(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2015년12월23일
(11) 등록번호 10-1580379
(24) 등록일자 2015년12월18일

(51) 국제특허분류(Int. Cl.)
H04L 9/32 (2006.01)
(21) 출원번호 10-2013-7027797
(22) 출원일자(국제) 2012년03월23일
심사청구일자 2013년10월23일
(85) 번역문제출일자 2013년10월22일
(65) 공개번호 10-2014-0002770
(43) 공개일자 2014년01월08일
(86) 국제출원번호 PCT/US2012/030352
(87) 국제공개번호 WO 2012/129503
국제공개일자 2012년09월27일
(30) 우선권주장
61/466,662 2011년03월23일 미국(US)
(뒷면에 계속)
(56) 선행기술조사문헌
KR1020100075603 A*
US20060020791 A1*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
인터디지탈 패튼 홀딩스, 인크
미국, 델라웨어주 19809, 윌밍턴, 벨뷰 파크웨이 200, 스위트 300
(72) 발명자
차 인혁
대한민국 서울 강남구 삼성동 14-1 중앙 하이츠 빌리지 102동 202호
구치오네 루이스 제이
미국 뉴욕주 10709 이스트 체스터 링컨 플레이스 211
(뒷면에 계속)
(74) 대리인
김태홍

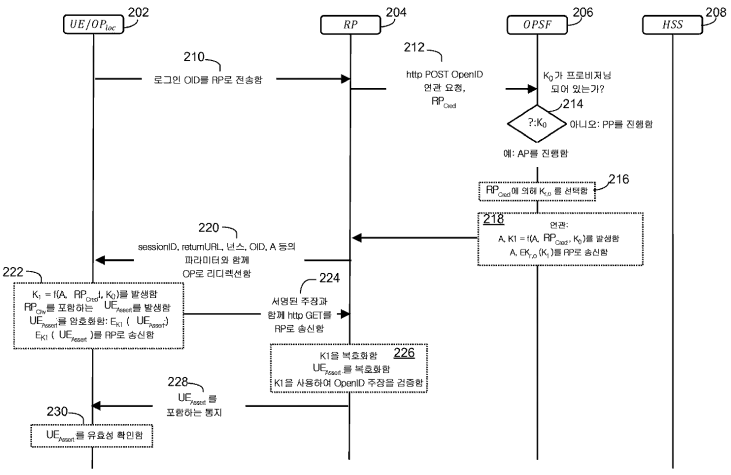
전체 청구항 수 : 총 10 항 심사관 : 양종필

(54) 발명의 명칭 **네트워크 통신 보호 시스템 및 방법**

(57) 요 약

네트워크 엔터티들의 인증 및/또는 검증을 수행하기 위해 네트워크 엔터티들 간에 보안 통신이 설정될 수 있다. 예를 들어, 사용자 장비(UE)는 사용자/UE의 인증을 위한 사용자 ID를 발행할 수 있는 ID 제공자와 보안 채널을 설정할 수 있다. UE는 또한 네트워크를 통해 UE에 서비스를 제공할 수 있는 서비스 제공자와 보안 채널을 설정할 수 있다. ID 제공자는 보안 통신을 수행하기 위해 심지어 서비스 제공자와 보안 채널을 설정할 수 있다. 이들 보안 채널 각각을 설정하는 것은 각각의 네트워크 엔터티가 다른 네트워크 엔터티에 대해 인증될 수 있게 해줄 수 있다. 보안 채널은 또한 UE가 자신과 보안 채널을 설정한 서비스 제공자가 서비스에 액세스하는 의도된 서비스 제공자임을 검증할 수 있게 해줄 수 있다.

대 표 도



(72) 발명자
슈미트 안드레아스
독일 프랑크푸르트 암 마인 65929 테토넨베그 37
라이셔 안드레아스
독일 프랑크푸르트 60385 하이드스트라쎄 131
샤 요겐드라 씨
미국 펜실베니아주 19341 엑스톤 리전시 커트 10

(30) 우선권주장
61/466,852 2011년03월23일 미국(US)
61/525,575 2011년08월19일 미국(US)

명 세 서

청구범위

청구항 1

삭제

청구항 2

삭제

청구항 3

삭제

청구항 4

삭제

청구항 5

삭제

청구항 6

삭제

청구항 7

삭제

청구항 8

삭제

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

청구항 12

삭제

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

삭제

청구항 22

삭제

청구항 23

삭제

청구항 24

삭제

청구항 25

삭제

청구항 26

사용자 장비(user equipment; UE), NAF(network application function), 및 BSF(bootstrapping server function)를 포함하는 시스템에서의 방법에 있어서,

상기 UE에서, 상기 UE와 상기 BSF 사이에 TLS(transport-layer security) 터널을 설정(establish)하는 단계 - 상기 TLS 터널은 그와 연관되어 있는 TLS 마스터 키를 가짐 -;

상기 BSF로부터의 신청(challenge)에 응답하여, 상기 UE에서, 랜덤한 넌스를 발생하고 인증 응답을 계산하는 단계;

상기 UE에서, 상기 UE의 사용자의 성공적인 인증을 나타내는 메시지를 수신하는 단계 - 상기 성공적인 인증의 결과, 제2 키가 도출됨 -; 및

상기 UE에서, 차후에 상기 UE와 상기 NAF 사이의 통신을 보호할 시에 사용하기 위한 제3 키를 도출하는 단계 - 상기 제3 키의 도출은 상기 제2 키 및 상기 TLS 마스터 키 둘 다에 적어도 부분적으로 의존함 - 를 포함하는, UE, NAF 및 BSF를 포함하는 시스템에서의 방법.

청구항 27

제26항에 있어서, 상기 인증 응답을 계산하는 단계는 하나 이상의 SIP 다이제스트(session initiation protocol

digest) 자격 증명(credential)을 사용하여 수행되는 것인, UE, NAF 및 BSF를 포함하는 시스템에서의 방법.

청구항 28

제26항에 있어서, 상기 제2 키는 GBA(generic bootstrapping architecture) 세션 키(Ks)이고, 상기 제3 키를 도출하는 단계는,

GBA 프로토콜을 사용하여 상기 Ks로부터 상기 제3 키를 도출하는 단계를 포함하고, 상기 제3 키는 응용프로그램 특유의 키(Ks_NAF)인 것인, UE, NAF 및 BSF를 포함하는 시스템에서의 방법.

청구항 29

제28항에 있어서, 상기 Ks_NAF를 사용하여 상기 UE와 상기 NAF 사이의 통신을 보호하는 단계를 더 포함하는, UE, NAF 및 BSF를 포함하는 시스템에서의 방법.

청구항 30

제28항에 있어서, 상기 Ks는 무결성 키(integrity key) 또는 기밀성 키(confidentiality key) 중 적어도 하나를 포함하는 것인, UE, NAF 및 BSF를 포함하는 시스템에서의 방법.

청구항 31

NAF(network application function) 및 BSF(bootstrapping server function)와 통신하도록 구성되어 있는 사용자 장비(user equipment; UE)에 있어서,

컴퓨터 실행가능 명령어가 저장되어 있는 메모리와;

프로세서를 포함하고,

상기 프로세서는,

상기 UE와 상기 BSF 사이에 TLS(transport-layer security) 터널을 설정(establish)하는 것 - 상기 TLS 터널은 그와 연관되어 있는 TLS 마스터 키를 가짐 -;

상기 BSF로부터의 신청에 응답하여, 랜덤한 넌스를 발생하고 인증 응답을 계산하는 것;

상기 UE의 사용자의 성공적인 인증을 나타내는 메시지를 수신하는 것 - 상기 성공적인 인증의 결과, 제2 키가 도출됨 -; 및

차후에 상기 UE와 상기 NAF 사이의 통신을 보호할 시에 사용하기 위한 제3 키를 도출하는 것 - 상기 제3 키의 도출은 상기 제2 키 및 상기 TLS 마스터 키 둘 다에 적어도 부분적으로 의존함 -

을 수행하기 위해 상기 컴퓨터 실행가능 명령어를 실행하도록 구성되어 있는 것인, 사용자 장비(UE).

청구항 32

제31항에 있어서, 상기 프로세서는 또한 하나 이상의 SIP 다이제스트(session initiation protocol digest) 자격 증명(credential)을 사용하여 상기 인증 응답을 계산하기 위해 상기 컴퓨터 실행가능 명령어를 실행하도록 구성되어 있는 것인, 사용자 장비(UE).

청구항 33

제31항에 있어서, 상기 제2 키는 GBA(generic bootstrapping architecture) 세션 키(Ks)이고, 상기 프로세서는 또한 GBA 프로토콜을 사용하여 상기 Ks로부터 상기 제3 키를 도출하기 위해 상기 컴퓨터 실행가능 명령어를 실행하도록 구성되어 있고, 상기 제3 키는 응용프로그램 특유의 키(Ks_NAF)인 것인, 사용자 장비(UE).

청구항 34

제33항에 있어서, 상기 프로세서는 또한 상기 Ks_NAF를 사용하여 상기 UE와 상기 NAF 사이의 통신을 보호하기 위해 상기 컴퓨터 실행가능 명령어를 실행하도록 구성되어 있는 것인, 사용자 장비(UE).

청구항 35

제33항에 있어서, 상기 Ks는 무결성 키(integrity key) 또는 기밀성 키(confidentiality key) 중 적어도 하나를 포함하는 것인, 사용자 장비(UE).

발명의 설명

기 술 분 야

[0001] 관련 출원의 상호 참조

[0002] 본 출원은 미국 가특허 출원 제61/466,662호(2011년 3월 23일자로 출원됨), 미국 가특허 출원 제61/525,575호(2011년 8월 19일자로 출원됨), 및 미국 가특허 출원 제61/466,852호(2011년 3월 23일자로 출원됨)(이들의 내용이 참조 문헌으로서 본 명세서에 포함됨)를 기초로 우선권을 주장한다.

배 경 기 술

[0003] 통신 네트워크에서, 네트워크 엔터티들 간의 다양한 형태의 통신이 제3자 공격에 취약할 수 있다. 예를 들어, 일 실시예에 따르면, 사용자 디바이스는 통신 네트워크를 통해 서비스 제공자로부터의 서비스(예컨대, 웹 사이트)에 액세스하려고 시도할 수 있다. 사용자 디바이스로부터의 이 액세스 시도 및/또는 기타 통신이 제3자 또는 MitM(man-in-the-middle)에 의해 가로채기될 수 있다. 이 제3자는, 예를 들어, 인증 정보(예컨대, 사용자 이름 및/또는 비밀 번호) 등의 사용자 디바이스와 연관되어 있는 정보에 액세스하기 위해 의도된 서비스 제공자로서 역할할지도 모른다. 제3자가 사용자 디바이스로부터 인증 정보를 획득하는 데 성공하는 경우, 제3자는 의도하지 않은 또는 악의적 목적을 위해 인증 정보를 사용할지도 모른다. 예를 들어, 제3자는 의도된 서비스 제공자로부터의 서비스 및/또는 기타 정보에 액세스하기 위해 사용자 디바이스로서 역할할지도 모른다.

[0004] 일 실시예에서, 네트워크 통신은 공격에 취약할 수 있는데, 그 이유는 통신이 충분히 보호되지 않을 수 있고 및/또는 통신을 송신받고 있는 네트워크 엔터티가 통신을 수신하기 위한 확실한 또는 의도된 네트워크 엔터티라는 적절한 보장 없이 송신될 수 있기 때문이다. 예를 들어, 네트워크 통신이, 예를 들어, 공개 키의 전송을 통해 단방향 인증 프로토콜(one-sided authentication protocol)을 사용하여 구현될 수 있고, 이는 네트워크 통신을 제3자 또는 MitM 공격에 취약하게 둘 수 있다.

발명의 내용

[0005] 이 요약은 이하에서 상세한 설명에 추가로 기술되어 있는 다양한 개념들을 간략화된 형태로 소개하기 위해 제공된다.

[0006] 서비스 제공자와 사용자 장비(UE) 간의 보안 통신을 설정하는 시스템, 방법 및 장치 실시예가 본 명세서에 기술되어 있다. 예를 들어, UE, 서비스 제공자, 및/또는 ID 제공자(identity provider)를 포함하는 시스템에서 네트워크 통신이 구현될 수 있다. UE와 서비스 제공자 사이에 보안 채널이 설정될 수 있다. ID 제공자에 대한 UE의 인증을 수행하기 위해 인증 파라미터가 ID 제공자로 송신될 수 있다. UE의 성공적인 인증을 나타내는 UE 인증 주장(authentication assertion)이 UE에서 결정될 수 있다. 예를 들어, UE 인증 주장이 외부 네트워크 엔터티로부터 수신되거나 UE에서 로컬적으로 결정될 수 있다. UE는 보안 채널이 설정되어 있는 서비스 제공자가 의도된 서비스 제공자인지를 검증할 수 있다. 의도된 서비스 제공자는 그로부터 서비스를 수신하기로 의도되어 있는 서비스 제공자 및/또는 이러한 서비스에 액세스하기 위해 그에 대해 인증이 수행되어야 하는 서비스 제공자를 포함할 수 있다. ID 제공자에 대한 UE의 인증 동안 및/또는 보안 채널의 설정 동안 발생된 적어도 하나의 파라미터를 사용하여, 서비스 제공자가 의도된 서비스 제공자로서 검증될 수 있다.

[0007] 다른 예시적인 실시예에 따르면, UE는 서비스 제공자와 보안 통신을 설정하도록 구성될 수 있다. UE는 컴퓨터 실행가능 명령어가 저장되어 있는 메모리, 및 컴퓨터 실행가능 명령어를 실행하도록 구성되어 있는 프로세서를 포함할 수 있다. UE는 UE와 서비스 제공자 사이에 보안 채널을 설정하도록 구성될 수 있다. UE는 ID 제공자에 대한 UE의 인증을 수행하기 위해 인증 파라미터를 ID 제공자로 송신할 수 있다. UE의 성공적인 인증을 나타내는 UE 인증 주장이 UE에서 결정될 수 있다. 예를 들어, UE 인증 주장이 외부 네트워크 엔터티로부터 수신되거나 UE에서 로컬적으로 결정될 수 있다. UE는 또한 보안 채널이 설정되어 있는 서비스 제공자가 서비스에 대한 인증을 수행하는 의도된 서비스 제공자인지를 검증하도록 구성되어 있을 수 있다. 의도된 서비스 제공자는 그로부터 서비스를 수신하기로 의도되어 있는 서비스 제공자 및/또는 이러한 서비스에 액세스하기 위해 그에 대해 인증이 수행되어야 하는 서비스 제공자를 포함할 수 있다. UE는 ID 제공자에 대한 UE의 인증 동안 및/또는 보안 채널의 설정 동안 발생된 적어도 하나의 파라미터를 사용하여, 서비스 제공자가 의도된 서비스 제공자인지를

검증할 수 있다.

[0008] 다른 예시적인 실시예에 따르면, ID 제공자와 서비스 제공자 사이에 보안 채널이 설정될 수 있다. 예를 들어, ID 제공자와 서비스 제공자 사이의 보안 채널을 통해 키 정보(key information)가 서비스 제공자에 수신될 수 있다. 예를 들어, 수신된 키 정보를 사용하는 등에 의해, 서비스 제공자와 UE 사이에도 보안 채널이 또한 설정될 수 있다. 서비스 제공자에서, UE의 인증을 나타내는 인증 주장이 수신될 수 있다. ID 제공자와 서비스 제공자 사이의 보안 채널 및/또는 서비스 제공자와 UE 사이의 보안 채널을 통해 수신된 정보를 사용하여 서비스 제공자에서 인증 주장이 검증될 수 있다.

[0009] 이 요약은 이하에서 상세한 설명에 추가로 기술되어 있는 일련의 개념들을 간략화된 형태로 소개하기 위해 제공된 것이다. 이 요약은 청구된 발명 요지의 주요 특징 또는 필수적인 특징을 확인하기 위한 것도 아니고, 청구된 발명 요지의 범위를 제한하기 위해 사용되기 위한 것도 아니다. 게다가, 청구된 발명 요지는 본 개시 내용의 임의의 부분에 열거된 임의의 또는 모든 단점을 해결하는 것으로 제한되지 않는다.

도면의 간단한 설명

[0010] 일례로서 첨부 도면과 관련하여 주어진 이하의 설명으로부터 보다 상세하게 이해할 수 있다.

도 1은 ID 제공자(identity provider)와 사용자 장비(UE) 사이에 보안 채널을 설정하는 프로비저닝 단계에 대한 예시적인 메시지 흐름도.

도 2는 로컬 ID 제공자를 사용하는 인증 단계에 대한 예시적인 메시지 흐름도.

도 3은 서비스 제공자 인증을 위한 메시지 교환에 대한 예시적인 메시지 흐름도.

도 4는 서비스 제공자 인증을 위한 메시지 교환에 대한 다른 예시적인 메시지 흐름도.

도 5는 UE와 서비스 제공자 사이의 사전 설정된 보안 채널을 사용하는 로컬 ID 제공자 인증을 위한 보안 채널의 설정을 나타내는 예시적인 메시지 흐름도.

도 6은 GBA/GBA_H(Generic Bootstrap Architecture) 프로토콜의 한 예에 대한 예시적인 메시지 흐름도.

도 7은 TLS(Transport-Layer Security) 및 GBA를 SIP 다이제스트(Session Initiation Protocol Digest) 인증과 바인딩하는 예시적인 메시지 흐름도.

도 8은 로컬 인증 엔터티/ID 제공자 및 클라우드/원격 컴퓨팅 서비스를 구현하는 예시된 통신 시스템을 나타낸 도면.

도 9는 SIP 다이제스트 인증을 사용하고 서비스 제공자 인증을 포함하는 예시적인 메시지 흐름도.

도 10은 ID 제공자에 대한 서비스 제공자 인증에서의 예시적인 프로토콜의 예시적인 메시지 흐름도.

도 11은 로컬 ID 제공자에서의 프로비저닝 단계의 예시적인 메시지 흐름도.

도 12는 로컬 주장 제공자(local assertion provider)에서의 예시적인 인증 단계의 예시적인 메시지 흐름도.

도 13a는 하나 이상의 개시된 실시예가 구현될 수 있는 예시적인 통신 시스템의 시스템도.

도 13b는 도 13a에 예시된 통신 시스템 내에서 사용될 수 있는 예시적인 WTRU(wireless transmit/receive unit, 무선 송수신 유닛)의 시스템도.

도 13c는 도 13a에 예시된 통신 시스템 내에서 사용될 수 있는 예시적인 RAN(radio access network, 무선 액세스 네트워크) 및 예시적인 코어 네트워크의 시스템도.

도 13d는 일 실시예에 따른, 예시적인 RAN 및 코어 네트워크의 다른 시스템도.

도 13e는 일 실시예에 따른, 예시적인 RAN 및 코어 네트워크의 다른 시스템도.

발명을 실시하기 위한 구체적인 내용

[0011] 본 명세서에 개시되어 있는 시스템, 방법 및 장치 실시예는, 예를 들어, 사용자/사용자 장비(UE), 서비스 제공자, 및/또는 ID 제공자 등의 네트워크 엔터티들 간의 보안 통신을 제공한다. 본 명세서에 기술된 바와 같이, 엔터티들 간의 공유 키/비밀을 사용하여 및/또는 공개/비밀 키를 사용하여 네트워크 엔터티들 간에 설정된 보안

채널을 통해 보안 통신이 수행될 수 있다. 이들 보안 채널은, 예를 들어, MitM(man-in-the-middle) 공격 등의 제3자로부터의 공격을 방지하는 데 사용될 수 있다.

[0012] 본 명세서에 기술되어 있는 일 실시예에서, 통신을 송신 및/또는 수신하는 의도된, 인증된 엔터티를 식별해주는 공유 키 또는 공유 비밀을 사용하여 보안 통신이 수행될 수 있다. 예를 들어, 네트워크 엔터티의 진정성(authenticity)을 나타내는 네트워크 엔터티들 간에 송신되는 메시지를 암호화하고 및/또는 그에 서명하기 위해 공유 키 또는 공유 비밀이 사용될 수 있다.

[0013] 예시적인 실시예에서, 본 명세서에 기술되어 있는 보안 통신은 OpenID 인증 프로토콜에 기초하고 및/또는 그에 바인딩(binding)되어 있을 수 있다. OpenID 인증에서, 서비스 제공자는 RP(relying party)일 수 있고 및/또는 ID 제공자는 OP(OpenID identity provider)일 수 있다. OpenID 인증은 OpenID 및/또는 OpenID에서의 OP의 일부 기능이 로컬 엔터티[UE, 게이트웨이, 스마트 카드, UICC(universal integrated circuit card), 기타 등등]에 의해 수행되는 로컬 OpenID라고 하는 변형의 사용을 포함할 수 있다.

[0014] OpenID 인증 흐름에서의 RP의 인증이 본 명세서에 기술되어 있다. 예를 들어, 사용자/UE 및 RP가, 예를 들어, AAA 데이터베이스로부터 RP에 의해 액세스가능한 UE에 대한 웹 사이트 인증서 및/또는 한 세트의 자격 증명을 사용하여 설정될 수 있는 것과 같은 신뢰 관계를 갖지 않을 수 있는 경우에, 이것이 유용할 수 있다. 다른 실시예는, 본 명세서에 기술되어 있는 바와 같이, 로컬 OP-RP 사적 공유 비밀(private shared secret)의 설정을 포함할 수 있다.

[0015] 로컬 모바일 SSO(single sign-on)는 SSO의 일부 또는 전부, 및/또는 통신 디바이스 자체의 일부 또는 전부일 수 있는 로컬-기반 엔터티 또는 모듈(예컨대, UE, 스마트 카드, 또는 UICC에 존재하는 보안 환경)[또는 이러한 엔터티/모듈은 물리적으로 및/또는 논리적으로 통신 디바이스 및/또는 그의 사용자에 아주 근접하여 위치해 있음(예컨대, 게이트웨이 등을 통해 연결되는 것과 같이 로컬적으로 위치해 있음)]에 의해 수행되는 관련 ID(identity) 관리 기능 - 종래에는, 예를 들어, 웹-기반 SSO 서버에 의해 수행될 수 있음 - 을 총칭하여 나타내기 위해 사용되는 용어이다. 예를 들어, 엔터티/모듈은 디바이스에 내장되어 있고, 디바이스에 부착되어 있고, 및/또는 로컬 인터페이스, 배선, 또는 단거리 무선 수단에 의해 디바이스에 연결되어 있을 수 있다.

[0016] 한 유형의 로컬 모바일 SSO를 나타내는 용어로서 로컬 OpenID가 사용될 수 있고, 그에 의해 SSO 또는 ID 관리는 OpenID 프로토콜에 기초하고 있다. 예를 들어, 로컬적으로 위치되는 엔터티/모듈에 의해 수행될 수 있는 OpenID ID 제공자(OP 또는 IdP)의 기능을 나타내기 위해 로컬 OpenID가 사용될 수 있다.

[0017] 로컬 IdP는 로컬 인증 및/또는 주장 기능을 수행하는 로컬 엔터티 또는 모듈을 나타내기 위해 사용되는 용어이다. 예를 들어, 로컬 IdP는 로컬 OpenID에 대한 OpenID 서버의 인증 및/또는 주장 기능을 수행할 수 있다. OpenID 기능을 구현하는 로컬 IdP를 나타내기 위해 약어 OP_{loc}가 사용될 수 있지만, 로컬 IdP가 유사한 기능을 수행할 수 있고 OpenID 프로토콜을 구현하는 데 필요하지 않을 수 있다. 로컬 IdP의 한가지 기능은 사용자 및/또는 디바이스의 ID에 관한 주장(들)을 통해 사용자 및/또는 디바이스의 인증을 용이하게 해주는 것일 수 있다. 예시적인 실시예에서, 이러한 인증 주장은 로컬 IdP로부터 디바이스 상에서 실행 중인 BA(browser agent)로 송신될 수 있고, BA는 인증 주장을 외부 RP로 전달할 수 있다. 로컬 IdP에 의해 제공되는 기능(들)이 주로 이러한 인증 주장을 제공하는 것으로 제한되어 있을 때, 로컬 IdP는 LAE(Local Assertion Entity)라고 할 수 있다.

[0018] 로컬 IdP는 인증 주장 메시지를 처리, 생성, 관리 및/또는 하나 이상의 외부 수신자로 송신할 수 있다. 인증 주장 메시지는 사용자 및/또는 디바이스에 관련된 하나 이상의 ID의 검증의 상태를 주장할 수 있다. 예를 들어, OpenID 프로토콜에서, RP 등의 제3자 엔터티는 인증 주장 메시지의 수신자들 중 하나일 수 있다. 로컬 IdP는 또한, 예를 들어, 각종의 공유 키 또는 공개/비밀 키 등의 암호 기법을 사용하여 인증 주장 메시지에 서명할 수 있다.

[0019] 로컬 OpenID 구현예는 루트 세션 키(root session key) 등의 하나 이상의 암호 키를 사용할 수 있다. 루트 세션 키는 RP와 UE 상에 존재하는 OP_{loc} 사이에서 사용하기 위한 것일 수 있다. 이러한 키는 RP와 OP 사이의 루트 세션 키 - 이로부터 다른 키들이 도출될 수 있음 - 로서 역할할 수 있다. 로컬 OpenID 방법은 또한 인증 주장 키를 사용할 수 있고, 이 인증 주장 키는 사용자의 인증을 위해 인증 주장 메시지(들) 중 하나 이상에 서명하는 데 사용될 수 있다. 이러한 인증 주장 키는 루트 세션 키로부터 도출될 수 있다.

[0020] 로컬 OpenID 구현예는 OPSF(OpenID Server Function)라고 하는 서비스를 사용할 수 있고, 이 OPSF의 역할은 로컬 IdP 및/또는 RP에 의해 사용될 수 있는 비밀을 발생, 공유 및/또는 분배하는 것일 수 있다. 예시적인 실시

예에서, OPSF 및 로컬 IdP는 외부 RP에 의해 단일 엔터티로서 간주될 수 있다. OPSF는 로컬 OpenID에 의해 발행된 서명을 검증할 수 있고, 및/또는, 예를 들어, 공중 인터넷을 통해 RP에 의해 직접 도달가능할 수 있다. OPSF의 주소가 로컬 IdP에 매핑되도록 디바이스 상의 로컬 DNS 분석 모듈(local DNS resolving module)을 수정함으로써 디바이스 상의 브라우저가 로컬 IdP로 리디렉션될 수 있다.

[0021] OpenID 구현예는 RP를 대신하여 로컬 IdP의 발견을 용이하게 해주는 서비스를 사용할 수 있다. 이러한 서비스는, 예를 들어, OP-agg라고 표시될 수 있다.

[0022] OpenID(예를 들어, OpenID 및/또는 로컬 OpenID를 포함함)를 사용하여 구현될 수 있는 보안 시스템, 방법 및 장치가 본 명세서에 개시되어 있다. 본 명세서에 기술되어 있는 실시예들 중 일부 실시예는, 예를 들어, UE에서 구현될 수 있다. 사용자 장비는 OpenID 요청을 OP로 전달할 수 있다. 본 명세서에 추가로 기술되는 바와 같이, OP는 UE 및/또는 RP를 인증하는 데 사용될 수 있다.

[0023] RP에서 로컬 OP로의 투명한 인증 위임을 위한 실시예들이 기술되어 있다. 본 명세서에 기술되어 있는 실시예에 따르면, OpenID를 사용하여 어떻게 RP 인증을 수행하는지 및/또는, 예를 들어, OP_{loc} 등의 서명된 인증 주장의 로컬 제공자를 사용하는 것을 나타내는 프로토콜이 개시되어 있다. 본 명세서에 기술된 바와 같이, 재생 보호(replay protection)를 위해 신청 값(challenge value) 및/또는 넌스(nonce)가 부가될 수 있다[예컨대, 도 1에서의 프로토콜의 단계(112 및 120)].

[0024] RP를 인증하는 기술된 구현예의 한 측면은 OPSF 노드에 의한 인증 위임의 구현예를 포함할 수 있다. 이는 OP_{loc}가 신청 RP_{Chv}를 제기하는 일반적인 신청-응답(challenge-response) 전략을 따를 수 있다. 이 신청은 진정한 RP가 그 신청을 복호화할 수 있도록 적절한 방식으로 OPSF에 의해 암호화될 수 있다. 예를 들어, RP 및 OPSF는 신청을 암호화 및 복호화하는 데 사용될 수 있는 비밀 $K_{r,o}$를 공유할 수 있다.

[0025] 도 1은 예시적인 프로비저닝 단계(provisioning phase, PP)의 메시지 흐름도를 나타낸 것이다. 도 1에 예시된 바와 같이, 프로비저닝 단계는 UE/OP_{loc}(102), RP(104), OPSF(106), 및/또는 HSS(Home Subscription Service)(108)를 포함할 수 있다. 110에서, UE/OP_{loc}(102)는 로그인 식별자(login identifier)[예컨대, http 주소 또는 이메일 등의 OpenID 식별자(OpenID identifier, OID)]를 RP(104)로 전송할 수 있다. 110에서의 메시지는 RP 신청 값 RP_{Chv}를 포함할 수 있다. RP 신청 값 RP_{Chv}는 RP(104)가 그의 진정성을 증명하기 위해 그에 대해 적절히 응답할 수 있는 값이다. 예를 들어, 이것은 일회용일 수 있는 랜덤한 값일 수 있다. 112에서, RP(104)는 연관 요청(association request)(예컨대, http POST OpenID 연관 요청)을 OPSF(106)로 송신할 수 있다. 연관 요청은 RP(104)에 대응하는 RP 자격 증명(RP credential) RP_{Cred} 및/또는 RP 신청 값 RP_{Chv}를 포함할 수 있다. RP_{Cred}는 OPSF(106)가 OPSF(106)와 RP(104) 사이에서 공유되는 올바른 사전 공유 키(pre-shared key) $K_{r,o}$를 선택할 수 있게 해줄 수 있는 RP(104)의 식별자일 수 있다. RP_{Cred}는 OPSF(106)가 다른 수단(예컨대, 인터넷 URL)에 의해 RP(104)를 식별하는 경우 메시징으로부터 생략될 수 있다. 114에서, OPSF(106)는 OPSF(106)와 UE/OP_{loc}(102) 사이의 공유 비밀 K_0가 프로비저닝되어 있는지를 결정할 수 있다. 그러한 경우, OPSF(106)는 (예컨대, 도 2에 예시된 바와 같이) 인증 단계(authentication phase, AP)로 진행할 수 있다. 그렇지 않은 경우, 프로비저닝 단계가 계속될 수 있다.

