



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2007-0104939
(43) 공개일자 2007년10월29일

(51) Int. Cl.

H04L 9/32 (2006.01) H04L 9/08 (2006.01)

(21) 출원번호 10-2007-7021336

(22) 출원일자 2007년09월17일

심사청구일자 2007년09월17일

번역문제출일자 2007년09월17일

(86) 국제출원번호 PCT/US2006/006650

국제출원일자 2006년02월24일

(87) 국제공개번호 WO 2006/091844

국제공개일자 2006년08월31일

(30) 우선권주장

60/656,796 2005년02월25일 미국(US)

(71) 출원인

칼컴 인코포레이티드

미국 캘리포니아 샌디에고 모어하우스
드라이브5775 (우 92121-1714)

(72) 발명자

간트맨, 알렉산더

미국 92064 캘리포니아 포웨이 에디나 웨이 13242

로즈, 그레고리 고든

미국 92117 캘리포니아 샌디에고 노쓰 스타 드라이브 3234

(뒷면에 계속)

(74) 대리인

남상선

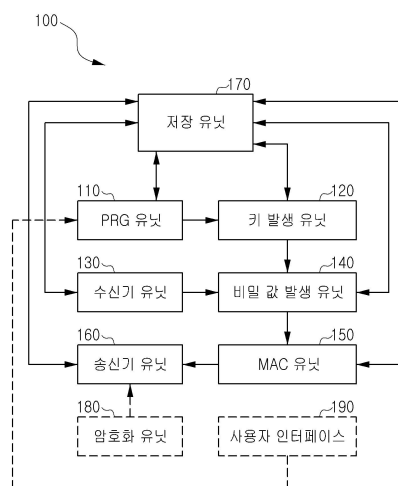
전체 청구항 수 : 총 42 항

(54) 인증을 위한 작은 공개-키 기반 디지털 서명

(57) 요약

개시된 실시예들은 공통 모듈러스(modulus) N 의 이용에 합의한 두 개의 엔티티들 간의 인증(authentication)을 허용한다. 상기 인증은 의사난수 스트링 값(pseudorandom string value)을 발생시키는 단계; 상기 모듈러스 N 및 상기 의사난수 스트링 값에 기초하여 공개 키 값(public key value)을 발생시키는 단계; 상기 공개 키 값에 대응하는 개인 키 값(private key value)을 발생시키는 단계; 검증자(verifier)의 공개 키 값을 수신하는 단계; 상기 모듈러스 N , 상기 개인 키 값 및 상기 검증자의 공개 키 값에 기초하여 공유 비밀 값(shared secret value)을 발생시키는 단계; 상기 공유 비밀 값을 이용하여 인증 서명 값(authentication signature value)을 계산하는 단계; 및 인증을 위해 상기 인증 서명 값을 전송하는 단계를 포함한다. 상기 인증 서명이 수신될 때, 상기 공개 키 값 및 상기 공유 값이 발생되어 인증 서명 값을 계산한다. 이후에, 상기 인증 서명 값들이 비교되고 인증된다.

대표도 - 도1



(72) 발명자

너렌버그, 존 더블유., 투

미국 92131 캘리포니아 샌디에고 피그 트리 스트리트 12343

호크스, 필립 마이클

오스트레일리아 2131 뉴 사우스 웨일즈 애쉬필드
낙클레이드스트리트 #30 18-20

특허청구의 범위

청구항 1

공통 모듈러스(modulus) N 의 이용에 합의한 두 개의 엔티티(entity)들 간의 인증(authentication)에 이용하기 위한 방법으로서:

입력 값으로부터 의사난수 스트링 값(pseudorandom string value)을 발생시키는 단계;

상기 모듈러스 N 및 상기 의사난수 스트링 값에 기초하여 제 1 공개 키(public key)를 발생시키는 단계;

상기 제 1 공개 키 값에 대응하는 제 1 개인 키(private key)를 발생시키는 단계;

제 2 공개 키 값을 수신하는 단계; 및

상기 모듈러스 N , 상기 제 1 개인 키 값 및 상기 제 2 공개 키 값에 기초하여 공유 비밀 값(shared secret value)을 발생시키는 단계를 포함하는, 인증에 이용하기 위한 방법.

청구항 2

제 1 항에 있어서,

상기 공유 비밀 값을 이용하여 인증 서명 값(authentication signature value)을 계산하는 단계; 및

상기 인증 서명 값을 전송하는 단계를 더 포함하는, 인증에 이용하기 위한 방법.

청구항 3

제 2 항에 있어서,

상기 인증 서명 값을 절단(truncate)하는 단계를 더 포함하며; 여기서 상기 인증 서명 값을 전송하는 단계는 절단된 인증 서명 값(truncated authentication signature value)을 포함하는, 인증에 이용하기 위한 방법.

청구항 4

제 2 항 또는 제 3 항에 있어서,

상기 인증 서명 값을 계산하는 단계는 상기 공유 비밀 값 및 상기 입력 값을 이용하여 상기 인증 서명 값을 계산하는 단계를 포함하는, 인증에 이용하기 위한 방법.

청구항 5

제 2 항 내지 제 4 항 중 어느 한 항에 있어서,

패스워드(password) 값을 수신하는 단계;

상기 제 1 개인 키 값을 이용하여 상기 패스워드 값을 암호화(encrypt)하는 단계; 및

상기 암호화된 패스워드 값을 전송하는 단계를 더 포함하는, 인증에 이용하기 위한 방법.

청구항 6

제 5 항에 있어서,

상기 인증 서명 값을 계산하는 단계는 상기 공유 비밀 값 및 상기 암호화된 패스워드 값을 이용하여 상기 인증 서명 값을 계산하는 단계를 포함하는, 인증에 이용하기 위한 방법.

청구항 7

제 1 항 내지 제 6 항 중 어느 한 항에 있어서,

상기 제 1 개인 키 값을 발생시키는 단계는 이산 대수 문제 모듈로 N (discrete logarithm problem modulo N)을 풀어서 상기 제 1 개인 키 값을 발생시키는 단계를 포함하는, 인증에 이용하기 위한 방법.

청구항 8

제 1 항 내지 제 7 항 중 어느 한 항에 있어서,

사용자 입력 데이터를 수신하는 단계를 더 포함하며 상기 입력 값은 상기 사용자 입력 데이터에 기초하는, 인증에 이용하기 위한 방법.

청구항 9

제 1 항 내지 제 8 항 중 어느 한 항에 있어서,

상기 입력 값을 전송하는 단계를 더 포함하는, 인증에 이용하기 위한 방법.

청구항 10

제 1 항 내지 제 9 항 중 어느 한 항에 있어서,

상기 이산 대수 문제 모듈로 N (discrete logarithm problem modulo N)을 풀어서 상기 제 2 공개 키 값에 대응하는 제 2 개인 키를 발생시키는 단계; 및

공유 비밀 키(shared secret key)로서 상기 제 2 개인 키를 이용하는 단계를 더 포함하는, 인증에 이용하기 위한 방법.

청구항 11

제 1 항 내지 제 10 항 중 어느 한 항의 방법의 수행을 제어하도록 구성되는 처리기.

청구항 12

제 1 항 내지 제 10 항 중 어느 한 항의 방법을 수행하는 명령(instruction)들을 저장하도록 구성되는 저장 매체.

청구항 13

공통 모듈러스 N 의 이용에 합의한 두 개의 엔티티들 간의 인증에 이용하기 위한 장치로서:

입력 값으로부터 의사난수 스트링 값을 발생시키기 위한 수단;

상기 모듈러스 N 및 상기 의사난수 스트링 값에 기초하여 제 1 공개 키 값을 발생시키기 위한 수단;

상기 제 1 공개 키 값에 대응하는 제 1 개인 키 값을 발생시키기 위한 수단;

제 2 공개 키 값을 수신하기 위한 수단; 및

상기 모듈러스 N , 상기 제 1 개인 키 값 및 상기 제 2 공개 키 값에 기초하여 공유 비밀 값을 발생시키기 위한 수단을 포함하는, 인증에 이용하기 위한 장치.

청구항 14

제 13 항에 있어서,

상기 공유 비밀 값을 이용하여 인증 서명 값을 계산하기 위한 수단; 및

상기 인증 서명 값을 전송하기 위한 수단을 더 포함하는, 인증에 이용하기 위한 장치.

청구항 15

제 14 항에 있어서,

상기 인증 서명 값을 절단(truncate)하기 위한 수단을 더 포함하며; 여기서 상기 인증 서명 값을 전송하는 단계는 절단된 인증 서명 값을 전송하는 단계를 포함하는, 인증에 이용하기 위한 장치.

청구항 16

제 14 항 또는 제 15 항에 있어서,

상기 인증 서명 값을 계산하기 위한 수단은 상기 공유 비밀 값 및/또는 상기 입력 값을 이용하여 상기 인증 서

명 값을 계산하기 위한 수단을 포함하는, 인증에 이용하기 위한 장치.

청구항 17

제 14 항 내지 제 16 항 중 어느 한 항에 있어서,

패스워드 값을 수신하기 위한 수단;

상기 제 1 개인 키 값을 이용하여 상기 패스워드 값을 암호화(encrypt)하기 위한 수단; 및

상기 암호화된 패스워드 값을 전송하기 위한 수단을 더 포함하는, 인증에 이용하기 위한 장치.

청구항 18

제 17 항에 있어서,

상기 인증 서명 값을 계산하기 위한 수단은 상기 공유 비밀 값 및 상기 암호화된 패스워드 값을 이용하여 상기 인증 서명 값을 계산하기 위한 수단을 포함하는, 인증에 이용하기 위한 장치.

청구항 19

제 13 항 내지 제 18 항 중 어느 한 항에 있어서,

상기 제 1 개인 키 값을 발생시키기 위한 수단은:

상기 이산 대수 문제 모듈로 N (discrete logarithm problem modulo N)을 풀어서 상기 제 1 개인 키 값을 발생시키기 위한 수단을 포함하는, 인증에 이용하기 위한 장치.

청구항 20

제 13 항 내지 제 19 항 중 어느 한 항에 있어서,

상기 이산 대수 문제 모듈로 N (discrete logarithm problem modulo N)을 풀어서 상기 제 2 공개 키 값에 대응하는 제 2 개인 키를 발생시키기 위한 수단; 및

공유 비밀 키로서 상기 제 2 개인 키를 이용하기 위한 수단을 더 포함하는, 인증에 이용하기 위한 장치.

청구항 21

공통 모듈러스 N 의 이용에 합의한 두 개의 엔티티들 간의 인증에 이용하기 위한 장치로서:

입력 값으로부터 의사난수 스트링 값을 발생시키도록 구성되는 의사-난수(pseudo-random) 발생기;

상기 모듈러스 N 및 상기 의사난수 스트링 값에 기초하여 제 1 공개 키 값을 발생시키고, 상기 제 1 공개 키 값에 대응하는 제 1 개인 키 값을 발생시키도록 구성되는 키 발생 유닛(key generating unit);

제 2 공개 키 값을 수신하도록 구성되는 수신기 유닛; 및

상기 모듈러스 N , 상기 제 1 개인 키 값 및 상기 제 2 공개 키 값에 기초하여 공유 비밀 값을 발생시키도록 구성되는 비밀 값 발생 유닛을 포함하는, 인증에 이용하기 위한 장치.

