서도이다.

[0032] 상기 범위 테스트 프로토콜을 수행하기 전에 선행되어야 하는 기본설정을 살펴보면 다음과 같다. 집합연산이 적용되는 암호화 환경 n 개의 참여모듈이 있고, 각각의 참여모듈 P_i 가 하나의 암호문 $E_{pk}(a)$ 와 범위 t_i, t_2 를 알고 있으며, 여기서 $t_1 \leq t_2$ 이다. 또한 상기 참여모듈 P_i 는 준동형 암호화의 공개키 pk 에 대응되는 비밀키 sk 를 공유한다.

[0033] 도 3에 도시된 바와 같이, 참여모듈 P_1 이 $t_2 - t_1 + 1$ 개의 랜덤한 수 $r_{t_1}^1, \dots, r_{t_2}^1$ 를 선택하고, $A_{t_1}^1 = E_{pk}((m - t_1) \cdot r_{t_1}^1), \dots, A_{t_2}^1 = E_{pk}((m - t_2) \cdot r_{t_2}^1)$ 를 계산한 후, $(A_{t_1}^1, \dots, A_{t_2}^1)$ 를 참여모듈 P_2 로 전달한다(S31).

[0034] 이후, 각각의 참여모듈 P_i ($2 \leq i \leq n$) 가 차례대로 $t_2 - t_1 + 1$ 개의 랜덤한 수 $r_{t_1}^i, \dots, r_{t_2}^i$ 를 선택하고, $A_{t_1}^i = E_{pk}((m - t_1) \cdot \prod_{k=1}^i r_{t_1}^k), \dots, A_{t_2}^i = E_{pk}((m - t_2) \cdot \prod_{k=1}^i r_{t_2}^k)$ 를 계산한 후, 참여모듈 $P_{i+1 \bmod n}$ 으로 $(A_{t_1}^i, \dots, A_{t_2}^i)$ 를 전달하며, 참여모듈 P_n 이 $(A_{t_1}^n, \dots, A_{t_2}^n)$ 를 다른 참여모듈로 전달한다(S32).

[0035] 따라서, 모든 참여모듈이 $(A_{t_1}^n, \dots, A_{t_2}^n)$ 에 대해 셔플 프로토콜을 수행하여 $(A_{t_1}, \dots, A_{t_2})$ 를 획득한다(S33).

[0036] 이후, 모든 참여모듈이 $(A_{t_1}, \dots, A_{t_2})$ 에 대해 그룹복호화를 수행한다. 만약, 복호화된 값 중 하나의 값이라

도 0이면, 모든 참여모듈이 $t_1 \leq a \leq t_2$ 임을 판단한다(S34).

[0037] 다시 도 1로 돌아가서, 다음으로 상기 참여모듈들의 각각의 $P_i (1 \leq i \leq n)$ 는 $|U|$ 개의 랜덤(random)한 수

$r_1^i, \dots, r_{|U|}^i$ 를 선택하고, 각각의 원소 $e_j^i (\in S_i)$ 와 $C_k = (A_k, B_k) \in \Gamma$ 에 대해

$D_{j,k}^i = \text{Rand}(r_j^i, \text{Mult}(e_j^i, B_k)) \parallel A_k$ 를 산출한 후, $D_{j,k}^i$ 를 다른 모든 참여모듈에게 배포한다

(S13). 여기서 e_j^i 는 참여모듈 P_i 가 가지고 있는 비밀 집합 S_i 의 j 번째 원소를 의미하며, $D_{j,k}^i$

e_j^i 와 B_k 의 곱셈(Mult) 알고리즘을 수행하고, 그 알고리즘의 결과값을 랜덤화시킨 다음 A_k 를 연결(concatenation)한 것을 의미한다. A_k 와 B_k 는 Γ 에 포함된 암호화된 각각의 원소 C_k 에 대해 $C_k = (A_k,$

$B_k) = (E_{pk}(a_k), E_{pk}(b_k))$ 를 만족하는 원소를 의미한다. $\text{Mult}(e_j^i, B_k)$ 는

$E_{pk}(b_k)$ 에 대해서 $e_j^i \cdot B_k = D_{sk}(\text{Mult}(e_j^i, B_k))$ 를 만족하는 암호문을 만드는 것을 말

하며, $D_{sk}(\cdot)$ 는 비밀키 sk 를 가지고 있는 사람이 암호문 $E_{pk}(\circ)$ 을 복호화하는 것을 말한다.

