Journal of the Korea Academia-Industrial cooperation Society
Vol. 24, No. 11 pp. 886-892, 2023

https://doi.org/10.5762/KAIS.2023.24.11.886
ISSN 1975-4701 / eISSN 2288-4688

USN/IoT 통신환경에 적용 가능한 분산-해시함수 알고리즘에 관한 연구

이선근
우석대학교 전기자동차공학부

A Study on the Distributed-Hash Function Algorithm Applicable to USN/IoT Communication Environment

Seon-Keun Lee
School of Electric Vehicle Engineering, Woosuk University

요 약 저용량/저속 데이터들을 사용하는 USN/IoT 환경의 복잡도 증가에 적합한 안전성을 확보하기 위하여 경량화 암호알고리즘이 사용되지만 본 논문은 암호알고리즘 대신에 보안기능을 위한 분산 해시함수를 적용한 DHC를 제안하였다. 제안된 DHC는 USN/IoT환경의 다양한 변화 및 복잡도 증가 등에 유리함과 동시에 최소한의 안전성을 보장할 수 있기 때문에 경량화 암호알고리즘의 대체 및 PQC로써의 가능성을 제시한다. 또한 USN/IoT 센서노드들의 확장성에 독립적으로 작용할 수 있는 DHC는 향후 네트워크의 복잡도 증가에 따른 보안 성능의 감소 등에 영향을 끼치지 않을 것으로 생각한다.

Abstract A lightweight cryptographic algorithm was used to ensure security appropriate for the increased complexity of the ubiquitous sensor network/Internet of Things (USN/IoT) environment using low-capacity/low-speed data. On the other hand, this paper proposes a distributed hash-function in embedded cryptosystem(DHC) that applies a distributed hash function for security functions instead of a cryptographic algorithm. The proposed DHC is advantageous for various changes and increased complexity in the USN/IoT environment and can guarantee minimum safety concurrently, suggesting its potential as a replacement for lightweight cryptographic algorithms and as post-quantum cryptography (PQC). In addition, DHC, which can operate independently of the scalability of USN/IoT sensor nodes, is not expected to affect the decrease in security performance due to increased network complexity in the future.

Keywords : Lightweight, USN/IoT, Hash Function, Avalanche Effect, Distributed Processing

1. 서론

개인, 물류 유통, 국방, 교통, 보건, 복지, 그리고 다양한 환경 등의 분야에 무궁무진하게 적용되는 유비쿼터스/IoT[1] 환경은 미래 사회의 기반 인프라로 자리 잡았으며 더욱 큰 발전을 이루게 될 것이다. 이러한 유비쿼터스/IoT 환경을 구성하는 USN/IoT(Ubiquitous Sensor Network/Internet of Things)은 다수의 센서 노드로 구성된 네트워크로써 다양한 위치에 설치된 노드들로부터 환경 정보를 인식하고 인식된 정보를 통합/가공하여 언제, 어디서나, 안전하고 자유롭게 이용할 수 있게 하는 정보서비스이다[2].

USN/IoT는 다양한 환경에서 주변상황을 모니터링하고 필요한 정보를 센싱하는 용도로 사용되기 때문에 센

*Corresponding Author : Seon-Keun Lee(Woosuk Univ.)
email: caiserrisk@woosuk.ac.kr
Received July 26, 2023 Revised September 20, 2023
Accepted November 3, 2023 Published November 30, 2023

서노드들에 대한 신뢰성 및 안전성이 매우 중요하다[3]. 이를 위하여 다양한 암호화 기법들이 개발되었으며 직접 적용되고 있다. 그러나 전송량, 소비전력, 복잡도 등의 다양한 파라미터 측면에서 센서네트워크 환경에 적용되던 대칭형/비대칭형 암호알고리즘들은 보안성에만 중점을 두지 않고 전체적인 성능을 고려할 경우, USN/IoT에 적용하기에는 다소 무리가 있다. 비대칭형 방식의 RSA, ElGamal[4], 대칭형 방식의 AES 등의 암호알고리즘은 USN/IoT 영역에서 적은 데이터량, 소비전력, 키 관리의 어려움과 같은 다양한 요인으로 USN/IoT 환경에서 적합하다고 볼 수 없다. 또한 경량화 암호알고리즘의 경우에도, 센서 노드들의 수가 많을수록 효용성은 크지 않다[5].

