논문 15-40-10-14 The Journal of Korean Institute of Communications and Information Sciences '15-10 Vol.40 No.10
http://dx.doi.org/10.7840/kics.2015.40.10.1969

해시 트리 기반의 경량화된 DTLS 메시지 인증

이 부 형◆, 이 성 범*, 문 지 연**, 이 종 혁°

Lightweight DTLS Message Authentication Based on a Hash Tree

Boo-Hyung Lee◆, Sung-Bum Lee*, Ji-Yeon Moon**, Jong-Hyouk Lee°

요 약

제한된 자원을 가지는 장비들이 서로 통신을 하는 사물인터넷 환경에서는 경량화된 보안 프로토콜이 요구된다. 본 논문은 DTLS 프로토콜의 메시지 인증 경량화를 위해 해시 트리를 이용한 새로운 메시지 인증 기법을 제안한다. 제안된 기법은 DTLS 프로토콜의 기본 동작에 비해 경량화된 보안 동작을 제공하며, 통신과정 중에서도 불필요한 메시지 인증 코드의 사용을 줄여 사물인터넷 환경에 더욱 적합한 성능을 나타낸다.

Key Words : Hash Tree, DTLS, Authentication, Network, Lightweight

ABSTRACT

The Internet of Things (IoT), in which resource constrained devices communicate with each other, requires a lightweight security protocol. In this paper, we propose a new message authentication scheme using a hash tree for lightweight message authentication in the Datagram Transport Layer Security (DTLS) protocol. The proposed scheme provides lightweight secure operations compared with those of the DTLS protocol. Besides, it provides more suitable performance than the DTLS protocol for an IoT environment, thanks to the reduced use of message authentication code.

Ⅰ. 서 론

사물인터넷 시대[1], 각종 센서와 기기들에 의해 생성되는 엄청난 양의 데이터는 안전한 전송, 저장 및 이용을 필요로 한다. 이러한 데이터를 이용하는 대부분의 장치들은 블루투스, BLE(Bluetooth Low Energy), 6LoWPAN 등을 포함하는 다수의 연결성 채널들을 포함하게 된다. 따라서, 기존의 인터넷 환경과 마찬가지로 도래하게 될 사물인터넷 환경에서도 무결성, 기밀성, 가용성 등이 제공되어야 한다. 특히, 메시지 인증(전송 되는 메시지에 대한 무결성을 통한 데이터 인증)은 전송 되는 데이터에 대한 신뢰성을 제공한다. 이러한 메시지 인증을 통해 프라이버시 침해나 악의적인 의도의 데이터 위변조, 비인가된 접근으로 인한 금전적인 피해나 인명 피해를 줄일 수 있다.

본 논문에서는 해시 트리(Hash Tree)를 응용한 새로운 메시지 인증 기법을 제안한다. 사물인터넷 환경에서의 메시지 인증 기법으로 DTLS(Datagram Transport Layer Security) 프로토콜이 가장 많이 제안되고 있는데, DTLS 프로토콜에서 메시지 인증 방

※ 본 연구은 2014년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2014R1A1A1006770).
◆ First Author : Dept. of Computer Science & Engineering, Sangmyung University, boohyung@pel.smuc.ac.kr, 학생회원
° Corresponding Author : Dept. of Computer Science & Engineering, Sangmyung University, jonghyouk@pel.smuc.ac.kr, 정회원
* Dept. of Computer Science & Engineering, Sangmyung University, sungbum@pel.smuc.ac.kr, 학생회원
** Dept. of Computer Science & Engineering, Sangmyung University, jiyeon@pel.smuc.ac.kr, 학생회원
논문번호 : KICS2015-06-185, Received June 12, 2015; Revised September 4, 2015; Accepted October 15, 2015

법은 TLS를 기반으로 만들어졌기 때문에 반드시 경량화가 필요하다. HMAC을 이용하는 DTLS와는 달리 제안하는 기법은 Top Hash를 이용하여 수신한 메시지를 인증하기 때문에 단편화된 메시지 블록의 개수와 상관없이 하나의 인증자만을 사용한다. 이는 데이터 위변조와 인증자를 이용한 공격에 안전하며, 통신 과정과 동작 과정에서 생기는 비용을 줄이는 효과가 있다.