[0026] 116에서, OPSF(106)는, 예를 들어, RP_{Cred} 또는 RP(104)의 다른 신뢰된 식별자에 기초하여 공유 비밀 $K_{r,o}$를 선택할 수 있다. 118에서, OPSF(106)는 RP(104)와의 연관을 수행할 수 있다. 118에서, OPSF(106)는 연관 핸들(association handle) A 및/또는 서명 키(signing key) S를 발생할 수 있다. 연관 핸들 A의 함수에 기초하여 서명 키 S가 발생될 수 있다. OPSF(106)는 연관 핸들 A 및 서명 키 S를 RP(104)로 송신할 수 있다. 서명 키 S는 공유 키 $K_{r,o}$로 암호화될 수 있고, 이 암호화된 것은, 예를 들어, $EK_{r,o}(S)$라고 할 수 있다. 120에서, RP(104)는 리디렉션 메시지(redirect message)를 UE/OP_{loc}(102)로 송신할 수 있다. 리디렉션 메시지는, 예를 들어, sessionID, returnURL, 넌스, 로그인 식별자(예컨대, OID), 및/또는 연관 핸들 A 등의 파라미터를 포함할 수 있다. 122에서, UE/OP_{loc}(102)는 요청(예컨대, http GET 요청)을 OPSF(106)로 송신할 수 있다. 요청(예컨대, http GET 요청)은, 예를 들어, sessionID, returnURL, 넌스, 로그인 식별자(예컨대, OID), 및/또는 연관 핸들 A 등의 파라미터를 포함할 수 있다.

[0027] 124에서, OPSF(106)는 HSS(108)로부터 인증 벡터(authentication vector) 및/또는 기타 정보를 가져올 수

있다. 126에서, OPSF(106)는 인증 신청(authentication challenge)을 UE/OP_{loc}(102)로 송신할 수 있다. 128에서, UE/OP_{loc}(102)는 인증 응답(authentication response)을 계산하고 인증 응답을 OPSF(106)로 송신할 수 있다. 130에서, OPSF(106)는 인증 응답을 유효성 확인하고 OPSF(106)와 UE/OP_{loc}(102) 사이에서 공유되는 공유 비밀 K_0를 발생할 수 있다. 인증 응답의 유효성 확인 후에 공유 비밀 K_0의 이러한 발생은 UE/OP_{loc}(102)와 OPSF(106) 사이의 보안 연관(security association)의 설정을 이 인증에 바인딩시킬 수 있다. 예를 들어, 도 1에 도시된 바와 같이, 이 바인딩은 공유 비밀 K_0의 발생에 대한 인증 응답의 유효성 확인의 절차적 바인딩(procedural binding)일 수 있다. 132에서, UE/OP_{loc}(102)는 공유 비밀 K_0를 발생할 수 있다. 134에서, OPSF(106)는 UE/OP_{loc}(102)를 인증한 후에 인증 주장 메시지 UE_{Assert}를 발생할 수 있다. 인증 주장은 K_0에 의해 암호화되어 있는 RP_{Cred} 및 RP_{Chv}[이 암호화된 것은, 예를 들어, $K_0(RP_{Cred}, RP_{Chv})$라고 할 수 있음]를 포함할 수 있다. $K_0(RP_{Cred}, RP_{Chv})$를 포함하는 이 인증 주장은 OPSF(106)가 RP(104)를 인증했다는 것을 UE/OP_{loc}(102)에 알려줄 수 있고, 따라서 UE/OP_{loc}(102)는 자신이 적법한 RP(104)와 대화하고 있다는 것으로 확신할 수 있다. 하나의 예시적인 실시예에서, RP_{Cred}는 UE/OP_{loc}(102)에 의해 식별가능한 RP(10)에 대한 이름(또는 기타 텍스트 값)일 수 있다. OPSF(106)는 또한 인증 주장 메시지 UE_{Assert}를 서명 키 S로 암호화할 수 있고, 이 암호화된 것은, 예를 들어, $E_S(UE_{Assert})$라고 할 수 있다. 136에서, OPSF(106)는 리디렉션 메시지를 UE/OP_{loc}(102)로 송신할 수 있다. 리디렉션 메시지는 서명된 주장 메시지와 함께 UE/OP_{loc}(102)를 RP(104)로 리디렉션할 수 있다. 138에서, UE/OP_{loc}(102)는 서명된 주장 메시지와 함께 요청(예컨대, http GET 요청)을 RP(104)로 송신할 수 있다. 140에서, RP(104)는 공유 키 $K_{r,o}$를 사용하여 서명 키 S를 복호화하고 및/또는 $E_S(UE_{Assert})$를 복호화함으로써 서명 키 S를 사용하여 인증 주장 메시지(예컨대, OpenID 주장 메시지)를 검증할 수 있다. 142에서, RP(104)는 인증 주장 UE_{Assert}를 포함하는 통지를 UE/OP_{loc}(102)로 송신할 수 있다. 144에서, UE/OP_{loc}(102)는 RP_{Chv} 및/또는 RP_{Cred}를 복호화함으로써 인증 주장 UE_{Assert}를 유효성 확인할 수 있다.

[0028] 도 1에 예시된 바와 같이, OPSF(106)와 UE/OP_{loc}(102) 사이의 공유 비밀 K_0가 설정될 수 있는 프로토콜이 구현될 수 있다. 예시적인 실시예에서, 프로비저닝 단계 이전에, 또는 그 동안에, OPSF(106) 및 UE/OP_{loc}(102)는 아직 비밀을 공유하지 않을 수 있다. 예를 들어, 네트워크 엔터티 HSS(108)를 사용하여 네트워크-기반 인증을 포함시키는 것에 의해 그 프로토콜이 실행될 때, 이 공유 비밀이 설정될 수 있다. RP_{Chv} 및 RP_{Cred}를 K_0로 암호화되어 있는 UE_{Assert}에 포함시키는 것에 의해, UE/OP_{loc}(102)는 수신된 메시지가 RP_{Cred}에 의해 식별되는 RP(104)로부터 온 것으로 확신할 수 있다. RP_{Cred}에서 주장된 ID와 RP(104)의 ID를 비교함으로써, UE/OP_{loc}(102)는 어떤 다른 RP도 인증 정보를 수신하지 않았다는 것과 RP(104)가 UE/OP_{loc}(102)가 그에 대해 인증을 수행하고자 했던 의도된 RP라는 것을 검증할 수 있다. UE_{Assert}에서의 정보 단편(information piece) RP_{Cred}는 RP(104) ID를 UE(102)에 알려주기 위해 OPSF(106)에 의해 발생되는 어떤 명확한 의사 표시(explicit statement) RP_{Assert}로 대체될 수 있다. UE_{Assert}는 서명된 OpenID 주장 메시지(서명 키 S로 서명되어 있음)일 수 있다.

[0029] 도 1은 또한 RP(104)가 UE/OP_{loc}(102)에 대해 인증(예컨대, 암시적으로 인증)될 수 있다는 것을 나타내고 있다. RP(104)는, RP_{Cred}에 의해 식별되는 진정한 RP인 경우, UE/OP_{loc}(102)의 OpenID 인증을 수행할 수 있다[그 후에 RP(104)는 서명 키 S를 복호화할 수 있음]. 그 프로토콜에서 OPSF(106)에 의해 RP(104)에 대해 인증되는 고유의 UE/OP_{loc}(102)는 RP(104)를 인증할 수 있다. 예시적인 실시예에서, 프로토콜 흐름이 로컬 OpenID 인증으로부터 수정되지 않을 수 있다. 또한, 네트워크 인증이 영향을 받지 않은 채로 있을 수 있다. 추가의 보호를 보장해 주기 위해 프로토콜에서의 하나 이상의 당사자에서 부가의 암호 동작이 구현될 수 있다.

[0030] 로컬 OpenID와 GBA(Generic Bootstrapping Architecture)(예컨대, 3GPP GBA) 간의 연동에 대한 가능한 구현의 경우, UE/OP_{loc}(102)와 OPSF(106) 간의 사전 공유 비밀 K_0가 존재하는 경우 프로토콜이 구현될 수 있다.

[0031] 도 2는 인증 단계(Authentication Phase, AP)의 예시적인 메시지 흐름도를 나타낸 것이다. 예를 들어, 인증 단계는 UE/OP_{loc}(202), RP(204), OPSF(206), 및/또는 HSS(208)를 구현할 수 있다. 도 2에 예시된 프로토콜 흐름

은, UE/OP_{loc}(102)와 OPSF(106) 간의 공유 비밀을 사용하여 보안 채널을 설정하기 위해, 예를 들어, 그 공유 비밀이 사전 공유 키로서 아직 존재하지 않는 경우 등에, 독립적으로 또는 도 1에 기술된 프로토콜 프로비저닝 단계(PP)와 관련하여 적용될 수 있다.

[0032] 도 2에 예시된 바와 같이, 210에서, UE/OP_{loc}(202)는 로그인 식별자(login identifier)[예컨대, http 주소 또는 이메일 등의 OpenID 식별자(OpenID identifier, OID)]를 RP(204)로 전송할 수 있다. 212에서, RP(204)는 연관 요청(예컨대, http POST OpenID 연관 요청)을 OPSF(206)로 송신할 수 있다. 연관 요청은 RP(204)를 식별해주는 RP 자격 증명 RP_{Cred}를 포함할 수 있다. 214에서, OPSF(206)는 공유 키 K_0가 결정되거나 프로비저닝되었는지를 결정할 수 있고, 그렇지 않은 경우, 프로토콜은 프로비저닝 단계에서 K_0를 프로비저닝하는 것을 계속할 수 있다. K_0가 이미 프로비저닝되어 있는 경우, 프로토콜은 인증 단계를 계속할 수 있다. 예를 들어, 216에서, OPSF(206)는 RP(204)에 대응하는 RP_{Cred}에 기초하여 공유 키 $K_{r,o}$를 선택할 수 있다. 218에서, OPSF(206)는 RP(204)와의 연관을 수행할 수 있다. OPSF(206)는 연관 핸들 A 및/또는 공유 키 K_1을 발생할 수 있다. 공유 키 K_1은, 예를 들어, 연관 핸들 A, RP_{Cred}, 및/또는 공유 키 K_0의 함수로부터 발생되는 OPSF(206), UE/OP_{loc}(202), 및/또는 RP(204) 사이의 공유 키일 수 있다. 예를 들어, UE/OP_{loc}(202) 및/또는 OPSF(206)는 공유 키 K_1을 발생하도록 구성될 수 있다. RP(204)는 공유 키 K_1을 수신하고, UE/OP_{loc}(202)와의 보안 통신을 위해, 이를 사용할 수 있다. OPSF(206)는 연관 핸들 A 및 암호화된 K_1을 RP(204)로 송신할 수 있고, 여기서 K_1은 공유 키 $K_{r,o}$에 의해 암호화되며, 이 암호화된 것은, 예를 들어, $EK_{r,o}$ (K_1)이라고 할 수 있다. 220에서, RP(204)는 sessionID, returnURL, 넌스, 로그인 식별자(예컨대, OID), 연관 핸들 A, 및/또는 RP_{Cred} 등의 파라미터를 포함하는 메시지를 UE/OP_{loc}(202)로 송신할 수 있다. 220에서의 메시지는 UE/OP_{loc}(202)를, 예를 들어, RP(204)로 리디렉션하는 리디렉션 메시지일 수 있다. 222에서, UE/OP_{loc}(202)는 K_1을 발생할 수 있다. 예를 들어, K_1은 연관 핸들 A, RP_{Cred}, 및/또는 K_0의 함수로부터 발생될 수 있다. UE/OP_{loc}(202)는, 222에서, 로컬 인증을 수행할 수 있고, 222에서, RP_{Chv}를 포함하는 인증 주장 메시지 UE_{Assert}를 발생할 수 있고 및/또는 키 K_1으로 UE_{Assert}를 암호화할 수 있으며, 이 암호화된 것은, 예를 들어, $EK_1(UE_{Assert})$라고 할 수 있다. UE_{Assert}는, 예를 들어, OpenID 주장 메시지일 수 있다. UE/OP_{loc}(202)는 암호화된 주장 메시지 UE_{Assert}를 RP(204)로 송신할 수 있다. 224에서, UE/OP_{loc}(202)는 서명된 주장과 함께 요청(예컨대, http GET 요청)을 RP(204)로 송신할 수 있다. 226에서, RP(204)는 $K_{r,o}$를 사용하여 K_1을 복호화할 수 있다. RP(204)는, 226에서, 복호화된 K_1을 사용하여 인증 주장 메시지 UE_{Assert}를 복호화할 수 있다. RP(204)는 공유 키 K_1을 사용하여 OpenID 주장을 검증할 수 있다. 228에서, RP(204)는 인증 주장 메시지 UE_{Assert}를 포함하는 통지를 UE/OP_{loc}(202)로 송신할 수 있다. 230에서, UE/OP_{loc}(202)는 인증 주장 메시지 UE_{Assert}를 유효성 확인할 수 있다.

[0033] 228에서 수신된 UE_{Assert}에서의 정보가 224에서 송신된 UE_{Assert}에서의 정보와 일치하는 것으로 유효성 확인함으로써, UE/OP_{loc}(202)는 228에서의 수신된 메시지가 RP_{Cred}에 의해 식별되고 UE/OP_{loc}(202)가 210에서 로그인 정보를 전송한 RP(204)로부터 발신된 것임을 확신할 수 있다. 예를 들어, RP_{Cred}에서 주장된 ID와 RP(104)의 ID를 비교함으로써, UE/OP_{loc}(202)는 어떤 다른 RP도 인증 정보를 수신하지 않았다는 것과 RP(104)가 UE/OP_{loc}(202)가 그에 대해 인증을 수행하고자 했던 의도된 RP라는 것을 검증할 수 있다.

[0034] UE_{Assert}에 새로운 신청 RP_{Chv}를 포함시킴으로써 인증의 신선성(freshness)이 보장될 수 있다. UE/OP_{loc}(202)는 수신된 UE_{Assert}가 이 신청 값을 포함하는지를 검증함으로써 수신된 UE_{Assert}를 유효성 확인할 수 있고, RP(204)는 UE/OP_{loc}(202) 및 RP(204)에 의해 공유될 수 있는 진정한 K_1으로 UE_{Assert}를 복호화할 수 있는지를 알 수 있다. 진정한 K_1의 사용은 RP(204)가 OPSF(206) 및 RP_{Cred}에 의해 식별되는 RP에 의해 공유되는 $K_{r,o}$를 소유하고 있다는 것을 증명할 수 있다.

[0035] 예시적인 실시예에 따르면, 로컬 OpenID 없이 OP를 사용하여(예컨대, 비로컬 OpenID를 사용하여) RP 인증이 수행될 수 있다. OpenID 프로토콜에 RP 인증을 포함시키는 것은 OpenID 프로토콜 자체에 대한 변경 및/또는 OP

및/또는 RP의 구현예에 대한 변경을 포함할 수 있다. RP 인증은, 예를 들어, 가짜 또는 불법 RP(rogue RP)에 의한 가능한 공격에 대한 대책을 제공하는 등의 보안 이점을 부가할 수 있다. OpenID(또는 로컬 OpenID)에 대한 UE 상에서의 구현은 임의의 이러한 RP 인증에 의해 영향을 받지 않을 수 있다. 예를 들어, UE는 로컬 OP 기능을 포함하지 않을 수 있고, 일 실시예에서, 신청 RP_{Chv}를 RP로 송신할 수 없을지도 모른다. RP 인증은 OP와 RP 사이의 신청-응답 단계를 포함할 수 있고, 여기서 OP는 신청을 신선성의 증거와 함께 (예컨대, 암호화된 넌스를 통해) RP로 송신할 수 있다. RP는 이 넌스를 복호화하고 대답을 OP로 반송하기 위해 사전 설정된 공유 비밀 $K_{r,o}$를 사용할 수 있다. 다른 대안으로서 또는 그에 부가하여, 넌스는 암호화되지 않을 수 있고, 그의 대답에서 RP에 의해 서명될 수 있다. 인증 신청에 대한 응답은 OP 인증 신청에 대한 직접적인 응답일 수 있거나, 예를 들어, UE를 OP로 보낼 수 있는 리디렉션 메시지에 통합되어 있을 수 있다. 어느 경우든지, OP는 UE 인증에 관여하기 전에 RP의 인증에 관한 신뢰할 만한 증거를 가질 수 있다. 이것은 RP 인증 실패의 경우에 프로토콜의 중단을 가능하게 해줄 수 있고, 및/또는 이러한 RP 인증 실패의 경우에 UE와 OP 간의 통신 노력을 절감할 수 있다. OP는 이어서 RP 인증 실패에 관한 정보를 UE로 직접 전달할 수 있다.

[0036] 도 3은 RP(304) 인증을 위한 메시지 교환의 예시적인 부분의 메시지 흐름도를 나타낸 것이다. 메시지 흐름도는 UE(302), RP(304) 및 OP(306) 간의 통신을 포함하고 있다. 인증 실패의 경우에, OP(306)는 강제로 UE(302)와 HTTPS(Hypertext Transfer Protocol Secure) 통신을 할 수 있고 및/또는 UE(302)에 실패를 통지할 수 있다. 그렇지 않은 경우, OpenID 인증이 계속될 수 있다.

[0037] 도 3에 예시된 바와 같이, 308에서, UE(302)는 로그인 식별자(예컨대, OID)를 RP(304)로 전송할 수 있다. 310에서, RP(304)는 연관 요청(예컨대, http POST OpenID 연관 요청)을 OP(306)로 송신할 수 있다. 310에서의 연관 요청은 RP_{Cred}를 포함할 수 있다. 312에서, OP(306)는, 예를 들어, RP_{Cred} 또는 RP(304)의 다른 신뢰된 식별자에 기초하여 OP(306)와 RP(304) 사이의 공유 비밀 $K_{r,o}$를 선택할 수 있다. 314에서, OP(306)는 RP(304)와의 연관을 수행할 수 있다. 314에서, OP(306)는 연관 핸들 A, 서명 키 S, 및/또는 RP_{Chv}를 발생할 수 있다. RP_{Chv}는 $K_{r,o}$를 사용하여 암호화될 수 있고, 이 암호화된 것은, 예를 들어, $EK_{r,o}(RP_{Chv})$라고 할 수 있다. OP(306)는 연관 핸들 A, 서명 키 S, 및/또는 $EK_{r,o}(RP_{Chv})$를 RP(304)로 송신할 수 있다.

[0038] 316에서, RP(304)는 공유 키 $K_{r,o}$를 사용하여 RP_{Chv}를 복호화할 수 있다. 318에서, RP(304)는 sessionID, returnURL, 넌스, 로그인 식별자(예컨대, OID), 연관 핸들 A, 및/또는 RP_{Chv} 등의 파라미터를 포함할 수 있는 메시지를 UE(302)를 통해 OP(306)로 송신할 수 있다. 예를 들어, 318에서의 메시지는 UE(302)를 OP(306)로 리디렉션시킬 수 있는 리디렉션 메시지를 포함할 수 있다. 320에서, UE(302)는 메시지(예컨대, http GET 요청)을 OP(306)로 송신할 수 있다. 320에서의 메시지는 sessionID, returnURL, 넌스, 로그인 식별자(예컨대, OID), 연관 핸들 A, 및/또는 RP_{Chv} 등의 파라미터를 포함할 수 있다. 322에서, OP(306)는 RP_{Chv}로 RP(304)의 ID를 유효성 확인할 수 있다. 324에서, RP(304)의 ID가 유효하지 않은 것으로 결정되는 경우, 326에서, OP(306)는 RP(304)의 유효하지 않음을 나타내는 통지를 [예컨대, RP(304)가 유효하지 않음을 나타내는 HTTPS 통지를 통해] UE(302)로 송신할 수 있다. RP(304)의 ID가 유효한 경우, 328에서 인증(예컨대, OpenID 인증)이 계속될 수 있고 및/또는 OP(306)는 RP(304)의 ID가 유효하다는 것을 나타내는 통지를 송신할 수 있다(도시 생략).

[0039] 다른 실시예에서, RP(304)가 OP(306)와의 보안 연관을 설정하는 경우, 보안 연관을 설정하기 위한 OP(306)로부터의 신청을 프로토콜에 포함시키기 위해 대응하는 단계들이 수정될 수 있다. 연관 설정 동안, OP(306) 및 RP(304)는 인증 주장 메시지 UE_{Assert}에 서명하는 데 사용될 수 있는 MAC(message authentication code, 메시지 인증 코드) 키를 설정할 수 있다. 이 키는 [예컨대, DH(Diffie-Hellman) 절차를 사용하여] OP(306)와 RP(304) 사이에서 협상될 수 있는 임시 비밀 키(temporary secret key)를 사용하여 암호화된 채로 송신될 수 있다. 임시 비밀 키에 부가하여, OP(306)는 RP(304)에 대한 응답에 넌스를 포함시킬 수 있다. 이 넌스는, 예를 들어, 임시 비밀 키(예컨대, DH 키)로 암호화될 수 있다.

[0040] RP(304)는 협상된 키(예컨대, DH 키)에 기초하여 넌스 및/또는 MAC 키를 복호화할 수 있다. RP(304)는 그 자신의 사전 설정된 $K_{r,o}$ 키를 사용하여 OP(306)로부터 수신되는 넌스를 암호화하거나 그에 서명할 수 있다. RP(304)는 이 키를, 예를 들어, UE(302)로 송신될 수 있는 리디렉션 메시지에 대한 파라미터로서 부가할 수 있다. UE(302)가 OP(306)로의 리디렉션을 따를 수 있기 때문에, OP(306)는 서명되거나 암호화된 넌스를 수신할 수 있고, 공유 키 $K_{r,o}$를 사용하여 RP(304)를 인증할 수 있다. 인증 실패의 경우에, OP(306)는 UE(302)를 인증

되지 않은 RP로부터 보호하기 위해 경보 메시지를 UE(302)로 송신할 수 있다. RP 인증 성공의 경우에, OP(306)는 프로토콜을 계속할 수 있다.

[0041] 예시적인 실시예에서, OP(306)와 RP(304) 사이에 연관이 설정되지 않은 경우에[예컨대, OpenID에서의 무상태 모드(stateless mode)], OP(306)는 정보를 RP(304)로 송신할 수 있다. 무상태 모드에서, 예를 들어, 발견 동안 등에 OP(306)와 RP(304) 사이에서 정보가 교환될 수 있다. 그렇지만, [예컨대, 사용자 식별자가, 예를 들어, http://myblog.blog.com에 있을 수 있고 및/또는 http://myblog.myopenid.com에 있는 OP에서의 OpenID OP 종단점 URL을 가리킬 수 있는 발견 위임(delegated discovery)의 경우에] 발견이 OP(306)를 수반하는 것이 보장되지 않을 수 있다. 따라서, myopenid.com에 있는 OP(306)가 발견에 직접 관여되지 않을 수 있고 이 스테이지에서 RP(304)를 인증할 수 없을지도 모른다.

[0042] OP(306)가 발견 단계 동안 RP(304)에 정보를 제공할 수 있는 경우[예컨대, 사용자 식별자 페이지가 OP(306) 자체에서 호스팅될 수 있는 경우], OP(306)는 발견 정보 페이지의 일부로서 넌스를 동적으로 발생할 수 있고 및/또는 이를 HTTP 요청 RP(304)의 식별자(예컨대, URL 또는 이메일 주소)와 연관시킬 수 있다. OP(306)는 RP(304)가 이 넌스에 서명하거나 그를 암호화하고 및/또는 정보를 리디렉션 메시지에 포함시킬 것으로 예상할 수 있다.

[0043] OP(306)는 강제로 HTTPS를 사용할 수 있다. 예를 들어, UE(302)와 OP(306) 사이의 임의의 차후의 통신이 HTTPS를 사용하여 보호될 수 있도록, UE(302)가 OP(306)에 의해 HTTPS의 사용으로 리디렉션될 수 있다. 이 특징은, 예를 들어, OpenID Authentication 2.0 등의 OpenID 표준 실시예에 의해 명백하게 가능하게 될 수 있다. 이러한 보호는, 예를 들어, OP(306)로부터 UE(302)로의 OpenID 인증 신청 메시지에 대한 MitM(man-in-the-middle) 공격의 방지를 가능하게 해줄 수 있다. 이는 RP 인증 실패의 경우에 경보 메시지가 보호된 방식으로 UE(302)로 송신될 수 있게 해줄 수 있다.

[0044] 분할 단말(split terminal) 구현예에 대한 예시적인 실시예가 본 명세서에 기술되어 있다. 분할 단말 구현예는 2개의 엔터티가 네트워크의 사용자측에 존재할 수 있는 시나리오를 말하는 것일 수 있다. 예를 들어, AA(Authentication Agent, 인증 에이전트) 및 BA(Browsing Agent, 브라우징 에이전트)가, 예를 들어, UE(302) 등의 UE와 연관되어 있고 및/또는 UE 상에 존재할 수 있다. AA는 인증을 위한 단계들을 수행할 수 있는 반면, BA는 서비스의 뷰어(viewer) 또는 소비 엔터티(consuming entity)일 수 있다. 분할 단말 구현예의 한 예에서, 사용자는, 예를 들어, RP(304) 등의 RP로부터의 어떤 서비스(예컨대, 웹 사이트)를 검색하기 위해 브라우저를 열 수 있다. RP(304)는 OP(306) 및 사용자의 AA와 어떤 단계들(예컨대, 연관 및/또는 발견)을 수행할 수 있다. 예를 들어, UE(302)가 OP(306)에 의해 접촉될 수 있다. OP(306) 및 UE(302)는, 예를 들어, BA가 모르고 있을지도 모르는 GBA 네트워크 자격 증명에 기초하여 인증을 수행할 수 있다. 예를 들어, OP(306)와 AA 사이의 인증이 성공한 경우 등에, BA는 RP(304)에서의 서비스에 액세스할 수 있다. 구현될 수 있는 다수의 변형례가 있을 수 있다. 각각의 변형례는, 예를 들어, 로컬 인터페이스(예컨대, 블루투스® 등)일 수 있는 AA와 BA 사이의 물리 채널 또는 논리 채널을 포함할 수 있다. 논리 채널은, 예를 들어, 2개의 세션이 논리적으로 결합될 수 있도록, 사용자가 AA 상에 보여지는 정보를 BA에 입력하는 것에 의해 생성될 수 있다.

[0045] MNO(Mobile Network Operator, 모바일 네트워크 통신 사업자) 자신의 서비스 및/또는 제3자 서비스 제공자의 서비스가 UE(302)에 또는 MNO가 알고 있는 디바이스에 제공될 수 있다. MNO가 사용자들이 하나의 인증기(authenticator)[예컨대, UE(302)]로 상이한/다수의 디바이스를 연결시킬 수 있게 해주고자 하는 경우, 분할 단말 구현예가 사용될 수 있다.

[0046] 분할 단말 구현예에 대한 예시적인 옵션은 2개의 세션 사이의 암호 바인딩(cryptographic binding)이 생성되는 옵션을 포함할 수 있다. 구현예는 또한 AA가 자격 증명 정보를 사용자에게 디스플레이하는 시나리오를 포함할 수 있고, 사용자는 RP(304)에 대해 인증하기 위해 이 자격 증명 정보를 BA에 입력할 수 있다.

[0047] 다른 대안으로서 또는 그에 부가하여, 자격 증명이 BA와 AA 사이의 보안 로컬 링크(secured local link)를 통해(예컨대, 본 명세서에 기술되어 있는 물리 채널을 사용하여) 송신될 수 있다. 이 구현예에서, AA는 인증 토큰/비밀 번호 발생기로서 사용될 수 있다. 예시적인 실시예에서, BA는 공유 키 K_1 및 인증 주장 메시지 UE_{Assert}[$K_{r,o}$에 의해 암호화되어 있을 수 있고, 이 암호화된 것은, 예를 들어, $EK_{r,o}(K_1,UE_{Assert})$라고 할 수 있음]를 AA로부터 수신하고 이를 RP(304)로 송신할 수 있다. 이 정보는 사용자를 인증하기 위해 RP(304)에 의해 사용될 수 있다. 예시적인 실시예에서, 분할 단말 구현예는 UE(302)/AA 내부에서 인증 주장 메시지 UE_{Assert}를 발생하는 로컬 주장 제공자에 의해 설정될 수 있다.

[0048] 로컬 OpenID에 기초한 인증에 따라, 부가된 보안 기능이 구현될 수 있다. 인증은 사적 비밀(private secret)[예컨대, 도 4에서 410 및 414에서 예시된 암호화 키 E]을 제공하기 위해 로컬 OpenID에 기초할 수 있다. 이 비밀은, 예를 들어, OP_{loc} 및/또는 OP_{loc}가 존재하는 신뢰된 환경(예컨대, 스마트 카드 또는 다른 신뢰된 컴퓨팅 환경)과 RP 사이에 사적 보안 채널(private, secure channel)을 설정하기 위해 사용될 수 있다. 다른 대안으로서, 보안 채널은 UE 플랫폼이라고 할 수 있는 UE의 어떤 비교적 보안되지 않은 부분에 있는 종단점을 가질 수 있다.

[0049] 이러한 보안 채널을 로컬 OpenID 인증에 바인딩시키는 옵셩이 본 명세서에 기술되어 있다. 예시적인 실시예에서, 보안 채널이 UE 플랫폼에 의해 설정될 수 있고, RP 및 로컬 OpenID 인증이 이 보안 채널 내부에서 수행될 수 있다. 이 예시적인 실시예는 어떤 구현예에 대해 충분할 수 있지만, 다른 구현예의 보안 요구를 충족시키지 않을 수 있다. 예를 들어, 보안 채널을 설정하는 UE 플랫폼은 OP_{loc}가 존재하는 신뢰된 환경(예컨대, 스마트 카드 또는 다른 신뢰된 컴퓨팅 환경)보다 덜 안전할 수 있다. 동일한 신뢰된 환경으로부터 오고 RP 쪽으로 보내지는 사적 데이터는 UE 내의 비교적 안전하지 않은 내부 노드를 가지는 채널을 통해 이동할 수 있다. 따라서, OP_{loc} 및/또는 OP_{loc}가 존재하는 신뢰된 컴퓨팅 환경이, UE 플랫폼의 특성과 관계없이, RP와 비밀을 교환할 수 있게 해주고 메시지의 이러한 프라이버시 특성을 RP에 대한 로컬 OpenID 인증에 바인딩시킬 수 있게 해줄 수 있는 대안의 실시예가 구현될 수 있다.

