(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2020년11월12일
(11) 등록번호 10-2177956
(24) 등록일자 2020년11월06일

(51) 국제특허분류(Int. Cl.)
H04L 9/32 (2006.01) *H04L 9/30* (2006.01)
(52) CPC특허분류
H04L 9/3263 (2013.01)
H04L 9/302 (2013.01)
(21) 출원번호 10-2017-0055361(분할)
(22) 출원일자 2017년04월28일
심사청구일자 2020년04월06일
(65) 공개번호 10-2017-0052548
(43) 공개일자 2017년05월12일
(62) 원출원 특허 10-2015-0050155
원출원일자 2015년04월09일
심사청구일자 2015년04월09일
(30) 우선권주장
1020140042360 2014년04월09일 대한민국(KR)
(56) 선행기술조사문헌
KR1020090017538 A*
US20110103589 A1*
US20120321077 A1*
QG.N.Shinde 외 1명, "Faster RSA Algorithm for Decryption Using Chinese Remainder Theorem", ICCES, vol.5, no.4, pp.255-261 (2008.)*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
주식회사 아이씨티케이 홀딩스
경기도 성남시 분당구 판교로 323 ,5층(삼평동,V-Forum빌딩)
한양대학교 산학협력단
서울특별시 성동구 왕십리로 222(행당동, 한양대학교내)
(72) 발명자
김동규
서울특별시 동대문구 장한로14길 81, 105동 2103호 (장안동, 삼성래미안1차아파트)
최병덕
서울특별시 강동구 올림픽로62길 9-24
(뒷면에 계속)
(74) 대리인
특허법인 무한

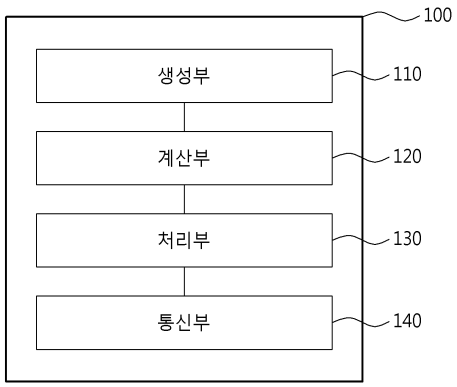
전체 청구항 수 : 총 20 항 심사관 : 양종필

(54) 발명의 명칭 **인증 장치 및 방법**

(57) 요 약

공개키 암호화 알고리즘을 이용하는 인증 장치가 제시된다. 일실시예에 따르면 장치는, 메시지에 대응하는 전자서명 생성 요청에 응답하여, 랜덤 넘버 생성 과정을 통해 제1 인스턴트 공개키를 생성한다. 그리고 상기 제1 인스턴트 공개키를 이용해서, 상기 제1 인스턴트 공개키와 쌍을 이루는 제1 인스턴트 개인키를 계산하여 사용한다.

대 표 도 - 도1



(52) CPC특허분류
H04L 9/3247 (2013.01)
H04L 9/3249 (2013.01)
H04L 9/3278 (2013.01)
(72) 발명자
김동현
경기도 용인시 수지구 상현로 25, 176동 1201호 (상현동, 상현마을쌍용아파트)
박상선
경기도 성남시 분당구 불정로 179, 208동 102호 (정자동, 정든마을동아아파트)

명 세 서

청구범위

청구항 1

적어도 하나의 프로세서를 포함하고 공개키 기반 암호화 알고리즘(public-key cryptosystems)에 따른 인증 절차를 수행하는 인증 장치에 있어서, 상기 인증 장치는 상기 적어도 하나의 프로세서에 의해 적어도 일시적으로 구현되는:

상기 알고리즘에 대응하는 전자 서명이 필요한 경우에 응답하여, 메시지에 대한 제1 인스턴트 공개키를 생성하는 생성부;

상기 제1 인스턴트 공개키, 고정 개인키 및 상기 알고리즘을 이용해서, 상기 알고리즘에서 상기 제1 인스턴트 공개키와 쌍을 이루는 제1 인스턴트 개인키를 계산하는 계산부; 및

상기 제1 인스턴트 개인키, 상기 메시지 및 상기 고정 개인키를 이용하여 상기 알고리즘에 의한 전자서명을 생성하는 처리부

를 포함하고,

상기 제1 인스턴트 공개키는 상기 인증 절차를 위해 일시적으로 생성되어 제2 인스턴트 공개키는 상기 제1 인스턴트 공개키와 상이하고,

각 인스턴트 개인키는 대응하여 생성된 인스턴트 공개키에 기초하여 계산되는 인증 장치.

청구항 2

제1항에 있어서,

전송할 메시지를 상기 전자서명 및 상기 제1 인스턴트 공개키와 함께 상대 기기에 전송하는 통신부

를 더 포함하는 인증 장치.

청구항 3

제2항에 있어서,

상기 상대 기기로부터 재전송 요청이 수신되는 것에 응답하여, 통신 Ack가 수신되어 통신 오류가 없었던 것으로 판단되는 경우에 해당한다면 상기 생성부가 상기 제1 인스턴트 공개키와 상이한 제2 인스턴트 공개키를 생성하는 인증 장치.

청구항 4

제1항에 있어서,

상기 생성부는, 랜덤 넘버 생성 과정을 통해 상기 제1 인스턴트 공개키를 생성하는 인증 장치.

청구항 5

제4항에 있어서,

상기 인증 장치는 무작위적으로 발생하는 공정 편차를 이용하여 하드웨어 핑거프린트를 제공하는 PUF(Physically Unclonable Function)를 더 포함하고,

상기 랜덤 넘버 생성 과정은, 상기 하드웨어 핑거프린트를 근원 값으로 이용하는 랜덤 넘버 생성 알고리즘을 포함하는 인증 장치.

청구항 6

제4항에 있어서,

상기 계산 결과에서 상기 제1 인스턴트 공개키와 쌍을 이루는 상기 제1 인스턴트 개인키가 존재하지 않는 것으로 판단되는 경우:

상기 생성부는 상기 제1 인스턴트 공개키와 상이한 제2 인스턴트 공개키를 생성하고; 상기 계산부는 상기 알고리즘에서 상기 제2 인스턴트 공개키와 쌍을 이루는 제2 인스턴트 개인키를 계산하는 인증 장치.

청구항 7

제6항에 있어서,

상기 생성부는 상기 제2 인스턴트 공개키를 생성하기 위해 상기 랜덤 넘버 생성 과정을 수행하는 대신 상기 제1 인스턴트 공개키에 정수 2를 더한 수를 상기 제2 인스턴트 공개키로 결정하여 제공하는 인증 장치.

청구항 8

제6항에 있어서,

상기 암호화 알고리즘이 RSA-CRT (Chinese Remainder Theorem) 알고리즘인 경우, 상기 계산부가 계산하는 상기 제1 인스턴트 개인키는 제1 dP 값 및 제1 dQ 값을 포함하고,

상기 계산 결과에서 상기 제1 dP 값 및 상기 제1 dQ 값 중 어느 하나라도 존재하지 않는 것으로 판단되는 경우:

상기 생성부는 상기 제1 인스턴트 공개키와 상이한 제2 인스턴트 공개키를 생성하고; 상기 계산부는 상기 알고리즘에서 상기 제2 인스턴트 공개키와 쌍을 이루는 제2 dP 값 및 제2 dQ 값을 계산하는 인증 장치.

청구항 9

적어도 하나의 프로세서를 포함하고 공개키 기반 암호화 알고리즘에 따라 상대 기기가 전송한 전자서명을 검증하는 인증 장치에 있어서, 상기 인증 장치는 상기 적어도 하나의 프로세서에 의해 적어도 일시적으로 구현되는:

보유하고 있던 상기 상대 기기의 고정 공개키, 및 상기 상대 기기가 전자서명에 대한 요청에 응답하여 메시지에 대해 인스턴트하게 생성하여 상기 전자서명과 함께 전송한 제1 인스턴트 공개키를 이용하여, 상기 전자서명을 검증하는 처리부

를 포함하고,

상기 제1 인스턴트 공개키는 상기 공개키 기반 암호화 알고리즘에 따른 인증 절차를 위해 일시적으로 생성되어 제2 인스턴트 공개키는 상기 제1 인스턴트 공개키와 상이하고,

각 인스턴트 개인키는 대응하여 생성된 인스턴트 공개키에 기초하여 계산되고,

상기 전자서명은 상기 상대 기기에서 제1 인스턴트 개인키, 상기 메시지 및 고정 개인키를 이용하여 상기 공개키 기반 암호화 알고리즘에 의해 생성되고,

상기 제1 인스턴트 개인키는 상기 상대 기기에서 상기 제1 인스턴트 공개키, 상기 고정 개인키 및 상기 공개키 기반 암호화 알고리즘을 이용해서 계산되는 인증 장치.

청구항 10

제9항에 있어서,

상기 제1 인스턴트 공개키가 3 이상의 홀수가 아니라면, 상기 제1 인스턴트 공개키를 유효하지 않은 것으로 판단하는 판단부

를 더 포함하는 인증 장치.

청구항 11

제9항에 있어서,

상기 제1 인스턴트 공개키가 반복 생성된 것이라면 상기 제1 인스턴트 공개키를 유효하지 않은 것으로 판단하는 판단부

를 더 포함하는 인증 장치.