본 논문의 2장에서는 DTLS의 메시지 인증 과정과 Hash Tree에 대하여 기술한다. 3장에서는 해시 트리를 이용한 제안하는 메시지 인증 기법을 소개한다. 제 4장은 성능분석 결과를 제시하며, 제 5장에서 본 논문을 결론짓는다.

Ⅱ. 관련 연구

2.1 DTLS

DTLS[2,3]는 통신계층에서 보안성을 제공하는 TLS[4]를 경량화하여 만든 프로토콜로써, TLS를 UDP에 적용가능하게 해주는 UDP를 위한 보안 프로토콜이라 할 수 있다. 컴퓨팅 자원이 부족한 각종 센서나 기기와의 통신에서 무결성을 제공하고 상호인증 및 암호화 통신을 제공하기 위해 IETF에서 표준화되었다. DTLS는 다음과 같이 크게 2가지 단계로 이루어진다.

- Handshake 과정: 암호화 통신에 필요한 인자들을 노드 간 협상하고 상호인증 제공
- Record 과정: 데이터 단편화 및 전송 메시지에 대한 무결성과 기밀성 제공

표 1은 본 논문에서 사용되는 표기이다.

그림 1은 DTLS에서의 Handshake 과정을 나타낸다. 연결된 두 장치를 각각 클라이언트와 서버라고 할 때, 클라이언트는 서버에게 자신이 생성한 난수와 지원하는 알고리즘 목록을 전송한다. 서버는 자신이 생성한 난수와 자신이 선택한 알고리즘 목록과 함께 인증서나 서명을 이용하여 자신의 공개키를 클라이언트에게 전송한다. 클라이언트는 서버의 인증서나 서명을 검증하고 PreMaster Secret을 서버의 공개키로 암호화하여 서버에게 전송한다. PreMaster Secret은 클라이언트가 생성하는데, 이전 과정에서 교환한 서로의 난수와 함께 Master Secret을 만들 때 사용된다. Master Secret은 암호화에 쓰이는 Ks와 IV, 메시지 인증에 쓰이는 K_{HMAC}를 만드는 데 사용된다. Ks와 IV, K_{HMAC}의 생성이 끝나면 서로 지원하는 알고리즘 목록 중 실제로 사용할 알고리즘을 협상한다. 이 때, 협상하는 알고리즘은 암호화 알고리즘, HMAC에서 쓰이는 해시 함수, 압축 알고리즘이다. 따라서, Handshake 과정이 끝나면 두 장비는 서로의 공개키와 IV, K_{HMAC}, Ks를 가지게 된다.

DTLS의 Handshake 과정을 통해 통신 노드들은

표 1. 표기 정리
Table 1. Notations used in this paper

Notation	Definition
Ks	Session key(Server-Client)
IV	Initialization Vector using CBC
K_{HMAC}	Shared Key using HMAC
PUs	Public Key(Server)
PRs	Private Key(Server)
PUc	Public Key(Client)
PRc	Private Key(Client)
TH	Top Hash value generated by Client
TH'	Top Hash value generated by Server
$M(B_i)$	The i-th data block of n-data blocks ($i \leq n$)
$HMAC(M(B_i))$	Concatenated MAC to transmitted i-th data block of n-data blocks ($i \leq n$)
$HMAC'(M(Bi))$	Concatenated MAC to received i-th data block of n-data blocks ($i \leq n$)