[0038] 원래 a_i/b_i 꼴이 합집합의 원소가 된다. 여기서 $D_{j,k}^i$ 는 결과적으로 자신이 가지고 있는 원소에다가 b_k

를 곱하고, a_k 를 연결시킨 것에 대한 암호문이 된다. 즉, 참여모듈 P_i 가 자신의 j 번째 원소 e_j^i 를 사용하

여 $D_{j,k}^i$ 를 계산하게 되는데, 이 경우 e_j^i 가 합집합에 정말 포함되어 있는 값이라면

$D_{j,k}^i = E_{pk}(1)$ 이 된다. 즉, 1에 대한 암호문과 같아진다.

[0039] 다음으로 상기 다른 모든 참여모듈들은 각자 이퀄리티 테스트 프로토콜을 수행하여(S14), 만일 어떤

$u_k (\in U)$ 에 대하여 $e_j^i = u_k$ 이면 상기 다른 모든 참여모듈들의 각각의 참여모듈 P_i 는

$T_k^i = E_{pk}(1)$ 을 생성하고, 그렇지 않으면 참여모듈 P_i 는 $T_k^i = E_{pk}(0)$ 을 생성한다.

T_k^i 는 암호화되지 않은 평문 상태의 합집합의 원소를 의미하며, $T_k^i = E_{pk}(1)$ 은 참여모듈 P_i 의 어떤

$$T_k^i = E_{pk}(0)$$

원소가 암호화되지 않은 평문 상태의 합집합의 $k(=|U|)$ 번째 원소를 가지고 있다는 의미이고, 은 참여모듈 P_i 의 어떤 원소가 암호화되지 않은 평문 상태의 합집합의 k 번째 원소를 가지고 있지 않다는 의미이다.

[0040] 이해의 편의를 위해 이퀄리티 테스트 프로토콜을 도 4에 제시하였다.

[0041] 도 4는 이퀄리티 테스트 프로토콜의 수행에 따른 순서도이다.

[0042] 상기 이퀄리티 테스트 프로토콜을 수행하기 전에 선행되어야 하는 기본설정을 살펴보면 다음과 같다. 집합연산이 적용되는 암호화 환경 n 개의 참여모듈이 있고, 각각의 참여모듈 P_i 가 두 개의 암호문인 $E_{pk}(a)$ 와, $E_{pk}(b)$ 를 알고 있으며, 준동형 암호화의 공개키 pk 에 대응되는 비밀키 sk 를 공유한다.

[0043] 도 4에 도시된 바와 같이, 참여모듈 P_1 이 랜덤한 수 r_1 을 선택하고, 선택한 r_1 을 $E_{pk}(a-b)$ 에 곱한 후, 참여모듈 P_2 로 $E_1 = E_{pk}((a-b) \cdot r_1)$ 을 전송한다(S41).

[0044] 각각의 참여모듈 P_i ($2 \leq i \leq n$)가 차례대로 랜덤한 수 r_i 를 선택하고, 선택한 r_i 를 암호문 $E_{i-1} = E_{pk}((a-b) \cdot \prod_{j=1}^{i-1} r_j)$ 에 곱한 후, 참여모듈 $P_{i+1 \bmod n}$ 으로 $E_i = E_{pk}((a-b) \cdot \prod_{j=1}^i r_j)$ 를 전송한다(S42).

[0045] 이후, 참여모듈 P_1 가 랜덤한 수 R 를 선택하고, $E_n + E_{pk}(R)$ 을 계산하여 $E_{pk}((a-b) \cdot \prod_{i=1}^n r_i + R)$ 을 획득한 후, 획득한 $E_{pk}((a-b) \cdot \prod_{i=1}^n r_i + R)$ 을 다른 모든 참여모듈로 전송한다(S43).

[0046] 이에 따라, 모든 참여모듈이 그룹복호화를 수행하여 $x = (a-b) \cdot \prod_{i=1}^n r_i + R$ 을 획득한다. 만약 $x = R$ 이면, P_1 은 $a=b$ 임을 알 수 있게 된다(S44).

[0047] 다시 도 1로 돌아가서, 다음으로 각각의 참여모듈 P_i 는 모든 다른 참여모듈들에게 $T^i = (T_1^i, \dots, T_{|U|}^i)$ 를 배포한다(S15).

[0048] 다음으로 각각의 참여모듈 P_i 는 $T_j = Add(T_j^1, \dots, T_j^n) (1 \leq j \leq |U|)$ 를 생성한다(S16). $Add(\cdot)$ 알고리즘은 암호문들의 합을 각 암호문에 대응하는 평문을 합한 결과의 암호문을 생성해주는 알고리즘으로

$T_j^1, T_j^2, \dots, T_j^n$ 들은 모두 0 또는 1을 암호화한 값을 의미한 $Add(E_{pk}(a), E_{pk}(b)) = E_{pk}(a+b)$ 이다. 즉,

다. T_j 는 이 값들의 평균을 더한 값을 암호화한 것과 동일하고, j번째 원소를 가지고 있는 참여모듈이 몇 개인지가 암호화된 상태로 저장되어 있는 값을 의미한다.