청구항 12

적어도 하나의 프로세서를 포함하는 인증 장치에 있어서, 상기 인증 장치는 상기 적어도 하나의 프로세서에 의해 적어도 일시적으로 구현되는:

상대 기기로부터 수신되는 제1 인스턴트 공개키가 유효한 값인지 판단하는 판단부; 및

상기 제1 인스턴트 공개키가 유효한 값인 경우, 보유하고 있던 상기 상대 기기의 고정 공개키와 상기 제1 인스턴트 공개키를 이용하여, 전송할 데이터를 인코딩하는 처리부

를 포함하고,

상기 제1 인스턴트 공개키는 인증 절차를 위해 일시적으로 생성되어 제2 인스턴트 공개키는 상기 제1 인스턴트 공개키와 상이하고,

각 인스턴트 개인키는 대응하여 생성된 인스턴트 공개키에 기초하여 계산되고,

상기 제1 인스턴트 공개키는 전자서명에 대한 요청에 응답하여 상기 상대 기기에서 메시지에 대해 생성되고,

상기 전자서명은 상기 상대 기기에서 제1 인스턴트 개인키, 상기 메시지 및 고정 개인키를 이용하여 공개키 기반 암호화 알고리즘에 의해 생성되고,

상기 제1 인스턴트 개인키는 상기 상대 기기에서 상기 제1 인스턴트 공개키, 상기 고정 개인키 및 상기 공개키 기반 암호화 알고리즘을 이용해서 계산되는 인증 장치.

청구항 13

제12항에 있어서,

상기 제1 인스턴트 공개키는, 상기 상대 기기에서 랜덤 넘버 생성 과정을 통해 생성된 것인 인증 장치.

청구항 14

제12항에 있어서,

상기 판단부는, 상기 제1 인스턴트 공개키가 3 이상의 홀수가 아니라면, 상기 제1 인스턴트 공개키를 유효하지 않은 것으로 판단하는 인증 장치.

청구항 15

제12항에 있어서,

상기 판단부는, 상기 제1 인스턴트 공개키가 미리 지정된 재사용 횟수 이상 반복 사용되고 있는 경우라면, 상기 제1 인스턴트 공개키를 유효하지 않은 것으로 판단하는 인증 장치.

청구항 16

적어도 하나의 프로세서를 포함하고 공개키 기반 암호화 알고리즘에 따른 인증 절차를 수행하는 인증 장치에 있어서, 상기 인증 장치는 상기 적어도 하나의 프로세서에 의해 적어도 일시적으로 구현되는:

상기 인증 절차가 수행되어야 하는 것에 응답하여, 랜덤 넘버 생성 과정을 통해 메시지에 대한 제1 인스턴트 공개키를 생성하는 생성부;

상기 제1 인스턴트 공개키, 고정 개인키 및 상기 알고리즘을 이용해서, 상기 알고리즘에서 상기 제1 인스턴트 공개키와 쌍을 이루는 제1 인스턴트 개인키를 계산하는 계산부; 및

상기 제1 인스턴트 공개키, 상기 메시지 및 상기 고정 개인키를 전송 받은 상대 기기로부터, 상기 상대 기기에 미리 보유되는 고정 공개키와 상기 제1 인스턴트 공개키를 이용하여 인코딩한 메시지가 수신되면, 상기 제1 인스턴트 개인키를 이용하여 상기 메시지를 디코딩하는 처리부

를 포함하고,

상기 제1 인스턴트 공개키는 상기 인증 절차를 위해 일시적으로 생성되어 제2 인스턴트 공개키는 상기 제1 인스턴트 공개키와 상이하고,

각 인스턴트 개인키는 대응하여 생성된 인스턴트 공개키에 기초하여 계산되는 인증 장치.

청구항 17

제16항에 있어서,

상기 인증 장치는 무작위적으로 발생하는 공정 편차를 이용하여 하드웨어 핑거프린트를 제공하는 PUF(Physically Unclonable Function)를 더 포함하고,

상기 랜덤 넘버 생성 과정은, 상기 하드웨어 핑거프린트를 근원 값으로 이용하는 랜덤 넘버 생성 알고리즘을 포함하는 인증 장치.

청구항 18

제1항에 있어서,

상기 계산 결과에서 상기 제1 인스턴트 공개키와 쌍을 이루는 상기 제1 인스턴트 개인키가 존재하지 않는 것으로 판단되는 경우:

상기 생성부는 상기 제1 인스턴트 공개키와 상이한 제2 인스턴트 공개키를 생성하고; 상기 계산부는 상기 알고리즘에서 상기 제2 인스턴트 공개키와 쌍을 이루는 제2 인스턴트 개인키를 계산하는 인증 장치.

청구항 19

컴퓨터-판독가능 기록매체에 저장되는 비-일실시적 컴퓨터 프로그램에 있어서, 상기 프로그램은 프로세서를 포함하는 컴퓨팅 장치에서 실행되는 경우 상기 프로세서가:

메시지에 대응하는 전자 서명 생성 요청에 응답하여, 랜덤 넘버 생성 과정을 통해 메시지에 대한 제1 인스턴트 공개키를 생성하도록 하는 명령어 세트;

상기 제1 인스턴트 공개키, 고정 개인키 및 공개키 기반 암호화 알고리즘을 이용해서, 상기 제1 인스턴트 공개키와 쌍을 이루는 제1 인스턴트 개인키를 계산하도록 하는 명령어 세트; 및

상기 제1 인스턴트 개인키, 상기 메시지 및 상기 고정 개인키를 이용하여 상기 공개키 기반 암호화 알고리즘에 의한 전자서명을 생성하도록 하는 명령어 세트

를 포함하고,

상기 제1 인스턴트 공개키는 인증 절차를 위해 일시적으로 생성되어 제2 인스턴트 공개키는 상기 제1 인스턴트 공개키와 상이하고,

각 인스턴트 개인키는 대응하여 생성된 인스턴트 공개키에 기초하여 계산되는 컴퓨터-판독가능 기록매체에 저장되는 프로그램.

청구항 20

컴퓨터-판독가능 기록매체에 저장되는 비-일실시적 컴퓨터 프로그램에 있어서, 상기 프로그램은 프로세서를 포함하는 컴퓨팅 장치에서 실행되는 경우 상기 프로세서가:

상대 기기로부터 메시지 및 전자서명과 함께 수신된 상기 메시지의 제1 인스턴트 공개키가 유효한 지 판단하도록 하는 명령어 세트; 및

상기 제1 인스턴트 공개키가 유효한 경우, 상기 컴퓨팅 장치에 저장되어 있던 상기 상대 기기의 고정 공개키, 및 수신한 상기 제1 인스턴트 공개키를 이용하여 상기 전자서명을 검증하도록 하는 명령어 세트

를 포함하고,

상기 제1 인스턴트 공개키는 인증 절차를 위해 일시적으로 생성되어 제2 인스턴트 공개키는 상기 제1 인스턴트 공개키와 상이하고,

각 인스턴트 개인키는 대응하여 생성된 인스턴트 공개키에 기초하여 계산되고,

상기 전자서명은 상기 상대 기기에서 제1 인스턴트 개인키, 상기 메시지 및 고정 개인키를 이용하여 공개키 기반 암호화 알고리즘에 의해 생성되고,

상기 제1 인스턴트 개인키는 상기 상대 기기에서 상기 제1 인스턴트 공개키, 상기 고정 개인키 및 상기 공개키 기반 암호화 알고리즘을 이용해서 계산되는 컴퓨터-판독가능 기록매체에 저장되는 프로그램.

발명의 설명

기 술 분 야

[0001] 인증 장치 및 방법에 연관되며, 보다 특정하게는 공개키 암호 알고리즘 기반 인증 장치의 보안 공격에 대한 강인함 향상에 연관된다.

배 경 기 술

[0003] 기기(device) 인증이나 메시지 전자 서명에서 공개키 암호 알고리즘을 이용한 서명 기법이 사용된다. 예를 들어, RSA (Rivest Shamir Adleman) 알고리즘은 공개키와 개인키를 세트로 만들어서 암호화와 복호화를 수행하는 인터넷 암호화 및 인증 방법으로, 개인키는 기기에, 공개키는 인증기관과 같은 상대 기기에 전달되어 보관된다.

[0004] 한편, RSA와 같은 공개키 알고리즘 기반에서 개인키가 노출/유출되는 경우 서명을 위조할 수 있으므로 개인키는 보안 공격, 이를테면 부채널 공격(side channel attack)의 목표가 되고 있다. 부채널 공격 방법 중, 대량의 데이터를 수집하여 통계적으로 분석하는 DPA(Differential Power Analysis) 공격은 매우 강력한 것으로 알려져 있다.

[0005] 한편, PUF (Physically Unclonable Function)는 예측 불가능한 (unpredictable) 디지털 값을 제공할 수 있다. 개개의 PUF들은 정확한 제조 공정이 주어지고, 동일한 공정에서 제조되더라도, 상기 개개의 PUF들이 제공하는 디지털 값은 다르다. PUF는 복제가 불가능한 POWF (Physical One-Way Function practically impossible to be duplicated)로 지칭될 수도 있다.