본 논문은 이와 같은 이유 때문에 USN/IoT 환경에 기존 비대칭형/대칭형 암호알고리즘을 사용하지 않고 해시함수를 적용하여 센서시스템이 포함된 네트워크의 부하를 줄이며 안전성을 보장할 수 있도록 하는 분산형 해시함수 기반 임베디드 암호시스템(DHC : Distributed Hash-function in embedded Cryptosystem)을 설계하기 위한 전처리과정으로 분산처리가 가능한 변형된 해시함수 알고리즘을 제안하였고 이에 대한 모델링을 수행하였다.

제안된 DHC 알고리즘은 소비전력, 키 관리, 복잡도, 안전성 등의 다양한 파라미터에 대하여, USN/IoT 분야에 적용할 경우, 분산된 네크워크 환경에서 보다 가벼운 보안시스템으로 작용할 것이다.

2. 해시함수 및 관련연구

2.1 해시함수

해시함수는 임의의 긴 입력값을 적절하게 처리하여 짧은 출력값을 산출하는 함수이다. 해시함수에 의해 축약된 메시지에 서명을 하면, 서명에 필요한 계산량, 메모리, 전송량이 크게 줄어든다[6,7].

해시함수는 중요한 정보의 인증과 무결성을 해결하기 위한 수단으로 사용할 수 있다. 대부분의 컴퓨터 바이러스는 정상적인 프로그램에 불법적으로 은밀하게 삽입되어 무작위 또는 인위적으로 전파된다. 따라서 분산되어 있는 센서시스템들의 데이터를 임의 또는 정해진 경로를 통해 접한 경우, 그 데이터가 오염되지 않은 정상적인 데이터인지 반드시 확인할 필요가 있다. D를 전송 받은 데이터이고 $h(D)$를 안전한 장소에 저장되어 있는 D의 해시값일 경우, 데이터의 무결성을 확인하려면 해시값을 계산한 다음 $h(D)$와 비교하여 보면 된다. 만약 바이러스 등과 같은 불법적인 코드가 삽입되거나 데이터의 변조가 발생한다면 해시값은 당연히 일치하지 않게 되므로 D에 대한 오염을 확인할 수 있다.

전자서명의 효율성을 높이고 중요한 정보의 무결성을 확인하기 위한 수단으로서의 해시함수 h는 다음과 같은 조건을 만족하고 있어야 한다[8].

i) 임의의 입력값 x를 고정된 길이의 해시값 $(y=h(x))$으로 출력한다. 이때 해시값은 가능하면 작아야 한다.

ii) 해시함수값 $y=h(x)$로부터 x를 역으로 계산하는 것이 불가능(computationally infeasible)하여야 한다. 이러한 성질을 가진 해시함수를 일방향 함수라고도 한다.

iii) $h(x)=h(x')$인 x'을 찾는 것이 계산상 불가능하여야 한다(collision-free).

iv) 계산 효율이 좋아야 한다.

해시함수는 기본적으로 긴 길이의 입력값을 짧은 길이로 출력하므로 충돌쌍은 이론적으로 반드시 존재한다. 따라서 완벽한 충돌회피성을 갖는 해시함수는 있을 수가 없다. 여기서 말하는 충돌회피성은 충돌쌍이 존재하기는 하지만 그 쌍을 찾아내기가 매우 어렵다는 의미이다.

충돌쌍이 발견되면 전자서명 때 송신자가 메시지 x를 보내고, 나중에 메시지 x'를 보냈다고 하는 내부 부정이 가능해진다. 또한 메시지의 무결성을 확인할 때 실제 메시지의 변화를 확인자가 감지 할 수 없는 경우도 발생한다. 그러므로 해시함수의 안전성은 충돌쌍 발견의 어려움에 의존하게 되며, 반대로 충돌쌍을 찾는 것이 해시함수를 공격하는 첫 번째 단계이다. 해시함수가 안전하다는 것은 적용 가능한 모든 공격이 전수공격(brute force)보다 더 어렵다는 것을 의미한다[9,10].