청구항 22

제 21 항에 있어서,

상기 공유 비밀 값을 이용하여 인증 서명 값(authentication signature value)을 계산하도록 구성되는 메시지 인증 코드 유닛(message authentication code unit); 및

상기 인증 서명 값을 전송하도록 구성되는 송신기 유닛을 더 포함하는, 인증에 이용하기 위한 장치.

청구항 23

제 22 항에 있어서,

상기 송신기 유닛은 절단된 인증 서명(truncated authentication signature)을 전송하는, 인증에 이용하기 위

한 장치.

청구항 24

공통 모듈러스 N 의 이용에 합의한 두 개의 엔티티들 간의 인증에 이용하기 위한 방법으로서:

제 1 인증 서명 값을 수신하는 단계;

입력 값으로부터 의사난수 스트링 값을 발생시키는 단계;

상기 모듈러스 N 및 상기 의사난수 스트링 값에 기초하여 제 1 공개 키 값을 발생시키는 단계;

상기 모듈러스 N , 개인 키 값 및 상기 제 1 공개 키 값에 기초하여 공유 비밀 값을 발생시키는 단계;

상기 공유 비밀 값을 이용하여 제 2 인증 서명 값을 계산하는 단계; 및

상기 제 1 인증 서명 값을 상기 제 2 인증 서명 값과 비교하는 단계를 포함하는, 인증에 이용하기 위한 방법.

청구항 25

제 24 항에 있어서,

상기 제 1 및 제 2 인증 서명 값들을 비교하는 단계는 상기 제 1 인증 서명 값을 상기 제 2 인증 서명 값의 절단된(truncated) 부분과 비교하는 단계를 포함하는, 인증에 이용하기 위한 방법.

청구항 26

제 24 항 또는 제 25 항에 있어서,

상기 입력 값을 수신하는 단계를 더 포함하는, 인증에 이용하기 위한 방법.

청구항 27

제 24 항 내지 제 26 항 중 어느 한 항에 있어서,

상기 인증 서명 값을 계산하는 단계는 상기 공유 비밀 값 및 상기 입력 값을 이용하여 상기 인증 서명 값을 계산하는 단계를 포함하는, 인증에 이용하기 위한 방법.

청구항 28

제 24 항 내지 제 27 항 중 어느 한 항에 있어서,

암호화된(encrypted) 패스워드 값을 수신하는 단계;

상기 제 1 공개 키 값을 이용하여 상기 패스워드 값을 복호화(decrypt)하는 단계; 및

상기 복호화된 패스워드 값을 검증(validate)하는 단계를 더 포함하는, 인증에 이용하기 위한 방법.

청구항 29

제 28 항에 있어서,

상기 인증 서명 값을 계산하는 단계는 상기 공유 비밀 값 및 상기 암호화된 패스워드 값을 이용하여 상기 인증 서명 값을 계산하는 단계를 포함하는, 인증에 이용하기 위한 방법.

청구항 30

제 24 항 내지 제 29 항 중 어느 한 항의 방법의 수행을 제어하도록 구성되는 처리기.

청구항 31

제 24 항 내지 제 29 항 중 어느 한 항의 방법을 수행하는 명령들을 저장하도록 구성되는 저장 매체.

청구항 32

공통 모듈러스 N 의 이용에 합의한 두 개의 엔티티들 간의 인증에 이용하기 위한 장치로서:

제 1 인증 서명 값을 수신하기 위한 수단;

입력 값으로부터 의사난수 스트링 값을 발생시키기 위한 수단;

상기 모듈러스 N 및 상기 의사난수 스트링 값에 기초하여 제 1 공개 키 값을 발생시키기 위한 수단;

상기 모듈러스 N , 개인 키 값 및 상기 제 1 공개 키 값에 기초하여 공유 비밀 값을 발생시키기 위한 수단;

상기 공유 비밀 값을 이용하여 제 2 인증 서명 값을 계산하기 위한 수단; 및

상기 제 1 인증 서명 값과 상기 제 2 인증 서명 값을 비교하기 위한 수단을 포함하는, 인증에 이용하기 위한 장치.

청구항 33

제 32 항에 있어서,

상기 제 1 및 제 2 인증 서명 값들을 비교하기 위한 상기 수단은 상기 제 1 인증 서명 값을 상기 제 2 인증 서명 값의 절단된(truncated) 부분과 비교하는 것을 포함하는, 인증에 이용하기 위한 장치.

청구항 34

제 32 항 또는 제 33 항에 있어서,

상기 입력 값을 수신하기 위한 수단을 더 포함하는, 인증에 이용하기 위한 장치.

청구항 35

제 32 항 내지 제 34 항 중 어느 한 항에 있어서,

상기 인증 서명 값을 계산하기 위한 상기 수단은 상기 공유 비밀 값 및 상기 입력 값을 이용하여 상기 인증 서명 값을 계산하기 위한 수단을 포함하는, 인증에 이용하기 위한 장치.

청구항 36

제 32 항 내지 제 35 항 중 어느 한 항에 있어서,

암호화된 패스워드 값을 수신하기 위한 수단;

상기 제 1 공개 키 값을 이용하여 상기 패스워드 값을 복호화(decrypt)하기 위한 수단; 및

상기 복호화된 패스워드 값을 검증(validate)하기 위한 수단을 더 포함하는, 인증에 이용하기 위한 장치.

청구항 37

제 36 항에 있어서,

상기 인증 서명 값을 계산하기 위한 상기 수단은 상기 공유 비밀 값 및 상기 암호화된 패스워드 값을 이용하여 상기 인증 서명 값을 계산하기 위한 수단을 포함하는, 인증에 이용하기 위한 장치.

청구항 38

공통 모듈러스 N 의 이용에 합의한 두 개의 엔티티들 간의 인증에 이용하기 위한 장치로서:

제 1 인증 서명 값을 수신하도록 구성되는 수신기 유닛;

입력 값으로부터 의사난수 스트링 값을 발생시키도록 구성되는 의사-난수 발생기;

상기 모듈러스 N 및 상기 의사난수 스트링 값에 기초하여 제 1 공개 키 값을 발생시키도록 구성되는 키 발생 유닛;

상기 모듈러스 N , 개인 키 값 및 상기 제 1 공개 키 값에 기초하여 공유 비밀 값을 발생시키도록 구성되는 공유 비밀 값 발생 유닛(shared secret value generating unit);

상기 공유 비밀 값을 이용하여 제 2 인증 서명 값을 계산하도록 구성되는 메시지 인증 코드 유닛; 및

상기 제 1 인증 서명 값을 상기 제 2 인증 서명 값과 비교하도록 구성되는 비교 유닛을 포함하는, 인증에 이용하기 위한 장치.

청구항 39

제 38 항에 있어서,

상기 비교 유닛은 상기 제 1 인증 서명 값과 상기 제 2 인증 서명 값의 절단된(truncated) 부분을 포함하는, 인증에 이용하기 위한 장치.

청구항 40

공통 모듈러스 N 의 이용에 합의한 두 개의 엔티티들 간의 인증에 이용하기 위한 방법으로서:

상기 모듈러스 N 에 기초하여 발생하는 공개 키 값을 수신하는 단계;

상기 이산 대수 문제 모듈로 N (discrete logarithm problem modulo N)을 풀어서 상기 공개 키 값에 대응하는 개인 키를 발생시키는 단계; 및

상기 개인 키를 공유 비밀 키로서 이용하는 단계를 포함하는, 인증에 이용하기 위한 방법.

청구항 41

제 40 항에 있어서,

상기 개인 키를 발생시키는 단계는:

중국인의 나머지 정리(Chinese remainder theorem)를 이용하여 상기 개인 키를 계산하는 단계를 포함하는, 인증에 이용하기 위한 방법.

청구항 42

공통 모듈러스 N 의 이용에 합의한 두 개의 엔티티들 간의 인증에 이용하기 위한 장치로서:

상기 모듈러스 N 에 기초하여 발생하는 공개 키 값을 수신하기 위한 수단;

상기 이산 대수 문제 모듈로 N (discrete logarithm problem modulo N)을 풀어서 상기 공개 키에 대응하는 개인 키를 발생시키기 위한 수단; 및

상기 개인 키를 공유 비밀 키로서 이용하기 위한 수단을 포함하는, 인증에 이용하기 위한 장치.

명세서

기술분야

<1> 본 발명은 일반적으로 암호기법 및 더 상세하게는 인증에 관한 것이다.

배경기술

<2> 인증은 어떠한 형태의 디지털 서명 또는 메시지 인증 코드를 수반한다. 디지털 서명들과 메시지 인증 코드들은 공개-키(public-key) 및/또는 대칭-키(symetric-key) 암호 시스템들에 기초한다. 상기 두 개의 시스템들은 다른 특징들을 가지고, 하나가 어느 상황에서 더 적절한 반면 다른 하나는 다른 상황에서 더 적절하다.

<3> 일반적으로, 공개-키 시스템들은 키 관리 및 분배의 면에서 이점을 갖는다. 그러나, 공개 키 시스템들은 종종 계산상으로 어렵고 큰 디지털 서명들을 수반할 수 있다. 따라서, 공개 키 시스템들은 장치의 비용, 무게 및/또는 크기의 증가를 초래할 수 있고; 지연을 유발할 수 있는 추가적인 처리 시간을 필요로 하고, 더 많은 에너지 및/또는 배터리들을 방전시킬 수 있고; 또는 상기 결과들의 조합을 초래할 수 있는 강력한 처리기를 필요로 할 수 있다.

<4> 결과적으로, 대부분의 소형 인증 장치들은, 계산상의 요구조건들과 작은 출력들에 대한 필요성 모두에 의해, 대칭-키 암호기법(cryptography)을 이용하도록 강제되어 왔다. 그러나, 이동 전화와 같은 장치들에 더욱 더 강력한 처리기들이 탑재되어, 상기 계산상의 요구조건들은 더 이상 지배적인 요소가 아니다. 그럼에도, 더 작은

디지털 서명에 대한 필요가 여전히 존재한다.

