(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2019-0049133
(43) 공개일자 2019년05월09일

(51) 국제특허분류(Int. Cl.)
H04L 9/08 (2006.01) *H04L 9/06* (2006.01)
(52) CPC특허분류
H04L 9/085 (2013.01)
H04L 67/12 (2013.01)
(21) 출원번호 10-2017-0144660
(22) 출원일자 2017년11월01일
심사청구일자 2017년11월01일

(71) 출원인
경희대학교 산학협력단
경기도 용인시 기흥구 덕영대로 1732 (서천동, 경희대학교 국제캠퍼스내)
(72) 발명자
조진성
경기도 수원시 영통구 봉영로 1620, 101동 1801호 (영통동, 대우 월드마크 영통)
김병선
경기도 용인시 기흥구 덕영대로 1732 경희대학교 (서천동)
(뒷면에 계속)
(74) 대리인
특허법인도담

전체 청구항 수 : 총 16 항

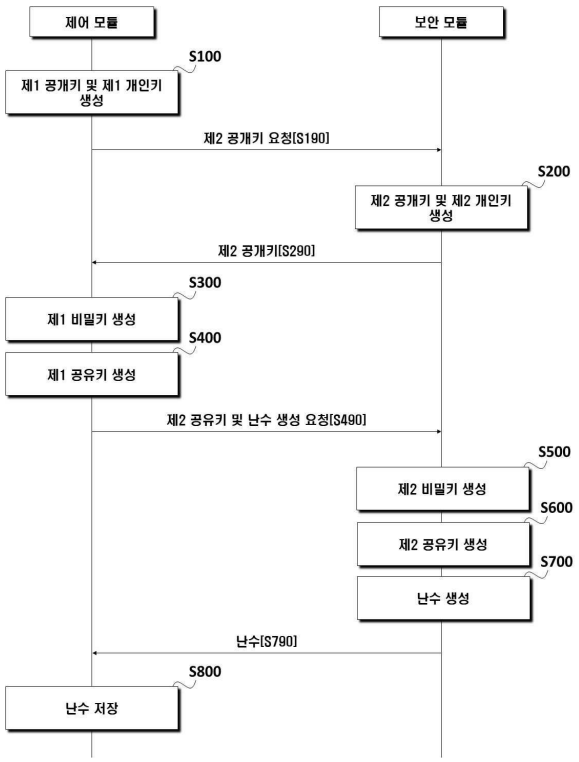
(54) 발명의 명칭 **보안 공유키 교환 방법 및 시스템**

(57) 요 약

본 발명은 제어 모듈과 보안 모듈 간 보안 공유키 교환 방법 및 시스템에 대한 것으로, IoT 디바이스가 부팅되면, 상기 제어 모듈이 제1 공개키 및 제1 개인키를 생성하고, 상기 제어 모듈이 상기 보안 모듈에 비밀키 생성을 위한 상기 보안 모듈의 공개키를 요청하고, 상기 보안 모듈이 제2 공개키 및 제2 개인키를 생성하고 상기

(뒷면에 계속)

대 표 도



제어 모듈에 상기 제2 공개키를 전송하고, 상기 제어 모듈은 상기 제1 개인키와 상기 제2 공개키를 이용하여 제1 비밀키를 생성하고, 상기 제1 비밀키를 이용하여 제1 공유키를 생성하고, 상기 제어 모듈이 상기 보안 모듈에 상기 제1 공개키를 전송하면, 상기 보안 모듈이 상기 제2 개인키와 상기 제1 공개키를 이용하여 제2 비밀키를 생성하고, 상기 제2 비밀키를 이용하여 제2 공유키를 생성하는 단계를 포함하고, 상기 제1 공유키와 상기 제2 공유키는 상기 보안 모듈과 상기 제어 모듈 간 송수신 데이터의 암호화 및 복호화에 사용되는 것으로, 동일한 값을 갖는 것을 특징으로 한다.

(52) CPC특허분류

H04L 9/0643 (2013.01)

H04L 9/0825 (2013.01)

(72) 발명자

이기영

경기도 성남시 중원구 도촌남로 22, 113동 404호 (도촌동, 휴먼시아섬마을1단지아파트)

이기웅

경기도 군포시 수리산로 203번길 14 (산본동)

조성희

경기도 용인시 수지구 성복2로 86, 203동 1104호 (성복동, 성동마을LG빌리지2차아파트)

윤중철

경기도 화성시 삼성전자로 1-1 DSR Security 제품개발팀 (반월동)

강태규

경기도 수원시 영통구 영일로 22, 103호 (영통동)

명 세 서

청구범위

청구항 1

제어 모듈과 보안 모듈 간 보안 공유키 교환 방법에 있어서,

IoT 디바이스가 부팅되면, 상기 제어 모듈이 제1 공개키 및 제1 개인키를 생성하는 단계;

상기 제어 모듈이 상기 보안 모듈에 비밀키 생성을 위한 상기 보안 모듈의 공개키를 요청하는 단계;

상기 보안 모듈이 제2 공개키 및 제2 개인키를 생성하고 상기 제어 모듈에 상기 제2 공개키를 전송하는 단계;

상기 제어 모듈은 상기 제1 개인키와 상기 제2 공개키를 이용하여 제1 비밀키를 생성하고, 상기 제1 비밀키를 이용하여 제1 공유키를 생성하는 단계;

상기 제어 모듈이 상기 보안 모듈에 상기 제1 공개키를 전송하면, 상기 보안 모듈이 상기 제2 개인키와 상기 제1 공개키를 이용하여 제2 비밀키를 생성하고, 상기 제2 비밀키를 이용하여 제2 공유키를 생성하는 단계를 포함하고,

상기 제1 공유키와 상기 제2 공유키는 상기 보안 모듈과 상기 제어 모듈 간 송수신 데이터의 암호화 및 복호화에 사용되는 것으로, 동일한 값을 갖는 것을 특징으로 하는 보안 공유키 교환 방법.

청구항 2

제1항에 있어서,

상기 제1 공개키 및 상기 제1 비밀키는 한 쌍의 비대칭키이고,

상기 제2 공개키 및 상기 제2 비밀키는 한 쌍의 비대칭키인 것을 특징으로 하는 보안 공유키 교환 방법.

청구항 3

제1항에 있어서,

상기 제1 공개키, 상기 제1 개인키, 상기 제2 공개키, 상기 제2 개인키, 상기 제1 비밀키 및 상기 제2 비밀키 중 적어도 하나는 ECDH 함수를 이용하여 생성되는 보안 공유키 교환 방법.

청구항 4

제1항에 있어서,

상기 제1 비밀키 및 상기 제2 비밀키는 SHA1 함수를 이용하여 암호화되는 보안 공유키 교환 방법.

청구항 5

제4항에 있어서,

상기 제1 공유키 및 상기 제2 공유키는 암호화된 제1 비밀키 및 암호화된 제2 비밀키의 상위 16 bytes를 추출하여 생성되는 보안 공유키 교환 방법.

