JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 27, No. 3, pp. 839-848, Mar. 2026

Check for updates

IoT 환경에서 센서 노드와 클러스터 헤드, 게이트웨이의 3자 간 상호 인증

이 구 연[1] · 이 용[2*]
[1]강원대학교 컴퓨터공학과 교수
[2]덕성여자대학교 디지털소프트웨어공학부 초빙교수

Three-Party Mutual Authentication Between Sensor Nodes, Cluster Heads, and Gateways in IoT Environments

Goo Yeon Lee[1] · Yong Lee[2*]

[1]Professor, Department of Computer Engineering, Kangwon National University, Chuncheon 24341, Korea
[2]Invited Professor, College of Science and Technology, Duksung Women's University, Seoul 01369, Korea

[요 약]

오늘날 무선 센서 네트워크는 다양한 분야에서 중요한 역할을 수행하고 있다. 그러나 무선 센서 네트워크에 대한 외부 공격, 데이터 위조, 노드의 침해 등 다양한 보안 위협이 존재한다. 이러한 위협을 해결하기 위한 방법 중 하나가 상호 인증이다. 상호 인증은 네트워크 내의 노드가 서로의 신원을 검증하고, 신뢰할 수 있는 통신이 이루어지도록 보장하는 중요한 보안 메커니즘이다. 특히 센서 네트워크에서는 센서 노드와 클러스터 헤드 그리고 게이트웨이 간의 상호 인증 절차가 중요한 역할을 한다. 이에 본 연구에서는 센서 노드와 클러스터 헤드 그리고 게이트웨이 간의 3자 간 상호 인증 프로토콜 절차를 제안하고, 보안 분석을 수행한다. 기존의 인증 방법들은 주로 두 당사자 간의 인증에 집중한 반면, 본 연구에서의 3자 간 상호 인증은 3개의 개체 간의 상호 인증을 수행함으로 보안을 더욱 강화할 수 있어 무선 센서 네트워크에 효과적으로 활용될 수 있다.

[Abstract]

Wireless sensor networks play a vital role in diverse fields. However, they face various security threats, including external attacks, data forgery, and node compromise. One way to address these threats is through mutual authentication. Mutual authentication is a security mechanism that allows nodes within networks to verify each other's identities and ensures trustworthy communication. The mutual authentication process among sensor nodes, cluster heads, and gateways is especially critical in sensor networks. This study proposes three-party mutual authentication protocols involving sensor nodes, cluster heads, and gateways, and conducts a security analysis. While existing authentication methods mainly focus on two-party authentication, the proposed three-party mutual authentication strengthens security by enabling mutual authentication among the three entities, making it more applicable to wireless sensor networks.

색인어 : IoT, 센서 노드, 클러스터 헤드, 게이트웨이, 상호 인증
Keyword : IoT, Sensor Node, Cluster Head, Gateway, Mutual Authentication

http://dx.doi.org/10.9728/dcs.2026.27.3.839

Received 10 January 2026; **Revised** 04 February 2026
Accepted 05 March 2026

***Corresponding Author; Yong Lee**

Tel: +82-2-901-8846
E-mail: yonglee@duksung.ac.kr

Ⅰ. 서 론

무선 센서 네트워크(Wireless Sensor Network, WSN)는 IoT(Internet of Things) 기술과 함께 다양한 분야에서 지속적으로 중요한 역할을 하고 있으며, 최근에는 특히 환경 모니터링, 군사 감시, 의료 모니터링, 스마트 홈 등 여러 응용 분야에서 널리 전개되고 있다[1],[2]. 센서 네트워크는 물리적 환경을 감지하고 그 정보를 실시간으로 처리하여 필요한 시스템에 전달하는 기능을 수행한다. 그림 1은 무선 센서 네트워크 구성도의 한 예를 보여 준다.

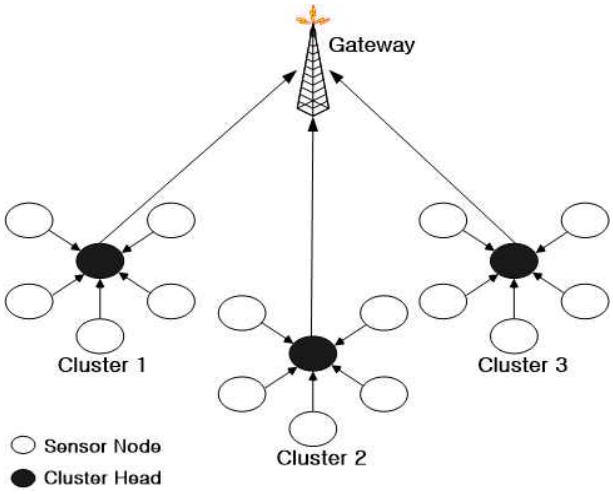

그림 1. 무선 센서 네트워크 구성도의 예
Fig. 1. An example of wireless sensor network configuration diagram

각 센서 노드는 주변 환경에 대한 데이터를 수집하여 무선으로 클러스터 헤드에 전송하고, 최종적으로 중요한 정보를 게이트웨이를 통하여 중앙 서버나 클라우드로 전송한다. 이러한 시스템은 다양한 장점에도 불구하고 여러 보안 위협에 취약한 특성을 지닌다. 무선 통신이라는 특성상 외부 공격자가 센서 네트워크에 침입하여 데이터를 탈취하거나 위조할 가능성이 높다. 또한, 센서 노드는 자원 제한이 크기 때문에 보안이 강화된 프로토콜을 구현하기 어려운 점이 있다. 따라서 센서 네트워크에서의 보안 문제는 매우 중요한 연구 주제로, 특히 인증과 데이터 무결성의 확보가 중요한 문제가 되고 있다[1],[2].

무선 센서 네트워크에서 상호 인증은 두 노드가 서로 신뢰할 수 있는 통신을 할 수 있도록 보장하는 핵심적인 보안 메커니즘이다. 상호 인증은 두 당사자가 서로의 신원을 확인하고, 서로가 신뢰할 수 있는 존재임을 검증하는 과정으로, 네트워크에서 발생할 수 있는 악의적인 공격, 예를 들어, 중간자 공격(Man-in-the-Middle Attack), 재전송 공격(Replay Attack), 부인 방지(Non-repudiation) 등을 방지할 수 있다. 무선 센서 네트워크의 보안에서 상호 인증은 기본적으로 센서 노드와 클러스터 헤드 간의 신뢰를 확보하는 과정에서 시작되며, 센서 노드와 게이트웨이 간의 인증이 결합되어 3자 상호 인증을 형성할 수 있다. 기존의 인증 방식들은 주로 두 노드 간의 인증을 중심으로 구현되어 왔으나, 무선 센서 네트워크에서는 클러스터 헤드와 게이트웨이가 중요한 역할을 하기 때문에, 센서 노드와 클러스터 헤드 간의 신뢰 관계만으로는 충분하지 않아, 지속적으로 공격대상이 되고 중요한 정보가 노출되는 사고가 발생하고 있다. 따라서 센서 노드, 클러스터 헤드, 그리고 게이트웨이 간의 3자 상호 인증을 통해 더 높은 보안성을 구현할 필요가 있다[2],[3].