발명의 상세한 설명

- <5> 일 특징으로, 공통 모듈러스(modulus) N 의 이용에 동의한 두 개의 엔티티(entity)들 사이의 인증에 이용하기 위한 수단을 갖는 방법 및 장치는, 입력 값으로부터 의사난수(pseudorandom) 스트링(string) 값을 발생시키는 단계; 상기 모듈러스 N 과 상기 의사난수 스트링 값에 기초하여 제 1 공개 키를 발생시키는 단계; 상기 제 1 공개 키 값에 대응하는 제 1 개인 키(private key) 값을 발생시키는 단계; 제 2 공개 키 값을 수신하는 단계; 및 상기 모듈러스 N , 상기 제 1 개인 키와 상기 제 2 공개 키 값에 기초하여 공유 비밀 값(shared secret value)을 발생시키는 단계를 포함한다. 후에, 인증 서명 값이 상기 공유 비밀 값을 이용하여 계산되고; 전송될 수 있다.
- <6> 다른 특징으로, 공통 모듈러스 N 의 이용에 동의한 두 개의 엔티티들 사이의 인증에 이용하기 위한 장치는, 상기 장치는 입력 값으로부터 의사난수 스트링 값을 발생시키도록 구성되는 의사-난수(pseudo-random) 발생기; 상기 모듈러스 N 및 상기 의사난수 스트링 값에 기초하여 제 1 공개 키 값을 발생시키고, 상기 제 1 공개 키 값에 대응하는 제 1 개인 키 값을 발생시키도록 구성되는 키 발생 유닛(key generating unit); 제 2 공개 키 값을 수신하도록 구성되는 수신기 유닛; 및 상기 모듈러스 N , 상기 제 1 개인 키 값과 상기 제 2 공개 키 값에 기초하여 공유 비밀 값을 발생시키도록 구성되는 비밀 값 발생 유닛을 포함한다.
- <7> 또 다른 특징으로, 공통 모듈러스 N 의 사용에 동의한 두 개의 엔티티들 사이의 인증에 이용하기 위한 수단을 갖는 방법 및 장치는, 제 1 인증 서명 값을 수신하는 단계; 입력 값으로부터 의사난수 스트링 값을 발생시키는 단계; 상기 모듈러스 N 및 상기 의사난수 스트링 값에 기초하여 제 1 공개 키 값을 발생시키는 단계; 상기 모듈러스 N , 개인 키 값 및 상기 제 1 공개 키 값에 기초하여 공유 비밀 값을 발생시키는 단계; 상기 공유 비밀 값을 이용하여 제 2 인증 서명 값을 계산하는 단계; 및 상기 제 1 인증 서명 값을 상기 제 2 인증 서명 값과 비교하는 단계를 포함한다.
- <8> 또 다른 특징으로, 공통 모듈러스 N 의 이용에 동의한 두 개의 엔티티들 사이의 인증에 이용하기 위한 장치는: 제 1 인증 서명 값을 수신하도록 구성되는 수신기 유닛; 입력 값으로부터 의사난수 스트링 값을 발생시키도록 구성되는 의사-난수 유닛; 상기 모듈러스 N 및 상기 의사난수 스트링 값에 기초하여 제 1 공개 키 값을 발생시키도록 구성되는 키 발생 유닛; 상기 모듈러스 N , 개인 키 값 및 상기 제 1 공개 키 값에 기초하여 공유 비밀 값을 발생시키도록 구성되는 공유 비밀 값 발생 유닛; 상기 공유 비밀 값을 이용하여 제 2 인증 서명 값을 계산하도록 구성되는 메시지 인증 코드 유닛; 및 상기 제 1 인증 서명 값과 상기 제 2 인증 서명 값을 비교하도록 구성되는 비교 유닛을 포함한다.
- <9> 또 다른 특징으로, 공통 모듈러스 N 의 이용에 동의한 두 개의 엔티티들 사이의 인증에 이용하기 위한 수단을 갖는 방법 및 장치는 상기 모듈러스 N 에 기초하여 발생하는 공개 키 값을 수신하는 단계; 이산 대수 문제(discrete logarithm problem) 모듈로 N (modulo N)을 풀어 상기 공개 키 값에 대응하는 개인 키를 발생시키는 단계; 및 상기 개인 키를 공유 비밀 키로서 이용하는 단계를 포함한다.
- <10> 상기 방법들은 처리기에 의해 실행되고 그리고/또는 제어될 수 있다. 유사하게, 상기 방법들은 기계 판독가능 매체에 저장된 명령들에 의해 실행되고 그리고/또는 수행될 수 있다.

실시예

- <18> 이하의 설명에서, 특정한 상세내용들은 실시예들에 대한 총괄적인 이해를 제공하기 위한 것이다. 그러나, 당업자는 상기 실시예들이 이러한 특정 세부내용 없이도 실시될 수 있음을 알 것이다. 예를 들어, 회로들은 불필요하게 상세하게 실시예들을 회색하지 않기 위해 블록 다이어그램들로 도시될 수 있다. 다른 예로, 공지된 회로들, 구조들 및 기술들이 실시예들을 회색하지 않게 하기 위하여 상세하게 도시될 수 있다.
- <19> 또한, 상기 실시예들은 순서도, 순서 다이어그램, 구조 다이어그램, 또는 블록도로서 표현되는 프로세스로 기술될 수 있음에 유의하여야 한다. 순서도가 순차적 프로세스로서 연산들을 기술할 수 있을지라도, 다수의 연산들은 병행하여 또는 동시에 수행될 수 있다. 추가로, 상기 연산들의 순서는 재-배열될 수 있다. 프로세스는 그 연산들이 완료될 때 종결된다. 프로세스는 방법, 함수, 프로시저, 서브루틴(subroutine), 서브프로그램(subprogram) 등에 대응할 수 있다. 프로세스가 함수에 대응할 때, 그 종결은 상기 함수의 호출 함수 또는 주 함수(main function)로의 복귀에 대응한다.
- <20> 또한, 실시예들은 하드웨어, 소프트웨어, 펌웨어, 미들웨어, 마이크로코드(microcode), 또는 이들의 임의의 조

함으로 구현될 수 있다. 소프트웨어, 펌웨어, 미들웨어 또는 마이크로코드로 구현될 때, 필수적인 태스크들을 수행하는 프로그램 코드 또는 코드 세그먼트(segment)들은 다양한 저장 매체와 같은 기계 판독가능 매체에 저장될 수 있다. 처리기는 상기 필수적인 태스크들을 수행할 수 있다. 코드 세그먼트는 프로시저, 함수, 서브프로그램, 프로그램, 루틴, 서브루틴, 모듈, 소프트웨어 패키지, 클래스, 또는 명령, 데이터 구조들, 또는 프로그램 구문(statement)들의 임의의 조합을 나타낼 수 있다. 코드 세그먼트는 정보, 데이터, 인수(argument)들, 파라미터들, 또는 메모리 내용들을 전달 및/또는 수신함으로써 다른 코드 세그먼트나 하드웨어 회로에 연결될 수 있다. 정보, 인수들, 파라미터들, 데이터 등은 메모리 공유, 메시지 전달, 토큰 전달, 네트워크 전송 등을 포함하는 임의의 적절한 수단을 통해 전달, 회송, 또는 전송될 수 있다.

- <21> 일반적으로, 이하의 기술내용은 디지털 서명이, 예를 들어 6개의 디지털(digit)들의 차수로, 상대적으로 작게 되는 것을 허용한다. 이는 상기 서명이, 일반적으로 인증 목적으로, 상기 서명이 저-대역폭 채널을 통해 전송되거나 상기 서명이 사람에게 의해 타이핑 되어야 할 필요가 있는 것과 같은 다양한 상황에서 편리하게, 용이하게, 저렴하게 및/또는 효율적으로 이용되게 하여 준다. 또한 후자의 경우, 작은 디지털 서명이 사용자-편의적일 것이다.
- <22> 이하에서, 본 발명과 관련된 일부 개념들은 본 발명의 설명 및/또는 더 나은 이해를 목적으로 논의될 것이다. 특정한 상황, 정리(theorem), 및/또는 공식들에 대한 참조는 예시이며 본 발명의 범위를 제한하는 것으로 해석되어서는 안된다.
- <23> 공개-키 서명
- <24> 일반적으로, 오늘날 이용되는 공개-키 디지털 서명 알고리즘들의 두 개의 주 클래스들이 있다.
- <25> 가장 일반적인 한가지는 RSA 시스템일 수 있으며, 공개 키는 큰 비소수(非素數, composite number) N 이고, 잠재적인 수학적 "난제"는 N 을 $p \cdot q$ 로 인수분해하는 것의 곤란성이다. 현재의 인수분해 알고리즘들은 복잡도에 있어서 준지수승수적이다. 그러므로, 예컨대, 80비트의 키로써 대칭 알고리즘을 파괴하는 것에 상당하는 보안을 위해, 모듈러스 N 은 대략 1024비트가 되어야 할 필요가 있다. RSA 시스템의 디지털 서명은 N 과 거의 동일한 크기의 정수이다. 디지털 서명의 검증은 (일반적으로 작은) 거듭제곱 e 에 대해 모듈로 N (modulo N) 거듭제곱(exponentiation)의 수행을 필요로 하며, 상대적으로 효율적이다.
- <26> 다른 일반적인 디지털 서명 방식은 El Gamal, Schnorr, 또는 디지털 서명 (표준) 알고리즘(Digital Signature (Standard) Algorithm, DSA)으로 불리우며, 그 정확한 실현에 의존한다. 이러한 방식들은 이산 대수들 모듈로(modulo) 소수 P 의 계산의 곤란성에 기초한다. Diffie-Hellman (D-H) 키 합의(agreement) 알고리즘이 발명된 최초의 공개-키 시스템이었으며, 이산 대수 문제에 의존한다. El Gamal은 상기 D-H의 디지털 서명에 대한 적용이다. Schnorr 및 DSA는 디지털 서명들의 크기를 감소시키는 트릭을 이용한다. 또한 상기 모듈러스 P 는 일반적으로 1024비트이다.
- <27> 상기 인수분해 및 이산 대수들의 문제가 관련되어 있다는 증거가 없을 수 있지만, 그 중 어느 문제를 해결하기 위한 알고리즘들도 일반적으로 다른 것에 대해 적용가능한 아날로그를 가지며, 이것이 유사한 키 크기들이 비교적 안전한 시스템을 만드는지에 대한 이유이다. El Gamal 디지털 서명은 각각 P 에 유사한 크기의 숫자들(r, s) 쌍인 반면, Schnorr의 방법은 더 작은 서명들을 가능하게 하여, r 과 s 가 각각 (80비트 보안에 상당하는) 160 비트이다.
- <28> 또한 타원 곡선 디지털 서명들도 El Gamal의 변형으로, 더 작은 키들과 다른 계산을 이용하지만, 결과적인 서명은 여전히 최소의 보안을 위해 대략 320 비트이다.
- <29> 그리하여 상기 기술들 중 임의의 것을 이용하는 가장 짧은 실제 디지털 서명은 320 비트이다. 십진수로 표현하면, 이는 90-디지털 숫자, 또는 통상적인 대문자 및 소문자 및 한 쌍의 구두 문자 인코딩을 이용하여 약 55 캐릭터(character)이다. 그리하여 이는 사용자가, 예컨대 상기 디지털 서명을 신뢰할 수 있게 타이핑하는데 매우 불편하고 어려울 것이다.
- <30> 수학적 배경
- <31> 이하의 논의에서, 이산 대수를 계산하는 기본적인 문제가 기술될 것이다. 상기 문제는 본질적으로 Diffie-Hellman이지만, RSA 시스템과 유사하게, 그 인수분해가 공지되지 않은 모듈로 합성수(composite number) N 이다.
- <32> 예를 들어, 상기 이산 대수 문제는 다음과 같이 기술될 수 있다: 정수 N , g , 및 X 에 대해, 여기서 $X \equiv g^x \pmod{N}$

N 이면, x 를 구하십시오. 종래의 이산 로그 시스템들에서, N 은 보통 소수이고, g 는 곱셈군(multiplicative group) 모듈로 N 의 소수 위수 부분군(prime-order subgroup)의 생성원(generator)이다. 설명을 위해, 합성수 N 을 이용한다. 상기 이산 대수 문제 mod N 을 푸는 것은, $N \equiv 3 \pmod{4}$, 그리고 $N \neq pq$ 이고, 여기서 p 와 q 는 소수 라면, N 을 인수분해하는 능력을 포함한다; N 은 소위 블럼 정수(Blum Integer)이다. 이하에서, N 은 소수 p 와 q 의 곱을 지칭한다.