[0050] 도 4는, 예를 들어, UE/OP_{loc}(402) 등의 로컬 인증 엔터티와 Rp(104) 사이에 보안 채널을 생성 및/또는 구현하는 예시적인 실시예의 메시지 흐름도를 나타낸 것이다. 도 4에 예시된 흐름도는 UE/OP_{loc}(402), RP(404), 및/또는 OPSF(406) 사이의 통신을 포함하고 있다. 408에 나타낸 바와 같이, UE/OP_{loc}(402)가 410에서 서명된 인증 주장을 발생하는 지점까지 로컬 OpenID 인증이 수행될 수 있다. 410에서, UE/OP_{loc}(402)는 KDF(key derivation function, 키 도출 함수)를 사용하여 연관 핸들 A 및 공유 키 K_0의 함수로부터 도출될 수 있는 서명된 키 S를 발생할 수 있다. 공유 키 K_0는 보안 통신을 위해 UE/OP_{loc}(402)와 OPSF(406) 사이에서 공유될 수 있다. 서명 키 S는, 예를 들어, OpenID 서명 키일 수 있다. UE/OP_{loc}(402)는 로컬 인증을 수행할 수 있고, 인증 주장 메시지 UE_{Assert}가 410에서 발생될 수 있으며 암호화된 씨드 값(Seed)을 포함할 수 있다. Seed는 2개 이상의 당사자 사이의 공유 비밀을 은폐시키기 위해 사용될 수 있다. 예를 들어, 공유 비밀이 당사자들 사이에서 전송될 수 없기 때문에, 공유 비밀이 은폐될 수 있다. Seed는 그 대신에 전송되어, 비밀이 공유되는 당사자들 각각에서 공유 비밀을 (예컨대, 로컬적으로) 도출하기 위해 사용될 수 있다.

[0051] 인증 주장 메시지 UE_{Assert}는, 예를 들어, OpenID 주장일 수 있다. UE/OP_{loc}(402)는 OPSF(406), UE/OP_{loc}(402), 및/또는 RP(404)에 사적일 수 있는 서명 키 $S[E_S(Seed)$라고 함]로 Seed를 암호화할 수 있다. 대안의 실시예에서, UE/OP_{loc}(402)는 소정의 방식으로 S로부터 도출된 키를 사용하여 Seed를 암호화할 수 있다. UE/OP_{loc}(402)는, 예를 들어, RP(404)가 알고 있을 수 있는 소정의 방식으로 Seed로부터 암호화 키 E를 발생할 수 있다. UE/OP_{loc}(402)는 서명 키 S로 인증 주장 메시지 UE_{Assert}에 서명할 수 있다. 로컬 인증으로부터 암호화 키 E를 이와 같이 발생하는 것은 UE/OP_{loc}(402)와 RP(404) 사이에 보안 채널을 설정하는 것을 이 로컬 인증에 바인딩시킬 수 있다.

[0052] 412에서, UE/OP_{loc}(402)는 서명된 주장 UE_{Assert}와 함께 메시지(예컨대, http GET 요청)를 RP(404)로 송신할 수 있다. 414에서, RP(404)는 인증 주장 메시지 UE_{Assert}를 검증할 수 있고 서명 키 S를 사용하여 Seed 정보를 복호화할 수 있다. RP(404)는 Seed 정보에 기초하여 암호화 키 E를 발생할 수 있다. 예를 들어, RP(404)는 UE/OP_{loc}(402)가 알고 있을 수 있는 소정의 방식으로 Seed 정보로부터 암호화 키 E를 발생할 수 있다. 암호화 키 E는 UE/OP_{loc}(402) 및 RP(404)에 사적일 수 있다.

[0053] RP(404)는 앞서 검증된 인증 주장 UE_{Assert}를 암호화 키 E로 암호화하고 이를 다시 UE/OP_{loc}(402)로 송신할 수 있다. 예를 들어, 416에서, RP(404)는, 예를 들어, 암호화 키 E ($E_E(UE_{Assert})$)로 암호화되어 있을 수 있는 인증 주장 메시지 UE_{Assert}를 포함하는 통지를 UE/OP_{loc}(402)로 송신할 수 있다. 이것은 비밀 설정의 확인을 UE/OP_{loc}(402)에 제공할 수 있다. 418에서, UE/OP_{loc}(402)는 인증 주장 메시지 UE_{Assert}를 암호화 키 E를 사용하여

복호화함으로써 UE_{Assert}를 유효성 확인할 수 있다. 416에서 수신된 UE_{Assert} 내의 정보가 412에서 송신된 정보 UE_{Assert}와 일치하는 것을 검증함으로써, UE/OP_{loc}(402)는 416에서 수신된 메시지가 의도된 RP(404)로부터 발신되었다는 것을 확신할 수 있다. 예를 들어, 416에서 RP(404)로부터 수신된 통지에서의 Seed를 410에서의 UE_{Assert}에 포함되어 있는 Seed와 비교함으로써, UE/OP_{loc}(402)는 어떤 다른 RP도 인증 정보를 수신하지 않았다는 것과 RP(404)가 UE/OP_{loc}(402)가 그에 대해 인증을 수행하고자 했던 의도된 RP라는 것을 검증할 수 있다. UE/OP_{loc}(402)는 이 검증을 RP(404)가 Seed를 복호화하고 E를 도출하는 데 사용할 수 있는 키 S를 획득했다는 표시로서 신뢰할 수 있다. 420에서, UE/OP_{loc}(402)와 RP(404) 사이에 보안 채널을 설정하기 위해 (예컨대, 다른 프로토콜에서) 암호화 키 E가 사용될 수 있다. 이 보안 채널을 설정하는 데 사용될 수 있는 한 예시적인 프로토콜은, 사전 공유 키를 입력으로서 받고 사전 공유 키에 기초하여 보안 채널을 실현하는 통상의 TLS 프로토콜의 한 변형일 수 있는 TLS-PSK 프로토콜을 포함할 수 있다. TLS-PSK의 한 예시적인 실시예가 IETF(Internet Engineering Task Force)에 의해 RFC(Request for Comments) 문서 4279 및 4785에 예시되어 있다.

[0054] 도 4에 예시된 바와 같이, Seed의 정보 및 공개되어 있을 수 있는 KDF를 사용하여 암호화 키 E의 도출이 수행될 수 있다. Seed는 RP(404)가 알고 있을 수 있고, 다른 것들로부터 보호될 수 있는데, 그 이유는 Seed가 서명 키 S로 암호화되어 있기 때문이다. S는, 예를 들어, 인증서-기반 TLS(transport layer security, 전송 계층 보안) 등의 보안 채널을 통해 OPSF(406)에 의해 RP(404)에 노출되어 있을 수 있다. RP(404)가 $E_E(UE_{ASsert})$를 다시 UE(402)로 송신할 수 있기 때문에, UE(402)는 RP(404)가 서명 키 S를 소유하고 있다는 확인을 획득할 수 있고, RP(404)가 Seed를 복호화할 수 있는 경우 RP(404)는 이것을 할 수 있다. 따라서, UE(402)는 RP(404)로부터 키 확인(key confirmation)을 획득할 수 있다. 도 4에 예시되어 있는 프로토콜 흐름은, 보안 통신을 가능하게 해주기 위해, 본 명세서에 기술되어 있는 RP 인증 프로토콜 등의 RP 인증 프로토콜과 결합될 수 있다.

[0055] Seed 정보가 엔터티들 간의 사적 공유 키를 도출하는 데 사용될 수 있는 것으로 예시되어 있지만, 사적 공유 키가 다른 방식으로도 도출될 수 있다. 예를 들어, 실시예들은 Diffie-Hellman 키 설정을 구현할 수 있다.

[0056] 본 명세서에 기술된 바와 같이, 예를 들어, Seed 등의 어떤 초기값이 공유 비밀을 설정하고자 하는 엔터티들 간에 전송될 수 있다. 중간자 공격(man-in-the-middle attack)으로부터 Seed를 보호하기 위해 Seed의 암호화가 사용될 수 있다. 서명 키 S 또는 S로부터 도출된 키에 의한 특정의 암호화가 로컬 OpenID 인증에의 바인딩을 위해 사용될 수 있다. 암호화된 통지 메시지가 로컬 OpenID 인증에의 바인딩을 위해 사용될 수 있다. 이것은 UE/OP_{loc}(402)에 대해 비밀의 설정을 확인시켜 주는 특징을 부가할 수 있다.

[0057] 비밀의 설정은 RP(404)가 암호화된 Seed를 리디렉션 메시지에서 UE/OP_{loc}(402)로 송신하는 것에 의해 로컬 OpenID 프로토콜 흐름에서 조기에 시작될 수 있다.

[0058] 다른 실시예에서, RP(404)는 원하는 보안 채널의 종단점까지의 경로 상의 중간 노드일 수 있다. 이 경우에, RP(404)는 이 종단점으로부터 Seed를 수신할 수 있고, 이 종단점은 UE/OP_{loc}(402)가 그와의 보안 채널을 설정하고자 할 수 있는 그리고 RP(404)가 그에 대해 인증 게이트웨이, 및 선택적으로 허가 게이트웨이로서 동작할 수 있는 서버일 수 있다. 암호화 키 E는, 다른 프로토콜에서, UE/OP_{loc}(402) 또는 UE 플랫폼과 RP(404) 사이에 보안 채널을 설정하는 데 사용될 수 있다. 이러한 방식으로 암호화 키 E를 사용하는 후보 프로토콜은, 사전 공유 키를 입력으로서 받고 사전 공유 키에 기초하여 보안 채널을 실현하는 TLS 프로토콜의 한 변형일 수 있는 TLS-PSK 프로토콜을 포함할 수 있다. 일부 실시예에서, 비밀의 설정은 RP 인증과 결합될 수 있다.

[0059] 도 5는 사후 인증 키 확인(post- authentication key confirmation)에 의한 UE-RP 사전 설정된 보안 채널을 사용하는 로컬 OpenID 인증을 위한 보안 채널의 설정을 나타낸 흐름도이다. 예를 들어, 보안 채널 설정은 UE/OP_{loc}(502) 또는 UE 플랫폼 및 RP(504)가 보안 채널을 설정하고 로컬 OpenID 인증을 계속할 수 있게 해줄 수 있다. 도 5에 예시되어 있는 흐름도는 인증 동안 RP(504)에 대해 보안 채널 키를 확인시켜 주는 데 사용될 수 있고, 예를 들어, 인증에 바인딩되어 있을 수 있다. 이것은 키 자료(key material) XS를, 예를 들어, TLS(transport-layer security) 터널 등의 보안 채널로부터 추출함으로써 및/또는 바인딩 응답(binding response) B_{res}를 그로부터 도출함으로써 행해질 수 있다.

[0060] 도 5에 예시된 바와 같이, 508에서, UE/OP_{loc}(502) 및 RP(504)는 보안 채널을 설정할 수 있다. 예를 들어, 보안 채널은 TLS를 사용하여 설정될 수 있다. 510에서, UE/OP_{loc}(502)는 로그인 식별자(예컨대, OID)를 RP(504)로 전

송할 수 있다. 512에서, RP(504)는 연관 요청(예컨대, http POST OpenID 연관 요청)을 OPSF(506)로 송신할 수 있다. 514에서, OPSF(506)는 RP(504)와의 연관을 수행할 수 있다. 예를 들어, OPSF(506)는 연관 핸들 A 및/또는 공유 키 K_1을 발생할 수 있다. 공유 키 K_1은 OPSF(506), RP(504), 및/또는 UE/OP_{loc}(502) 사이의 공유 키일 수 있다. 공유 키 K_1은 연관 핸들 A 및/또는 공유 키 K_0로부터 도출될 수 있다. OPSF(506)는 연관 핸들 A 및/또는 공유 키 K_1을 RP(504)로 송신할 수 있다.

[0061] 516에서, RP(504)는 UE/OP_{loc}(502)를 UE/OP_{loc}(502) 상에 로컬적으로 존재하는 OP로 리디렉션시키는 리디렉션 메시지를 UE/OP_{loc}(502)로 송신할 수 있다. 리디렉션 메시지는 sessionID, returnURL, 넌스, 로그인 식별자(예컨대, OID), 및/또는 연관 핸들 A 등의 파라미터를 포함할 수 있다. 518에서, UE/OP_{loc}(502)는 로컬 인증을 수행할 수 있고, 공유 키 K_1을 발생할 수 있다. 공유 키 K_1은 연관 핸들 A 및/또는 공유 키 K_0로부터 발생될 수 있다. 로컬 인증으로부터 공유 비밀 K_1을 이와 같이 발생하는 것은 UE/OP_{loc}(502)와 RP(506) 사이에 보안 채널을 설정하는 것(508)을 이 로컬 인증에 바인딩시킬 수 있다. UE/OP_{loc}(502)는 보안 채널로부터 키 자료 XS를 추출할 수 있고, XS ($B_{res}=g(XS)$)로부터 바인딩 응답 B_{res}를 발생할 수 있다. 예시적인 실시예에 따르면, 바인딩 응답 B_{res}의 도출은, 예를 들어, 연관 핸들 A 등의 부가의 넌스로 MAC 알고리즘을 사용하여 행해질 수 있다. UE/OP_{loc}(502)는 바인딩 응답 B_{res}를 인증 주장 메시지 UE_{Assert}에 포함시킬 수 있다. B_{res}는, 예를 들어, OpenID에 의해 허용되는 바와 같이, 인증 주장 메시지 UE_{Assert}의 확장 필드에 포함될 수 있다. 인증 주장 메시지 UE_{Assert}는 공유 키 K_1를 사용하여 UE/OP_{loc}(502)에 의해 서명될 수 있고, 이 서명된 것은, 예를 들어, $SigK_1(UE_{Assert})$라고 할 수 있다. 520에서, UE/OP_{loc}(502)는 서명된 주장 메시지 $SigK_1(UE_{Assert})$를 RP(504)로 송신할 수 있다. 예를 들어, 서명된 주장 메시지는 http GET 요청에서 송신될 수 있다. 예시적인 실시예에서, XS가 RP(504)로의 메시지에서 직접 사용되지 않을 수 있는데, 그 이유는 이것이 보안 채널에 관한 정보를 공격자에게 누출시킬 수 있기 때문이다.

[0062] 522에서, RP(504)는 공유 키 K_1을 사용하여 서명된 주장 $SigK_1(UE_{Assert})$를 검증할 수 있다. 예를 들어, UE/OP_{loc}(502)로부터의 인증 주장을 성공적으로 검증한 후에, RP(504)는 RP(504) 자신의 보안 채널 키 자료 XS*로부터 비교 값 B_{res}*를 도출할 수 있고, 그것이 수신된 B_{res}와 일치한다는 것을 발견할 수 있다. 예를 들어, RP(504)는 보안 채널로부터 키 자료 XS*를 추출할 수 있고, 키 자료 XS*로부터 바인딩 응답 B_{res}* 를 발생할 수 있으며(B_{res}* = g(XS*)), 바인딩 응답 B_{res}*가 서명된 주장에 나타내어져 있는 바인딩 응답 B_{res}와 같은지를 검증할 수 있다. RP(504)는 인증된 당사자가 보안 채널 종단점이라는 것을 알고 있을 수 있는데, 그 이유는 RP(504)가 보안 채널 키의 키 확인(key confirmation)으로서 사용될 수 있는, 인증 프로토콜이 실행되었던 채널의 올바른 보안 채널 키를 소유하고 있기 때문이다. RP(504)가 바인딩 응답 B_{res}*이 바인딩 응답 B_{res}와 같은 것으로 검증하는 경우, 인증이 성공적인 것으로 결정될 수 있고 UE/OP_{loc}(502)와 RP(504) 사이의 채널이 안전할 수 있다. 524에서, RP(504)는 인증이 성공적이고 채널이 안전하다는 통지를 UE/OP_{loc}(502)로 송신할 수 있다.

[0063] 도 5에 예시된 바와 같이, 보안 채널은 TLS를 사용하여 설정될 수 있다. UE/OP_{loc}(502) 및 RP(504)는 (예컨대, OpenID 인증에 의해) 인증된 당사자가 또한 이전에 설정된 보안 채널의 종단점일 수 있다는 것을 RP(504)에 보장해줄 수 있는 키 확인을 프로토콜 내에 포함시킬 수 있다. 도 5에 예시되어 있는 예시적인 실시예는 OP_{loc}를 키 확인 및 보안 채널 설정은 물론 인증을 위한 신뢰 앵커(trust anchor)로서 사용하는 것을 포함할 수 있다. OP_{loc}를 사용함이 없이(예컨대, 외부 OP를 사용하여) 동일한 또는 유사한 보안을 달성하려고 시도하는 실시예는 RP(504)와 네트워크 OP 사이의 부가의 통신 단계를 필요로 할 수 있다. 도 5에 예시되어 있는 예시적인 실시예는 MitM(man-in-the-middle)이 보안(TLS) 채널의 설정 시에 그 자신을, 예를 들어, TLS 릴레이로서 설정하는 공격 등의 MitM 공격을 완화시킬 수 있다. 본 명세서에 기술되어 있는 실시예는 MitM이 RP(504)에 의해 명확히 검출될 수 있게 해줄 수 있다.

[0064] 인증 주장의 확장 필드를 사용하고자 하지 않는 경우, XS는 키 확인을 위해 사용될 수 있다. 예를 들어, UE/OP_{loc}(502)는 서명 키 $K_1' = g(K_1, XS)$(도시 생략)를 도출할 수 있고 이를 사용하여 인증 주장에 서명할 수 있다. RP(504)는 서명된 주장을 검증하기 위해 동일한 일을 행할 수 있다. 성공 시에, RP(504)는 보안 채널에

대한 인증 및 키 확인을 동시에 달성할 수 있다. 이것은 감소된 시맨틱스(reduced semantics)의 대가로 얻어질 수 있는데, 그 이유는 MitM의 존재가 인증의 실패로 인해 더 이상 분간할 수 없기 때문이다.

[0065] 도 5에 예시되어 있는 실시예는, 예를 들어, 본 명세서에 기술되어 있는 RP 인증 실시예 등의 RP 인증과 결합될 수 있다. 예를 들어, 채널 보안의 보장이, 도 5에서의 프로토콜에 예시되어 있는 바와 같이 단방향일 수 있다. 이를 양방향으로 만들기 위해, 이 프로토콜이, 예를 들어, 도 2 및 도 3에 예시되어 있는 RP 인증 프로토콜 등의 RP 인증 프로토콜과 결합될 수 있다. 이것을 위해, UE/OP_{loc}(502)는 암호화된 신청 값 $EK_1(RP_{Chv})$을 인증 주장 메시지에 포함시킬 수 있다. K_1이 결코 MitM으로 누설되지 않는 경우, UE/OP_{loc}(502)는, RP 신청 값 RP_{Chv}를 포함하는 통지를 수신할 시에, 유효한 RP(504)가 B_{res}의 성공적인 평가를 수행했고 따라서 MitM이 존재하지 않을 수 있는 것으로 가정할 수 있다. 이와 같이, RP(504)가 올바른 K_1을 소유하고 있는 경우, RP(504)는 RP_{Chv}를 복호화할 수 있다.

[0066] 다른 실시예에서, RP(504)는 바인딩 응답 B_{res}에 대한 정보를 가지고 있을 수 있다. 예를 들어, 524에서, B_{res}는 UE/OP_{loc}(502)로 반환되는 통지에서의 RP 신청 값 RP_{Chv}을 암호화하는 데 사용될 수 있다. UE/OP_{loc}(502)는 인증 주장 메시지 UE_{Assert} 내부에 있는 RP_{Chv}를 암호화시키기 위해, 예를 들어, K_0 또는 K_1보다는 K_1'를사용할 수 있다. 이어서, RP(504)는, 올바른 XS 값으로부터 도출된 K_1'를 소유하고 있는 경우, RP_{Chv}를 추출할 수 있다.

[0067] 본 명세서에 기술되어 있는 인증 및 키 합의 프로토콜은, 예를 들어, MitM 공격 등의 공격으로부터의 보호를 위한 다양한 구현예를 포함할 수 있다. 이러한 보호를 제공하는 한가지 방법은 인증 흐름 이전에, 예를 들어, TLS 터널 등의 외부 채널(outer channel)이라고 불릴 수 있는 보안 채널을 설정하는 것이다. 이 보안 채널 내부에서 인증이 수행될 수 있다. 예를 들어, GBA_H라고 하는 프로토콜은 TLS 터널에 의해 설정되는 외부 인증 프로토콜과 관련한 공격에 대해 충분히 안전할 수 있다. GBA_H는, 예를 들어, TLW를 통한 HTTP 다이제스트(HTTP digest)에 기초하고 있는 인증 절차를 포함할 수 있다. GBA_H의 한 예시적인 실시예는 3GPP(3rd Generation Partnership Project) TS(Technical Specification) 번호 33.220에 예시되어 있다.

[0068] 도 6은 HTTP-SIP 다이제스트(HTTP-SIP digest)를 사용하는 GBA_H의 한 예를 나타내는 메시지 흐름도를 나타낸 것이다. 도 6에 예시된 바와 같이, UE(602), BSF(604), 및/또는 HSS(606)를 사용하여 통신이 수행될 수 있다. 608에서, UE(602)는 BSF(604)와 TLS 터널을 설정할 수 있다. 610에서, UE(602)는, 예를 들어, TLS 터널을 사용하여 요청을 BSF(604)로 송신할 수 있다. 610에서의 요청은, 612에 예시되어 있는 바와 같이, 사적 ID(private identity)를 포함하는 허가 헤더(authorization header)를 포함할 수 있다. 614에서, BSF(604) 및 HSS(606)는 인증 정보를 교환하기 위해 Zh 참조점을 사용할 수 있다. 예를 들어, 616에 예시된 바와 같이, Zh BSF(604)는 Zh 참조점을 사용하여 HSS(606)로부터 인증 벡터(AV) 및/또는 사용자 프로파일 정보를 검색할 수 있다.

[0069] 618에서, BSF(604)는 인증 신청(예컨대, HTTP 401 허가되지 않음 응답에서의 인증 신청)을 UE(602)로 송신할 수 있다. 620에 예시되어 있는 바와 같이, 618에서의 메시지는 사적 ID 정보(private identity information), 영역(realm), 넌스(nonce), qop(quality of protection, 보호 품질) 값, 인증 알고리즘, 도메인(domain), 및/또는 불투명(opaque)을 포함할 수 있다. 예시적인 실시예에서, 이 정보는 메시지의 인증 헤더에 포함될 수 있다. 사적 ID 정보는 네트워크가 사용자를 식별하기 위해 사용하는 ID를 포함할 수 있다. 이 사적 ID는 네트워크가 사용자 프로파일 및/또는 신청에 대한 인증 벡터를 검색할 수 있게 해줄 수 있다. 예시적인 실시예에서, 영역, 넌스, qop 값, 인증 알고리즘, 도메인, 및/또는 불투명은 IETF에 의해 RFC 문서 2617에 예시되어 있을 수 있다. 622에서, UE(602)는 인증 응답을 계산할 수 있다. 624에서, UE는 인증 요청을 BSF(604)로 송신할 수 있다. 626에 예시되어 있는 바와 같이, 인증 요청은 사적 ID 정보, 영역, 넌스, cnonce, qop 값, 넌스 카운트(nonce count), 인증 요청, 다이제스트 uri(digest uri), 및 불투명을 포함할 수 있다. 예시적인 실시예에서, cnonce, 넌스 카운트, 및/또는 다이제스트 uri가 IETF에 의해 RFC 문서 2617에 예시되어 있을 수 있다. 628에서, BSF(604)는 응답을 계산하고 UE(602)로부터 수신된 값을 BSF(604)에서 계산된 값과 비교할 수 있다. 630에서, BSF(604)는 인증이 성공적이었다는 것을 UE(602)에 확인해주는 메시지(예컨대, 200 OK 메시지)를 UE(602)로 송신할 수 있다. 630에서의 메시지는, 632에 예시되어 있는 바와 같이, B_TID(binding trusted identifier) 및/또는 키 Ks 수명을 포함할 수 있다. 예시적인 실시예에서, B_TID 및 Ks 수명은 3 GPP TS 번호 33.220에 예시되어 있을 수 있다. 634에서, UE(602) 및 BSF(604)는 Ks_NAF를 계산할 수 있다.

[0070] 다른 예시적인 실시예는, 본 명세서에 기술되어 있는 바와 같이, TLS 외부 인증 및 GBA 메커니즘에 의해 설정된

인증 간의 바인딩을 포함할 수 있다. 제안된 바인딩 해결 방안은, 예를 들어, 624에서, UE(602)가 바인딩 응답 B_{res}를 메시지에 부가하는 것에 의해 만들어질 수 있다. B_{res}는 MitM가 아니라 BSF(604) 및 UE(602)가 알고 있는 방식으로 보안 채널에 의존할 수 있다. B_{res}는, 내부 인증(예컨대, AKA) 응답과 유사한(또는 심지어 동일한) 방식으로, 보안 채널 메시지로부터 도출될 수 있지만, 응답에 독립적일 수 있다. 예를 들어, B_{res}는 통상의 공개적으로 알려져 있는 방식으로 응답으로부터 도출되지 않을 수 있으며, 그렇지 않고 MitM가 유사한 방식으로 B_{res}를 도출할 수 있다. MitM이 존재하는 경우, BSF(604)는 보안 채널 UE(602)-MitM의 파라미터와 상이한 보안 채널 BSF(604)-MitM로부터의 파라미터를 사용하여 B_{res}의 검증을 수행할 수 있다. 이것에 대한 전제 조건은, 예를 들어, TLS 등의 프로토콜에 의해 만족될 수 있는 보안 채널의 유일성(uniqueness)을 포함할 수 있고, 여기서 BSF(604) 및 UE(602) 둘 다는 채널 설정에서 그 자신의 선택된 파라미터(예컨대, 넌스)를 도입할 수 있다. B_{res}의 검증 및/또는 재계산은, MitM에 의해 수행되는 경우, MitM가 타당한 B_{res} 값을 어떻게 도출하는지를 모를 수 있기 때문에, 실패할 수 있는 반면, MitM에 의한 GBA 응답의 재계산은 성공할 수 있다. 이러한 방식으로, MitM가 탐지될 수 있다.

[0071] 예시적인 실시예에서, UE(602)는 TLS 암호화 키를 받고 키 해시 함수(keyed hash function) H를 사용하여 이를 해싱할 수 있고, 여기서 키는 AKA 인증 신청에 의존한다. 이것은 618에서의 메시지에서 BSF(604)에 의해 제기될 수 있다. 예를 들어, AV가 적절히 형식 설정될 수 있고, AKA 신청 값 대신에, GBA 응답 계산 알고리즘에 직접 피드될 수 있다. 이것은 재생(replay)을 완화시킬 수 있고 보안 TLS 채널을 GBA 인증 실행(authentication run)에 바인딩시킬 수 있다(608).

[0072] 예시적인 실시예에 따르면, 보안 채널을 신청-응답 인증(618 내지 630)에 바인딩시키는 것(608)이 설정될 수 있다. 예를 들어, UE(602)는, 인증 신청(620)[예컨대, 내부 인증 신청(inner auth challenge)]을 수신한 후에, 수정된 신청*(modified challenge*)을 획득하기 위해 608에서 TLS 채널로부터 추출된 TLS 키로 다이제스트 알고리즘(digest algorithm) H(예컨대, HMAC 알고리즘)를 적용할 수 있다. 이것은, 예를 들어, H(TLS_key, inner_auth_challenge) -> challenge*로서 표현될 수 있다. TLS에 대한 키 추출 방법의 예시적인 실시예는 IETF에 의해 RFC 문서 5705에 예시되어 있다. UE(608)는 622에서 BSF(604)에 의해 제기된 신청에 대한 응다바을 계산할 수 있고, 이와 동시에 동일한 또는 유사한 알고리즘을 사용하여 바인딩 응답 B_{res}를 계산할 수 있다. 이것은, 예를 들어, AKA-RESPONSE (inner_auth_challenge)-> 응답; AKA-RESPONSE (challenge*, IK) -> B_{res}로서 표현될 수 있다. 624에서, UE는 응답 및 B_{res} 둘 다를 다시 BSF(604)로 송신할 수 있다.

[0073] BSF(604)는 UE(602) 응답을 검사함으로써 바인딩의 보장을 달성할 수 있다. 응답이 확인되는 경우, BSF(604)는 통신의 다른쪽 종단에 있는 엔터티가 인증되었다는 것을 알게 된다. B_{res}가 또한 확인되는 경우 - 이 때 BSF(604)는 검증을 위해 그 자신의 TLS 키를 사용함 -, 인증된 엔터티는 BSF(604)와 TLS 채널을 가지는 것일 수 있고, 그렇지 않은 경우, MitM가 의심될 수 있다.

[0074] 도 7은 SIP 다이제스트 인증에서 TLS와 GBA를 바인딩시키는 예시적인 호 흐름의 다이어그램이다. 도 7에 예시된 바와 같이, UE(702)는 BSF(704)와 TLS 세션을 개시함으로써 부트스트랩 절차(bootstrapping procedure)를 시작할 수 있다. UE(702)는 BSF(704)에 의해 제시된 인증서에 의해 BSF(704)를 인증할 수 있다. BSF(704)는 이 시점에서 UE(702)로부터의 인증을 필요로 하지 않을 수 있다. 708에서의 TLS 터널의 설정 이후에, 710에서, UE(702)는 사적 ID[즉, IMPI(IP multimedia subsystem private identifier)]를 포함하는 요청 메시지(예컨대, HTTP GET 요청)를 BSF(704)로 송신할 수 있다. 712에서, BSF(704)는 HSS(706)로부터 인증 정보[예컨대, AV(들)]를 요청할 수 있다. 714에서, HSS(706)는 [예컨대, AV(들)를 포함하는] 요청된 데이터를 BSF(704)에 제공할 수 있다. 716에서, BSF(704)는 인증 신청을 (예컨대, HTTP 401 허가되지 않음 응답에서) UE(702)로 송신할 수 있다. 인증 신청은 인증 헤더 및/또는 랜덤하게 발생된 넌스를 포함할 수 있다. 넌스에 부가하여, 인증 헤더는 사전 ID, 영역, qop 값, 알고리즘 정보, 및/또는 도메인 등의 부가의 파라미터를 포함할 수 있다.

[0075] 718에 나타낸 바와 같이, BSF(704)로부터 신청에 응답할 때, UE(702)는 랜덤한 cnonce를 발생하고, SIP 다이제스트 자격 증명에 의해 인증 응답을 계산할 수 있다. UE(702)는 또는, 예를 들어, TLS 터널 세션 키 및 세션 키 둘 다를 사용하여, 메시지 인증 코드(messages authentication code)(MAC) 값 B_{res}를 발생할 수 있다. TLS 터널 세션 키 및/또는 세션 키는, 예를 들어, 무결성 키(integrity key, IK) 또는 기밀성 키(confidentiality key, CK)를 포함할 수 있다. 예시적인 실시예에서, 무결성 보호를 위해 IK가 사용되도록 지정될 수 있기 때문

에, IK가 CK 대신에 사용될 수 있다. 이들 키는 UE(702)가 수신한 AV로부터 얻은 인증 신청 RAND로부터 발생될 수 있다. 이것은 TLS 터널 인증을 GBA 프로토콜과 바인딩시킬 수 있다. 720에서, 인증 신청 응답 및 $_{Bres}$ 둘 다는 허가 헤더에 넣어지고 다시 요청 메시지(예컨대, HTTP GET 요청 메시지)에서 BSF(704)로 송신될 수 있다. B_{res}는 인증 응답과 동일한 알고리즘에 의해 계산될 수 있지만, 기술된 바와 같이, 상이한 입력 파라미터로 계산될 수 있다.