클러스터 헤드는 센서 네트워크에서 중요한 역할을 하는 장치로, 네트워크 내의 여러 센서 노드를 관리하고 데이터를 수집하여 게이트웨이로 전송하는 기능을 담당한다. 즉 클러스터 헤드는 여러 센서 노드들과의 무선 통신을 통해 데이터를 수집하고, 이를 게이트웨이로 전송하는 중간 단계의 역할을 한다. 이 과정에서 클러스터 헤드가 악성 공격자에 의해 침해되거나, 센서 노드와 클러스터 헤드 간의 통신이 위조되거나 중단되는 등의 보안 위협이 발생할 수 있다.

3자 상호 인증 절차는 센서 네트워크의 보안을 더욱 강화할 수 있는 방법이다. 기존의 인증 방식에서는 센서 노드와 클러스터 헤드 간의 인증만을 고려하는 경우가 많았으나, 3자 상호 인증은 인증 과정에 게이트웨이를 포함시켜, 센서 노드, 클러스터 헤드, 게이트웨이 간의 신뢰 관계를 동시에 구축한다. 이 방식은 중간자 공격이나 클러스터 헤드의 침해와 같은 보안 위협을 방지하는 데 효과적이다. 예를 들어, 게이트웨이는 클러스터 헤드가 실제로 신뢰할 수 있는 장치인지 확인하며, 센서 노드는 클러스터 헤드와 통신하기 전에 클러스터 헤드의 신원을 검증한다. 또한, 게이트웨이는 센서 노드의 신원도 검증하여 불법적인 노드가 네트워크에 접근하는 것을 방지한다. 이러한 다단계 인증 절차는 센서 네트워크의 보안성을 크게 향상시키며, 악의적인 공격자가 네트워크에 침투하는 것을 어렵게 만든다.

본 연구에서는 3자 상호 인증 방법으로 두 가지 프로토콜을 제안한다. 첫 번째는 패스워드 방식의 3자 상호 인증 절차인데, 이 프로토콜에서는 클러스터 헤드와 게이트웨이 간 공개키 방식을 통한 상호 패스워드로 검증하고, 클러스터 헤드에서 센서 노드의 패스워드에 대한 검증은 게이트웨이와 센서 노드 사이에서 암호화된 형태로 진행됨으로써 센서 노드의 패스워드에 대한 노출 없이도 클러스터 헤드는 센서 노드의 패스워드를 검증할 수 있게 된다. 두 번째 프로토콜은 키 교환과 CHAP (Challenge Handshake Authentication Protocol)을 이용한 상호 인증 프로토콜로서, 먼저 상호 인증을 원하는 클러스터 헤드와 게이트웨이 사이에서 키 교환 프로토콜을 통해 비밀키를 공유한 후, 이 후 CHAP을 사용하여 3자간 상호 인증을 완료한다. 이렇게 제안된 방식들은 센서 노드와 클러스터 헤드 그리고 게이트웨이간의 간결한 3자 인증을 제공함으로서 IoT 무선 센서 네트워크 환경에서 효과적으로 활용될 것으로 판단된다.

II. 관련 연구

무선 센서 네트워크 및 IoT 환경에서 인증과 키 교환은 보안성 확보의 핵심 문제로 꾸준히 연구되어 왔다. 특히 센서 노드, 클러스터 헤드, 게이트웨이 등의 여러 주체 간 상호 인증(Mutual Authentication)과 세션 키 협상(Key Agreement)은 네트워크 신뢰성과 데이터 무결성을 확보하기 위한 중요한 기술로 인식되고 있다. 이런 문제를 해결하기 위하여 최근 인증 연구는 경량화, 형식적 검증(Formal Verification), 그리고 다중 요소(Multi-Factor) 기반 프로토콜로 진화하고 있다.

3자 상호 인증(Three-Party Authentication)은 두 통신 주체가 신뢰할 수 있는 제 3자의 도움을 받아 상호 인증과 세션 키 합의를 수행하는 구조로, 기존의 2자 인증 방식의 한계를 보완하기 위해 제안되었다. 이 개념은 특히 두 통신 주체가 직접적으로 비밀 정보를 공유하지 않으면서도 안전한 통신을 수행해야 하는 환경에서 효과적인 보안 메커니즘으로 활용된다. 3자 인증 구조는 일반적으로 두 사용자 또는 노드와 중앙 서버(또는 게이트웨이)로 구성되며, 제 3자는 인증 정보의 검증과 세션 키 생성 과정에서 핵심적인 역할을 수행한다. 3자 인증에 대한 대표적인 기초 연구로는 [2]에서 제안하는 Three-Party Password-Based Authenticated Key Exchange 프로토콜이 있다. 이 연구는 두 사용자가 서로 비밀번호를 공유하지 않고, 각각 신뢰된 서버와 공유한 비밀번호만을 이용하여 안전하게 상호 인증과 세션 키를 생성할 수 있음을 이론적으로 정립하였다. 해당 연구는 3자 인증 환경에서의 보안 모델을 체계적으로 제시하였으며, 비밀번호 추측 공격, 중간자 공격, 세션 키 노출 공격에 대한 안전성을 분석함으로써 이후 3자 인증 및 키 합의(3PAKE; Password-Based Authenticated Key Exchange) 연구의 기반을 마련하였다. 이 구조는 이후 다양한 응용 환경에서 확장·응용되며, 3자 인증 프로토콜의 기본 설계 원칙으로 널리 인용되고 있다.

[4]에서는 3자 인증(3PAKE)의 현대적 적용과 보안성을 강화하고 비밀번호 기반 인증과 키 교환을 다루고 있다. 여기서는 MLWE(Modular Learning With Errors) 기반 암호학을 활용해 양자 컴퓨터 공격에도 안전한 3자 인증 및 키 교환 프로토콜을 설계하고 3자 구조에서 사용자·서버·게이트웨이(또는 인증 서버) 간 상호 인증과 세션 키 생성을 구현하고자 하였다.

최근에는 이러한 전통적인 3자 인증 개념을 현대 보안 요구사항에 맞게 확장한 연구들이 등장하고 있다. 3자 비밀번호 기반 인증과 키 교환 구조를 양자 컴퓨팅 환경에서도 안전하도록 개선한 연구에서는 기존 3자 인증 모델을 유지하면서도 새로운 암호학적 가정을 도입하여 보안성을 강화하였다. 이러한 연구들은 기본적인 3자 인증 구조가 다양한 암호 기법과 결합될 수 있음을 보여주며, IoT 및 무선 센서 네트워크와 같은 자원 제한 환경에서도 적용 가능한 설계 방향을 제시한다.

Thakur 등은 ECC(타원 곡선 암호)를 기반으로 3요소 인증(Three-Factor Authentication) 시스템을 제안하며, 기존 스킴의 보안 취약점(사용자·게이트웨이·노드 위조 공격 등)을 분석하고 개선된 IoT-WSN 환경의 인증 절차를 제시하였여 상호 인증 성능을 강화하고자 하였다[5].