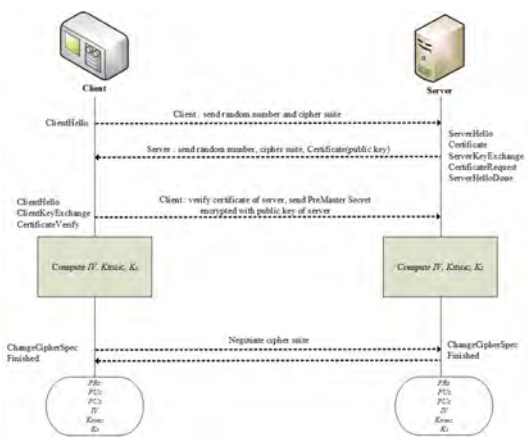

그림 1. DTLS에서의 Handshake 과정
Fig. 1. Handshake procedure in DTLS

서로의 공개키, 메시지 인증을 위한 K_{HMAC}, 데이터 암호화를 위한 관용 세션 키 등을 협상하며, 실제 데이터 전송은 Record 과정을 통해 이루어진다.

그림 2는 DTLS Record 과정에서의 메시지 인증을 나타낸다. 그림과 같이 Record 과정에서 송신 측은 데이터를 MTU 크기에 따라 단편화하고 압축한다. 단편화된 데이터에 메시지 인증을 제공하기 위해, HMAC을 이용해 MAC(Message Authentication Code)을 생성하고 *Ks*로 MAC을 포함한 메시지를 암호화하여 전송한다.

HMAC을 이용한 MAC 생성에서, K_{HMAC}과 *ipad*, *opad*가 사용되며, 단편화된 데이터 블록마다 해시 연산이 2번 실행된다. 즉, 송신될 데이터가 n개의 블록으로 단편화 될 경우, *n* 개의 MAC이 송신 측에서 생성되고 전송되어야 한다. 수신 측에서도 *n* 개의 블록에 대해 각각 전송된 MAC에 대해 메시지 인증을 확인해야 하는 과정이 요구된다.

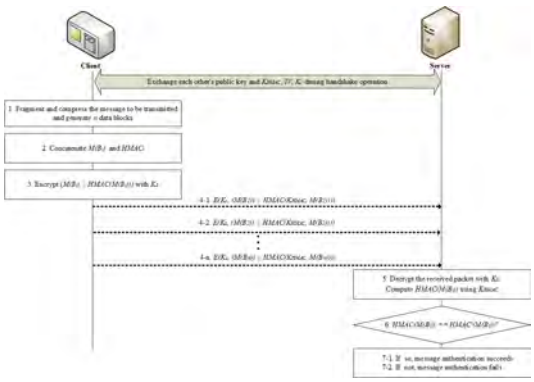

그림 2. DTLS의 메시지 인증
Fig. 2. Message authentication in DTLS

2.2 메시지 인증

데이터의 무결성을 제공하는 메시지 인증 방법에는 앞서 설명한 DTLS에서 사용하는 HMAC과 전자서명 등이 있다.

- HMAC(Hash-based Message Authentication Code) : 일방향 해시 함수를 이용하여 메시지 인증 코드를 구성하는 방법이다. 일방향 해시 함수로는 SHA1과 MD5가 많이 사용 된다. 송신자와 수신자가 공유하고 있는 키와 메시지를 혼합하여 해시 값을 만들고 만든 해시 값을 메시지 인증 코드로 사용한다.
- 전자서명(Digital Signature) : 전자서명은 문서나 메시지를 보낸 사람의 신원이 진짜임을 증명하기 위해 사용되는 서명이다. 전자서명은 위조가 불가능하고 타임스탬프가 자동으로 유지되며 부인방지 기능도 제공한다.