[0006] 이러한 PUF의 복제 불가능한 특성은 보안 및/또는 인증을 위한 기기의 식별자(identifier)를 생성하는 데에 이용될 수 있다. 이를테면, 디바이스를 다른 디바이스와 구별하기 위한 유니크 키(unique key to distinguish devices from one another)를 제공하기 위해 PUF가 이용될 수 있다.

[0007] 한국 등록특허 10-1139630호(이하 '630 특허)에서 PUF를 구현하는 방법이 제시된 바 있다. '630 특허에서는 반도체의 공정 편차(process variation)를 이용하여 반도체의 전도성 레이어들 사이의 인터-레이어 콘택(inter-layer contact) 또는 비아(via)의 생성 여부가 확률적으로 결정되도록 한 방법이 제시되었다.

발명의 내용

해결하려는 과제

과제의 해결 수단

[0009] 실시예들에 따르면, 부채널 공격에 강인한 인증 장치 및 인증 방법이 제시된다. 이를테면, 공개키 기반 알고리즘을 이용하면서도 DPA 공격을 불가능하고 또한 무의미하게 만드는 인증 장치 및 방법이 제시된다. 실시예들에 따르면 공개키-개인키 쌍이 고정된 값으로 되어 반복적으로 사용되지 않고, 대신 인증이 필요한 경우에 인스턴트하게(instantly) 생성되어 사용될 수 있다.

[0010] 일측에 따르면, 적어도 하나의 프로세서를 포함하고 공개키 기반 암호화 알고리즘(encryption algorithm using public key), 이를테면 비대칭키 암호화 알고리즘(asymmetric keys encryption algorithm)인 RSA 에 따른 인증 절차를 수행하는 인증 장치가 제공된다. 장치에는 기 생성된 고정 개인키 p, q를 가지고 있다. 일실시예에 따르면 장치는: 상기 알고리즘에 대응하는 전자 서명이 필요한 경우에 응답하여, 제1 인스턴트 공개키를 생성하는 생성부; 상기 고정 개인키와 상기 제1 인스턴트 공개키를 이용해서, 상기 알고리즘에서 상기 제1 인스턴트 공개키와 쌍을 이루는 제1 인스턴트 개인키를 계산하는 계산부; 및 상기 제1 인스턴트 개인키를 이용하여 상기 알고

리즘에 의한 전자서명을 생성하는 처리부를 포함할 수 있다. 상기 생성부, 상기 계산부 및 상기 처리부는 상기 적어도 하나의 프로세서에 의해 적어도 일시적으로 구현될 수 있다. 일실시예에 따르면 장치는 전송할 메시지를 상기 전자서명 및 상기 제1 인스턴트 공개키와 함께 상대 기기에 전송하는 통신부를 더 포함한다.

[0011] 일실시예에 따르면 장치는 공개키 키 생성 알고리즘을 통해 상기 고정 개인키를 생성하여 보유하고 있다. 상기 고정 개인키는 상기 제1 인스턴트 공개키를 이용해서 상기 제1 인스턴트 개인키를 계산할 때와 사용에 앞서 발급 과정에서 다른 기기로 전달되어 추후 서명 검증에 함께 사용되는 고정 공개키를 생성할 때 이용된다. 예시적으로, 그러나 한정되지 않게, 상기 인증 장치에는 무작위적으로 발생하는 공정 편차를 이용하여 하드웨어 핑거프린트를 제공하는 PUF가 더 포함될 수 있다. 일실시예에 따르면 이 PUF의 값이 상기 고정 개인키 값으로 직접 또는 간접적으로 사용할 수 있다. 이 경우에는 메모리에 상기 고정 개인키를 직접적으로 저장할 필요가 없어 물리적 공격으로부터 상기 고정 개인키를 보호할 수 있다. 이는 상기 고정 개인키가 해당 기기 내에만 존재한다는 점이 강하게 보장되기 때문에 다른 기기에서 인스턴트 공개키-개인키 쌍을 임의로 만들 수 없음 또한 보장된다. 또한 다른 실시예에서는, 상기 PUF의 값이 인스턴트 공개키들을 만드는 데에 직접 또는 간접적으로 이용될 수도 있다. 이를테면 랜덤 넘버 생성 과정에서의 시드 값 또는 근원 값으로 이용될 수 있다.. 이 경우, PUF를 랜덤 넘버 생성 알고리즘의 근원 값으로 사용함으로써, 기기 마다 생성되는 랜덤 넘버 생성 결과가 서로 독립적이고 또한 값이 상이해지도록 하는 효과를 추가적으로 기대할 수 있다.

[0012] 한편 상기 상대 기기로부터 재전송 요청이 수신되는 것에 응답하여 아래와 같이 대응할 수 있다. 단순 통신 오류로 인한 재전송 요청의 경우 생성했던 메시지를 재전송 할 수 있다. 그러나 그 외의 경우, 이를테면 잘못된 서명, 또는 공격이 있는 것으로 의심되는 경우, 제 1 인스턴트 공개키를 폐기하고, 이와 상이한 새로운 제2 인스턴트 공개키를 생성하여 다시 전자서명을 생성하여 전송한다.

[0013] 일실시예에 따르면 상기 계산 결과에서 상기 제1 인스턴트 공개키와 쌍을 이루는 상기 제1 인스턴트 개인키가 존재하지 않는 것으로 판단되는 경우에, 상기 생성부는 상기 제1 인스턴트 공개키와 상이한 제2 인스턴트 공개키를 생성한다. 그러면 상기 계산부는 상기 알고리즘에서 상기 제2 인스턴트 공개키와 쌍을 이루는 제2 인스턴트 개인키를 계산한다. 한편, 상기 생성부는 상기 제2 인스턴트 공개키를 생성하기 위해 상기 랜덤 넘버 생성 과정을 수행하는 대신 상기 제1 인스턴트 공개키에 정수 2를 더한 수를 상기 제2 인스턴트 공개키로 결정하여 제공할 수도 있다. 제1 인스턴트 공개키에 더해지는 "2"라는 정수는 예시적인 것이므로, 다른 값으로 변경될 수 있다. 만약 상기 암호화 알고리즘이 RSA-CRT (Chinese Remainder Theorem) 알고리즘인 경우, 상기 계산부가 계산하는 상기 제1 인스턴트 개인키는 제1 dP 값 및 제1 dQ 값을 포함하는 것이다. 이 RSA-CRT 실시예에서 상기 계산 결과에 따른 상기 제1 dP 값 및 상기 제1 dQ 값 중 어느 하나라도 존재하지 않는 것으로 판단되는 경우라면, 상기 생성부는 상기 제1 인스턴트 공개키와 상이한 제2 인스턴트 공개키를 생성한다. 그리고 상기 계산부는 상기 알고리즘에서 상기 제2 인스턴트 공개키와 쌍을 이루는 제2 dP 값 및 제2 dQ 값을 계산한다.

[0014] 다른 일측에 따르면, 적어도 하나의 프로세서를 포함하고 공개키 기반 암호화 알고리즘에 따라 상대 기기가 전송한 전자서명을 검증하는 인증 장치, 이를테면 인증 기관(CA: Certification Authority)가 제공된다. 장치는, 보유하고 있던 상기 상대 기기의 고정 공개키, 및 상기 상대 기기가 인스턴트하게 생성하여 상기 전자서명과 함께 전송한 제1 인스턴트 공개키를 이용하여, 상기 전자서명을 검증하는 처리부를 포함할 수 있다.

[0015] 일실시예에 따르면 장치는, 상기 제1 인스턴트 공개키가 3 이상의 홀수가 아니라면, 상기 제1 인스턴트 공개키를 유효하지 않은 것으로 판단하는 판단부를 더 포함할 수 있다. 한편, 상기 처리부와 상기 판단부는 상기 적어도 하나의 프로세서에 의해 적어도 일시적으로 구현될 수 있다.

[0016] 또 다른 일측에 따르면, 적어도 하나의 프로세서를 포함하는 인증 장치에 있어서, 상대 기기로부터 수신되는 제1 인스턴트 공개키가 유효한 값인지 판단하는 판단부; 및 상기 제1 인스턴트 공개키가 유효한 값인 경우, 보유하고 있던 상기 상대 기기의 고정 공개키와 상기 제1 인스턴트 공개키를 이용하여, 전송할 데이터를 인코딩하는 처리부를 포함하는 인증 장치가 제공된다. 상기 판단부와 상기 처리부는 상기 적어도 하나의 프로세서에 의해 적어도 일시적으로 구현될 수 있다. 예시적으로, 그러나 한정되지 않게, 상기 제1 인스턴트 공개키는 상기 상대 기기에서 랜덤 넘버 생성 과정을 통해 생성된 것일 수 있다.

[0017] 일실시예에 따르면 상기 제1 인스턴트 공개키가 3 이상의 홀수가 아니라면, 상기 판단부가 상기 제1 인스턴트 공개키를 유효하지 않은 것이라고 판단할 수 있다. 또 다른 일측에 따르면, 적어도 하나의 프로세서를 포함하고 공개키 기반 암호화 알고리즘에 따른 인증 절차를 수행하는 인증 장치가 제공된다. 장치는: 상기 인증 절차가 수행되어야 하는 것에 응답하여, 랜덤 넘버 생성 과정을 통해 제1 인스턴트 공개키를 생성하는 생성부; 상기 제1 인스턴트 공개키를 이용해서, 상기 알고리즘에서 상기 제1 인스턴트 공개키와 쌍을 이루는 제1 인스턴트 개

인키를 계산하는 계산부; 및 상기 제1 인스턴트 공개키를 전송 받은 상대 기기로부터, 상기 상대 기기에 미리 보유되는 고정 공개키와 상기 제1 인스턴트 공개키를 이용하여 인코딩한 메시지가 수신되면, 상기 제1 인스턴트 개인키를 이용하여 상기 메시지를 디코딩하는 처리부를 포함할 수 있다. 상기 생성부, 상기 계산부, 상기 처리부는 상기 적어도 하나의 프로세서에 의해 적어도 일시적으로 구현될 수 있다.