청구항 6

제1항에 있어서,

상기 제1 비밀키와 상기 제2 비밀키는 동일한 값을 갖는 보안 공유키 교환 방법.

청구항 7

제1항에 있어서,

상기 보안 모듈이 상기 제2 공유키를 생성하면, 상기 보안 모듈은 데이터 중복 방지를 위한 난수를 추가로 생성하여 상기 제어 모듈에 전송하는 보안 공유키 교환 방법.

청구항 8

제7항에 있어서,

상기 제어 모듈과 상기 보안 모듈은 상기 송수신 데이터에 상기 난수를 추가하고, 상기 송수신 데이터를 상기 제1 공유키 및 상기 제2 공유키 중 적어도 하나로 암호화하여 통신하는 보안 공유키 교환 방법.

청구항 9

보안 공유키 교환 시스템의 제어 모듈에 있어서,

상기 제어 모듈의 비대칭키 쌍인 제1 공개키 및 제1 개인키를 생성하고, 상기 제1 개인키와 보안 모듈의 공개키를 이용하여 제1 비밀키를 생성하는 비밀키 생성부;

상기 제1 비밀키를 이용하여 제1 공유키를 생성하는 공유키 생성부;

상기 보안 모듈에 상기 보안 모듈의 공개키를 요청하며, 상기 보안 모듈에 상기 제1 공개키를 전송하고, 데이터 중복 방지를 위해 상기 보안 모듈이 생성한 난수를 수신하는 통신부를 포함하는 보안 공유키 교환 시스템의 제어 모듈.

청구항 10

제9항에 있어서,

상기 비밀키 생성부는 ECDH 함수를 이용하여 상기 제1 비밀키를 생성하는 보안 공유키 교환 시스템의 제어 모듈.

청구항 11

제9항에 있어서,

상기 공유키 생성부는 SHA1 함수를 이용하여 상기 제1 비밀키를 암호화하고, 암호화된 제1 비밀키의 상위 16 bytes를 추출하여 상기 제1 공유키를 생성하는 보안 공유키 교환 시스템의 제어 모듈.

청구항 12

제9항에 있어서,

상기 제1 공유키는 상기 난수가 추가된 송수신 데이터의 암호화 및 복호화에 사용되는 보안 공유키 교환 시스템의 제어 모듈.

청구항 13

보안 공유키 교환 시스템의 보안 모듈에서 있어서,

상기 보안 모듈의 비대칭키 쌍인 제2 공개키 및 제2 개인키를 생성하고, 상기 제2 개인키와 제어 모듈의 공개키를 이용하여 제2 비밀키를 생성하는 비밀키 생성부;

상기 제2 비밀키를 이용하여 제2 공유키를 생성하는 공유키 생성부;

데이터 중복 방지를 위한 난수를 생성하는 난수 생성부;

상기 제어 모듈로부터 상기 제어 모듈의 공개키를 수신하고, 상기 제2 공개키 및 상기 난수를 제어 모듈에 전송하는 통신부;

상기 난수를 저장하는 데이터부를 포함하는 보안 공유키 교환 시스템의 보안 모듈.

청구항 14

제13항에 있어서,

상기 비밀키 생성부는 ECDH 함수를 이용하여 상기 제2 비밀키를 생성하는 보안 공유키 교환 시스템의 보안 모듈.

청구항 15

제13항에 있어서,

상기 공유키 생성부는 SHA1 함수를 이용하여 상기 제2 비밀키를 암호화하고, 암호화된 제2 비밀키의 상위 16 bytes를 추출하여 상기 제2 공유키를 생성하는 보안 공유키 교환 시스템의 보안 모듈.

청구항 16

제13항에 있어서,

상기 제2 공유키는 상기 난수가 추가된 송수신 데이터의 암호화 및 복호화에 사용되는 보안 공유키 교환 시스템의 보안 모듈.

발명의 설명

기 술 분 야

[0001] 본 발명은 보안 공유키 교환 방법 및 시스템에 관한 것으로, 보다 구체적으로 보안 모듈과 제어 모듈 사이에서 보안 공유키를 교환하는 방법 및 시스템에 관한 것이다.

배 경 기 술

[0003] IoT(Internet Of Things)는 각종 사물에 센서와 통신 기능을 내장하여 인터넷에 연결하는 기술로 최근 가전, 자동차, 스마트시티 등과 같은 서비스 분야에 폭넓게 활용되고 있어 실생활에서 쉽게 접할 수 있다. 미국의 정보기술 연구 및 자문 회사인 가트너가 2020년에 약 250억개의 IoT 디바이스가 존재할 것이라고 관측할 정도로 IoT 디바이스의 성장 및 보급 속도가 증가하고 있는 추세이다.

[0004] 그러나 IoT가 보편화됨과 비례하여 다양한 부작용이 발생할 수 있다. IoT는 센서 기술, 통신 기술, 칩 기술, 운영체제 기술, 임베디드 시스템 기술, 플랫폼 기술, 빅데이터 기술, 텍스트 마이닝 기술, 오픈 API 기술 등 다양한 요소기술이 통합되어 사용되기 때문에 기술 혹은 구현하는 방법의 문제에 따라 보안 취약점이 존재할 수 있다. 또한 IoT 디바이스는 사양이 낮기 때문에 다양한 공격을 받을 위험이 존재한다. 실제로 IoT 환경의 보안적

인 취약점을 악용하여 사생활을 침해한 사례도 있다. 예를 들어 CCTV의 영상을 불특정 다수에게 유포하고, 아파트 도어록 해킹 후 출입문을 열거나, 자동차의 문 잠금을 해제한 뒤 절도하는 등의 다양한 피해가 발생하였다.

[0005] 그에 따라 보안 솔루션 기업에서는 IoT 디바이스의 보안성을 향상시키기 위한 연구를 진행하고 있으며, 그 중 하드웨어 보안 모듈을 IoT 디바이스에 장착하여 IoT 디바이스가 수행하지 못하는 검증 및 인증을 수행함으로써 IoT 디바이스의 취약점을 해결하고 있다. 그러나 IoT 디바이스는 검증 및 인증 절차를 수행하기에 앞서 검증 대상 데이터를 하드웨어 보안 모듈에 전송해야 하기 때문에 그 과정에서 검증 대상 데이터가 외부에 노출될 수 있는 위험성이 존재한다.

[0006] 이러한 위험으로 인해 주로 발생하는 공격 중 재전송 공격이나 도청 공격은 IoT 디바이스와 보안 모듈 사이에서 교환되는 데이터를 탐지하고 IoT 디바이스의 정보를 탈취하여 악의적인 행위에 사용할 수 있다. 이러한 공격에 노출되게 되면 외부에서 데이터를 분석하여 검증 및 인증 절차를 우회함으로써 악의적인 목적을 갖는 데이터를 기존 데이터에 주입할 수 있기 때문에 IoT 디바이스의 제어권을 탈취당하거나 중요 데이터 및 개인 정보가 유출될 수 있다.