Huang은 ECC 기반으로 스마트 카드, 패스워드, 생체인식 요소를 결합하여 3요소 상호 인증 및 키 합의 기법을 제시하였으며, 챌린지/응답 기반 인증 절차를 수행함으로써 세션 키를 안전하게 협상하는 메커니즘을 제공하였다[6]. 이 연구는 BAN 논리((Burrows, Abadi and Needham logic) 및 ProVerif(Protocol Verifier)와 같은 검증 도구를 활용하여 프로토콜의 보안성을 분석하였고, 익명성 보장 및 추적 공격 방지 측면에서도 기존 연구 대비 향상된 결과를 보였다[6]. 이러한 연구들은 주로 사용자–게이트웨이–센서 노드 구조를 가정하고 있으나, 클러스터 헤드의 보안 위협을 충분히 고려하지 못한 한계를 지닌다. 또한, Wang 등의 연구에서는 PUF (Physical Unclonable Function) 및 혼돈(Chaotic Map) 기반의 3자 인증 및 키 합의 구조가 제안되어, 익명성 보호와 머신러닝 기반 공격 저항성을 향상시키는 설계가 소개되고 있다[7]. 이와 같은 다중 인증 및 키 합의는 전통적인 ECC 기반 설계와 병행하여 WSN의 보안성과 효율성을 더욱 강화하기 위한 대안으로 평가받고 있다. [8]에서는 Chen 등이 제안한 기존 인증 기법의 취약점을 분석하고, 이를 보완하는 새로운 사용자 인증 및 키 합의 스킴을 제안하였다. 이 논문에서 제안된 방식은 암호 분석을 통해 다양한 공격에 대한 안전성을 입증하였으며, 성능 분석과 NS-3 시뮬레이션 결과를 통해 계산 비용과 통신 효율 측면에서도 기존 기법들보다 우수함을 보였다.

WSN에서 게이트웨이 및 다중 인증 구조를 고려한 다중 게이트웨이 인증 및 키 합의 연구도 진행되어 사용자, 센서, 게이트웨이가 동적으로 여러 영역에 참여하는 시나리오를 처리할 수 있도록 설계되었다[9]. 이러한 접근은 단일 경로 인증 구조의 한계를 극복하고, 네트워크 확장성 측면에서도 유리한 결과를 도출하였다[9].

전통적인 상호 인증과 세션 키 협상 프로토콜과 관련하여 키 분배 및 인증 프로토콜을 포괄적으로 조사한 연구로 [2]에서는 낮은 연산 능력과 제한된 신뢰 기반을 갖는 WSN의 특성과 관련하여 다양한 인증/세션 키 협상 메커니즘의 장단점을 체계적으로 분석하여 향후 연구 방향성을 제시하였다. 또한, IoT 보안 전체를 다루는 인증 및 키 합의 메커니즘에 대한 종합적 리뷰 연구에서도 기존 기법의 한계와 새로운 방향을 논의함으로써 WSN 보안 연구가 보다 확장되고 있음을 보여준다[10].

의료, 스마트 홈, 스마트 공장 등 다양한 응용 분야에서는 익명성(Anonymity), 전방 안전성(Forward Secrecy), 스마트 카드 분실 공격 방어와 같은 세분화된 보안 요건을 만족시키기 위한 프로토콜이 제안되고 있다. Lee 등은 무선 의료 센서 네트워크(WMSN; Wireless Medical Sensor Network)를 대상으로 한 인증 프로토콜을 제안하고 AVISPA 및 ROR 모델을 이용한 형식적 분석을 통해 여러 공격에 대한 저항성

을 검증하고, 계산 및 통신 비용 측면에서도 효율적임을 주장하였다[11].

그러나 이러한 연구들은 주로 사용자–센서 또는 센서–게이트웨이 간 인증 구조에 초점을 맞추는 경우가 많다. 즉 IoT 환경에서 일반 사용자는 센서로부터 데이터를 수집하기 위해 게이트웨이를 거쳐야 하는데, 이때 일반 사용자, 게이트웨이, 센서 간의 3자 간 상호 인증이 필요하기 때문이다. 이는 의료 IoT 망 등에서 개인정보보호를 위한 조치로 법률이나 규정 등에서 요구하는 사항으로 구현 영역에서 많은 연구가 진행되고 있는 분야이다. 반면 본 연구에서 다루는 게이트웨이와 센서에 추가로 클러스터 헤드를 포함한 무선 센서 네트워크 내의 3자 간 상호 인증에 관한 연구는 상대적으로 부족하다. 특히 클러스터 헤드는 다수의 센서 노드를 집계하고 데이터를 전송하는 핵심 구성 요소로, 그 자체가 공격 취약점으로 작용할 수 있기 때문에 이를 포함하는 보안 구조의 중요성은 계속 강조되고 있다. 본 연구는 이러한 한계를 보완하여 클러스터 헤드, 게이트웨이, 센서 노드를 포함한 3자 상호 인증 구조를 설계하고, 이를 패스워드 기반과 키 교환 및 CHAP 기반 인증 방식으로 구현함으로써 기존 연구 대비 실용성과 보안성을 동시에 향상시키는 것을 목표로 한다.

Ⅲ. 3자간 상호 인증 프로토콜

3-1 3자간 상호 인증의 필요성

무선 센서 네트워크에서 장비 셋업시에 게이트웨이와 센서 노드, 그리고 게이트웨이와 클러스터 헤드 간의 인증 정보는 사전에 공유 관리된다. 다만 이동이 빈번하고 접속 클러스터가 수시로 바뀔 수 있는 클러스터 헤드와 센서 노드 간의 인증 정보 공유는 클러스터 헤드의 많은 자원을 요구하고 또한 센서 노드 인증 정보의 추가 및 삭제 등의 복잡한 절차가 필요하므로 쉽지 않다. 즉 클러스터 헤드가 모든 센서 노드와의 인증 정보를 전부 공유 관리하는 것은 현실적으로 어려우며, 또한 클러스터 헤드들 끼리 부분적으로 센서 노드 들의 인증 정보를 나누어 저장하는 것도 이동이 빈번한 센서 노드의 특성상 클러스터 헤드 들끼리 센서 노드들의 인증 정보 교환 및 검증 등 복잡한 절차가 필요하기 때문이다.

이러한 상황에서 센서 노드와 클러스터 헤드 그리고 게이트웨이 간의 일반적인 인증은 양자 간의 인증 방식을 통하여 구현될 수 있다. 이 경우 그림 2와 같이 게이트웨이와 센서 노드, 그리고 게이트웨이와 클러스터 헤드 간의 양자 간 인증을 이용할 수 있다.

그림 2에서 게이트웨이와 클러스터 헤드는 공유한 인증 정보를 이용하여 상호 인증을 수행하며, 센서 노드도 게이트웨이와 공유된 인증 정보를 클러스터 헤드를 거쳐 교환함으로써 상호 인증을 수행한다. 그러나 이 경우 클러스터 헤드와 센서 노드는 공유된 인증 정보가 없으므로 상호 인증을 수행할 수 없다. 또한 센서 노드와 게이트웨이 간의 정보는 클러스터 헤드를 통하여 교환되므로, 클러스터 헤드가 이를 알 수 없도록 하기 위해 암호화 하여 전달하여야 한다. 즉 클러스터 헤드는 전달하는 센서 노드 데이터의 내용을 알 수 없으며, 단지 데이터의 전달 역할에 국한되는 구조이다. 이는 클러스터 헤드가 제한된 역할만 수행하는 구조로 현재의 무선 센서 네트워크의 구성상 비효율적인 운영 방법이 된다.

