(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2010-0085424
(43) 공개일자 2010년07월29일

(51) Int. Cl.
H04L 9/08 (2006.01) *H04W 12/04* (2009.01)
(21) 출원번호 10-2009-0004700
(22) 출원일자 2009년01월20일
심사청구일자 2009년01월20일

(71) 출원인
성균관대학교산학협력단
경기 수원시 장안구 천천동 300 성균관대학교내
(72) 발명자
최형기
서울특별시 서초구 반포본동 반포아파트 53동 106호
김정윤
경기도 수원시 장안구 천천동 성균관대학교 제2공학관 27313호 인터넷보안연구실
(뒷면에 계속)
(74) 대리인
김인철, **특허법인세하**

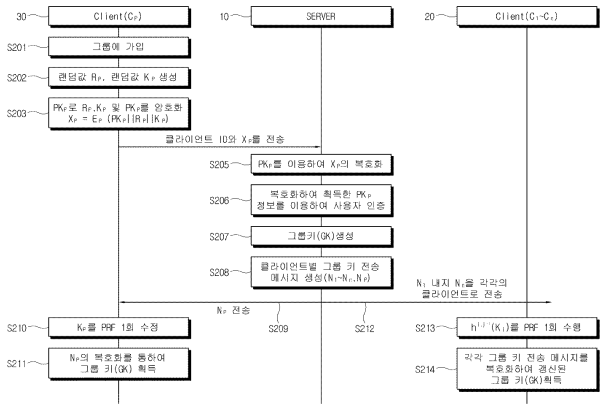
전체 청구항 수 : 총 26 항

(54) 그룹키 분배 방법 및 이를 위한 서버 및 클라이언트

(57) 요 약

본 발명은 그룹키 분배 방법 및 이를 위한 서버 및 클라이언트에 관한 것으로, 제1 클라이언트는 제1 랜덤값 및 제2 랜덤값을 생성하고, 서버와 공유하는 비밀키를 암호화하여 서버로 전송하는 단계, 서버는 암호화된 값을 복호화하여 사용자 인증을 수행하는 단계, 서버는 서버 랜덤값을 생성하고 그룹에 속해있는 클라이언트들의 제1 랜덤값을 이용하여 그룹키를 생성하는 단계, 서버는 생성된 그룹키로부터 클라이언트들 별 그룹키 보안 메시지를 생성하여 각 클라이언트로 전송하는 단계 및 그룹에 속해 있는 각각의 클라이언트는 그룹키 보안 메시지를 이용하여 그룹키를 획득하는 단계를 포함하는 것을 특징으로 하는 그룹키 분배 방법 및 이를 위한 서버 및 클라이언트를 제공함으로써 안전하고 빠른 키 분배를 구현할 수 있다.

대 표 도



(72) 발명자

임이진

대구광역시 수성구 지산1동 청구아파트 102동 708호

신기은

경기도 부천시 원미구 상동 사랑마을아파트 1618-304

특허청구의 범위

청구항 1

그룹키 분배 방법에 있어서,

제1 클라이언트는 제1 랜덤값 및 제2 랜덤값을 생성하고, 상기 제1 랜덤값, 제2 랜덤값 및 서버와 상기 제1 클라이언트가 공유하는 비밀키를 암호화하여 서버로 전송하는 단계;

상기 서버는 상기 암호화된 값을 복호화하여 상기 제1 랜덤값, 제2 랜덤값 및 비밀키 정보를 획득하는 단계;

상기 서버는 서버 랜덤값을 생성하고, 상기 서버 랜덤값과 상기 그룹에 속해있는 적어도 하나 이상의 클라이언트들의 제1 랜덤값을 이용하여 그룹키를 생성하는 단계;

상기 서버는 상기 생성된 그룹키로부터 상기 그룹에 속해있는 적어도 하나 클라이언트들 별 그룹키 보안 메시지를 생성하고 이를 그룹에 속해 있는 각각의 클라이언트로 전송하는 단계; 및

상기 그룹에 속해 있는 각각의 클라이언트는 자신의 제1 랜덤값, 제2 랜덤값 및 상기 그룹키 보안 메시지를 이용하여 그룹키를 획득하는 단계를 포함하는 것을 특징으로 하는 그룹키 분배 방법.

청구항 2

제1항에 있어서,

상기 제1 클라이언트가 상기 그룹에서 탈퇴한 경우,

상기 서버는 새로운 서버 랜덤값을 생성하고, 상기 서버 랜덤값과 상기 그룹에 남아있는 적어도 하나 이상의 클라이언트 별 제1 랜덤값을 이용하여 새로운 그룹키를 생성하는 단계;

상기 서버는 상기 새롭게 생성된 그룹키를 이용하여 상기 그룹에 남아있는 적어도 하나 이상의 클라이언트 별로 그룹키 보안 메시지를 생성하여, 각각의 클라이언트로 전송하는 단계; 및

상기 그룹에 남아있는 적어도 하나 이상의 클라이언트는 상기 그룹키 보안 메시지로부터 상기 새롭게 생성된 그룹키를 추출하는 단계를 더 포함하는 그룹키 분배 방법.

청구항 3

제1항 또는 제2항에 있어서,

상기 제1 클라이언트가 비밀키 정보를 암호화하는 단계는,

상기 제1 랜덤값, 제2 랜덤값 및 및 서버와 제1 클라이언트가 공유하는 비밀키를 상기 비밀키를 이용하여 암호화하는 것을 특징으로 하는 그룹키 분배 방법.

청구항 4

제1항 또는 제2항에 있어서,

상기 복호화하여 획득한 상기 제1 랜덤값, 제2 랜덤값 및 비밀키 정보를 이용하여 상기 제1 클라이언트의 인증을 수행하는 단계를 더 포함하는 그룹키 분배 방법.

청구항 5

제4항에 있어서,

상기 서버가 제1 클라이언트의 인증을 수행하는 단계는,

상기 복호화하여 획득한 비밀키와 상기 서버가 가지고 있던 제1 클라이언트의 비밀키 정보를 비교함으로써 상기 제1 클라이언트의 인증을 수행하는 것을 특징으로 하는 그룹키 분배 방법.

청구항 6

제1항 또는 제2항에 있어서,

상기 서버 랜덤값과 상기 그룹에 속해있는 적어도 하나 이상의 클라이언트들의 제1 랜덤값을 이용하여 그룹키를 생성하는 단계는,

상기 서버 랜덤값과 상기 그룹에 속해있는 적어도 하나 이상의 클라이언트들의 제1 랜덤값을 XOR 연산하여 그룹키를 생성하는 것을 특징으로 하는 그룹키 분배 방법.

청구항 7

제6항에 있어서,

상기 서버가 그룹키 보안 메시지를 생성하는 단계는,

상기 생성된 그룹키, 클라이언트 별 제1 랜덤값 및 클라이언트별 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값을 XOR 연산하여 상기 그룹키 보안 메시지를 생성하는 것을 특징으로 하는 그룹키 분배 방법.

청구항 8

제7항에 있어서,

상기 서버는 클라이언트별 제2 랜덤값을, 상기 클라이언트별 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값으로 업데이트하는 것을 특징으로 하는 그룹키 분배 방법.

청구항 9

제8항에 있어서,

상기 클라이언트는 상기 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값, 고정 랜덤 값 및 그룹키 보안 메시지를 XOR 연산함으로써 그룹키를 획득하는 것을 특징으로 하는 그룹키 분배 방법.

청구항 10

제9항에 있어서,

상기 클라이언트는 자신이 가지고 있는 제2 랜덤값을, 상기 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값으로 업데이트하는 것을 특징으로 하는 그룹키 분배 방법.

청구항 11

그룹키 분배를 위한 서버에 있어서,

클라이언트로부터 전달되는 암호화 메시지를 복호화하여 제1 랜덤값, 제2 랜덤값 및 비밀키를 획득하는 복호화부;

서버의 제1 랜덤값을 생성하고, 상기 서버의 제1 랜덤값과 그룹에 속하는 적어도 하나 이상의 클라이언트 별 제

1 랜덤값을 이용하여 그룹키를 생성하는 그룹키 생성부; 및

상기 생성된 그룹키로부터 상기 그룹에 속하는 클라이언트 별로 그룹키 보안 메시지를 생성하여 이를 각각의 클라이언트로 전송하도록 하는 그룹키 보안 메시지 생성부를 포함하는 것을 특징으로 하는 서버.

청구항 12

제11항에 있어서,

상기 복호화부는,

상기 암호화 메시지를 전달한 클라이언트와 공유하고 있던 비밀키를 이용하여 상기 암호화 메시지를 복호화하는 것을 특징으로 하는 서버.

청구항 13

제11항에 있어서,

상기 복호화부가 획득한 비밀키를 이용하여 클라이언트의 인증을 수행하고, 인증이 수행한 경우 상기 그룹키 생성부로 하여금 그룹키를 생성토록 제어하는 사용자 인증부를 더 포함한 것을 특징으로 하는 서버.

청구항 14

제11항에 있어서,

상기 그룹키 생성부는,

상기 서버의 제1 랜덤값과 그룹에 속한 적어도 하나 이상의 클라이언트 별 제1 랜덤값을 XOR 연산하여 그룹키를 생성하는 것을 특징으로 하는 서버.