[0007] 이를 방지하기 위해 상호 인증을 통한 공개키 기반 구조, 암호키 등을 이용한 상호 인증, 대기 시간 점검, 비공개 채널로 키를 주고 받고 공개 채널로 암호문을 보내는 양자 암호 등의 방법이 사용되고 있다. 대표적으로 공개 키 기반 구조는 요청자에게 제공되고 타인과 공유할 수 없는 개인키와 모든 사람이 접근할 수 있는 공개키를 이용한다. 보다 구체적으로 A가 B에게 메시지를 송신할 때, A는 B의 공개키를 사용하여 메시지를 암호화하고 A의 개인키를 이용하여 디지털 인증서를 암호화하여 함께 송신한다. B가 메시지를 수신하게 되면 B의 개인키를 통해 복호화하여 메시지를 확인하고, 함께 수신한 디지털 인증서를 통해 A에게 온 메시지라는 것을 확인한다.

[0008] 상술한 바와 같이 데이터 교환 시에 발생하는 다양한 보안상의 취약점을 대비하기 위한 종래의 기술로는 대한민국 공개특허공보 제10-2016-0035999호 “데이터 통신 보안을 위한 방법, 장치 및 시스템”이 있으며, 제1 디바이스가 데이터를 암호화하기 위한 암호화키를 생성하여 키 식별정보를 생성하고, 데이터 암호화를 수행하며, 암호화된 데이터 및 키 식별정보를 포함하는 데이터를 제2 디바이스로 송신하는 것을 특징으로 한다.

[0009] 기존 데이터 교환 방법의 암호화 및 검증 방법은 별도로 구비된 보안 모듈과의 데이터 교환에 있어서 보안성이 취약하다는 단점이 있다. 이를 해결하기 위한 방안으로 데이터 암호화에 필요한 공유키를 생성하여 데이터를 암호화 및 복호화하고 난수를 생성하여 데이터를 랜덤화하여 데이터를 검증함으로써 IoT 디바이스 내부에서 발생하는 데이터 교환의 보안성을 향상시킬 필요가 있다.

선행기술문헌

특허문헌

[0011] (특허문헌 0001) 대한민국 공개특허공보 제10-2016-0035999호(2016년 04월 01일)

발명의 내용

해결하려는 과제

[0012] 본 발명은 전술한 문제를 해결하기 위한 것으로, IoT 디바이스 내 제어 모듈과 보안 모듈 간 통신에서 발생할 수 있는 보안 상의위험성을 차단하는 것을 일 목적으로 한다.

[0013] 또한, 본 발명은 보안 공유키를 통해 데이터를 암호화하여 외부의 공격으로부터 데이터를 안전하게 보호하는 것을 일 목적으로 한다.

[0014] 또한, 본 발명은 보안 공유키뿐만 아니라 난수를 생성하여 데이터를 재사용하는 공격을 차단하는 것을 일 목적으로 한다.

[0015] 또한, 본 발명은 매 부트 단계마다 보안 공유키를 초기화하고 생성함으로써 보안성을 더욱 향상시킬 수 있도록 하는 것을 일 목적으로 한다.

과제의 해결 수단

[0017] 이러한 목적을 달성하기 위한 본 발명은 보안 공유키 교환 방법에 있어서, IoT 디바이스가 부팅되면, 상기 제어 모듈이 제1 공개키 및 제1 개인키를 생성하는 단계, 상기 제어 모듈이 상기 보안 모듈에 비밀키 생성을 위한 상기 보안 모듈의 공개키를 요청하는 단계, 상기 보안 모듈이 제2 공개키 및 제2 개인키를 생성하고 상기 제어 모듈에 상기 제2 공개키를 전송하는 단계, 상기 제어 모듈은 상기 제1 개인키와 상기 제2 공개키를 이용하여 제1 비밀키를 생성하고, 상기 제1 비밀키를 이용하여 제1 공유키를 생성하는 단계, 상기 제어 모듈이 상기 보안 모듈에 상기 제1 공개키를 전송하면, 상기 보안 모듈이 상기 제2 개인키와 상기 제1 공개키를 이용하여 제2 비밀키를 생성하고, 상기 제2 비밀키를 이용하여 제2 공유키를 생성하는 단계를 포함하고, 상기 제1 공유키와 상기 제2 공유키는 상기 보안 모듈과 상기 제어 모듈 간 송수신 데이터의 암호화 및 복호화에 사용되는 것으로, 동일한 값을 갖는 것을 일 특징으로 한다.

[0018] 또한 상기 제1 공개키 및 상기 제1 비밀키는 한 쌍의 비대칭키이고, 상기 제2 공개키 및 상기 제2 비밀키는 한 쌍의 비대칭키인 것을 일 특징으로 한다.

[0019] 나아가 상기 제1 공개키, 상기 제1 개인키, 상기 제2 공개키, 상기 제2 개인키, 상기 제1 비밀키 및 상기 제2 비밀키 중 적어도 하나는 ECDH 함수를 이용하여 생성되는 것을 일 특징으로 한다.

[0020] 또한 상기 제1 비밀키 및 상기 제2 비밀키는 SHA1 함수를 이용하여 암호화되는 것을 일 특징으로 한다.

[0021] 나아가 상기 제1 공유키 및 상기 제2 공유키는 암호화된 제1 비밀키 및 암호화된 제2 비밀키의 상위 16 bytes를 추출하여 생성되는 것을 일 특징으로 한다.

[0022] 또한 상기 제1 비밀키와 상기 제2 비밀키는 동일한 값을 갖는 것을 일 특징으로 한다.

[0023] 나아가 상기 보안 모듈이 상기 제2 공유키를 생성하면, 상기 보안 모듈은 데이터 중복 방지를 위한 난수를 추가로 생성하여 상기 제어 모듈에 전송하는 것을 일 특징으로 한다.

[0024] 또한 상기 제어 모듈과 상기 보안 모듈은 상기 송수신 데이터에 상기 난수를 추가하고, 상기 송수신 데이터를 상기 제1 공유키 및 상기 제2 공유키 중 적어도 하나로 암호화하여 통신하는 것을 일 특징으로 한다.

[0025] 나아가 본 발명은 보안 공유키 교환 시스템의 제어 모듈에 있어서, 상기 제어 모듈의 비대칭키 쌍인 제1 공개키 및 제1 개인키를 생성하고, 상기 제1 개인키와 보안 모듈의 공개키를 이용하여 제1 비밀키를 생성하는 비밀키 생성부, 상기 제1 비밀키를 이용하여 제1 공유키를 생성하는 공유키 생성부, 상기 보안 모듈에 상기 보안 모듈의 공개키를 요청하며, 상기 보안 모듈에 상기 제1 공개키를 전송하고, 데이터 중복 방지를 위해 상기 보안 모듈이 생성한 난수를 수신하는 통신부를 포함하는 것을 일 특징으로 한다.