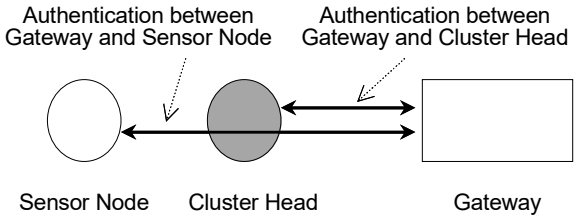

그림 2. 게이트웨이와 클러스터 헤드, 게이트웨이와 센서 노드 간의 양자 간의 인증 방식의 예

Fig. 2. An example of two-party authentications between gateway and cluster head, and between gateway and sensor node

클러스터 헤드는 데이터의 단순한 수집 및 전달을 넘어서 보다 고도화된 역할의 수행을 요구받고 있다. 즉 클러스터 헤드는 센서 노드로부터 수집된 데이터에 대한 간단한 선처리로부터 부분적으로 엣지 컴퓨팅의 기능을 수행할 수 있어야 한다. 예를 들어 클러스터 헤드는 센서 노드가 보내 온 데이터에 대하여 이전 데이터와 중복된 부분 처리, 여러 센서 노드로부터의 데이터를 합쳐 보내거나 또는 클러스터 헤드와 게이트웨이 간의 채널 용량을 고려한 압축 전송 등을 수행할 수 있어야 한다. 또한 이미지 데이터의 경우 간단한 화상 인식 기능을 선제적으로 수행할 수도 있어야 하며, 센서 노드들의 여러 데이터에 대한 간단한 데이터 분석 등을 수행할 수도 있어야 한다.

이러한 고도화된 기능의 수행을 위해서는 클러스터 헤드는 센서 노드로부터 수집된 데이터에 대한 액세스가 가능해야 한다. 통상적으로 센서 노드는 데이터를 암호화하여 보내야 하는데, 클러스터 헤드가 암호화된 센서 노드의 데이터를 풀어보기 위해서는 센서 노드와의 상호 인증이 필요하며, 이에 기반한 세션 키 생성이 필요하다. 이와 같은 필요에 의하여, 기본적인 게이트웨이와 클러스터 헤드 간의 인증, 게이트웨이와 센서 노드 간의 인증에 추가적으로 클러스터 헤드와 센서 노드간의 상호 인증 과정이 필요하다. 이러한 클러스터 헤드와 센서 노드간의 상호 인증 과정은 게이트웨이와 클러스터 헤드 간의 인증, 게이트웨이와 센서 노드 간의 인증과 결합하여 3자간 상호 인증의 형태로 설계하는 것이 효과적이다. 이에 이어지는 절에서는 본 연구에서 제안한 “패스워드 기반의 3자간 상호 인증 프로토콜”과 “키 교환과 CHAP 기반의 3자간 상호 인증 프로토콜”에 대하여 기술한다.

3-2 패스워드 기반의 3자간 상호 인증 프로토콜

본 프로토콜의 기본적인 가정은 다음과 같다. 클러스터 헤드와 게이트웨이 간은 사전에 상대에 대한 인증으로 각각 서로에 대한 패스워드를 공유한다. 즉 클러스터 헤드와 게이트웨이 간의 상호 인증 시 각각 상대에 대한 자기의 패스워드를 제시하여야 한다. 또한 클러스터 헤드와 게이트웨이는 상대적으로 컴퓨팅 자원이 풍부하므로 공개키 연산을 수행할 수 있다고 가정하며, 각각 공개키를 게시하며, 해당되는 개인키를 갖고 있다고 가정한다.

그리고 간편하게 설치 운영되어야 하는 센서 노드는 등록 시점에 게이트웨이에 대한 패스워드만 공유한다. 이는 클러스터 헤드와는 다르게, 설치 장소의 이동이 빈번하고 접속 클러스터 헤드가 수시로 바뀔 수 있는 센서 노드의 경우 특정 클러스터 헤드와의 고정된 인증보다는 게이트웨이와의 패스워드를 통하여 임의의 클러스터 헤드와 3자 간 상호 인증을 수행하는 방법이 현실적이기 때문이다.

따라서 센서 노드가 현장에 처음 설치되어 클러스터 헤드와 통신할 때, 센서 노드와 클러스터 헤드, 그리고 게이트웨이 간의 패스워드 방식의 3자 간 상호 인증 절차가 진행된다. 클러스터 헤드와 게이트웨이 간은 상대방의 공개키를 이용하여 패스워드를 확인한다. 또한 클러스터 헤드에서 센서 노드의 패스워드에 대한 검증을 게이트웨이와 센서 노드 사이에서 암호화된 형태로 진행한다. 이에 클러스터 헤드에서 센서 노드의 패스워드에 대한 노출 없이도 센서 노드의 패스워드를 검증할 수 있게 된다.

다음은 본 절에서의 패스워드 기반의 3자간 상호 인증 프로토콜의 절차에 대한 설명이다. 먼저 절차에서 사용되는 약어는 다음과 같다.

SN : Sensor node
CH : Cluster head
GW : Gateway

SN_ID : Identification of sensor node
CH_ID : Identification of cluster head
GW_ID : Identification of gateway

SN_TID : Temporary ID of SN
GW_AID : Alias of GW

SN_Pass : Password of SN
CH_Pass : Password of CH
GW_Pass : Password of GW

n1, n2, n3, n4, n5 : Nonces

KX+ : Public Key of entity X
KX- : Private Key of entity X
KX/Y : Shared secret key of entity X and Y
{M}K : Message M encrypted with key K

본 절에서 제안하는 패스워드 기반의 3자 간 상호 인증 절차는 GW/CH 사이의 상호 인증 과정과 SN/CH 간의 상호 인증 과정의 두 단계로 이루어진다.

1) GW/CH 사이의 상호인증

(M1) SN→CH:

$$\{SN\_ID,\ CH\_ID,\ n1\}_{K_{SN/GW}},SN\_TID,\ GW\_AID \quad (1)$$

메시지 1(M1)에서 $K_{SN/GW}$는 SN과 GW 간의 공유 비밀 키(SN이 GW에 등록될 때 생성된 키)로 GW가 SN를 식별 인증하는 데 사용된다. SN_TID는 SN의 임시 식별자로, SN이 CH에 처음 인식 등록될 때 CH에 의해 사전에 부여된 임시 ID이며 SN_ID가 무선상에 자주 노출되는 것을 막기 위해 사용된다. 또한 GW_AID는 GW의 별칭이며, 클러스터 헤드와 게이트웨이는 서로 간에 별칭 테이블을 관리하므로, CH는 GW_AID를 실제 GW와 일치시킬 수 있다. 본 메시지에서는 SN은 CH에게 자기의 임시 ID인 SN_TID와 자신이 등록되어진, 그래서 인증을 수행할 GW의 별칭인 GW_AID를 전달한다. 또한 자신과 GW 사이의 공유 비밀키로 암호화한 자신의 ID (SN_ID), 접속하고자 하는 CH의 ID (CH_ID), 그리고 한시적인 시간 동안만 유효한 난스 n1을 GW와 SN사이의 공유 비밀키 $K_{SN/GW}$로 암호화한 내용을 같이 전송한다. 여기서 CH는 $K_{SN/GW}$를 모르므로 암호화된 내용을 알 수 없게 된다.

(M2) CH→GW:

$$\{\{SN\_ID,\ CH\_ID,\ n1\}_{K_{SN/GW}},\ CH\_Pass,\ CH\_ID,\ SN\_ID,\ n2\}_{K_{GW+}} \quad (2)$$

메시지 2(M2)에서 CH는 GW에게, SN으로부터 받은 암호문, 자신의 ID(CH_ID)와 패스워드(CH_Pass), 그리고 SN_TID에 해당하는 SN_ID, 난스 n2를 GW의 공개키로 암호화하여 GW에게 전송한다. 이 메시지는 GW의 개인키로만 풀어 볼 수 있으므로 제 3자가 탈취한다 하더라도 내용을 알 수 없다. 즉 이 메시지가 처리되었다는 것은 개인키를 가지고 있는 GW가 수신하였음을 의미한다. GW는 자신의 개인키로 (M2)를 풀어보고, 이어서 $K_{SN/GW}$를 이용하여 SN이 보낸 암호문을 풀어 본 후, SN_ID를 갖는 SN이 해당 CH에 접속을 원하고 또한 인증을 받기를 원하는지를 알게 된다.