<33> 또한 중국인의 나머지 정리(Chinese Remainder Theorem, CRT)가 유용한 기법이다. 간단한 경우, 임의의 정수 $X \bmod N$ 은 한 쌍의 정수들 (a, b) 로 표현될 수 있으며, 여기서 $a \equiv X \pmod{p}$ 이고, $b \equiv X \pmod{q}$ 이다. 상기 두 개의 표현들 사이를 전환하는 것이 수월하다. 요구되는 정답 모듈로 N 으로 다시 전환하기 전에 후자의 표현을 이용하여 곱셈 및 나누셈과 같은 연산들을 수행하는 것이 종종 편리하다. 이는 두 개의 인수들로서 표현되지만, 다수의 인수들로 일반화된다. 유일한 소인수 분해 정리(Unique Factorization Theorem)는 모든 정수가 소수 거듭제곱들의 곱으로서 고유한 표현을 갖는다는 것이다. 결과적으로, 상기 CRT는 일반적으로 소인수들에 대하여 이용될 수 있다.

<34> 또한, 작은 소인수들만을 갖는 정수는 평활하다(smooth)고 칭한다. 여기서 "작은"은 일반적으로 평활도 경계(smoothness bound) B 로써 정의된다. 만일 소인수들 중 하나인 p 에 대해, $p-1$ 이 평활하다면, Pollard의 $p-1$ 알고리즘이, 비소수들을 인수분해하는데 이용될 수 있다. 만일 $p-1$ 이 평활하다면 동일한 알고리즘의 변형을 이용하여 소수 p 를 범으로 하여 이산 대수들을 계산할 수 있다. Pohlig-Hellman 알고리즘은, 만일 이산 대수 모듈로 모든 N 의 인수들(modulo all of the factors of N)을 계산하는 것이 가능하다면, 이를 구성하기 위해 본질적으로 CRT를 이용함으로써, 이산 대수들 모듈로 합성수 N 을 계산할 수 있다.

<35> 또한, 상기 Diffie-Hellman 키 합의 프로토콜은 다음과 같이 작동한다. 예를 들어, 앨리스와 밥과 같은 쌍방 또는 엔티티들은 공통 모듈러스 N 과 단위 군(group of units)의 큰 부분군의 생성원 g 모듈로 N 의 이용에 동의하였다. 보통은 $D-H$ 는 소수 모듈러스로써 연산하지만, 비소수 모듈러스가 여기에서 이용된다. 일반적으로, 앨리스와 밥은 각기 그들의 비밀 키들 a 및 b 를 선택하고서, 각각 $A = g^a \pmod{N}$ 및 $B = g^b \pmod{N}$ 을 계산하고, A 와 B 를 교환한다. 그리고 나서 앨리스는 공유 비밀 $S = B^a \pmod{N} \equiv g^{ab} \pmod{N}$ 을 계산할 수 있고 또한 밥도 상기 공유 비밀 $S = A^b \pmod{N} \equiv g^{ab} \pmod{N}$ 을 계산할 수 있다. 도청자들은 일반적으로 상기 이산 대수 문제 모듈로 N 을 풀지 않고서는 S 를 계산할 수 없다.

<36> 또한, 몇가지 암호기법 원형이 이하에서 기술될 것이다. 하나는 메시지 인증 코드(Message Authentication Code, MAC)로서, 비밀 대칭 키를 이용하여 메시지상의 태그(tag)를 생성하는 것이다. 동일한 비밀 키를 갖는 엔티티는 상기 태그가 정확하다는 것을 검증할 수 있으며, 이는 상기 태그가 상기 키를 알고 있는 자에 의해 생성되었고 상기 메시지가 변경되지 않았음을 의미한다. 또한 의사-난수 발생기(Pseudo-Random Generator, PRG)는 상대적으로 작은 입력을 취하고 이를 임의의 요구되는 크기의 난수로-보이는 출력에 "연장(stretch)"시킬 수 있다.

<37> 이러한 원형들 모두가 해시 함수(hash function)들, 스트림 암호(stream cipher)들, 또는 블록 암호(block cipher)들로부터 공지된 방법들로 구성될 수 있다. SHA-256과 같은 해시 함수(HASH)는, 예를 들어, 상기 Diffie-Hellman 키 합의로부터 유도되는 상기 공유 비밀을 해시함으로써 키를 유도하는데 이용될 수 있다. 결국, 키 생성을 위해, 암호적 보안 난수 발생기(cryptographically secure random number generator)가 키들을 생성하는데 이용될 수 있다.

<38> 시스템 개요

<39> 이산 대수들에 기초하는 시스템을 이용하는 디지털 서명에 두 개의 부분이 일반적으로 존재한다; 하나는 임시 키이고, 나머지 하나는 Diffie-Hellman 공유 비밀 S 로부터 유도되는 대칭 키를 이용하여 서명되는 문서의 MAC이다. 상기 임시 공개 키는 종종 큰데, 이는 서명자, 예를 들어 앨리스가, 이를 무작위로 선택된 비밀 키로부터 생성하며, 일반적으로 상기 키를 생성하는데 제어가 안되거나 거의 안되기 때문이다.

<40> 이하에서 기술되는 실시예들에서, 상기 임시 공개 키는 상기 PRG를 이용하여 작은 길이의 워드(word) 또는 개수의 문자(character)들을 공개 키로서 이용될 수 있는 어떠한 것으로 확장함으로써 생성된다. 상기 작은 길이의 워드는, 예를 들어, 숫자 시퀀스, 선택된 번호, 난수, 시간, 알파벳 또는 숫자문자 워드, 심볼 문자들, 또는 이들의 조합일 수 있다. 그리고 나서, 모듈러스 N 과 알려진 인수들의 특별한 구성을 이용하여, 앨리스는 효과적으로 그녀 고유의 공개 키를 "파괴(break)"하여 그에 상응하는 비밀 키를 구할 수 있다. 상기 인수분해를 알지

못하는, 공격자는 그렇게 할 수 없다. 상기 MAC 함수는 상기 데이터를 인증하기 위해 서명된 데이터 및 상기 유도된 키와 함께, 상기의 상황에서 유용한 임의의 크기로 절단(truncate)되어, 이용된다. 여기서, 또한 상기 출력 MAC을 발생시키는데 이용되는 키는, 만일 적절하다면, 임의의 통신에 대한 세션(session) 키로서 차후에 이용될 수도 있다.

<41> 짧은 서명들이 본질적으로 오프라인 계산에 취약할 수 있음을 유의하여야 한다. 따라서, 이러한 서명들은 특정 검증자(verifier)로 향할 수 있다. 이들은 즉석(on-the-spot) 인증(로그인, ATM 트랜잭션 등) 또는 짧고, 시간에-민감한 메시지들을 확인하는데 더 유용할 수 있다. 상기 검증자는 2-인수(two-factor) 인증을 이용하고 과도한 실패(failure)가 감지될 때 계정들을 불능화(disable)하는 것과 같은, 맹목적 공격들을 방지하기 위해 추가적인 대책들을 취할 수 있다.

<42> 따라서, 어떠한 종류의 장치들을 이용하여 상기 비밀 키를 저장하고 상기 필수적인 계산들을 수행할 수 있다. 이는 인증을 위한 인수들 중 하나를 형성한다. 2 인수 인증을 위해, 패스워드, 개인 식별 번호(personal identification number, PIN) 또는 바이오메트릭(biometric)과 같은(이에 제한되는 것은 아님) 다양한 제 2 인수가 구현될 수 있다. 예를 들어, PIN는 검증 장치에 입력될 수 있고, 서명 장치를 해제하는데 이용될 수 있고, 그리고/또는 서명 계산 자체에 편입될 수 있다.

<43> 그러므로, 앨리스의 비밀 키는 그녀가 지참하는, 예를 들어, 핸드폰과 같은, 어떠한 장치 내에 보관될 수 있다. 일반적으로, 이는 난수들로부터 상기 장치 내에서 발생되고, 그녀가 새로운 전화기를 가질 때, 일종의 백업인, 재구성하는데 이용될 수 있는 아마도 "비밀 공유(secret shares)"의 형태를 제외하고는, 상기 장치를 떠나지 않는다.

<44> 이하의 논의에서, "6 디지트 숫자"인 서명이 설명을 위해 이용된다. 그러나, 6 디지트 초과 또는 미만인 다른 워드들, 16진 디지트들, 인코딩된 숫자문자들, 또는 상기 입력 장치에 적합한 무엇도 구현될 수 있음이 당업자들에게는 명백하다. 이는 상이한 검증자에 대해서는 상이할 수 있다. 6 (십진) 디지트 숫자는 대략적으로 2^{20} 개의 가능성을 가지는데, 반면 8 숫자문자(alphanumeric) 캐릭터들의 워드는 2^{40} 개의 가능성을 가지며, 이는 맹목적-공격들에 대한 보호에 충분하다. 일부 사용례들은 아래와 같을 수 있다. 다시금, 상기 이용례들은 예시 들임을 유의하여야 하며, 다른 이용례들 및/또는 이하에서 기술되는 이용례들 내의 변형들이 구현될 수 있음은 당업자에게 명백하다.

<45> 인트라넷 로그인: 그녀의 회사 인트라넷에 로그인 하기 위해, 앨리스는, 그녀의 전화기와 같은, 장치를 이용하여 "BigCom"을 입력하고, 이는 6-디지트 숫자를 디스플레이한다. 그리고 나서 앨리스는 그녀의 사용자 이름, 그리고 패스워드 필드에, 그녀의 패스워드와 인증 번호를 타이핑할 수 있다. 이 경우, 상기 디지털 서명은 그러한 번호가 요청될 때 매번 증가되는 시퀀스 번호에 기초할 수 있다. 상기 인트라넷 인증 서버는 상기 패스워드 및 번호 모두를 검증하고, 상기 시퀀스 번호를 이용하여 재연(replay) 공격들을 방지한다.

<46> 전화기에 의한 인증: 앨리스가 그녀 자신을 전화를 통해, 예를 들어 그녀의 주식브로커와 같은, 제 3 엔티티에게 인증하고 싶어한다고 가정한다. 앨리스의 패스워드는 그녀의 전화에 입력될 수 있다. 상기 패스워드는 서명 계산에 넣어질 것이다. 그 결과인 숫자가 패스워드로서 타이핑될 것이다. 이 방법은, 예를 들어, 상기 서명이 전화를 통해 판독되거나 DTMF 톤으로서 전송될 때 유리할 수 있다. 또한 상기 패스워드는 주식브로커나 임의의 도청자들에게는 드러나지 않을 필요가 있는 것이 바람직할 수 있다.

<47> 물리적 접근: 예를 들어, 사무실 빌딩과 같은, 구조물에 들어가기 위해, 앨리스는 그녀의 전화 애플리케이션 상의 "사무실"을 선택하고, 그녀의 직원 번호 및 서명 번호를 타이핑할 수 있다. 이는 소리, DTMF 톤들, 적외선, 블루투스, 또는 RFID를 이용하여 전송함으로써 자동화될 수 있다.

<48> 집 입장: 앨리스가 집에 돌아올 때, 그녀는 그녀의 PIN을 전화기로 입력함으로써 문을 열 수 있으며, 이는 상기 인증 애플리케이션을 해제하는 효과를 갖는다. 이는 상기 문 제어기가 그녀의 키에 대해 설정된 패스워드를 가질 필요가 없지만, 그럼에도 앨리스의 전화를 홈치는 것이 들어가는데 충분하지 않을 것임을 뜻한다. 그리고 나서 그녀는 그 결과인 번호를 문을 제어하는 독립형 장치에 입력한다. 이 장치는 앨리스 및 그녀의 가족 구성원들에 대해, 다수의 상이한 공개 키들을 인식하도록 이전에 설정되었으며, 상기 서명 번호를 이용하여 가능한 키들 중 어느 것(그리고 어느 가족 구성원)이 입장이 허용될 것인지를 결정한다. (즉, 자동화된 전송 방법들이 바람직할 수 있다.)