그러나 사물인터넷 환경에서는 기기의 하드웨어적인 제약 때문에, 많은 계산량이 요구되는 전자서명은 부적합하다. HMAC의 경우에는 20byte의 MAC을 사용하는데, 데이터의 크기가 작은 경우 HMAC 자체가 오버헤드가 될 수 있다.[5]

2.3 Hash Tree

Hash Tree[6]는 1979년 Ralph C. Merkle이 제안한 기법으로, 각각의 속성들의 해시로 구성된 트리 구조를 말한다. 자식 노드는 각 데이터 블록의 해시 값을 가지고, 부모 노드는 각 자식 노드들의 해시 값을 가진다. 여기서 트리의 루트 노드를 Top Hash라고 한다. 이렇게 생성된 Top Hash를 이용하여 메시지를 검증한다. Top Hash를 생성하기 위해 사용하는 해시 함수는 SHA1[7], Tiger, WHIRLPOOL 등이 있다. 본 논문에서는 제안 기법을 위해 출력 값이 160bit인 SHA1을 사용한다.

Hash Tree를 이용한 데이터 검증은 Top Hash 값 검증으로 이루어진다. 신뢰할 수 있는 Top Hash 값을 사용자가 가지고 있다면, 전체 트리를 사용하지 않고도 특정 부분만으로 노드들의 무결성을 검증할 수 있다. 예를 들어 그림 3의 Data block 2를 검증하려면, Hash 0-0과 Hash 0-1으로 만들 수 있는 Hash 0와 Hash 1값을 통해 계산된 Top Hash를 신뢰할 수 있는 Top Hash와 비교하여 데이터 블록의 무결성을 검증

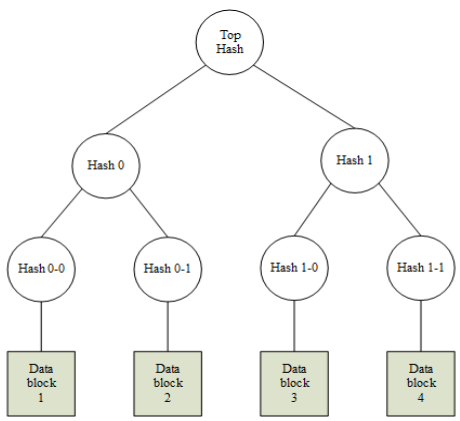

그림 3. Hash Tree의 구성 예
Fig. 3. An example of a Hash Tree

할 수 있다. 따라서 많은 데이터 블록이 존재하더라도 특정한 데이터 블록 검증이 가능하다.

본 논문에서는 메시지 인증 기법으로 DTLS에서 사용하는 HMAC을 대신해 Hash Tree를 사용함으로써 동작과 통신과정의 오버헤드를 줄이고자 한다.

Ⅲ. 제안하는 기법

이 장에서는 제안하는 기법을 상세히 소개하고, 제안하는 기법이 가지는 보안성에 대해 설명한다.

3.1 제안하는 기법

제안하는 기법은 DTLS Handshake 과정 이후에 상호 인증이 완료되고, 클라이언트와 서버 간에 암호화된 안전한 채널이 수립된 후 수행된다.

Handshake 과정이 종료되면 클라이언트와 서버는 서로의 공개키를 교환하고, *IV*와 K_{HMAC}, 암호화에 쓰이는 관용 세션 키 *Ks*를 가진다. 그림 4는 제안하는 기법의 메시지 인증 과정을 나타낸다.

제안하는 기법은 총 7개의 과정으로 이루어지며, 그 절차는 아래와 같다.

1. 클라이언트는 송신할 데이터를 단편화하고, 단편화한 데이터 블록을 해시 트리의 속성값으로 하여 해시 트리를 구성한다. 구성한 해시 트리를 이용해 *TH*를 계산한다. *TH*는 송신할 데이터 블록 단편의 해시 값을 연접한 값이다. 수신자는 여기서 계산한 *TH*를 이용하여 메시지를 인증한다.

2. 데이터 블록들을 관용 세션 키 *Ks*로 암호화하고, 1번 과정에서 계산한 *TH*는 서버의 공개키 *PUs*로 암호화한다. *TH*를 서버의 공개키 *PUs*로 암호화하는 이유는 서버만이 자신의 개인키를 이용하여 *TH*를 복호화할 수 있게 하기 위해서이다.