[0076] BSF(704)는 B_{res}를 그 자신의 예상된 값 B_{res}*와 대조할 수 있다. BSF(704)는, B_{res}의 계산에서 사용된 키와 예상된 인증 응답 둘 다를 알고 있기 때문에, 그렇게 할 수 있다. 수신된 B_{res}가 B_{res}*와 일치하고, 수신된 인증 응답이 그의 예상된 값과 일치하는 경우, BSF(704)는 UE(702)가 진정한 것으로 결정할 수 있고 또한, 2번의 비교의 일치로부터 검증된 바인딩 효과로 인해, BSF(704)가 TLS 터널의 형성에서 인증했던 UE(702)가 BSF(704)가 프로토콜의 GBA 측면에서 인증했던 바로 그 UE(702)인 것을 확인할 수 있다. 722에서, BSF(704)는 GBA/GAA 마스터 세션 키 Ks의 키 수명 및 B-TID 등의 부트스트랩 키 자료를 발생할 수 있다. 724에서, BSF(704)는 B-TID 및 키 Ks를 포함하는 메시지(예컨대, 200 OK 메시지)를 UE(702)로 송신할 수 있다. UE(702) 및/또는 BSF(704)는 Ks를 사용하여 부트스트랩 키 자료 Ks_NAF를 도출할 수 있다. 예를 들어, 726에서, UE(702)는 Ks로부터 Ks_NAF를 발생할 수 있다. Ks_NAF는 Ua 참조점을 보호하는 데 사용될 수 있다.

[0077] [UE(702)와 NAF(network authentication function)(도시 생략) 사이의] Ua 참조점에 대한 보안을 위한 응용 관련 키가, 적어도 부분적으로, GBA를 통해 부트스트랩된 키로부터 도출될 수 있다. 예를 들어, Ks_NAF는 Ks = CK || IK로부터 도출될 수 있고, 여기서 CK 및 IK는 714에서 HSS(706)로부터 BSF(704)로 전달된 AV의 일부이다. Ks_NAF가 Ks 및 TLS 터널의 형성 동안 설정된 마스터 키 둘 다로부터 도출되는 경우, 이 바인딩은 여전히 유효할 수 있다. 따라서, Ks_NAF가 UE(702)와 네트워크 사이에서 공유될 수 있다. 이는 임의의 MitM에 의해 이용가능하지 않을 수 있다.

[0078] 본 명세서에 기술되어 있는 실시예는 클라우드 컴퓨팅 시나리오에서 구현될 수 있다. 예시적인 실시예에 따르면, 하나 이상의 사적 디바이스로부터 멀티-테넌트 지원 클라우드 액세스(multi-tenant capable cloud access)를 가능하게 해주기 위해 로컬 OpenID의 특성 및/또는 기술적 특징이 결합될 수 있다. 예를 들어, 로컬 OP 인증, RP 인증, 비밀 설정, 및/또는 등록 절차가 결합될 수 있다. 조직의 컴퓨팅 자원에 대한 아웃소싱의 적어도 2가지 측면이 본 명세서에 기술되어 있는 바와 같이 결합될 수 있다. 한 예시적인 측면에서, 원격, 외부, 이동 및 현장 작업자의 최근의 노동력 등급은 조직으로 하여금 작업을 위해 작업자의 사적 디바이스를 사용하도록 권장할 수 있다. 다른 예시적인 측면에서, 정보 및 컴퓨팅 자원이 점점 더 컴퓨터 클라우드(예컨대, 멀티-테넌트 호스팅 인프라 및/또는 서버)에 아웃소싱될 수 있다. 이 듀얼 아웃소싱(dual outsourcing) 시나리오에서의 아웃소싱 조직의 보안 요구사항은 그의 구현을 위해 선택된 보안 아키텍처에 대한 제약조건을 설정할 수 있다. 이들은, 예를 들어, 조직의 자산을 보호하기 위해 사용될 수 있는 보호 목표 및/또는 보안 제어의 점에서 기술될 수 있다.

[0079] 사용자 디바이스는 안전하지 않은 것으로 간주될 수 있다. 회사 데이터의 완전한 보호가 디바이스 상에서 가능하지 않을 수 있을지라도, 사용자 디바이스를 통한 데이터 손실 및/또는 누설을 가능한 범위까지 예방하는 등을 위해, 조직의 데이터가 적어도 클라우드 저장 장치에서 보호될 수 있다. 이것을 하는 한가지 방식은, 예를 들어, 클라우드에 있는 가상 워크스테이션에 연결될 수 있는 원격 데스크톱 응용 프로그램을 통한 클라우드에의 액세스를 가능하게 해주는 것일 수 있다. 한 이점으로서, 이것은 원격 작업자 및/또는 가상 워크스테이션이 상이한 운영 체제(OS)를 사용하는 것을 가능하게 해줄 수 있다. 예를 들어, 사용자 디바이스는 ANDROID™ 또는 APPLE® OS를 실행하는 태블릿일 수 있고, 예를 들어, 어떤 RDP(remote desktop protocol, 원격 데스크톱 프로토콜) 클라이언트 응용 프로그램 등을 통해 MICROSOFT WINDOWS® 가상 기계에 연결될 수 있다. 사용자 인증이, 예를 들어, 스마트 카드 또는 다른 신뢰된 환경에 바인딩되어 있을 수 있는 사용자측에서의 하드웨어 보호 대책에 의해 보호될 수 있다. 본 명세서에 기술된 바와 같이, 사용자 장비의 사용자(들)에 대한 로컬 OpenID로 인에이블되는 스마트 카드 또는 다른 신뢰된 환경이 발행될 수 있다. 본 명세서에 기술되어 있는 스마트 카드 또는 다른 보안 환경 실시예에서 사용하기 위해 사용자 계정이 등록될 수 있다.

[0080] 클라우드 호스트는 어떤 보안 제어 및/또는 계약상 보증을 제공할 수 있다. 클라우드 서비스를 사용하는 조직은 이러한 멀티-테넌트 환경에서의 데이터 손실 및/또는 누설에 대한 추가의 독립적인 보안 제어를 설정할 수 있다. 한 예로서, 조직의 IT 부서는 클라우드 워크스테이션의 (가상) 하드 드라이브에 대한 디스크 암호화 솔루션을 설치할 수 있다.

[0081] 클라우드 컴퓨터 상에서의 디스크 암호화에 의해 제공되는 보호는 제한될 수 있다. 클라우드 호스트의 하이퍼바이저는 가상 워크스테이션이 동작 중인 동안 완전한 데이터 액세스를 할 수 있다. 사용자가 워크스테이션에 로그온할 때, 클라우드 호스트의 하이퍼바이저는 하드 드라이브를 복호화하는 데 사용되는 송신된 자격 증명을 리스닝할 수 있다. 디스크 복호화는, 예를 들어, 신뢰된 컴퓨팅 기반 가상화 지원 기술을 사용하는 등에 의해, 어떤 방식으로 호스팅 하드웨어에 바인딩되어 있을 수 있다.

[0082] 원격 사용자 디바이스는, 예를 들어, 디스크 암호화 자격 증명(예컨대, 비밀 번호) 등의 비밀 데이터를 클라우드에 있는 가상 기계로 전송할 수 있다. 이러한 데이터는 그의 목적지에 은밀히 도달하도록 보호될 수 있고, 사용자에게 알려지지 않을 수 있다. 이 자격 증명은 지정된 가상 기계로 전달되도록 로컬 OpenID로 인에이블되는 스마트 카드 또는 다른 신뢰된 환경에 은밀히 저장될 수 있다.

[0083] 도 8은 로컬 인증 엔터티 및 클라우드/원격 컴퓨팅 서비스를 구현하는 예시된 통신 시스템의 다이어그램을 나타낸 것이다. 도 8에 예시된 바와 같이, 916에서, 회사 사용자는, 예를 들어, 스마트 카드(818) 또는 다른 신뢰된 환경을 회사(814)로부터 획득할 수 있다. 스마트 카드는 로컬 OpenID-지원(local OpenID-enabled) 스마트 카드일 수 있다. 스마트 카드(818)는, 예를 들어, OP_{loc}를 포함할 수 있다. 스마트 카드(818)는 클라우드 호스팅(cloud-hosted) 가상 기계(VM)(810) 등의 다른 곳에서 호스팅되는 회사(814) 자원에의 사적 액세스를 위한 자격 증명 보관소(credential vault)를 포함할 수 있다. 812에서, 회사(814)는 클라우드 호스팅 VM(810)에 연결되고 스마트 카드(818)를 통해 사용자 디바이스(802)에 의해 액세스하기 위한 회사(814) 정보, 서비스, 문서 등을 저장/업로드할 수 있다.

[0084] 820에서, 사용자는 스마트 카드(818)[예컨대, OP_{loc} 기능을 수행하기 위해 로컬 OpenID 기술로 인에이블되는 스마트 카드]를 사용자 디바이스(802)에 삽입할 수 있다. 사용자 디바이스(802)는, 예를 들어, 태블릿, 스마트폰, 휴대폰, 랩톱 컴퓨터, 또는 기타 모바일 디바이스일 수 있다. 사용자 디바이스(802)는 모바일 디바이스일 필요가 없고, 스마트 카드(818) 또는 다른 신뢰된 환경을 사용하여 클라우드 호스팅 VM(810) 상의 서비스에 액세스하도록 구성되어 있는 임의의 다른 컴퓨팅 디바이스일 수 있다. 어떤 응용 프로그램이, 예를 들어, 클라이언트 VM(810) 상의 원격 데스크톱에 액세스하기 위해 RDP(remote desktop protocol) 클라이언트를 포함할 수 있는 사용자 디바이스(802) 상에 설치될 수 있다. 원격 데스크톱에의 로그인은 스마트 카드 인증(예컨대, OpenID 인증) 절차를 위한 RP로서 역할할 수 있는 웹-기반 게이트웨이(806)를 통해 중재될 수 있다. 이 RP(806)는 클라우드 호스팅 VM(810)에 존재할 수 있거나, 독립적인 엔터티일 수 있다. RP(806)는 아웃소싱 회사에 보안 서비스로서 제공될 수 있거나, 회사(814) 자체에 의해 운영될 수 있다. 게이트웨이 RP(806)는, 808에서, 클라우드 호스팅 VM(810)에의 사적 보안 연결을 가질 수 있다.

[0085] 로컬 OpenID-기반 로그온은 본 명세서에 기술되어 있는 적어도 3개의 보안 특징 중 하나 이상을 결합할 수 있다. 예를 들어, 로컬 OpenID-기반 로그온은 (1) OP_{loc}를 통한 사용자의 인증; (2) 스마트 카드(818) 상의 OP_{loc}에 대한 RP(806)(예컨대, 보안 게이트웨이)의 인증; 및/또는 (3) 스마트 카드(818)와 RP(806) 간의 종단간, 그리고 선택적으로 클라우드 호스팅 VM(810)에 추가로 위임되는 사적 비밀의 설정을 포함할 수 있다. 스마트 카드(818) 상의 OP_{loc}를 통한 사용자의 인증은 스마트 카드(818)를 소유하는 것 및 인증 비밀(authentication secret)을 아는 것에 의한 (적어도) 2-요소 인증, 및 생체 인식 사용자 인증을 포함할 수 있다. 804에서, 인증 및/또는 비밀 전달은 사용자 디바이스(802)와 RP(806) 사이의 보안 통신을 통해 수행될 수 있다. 사용자가 스푸핑된 사이트(spoofed site)가 아니라 지정된 회사 자원에 연결하는 것을 보장해주기 위해 스마트 카드(818) 상의 OP_{loc}에 대한 RP(806)의 인증이 사용자로 확장될 수 있다. 예를 들어, RP(806) 인증을 위한 자격 증명이 스마트 카드(818)에 은밀히 포함될 수 있다. RP(806)는 사용자 디바이스(802)에 대한 비밀을 클라우드 호스팅 VM(810)에 위임할 수 있거나, 예를 들어, 2개의 보안 채널의 중간점으로서 역할할 수 있다.

[0086] 스마트 카드(818) 상의 OP_{loc}와 RP(806) 사이에 비밀이 설정될 때, 스마트 카드(818) 상의 자격 증명 보관소가 잠금 해제될 수 있다. 클라우드 호스팅 VM(810) 상의 데이터 액세스를 위한 자격 증명은 (예컨대, 카드 상의) 설정된 비밀로 암호화되고 및/또는 클라우드 호스팅 VM(810)으로 전송될 수 있다. 그곳에서, 자격 증명은 복호화되고 검증될 수 있으며, 검증이 성공적인 경우, 비밀은 사용자 데이터를 복호화하는 데 사용될 수 있다. 사용자는 원격 데스크톱 응용 프로그램을 통해 클라우드 호스팅 VM(810) 상에서 작업을 할 수 있다. 사용자는, 예를 들어, 클라우드 호스팅 VM(810)으로부터 회사 인트라넷으로의 보안 연결을 통해 회사 자원에 액세스할 수 있다.

[0087] 도 9는 SIP 다이제스트 인증을 사용하고 OpenID에서의 RP(904) 인증을 포함하는 예시적인 프로토콜 흐름을 나타낸 것이다. 인증은 RP(904)와 OP(908) 사이의 사전 공유 키 $K_{r,o}$를 사용한 OP(908)에 대한 UE(902)의 인증을 포함할 수 있다. OpenID 인증에서의 RP 인증은, 차례로, SIP 다이제스트 인증으로부터 부트스트랩될 수 있다. 도 9에 예시되어 있는 프로토콜 흐름은 UE(902), RP(904)(예컨대, 응용 프로그램 서버), OP(908)[예컨대, SSO(Single-Sign-on) 서버], 및 HSS(910) 사이의 통신을 포함하고 있다. RP(904) 및 OP(908)는, 906에서, 엔터티들 간의 보안 통신을 위해 사용되는 공유 비밀 $K_{r,o}$를 사전 설정했을 수 있다.

[0088] 도 9에 예시되어 있는 프로토콜에서, OpenID는 UE(902) 인증을 위한 그의 무상태 모드에서 사용될 수 있다. 단계(912) 내지 단계(918)의 조합이 OP(908)에서의 RP(904) 인증을 달성하기 위해 사용될 수 있다. 912에서, UE(902)는 IMS[IP(internet protocol) multimedia subsystem]에 등록할 수 있다. 914에서, UE(902)는 인증 요청(예컨대, OpenID 인증 요청)을 RP(904)로 송신할 수 있다. 인증 요청은 인증 식별자(예컨대, OID)를 포함할 수 있다. 916에서, RP(904)는 리디렉션 요청을 UE(902)로 송신할 수 있다. 916에서의 리디렉션 요청은 UE(902)를 OP(908)로 리디렉션할 수 있다. 리디렉션 요청은 인증 식별자(예컨대, OID) 및/또는 RP(904)에 대응하는 RP 자격 증명 RP_{Cred}를 포함할 수 있다. RP_{Cred}는 OP(908)와 공유되는 사전 공유 키 $K_{r,o}$로 서명될 수 있다. 918에서, UE(902)는 리디렉션 요청 메시지를 OP(908)로 송신할 수 있다. 리디렉션 요청 메시지는 인증 식별자(예컨대, OID) 및 916에서 RP(904)로부터 수신된 RP 자격 증명 RP_{Cred}를 포함할 수 있다.

[0089] 920에서, OP(908)는 RP_{Cred}를 사용하여 RP(904)의 인증을 수행하고 및/또는 RP 인증 주장을 발생할 수 있다. OP(908)는 또한, UE(902)와 OP(908) 사이의 보안 통신을 보장해주기 위해, UE(902)와 OP(908) 사이의 공유 키일 수 있는 공유 키 K_0의 검사를 수행할 수 있다. 922에서, OP(908)는 RP(904)가 인증되었는지를 결정할 수 있다. RP(904)가 922에서 제대로 인증되지 않은 경우, 924에서, OP(908)는 RP(904)가 불량 RP라는 것을 나타내는 경보를 UE(902)로 송신하고 절차를 종료할 수 있다. RP(904)가 922에서 제대로 인증되는 경우, OP(908)는 프로토콜을 계속할 수 있다. 예시적인 실시예에서, 926에서 RP(904)가 진정한 것으로 결정되는 경우, 920에서의 RP(904) 인증 주장의 발생이 행해질 수 있다. 예시적인 실시예에서(도 9에 예시되지 않음), 922에서의 RP(904) 인증 결정이 OP(908)가 RP 인증에 대한 결정을 하는 지점으로서 간주되는 경우, RP(904) 인증 결정 이후의 단계들에서 RP_{Assert} 사용이 프로토콜로부터 생략될 수 있다.

[0090] 예시적인 변형례에서, RP_{Cred}는 RP(904)의 평문 식별자일 수 있고(즉, 어떤 키로도 서명되어 있지 않음), 이는 OP(908)가 장래의 사용을 위한 올바른 공유 키 $K_{r,o}$를 선택할 수 있게 해줄 수 있다. 이 경우에, RP_{Cred}가 OP(908)이 알고 있는 어떤 RP에도 대응하지 않는 경우, OP(908)는 절차를 종료하기로 결정하고 UE(902)에 통지할 수 있다.

[0091] 도 9에 예시되어 있는 예시적인 메시지 흐름을 계속하여, SIP 다이제스트 인증이 수행될 수 있다. 예를 들어, 928에서, OP(908)는 SIP 다이제스트 인증 벡터(SIP digest authentication vector, SD-AV) 및/또는 사용자 프로파일 정보를 HSS(910)로부터 획득할 수 있다. OP(908)는 사용자 자격 증명(예컨대, 사용자 이름/비밀 번호)에 기초하여 이러한 정보를 획득할 수 있다. OP(908)는 또한 사용자 자격 증명, 영역, qop 값, 인증 알고리즘, 및/또는 해쉬 H(A1)를 HSS(910)로부터 획득할 수 있다. 예시적인 실시예에서, 영역, qop 값, 인증 알고리즘, 및/또는 해쉬 H(A1)는 IETF에 의해 RFC 문서 2069 및 2617에 예시되어 있을 수 있다.

[0092] 930에서, OP(908)는 넌스를 발생하고 넌스 및 H(A1)을 저장할 수 있다. 932에서, OP(908)는 인증 신청(예컨대, 인증 신청을 갖는 HTTP 401 허가되지 않음 메시지)을 UE(902)로 송신할 수 있다. 인증 신청은 사용자 자격 증명, 넌스, 영역, qop 값, 및/또는 인증 알고리즘을 포함할 수 있다. 934에서, UE(902)는 cnonce, H(A1), 및/또는 보안 통신을 위해 OP(908)와 공유되는 비밀키 K_0를 발생할 수 있다. 936에서, UE(902)는 또한 신청 응답을 계산하고 신청 응답(예컨대, 인증 응답을 갖는 HTTP GET 메시지)을 OP(908)로 송신할 수 있다. 신청 응답은 cnonce, 응답, 넌스, 사용자 자격 증명, 영역, qop 값, 인증 알고리즘, 다이제스트 uri, 및/또는 넌스 카운트를 포함할 수 있다. 예시적인 실시예에서, cnonce, 넌스, 영역, qop 값, 인증 알고리즘, 다이제스트 uri, 및/또는 넌스 카운트는 IETF에 의해 RFC 문서 2617에 예시되어 있을 수 있다. 공유 키 K_0는 공유 키 K_0를 SIP 다이제스트 인증에 바인딩시킬 수 있는 인증 응답으로부터 도출될 수 있다. 938에서, OP(908)는 넌스와 대조하고, Xresponse를 계산하며, 및/또는 Xresponse를 UE(902)로부터 수신된 응답과 비교할 수 있다.

[0093] SIP 다이제스트 인증이 성공하는 경우(예컨대, Xresponse, 또는 그 안에 있는 어떤 파라미터가 응답 또는 그 안

에 있는 어떤 파라미터와 일치하는 경우), 938에서, OP(908)는 UE 인증 주장 UE_{Assert} 및/또는 공유 키 K_0를 발생할 수 있다. 940에서, OP(908)는 nonce1 및/또는 UE(902)와 RP(904) 사이에 보안 채널을 설정하는 데 사용되는 UE(902), OP(908), 및/또는 RP(904) 사이의 공유 키일 수 있는 K_1을 발생할 수 있다. K_1은 신선성을 위해 발생에서 nonce1을 사용하여 OP(908)에 의해 발생될 수 있다. K_0는 nonce1 및/또는 RP 인증 주장 메시지 RP_{Assert}를 암호화하는 데 사용될 수 있고, 이 암호화된 것은, 예를 들어, EK_0 (nonce1, RP_{Assert})라고 할 수 있다. K_0에 의한 암호화는 정당한 인증된 UE(902)가 자신이 의도된 진정한 RP(904)와 통신하고 있다는 UE(902)에 대한 확인일 수 있는 RP_{Assert}를 획득할 수 있게 해줄 수 있다. OP(908)는 공유 키 $K_{r,o}$를 사용하여 키 K_1 및/또는 UE 인증 주장 메시지 UE_{Assert}를 암호화할 수 있고, 이 암호화된 것은, 예를 들어, $EK_{r,o}$(K_1, UE_{Assert})라고 할 수 있다. 942에서, OP(908)는 UE(902)를 RP(904)로 리디렉션할 수 있는 리디렉션 메시지를 UE(902)로 송신할 수 있다. 리디렉션 메시지는 EK_0(nonce1, RP_{Assert}) 및/또는 $EK_{r,o}$(K_1, UE_{Assert})를 포함할 수 있다. 예시적인 실시예에서, 944에 예시되어 있는 바와 같이, OP(908)가 RP(904) 신뢰성에 대한 결정 지점일 수 있기 때문에 프로토콜 흐름에서의 특정의 지점에서 RP 인증 주장 메시지 RP_{Assert}가 쓸모없게 될 수 있다. UE(902)가 의도된 RP(904)와 안전하게 통신하고 있도록 보장해주기 위해, 예를 들어, 구현 관련 단계(952 및/또는 954)를 수행하는 등, RP(904)가 UE(902)와 통신을 수행할 때, K_1이 사용될 수 있다.

[0094] 946에서, UE(902)는 K_0를 사용하여 nonce1 및/또는 RP 인증 주장 메시지 RP_{Assert}를 복호화할 수 있다. K_0를 사용하여 RP 인증 주장 RP_{Assert}를 복호화할 수 있는 것에 의해, UE(902)는 자신이 의도된 진정한 RP(904)와 통신하고 있다는 것을 확인할 수 있다. UE(902)는 RP 인증 주장 메시지 RP_{Assert} 및 nonce1을 획득할 수 있다. UE(902)는 수신된 RP 인증 주장 RP_{Assert}에 기초하여 RP(904)를 인증할 수 있다. UE(902)는 nonce1을 사용하여 K_1을 발생할 수 있다. 공유 키 K_1에 의한 암호화는 정당한 인증된 UE(902)가 서비스에 대해 사용하기 위한 액세스 토큰으로서 역할할 수 있는 UE_{Author}를 획득할 수 있게 해줄 수 있다. 948에서, UE(902)는 RP(904)로 리디렉션될 수 있다. 948에서, UE(902)는 키 K_1 및 UE 인증 주장 메시지 UE_{Assert}를 RP(904)로 송신할 수 있다. 키 K_1 및 UE_{Assert}는 공유 키 $K_{r,o}$로 암호화될 수 있고, 이 암호화된 것은, 예를 들어, $EK_{r,o}$(K_1, UE_{Assert})라고 할 수 있다. 이 암호화는 OP(908)에 의해 이전에 수행되었을 수 있다. 950에서, RP(904)는 $K_{r,o}$를 사용하여 $EK_{r,o}$(K_1, UE_{Assert})를 복호화하고 UE_{Assert} 및 K_1을 획득할 수 있다. 950에서, UE(902)에 대한 정보가 허가될 수 있다. 예를 들어, RP(904)는 K_1을 사용하여 UE_{Assert}의 서명을 검증할 수 있다. UE_{Assert}를 성공적으로 검증한 후에, RP(904)는 키 K_1으로 암호화될 수 있는 허가 정보 UE_{Author}를 발생할 수 있고, 이 암호화된 것은, 예를 들어, $EK_1(UE_{Author})$라고 할 수 있다. UE_{Author}는 UE(902)가 RP(904)에서의 하나 이상의 서비스에 액세스하도록 허가되어 있다는 것을 나타내는 허가 정보 또는 허가 파라미터를 포함할 수 있다. 952에서, RP(904)는 UE(902)가 RP(904)에서의 서비스에 대해 허가되어 있는지를 UE(902)에 통지할 수 있다. 예를 들어, RP(904)는 UE 허가 파라미터 또는 정보 UE_{Author}를 송신할 수 있다. UE_{Author}는 UE(902)와 RP(904) 사이에서 공유되는 비밀 키 K_1 ($EK_1(UE_{Author})$)로 암호화될 수 있다. 954에서, UE(902)는 $EK_1(UE_{Author})$를 복호화하고 UE_{Author}를 사용하여 RP(904)로부터의 요청된 서비스에 액세스할 수 있다. 단계(952 및/또는 954)는, 선택적일 수 있고 UE(902) 및/또는 RP(904)의 서비스 구현에 의존할 수 있는, 구현 관련 단계일 수 있다. 예를 들어, 이들은 인증 후에 UE(902)에 일반 서비스 액세스를 제공하는 원하는 응용에 특유한 것일 수 있다. 이들 단계가 사용되지 않는 경우, K_1은 필요하지 않을 수 있다.

[0095] 예시적인 실시예에서, 도 9에 예시되어 있는 프로토콜 흐름은 OP(908)에 대한 RP(904) 인증을 달성하기 위해 비밀 $K_{r,o}$를 사용할 수 있다. 예를 들어, 비밀 $K_{r,o}$는, OP(908)에 대해 [예컨대, 단계(912) 내지 단계(918)에서] RP_{Cred}로 메시지에 서명하기 위해 사용되지 않는 경우, 인증을 위해 사용될 수 있다. 예를 들어, OP(908) 및 RP(904)가 이미 비밀 $K_{r,o}$를 공유하고 있는 경우, 이 비밀은 OP(908)에 대한 RP(904) 인증을 위해 사용될 수 있다. 인증 프로토콜(예컨대, OpenID 프로토콜)의 발견 및 (선택적인) 연관 생성 단계가 도 9에 예시되어 있는 프로토콜에 나타내어져 있지 않다. UE(902) 상에서의 구현은 임의의 이러한 RP(904) 인증에 의해 영향을 받지 않을 수 있다. 예를 들어, 일 실시예에서, UE(902)는 OP_{loc} 기능을 포함하지 않을 수 있고, 따라서, 신청 RP_{Chv}

를 RP로 송신할 수 없을지도 모른다.

[0096] 도 10은 OP(1008)에 대한 RP(1004) 인증에서의 예시적인 프로토콜의 메시지 흐름도를 나타낸 것이다. 도 10에서, UE(1002), RP(1004)(예컨대, 응용 프로그램 서버), OP(1008)(예컨대, SSO 서버), 및/또는 HSS(1010) 사이에서 통신이 수행될 수 있다. RP(1004) 및 OP(1008)는, 1006에 예시되어 있는 바와 같이, 보안 채널을 통한 보안 통신을 가능하게 해주는 사전 설정된 공유 비밀을 가질 수 있다.

[0097] 도 10에 예시된 바와 같이, 1012에서, UE(1002)는 로그인 식별자(예컨대, URL 또는 이메일 주소 등의 OpenID 식별자)를 포함하는 인증 요청(예컨대, OpenID 인증 요청)을 RP(1004)에 발행할 수 있다. 1014에서, RP(1004)는 OP(1008)를 발견할 수 있다. 1016에서, RP(1004)는 연관 요청(예컨대, OpenID 연관 요청)을 OP(1008)로 송신할 수 있다. RP(1004) 및 OP(1008)는 Diffie-Hellman 키 D-H를 설정할 수 있다. OP(1008)는 연관 비밀 및/또는 연관 핸들(모두 합하여 연관이라고 할 수 있음)을 발생할 수 있다. 1018에서, OP(1008) 및 RP(1004)는 연관 비밀 및 nonce0를 포함할 수 있는 연관 응답을 RP(1004)로 송신할 수 있다. 연관 비밀 및/또는 nonce0는 설정된 D-H 키로 암호화될 수 있다. 1020에서, RP(1004)는 수신된 암호화된 nonce0 및 암호화된 연관 비밀을 복호화할 수 있다. RP(1004)는 이어서 RP(1004)와 OP(1008) 사이에 공유되는 사전 설정된 키일 수 있는 공유 키 $K_{r,o}$로 nonce0에 서명할 수 있다. nonce0에 서명하기 위해 HMAC 또는 다른 적당한 대칭 서명 알고리즘이 사용될 수 있다. RP(1004) 및 OP(1008)는 공지의 메커니즘을 사용하는, 예를 들어, Diffie-Hellman 키 교환 프로토콜 또는 사전 공유 비밀을 사용하는 공유 비밀 $K_{r,o}$을 가질 수 있다. 이 공유 비밀로, OP(1008) 및 RP(1004)는 메시지에 서명할 수 있고, 공유 비밀 $K_{r,o}$로 서명되어 있는 서로의 메시지를 검증할 수 있다.

[0098] 1022에서, RP(1004)는 리디렉션 메시지를 사용하여 UE(1002)에 의해 송신된 인증 요청을 리디렉션할 수 있다. 리디렉션 메시지는 로그인 식별자(예컨대, OpenID 식별자), RP(1004) 식별자(RP_{Cred}), 및/또는 서명된 nonce0를 포함할 수 있다. 예를 들어, UE(1002)가 OP(1008)로 리디렉션될 수 있다. 1024에서, 인증 요청이 OP(1008)로 리디렉션될 수 있다. 리디렉션은 로그인 식별자(예컨대, OpenID 식별자) 및/또는 RP_{Cred}를 포함할 수 있다. 1026에서, OP(1006)는 보안 통신을 위해 UE(1002)와의 통신에 대해 HTTPS의 사용을 시행할 수 있다. HTTPS의 사용의 시행은 OP(1002)의 웹 서버의 구성[예컨대, 주소 다시 쓰기(address rewrite)]에 의해 수행될 수 있다. 1028에서, OP(1008)는 RP(1004)를 인증하기 위해 nonce0의 서명을 검증할 수 있다. 예를 들어, OP(1008)는 공유 키 $K_{r,o}$를 사용하여 서명을 검증할 수 있다. 단계(1028)의 RP(1004) 인증은 1030에서 결정될 수 있고, 인증이 실패하는 경우, 1032에서, OP(1008)는, RP(1004) 인증 실패를 알려주기 위해, 예를 들어, HTTPS에 의해 보호될 수 있는 경보 메시지를 UE(1002)로 송신할 수 있다. 1028에서의 RP(1004) 인증이 성공하는 경우, 예를 들어, 단계(1034) 등에서 프로토콜 흐름이 계속될 수 있다.