청구항 15

제11항 내지 제13항 중 어느 한 항에 있어서,

그룹에 속한 적어도 하나 이상의 클라이언트 별 제1 랜덤값, 제2 랜덤값 및 비밀키를 저장하기 위한 암호 저장부를 더 포함하는 것을 특징으로 하는 서버.

청구항 16

제15항에 있어서,

상기 그룹키 보안 메시지 생성부는,

상기 생성된 그룹키, 클라이언트별 제1 랜덤값 및 제2 랜덤값을 이용하여 클라이언트별 그룹키 보안 메시지를 생성하는 것을 특징으로 하는 서버.

청구항 17

제16항에 있어서,

상기 그룹키 보안 메시지 생성부는,

클라이언트 별 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값, 클라이언트 별 제1 랜덤값 및 상기 생성된 그룹키를 XOR 연산하여 클라이언트 별 그룹키 보안 메시지를 생성하는 것을 특징으로 하는 서버.

청구항 18

제17항에 있어서,

상기 일방향 함수는,

해쉬 함수(Hash Function) 또는 의사 난수 함수(Pseudo Random Function)인 것을 특징으로 하는 서버.

청구항 19

제17항에 있어서,

상기 그룹키 생성부는 클라이언트로부터 그룹 탈퇴 요청을 수신한 경우, 새로운 서버의 제1 랜덤값을 생성하여, 상기 새로운 서버의 제1 랜덤값과 그룹에 남아있는 클라이언트 별 제1 랜덤값을 이용하여 새로운 그룹키를 생성하고,

상기 그룹키 보안 메시지 생성부는 상기 새롭게 생성된 그룹키로부터 상기 그룹에 남아있는 클라이언트 별 그룹키 보안 메시지를 생성하여 각각의 클라이언트로 전송하토록 하는 것을 특징으로 하는 서버.

청구항 20

제19항에 있어서,

상기 그룹키 보안 메시지 생성부는,

상기 클라이언트 별 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값을 다시 기설정된 횟수만큼 일방향 함수에 적용한 값, 클라이언트별 제1 랜덤값 및 상기 새롭게 생성된 그룹키를 XOR 연산하여 클라이언트별 그룹키 보안 메시지를 생성하는 것을 특징으로 하는 서버.

청구항 21

제1 랜덤값과 제2 랜덤값을 생성하는 랜덤값 생성부;

상기 제1 랜덤값과 제2 랜덤값 및 서버와 공유하고 있는 비밀키를 암호화하고 상기 암호화된 메시지를 서버로 전송케 하는 암호화부; 및

서버로부터 그룹키가 암호화된 그룹키 보안 메시지를 수신하고 이를 복호화하여 그룹키를 획득하는 그룹키 보안 메시지 복호화부를 포함하는 것을 특징으로 하는 클라이언트.

청구항 22

제21항에 있어서,

상기 암호화부는,

상기 제1 랜덤값과, 제2 랜덤값 및 서버와 공유하고 있는 비밀키를 상기 비밀키를 이용하여 암호화하는 것을 특징으로 하는 클라이언트.

청구항 23

제21항 또는 제22항에 있어서,

상기 제1 랜덤값, 제2 랜덤값 및 비밀키를 저장하기 위한 암호 저장부를 더 포함하는 것을 특징으로 하는 클라

이언트.

청구항 24

제23항에 있어서,

상기 그룹키 보안 메시지 복호화부는,

상기 암호 저장부에 포함되어 있는 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값, 고정 랜덤 값 및 그룹키 보안 메시지를 이용하여 그룹키를 획득하는 것을 특징으로 하는 클라이언트.

청구항 25

제24항에 있어서,

상기 그룹키 보안 메시지 복호화부는,

상기 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값, 고정 랜덤 값 및 그룹키 보안 메시지를 XOR 연산함으로써 그룹키를 획득하는 것을 특징으로 하는 클라이언트.

청구항 26

제24항에 있어서,

상기 그룹키 보안 메시지 복호화부는,

상기 암호 저장부에 저장된 제2 랜덤값을, 상기 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값으로 업데이트하는 것을 특징으로 하는 클라이언트.

명 세 서

발명의 상세한 설명

기 술 분 야

[0001] 본 발명은 그룹키 분배 방법 및 이를 위한 서버 및 클라이언트에 관한 것이다.

배 경 기 술

[0002] 최근 네트워크 기술이 발전함에 따라 VoIP(Voice over Internet Protocol), IP-TV(Internet Protocol Television) 등 인터넷을 기반으로 하는 다양한 서비스들이 등장하였다. VoIP는 기존 인터넷 망을 활용함으로써 저렴한 비용의 전화 서비스가 가능하기 때문에 기존 PSTN(Public Switched Telephone Network) 기반의 전화를 대체할 새로운 서비스로 기재되고 있다. IP-TV는 기존의 방송 시스템과 달리 서비스 제공자와 사용자가 상호 작용함으로써 양방향 서비스를 제공할 수 있다. 이러한 양방향 통신은 사용자에 따라 차별화된 서비스를 제공한다.

[0003] 한편 VoIP 및 IP-TV 서비스는 모두 오픈 네트워크인 인터넷을 기반으로 제공되기 때문에 각종 위협에 노출되어 있다. 예를 들어, VoIP의 경우 통화 주체들 외의 다른 사용자들이 통화 내용을 도청할 수 없도록 통화 내용을 안전하게 보호해야 한다. 특히, 다자간 통화나 화상 회의의 경우 여러 사용자들이 통화에 참석할 수 있기 때문에 권한이 없는 사용자가 통화에 참여하거나 통화 내용을 도청할 수 없도록 접근 제어를 수행하여야 한다.

[0004] IP-TV의 경우, 서비스 제공자는 과금 처리가 완료된 합법적인 사용자만이 서비스를 수신할 수 있도록 하는 접근 제어를 필요로 한다. 즉, 합법적인 사용자들만이 공통으로 알고 있는 그룹키를 이용하여 컨텐츠를 암호화 함으로써 컨텐츠를 보호할 수 있다. 이 경우, 그룹키의 효율적인 재분배 문제를 해결해야 한다. 예를 들어, 과금 기간이 만료되거나, 혹은 가입 해지를 함으로써 더 이상 컨텐츠의 수신 권한이 없는 사용자가 발생할 경우, 서비스 제공자는 해당 사용자가 더 이상 컨텐츠를 수신할 수 없도록 기존의 그룹키를 갱신해야 하며, 갱신된 그룹키

를 합법적인 사용자들에게 다시 전송해야 한다.

[0005] 단말에게 그룹키를 분배하기 위한 방식에는, 키 전달 프로토콜과 키 동의 프로토콜이 존재한다. 키 전달 프로토콜은, 키 관리 서버가 그룹키를 생성 및 전달하는 방식을 의미한다. 키 동의 프로토콜은, 키 관리 서버 없이 단말들이 메시지를 송수신하여 그룹키를 공유하게 되는 방식을 의미한다. 우리는 기존에 연구된 키 전달 프로토콜과 키 동의 프로토콜을 분석하였고, 그 설명은 다음과 같다.

[0006] 1. 키 동의 프로토콜

[0007] 셔먼(Sherman) 등은 머클 트리(Merkle Tree)를 이용한 그룹키 분배 프로토콜인 One-way Function Trees(OFT)를 제안하였다. OFT의 뿌리 노드(Root Node)는 그룹키를 의미하고, 종단 노드(Leaf Node)는 단말과 키 관리 서버가 공유하는 비밀 값을 의미한다. 단말은 그룹키를 획득하기 위해, 먼저 종단 노드에 해당하는 비밀 값을 해쉬 함수로 연산한 값과, 형제 노드(Sibling Node)에 해당하는 비밀 값을 해쉬 함수로 연산한 값을 XOR 함으로써, 부모 노드(Parent Node)를 갖게 된다. 그리고 이 부모 노드에 대해서도 위와 같은 과정을 거쳐, 궁극적으로 그룹키인 뿌리 노드가 계산된다. OFT는, 단말의 가입이나 탈퇴가 발생할 때마다, 대칭 키 기반 암호화 알고리즘과 해쉬 함수를 각각 log2(n+1) 씩 수행해야 한다.

[0008] Jung은 연산 자원이 제한된 단말들이 효율적으로 그룹키를 공유하기 위한 키 동의 프로토콜을 제안하였다. Jung의 프로토콜은 디피-헬만 (Diffie-Hellman) 기반의 그룹키 동의 방식을 사용하고 있으며, XOR, 해쉬 등 가벼운 연산만으로 구현이 가능하다. 그러나 Lee 등은 Jung의 프로토콜에 보안 취약점이 존재한다는 사실을 증명하였다. Lee 등에 따르면, Jung의 프로토콜은 내부 공격자에 의한 서비스 거부 공격이 발생할 수 있다.

[0009] 2. 키 전달 프로토콜

[0010] Dondeti 등은 그룹의 계층 구조에 기반하여 그룹키를 분배하는 Dual-Encryption Protocol(DEP)을 제안하였다. 이 연구에 의하면, 그룹키를 효율적으로 갱신하기 위해서는, 하나의 그룹을 다수 개의 서브 그룹으로 분할해야 하며, 각 서브 그룹의 그룹키를 관리하는 서브 그룹 매니저가 존재해야 한다. 또한, 서브 그룹 매니저가 메시지를 수신할 권한이 없는 경우, 서브 그룹 매니저의 접근을 제한하기 위해 키 서버는 그룹 가입자에게만 알려진 그룹키를 이용하여 메시지를 암호화 한다. DEP는 서브 그룹 매니저의 접근 제어가 필요한 상황에는 적합하지만, 그렇지 않은 경우에는 불필요한 암호화에 의한 오버헤드가 발생한다.