2.2 관련연구

해싱연구는 데이터 독립형과 데이터 종속형으로 분류되며, 이후 이에 대한 변형 연구가 진행되고 있다. 최근 연구에서는 데이터 구조나 부가정보를 고려하여 해시함수를 학습하는 데이터 종속기술을 개발하는데 중점을 두고 있다. 대표적으로 앵커 그래프 해싱[11], 반복 양자화(ITQ)[12], 구형 해싱[13] 등이 있다.

대부분의 기존 해시 알고리즘은 중앙 집중식을 위해 개발되었다. 즉, 단일 시스템 접근 방식이다. 그러나 기술이 발달하면서 실제 응용 프로그램에는 점점 더 많은 대용량 데이터 그룹이 등장하고 이에 따라 데이터는 분

산 데이터베이스[14], 네트워크를 통한 스트림 데이터들에 대한 다양한 위치에 분산되는 경우가 많다. 특히, 모바일 사용 및 센서 네트워크에서 데이터는 분산된 사이트에서 수집된다[15]. 이러한 맥락에서 데이터에 대한 오류없는 코드값을 얻기 위해서는 전체 데이터를 기반으로 해시 함수를 학습해야 한다. 이를 위한 방식중의 하나는 모든 데이터를 융합센터에 수집하는 것이다. 그러나 이는 비현실적인 통신 오버헤드로 인해 실행 가능한 옵션이 아니다. 특히 대용량 데이터를 직접 수집하는 것은 시간과 공간 측면에서 보았을 때 엄청난 계산비용이 들기 때문에 실제 적용이 비현실적이다. 결과적으로 분산 환경에서 효과적인 해시 알고리즘을 개발하는 것이 가장 중요하다.

3. DHC 알고리즘

3.1 USN/IoT

USN/IoT는 여러 개의 센서 네트워크 필드가 게이트웨이를 통해 외부 네트워크에 연결되는 구조를 갖는다. 센서 노드들은 가까운 싱크 노드로 데이터를 전송하고, 싱크 노드로 집속된 데이터는 게이트웨이로 전송된다. 게이트웨이에서 관리자에게 전달되는 데이터는 위성통신, 유/무선 인터넷 등을 통해 전송될 수 있으며, 이런 접속망(Access Network)은 기존 인프라를 사용한다[16].

USN의 전체적인 구조는 Fig. 1과 같다. 접속망은 IPv6 기반의 BcN으로 인터넷 통합망을 의미하게 되는데, 이것은 모든 센서 노드에 IPv6가 적용될 수 있음을 뜻한다[16,17]. 또한 센서 네트워크의 애플리케이션을 위해 미들웨어로써 서비스 플랫폼이 제공되어 사용자는 이를 통해 차세대 네트워크인 지능형 센서 네트워크를 자유롭게 이용하게 된다.

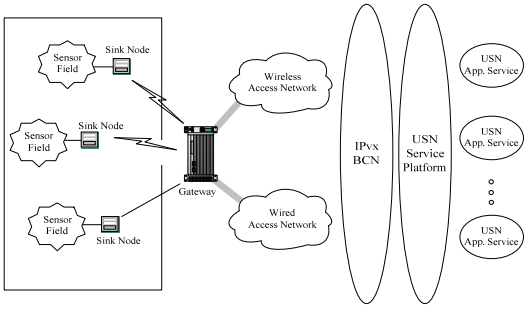

Fig. 1. USN architecture

USN이 완성되기 위해 주목해야 하는 부분은 센서 네트워크 필드 부분이다. 싱크 노드에서 게이트웨이를 거치는 접속망 이상의 분야는 USN의 통합적인 발전을 위한 기술임과 동시에 정책적 사안으로 발전되며, 일반 통신망과 크게 다르지 않다. 그러나 센서 네트워크 필드 분야는 USN의 특징을 가장 많이 가지면서 일반적이지 않은 특징을 가진다.

