(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2020-0055672
(43) 공개일자 2020년05월21일

(51) 국제특허분류(Int. Cl.)
H04L 9/30 (2006.01) *H04L 9/00* (2006.01)
H04L 9/06 (2006.01) *H04L 9/08* (2006.01)
(52) CPC특허분류
H04L 9/30 (2013.01)
H04L 9/003 (2013.01)
(21) 출원번호 10-2019-0144235
(22) 출원일자 2019년11월12일
심사청구일자 2019년11월12일
(30) 우선권주장
1020180138716 2018년11월13일 대한민국(KR)

(71) 출원인
(주)블루팝콘
서울특별시 강서구 마곡중앙6로 63, 6층 627호 (마곡동, 마곡테크노타워)
(72) 발명자
안세환
서울 양천구
(74) 대리인
특허법인키

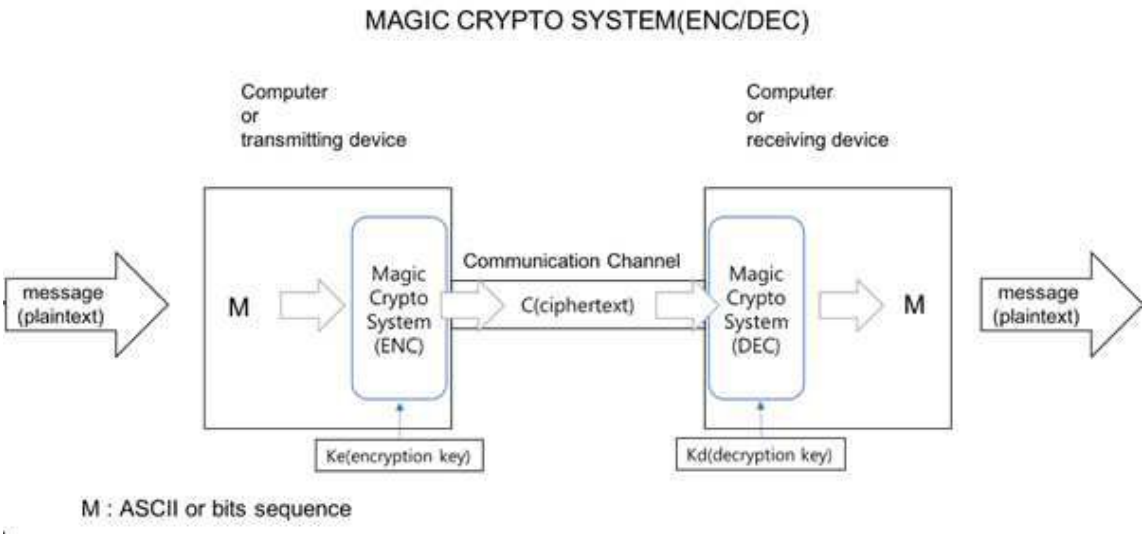
전체 청구항 수 : 총 31 항

(54) 발명의 명칭 **순열그룹 기반의 암호화 기술을 적용한 암호화시스템 및 방법**

(57) 요 약

본 발명은 메시지를 암호화 하여 송신하는 송신 단말과, 상기 암호화된 메시지를 복호화하는 수신 단말을 포함하며, 상기 송신 단말 및 수신 단말은 순열그룹에 기반하여 순열형태인 대칭키와 비대칭키를 동시에 합성하여 메시지를 암호화 한 후 송수신하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템을 개시한다.

대 표 도 - 도4



(52) CPC특허분류
H04L 9/0662 (2013.01)
H04L 9/0863 (2013.01)

명 세 서

청구범위

청구항 1

메시지를 암호화하는 암호화 수행객체; 및

상기 암호문을 원문 메시지로 복호화 하는 복호화 수행객체를 포함하며,

상기 암호화 수행객체 및 복호화 수행객체는 순열 그룹에 기반하여 순열 형태인 대칭키와 비대칭키를 동시에 합성하여 메시지를 암복호화하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 2

제1항에 있어서,

상기 암호화 수행객체 및 복호화 수행객체는,

암호화키(K_e)를 이용하여 메시지를 암호문으로 암호화 하는 암호화기(ENC)와, 암호화된 암호문을 복호화키(K_d)를 이용하여 복호화 하는 복호화기(DEC)와, 암호키생성기(MKG)를 포함하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 3

제2항에 있어서,

상기 암호화기(ENC)는,

메시지 입력을 처리하는 입력큐;

대칭키와 복호화 수행객체의 비대칭키를 이용하여 순열연산을 통해 암호문을 생성하는 GA 연산기; 및

상기 생성된 암호문의 출력을 처리하는 출력큐를 포함하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 4

제3항에 있어서,

상기 GA 연산기는,

상기 암호키생성기(MKG)로부터 대칭키(Q_{AB})와 복호화 수행객체의 비대칭키 쌍 중 공개키(G_B)를 제공받아 순열연산을 통해 암호문을 생성하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 5

제4항에 있어서,

상기 순열연산은 $Q_{AB}^{-1}G_BQ_{AB}(M) = C$ 수식에 의하여 이루어지며, Q_{AB}는 암호화 수행객체와 복호화 수행객체의 대칭키, G_B 는 복호화 수행객체의 공개키, M은 메시지 공간, C는 암호문 공간인 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 6

제2항에 있어서,

상기 복호화기(DEC)는,

암호문 입력을 처리하는 입력큐;

대칭키와 복호화 수행객체의 비대칭키를 이용하여 순열연산을 통해 원문메시지를 복원하는 GA 연산기; 및

상기 복원된 원문메시지의 출력을 처리하는 출력큐를 포함하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 7

제6항에 있어서,

상기 GA 연산기는,

상기 암호키생성기(MKG)로부터 대칭키(Q_{AB})와 복호화 수행객체의 비대칭키 쌍 중 개인키(H_B)를 제공받아 순열연산을 통해 원문 메시지를 복원하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 8

제7항에 있어서,

상기 순열연산은 $H_BQ_{AB}Q_{AB}(C) = M$ 수식에 의하여 이루어지며, Q_{AB}는 암호화 수행객체와 복호화 수행객체의 대칭키, H_B 는 복호화 수행객체의 비대칭키 쌍 중의 개인키, M은 메시지 공간, C는 암호문 공간인 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 9

제2항에 있어서,

상기 암호키생성기는,

복수의 매개변수를 사용한 키유도함수 KDF(key derivation function)를 통해 일회용 유사난수를 생성하는 난수생성기(PRNG); 및

상기 키유도함수(KDF)를 통해 일회용 유사난수순열(PRP)를 생성하여 키생성모듈에 제공하는 순열생성기를 포함하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 10

제9항에 있어서,

상기 키생성모듈은,

복호화 수행객체의 비밀함수그룹(secret permutation group)을 나타내는 벡터함수인 마스터키벡터(MSK_$\overrightarrow{tp}$) 모듈;

복호화할 때 사용될 비밀함수(secret permutation)를 결정하는 마스터키스칼라모듈(MSK_$\overrightarrow{tv}$);

암호화 수행객체와 복호화 수행객체 간 공유하는 대칭키를 생성하는 대칭키모듈(MPK); 및

메시지 암호화와 복호화 시 사용되는 복호화 수행객체의 한 쌍의 비대칭키인 공개키와 개인키를 함께 생성되는 공개키모듈(PK)과 개인키모듈(SK)을 포함하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 11

제10항에 있어서,

상기 마스터키벡터(MSK_$\overrightarrow{tp}$) 모듈, 마스터키스칼라모듈(MSK_$\overrightarrow{tv}$) 및 대칭키모듈(MPK)은 객체들을 상호 구분할 수 있는 하나 이상의 식별정보를 매개변수로 하여 키유도함수(KDF)를 통해 키 값들을 생성하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 12

암호키생성기가 식별인자를 이용하여 마스터키를 생성하는 단계;

암호키생성기에서 암호화와 복호화에 필요한 대칭키, 개인키 및 공개키 쌍을 생성하는 단계;

암호화 수행객체가 암호화키인 대칭키와 복호화 수행객체가 생성한 공개키를 수신하여 암호문을 생성하는 단계; 및

복호화 수행객체가 암호키생성기를 통해 생성된 대칭키와 비대칭키 중 자신의 개인키를 통해 암호화 수행객체를 통해 생성된 암호문을 복원하는 단계를 포함하는 것을 특징으로 하는 암호화 및 복호화 방법.

청구항 13

제12항에 있어서,

상기 식별인자는,

사용자의 개인정보를 포함하는 사용자 식별인자, 사용자 단말 정보를 포함하는 단말장치 식별인자 및 비밀함수 생성인자 중 적어도 하나를 포함하는 것을 특징으로 하는 암호화 및 복호화 방법.

청구항 14

제13항에 있어서,

상기 암호문은 상기 생성된 대칭키 및 복호화 수행객체의 공개키를 이용하여 GA 연산기에 연산시켜 생성하는 것을 특징으로 하는 암호화 및 복호화 방법.

청구항 15

제14항에 있어서,

상기 암호문(C)은 암호화 함수(E)에 암호화키(K_e)를 대입하여 순열연산($M \times K \rightarrow C$)에 의해 아래 수식과 같이 생성되는 것을 특징으로 하고,

$E(M, K_e) = Q^{-1}GQ(M) = C$

여기서 $Q^{-1}GQ$ 는 순열함수 G, Q의 순열연산(left multiplication)이고,

메시지 $M = (m_1 \cdots m_n)$, 키 $K = (k_1 \cdots k_n)$, 암호문 $C = (c_1 \cdots c_n)$는 메시지집합 $M = \{m_1 \cdots m_n\}$ 상에서의 순열그룹 G의 원소인 것($M,K,C \in G$)을 특징으로 하는 암호화 및 복호화 방법.

청구항 16

제12항에 있어서,

상기 암호문의 복원은 복호화 함수(D)에 복호화키(K_d)를 대입하여 순열연산($C \times K \rightarrow M$)에 의해 아래 수식과 같이 생성되는 것을 특징으로 하고,

$D(C, K_d) = HQQ(C) = M$

여기서 HQQ 는 순열함수 H, Q의 순열연산(left multiplication)이고,

메시지 $M = (m_1 \cdots m_n)$, 키 $K = (k_1 \cdots k_n)$, 암호문 $C = (c_1 \cdots c_n)$는 메시지집합 $M = \{m_1 \cdots m_n\}$ 상에서의 순열그룹 G의 원소인 것($M,K,C \in G$)을 특징으로 하는 암호화 및 복호화 방법.

청구항 17

암호문 작성시 서명문을 생성하는 서명 수행객체; 및

상기 암호문을 원문 메시지로 복호화 하기 위하여 상기 서명문을 검증하도록 형성되는 검증 수행객체를 포함하며,

순열 그룹에 기반하여 순열 형태인 대칭키와 비대칭키를 이용하여 서명문을 생성하고 검증하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 서명 검증 시스템.

청구항 18

제17항에 있어서,

상기 서명 수행객체 및 검증 수행객체는,

순열연산을 통해 서명문을 생성하는 서명기와, 순열연산을 통해 상기 서명문을 검증하는 검증기와, 암호키생성기(MKG)를 포함하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 서명 검증 시스템.

청구항 19

제18항에 있어서,

상기 서명기는,

메시지 입력을 처리하는 입력큐;

일회용개인키를 이용하여 순열연산을 통해 서명문을 생성하는 GA 연산기; 및

상기 생성된 서명문의 출력을 처리하는 출력큐를 포함하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 20

제19항에 있어서,

상기 서명기의 GA연산기는,

상기 암호키생성기(MKG)로부터 대칭키(Q_{AB})와 서명 수행객체의 개인키(H_A)를 제공받아 순열연산을 통해 서명문을 생성하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 21

제20항에 있어서,

상기 서명기의 순열연산은 $Q_{AB}^{-1}H_AQ_{AB}(M) = M_s$ 수식에 의하여 이루어지며, Q_{AB}는 서명 수행객체와 검증 수행객체의 대칭키, H_A는 서명 수행객체의 개인키, M은 메시지($m_1 \cdots m_n$), M_S는 서명문($s_1 \cdots s_n$)인 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 22

제21항에 있어서,

상기 검증기는,

서명문을 입력받아 처리하는 입력큐;

일회용공개키를 이용하여 순열연산을 통해 서명문을 검증하여 수락된 원문메시지를 생성하는 GA 연산기; 및

상기 수락된 원문메시지의 출력을 처리하는 출력큐를 포함하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 23

제22항에 있어서,

상기 검증기의 GA연산기는,

상기 암호키생성기(MKG)로부터 대칭키(Q_{AB})와 서명 수행객체의 공개키(G_A)를 제공받아 순열연산을 통해 서명문을 검증하고 수락 또는 거절 여부를 확인하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시

스템.

청구항 24

제23항에 있어서,

상기 검증기의 순열연산은 $G_AQ_{AB}Q_{AB}(M_s) = M$ 수식에 의하여 이루어지며, Q_{AB}는 서명 수행객체와 검증 수행객체의 대칭키, G_A는 서명 수행객체의 공개키, M은 메시지($m_1 \cdots m_n$), M_S는 서명문($s_1 \cdots s_n$)인 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 25

제18항에 있어서,

상기 암호키생성기는,

복수의 매개변수를 사용한 키유도함수 KDF(key derivation function)를 통해 일회용 유사난수를 생성하는 난수생성기(PRNG); 및

상기 키유도함수(KDF)를 통해 일회용 유사난수순열(PRP)를 생성하여 키생성모듈에 제공하는 순열생성기를 포함하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 26

제25항에 있어서,

상기 키생성모듈은,

서명 수행객체의 비밀함수그룹(secret permutation group)을 나타내는 벡터함수인 마스터키벡터(MSK_$\overrightarrow{tp}$) 모듈;

복호화할 때 사용될 비밀함수(secret permutation)를 결정하는 마스터키스칼라모듈(MSK_$\overrightarrow{tv}$);

서명 수행객체와 검증 수행객체 간 공유하는 대칭키를 생성하는 대칭키모듈(MPK); 및

메시지 암호화와 복호화 시 사용되는 복호화 수행객체의 한 쌍의 비대칭키인 공개키와 개인키를 함께 생성되는 공개키모듈(PK)과 개인키모듈(SK)을 포함하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 27

제26항에 있어서,

상기 마스터키벡터(MSK_$\overrightarrow{tp}$) 모듈, 마스터키스칼라모듈(MSK_$\overrightarrow{tv}$) 및 대칭키모듈(MPK)은 객체들을 상호 구분할 수 있는 하나 이상의 식별정보를 매개변수로 하여 키유도함수(KDF)를 통해 키 값들을 생성하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템.

청구항 28

암호키생성기가 식별인자를 이용하여 마스터키를 생성하는 단계;

암호키생성기에서 암호화와 복호화에 필요한 대칭키, 개인키 및 공개키 쌍을 생성하는 단계;

서명 수행객체가 상기 생성된 대칭키와 서명키인 검증 수행객체의 개인키를 수신하여 서명문을 생성하는 단계; 및

검증 수행객체가 상기 대칭키와 검증키인 서명 수행객체의 일회용공개키를 수신하여 서명 수행객체를 통해 생성된 서명문을 검증하고 상기 검증 결과에 따라 원문메시지를 수락하거나 거절하는 단계를 포함하는 것을 특징으로 하는 암호문 서명 및 검증 방법.

청구항 29

제28항에 있어서,

상기 식별인자는,

사용자의 개인정보를 포함하는 사용자 식별인자, 사용자 단말 정보를 포함하는 단말장치 식별인자 및 비밀함수 생성인자 중 적어도 하나를 포함하는 것을 특징으로 하는 암호문 서명 및 검증 방법.

청구항 30

제28항에 있어서,

상기 서명문(M_S)은 서명 함수(S)에 서명키(K_S)를 대입하여 순열연산($M\times K \rightarrow S$)에 의해 아래 수식과 같이 생성되는 것을 특징으로 하고,

$S(M, K_S) = Q^{-1}HQ(M) = M_s$

여기서 $Q^{-1}HQ$는 순열함수 H, Q의 순열연산(left multiplication)이고,

메시지 $M = (m_1 \cdots m_n)$, 키 $K = (k_1 \cdots k_n)$, 서명문 $M_s = (s_1 \cdots s_n)$ 는 각각 메시지집합 $M=\{m_1 \cdots m_n\}$ 상에서의 순열 그룹 G의 원소인 것(M,K,S $\in$G)을 특징으로 하는 암호문 서명 및 검증 방법.

청구항 31

제28항에 있어서,

상기 서명문의 검증은 검증 함수(V)에 검증키(K_V)를 대입하여 순열연산($S \times K \rightarrow S$)에 의해 아래 수식과 같이 생성되는 것을 특징으로 하고,

$V(S, K_v) = GQQ(M_s) = M$

여기서 GQQ는 순열함수 G, Q의 순열연산(left multiplication)이고,

메시지 $M = (m_1 \cdots m_n)$, 키 $K = (k_1 \cdots k_n)$, 서명문 $M_s = (s_1 \cdots s_n)$ 는 각각 메시지집합 $M=\{m_1 \cdots m_n\}$ 상에서의 순열 그룹 G의 원소인 것(M,K,S $\in$G)을 특징으로 하는 암호문 서명 및 검증 방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 순열 그룹 기반의 암호 기술(cryptographic technologies)을 이용한 암호화 방법 및 시스템에 관한 것으로, 보다 상세하게는 암호 생성 객체(이하 “송신자”)와 암호 해제 객체(이하 “수신자”)가 메시지를, 원래의 메시지를 구성하는 메시지 공간과는 다른 공간으로, 이동 또는 확장 등을 통해 새로운 공간으로 변경시키는 비밀함수그룹(secret permutations group)을 서로 다르게, 사전에 또는 필요시 동적으로 사전 지정하거나 임의의 횟수나 시간 또는 특정한 통신 시점마다 매번 다르게 사전 약정한 시간 동안 유효하도록 생성한 후, 송신자와 수신자가 사용할 서로 다른 공간이 상호 연결/매핑 할 수 있도록 하는 유일한 비밀 함수(secret permutation)을 매번 다르게 생성하여 상호 연결된 공간을 통해 안전하게 암호문을 생성/전송/복원하기 위한 방법 및 그를 이용한 시스템에 관한 것이다. 이렇게 함으로써 기존 암호 기술이 매번 동일한 메시지 공간과 동일한 특정 값만 사용함에 따른 비밀 정보가 노출되는 보안상 문제점을 해결하고자 한다.

배 경 기 술

[0003] 1994년 MIT 응용 수학자인 Peter Shor에 의해 양자 컴퓨팅을 이용한 암호화 알고리즘(이하 양자 기반 알고리

즘)을 이용할 경우 소인수분해에 걸리는 시간이 획기적으로 줄어들 수 있다는 것이 증명되자 전세계 보안전문가들은 충격에 빠졌다. Shor 알고리즘에 의하여 현재 지수적 연산을 수행하는 수인수분해 기법과 이산로그문제에 기반한 공개키 암호화가 짧은 시간에 해독이 가능하게 되기 때문이다.

[0004] 양자 기반 알고리즘에는 위에 언급한 Shor 알고리즘과 Grover 알고리즘이 있다. 대칭키 암호화에 영향을 끼치는 Grover 알고리즘에 따르면 대부분의 대칭키 암호화 방식은 암호키를 두배로 늘리는 것으로 기존과 동일한 수준의 보안이 가능하지만, Shor 알고리즘이 구현된 양자컴퓨터가 개발되면 현재 사용하는 공개키 암호화 방식은 더 이상 사용할 수 없게 된다.