[0011] Sun 등은 Pay-TV에 적합한 새로운 CAS를 제안하였다. 그들은 그룹키 전달에 사용되는 모든 값들을 사전에 오프라인으로 전달하는 방식을 사용하였다. 즉, 저자들은 단말마다 저장해야 하는 정보를 늘리는 대신, 전송 오버헤드를 감소시키는 그룹키 분배 프로토콜을 제안하였다. 시스템에 존재하는 모든 단말 Ci (1≤i≤n)에는 각 단말과 관련된 고유 정보 Ii (1≤i≤n)가 있으며, Ii는 Ci를 제외한 나머지 n-1개의 단말들이 모두 알고 있는 값이다. 만약 단말 Ci가 탈퇴하면, 나머지 n-1개의 단말들은 Ii를 기존 키에 XOR 함으로써 새로운 키를 획득하게 된다. 따라서, 탈퇴한 단말은 갱신된 키를 알 수 없고, 탈퇴한 단말을 제외한 모든 단말은 갱신된 키를 알 수 있다.

[0012] 그러나 이상에서 설명한 프로토콜 또는 접근 제어에 따라 통화 내용에 대한 암호화를 수행하게 되면, 통화 내용의 송수신에 지연이 발생할 수 있으며, 이는 통화의 품질을 떨어뜨리는 원인이 된다. 다자간 통화나 화상 회의의 경우에는 통화 주체가 셋 이상이 될 수 있기 보안에 소요되는 오버헤드가 일반 통화에 비해 더욱 클 수밖에 없다. 따라서 다자간 통화 및 화상 회의의 경우는 1:1 통화에 비해 통화 품질의 저하를 최소화시키면서 통화 내용을 안전하게 보호할 수 있는 방법이 더욱 절실히 요구된다.

발명의 내용

해결 하고자하는 과제

[0013] 따라서 본 발명은 상기한 종래 기술에 따른 문제점을 해결하기 위한 것으로, Pseudo Random Function과 XOR 연산과 같이 매우 빠른 연산만을 이용하고, Pseudo Random Function의 반복 사용에 의해 발생할 수 있는 전방향 안전성 문제를 XOR 연산을 통해 극복한 안전하고 효율적인 그룹키 분배 방법 및 이를 구현하기 위한 서버 및 클라이언트의 제공을 그 목적으로 한다.

과제 해결수단

[0014] 본 발명의 일 측면에 따른 그룹키 분배 방법은 제1 클라이언트는 제1 랜덤값 및 제2 랜덤값을 생성하고, 상기 제1 랜덤값, 제2 랜덤값 및 서버와 상기 제1 클라이언트가 공유하는 비밀키를 암호화하여 서버로 전송하는 단계; 상기 서버는 상기 암호화된 값을 복호화하여 상기 제1 랜덤값, 제2 랜덤값 및 비밀키 정보를 획득하는 단계; 상기 서버는 서버 랜덤값을 생성하고, 상기 서버 랜덤값과 상기 그룹에 속해있는 적어도 하나 이상의 클라이언트들의 제1 랜덤값을 이용하여 그룹키를 생성하는 단계; 상기 서버는 상기 생성된 그룹키로부터 상기 그룹에 속해있는 적어도 하나 클라이언트들 별 그룹키 보안 메시지를 생성하고 이를 그룹에 속해 있는 각각의 클라이언트로 전송하는 단계; 및 상기 그룹에 속해 있는 각각의 클라이언트는 자신의 제1 랜덤값, 제2 랜덤값 및 상기 그룹키 보안 메시지를 이용하여 그룹키를 획득하는 단계를 포함한다.

[0015] 이 경우 상기 제1 클라이언트가 상기 그룹에서 탈퇴한 경우, 상기 서버는 새로운 서버 랜덤값을 생성하고, 상기 서버 랜덤값과 상기 그룹에 남아있는 적어도 하나 이상의 클라이언트 별 제1 랜덤값을 이용하여 새로운 그룹키를 생성하는 단계; 상기 서버는 상기 새롭게 생성된 그룹키를 이용하여 상기 그룹에 남아있는 적어도 하나 이상의 클라이언트 별로 그룹키 보안 메시지를 생성하여, 각각의 클라이언트로 전송하는 단계; 및 상기 그룹에 남아있는 적어도 하나 이상의 클라이언트들은 상기 그룹키 보안 메시지로부터 새롭게 생성된 그룹키를 추출하는 단계를 더 포함할 수 있다.

[0016] 상기 제1 클라이언트가 비밀키 정보를 암호화하는 단계는, 상기 제1 랜덤값, 제2 랜덤값 및 및 서버와 제1 클라이언트가 공유하는 비밀키를 상기 비밀키를 이용하여 암호화하는 것을 특징으로 할 수 있다.

[0017] 본 발명에 따른 그룹키 분배 방법에는 상기 복호화하여 획득한 상기 제1 랜덤값, 제2 랜덤값 및 비밀키 정보를 이용하여 상기 제1 클라이언트의 인증을 수행하는 단계가 더 포함될 수 있다.

[0018] 상기 서버가 제1 클라이언트의 인증을 수행하는 단계는, 상기 복호화하여 획득한 비밀키와 상기 서버가 가지고 있던 제1 클라이언트의 비밀키 정보를 비교함으로써 상기 제1 클라이언트의 인증을 수행하는 것을 특징으로 할 수 있다.

[0019] 상기 서버 랜덤값과 상기 그룹에 속해있는 적어도 하나 이상의 클라이언트들의 제1 랜덤값을 이용하여 그룹키를 생성하는 단계는, 상기 서버 랜덤값과 상기 그룹에 속해있는 적어도 하나 이상의 클라이언트들의 제1 랜덤값을 XOR 연산하여 그룹키를 생성하는 것을 특징으로 할 수 있다.

[0020] 상기 서버가 그룹키 보안 메시지를 생성하는 단계는, 상기 생성된 그룹키, 클라이언트 별 제1 랜덤값 및 클라이언트별 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값을 XOR 연산하여 상기 그룹키 보안 메시지를 생성하는 것을 특징으로 할 수 있다.

[0021] 상기 서버는 클라이언트별 제2 랜덤값을, 상기 클라이언트별 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값으로 업데이트하는 것을 특징으로 할 수 있다. 상기 클라이언트는 상기 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값, 고정 랜덤 값 및 그룹키 보안 메시지를 XOR 연산함으로써 그룹키를 획득하는 것을 특징으로 할 수 있다. 상기 클라이언트는 자신이 가지고 있는 제2 랜덤값을, 상기 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값으로 업데이트하는 것을 특징으로 할 수 있다.

[0022] 본 발명의 다른 측면에 따른 그룹키 분배를 위한 서버는 클라이언트로부터 전달되는 암호화 메시지를 복호화하여 제1 랜덤값, 제2 랜덤값 및 비밀키를 획득하는 복호화부; 서버의 제1 랜덤값을 생성하고, 상기 서버의 제1 랜덤값과 그룹에 속하는 적어도 하나 이상의 클라이언트 별 제1 랜덤값을 이용하여 그룹키를 생성하는 그룹키 생성부; 및 상기 생성된 그룹키로부터 상기 그룹에 속하는 클라이언트 별로 그룹키 보안 메시지를 생성하여 이를 각각의 클라이언트로 전송하도록 하는 그룹키 보안 메시지 생성부를 포함한다.

[0023] 이 경우 상기 복호화부는, 상기 암호화 메시지를 전달한 클라이언트와 공유하고 있던 비밀키를 이용하여 상기 암호화 메시지를 복호화하는 것을 특징으로 할 수 있다.

[0024] 본 발명에 따른 서버는 상기 복호화부가 획득한 비밀키를 이용하여 클라이언트의 인증을 수행하고, 인증이 수행한 경우 상기 그룹키 생성부로 하여금 그룹키를 생성토록 제어하는 사용자 인증부를 더 포함할 수 있다.

[0025] 상기 그룹키 생성부는, 상기 서버의 제1 랜덤값과 그룹에 속한 적어도 하나 이상의 클라이언트 별 제1 랜덤값을 XOR 연산하여 그룹키를 생성하는 것을 특징으로 할 수 있다.

[0026] 본 발명에 따른 서버는 그룹에 속한 적어도 하나 이상의 클라이언트 별 제1 랜덤값, 제2 랜덤값 및 비밀키를 저장하기 위한 암호 저장부를 더 포함할 수 있다.

[0027] 이 경우 상기 그룹키 보안 메시지 생성부는, 상기 생성된 그룹키, 클라이언트별 제1 랜덤값 및 제2 랜덤값을 이용하여 클라이언트별 그룹키 보안 메시지를 생성하는 것이 바람직하다.

[0028] 또한 그룹키 보안 메시지 생성부는, 클라이언트 별 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값, 클라이언트 별 제1 랜덤값 및 상기 생성된 그룹키를 XOR 연산하여 클라이언트 별 그룹키 보안 메시지를 생성하는 것을 특징으로 할 수 있다. 상기 일방향 함수는, 해쉬 함수(Hash Function) 또는 의사 난수 함수(Pseudo Random Function)일 수 있다.

