한국정보전자통신기술학회논문지(jkiiect)'17-02, Vol.10 No.1
논문 17-10-01-66 http://dx.doi.org/10.17661/jkiiect.2017.10.1.66

IoT 통신 환경을 위한 경량 IKEv2 프로토콜 연구

김홍성*, 송인아**, 이영석***

A Study on Lightweight IKEv2 protocol for IoT communication environments

Hong-Sung Kim*, In-A Song**, Young-Seok Lee***

요 약 IoT 통신 환경이 구축되면서 고사양 기기뿐만 아니라 저사양기기를 사용하는 통신이 증가하였지만 그에 따른 보안 위협도 증가하고 있다. 최근 저사양 기기에 적용할 수 있는 효율적인 보안 기술을 위해 기존 IP 계층에서 쓰이고 있는 보안 기술인 IPsec와 IKEv2의 경량화 시도가 이루어 지고 있으며 대표적으로 Smyslov는 IoT 환경에서 사용 가능할 수 있도록 경량화 IKEv2 프로토콜을 제안하였다. 하지만 이 제안 방법은 기존 IKEv2와 같이 사용하기엔 호환성 문제가 발생하여 IoT 통신에서의 보안성 향상을 기대하기 힘들다. 본 논문에서는 Smyslov의 제안 방법과는 달리 기존 IKEv2와 호환이 가능하고 페이로드에 무손실 압축 알고리즘을 적용한 효율적인 경량 IKEv2 프로토콜을 제안한다. 제안 경량 IKEv2 프로토콜 연구를 위하여 IKEv2와 기존 경량화 IKEv2 프로토콜을 분석하며 성능 평가를 통하여 제안 프로토콜이 기존 경량 IKEv2 프로토콜에 비해 효율적임을 증명하였다.

Abstract As the IoT communication environment has been established, communications that utilize not only high-spec machines but also low-spec machines are increasing, but security threats are increasing, too. In recent times, a lot of papers have attempted to reduce the weight of IP layer security techniques such as IPsec and IKEv2 for low-spec machines. Typically, Smyslov proposed Lightweight IKEv2 protocol which is used in IoT environment. However, This proposed protocol had compatibility problem with IKEv2 protocol, So, It is hard to be expected to be used in IoT communication environment. Unlike the Smyslov's protocol, this paper proposed Lightweight IKEv2 protocol which can be compatible of IKEv2 protocol and applied lossless compression algorithm to payload. To suggest lightweight IKEv2 protocol, this paper analyzed IKEv2 protocol and existed lightweight IKEv2 protocol. Furthermore, This paper proved that proposed protocol is more efficient than existed lightweight IKEv2 protocol through performance evaluation as a method.

Key Words : IKEv2, IPsec, Internet Key Exchange Protocol, Lightweight, IoT

1. 서 론

IoT 통신 환경이 구축되면서 고사양 기기뿐만 아니라 저사양기기를 사용하는 통신이 증가면서 그에 따른 보안 위협도 증가하고 있다.

보안성을 갖추기 위해서는 메시지 암호화와 인증을 제공하는 블록 암호 알고리즘과 메시지 인증 코드가 필수적으로 사용되어야 하지만 기존 암호 기술을 통신, 계산 기능이 제약된 저사양 기기에 그대로 사용하기에는 어려움이 따른다. 이러한 상

This Paper was performed with the support ICT development projects promising technologies of information and communication technology promotion center.(No.R6910-15-11-2)
*National Security Research Institute
**Dept. of Information & Telecommunication Engineering, Kunsan National University
***Corresponding Author : Dept. of Information & Telecommunication Engineering, Kunsan National University (leeys@kunsan.ac.kr)
Received January 26, 2017 Revised February 02, 2017 Accepted February 10, 2017

황에서 보안 기술 적용을 위해 IP 계층에서 쓰이고 있는 IPsec과 IKEv2의 경량화 시도가 이루어지고 있다.

Smyslov는 IoT 환경에서 사용 가능할 수 있도록 경량화 IKEv2 프로토콜을 제안하였다. 하지만 이 제안 방법은 기존 IKEv2와 같이 사용하기엔 호환성에 대한 문제가 있어 보안성 향상을 기대하기 힘들다.