[0099] 1034에서, OP(1008)는 OP(1008)와 UE(1002) 사이에 보안 채널이 설정되었는지를 결정할 수 있다. 예를 들어, OP(1008)는 유효한 키 K_0가 존재하는지를 결정할 수 있다. 유효한 키 K_0가 존재하는 경우, 프로토콜 흐름은 UE 인증 주장 UE_{Assert}를 발생하는 단계(1048)로 진행할 수 있다. 유효한 키 K_0가 존재하지 않는 경우, 프로토콜 흐름은 계속하여 UE(1002)의 인증을 수행할 수 있다. 예시적인 실시예에서, 보안 채널의 설정(예컨대, 도 4에 예시되어 있음) 및 UE(1002)의 인증이 동일한 프로토콜 흐름에서 서로 바인딩되어 있을 수 있다. 1036에 예시되어 있는 바와 같이, OP(1008)는 인증 요청을 HSS(Home Subscription Server)(1010)로 송신할 수 있고, HSS(1010)로부터의 사용자 자격 증명에 기초하여 SIP 다이제스트 인증 벡터(SD-AV) 및/또는 사용자 프로파일을 획득할 수 있다. SD-AV는 qop 값, 인증 알고리즘, 영역, 및 사용자 자격 증명, 영역 및 비밀 번호의 해쉬[H(A1)이라고 함]를 포함할 수 있다. 다중 HSS 환경에서, OP(1008)는 SLF(Service Layer Function)에 질의함으로써 UE(1002)의 가입의 상세가 저장되어 있는 HSS(1010)의 주소를 획득할 수 있다. 1038에서, OP(1008)는 랜덤한 넌스를 발생할 수 있고, 사용자 자격 증명에 대한 해쉬 H(A1) 및 넌스를 저장할 수 있다. 1040에서, OP(1008)는 넌스, 영역, qop 값, 인증 알고리즘, 및 사용자 자격 증명을 포함할 수 있는 인증 신청 메시지(예컨대, SIP 다이제스트 인증 신청으로서의 401 인증 신청)를 (예컨대, 보호된 HTTPS 메시지에서) UE(1002)로 송신할 수 있다.

[0100] 1040에서 신청을 수신할 시에, 1042에서, UE(1002)는 랜덤한 cnonce 및 H(A1)을 발생할 수 있다. UE(1002)는 H(A1), cnonce 및/또는 기타 정보(예를 들어, 인증 신청에 포함되어 있는 자료 등)에 기초하여 공유 비밀 K_0를 발생할 수 있다. 공유 비밀 K_0는 UE(1002)와 OP(1008) 사이의 통신이 보안 채널을 사용하여 전송될 수 있게 해줄 수 있는 UE(1002)와 OP(1008) 사이의 공유 비밀일 수 있다. UE(1002)는 cnonce 및/또는 인증 신청에서 제

공되는 기타 파라미터(예를 들어, 넌스, 사용자 자격 증명, 및/또는 qop 값 등)를 사용하여 인증 응답을 계산할 수 있다. 1044에서, UE(1002)는 신청 응답(예컨대, 보호된 HTTPS 메시지일 수 있음)을 OP(1008)로 송신할 수 있다. 신청 응답은, 예를 들어, cnonce, 넌스, 응답, 영역, 사용자 자격 증명, qop 값, 인증 알고리즘, 넌스 카운트, 및/또는 다이제스트 uri를 포함할 수 있다. 1044에서 응답을 수신할 시에, OP(1008)는 이전에 저장된 넌스를 사용하여 응답에 포함되어 있는 넌스와 대조할 수 있다. 대조가 성공적인 경우, OP(1008)는, 응답에 포함되어 있는 기타 파라미터(예컨대, cnonce, 넌스 카운트, qop 값 등)와 함께 이전에 저장된 해쉬 H(A1) 및 넌스를 사용하여, 예상된 응답(Xresponse)을 계산할 수 있고, 이것을 사용하여 UE(1002)로부터 수신된 응답과 대조할 수 있다. 대조가 성공적인 경우, UE(1002)의 인증이 성공한 것으로 간주될 수 있다. 대조가 성공적이지 않은 경우, 인증이 실패한 것으로 간주될 수 있다. UE(1002)가 성공적으로 인증된 경우, OP(1008)는 H(A1), cnonce 및/또는 기타 정보(예를 들어, 인증 신청에 포함되어 있는 자료 등)에 기초하여 발생될 수 있는 공유 비밀 K_0를 발생할 수 있다. 다른 대안으로서 또는 그에 부가하여, 1044에서 응답을 수신할 시에, OP(1008)는 인증 주장 UE_{Assert}를 생성할 수 있다. UE_{Assert}는, 예를 들어, 1018에서 메시지에 사용된 연관 비밀일 수 있는 연관 비밀을 사용하여 서명될 수 있다.

[0101] 1050에서, OP(1008)는 랜덤한 nonce1을 발생할 수 있고 및/또는 K_0 및 nonce1에 기초하여 공유 비밀 K_1을 발생할 수 있다. 공유 비밀 K_1은 UE(1002)와 RP(1004) 사이에 보안 채널을 설정하기 위한 UE(1002), OP(1008), 및/또는 RP(1004) 사이의 공유 비밀일 수 있다. OP(1008)는 K_0를 사용하여 nonce1을 암호화할 수 있고[이 암호화된 것은, 예를 들어, EK_0(nonce1)이라고 할 수 있음], $K_{r,o}$를 사용하여 K_1 및 서명된 주장 메시지 UE_{Assert}를 암호화할 수 있고, 이 암호화된 것은, 예를 들어, $EK_{r,o}$(K_1, signed(UE_{Assert}))라고 할 수 있다. 1052에서, OP(1008)는 RP(1004)로의 리디렉션과 함께 EK_0(nonce1) 및/또는 $EK_{r,o}$(K_1, signed(UE_{Assert}))를 포함할 수 있는 메시지(예컨대, 리디렉션 메시지)를 UE(1002)로 송신할 수 있다. 1054에서, UE(1002)는 공유 키 K_0를 사용하여 EK_0(nonce1)를 복호화할 수 있고 nonce1을 획득할 수 있다. UE(1002)는 K_0 및 nonce1에 기초하여 공유 비밀 K_1을 발생할 수 있다. 1056에서, OP(1008)에 의해 송신된 메시지는 RP(1004)로 리디렉션될 수 있다. 1056에서의 메시지는 $EK_{r,o}$(K_1, signed UE_{Assert})를 포함할 수 있다. 1058에서, RP(1004)는 $EK_{r,o}$(K_1, signed UE_{Assert})를 복호화할 수 있고, UE_{Assert} 및 K_1을 획득할 수 있다. RP(1004)는 OP(1008)와 공유되는 연관 비밀을 사용하여 주장 메시지 UE_{Assert}의 서명을 검증할 수 있다. 주장 메시지 UE_{Assert}를 검증한 후에, RP(1004)는 UE(1002)에 대한 허가 정보를 발생할 수 있다. 예를 들어, RP(1004)는 허가 정보 UE_{Author}를 발생하고 K_1을 사용하여 UE_{Author}를 복호화할 수 있고, 이 복호화된 것은, 예를 들어, EK_1(UE_{Author})라고 할 수 있다. 1060에서, RP(1004)는 K_1으로 암호화되어 있는, 이 메시지에 포함되어 있는 응용 프로그램 관련 허가 정보에 관하여 UE(1002)에 통지할 수 있다. 1062에서, UE(1002)는 공유 키 K_1을 사용하여 EK_1(UE_{Author})를 복호화하고 이어서 요청된 서비스에 액세스할 수 있다.

[0102] 도 10에서, 허가 정보 또는 파라미터 UE_{Author}는 응용 프로그램에 고유한 것이고 및/또는 OP(1008)에 고유한 것일 수 있다. UE_{Author}가 OP(1008)에 고유한 것인 경우, UE_{Author}는 K_0에 의해 서명될 수 있다. 허가 정보 또는 파라미터 UE_{Author}가 응용 프로그램에 고유한 것인 경우, UE_{Author}는 $K_{r,o}$ 또는 서명 키 S에 의해 서명될 수 있다. 전송은 서명 키 S로 동작할 수 있다.

[0103] 예시적인 실시예에서, 본 명세서에 기술되어 있는 바와 같은 분할 단말 시나리오를 사용할 시에 도 10에 예시되어 있는 프로토콜 흐름이 구현될 수 있다.

[0104] 다른 예시적인 실시예에서, RP(1004) 인증은 OP(1008)와 RP(1004) 사이의 신청-응답 단계에 포함될 수 있고, 여기서 OP(1008)는 신청을 신선성의 증거와 함께 (예컨대, 넌스를 통해) RP(1004)로 송신할 수 있다. RP(1004)는 사전 설정된 공유 비밀 $K_{r,o}$를 사용하여 이 넌스에 서명하고 대답을 OP(1008)로 반환할 수 있다. 인증 신청에 대한 응답은 OP(1008) 인증 신청에 대한 직접적인 응답일 수 있거나, UE(1002)를 OP(1008)로 보내는 리디렉션 메시지에 통합되어 있을 수 있다. 어느 경우든지, OP(1008)는 (예컨대, UE 인증에 관여하기 전에) RP(1004)의 인증에 관한 신뢰할 만한 증거를 가질 수 있다. 이것은 RP(1004) 인증 실패의 경우에 OP(1008)가 프로토콜을 중단시킬 수 있게 해줄 수 있고, 이러한 RP(1004) 인증 실패의 경우에 UE(1002)와 OP(1008) 간의 통신 노력을 절감할 수 있다. 예를 들어, 1032에 예시되어 있는 바와 같이, OP(1008)는 RP(1004) 인증 실패에 관한 정보를

UE(1002)로 직접 전달할 수 있다.

[0105] 본 명세서에 기술된 바와 같이, RP(1004) 인증을 위해 연관이 사용될 수 있다. 예를 들어, RP(1004)가 OP(1008)와의 연관을 설정하는 경우, OP(1008)로부터의 신청을 포함시키기 위해 대응하는 단계들이 수정될 수 있다. 연관 설정 동안, OP(1008) 및 RP(1004)는 인증 주장 메시지에 서명하는 데 사용될 수 있는 MAC 키를 설정할 수 있다. 이 키는, 예를 들어, DH(Diffie-Hellman) 키를 사용하여 OP(1008)와 RP(1004) 사이에서 협상될 수 있는 임시 비밀 키를 사용하여 암호화된 채로 송신될 수 있다. 임시 비밀 키에 부가하여, OP(1008)는, RP(1004)에 응답하여, 예를 들어, 역시 DH 키로 암호화될 수 있는 넌스를 포함할 수 있다.

[0106] RP(1004)는 협상된 DH 키에 기초하여 넌스 및/또는 MAC 키를 복호화할 수 있다. RP(1004)는 그 자신의 사전 설정된 공유 키 $K_{r,o}$를 사용하여 OP(1008)로부터 수신되는 넌스에 서명하거나 그를 암호화할 수 있고 이를 부가의 파라미터로서 UE(1002)로 송신되는 리디렉션 메시지에 부가할 수 있다. UE(1002)가 OP(1008)로의 리디렉션을 따르기 때문에, OP(1008)는 서명되거나 암호화된 넌스를 수신할 수 있고, 공유 키 $K_{r,o}$를 사용하여 RP(1004)를 인증할 수 있다. 인증 실패의 경우에, OP(1008)는 UE(1002)를 인증되지 않은 RP로부터 보호하기 위해 경보 메시지를 UE(302)로 송신할 수 있다. RP(1004) 인증 성공의 경우에, OP(1002)는 프로토콜을 계속할 수 있다.

[0107] RP(1004) 인증에 대해 발견 모드를 사용하는 예시적인 실시예가 기술되어 있다. 예를 들어, OP(1008)와 RP(1004) 사이에 연관이 설정되지 않은 경우에[즉, OpenID에서의 무상태 모드], OP(1008)는 정보를 RP(1004)로 송신할 수 있다. 무상태 모드에서, 발견 동안 OP(1008)와 RP(1004) 사이의 정보 교환이 행해질 수 있다. 그렇지만, 예를 들어, 발견 위임의 경우에서와 같이, 발견은 OP(1008)을 수반하거나 그렇지 않을 수 있다. 발견 위임에서, 사용자 식별자는, 예를 들어, http://myblog.blog.com에 있을 수 있고, 그러면 OP(1008)에 있는 OP 종단점 URL(예컨대, http://myblog.myopenid.com에 있음)을 가리킬 수 있다. 따라서, (예컨대, myopenid.com에 있는) OP(1008)가 발견에 직접 관여되지 않을 수 있고 이 스테이지에서 RP(1004)를 인증할 수 없을지도 모른다. 예를 들어, 도 10에 예시되어 있는 바와 같이, 1018, 1030에서의 인증을 결정하는 단계 대신에, 1016, 1018에서의 연관 동안 OP(1008)는 RP(1004)를 인증할 수 있다.

[0108] OP(1008)가 발견 단계 동안 RP(1004)에 부가의 정보를 제공할 수 있는 경우[예컨대, 사용자 식별자 페이지가 OP(1008) 자체에서 호스팅되는 경우], OP(1008)는 발견 정보 페이지의 일부로서 넌스를 동적으로 발생할 수 있고 이를 HTTP 요청 RP(1004)의 식별자(예컨대, URL 또는 이메일 주소)와 연관시킬 수 있다. 그러면 OP(1008)는 RP(1004)가 이 넌스에 서명하거나 그를 암호화하고 정보를 리디렉션 메시지에 포함시킬 것으로 예상할 수 있다.

[0109] 본 명세서에 예시되어 있는 바와 같이, OP(1008)는 OP(1008)와 UE(1002) 사이의 통신을 보호할 수 있다. 예를 들어, 1026에 예시되어 있는 바와 같이, OP(1008)는 강제로 HTTPS를 사용할 수 있다[즉, UE(1002)와 OP(1008) 사이의 임의의 차후의 통신이 보호될 수 있도록, UE(1002)가 OP(1008)에 의해 HTTPS의 사용으로 리디렉션될 수 있다]. 예를 들어, TLS가 사용될 수 있다. UE(1002)로 하여금 강제로 OP(1008)의 인증서를 자동으로 임포트하게 하거나 사전 설치된 OP 인증서를 사용하게 함으로써 TLS가 동작할 수 있다. 둘 다가, 예를 들어, BA에 의해 (예컨대, 루트 CA에 의해 서명되어 있는) 루트 인증서와 대조될 수 있다. 이러한 보호에 의해, 이는 예를 들어, 1040에서 OP(1008)로부터 UE(1002)로 인증 신청 메시지에 대한 MitM 공격을 방지할 수 있게 될 수 있다. 또한, RP(1004) 인증 실패의 경우에, 이는 OP(1008)가 경보 메시지를 보호된 방식으로 UE(1002)로 송신할 수 있게 해줄 수 있다.

[0110] 본 명세서에 기술되어 있는 실시예는 로컬 주장 제공자에서 구현될 수 있다. OpenID에서의 RP 인증을 조정하고 로컬 주장 제공자를 사용하는 예시적인 프로토콜이 본 명세서에 기술되어 있다. 기술된 실시예는, RP와 OP 사이의 사전 설정된 공유 비밀 $K_{r,o}$에 기초하여, RP와 (네트워크측) OP 사이의 접촉(예컨대, 제1 접촉)이 있을 때 RP의 인증을 가능하게 해줄 수 있다. OpenID의 연관 모드에서, 이것은 연관 관계이다.

[0111] 도 11은 로컬 주장 제공자에서의 프로비저닝 단계의 메시지 흐름도의 예시적인 실시예를 나타낸 것이다. 도 11에 예시된 바와 같이, 로컬 주장 제공자에 의한 프로비저닝 단계에서 수행되는 통신에서 UE(1102), RP(1104), OP(1106), 및/또는 HSS(1108)가 구현될 수 있다. 프로비저닝 단계에서의 다양한 스테이지에서, 재생 보호를 위해 넌스가 구현될 수 있다.

[0112] 도 11에 도시된 바와 같이, 1110에서, UE(1102)는 로그인 식별자(예컨대, OID)를 RP(1104)로 전송할 수 있다. 1112에서, RP(1104)는 연관 요청(예컨대, http POST OpenID 연관 요청)을 OP(1106)로 송신할 수 있다. 연관 요청은 RP(1104)와 OP(1106) 사이에서 공유되는 공유 키 $K_{r,o}$로 암호화되어 있을 수 있는 RP(1104) 자격 증명

RP_{Cred}를 포함할 수 있다. 이 암호화된 RP_{Cred}는, 예를 들어, $EK_{r,o}(RP_{Cred})$라고 할 수 있다. RP 자격 증명 RP_{Cred}는 사전 공유 비밀 또는 식별자를 포함할 수 있는 일반 유형의 자격 증명일 수 있다. 1114에서, OP(1106)는 공유 키 K_0가 존재하는지를 결정할 수 있다. 공유 키 K_0가 존재하는 경우, OP(1106)는 인증 단계(AP)를 계속할 수 있다. 공유 키 K_0가 존재하지 않는 경우, OP(1106)는 프로비저닝 단계를 계속할 수 있다. 예를 들어, OP(1106)는 단계(1116)로 진행할 수 있다.

[0113] 1116에서, OP(1106)는 RP(1104)와의 연관을 수행할 수 있다. 예를 들어, OP(1106)는 연관 핸들 A 및/또는 서명 키 S를 발생할 수 있다. 연관 핸들 A의 함수로부터 서명 키 S가 발생될 수 있다. OP(1106)는 키 $K_{r,o}$로 서명 키 S를 암호화할 수 있고, 이 암호화된 것은, 예를 들어, $EK_{r,o}(S)$라고 할 수 있다. OP(1106)는 연관 핸들 A 및/또는 암호화된 서명 키 S를 RP(1104)로 송신할 수 있다. 1118에서, RP(1104)는 UE(1102)를 OP(1106)로 리디렉션시키는 메시지(예컨대, 리디렉션 메시지)를 UE(1102)로 송신할 수 있다. 1118에서의 메시지는 sessionID, returnURL, 넌스, 로그인 식별자(예컨대, OID), 및/또는 연관 핸들 A 등의 파라미터를 포함할 수 있다. 1120에서, UE(1102)는 RP(1104)로부터 수신되는 파라미터들 중 하나 이상을 포함하는 메시지(예컨대, http GET 요청)를 OP(1106)로 송신할 수 있다. 예를 들어, 1120에서의 메시지는 sessionID, returnURL, 넌스, 로그인 식별자(예컨대, OID), 및/또는 연관 핸들을 포함할 수 있다.

[0114] 1122에서, OP(1106)는 SIP 다이제스트 인증 벡터(SIP digest authentication vector, SD-AV) 및/또는 기타 정보를 HSS(1108)로부터 획득할 수 있다. 1124에서, OP(1106)는 인증 신청을 UE(1102)로 송신할 수 있다. 1126에서, UE(1102)는 공유 키 K_0를 발생할 수 있다. 1126에서, UE(1102)는 또한 인증 응답을 발생하고 및/또는 인증 응답을 OP(1106)로 송신할 수 있다. 예를 들어, 인증 응답은 사전-프로비저닝된 사용자 자격 증명(예컨대, 사용자 이름 및 비밀 번호)을 사용하여 UE(1102)에 의해 계산될 수 있다. 1128에서, OP(1106)는, 수신된 응답을, 예를 들어, 인증 벡터(SD-AV)로부터 계산되는 예상된 응답과 비교하는 등에 의해, 인증 응답을 유효성 확인할 수 있다. 사용자/UE(1102)가 OP(1106)에서 인증되었으면, OP(1106)는 UE(1102)와 OP(1106) 사이에서 공유될 수 있는 공유 비밀 K_0를 발생할 수 있다. K_0에 의한 암호화는 정당한 인증된 UE(1102)가 서비스에 대해 나중에 사용하기 위한 서비스 액세스 토큰일 수 있는 UE_{Author}를 획득하도록 보장해줄 수 있다. 예시적인 실시예에서, K_0는 난수일 수 있고, 암호 함수를 사용하여 발생될 수 있다.

[0115] 1130에서, OP(1106)는 사용자/UE(1102)의 성공적인 인증을 나타내는 인증 주장 메시지 UE_{Assert}에 서명할 수 있다. 예를 들어, OP(1106)는 서명 키 S를 사용하여 UE_{Assert}에 서명할 수 있다. 서명된 UE_{Assert}는 $Sig_S(UE_{Assert})$라고 할 수 있다. OP(1106)는 연관 핸들 A, 서명된 주장 UE_{Assert}, 및/또는 허가 메시지 UE_{Author}를 UE(1102)로 송신할 수 있다. 서명된 주장 UE_{Assert}는 서명 키 S로 암호화될 수 있고, 이 암호화된 것은, 예를 들어, $E_S(Sig_S(UE_{Assert}))$라고 할 수 있다. 허가 메시지 UE_{Author}는 서명된 키 K_0로 암호화될 수 있고, 이 암호화된 것은, 예를 들어, $EK_0(UE_{Author})$라고 할 수 있다. 예시적인 실시예에서, 인증 주장 메시지 UE_{Assert}를 암호화하는 것 및 그에 서명하는 것 둘 다를 하는 것 대신에, 서명 키 S를 사용하여 단순히 인증 주장 메시지에 서명하는 것으로 충분할 수 있다. 1132에서, 연관 핸들 A, UE_{Assert}, 및/또는 UE_{Author}는 UE(1102)를 RP(1104)로 리디렉션시킬 수 있는 리디렉션 메시지에서 송신될 수 있다. 1134에서, UE(1102)는 연관 핸들, $E_S(Sig_S(UE_{Assert}))$, 및/또는 $EK_0(UE_{Author})$를 포함할 수 있는 메시지(예컨대, http GET 요청)를 RP(1104)로 송신할 수 있다. 1136에서, RP(1104)는 서명 키 S를 복호화하고, 서명된 주장 $Sig_S(UE_{Assert})$를 복호화하며, S를 사용하여 주장(예컨대, OpenID 주장)을 검증하고, 및/또는 암호화된 허가 메시지 $EK_0(UE_{Author})$를 복호화할 수 있다. 1138에서, RP(1104)는 $EK_0(UE_{Author})$를 포함하는 통지를 UE(1102)로 송신할 수 있다. $EK_0(UE_{Author})$는 RP(1104)가 적절한 RP로서 인증되었고 불법 RP 또는 기타 MitM이 아니라는 것을 UE(1102)에 알려줄 수 있는데, 그 이유는 통지가 불법 RP 또는 기타 MitM이 복호화할 수 없을 것인 $EK_0(UE_{Author})$를 포함할 수 있기 때문이다.

[0116] 도 11에 예시되어 있는 RP 인증은 본 명세서에 기술되어 있는 다른 실시예에서 이와 유사하게 구현될 수 있다. 예를 들어, 도 11에 예시되어 있는 인증 구현예는 도 2에 예시되어 있는 인증 단계에서 이와 유사하게 구현될 수 있다.

[0117] 도 12는 본 명세서에 기술되어 있는 실시예에 따른, 로컬 주장 제공자(local assertion provider)에서의 예시적

인 인증 단계의 메시지 흐름도를 나타낸 것이다. 도 12에 예시된 바와 같이, 인증 단계는 UE(1202), RP(1204), OP(1206), 및/또는 HSS(1208) 사이의 통신을 포함할 수 있다. 예시적인 실시예에서, UE(1202)는 로컬 인증 및 인증 주장(예컨대, OpenID 인증 주장)의 서명을 수행하는 로컬 OP 기능 OP_{loc}를 포함할 수 있는 반면, OP(1206)는, 예를 들어, 네트워크에 위치해 있을 수 있는 외부 OP일 수 있다. 1210에서, UE(1202)는 로그인 식별자(예컨대, OID)를 RP(1204)로 송신할 수 있다. 1212에서, RP(1204)는 연관 요청 메시지(예컨대, http POST OpenID 연관 요청)을 OP(1206)로 송신할 수 있다. 연관 요청 메시지는 RP(1204)에 대응하는 RP 자격 증명 RP_{Cred}를 포함할 수 있다. RP_{Cred}는 RP(1204)와 OP(1206) 사이에서 공유되는 공유 키 $K_{r,o}$로 암호화될 수 있다.

[0118] 1214에서, OP(1206)는 UE(1202)와 OP(1206) 사이의 보안 통신을 위해 이들 엔터티 간에 공유되는 공유 키 K_0가 프로비저닝되었는지를 결정할 수 있다. 공유 키 K_o가 프로비저닝되어 있지 않은 경우, 프로토콜은 공유 키 K_0를 프로비저닝하기 위해 프로비저닝 단계를 계속할 수 있다. 공유 키 K_0가 프로비저닝되어 있는 경우, 프로토콜은 인증 단계를 계속할 수 있다. 예시적인 실시예에서, OP(1206)는 공유 키 K_0가 프로비저닝되어 있는지를 결정하지 않을 수 있고, 프로토콜 흐름은 이러한 결정 없이 계속될 수 있다.

[0119] 1216에서, OP(1206)는 RP(1204)와의 연관을 수행할 수 있다. 예를 들어, OP(1206)는 연관 핸들 A 및/또는 공유 키 K_1을 발생할 수 있다. 공유 키 K_1은, 예를 들어, 공유 키 K_0 및 연관 핸들 A의 함수로부터 도출될 수 있다. 공유 키 K_1은 공유 키 $K_{r,o}$로 암호화될 수 있고, 이 암호화된 것은, 예를 들어, $EK_{r,o}(K_1)$이라고 할 수 있다. 연관 핸들 A 및 암호화된 키 K_1은 RP(1204)로 송신될 수 있다. RP(1204)는 sessionID, returnURL, 넌스, 로그인 식별자(예컨대, OID), 및/또는 연관 핸들 A 등의 파라미터를 포함하는 1218에서의 메시지를 UE(1202)로 송신할 수 있다. 1218에서의 메시지는 인증을 위해 그 UE(1202)를 UE(1202) 상의 OP_{loc}(도시 생략)으로 리디렉션하는 리디렉션 메시지일 수 있다. 1220에서, UE(1202)는 로컬 인증을 수행할 수 있다. 1220에서, UE(1202)는 공유 키 K_0 및 연관 핸들 A의 함수를 사용하여 공유 키 K_1을 발생할 수 있다. K_0에 의한 암호화는 정당한 인증된 UE(1202)가 서비스에 대해 나중에 사용하기 위한 서비스 액세스 토큰일 수 있는 UE_{Author}를 획득하도록 보장해줄 수 있다. UE(1202)는 공유 키 K_1로 인증 주장 메시지 UE_{Assert}에 서명할 수 있고, 이 서명된 것은 $SigK_1(UE_{Assert})$라고 할 수 있다. UE(1202)는 [예컨대, UE(1202) 상의 로컬 OP를 사용하여] 허가 정보 또는 파라미터 UE_{Author}를 발생할 수 있다. UE(1202)는 공유 키 K_0로 UE_{Author}를 암호화할 수 있고, 이 암호화된 것은, 예를 들어, $EK_0(UE_{Author})$라고 할 수 있다. UE(1202)는 공유 키 K_1로 $SigK_1(UE_{Assert})$ 및/또는 $EK_0(UE_{Author})$를 암호화할 수 있고 [이 암호화된 것은 $EK_1(SigK_1(UE_{Assert}), EK_0(UE_{Author}))$라고 할 수 있음], 연관 핸들 A 및 $EK_1(SigK_1(UE_{Assert}), EK_0(UE_{Author}))$를 RP(1204)로 송신할 수 있다. 1222에 예시되어 있는 바와 같이, UE(1202)는 서명된 주장 UE_{Assert}와 함께 메시지(예컨대, http GET 요청)를 RP(1204)로 송신할 수 있다.

[0120] 1224에서, RP(1204)는 공유 키 $K_{r,o}$를 사용하여 K_1을 복호화할 수 있다. RP(1204)는 $SigK_1(UE_{Assert})$를 복호화할 수 있고 K_1을 사용하여 인증 주장 메시지 UE_{Assert}를 검증할 수 있다. 1224에서, RP(1204)는 K_1을 사용하여 $EK_0(UE_{Author})$를 복호화할 수 있다. RP(1204)는 UE_{Author}를 복호화할 수 없을지도 모르는데, 그 이유는 UE_{Author}가 UE(1202)와 OP(1206) 사이에서 공유되는 공유 키 K_0에 의해 암호화되어 있을 수 있기 때문이다. 1226에서, RP(1204)는 RP(1204)가 K_1을 사용하여 보안 채널을 설정한 적절한 RP이고 불법 RP 또는 다른 MitM이 아니라는 것을 나타내는 통지를 UE(1202)로 송신할 수 있는데, 그 이유는 이 통지가 불법 RP 또는 다른 MitM이 복호화할 수 없을 것인 정보 $EK_0(UE_{Author})$를 포함할 수 있기 때문이다.

[0121] 도 13a 내지 도 13e는 본 명세서에 기술되어 있는 실시예를 수행할 시에 구현될 수 있는 예시적인 네트워크 시스템 및 디바이스를 나타낸 것이다. 도 13a는 하나 이상의 개시된 실시예가 구현될 수 있는 예시적인 통신 시스템(1300)의 도면이다. 통신 시스템(1300)은 음성, 데이터, 비디오, 메시징, 방송 등과 같은 콘텐츠를 다수의 무선 사용자에게 제공하는 다중 접속 시스템일 수 있다. 통신 시스템(1300)은 다수의 무선 사용자가 시스템 자원(무선 대역폭을 포함함)의 공유를 통해 이러한 콘텐츠에 액세스할 수 있게 해줄 수 있다. 예를 들어, 통신 시스템(1300)은 CDMA(code division multiple access, 코드 분할 다중 접속), TDMA(time division multiple access, 시분할 다중 접속), FDMA(frequency division multiple access, 주파수 분할 다중 접속),

OFDMA(orthogonal FDMA, 직교 FDMA), SC-FDMA(single-carrier FDMA, 단일 반송파 FDMA) 및/또는 기타와 같은 하나 이상의 채널 접속 방법을 이용할 수 있다.

[0122] 도 13a에 도시된 바와 같이, 통신 시스템(1300)은 WTRU(wireless transmit/receive unit, 무선 송수신 유닛)(1302a, 1302b, 1302c, 1302d), RAN(radio access network, 무선 액세스 네트워크)(1304), 코어 네트워크(1306), PSTN(public switched telephone network, 공중 교환 전화망)(1308), 인터넷(1310), 및 기타 네트워크(1312)를 포함할 수 있지만, 개시된 실시예가 임의의 수의 WTRU, 기지국, 네트워크 및/또는 네트워크 요소를 생각하고 있다는 것을 잘 알 것이다. WTRU(1302a, 1302b, 1302c, 1302d) 각각은 무선 환경에서 동작하고 및/또는 통신하도록 구성되어 있는 임의의 유형의 디바이스일 수 있다. 일례로서, WTRU(1302a, 1302b, 1302c, 1302d)는 무선 신호를 전송 및/또는 수신하도록 구성될 수 있고, UE(user equipment), 이동국, 고정형 또는 이동형 가입자 유닛, 페이저, 휴대폰, PDA(personal digital assistant), 스마트폰, 랩톱, 넷북, 개인용 컴퓨터, 무선 센서, 가전 제품 등을 포함할 수 있다.