(M3) GW→CH:

$$\{GW\_ID,\ GW\_Pass,\ n3\}_{K_{CH+}} \quad (3)$$

SN에 대한 인증 진행 전에 먼저 CH와 GW 간의 인증을 완

료한다. 이를 위해 메시지 3(M3)에서 GW는 자신의 ID (GW_ID)와 패스워드(GW_Pass), 그리고 난스 n3를 CH의 공개키로 암호화하여 CH에게 전송한다. CH는 이 메시지를 자신의 개인키로 풀어 보고, GW_ID 및 GW_Pass를 확인한 후 자신이 상대하고 있는 엔터티가 적절한 GW임을 인증할 수 있게 된다. 따라서 CH에서 본 메시지가 처리되면 CH와 GW 사이의 상호 인증이 완료된다.

2) SN/CH 사이의 상호인증

(M4) GW→CH:

$$\{SN\_ID,\ \{SN\_Pass,\ n4\}_{K_{SN/GW,}}\ n5,\ \{CH\_ID,\ SN\_ID,\ n4\}_{K_{SN/GW}}\}_{K_{CH+}} \quad (4)$$

메시지 4(M4)에서 GW은 SN의 패스워드(SN_Pass)와 난스 n4를 SN과 GW가 공유하고 있는 비밀키($K_{SN/GW}$)로 암호화한 내용, 또 SN이 접속하고자 하는 CH의 ID (CH_ID)와 SN의 ID(SN_ID)를 같은 난스 n4를 포함하여 $K_{SN/GW}$로 암호화한 내용을 CH의 공개키로 암호화하여 CH에게 전송한다. CH는 (M4)를 자신의 개인키로 풀어 본 후 이 메시지가 SN_ID에 해당하는 SN의 상호 인증을 위한 메시지임을 알 수 있게 된다. 이어 (M5), (M6) 메시지에서 볼 수 있듯이 $K_{SN/GW}$로 암호화한 두 개의 내용 중 하나를 CH에게 보내고, 다른 하나는 SN의 인증용으로 사용한다.

M5: CH→SN:

$$\{CH\_ID,\ SN\_ID,\ n4\}_{K_{SN/GW}} \quad (5)$$

CH은 (M4)의 두 개의 암호화된 내용 중 첫 번째인 $\{SN_Pass,\ n4\}_{K_{SN/GW}}$을 저장한 후, 두 번째인 $\{CH_ID,\ SN_ID,\ n4\}_{K_{SN/GW}}$를 메시지 5(M5)에서 SN에 전달한다. SN는 (M5) 메시지를 $K_{SN/GW}$로 풀어보고, GW가 CH_ID에 해당하는 CH를 인증했음을 알게 된다. CH는 (M5)의 추출 내용 중 난스 n4와 자신의 패스워드(SN_Pass)를 이용하여 $\{SN_Pass,\ n4\}_{K_{SN/GW}}$를 생성한다.

(M6) SN→CH:

$$\{SN\_Pass,\ n4\}_{K_{SN/GW}} \quad (6)$$

SN은 메시지 6(M6)에서 CH에게 생성한 $\{SN_Pass,\ n4\}_{K_{SN/GW}}$를 보낸다. CH은 (M6)을 수신한 후 (M4)에서 저장했던 $\{SN_Pass,\ n4\}_{K_{SN/GW}}$와 비교한다. 이 두 개의 암호화된 내용이 같으면, CH는 SN을 인증하게 된다. CH는 비교하는 두 개의 암호화된 내용은 알 수는 없지만, GW가 만들어 보낸 $\{SN_Pass,\ n4\}_{K_{SN/GW}}$에는 SN의 패스워드가 포함되어 있고, 또 난스 n4는 $K_{SN/GW}$을 알고 있는 SN만이 풀어 볼 수 있는 정보이므로, (M6)의 메시지는 적법한 인증된 SN만이 생성할 수 있는 내용이기 때문이다. 이러한 절차가 끝나면 SN과 CH 사이의 상호인증이 완료되며, 전체적으로 3자간 상호인증이 완료된다.

3-3 키 교환과 CHAP 기반의 3자간 상호 인증 프로토콜

본 절에서는 키 교환과 CHAP 기반의 3자 간 인증 프로토콜 절차를 설명한다. 본 프로토콜의 기본적인 가정은 3-2절의 가정과 유사하다. 본 절차에서 사용되는 약어도 3-2절의 약어를 동일하게 사용하며, 그 외에 추가적으로 사용되는 약어는 다음과 같다.

X_{GW}: Diffie-Hellman 키 교환에서 사용되는 GW의 랜덤 개인키.

X_{CH}: Diffie-Hellman 키 교환에서 사용되는 CH의 랜덤 개인키.

$pubValue_{GW}$: Diffie-Hellman 키 교환에서 사용되는 GW의 공개키로 $a^{X_{GW}} \bmod q$의 값을 가짐(a와 q는 전역 공개값).

$pubValue_{CH}$: Diffie-Hellman 키 교환에서 사용되는 CH의 공개키로 $a^{X_{CH}} \bmod q$의 값을 가짐.

Rand1, Rand2: 랜덤 챌린저 값

COUNT1, COUNT2: 서로 연관된 값으로 약속하며, 보통 순서를 나타내는 값으로 연관 지음(예: COUNT2=COUNT+ 1).

본 절에서 제안하는 키 교환과 CHAP 기반의 3자 간 상호 인증 절차는 GW/CH 사이의 상호 인증 및 키 교환 과정과 SN/CH 간의 상호 인증 및 키 설정 과정의 두 단계로 이루어진다.

1) GW/CH 사이의 상호 인증 및 키 교환

(M1) SN→CH:

$$\{SN\_ID,\ CH\_ID,\ n1\}_{K_{SN/GW}},\ SN\_TID,\ GW\_AID \quad (7)$$

메시지 1(M1)의 설명은 3-2절의 패스워드 기반 상호 인증 프로토콜의 식 (1)에 대한 설명과 동일하다.

(M2) CH→GW:

$$\{\{SN\_ID,\ CH\_ID,\ n1\}_{K_{SN/GW}},\ CH\_ID,\ SN\_ID,\ n2,\ COUNT1\}_{K_{GW+}} \quad (8)$$

메시지 2(M2)에서 CH는 GW에게, SN으로부터 받은 암호문, 자신의 ID (CH_ID), 그리고 SN_TID에 해당하는 SN_ID, 난스 n2와 COUNT1을 GW의 공개키로 암호화하여 GW에게 전송한다. 이 메시지는 GW의 개인키로만 풀어 볼 수 있다. GW는 자신의 개인키로 (M2)를 풀어보고, 이어서 $K_{SN/GW}$를 이용하여 SN이 보낸 암호문을 풀어 본 후, SN_ID를 갖는 SN이 해당 CH에 접속을 원하고 또한 인증을 받기를 원하는지를 알게 된다.