센서 네트워크 필드는 네트워크를 구성하는 일정 지역에 크기가 작은 노드들이 수 개에서 수천 개까지 설치되어 통신하는 구조를 갖는다. 또한 노드들이 주고 받는 데이터는 그 크기도 작고 데이터의 발생 빈도 또한 매우 낮아 통신하는 양은 많지 않으며, 통신하는 데이터들은 hard/soft-RTOS의 특징을 모두 가진다[17]. 이런 특징 때문에 센서 네트워크 필드에 암호화를 적용하게 되지만 암호 모듈의 복잡성, 계산량, 소비전력 등은 무시할 수 없는 파라미터이다[7].

3.2 DHC 알고리즘

본 논문은 USN/IoT 센서 네트워크 필드에서 다양한 저용량/저속의 디바이스들에 데이터 무결성이 가능하도록 하기 위하여 해시함수를 분산처리가 가능함과 동시에 수많은 디바이스들에 대한 충돌쌍 발생확률을 줄일 수 있도록 하는 USN/IoT 환경용 DHC 해시 알고리즘을 제안하였다.

Fig. 2는 센서 네트워크 필드가 ZigBee로 구성되어 있을 경우에 토포로지를 나타낸 것이다[18,19]. 여기에서 라우터에 집속되는 각 코디네이터들에 대한 데이터들은 해시함수를 이용하여 분산처리되며 하나의 해시함수 f는 모두 동일한 기초 해시함수를 사용한다. 일반적으로 USN/IoT 환경은 Fig. 2와 같이 디바이스들이 계층적인 토포라지를 가진다. 그러므로 본 연구에서 제안하는 DHC 알고리즘 역시 이러한 계층적 토포라지 환경에서 적용할 수 있도록 하였다.

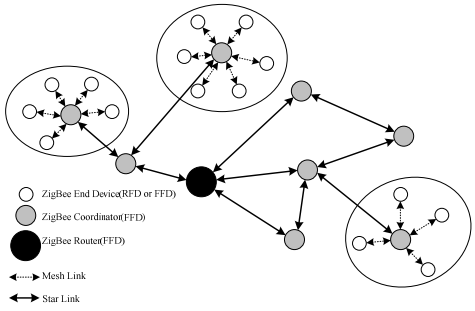

Fig. 2. ZigBee sensor field topology

Fig. 3은 ZigBee 토포로지[19,20]에서 각 구성요소인 엔드 디바이스, 코디네이터, 라우터 등에 대해서 각 레벨마다 공통의 해시함수를 사용하는 DHC의 기본 구조이다.

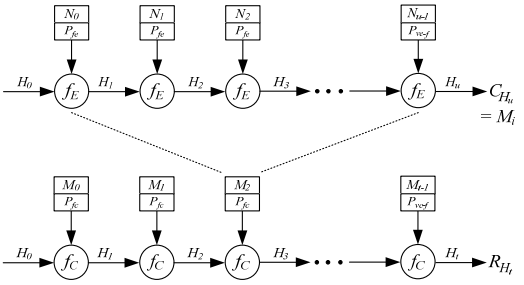

Fig. 3. DHC hash function level

이때 P_{fe} : 고정 길이를 가지는 엔드 디바이스의 패딩값, P_{ve-f} : 가변길이를 가지는 엔드 디바이스의 패딩값, P_{fc} : 고정 길이를 가지는 코디네이터의 패딩값, P_{vc-f} : 가변 길이를 가지는 코디네이터의 패딩값, N_i : M_i를 구성하는 세부 블록크기, M_i : 전체 블록크기인 M에 대한 세부 블록크기, f_E : 엔드 디바이스 레벨에서 사용되는 해시함수, f_C : 코디네이터 레벨에서 사용되는 해시함수, H_0 : 초기값, H_u : 엔드 디바이스 레벨에서 최종 해시값, H_t : 코디네이터 레벨에서 최종 해시값이다.