[0029] 상기 그룹키 생성부는 클라이언트로부터 그룹 탈퇴 요청을 수신한 경우, 새로운 서버의 제1 랜덤값을 생성하여, 상기 새로운 서버의 제1 랜덤값과 그룹에 남아있는 클라이언트 별 제1 랜덤값을 이용하여 새로운 그룹키를 생성하고, 상기 그룹키 보안 메시지 생성부는 상기 새롭게 생성된 그룹키로부터 상기 그룹에 남아있는 클라이언트별 그룹키 보안 메시지를 생성하여 각각의 클라이언트로 전송하토록 하는 것을 특징으로 할 수 있다.

[0030] 상기 그룹키 보안 메시지 생성부는, 상기 클라이언트 별 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값을 다시 기설정된 횟수만큼 일방향 함수에 적용한 값, 클라이언트별 제1 랜덤값 및 상기 새롭게 생성된 그룹키를 XOR 연산하여 클라이언트별 그룹키 보안 메시지를 생성할 수 있다.

[0031] 본 발명의 또 다른 측면에 따른 클라이언트는 제1 랜덤값과 제2 랜덤값을 생성하는 랜덤값 생성부; 상기 제1 랜덤값과 제2 랜덤값 및 서버와 공유하고 있는 비밀키를 암호화하고 상기 암호화된 메시지를 서버로 전송케 하는 암호화부; 및 서버로부터 그룹키가 암호화된 그룹키 보안 메시지를 수신하고 이를 복호화하여 그룹키를 획득하는 그룹키 보안 메시지 복호화부를 포함한다.

[0032] 상기 암호화부는, 상기 제1 랜덤값과, 제2 랜덤값 및 서버와 공유하고 있는 비밀키를 상기 비밀키를 이용하여 암호화하는 것을 특징으로 할 수 있다.

[0033] 본 발명에 따른 클라이언트는 상기 제1 랜덤값, 제2 랜덤값 및 비밀키를 저장하기 위한 암호 저장부를 더 포함할 수 있다.

[0034] 상기 그룹키 보안 메시지 복호화부는, 상기 암호 저장부에 포함되어 있는 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값, 고정 랜덤 값 및 그룹키 보안 메시지를 이용하여 그룹키를 획득하는 것을 특징으로 할 수 있다.

[0035] 상기 그룹키 보안 메시지 복호화부는, 상기 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값, 고정 랜덤 값 및 그룹키 보안 메시지를 XOR 연산함으로써 그룹키를 획득하는 것을 특징으로 할 수 있다.

[0036] 상기 그룹키 보안 메시지 복호화부는, 상기 암호 저장부에 저장된 제2 랜덤값을, 상기 제2 랜덤값을 기설정된 횟수만큼 일방향 함수에 적용한 값으로 업데이트하는 것을 특징으로 할 수 있다.

효 과

[0037] 상기한 바와 같이 본 발명에 따른 그룹키 분배 방법 및 이를 위한 서버 및 클라이언트를 이용하는 경우 PRF(Pseudo Random Function) 및 XOR(eXclusive) 연산만을 이용하므로 매우 빠른 연산이 가능하다. 특히 단말 측에서는 그룹 내 가입자의 수와 관계없이 PRF 연산 1회와 XOR 연산 2회만으로 그룹키의 갱신이 가능케 된다. 뿐만 아니라, 이 연산들 중 PRF 1회와 XOR 1회는 그룹키 갱신 과정에서 이루어질 필요가 없으며 미리 연산이 가능하므로(Pre-Computable) 보다 빠른 그룹키 갱신이 가능하게 된다.

[0038] 본 발명은 PRF를 이용하여 내부 및 외부 공격자에 의한 키의 노출을 차단하였으며, 반복적인 PRF의 사용시 발생할 수 있는 전방향 안전성의 문제도 PRF와 함께 XOR을 사용함으로써 극복할 수 있다.

[0039] 또한, PRF 및 XOR과 같이 가벼운 연산만을 사용함으로써 단말의 배터리 소모량도 최소화하였고, 단말에서 저장

해야 하는 정보의 크기도 매우 작기 때문에 단말의 실제 구현에 소요되는 비용도 매우 적다는 효과를 얻을 수 있다.

발명의 실시를 위한 구체적인 내용

[0040] 이하, 본 발명에 따른 그룹키 분배 방법 및 이를 위한 서버 및 클라이언트에 대하여 첨부된 도면을 참조하여 상세히 설명한다.

[0041] 도 1은 본 발명의 일 실시예에 따른 단말이 그룹 가입 또는 그룹 탈퇴하는 상황을 나타낸 도면이다.

[0042] 도 1에 도시된 바와 그룹(1)은 IP TV 그룹, VoIP 통화 그룹, 화상 회의 그룹 등 다양한 그룹을 의미한다. 본 발명은 기재된 그룹에만 적용되는 것이 아니며, 그룹키를 이용하여 그룹에 속한 클라이언트를 인증하는 그룹에는 모두 적용이 가능하다.

[0043] 도 1에 도시된 그룹(1)에는 현재 ID가 1 내지 n의 값을 가지는 클라이언트(20)와 상기 그룹(1)에 속한 클라이언트(20)를 관리하고 상기 클라이언트(20)에게 서비스를 제공하는 서버(10)가 존재한다. 여기서, n은 현재 그룹에 존재하는 클라이언트의 수를 나타낸다. n이 100이면 상기 그룹에는 100개의 단말이 존재하는 것이다.

[0044] 이와 같이 구성된 상태에서 클라이언트 C_p(30)가 가입을 하는 이벤트가 발생할 수 있다(① 과정). 또한, 그룹(1)에 가입되어 서비스를 제공받던 클라이언트 C_P(40)가 그룹(1)으로부터 탈퇴하는 이벤트가 발생할 수도 있다(② 과정).

[0045] 이와 같이 새롭게 클라이언트(C_p)가 새롭게 가입을 하거나, 그룹에 속해있던 클라이언트(C_P)가 탈퇴를 하게 되는 경우 서버(10)는 사용하고 있던 그룹키를 갱신하고 이를 그룹에 가입되어 있는 상태인 클라이언트에게 전달하게 된다.

[0046] 이하, 본 발명에 따른 그룹키 분배 과정에 대하여 살펴보기로 한다.

[0047] 도 2는 본 발명의 다른 실시예에 따른 클라이언트의 그룹 가입 시 그룹키 분배 방법을 나타낸 도면이다.

[0048] 먼저 클라이언트 C_p(30)는 도 1에 도시된 그룹(1)에 가입을 시도하게 된다(S201). 사용자 인증과 그에 따른 그룹키 생성을 위하여 클라이언트 C_p(30)는 제1 랜덤값(R_p)과 제2 랜덤값(K_p)을 생성한다(S202). 이 때 제1 랜덤값(R_p)과 제2 랜덤값(K_p)은 아래 식과 같이 생성될 수 있다.

[0049] $R_p \leftarrow \{0, 1\}^a$, $K_p \leftarrow \{0, 1\}^a$

[0050] 제1 랜덤값(R_p)과 제2 랜덤값(K_p)을 생성한 후 클라이언트 C_p(30)는 서버(10)와 공유하고 있는 비밀키인 PK_p을 이용하여 제1 랜덤값(R_p), 제2 랜덤값(K_p) 및 비밀키(PK_p)를 암호화한다(S203). S203 단계에서 암호화되어 생성되는 메시지를 X_p이라고 정의하며, 이는 $X_p \leftarrow E_{PKp}(PK_p \| R_p \| K_p)$와 같이 정의될 수 있다.

[0051] 이와 같이 PK_p를 이용하여 PK_p를 암호화하는 기법, 즉 Self-Encryption 기법은 비밀정보의 노출을 최소화하면서 동시에 사용자 인증을 가능하게 하는 기법으로, Challenge-Response 기법에서 사용되는 넌스(nonce), 타임스탬프(Timestamp)의 전송이 불필요하기 때문에 효율적인 인증을 가능하게 한다.

[0052] 그러나 Self-Encryption은 재전송 공격에 취약하다는 단점이 있다. 본 발명에서는 그룹키 분배에 사용되는 랜덤값을 Self-Encryption의 입력값에 포함시킴으로써 Self-Encryption에서 발생하게 되는 재전송 공격 문제를 해결한다. 또한, Self-Encryption을 사용함으로써 비밀정보의 노출을 최소화하는 동시에 불필요한 전송값을 최소화하고, 사용자 인증을 가능하게 한다.

[0053] S203 단계에서 사용되는 비밀키(PK_p)는 서버(10)와 클라이언트 C_p(30)가 사전에 공유하고 있는 값이거나, 공개키 기법을 이용하여 서버(10)가 클라이언트 C_p(30)에게 사전에 전달한 값이다.

[0054] 또한, 위 식에서 a는 보안 파라미터(Security Parameter)이며, 생성하는 랜덤값의 크기(Bit)를 나타낸다. 즉, a=127이라면 생성되는 R_p, K_p은 127의 비트를 가지게 되는 것이다.

[0055] 클라이언트 C_p(30)는 S203 단계에서 생성한 X_p와 자신의 ID를 서버(10)로 전송하여 사용자 인증을 요청한다(S204).