[0005] 지금껏 대부분의 비밀은 비대칭 암호화의 형태로 보호되어왔다. 1976년 위트필드 디피(Whitfield Diffie), 마크 헬만(Mark Hellman), 랄프 메클(Ralph Merkle)이 '암호의 새로운 방향'(New Directions in Cryptography)이라는 세미나 논문에서 해당 개념을 공개한 이후의 이야기다. RSA, SSL, TLS, HTTPS를 생각해 보자. 대부분의 웹사이트, 전자 서명 다운로드, 온라인 금융 거래, VPN, 스마트카드, 대부분의 무선 네트워크에 이 개념이 적용된다. 현대의 보안 통신은 전통적인 디지털 컴퓨터가 큰 소수를 포함해 다인자 수식을 쉽게 처리할 수 없다는 점에 기초한다. 그러나 양자 컴퓨팅의 도입으로 이런 보호 장치로 암호화된 비밀이 모두 무효화되게 된다. 실제로 세계의 주요 국가들이 추후에 복호화를 하기 위해 암호화된 네트워크 트래픽을 상당 부분을 기록 및 저장하고 있으며 그 날이 오기만을 기다리고 있다는 주장도 있다.

[0006] 아래 표 1은 현재 많이 사용되고 있는 암호화 기법에 대하여 미치는 영향을 정리하였다.

표 1

[0007]

암호알고리즘	형태	목적	양자 컴퓨터의 영향
AES-256	Symmetric key	Encryption	Secure
SHA-256, SHA-3	-	Hash functions	Secure
RSA	Public key	전자서명, 키설정	No longer secure
ECDSA, ECDH(타원암호)	Public key	전자서명, 키교환	No longer secure
DSA (finite field Cryptography)	Public key	전자서명, 키교환	No longer secure

[0008] 아래 표 2는 현재 사용하는 암호기법에 대하여 양자컴퓨팅 환경에서 보안수준이 어떻게 변하는 지 비교하여 보여준다.

표 2

[0009]

암호기법	키사이즈	효과적인 키강도/보안수준(bits)	
		기존 컴퓨팅	퀀텀 컴퓨팅
RSA-1024	1024	80	0
RSA-2048	2048	112	0
ECC-256	256	128	0
ECC-384	384	256	0
AES-128	128	128	64
AES-256	256	256	128

[0010] 다가오는 양자컴퓨팅 시대에 더이상 안전하지 않은 공개키 암호화 기법으로 인해 발생할 혼란을 대비하기 위해서도 양자컴퓨터가 풀 수 없는 양자내성 암호화 기법이 필요한 실정이다.

[0011] 따라서 이러한 문제를 해결하고자 본 발명에서는 현재 컴퓨팅 환경에서도 효율적으로 작동하고 양자컴퓨팅 환경에서도 안전하게 데이타를 보호하는 양자내성(post-quantum) 암호화 기법 및 시스템을 제안한다.

선행기술문헌

특허문헌

[0013] (특허문헌 0001) 미국 등록특허 US 6,212,279호

(특허문헌 0002) 미국 등록특허 US 6,243,467호

(특허문헌 0003) 미국 등록특허 US 6,782,100호

발명의 내용

해결하려는 과제

[0014] 제안되는 양자저항(post quantum) 암호화 기법은 양자컴퓨팅 환경으로 인해 더이상 안전하지 않은 공개키 암호화 기법을 대체해야 하는 만큼 성능, 보안성, 사용성 면에서 기존의 공개키 방법보다 개선되어야 하며, 양자컴퓨터 뿐아니라 현재 사용하는 컴퓨팅환경에도 적합하여야 한다.

[0015] 따라서 본 발명에서는 첫째, 양자 기반 알고리즘으로 해독되는 기존의 복잡한 수학연산에 기반한 방법을 사용하지 않고 순열그룹(permutation group)에 기반하여 각각 순열형태인 대칭키와 비대칭키를 동시에 합성(composition)하는 연산 방식을 사용하고 키배열 변환 방식으로 다차원 확장을 통해 복잡한 수학연산을 사용하지 않고 키공간을 확장하여 복잡도를 높이고 복잡한 수학연산을 수행하지 않고 대칭키 암호화 기법의 SPN(substitution-permutation-network)에서 처리하는 방식과 같은 값의 대치, 변환 등 연산을 수행하여 빠른 암호화 처리를 가능하게 하였다.

[0016] 둘째, 기존 방식이 도 1과 같이 한 번 생성된 키값을 사용하여 고정된 키함수의 수학연산을 하는 방식을 사용하는데 비해 본 발명에서는 송신자가 메시지를 전송할 때마다 도 3과 같이 수신자가 선택한 다차원 공간의 키함수들을 사용하여 매번 다른 공간의 키함수들을 통해 매번 다른 키값들을 생성하여 사용하므로 양자컴퓨터와 같은 컴퓨팅 파워의 향상 등 계산능력이 향상된다 하더라도 안전한 보안성을 제공한다.

[0017] 셋째, 아래 표 3과 같이 RSA 등 기존의 공개키에 비하여 더 작은 키사이즈로 높은 보안성을 보여준다. 따라서 기존의 컴퓨터에 사용하는 공개키를 무리없이 대체 가능하다.

표 3

[0018]

키사이즈(bits)	효과적인키강도/보안수준(bits)	
	기존 컴퓨팅	양자 컴퓨팅
256(|M|=32)	128(AES)	64
512(|M|=64)	256(AES)	128
1024(|M|=128)	768(AES)	384
2048(|M|=256)	1536(AES)	768

과제의 해결 수단

[0020] 본 발명은 메시지를 암호화 하는 암호화 수행객체와, 상기 암호화된 메시지를 복호화하는 복호화 수행객체를 포함하며, 상기 암호화 수행객체 및 복호화 수행객체는 순열그룹에 기반하여 순열형태인 대칭키와 비대칭키 중 공개키를 동시에 합성하여 메시지를 암호화 한 후 이를 다시 복호화 시에는 순열 그룹에 기반하여 순열 형태인 대칭키와 비대칭키 중 개인키를 사용하여 원문을 복호화하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 암호화시스템을 개시한다.

[0021] 본 발명의 일 실시예에 따르면, 상기 암호화 수행객체 및 복호화 수행객체는 암호화키(K_e)를 이용하여 메시지를 암호화 하는 암호화기(ENC)와, 메시지를 복호화키(K_d)를 이용하여 복호화 하는 복호화기(DEC)와, 암호키생성기(MKG)를 포함한다.

[0022] 본 발명의 일 실시예에 따르면, 상기 암호화기는 메시지 입력을 처리하는 입력큐와, 일회용 공개키를 이용하여 순열연산을 통해 암호문을 생성하는 GA 연산기와, 상기 생성된 암호문의 출력을 처리하는 출력큐를 포함한다.

[0023] 본 발명의 일 실시예에 따르면, 상기 GA 연산기는 상기 암호키생성기(MKG)로부터 대칭키(Q_{AB})와 복호화 수행객체의 공개키(G_B)를 제공받아 순열연산을 통해 암호문을 생성한다.

[0024] 본 발명의 일 실시예에 따르면, 상기 순열연산은 $Q_{AB}^{-1}G_BQ_{AB}(M) = C$ 수식에 의하여 이루어지며, Q_{AB}는 암호화 수행객체와 복호화 수행객체의 대칭키, G_B 는 복호화 수행객체의 공개키, M은 메시지 공간, C는 암호문 공간인 것을 특징으로 한다.

[0025] 본 발명의 일 실시예에 따르면, 상기 복호화기는 암호문 입력을 처리하는 입력큐와, 일회용개인키를 이용하여 순열연산을 통해 원문메시지를 복원하는 GA 연산기와, 상기 복원된 원문메시지의 출력을 처리하는 출력큐를 포함한다.

[0026] 본 발명의 일 실시예에 따르면, 상기 GA 연산기는 상기 암호키생성기(MKG)로부터 대칭키(Q_{AB})와 복호화 수행객체의 개인키(H_B)를 제공받아 순열연산을 통해 원문 메시지를 복원한다.

[0027] 본 발명의 일 실시예에 따르면, 상기 순열연산은 $H_BQ_{AB}Q_{AB}(C) = M$ 수식에 의하여 이루어지며, Q_{AB}는 암호화 수행객체와 복호화 수행객체의 대칭키, H_B 는 복호화 수행객체의 개인키, M은 메시지 공간, C는 암호문 공간인 것을 특징으로 한다.

[0028] 본 발명의 일 실시예에 따르면, 상기 암호키생성기는 복수의 매개변수를 사용한 키유도함수 KDF(key derivation function)를 통해 일회용 유사난수를 생성하는 난수생성기(PRNG)와, 상기 키유도함수(KDF)를 통해 일회용 유사난수순열(PRP)를 생성하여 키생성모듈에 제공하는 순열생성기를 포함한다.

[0029] 본 발명의 일 실시예에 따르면, 상기 키생성모듈은 복호화 수행객체의 비밀함수그룹(secret permutation group)을 나타내는 벡터함수인 마스터키벡터(MSK_$\overrightarrow{tp}$) 모듈과, 복호화 단계에 사용될 비밀함수(secret permutation)를 결정하는 마스터키스칼라모듈(MSK_$\overrightarrow{tv}$)과, 암호화 수행객체와 복호화 수행객체 간 공유하는 대칭키를 생성하는 대칭키모듈(MPK)과, 메시지 암호화와 복호화 시 사용되는 공개키와 개인키를 함께 생성되는 공개키모듈(PK)과 개인키모듈(SK)을 포함한다.

[0030] 본 발명의 일 실시예에 따르면, 상기 마스터키벡터(MSK_$\overrightarrow{tp}$) 모듈, 마스터키스칼라모듈(MSK_$\overrightarrow{tv}$) 및 대칭키모듈(MPK)은 복수의 고유 개인 식별정보를 매개변수로 하여 키유도함수(KDF)를 통해 같은 값이나 상호 약정한 방식에 따라 다른 키 값들을 생성한다.

[0031] 또한, 본 발명은 암호키생성기가 식별인자를 이용하여 마스터키를 생성하는 단계와, 암호키생성기에서 암호화와 복호화에 필요한 대칭키, 개인키 및 공개키 쌍을 생성하는 단계와, 암호화 수행객체가 대킹키와 암호화키인 복호화 수행객체의 공개키를 수신하여 암호문을 생성하는 단계와, 복호화 수행객체가 암호키생성기를 통해 대칭키와 복호화 수행객체의 개인키를 통해 생성된 암호문을 복원하는 단계를 포함하는 암호화 및 복호화 방법을 개시한다.

[0032] 본 발명의 일 실시예에 따르면, 상기 식별인자는 사용자의 개인정보를 포함하는 사용자 식별인자, 사용자 단말정보를 포함하는 단말장치 식별인자 및 비밀함수 생성인자 중 적어도 하나를 포함한다.

[0033] 본 발명의 일 실시예에 따르면, 상기 암호문은 상기 생성된 대칭키 및 복호화 수행객체의 공개키를 이용하여 GA 연산기에 연산시켜 생성한다.

[0034] 본 발명의 일 실시예에 따르면, 상기 암호문(C)은 암호화 함수(E)에 암호화키(K_e)를 대입하여 순열연산($M \times K \rightarrow C$)에 의해 아래 수식과 같이 생성되는 것을 특징으로 하고,

[0035] $E(M, K_e) = Q^{-1}GQ(M) = C$

[0036] 여기서 $Q^{-1}GQ$ 는 순열함수 G, Q의 순열연산(left multiplication)이고, 메시지 $M = (m_1 \cdots m_n)$, 키 $K = (k_1 \cdots k_n)$, 암호문 $C = (c_1 \cdots c_n)$는 메시지집합 $M = \{m_1 \cdots m_n\}$ 상에서의 순열그룹 G의 원소인 것($M,K,C \in G$)을 특징으로 한다.

[0037] 본 발명의 일 실시예에 따르면, 상기 암호문의 복원은 복호화 함수(D)에 복호화키(K_d)를 대입하여 순열연산(C

× K → M)에 의해 아래 수식과 같이 생성되는 것을 특징으로 하고,

[0038] $D(C, K_d) = HQQ(C) = M$

[0039] 여기서 HQQ 는 순열함수 H, Q의 순열연산(left multiplication)이고, 메시지 $M = (m_1 \cdots m_n)$, 키 $K = (k_1 \cdots k_n)$, 암호문 $C = (c_1 \cdots c_n)$는 메시지집합 $M = \{m_1 \cdots m_n\}$ 상에서의 순열그룹 G의 원소인 것(M,K,C ∈ G)을 특징으로 한다.

[0040] 또한, 본 발명은 암호문 작성시 서명문을 생성하는 서명 수행객체과, 상기 암호문을 원문 메시지로 복호화 하기 위하여 상기 서명문을 검증하도록 형성되는 검증 수행객체를 포함하며, 순열 그룹에 기반하여 순열 형태인 대칭키와 비대칭키를 이용하여 서명문을 생성하고 검증하는 것을 특징으로 하는 순열그룹 기반의 암호화 기술을 적용한 서명 검증 시스템을 제공한다.

[0041] 본 발명의 일 실시예에 따르면, 상기 서명 수행객체 및 검증 수행객체는 순열연산을 통해 서명문을 생성하는 서명기와, 순열연산을 통해 상기 서명문을 검증하는 검증기와, 암호키생성기(MKG)를 포함한다.

[0042] 본 발명의 일 실시예에 따르면, 상기 서명기는 메시지 입력을 처리하는 입력큐와, 일회용개인키를 이용하여 순열연산을 통해 서명문을 생성하는 GA 연산기와, 상기 생성된 서명문의 출력을 처리하는 출력큐를 포함한다.

[0043] 본 발명의 일 실시예에 따르면, 상기 서명기의 GA연산기는 상기 암호키생성기(MKG)로부터 대칭키(Q_{AB})와 서명 수행객체의 개인키(H_A)를 제공받아 순열연산을 통해 서명문을 생성한다.

[0044] 본 발명의 일 실시예에 따르면, 상기 서명기의 순열연산은

[0045] $Q_{AB}^{-1} H_A Q_{AB}(M) = M_s$ 수식에 의하여 이루어지며, Q_{AB}는 서명 수행객체와 검증 수행객체의 대칭키, H_A는 서명 수행객체의 개인키, M은 메시지($m_1 \cdots m_n$), M_S는 서명문($s_1 \cdots s_n$)인 것을 특징으로 한다.

[0046] 본 발명의 일 실시예에 따르면, 상기 검증기는 서명문을 입력받아 처리하는 입력큐와, 일회용공개키를 이용하여 순열연산을 통해 서명문을 검증하여 수락된 원문메시지를 생성하는 GA 연산기와, 상기 수락된 원문메시지의 출력을 처리하는 출력큐를 포함한다.

[0047] 본 발명의 일 실시예에 따르면, 상기 검증기의 GA연산기는 상기 암호키생성기(MKG)로부터 대칭키(Q_{AB})와 서명 수행객체의 공개키(G_A)를 제공받아 순열연산을 통해 서명문을 검증하고 수락 또는 거절 여부를 확인한다.

[0048] 본 발명의 일 실시예에 따르면, 상기 검증기의 순열연산은 $G_A Q_{AB} Q_{AB}(M_s) = M$ 수식에 의하여 이루어지며, Q_{AB}는 서명 수행객체와 검증 수행객체의 대칭키, G_A는 서명 수행객체의 공개키, M은 메시지($m_1 \cdots m_n$), M_S는 서명문($s_1 \cdots s_n$)인 것을 특징으로 한다.

[0049] 본 발명의 일 실시예에 따르면, 상기 암호키생성기는 복수의 매개변수를 사용한 키유도함수 KDF(key derivation function)를 통해 일회용 유사난수를 생성하는 난수생성기(PRNG)와, 상기 키유도함수(KDF)를 통해 일회용 유사난수순열(PRP)를 생성하여 키생성모듈에 제공하는 순열생성기를 포함한다.

[0050] 본 발명의 일 실시예에 따르면, 상기 키생성모듈은 서명 수행객체의 비밀함수그룹(secret permutation group)을 나타내는 벡터함수인 마스터키벡터($MSK\_\vec{tp}$) 모듈과, 복호화할 때 사용될 비밀함수(secret permutation)를 결정하는 마스터키스칼라모듈($MSK\_\vec{tv}$)과, 서명 수행객체와 검증 수행객체 간 공유하는 대칭키를 생성하는 대칭키모듈(MPK)과, 메시지 암호화와 복호화 시 사용되는 복호화 수행객체의 한 쌍의 비대칭키인 공개키와 개인키를 함께 생성되는 공개키모듈(PK)과 개인키모듈(SK)을 포함한다.

[0051] 본 발명의 일 실시예에 따르면, 상기 마스터키벡터($MSK\_\vec{tp}$) 모듈, 마스터키스칼라모듈($MSK\_\vec{tv}$) 및 대칭키모듈(MPK)은 객체들을 상호 구분할 수 있는 하나 이상의 식별정보를 매개변수로 하여 키유도함수(KDF)를 통해 키 값들을 생성한다.

[0052] 또한, 본 발명은 암호키생성기가 식별인자를 이용하여 마스터키를 생성하는 단계와, 암호키생성기에서 암호화와 복호화에 필요한 대칭키, 개인키 및 공개키 쌍을 생성하는 단계와, 서명 수행객체가 상기 생성된 대칭키와 서명키인 검증 수행객체의 개인키를 수신하여 서명문을 생성하는 단계와, 검증 수행객체가 상기 대칭키와 검증키인 서명 수행객체의 일회용공개키를 수신하여 서명 수행객체를 통해 생성된 서명문을 검증하고 상기 검증 결과에 따라 원문메시지를 수락하거나 거절하는 단계를 포함하는 암호문 서명 및 검증 방법을 개시한다.

[0053] 본 발명의 일 실시예에 따르면, 상기 식별인자는 사용자의 개인정보를 포함하는 사용자 식별인자, 사용자 단말정보를 포함하는 단말장치 식별인자 및 비밀함수 생성인자 중 적어도 하나를 포함한다.

[0054] 본 발명의 일 실시예에 따르면, 상기 서명문(M_S)은 서명 함수(S)에 서명키(K_S)를 대입하여 순열연산($M \times K \rightarrow S$)에 의해 아래 수식과 같이 생성되는 것을 특징으로 하고,

[0055] $S(M, K_S) = Q^{-1}HQ(M) = M_s$

[0056] 여기서 $Q^{-1}HQ$는 순열함수 H, Q의 순열연산(left multiplication)이고, 메시지 $M = (m_1 \cdots m_n)$, 키 $K = (k_1 \cdots k_n)$, 서명문 $M_s = (s_1 \cdots s_n)$ 는 각각 메시지집합 $M=\{m_1 \cdots m_n\}$ 상에서의 순열그룹 G의 원소인 것($M,K,S \in G$)을 특징으로 한다.

[0057] 본 발명의 일 실시예에 따르면, 상기 서명문의 검증은 검증 함수(V)에 검증키(K_V)를 대입하여 순열연산($S \times K \rightarrow S$)에 의해 아래 수식과 같이 생성되는 것을 특징으로 하고,

[0058] $V(S, K_v) = GQQ(M_s) = M$

[0059] 여기서 GQQ는 순열함수 G, Q의 순열연산(left multiplication)이고, 메시지 $M = (m_1 \cdots m_n)$, 키 $K = (k_1 \cdots k_n)$, 서명문 $M_s = (s_1 \cdots s_n)$ 는 각각 메시지집합 $M=\{m_1 \cdots m_n\}$ 상에서의 순열그룹 G의 원소인 것($M,K,S \in G$)을 특징으로 한다.

발명의 효과

[0061] 본 발명에서 제시한 방법은 양자 컴퓨팅 환경하에서 안전하지 않은 문제로 더이상 사용하지 못하는 수학적 기반의 공개키 암호 시스템을 대체 가능하게 하여 도래하는 양자컴퓨팅 시대에 데이타 보안 문제로 야기되는 혼란을 예방할 수 있도록 한다.

[0062] 본 발명에 따르면, 수신자의 개인키와 대칭키는 송수신 접속이 있을 경우에 한 번만 생성하는 일회용 키이므로 허가받지 않은 참여자가 관련정보를 탈취한다 하더라도 같은 키를 생성할 수가 없으므로 탈취한 암호문에 대해 복화화가 불가능하고, 중간자 공격 등 해킹공격에도 안전하다.