라우터에서 임의의 길이를 가진 데이터 M을 k비트 블록으로 잘라서 각각을 $M_0, M_1, M_2, \cdots, M_{t-1}$라고 하며, M_i는 k비트의 크기를 가지는 코디네이터 레벨에서 세부 블록의 값이다.

Merkle의 Meta 방식[21]에서 일반적으로 메시지 전체의 비트 길이가 k의 배수인 경우는 별로 없으므로 마지막 블록 M_{t-1}은 적절한 비트를 패딩하여 k비트가 되도록 조절하지만, 본 논문에서 제안한 DHC 기법은 각 세부 블록마다 고정된 패딩을 사용하여 각 하부 디바이스 세부블록들이 항상 일정한 크기의 값을 가지도록 하며 다음 레벨로 넘어갈 때는 전체에 균일하게 배정하였던 패딩값을 제거하여 순수한 데이터만을 전송한다.

DHC 알고리즘에 대한 모델링을 위하여 Keccak [22,23] 알고리즘군 중 하나인 Keccak-256 알고리즘을 중심으로 진행하였다. Keccak-256은 미국 NIST가 승인하여 사용중인 해시함수 SHA-1, SHA-2에 내재하는 보안 취약점을 개선하고자 개발된 암호화 해시알고리즘이다[23].

분산된 해시함수 모델에서, 전역 데이터 $M = \sum_{i=0} M_i$에 대한 네트워크 계층 토포라지를 위하여 DHC 알고리즘에 대한 내용은 다음과 같다.

$$H_u = f_C(H_{j-1}, M_{j-1} \| P_{f1}) \quad (1)$$
$$j = 1,2,3,4,\cdots,u$$

$$R_{H_t} = \quad (2)$$
$$= H_t$$
$$= f_C(H_{i-1}, M_{i-1} \| P_{f2})$$
$$i = 1,2,3,4,\cdots,t$$

$$P_{f1} = P_{fe} \ \& \ P_{ve-f} \quad (3)$$

$$P_{f2} = P_{fc} \ \& \ P_{vc-f}$$

Meta 방식에 의해 설계된 해시함수는 기초 해시함수 f의 안전성에 관계없이 서로 길이가 다른 충돌쌍을 쉽게 찾을 수 있다. 그러나 이러한 약점은 Eq. (4)와 같이 메시지의 전체 길이를 나타내는 값을 비트열로 표현한 블록 N_{u+1}, M_{t+1}을 추가함으로써 해결할 수 있다.

$$a = N + N_{u+1} \quad (4)$$
$$b = M + M_{t+1}$$

$$a < 2^u, \ \ b < 2^t$$

그러므로 DHC는 메시지 $N = N_1 N_2 \cdots N_u$, $M = M_1 M_2 \cdots M_t$를 해시하기 전에 메시지의 길이를 나타내는 a, b를 이진 수열로 표현한 블록 N_{u+1}, M_{t+1}을 메시지 블록에 덧붙이며, 각 분리된 메시지 블록들에 대하여 고정 및 가변 패딩을 수행하였다.

이러한 일련의 과정은 하나의 해시데이터를 분리하여 해싱을 수행한 후, 데이터를 전송하고 해시된 데이터들을 결합하여 해시정보를 확인함으로써 데이터의 무결성을 보장하게 된다.

이와 같이 구성한 DHC의 특징은 Eq. (2)와 Fig. 3에서 보는것과 같이 분산되어 구성된 디바이스들에 대한 각각의 해시값을 이용하여 데이터의 무결성을 확인하는 것이 아니라, 하나의 해시값을 모든 디바이스에 각각 분리하여 적용한 후, 최종값을 해시함으로써 디바이스들에 대한 데이터 무결성을 확보함과 동시에 각 디바이스들에 대한 상태여부를 확인할 수 있다. 이와 동시에 충돌쌍에 대한 내용을 찾기 어렵게 하기 위하여 본 DHC는 하나의 해시값을 모든 디바이스에 분리하여 분산배치하고 이를

최종 해시값으로 다시 활용하는 것이다. 이러한 DHC 알고리즘에 대한 분산후 결합과정에서 발생할 수 있는 충돌쌍에 대한 검증을 우선 확인할 필요성이 있다. 이러한 이유로 본 논문에서는 제안한 DHC에 대한 충돌쌍 발생 여부에 대한 모델링을 수행하는 것을 최우선시 하였다.