(M3) GW→CH:
$$\{pubValue_{GW}, GW\_ID,\ n3\}_{K_{CH+}} \quad (9)$$

(M4) CH→GW:
$$\{pubValue_{CH}, CH\_ID,\ n4\}_{K_{GW+}} \quad (10)$$

GW와 CH 간의 공유 비밀키 설정 단계로서 메시지 3 (M3)과 메시지 4 (M4)에서는 키 교환 프로토콜이 실행된다. 본 절차에서는 대표적인 키 교환 프로토콜인 Diffie-Hellman 키 교환 프로토콜로서 설명한다. (M3)에서 GW은 임의의 숫자 X_{GW}를 선택하고 이 숫자로 $pubValue_{GW}$을 계산하며 이 값을 CH의 공개키로 암호화하여 CH에 전송한다. (M4)에서 CH은 임의의 숫자 X_{CH}을 선택하고 이 숫자로 $pubValue_{CH}$을 계산하며, 이 값을 GW의 공개키로 암호화하여 GW에 전송한다. (M3)과 (M4) 교환 이후, GW과 CH은 서로 교환된 pubValue 값들로 부터 공유 비밀키를 얻게 된다. Diffie-Hellman 키 교환 프로토콜의 경우 공유 비밀키 $K_{SN/GW}$는 다음과 같이 계산된다.

$$\begin{aligned} K_{CH/GW} &= (pubValue_{CH})^{X_{GW}} \bmod q \\ &= (a^{X_{CH}})^{X_{GW}} \bmod q \\ &= (a^{X_{GW}})^{X_{CH}} \bmod q \\ &= (pubValue_{GW})^{X_{CH}} \bmod q \end{aligned} \quad (11)$$

식 (11)의 이해를 돕기 위해 간단한 계산 예를 들면 다음과 같다. 먼저 q = 353, a = 3이라고 하자. 실제로 사용되는 q, a 값은 매우 큰 값이지만, 여기서는 설명을 위해 작은 값으로 예를 든다. 다만 계산과정은 동일하다. 다음으로 X_{GW}=97 그리고 X_{CH}=233이 선택되었다고 하자. 그러면

$$pubValue_{GW} = 3^{97} \bmod 353 = 40$$
$$pubValue_{CH} = 3^{233} \bmod 353 = 248$$

으로 계산된다.

두 개의 pubValue가 서로 교환된 후 GW와 CH는

$$(pubValue_{GW})^{233} \bmod 353 = (40)^{233} \bmod 353 = 160$$
$$(pubValue_{CH})^{40} \bmod 353 = (248)^{97} \bmod 353 = 160$$

의 공유 비밀키 $K_{CH/GW}$를 얻게 된다.

(M5) GW→CH:
$$\{Rand1,\ n3\}_{K_{CH/GW}} \quad (12)$$

(M6) CH→GW:
$$\{Rand1,\ COUNT2\}_{K_{CH/GW}} \quad (13)$$

메시지 5(M5)와 메시지 6(M6)에서는 키 교환 알고리즘으로 만들어진 공유 비밀키에 대한 키 소유 증명을 챌린지 및 응답 과정을 통하여 진행한다. (M5)에서 GW는 랜덤 챌린지 Rand1을 키 교환 알고리즘으로 만들어진 공유 비밀키 $K_{CH/GW}$로 암호화하면 보내면, CH는 동일 키로 이를 풀어 Rand1을 확인하고, 이를 다시 COUNT2와 묶어 다시 공유 비밀키 $K_{CH/GW}$로 암호화해서 보낸다. 이를 수신한 GW는 공유 비밀키 $K_{CH/GW}$로 이를 풀어 Rand1을 확인함으로써 상호 키 소유 증명을 할 수 있다. (M2)와 (M6)에서 COUNT 필드가 사용되는데, 이는 재전송 공격을 방지하기 위해 사용된다. COUNT2의 값은 COUNT1의 값과 상호 연관되게 약속하여 만든다. GW는 (M6)의 COUNT2 필드를 사용하여 (M6)을 보낸 엔터티가 (M2)를 보낸 CH임을 확인한다. 이 절차가 끝나면 GW/CH 사이의 상호 인증 및 키 교환이 완료된다.

2) SN/CH 사이의 상호 인증 및 키 설정

(M7) GW→CH:
$$\{\{SN\_ID,\ Rand2,\ K_{SN/CH},\ n4\}_{K_{SN/GW}},\ CH\_ID,\ K_{SN/CH},\ Rand2,\ n5\}_{K_{CH/GW}} \quad (14)$$

메시지 7(M7)에서 GW은 CH과 SN 간에 사용될 공유 비밀키를 생성하고 이를 CH과 SN에 배포한다. 이를 위해 먼저 SN에게 전달할 내용으로 SN_ID와 Rand2, 그리고 GW가 생성한 CH와 SN 간의 공유 비밀키 $K_{SN/CH}$를 SN과 GW 사이의 공유 비밀키 $K_{SN/GW}$로 암호화한 내용과, CH에게 전달할 $K_{SN/GW}$를 포함한 전체 메시지를 (M3)과 (M4)에서 만들어진 $K_{CH/GW}$로 암호화하여 CH에게 전달한다. (M7)을 받은 CH는 $K_{SN/GW}$를 알게 되며, $K_{SN/GW}$로 암호화된 앞부분은 내용은 알 수 없으나 이를 그대로 메시지 8 (M8)을 통하여 SN에게 전달한다.

(M8) CH→SN:
$$\{SN\_ID,\ Rand2,\ K_{SN/CH},\ n4\}_{K_{SN/GW}} \quad (15)$$

SN은 메시지 8(M8)을 통하여 받은 내용에 대하여 $K_{SN/GW}$을 이용하여 풀게 되면 CH와 SN 사이의 공유 비밀키 $K_{SN/CH}$을 알 수 있게 된다. (M7)과 (M8)에서 Rand2는 SN와 CH의 CHAP 절차에 대한 챌린지로 사용된다. 일반적으로 챌린지는 인증을 원하는 두 엔터티 중 하나가 생성하여 평문으로 다른 엔터티에게 전달된다. 그러나 본 제안 프로토콜에서는 GW가 Rand2를 생성하고 이를 암호화하여 전송함으로서 외부에 노출하지 않도록 하였다.

(M9) SN→CH:

$$\{Rand2,\ n6\}_{K_{SN/CH}} \quad (16)$$

(M10) CH→SN:

$$\{Rand2+1,\ n6\}_{K_{SN/CH}} \quad (17)$$

메시지 9(M9)와 메시지 10(M10)에서는 GW가 생성해 분배해준 CH와 SN 사이의 공유 비밀키($K_{SN/CH}$)에 대한 키 소유 증명을 챌린지 및 응답 과정을 통하여 진행한다. (M9)에서 SN은 (M8)에서의 Rand2를 $K_{SN/CH}$로 암호화하면 보내면, CH는 동일 키로 이를 풀어 추출한 Rand2가 (M7)에서의 Rand2와 동일함을 확인함으로서 SN을 인증한다. CH는 (M10)에서 Rand2+1을 $K_{SN/CH}$로 암호화하여 SN에게 보낸다. SN은 이를 풀어 Rand2+1을 확인함으로서 CH를 인증한다. 이러한 절차가 끝나면 SN과 CH 사이의 상호 인증 및 키 공유가 완료되며, 전체적으로 3자 간 상호 인증이 완료된다.