[0063] 또한, 본 발명에 따른 서명 알고리즘을 통해, 생성된 암호문을 악의적인 공격자가 탈취하여 조작하였는지 여부를 판단할 수 있고 또한 자신이 보낸 메시지를 부인하지 못하게 하는 부인방지(non-repudiation) 기능을 제공한다.

[0064] 한편, 현재 컴퓨팅 환경하에서도 복잡한 수학처리 과정을 거치지 않고, 수학적인 암호화 시스템 대비 충분히 안전하고 큰 암호화 키공간의 확보가 가능한 치환방식(S-box) 방식의 대칭키 장점을 승계하면서도, 기존의 치환기반(S-box) 대칭키 암호화 시스템이 가지던 키교환 문제점과 암호문 유출로 인한 보안의 문제점 들을 상기에서 제시한 발명의 효과들을 통해 해결함으로서, 기존 시스템은 물론 저용량/저속/저렴한 운영이라는 요건들을 충족해야 하는 사물인터넷 기기(IoT Devices)나 Cloud와 같은 새로운 컴퓨팅환경하에서 효율적이고 안전한 암호화 시스템 및 암호 통신 시스템의 구축할 수 있다.

도면의 간단한 설명

[0066] 도 1은 RSA 암호시스템의 cipher(K, E, C) space 예시를 보여주는 개념도.

도 2는 순열연산(group action) 예시를 보여주는 개념도.

도 3은 발명시스템의 cipher(K, E, C) space 예시를 보여주는 개념도.

도 4는 암호 통신시스템의 개념도.

도 5는 암호/복호를 위한 시스템의 구조도.

도 6은 암호화기의 구조도.

도 7은 복호화기의 구조도.

도 8은 암호키 생성기의 개념도.

도 9는 암호 연산 예시를 보여주는 개념도.

도 10은 암호/복호화 절차 흐름도.

도 11은 암호/복호화 단계 1 설정 예시를 보여주는 개념도.

도 12는 암호화/복호화 단계 2 키 생성 예시를 보여주는 개념도.

도 13은 암호화 단계 3 암호문 생성을 보여주는 개념도.

도 14는 복호화 단계 3 메시지 생성을 보여주는 개념도.

도 15는 서명/검증을 위한 시스템의 구조도.

도 16은 서명기의 구조도.

도 17은 검증기의 구조도.

도 18는 서명/검증 절차 흐름도.

도 19은 서명/검증 단계 1 설정 예시를 보여주는 개념도.

도 20은 서명/검증 단계 2 키생성을 보여주는 개념도.

도 21는 서명 단계 4 서명(signature) 생성을 보여주는 개념도.

도 22은 검증 단계 4 검증 메시지 수락/거절을 보여주는 개념도.

도 23는 서명된 메시지의 암호화기시스템의 구조도.

도 24는 서명된 메시지의 복호화기시스템의 구조도.

도 25은 서명된 메시지의 암호화를 보여주는 개념도.

도 26은 서명된 메시지의 암복호화 및 검증 절차 흐름도.

도 27은 서명된 암호문의 복호화를 보여주는 개념도.

발명을 실시하기 위한 구체적인 내용

[0067] 이하, 첨부된 도면을 참조하여 본 명세서에 개시된 실시 예를 상세히 설명하되, 도면 부호에 관계없이 동일하거나 유사한 구성요소는 동일한 참조 번호를 부여하고 이에 대한 중복되는 설명은 생략하기로 한다.

[0068] 이하의 설명에서 사용되는 구성 요소에 대한 접미사 "모듈" 및 "부"는 명세서 작성의 용이함만이 고려되어 부여되거나 혼용되는 것으로서, 그 자체로 서로 구별되는 의미 또는 역할을 갖는 것은 아니다.

[0069] 또한, 본 명세서에 개시된 실시 예를 설명함에 있어서 관련된 공지 기술에 대한 구체적인 설명이 본 명세서에 개시된 실시 예의 요지를 흐릴 수 있다고 판단되는 경우 그 상세한 설명을 생략한다.

[0070] 또한, 첨부된 도면은 본 명세서에 개시된 실시 예를 쉽게 이해할 수 있도록 하기 위한 것일 뿐, 첨부된 도면에 의해 본 명세서에 개시된 기술적 사상이 제한되지 않으며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.

[0071] 제1, 제2 등과 같이 서수를 포함하는 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 상기 구성요소들은 상기 용어들에 의해 한정되지는 않는다.

[0072] 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다.

[0073] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다.

[0074] 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다.

[0075] 본 명세서에서, "포함한다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0076] 본 명세서에서 설명하는 수행 객체들(암호화 수행객체, 복호화 수행객체, 서명 수행객체, 검증 수행객체 등)과 수행객체를 구성하는 구성요소들(암호화기, 복호화기, 서명기, 검증기, 암호키생성기 등)은 각각 물리적으로 구분되는 구조를 가질 수도 있고 기능적으로만 구분되어 있을 수 있다.

[0077] 기능적으로만 구분되는 경우, 이러한 수행 객체들과 구성요소들은 하나의 제어부에 포함될 수 있다.

[0078] 상기 제어부는 단일 시스템이나 클라우드 서비스와 같은 분산 응용 프로그램 환경 내 특정 기능을 수행하는 API, 특정 기능을 수행하는 모듈, 컴포넌트(component), 칩, 단말 등의 하드웨어 또는 어플리케이션, 프로그램 등의 소프트웨어를 포함할 수 있다.

[0080] **I. 용어의 정의**

[0081] a) 정보의 표현 및 처리 방법

[0082] 컴퓨터 또는 통신시스템을 통해 보내고자 하는 정보는 숫자, 문자, 그림, 동영상, 소프트웨어 등 다양한 형태를 띄고 있으나 시스템 내에서는 binary, 즉 bit로 구성된 byte단위로 다루어진다. 이것은 ASCII, UNICODE 등 코드 형태로 변환되어 시스템 내의 응용프로그램이 인식하게 되고 이를 통해 사람들은 다시 숫자, 문자, 그림 등 정보의 형태로 전달 받는다.

[0083] 통상적으로 메시지라 함은 사람들이 컴퓨터 또는 통신시스템을 통해 상대방에게 전달하고자 하는 정보를 말하며, 이는 시스템에서는 앞서 언급한 바와 같이 byte 등 시스템 내 메시지단위(message unit)로 다루어지고, 시스템 내에서 모든 정보는 처리 가능한 하나의 시스템 내 메시지단위(message unit)의 열로 변환된다.

[0084] 즉, 전달하고자 하는 메시지는 메시지단위(message unit)에 의해 표현할 수 있고 메시지 단위에 의해 표현 가능한 경우를 나열한 것을 메시지 집합 M이라 한다.

[0085] 예를 들어, 메시지 단위가 bit인 경우 M = {0,1}이 되고 메시지는 00110101와 같이 메시지단위의 열로 표현될 수 있고, 메시지 단위가 byte인 경우 M = {0,1, … 255} 이고(십진수 표현으로 나타내면) 메시지는 64 68 72 82 와 같이 나타낼 수 있다.

[0086] 이 때 이를 수학적으로 표현하면, 메시지 집합 M은 $M = \{m_1, \cdots, m_i, \cdots, m_n\}$ 과 같이 나타내며, M의 원소는 m_i이고, $1 \leq i \leq n$이며, 집합 M의 원소의 갯수는 $|M| = n$과 같이 나타낼 수 있다.

[0087] 메시지 집합 M의 각 원소를 순서대로 나열하고 그 순서에 따라 나열된 원소의 인덱스를 집합으로 나타내면 $I_M = \{ 1, \cdots, i, \cdots, n \}$ 이 되고 이를 메시지 집합 M에 대한 인덱스 집합이라 말하며, 인덱스 집합의 i번째 원소는 $I_M(i) = m_i$ 과 같다.

[0088] 이 때 메시지의 인덱스 집합은 $I_M : I \rightarrow M$ 과 같이 표현하고 인덱스 집합 I에서 메시지 집합 M으로 가는 사상(morphism)이 존재한다고 말한다.

[0089] 순열(permutation) P = (p_1, …, p_i, …, p_n) 는 n개의 자연수(Z)로 이루어진 집합 S = {1, 2, …, n}의 permutation(순열)이라 하며, $1 \le p_i \le n$ 이다.

[0090] 예를 들면, 순열 $\sigma=\begin{pmatrix}1&2&3&4&5\\2&4&2&5&1\end{pmatrix}$ 와 같이 표현할 수 있고, 이는 집합 S= {1, 2, 3, 4, 5}의 permutation으로 σ : S → S 과 같이 전단사 함수(bijection)에 의해 표현되며, 즉 함수이다.

[0091] 다시 말하면, σ(1)=3, σ(2)=4, …, σ(5)=1 과 같으며, 일반적으로 집합 S = {x_1,x_2,…,x_n} 의 permutation은 $\sigma=\begin{pmatrix}x_1&\cdots&x_n\\ \sigma(x_1)&\cdots&\sigma(x_n)\end{pmatrix}$ 와 같이 표현된다.

[0092] 순열은 함수적으로 표현하면, 도메인 X에서 코도메인 Y로 가는 전단사 함수 F : X → Y 에서 X와 Y의 대응관계를 나타내며, 집합의 의미로 표현하면, 임의의 집합의 원소에 대한 배열순서를 나타낸다.

[0093] n개의 원소로 이루어진 임의의 집합의 순열(permutation)은 n개의 숫자 또는 문자의 순열(permutation)이라고도 말할 수 있으며 앞서 언급한 메시지 집합 M에 대해서도 마찬가지로 적용된다.

[0094] 따라서 모든 메시지는 메시지 집합 M의 순열에 의해 표현가능하다.

[0095] 순열(permutation)은 임의의 집합의 각 원소들의 배열에 따라 다른 순열(permutaion)이 되고, 이 모든 경우의 순열들이 모여 순열그룹(Group)을 형성한다. 즉 순열그룹(Permutaion Group)은 임의의 집합의 모든 경우의 순열(premutation)을 원소로 이루어진 집합이다.

[0096] 순열그룹(permutation group) G = { σ | σ : S → S , σ는 S = {x_1,x_2,…,x_n}의 permutation }이며, 집합 M = {1,2,…n}의 모든 순열들의 순열그룹(permutation group)은 symmetric group Sym(M)이 된다.(모든 전단사 함수는 symmetric group임)

[0097] 즉, M = {1,2, …,n} 이면, n개의 문자에 대한 Sym(M)은 S_n 으로 나타낸다.

[0098] 순열 σ, π ∈ G(순열그룹) 일 때, 그룹연산(composition of permutation)의 결과 또한 순열이 되고 결과의 순열 역시 G의 원소가 된다. 즉, 순열그룹 G는 그룹연산에 대하여 닫혀 있다.

[0099] G를 구성하는 순열(permutation)의 수는 |G| = n! 개 이다.

[0100] 순열연산(Group Action)은 순열그룹(permutation group) G의 각 원소(P: permutation)들이 집합(S)의 원소들을 배열하는 방법을 말하며, 일종의 함수와 같이 작용한다. 다시 말하면, 집합 S의 원소에 순열 P의 순열연산(Group Action)를 적용하면 집합 S의 원소가 순열 P에 의해 재배열되는 것을 의미한다. 즉, 집합 S의 원소 배열 순서를 바꾸는 연산을 순열연산(group action)이라고 한다.

[0101] (즉, 집합 S에 대해 순열 P의 방법에 의해 S의 원소를 재배열한다는 것이고 즉 S의 인덱스 집합 I_S이 바뀐다)

[0102] G : Permutation Group, M : non-empty set 일 때, 집합 M에 대한 순열그룹 G의 순열연산(group action)은 아래의 3가지 성질을 만족하는 함수 f : G × M → M 이다.

[0103] * 집합 M에 속하는 모든 원소 x에 대하여 f(1, x) = x (group G의 항등원은 1)

[0104] * f(x, y) = 1을 만족하는 $y = x^{-1}$, x, y ∈ G인 x의 역원이 존재

[0105] * G에 속하는 모든 permutation g, h와 M에 있는 모든 원소 x에 대하여f(g, f(h, x)) = f(gh, x) (결합법칙 성립, left multiplication)

[0107] 도 2에서 순열 연산의 예시를 보여준다.

[0108] G의 degree 는 G를 구성하는 집합 M의 원소의 갯수 |M|이고, G의 order는 G의 원소의 갯수(cardinality)는 |G| 이다. 즉, n개의 원소로 된 집합 M에 대한 degree of group G 는 n 이고, order of group G 는 n! 이다.

[0110] b) 암호학 및 발명시스템

[0111] PRNG(Pseudo Random Number Generator) : 난수를 흉내내기 위해 알고리즘을 통해 생성하는 난수 값을 유사난수라 하고, 이때 유사난수를 생성하는 알고리즘을 유사난수 생성기(pseudorandom number generator, PRNG)라고 한다. 이는 다음과 같은 함수 F : X → Y over (X,Y)와 같이 표현할 수 있다. 임의의 입력 값 X에 대해서 임의의 유사난수값 Y가 발생한다.

[0112] PRF(Pseudo Random Function) : PRNG를 기반으로 유도된 함수로 임의의 입력값을 받아 항상 유사난수 수열을 발생한다.(유사난수 함수) 이는 다음과 같은 함수 F : K × X → Y over (k,X,Y)와 같이 표현한다.

[0113] PRP(Pseudo Random Permutaion) : PRF와 유사한 방식으로 유사난수 수열을 생성하나 항상 같은 도메인으로 작용하는 일대일 사상이 존재하고 효율적인 역함수 D(k,X)가 존재한다. PRNG로부터 생성된 난수로부터 PRP로 부터 생성된 수열을 구별할 수 없으면 secure PRP라고 한다. 또한, 충분히 큰 X에서 정의된 secure PRP는 secure PRF이다. (유사난수 순열)

[0114] 이는 다음과 같은 함수 E : K × X → X over (k,X) 와 같이 표현한다.

[0115] TDF(Trapdoor Function) : 트랩도어 함수(trapdoor function, 비밀통로 일방향함수)는 일방향함수의 한 종류이다. 보통 일방향함수처럼 함수의 역을 구하는 것은 어렵지만, 트랩도어라고 부르는 특수한 정보가 있으면 쉽게 역을 구할 수 있는 함수이다. 트랩도어 함수를 수학적으로 정의하면 다음과 같다. 어떤 비밀값 y가 있어서, 어떤 x에 대해서 y가 없을 때는 f(x)를 구하기 어렵지만 y가 주어진다면 f(x)에서 x 값을 쉽게 찾을 수 있다면 함수 f는 트랩도어 함수이다.

[0116] 암호(Cipher) = (G, E, D), 암호공간(cipher space)= (k, M, C) : 암호(cipher)는 암호화 및 복호화를 수행하는 알고리즘이며 암호공간(K, M, C)상에서 작용하는 일종의 함수와 같다. 암호(cipher)는 G, E, D 등 세개의 알고리즘(함수)으로 구성되어 있다. 각각은 다음과 같은 약어를 나타낸다.

[0117] G : 키생성 함수

[0118] E : 암호화(Encryption) 함수

[0119] D : 복호화(Decryption) 함수

[0120] K : 키공간(Key Space)

[0121] M : 메시지공간(Message Space)

[0122] C : 암호문공간(Ciphertext Space)

[0123] MKG(Magic Key Generator)는 암호키생성기로 사용자가 암호화/복호화를 위해 필요한 사용자 식별 및 등록, 키생성, 배포 등을 처리하기 위한 키관리 장치를 말한다. 암호화기나 복호화기와 같은 시스템 내에 설치할 수도 있고 다른 제 3의 시스템에 설치하여 연동할 수도 있다. MKG에 대한 접속은 허용된 참여자만이 할 수 있도록 사용자 인증을 통해 안전한 정보채널을 보장한다.

[0124] 비밀함수그룹 SPG(Secret Permutation Group)는 메시지 집합 M상의 모든 순열그룹(permutation group) G의 subset을 말하며, 이 subset을 형성하는 각 순열(permutation)들을 비밀함수후보 SPC(Secret Permutation Candidates)라 부르며, 이 때 비밀함수후보중에서 특별히 지정된 한 개의 후보를 비밀함수 SP(Secret Permutation)라 한다. 도 3에서 SPG, SP에 대한 예시를 보여준다.

[0126] **II.시스템 구조**

[0127] 도 4는 발명시스템의 일 실시예를 보여주는 개념도이다. 이 시스템은 암호문을 전송하는 통신채널과 이에 연결된 두개의 단말을 포함하고 각 단말은 암복호화에 관련된 암호화키 K_e, 복호화키 K_d를 갖고 있다. 통신채널은 통상적인 전송케이블과 전송장치를 포함하고 도 4에서는 일방향 통신의 예를 보여주나 같은 방법으로 반대방향으로도 작동 가능하여 양방향통신이 가능하다.

[0129] 1. 암호/복호 시스템

[0130] 도 4의 송신 및 수신의 각 단말은 도 5와 같이 암호화기(ENC), 복호화기(DEC) 그리고 암호키생성기(MKG)로 구성되어 있다.

[0131] 암호화기(ENC)는 도 6과 같이 메시지 입력을 처리하는 입력큐와 본 발명의 일 실시예에 따른 알고리즘을 통해 일회용 공개키를 이용하여 순열연산(Group Action)을 통해 암호문을 생성하는 GA 연산기, 그리고 생성된 암호문의 출력을 처리하는 출력큐가 있다.

[0132] GA 연산기는 메시지를 입력으로 암호키생성기(MKG)로 부터 송수신 단말의 대칭키(Q_{AB})와 수신 단말의 공개키(G_B)를 제공받아 순열연산(GA 연산)을 통해 암호문을 생성한다. GA 연산기에서 처리하는 순열연산은 $Q_{AB}^{-1}G_BQ_{AB}(M) = C$ 이다.

[0133] 한편, 다른 일 실시예에서 암호복잡도에는 큰 영향은 없으나 송신단말에 입력된 메시지가 중복된 문자열을 포함하는 경우를 제거하기 위해 확산함수(diffusion function) F(x)를 XOR 연산기를 통해 전처리하고 암호화기의 메시지큐는 전처리된 메시지를 입력받아 암호문을 생성할 수 있다.

[0134] 복호화기(DEC)는 도 7과 같이 암호문 입력을 처리하는 입력큐와 본 발명의 일 실시예에 따른 알고리즘을 통해 일회용개인키를 이용하여 순열연산(Group Action)을 통해 원문메시지를 복원하는 GA 연산기, 그리고 복원된 원문메시지의 출력을 처리하는 출력큐가 있다.

[0135] GA 연산기는 암호문을 입력으로 암호키생성기(MKG)로 부터 송수신 단말의 대칭키(Q_{AB})와 수신단말의 개인키(H_B)를 제공받아 순열연산(GA 연산)을 통해 원문메시지을 복원한다.

[0136] GA 연산기에서 처리하는 순열연산은 $H_BQ_{AB}Q_{AB}(C) = M$ 이다.

[0137] 한편, 다른 일 실시예에서는 송신단말에 확산함수(diffusion function)을 적용한 경우 복호화기에서 복원된 메시지를 송신단말에 적용된 동일한 확산함수(diffusion function) F(x)를 XOR 연산기를 통해 후처리하여 원문메시지를 복원할 수 있다.

[0138] 암호키생성기(MKG)는 도 8에서 보는 바와 같이 난수생성기(PRNG), 순열생성기(permutation generator), 복수의 키생성모듈들(MSK_$\overrightarrow{tv}$, MSK_$\overrightarrow{tp}$, MPK, SK, PK) 그리고 순열연산기(GA Operator) 등으로 구성된다.