3. 관용 세션 키 *Ks*로 암호화한 데이터 블록들과 서버의 공개키 *PUs*로 암호화한 1개의 *TH*를 서버에 전송한다. 암호화한 *TH*는 단편화한 데이터 블록들 중 마지막 데이터 블록에 연접하여 전송한다.

4. 서버는 수신한 데이터 블록들을 각각 관용 세션 키 *Ks*로 복호화한다. 또한, *PUs*로 암호화된 *TH*를 서버의 개인키 *PRs*로 복호화한다.

5. 복호화한 데이터 블록들을 해시 트리의 속성값으로 하여 해시 트리를 구성하고, 구성한 해시 트리를 이용하여 *TH'*를 계산한다. *TH'*는 수신한 데이터 블록 단편의 해시 값을 연접한 값이다.

6. 4번 과정에서 복호화한 *TH*와 5번 과정에서 새로 계산한 *TH'*를 서로 비교한다.

7. 비교 결과, *TH*와 *TH'*가 서로 일치하면 메시지 인증 성공, 불일치하면 메시지 인증 실패로 판단한다.

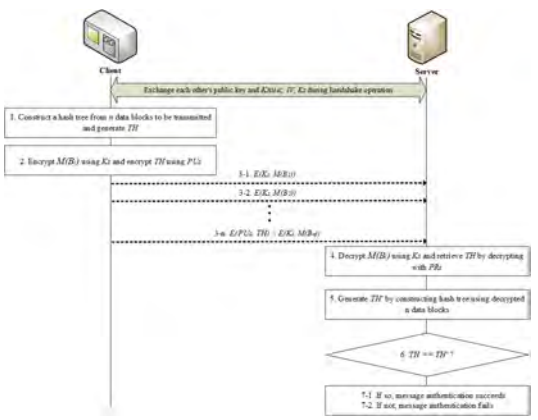

그림 4. 제안하는 메시지 인증 과정
Fig. 4. Proposed message authentication procedure

본 논문에서 제안하는 기법은 기존의 DTLS를 이용한 메시지 인증 기법보다 간편하다. 기존 기법은 데이터를 단편화하고, 단편화한 데이터 블록에 대하여 각각 HMAC을 사용하여 데이터를 인증한다. HMAC을 사용한 메시지 인증 코드를 구성하는 기법에서 K_{HMAC}과 *ipad, opad*를 이용하고, 단편화된 데이터 블록마다 해시 연산이 총 2번 실행된다. 만약 송신할 데이터가 *n*개의 블록으로 단편화된 경우 *n*개의 MAC이 전송된다. 그러나 제안 기법을 사용하면 K_{HMAC}, *ipad, opad*를 사용할 필요가 없다. 만약, 송신할 데이터가 *n*개의 블록으로 단편화된 경우에도 MAC은 1개만 전송된다. 이 때, 사용되는 MAC은 Top Hash이다. 메시지를 송신하기 전에 데이터 블록으로 클라이언트가 Top Hash를 만들고, 메시지를 수신한 후에 서버가 수신한 데이터 블록으로 Top Hash를 만든다. 클라이언트가 만든 Top Hash와 서버가 만든 Top Hash를 서로 비교하여 메시지 인증을 시도한다.

따라서, 제안 기법을 이용한 메시지 인증은 1개의 MAC만을 만들기 때문에, 단편화한 데이터 블록 수만큼 MAC을 만드는 과정에 필요한 오버헤드를 감소시킨다.

3.2 보안성 분석[8]

공격자는 전송되는 메시지에 대한 도청, 위변조 등의 악의적인 공격 및 프라이버시 침해에 관한 위협을 할 수 있다. 이 절에서는 제안하는 메시지 인증 방법

이 보안적으로 어떠한 공격에 대응할 수 있는지 분석한다.