[0026] 또한 상기 비밀키 생성부는 ECDH 함수를 이용하여 상기 제1 비밀키를 생성하는 것을 일 특징으로 한다.

[0027] 나아가 상기 공유키 생성부는 SHA1 함수를 이용하여 상기 제1 비밀키를 암호화하고, 암호화된 제1 비밀키의 상위 16 bytes를 추출하여 상기 제1 공유키를 생성하는 것을 일 특징으로 한다.

[0028] 또한 상기 제1 공유키는 상기 난수가 추가된 송수신 데이터의 암호화 및 복호화에 사용되는 것을 일 특징으로 한다.

[0029] 나아가 본 발명은 보안 공유키 교환 시스템의 보안 모듈에서 있어서, 상기 보안 모듈의 비대칭키 쌍인 제2 공개키 및 제2 개인키를 생성하고, 상기 제2 개인키와 제어 모듈의 공개키를 이용하여 제2 비밀키를 생성하는 비밀키 생성부, 상기 제2 비밀키를 이용하여 제2 공유키를 생성하는 공유키 생성부, 데이터 중복 방지를 위한 난수를 생성하는 난수 생성부, 상기 제어 모듈로부터 상기 제어 모듈의 공개키를 수신하고, 상기 제2 공개키 및 상기 난수를 제어 모듈에 전송하는 통신부, 상기 난수를 저장하는 데이터부를 포함하는 것을 일 특징으로 한다.

[0030] 또한 상기 비밀키 생성부는 ECDH 함수를 이용하여 상기 제2 비밀키를 생성하는 것을 일 특징으로 한다.

[0031] 나아가 상기 공유키 생성부는 SHA1 함수를 이용하여 상기 제2 비밀키를 암호화하고, 암호화된 제2 비밀키의 상위 16 bytes를 추출하여 상기 제2 공유키를 생성하는 것을 일 특징으로 한다.

[0032] 또한 상기 제2 공유키는 상기 난수가 추가된 송수신 데이터의 암호화 및 복호화에 사용되는 것을 일 특징으로 한다.

발명의 효과

[0034] 전술한 바와 같은 본 발명에 의하면, IoT 디바이스 내부에서 제어 모듈과 보안 모듈 간 통신에서 발생할 수 있는 보안 상의 위험성을 차단할 수 있다.

[0035] 또한, 본 발명은 보안 공유키를 통해 데이터를 암호화하여 외부의 공격으로부터 데이터를 안전하게 보호할 수 있다.

[0036] 또한, 본 발명은 보안 공유키뿐만 아니라 난수를 생성하여 데이터를 재사용하는 공격을 차단할 수 있다.

[0037] 또한, 본 발명은 매 부트 단계마다 보안 공유키를 초기화하고 생성함으로써 보안성을 더욱 향상시킬 수 있다.

도면의 간단한 설명

[0039] 도1은 본 발명의 일 실시 예에 의한 보안 공유키 교환 시스템의 구성을 도시한 도면이다.

도2는 본 발명의 일 실시 예에 의한 보안 공유키 교환 방법을 설명하기 위한 도면이다.

도3은 본 발명의 일 실시 예에 의한 데이터 검증 방법을 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

[0040] 전술한 목적, 특징 및 장점은 첨부된 도면을 참조하여 상세하게 후술되며, 이에 따라 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 본 발명의 기술적 사상을 용이하게 실시할 수 있을 것이다. 본 발명을 설명함에 있어서 본 발명과 관련된 공지 기술에 대한 구체적인 설명이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우에는 상세한 설명을 생략한다.

[0041] 도면에서 동일한 참조부호는 동일 또는 유사한 구성요소를 가리키는 것으로 사용되며, 명세서 및 특허청구의 범위에 기재된 모든 조합은 임의의 방식으로 조합될 수 있다. 그리고 다른 식으로 규정하지 않는 한, 단수에 대한 언급은 하나 이상을 포함할 수 있고, 단수 표현에 대한 언급은 또한 복수 표현을 포함할 수 있음이 이해되어야 한다.

[0042] 본 명세서에서 사용되는 용어는 단지 특정 예시적 실시 예들을 설명할 목적을 가지고 있으며 한정할 의도로 사용되는 것이 아니다. 본 명세서에서 사용된 바와 같은 단수적 표현들은 또한, 해당 문장에서 명확하게 달리 표시하지 않는 한, 복수의 의미를 포함하도록 의도될 수 있다. 용어 "및/또는," "그리고/또는"은 그 관련되어 나열되는 항목들의 모든 조합들 및 어느 하나를 포함한다. 용어 "포함한다", "포함하는", "포함하고 있는", "구비하는", "갖는", "가지고 있는" 등은 내포적 의미를 갖는 바, 이에 따라 이러한 용어들은 그 기재된 특징, 정수, 단계, 동작, 요소, 및/또는 컴포넌트를 특정하며, 하나 이상의 다른 특징, 정수, 단계, 동작, 요소, 컴포넌트, 및/또는 이들의 그룹의 존재 혹은 추가를 배제하지 않는다. 본 명세서에서 설명되는 방법의 단계들, 프로세스들, 동작들은, 구체적으로 그 수행 순서가 확정되는 경우가 아니라면, 이들의 수행을 논의된 혹은 예시된 그러한 특정 순서로 반드시 해야 하는 것으로 해석돼서는 안 된다. 추가적인 혹은 대안적인 단계들이 사용될 수 있음을 또한 이해해야 한다.

[0043] 또한, 각각의 구성요소는 각각 하드웨어 프로세서로 구현될 수 있고, 위 구성요소들이 통합되어 하나의 하드웨어 프로세서로 구현될 수 있으며, 또는 위 구성요소들이 서로 조합되어 복수 개의 하드웨어 프로세서로 구현될 수도 있다.

[0044] 이하, 첨부된 도면을 참조하여 본 발명에 따른 바람직한 실시 예를 상세히 설명하기로 한다.

[0046] 도1은 본 발명의 일 실시 예에 의한 보안 공유키 교환 시스템의 구성을 도시한 도면이다. 도1을 참조하면, 본 발명의 일 실시 예에 의한 보안 공유키 교환 시스템은 제어 모듈(100)과 보안 모듈(200)을 포함한다.

[0047] 제어 모듈(100)은 디바이스가 부팅되면 기 저장된 공유키를 초기화하고, 공유키를 재생성하기 위해 제어 모듈(100) 내부에 기 저장된 ECDH 함수를 이용하여 한 쌍의 비대칭키인 제1 공개키와 제1 개인키를 생성하고, 제1 개인키와 보안 모듈(200)의 공개키를 이용하여 제1 비밀키를 생성하고, 제1 비밀키를 이용하여 제1 공유키를 생성하는 모듈로, 생성부(130), 통신부(190) 그리고 데이터부(미도시)를 포함할 수 있다.