[0056] 서버(10)는 클라이언트 C_p(30)의 ID에 상응하는 PK_p를 가지고 있는 상태이다. 서버(10)는 자신이 소유하고 있던 PK_p를 이용하여 수신한 X_p를 복호화한다(S205). X_p를 복호화하게 되면 서버(10)는 암호화되었던 R_p, K_p 및 클라이언트 C_p(30)의 비밀키 PK_p를 획득할 수 있다.

[0057] 서버(10)는 X_p를 복호화하여 획득한 PK_p와 자신이 미리 저장하고 있던 PK_p를 비교함으로써 클라이언트 C_p(30)를 인증한다(S206). 만일 S206 단계에서 클라이언트 C_p(30)의 사용자 인증이 실패했다면 서버(10)는 그룹키를 갱신하지 않으며, 클라이언트 C_p(30)로 인증 실패를 알리고 동작을 종료한다.

[0058] 만일 S206 단계에서 클라이언트 C_p(30)의 인증이 성공적으로 이루어진 경우 서버(10)는 그룹키(GK)를 생성한다(S207).

[0059] 그룹키를 생성하기 위하여 서버(10)는 먼저 서버 랜덤값(R_0)을 생성한다. 그 후 서버(10)는 클라이언트 C_1 내지 C_n, 및 C_p로부터 전달받았던 R_1 내지 R_n와 R_p 및 서버 랜덤값(R_0)을 XOR(eXlusive OR) 연산하여 그룹키를 생성한다. 즉, 서버(10)는 서버 랜덤값(R_0)을 생성하여 아래의 식과 같이 그룹키를 생성한다.

수학식 1

[0060] $$GK = R_0 \otimes R_1 \otimes \ldots \otimes R_n \otimes R_p$$

[0061] 즉, 서버(10)는 자신이 생성하여 비밀리에 보관하는 서버 랜덤값(R_0)과 그룹(1)에 속하는 n+1개의 클라이언트들이 생성한 제1 랜덤값(R_1, R_2, ..., R_n, R_p)을 XOR 연산하는 것이다.

[0062] 그 후 서버(10)는 생성한 그룹키(GK)를 그룹에 속한 클라이언트에게 전송한다. 그룹키의 전송은 보안을 위해 암호화를 수행하여야 한다. 이를 위하여 서버(10)는 각각의 클라이언트 별로 그룹키를 전송하기 위한 그룹키 보안 메시지를 생성한다.

[0063] 서버(10)가 생성하는 그룹키 보안 메시지를 N_i라고 정의하기로 한다. 여기서 i는 1부터 n의 값 또는 p이다. 즉, N_i는 서버(10)가 클라이언트 C_i로 보내기 위한 메시지를 의미한다. 이 때 서버(10)는 아래 수학식을 이용하여 N_i를 생성한다.

수학식 2

[0064] $$N_i = h^{i,j}(K_i) \otimes GK \otimes R_i$$

[0065] 여기서 $\otimes$는 XOR(eXclusive OR) 연산을 의미한다.

[0066] 또한, $h^{i,j}(K_i)$값에서 i는 클라이언트 C_i를 의미한다. j는 클라이언트 C_i가 그룹키를 갱신한 횟수를 나타낸다. 현재의 예에서 클라이언트 C_p(30)은 그룹에 가입하여 첫 번째로 그룹키를 분배받는 상황이므로 $h^{p,1}(K_p)$를 연산하게 된다.

[0067] $h^{i,j}(K_i)$란 K_i의 값을 j번 해쉬 함수에 적용한 값을 의미한다. 여기서 해쉬 함수로서 PRF(Pseudo Random Function)를 이용하는 것이 바람직하다. 즉, $h^{i,j}(K_i) = h(h^{i,j-1}(K_i) = h(h(h^{i,j-2}(K_i) = \ldots$ 과 같이 정의되며,

$h^{i,1}(K_i)$는 K_i를 한 번 해쉬 함수(또는 PRF 함수)에 적용한 값을 의미한다.

[0068] 이와 같이 생성된 N_i 메시지는 상응하는 클라이언트 C_i에게 전송된다(S209, S212). 예를 들어, N_p 메시지는 클라이언트 C_p(30)에게, N_2 메시지는 클라이언트 C_2(22)에게 전송되는 것이다.

[0069] 클라이언트 C_p(30)는 N_p의 메시지를 복호화함으로써 그룹키를 획득하게 된다(S211). 이 때 클라이언트 C_p는 다음의 식을 이용하여 N_p 메시지를 복호화할 수 있다.

수학식 3

[0070] $$GK = N_p \otimes h^{p,j}(K_p) \otimes R_p$$

[0071] 여기서 N_p은 클라이언트 C_p(30)가 서버(10)로부터 수신한 그룹키 보안 메시지이며, R_p는 S202 단계에서 생성하여 기억하고 있던 값이다. 또한, $h^{p,j}(K_p)$에서 j=1의 값을 가지며 $h^{p,j}(K_p)$ 또는 $h^{p,1}(K_p)$은 K_p를 한 번 해쉬 함수(또는 PRF)에 적용한 값을 의미한다. 클라이언트 C_p(30)는 수학식 3을 수행하기 위하여 S202 단계에서 생성한 K_p에 대하여 해쉬 함수(PRF 함수)에 적용함으로써 $h^{p,1}(K_p)$의 값을 획득할 수 있다(S210). 이 때 S210 단계는 N_p 메시지를 수신하기 전에 수행되는 경우 그룹키 획득을 위한 연산 시간을 더 줄일 수 있어 보다 효율적이다.

[0072] 이와 같은 과정을 통하여 클라이언트 C_p(30)는 N_p 메시지로부터 그룹키(GK)를 획득할 수 있는 것이다. 그 후 클라이언트 C_p(30)는 해쉬 함수(또는 PRF)를 적용한 값인 $h^{p,1}(K_p)$의 값을 추후 발생가능한 그룹키 갱신에 대비하여 저장하여 둔다.

[0073] 한편, 기존부터 그룹에 속해있던 클라이언트들, 즉 클라이언트 C_1 내지 C_n 도 각각 N_1 내지 N_n의 그룹키 보안 메시지를 수신하게 된다(S212). N_1 내지 N_n의 그룹키 보안 메시지를 수신한 클라이언트(20)는 아래의 식을 이용하여 그룹키를 획득하게 된다(S214).

수학식 4

[0074] $$GK = N_i \otimes h^{i,j}(K_i) \otimes R_i$$

[0075] 여기서 N_i는 서버(10)가 클라이언트 C_i에게 전달한 그룹키 보안 메시지이며, R_i은 클라이언트 C_i가 그룹에 가입하기 위하여 생성했던 값을 의미한다.

[0076] $h^{i,j}(K_i)$는 위에서 설명한 바와 같이 $h^{i,j-1}(K_i)$를 한 번 해쉬 함수(또는 PRF)에 적용된 값이다. $h^{i,j-1}(K_i)$는 클라이언트 C_p(30)가 가입하기 직전 그룹키가 갱신될 때 클라이언트 C_i가 생성하여 저장해둔 값이다.

[0077] 클라이언트 C_i는 저장해둔 $h^{i,j-1}(K_i)$의 값에다가 한 번 해쉬 함수(또는 PRF)를 적용함으로써 $h^{i,j}(K_i)$의 값을 획득할 수 있다(S213).

[0078] 이들 값들을 이용하여 그룹에 계속 존재하던 클라이언트 C_i들도 새로운 그룹키(GK)를 획득할 수 있게 된다. 또한, 다음 그룹키 갱신을 위하여 클라이언트 C_i는 $h^{i,j-1}(K_i)$를 삭제하고, $h^{i,j}(K_i)$를 저장하여 둔다.

[0079] 도 3은 본 발명의 또 다른 실시예에 따른 클라이언트의 그룹 탈퇴 시 그룹키 분배 방법을 나타낸 도면이다.

[0080] 도 3의 실시예는 그룹에 존재하였던 클라이언트 C_p(40)가 그룹으로부터 탈퇴를 하는 경우이다. 가입과 탈퇴를 구분하기 위하여 클라이언트 C_p를 40으로 표현하였다. 물론 클라이언트 C_p(40)가 아니라 다른 클라이언트가 탈퇴를

하여도 그룹키 분배 방법은 동일하다.

[0081] 먼저 클라이언트 C_p(40)는 그룹의 탈퇴를 서버(10)로 통보한다(S301). 이 경우 서버(10)는 새로운 서버 랜덤값인 R_0`을 생성한다. 이 때 생성된 $R_0$`은 R_0와 구분하기 위하여 "`"를 마킹하였다. $R_0$`도 R_0와 마찬가지로 서버(10)만이 비밀리에 보관하는 값에 해당한다.

[0082] 그 후 서버(10)는 새롭게 생성한 서버 랜덤값 R_0`을 이용하여 새로운 그룹키(GK`)를 생성한다(S302). 이 때 서버(10)는 아래의 수학식 5의 (1) 또는 (2) 중 하나를 이용하여 새로운 그룹키(GK`)를 생성할 수 있다.