본 논문에서 제안하는 방식은 기존 경량화 IKEv2 제안의 문제점인 호환성을 해결하고 더욱 효율적으로 경량화 할 수 있는 방법에 대해 제안하고자 한다.

본 논문의 2장에서는 기존 IKEv2와 Smyslov가 제안한 경량 IKEv2을 살펴본다. 3장에서는 본 논문에서 제안한 경량 IKEv2을 살펴본다. 4장에서는 구성한 환경을 토대로 성능평가를 진행하고, 마지막으로 5장에서는 결론과 향후 연구 방향을 제시한다.

2. 관련 연구

2.1 IKEv2

IKE(Internet Key Exchange)은 보안연계의 수립을 위하여 인증된 키 자료를 보호된 방식으로 협상하고 제공하는 프로토콜이다. 보안연계를 위해 3개의 서로 다른 프로토콜의 관련 부분을 결합한 하이브리드 프로토콜로서, ISAKMP(Internet Security Associationand Key Management Protocol), Oakley Key Determination Protocol, SKME 로 결합되어 있다.

IKE는 서비스 거부 공격 및 중간자 공격을 방지하며 PFS(Perfect Forward Security)를 제공하도록 설계되었다. IKE는 ISAKMP 협상의 단계에서 동작하는 여러 교환 모드를 제공하며 1단계 교환에서는 ISAKMP 보안연계를 수립하고 2단계에서는 AH, ESP와 같은 다른 보안 서비스를 위한 보안연계를 수립하는데 이용한다.

ISAKMP 교환은 쿠키로 시작하는데 이는 방해 대응용으로 어떤 특정 메시지가 통신으로부터 온 것임을 확인하는 방법으로 사용된다. 만약 수신한 응답자 쿠키 값이 응답자로부터 그 전에 받은 쿠키의 값과 일치하지 않으면 메시지를 폐기하게 된다. 쿠키는 보안연계마다 고유하여야 한다. [1]

IKE 프로토콜은 기능이 복잡하고 이기종 제품 간의 상호 연동성 문제를 가지고 있기 때문에 IETF IPSec Working Group에서는 새로운 키 관리 프로토콜을 개발하였으며, 이는 IKEv2가 표준화가 되었다.

IKEv2에서는 거의 사용되지 않는 옵션을 제거하였지만, 기존의 IKE 프로토콜의 개념을 그대로 계승하면서 그 기능을 축약시켰다. IKEv2에서는 개시자가 보안연계 이루어진 구간에서 보안 게이트웨이에 의해 자체적인 IP주소를 요청하는 매커니즘을 포함하고 있다. 또한 IKEv2는 네트워크 노드들 사이에서 IPsec 보안연계를 자동으로 설정해 주는 기능을 제공한다. 네트워크 노드들 사이의 인증은 AUTH Payload을 계산함으로써 이루어진다.

IKEv2는 일반적으로 3개의 라운드 트랩으로 구성되고 서비스 거부 공격 방지 라운드 트랩을 사용하지 않으면 2개의 라운드 트랩으로 구성된다.[2]

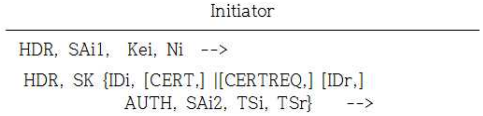

그림 1. IKEv2에서의 개시자 메시지
Figure 1. Initiator message in IKEv2

개시자는 HDR를 이용하여 암호학적 파라미터를 포함하는 Kei를 생성한다. Kei는 IKE_SA와 Diffie-Hellman의 값을 포함한다. 그리고 개시자가 생성한 Nonce를 포함한 IKE_SA_INIT Request 메시지를 응답자에게 전송한다.

Request 메시지를 수신받은 뒤 개시자는 검증하기 위한 자신의 식별자 IDi와 메시지의 무결성을 위한 AUTH Payload를 전송한다. CERT는 인증서이며 이와 관련된 인증기관 리스트와 공개키를 보낸다. 이러한 공개키는 CERTREQ이다.