1. 인증자 위조 공격

공격자는 메시지 인증을 위해 존재하는 인증자 *TH*를 위조할 수 없다. 만약, 인증자 *TH*를 위조하려면 데이터 블록의 해시 값이 필요한데 공격자가 서버와 클라이언트 간에 미리 협상한 관용 세션 키 *Ks*를 획득하지 못한다면 데이터 블록을 복호화할 수 없어 *TH*도 생성할 수 없다. 만약, 서버와 클라이언트가 서로 통신하는 중에 공격자가 통신내용을 도청하여 서버의 공개키로 암호화된 *TH*를 얻었다고 해도 서버의 개인키가 안전하다면 공개키 암호화 방식으로 암호화된 *TH*는 복호화할 수 없다.

2. 데이터 위변조

서버는 제안하는 기법을 사용하여 클라이언트로부터 받은 데이터 블록에 대한 무결성을 검증할 수 있다. 제안하는 기법은 해시 트리를 이용한 *TH*로 메시지를 인증하기 때문에, 만약 무결성이 깨짐이 확인되면 클라이언트로부터 무결성에 위반한 데이터 블록에 대해 재전송 요청을 할 수 있다.

3. 인증자 재사용 공격

제안하는 기법을 사용하면 메시지 인증에 쓰이는 인증자 *TH*는 클라이언트에 의해 전송할 데이터 블록에 의해 송신 전에 한번 생성되고, 서버가 수신한 메시지를 인증할 때 인증자 *TH'* 를 생성한다. 만약, 공격자가 도청을 통해 인증자 *TH*를 얻었다고 해도, 인증자 *TH*는 다시 사용하지 않기 때문에 인증자 재사용 공격을 방어할 수 있다.

Ⅳ. 성능 분석

4.1 동작 비용 분석

이 절에서는 DTLS와 제안된 기법이 작동하는 과정에서 동작되는 요소들로 인해 생기는 동작 비용을 표 1에서 비교 분석하여 나타낸다. *n*은 단편화된 데이터 블록의 개수이다.

- Encryption, Decryption : AES_256_CBC 암호화 알고리즘을 사용한다.
- Encryption, Decryption(Public Key) : Top Hash를 전달할 때 공개키 암호 방식을 사용한다.
- SHA1(•) : DTLS는 HMAC을 사용하기 때문에 *n* * 2번의 일방향 해시 함수 SHA1이 실행되고, 제안기법은 Hash Tree에서 2진 트리를 사용하기 때문에 (*n* * 2) - 1번의 SHA1 연산이 실행된다.

표 2. 동작 비용 비교 분석
Table 2. Analysis of operation cost

	DTLS		Proposed scheme	
	client	server	client	server
Encryption	*n*	-	*n*	-
Decryption	-	*n*	-	*n*
Encryption (Public Key)	-	-	1	-
Decryption (Private Key)	-	-	-	1
SHA1(•)	*n**2	*n**2	(*n**2)-1	(*n**2)-1
Hash Function Block Processing	*n*+2	*n*+2	*n*	*n*
XOR	*n**2	*n**2	-	-

- Hash Function Block Processing : 해시 함수에 따라 블록이 정해지고, 단편화된 데이터를 블록의 크기로 나눈 뒤 각각의 블록 별로 메시지 스케줄 및 라운드처리를 하게 된다. 본 논문에서는 이를 Block Processing이라고 표현한다. DTLS에서 사용하는 HMAC의 경우는 데이터 앞에 *ipad XOR key*, *opad XOR key* 연산한 값을 이용해 추가적으로 Block Processing을 하여야 한다.
- *XOR* : DTLS에서 HMAC은 *ipad XOR key, opad XOR key* 연산을 한 뒤, 해시 함수를 실행한다.[9]

4.2 통신 비용 분석

저속도 무선 개인 통신망을 위한 표준 802.15.4[10,11]에서는 MTU의 크기를 최소 1,280 bytes로 정하고 있어서, 데이터는 1,280 bytes로 단편화하여 전송한다.