<49> 시스템 구현

- <50> 일반적으로 인증 프로세스에는 다수의 단계들이 존재할 수 있다. 설명을 위해, 앨리스가 서명하는 엔티티이고 빅터는 검증 엔티티라고 가정하면, 상기 인증 프로세스는 다음과 같을 수 있다.
- <51> 1. 앨리스는 그녀의 공개 키를 생성한다. 앨리스는 이를 한 번만 하면 되는데, 그녀가 원하는 만큼 많은 상이한 검증자(verifier)들에게 그녀의 공개 키를 이용할 수 있기 때문이다. 그러나, 상이한 검증자들 또는 검증자들 세트에 대해 둘 이상의 공개 키들이 생성될 수 있음이 당업자에게는 명백할 것이다.
- <52> 2. 빅터는 앨리스의 공개 키를 수신하고 저장하여, 이를 예컨대, 계정, 자물쇠 및 차량들과 같이 보호할 적절한 객체와 연관시킨다. 복수의 검증자들이 있을 수 있으며, 이들은 서로 간에 신뢰할 필요가 없음을 유의하라.
- <53> 3. 빅터는 그 자신의 개인-공개 키 쌍을, 앨리스와 이용하기 위해 생성하고, 이를 앨리스에게 전달한다. 앨리스는 이를 빅터와의 상호작용(interaction)들을 위해 저장한다. 본 키 쌍은 모듈로 앨리스의 N 으로 계산되어야만 하며 다른 서명자와는 이용될 수 없음에 유의하라.
- <54> 4. 앨리스와 빅터는 입력 값을 동기화한다. 상기 입력값이 난수 또는 숫자문자(alphanumeric) 값일 수 있는 반면, 숫자 시퀀스를 이용하는 것이 이를 행하는데 효과적일 것이다. 숫자 시퀀스를 이용한다면, 상기 서명 프로세스에 대한 입력이 전달될 필요가 없다. 그렇지 않으면, 짧은 난수와 같은 입력 값이 서명의 일부로서 이용될 수 있다.
- <55> 5. 앨리스가 그녀 자신을 빅터에게 인증하기를 원할 때마다, 그녀는 상기 입력 값의 현재 값에 기초하여 짧은 인증자(authenticator)를 발생시키고, 이를 타이핑하거나, 전화 상으로 읽어서 들려주거나, DTMF 톤들을 이용하여 들을 수 있게 이를 전송하거나, 또는 필요로 하는 다른 다양한 수단들을 이용하여 전달할 수 있다.
- <56> 상기 프로세스에서, 제 1 단계는 한 차례 발생할 수 있다. 그 결과가 공개 키이기 때문에, 예를 들어 이는 공인 기관(Certificate Authority)에 제출되어, 공중 디렉토리(public directory)에 저장되는 등, 검증자들에 의한 차후의 확인을 용이하게 할 수 있다. 단계(2-4)들은 각각의 검증자에게 특정하며, 단계 2와 3은, 앨리스와 빅터의 공개키들 모두가 클 수 있어서, 온라인으로 수행될 필요가 있을 수 있다. 모듈러스 N 의 인수분해를 알고 있는, 앨리스는 빅터의 공개 키를 "과파"하여 그녀의 비밀 키를 복구할 수 있지만, 이러한 비밀 키는 앨리스에게만 특정함을 유의하라. 따라서, 앨리스는 그렇게 하여야 할 특별한 동기를 갖지 않는다.
- <57> 도 1은 예를 들어 앨리스에 의해, 디지털 서명을 발생시키는데 이용될 수 있는 예시 장치(100)를 나타낸다. 장치(100)는 입력 값으로부터 의사난수 스트링 값을 발생시키도록 구성되는 의사난수 발생(PRG) 유닛(110)을 포함할 수 있다. 또한 장치(100)는 상기 모듈러스 N 과 상기 의사난수 스트링 값에 기초하여 어느 값을 발생시키도록 구성되는 키 발생 유닛(120)을 포함할 수 있으며, 그러한 값은 공개 키로서 이용되기 위한 것이다. 키 발생 유닛(120)은 상기 발생된 공개 키에 상응하는 개인 키 값을 발생시키도록 추가로 구성될 수 있다. 추가로, 장치(100)는 상기 검증 엔티티로부터 공개 키 값을 수신하도록 구성되는 수신기(130); 상기 모듈러스 N , 상기 개인 키 값 및 상기 검증자의 공개 키 값에 기초하여 공유 비밀 값을 발생시키도록 구성되는 비밀 값 발생 유닛(140); 상기 공유 비밀 값을 이용하여 인증 서명 값을 계산하도록 구성되는 MAC 유닛(150); 및 상기 인증 서명 값 또는 디지털 서명을 전송하도록 구성되는 송신기 유닛(160)을 포함할 수 있다.
- <58> 도 2는 장치(100)에 대응하고, 예를 들어 빅터에 의해, 수신된 디지털 서명을 인증하는데 이용될 수 있는 예시 장치(200)를 나타낸다. 장치(200)는 인증 서명 값을 수신하도록 구성되는 수신기 유닛(210); 입력 값으로부터 의사난수 스트링 값을 발생시키도록 구성되는 PRG 유닛(220); 상기 모듈러스 N 과 상기 의사난수 스트링 값에 기초하여 공개 키 값을 발생시키도록 구성되는 키 발생 유닛(230); 상기 모듈러스 N , 개인 키 값 및 상기 공개 키 값에 기초하여 공유 비밀 값을 발생시키도록 구성되는 비밀 값 발생 유닛(240); 상기 공유 비밀 값을 이용하여 인증 서명 값을 계산하도록 구성되는 MAC 유닛(250); 및 상기 수신된 인증 서명 값을 상기 계산된 인증 서명 값과 비교하도록 구성되는 비교 유닛(260)을 포함할 수 있다. 상기 인증 서명들이 매치(match)되면, 상기 수신된 인증 서명은 인증될 수 있다.
- <59> 일반적으로, 하나의 엔티티의 디지털 서명은 다른 엔티티에 의해 인증될 것이며, 여기서 상기 두 개의 엔티티들은 이하에서 설명되는 바와 같이 공통 모듈러스 N 의 이용에 동의하였다. 장치(100)는 필요한 다양한 정보 또는 데이터를 저장하도록 구성되는 저장 유닛(170), 필요한 암호화(encryption)를 수행하도록 구성되는 암호화 유닛(180) 및 사용자로 하여금 장치(100)와 상호작용하게 하여주도록 구성되는 사용자 인터페이스(190)와 같은(이에 제한되는 것은 아님) 부가적인 구성요소들을 포함할 수 있음을 유의하여야 한다. 유사하게, 장치(200)는 필요한 다양한 정보 또는 데이터를 저장하도록 구성되는 저장 유닛(270)과 필요한 복호화(decryption)를 수행하도록 구성되는 복호화 유닛(280)과 같은(이에 제한되는 것은 아님) 부가적인 구성요소들을 포함할 수 있다.

- <60> 또한 상기 구성요소들(110-190) 중 하나 또는 조합이 장치(100)의 동작에 영향을 미치지 않고 재배열될 수 있음을 유의하여야 한다. 유사하게, 상기 구성요소들(210-280) 중 하나 또는 조합이 장치(200)의 동작에 영향을 미치지 않고 재배열될 수 있다. 결국, 상기 구성요소들(110-190) 중 하나 또는 조합이 장치(100)의 동작에 영향을 미치지 않고 조합되어 구현될 수 있으며 구성요소들(210-280) 중 하나 또는 조합이 장치(200)의 동작에 영향을 미치지 않고 조합되어 구현될 수 있다. 장치들(100 및 200)의 동작은 이하에서 상세하게 기술될 것이다.
- <61> 이하의 기술에서, 설명을 위해 상기 모듈러스 N 은 1024 비트라고 가정한다. 그러나, 그 크기는 1024 비트보다 작거나 또는 더 클 수 있음이 명백하다. 실제 크기가 중요한 것이 아니며, 크기를 늘리는 것이 더 큰 암호적 보안을 가져올 것이지만, 계산상의 오버헤드(overhead)를 희생하게 된다. 또한, 다른 특정 값들, 정리(theorem)들 및/또는 알고리즘들에 대한 참조는 설명을 위한 예시들이다. 따라서, 상이한 값들, 정리들 및/또는 알고리즘들도 구현될 수 있음은 당업자에게 명백할 것이다.
- <62> 단계 1: 엘리스의 공개 키를 생성
- <63> 본 시스템의 공개 키는 비소수 모듈러스 N , 그리고 유닛(unit)들(즉, 역수(multiplicative inverse)들 모듈로 N 을 갖는 엘리먼트들) 군의 생성원 g 를 포함할 수 있다. 일반적으로, 예를 들어, 2는 적어도 절반의 경우 만족스러운 생성원일 수 있으며, 그래서 이것이 참(true)이 될 때까지 가능한 N 들을 발생시키고, 변수 g 의 가능성을 무시할 가치가 있을 것이다.
- <64> N 은 다음의 기준을 만족하도록 무작위로 발생된다:
- <65> 1. N 은 소수 p 와 q 의 곱이다. $N \equiv 3 \pmod{4}$ (즉, 블럼 정수(Blum Integer)).
- <66> 2. 또는: 2는 유닛들 군의 생성원 모듈로 N 이거나, 그러한 생성원인 작은 g 를 구한다.
- <67> 3. $(p-1)/2$ 는 큰 소수 p_1 및 p_2 의 곱이다. 유사하게, $(q-1)/2$ 는 큰 소수 q_1 및 q_2 의 곱이다.
- <68> 4. p_1 , p_2 , 그리고 q_2 는 각각 평활하여, 즉, (p_1-1) 과 각각 (p_2-1) , (q_1-1) , (q_2-1) 은 1,000,000 보다 작은 별개의 소수들의 곱이다.
- <69> N 을 생성하는 상기 프로세스는 실제로는 거꾸로 진행한다. 1,000,000 보다 작은 약 50,000 개의 소수들이 존재하며, 이들 중 약 12개는 4개의 3단계 인수들(대략 총 50개) 각각에 대해 필요하다. 대략 2.8×10^{170} 개의 그러한 조합들이 존재하므로, 후보(candidate)들이 부족하지 않다; 상기 프로세스는 길 수 있지만, 한 번 수행된다. 가능한 N 들의 개수는, 대략 10^{305} 1024-비트 소수들의 상대적으로 작은 부분일지라도, 그 특별한 특징에 기초한 임의의 종류의 탐색을 방지하는데 충분하다.
- <70> 상기 50,000 또는 기초 소수들의 테이블은 애플리케이션에 제공되고 키 생성후에 폐기될 수 있거나, 또는 에라토스테네스의 체(Sieve of Eratosthenes)를 이용하여 동적으로 생성될 수 있다. 더 큰 후보 소수들의 소수성(primality)을 점검하는 것은 밀러-라빈(Miller-Rabin)과 같은 확률적 소수성 점검 알고리즘(probabilistic primality checking algorithm)이 복수 회 이어지는 시험 제법(trial division)과 같이 아마도 공지된 알고리즘들을 이용할 것이다.
- <71> 엘리스의 비밀 키는 상기 언급된 모든 다양한 소수들이며, 어떻게 이들이 함께 묶여서 N 을 형성하는가이다. 다양한 중국인의 나머지 정리 계산들을 용이하게 하는 추가적인 데이터도 저장될 수 있다.
- <72> 단계 2: 공개 키 불출
- <73> 엘리스의 공개 키는 모듈러스 N (그리고 그것이 상수 2가 아니라면, g)에 기초한다. 이는 엘리스가 그녀의 신원을 빅터에게 입증할 때 직접 빅터에게 전송될 수 있거나, 또는 그녀의 공개 키를 포함하는 증명서를 생성하고, 필요에 따라 분배할 수 있는 인증 기관에 전송될 수 있다. 공개 키이므로, 빅터가 엘리스와의 연결을 확인하는 한, 어떻게 빅터에게 도달되는지는 중요하지 않다.
- <74> 공개-키 기반이므로, 동일한 기초 키가 많은 다른 상황들에서 인증을 위해 dlyd될 수 있으며, 상기 키가 분배되는 방식은, 공개 키 기반구조(PKI) 인증서, LDAP/활성 디렉토리(LDAP/Active Directory) 등을 이용하는 것을 포함하여, 매우 유연할 수 있다.
- <75> 단계 3: 빅터의 공개 키 쌍
- <76> 빅터는 Diffie-Hellman 합의를 위한 보통의 방식으로 엘리스와 이용될 그의 키 쌍을 생성할 수 있는데, 즉 그는