Ⅳ. 프로토콜 분석

본 절에서는 제안된 프로토콜의 특징과 보안성에 관하여 논의한다. 보안성에 관련된 보안 위협은 제안된 프로토콜에서 사용하는 암호화 알고리즘이 견고하다는 전제하에 프로토콜 절차상의 인증, 재전송 공격, 중간자 공격, 기밀성, 변조 등을 고려하며, 게이트웨이나 클러스터 헤드의 시스템 보안은 안전하다고 가정한다. 또한 센서 노드 및 클러스터 헤드의 탈취 및 복제를 통한 공격에도 견고하다고 가정한다. 편의상, 패스워드 기반 상호 인증을 첫 번째 프로토콜, 키 교환 및 CHAP 기반 상호 인증 프로토콜을 두 번째 프로토콜이라고 한다.

• 운영 구조: 첫 번째 프로토콜에서는 게이트웨이는 모든 클러스터 헤드의 패스워드 테이블을 유지한다. 클러스터 헤드가 생성되거나 삭제될 때, 패스워드 등록이나 철회 처리를 수행한다. 두 번째 프로토콜에서는 게이트웨이와 클러스터 헤드 간의 키 교환 프로토콜이 미리 정의되어야 한다. 본 논문에서는 대표적인 키 교환 프로토콜인 Diffie-Hellman 키 교환 프로토콜로서 설명하였으나, 다른 방법의 채용도 가능하다. 따라서 어떤 보안 키 교환 알고리즘을 선택할지, 그리고 교환된 공유 비밀키가 업데이트되기 전에 얼마나 오래 사용될지를 고려해야 한다.

• 인증: 본 논문에서 설명한 두 가지 프로토콜에서, 게이트웨이는 $K_{SN/GW}$으로 암호화된 (M1)을 성공적으로 복호화하여 센서 노드를 인증할 수 있다. 두 번째 프로토콜에서는 키 교환 프로토콜을 통해 키 교환이 안전하게 실행되면 게이트웨이와 클러스터 헤드 간의 상호 인증이 완료되었다고 볼 수 있다. 센서 노드와 클러스터 헤드 간의 상호 인증의 경우, 첫 번째 프로토콜에서는 인증 메시지가 $K_{SN/GW}$ 및 K_{CH+}로 암호화되므로, 의도된 센서 노드와 클러스터 헤드만 이들을 복호화할 수 있으며, 이로 인해 센서 노드와 클러스터 헤드 간의 상호 인증이 이루어진다. 두 번째 프로토콜에서는 클러스터 헤드와 상호 인증된 게이트웨이가 암호화된 공유 비밀키를 전송하므로 오직 의도된 센서 노드와 클러스터 헤드만 이를 복호화하여 상호 인증을 시작할 수 있다.

• 엔터티 프라이버시: 본 논문에서 설명한 두 가지 프로토콜의 (M1)에서 SN_TID와 GW_AID를 사용함으로써, 센서 노드와 게이트웨이의 실제 식별자는 무선 인터페이스에서 전송되는 메시지에 나타나지 않으며, 이는 트래픽 분석 공격을 방지하는 데 효과적이다. 또한 게이트웨이와 클러스터 헤드 간의 모든 메시지는 둘 사이의 공유 비밀키 또는 공개키로 암호화되므로 엔터티의 프라이버시가 보장된다.

• 재전송 방지: 대부분의 메시지에 포함된 난스는 시간 관련 정보를 포함한다. 이런 경우 이러한 난스들은 메시지의 유효 기간을 확인하는 데 사용될 수 있다. 또한 일부 메시지에서는 순차적인 COUNT 값 필드가 사용된다. 이 경우 엔터티는 처음에 COUNTx 값을 로컬에 저장하고 이를 전송하며 엔터티가 COUNTy 값을 포함한 응답 메시지를 받으면, 이를 저장된 COUNTx 값과 비교하며, 이를 통해 엔터티는 수신된 메시지가 재전송되지 않았으며 정상적인 순서로서 연속적으로 수신되었음을 확인한다.

• 변조 방지: 본 논문에서 설명한 두 가지 프로토콜의 (M2)에서 $K_{SN/GW}$으로 암호화된 CH_ID는 클러스터 헤드와 상호 인증된 게이트웨이 그리고 클러스터 헤드에 현재 센서 노드가 실제로 접속하고 있는지를 확인하는 데 사용된다. 클러스터 헤드는 $\{SN_ID,\ CH_ID,\ n1\}_{K_{SN/GW}}$을 생성하거나 수정할 수 없기 때문에, 게이트웨이는 이 정보를 복호화한 후 클러스터 헤드가 정상적으로 인증을 요청하고 있음을 확인할 수 있다. 첫 번째 프로토콜의 (M5), 두 번째 프로토콜의 (M8)을 받은 후, 센서 노드는 $K_{SN/GW}$ 키로 암호화된 메시지를 복호화하여 클러스터 헤드로부터 온 메시지가 게이트웨이에서 나온 것임을 확인할 수 있으며 이는 변조 방지가 제공된다는 것을 의미한다.

• 자원 사용: 연산 능력의 경우 센서 노드는 제한된 연산 능력을 가지고 있고, 클러스터 헤드와 게이트웨이는 충분한 연산 능력을 가지고 있으며, 또한 게이트웨이와 클러스터 헤드 간의 무선 구간 그리고 클러스터 헤드와 센서 노드들간의 무선 구간은 제한된 대역폭을 갖고 있다. 이에 3자 상호 인증에 필요한 자원 (연산 능력, 대역 폭 등) 사용량에 대해 살펴볼 필요가 있다. 본 프로토콜의 3자 상호 인증 절차는 센서 노드가 클러스터 헤드에 접속할 때 1회성으로 진행되며, 또한 센서 노드는 한 번 설치되면 상당 시간 고정된 상태로 운영되므로, 지속적인 자원 (연산, 대역폭 등) 소모를 하지 않는다. 따라서 본 연구에서의 3자 상호 인증에 사용되는 자원의 양은 전체 무선 센서 네트워크 사용 시간에 비추어 보면 무시할 정도라고 볼 수 있다. 그러므로 본 연구에서 제안한 프로토콜의 정량적, 정성적 자원 사용 비교는 크게 중요하지 않다. 그러나 제안된 프로토콜의 자원 사용량에 대한 기본적인 이해를 위하여 제안한 프로토콜에서의 메시지 수와 암호화 횟수 등을 표 1과

표 1. 메시지 수와 암호화 횟수
Table 1. Number of messages and number of encryptions

	SN ↔ CH			CH ↔ GW		
	No. of messages	No. of encryptions		No. of messages	No. of encryptions	
		Symmetry	Asymmetry		Symmetry	Asymmetry
1st protocol	3	3	0	3	2	2
2nd protocol	4	4	0	6	5	3

같이 분석할 수 있다. 메시지 수는 통신 오버헤드와 관련이 있으며 암호화 횟수는 연산 비용과 관련이 있는 수치이다.