[0123] 통신 시스템(1300)은 또한 기지국(1314a) 및 기지국(1314b)을 포함할 수 있다. 기지국(1314a, 1314b) 각각은 하나 이상의 통신 네트워크 - 코어 네트워크(1306), 인터넷(1310) 및/또는 네트워크(1312) 등 - 에의 액세스를 용이하게 해주기 위해 WTRU(1302a, 1302b, 1302c, 1302d) 중 적어도 하나와 무선으로 인터페이스하도록 구성되어 있는 임의의 유형의 디바이스일 수 있다. 일례로서, 기지국(1314a, 1314b)은 BTS(base transceiver station, 기지국 송수신기), 노드-B, eNode B, 홈 노드 B, 사이트 제어기, AP(access point), 무선 라우터 및/또는 기타일 수 있다. 기지국(1314a, 1314b) 각각이 단일 요소로서 나타내어져 있지만, 기지국(1314a, 1314b)이 임의의 수의 상호연결된 기지국 및/또는 네트워크 요소를 포함할 수 있다는 것을 잘 알 것이다.

[0124] 기지국(1314a)은 다른 기지국 및/또는 네트워크 요소 - BSC(base station controller, 기지국 제어기), RNC(radio network controller, 무선 네트워크 제어기), 중계 노드, 기타 등등 - (도시 생략)도 포함할 수 있는 RAN(1304)의 일부일 수 있다. 기지국(1314a) 및/또는 기지국(1314b)은 특정의 지리적 지역 - 셀(도시 생략)이라고 할 수 있음 - 내에서 무선 신호를 전송 및/또는 수신하도록 구성될 수 있다. 셀은 여러 셀 섹터(cell sector)로 추가로 나누어질 수 있다. 예를 들어, 기지국(1314a)과 연관된 셀이 3개의 섹터로 나누어질 수 있다. 따라서, 일 실시예에서 기지국(1314a)은 3개의 송수신기(즉, 셀의 각각의 섹터마다 하나씩)를 포함할 수 있다. 다른 실시예에서, 기지국(1314a)은 MIMO(multiple-input multiple output, 다중 입력 다중 출력) 기술을 이용할 수 있고, 따라서, 셀의 각각의 섹터에 대해 다수의 송수신기를 이용할 수 있다.

[0125] 기지국(1314a, 1314b)은 임의의 적당한 무선 통신 링크[예컨대, RF(radio frequency, 무선 주파수), 마이크로파, IR(infrared, 적외선), UV(ultraviolet, 자외선), 가시광 등]일 수 있는 공중 인터페이스(1316)를 통해 WTRU(1302a, 1302b, 1302c, 1302d) 중 하나 이상과 통신할 수 있다. 임의의 적당한 RAT(radio access technology, 무선 액세스 기술)를 사용하여 공중 인터페이스(1316)가 설정될 수 있다.

[0126] 보다 구체적으로는, 앞서 살펴본 바와 같이, 통신 시스템(1300)은 다중 접속 시스템일 수 있고, CDMA, TDMA, FDMA, OFDMA, SC-FDMA 등과 같은 하나 이상의 채널 접속 방식을 이용할 수 있다. 예를 들어, RAN(1304) 내의 기지국(1314a) 및 WTRU(1302a, 1302b, 1302c)는 WCDMA(wideband CDMA, 광대역 CDMA)를 사용하여 공중 인터페이스(1316)를 설정할 수 있는 UTRA[UMTS(Universal Mobile Telecommunications System) Terrestrial Radio Access]와 같은 무선 기술을 구현할 수 있다. WCDMA는 HSPA(High-Speed Packet Access, 고속 패킷 액세스) 및/또는 HSPA+(Evolved HSPA)와 같은 통신 프로토콜을 포함할 수 있다. HSPA는 HSDPA(High-Speed Downlink Packet Access, 고속 하향링크 패킷 액세스) 및/또는 HSUPA(High-Speed Uplink Packet Access, 고속 상향링크 패킷 액세스)를 포함할 수 있다.

[0127] 다른 실시예에서, 기지국(1314a) 및 WTRU(1302a, 1302b, 1302c)는 LTE(Long Term Evolution) 및/또는 LTE-A(LTE-Advanced)를 사용하여 공중 인터페이스(1316)를 설정할 수 있는 E-UTRA(Evolved UMTS Terrestrial Radio Access)와 같은 무선 기술을 구현할 수 있다.

[0128] 다른 실시예에서, 기지국(1314a) 및 WTRU(1302a, 1302b, 1302c)는 IEEE 802.16[즉, WiMAX(Worldwide Interoperability for Microwave Access)], CDMA2000, CDMA2000 1X, CDMA2000 EV-DO, IS-2000(Interim Standard 2000), IS-95(Interim Standard 95), IS-856(Interim Standard 856), GSM(Global System for Mobile communications), EDGE(Enhanced Data rates for GSM Evolution), GSM EDGE(GERAN) 및/또는 기타와 같은 무선 기술을 구현할 수 있다.

[0129] 도 13a의 기지국(1314b)은, 예를 들어, 무선 라우터, 홈 노드 B, 홈 eNode B, 또는 액세스 포인트(access

point)일 수 있고, 사업장, 가정, 차량, 캠퍼스 등과 같은 국소화된 지역에서의 무선 연결을 용이하게 해주는 임의의 적당한 RAT를 이용할 수 있다. 일 실시예에서, 기지국(1314b) 및 WTRU(1302c, 1302d)는 WLAN(wireless local area network, 무선 근거리 통신망)을 설정하기 위해 IEEE 802.11과 같은 무선 기술을 구현할 수 있다. 다른 실시예에서, 기지국(1314b) 및 WTRU(1302c, 1302d)는 WPAN(wireless personal area network, 무선 개인 영역 네트워크)을 설정하기 위해 IEEE 802.15와 같은 무선 기술을 구현할 수 있다. 또 다른 실시예에서, 기지국(1314b) 및 WTRU(1302c, 1302d)는 피코셀(picocell) 또는 펨토셀(femtocell)을 설정하기 위해 셀룰러-기반 RAT(예컨대, WCDMA, CDMA2000, GSM, LTE, LTE-A 등)를 이용할 수 있다. 도 13a에 도시된 바와 같이, 기지국(1314b)은 인터넷(1310)에의 직접 연결을 가질 수 있다. 따라서, 기지국(1314b)은 코어 네트워크(1306)를 통해 인터넷(1310)에 액세스할 필요가 없을 수 있다.

[0130] RAN(1304)은 음성, 데이터, 응용 프로그램, 및 VoIP(voice over internet protocol) 서비스를 WTRU(1302a, 1302b, 1302c, 1302d) 중 하나 이상의 WTRU에 제공하도록 구성되어 있는 임의의 유형의 네트워크일 수 있는 코어 네트워크(1306)와 통신하고 있을 수 있다. 예를 들어, 코어 네트워크(1306)는 호출 제어, 대금 청구 서비스, 모바일 위치-기반 서비스, 선불 전화(pre-paid calling), 인터넷 연결, 비디오 배포 등을 제공하고 및/또는 사용자 인증과 같은 고수준 보안 기능을 수행할 수 있다. 도 13a에 도시되어 있지는 않지만, RAN(1304) 및/또는 코어 네트워크(1306)가 RAN(1304)과 동일한 RAT 또는 상이한 RAT를 이용하는 다른 RAN과 직접 또는 간접 통신을 하고 있을 수 있다는 것을 잘 알 것이다. 예를 들어, E-UTRA 무선 기술을 이용하고 있을 수 있는 RAN(1304)에 연결되는 것에 부가하여, 코어 네트워크(1306)는 또한 GSM 무선 기술을 이용하는 다른 RAN(도시 생략)과 통신하고 있을 수 있다.

[0131] 코어 네트워크(1306)는 또한 WTRU(1302a, 1302b, 1302c, 1302d)가 PSTN(1308), 인터넷(1310) 및/또는 기타 네트워크(1312)에 액세스하기 위한 게이트웨이로서 역할할 수 있다. PSTN(1308)은 POTS(plain old telephone service)를 제공하는 회선-교환 전화 네트워크를 포함할 수 있다. 인터넷(1310)은 TCP/IP 인터넷 프로토콜군 내의 TCP(transmission control protocol, 전송 제어 프로토콜), UDP(user datagram protocol, 사용자 데이터그램 프로토콜) 및 IP(internet protocol, 인터넷 프로토콜)와 같은 공통의 통신 프로토콜을 사용하는 상호연결된 컴퓨터 네트워크 및 디바이스의 전세계 시스템을 포함할 수 있다. 네트워크(1312)는 다른 서비스 공급자가 소유하고 및/또는 운영하는 유선 또는 무선 통신 네트워크를 포함할 수 있다. 예를 들어, 네트워크(1312)는 RAN(1304)과 동일한 RAT 또는 상이한 RAT를 이용할 수 있는 하나 이상의 RAN에 연결된 다른 코어 네트워크를 포함할 수 있다.

[0132] 통신 시스템(1300) 내의 WTRU(1302a, 1302b, 1302c, 1302d) 중 일부 또는 전부는 다중-모드 기능을 포함할 수 있다 - 즉, WTRU(1302a, 1302b, 1302c, 1302d)가 상이한 무선 링크를 통해 상이한 무선 네트워크와 통신하기 위한 다수의 송수신기를 포함할 수 있다 -. 예를 들어, 도 13a에 도시된 WTRU(1302c)는 셀룰러-기반 무선 기술을 이용할 수 있는 기지국(1314a)과 통신하도록, 그리고 IEEE 802 무선 기술을 이용할 수 있는 기지국(1314b)과 통신하도록 구성될 수 있다.

[0133] 도 13b는 예시적인 WTRU(1302)의 시스템도이다. 도 13b에 도시된 바와 같이, WTRU(1302)는 프로세서(1318), 송수신기(1320), 송신/수신 요소(1322), 스피커/마이크(1324), 키패드(1326), 디스플레이/터치패드(1328), 비이동식 메모리(1330), 이동식 메모리(1332), 전원 공급 장치(1334), GPS(global positioning system) 칩셋(1336), 및 기타 주변 장치(1338)를 포함할 수 있다. 실시예와 부합한 채로 있으면서 WTRU(1302)가 상기한 요소들의 임의의 서브컴비네이션을 포함할 수 있다는 것을 잘 알 것이다.

[0134] 프로세서(1318)가 범용 프로세서, 전용 프로세서, 종래의 프로세서, DSP(digital signal processor), 복수의 마이크로프로세서, DSP 코어와 연관된 하나 이상의 마이크로프로세서, 제어기, 마이크로제어기, ASIC(Application Specific Integrated Circuit), FPGA(Field Programmable Gate Array) 회로, 임의의 다른 유형의 IC(integrated circuit), 상태 기계 및/또는 기타일 수 있다. 프로세서(1318)는 WTRU(1302)가 무선 환경에서 동작할 수 있게 해주는 신호 코딩, 데이터 처리, 전력 제어, 입력/출력 처리, 및/또는 임의의 다른 기능을 수행할 수 있다. 프로세서(1318)는 송신/수신 요소(1322)에 결합되어 있을 수 있는 송수신기(1320)에 결합될 수 있다. 도 13b가 프로세서(1318) 및 송수신기(1320)를 개별 구성요소로서 나타내고 있지만, 프로세서(1318) 및 송수신기(1320)가 전자 패키지 또는 칩에 함께 통합되어 있을 수 있다는 것을 잘 알 것이다.

[0135] 송신/수신 요소(1322)는 공중 인터페이스(1316)를 통해 기지국[예컨대, 기지국(1314a)]으로 신호를 전송하거나 기지국으로부터 신호를 수신하도록 구성될 수 있다. 예를 들어, 일 실시예에서, 송신/수신 요소(1322)는 RF 신호를 전송 및/또는 수신하도록 구성되어 있는 안테나일 수 있다. 다른 실시예에서, 송신/수신 요소(1322)는,

예를 들어, IR, UV 또는 가시광 신호를 전송 및/또는 수신하도록 구성되어 있는 방출기/검출기일 수 있다. 또 다른 실시예에서, 송신/수신 요소(1322)는 RF 신호 및 광 신호 둘 다를 전송 및 수신하도록 구성될 수 있다. 송신/수신 요소(1322)가 무선 신호의 임의의 조합을 전송 및/또는 수신하도록 구성될 수 있다는 것을 잘 알 것이다.

[0136] 그에 부가하여, 송신/수신 요소(1322)가 도 13b에 단일 요소로서 나타내어져 있지만, WTRU(1302)는 임의의 수의 송신/수신 요소(1322)를 포함할 수 있다. 보다 구체적으로는, WTRU(1302)는 MIMO 기술을 이용할 수 있다. 따라서, 일 실시예에서, WTRU(1302)는 공중 인터페이스(1316)를 통해 무선 신호를 전송 및 수신하기 위한 2개 이상의 송신/수신 요소(1322)(예컨대, 다수의 안테나)를 포함할 수 있다.

[0137] 송수신기(1320)는 송신/수신 요소(1322)에 의해 전송되어야 하는 신호를 변조하고 송신/수신 요소(1322)에 의해 수신되는 신호를 복조하도록 구성될 수 있다. 앞서 살펴본 바와 같이, WTRU(1302)는 다중-모드 기능을 가질 수 있다. 따라서, 송수신기(1320)는 WTRU(1302)가, 예를 들어, UTRA 및 IEEE 802.11과 같은 다수의 RAT를 통해 통신할 수 있게 해주는 다수의 송수신기를 포함할 수 있다.

[0138] WTRU(1302)의 프로세서(1318)는 스피커/마이크(1324), 키패드(1326), 및/또는 디스플레이/터치패드(1328)[예컨대, LCD(liquid crystal display, 액정 디스플레이) 디스플레이 유닛 또는 OLED(organic light-emitting diode, 유기 발광 다이오드) 디스플레이 유닛]에 결합될 수 있고 그로부터 사용자 입력 데이터를 수신할 수 있다. 프로세서(1318)는 또한 사용자 데이터를 스피커/마이크(1324), 키패드(1326) 및/또는 디스플레이/터치패드(1328)로 출력할 수 있다. 그에 부가하여, 프로세서(1318)는 비이동식 메모리(1330) 및/또는 이동식 메모리(1332)와 같은 임의의 유형의 적당한 메모리로부터의 정보에 액세스하고 그 메모리에 데이터를 저장할 수 있다. 비이동식 메모리(1330)는 랜덤 액세스 메모리(RAM), 판독 전용 메모리(ROM), 하드 디스크, 또는 임의의 다른 유형의 메모리 저장 디바이스를 포함할 수 있다. 이동식 메모리(1332)는 GSM SIM(Subscriber Identity Module, 가입자 식별 모듈) 카드, UICC(즉, SIM 카드의 UMTS 버전), 메모리 스틱, SD(secure digital) 메모리 카드 및/또는 기타를 포함할 수 있다. 다른 실시예에서, 프로세서(1318)는 WTRU(1302) 상에 물리적으로 위치하지 않은[예컨대, 서버 또는 가정용 컴퓨터(도시 생략) 상의] 메모리로부터의 정보에 액세스하고 그 메모리에 데이터를 저장할 수 있다.

[0139] 프로세서(1318)는 전원 공급 장치(1334)로부터 전력을 받을 수 있고, WTRU(1302) 내의 다른 구성요소로 전력을 분배하고 및/또는 전력을 제어하도록 구성될 수 있다. 전원 공급 장치(1334)는 WTRU(1302)에 전원을 제공하는 임의의 적당한 디바이스일 수 있다. 예를 들어, 전원 공급 장치(1334)는 하나 이상의 건전지[예컨대, 니켈-카드뮴(NiCd), 니켈-아연(NiZn), 니켈 수소화금속(NiMH), 리튬-이온(Li--ion) 등], 태양 전지, 연료 전지 및/또는 기타를 포함할 수 있다.

[0140] 프로세서(1318)는 또한 WTRU(1302)의 현재 위치에 관한 위치 정보(예컨대, 경도 및 위도)를 제공하도록 구성될 수 있는 GPS 칩셋(1336)에 결합될 수 있다. GPS 칩셋(1336)으로부터의 정보에 부가하여 또는 그 대신에, WTRU(1302)는 기지국[예컨대, 기지국(1314a, 1314b)] 공중 인터페이스(1316)를 통해 위치 정보를 수신하고 및/또는 2개 이상의 근방의 기지국으로부터 수신되는 신호의 타이밍에 기초하여 그의 위치를 결정할 수 있다. 실시예와 부합한 채로 있으면서 WTRU(1302)가 임의의 적당한 위치 결정 방법에 의해 위치 정보를 획득할 수 있다는 것을 잘 알 것이다.

[0141] 프로세서(1318)는 또한 부가의 특징, 기능 및/또는 유선 또는 무선 연결을 제공하는 하나 이상의 소프트웨어 및/또는 하드웨어 모듈을 포함할 수 있는 다른 주변 장치(1338)에 결합될 수 있다. 예를 들어, 주변 장치(1338)는 가속도계, 전자 나침반, 위성 송수신기, 디지털 카메라(사진 또는 비디오용), USB(universal serial bus) 포트, 진동 디바이스, 텔레비전 송수신기, 핸즈프리 헤드셋, 블루투스® 모듈, FM(frequency modulated, 주파수 변조) 라디오 유닛, 디지털 음악 플레이어, 미디어 플레이어, 비디오 게임 플레이어 모듈, 인터넷 브라우저 및/또는 기타를 포함할 수 있다.

[0142] 도 13c는 일 실시예에 따른, RAN(1304) 및 코어 네트워크(1306)의 시스템도이다. 앞서 살펴본 바와 같이, RAN(1304)은 공중 인터페이스(1316)를 통해 WTRU(1302a, 1302b, 1302c)와 통신하기 위해 UTRA 무선 기술을 이용할 수 있다. RAN(1304)은 또한 코어 네트워크(1306)와 통신하고 있을 수 있다. 도 13c에 도시된 바와 같이, RAN(1304)은 각각이 공중 인터페이스(1316)를 통해 WTRU(1302a, 1302b, 1302c)와 통신하기 위한 하나 이상의 송수신기를 포함할 수 있는 노드-B(1340a, 1340b, 1340c)를 포함할 수 있다. 노드-B(1340a, 1340b, 1340c) 각각은 RAN(1304) 내의 특정의 셀(도시 생략)과 연관되어 있을 수 있다. RAN(1304)은 또한 RNC(1342a, 1342b)도 포함할 수 있다. 실시예와 부합한 채로 있으면서 RAN(1304)이 임의의 수의 노드-B 및 RNC를 포함할 수 있다는

것을 잘 알 것이다.

[0143] 도 13c에 도시된 바와 같이, 노드-B(1340a, 1340b)는 RNC(1342a)와 통신하고 있을 수 있다. 그에 부가하여, 노드-B(1340c)는 RNC(1342b)와 통신하고 있을 수 있다. 노드-B(1340a, 1340b, 1340c)는 Iub 인터페이스를 통해 각자의 RNC(1342a, 1342b)와 통신할 수 있다. RNC(1342a, 1342b)는 Iur 인터페이스를 통해 서로 통신하고 있을 수 있다. 각각의 RNC(1342a, 1342b)는 RNC가 연결되어 있는 각자의 노드-B(1340a, 1340b, 1340c)를 제어하도록 구성되어 있을 수 있다. 그에 부가하여, 각각의 RNC(1342a, 1342b)는 외측 루프 전력 제어, 부하 제어, 허가 제어, 패킷 스케줄링, 핸드오버 제어, 매크로다이버시티(macrodiversity), 보안 기능, 데이터 암호화 등과 같은 다른 기능을 수행하거나 지원하도록 구성되어 있을 수 있다.

[0144] 도 13c에 도시된 코어 네트워크(1306)는 MGW(media gateway, 미디어 게이트웨이)(1344), MSC(mobile switching center, 이동 교환국)(1346), SGSN(serving GPRS support node, 서비스 제공 GPRS 지원 노드)(1348), 및/또는 GGSN(gateway GPRS support node, 게이트웨이 GPRS 지원 노드)(1350)을 포함할 수 있다. 상기 요소들 각각이 코어 네트워크(1306)의 일부로서 나타내어져 있지만, 이들 요소 중 임의의 것이 코어 네트워크 운영자 이외의 엔터티에 의해 소유되고 및/또는 운영될 수 있다는 것을 잘 알 것이다.

[0145] RAN(1304) 내의 RNC(1342a)는 IuCS 인터페이스를 통해 코어 네트워크(1306) 내의 MSC(1346)에 연결될 수 있다. MSC(1346)는 MGW(1344)에 연결될 수 있다. MSC(1346) 및 MGW(1344)는, WTRU(1302a, 1302b, 1302c)와 종래의 지상선(land-line) 통신 디바이스 사이의 통신을 용이하게 해주기 위해, PSTN(1308)과 같은 회선 교환 네트워크에의 액세스를 WTRU(1302a, 1302b, 1302c)에 제공할 수 있다.

[0146] RAN(1304) 내의 RNC(1342a)는 또한 IuPS 인터페이스를 통해 코어 네트워크(1306) 내의 SGSN(1348)에 연결될 수 있다. SGSN(1348)은 GGSN(1350)에 연결될 수 있다. SGSN(1348) 및 GGSN(1350)은, WTRU(1302a, 1302b, 1302c)와 IP-기반 디바이스 사이의 통신을 용이하게 해주기 위해, 인터넷(1310)과 같은 패킷 교환 네트워크에의 액세스를 WTRU(1302a, 1302b, 1302c)에 제공할 수 있다.

[0147] 앞서 살펴본 바와 같이, 코어 네트워크(1306)는 또한 다른 서비스 공급자에 의해 소유되고 및/또는 운영되는 다른 유선 또는 무선 네트워크를 포함할 수 있는 네트워크(1312)에 연결될 수 있다.

[0148] 도 13d는 일 실시예에 따른, RAN(1304) 및 코어 네트워크(1306)의 시스템도이다. 앞서 살펴본 바와 같이, RAN(1304)은 공중 인터페이스(1316)를 통해 WTRU(1302a, 1302b, 1302c)와 통신하기 위해 E-UTRA 무선 기술을 이용할 수 있다. RAN(1304)은 또한 코어 네트워크(1306)와 통신하고 있을 수 있다.

[0149] RAN(1304)은 eNode B(1340a, 1340b, 1340c)를 포함할 수 있지만, 실시예와 부합한 채로 있으면서 RAN(1304)이 임의의 수의 eNode B를 포함할 수 있다는 것을 잘 알 것이다. eNode B(1340a, 1340b, 1340c) 각각은 공중 인터페이스(1316)를 통해 WTRU(1302a, 1302b, 1302c)와 통신하기 위한 하나 이상의 송수신기를 포함할 수 있다. 일 실시예에서, eNode B(1340a, 1340b, 1340c)는 MIMO 기술을 구현할 수 있다. 따라서, 예를 들어, eNode B(1340a)는 WTRU(1302a)로 무선 신호를 전송하고 그로부터 무선 신호를 수신하기 위해 다수의 안테나를 사용할 수 있다.

[0150] eNode B(1340a, 1340b, 1340c) 각각은 특정의 셀(도시 생략)과 연관되어 있을 수 있고, 무선 자원 관리 결정, 핸드오버 결정, 상향링크 및/또는 하향링크에서의 사용자의 스케줄링 등을 처리하도록 구성되어 있을 수 있다. 도 13d에 도시된 바와 같이, eNode B(1340a, 1340b, 1340c)는 X2 인터페이스를 통해 서로 통신할 수 있다.

[0151] 도 13d에 도시된 코어 네트워크(1306)는 MME(mobility management gateway, 이동성 관리 게이트웨이)(1360), SGW(serving gateway, 서비스 제공 게이트웨이)(1362), 및/또는 PDN(packet data network, 패킷 데이터 네트워크) 게이트웨이(1364)를 포함할 수 있다. 상기 요소들 각각이 코어 네트워크(1306)의 일부로서 나타내어져 있지만, 이들 요소 중 임의의 것이 코어 네트워크 운영자 이외의 엔터티에 의해 소유되고 및/또는 운영될 수 있다는 것을 잘 알 것이다.

[0152] MME(1360)는 S1 인터페이스를 통해 RAN(1304) 내의 eNodeB(1340a, 1340b, 1340c) 각각에 연결되어 있을 수 있고, 제어 노드로서 역할할 수 있다. 예를 들어, MME(1360)는 WTRU(1302a, 1302b, 1302c)의 사용자를 인증하는 것, 베어러 활성화/비활성화, WTRU(1302a, 1302b, 1302c)의 초기 접속(initial attach) 동안 특정의 SGW(serving gateway)를 선택하는 것 등을 책임지고 있을 수 있다. MME(1360)는 또한 RAN(1304)과 GSM 또는 WCDMA와 같은 다른 무선 기술을 이용하는다른 RAN(도시 생략) 간에 전환하는 제어 평면 기능(control plane function)을 제공할 수 있다.

[0153] SGW(serving gateway)(1362)는 S1 인터페이스를 통해 RAN(1304) 내의 eNode B(1340a, 1340b, 1340c) 각각에 연결될 수 있다. 서비스 제공 게이트웨이(1362)는 일반적으로 WTRU(1302a, 1302b, 1302c)로/로부터 사용자 데이터 패킷을 라우팅하고 전달할 수 있다. SGW(serving gateway)(1362)는 eNode B간 핸드오버 동안 사용자 평면을 앵커링(anchoring)하는 것, WTRU(1302a, 1302b, 1302c)에 대해 하향링크 데이터가 이용가능할 때 페이징(paging)을 트리거하는 것, WTRU(1302a, 1302b, 1302c)의 컨텍스트를 관리하고 저장하는 것 및/또는 기타와 같은 다른 기능도 수행할 수 있다.

[0154] SGW(serving gateway)(1362)는, WTRU(1302a, 1302b, 1302c)와 IP-기반(IP-enabled) 디바이스 사이의 통신을 용이하게 해주기 위해, 인터넷(1310)과 같은 패킷 교환 네트워크에의 액세스를 WTRU(1302a, 1302b, 1302c)에 제공할 수 있는 PDN 게이트웨이(1364)에도 연결될 수 있다.

[0155] 코어 네트워크(1306)는 기타 네트워크와의 통신을 용이하게 해줄 수 있다. 예를 들어, 코어 네트워크(1306)는, WTRU(1302a, 1302b, 1302c)와 종래의 지상선(land-line) 통신 디바이스 사이의 통신을 용이하게 해주기 위해, PSTN(1308)과 같은 회선 교환 네트워크에의 액세스를 WTRU(1302a, 1302b, 1302c)에 제공할 수 있다. 예를 들어, 코어 네트워크(1306)는 코어 네트워크(1306)와 PSTN(1308) 사이의 인터페이스로서 역할하는 IP 게이트웨이[예컨대, IMS(IP multimedia subsystem, IP 멀티미디어 서브시스템) 서버]를 포함할 수 있거나 그와 통신할 수 있다. 그에 부가하여, 코어 네트워크(1306)는 다른 서비스 공급자에 의해 소유되고 및/또는 운영되는 다른 유선 또는 무선 네트워크를 포함할 수 있는 네트워크(1312)에의 액세스를 WTRU(1302a, 1302b, 1302c)에 제공할 수 있다.

[0156] 도 13e는 일 실시예에 따른, RAN(1304) 및 코어 네트워크(1306)의 시스템도이다. RAN(1304)은 공중 인터페이스(1316)를 통해 WTRU(1302a, 1302b, 1302c)와 통신하기 위해 IEEE 802.16 무선 기술을 이용하는 ASN(access service network)일 수 있다. 이하에서 더 논의할 것인 바와 같이, WTRU(1302a, 1302b, 1302c)의 상이한 기능적 엔터티 간의 통신 링크, RAN(1304), 및 코어 네트워크(1306)가 기준점으로서 정의될 수 있다.

[0157] 도 13e에 도시된 바와 같이, RAN(1304)은 기지국(1340a, 1340b, 1340c) 및 ASN 게이트웨이(1370)를 포함할 수 있지만, RAN(1304)이 실시예와 부합한 채로 있으면서 임의의 수의 기지국 및 ASN 게이트웨이를 포함할 수 있다는 것을 잘 알 것이다. 기지국(1340a, 1340b, 1340c)은 각각이 RAN(1304) 내의 특정의 셀(도시 생략)과 연관될 수 있고, 각각이 공중 인터페이스(1316)를 통해 WTRU(1302a, 1302b, 1302c)와 통신하기 위한 하나 이상의 송수신기를 포함할 수 있다. 일 실시예에서, 기지국(1340a, 1340b, 1340c)은 MIMO 기술을 구현할 수 있다. 따라서, 예를 들어, 기지국(1340a)은 WTRU(1302a)로 무선 신호를 전송하고 그로부터 무선 신호를 수신하기 위해 다수의 안테나를 사용할 수 있다. 기지국(1340a, 1340b, 1340c)은 또한 핸드오프 트리거링, 터널 설정, 무선 자원 관리, 트래픽 분류, QoS(quality of service) 정책 시행, 및/또는 기타와 같은 이동성 관리 기능을 제공할 수 있다. ASN 게이트웨이(1370)는 트래픽 집계 지점으로서 역할할 수 있고, 페이징, 가입자 프로필의 캐싱, 코어 네트워크(1306)로의 라우팅 등을 책임지고 있을 수 있다.

[0158] WTRU(1302a, 1302b, 1302c)와 RAN(1304) 사이의 공중 인터페이스(1316)는 IEEE 802.16 규격을 구현하는 R1 기준점으로서 정의될 수 있다. 그에 부가하여, WTRU(1302a, 1302b, 1302c) 각각은 코어 네트워크(1306)와 논리 인터페이스(도시 생략)를 설정할 수 있다. WTRU(1302a, 1302b, 1302c)와 코어 네트워크(1306) 사이의 논리 인터페이스는 인증, 허가, IP 호스트 구성 관리, 및/또는 이동성 관리를 위해 사용될 수 있는 R2 기준점으로서 정의될 수 있다.

[0159] 기지국(1340a, 1340b, 1340c) 각각 사이의 통신 링크는 기지국들 사이의 WTRU 핸드오버 및 데이터 전송을 용이하게 해주는 프로토콜을 포함하는 R8 기준점으로서 정의될 수 있다. 기지국(1340a, 1340b, 1340c)과 ASN 게이트웨이(1370) 사이의 통신 링크는 R6 기준점으로서 정의될 수 있다. R6 기준점은 WTRU(1302a, 1302b, 1302c) 각각과 연관된 이동성 이벤트에 기초하여 이동성 관리를 용이하게 해주는 프로토콜을 포함할 수 있다.