난수 r 를 선택하고 $V = g^r \pmod{M}$ 을 계산한다. V 는 공개키로, 앨리스에게 전송되어야 할 필요가 있다(그리고, 앨리스가 빅터와의 접속을 확인하는 한, 민감한 정보가 아니다). 빅터의 비밀 키 v 는 앨리스에게 고유한데, 즉, 그는 다른 가입자들에 대해서는 다른 v 들을 이용하여야만 한다. 그는 v 를, 앨리스와 관련된 임의의 패스워드/생체인식(biometric) 정보와 함께, 그의 인증/권한수여 데이터베이스에 저장한다.

<77> 단계 4: 시퀀스 번호 동기화

<78> 인증 서명들을 발생시킬 때, 앨리스와 빅터 모두에 의해 이용가능한 어떠한 입력 아이템 또는 입력 값이 필요하다. 짧은 서명을 위해, 이러한 입력은 숫자 시퀀스와 같이 암시적으로 동기화되거나, 또는 작은 난수이거나, 또는 현재 시간과 관련되거나, 또는 상기 서명이 적용되는 데이터일 수도 있다. 리플레이 공격(replay attack)으로부터의 보호도 중요하며, 숫자 시퀀스는 단순한 방식으로 이러한 보호를 제공할 수 있다. 서명이 어떠한 방식으로 전송되는 데이터에 적용하기 위한 것이라면, 상기 데이터 자체는 어떻게든 리플레이 공격을 조심하여야만 한다.

<79> 설명 목적으로, 앨리스와 빅터는 숫자 시퀀스를 이용한다고 가정한다. 상기 숫자 시퀀스는 영으로 초기화되고 인증 번호가 발생/승인될 때 매번 증가될 수 있다. 빅터가 최후로 숫자 시퀀스 i 에 대한 서명을 보았다면, 그는 시퀀스 번호 $i+1, i+2, \dots$, 어떠한 상당히 작은 한도까지 가정하며 상기 서명을 체크할 수 있다. 서명이 승인될 때, 상기 시퀀스 번호는 그러한 값으로 갱신되어, 리플레이 공격들을 방지한다. 시간-기반(time-based) 시스템은 예컨대, 현재 시간의 작은 범위 내의 시간 값들에 기초하여 서명들을 승인할 것이다.

<80> 단계 5: 인증 서명 발생/점검

<81> 설명을 위해 서명되어야 하는 입력 값 또는 데이터 아이템(그것이 시퀀스 번호, 타임스탬프(timestamp), 또는 어떠한 종류의 짧은 메시지이든)이 A 라고 칭해진다고 가정한다.

<82> 1. A 가 연장된다. $D = \text{PRG}(A) \pmod{M}$ 을 계산한다. PRG는 그 입력으로부터 임의의 길이의 의사난수 스트링을 발생시키는 점을 상기하라. 상기 PRG는 모듈러스 M 내에 있는 만큼 많은 비트들을 발생시킬 수 있다. 발생될 수 있는 것보다 많거나 또는 적을 수 있는 반면, 이는 유용하지 않을 수 있으며, 보안을 위해, 정수들 모듈로 M 에 걸쳐 대략 균일하게 분배되는 것이 바람직하다. 불량한 D 값들은 무시할만한 확률로 발생할 것이다; 이들은 매우 작은 값들이거나, 또는 (M 이 우연히 인수분해된 경우인) $\text{GCD}(D, M) \neq 1$ 인 것들일 것이다.

<83> 2. D 는 Diffie-Hellman 키 합의에서 앨리스의 공개 값으로서 이용될 것이다. 그녀가 상기 프로세스를 완료하기 위해, 그녀는 상응하는 비밀 값(d)을 알 필요가 있다. 이를 위해, 그녀는 이산 대수 문제 모듈로 M 을 풀 수 있다. 앨리스는 다양한 숫자들의 인수들의 모든 세부내역들을 알기 때문에 그렇게 할 수 있다.

<84> a. 상기 대응하는 이산 대수 모듈로 p 및 모듈로 q 로부터의 중국인의 나머지 정리를 이용하여, 상기 비밀 값 d 가 계산될 수 있다.

<85> b. Pohlig-Hellman 알고리즘을 이용하여, 상기 이산 대수 모듈로 p 는 상기 이산 대수들 모듈로 p_1 및 모듈로 p_2 를 계산함으로써 결정될 수 있다. q 에 대해서도 유사하다.

<86> c. 다시 Pohlig-Hellman을 이용하면, 상기 이산 대수 모듈로 p_1 은 이산 대수들 모듈로 p_1-1 의 모든 소인수들(modulo all of the prime factors of p_1-1)을 구함으로써 결정될 수 있다.

<87> d. f 는 소수 $< 1,000,000$ 일 때, 이산 대수 모듈로 f 를 계산하는 것은, 다루기 쉬운 시간 내에, Pollard의 Rho 알고리즘, 인덱스 계산법(index calculus), 또는 역지기법(brute force)에 의해 수행될 수 있다. 이는 본 예시에서 약 50개의 f 의 값들에 대해 수행되어야 한다.

<88> 3. 앨리스는 보통의 Diffie-Hellman 방식으로 상기 공유 비밀 S 를 계산할 수 있다. $S = \text{HASH}(V^d \pmod{M})$. 시퀀스 번호가 이용된다면, 이러한 점에 대한 단계들은 상기 인증 서명이 전송되어야 하기 전에 모두 계산될 수 있다. 그러한 경우, 예를 들어, 전화기와 같은 장치는 이미 계산된 S 의 값을 이용할 수 있으며, 다음 단계들이 완료된 후에, 상대적으로 비용이 드는 다음 S 의 계산을 시작할 수 있다.

<89> 4. 앨리스는 $a = \text{MAC}(S)$ 로서 전송되는 상기 인증 서명 또는 디지털 서명을 계산할 수 있다. 대안적으로, 인증 서명은 $a = \text{MAC}(S, \text{입력 값})$, $a = \text{MAC}(S, d[\text{PIN}])$, $a = \text{MAC}(S, \text{입력 값}, d[\text{PIN}])$ 또는 $a = \text{MAC}(S, d[\text{입력 값}, \text{PIN}])$ 과 같은(이에 제한되는 것은 아님) 입력들로서 다른 값들을 이용하여 발생될 수 있으며, 여기서 $d[]$ 는 d 를 이용하는 $[]$ 의 암호화를 칭한다. 그리고 나서 상기 인증 서명이 적절히 절단(truncate)되고 요구되는 표현으로 변환될 수 있다. 상기 PIN/패스워드 및/또는 입력 값이 독립적으로 전송된다면, 이는 본 계산에 포함되지 않을 수 있

다.