DTLS에서 메시지 인증을 위한 기법은 단편화된 데이터마다 MAC을 포함하므로, 단편화된 패킷은 헤더(103 bytes), 페이로드(1,155 bytes) MAC(20 bytes), 트레일러(2 bytes)로 구성된다. 제안된 기법은 메시지 인증을 위해 전체 데이터에 대해 Top Hash(20 bytes)를 생성하고, 단 하나의 Top Hash만 전송한다. 즉, 단편화된 데이터마다 메시지 인증을 위한 MAC과 같은 정보가 불필요하다. 따라서 제안 기법을 이용하는 경우 단편화된 패킷은 헤더(103 bytes), 페이로드(1,175 bytes), 트레일러(2 bytes)로 구성된다. 그림 5는 제안하는 기법을 사용할 때와 DTLS의 메시지 인증 기법을 사용할 때의 Payload 크기를 비교한 그림이다.

제안 기법을 사용하면 데이터를 단편화함에 따라 사용하는 MAC의 수 차이가 발생한다. 그림 6은 단편

화된 데이터 개수에 따른 MAC의 수를 비교한 것이다. 제안된 기법은 DTLS와 달리 단편화된 데이터마다 MAC을 붙이지 않기 때문에 추가적으로 20 bytes의 Payload를 가질 수 있다. 따라서 같은 크기의 데이터를 보낸다고 할 때, 제안된 기법을 사용하면 n이 증가함에 따라 DTLS보다 적은 수의 패킷으로 전송이 가능하다.

그림 7은 제안 기법을 사용할 때와 비교하여 DTLS를 사용할 때, 송신할 데이터가 늘어남에 따라 더 사용해야 하는 패킷의 수를 의미한다.

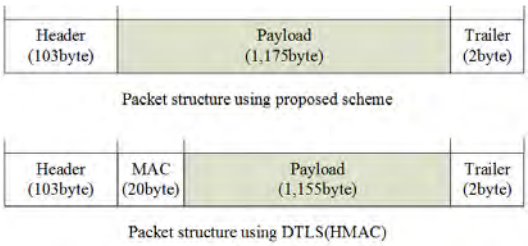

그림 5. Payload 크기 비교
Fig. 5. Comparison of payload

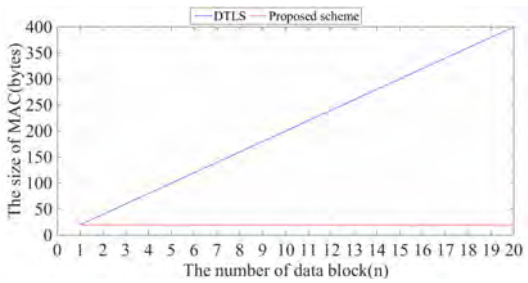

그림 6. 단편화된 데이터 개수에 따른 MAC의 크기 비교
Fig. 6. Comparison of the size of MAC according to fragmented data

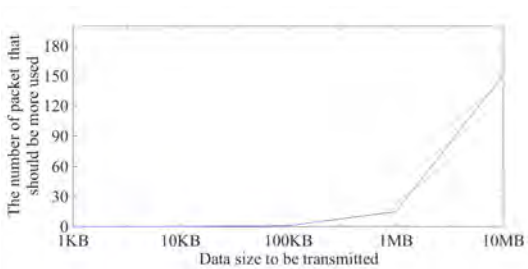

그림 7. 송신할 데이터에 따라 더 사용해야 하는 패킷의 수 (DTLS를 사용할 때)
Fig. 7. The number of packet more that should be used according to be transmitted data(using DTLS)

V. 결 론

본 논문은 DTLS에 적용될 수 있는 해시 트리 기반의 새로운 메시지 인증을 제안하였다. 제안된 기법은 해시 트리의 특성을 이용해 동작과정과 통신과정에서의 오버헤드를 줄이는 경량화된 메시지 인증을 제공한다. 적은 자원을 사용하는 사물인터넷 환경에서, TLS 기반으로 만들어진 DTLS를 대신할 수 있는 메시지 인증 기법이 될 수 있다. 제안하는 기법은 전송하는 패킷의 수가 많은 경우, 예를 들어 홈 오토메이션이나 빌딩 오토메이션 분야에서 무인 감시 카메라가 전송하는 영상 데이터의 메시지 인증에 효율적으로 사용할 수 있다.