[0049]

마지막으로 모든 참여모듈들은 범위 테스트 프로토콜을 수행하여(S17) T_j 가 $t_1 \leq m_j \leq t_2$ 인지

판단한다. 즉, T_j 가 소정 개수 범위의 참여모듈만이 갖는 것인지 판단한다. 여기서 $T_j = E_{pk}(m_j)$ 이다. 즉,

T_j m_j 는 T_j 의 평균이다. 상기 T_j 이 이 부등식을 만족하는 경우에만 모든 참여모듈들에 의해 그때의 튜플

$(A_j, B_j) = (E_{pk}(a_j), E_{pk}(b_j))$ 에 대하여 복호화가 허용된다.

[0050]

본 방법발명은 또한 컴퓨터로 읽을 수 있는 기록매체에 컴퓨터가 읽을 수 있는 코드로서 구현하는 것이 가능하다. 컴퓨터가 읽을 수 있는 기록매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록 장치를 포함한다.

[0051]

컴퓨터가 읽을 수 있는 기록매체의 예로는 ROM, RAM, CD-ROM, 자기 테이프, 플로피디스크, 광 데이터 저장장치 등이 있으며, 또한 캐리어 웨이브(예를 들어 유무선 네트워크를 통한 전송)의 형태로 구현되는 것도 포함한다. 또한 컴퓨터가 읽을 수 있는 기록매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어 분산방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수 있다.

[0052]

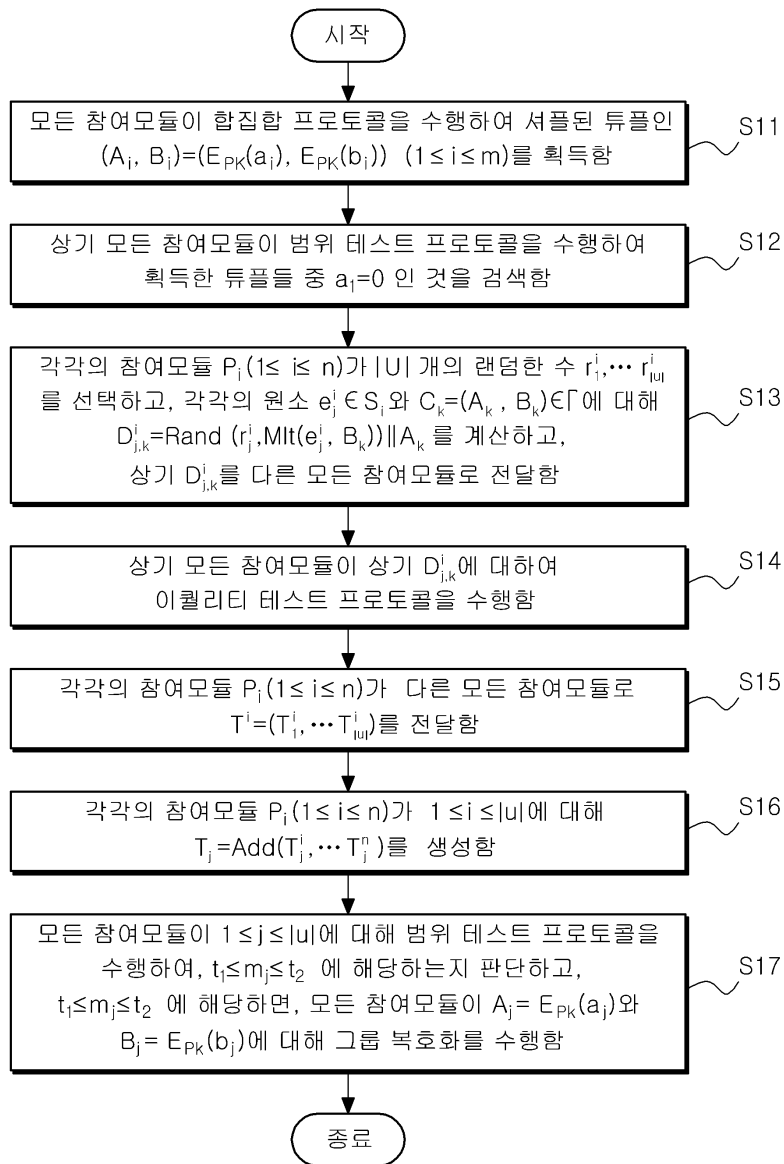
이제까지 본 발명에 대하여 그 바람직한 실시예를 중심으로 살펴보았다. 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자는 본 발명이 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 변형된 형태로 구현될 수 있음을 이해할 수 있을 것이다.