- <90> 5. 어떻게든, 앨리스는 그녀의 메시지(비어 있을 수 있는)를 전송하고 상기 인증 서명 a 를 전송한다. 이는 단지 숫자 키패드 상의 6개의 디지트를 타이핑함으로써 가능할 것이다.
- <91> 6. 빅터는 상기 인증 서명을 수신하고, A 가 무엇인지(그것이 현재 시퀀스 번호, 또는 현재 시각, 또는 수신된 메시지 중 임의의 것이기 때문에)를 안다. 그는 앨리스와 동일한 방식으로 $D=PRG(A)$ 를 계산한다.
- <92> 7. 빅터는 상기 공유 비밀 $S=HASH(D^V \pmod{N})$ 을 계산한다.
- <93> 8. 빅터는 상기 수신된 a 가 앨리스와 동일한 방식으로 그가 계산한 것과 같은지를 점검한다.
- <94> 9. 상기 서명들이 매치되고, 상기 PIN/패스워드가 독립적으로 전송되었을 경우, 그 역시 매치된다면 인증은 성공적인 것으로 간주될 수 있다. 상기 시퀀스 번호 방법이 이용되었다면, 빅터는 현재의 시퀀스 번호를 $A+1$ 로 갱신한다.
- <95> Pollard의 알고리즘은 만일 그 인수들 빼기 1이 평활(smooth)하다면 N 을 인수 분해할 수 있다. p 와 q 모두가 특별히 선택되어 이러한 확률을 회피하였다. ($p-1$ 과 $q-1$ 이 모두 평활할 때 이 방법을 이용하는 것이 가능하고; 이는 $p-1$ 과 $q-1$ 의 모든 인수들에 대한 특별한 선택을 필요로 하며 결과적으로 공개 키들을 파괴하기 위한 앨리스의 작업이 인수 N 에 대한 공격자의 작업의 제공근인 방법으로 귀결됨에 유의하라. 다른 계층의 큰 소수들을 도입함으로써, 현저한 이점이 얻어진다.) N 의 구조가 임의의 다른 인수분해 알고리즘으로 하여금 무작위 N 에 대한 것보다 더 나은 확률로 성공하게 할 것이라고 믿을 이유가 없다.
- <96> 또한 성공적인 인증 후에, 상기 공유 비밀 S 는, 제한된 대역폭이라는 가정 하에서, 비효율적일 수 있을지라도, 세션 키 또는 공유 비밀 키로서 이용될 수 있다. 대안적으로, 앨리스가 빅터의 공개 키 V 를 수신할 때, 그녀는 v (앨리스와의 이용을 위한 빅터의 비밀 키)를 얻기 위한 그녀 자신의 임시 값에 대해 수행하는 바와 같이 동일한 방법으로 이를 "파괴"할 수 있다. 효율적으로는, 발생한 일은 상기 공개 키 방법이 빅터가 그의 v 를 앨리스에게 전달하는데 이용되었다는 점이다. 상기 논의된 것 중 나머지는, 상기 공유 v 를 세션 키 또는 공유 비밀 키로서 이용하여, 앨리스의 장치 및 빅터의 장치에서 동일하게 수행될 수 있다.
- <97> 따라서, 디지털 서명과 인증의 발생이 수행될 수 있다. 도 3은 상기와 같이 공통 모듈러스 N 의 이용에 합의한 두 개의 엔티티들 간의 인증에 이용하기 위한 예시 방법(300)을 나타낸다. 방법(300)에서, 의사난수 스트링 값이 입력 값으로부터 발생된다(310). 상기 입력 값은 상기와 같이 기술될 수 있으며 사용자 입력 데이터에 기초할 수 있다. 예를 들어, 상기 사용자 입력 데이터는 (이에 한정되는 것은 아님) 숫자 시퀀스, 난수, 현재 시각, 알파벳 워드, 알파벳 구(phrase), 숫자문자(alphanumeric) 작업 또는 숫자문자 구(phrase)일 수 있다.
- <98> 그리고 나서 엔티티를 인증하거나 서명하기 위한 공개 키 값이 상기 기술된 바와 같이, 즉 상기 입력 값을 연장 또는 확장하고 상기 연장된 입력 값 모듈러스 N 을 풀어서 상기 모듈러스 N 과 상기 의사난수 스트링 값에 기초하여 발생(320)된다. 상기 공개 키 값에 대응하는 개인 키 값도 발생된다(330). 상기 개인 키 값은 상기 이산 대수 문제 모듈로 N 을 풀어서 발생할 수 있다. 예를 들어, 상기 개인 키 값은 중국인의 나머지 정리, Pohlig-Hellman 알고리즘 및 Pollard의 Rho 알고리즘을 이용하여 발생할 수 있다.
- <99> 그리고 나서 공유 비밀 값이 상기 모듈러스 N , 상기 개인 키 값 및 검증자의 공개 키 값에 기초하여 발생(340)될 수 있다. 여기서, 상기 검증자의 공개 키는 상기 공유 비밀 값의 발생 전 어느 때나 수신될 수 있다. 인증 서명 값 또는 디지털 서명이 상기 공유 비밀 값을 이용하여 계산(350)된다. 여기서, 또한 상기 인증 서명이 상기 입력 값 또는 암호화된 입력 값을 이용하여 발생할 수 있다. 대안적으로, 상기 인증 서명은 상기 공유 비밀 값 및 PIN 또는 패스워드를 이용하여 발생할 수 있다. 또한 대안적으로, 상기 인증 서명은 상기 공유 비밀 값, 상기 입력 값 및 상기 PIN을 이용하여 발생할 수 있다. 상기 PIN이 이용되는 경우에, PIN은 서명 엔티티로부터 수신되어 상기 인증 서명에 이용되도록 암호화될 수 있다. 상기 입력 값(필요시)과 상기 PIN은 상기 검증자가 암호화된 키를 복호화할 수 있는 한 상기 개인 키 또는 다른 합의된 암호 키를 이용하여 암호화될 수 있다. 또한, 상기 인증 서명 값을 발생시키는데 이용된다면, 상기 (암호화된) 입력 값 및/또는 암호화된 PIN이 상기 검증자로 전송된다. 일부 경우에, 상기 입력 값은 상기 인증 서명에서 이용될 수 있지만, 예를 들어, 숫자 시퀀스가 이용된다면 전송되지 않을 수 있음을 유의하여야 한다.
- <100> 이하에서, 상기 인증 서명 값은 상기 검증 엔티티로 전송될 수 있다. 상기 인증 서명 값은 전송 전에 절단(truncate)될 수 있다.

- <101> 다양한 수단이 수현되어 방법(300)을 달성 및/또는 수행할 수 있다. 만일 상기 예시적인 장치(100)를 이용한다면, PRG 유닛(110)은 의사난수 스트링 값을 발생시킬 수 있고, 키 발생 유닛(120)은 상기 공개 키와 상기 개인 키를 발생시킬 수 있고, 비밀 값 발생 유닛(140)은 상기 비밀 값을 발생시킬 수 있으며 MAC 유닛(150)은 상기 인증 서명을 계산할 수 있다. 저장 유닛(170)은 장치(100)의 동작에 필요한 다른 데이터 및/또는 정보와 더불어 공통 모듈러스에 합의한 것들에 대한 데이터 및/또는 정보를 저장할 수 있다. 수신기 유닛(130)은 상기 검증자의 공개 키를 수신할 수 있다. 송신기 유닛(160)은 상기 인증 서명과 필요하다면 상기 입력 값을 전송할 수 있다. 또한, 암호화 유닛(180)은 상기 입력 값 및/또는 상기 PIN을 암호화할 수 있으며, 사용자 인터페이스(190)를 이용하여 사용자 입력 데이터와 PIN을 수신할 수 있다. 또한, 상기 인증 서명은 발생 후 MAC 유닛(150)에 의해 또는 전송 전에 예를 들어, 처리기(미도시)나 송신기 유닛(160)과 같은 별도의 엘리먼트에 의해 절단될 수 있다.
- <102> 도 4는 공통 모듈러스 N 의 이용에 합의한 두 개의 엔티티들 간의 인증에 이용하기 위한 방법(300)에 대응하는 예시 방법(400)을 나타낸다. 방법(400)에서, 인증 서명 값이 인증을 위해 수신(410)된다. 인증을 위해, 의사난수 스트링 값이 입력 값으로부터 발생(420)된다. 여기서, 상기 의사난수 스트링 값은 방법(300)에서와 동일한 방식으로 발생될 수 있다.
- <103> 그리고 나서 공개 키 값이 상기 모듈러스 N 및 상기 의사난수 스트링 값을 기초로 방법(300)과 동일한 방식으로 발생되며; 공유 비밀 값은 상기 모듈러스 N , 개인 키 값 및 상기 공개 키 값을 기초로 방법(300)과 동일한 방식으로 발생된다. 여기서, 상기 개인 키는 상기 서명 엔티티(signing entity)의 공개 키에 대응하는 개인 키일 것이며 또한 방법(300)과 동일한 방식으로 발생될 수 있다. 즉, 상기 개인 키는 상기 이산 대수 문제 모듈로 N 을 풀어서 발생될 수 있다.
- <104> 인증 서명 값은 방법(300)에서와 동일한 방식으로 상기 공유 비밀 값을 이용하여 발생(450)된다. 따라서, 상기 인증 서명은 상기 입력 값 또는 암호화된 입력 값을 이용하여 발생될 수도 있다. 대안적으로, 상기 인증 서명은 상기 공유 비밀 값 및 PIN 또는 패스워드를 이용하여 발생될 수 있다. 또한 대안적으로, 상기 인증 서명은 상기 공유 비밀 값, 상기 입력 값 및 상기 PIN을 이용하여 발생될 수 있다. PIN이 이용되는 경우에, 암호화된 PIN은 상기 서명 엔티티로부터 수신되고 상기 인증 서명에 이용된다. 입력 값이 이용되는 경우에, 상기 입력 값은 일부의 경우에 상기 서명 엔티티로부터 수신될 수 있다. 그러나, 일부의 경우, 상기 입력 값은, 예를 들어, 숫자 시퀀스가 이용된다면 수신될 필요가 없다.
- <105> 상기 수신된 인증 서명 값은 계산된 인증 서명 값과 비교(460)된다. 상기 수신된 인증이 인증 서명의 절단(truncate)된 부분일 수 있어서, 상기 수신된 인증 서명 값은 상기 계산된 인증 서명 값의 절단된 부분과 비교될 수 있다. 만일 상기 두 개의 서명 값들이 매치(match)되면, 상기 수신된 인증 서명은 인증될 수 있다. 여기서, PIN이 상기 인증 프로세스에 이용될 수도 있다. 즉, 2-인수(two-factor) 인증을 제공하기 위해, 상기 서명 엔티티에 대응하는 PIN이 저장될 수 있다. 따라서, 암호화된 PIN이 수신된다면, 상기 PIN은 복호화(decrypt)되고 상기 저장된 PIN과 비교될 수 있다. 상기 인증 서명 값들과 상기 PIN들이 모두 매치되면, 상기 인증 서명은 인증된다. 필요하다면, 상기 입력 값과 상기 PIN은 상기 검증자가 상기 암호화된 키를 복호화할 수 있는 한 상기 서명 엔티티 또는 암호 키에 동의한 다른 엔티티의 개인 키를 이용하여 복호화될 수 있다.
- <106> 다양한 수단이 방법(400)을 달성 및/또는 수행하도록 구현될 수 있다. 상기 예시 장치(200)를 이용한다면, 수신기 유닛(210)은, 필요하다면 상기 인증 서명 값과 상기 입력 값 및/또는 상기 암호화된 PIN을 수신할 수 있다. PRG 유닛(220)은 상기 의사난수 스트링 값을 발생시킬 수 있고, 키 발생 유닛(230)은 상기 공개 키 및 상기 개인 키를 발생시킬 수 있고, 비밀 값 발생 유닛(240)은 상기 공유 비밀 값을 발생시킬 수 있고, MAC 유닛(250)은 상기 인증 서명을 계산할 수 있으며 비교 유닛(260)은 상기 인증 서명들과 PIN을 비교할 수 있다. 저장 유닛(270)은 장치(200)의 동작에 필요한 다른 데이터 및/또는 정보 뿐 아니라 상기 동의된 공통 모듈러스에 관련된 데이터 및/또는 정보를 저장할 수 있다. 복호화 유닛(280)은 필요에 따라, 상기 입력 값 및/또는 PIN을 복호화할 수 있다.
- <107> 그리하여, 디지털 서명이 발생, 전송 및 인증될 수 있다.
- <108> 다양한 다른 인증이 이뤄질 수 있음을 유의하여야 한다. 예를 들어, 도 5는 다른 엔티티의 모듈러스 N 에 기초하여 발생된 공개 키가 수신(510)되는 대안적인 방법(500)을 나타낸다. 상기 수신된 공개 키에 대응하는 개인 키도 이산 대수 문제 모듈로 N 을 풀어서 발생(520)될 수 있다. 이하에서, 상기 검증자의 공개 키에 대응하는 개인 키는 두 개의 엔티티들 간의 향후의 통신에서 공유 비밀 키 또는 대칭 키로서 이용(530)될 수 있다.