수학식 5

$$GK` = R_0` \otimes R_1 \otimes R_2 \otimes \ldots \otimes R_n \quad \ldots\ldots\ldots\ldots (1)$$

$$GK` = GK \otimes R_0` \otimes R_0 \otimes R_p \quad \ldots\ldots\ldots\ldots (2)$$

[0083]

[0084] 수학식 5의 (1) 식은 수학식 1과 크게 다르지 않으므로 그 설명을 생략한다. 즉, 서버(10)는 기존에 사용하던 그룹키에 새롭게 생성한 R_0`, 기존에 생성한 R_0 및 탈퇴를 요청한 클라이언트 C_p의 R_p값을 XOR 연산함으로써 새로운 그룹키를 생성하는 것이다.

[0085] 이 때 수학식 5 의 (2) 식은 수학식 5의 (1) 식으로부터 $GK` = GK \otimes R_0` \otimes R_0 \otimes R_p = (R_0 \otimes R_1 \otimes \ldots \otimes R_n \otimes R_p) \otimes R_0` \otimes R_0 \otimes R_p = R_0` \otimes R1 \otimes R_2 \otimes \ldots \otimes R_n$의 과정을 통해 유도될 수 있다.

[0086] 이 때 서버는 수학식 5의 (2) 식을 통하여 새로운 그룹키를 생성하는 것이 보다 바람직하다. 수학식 5의 (1) 식은 $\otimes$ 연산(XOR 연산)을 n-1 번 해야하지만, 수학식 5의 (2) 식은 $\otimes$ 연산(XOR 연산)을 3번만 하면 되기 때문이다.

[0087] 이와 같이 새로운 그룹키(GK`)를 생성한 후 서버(10)는 클라이언트 별 그룹키 보안 메시지를 생성한다. 물론 클라이언트 C_p(40)은 탈퇴를 요청하였으므로 서버(10)는 N_1 내지 N_n의 그룹키 보안 메시지만을 생성한다(S303).

[0088] 이 때 서버(10)는 아래 식과 같이 그룹키 보안 메시지를 생성할 수 있다.

수학식 6

$$N_i` = h^{i,j+1}(K_i) \otimes GK` \otimes R_i$$

[0089]

[0090] 이 때 GK`는 새롭게 생성된 그룹키이다. 또한 수학식 2와 비교할 때 j가 j+1로 바뀌었음을 알 수 있다. 즉, 서버(10)는 이전 그룹키 갱신 단계에서 저장해두었던 $h^{i,j}(K_i)$에 해쉬 함수(또는 PRF)를 한 번 적용한 결과값, 새롭게 생성된 그룹키(GK`) 및 클라이언트별 제1 랜덤값인 R_i를 XOR 연산하여 새로운 그룹키 전송 메시지를 생성하는 것이다.

[0091] 서버(10)는 생성한 N_1` 내지 $N_n$`의 그룹키 보안 메시지를 각각의 클라이언트C_1 내지 C_n에게 전송한다(S306).

[0092] 각각의 클라이언트는 서버(10)로부터 수신한 그룹키 보안 메시지를 복호화하여 갱신된 그룹키(GK`)를 획득한다(S307). 이를 위해 클라이언트 C_1 내지 C_n는 아래 수학식 7과 같은 연산을 수행한다.

수학식 7

$$GK` = N_i` \otimes h^{i,j+1}(K_i) \otimes R_i$$

[0093]

[0094] N_i`는 서버(10)로부터 수신한 그룹키 보안 메시지이며, $h^{i,j+1}(K_i)$는 이전 단계에서 저장해둔 $h^{i,j}(K_i)$를 한 번 해쉬 함수(또는 PRF)에 적용함으로써 획득할 수 있는 값이다. R_i는 클라이언트 C_i가 최초 그룹에 가입하는 단계에서 생성한 랜덤값이다.

[0095] 도 4는 본 발명의 또 다른 실시예에 따른 서버의 블록 구성을 나타낸 도면이다.

[0096] 도 4에 도시된 바와 같이 그룹키 분배를 위한 서버(100)는 그룹키 분배 기능과 관련하여 메시지 처리부(110), 그룹 관리부(120), 복호화부(130), 사용자 인증부(140), 클라이언트 키 저장부(150), 그룹키 생성부(160), 그룹키 보안 메시지 생성부(170) 등을 포함하여 구성될 수 있다.

[0097] 메시지 처리부(110)는 클라이언트 C_i로부터 메시지를 받아 그에 상응하는 처리를 수행하는 구성 요소이다.

[0098] 예를 들어 메시지 처리부(110)는 그룹 가입 요청 메시지 또는 그룹 탈퇴 요청 메시지를 수신한 경우, 이를 그룹 관리부(120)로 전달하여 그에 상응하는 처리를 하도록 제어한다.

[0099] 그룹 관리부(120)는 클라이언트 C_p로부터 그룹 가입 요청을 받은 경우, 그룹 가입 요청 메시지에 포함되어 있는 X_p와 클라이언트의 ID를 복호화부(130)로 전달한다. 복호화부(130)는 클라이언트 키 저장부(150)에 저장되어 있는 해당 클라이언트C_p의 비밀키(PK_p)를 로딩하여, 상기 X_p를 복호화한다.

[0100] X_p가 복호화되어 획득되는 제1 랜덤값(R_p), 제2 랜덤값(K_p) 및 비밀키(PK_p)는 사용자 인증부(140)로 전달된다. 사용자 인증부(140)는 X_p를 복호화하여 획득한 PK_p와 클라이언트 키 저장부(150)에 저장되어 있는 해당 클라이언트의 비밀키(PK_p)를 비교함으로써 사용자 인증을 수행한다.

[0101] 성공적으로 사용자 인증이 이루어진 경우 그룹 관리부(120)는 상기 클라이언트 C_p를 그룹에 가입시키는 일련의 동작을 수행한다. 그 후 그룹 관리부(120)는 새로운 그룹키의 생성 및 분배를 제어한다.

[0102] 그룹키 생성부(160)는 새로운 그룹키를 생성한다. 임의의 클라이언트가 그룹에 가입한 경우 새로운 그룹키를 생성하는 방법은 수학식 1에서 설명한 바와 같다. 생성된 그룹키는 그룹키 보안 메시지 생성부(170)로 전달된다.

[0103] 그룹키 보안 메시지 생성부(170)는 그룹키를 암호화하여 그룹에 속한 클라이언트에게 전송하기 위하여 수학식 2에 따라 그룹키 보안 메시지를 생성한다. 이와 같이 생성된 그룹키 보안 메시지는 메시지 처리부(110)를 통하여 그룹에 속한 클라이언트들로 전달된다.

[0104] 한편, 임의의 클라이언트(C_p)로부터 그룹 탈퇴 요청을 받은 경우, 그룹 관리부(120)는 클라이언트 C_p의 탈퇴를 처리하는 한편, 새로운 그룹키의 생성 및 분배를 제어한다.

[0105] 그룹키 생성부(160)는 수학식 5를 이용하여 새로운 그룹키를 생성한다. 특히 그룹키 생성부(160)는 수학식 5의 (2) 식을 이용하여 그룹키를 생성하는 것이 바람직하다.

[0106] 또한, 그룹키 보안 메시지 생성부(170)는 새롭게 생성된 그룹키를 수학식 6에 적용하여 그룹키 보안 메시지를 생성한다. 이와 같이 생성된 그룹키 보안 메시지는 메시지 처리부(110)를 통하여 그룹에 남아있는 클라이언트들로 전달된다.

[0107] 도 5는 본 발명의 또 다른 실시예에 따른 클라이언트의 블록 구성을 나타낸 도면이다.

[0108] 도 5에 도시된 바와 같이 클라이언트(200)는 메시지 처리부(210), 암호화 처리부(220), 랜덤값 생성부(230), 암호 저장부(240), 그룹키 보안 메시지 복호화부(250) 등을 포함하여 구성될 수 있다.

[0109] 암호 저장부(220)에는 클라이언트(200)가 서버와 사전에 공유하고 알고 있는 비밀키(PK_i)가 저장되어 있다. 클라이언트(200)의 그룹 가입을 위하여 랜덤값 생성부(230)는 두 개의 랜덤값, 즉 제1 랜덤값(R_i)와 제2 랜덤값(K_i)를 생성한다. 이와 같이 생성된 R_i와 K_i는 암호 저장부에(240)에 저장된다.

[0110] 암호화부(220)는 서버와 사전에 나누어 가진 비밀키인 PK_i를 이용하여 R_i와 K_i 및 PK_i를 암호화한다. 이와 같이

암호화한 메시지 X_i는 $E_{PK_i}(R_i \| K_i \| PK_i)$와 같이 정의될 수 있다.

[0111] 메시지 처리부(210)는 서버로 클라이언트(200)의 ID 및 암호화된 메시지 X_i를 전송한다. 이제 서버는 클라이언트(200)의 ID 및 암호화된 메시지 X_i를 이용하여 사용자 인증, 그룹키 생성 및 분배를 수행한다.

[0112] 클라이언트(200)의 메시지 처리부(210)는 서버로부터 그룹키 보안 메시지를 전송받아 그룹키 보안 메시지 복호화부(250)로 전달한다.

[0113] 그룹키 보안 메시지 복호화부(250)는 암호화부에 저장하고 있던 $h^{i,0}(K_i) = K_i$를 1회 해쉬 함수(또는 PRF)에 적용하고, 수학식 7을 이용하여 그룹키 보안 메시지를 복호화한다. 이러한 과정을 통해 그룹키 보안 메시지 복호화부(250)는 그룹키를 획득할 수 있다.