[0018] 일실시예에 따르면 상기 인증 장치에는 무작위적으로 발생하는 공정 편차를 이용하여 하드웨어 핑거프린트를 제공하는 PUF가 더 포함될 수 있다. 이 경우, 상기 랜덤 넘버 생성 과정은, 상기 하드웨어 핑거프린트 값을 근원 값으로 이용하는 랜덤 넘버 생성 알고리즘을 포함한다.

[0019] 일실시예에 따르면, 상기 계산 결과에서 상기 제1 인스턴트 공개키와 쌍을 이루는 상기 제1 인스턴트 개인키가 존재하지 않는 것으로 판단되는 경우에, 상기 생성부는 상기 제1 인스턴트 공개키와 상이한 제2 인스턴트 공개키를 생성한다. 그리고 상기 계산부는 상기 알고리즘에서 상기 제2 인스턴트 공개키와 쌍을 이루는 제2 인스턴트 개인키를 계산할 수 있다.

[0020] 또 다른 일측에 따르면, 컴퓨터-판독가능 기록매체에 저장되는 비-일실시적 컴퓨터 프로그램이 제공된다. 상기 프로그램은 프로세서를 포함하는 컴퓨팅 장치에서 실행되는 경우 상기 프로세서가 동작하도록 하는 이하의 명령어 세트들을 포함할 수 있다. 명령어 세트들은: 상기 프로세서가 메시지에 대응하는 전자 서명 생성 요청에 응답하여, 랜덤 넘버 생성 과정을 통해 제1 인스턴트 공개키를 생성하도록 하는 명령어 세트; 상기 프로세서가 상기 제1 인스턴트 공개키를 이용해서, 상기 제1 인스턴트 공개키와 쌍을 이루는 제1 인스턴트 개인키를 계산하도록 하는 명령어 세트; 및 상기 프로세서가 상기 제1 인스턴트 개인키를 이용하여 공개키-알고리즘에 의한 전자서명을 생성하도록 하는 명령어 세트를 포함할 수 있다.

[0021] 또 다른 일측에 따르면, 컴퓨터-판독가능 기록매체에 저장되는 비-일실시적 컴퓨터 프로그램이 제공된다. 상기 프로그램은 프로세서를 포함하는 컴퓨팅 장치에서 실행되는 경우 상기 프로세서가 동작하도록 하는 이하의 명령어 세트들을 포함할 수 있다. 명령어 세트들은: 상기 프로세서가 상대 기기로부터 메시지 및 전자서명과 함께 수신된 제1 인스턴트 공개키가 유효한 지 판단하도록 하는 명령어 세트; 및 상기 프로세서가 상기 제1 인스턴트 공개키가 유효한 경우, 상기 컴퓨팅 장치에 저장되어 있던 상기 상대 기기의 고정 공개키, 및 수신한 상기 제1 인스턴트 공개키를 이용하여 상기 전자서명을 검증하도록 하는 명령어 세트를 포함할 수 있다.

도면의 간단한 설명

[0023] 도 1은 일실시예에 따라 인스턴트 공개키와 인스턴트 개인키를 제공하는 인증 장치를 도시하는 블록도이다.

도 2는 일실시예에 따라 상대방이 생성한 인스턴트 공개키를 전달 받아 사용하는 인증 장치를 도시하는 블록도이다.

도 3은 일실시예에 따른 전자 서명 검증 과정을 도시하는 흐름도이다.

도 4는 일실시예에 따른 인코딩 및 디코딩 과정을 도시하는 흐름도이다.

도 5는 일실시예에 따라 인스턴트 개인키가 존재하지 않는 경우의 처리를 도시하는 흐름도이다.

도 6은 일실시예에 따라 RSA-CRT 알고리즘 적용 시 개인키가 존재하지 않는 경우의 처리를 도시하는 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0024] 이하에서, 실시예들을 첨부된 도면을 참조하여 상세하게 설명한다. 그러나, 권리범위는 이러한 실시예들에 의해 제한되거나 한정되는 것은 아니다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.

[0025] 아래 설명에서 사용되는 용어는, 연관되는 기술 분야에서 일반적이고 보편적인 것으로 선택되었으나, 기술의 발달 및/또는 변화, 관례, 기술자의 선호 등에 따라 다른 용어가 있을 수 있다. 따라서, 아래 설명에서 사용되는 용어는 기술적 사상을 한정하는 것으로 이해되어서는 안 되며, 실시예들을 설명하기 위한 예시적 용어로 이해되어야 한다.

[0026] 또한 특정한 경우는 출원인이 임의로 선정한 용어도 있으며, 이 경우 해당되는 설명 부분에서 상세한 그 의미를 기재할 것이다. 따라서 아래 설명에서 사용되는 용어는 단순한 용어의 명칭이 아닌 그 용어가 가지는 의미와 명세서 전반에 걸친 내용을 토대로 이해되어야 한다.

[0028] 도 1은 일실시예에 따라 인스턴트 공개키와 인스턴트 개인키를 제공하는 인증 장치를 도시하는 블록도이다. 장치(100)는 적어도 하나의 프로세서를 포함하는 컴퓨팅 단말의 적어도 일부분일 수 있다. 이 컴퓨팅 단말은, 예시적으로 그러나 한정되지 않게, 스마트폰, 태블릿, 랩탑, 범용 컴퓨터, 서버, 또는 암호화 통신용 전용 단말기 등일 수 있다. 장치(100)는 공개키 기반 암호화 알고리즘, 이를테면 RSA와 같은 비대칭키 암호화 알고리즘에 따른 인증 절차를 수행한다.

[0029] 일실시예에 따르면 장치(100)의 생성부(110)는 RSA 알고리즘에 대응하는 전자 서명이 필요한 경우에 응답하여, 제1 인스턴트 공개키(instant public-key) "E"를 생성한다. 이러한 제1 인스턴트 공개키 E의 생성은 랜덤 넘버 생성(random number generating) 과정을 통해 수행될 수 있다. 한편, 생성되는 E는 3이상의 홀수이어야 한다. 소수(prime) 또는 매우 큰 수인 경우에도 의사-소수(pseudo-prime)이어야 하기 때문이다. 생성부(110)는 생성된 E가 홀수인지와, 그 값이 1이 아닌지 확인할 수 있다.

[0030] 한편 일실시예에 따르면 장치(100)에는 무작위적으로 발생하는 반도체 공정 편차(process variation)를 이용하여 하드웨어 핑거프린트(hardware finger-print)를 제공하는 PUF (미도시)가 포함되어 있을 수 있다. 반도체 공정 편차를 이용하여 PUF를 구현하는 예에도 여러 가지가 있다. 예시적으로 반도체의 전도성 레이어들 사이에 배치되는 비아(via)들 또는 인터-레이어 컨택(inter-layer contact)들의 무작위적 형성-형성실패 결과를 이용하여 PUF가 구현될 수 있으며, 이를 자세히 공개한 '630 특허의 명세서가, 참고로서 이 명세서에 인용된다.

[0031] 한편 이러한 예시적 상황에서, PUF가 제공하는 하드웨어 핑거프린트는 상기 고정 개인키 생성 알고리즘에 이용될 수 있다. 장치가 기본적으로 가지고 있는(301) 고정 개인키(p, q)는 소수로, 이를 생성할 때 고정 개인키(p, q)의 후보 값으로 하드웨어 핑거프린트가 이용될 수 있다. 이 경우 하드웨어 핑거프린트 값과 실제 생성된 고정 개인키(p, q) 값의 차이만 메모리에 기록하면 된다. 기록되는 값은 32-bit로 충분하므로 이 값만 가지고 원래 고정 개인키(p, q)을 추측하기는 불가능하다. 또한, 필요 시 하드웨어 핑거프린트 값으로부터 고정 개인키 값을 빠르게 재생산하는 것도 가능하다.

[0032] 한편 이러한 예시적 상황에서, PUF가 제공하는 하드웨어 핑거프린트는 상기 랜덤 넘버 생성 알고리즘에 이용될 수도 있다. 이를테면 소프트웨어 및/또는 하드웨어 적으로 동작하는 랜덤 넘버 알고리즘에서 요구되는 근원 값, 또는 시드 키(seed key)로 하드웨어 핑거프린트가 이용될 수 있다.

수학식 1

[0033] $$E = random( )$$

[0034] 위 수학식 1에 의해 제1 인스턴트 공개키 E가 생성되면, 계산부(120)는 RSA 방식에서 규정된 계산 방식에 따라 알고리즘에 따라 제1 인스턴트 공개키 E를 이용해서, E의 RSA 쌍인, 다시 말해 E와 쌍을 이루는 제1 인스턴트 개인키(instant private-key) "D"를 계산해 낸다.