하지만, 공격자가 만약 통신 채널에 접근이 가능하다면 수집한 TH중 임의의 TH를 수신자에게 보내 고의로 메시지 인증을 방해할 수 있으므로 차후에 이에 대한 보안 연구가 필요하다.

References

[1] J. Park and N. Kang, "Entity authentication scheme for secure WEB of things applications," *J. KICS*, vol. 38B, no. 05, pp. 394-400, May 2013.

[2] E. Rescorla and N. Modadugu, *Datagram Transport Layer Security Version 1.2*, IETF RFC 6347, Jan. 2012.

[3] J. Park, S. Shin, and N. Kang, "Mutual authentication and key agreement scheme between lightweight devices in internet of things," *J. KICS*, vol. 38B, no. 09, pp. 707-714, Sept. 2013.

[4] T. Dierks and E. Rescorla, *The Transport Layer Security Protocol Version 1.2*, IETF RFC 5246, Aug. 2008.

[5] M.-H. Park, C.-K. Lee, J.-H. Son, and S.-W. Seo, "Efficient security mechanism using light-weight data origin authentication in sensor network," *J. KICS*, vol. 32, no. 05, pp. 402-408, May 2007.

[6] Y. Hu, A. Perrig, and D. B. Johnson, "Efficient security mechanisms for routing protocols," in *Proc. NDSS'03*, Feb. 2003.

[7] D. Eastlake and P. Jones, *US Secure Hash Algorithm 1(SHA1)*, IETF RFC 3174, Sept.

2001.

[8] F. Dötzer, “Privacy issues in vehicular ad hoc networks,” in *Proc. 2nd ACM Int. Workshop Veh. Ad Hoc Netw.*, Sept. 2005.

[9] H. Krawczyk, M. Bellare, and R. Canetti, *HMAC: Keyed-Hashing for Message Authentication*, IETF RFC 2104, Feb 1997.

[10] I. Ishaq, D. Carels, G. Teklemariam, J. Hoebeke, F. Abeele, E. Poorter, I. Moerman, and P. Demeester, “IETF standardization in the field of the internet of things (IoT): A survey,” *J. Sensor and Actuator Networks*, vol. 2, no. 2, pp. 235-287, Apr. 2013.

[11] H. R. Lee, K.-H. Jung, and Y.-J. Suh, “Contention/Collision mitigation scheme in IEEE 802.15.4 mesh sensor networks,” *J. KICS*, vo1. 38C, no. 08, pp. 683-691, Sept. 2013.

이 부 형 (Boo-Hyung Lee)

2009년 3월~현재 : 상명대학교 컴퓨터소프트웨어공학과 재학

<관심분야> 사물인터넷, 네트워크 보안, 시스템 보안

이 성 범 (Sung-Bum Lee)

2010년 3월~현재 : 상명대학교 컴퓨터소프트웨어공학과 재학

<관심분야> 네트워크 보안, 시스템 보안

문 지 연 (Ji-Yeon Moon)

2013년 3월~현재 : 상명대학교 컴퓨터소프트웨어공학과 재학

<관심분야> 사물인터넷, 네트워크 보안

이 종 혁 (Jong-Hyouk Lee)

2010년 2월 : 성균관대학교 공학박사

2009년 6월~2012년 2월 : 프랑스 INRIA 연구원

2012년 3월~2013년 8월 : 프랑스 그랑제꼴 TELECOM Bretagne 조교수

2013년 9월~현재 : 상명대학교 컴퓨터공학과 조교수

<관심분야> IP 이동성, 보안, 프라이버시 보호