[0114] 그룹키 보안 메시지 복호화부(250)는 다른 클라이언트의 가입 또는 탈퇴로 인한 그룹키 갱신이 되는 경우를 대비하여 $h^{i,0}(K_i) = K_i$를 1회 해쉬 함수(또는 PRF)에 적용한 결과, 즉 $h^{i,1}(K_i)$를 암호 저장부(240)에 저장해 둔다.

[0115] 한편, 메시지 처리부(210)는 그룹으로부터의 탈퇴를 위해 그룹 탈퇴 요청 메시지를 서버로 전송할 수 있다. 이 경우 서버는 그룹 탈퇴 요청을 한 클라이언트를 제외한 클라이언트에게 갱신된 그룹키 보안 메시지를 전송하며, 그룹 탈퇴 요청을 한 클라이언트(200)는 별도의 그룹키 분배 과정에 참여하지 않는다.

[0116] 이상에서 대표적인 실시예를 통하여 본 발명에 대하여 상세하게 설명하였으나, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자는 상술한 실시예에 대하여 본 발명의 범주에서 벗어나지 않는 한도 내에서 다양한 변형이 가능함을 이해할 것이다. 그러므로 본 발명의 권리 범위는 설명된 실시예에 국한되어 정해져서는 안 되며, 후술하는 특허청구범위뿐만 아니라 이 특허청구범위와 균등한 것들에 의하여 정해져야 한다.

도면의 간단한 설명

[0117] 도 1은 본 발명의 일 실시예에 따른 단말이 그룹 가입 또는 그룹 탈퇴하는 상황을 나타낸 도면.

[0118] 도 2는 본 발명의 다른 실시예에 따른 클라이언트의 그룹 가입 시 그룹키 분배 방법을 나타낸 도면.

[0119] 도 3은 본 발명의 또 다른 실시예에 따른 클라이언트의 그룹 탈퇴 시 그룹키 분배 방법을 나타낸 도면.

[0120] 도 4는 본 발명의 또 다른 실시예에 따른 서버의 블록 구성을 나타낸 도면.

[0121] 도 5는 본 발명의 또 다른 실시예에 따른 클라이언트의 블록 구성을 나타낸 도면.

[0122] <도면의 주요 부분에 대한 부호 설명>

[0123] 100 : 서버 110 : 메시지 처리부

[0124] 120 : 그룹 관리부 130 : 복호화부

[0125] 140 : 사용자 인증부 150 : 클라이언트 키 저장부

[0126] 160 : 그룹키 생성부 170 : 그룹키 보안 메시지 생성부

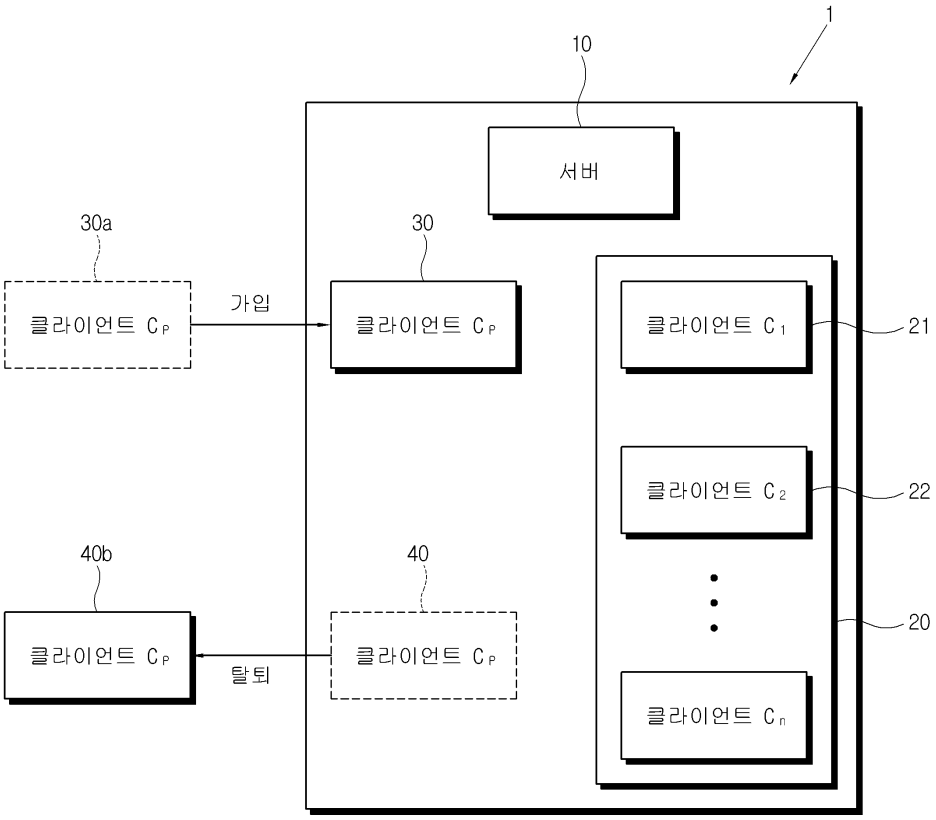
[0127] 200 : 클라이언트 210 : 메시지 처리부

[0128] 220 : 암호화부 230 : 랜덤값 생성부

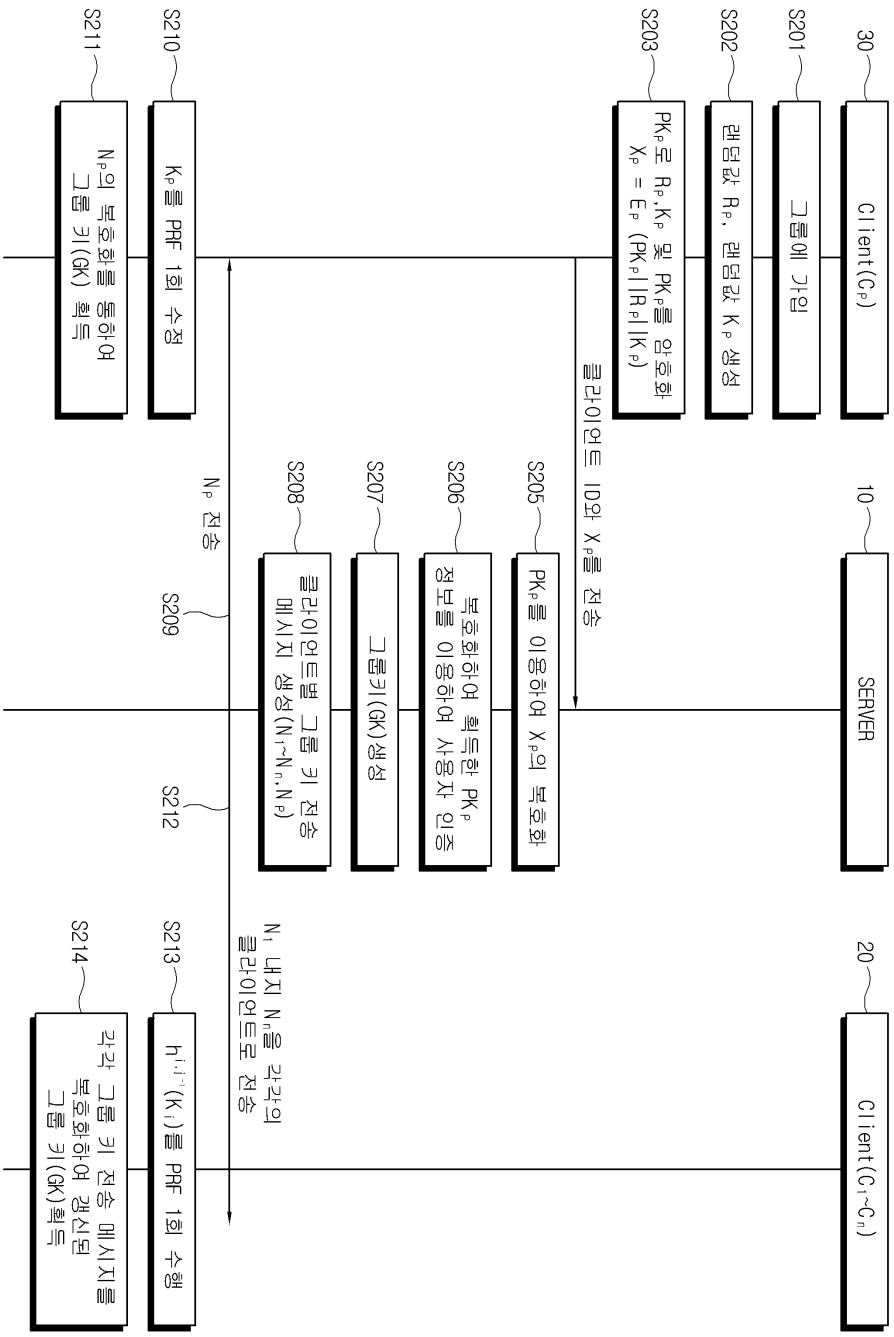
[0129] 240 : 암호 저장부 250 : 그룹키 보안 메시지 복호화부