[0048] 제어 모듈(100)이 디바이스가 부팅될 때마다 기 저장된 공유키를 초기화하고 다시 생성함으로써 외부에서 데이터를 탈취하여 분석하지 못하게 하는 효과를 가질 수 있다.

[0049] 생성부(130)는 공유키 교환을 위한 제1 공개키, 제1 개인키, 제1 비밀키 및 제1 공유키를 생성할 수 있다. 보다 구체적으로 생성부(130)는 비밀키 생성부(150)와 공유키 생성부(170)를 포함할 수 있다.

[0050] 비밀키 생성부(150)는 제어 모듈(100) 내부에 기 저장된 ECDH 함수를 이용하여 한 쌍의 비대칭키인 제1 공개키와 제1 개인키를 생성할 수 있다.

[0051] ECDH 함수는 타원 곡선 방정식과 이산 대수를 기반으로 만들어진 비대칭형 암호화 방식이다. 비대칭형 암호화 방식은 한 쌍의 키 조합인 공개키와 개인키를 이용하여 데이터를 암호화할 수 있다. 본 발명의 일 실시 예에 따르면, 비밀키 생성부(150)는 비대칭키를 생성함에 있어서 비대칭형 암호화 방식 중 키 교환 방식을 사용할 수 있다.

[0052] 키 교환 방식은 두 모듈이 암호화되지 않은 통신망을 통해 공통의 비밀키를 공유할 수 있도록 하는 방법이다. 예를 들어, A 모듈과 B 모듈이 키 교환을 수행할 경우, A 모듈은 A 모듈이 생성한 공개키를 B 모듈로 전송하고, B 모듈은 B 모듈이 생성한 공개키를 A 모듈로 전송하여 각 모듈 내부에 저장된 수학적 함수를 이용하여 비밀키를 생성할 수 있다.

[0053] 비대칭형 암호화 방식에서 통상적으로 사용되는 RSA와 같은 수학적 함수는 저사양 IoT 디바이스에 적용하기에 연산이 복잡하기 때문에 본원 발명에서는 ECDH 함수를 사용할 수 있다. ECDH 함수는 이산 대수 문제 기반 암호 함수로 짧은 길이의 키를 사용하여 저사양 IoT 디바이스에도 사용할 수 있으며 또한 높은 안전성을 장점으로 한다. ECDH 함수의 키 길이에 따른 파라미터와 각 키 길이에 따른 안전성을 RSA와 비교하면 [표1]과 같다.

표 1

[0055]

Parameters	Section	Strength	Size	RSA/DSA	Koblitz or random
secp112r1	2.2.1	56	112	512	r
secp112r2	2.2.2	56	112	512	r
secp128r1	2.3.1	64	128	704	r
secp128r2	2.3.2	64	128	704	r
secp160k1	2.4.1	80	160	1024	k
secp160r1	2.4.2	80	160	1024	r
secp160r2	2.4.3	80	160	1024	r
secp192k1	2.5.1	96	192	1536	k
secp192r1	2.5.2	96	192	1536	r
secp224k1	2.6.1	112	224	2048	k
secp224r1	2.6.2	112	224	2048	r
secp256k1	2.7.1	128	256	3072	k
secp256r1	2.7.2	128	256	3072	r
secp384r1	2.8.1	192	384	7680	r
secp521r1	2.9.1	256	521	15360	r

[0057] 비밀키 생성부(150)가 제1 공개키와 제1 개인키를 생성하고 나면, 통신부(190)를 통해 수신한 보안 모듈(200)의 공개키와 제1 개인키에 ECDH 함수를 적용하여 제1 비밀키를 생성할 수 있다. 일 실시 예로, ECDH 함수 중 secp224r1 파라미터를 이용하는 함수를 사용하는 경우, 제1 비밀키는 224bits의 길이로 생성될 수 있다. 한편, 제1 비밀키는 외부에 유출되지 않도록 데이터부에 안전하게 보관될 수 있다.

[0058] 공유키 생성부(170)는 비밀키 생성부(150)에서 생성된 제1 비밀키를 이용하여 제1 공유키를 생성할 수 있다. 공유키 생성부(170)는 비밀키 생성부(150)에서 생성된 제1 비밀키를 SHA1 해시 함수를 이용하여 암호화할 수 있다. SHA1 해시 함수는 임의의 길이의 데이터를 고정된 길이의 해시 값으로 출력하는 함수로, 입력 데이터의 오류나 변조를 탐지할 수 있도록 하는 것을 목적으로 한다. 본원 발명에서 사용되는 SHA1 함수는 입력되는 데이터를 20 bytes의 길이의 해시 값으로 출력할 수 있다. 따라서 암호화된 제1 비밀키는 20 bytes의 길이를 가질 수 있다. 본원 발명은 송수신 데이터의 암호화 및 복호화를 AES 128 함수를 통해 진행할 수 있다. AES 128 함수

는 128 bits, 즉 16 bytes의 입력 데이터를 사용할 수 있기 때문에 송수신 데이터의 암호화 및 복호화를 수행할 수 있는 제1 공유키를 암호화된 제1 비밀키의 상위 16 bytes를 추출하여 생성할 수 있다.

[0059] 통신부(190)는 보안 모듈(200)에 보안 모듈(200)의 공개키 및 난수를 요청하고, 보안 모듈(200)에 제1 공개키를 전송하고, 보안 모듈(200)이 생성한 난수를 수신할 수 있다. 제어 모듈(100)이 보안 모듈(200)과 통신하기 위해서는 APDU 프로토콜을 사용할 수 있다. APDU 프로토콜은 Command APDU와 Response APDU로 나눌 수 있다. Command APDU는 요청할 명령을 나타내는 4 byte 헤더부와 필요한 경우 사용하는 데이터 필드 등을 포함하는 옵션부로 구성될 수 있으며, Response APDU는 데이터 필드, 서로를 연결하여 결과 메시지를 나타내는 상태 코드인 SW1과 SW2로 구성될 수 있다.

[0060] 제어 모듈(100)이 보안 모듈(200)에 보안 모듈(200)의 공개키를 요청할 경우, 제어 모듈의 통신부(190)는 [표2]과 같은 구조의 Command APDU를 생성할 수 있다.

표 2

[0062]

Command APDU					
Header				Payload	
CLA	INS	P1	P2	Lc	Data Field
0x80	0xA0	0x0E	0x02	0x00	NULL

[0064] [표2]를 참조하면, CLA(Class of Instruction)은 APDU의 명령 유형을 의미하며 0xFF를 제외한 모든 8 bits의 값을 가질 수 있다. INS(Instruction code)는 명령을 나타내는 코드로 최상위 4 bit가 6 또는 9의 값을 나타내지 않을 때 유효한 값을 가질 수 있다. P1(Instruction Parameter 1)과 P2(Instruction Parameter 2)는 명령의 파라미터를 의미하며 사용하지 않을 경우 0x00의 값을 가질 수 있다. Lc는 데이터 필드의 길이를 나타낸다. 제어 모듈(100)이 보안 모듈(200)에 보안 모듈(200)의 공개키를 요청할 경우에 제어 모듈(100)이 보안 모듈(200)에 송신하는 데이터가 없기 때문에 데이터 필드에는 어떠한 값도 포함하지 않는다.