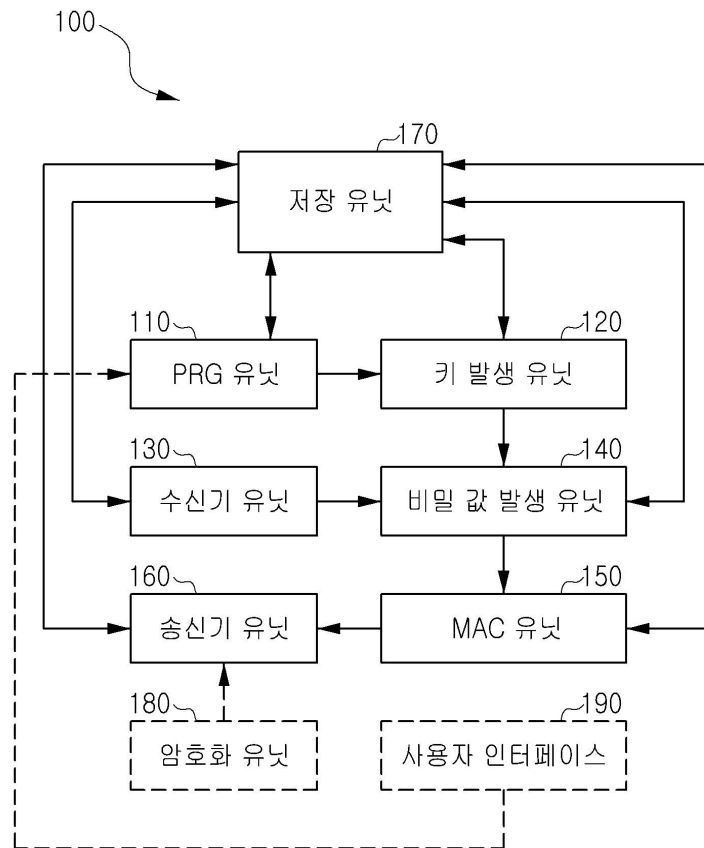
- <109> 또한, 방법들(300, 400 및 500)은 도 6에 도시된 바와 같이 저장 매체(610) 및 처리기(620)에 의해 구현될 수 있다. 저장 매체(610)는 방법들(300, 400 및 500)을 수행하도록 구성되는 명령(instruction)들을 포함할 수 있다. 처리기(620)는 예를 들어, 상기 저장 매체에 저장된 상기 명령들을 실행함으로써 방법들(300, 400 및 500)의 동작을 제어할 수 있다.
- <110> 추가로, 상기 디지털 서명들은 짧을 수 있기 때문에, 이들은 예를 들어, 전화를 통해 판독되고, 키패드로 입력되거나, 또는 서식 상에 수기(handwrite)될 수 있는, 모든 종류의 공간 또는 대역폭이 제한된 상황들을 따를 여지가 있다. 그러므로, 이들은 일반적 상황에서의 위조(forgery)를 완전히 방지할 수는 없다. 그럼에도, 그러한 서명들이 일반적 상황들에 이용될 수 있는 방안들이 존재한다. 예를 들어, 상기 서명이 시간-의존적(time-dependent)이고 짧은 시간 동안 유효하도록 구현될 수 있다. 상기 서명 입력은 상대적으로 타진불가(untappable)하게 만들어질 수도 있다(예를 들어, 국부 검증(local verification)을 행하는 문 위의 키패드에 상기 서명을 입력함). 검증 엔티티는, 인증하려는 성공적이지 못한 한 쌍 이상의 시도들이 빠르게 연속되면, 잠시 동안 디스에이블(disable), 또는 폐쇄할 수 있다. 이는 PIN이 연속적으로 3번 잘못 입력되면 ATM이 카드를 "삼키는(eat)" 방식에 대한 원리와 유사하다.
- <111> 대안적으로, 상기 서명들은 특정한 검증 엔티티, 즉, 하나의 특정한 검증자 스스로의 공개 키를 목표로 하도록 생성될 수 있다. 다시 말해, 상기 검증자는 너무 많은 성공적이지 않은 인증 시도들이 행해진다면 상기 계정을 "폐쇄(lock out)"할 수 있다. 그러나, 상기 서명이 반드시 시간-의존적일 필요는 없으며, 상기 시도들은 시간적으로 간격이 있을 수 있다. 이는 로그인 패스워드의 현재 처리와 유사하다.
- <112> 결국, 상기 실시예들이 엔티티들인 앨리스와 빅터를 참조로 하여 기술된 반면, 다른 종류의 엔티티도 적용가능함은 명백하다. 또한, 상기 구현들이 특정한 조건들과 값들을 참조로 하여 기술되었지만, 다른 조건들과 값들도 일반적인 인증 프로세스에 영향을 미치지 않고 적용될 수 있음이 명백하다.
- <113> 따라서, 전술한 실시예들은 단지 예시들이며 본 발명을 제한하는 것으로 해석되어서는 안된다. 상기 실시예들의 기술은 예시적인 것이며, 청구항들의 범위를 제한하고자 하는 것이 아니다. 그리하여, 본 설명들은 다른 종류의 장치들에 용이하게 적용될 수 있으며 많은 대안들, 수정들, 그리고 변형들은 다업자에게 명백할 것이다.

도면의 간단한 설명

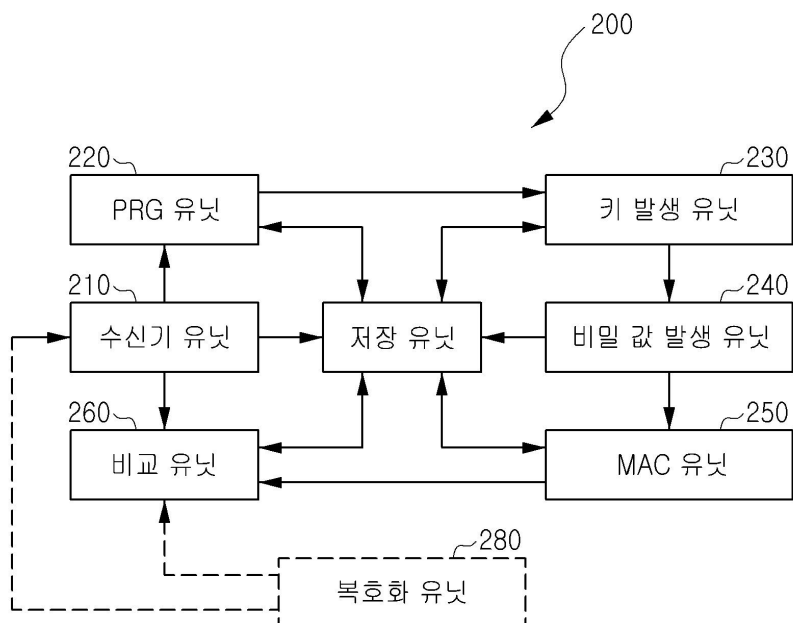
- <11> 동일한 참조 부호는 동일한 구성요소를 지시하는 이하의 도면들을 참조로하여 다양한 실시예들이 상세히 설명될 것이며:
- <12> 도 1은 디지털 서명을 발생시키기 위해 엔티티에 의해 이용되는 예시적인 장치를 나타낸다;
- <13> 도 2는 디지털 서명을 인증하기 위해 검증 엔티티(verifying entity)에 의해 이용되는 예시적인 장치를 나타낸다;
- <14> 도 3은 디지털 서명을 발생시키는데 이용되는 예시적인 방법을 나타낸다;
- <15> 도 4는 디지털 서명을 인증하는데 이용되는 예시적인 방법을 나타낸다;
- <16> 도 5는 인증에 이용되는 다른 예시적인 방법을 나타낸다; 그리고
- <17> 도 6은 인증에 이용되는 다른 예시적인 장치를 나타낸다.

도면

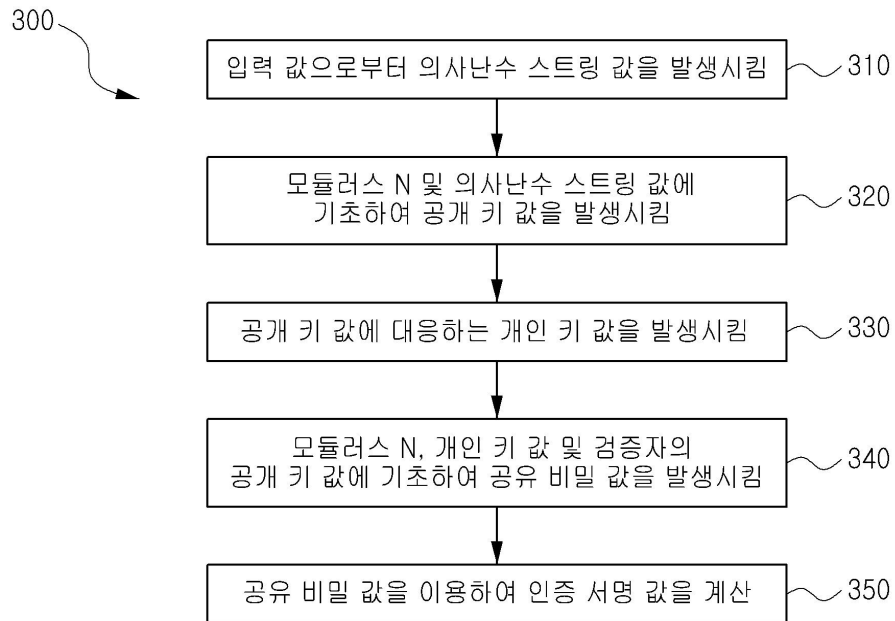
도면1



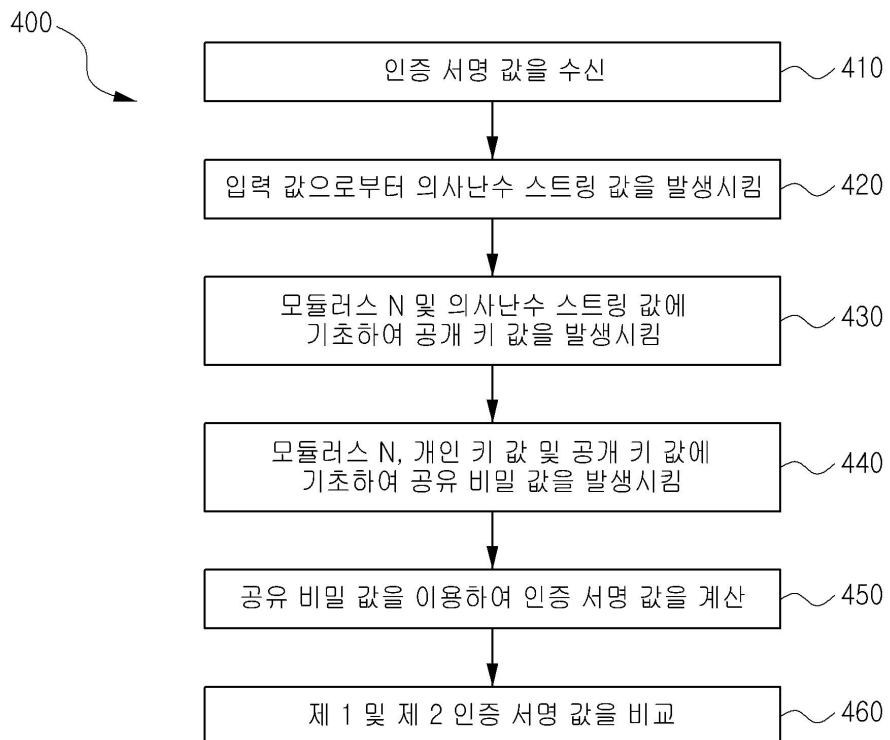
도면2



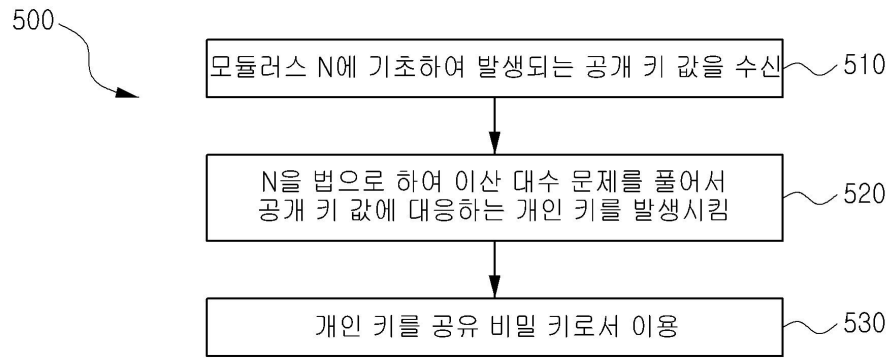
도면3



도면4



도면5



도면6