수학식 2

[0035] $$D = E^{-1} \bmod (p-1)(q-1)$$

[0036] 이렇게 RSA 알고리즘에서 키 발급 과정의 대상인, 개인키-공개키를 즉석에서 생성 및 계산하는 것은 아래와 같은 장점이 있다. RSA 방식에서 개인키는 {D, p, q}가 있으며, 상대방 기기에 제공되는 공개키는 {E, N}이 있다. 여기서 N은 p와 q의 곱이며, D와 E는 $D \times E = 1 \bmod (p-1)(q-1)$인 관계가 성립한다. 다른 기기나 인증기관으로 전달되는 E나 N과 달리 인증 장치 내부에만 존재하게 되는(그리고 존재해야만 하는) 개인키 {D, p, q}는 보안 공격의 대상이 된다. D를 제3자가 알게 되면, 직접적인 전자서명 위조가 가능하다. 그리고 D를 모르더라도 p 및/또는 q 값을 제3자가 알면, 외부에 공개되어 있는 E와 함께 사용하여 D를 계산해낼 수도 있기 때문에, 이 RSA 암호화는 더 이상 가치가 없게 된다.

[0037] 앞서 언급한 바와 같이, 부채널 공격, 이를테면 DPA (Differential Power Analysis, 차분 전력 분석)은 기기의

소모 전력 파형을 분석하는데, 하나의 전류 소모 파형만 관찰하는 SPA (Simple Power Analysis) 공격과 달리, 실제 키가 사용된 파형을 다수 수집 후 통계적 분석 방법에 의해 이 D, p 또는 q를 획득하려 하게 된다. 노이즈가 심한 환경이나, 전력 소모를 쉴드하여 관찰이 어렵도록 한 경우에 SPA 공격은 성공적이지 못할 수 있지만, 여전히 DPA 공격은 알고리즘의 안전성에 위협적이다. 다수의 파형을 통한 통계적 분석 과정에서 노이즈 등 기타 아웃라이어는 상쇄될 수 있고, 유의미한 패턴이 발견될 수 있기 때문이다.

[0038] 종래의 RSA 암호 알고리즘에서는 인증 장치가 고정된 개인키인 {D, p, q}를 가지고 있고, 상대방인 인증기관이 이 인증 장치의 고정된 공개키인 {E, N}을 가지고 있었다. 따라서 공격자가 기기에게 전자서명을 반복적으로 재요청하는 식으로 DPA에 필요한 다수의 파형을 수집하는 것이 가능하기에 위험이 있었다.

[0039] 그러나 상기 실시예에 따르면, 이 개인키와 공개키 중 적어도 D와 E가 고정된 값이 아닌 동적으로 변하는 값이 된다. 다시 말해, 전자서명과 같은 암호화 알고리즘 수행이 필요한 경우에 즉석에서(instantly) 잠시 사용하는 D와 E를 생성하는 것이다. 생성된 D와 E는 한 번(one time)만 사용되기 때문에 공격자에게 전자서명이 수집된다 하더라도 통계적으로 무의미 한 것이 된다. 실시예들은 이렇게 고정되지 않고 동적으로 변경되는 E와 D를 생성하여 사용함으로써 DPA 공격 등 부채널 공격을 원천적으로 방지하게 된다.

[0040] 더욱이, 앞서 설명한 바와 같이, 랜덤하게 제1 인스턴트 공개 키 E를 생성하는 데에 이용되는 PUF의 하드웨어 핑거프린트는 동일하게 설계되더라도 그 값이 달라서 물리적으로 복제한다는 것이 불가능하다. 그리고 PUF 값이나 구조도 외부에서 관찰 및 분석이 거의 불가능하다. 따라서, 이러한 실시예에 따르면 PUF값을 시드로 하여 랜덤한 E를 생성하는 알고리즘 자체를 공격하는 것도 방지된다.

[0041] 이렇게 제1 인스턴트 공개키 E가 생성되고, 이를 이용하여 E의 쌍인 제1 인스턴트 개인키 D가 계산되게 되는데, 경우에 따라서는 위 수학식 2를 만족하는 D가 존재하지 않을 수도 있다.

[0042] D가 존재하지 않는다면, 유효한 "인스턴트 D - 인스턴트 E의 쌍"이 만들어질 수 없으므로, 생성부(110)이 앞선 제1 인스턴트 공개키 E와 다른 제2 인스턴트 공개키 E'을 생성하게 된다. 이러한 제2 인스턴트 공개키 E'을 가지고 계산부(120)가 수학식 2에 따라 제2 인스턴트 개인키 D'을 계산해 본다. D'이 존재한다면 E'과 D'을 사용하고, 그렇지 않으면 이 과정을 반복할 수 있다.

[0043] 여기서, 생성부(110)가 제2 인스턴트 공개키 E'을 만드는 과정은 제1 인스턴트 공개키 E를 만들었던 것과 유사하게 다시 랜덤 넘버 생성 과정을 거치는 것일 수도 있다. 그러나 이러한 과정보다 더 간단한 다른 실시예도 가능하다. 이를테면, 생성부(110)는 다시 랜덤 넘버 생성 과정을 수행하는 대신, 생성되었으나 대응하는 D가 존재하지 않았던 제1 인스턴트 공개키 E에 정수 2를 더한 수 (다시 말해 E 바로 위의 홀수)를 제2 인스턴트 공개키 E'이라고 결정하여 제공할 수도 있다. 여기서 E'을 생성하기 위해 E에 더해지는 "2"라는 정수는 예시적인 것이므로, 다른 값으로 변경될 수 있다. D의 존재 여부에 따라 E를 다시 생성하는 것은 도 5를 참조하여 다시 설명할 것이다.

[0044] 한편, 상기 암호화 알고리즘이 CRT (Chinese Remainder Theorem)를 이용하는 알고리즘일 수도 있는데, 이 경우에는 E를 가지고 계산되어야 하는 인스턴트 개인키는 D 하나가 아니라 아래 수학식 3과 수학식 4에 따라 계산되는 {dP, dQ}일 수 있다.

수학식 3

[0045] $$dP = E^{-1} \bmod (p-1)$$

수학식 4

[0046] $$dQ = E^{-1} \bmod (q-1)$$

[0047] 이 RSA-CRT 실시예에서, 계산부(120)의 계산에 따라 E에 대응하는 유효한 dP 값 및 유효한 dQ 값 중 어느 하나라도 존재하지 않는 것으로 판단되면 마찬가지로 E를 다시 생성해야 할 수 있다. 이 경우도 랜덤 넘버 생성 과

정을 다시 거칠 수도 있으나, E에 2를 더해서 간단히 E'을 결정하고, 이 E'에 대응하는 dP' 값 및 dQ' 값을 계산할 수도 있다. 자세한 내용은 도 6을 참조하여 다시 설명할 것이다.

[0048] 이상의 과정을 통해 유효한 인스턴트 공개키(E)와 인스턴트 개인키(D 또는 {dP, dQ})가 결정되면 이를 통하여 RSA 방식의 암호화 알고리즘이 수행될 수 있다. 메시지 M을 서명하여 전자서명 S를 생성한다면 처리부(130)가 아래와 같은 과정을 수행할 수 있다.

수학식 5

[0049] $$S = GenSign(M, D, p, q)$$

[0050] GenSign()은 전자서명 생성 함수로서, RSA 알고리즘에 따라 규정되어 있다. 통신부(140)은 생성된 전자서명 S를 메시지 M과 함께 상대방 기기 (이를 테면 인증 기관)에 전송하되, 종래와는 다르게 위에서 생성한 인스턴트 공개키 E도 함께 전송한다. 그러면 상대방 기기는 이 인스턴트 공개키 E를 이미 보유하고 있던 장치(100)의 고정 공개키 N과 함께 사용하여 전자서명 S를 검증할 것이다. 이상의 과정은 도 3을 참조하여 다시 설명될 것이다.

[0051] 한편, 이렇게 전송된 인스턴트 공개키 E를 이용하여 상대방 기기가 메시지를 인코딩해서 보내온다면 장치(100)은 가지고 있던 인스턴트 개인키 D (또는 dP, dQ)를 p, q와 함께 이용하여 메시지를 디코딩할 수 있다. 이러한 과정이 도 4를 참조하여 후술된다.

[0052] 이처럼 실시예들에 따라 보안 공격의 위험을 상당히 낮추었으나, 예상할 수 있는 몇 가지의 또 다른 공격 시도들이 있을 수 있고, 이러한 다른 공격들은 아래와 같은 실시예들에 의해 구현될 수 있다. 만약 공격자가 매우 짧은 시간에 여러 번 키의 재전송을 요구할 수 있다. 일실시예에 따른 장치(100)는 통신오류의 경우 인스턴트 개인키를 재사용하여 서명을 재생성하지 않고 이미 생성된 생성을 재전송하기만 한다. 그리고 통신 오류가 아닌 경우 기 생성된 인스턴트 공개키를 폐기하고, 새로운 인스턴트 공개키를 생성할 수 있다. 통신 오류인지 여부는 전송 메시지에 대한 통신 Ack가 수신 여부로 확인할 수 있다. 상대방 기기 쪽에서 수행할 수 있는 공격 방지 대안들은 도 2를 참고하여 보다 상세히 설명할 것이다.