일반적으로, USN/IoT 네트워크에서 디바이스들에 대한 상태는 별도의 시스템을 이용하여 모니터링을 수행하고 있기 때문에 디바이스들에 대한 상태를 확인하는 것 또한 별도의 부담이 된다. 그런데 본 논문에서 제안한 DHC 알고리즘의 특징은 하나의 해시함수를 이용하여 디바이스 수에 무관하게 해시 무결성을 확인할 수 있음과 동시에 디바이스들에 대한 상태모니터링도 수행할 수 있다는 점이다.

4. 모의실험

제안된 DHC 알고리즘은 DHC에 대한 검증 및 구현을 위한 전처리 과정으로 DHC에 대한 동작 확인 및 이에 대한 검증을 위하여 일반적으로 사용되는 해시함수 avalanche effect를 중심으로 DHC 모델링을 수행하였다. 해시함수 avalanche effect는 입력의 작은 변화가 출력에 큰 영향을 미치고 통계적으로 무작위와 구별이 되지 않는 것을 의미한다.

Keccak-256은 고정길이 해시 알고리즘으로써 스펀지 함수라는 XOF(eXtendable-Output Function)로 구성되어 있다[23]. 스펀지 구조는 임의 길이의 데이터 입력 및 임의 길이 데이터 출력을 허용하기 때문에 네트워크상에서 매우 큰 융통성을 가진다.

Keccak-256은 임의 길이의 값을 입력하더라도 임의 길이의 결과값을 출력하기 때문에 본 논문에서 목표로 하는 USN/IoT 분야에 최적이라고 생각되기 때문에 Keccak-256을 중심으로 DHC 알고리즘을 전개하였다. USN/IoT 특성상, 적용분야 및 환경에 따라 디바이스의 수가 매우 가변적이다. 그러므로 환경변수에 따른 디바이스들의 가변에 최적화된 것이 Keccak-256이며 이를 DHC에 적용하였다.

Keccak 시리즈는 SHA3로써 가장 중요한 인자는 avalanche effect와 처리속도이다. 이 두 인자들은 모두 Keccak 스펀지 구조에 의해 발생되는 문제들이다.

스펀지 구조에 의해 발생되는 문제를 해결하기 위하여 스펀지 구조를 분산처리 할 수 있도록 하면 avalanche effect에 대한 특징이 강해져 이를 해결할 필요성이 있다.

분산처리에 의한 문제를 해결하기 위하여 스펀지 구조에 Fig. 3을 포함하는 DHC 알고리즘을 적용하여 avalanche effect에 영향을 미치는 영향을 최소화 하도록 하였다.

해시함수에서 avalanche effect는 입력 중 하나의 반전 비트가 평균 출력 비트의 절반의 변화를 일으킨다는 것이다. 이러한 특징을 Keccak-256 해시함수에 적용하여 avalanche effect가 나타나는지를 확인하였다. Fig. 4는 Keccak-256에 대한 avalanche effect를 확인한 그림이다.

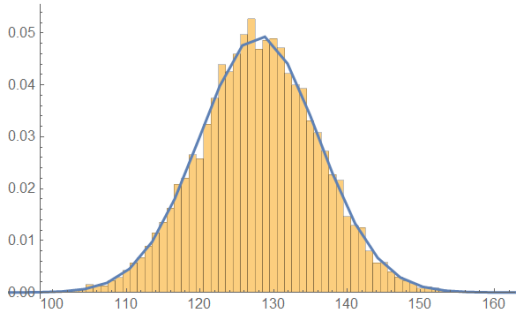

Fig. 4 DHC hash function using Keccak-256

Keccak-256을 Fig. 3과 같이 변형하여 바이트 배열을 해시하고, 각 쌍에 대하여 해밍거리를 계산한 후, 이에 대한 평균값을 계산하여 DHC가 avalanche effect 및 암호화에 효율적인지를 확인하였다. 이를 위하여 입/출력 데이터들에 대하여 계산되어진 해밍거리 및 표준편차를 갖는 정규분포 편미분 방정식을 구하고 이를 히스토그램으로 표현한 것이 Fig. 4이다. Fig. 4에서 정규분포를 약간 벗어나긴 하였어도 동작에 대한 검증은 이상 없음을 알 수 있다.