IP Address 내에서 다중 ID로 서비스 하고 있는 개시자가 선택적으로 응답자를 선택할 수 있도록 기능을 갖고 있는 IDr를 송신한다.

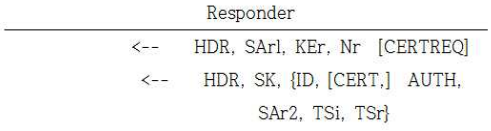

그림 2. IKEv2에서의 응답자 메시지
Figure 2. Responder message in IKEv2

선택적으로 서비스 거부 공격 방지를 위한 라운드 트랩을 이용할 수 있다. 메시지 교환 후 개시자가 제공한 암호학적 파라미터 중에 선택한 것을 포함하는 SAr1과 Diffie-Hellman 키 교환을 완성하기 위한 KEr과 응답자가 생성한 Nonce(Nr)를 포함하는 메시지를 개시자에게 전송한다.

다른 메시지는 응답자가 IDr을 검증하고 CERT 그리고 자신임을 증명하는AUTH Payload를 포함하는 메시지를 전송한다.

CREATE_CHILD_SA 메시지는 양쪽 노드들이 이미 기존에 SA를 가지고 있거나 만료했거나 다른 보안 정책을 가지고 있을 때 새로운 SA를 다시 생성하기 위해 사용된다. 이러한 메시지는 IKE_AUTH메시지에서 인증 부분이 없어진 것과 유사한 메시지이다.

이때 개시자 메시지는 새로운 보안연계와 새로운 키 교환을 위한 KEi을 포함해서 응답자에게 전송한다. 응답자는 개시자가 제공한 보안연계 중 사용 가능한 보안연계와 Diffie-Hellman 값을 포함하는 KEr을 응답 메시지로 전송한다. Information 메시지는 상황에 따라 선택 가능한 메시지이다.

2.2 기존 경량 IKEv2 프로토콜

Smyslov's는 IKEv2의 경량화에 대해 무손실 압축을 통한 IKEv2 메시지를 만들며 IoT환경의 저전력 장치에 적합한 키 교환 프로토콜을 만드는 시도를 하였다.[3]

2.2.1 IKE_SA_INIT Exchange

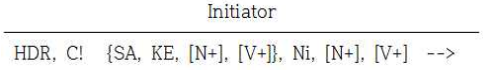

그림 3. 압축된 개시자의 메시지
Figure 3. Using Compression in Initiator Message

개시자가 압축을 원할 경우 압축 알고리즘을 통해 새로운 페이로드를 형성한다. 이 페이로드는 압축된 메시지 뿐 아니라 어떤 종류의 알고리즘을 사용하였는지도 표기한다. 개시자는 페이로드 타입에 적절한 압축 알고리즘을 선택해야 하며, 응답자는 이 압축 알고리즘을 잘못 선택 됐을 경우 개시자에게 이를 알리고 새로운 보안연계를 설정 해야 한다. 이때 Critical 필드의 압축 페이로드의 최 상위 헤더 비트를 1로 해야 한다.

개시자가 만약 INVALID_SYNTAX나 No response를 수신할 경우에는 반드시 압축을 하지 않은 페이로드로 재송신 해야 하며 UNSUPPORTED_CRITICAL_PAYLOAD를 수신할 경우 통지받는 알고리즘 리스트를 통해 새롭게 압축된 IKE_SA_INIT를 생성하여 송신 할 수 있다. 응답자는 수신한 IKE_SA_INIT를 압축을 해제하고 평소와 같이 처리한다.

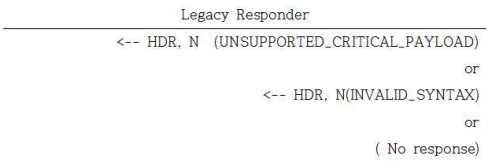

그림 4. 압축을 사용하지 않을 경우의 응답 메시지
Figure 4. Message in not using compression case

만약 [그림 5]와 같이 적절한 알고리즘이 없을 경우에도 압축을 진행하지 않는다.