[0054] 도 2는 일실시예에 따라 상대방이 생성한 인스턴트 공개키를 전달 받아 사용하는 인증 장치를 도시하는 블록도이다. 장치(200)은 이를 테면 인증 기관(CA: Certification Authority)일 수 있다. 그리고 장치(200)는 적어도 하나의 프로세서를 포함하는 컴퓨팅 단말의 적어도 일부분일 수 있고, 후술하는 판단부(210)와 처리부(220)는 상기 프로세서에 의해 적어도 일시적으로 구현될 수 있다.

[0055] 일실시예에 따른 장치(200)는, 보유하고 있던 상기 상대 기기의 고정 공개키 N과, 도 1에서 설명한 것과 같이 상대 기기가 인스턴트하게 생성하여 메시지 M, 전자서명 S와 함께 전송한 제1 인스턴트 공개키 E를 이용하여, 전자서명 S를 검증하는 처리부(220)를 포함할 수 있다.

[0056] 메시지를 인코딩하여 상대 기기에게 보내야 하는 경우에, 처리부(220)는 보유하고 있던 그 상대 기기의 고정 공개키 N과, 앞서 전달 받은 제1 인스턴트 공개키 E를 이용하여, 전송할 데이터 M을 인코딩한다. M이 정상적으로 인코딩된 M'이 상대 기기로 전송된다면, 상대 기기는 자신의 고정 개인키 p, q와 인스턴트 개인키 D (또는 dP와 dQ)를 이용하여 M'을 M으로 디코딩 할 것이다.

[0057] 한편, 일실시예에 따르면 장치(200)는, 상대 기기가 보내온 인스턴트 공개키 E가 유효한 것인지 판단하는 판단부(210)을 더 포함할 수도 있다. 공격자가 E와 D를 모두 1로 하여 전자서명을 생성하고 이를 전달하는 공격을 예상해 볼 수 있다. 이 경우 전자서명 생성 및 검증 연산은 실제로 이루어지지 않는다. E와 D는 RSA 연산에서 지수 값(exponential value)로서 사용되기 때문에 값이 1이라면 인코딩-디코딩을 위해 메시지의 "1"제곱을 하는 것은 연산을 하지 않는 것과 동일하기 때문이다. 이 때 인증기관 CA 입장에서 보면 서명 검증 결과가 유효하게 나올 수도 있게 된다. 따라서 이러한 상황을 방지하기 위해 판단부(210)는 상대 기기로부터 전달받은 인스턴트 공개키 E가 3이상의 홀수 인지 확인한다. 그래서 E가 짝수라거나, E가 1이라면 유효하지 않은 E라고 판단할 수 있다.

[0058] 나아가, 다른 일실시예에 따른 판단부(210)는 E가 지정된 재사용 횟수 이상 반복하여 사용되고 있는 경우에, 이

를 비정상적인 상황으로 인식하여 인스턴트 공개키 E를 유효하지 않은 것으로 취급할 수도 있다.

[0060] 도 3은 일실시예에 따른 전자 서명 검증 과정을 도시하는 흐름도이다. 상술한 바와 같이, 종래의 RSA와 차이점은 기기가 고정된 개인키 D를 가지고 있지 않고, 상대 인증기관도 이 기기의 고정된 공개키 E를 가지고 있지 않다는 점이다. 따라서, 실시예에 따르면 도 1 내지 도 2를 참조하여 상술한 대로 인증이 필요한 경우에 인스턴트 공개키 E와 이에 의해 계산되는 인스턴트 개인키 D가 생성되어 활용된다.

[0061] 도시된 실시예에서, 기기는 저장소(301)에 p와 q만을 가지고 있다. 그리고 인증기관은 이 기기의 고정 공개키 중 N만을 가지고 있다. 인증이 필요한 경우, 단계(310)이 수행된다. 단계(310)은 랜덤 넘버 생성 과정을 통해 랜덤한 인스턴트 공개키 E가 생성되는 과정으로, 생성되는 E는 3 이상의 홀수이다. 인스턴트 E의 생성 및 그 유효성 검증에 대해서는 도 1을 참조하여 설명한 바와 같다.

[0062] 그러면 단계(320)에서 인스턴트 공개키 E의 RSA 쌍인 D가 계산된다. RSA에서 규정된 방식에 의하므로 이 분야의 통상의 기술자는 이러한 D의 계산을 쉽게 이해할 수 있다. 그러면 단계(330)에서 전송할 메시지 M에 대한 전자서명 S가 생성된다. 그리고, 단계(340)에서 메시지 M, 전자서명 S와 함께 상기 인스턴트 공개키 E가 인증기관으로 전송된다. 그러면 인증기관은 전자서명 검증 알고리즘을 수행하는 단계(350)에서, 앞서 전달 받은 E를 미리 가지고 있던 N과 함께 사용할 것이다. 단계(350)의 수행에 앞서서 인증기관은 전달 받은 E가 유효한지를 검증하는 과정을 거칠 수도 있으며, 이는 도 2를 참조하여 상세히 설명한 바 있다.

[0064] 도 4는 일실시예에 따른 인코딩 및 디코딩 과정을 도시하는 흐름도이다. 인증기관이 메시지 M을 인코딩하여 기기에게 보내는 과정이 도시되어 있다. 인증기관은 단계(410)에서 미리 보유하고 있던 그 상대 기기의 고정 공개키 N과, 앞서 전달 받은 인스턴트 공개키 E를 이용하여, 전송할 데이터 M을 인코딩한다. 그리고 단계(420)에서 인코딩된 메시지 M'이 기기로 전송되고, 기기는 단계(430)에서 자신의 고정 개인키 p, q와 인스턴트 개인키 D (또는 dP와 dQ)를 이용하여 M'을 M으로 디코딩 할 수 있다.

[0066] 도 5는 일실시예에 따라 인스턴트 개인키가 존재하지 않는 경우의 처리를 도시하는 흐름도이다. 단계(510)에서 기기의 생성부가 랜덤 넘버 생성 과정을 통해 인스턴트 공개기 E를 생성한다. 그리고 단계(520)에서 기기의 계산부가 E를 이용하여 수학식 2에 따라 E의 RSA 쌍인 인스턴트 개인키 D를 계산해 본다. 단계(530)에서 이 유효한 D가 존재하는 경우라면 도 1 내지 도 4를 참조하여 설명한 바와 같이 인스턴트 D와 인스턴트 E를 사용한 인증 과정이 수행된다.

[0067] 그러나 단계(530)의 판단에서 D가 존재하지 않는 경우라면 기기의 생성부는 앞선 인스턴트 공개키 E와 다른 제2 인스턴트 공개키 E'을 생성한다. 이러한 과정은 다시 랜덤 넘버 생성 과정을 거치는 것일 수도 있으나 도 5에서 도시된 예에서는 보다 간단한 실시예가 제시되었다. 단계(540)에서 앞서 생성되었으나 D가 존재하지 않아 사용될 수 없는 인스턴트 공개키 E에 정수 2를 더하는 것이다. 이러한 과정이 유효한 E-D 쌍이 있을 때까지 반복된다.

[0069] 도 6은 일실시예에 따라 RSA-CRT 알고리즘 적용 시 개인키가 존재하지 않는 경우의 처리를 도시하는 흐름도이다. 먼저 CRT를 사용하지 않는 앞의 실시예의 경우 서명을 생성할 때와 CRT를 사용하는 경우 서명을 생성할 때를 비교하면 아래 표와 같다.

표 1

[0070]

CRT 사용 하지 않는 경우	CRT 사용하는 경우
$D = E^{-1} \bmod (p-1)(q-1)$ $N = pq$ $S = M''^{D} \bmod N$	$dP = E^{-1} \bmod (p-1)$ $dQ = E^{-1} \bmod (q-1)$ $qInv = q^{-1} \bmod p$ $s_1 = M''^{dp} \bmod p$ $s_2 = M''^{dq} \bmod q$ $h = qInv \times (s_1 - s_2) \bmod p$ $S = s_2 + h \times q$

[0071] 위 표 1에서 M''은 RSA 서명 포맷에 맞게 메시지 M의 해쉬값에 패딩 등을 추가한 값이다. 표 1에서 확인할 수

있듯이 CRT를 사용할 경우 식이 복잡해지나 RSA 연산에서 연산양의 상당 부분을 차지하는 "modular exponentiation"의 처리 데이터 길이가 1/2이 되기 때문에, CRT를 사용하지 않는 경우에 비해 오히려 연산 속도는 4배 내지 8배 가량 빨라질 수 있다.

[0072] 랜덤한 인스턴트 공개키 E가 생성된 후, CRT가 사용되는 경우이면(단계 610), 단계(620)에서 dP가 계산되고, 이 dP가 존재하는지 판단된다(630). dQ의 존재 여부와 관계 없이 이 dP가 존재하지 않는다면 E는 사용될 수 없다. 따라서 단계(631)에서 E에 2를 더한 수를 다시 E로 지정하고 dP를 계산하는 과정이 이 dP가 존재할 때까지 반복된다. 물론 앞서 설명한 것처럼 단계(631)은 선택적인 실시예이므로, 랜덤 넘버 생성 과정을 통해 완전히 새로운 E를 다시 생성하는 것도 가능하다.