5. 결론

본 논문은 USN/IoT 분야에서 분산된 디바이스들에 대한 데이터 보호 및 변조를 막기 위하여 해시 및 인증을 수행할 수 있는 SHA3 Keccak 계열의 Keccak-256을 이용한 DHC 알고리즘을 모델링하였다.

DHC 알고리즘은 기존 Keccak 계열의 스펀지 구조로 인한 단점을 없앰과 동시에 분산된 디바이스들에 대한 데이터 보호를 수행할 수 있도록 하나의 해시함수를 적용할 수 있는 방안을 제시하였다.

제안된 DHC 알고리즘은 USN/IoT 네트워크 내에 분

산되어 있는 다양한 디바이스들에 대하여 하나의 해시함수를 적용하고 이를 처리할 수 있는 방법을 제시함으로써 PQC에 대한 방안을 제시하고 소량, 다종류 특징을 가지는 데이터들에 대한 보안을 강화할 수 있을 것으로 생각된다.

References

[1] Kamal Gulati, Raja Sarath Kumar Boddu, Dhiraj Kapila, Sunil L. Bangare, Neeraj Chandnani, G. Saravanan, "A review paper on wireless sensor network techniques in Internet of Things (IoT)", *Materials Today: Proceedings 51*, pp. 161-165, 2022, DOI: https://doi.org/10.1016/j.matpr.2021.05.067

[2] Tahniyat Aslam, Seema Ansari, Fatima Maqbool, Adeel Ansari, "Internet of Things Based Monitoring of Remote Patients", *Part of the Lecture Notes on Data Engineering and Communications Technologies book series (LNDECT, volume 78)*, 2021.
DOI: https://doi.org/10.1007/978-3-030-79203-9_34

[3] Pampa Sadhukhan, Firoj Gazi, "An IoT based Intelligent Traffic Congestion Control System for Road Crossings", *International Conference on Communication, Computing and Internet of Things (IC3IoT)*, pp. 404-408, 2018.
DOI: https://doi.org/10.1109/IC3IoT.2018.8668131

[4] Fatma Mallouli, Aya Hellal, Nahla Sharief Saeed, Fatimah Abdulraheem Alzahrani, "A Survey on Cryptography: Comparative Study between RSA vs ECC Algorithms, and RSA vs El-Gamal Algorithms", *2019 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/ 2019 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, pp. 173-176, 2019.
DOI: https://doi.org/10.1109/CSCloud/EdgeCom.2019.00022

[5] Muhammad Rana, Quazi Mamun, Rafiqul Islam, "Lightweight cryptography in IoT networks: A survey", *Future Generation Computer Systems*, pp. 77-89, 2022, DOI: https://doi.org/10.1016/j.future.2021.11.011

[6] Rajeev Sobti, G. Geetha, "Cryptographic Hash Functions: A Review", *International Journal of Computer Science Issues*, Vol. 9, Issue 2, No 2, pp. 461-479, 2012.

[7] Y. J. Yang, Fei chen, Xiaomei Zhang, Jianping Yu, Peng Zhang, "Research on the Hash Function Structures and its Application", *Wireless Personal Communications* Vol. 94, pp. 2969-2985, 2017.
DOI: https://doi.org/10.1007/s11277-016-3760-4

[8] Jean-Sébastien Coron, Yevgeniy Dodis, Cécile Malinaud, Prashant Puniya, "Merkle-Damgård Revisited: How to Construct a Hash Function", *Advances in Cryptology–CRYPTO 2005*, pp. 430-448, 2005.
DOI: https://doi.org/10.1007/11535218_26