[0139] 난수생성기(PRNG)는 사전에 등록된 송수신 참여자만이 알 수 있는 참여자 고유의 개인식별자(ID), device ID, 이벤트, 시간 등 복수의 매개변수(parameter)를 사용한 키유도함수 KDF(key driven function)를 통해 일회용 유사난수를 생성한다. 생성한 난수는 각각 순열생성기와 키생성모듈로 제공된다.

[0140] 순열생성기(permutation generator)는 난수생성기와 고유의 키유도함수(KDF)를 통해 일회용 유사난수순열(PRP)를 생성한다. 생성한 난수수열은 각 키생성모듈로 제공된다.

[0141] 키생성모듈들은 수신자의 비밀함수그룹(Secret Permutation Group)을 나타내는 벡터함수인 마스터키벡터(MSK_$\overrightarrow{tp}$)모듈, 수신할 비밀함수(Secret Permutation)를 결정하는 마스터키스칼라모듈(MSK_$\overrightarrow{tv}$), 송신자 및 수신자만이 공유하는 대칭키를 생성하는 대칭키모듈(MPK), 송수신 이벤트마다 쌍으로 생성하는 공개키모듈(PK) 및 개인키모듈(SK) 등이 있다. 또한, 키생성모듈중 마스터 키모듈과 대칭키모듈들은 복수(multiple)의 고유 개인식별 정보를 매개변수로 키유도함수(KDF)를 통해 난수발생기 또는 순열생성기를 통해 매번 다른 키 값들을 생성하고 내부의 키보관소에 키를 보관하고 개인키 및 공개키 생성을 위해 개인키모듈과 공개키모듈로 생성된 키값을 배포한다. 또한 키를 요청하는 해당 단말의 암호화기와 복호화기에 배포한다.

[0142] 개인키모듈은 개인키를 생성한다. 개인키는 마스터키 스칼라모듈로 부터 생성된 난수값들을 마스터키벡터모듈이 지정한 위치에 먼저 배열하고 순열생성기에서 제공된 난수수열 들을 나머지 위치에 배열하여 생성한다.

[0143] 암호키생성기내의 순열연산기(GA Operator)는 암호화기 또는 복호화기의 순열연산기가 각각 암호문과 원문메시지 생성에 필요한 연산을 수행하는 것과 달리 대칭키와 개인키를 통해 공개키를 생성하는 역할을 한다. 개인키모듈 SK로 부터 생성되는 키를 H라 하고, 대칭키모듈 MPK로 부터 생성되는 키를 Q라 하고 순열연산기에서 연산되어 생성되는 공개키를 G라 하면, 순열연산기에서 연산되는 순열연산은 $G = Q^{-1}H^{-1}Q^{-1}$ 이 되고, 연산결과 산출되는 공개키 G는 공개키모듈 PK로 출력되어 보관되고 해당 암호화기 또는 복호화기로 배포된다.

[0144] 이제 도 9를 통해 암호화기의 작동에 대한 일 실시예를 살펴보고자 한다.

[0145] 메시지 집합 M이 0~9의 숫자로 구성된 경우, 즉 |M| = 10일 때, 사용자 A에서 사용자 B로 4581290367의 10개의 숫자메시지를 전송하고자 한다. 도 9는 사용자 A 단말의 암호화기를 통해 암호문 5301689742가 생성되는 것을 보여준다.

[0147] 2. 암호화/복호화 방법 및 절차

[0148] 본 발명의 상기 실시예에 따라 도 10에서 보여주는 방법 및 절차에 따라 순열그룹 기반의 메시지 암호화 전송방법을 구현할 수 있다.

[0149] 실시예에 따른 시스템을 이용하여 메시지 전송을 하기 위해서는 송수신 참여자는 사전에 시스템에 개인식별 정보 등을 등록하여 허용된 참여자로 승인받아야 한다.

[0150] 따라서 실시예에 따른 전송방법의 제 1단계는 설정(setup)단계로서 암호키생성기(MKG)에 참여객체를 식별할 수 있도록 사용자 식별인자(phone number, user id, email address etc.), 단말장치 식별인자(device id, MAC address, ip address, faceid, fingerprint etc.), 비밀함수(Secret Permutation) 생성인자 등과 같은 개인식별 정보를 등록하고 암호키생성기(MKG)는 이 정보에 따라 등록하는 객체의 식별번호, 마스터키 등을 생성한다.

[0151] 마스터키는 메시지 전체 순열그룹중에서 비밀함수후보(SPC)들을 특정할 수 있는 벡터함수이며 이러한 키벡터함수 T는 (tp,tv) 벡터쌍을 이루며 $T \subset I_M \times M$ 이다. 마스터키 T의 tp는 설정시에 생성 또는 등록되고, 마스터키 T의 함수값 tv는 암호화 실행시에 결정되어 비밀함수후보(SPC)중에서 특정 비밀함수(SP)를 지정하고 이와 관련하여 해당 참여객체의 개인키를 생성하게 한다.

[0152] 제 2단계는 암호키생성기에서 암호화를 위한 키생성단계로 암호화와 복호화에 필요한 대칭키와 개인키 및 공개키 쌍을 생성한다. 송수신 참여객체만의 사전등록 정보를 통해 송수신 양자만이 알 수 있는 대칭키를 생성한다. 또한 설정단계에서 생성한 마스터키(함수)에 일회용 함수값을 할당하여 비밀함수(SP)를 지정하고 이와 함께 설정단계에서 사전에 등록된 개인식별 정보를 바탕으로 개인키를 생성한다. 공개키는 생성된 대칭키와 개인키의 순열연산(GA)에 의해 생성된다.

[0153] 제 3단계에서는 암호문 생성을 위해 송신자가 암호키생성기를 통해 암호화 키인 수신자의 공개키를 요청하여 얻는다. 대칭키는 각 송수신 단말은 2단계를 통해 이미 보유하고 있다. 이 때 해당 참여객체들의 키들은 제 2단계인 키생성단계에서 이미 생성되어 있으므로 쉽게 획득가능하다. 수신자의 공개키와 이미 갖고 있는 대칭키를 순열연산기를 통해 연산하여 암호문을 생성한다. 이 과정을 수학적으로 표현하면 다음과 같다.

[0154] 메시지 $M = (m_1 \cdots m_n)$, 키 $K = (k_1 \cdots k_n)$, 암호문(ciphertext) $C = (c_1 \cdots c_n)$는 각각 메시지집합 $M=\{m_1 \cdots m_n\}$ 상에서의 순열로 순열그룹 G의 원소이며, $E : M \times K \rightarrow C$, $M, K, C \in G$ 와 같으며, 암호화키 K_e는 (MPK, PK) 쌍이고 순열함수쌍 (Q, G)로 나타낸다. 즉 $K_e=(Q, G)$ 이다. 암호화 함수 E는 K_e를 구성하는 순열함수 Q, G의 순열연산(Group Action)인 left multiplication에 의해 $E=Q^{-1}GQ$ 와 같이 나타낼 수 있다. 따라서 $E(M, K_e) = Q^{-1}GQ(M) = C$ 와 같다.

[0155] $D = d_1d_2 \cdots d_k$: 메시지 시퀀스 D는 d_i가 메시지 집합 M의 원소로 구성된 연속된 메시지 문자열이라 할 때, 결과 암호문열이 $x = x_1x_2 \cdots x_k$ 라고 하면,

[0156] 이 때 $E(d_i, K_e) = Q^{-1}GQM((d_i)) = x_i$이다. $R = Q^{-1}$라 하면,

[0157] $$G=\begin{pmatrix}1 & \cdots & n\\ g_1 & \cdots & g_n\end{pmatrix},\quad Q=\begin{pmatrix}1 & \cdots & n\\ q_1 & \cdots & q_n\end{pmatrix},\quad R=\begin{pmatrix}1 & \cdots & n\\ r_1 & \cdots & r_n\end{pmatrix},\quad M=\begin{pmatrix}1 & \cdots & n\\ m_1 & \cdots & m_n\end{pmatrix}$$ 일 때,

[0158] $$Q^{-1}GQ(M(d_i))=\begin{pmatrix}1 & \cdots & n\\ r_1 & \cdots & r_n\end{pmatrix}\begin{pmatrix}1 & \cdots & n\\ g_1 & \cdots & g_n\end{pmatrix}\begin{pmatrix}1 & \cdots & n\\ q_1 & \cdots & q_n\end{pmatrix}\begin{pmatrix}1 & \cdots & n\\ m_1 & \cdots & m_n\end{pmatrix}(d_i)=x_i$$ 와 같으며 E는 left multiplication

에 의해 계산된다.

[0159] 제 4단계에는 수신된 암호문을 복원하기 위해 수신자가 암호키생성기를 통해 암호화키인 수신자의 개인키를 얻는다. 대칭키는 각 송수신 단말은 2단계를 통해 이미 보유하고 있다. 이 때 해당 참여객체들의 키들은 제 2단계인 키생성단계에서 이미 생성되어 있으므로 쉽게 획득가능하다. 수신자의 개인키와 이미 갖고 있는 대칭키를 순열연산기를 통해 연산하여 원문메시지를 복원한다. 이 과정을 수학적으로 표현하면 다음과 같다.

[0160] 메시지 M = (m_1 … m_n), 키 K = (k_1 … k_n), 암호문(ciphertext) C = (c_1 …c_n)는 각각 메시지집합 M={m_1 … m_n} 상에서의 순열로 순열그룹 G의 원소이며, D : C ×K → M, M, K, C ∈ G 와 같으며, 복호화키 K_d는 (MPK, SK) 쌍이고 순열함수쌍 (Q, H)로 나타낸다. 즉 K_d=(Q, H) 이다.

[0161] 복호화 함수 D는 K_d를 구성하는 순열 Q, H의 순열연산(Group Action)인 left multiplication에 의해 D=HQQ 와 같이 나타낼 수 있다. 따라서 D(C, K_d) = HQQ(C) = M 가 된다.

[0162] X = x_1x_2 …x_k 암호문 시퀀스 X는 x_i가 암호문 집합 C의 원소로 구성된 연속된 암호문 문자열이라 하고, D=d_1d_2… d_k : 메시지 시퀀스 D는 d_i가 메시지 집합 M의 원소로 구성된 연속된 메시지 문자열이라 할 때 D(x_i, K_d) = HQQ(C(x_i)) = d_i 이다.

[0163] $H=\begin{pmatrix}1 & \cdots & n\\ h_1 & \cdots & h_n\end{pmatrix}$, $Q=\begin{pmatrix}1 & \cdots & n\\ q_1 & \cdots & q_n\end{pmatrix}$, $C=\begin{pmatrix}1 & \cdots & n\\ c_1 & \cdots & c_n\end{pmatrix}$ 일 때,

[0164] $$HQQ(C(x_i))=\begin{pmatrix}1 & \cdots & n\\ h_1 & \cdots & h_n\end{pmatrix}\begin{pmatrix}1 & \cdots & n\\ q_1 & \cdots & q_n\end{pmatrix}\begin{pmatrix}1 & \cdots & n\\ q_1 & \cdots & q_n\end{pmatrix}\begin{pmatrix}1 & \cdots & n\\ c_1 & \cdots & c_n\end{pmatrix}(x_i)=d_i$$ 와 같으며 D는 left multiplication에 의해 계산된다.

[0166] 3. 메시지 암호화/복호화 전송방법 실시예

[0167] 도 11내지 도 14에서는 본 발명의 일 실시예로 메시지 암호화 전송방법의 단계별 구현에 대한 구체적인 예시를 보여준다.

[0168] 예시는 단말 A에서 단말 B로 0~9의 숫자를 원소로 하는 메시지 집합에서 숫자 10개로 구성된 숫자열 “4581290367”을 메시지로 입력받아 암호문 생성하여 전송하고 이를 수신하여 원문 메시지를 복원하는 과정을 수행 단계별로 구체적으로 도 11에서 도 14를 통해 보여준다.

[0169] 도 11에서는 1단계 설정단계에서 송수신을 위해 암호키생성기에 송신단말 A와 수신단말 B의 ID 등록과 이를 통해 마스터개인키 벡터함수 {(2, v_1), (4, v_2), (6, v_3), (8, v_4)}가 생성되고, 마스터공개키 생성함수가 설정되는 것을 보여준다.

[0170] 도 12에서는 2단계 키생성단계로 마스터키 벡터함수의 벡터값 할당과 이를 통해 어떻게 개인키가 생성되는지 보여준다. 또한 순열생성기를 통해 대칭키 생성함수에 함수값이 어떻게 할당되는 지와 이와 함께 개인키와 순열연산기를 통해 공개키가 생성되는 것을 보여준다.

[0171] 도 13에서는 3단계 암호문 생성단계로 1, 2단계를 통해 생성된 암호화키 MPK, SK, PK를 통해 순열연산기(GA Operator)에서 순열연산을 통해 어떻게 연산이 이루어지고 암호문이 어떻게 생성되는 지 구체적인 예시를 통해 보여준다.

[0172] 도 14에서는 4단계 암호문 복호화 단계로 1, 2단계를 통해 생성된 암호화키 MPK, SK, PK를 통해 순열연산기(GA Operator)에서 순열연산을 통해 어떻게 연산이 이루어지고 수신된 암호문이 어떻게 원문메시지로 복호화되어 복원되는 지 구체적인 예시를 통해 보여준다.

[0174] 4. 서명/검증 시스템

[0175] 전자서명 시스템의 각 송수신 단말은 기능적으로는 앞서 일 실시예에서 설명한 암호화기나 복호화기와 동일한 구조로 동일하게 작동하지만 다른 키와 다른 입력을 사용하여 작동하는 점이 다르다. 서명/검증 시스템의 송신 및 수신의 각 단말은 도 15와 같이 서명기(SIGN), 검증기(VERIFY) 그리고 암호키생성기(MKG)로 구성되어 있다.

[0176] 서명기(SIGN)는 도 16과 같이 메시지 입력을 처리하는 입력큐와 본 발명의 일 실시예에 따른 알고리즘을 통해 일회용개인키를 이용하여 순열연산(Group Action)을 통해 서명문(signature)을 생성하는 GA 연산기, 그리고 생성된 서명문(signature)의 출력을 처리하는 출력큐가 있다.

[0177] GA 연산기는 메시지를 입력으로 암호키생성기(MKG)로 부터 송수신 단말의 대칭키(Q_{AB})와 송신단말의 개인키(H_A)를 제공받아 순열연산(GA 연산)을 통해 서명문(signature)을 생성한다. GA 연산기에서 처리하는 순열연산은 $Q_{AB}^{-1}H_AQ_{AB}(M) = M_s$ 이다.

[0178] 한편, 다른 일 실시예에서 암호복잡도에는 큰 영향은 없으나 송신단말에 입력된 메시지가 중복된 문자열을 포함하는 경우를 제거하기 위해 확산함수(diffusion function) F(x)를 XOR 연산기를 통해 전처리하고 서명기의 메시지큐는 전처리된 메시지를 입력받아 서명문을 생성할 수 있다.

[0179] 검증기(VERIFY)는 도 17과 같이 서명문(signature)을 입력받아 처리하는 입력큐와 본 발명의 일 실시예에 따른 알고리즘을 통해 일회용공개키를 이용하여 순열연산(Group Action)을 통해 서명문을 검증하여 수락된 원문메시지를 생성하는 GA 연산기, 그리고 검증/수락된 원문메시지의 출력을 처리하는 출력큐가 있다.

[0180] GA 연산기는 서명문를 입력으로 암호키생성기(MKG)로 부터 송수신 단말의 대칭키(Q_{AB})와 송신단말의 공개키(G_A)를 제공받아 순열연산(GA 연산)을 통해 서명문을 검증하고 수락/거절여부를 확인하여 원문메시지를 검증한다.

[0181] GA 연산기에서 처리하는 순열연산은 $G_AQ_{AB}Q_{AB}(M_s) = M$ 이다.

[0182] 한편, 다른 일 실시예에서는 송신단말에 확산함수(diffusion function)을 적용한 경우 검증기에서 검증된 메시지를 송신단말에 적용된 동일한 확산함수(diffusion function) F(x)를 XOR 연산기를 통해 후처리하여 원문메시지를 복원할 수 있다.

[0183] 도 15의 서명/검증시스템에 나타나는 암호키생성기(MKG)는 도 8에서 보는 바와 같이 난수생성기(PRNG), 순열생성기(permutation generator), 복수의 키생성모듈들($MSK\_\overrightarrow{tv}$, $MSK\_\overrightarrow{tp}$, MPK, SK, PK) 그리고 순열연산기(GA Operator) 등으로 암호화기나 복호화기에서와 같은 구조와 기능을 제공하고 동일하게 작동한다.

[0185] 5. 서명/검증 방법 및 절차

[0186] 본 발명의 상기 실시예에 따라 도 18에서 보여주는 방법 및 절차에 따라 순열그룹 기반의 메시지 암복호화 알고리즘을 이용하여 서명/검증 방법을 수행할 수 있다.

[0187] 도 18의 일 실시예에 따른 메시지에 대한 서명 및 검증 방법은 4단계의 절차에 의해 수행될 수 있으며, 제 1단계 송수신 참여객체에 대한 등록 및 설정 방법과 제 2단계 키생성 방법 및 절차는 상기 암복호화 방법의 일 실시예에서 설명한 방법과 절차와 동일하게 수행한다.

[0188] 제 3단계에서는 서명문 생성을 위해 송신자가 암호키생성기를 통해 서명키인 송신자의 개인키를 요청하여 얻는다. 대칭키는 각 송수신 단말은 2단계를 통해 이미 보유하고 있다. 이 때 해당 참여객체들의 키들은 제 2단계인 키생성단계에서 이미 생성되어 있으므로 쉽게 획득가능하다. 송신자의 개인키와 이미 갖고 있는 대칭키를 순열연산기를 통해 연산하여 서명문을 생성한다. 이 과정을 수학적으로 표현하면 다음과 같다.

[0189] 메시지 $M = (m_1 \cdots m_n)$, 키 $K = (k_1 \cdots k_n)$, 서명문(signature) $M_s = (s_1 \cdots s_n)$ 는 각각 메시지집합 $M=\{m_1 \cdots m_n\}$ 상에서의 순열로 순열그룹 G의 원소이며, $S : M \times K \rightarrow S$, $M,K,S \in G$ 와 같으며, 서명키 K_s는 (MPK, SK) 쌍이고 순열함수쌍 (Q, H)로 나타낸다. 즉 $K_s=(Q, H)$ 이다. 서명 함수 S는 K_s를 구성하는 순열함수 Q, H의 순열연산(Group Action)인 left multiplication에 의해 $S=Q^{-1}HQ$ 와 같이 나타낼 수 있다. 따라서 $S(M, K_S) = Q^{-1}HQ(M) = M_s$ 와 같다.

[0190] D = d_1d_2 …d_k: 메시지 시퀀스 D는 d_i가 메시지 집합 M의 원소로 구성된 연속된 메시지 문자열이라 할 때, 결과 서명문열이 X = x_1x_2 …x_k 라고 하면,

[0191] 이 때 S=(d_i, K_s) = Q^{-1}HQM((d_i)) = x_i이다. R=Q^{-1}라 하면,

[0192] $$H=\begin{pmatrix}1 & \cdots & n\\ h_1 & \cdots & h_n\end{pmatrix},\quad Q=\begin{pmatrix}1 & \cdots & n\\ q_1 & \cdots & q_n\end{pmatrix},\quad R=\begin{pmatrix}1 & \cdots & n\\ r_1 & \cdots & r_n\end{pmatrix},\quad M=\begin{pmatrix}1 & \cdots & n\\ m_1 & \cdots & m_n\end{pmatrix}$$ 일 때,

[0193] $$Q^{-1}HQ(M(d_i))=\begin{pmatrix}1 & \cdots & n\\ r_1 & \cdots & r_n\end{pmatrix}\begin{pmatrix}1 & \cdots & n\\ h_1 & \cdots & h_n\end{pmatrix}\begin{pmatrix}1 & \cdots & n\\ q_1 & \cdots & q_n\end{pmatrix}\begin{pmatrix}1 & \cdots & n\\ m_1 & \cdots & m_n\end{pmatrix}(d_i)=x_i$$ 와 같으며 S는 left multiplication에 의해 계산된다.