[0073] dP가 존재하는 경우, dQ에 대해서도 동일한 과정이 반복된다. dQ의 계산(640) 및 존재 여부 판단(650) 및 dQ 부존재에 따른 E의 재생성(651)은 단계(620, 630 및 631)의 설명과 같다. 그리고 유효한 dP와 dQ가 모두 존재하는 경우, 단계(660)에서 CRT를 사용하는 암호화 알고리즘이 수행될 것이다.

[0074] 상술한 실시예들에 따른 효과 및 성능 이슈를 잠시 설명하기로 한다. 실시예들 중 PUF 하드웨어 핑거프린트를 근원 값으로 하는 랜덤 넘버 생성의 경우 외부의 공격이 불가능하다. E와 D의 쌍(pair)을 항상 재생성하여 사용하더라도 E의 값이 너무 작다면(예를 들어 16-bit 미만) 동일한 E-D 쌍이 우연히 사용될 수도 있을 것이나, 이는 E를 조금 키움으로써 방지할 수 있다. E의 크기가 커지는 경우, 전자서명 검증에 필요한 연산양은 증가하지만, 이러한 연산을 하는 인증기관의 하드웨어 리소스를 고려하면 부담이 되지 않는 정도이다. 따라서, E 값의 크기는 성능을 저하하지 않는 선에서 충분히 큰 값, 이를테면 128-bit 이상으로 정해질 수 있다.

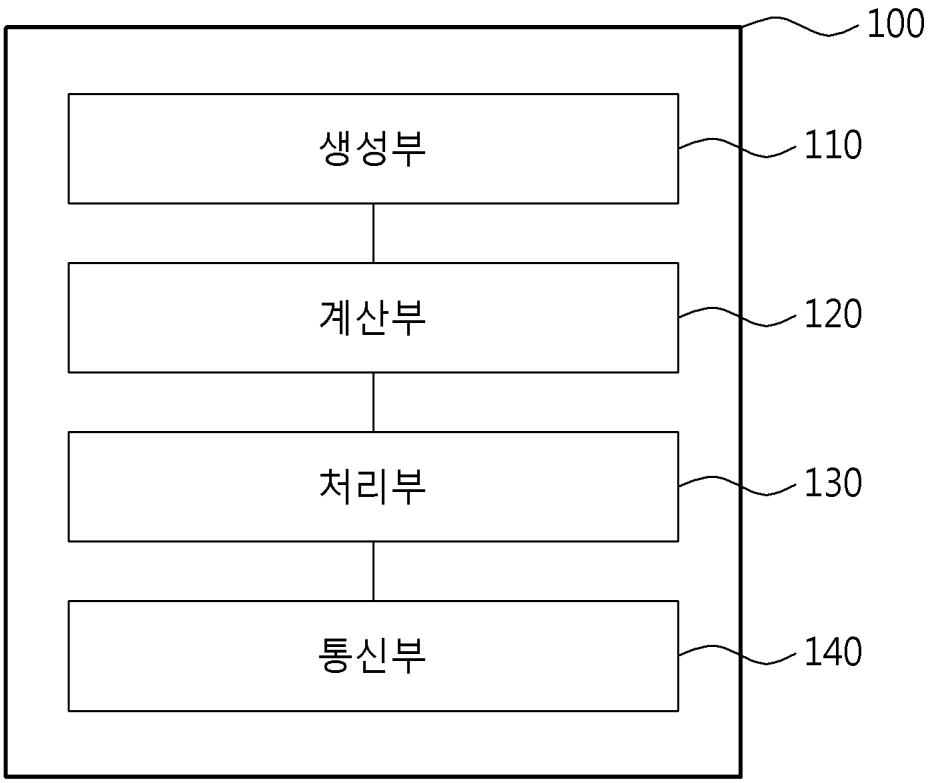
[0075] 한편, 실시예들에 따라 E- D 쌍이 지속적으로 재생성 됨으로써 발생하는 성능 저하도 크지 않다. E-D 쌍을 생성하는 연산의 시간 복잡도는 키의 길이 n에 비례하고(시간 복잡도 O(n)), 서명 생성 연산은 구현 방식에 따라 키의 길이 n의 제곱 또는 세제곱에 비례한다(시간 복잡도 $O(n^2)$ 또는 $O(n^3)$). 그런데 최근 사용되는 RSA의 최저 키 길이는 1024, 2048-bit 이상이기 때문에 E-D 쌍을 생성하는 연산 시간은 서명 생성 시간의 수천분의 1에 해당하며, 이 정도 소모 시간은 전체적으로 봤을 때 그 비중을 무시할 수 있을 정도로 작은 편이다. 따라서 실시예들에 따르면 하드웨어적인 무리나 성능 저하 없이 DPA와 같은 부채널 공격이 원천적으로 방지된다.

[0077] 이상에서, 실시예들이 비록 한정된 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 상기의 기재로부터 다양한 수정 및 변형이 가능하다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다.

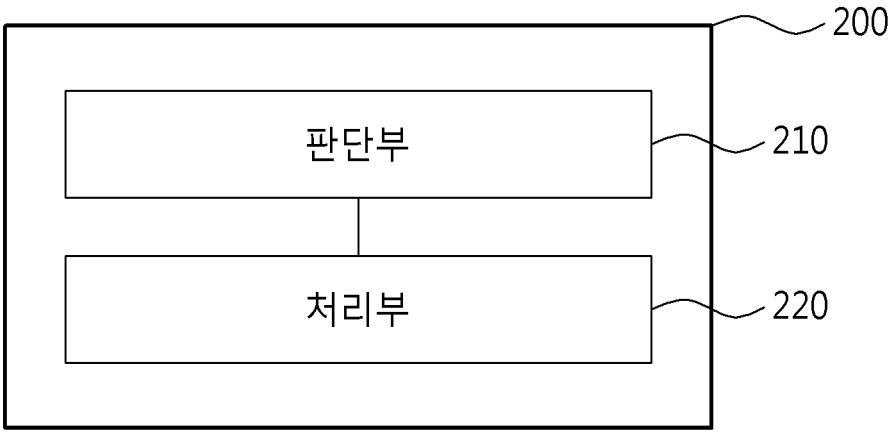
[0078] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