[0065] 또한, 제어 모듈(100)이 보안 모듈(200)에 제1 공개키를 전송할 경우, 제1 공개키를 전송함과 동시에 보안 모듈(200)의 공유키 및 난수 생성을 요청하기 위해 제어 모듈의 통신부(190)는 [표3]과 같은 구조의 Command APDU를 생성할 수 있다.

표 3

[0067]

Command APDU					
Header				Payload	
CLA	INS	P1	P2	Lc	Data Field
0x80	0xA3	0x0E	0x02	0x39	제1 공개키

[0069] 또한, 통신부(190)는 보안 모듈(200)이 생성한 난수를 수신하여 데이터부(미도시)의 난수 버퍼에 배열의 형태로 저장할 수 있다.

[0071] 보안 모듈(200)은 각종 보안 기능을 하드웨어적으로 구현한 모듈로, 기본적으로 디바이스 인증을 제공하고, 저장된 개인정보 및 데이터를 보호하기 위한 암호화키를 생성하며, 생성된 키는 보안 모듈(200) 내부에 저장되어 소프트웨어 기반 악성 프로그램으로부터 디바이스를 보호하는 기능을 제공한다. 또한, 보안 모듈(200) 내부에는 JavaCard OS와 같은 별도의 운영체제가 포함되어 있어 기존 시스템과 통합하기 쉽고, 시스템의 요구사항에 맞춰 보안 모듈(200) 내부에 개별적인 보안 솔루션을 프로그래밍 할 수 있다는 장점이 있다.

[0072] 보안 모듈(200)은 비대칭키 쌍인 제2 공개키 및 제2 개인키를 생성하고, 제2 개인키와 제어 모듈(100)의 공개키

를 이용하여 제2 비밀키를 생성하고, 제2 비밀키를 이용하여 제2 공유키를 생성하고, 데이터 중복 방지를 위한 난수를 생성하고, 제어 모듈(100)로부터 제어 모듈(100)의 공개키를 수신하고, 제2 공개키 및 난수를 제어 모듈(100)에 전송하고, 난수를 저장하는 모듈로, 생성부(230)와 데이터부(270), 그리고 통신부(290)를 포함할 수 있다.

[0073] 도1의 생성부(230)는 부트 단계에서 각종 검증 및 키 교환과 관련된 역할을 하는 보안 애플릿 영역으로 분류할 수 있으며, 이하에서는 도면에 도시된 구성 요소를 중심으로 보안 모듈(200)의 구성을 설명한다.

[0074] 생성부(230)는 공유키 교환을 위한 제2 공개키, 제2 개인키, 제2 비밀키 및 제2 공유키를 생성한다. 보다 구체적으로 생성부(230)는 비밀키 생성부(240)와 공유키 생성부(250), 그리고 난수 생성부(260)을 포함할 수 있다.

[0075] 비밀키 생성부(240)는 제어 모듈(100)로부터 보안 모듈(200)의 공개키를 송신하라는 요청을 받으면 보안 모듈(200) 내부에 기 저장된 ECDH 함수를 이용하여 한 쌍의 비대칭키인 제2 공개키와 제2 개인키를 생성할 수 있다.

[0076] 비밀키 생성부(240)가 제2 공개키와 제2 개인키를 생성하고 나면, 제2 개인키와 통신부(290)를 통해 수신한 제어 모듈(100)의 공개키인 제1 공개키에 ECDH 함수를 적용으로 제2 비밀키를 생성할 수 있다. ECDH 함수 중 secp224r1 파라미터를 이용하는 함수를 사용함으로써 제2 비밀키는 224bits의 길이로 생성될 수 있다. 제2 비밀키는 외부에 유출되지 않도록 데이터부(270)에 저장될 수 있다.

[0077] 공유키 생성부(250)는 비밀키 생성부(240)에서 생성된 제2 비밀키를 이용하여 제2 공유키를 생성할 수 있다. 공유키 생성부(250)는 비밀키 생성부(240)에서 생성된 제2 비밀키를 SHA1 해시 함수를 이용하여 암호화할 수 있다. 따라서 암호화한 제2 비밀키는 20 bytes의 길이를 가질 수 있다. 본원 발명은 송수신 데이터의 암호화 및 복호화를 16 bytes의 입력 데이터를 사용하는 AES 128 함수를 이용하여 진행할 수 있다. 따라서 송수신 데이터의 암호화 및 복호화를 수행할 수 있는 제2 공유키는 암호화된 제2 비밀키의 상위 16 bytes를 추출하여 생성할 수 있다.

[0078] 난수 생성부(260)는 데이터 중복 방지를 위한 난수를 생성할 수 있다. 난수는 제어 모듈(100)과 보안 모듈(200) 간 송수신되는 데이터를 랜덤화하는 데이터로, 0부터 10까지의 값을 가질 수 있다. 생성된 난수는 데이터부(270)의 난수 버퍼에 배열의 형태로 저장될 수 있다. 제어 모듈(100)과 보안 모듈(200)은 데이터를 송신할 때 Command APDU의 Payload에 난수 버퍼를 추가하여 암호화할 수 있다. 난수 버퍼가 추가된 데이터를 수신한 모듈은 데이터를 복호화한 후 난수를 검증할 수 있다. 예를 들어, 난수 생성부(260)가 3이라는 난수를 생성하면 난수 버퍼의 3번째 인덱스 값을 1로 설정하여 송신할 데이터와 함께 암호화한 후 Command APDU의 Payload에 추가할 수 있다. Command APDU를 수신한 모듈은 Payload를 복호화하고 모듈의 데이터부에 저장된 난수 버퍼와 Payload에 포함된 난수 버퍼가 일치할 경우 정상적인 Command APDU라고 판단할 수 있다. Command APDU의 검증을 완료하면 Payload에 추가된 난수 버퍼의 값을 초기화할 수 있다.

[0079] 데이터부(270)는 보안 모듈(200)이 생성한 제2 공개키, 제2 개인키, 제2 비밀키, 제2 공유키 및 난수 중 하나 이상을 저장할 수 있다.

[0080] 통신부(290)는 제어 모듈(100)로부터 제어 모듈(100)의 공개키인 제1 공개키를 수신하고, 비밀키 생성부(240)가 생성한 제2 공개키 및 난수 생성부(260)가 생성한 난수를 제어 모듈(100)에 전송할 수 있다. 보안 모듈(200)이 제어 모듈(100)과 통신하기 위해서는 APDU 프로토콜을 사용할 수 있다.

[0081] 통신부(290)가 제어 모듈(100)에 제2 공개키를 송신할 경우, 보안 모듈(200)은 [표4]와 같은 구조의 Response APDU를 생성할 수 있다.