[0053]

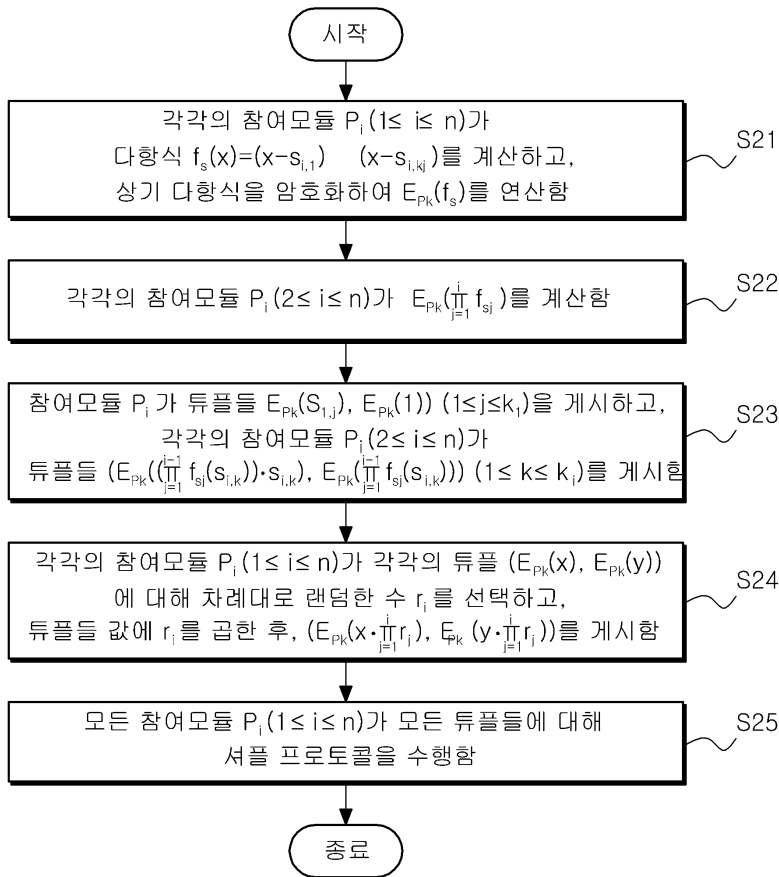
그러므로 개시된 실시예들은 한정적인 관점이 아니라 설명적인 관점에서 고려되어야 한다. 본 발명의 범위는 전술한 설명이 아니라 특허청구범위에 나타나 있으며, 그와 균등한 범위 내에 있는 모든 차이점은 본 발명에 포함된 것으로 해석되어야 할 것이다.

도면

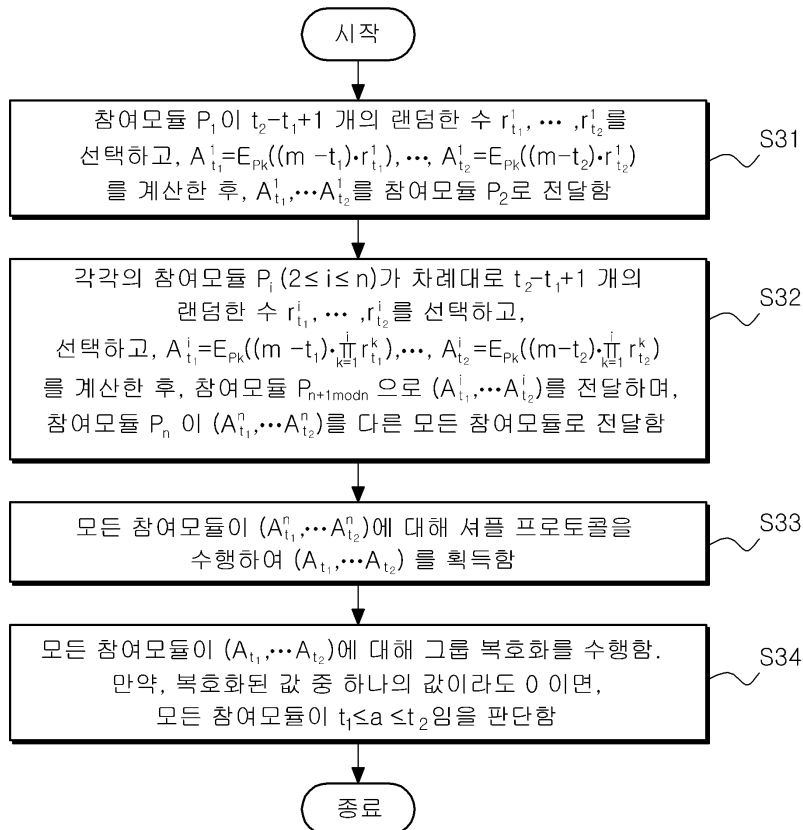
도면1



도면2



도면3



도면4