도면

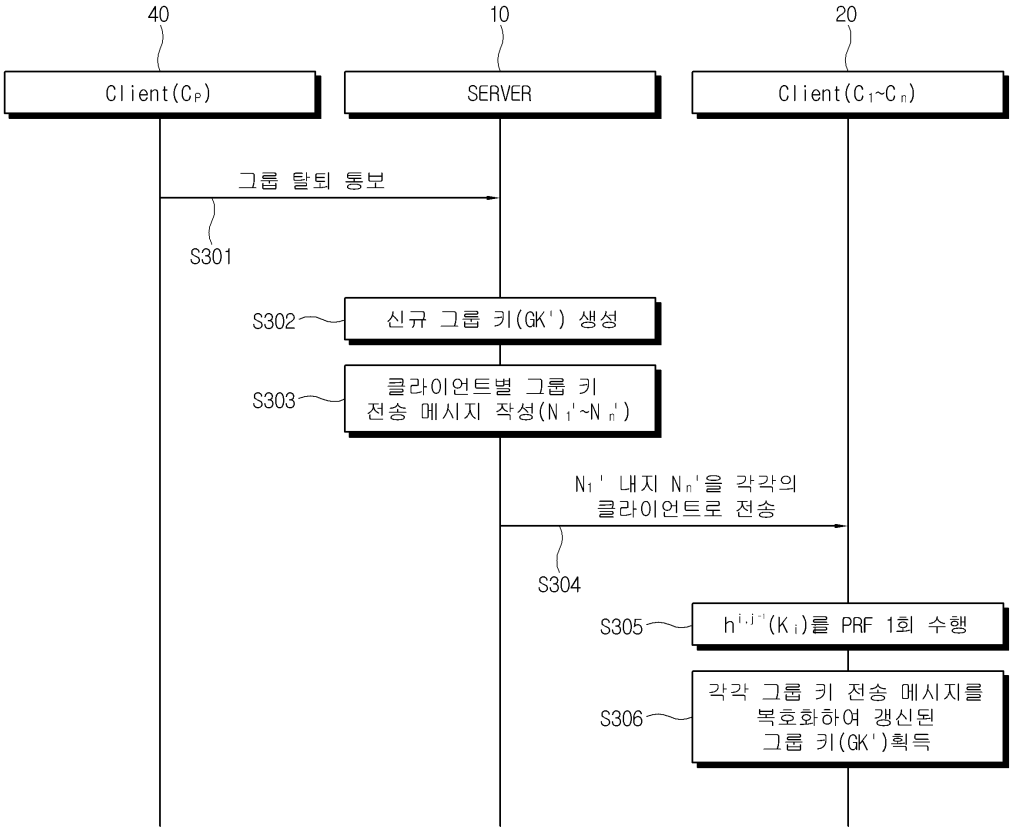
도면1


1
10
서버
30a
클라이언트 C_P
가입
30
클라이언트 C_P
클라이언트 C_1
21
클라이언트 C_2
22
20
클라이언트 C_n
40b
클라이언트 C_P
탈퇴
40
클라이언트 C_P

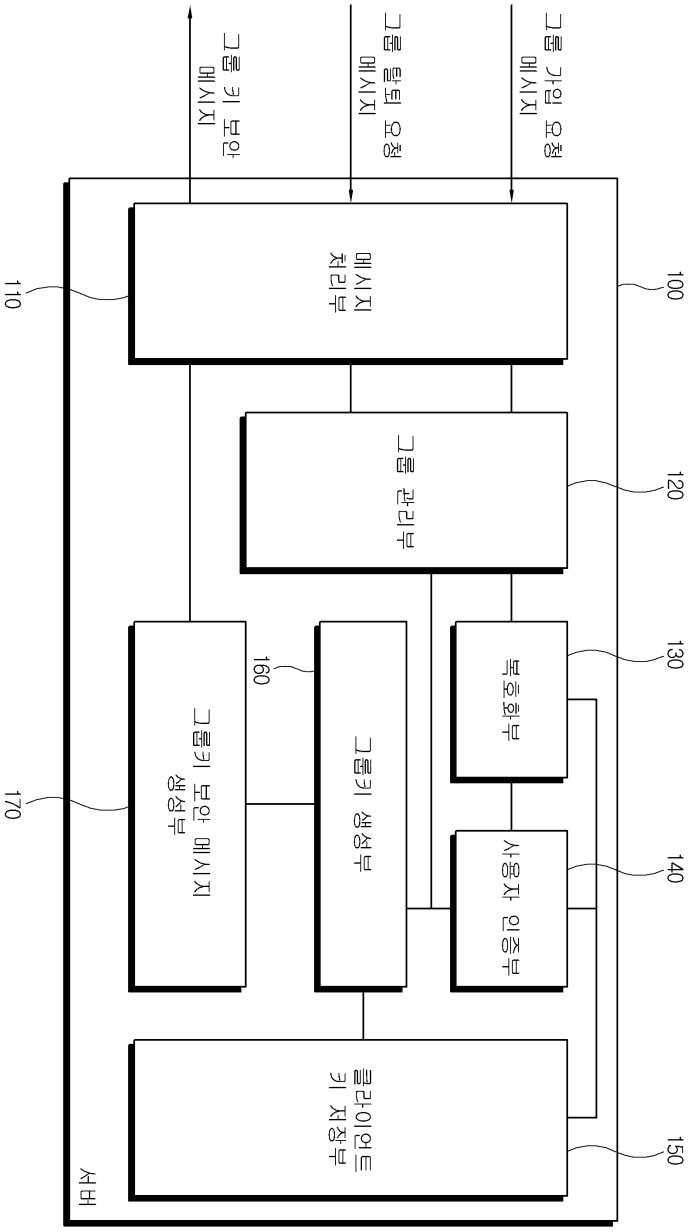
도면2


30
Client(Cp)
S201
그룹에 가입
S202
랜덤값 Rp, 랜덤값 Kp 생성
S203
PKp로 Rp,Kp 및 PKp를 암호화
Xp = Ep (PKp||Rp||Kp)
클라이언트 ID와 Xp를 전송
10
SERVER
S205
PKp를 이용하여 Xp의 복호화
S206
복호화하여 획득한 PKp
정보를 이용하여 사용자 인증
S207
그룹키(GK)생성
S208
클라이언트별 그룹 키 전송
메시지 생성(N1~Nn,Np)
Np 전송
S209
S210
Kp를 PRF 1회 수정
S211
Np의 복호화를 통하여
그룹 키(GK) 획득
S212
N1 내지 Nn을 각각의
클라이언트로 전송
20
Client(C1~Cn)
S213
h'·j'(Ki)를 PRF 1회 수행
S214
각각 그룹 키 전송 메시지를
복호화하여 갱신된
그룹 키(GK)획득

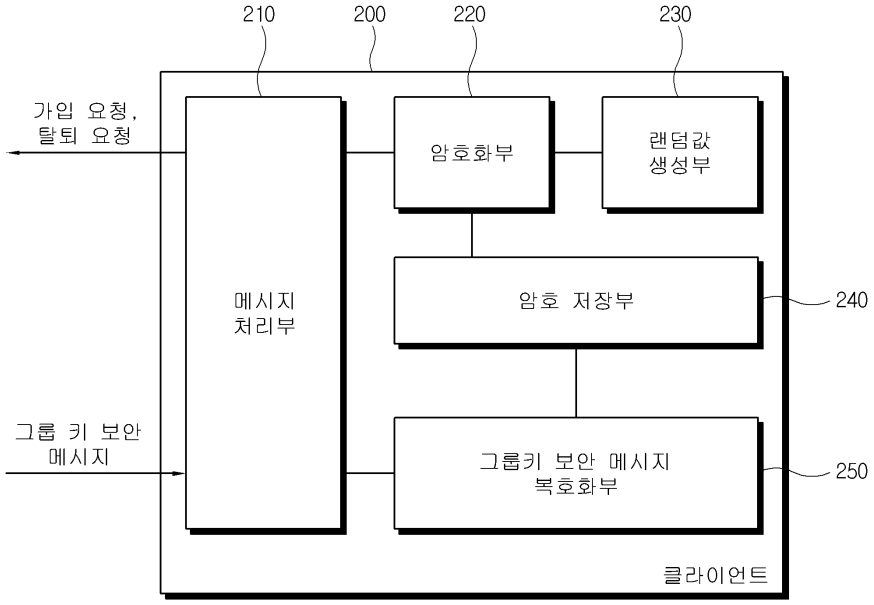
도면3


40
10
20
Client(Cp)
SERVER
Client(C1~Cn)
그룹 탈퇴 통보
S301
S302
신규 그룹 키(GK') 생성
S303
클라이언트별 그룹 키
전송 메시지 작성(N1'~Nn')
N1' 내지 Nn'을 각각의
클라이언트로 전송
S304
S305
h^[i,j^-1](Ki)를 PRF 1회 수행
S306
각각 그룹 키 전송 메시지를
복호화하여 갱신된
그룹 키(GK')획득

도면4


그룹 가입 요청 메시지
그룹 탈퇴 요청 메시지
그룹 키 보안 메시지
서버
메시지 처리부
그룹 관리부
복호화부
사용자 인증부
그룹키 생성부
그룹키 보안 메시지 생성부
클라이언트 키 저장부
100
110
120
130
140
150
160
170

도면5


210
200
220
230
가입 요청,
탈퇴 요청
암호화부
랜덤값
생성부
메시지
처리부
암호 저장부
240
그룹 키 보안
메시지
그룹키 보안 메시지
복호화부
250
클라이언트