표 4

[0083]

Response APDU		
Payload(Optional)	Status	
Data Field	SW1	SW2
제2 공개키	0x90	0x00

[0085] [표4]를 참조하면, 상태 코드인 SW1에 “0x90”의 값을, SW2에 “0x00”을 삽입하여 제어 모듈(100)에 Command

APDU를 정상적으로 수신했음을 알릴 수 있다.

[0086] 또한, 통신부(290)가 제어 모듈(100)에 난수를 송신할 경우, 보안 모듈(200)은 [표5]와 같은 구조의 Response APDU를 생성할 수 있다.

표 5

[0088]

Response APDU		
Payload(Optional)	Status	
Data Field	SW1	SW2
난수	0x90	0x00

[0090] 이하에서는 도2을 참조하여 본 발명의 일 실시 예에 의한 보안 공유키 교환 방법을 설명한다. 보안 공유키 교환 방법에 관한 설명에서 전술한 보안 공유키 교환 시스템과 중복되는 세부 실시 예는 생략될 수 있다.

[0091] 도2를 참조하면 제어 모듈(100)은 디바이스가 부팅되면 제어 모듈(100)에 기 저장된 ECDH 함수를 이용하여 한 쌍의 비대칭키인 제1 공개키 및 제1 개인키를 생성할 수 있다(S100).

[0092] 제어 모듈(100)이 제1 공개키 및 제1 개인키를 생성하면, 제어 모듈(100)은 보안 모듈(200)에 보안 모듈(200)이 생성할 수 있는 보안 모듈(200)의 공개키를 요청할 수 있다(S190).

[0093] 보안 모듈(200)이 제어 모듈(100)로부터 보안 모듈(200)의 공개키를 요청 받으면, 보안 모듈(200)은 보안 모듈(200)에 기 저장된 ECDH 함수를 이용하여 한 쌍의 비대칭키인 제2 공개키 및 제2 개인키를 생성할 수 있다(S200).

[0094] 보안 모듈(200)이 제2 공개키 및 제2 개인키를 생성하면, 보안 모듈(200)은 제어 모듈(100)에 보안 모듈(200)의 공개키인 제2 공개키를 전송할 수 있다(S290).

[0095] 제어 모듈(100)이 보안 모듈(200)로부터 제2 공개키를 수신하면, 제어 모듈(100)은 제1 개인키와 제2 공개키에 ECDH 함수를 적용하여 제1 비밀키를 생성할 수 있다(S300). 보다 구체적으로 제1 비밀키를 생성하는 ECDH 함수는 ECDH 함수 중 secp224r1 파라미터를 이용하는 함수를 이용하기 때문에 224 bits의 길이를 가질 수 있다.

[0096] 제어 모듈(100)이 제1 비밀키를 생성하면, 제1 공유키를 생성할 수 있다(S400). 보다 구체적으로 제1 비밀키에 SHA1 함수를 사용하여 20 bytes의 해시 값을 얻은 후 상위 16 bytes를 추출하여 제1 공유키를 생성할 수 있다.

[0097] 제어 모듈(100)이 제1 공유키를 생성하면, 보안 모듈(200)에 제1 공개키를 전송하고 제2 공유키 생성 및 난수를 요청할 수 있다(S490).

[0098] 보안 모듈(200)이 제어 모듈(100)로부터 제2 공유키 생성 및 난수를 요청 받으면, 보안 모듈(200)은 제2 개인키와 제1 공개키에 ECDH 함수를 적용하여 제2 비밀키를 생성할 수 있다(S500). 보다 구체적으로 제2 비밀키를 생성하는 ECDH 함수는 ECDH 함수 중 secp224r1 파라미터를 이용하는 함수를 이용하기 때문에 224 bits의 길이를 가질 수 있다.

[0099] 보안 모듈(200)이 제2 비밀키를 생성하고 나면, 제2 공유키를 생성할 수 있다(S600). 보다 구체적으로 제2 비밀키에 SHA1 함수를 사용하여 20 bytes의 해시 값을 얻은 후 상위 16 bytes를 추출하여 제2 공유키를 생성할 수 있다.

[0100] 보안 모듈(200)이 제2 공유키를 생성하면, 보안 모듈(200)은 난수를 생성할 수 있다(S700). 난수는 0부터 10까지의 값을 가지며 송수신 데이터를 랜덤화할 수 있다. 데이터를 랜덤화함으로써 데이터 중복을 방지할 수 있다는 효과가 있다.

[0101] 보안 모듈(200)이 난수를 생성하면, 보안 모듈(200)은 제어 모듈(100)에 난수를 전송할 수 있다(S790).

[0102] 제어 모듈(100)이 보안 모듈(200)로부터 난수를 수신하면, 제어 모듈(100)은 난수를 저장할 수 있다(S800).

[0103] 도2에 도시되지 않았으나 제어 모듈(100)과 보안 모듈(200)은 저장된 난수를 송신하고자 하는 데이터와 함께 암호화할 수 있다. 보다 구체적으로 제어 모듈(100)과 보안 모듈(200)은 난수를 포함하는 데이터와 AES 128 함수

를 이용하여 암호화할 수 있다. AES 함수는 저사양 IoT 디바이스에서 사용할 수 있으며 또한 보안성이 입증된 암호화 및 복호화 과정에서 동일한 키를 사용하는 대칭키 알고리즘으로, 본원 발명의 제어 모듈(100)은 데이터를 암호화하는 데에 제1 공유키를, 보안 모듈(200)은 제2 공유키를 사용할 수 있다. 본원 발명의 제어 모듈(100)의 제1 공유키와 보안 모듈(200)의 제2 공유키는 같은 값을 가질 수 있다.

[0104] 이에 따라 외부에서 제어 모듈(100)과 보안 모듈(200) 간 송수신되는 데이터를 탈취하여 외부에서 목적하는 데이터를 추가하여 재전송하는 것을 난수를 이용하여 방지할 수 있다. 보다 구체적으로 외부에서 탈취한 데이터의 난수가 3일 경우에, 재전송되는 데이터의 난수는 3의 값을 가질 수 있다. 그러나 보안 모듈(200)은 매 부트 단계마다 난수를 새로 생성하기 때문에, 재전송되는 데이터와 기 저장된 난수 값이 다를 수 있다. 따라서 수신된 데이터와 난수가 일치하지 않으면 악의적인 목적을 갖는 데이터라고 판단할 수 있다.

[0106] 이하에서는 도3을 참조하여 본원 발명에서 생성한 공유키를 이용하여 송수신 데이터를 검증하는 방법을 설명한다.