도면

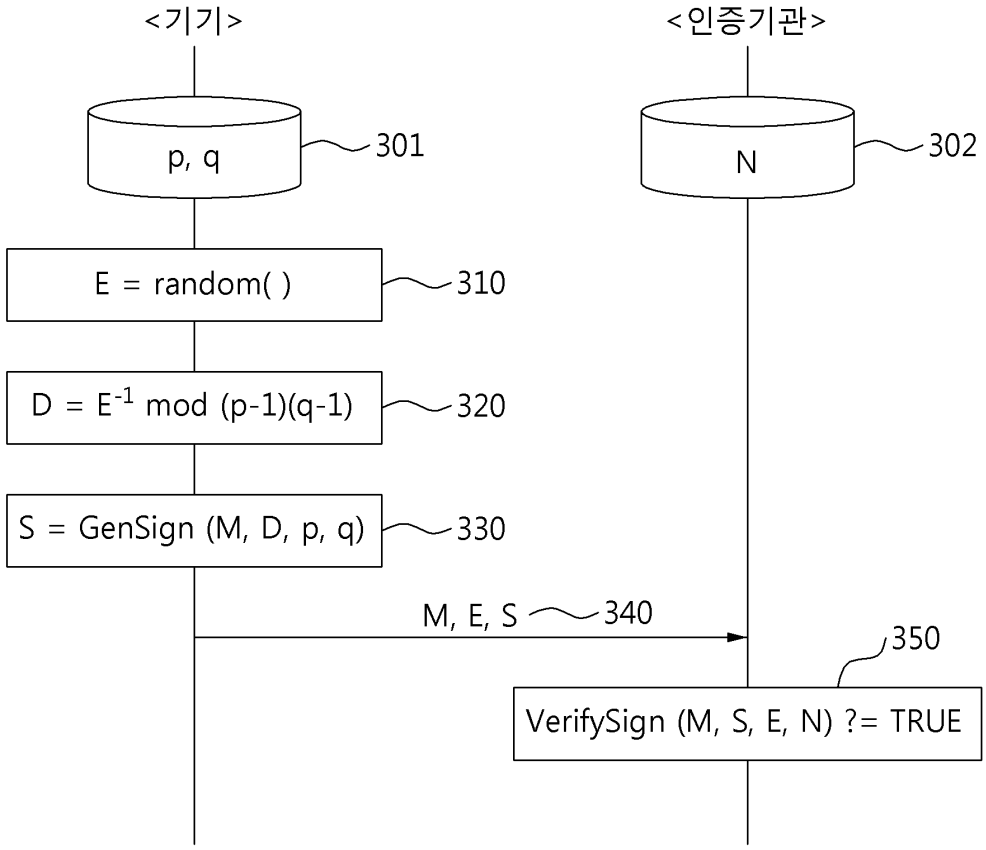
도면1



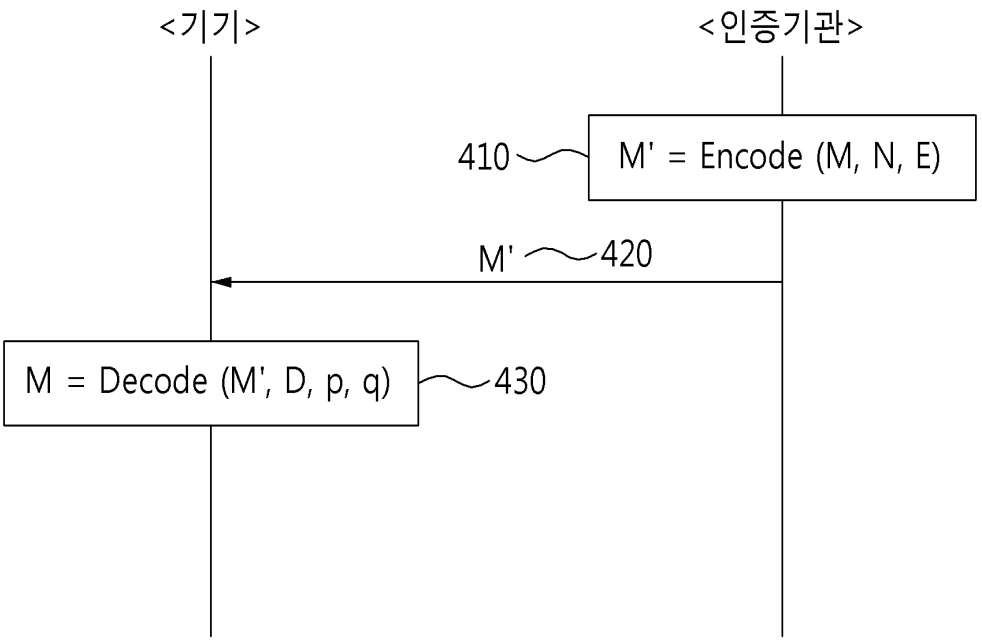
도면2



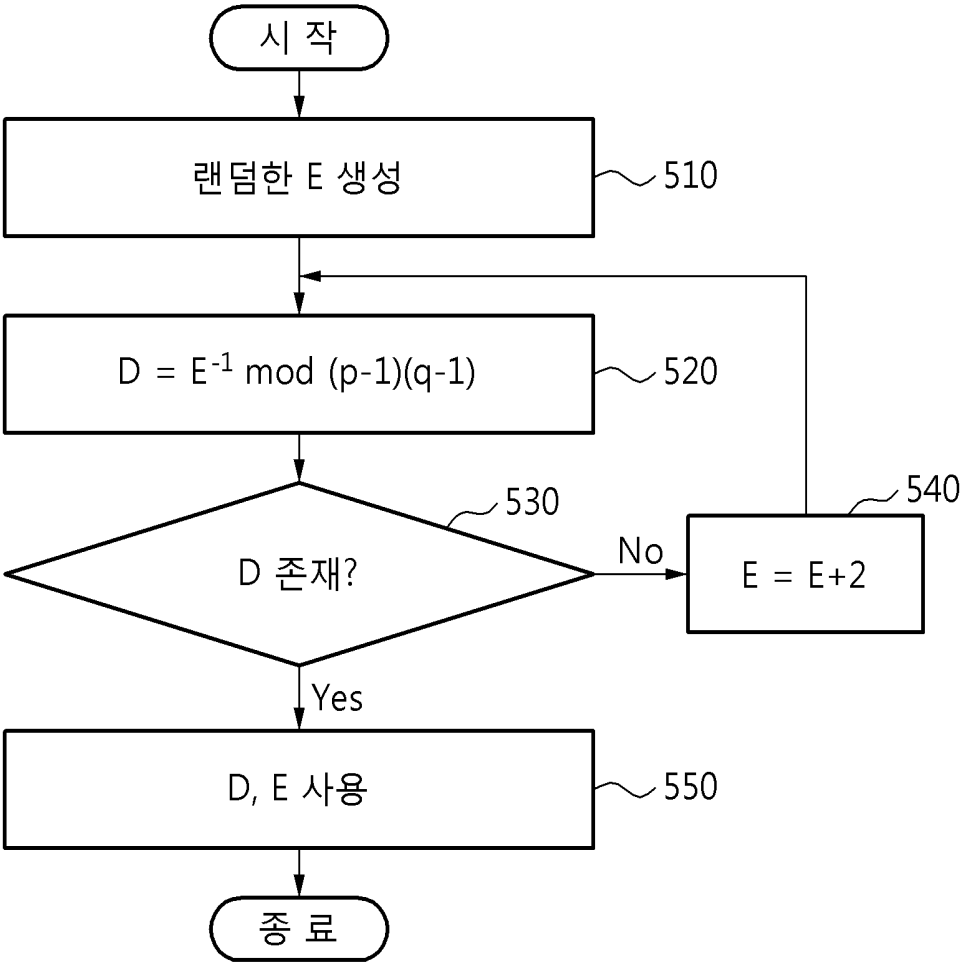
도면3


<기기>
<인증기관>
p, q
301
N
302
E = random( )
310
D = E-1 mod (p-1)(q-1)
320
S = GenSign (M, D, p, q)
330
M, E, S
340
350
VerifySign (M, S, E, N) ?= TRUE

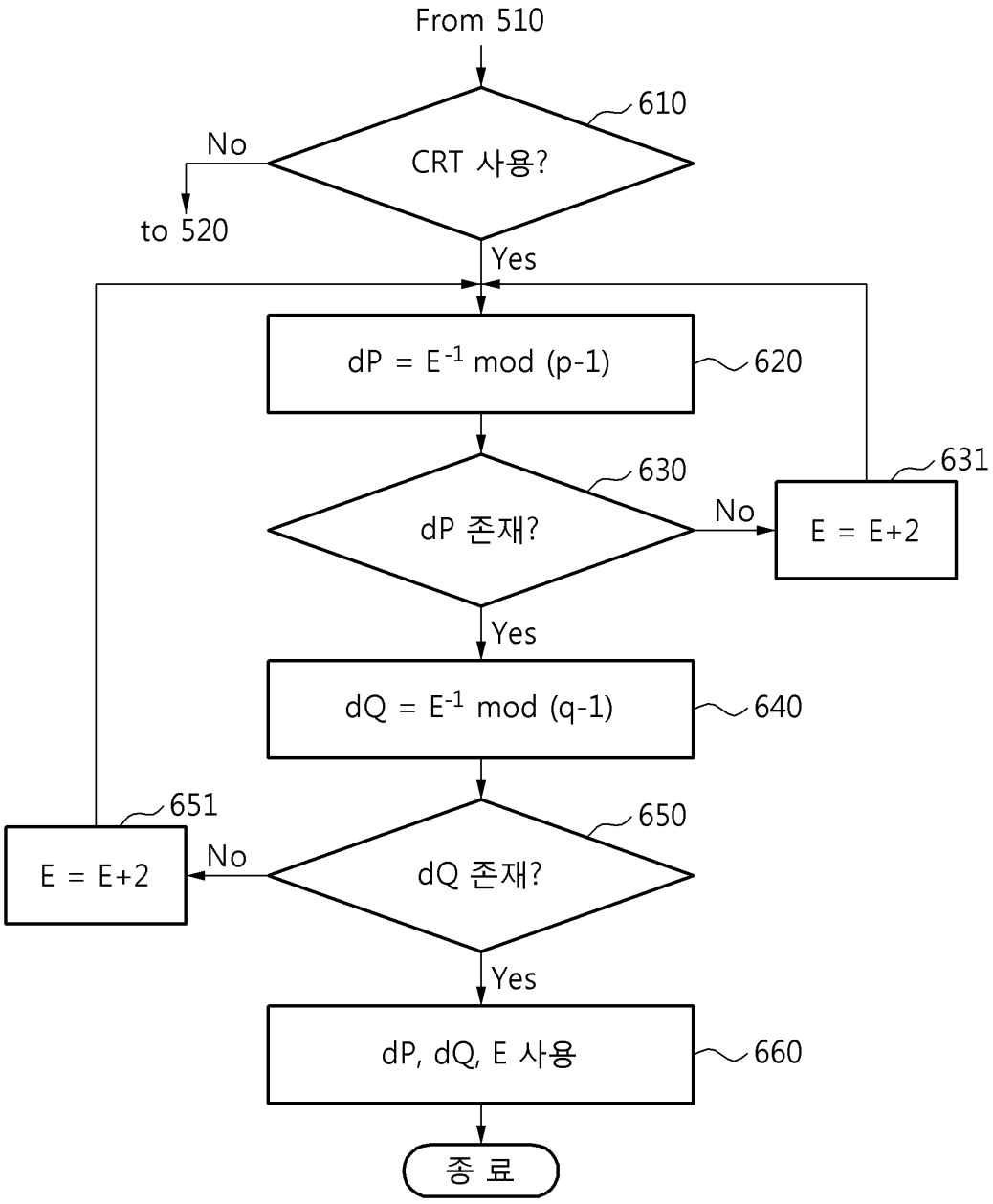
도면4


<기기>
<인증기관>
410
M' = Encode (M, N, E)
M'
420
M = Decode (M', D, p, q)
430

도면5


시 작
랜덤한 E 생성
510
D = E-1 mod (p-1)(q-1)
520
530
D 존재?
No
540
E = E+2
Yes
D, E 사용
550
종 료

도면6


From 510
610
CRT 사용?
No
to 520
Yes
dP = E-1 mod (p-1)
620
630
dP 존재?
No
631
E = E+2
Yes
dQ = E-1 mod (q-1)
640
650
dQ 존재?
No
651
E = E+2
Yes
dP, dQ, E 사용
660
종 료