[0194] 제 4단계에는 수신된 서명문을 검증하기 위해 수신자가 암호키생성기를 통해 검증키인 송신자의 일회용공개키를 얻는다. 대칭키는 각 송수신 단말은 2단계를 통해 이미 보유하고 있다. 이 때 해당 참여객체들의 키들은 제 2단계인 키생성단계에서 이미 생성되어 있으므로 쉽게 획득가능하다. 송신자의 공개키와 이미 갖고 있는 대칭키를 순열연산기를 통해 연산하여 서명문을 검증하고 검증된 원문메시지를 수락 또는 거절한다. 이 과정을 수학적으로 표현하면 다음과 같다.

[0195] 메시지 M = (m_1 … m_n), 키 K = (k_1 … k_n), 서명문(signature) M_s = (s1 … s_n)는 각각 메시지집합 M={m_1 … m_n} 상에서의 순열로 순열그룹 G의 원소이며, V : S ×K → S, M, K, S ∈ G 와 같으며, 검증키 K_v는 (MPK, PK) 쌍이고 순열함수쌍 (Q, G)로 나타낸다. 즉 K_v=(Q, V) 이다.

[0196] 검증 함수 V는 K_v를 구성하는 순열 Q, G의 순열연산(Group Action)인 left multiplication에 의해 V=GQQ 와 같이 나타낼 수 있다. 따라서 V(S, K_v) = GQQ(M_s) = M 가 된다.

[0197] X = x_1x_2 …x_k 서명문 시퀀스 X는 x_i가 서명문 집합 M_s의 원소로 구성된 연속된 서명문 문자열이라 하고 결과 메시지 문자열이 D=d_1d_2…d_k 라 할 때 V(x_i, K_v) = GQQ(M_s(x_i)) = d_i 이다.

[0198] $$G=\begin{pmatrix}1 & \cdots & n\\ g_1 & \cdots & g_n\end{pmatrix},\quad Q=\begin{pmatrix}1 & \cdots & n\\ q_1 & \cdots & q_n\end{pmatrix},\quad M_s=\begin{pmatrix}1 & \cdots & n\\ m_1 & \cdots & m_n\end{pmatrix}$$ 일 때,

[0199] $$GQQ(M_s(x_i))=\begin{pmatrix}1 & \cdots & n\\ g_1 & \cdots & g_n\end{pmatrix}\begin{pmatrix}1 & \cdots & n\\ q_1 & \cdots & q_n\end{pmatrix}\begin{pmatrix}1 & \cdots & n\\ q_1 & \cdots & q_n\end{pmatrix}\begin{pmatrix}1 & \cdots & n\\ m_1 & \cdots & m_n\end{pmatrix}(x_i)=d_i$$ 와 같으며 E는 left multiplication에 의해 계산된다.

[0201] 6. 메시지 서명/검증 전송 방법 예시

[0202] 도 19 내지 도 22에서는 본 발명의 일 실시예로 메시지 서명문 전송방법의 단계별 구현에 대한 구체적인 예시를 보여준다.

[0203] 예시는 단말 A에서 단말 B로 0~9의 숫자를 원소로 하는 메시지 집합에서 숫자 10개로 구성된 숫자열 “4581290367”을 메시지로 입력받아 서명문 생성하여 전송하고 이를 수신하여 원문 메시지를 검증하는 과정을 수행 단계별로 구체적으로 도 19에서 도 22을 통해 보여준다.

[0204] 도 19에서는 1단계 설정단계에서 송수신을 위해 암호키생성기에 송신단말 A와 수신단말 B의 ID 등록과 이를 통해 마스터키 벡터함수 {(1, v_1), (3, v_2), (5, v_3), (7, v_4)}가 생성되고, 대칭키 생성함수가 설정되는 것을 보

여준다.

[0205] 도 20에서는 2단계 키생성단계로 마스터키 벡터함수의 벡터값 할당과 이를 통해 어떻게 일회용개인키가 생성되는지 보여준다. 또한 순열생성기를 통해 대칭키 생성함수에 함수값이 어떻게 할당되는 지와 이와 함께 개인키와 순열연산기를 통해 공개키가 생성되는 것을 보여준다.

[0206] 도 21에서는 3단계 서명문 생성단계로 1, 2단계를 통해 생성된 암호화키 MPK, SK, PK를 통해 순열연산기(GA Operator)에서 순열연산을 통해 어떻게 연산이 이루어지고 서명문이 어떻게 생성되는 지 구체적인 예시를 통해 보여준다.

[0207] 도 22에서는 4단계 검증 메시지 수락/거절 단계로 1, 2단계를 통해 생성된 암호화키 MPK, SK, PK를 통해 순열연산기(GA Operator)에서 순열연산을 통해 어떻게 연산이 이루어지고 수신된 서명문이 어떻게 원문메시지로 검증되고 수락/거절되는 지 구체적인 예시를 통해 보여준다.

[0209] 7. 메시지의 서명/검증을 포함한 암호화/복호화 시스템

[0210] 전자서명 및 검증을 제공하는 암호시스템에서의 각 송수신 단말은 도 5과 같이 암호화기(ENC), 복호화기(DEC) 그리고 암호키생성기(MKG) 등 동일한 구조를 가지나, 암호화기(ENC)와 복호화기(DEC)는 각각 도 23과 도 24와 같이 서명기와 검증기를 포함하도록 변경될 수 있다.

[0211] 여기에서 암호화기(ENC)는 도 23에서 보는 바와 같이 도 6의 암호화기에 도 25의 서명기를 결합한 구조로 메시지 입력을 처리하는 입력큐와 암호문 생성용 GA연산기와 서명문 생성용 GA 연산기 등 2개의 다른 GA연산기가 포함되고 암호문 생성용 GA연산기는 입력큐로부터 메시지를 전달받고 서명문 생성용 GA연산기로 부터 서명문을 전달받아 도 25의 예시와 같이 (메시지+서명문) 에 대하여 순열연산을 수행하여 암호문을 생성한다.

[0212] 한편 복호화기(DEC)는 도 24에서 보는 바와 같이 도 6의 복호화기에 도 27의 검증기를 결합한 구조로 암호문 입력을 처리하는 입력큐와 메시지 복원용 (복호화용) GA연산기와 서명문 검증용 GA 연산기 등 2개의 다른 GA연산기가 포함되고 메시지 복원용(복호화용) GA연산기는 입력큐로부터 암호문을 전달받아 복호화하여 (메시지+서명문)을 복원하고 여기에서 서명문은 검증용 GA연산기로 전달하고 검증용 GA연산기는 도 27의 예시와 같이 검증된 메시지를 생성한다. 각각 다른 두개의 GA연산기로 부터 출력된 메시지는 AND 연산을 통해 메시지 수락 또는 거절 여부를 결정한다.

[0214] 8. 서명된 메시지의 암호화 전송 및 복호화/검증 방법

[0215] 본 발명의 일 실시예에 따라 순열그룹 기반의 공개키를 이용하여 서명된 메시지에 대한 암호화 전송방법을 도 26과 같이 구현할 수 있다.

[0216] 도 26의 일 실시예에 따른 서명된 메시지에 암호화 전송방법은 6단계의 절차에 의해 수행될 수 있으며, 제 1단계 송수신 참여객체에 대한 등록 및 설정 방법과 제 2단계 키생성 방법 및 절차는 상기 도 10의 암복호화 방법의 일 실시예에서 설명한 방법과 절차와 동일하게 수행한다.

[0217] 제 3단계의 서명문 생성 방법 및 절차는 도 18의 메시지 서명/검증 방법에서와 같다.

[0218] 제 4단계는 도 25의 예시와 같이 전송하고자 하는 메시지와 제 3단계에서 생성된 서명문을 결합하여 수신자의 공개키로 (메시지+서명문)을 암호화한다.

[0219] 즉, $E(M', K_e) = E((M+M_s), K_e) = Q_{AB}^{-1}G_BQ_{AB}(M+M_s) = C'$ (K_e는 수신자 B의 공개키(G_B)) 제 5단계는 수신된 암호문 M'를 도 27의 예시와 같이 복호화하여 $M+M_s$를 복원한다.

[0220] 즉, $D(C', K_d) = H_BQ_{AB}Q_{AB}(C') = M'$ (K_d는 수신자 B의 개인키(H_B)), $M' = M+M_s$

[0221] 제 6단계는 서명문 M_s를 $V(M_s, Kv) = Q_{AB}^{-1}G_AQ_{AB}(M_s) = M''$ (K_v는 송신자 A의 공개키(G_A)) 과 같이 검증하여 검증된 메시지 M"을 얻고 5단계에서 복원한 원문메시지 M과 검증된 메시지 M"이 일치하는 지 여부를 확인하여 메시지를 수락 또는 거절할 지 결정한다. 이 서명된 메시지 전송방법을 통해 메시지의 위조 또는 변조 여부를 판단하고 위변조되지 않은 메시지만 수신하여 무결성을 보장할 수 있다. 또한 서명은 서명한 참여자의 유일하게 한번

생성되는 개인키를 사용하므로 서명한 참여자를 제외한 누구도 생성할 수 없다. 따라서 일 실시예에 따라 서명한 메시지 전송방법을 통해 전송한 메시지에 대하여는 송신자는 메시지 송신을 부인(repudiation)할 수 없다.

[0223] 9. 기타 구현 및 적용 예시

[0224] 암호화에 사용하는 암호화 키는 Digit, Character, Images 등 메시지 스페이스를 구성하는 요소를 활용한다. 예를 들어, 문자를 암호화하기 위해서 확장 아스키코드(extended ASCII Code)를 암호화 키 스페이스로 확장해 256 byte 암호화 시스템을 구현하는 것이 가능하다.

[0225] 본 발명 시스템은 2단 구조(2 tier) 또는 3단 구조(3 tier)로 구현될 수 있다.

[0226] 2단 구조(2 Tier)에서는 암호화 메시지를 전송하는 전송자(Sender) 와 암호화 메시지를 수신하여 복호화하는 수신자(Receiver) 간에 암호화 통신 중간 매개체를 거치지 않고 통신하는 구조에 적용할 수도 있다.

[0227] 암호화 메시지를 전송하고 수령하는 전송자(Sender)와 수신자(Receiver) 간의 역할이 일방적이며 고정적이고 변동되지 않을 경우인 단방향(One-way) 통신 방식과 암호화 메시지를 상호 전송/수령하는 경우 양방향(two way) 통신이 모두 구현 가능하며 이경우, 전송자(Sender)와 수신자(Reciever) 모두 암호화 수행객체와 복호화 수행객체 모두가 탑재될 수 있다.

[0228] 이러한 시스템의 구현 예시는 1대1통신, 또는 피어투피어 통신(Peer to Peer Communication), 일대다수 통신 방식(One to Many Communication) 등에도 적용될 수 있다.

[0229] 3단 구조(3Tier)에서는 암호화 메시지를 전송하는 전송자(Sender)와 암호화 메시지 메시지 또는 평문을 수신하는 수신자(Reciever) 간에 암/복호화 기능 수행 또는 다른 통신 프로토콜로의 변환 등과 같은 타시스템과의 중계 또는연동 기능을 수행하는 게이트웨이(Gateway) 를 통해 통신하는 구조에 적용할 수도 있다.

[0230] 이때 게이트웨이(Gateway)는 송신자(Sender) 가 지정한 수신자(Receiver)에게 메시지를 전송할 때 자체 복호화를 실행하여 평문 자체를 전송하거나, 이를 수신자(Receiver) 가 원하는 다른 암호화 방식 또는 다른 통신 프로토콜 형식으로의 변환 작업을 통해 변경된 형식의 메시지를 전송하거나, 송신자(Sender)가 보낸 암호화된 메시지 자체를 수신자(Receiver)에게 전송할 수도 있다.

[0231] 이러한 시스템은 센서-게이트웨이-서버(Sensor-Gateway-Server), 센서-게이트웨이-센서(Sensor-Gateway-Sensor) 등과 같은 IoT 네트워크 방식이나, 전통적인 3단 구조(3-Tier) 방식이나 N단 구조(N-Tier) 방식이라 칭하는 다중 객체 참여 통신 시스템에 적용될 수 있다.

[0232] 한편, 본 명세서에서 사용하는 용어 중 송신 단말이나 수신 단말은 적어도 하나의 네트워크를 통해 통신 가능하도록 연결된 단말기를 가리키며, 일 예로, 휴대폰, 스마트 폰(smart phone), 노트북 컴퓨터(laptop computer), 디지털방송용 단말기, PDA(personal digital assistants), PMP(portable multimedia player), 슬레이트 PC(slate PC), 태블릿 PC(tablet PC), 울트라북(ultrabook) 등과 같은 이동 단말기나, 디지털 TV, 데스크탑 컴퓨터 등과 같은 고정 단말기일 수 있으나, 특별히 한정하는 것은 아니다.

[0233] 본 발명의 일 실시예에 따르면, 안전하고 다양한 환경에 적용 가능한 순열그룹기반 일회용공개키를 이용하는 비대칭방식의 암호통신시스템을 구축할 수 있다.

[0234] 메시지를 암호화하는 키로서 일회용으로 생성되는 공개순열(Public Permutation)을 이용하는 일회용공개키와 이를 통해 생성된 암호문을 평문으로 복호화하기 위해서는 오직 수신단말이 일회용으로 생성한 개인순열(Private Permutation)을 이용하는 개인키를 통해서만 가능한 비대칭 키를 사용하는 암호 통신 시스템이 구현 가능케 된다.

[0235] 이 때 일회용공개키와 일회용개인키는 모두 암호문을 복호화할 수 있는 수신 단말의 마스터개인키를 통해서만 생성가능하며, 안전한 방법을 통해 일회용공개키는 송신 단말에게 공유되는 시스템의 구현이 가능하다. 송신 단말이 보유한 일회용공개키 관련정보나 이를 통해 생성된 암호문을 가지고 원문을 복원 또는 추정하기 어려운 암호학의 Trap Door function을 가지고 구현된다.

[0236] 공개키와 개인키는 보안성 향상 등을 목적으로 매번 랜덤으로 암호화 통신 과정 중 또는 통신 이후에도 자동적 또는 수동적으로 변경 가능하며, 이러한 생성 변경 작업은 오직 복호화 권한을 가진 마스터개인키를 보유한 사용자/시스템/기기에 의해 수행된다. 이러한 기능을 통해 암호 통신 시스템 내 사용된 공개키 및 개인키의 노출

및 암호문의 수집 및 역추적 공학(Reverse Engineering) 등을 통한 추정이 어려운 특성의 구현이 가능하다.

[0238] 키교환 없는 대칭키 암호통신

[0239] 또한, 암호화 통신 시스템에 필요한 암호관련 키인 순열키(Permutation Key)를 직접 전송하지 않는 방식으로 구현이 가능하다. 예를 들면 암호화과정과 복호화 과정에서 필요한 공개키/개인키는 수신 단말에 의해 생성된 이후, 생성 및 변경 발생시 해당 생성 조건(시간, 공간 등) 및 변경 조건을 사전 약정한 송신 단말과 수신 단말이 각각 자체 순열연산기를 포함하도록 하고 이를 통해 송신 단말 스스로 동일한 가상의 공개키를 생성할 수 있도록 구현함으로서 암호화 통신과정에서 필수적인 암호화 관련 키 정보를 직접 송/수신하는 단계를 수행하지 않고 마치 키교환 없는 대칭키 암호통신을 하는 것과 같은 시스템을 구축할 수 있다.

[0241] 암호화에 사용된 공개키를 통해 전송되는 값은 실제 값의 추정이 어려운 수신자만이 가지고 있는 마스터개인키 내 정보 중 일부 정보만을 이용하여 생성하는 개인키를 통해 유도되어지기 때문에 이에 대한 정보를 가지고 평문을 복호화하거나 추정하기는 어려운 암호화 통신 시스템의 구현이 가능하다.

[0243] 키노출에도 안전유지

[0244] 공개키와 개인키는 보안 정책 또는 시스템 요건에 따라 매번 랜덤하게 변경되기 때문에 관련 정보가 노출된 이후에도 이전 정보를 소유한 악의적 사용자가 탈취한 이후 생성된 암호문을 임의로 또는 무단으로 복호할 수 없는 시스템의 구현이 가능하다.

[0246] 상기에서 살펴본 바와 같이 본원발명은 순열기반으로 비대칭 방식을 이용하고 암호화 처리를 byte 단위 또는 원하는 크기의 메시지처리단위로 할 수 있으므로 메시지 유형에 따라 다양한 방식으로 구현이 가능하다.

[0247] 또한, 본원발명에 따르면 어플리케이션 메시지 처리 단위에서 바로 연산이 이루어져, 메시지를 블록 단위로 암호화 한 후 이를 어플리케이션에서 사용할 수 있는 형태로 재구성하여야 하는 종래 기술에 비해 처리 속도를 비약적으로 빠르게 할 수 있다. 이를 통해 본원발명은 저성능의 CPU 기기에도 구현이 가능해진다.

[0248] 또한, 본원발명은 단일 암호시스템 내 대칭키/비대칭키 방식 구현이 모두 가능하고, 응용 어플리케이션내 다양한 메시지 형태를 모두 처리 가능하며, 2-Tier 및 3-Teir 통신 구조하에서 유연한 기능 구현이 가능하고, 패스워드/PIN을 기반으로 하는 Human to Machine 방식의 기존 시스템 또는 새로운 Machine to Machine 방식을 채택한 시스템에 모두 적용할 수 있다.

[0249] 즉, 본원발명의 시스템은 경량화/저용량 기기를 기반으로 다양한 통신 구조하에 동작하는 새로운 iot 환경하에서도 단일 시스템으로 적용하는 것이 가능하고, 또한 기존 암호화 기술 기반 시스템과 연동하는 것이 가능하다.

[0251] 컴퓨터 판독 가능한 기록매체

[0252] 이상 설명된 본 발명의 일 실시예에 따른 순열그룹기반 일회용공개키를 이용한 메시지 전송 방법은 다양한 컴퓨터 구성요소를 통하여 수행될 수 있는 프로그램 명령어의 형태로 구현되어 컴퓨터 판독 가능한 기록 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능한 기록 매체는 프로그램 명령어, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 컴퓨터 판독 가능한 기록 매체에 기록되는 프로그램 명령어는 본 발명을 위하여 특별히 설계되고 구성된 것들이나 컴퓨터 소프트웨어 분야의 당 업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능한 기록 매체의 예에는, 하드 디스크, 플로피 디스크, 및 자기 테이프와 같은 자기 매체, CD-ROM, DVD와 같은 광기록 매체, 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 ROM, RAM, 플래시 메모리 등과 같은 프로그램 명령어를 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령어의 예에는, 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드도 포함된다. 상기 하드웨어 장치는 본 발명에 따른 처리를 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

[0254] **III.발명의 효과 등**

[0255] 매번 바뀌는 키로 인해 키공간 및 암호공간이 매번 다른 공간을 제공하여 다차원 공간으로 확장되므로 기존 방식이 brute-force 공격에 대해 시도마다 공간확률이 줄어들어 취약한데 반해 본 발명시스템의 공간확률은 항상 동일한 확률을 제공하므로 키를 유도하는 랜덤함수가 even한 확률분포를 제공한다면 brute-force 공격이 확률적으로 어렵다.

[0256] 또한 복잡한 수학연산을 통해 암호화하지 않으며, 기존 방식과 같이 고정된 함수값을 사용하지 않아 앞서 언급한 바와 같이 순열그룹에 포함된 변동함수를 사용하여 키공간과 암호공간이 다차원 공간으로 확장되므로 암호화 결과에 대하여 양자컴퓨터 등 컴퓨팅 파워가 향상된 컴퓨터를 통해 암호해독을 수행하여도 해독이 어려워 양자저항성(quantum resistant)을 갖는다.