표 1에서 클러스터 헤드와 게이트웨이 사이에는 대칭키 암호화와 공개키 암호화가 모두 사용되는 방면, 센서 노드와 클러스터 헤드 사이에는 대칭키 암호화만 있고 공개키 암호화가 없음을 알 수 있다. 이는 센서 노드의 제한된 컴퓨팅 자원을 고려하여 센서 노드에서는 공개키 암호화 없이 대칭키 암호화만 수행하도록 설계하였기 때문이다. 그리고 2번째 프로토콜에서의 메시지 수가 첫 번째 프로토콜보다 많은 이유는 CHAP 형태로 설계되었기 때문에 챌린지와 응답 과정에서 메시지 수가 증가하였기 때문이다. 그러나 전체적으로 보면 첫 번째 프로토콜에서는 전체 메시지 수는 6개, 암호화 횟수는 대칭키 암호화 5번, 공개키 암호화 2번, 그리고 두 번째 프로토콜에서는 전체 메시지 수 10개에 대칭키 암호화 9번, 공개키 암호화 3번으로 인증에 사용되는 자원의 양은 전체 무선 센서 네트워크 사용 시간에 비추어 무시할 수 있음을 확인할 수 있다.

V. 결 론

본 논문에서는 무선 센서 네트워크를 구성하는 핵심 요소인 센서 노드, 클러스터 헤드, 그리고 게이트웨이 간의 3자 간 상호 인증 과정에 대해 연구하였다. 특히 자원이 제한적인 센서 노드 환경에 적용이 용이한 인증 구조를 고려하였으며, 이를 바탕으로 패스워드 기반 상호 인증 프로토콜과 키 교환 및 CHAP 기반의 상호 인증 프로토콜을 제안하였다. 또한 제안된 각 프로토콜에 대해 운영 구조, 인증, 엔터티 프라이버시, 재전송 방지, 변조 방지 등 다양한 요소를 고려한 보안 분석을 수행함으로써, 프로토콜의 안전성과 신뢰성을 검증하였다. 기존의 인증 기법들은 주로 두 당사자 간의 인증에 초점을 맞추고 있어, 무선 센서 네트워크와 같이 다수의 노드가 계층적으로 구성된 환경에서는 보안상 한계가 존재하였다. 반면, 본 연구에서 제안한 3자 간 상호 인증 방식은 센서 노드, 클러스터 헤드, 게이트웨이라는 세 개체가 모두 상호 신뢰를 형성하도록 설계됨으로써, 네트워크 전반의 보안 수준을 향상시키는 동시에 인증 과정의 신뢰성을 강화할 수 있었다. 결론적으로, 본 논문에서 제안한 3자 간 상호 인증 방식은 센서 노드와 클러스터 헤드, 그리고 게이트웨이 간의 간결하면서도 안전한 인증 메커니즘을 제공함으로써 IoT 무선 센서 네트워크 환경에서 효과적으로 활용될 수 있을 것으로 판단된다. 향후 연구에서는 ProVerif 등의 형식적 검증 도구를 적용한 보안성 분석 및 실제 네트워크 환경에서의 다양한 공격 모델에 대한 추가 검증을 통해 제안 기법의 실용성과 확장성을 더욱 강화할 수 있을 것으로 기대된다.

참고문헌

[1] J. H. Choi, “A Blockchain-Based Mutual Authentication Scheme between IoT Devices and Edge Servers in Edge Cloud Environments,” *Journal of Digital Contents Society*, Vol. 24, No. 4, pp. 815-825, April 2023. https://doi.org/10.9728/dcs.2023.24.4.815

[2] S. Szymoniak, “Key Distribution and Authentication Protocols in Wireless Sensor Networks: A Survey,” *ACM Computing Surveys*, Vol. 56, No. 6, pp. 1-31, January 2024. https://doi.org/10.1145/3638043

[3] M. Abdalla, P.-A. Fouque, and D. Pointcheval. “Password-Based Authenticated Key Exchange in the Three-Party Setting,” in *Proceedings of the 8th International Workshop on Theory and Practice in Public Key Cryptography - PKC 2005*, Les Diablerets, Switzerland, pp. 65-84, January 2005. https://doi.org/10.1007/978-3-540-30580-4_6

[4] S. Guo, Y. Song, S. Guo, Y. Yang, and S. Song, “Three-Party Password Authentication and Key Exchange Protocol Based on MLWE,” *Symmetry*, Vol. 15, No. 9, September 2023. https://doi.org/10.3390/sym15091750

[5] G. Thakur, S. Prajapat, P. Kumar, and C.-M. Chen, “A Privacy-Preserving Three-Factor Authentication System for IoT-Enabled Wireless Sensor Networks,” *Journal of System Architecture*, Vol. 154, 103245, 2024. https://doi.org/10.1016/j.sysarc.2024.103245

[6] W. Huang, “ECC-Based Three-Factor Authentication and Key Agreement Scheme for Wireless Sensor Networks,” *Scientific Reports*, Vol. 14, 1787, January 2024. https://doi.org/10.1038/s41598-024-52134-z

[7] L. Wang and C. Han, “Multi-Factor Authentication and kEy Agreement Scheme Based on PUF and Chebyshev Chaotic Map for Wireless Sensor Networks,” *Scientific Reports*, Vol. 16, December 2025. https://doi.org/10.1038/s41598-025-332

17-x

[8] S. Byun, J. Ryu, Y. Choi, and H. Lee, “Improved Secure Three-Factor-Based Mutual Authentication Scheme for Wireless Sensor Networks in Internet of Things Environments,” *Cluster Computing*, Vol. 29, 47, November 2025. https://doi.org/10.1007/s10586-025-05820-0

[9] J.-H. Yang, “A Multi-Gateway Authentication and Key-Agreement Scheme on Wireless Sensor Networks for IoT,” *EURASIP Journal on Infomation Security*, Vol. 2023. 2, March 2023. https://doi.org/10.1186/s13635-023-00138-z

[10] R. Mishra and A. Mishra, “Current Research on Internet of Things (IoT) Security Protocols: A Survey,” *Computers & Security*, Vol. 151, April 2025. https://doi.org/10.1016/j.cose.2024.104310

[11] J. Lee, J. Oh, and Y. Park. “A Secure and Anonymous Authentication Protocol Based on Three-Factor Wireless Medical Sensor Networks,” *Electronics,* Vol. 12, No. 6, March 2023. https://doi.org/10.3390/electronics12061368

이구연(Goo Yeon Lee)

1986년 : 서울대학교 전자공학과 (학사)
1988년 : KAIST 전기및전자공학과 (석사)
1993년 : KAIST 전기및전자공학과 (박사)
1993년～1996년: 디지콤정보통신 연구소
1996년: 삼성전자
2004년 7월～2005년 2월: 미국 Cornell 대학교 방문교수
2010년 1월～2011년 1월: 미국 Cornell 대학교 방문교수
2012년 8월～2014년 2월: 강원대학교 IT대학 부학장
1997년～현 재: 강원대학교 컴퓨터공학과 교수
※관심분야 : 데이터통신, 컴퓨터네트워크, 네트워크 보안, 네트워크 성능분석, 암호학, 정보보호관리체계

이용(Yong Lee)

1997년 8월 : 연세대학교 컴퓨터과학과 (이학석사)
2001년 2월 : 연세대학교 컴퓨터과학과 (공학박사)
2001년～2003년: 한국정보보호진흥원 선임연구원
2004년～2005년, 2009년～2012년: 코넬대학교 방문연구원
2005년～2007년: 삼성전자 통신연구소 책임연구원
2007년～2011년: 충주대학교 전자통신공학전공 조교수
2021년～2022년: 배화여자대학교 조교수
2026년～현 재: 덕성여자대학교 디지털소프트웨어공학부 초빙교수
※관심분야 : 네트워크 보안, 차세대 인터넷, IoT보안, 이동통신망 보안, 정보보호