[0160] 도 13e에 도시된 바와 같이, RAN(1304)은 코어 네트워크(1306)에 연결될 수 있다. RAN(1304)과 코어 네트워크(1306) 사이의 통신 링크는, 예를 들어, 데이터 전송 및 이동성 관리 기능을 용이하게 해주는 프로토콜을 포함하는 R3 기준점으로서 정의될 수 있다. 코어 네트워크(1306)는 MIP-HA(mobile IP home agent, 이동 IP 홈 에이전트)(1372), AAA(authentication, authorization, accounting) 서버(1374), 및 게이트웨이(1376)를 포함할 수 있다. 상기 요소들 각각이 코어 네트워크(1306)의 일부로서 나타내어져 있지만, 이들 요소 중 임의의 것이 코어 네트워크 운영자 이외의 엔터티에 의해 소유되고 및/또는 운영될 수 있다는 것을 잘 알 것이다.

[0161] MIP-HA(1372)는 IP 주소 관리를 책임지고 있을 수 있고, WTRU(1302a, 1302b, 1302c)가 상이한 ASN 및/또는 상

이한 코어 네트워크 사이에서 로밍할 수 있게 해줄 수 있다. MIP-HA(1372)는, WTRU(1302a, 1302b, 1302c)와 IP-기반 디바이스 사이의 통신을 용이하게 해주기 위해, 인터넷(1310)과 같은 패킷 교환 네트워크에의 액세스를 WTRU(1302a, 1302b, 1302c)에 제공할 수 있다. AAA 서버(1374)는 사용자 인증 및 사용자 서비스를 지원하는 것을 책임지고 있을 수 있다. 게이트웨이(1376)는 다른 네트워크와의 연동을 용이하게 해줄 수 있다. 예를 들어, 게이트웨이(1376)는, WTRU(1302a, 1302b, 1302c)와 종래의 지상선(land-line) 통신 디바이스 사이의 통신을 용이하게 해주기 위해, PSTN(1308)과 같은 회선 교환 네트워크에의 액세스를 WTRU(1302a, 1302b, 1302c)에 제공할 수 있다. 그에 부가하여, 게이트웨이(1376)는 다른 서비스 공급자에 의해 소유되고 및/또는 운영되는 다른 유선 또는 무선 네트워크를 포함할 수 있는 네트워크(1312)에의 액세스를 WTRU(1302a, 1302b, 1302c)에 제공할 수 있다.

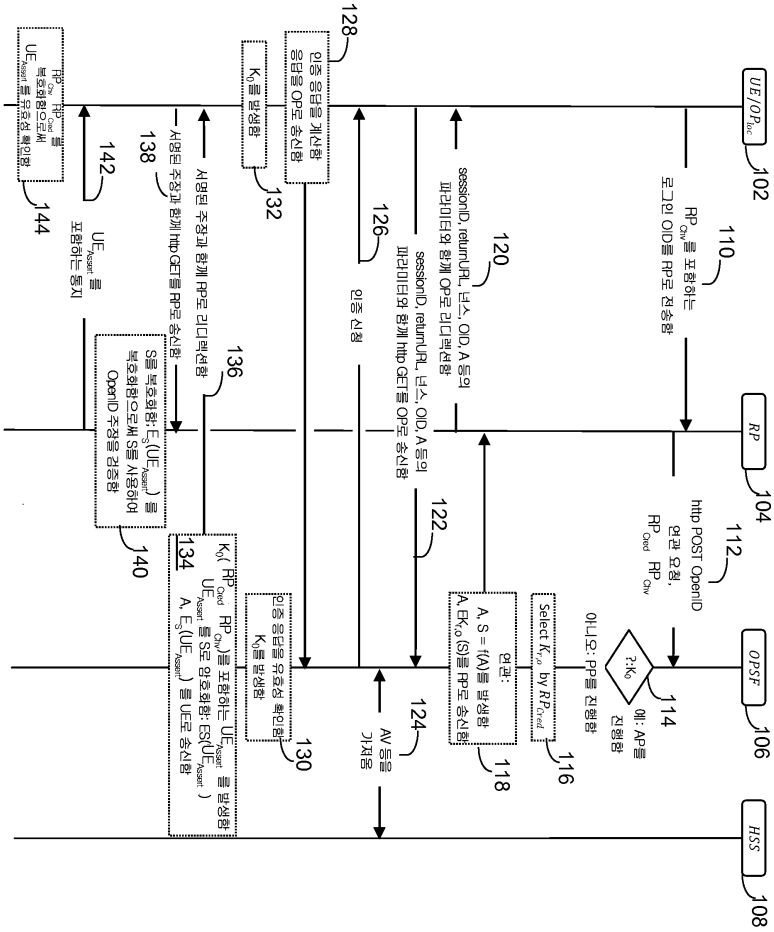
[0162] 도 13e에 도시되어 있지는 않지만, RAN(1304)이 다른 ASN에 연결될 수 있다는 것과 코어 네트워크(1306)가 다른 코어 네트워크에 연결될 수 있다는 것을 잘 알 것이다. RAN(1304)과 다른 ASN 사이의 통신 링크가 RAN(1304)과 다른 ASN 사이의 WTRU(1302a, 1302b, 1302c)의 이동성을 조정하는 프로토콜을 포함할 수 있는 R4 기준점으로서 정의될 수 있다. 코어 네트워크(1306)와 다른 코어 네트워크 사이의 통신 링크가 홈 코어 네트워크와 방문한 코어 네트워크 사이의 연동을 용이하게 해주는 프로토콜을 포함할 수 있는 R5 기준으로서 정의될 수 있다.

[0163] 본 명세서에 기술된 방법이 컴퓨터 또는 프로세서에서 실행하기 위해 컴퓨터 판독가능 매체에 포함되어 있는 컴퓨터 프로그램, 소프트웨어, 또는 펌웨어로 구현될 수 있다. 컴퓨터 판독가능 매체의 일례는 전자 신호(유선 또는 무선 연결을 통해 전송됨) 및 컴퓨터 판독가능 저장 매체를 포함한다. 컴퓨터 판독가능 저장 매체의 일례로는 ROM(read only memory), RAM(random access memory), 레지스터, 캐시 메모리, 반도체 메모리 디바이스, 내장형 하드 디스크 및 이동식 디스크 등의 자기 매체, 광자기 매체, 그리고 CD-ROM 디스크 및 DVD(digital versatile disk) 등의 광 매체가 있지만, 이들로 제한되지 않는다. 소프트웨어와 연관된 프로세서는 WTRU, UE, 단말, 기지국, RNC, 또는 임의의 호스트 컴퓨터에서 사용하기 위한 무선 주파수 송수신기를 구현하는 데 사용될 수 있다.

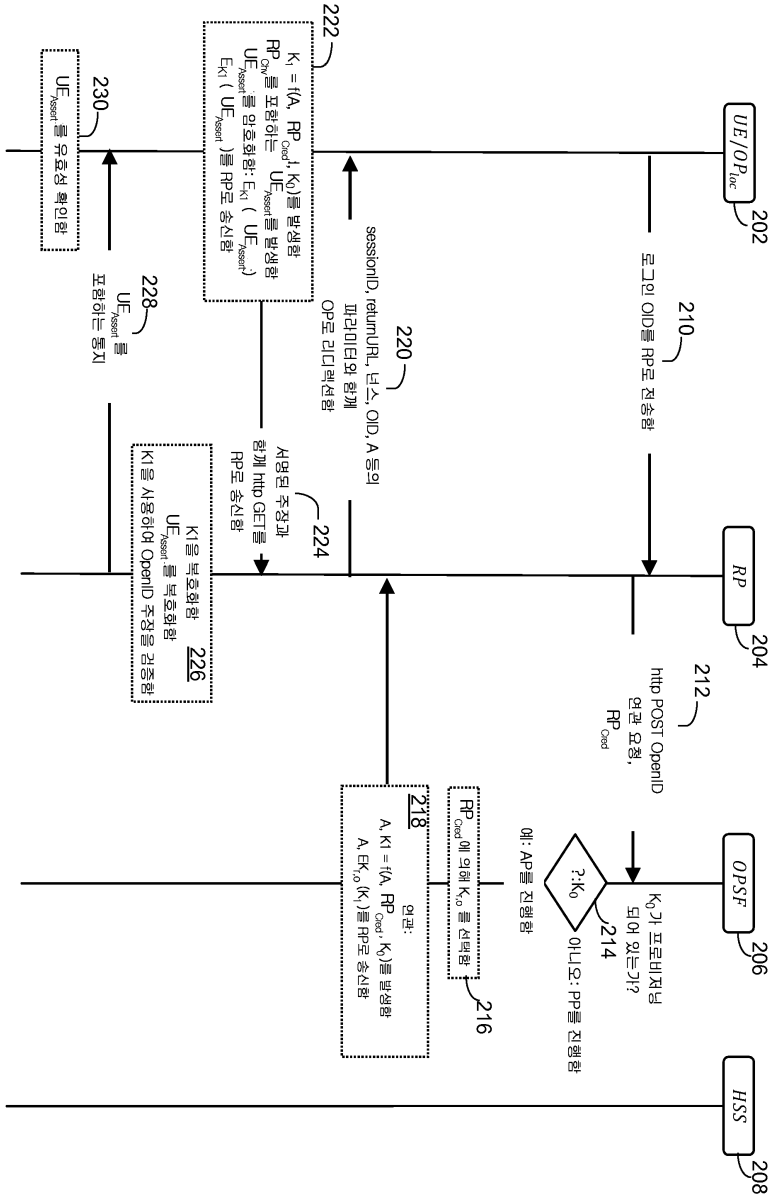
[0164] 특징 및 요소가 특정의 조합으로 앞서 기술되어 있지만, 각각의 특징 또는 요소가 단독으로 또는 다른 특징 및 요소와 임의의 조합으로 사용될 수 있다. 예를 들어, 본 명세서에 기술되어 있는 프로토콜 흐름 단계들은 이들이 기술되어 있는 순서로 제한되지 않는다. 그에 부가하여, 본 명세서에 기술되어 있는 실시예들이 OpenID 인증을 사용하여 기술되어 있을 수 있지만, 다른 형태의 인증이 구현될 수 있다. 이와 유사하게, 본 명세서에 기술되어 있는 실시예들은 OpenID 통신 또는 엔터티로 제한되지 않을 수 있다. 예를 들어, RP는 임의의 서비스 제공자를 제공할 수 있고, OP/OPSF는 임의의 ID 및/또는 주장 제공자(들)를 포함할 수 있으며, 및/또는 OP_{loc}는 임의의 로컬 ID 및/또는 주장 제공자일 수 있다. 게다가, 본 명세서에 기술되어 있는 UE의 임의의 인증은 UE 및 UE와 연관되어 있는 사용자의 인증을 포함할 수 있다.

도면

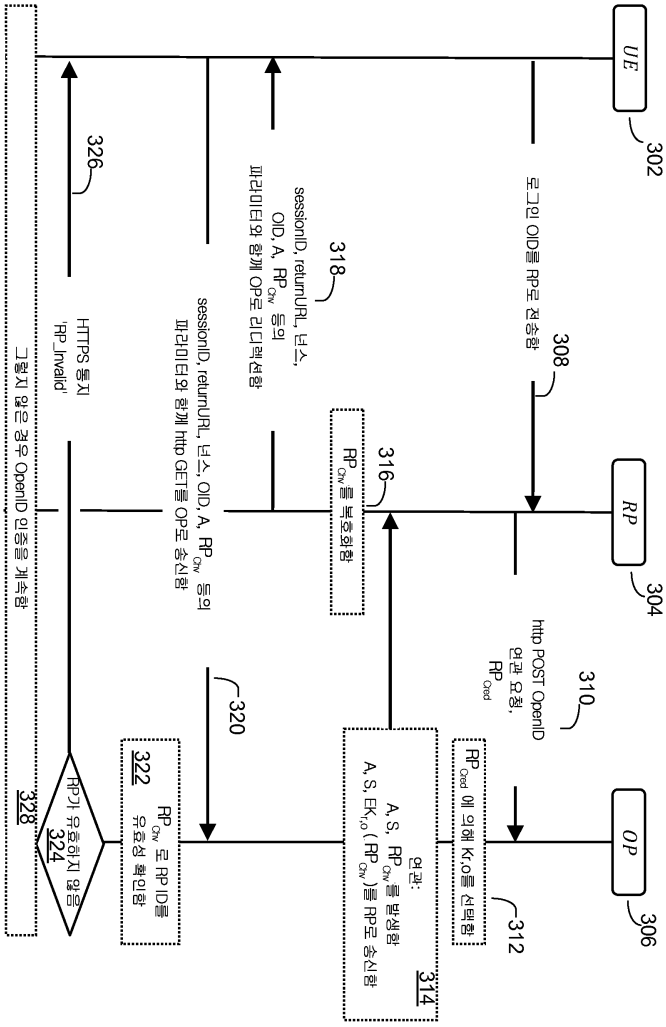
도면1


UE/OP_Dsc
RP
OPSF
HSS
102
104
106
108
110
RP_Clnt 를 포함하는 로그인 OID를 RP로 전송함
112
http POST OpenID 연관 요청, RP_Cred RP_Clnt
114
?:K_O
예: AP를 진행함
아니오: PP를 진행함
116
Select K_P,O by RP_Cred
118
연관: A, S = f(A)를 발생함 A, EK_P,O (S)를 RP로 송신함
120
sessionID, returnURL, 넌스, OID, A 등의 파라미터와 함께 OP로 리디렉션함
122
sessionID, returnURL, 넌스, OID, A 등의 파라미터와 함께 http GET을 OP로 송신함
124
AV 등을 가져옴
126
인증 신청
128
인증 응답을 계산함
응답을 OP로 송신함
130
인증 응답을 유효성 확인함
K_S를 발생함
132
K_S를 발생함
134
K_S(RP_Cred UE_Assert RP_Clnt)를 포함하는 UE_Assert 를 S로 암호화함: ES(UE_Assert) A, E_S(UE_Assert) 를 UE로 송신함
136
서명된 주장과 함께 http GET을 RP로 리디렉션함
138
서명된 주장과 함께 http GET을 RP로 송신함
140
S를 복호화함: E_S(UE_Assert) 를 복호화함으로써 S를 사용하여 OpenID 주장을 검증함
142
UE_Assert 를 포함하는 통지
144
RP_Cred RP_Clnt 를 복호화함으로써 UE_Assert 를 추출하고 확인함

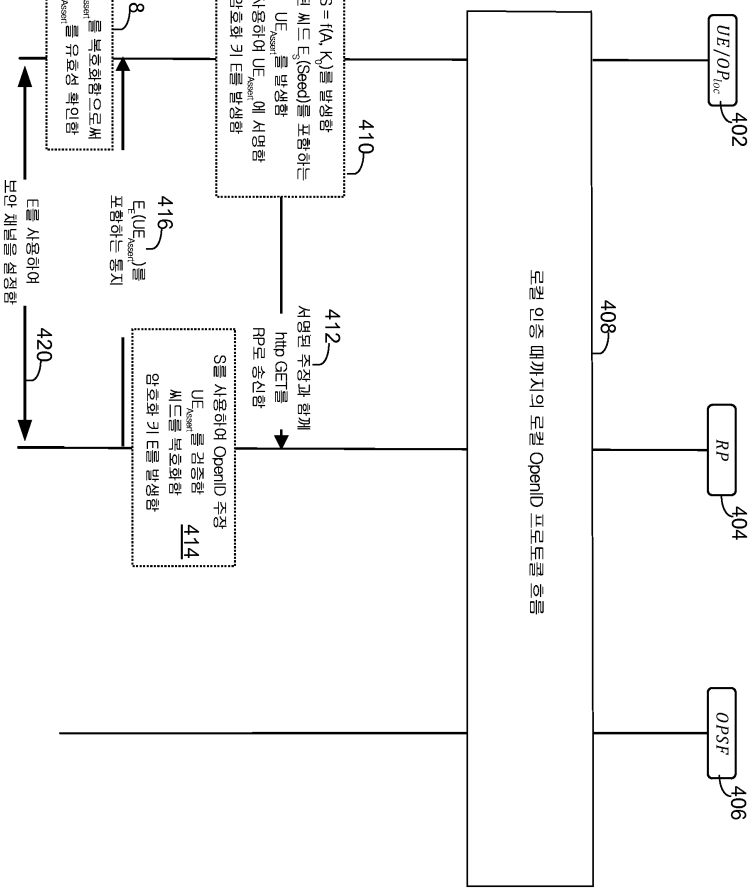
도면2


UE/OP_Unc
202
RP
204
OPSF
206
HSS
208
210
로그인 OID를 RP로 전송함
212
http POST OpenID 연관 요청, RP_Cred
?:K0
214
K0가 프로비저닝 되어 있는가?
예: AP를 진행함
아니오: PP를 진행함
216
RP_Cred에 의해 K0를 선택함
218
연관:
A, K1 = f(A, RP_Cred', K0')를 발생함
A, EK0(K1)를 RP로 송신함
220
sessionID, returnURL, 넌스, OID, A 등의 파라미터와 함께 OP로 리디렉션함
222
K1 = f(A, RP_Cred', K0')를 발생함
RP_Cred를 포함하는 UE_Assert를 발생함
UE_Assert를 암호화함: EK1 (UE_Assert)를 RP로 송신함
224
서명된 주장과 함께 http GET를 RP로 송신함
226
K1을 복호화함
UE_Assert를 복호화함
K1을 사용하여 OpenID 주장을 검증함
228
UE_Assert를 포함하는 통지
230
UE_Assert를 유효성 확인함

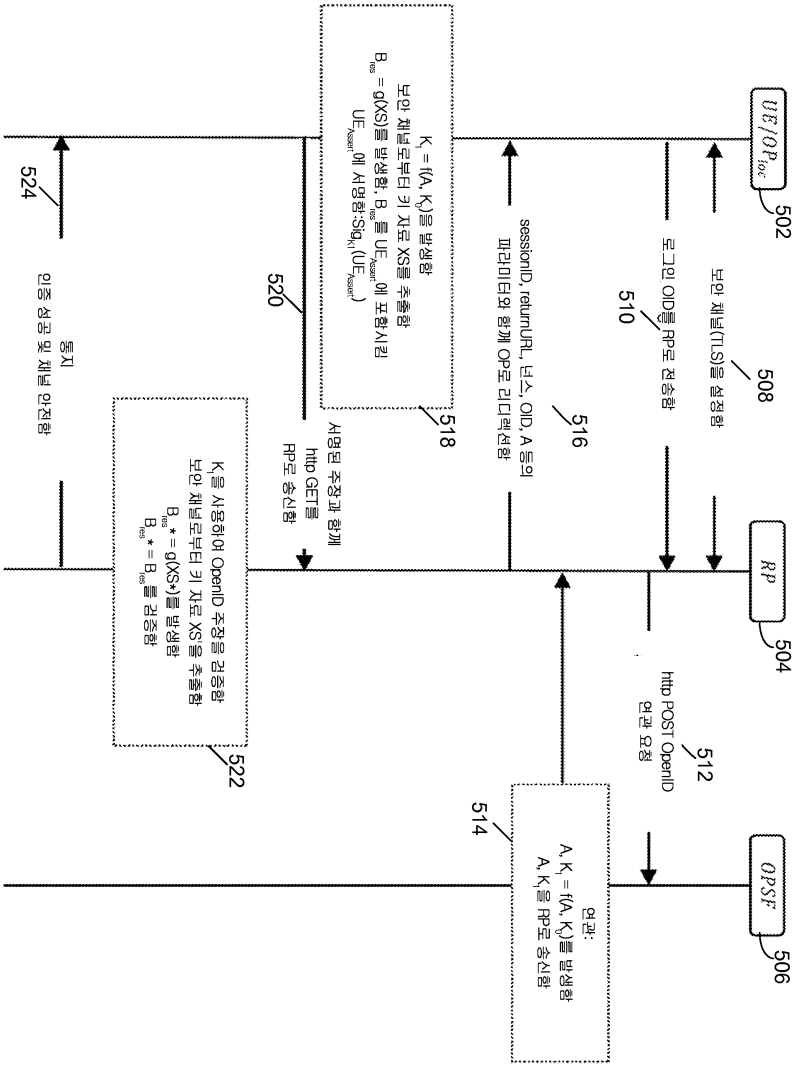
도면3


UE
302
RP
304
OP
306
로그인 OID를 RP로 전송함
308
http POST OpenID 연관 요청, RP Cnrd
310
RP Cnrd 에 의해 Kr,o를 선택함
312
연관:
A, S, RP Cnv를 발생함
A, S, EKr,o (RP Cnv)를 RP로 송신함
314
RP Cnv를 복호화함
316
sessionID, returnURL, 넌스, OID, A, RP Cnv 등의 파라미터와 함께 OP로 리디렉션함
318
sessionID, returnURL, 넌스, OID, A, RP Cnv 등의 파라미터와 함께 http GET를 OP로 송신함
320
RP Cnv 로 RP ID를 유효성 확인함
322
RP가 유효하지 않음
324
HTTPS 통지 'RP_Invalid'
326
그렇지 않은 경우 OpenID 인증을 계속함
328

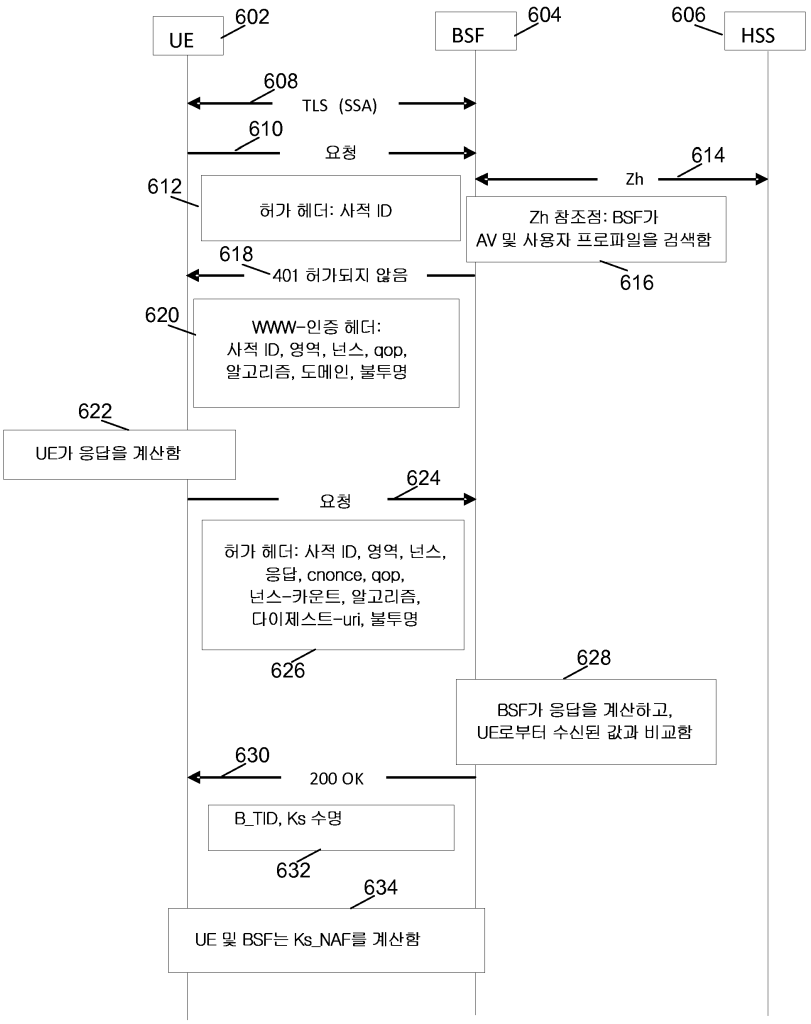
도면4


UE/OP_loc
402
RP
404
OPSF
406
로컬 인증 때까지의 로컬 OpenID 프로토콜 흐름
408
S = f(A, K_s)를 발생함
암호화된 씨드 E_s(Seed)를 포함하는
UE_Assert를 발생함
S를 사용하여 UE_Assert에 서명함
암호화 키 E를 발생함
410
서명된 주장과 함께
http GET를
RP로 송신함
412
S를 사용하여 OpenID 주장
UE_Assert를 검증함
씨드를 복호화함
암호화 키 E를 발생함
414
E_E(UE_Assert)를
포함하는 동지
416
UE_Assert를 복호화함으로써
UE_Assert를 유효성 확인함
418
E를 사용하여
보안 채널을 설정함
420

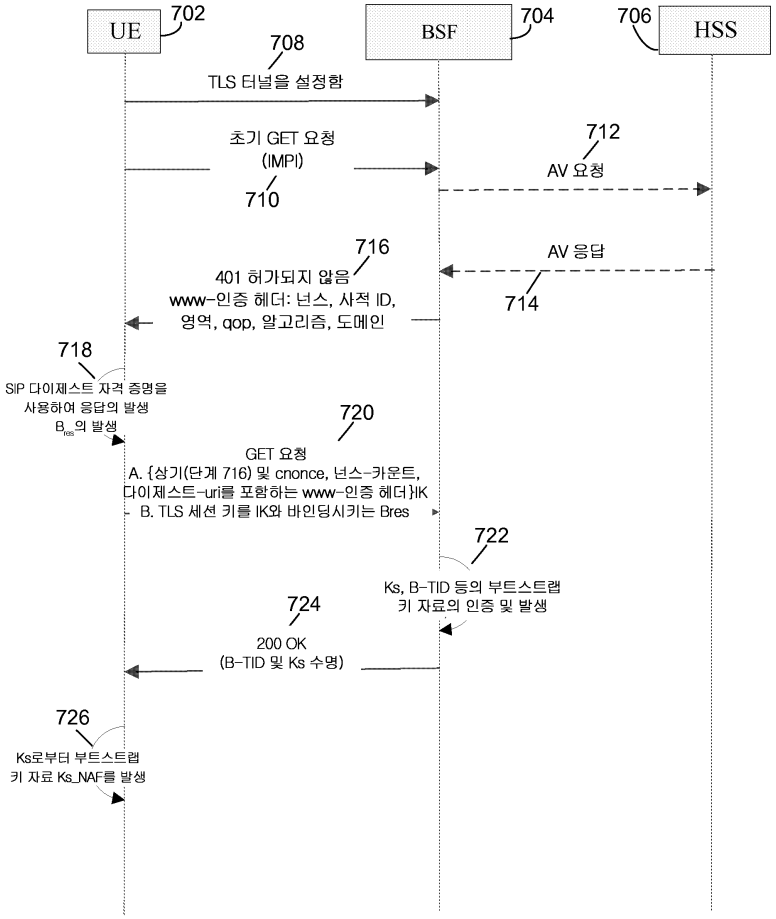
도면5


UE/OP_loc
502
보안 채널(TLS)을 설정함
508
로그인 OID를 RP로 전송함
510
RP
504
sessionID, returnURL, 넌스, OID, A 등의
파라미터와 함께 OP로 리디렉션함
516
http POST OpenID
연관 요청
512
OPSF
506
연관:
A, K_c = f(A, K_s)를 발생함
A, K_c을 RP로 송신함
514
K_c = f(A, K_s)을 발생함
보안 채널로부터 키 자료 XS를 추출함
B_res = g(XS)를 발생함; B_res를 UE_Assert에 포함시킴
UE_Assert에 서명함: Sig_c(UE_Assert)
518
서명된 주장과 함께
http GET을
RP로 송신함
520
K_c을 사용하여 OpenID 주장을 검증함
보안 채널로부터 키 자료 XS를 추출함
B_res* = g(XS*)를 발생함
B_res* = B_res를 검증함
522
통지
인증 성공 및 채널 안전함
524

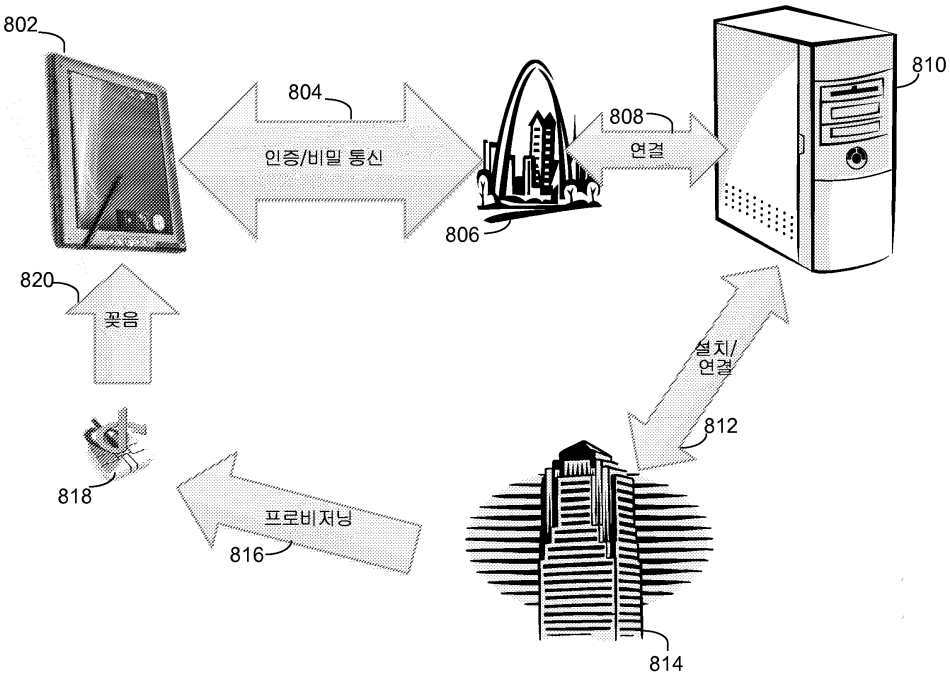
도면6


UE
602
BSF
604
606
HSS
608
TLS (SSA)
610
요청
614
Zh
612
허가 헤더: 사적 ID
Zh 참조점: BSF가
AV 및 사용자 프로파일을 검색함
618
401 허가되지 않음
616
620
WWW-인증 헤더:
사적 ID, 영역, 넌스, qop,
알고리즘, 도메인, 불투명
622
UE가 응답을 계산함
624
요청
허가 헤더: 사적 ID, 영역, 넌스,
응답, cnonce, qop,
넌스-카운트, 알고리즘,
다이제스트-uri, 불투명
626
628
BSF가 응답을 계산하고,
UE로부터 수신된 값과 비교함
630
200 OK
B_TID, Ks 수명
632
634
UE 및 BSF는 Ks_NAF를 계산함