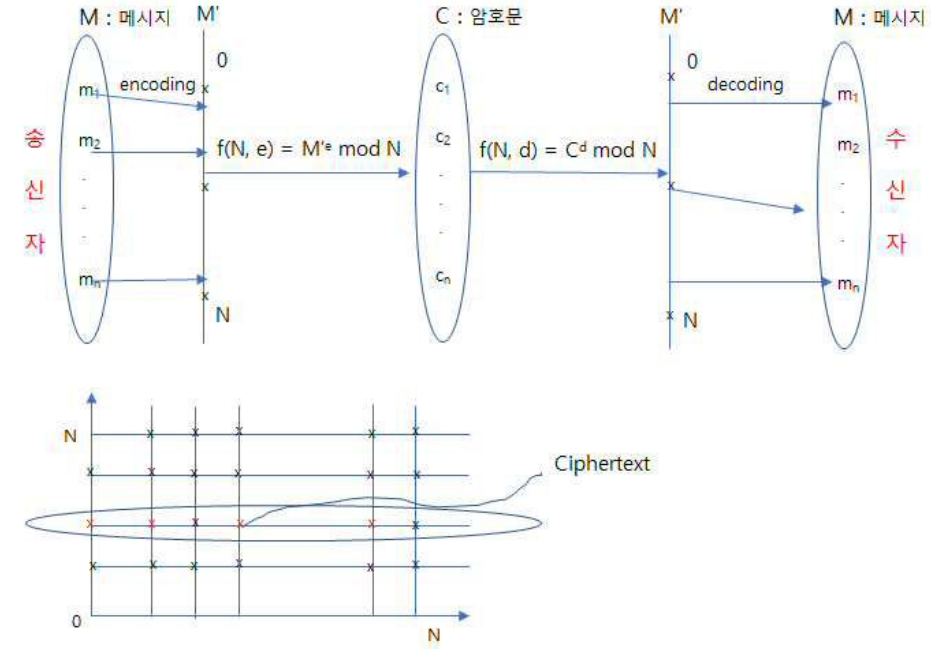
[0257] 또한 기존의 비대칭키 방식의 경우 중간자공격(Man-in-the-Middle)에 취약하여 이러한 문제를 해결하기 위해 제3의 신뢰기관(Certificate Authority)을 통해 암호통신에 참여하는 모든 참여자에게 신원보증서(Certificate)를 발급하고 이 보증서의 진위여부에 따라 암호통신이 가능하도록 인프라를 구축하는 것이 필요하다. 따라서 중간자공격에 안전한 비대칭키 방식의 암호통신을 위해 막대한 비용의 인프라구축이 필요하고 이러한 인프라로 인해 암호화 수행과정이 복잡하고 처리시간이 오래 걸리는 문제점이 있다. 본 발명의 시스템은 매번 다른 키를 생성하여 암복호화를 수행하므로 중간자공격이 불가능하고 따라서 이러한 문제를 해결하느라 도입된 신뢰기관(CA)이나 신원보증서가 필요없이 안전하게 암호통신하는 것이 가능하다.

[0259] 이상으로 본 발명의 바람직한 실시예를 도면을 참고하여 상세하게 설명하였다. 본 발명의 설명은 예시를 위한 것이며, 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자는 본 발명의 기술적 사상이나 필수적인 특징을 변경하지 않고서 다른 구체적인 형태로 쉽게 변형이 가능하다는 것을 이해할 수 있을 것이다.

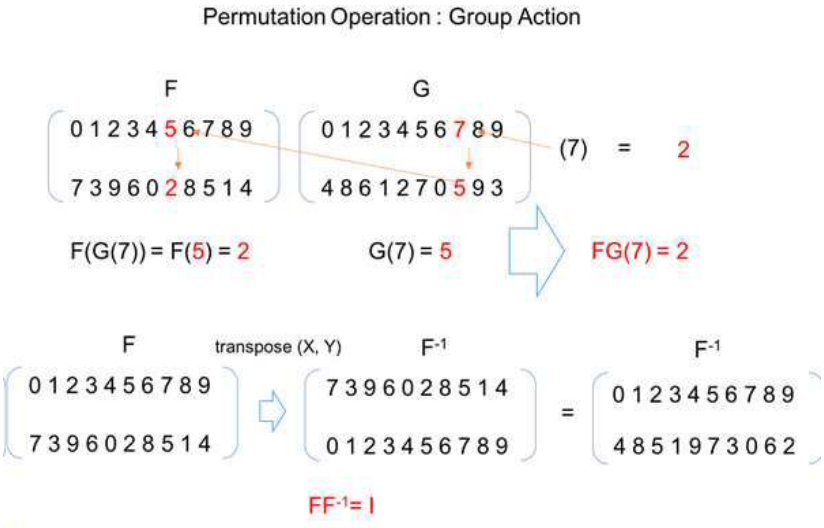
[0260] 따라서, 본 발명의 범위는 상기 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미, 범위 및 그 균등 개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본 발명의 범위에 포함되는 것으로 해석되어야 한다.

도면

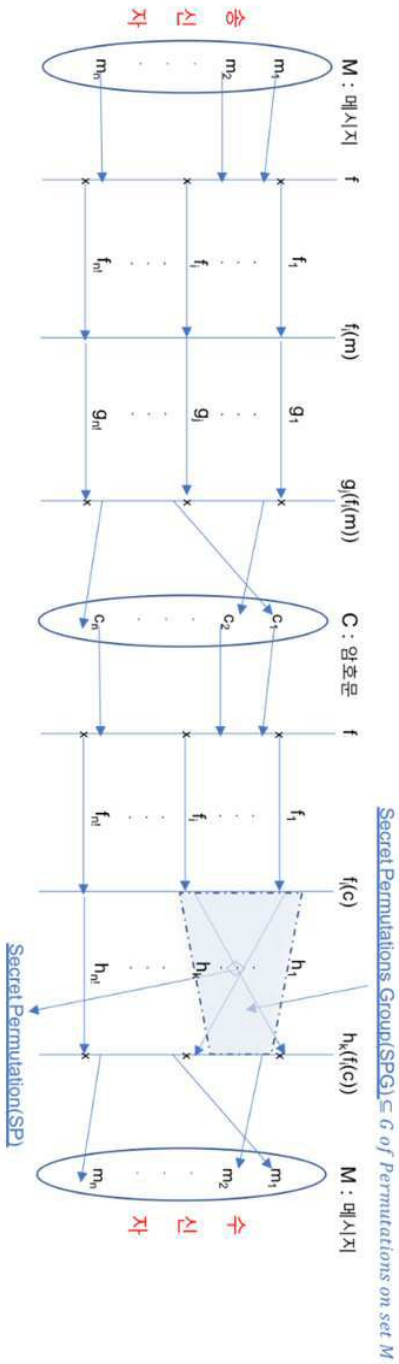
도면1


M : 메시지
M'
C : 암호문
M'
M : 메시지
0
encoding
m1
m2
mn
N
f(N, e) = M'e mod N
c1
c2
cn
f(N, d) = Cd mod N
decoding
송
신
자
수
신
자
Ciphertext
N
0
N

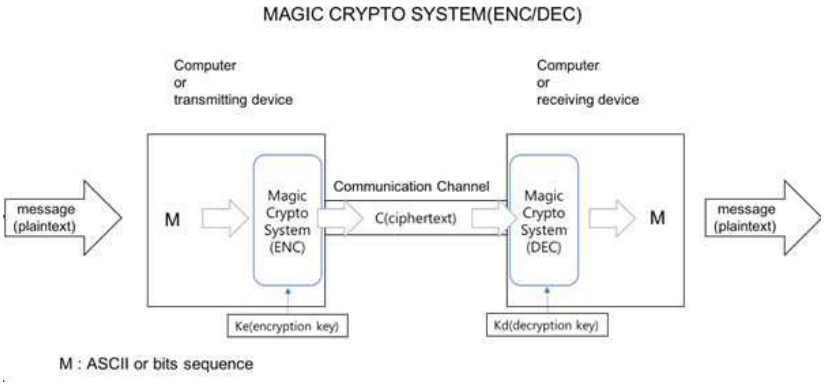
도면2


Permutation Operation : Group Action
F
G
0 1 2 3 4 5 6 7 8 9
7 3 9 6 0 2 8 5 1 4
0 1 2 3 4 5 6 7 8 9
4 8 6 1 2 7 0 5 9 3
(7) = 2
F(G(7)) = F(5) = 2
G(7) = 5
FG(7) = 2
F
transpose (X, Y)
F-1
0 1 2 3 4 5 6 7 8 9
7 3 9 6 0 2 8 5 1 4
7 3 9 6 0 2 8 5 1 4
0 1 2 3 4 5 6 7 8 9
=
F-1
0 1 2 3 4 5 6 7 8 9
4 8 5 1 9 7 3 0 6 2
FF-1= I

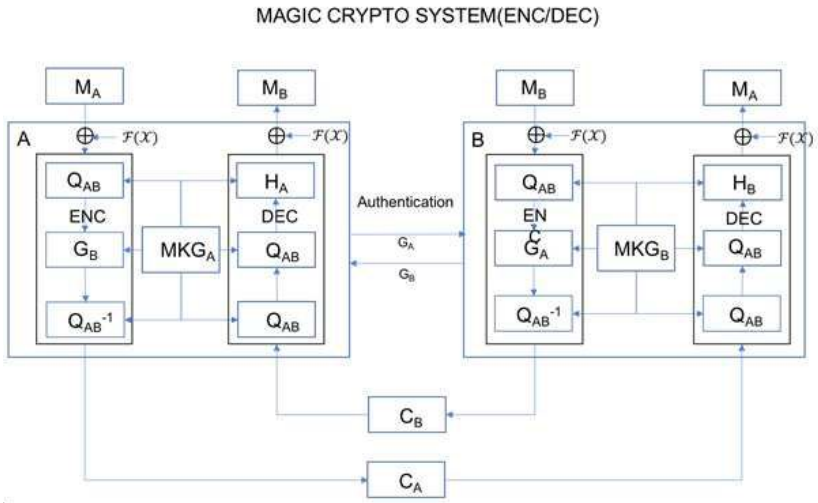
도면3


송 신 자
M : 메시지
f
f_1 f_i f_{nl}
$f_1(m)$
g_1 g_j g_{nl}
$g(f_1(m))$
C : 암호문
f
f_1 f_i f_{nl}
$f_1(c)$
h_1 h_k h_{nl}
$h_k(f_1(c))$
M : 메시지
수 신 자
Secret Permutations Group(SPG) ⊆ G of Permutations on set M
Secret Permutation(SP)

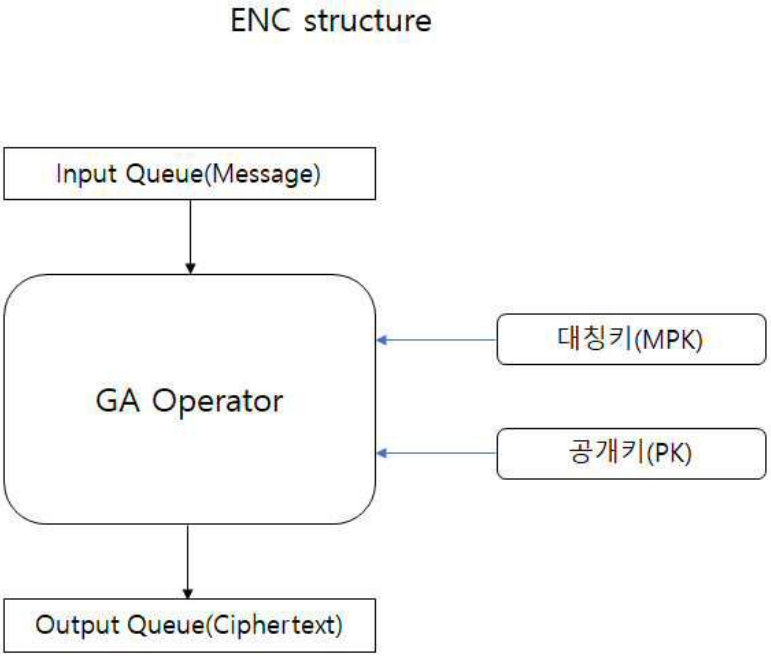
도면4


MAGIC CRYPTO SYSTEM(ENC/DEC)
Computer or transmitting device
Computer or receiving device
message (plaintext)
M
Magic Crypto System (ENC)
Communication Channel
C(ciphertext)
Magic Crypto System (DEC)
M
message (plaintext)
Ke(encryption key)
Kd(decryption key)
M : ASCII or bits sequence

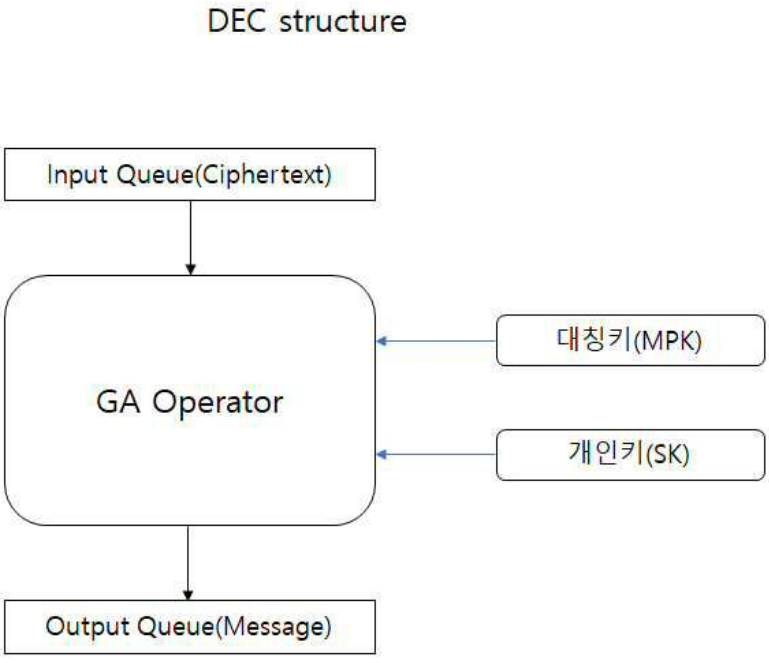
도면5


MAGIC CRYPTO SYSTEM(ENC/DEC)
A
B
Authentication
ENC
DEC
EN
C
DEC

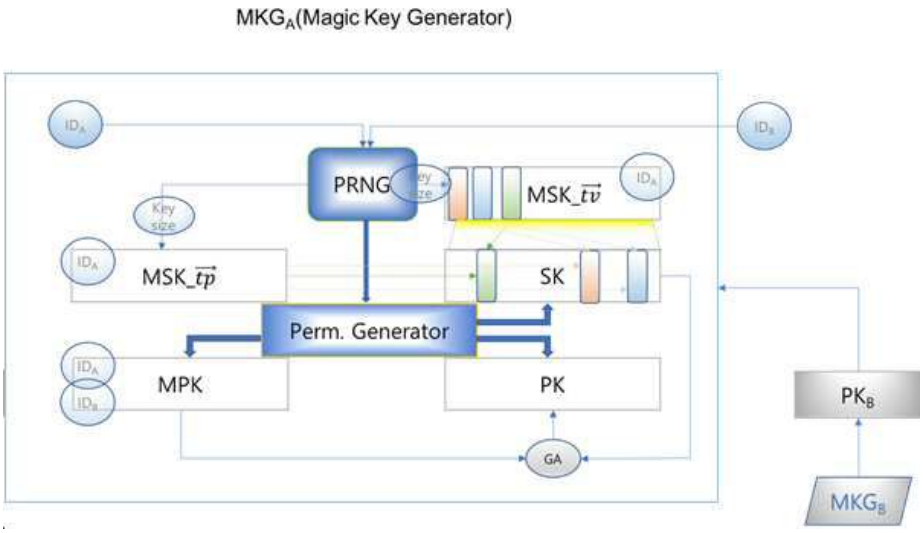
도면6


ENC structure
Input Queue(Message)
GA Operator
대칭키(MPK)
공개키(PK)
Output Queue(Ciphertext)

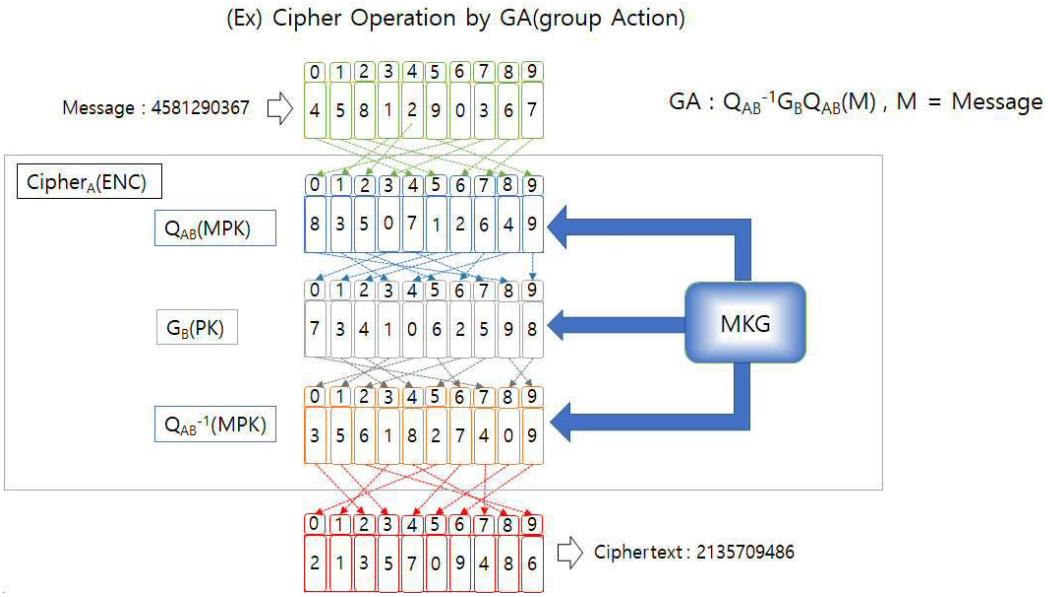
도면7


DEC structure
Input Queue(Ciphertext)
GA Operator
대칭키(MPK)
개인키(SK)
Output Queue(Message)

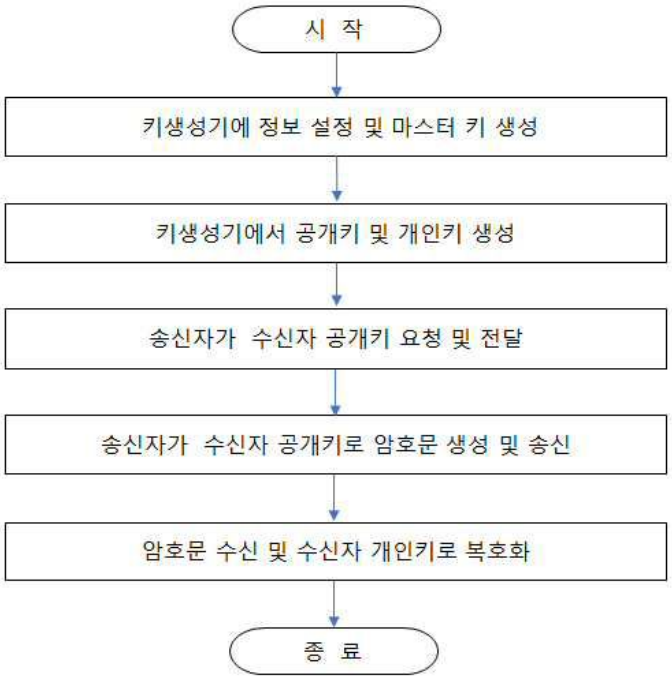
도면8


MKG$_A$(Magic Key Generator)
PRNG
MSK_$\vec{tv}$
MSK_$\vec{tp}$
SK
Perm. Generator
MPK
PK
GA
PK$_B$
MKG$_B$