[9] Ivan Bjerre Damgård, "A Design Principle for Hash Functions", *Advances in Cryptology—CRYPTO'89 Proceedings* pp.416-427, 1989.
DOI: https://doi.org/10.1007/0-387-34805-0_39

[10] Justin Kang, Wei Yu, "Minimum Feedback for Collision-Free Scheduling in Massive Random Access", *IEEE Transactions on Information Theory*, Volume 67 Issue 12, pp.8094-8108, 2021.
DOI: https://doi.org/10.1109/ISIT44484.2020.9174336

[11] Liu, W., Wang, J., Ji, R., Jiang, Y., and Chang, S. Supervised hashing with kernels. *In IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[12] Gong, Yunchao, Lazebnik, Svetlana, Gordo, Albert, and Perronnin, Florent. "Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(12):2916-2929, 2013.

[13] Heo, J., Lee, Y., He, J., Chang, S., and Yoon, S. Spherical hashing. *In IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[14] Corbett, James C, Dean, Jeffrey, Epstein, Michael, Fikes, Andrew, Frost, Christopher, Furman, JJ, Ghemawat, Sanjay, Gubarev, Andrey, Heiser, Christopher, Hochschild, Peter, et al. Spanner: Googles globally distributed database. *ACM Transactions on Computer Systems (TOCS)*, 31(3):8, 2013.

[15] Liang, Junli, Zhang, Miaohua, Zeng, Xianyu, and Yu, Guoyang, "Distributed dictionary learning for sparse representation in sensor networks", *Image Processing, IEEE Transactions on*, 23(6): 2528-2541, 2014.

[16] Zhijia Yue, Wanrong Sun, Peijia Li, Masood Ur Rehman, Xiaodong Yang, "Internet of things: Architecture, technology and key problems in implementation", *2015 8th International Congress on Image and Signal Processing*, pp.1298-1302, 2015.,
DOI: https://doi.org/10.1109/CISP.2015.7408082

[17] Aleksandar Milinković, Stevan Milinković, "Choosing the right RTOS for IoT platform", *INFOTEH-JAHORINA* Vol. 14, pp.504-509, 2015.

[18] Emerson Navarro, Nuno Costa, António Pereira, "A Systematic Review of IoT Solutions for Smart Farming", *Sensors 2020*, 2020.,
DOI: https://doi.org/10.3390/s20154231

[19] Foughali Karim, Fathalah Karim, Ali frihida, "Monitoring system using web of things in precision agriculture", *Procedia Comput. Sci.*, pp.402-409, 2017,
DOI: https://doi.org/10.1016/j.procs.2017.06.083

[20] Luis Manuel Fernández-Ahumada, Jose Ramírez-Faz, Marcos Torres-Romero, Rafael López-Luque, "Proposal for the Design of Monitoring and Operating Irrigation Networks Based on IoT, Cloud Computing and Free Hardware Technologies", *Sensors 2019.*
DOI: https://doi.org/10.3390/s19102318

[21] Muhammad Saqib Niaz, Gunter Saake, "Merkle hash tree based techniques for data integrity of outsourced

data", *CEUR Workshop Proceedings*. 1366, pp.66-71, 2015.

[22] José R.C. Cruz, "Keccak: The New SHA-3 Encryption Standard", Dr Dobb's, https://www.drdobbs.com/security/keccak-the-new-sha-3-encryption-standard/240154037?pgno=1, 2013-05-07

[23] Keccak team official website, https://keccak.team/

이 선 근(Seon-Keun Lee) [정회원]

- 1997년 8월 : 원광대학교 전자공학과 (공학석사)
- 2003년 2월 : 원광대학교 전자공학과 (공학박사)
- 2006년 4월 ~ 2008년 2월 : 원광대학교 전자공학과 교수
- 2017년 3월 ~ 2020년 2월 : 전북대학교 기계시스템공학부 강의전담교수
- 2020년 3월 ~ 현재 : 우석대학교 전기자동차공학부 교수

〈관심분야〉
IoT, 임베디드시스템, H/W 암호시스템, 프로세서설계