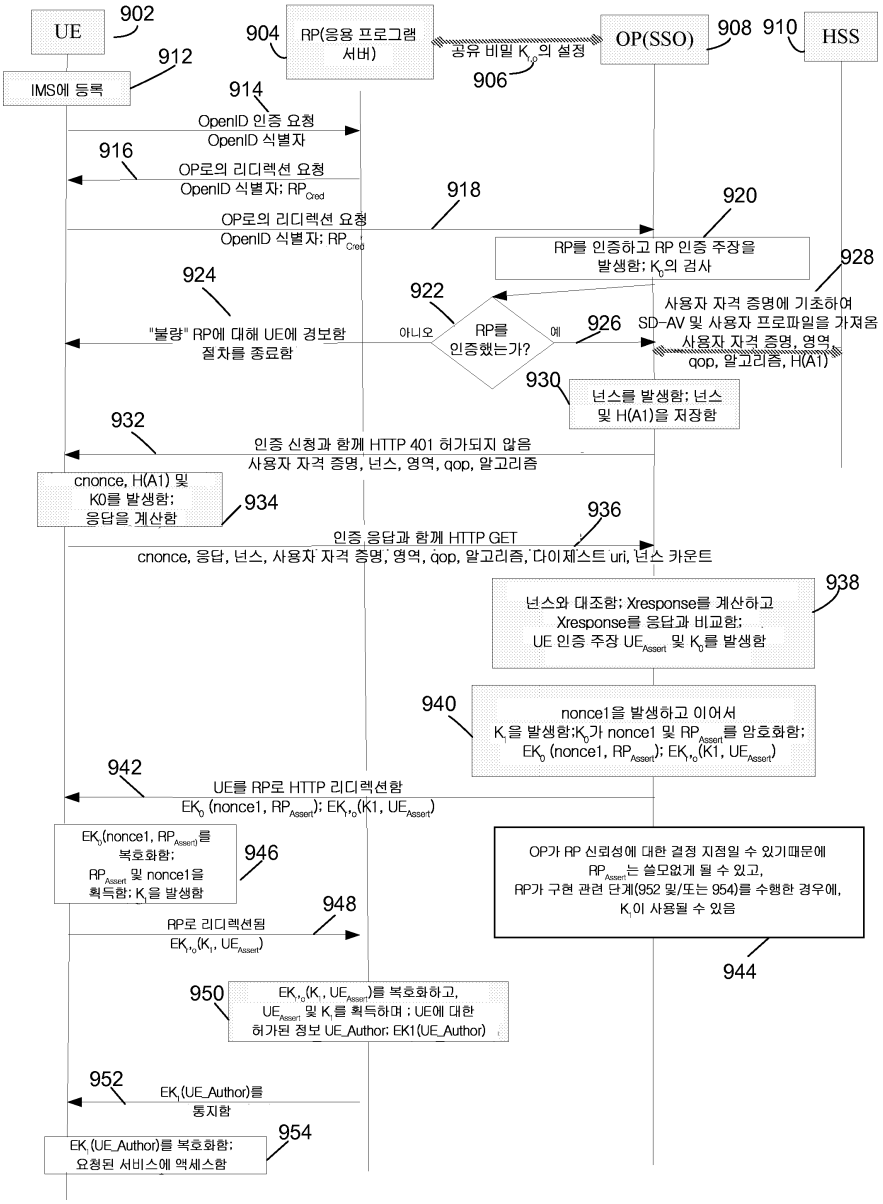
도면7


UE
702
BSF
704
706
HSS
708
TLS 터널을 설정함
초기 GET 요청
(IMPI)
710
712
AV 요청
716
401 허가되지 않음
www-인증 헤더: 넌스, 사적 ID,
영역, qop, 알고리즘, 도메인
AV 응답
714
718
SIP 다이제스트 자격 증명을
사용하여 응답의 발생
B_{res}의 발생
720
GET 요청
A. {상기(단계 716) 및 cnonce, 넌스-카운트,
다이제스트-uri를 포함하는 www-인증 헤더}IK
B. TLS 세션 키를 IK와 바인딩시키는 Bres
722
Ks, B-TID 등의 부트스트랩
키 자료의 인증 및 발생
724
200 OK
(B-TID 및 Ks 수명)
726
Ks로부터 부트스트랩
키 자료 Ks_NAF를 발생

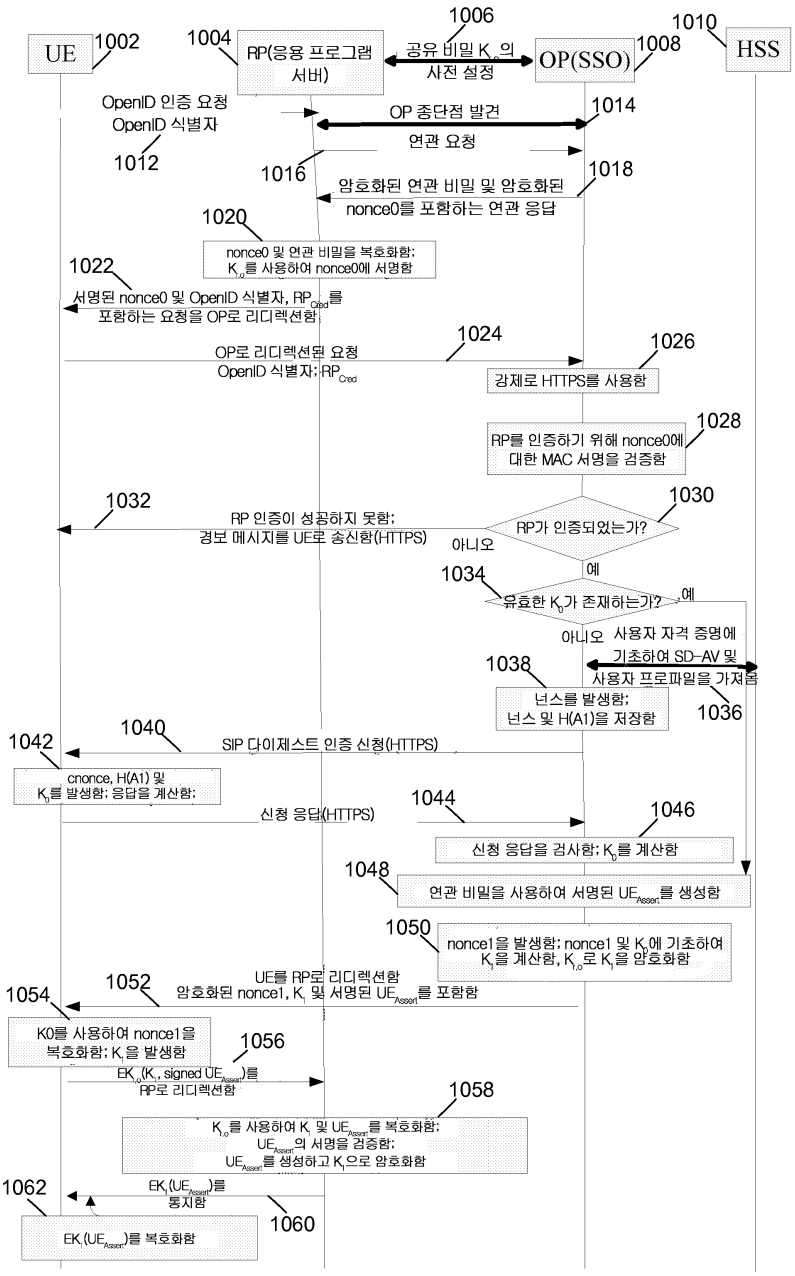
도면8



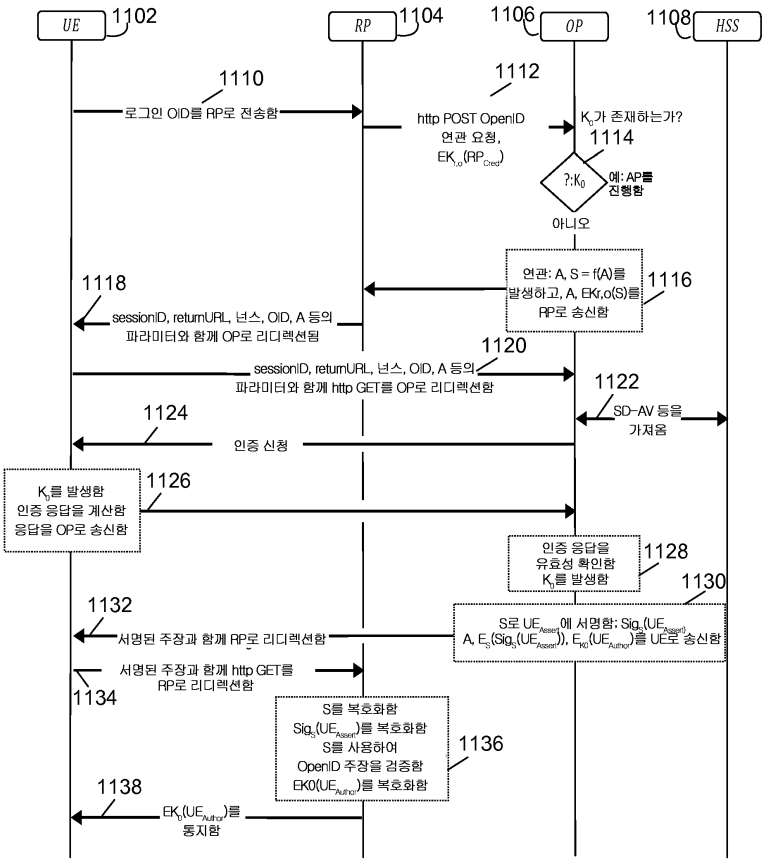
도면9


UE
902
RP(응용 프로그램 서버)
904
공유 비밀 K_0의 설정
906
OP(SSO)
908
HSS
910
IMS에 등록
912
OpenID 인증 요청
OpenID 식별자
914
OP로의 리디렉션 요청
OpenID 식별자; RP_{Cred}
916
OP로의 리디렉션 요청
OpenID 식별자; RP_{Cred}
918
RP를 인증하고 RP 인증 주장을 발생함; K_0의 검사
920
사용자 자격 증명에 기초하여 SD-AV 및 사용자 프로파일을 가져옴
사용자 자격 증명, 영역, qop, 알고리즘, H(A1)
928
RP를 인증했는가?
922
아니오
예
926
"불량" RP에 대해 UE에 경보함
절차를 종료함
924
넌스를 발생함; 넌스 및 H(A1)을 저장함
930
인증 신청과 함께 HTTP 401 허가되지 않음
사용자 자격 증명, 넌스, 영역, qop, 알고리즘
932
cnonce, H(A1) 및 K0를 발생함; 응답을 계산함
934
인증 응답과 함께 HTTP GET
cnonce, 응답, 넌스, 사용자 자격 증명, 영역, qop, 알고리즘, 다이제스트 uri, 넌스 카운트
936
넌스와 대조함; Xresponse를 계산하고 Xresponse를 응답과 비교함; UE 인증 주장 UE_{Assert} 및 K_0를 발생함
938
nonce1을 발생하고 이어서 K_1을 발생함;K_0가 nonce1 및 RP_{Assert}를 암호화함; EK_0 (nonce1, RP_{Assert}); $EK_{1,0}$(K1, UE_{Assert})
940
UE를 RP로 HTTP 리디렉션함
EK_0 (nonce1, RP_{Assert}); $EK_{1,0}$(K1, UE_{Assert})
942
EK_0(nonce1, RP_{Assert})를 복호화함; RP_{Assert} 및 nonce1을 획득함; K_1을 발생함
946
OP가 RP 신뢰성에 대한 결정 지점일 수 있기때문에 RP_{Assert}는 쓸모없게 될 수 있고, RP가 구현 관련 단계(952 및/또는 954)를 수행한 경우에, K_1이 사용될 수 있음
944
RP로 리디렉션됨
$EK_{1,0}$(K_1, UE_{Assert})
948
$EK_{1,0}$(K_1, UE_{Assert})를 복호화하고, UE_{Assert} 및 K_1를 획득하며 ; UE에 대한 허가된 정보 UE_Author; EK1(UE_Author)
950
EK_1(UE_Author)를 통지함
952
EK_1(UE_Author)를 복호화함; 요청된 서비스에 액세스함
954

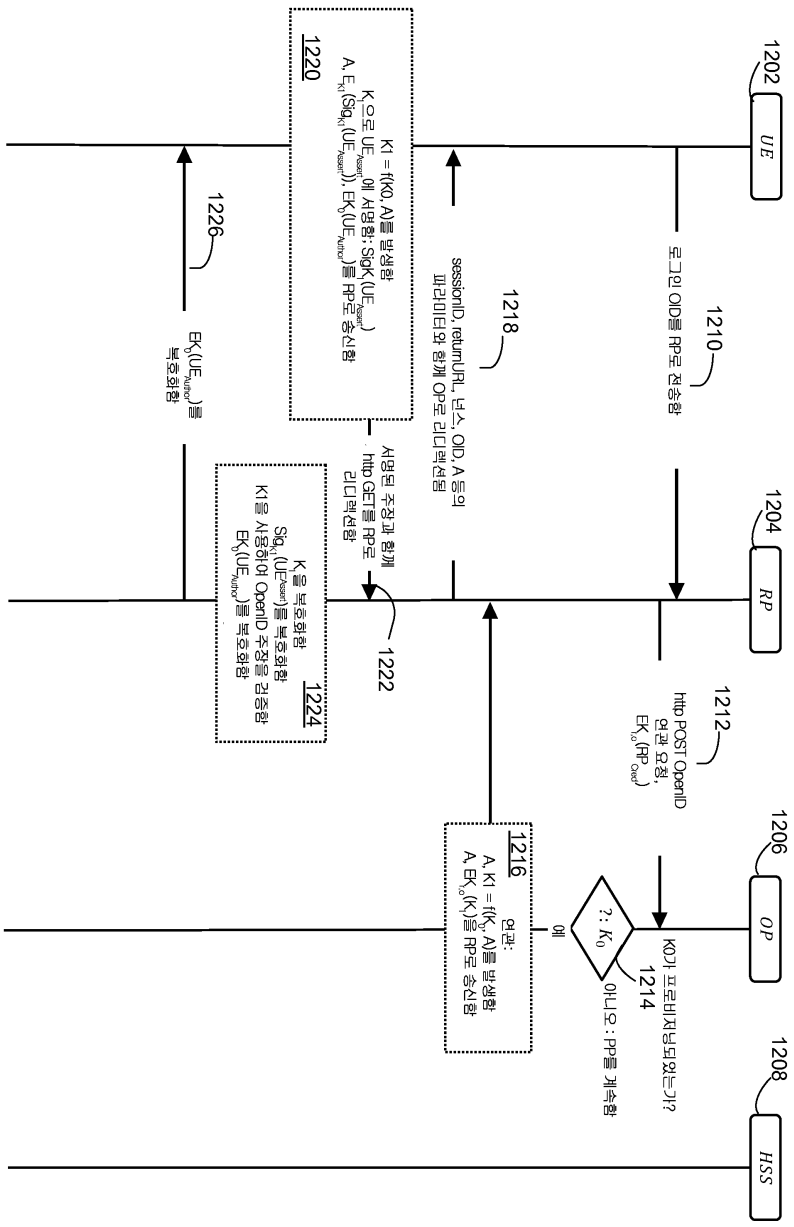
도면10


UE
1002
1004
RP(응용 프로그램 서버)
1006
공유 비밀 K_o의 사전 설정
OP(SSO)
1008
1010
HSS
OpenID 인증 요청
OpenID 식별자
1012
OP 종단점 발견
연관 요청
1014
1016
암호화된 연관 비밀 및 암호화된 nonce0를 포함하는 연관 응답
1018
1020
nonce0 및 연관 비밀을 복호화함; $K_{1,o}$를 사용하여 nonce0에 서명함
1022
서명된 nonce0 및 OpenID 식별자, RP_{Cred}를 포함하는 요청을 OP로 리디렉션함
1024
OP로 리디렉션된 요청 OpenID 식별자; RP_{Cred}
1026
강제로 HTTPS를 사용함
1028
RP를 인증하기 위해 nonce0에 대한 MAC 서명을 검증함
1030
RP가 인증되었는가?
아니오
1032
RP 인증이 성공하지 못함; 경보 메시지를 UE로 송신함(HTTPS)
예
1034
유효한 K_0가 존재하는가?
예
아니오
사용자 자격 증명에 기초하여 SD-AV 및 사용자 프로파일을 가져옴
1036
1038
넌스를 발생함; 넌스 및 H(A1)을 저장함
1040
SIP 다이제스트 인증 신청(HTTPS)
1042
cnonce, H(A1) 및 K_1를 발생함; 응답을 계산함;
신청 응답(HTTPS)
1044
1046
신청 응답을 검사함; K_0를 계산함
1048
연관 비밀을 사용하여 서명된 UE_{Assert}를 생성함
1050
nonce1을 발생함; nonce1 및 K_0에 기초하여 K_1을 계산함, $K_{1,o}$로 K_1을 암호화함
1052
UE를 RP로 리디렉션함
암호화된 nonce1, K_1 및 서명된 UE_{Assert}를 포함함
1054
K0를 사용하여 nonce1을 복호화함; K_1을 발생함
1056
$EK_{1,o}(K_1, \text{signed } UE_{Assert})$를 RP로 리디렉션함
1058
$K_{1,o}$를 사용하여 K_1 및 UE_{Assert}를 복호화함; UE_{Assert}의 서명을 검증함; UE_{Assert}를 생성하고 K_1으로 암호화함
1062
$EK_1(UE_{Assert})$를 통지함
1060
$EK_1(UE_{Assert})$를 복호화함

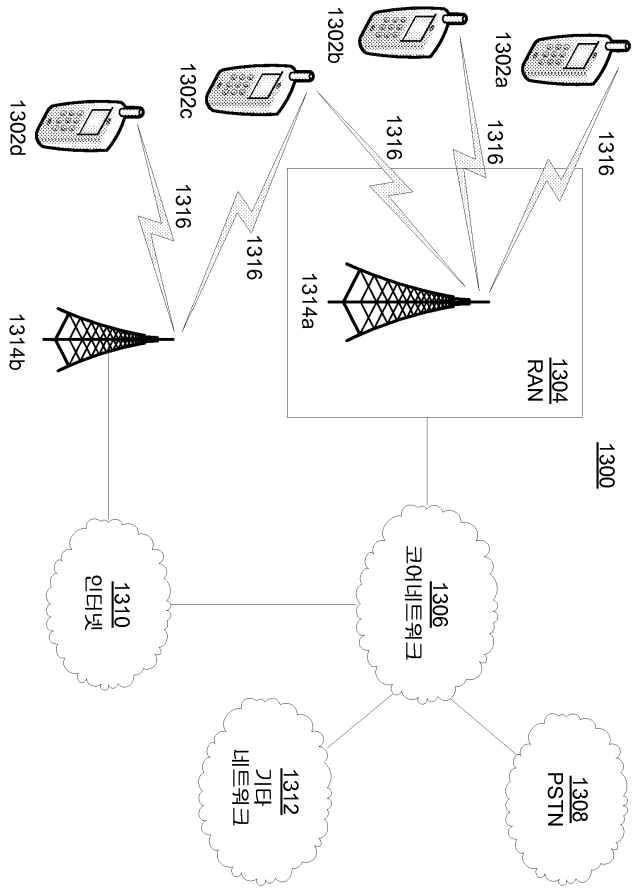
도면11


UE
1102
RP
1104
1106
OP
1108
HSS
1110
로그인 OID를 RP로 전송함
1112
http POST OpenID 연관 요청, EK$_{r,o}$(RP$_{Chal}$)
K$_0$가 존재하는가?
1114
?:K$_0$
예: AP를 진행함
아니오
연관: A, S = f(A)를 발생하고, A, EKr,o(S)를 RP로 송신함
1116
1118
sessionID, returnURL, 넌스, OID, A 등의 파라미터와 함께 OP로 리디렉션됨
1120
sessionID, returnURL, 넌스, OID, A 등의 파라미터와 함께 http GET를 OP로 리디렉션함
1122
SD-AV 등을 가져옴
1124
인증 신청
K$_0$를 발생함
인증 응답을 계산함
응답을 OP로 송신함
1126
인증 응답을 유효성 확인함
K$_0$를 발생함
1128
1130
S로 UE$_{Assert}$에 서명함; Sig$_S$(UE$_{Assert}$) A, E$_S$(Sig$_S$(UE$_{Assert}$)), E$_{K0}$(UE$_{Author}$)를 UE로 송신함
1132
서명된 주장과 함께 RP로 리디렉션함
서명된 주장과 함께 http GET를 RP로 리디렉션함
1134
S를 복호화함
Sig$_S$(UE$_{Assert}$)를 복호화함
S를 사용하여 OpenID 주장을 검증함
EK0(UE$_{Author}$)를 복호화함
1136
1138
EK$_0$(UE$_{Author}$)를 통지함

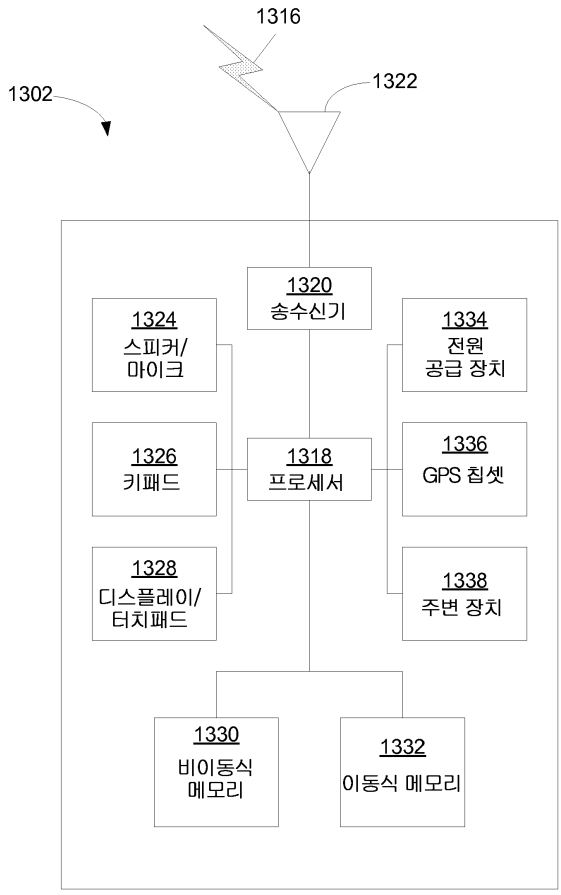
도면12


UE
RP
OP
HSS
1202
1204
1206
1208
로그인 OID를 RP로 전송함
1210
sessionID, returnURL, 넌스, OID, A 등의 파라미터와 함께 OP로 리디렉션됨
1218
서명된 주장과 함께 http GET를 RP로 리디렉션함
1222
http POST OpenID 연관 요청 $EK_{OP}(RP_{Ord})$
1212
K0가 프로비저닝되었는가?
1214
예
아니오 : PP를 계속함
$?: K_0$
연관: A, K1 = $f(K_0, A)$를 발생함 A, $EK_{OP}(K)$를 RP로 송신함
1216
K1 = f(K0, A)를 발생함 K으로 UE_{Assert}에 서명함: $SigK(UE_{Assert})$ A, $E_{K1}(Sig_{K1}(UE_{Assert}))$, $EK_0(UE_{Authn})$를 RP로 송신함
1220
K을 복호화함 $Sig_{K1}(UE^{Assert})$를 복호화함 K1을 사용하여 OpenID 주장을 검증함 $EK_0(UE_{Authn})$를 복호화함
1224
$EK_0(UE_{Authn})$를 복호화함
1226

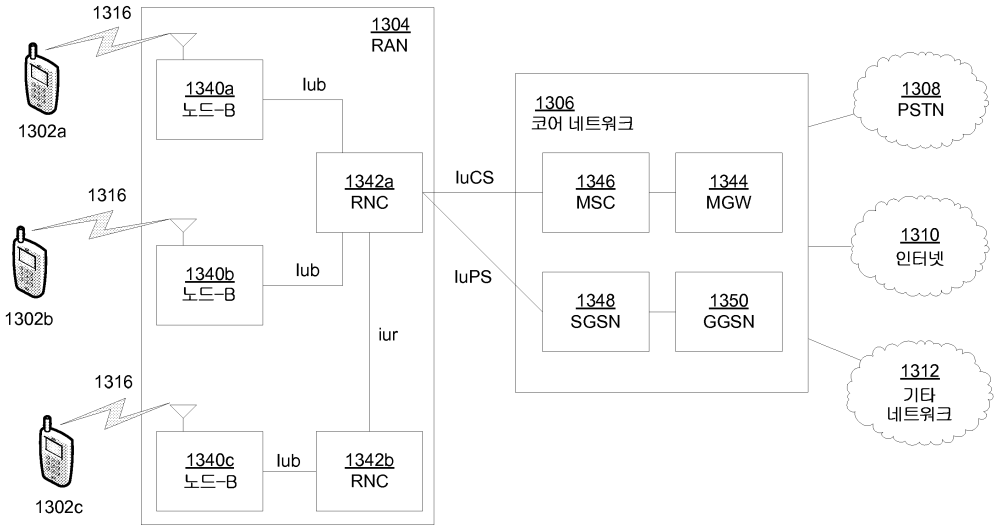
도면13a


1302a
1302b
1302c
1302d
1316
1316
1316
1316
1314a
1314b
1304
RAN
1300
1306
코어네트워크
1310
인터넷
1312
기타
네트워크
1308
PSTN

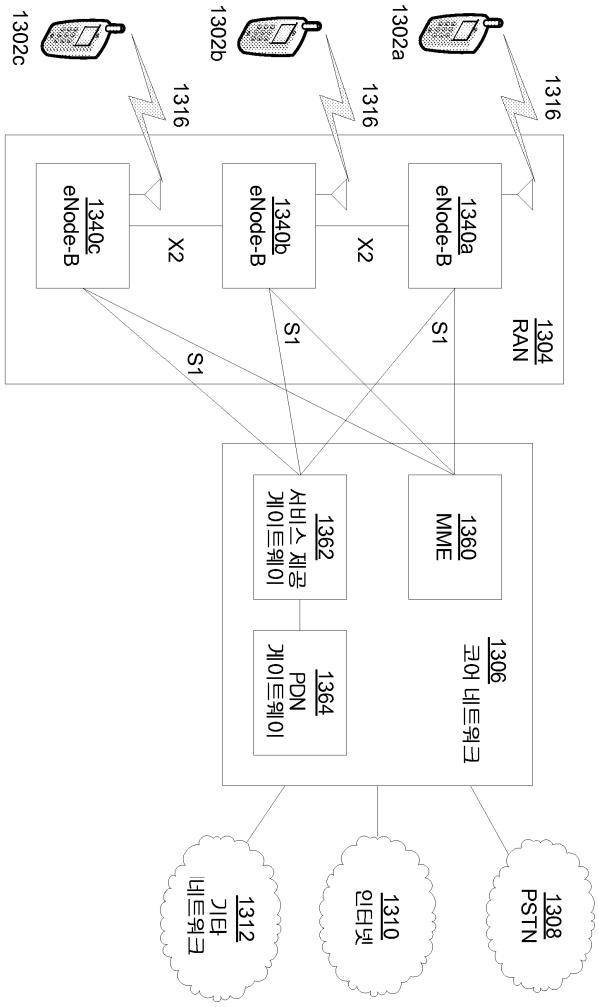
도면13b


1316
1322
1302
1320
송수신기
1324
스피커/
마이크
1334
전원
공급 장치
1326
키패드
1318
프로세서
1336
GPS 칩셋
1328
디스플레이/
터치패드
1338
주변 장치
1330
비이동식
메모리
1332
이동식 메모리

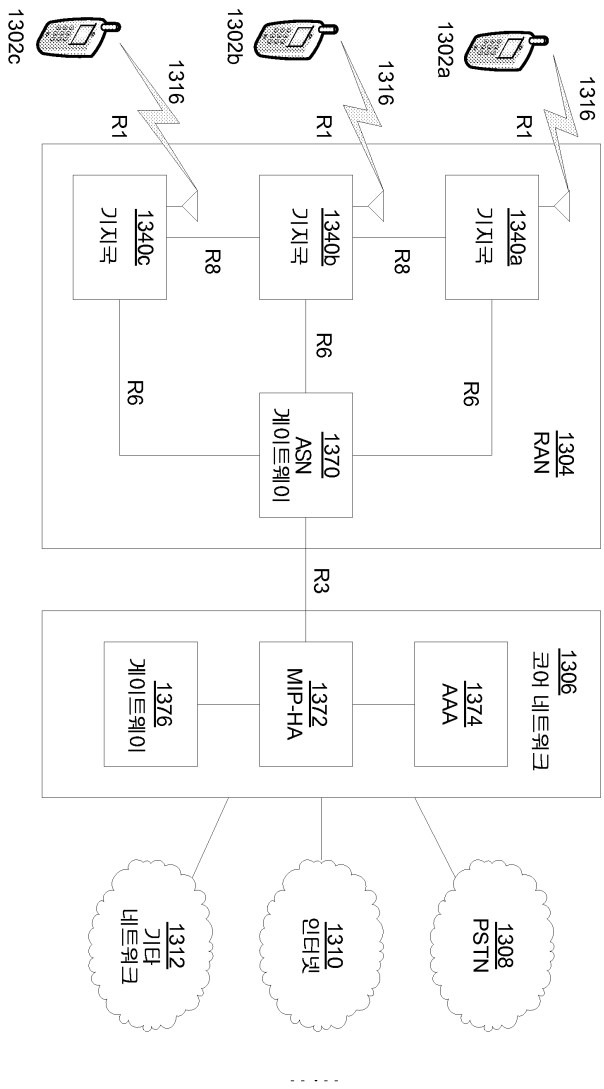
도면13c


1316
1304
RAN
1340a
노드-B
Iub
1302a
1316
1342a
RNC
IuCS
1306
코어 네트워크
1346
MSC
1344
MGW
1308
PSTN
1340b
노드-B
Iub
1302b
IuPS
1310
인터넷
1348
SGSN
1350
GGSN
iur
1316
1312
기타
네트워크
1340c
노드-B
Iub
1342b
RNC
1302c

도면13d


1302c
1302b
1302a
1316
1316
1316
1340c eNode-B
1340b eNode-B
1340a eNode-B
X2
X2
S1
S1
S1
1304 RAN
1362 서비스 제공 게이트웨이
1360 MME
1364 PDN 게이트웨이
1306 코어 네트워크
1312 기타 네트워크
1310 인터넷
1308 PSTN

도면13e


1302c
1302b
1302a
1316
R1
1340c 기지국
1340b 기지국
1340a 기지국
R8
R6
1370 ASN 게이트웨이
1304 RAN
R3
1376 게이트웨이
1372 MIP-HA
1374 AAA
1306 코어 네트워크
1312 기타 네트워크
1310 인터넷
1308 PSTN