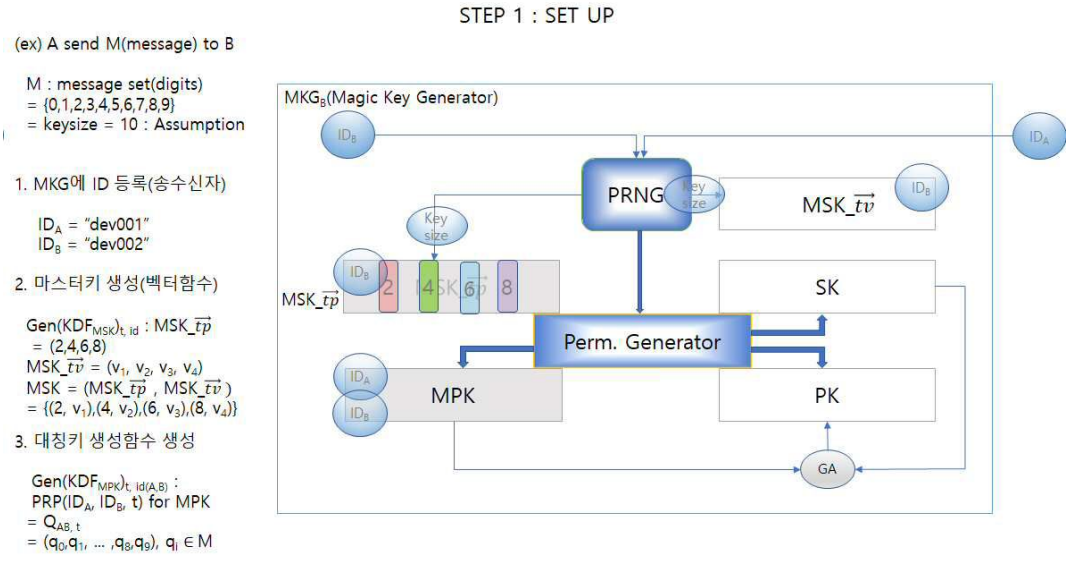
도면9


(Ex) Cipher Operation by GA(group Action)
Message : 4581290367
GA : $Q_{AB}^{-1}G_BQ_{AB}(M)$, M = Message
Cipher$_A$(ENC)
Q_{AB}(MPK)
G_B(PK)
Q_{AB}^{-1}(MPK)
MKG
Ciphertext : 2135709486

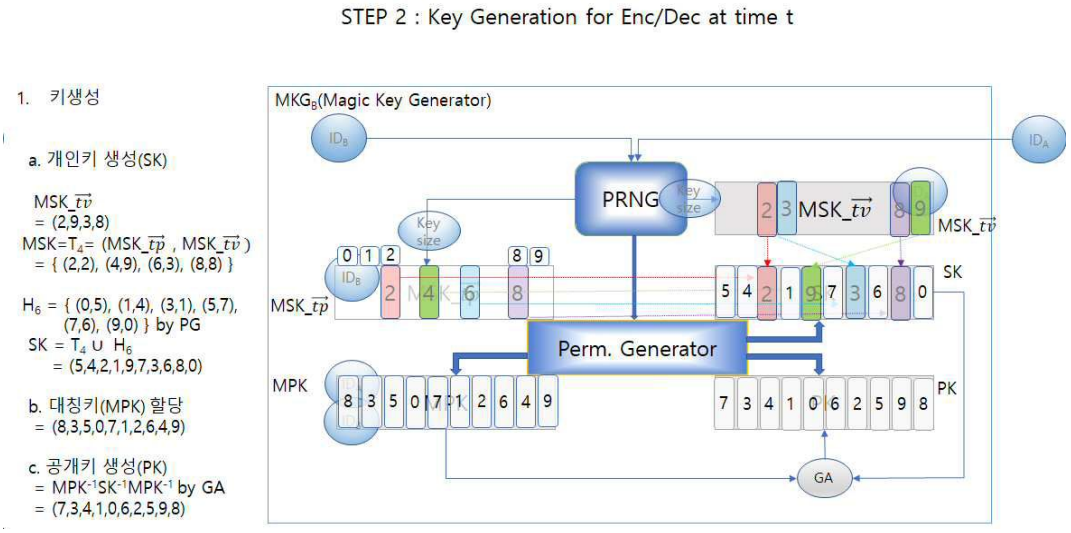
도면10


시 작
키생성기에 정보 설정 및 마스터 키 생성
키생성기에서 공개키 및 개인키 생성
송신자가 수신자 공개키 요청 및 전달
송신자가 수신자 공개키로 암호문 생성 및 송신
암호문 수신 및 수신자 개인키로 복호화
종 료

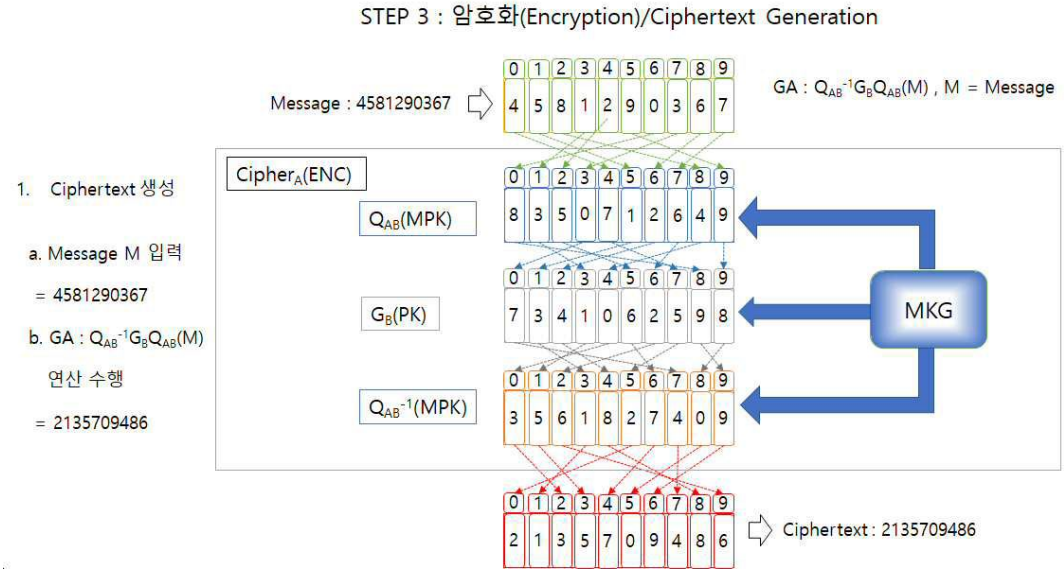
도면11


STEP 1 : SET UP
(ex) A send M(message) to B
M : message set(digits)
= {0,1,2,3,4,5,6,7,8,9}
= keysize = 10 : Assumption
1. MKG에 ID 등록(송수신자)
ID_A = "dev001"
ID_B = "dev002"
2. 마스터키 생성(벡터함수)
Gen(KDF_MSK)_t, id : MSK_tp
= (2,4,6,8)
MSK_tv = (v1, v2, v3, v4)
MSK = (MSK_tp , MSK_tv)
= {(2, v1),(4, v2),(6, v3),(8, v4)}
3. 대칭키 생성함수 생성
Gen(KDF_MPK)_t, id(A,B) :
PRP(ID_A, ID_B, t) for MPK
= Q_AB, t
= (q0,q1, ... ,q8,q9), qi ∈ M
MKG_B(Magic Key Generator)
PRNG
Key size
MSK_tv
MSK_tp
SK
Perm. Generator
MPK
PK
GA

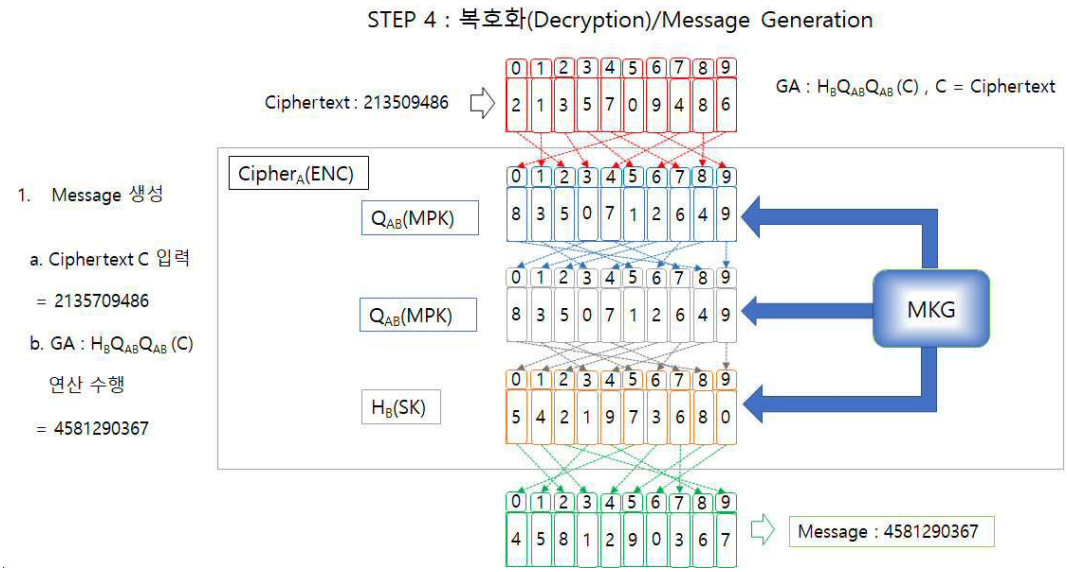
도면12


STEP 2 : Key Generation for Enc/Dec at time t
1. 키생성
a. 개인키 생성(SK)
MSK_$\vec{tv}$ = (2,9,3,8)
MSK=T_4= (MSK_$\vec{tp}$, MSK_$\vec{tv}$) = { (2,2), (4,9), (6,3), (8,8) }
H_6 = { (0,5), (1,4), (3,1), (5,7), (7,6), (9,0) } by PG
SK = T_4 ∪ H_6 = (5,4,2,1,9,7,3,6,8,0)
b. 대칭키(MPK) 할당 = (8,3,5,0,7,1,2,6,4,9)
c. 공개키 생성(PK) = $MPK^{-1}SK^{-1}MPK^{-1}$ by GA = (7,3,4,1,0,6,2,5,9,8)
MKG_B(Magic Key Generator)
PRNG
Key size
MSK_$\vec{tv}$
MSK_$\vec{tp}$
SK
Perm. Generator
MPK
PK
GA

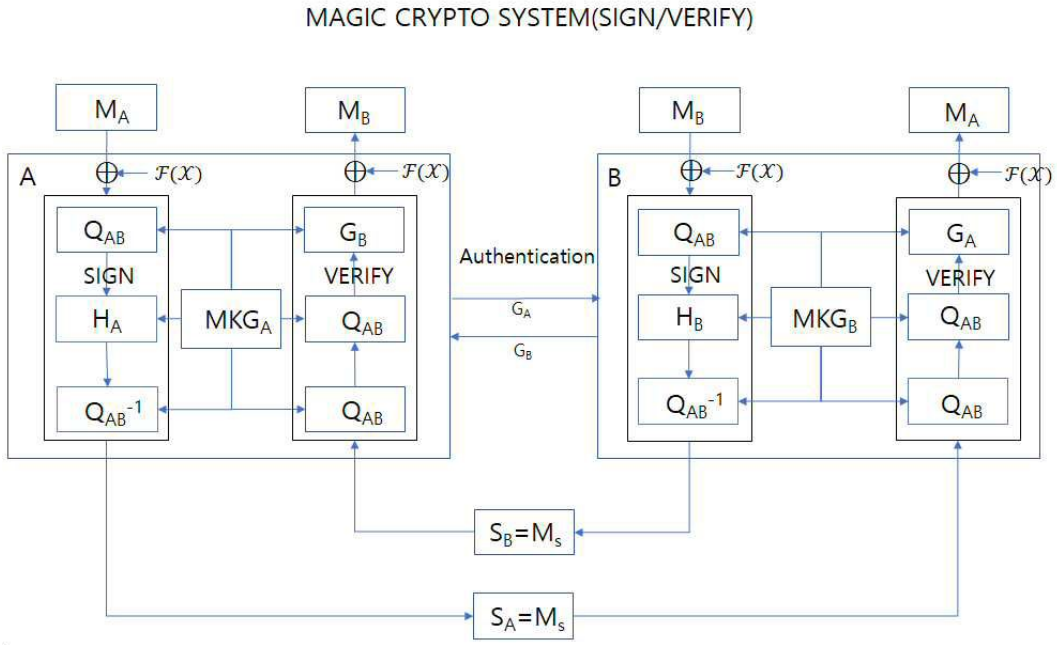
도면13


STEP 3 : 암호화(Encryption)/Ciphertext Generation
1. Ciphertext 생성
a. Message M 입력
= 4581290367
b. GA : $Q_{AB}^{-1}G_BQ_{AB}(M)$
연산 수행
= 2135709486
Message : 4581290367
GA : $Q_{AB}^{-1}G_BQ_{AB}(M)$, M = Message
Cipher_A(ENC)
Q_{AB}(MPK)
G_B(PK)
Q_{AB}^{-1}(MPK)
MKG
Ciphertext : 2135709486

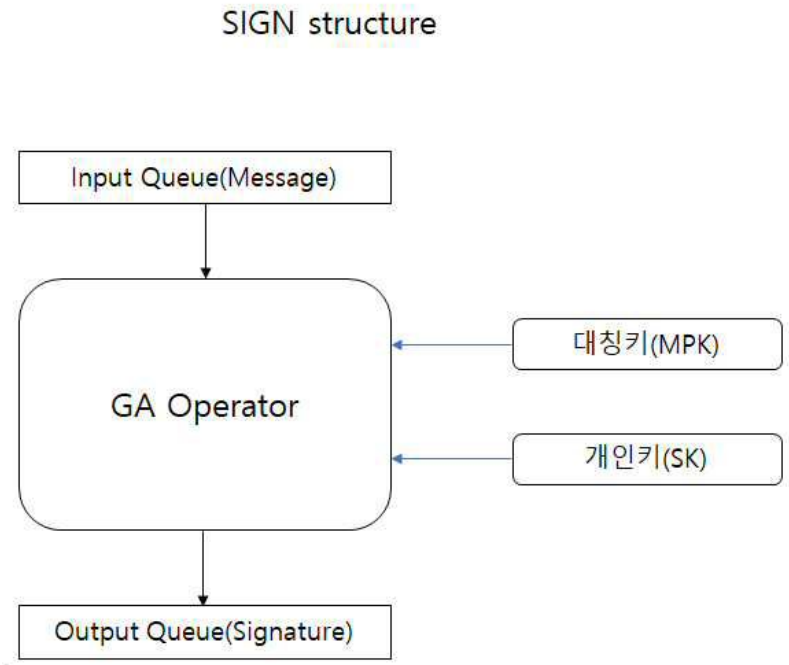
도면14


STEP 4 : 복호화(Decryption)/Message Generation
Ciphertext : 213509486
GA : $H_B Q_{AB} Q_{AB}$ (C) , C = Ciphertext
1. Message 생성
a. Ciphertext C 입력
= 2135709486
b. GA : $H_B Q_{AB} Q_{AB}$ (C)
연산 수행
= 4581290367
$Cipher_A$(ENC)
Q_{AB}(MPK)
Q_{AB}(MPK)
H_B(SK)
MKG
Message : 4581290367

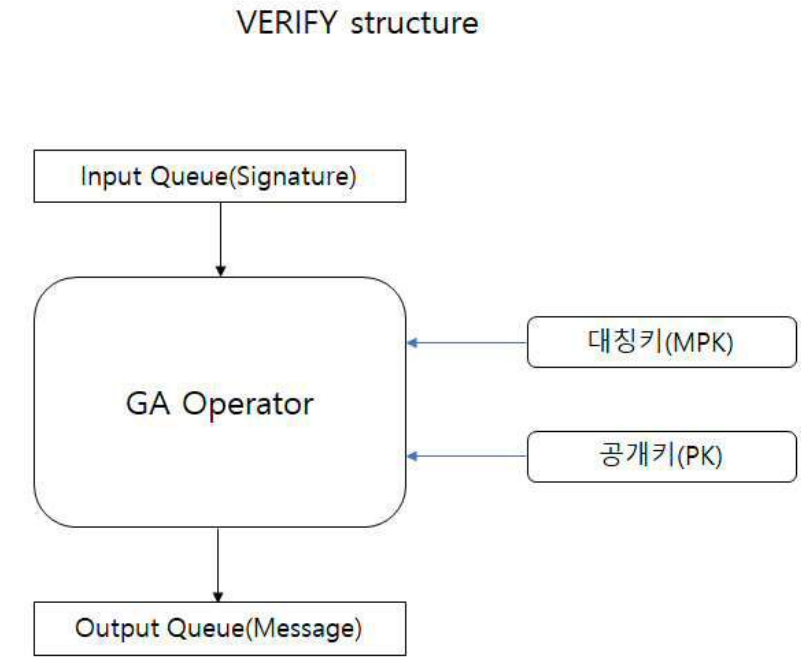
도면15


MAGIC CRYPTO SYSTEM(SIGN/VERIFY)
M_A
M_B
M_B
M_A
A
B
$\mathcal{F}(X)$
Q_{AB}
SIGN
H_A
Q_{AB}^{-1}
MKG_A
G_B
VERIFY
Q_{AB}
Authentication
G_A
G_B
H_B
MKG_B
G_A
$S_B=M_s$
$S_A=M_s$

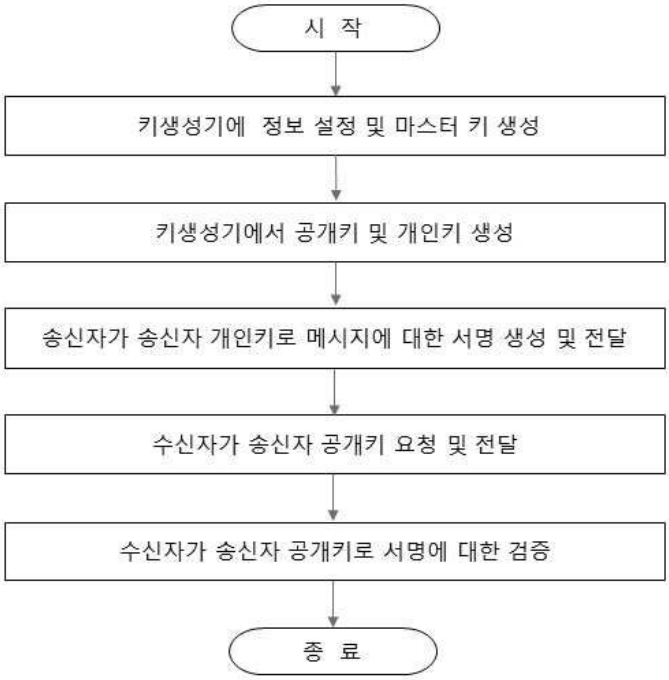
도면16


SIGN structure
Input Queue(Message)
GA Operator
대칭키(MPK)
개인키(SK)
Output Queue(Signature)