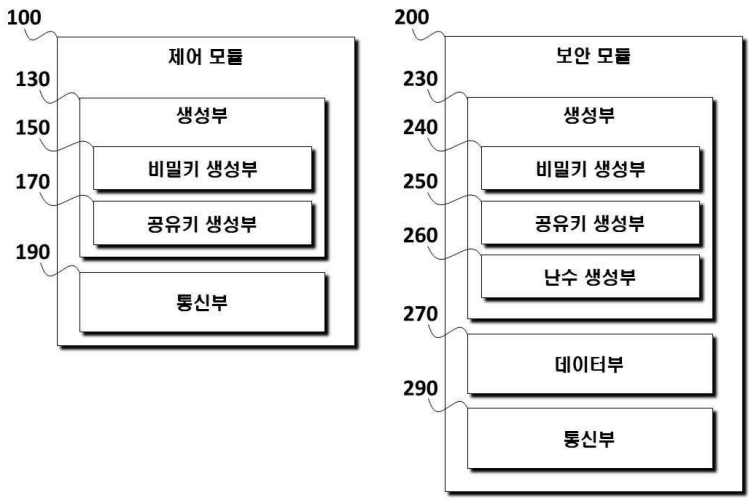
[0107] 보안 모듈(200)은 수신한 데이터를 AES 128 함수를 이용하여 복호화하여 데이터에 포함되어 있는 난수를 검증할 수 있다(S910). 보다 구체적으로 난수 버퍼의 인덱스를 카운트하였을 때 카운트 된 값이 짝수이면 수신된 데이터가 정상적이라고 판단하여 검증 성공 메시지를 생성하여 제어 모듈(100)에 전송할 수 있다(S930). 만약 난수 버퍼의 카운트 값이 홀수이면 검증 실패 메시지를 생성하여 제어 모듈(100)에 전송할 수 있다(S940).

[0108] 보안 모듈(200)이 난수의 검증을 완료하면, 수신된 데이터에서 난수를 제외한 데이터를 검증할 수 있다(S920). 보다 구체적으로 각 모듈의 데이터부에 기 저장된 검증 기준과 데이터를 비교 및 분석하여 데이터가 정상적이라고 판단하여 검증 성공 메시지를 생성하고, 그렇지 않으면 검증 실패 메시지를 생성하여 제어 모듈(100)에 전송할 수 있다. 기 저장된 검증 기준은 전자 서명일 수도 있고 또는 그 외의 것일 수 있다.

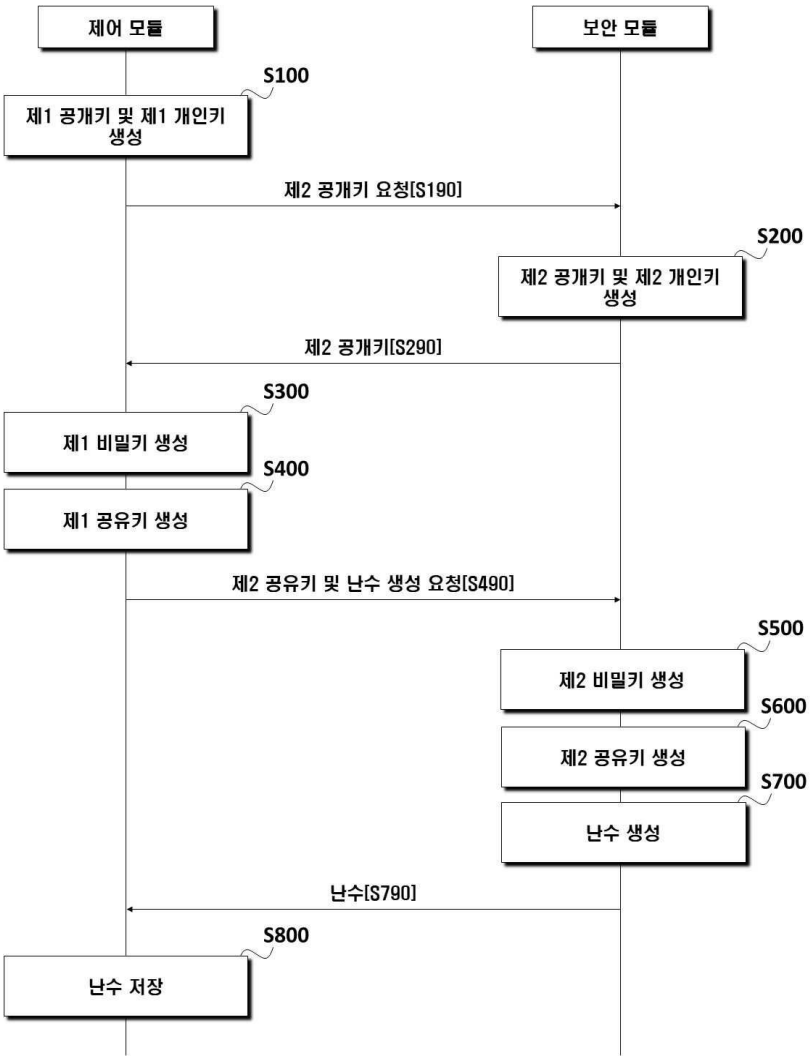
[0110] 본 명세서와 도면에 개시된 본 발명의 실시 예들은 본 발명의 기술 내용을 쉽게 설명하고 본 발명의 이해를 돕기 위해 특정 예를 제시한 것뿐이며, 본 발명의 범위를 한정하고자 하는 것은 아니다. 여기에 개시된 실시 예들 이외에도 본 발명의 기술적 사상에 바탕을 둔 다른 변형 예들이 실시 가능하다는 것은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에게 자명한 것이다.

도면

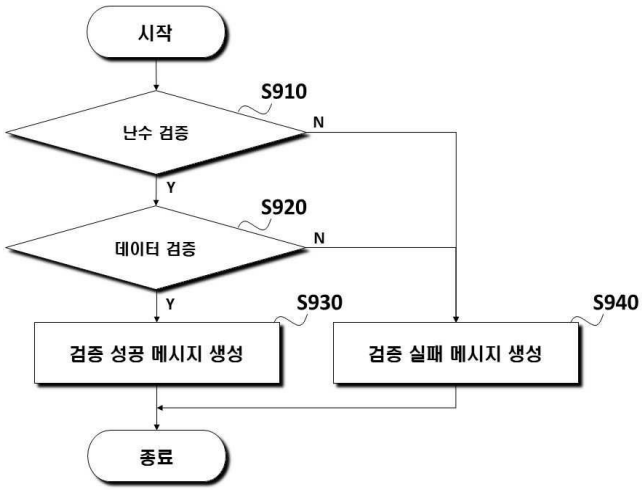
도면1



도면2


제어 모듈
보안 모듈
S100
제1 공개키 및 제1 개인키 생성
제2 공개키 요청[S190]
S200
제2 공개키 및 제2 개인키 생성
제2 공개키[S290]
S300
제1 비밀키 생성
S400
제1 공유키 생성
제2 공유키 및 난수 생성 요청[S490]
S500
제2 비밀키 생성
S600
제2 공유키 생성
S700
난수 생성
난수[S790]
S800
난수 저장

도면3


시작
S910
난수 검증
N
Y
S920
데이터 검증
N
Y
S930
검증 성공 메시지 생성
S940
검증 실패 메시지 생성
종료