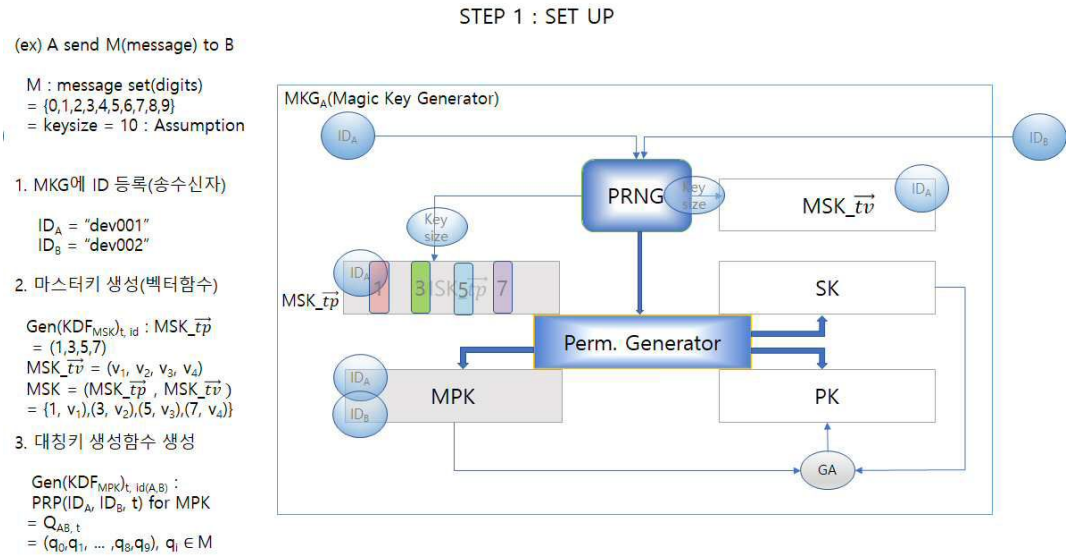
도면17


VERIFY structure
Input Queue(Signature)
GA Operator
대칭키(MPK)
공개키(PK)
Output Queue(Message)

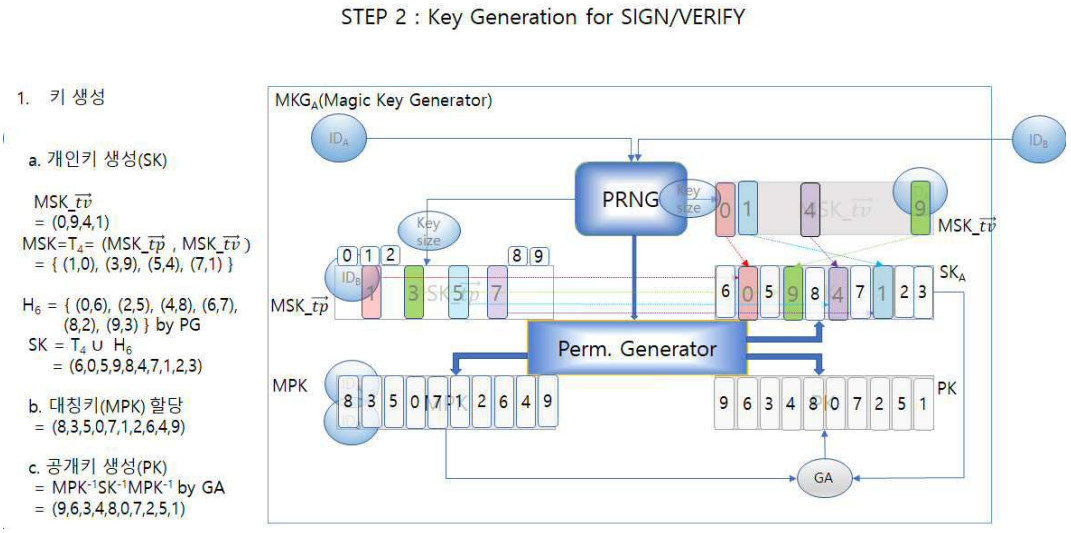
도면18


시 작
키생성기에 정보 설정 및 마스터 키 생성
키생성기에서 공개키 및 개인키 생성
송신자가 송신자 개인키로 메시지에 대한 서명 생성 및 전달
수신자가 송신자 공개키 요청 및 전달
수신자가 송신자 공개키로 서명에 대한 검증
종 료

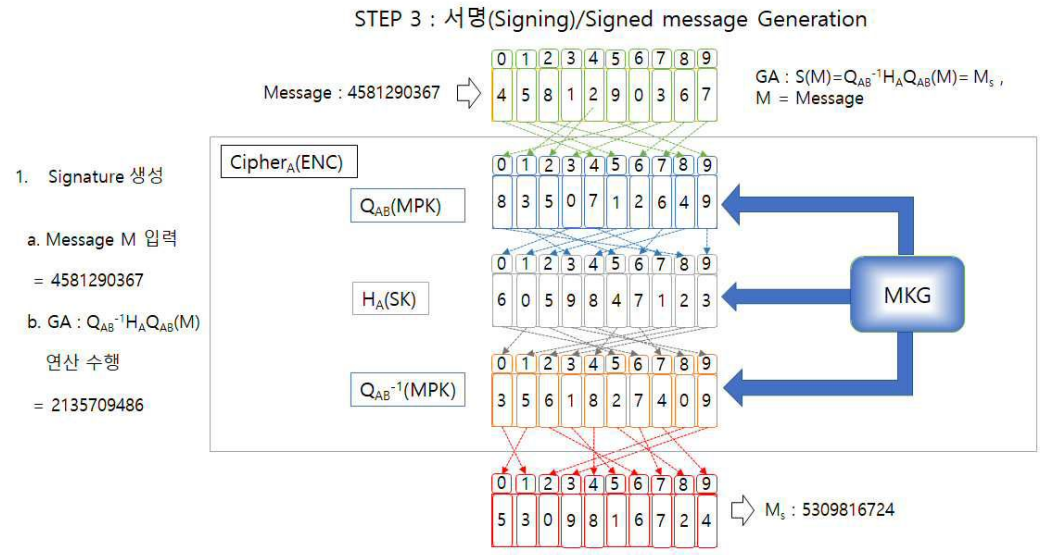
도면19


STEP 1 : SET UP
(ex) A send M(message) to B
M : message set(digits)
= {0,1,2,3,4,5,6,7,8,9}
= keysize = 10 : Assumption
1. MKG에 ID 등록(송수신자)
ID_A = "dev001"
ID_B = "dev002"
2. 마스터키 생성(벡터함수)
Gen(KDF_MSK)_t, id : MSK_tp
= (1,3,5,7)
MSK_tv = (v1, v2, v3, v4)
MSK = (MSK_tp , MSK_tv)
= {1, v1),(3, v2),(5, v3),(7, v4)}
3. 대칭키 생성함수 생성
Gen(KDF_MPK)_t, id(A,B) :
PRP(ID_A, ID_B, t) for MPK
= Q_AB, t
= (q0,q1, ... ,q8,q9), qi ∈ M
MKG_A(Magic Key Generator)
ID_A
ID_B
PRNG
Key size
MSK_tv
SK
MSK_tp
Perm. Generator
MPK
PK
GA

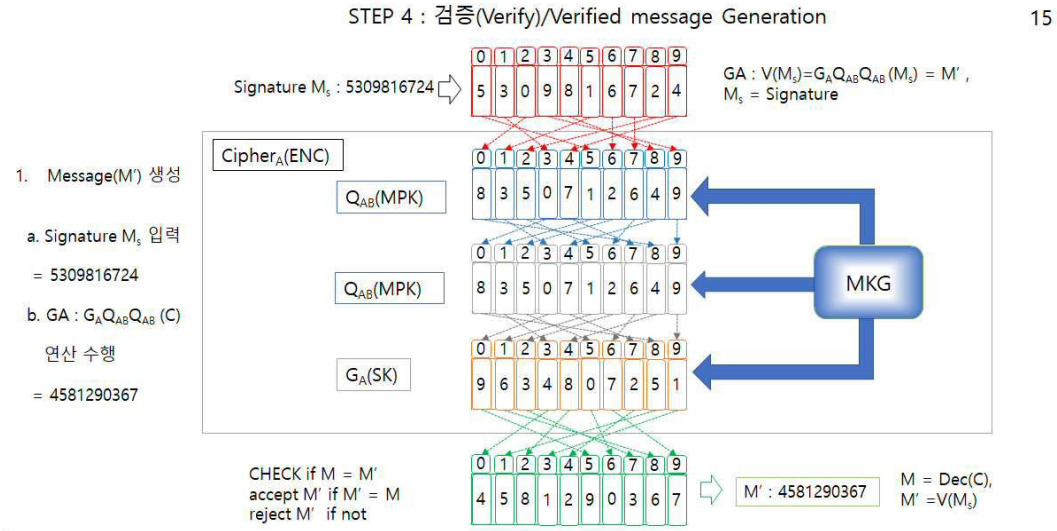
도면20


STEP 2 : Key Generation for SIGN/VERIFY
1. 키 생성
a. 개인키 생성(SK)
MSK_$\overrightarrow{tv}$ = (0,9,4,1)
MSK=T_4= (MSK_$\overrightarrow{tp}$, MSK_$\overrightarrow{tv}$) = { (1,0), (3,9), (5,4), (7,1) }
H_6 = { (0,6), (2,5), (4,8), (6,7), (8,2), (9,3) } by PG
SK = T_4 ∪ H_6 = (6,0,5,9,8,4,7,1,2,3)
b. 대칭키(MPK) 할당 = (8,3,5,0,7,1,2,6,4,9)
c. 공개키 생성(PK) = $MPK^{-1}SK^{-1}MPK^{-1}$ by GA = (9,6,3,4,8,0,7,2,5,1)
MKG$_A$(Magic Key Generator)
PRNG
Key size
MSK_$\overrightarrow{tv}$
MSK_$\overrightarrow{tp}$
SK$_A$
Perm. Generator
MPK
PK
GA

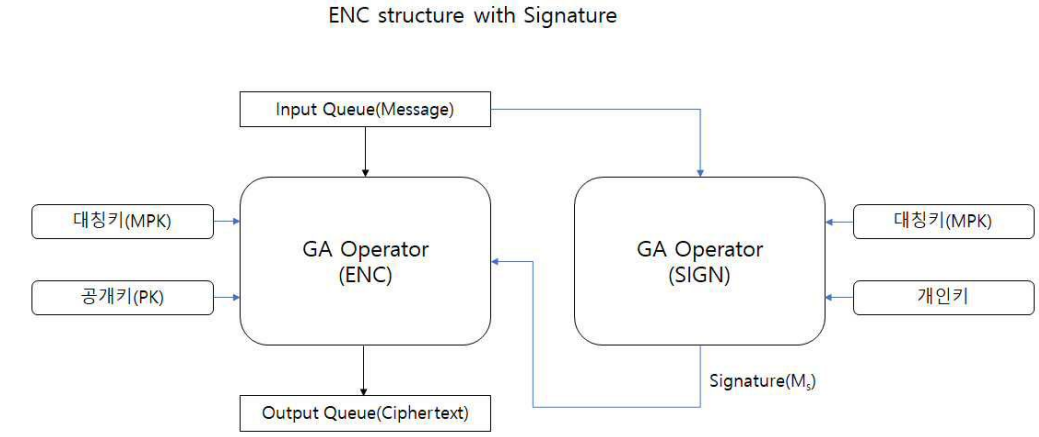
도면21


STEP 3 : 서명(Signing)/Signed message Generation
1. Signature 생성
a. Message M 입력
= 4581290367
b. GA : $Q_{AB}^{-1}H_AQ_{AB}(M)$ 연산 수행
= 2135709486
Message : 4581290367
GA : $S(M)=Q_{AB}^{-1}H_AQ_{AB}(M)= M_s$, M = Message
Cipher$_A$(ENC)
Q_{AB}(MPK)
H_A(SK)
Q_{AB}^{-1}(MPK)
MKG
M_s : 5309816724

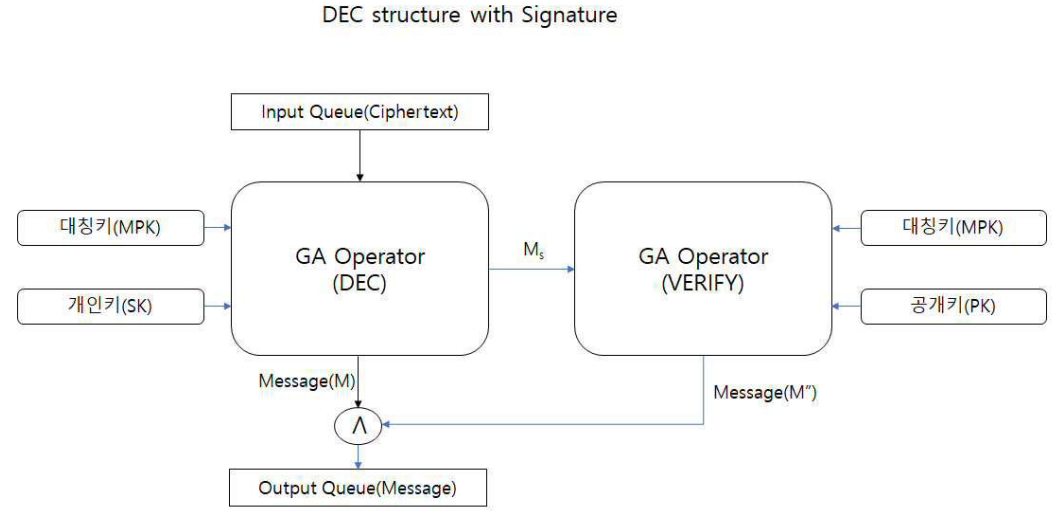
도면22



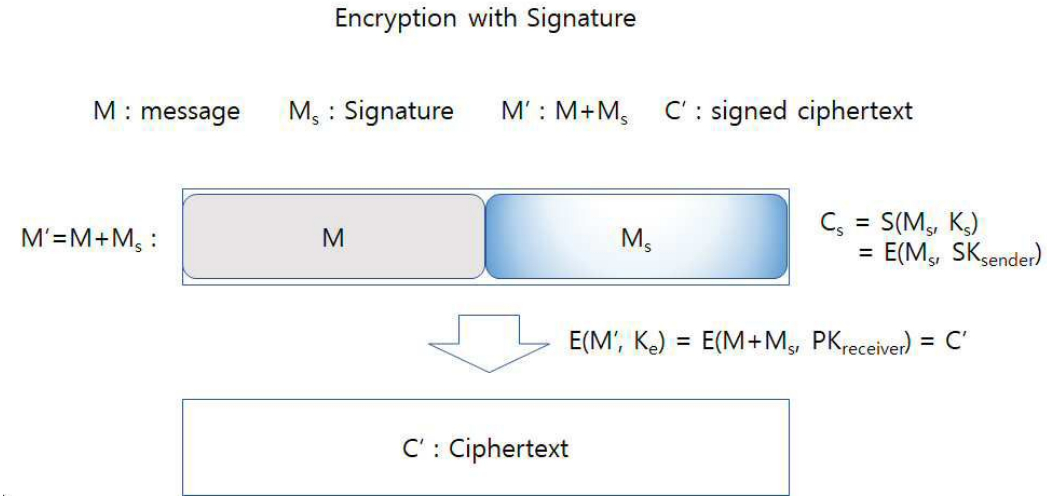
도면23



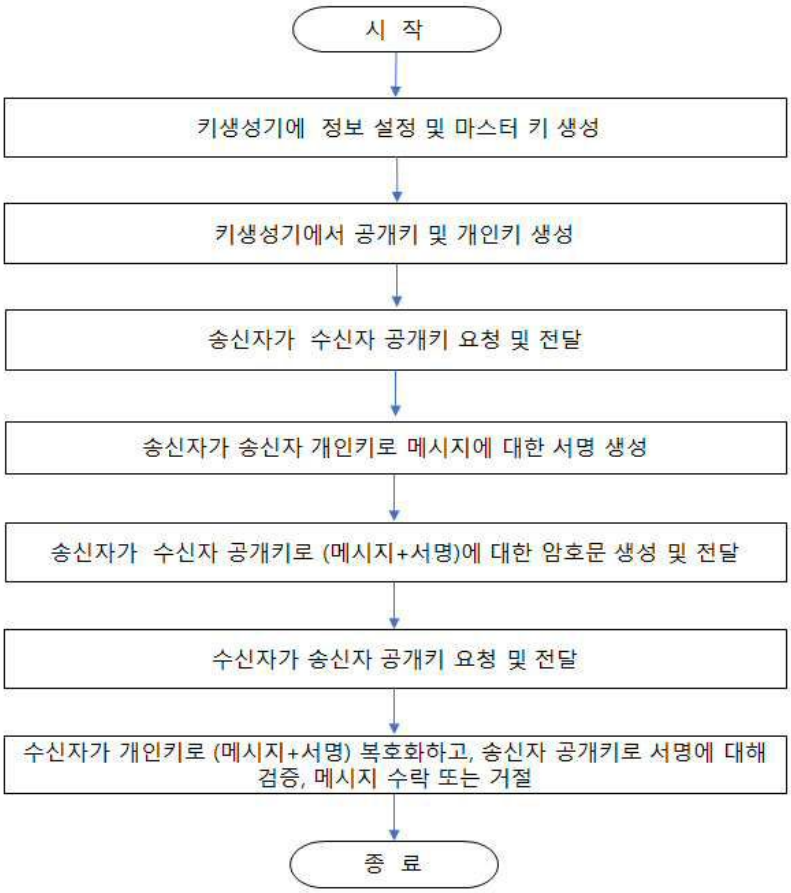
도면24



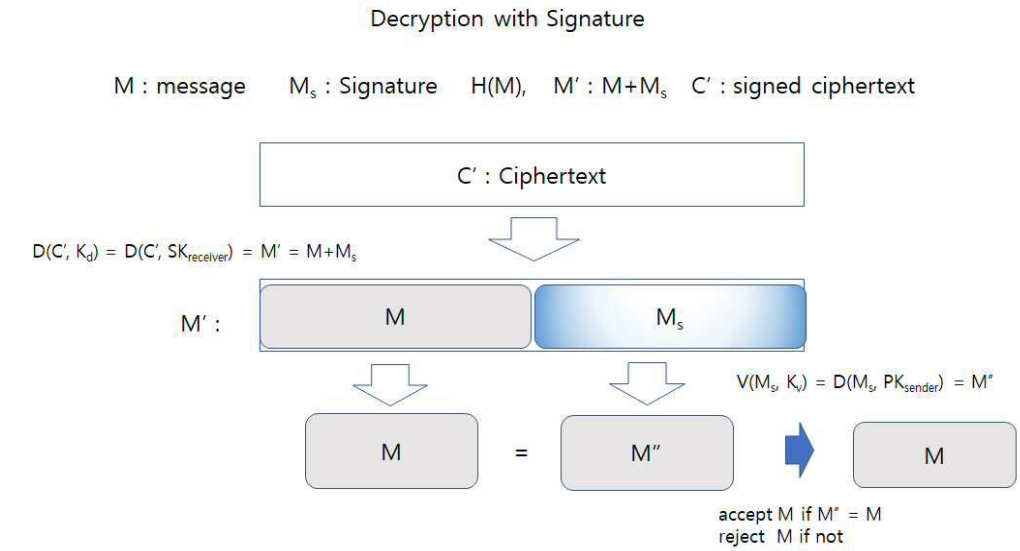
도면25


Encryption with Signature
M : message
M_s : Signature
M' : M+M_s
C' : signed ciphertext
M'=M+M_s :
M
M_s
C_s = $S(M_s, K_s)$
= $E(M_s, SK_{sender})$
E(M', K_e) = E(M+M_s, $PK_{receiver}$) = C'
C' : Ciphertext

도면26


시 작
키생성기에 정보 설정 및 마스터 키 생성
키생성기에서 공개키 및 개인키 생성
송신자가 수신자 공개키 요청 및 전달
송신자가 송신자 개인키로 메시지에 대한 서명 생성
송신자가 수신자 공개키로 (메시지+서명)에 대한 암호문 생성 및 전달
수신자가 송신자 공개키 요청 및 전달
수신자가 개인키로 (메시지+서명) 복호화하고, 송신자 공개키로 서명에 대해 검증, 메시지 수락 또는 거절
종 료

도면27


Decryption with Signature
M : message
M_s : Signature
H(M),
M' : M+M_s
C' : signed ciphertext
C' : Ciphertext
$D(C', K_d) = D(C', SK_{receiver}) = M' = M+M_s$
M' :
M
M_s
$V(M_s, K_v) = D(M_s, PK_{sender}) = M''$
M
=
M''
M
accept M if M'' = M
reject M if not