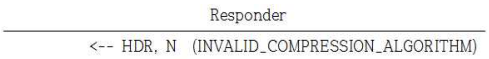

그림 5. 압축을 미지원 할 경우의 응답 메시지
Figure 5. Message in not support compression case

1만약 적절한 알고리즘이 있거나 지원하는 알고리즘이 있을 경우 Responder는 [그림 6]과 같은 메시지를 전송한다.

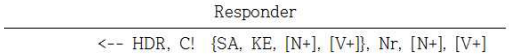

그림 6. 압축이 가능할 경우 응답 메시지
Figure 6. Message in using compression case

INIT Exchange와 Subsequent Exchanges에서 메시지를 교환할 때 C!{payload}의 압축부분으로써 그 형태는 다음과 같다.

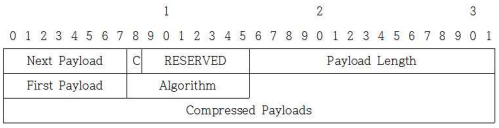

그림 7. 압축된 IKE_SA_INIT의 페이로드
Figure 7. Compressed Payload in IKE_SA_INIT

기존 IKEv2에서 생성하는 Payload 부분과 비교하였을 때 RESERVED 부분과 ID TYPE이 삭제되며, Identification Data는 Compressed Payloads로 변경되었다. 기본 헤더인 Next Payload와 C, Reserved, Payload Length는 기존의 IKEv2와 같다. 각 필드의 설명은 다음과 같다.

(1) Next Payload : 기존 IKEv2와 마찬가지로 메시지의 다음에 오는 페이로드의 타입을 나타낸다.

(2) C (Critical) : Critical을 나타내는 부분으로 반드시 1로 설정되어야 하며 이는 기존 IKEv2와 같다.

(3) RESERVED : 헤더 안에 포함되는 RESERVED 부분으로서 모든 영역이 0으로 셋팅되며 수신자는 이 필드는 확인하였을 때 무시한다.

(4) Payload Length : 헤더부터 페이로드 길이까지 나타내며 기존 IKEv2와 같은 2byte를 할당하게 된다.

(5) First Payload : 압축된 페이로드 필드에 포함된 페이로드의 타입을 나타내며 1byte의 크기를 할당한다.

(6) Algorithm : Compress inner Payload를 위해 사용되는 압축 알고리즘을 정의하고 있으며 총 256개의 값을 할당할 수 있다. 0번은 Reserved, 1번은 IPCOMP_OUI, 2번은 IPCOMP_DEFLATE, 3번은 IPCOMP_LZS, 4번은 IPCOMP_LZJH 이며, 5번부터 240까지는 Unassigned로 할당되지 않는다. 나머지 값인 241번부터 255번까지는 Private use로 개인활용 용도로 사용되게 된다.

표 1. IKEv2 Notification IPCOMP Transforms IDs
Table 1. IKEv2 Notification IPCOMP Transforms IDs

Value	Compression Type	Reference
0	Reserved	RFC7296
1	IPCOMP_OUI	UNSPECIFIED
2	IPCOMP_DEFLATE	RFC2394
3	IPCOMP_LZS	RFC2395
4	IPCOMP_LZJH	RFC3051
5-240	Unssigned	-
241-255	Private Use	RFC7296

(7) Compressed Payloads : 다음 페이로드의 마지막 비트는 0으로 설정되며 실제로 압축된 페이로드를 나타낸다. 길이는 알고리즘에 따라 달라지기 때문에 가변길이이다.

2.2.2 Subsequent Exchanges

IKE_SA_INIT에서 살펴 보았듯이 압축이 지원되거나 안되는 경우가 있으며 Subsequent Exchange 메시지도 압축의 지원 유 · 무에 따라 메시지 타입이 다르게 나타난다.

[그림 8]은 Subsequent Exchange에서 압축이 진행 되었을 경우의 페이로드를 나타낸다. Notification Type Payload 압축에 관련된 내용이며 Next Payload, C, RESERVED, Payload Length, Protocol ID, SPI Size, Notify Message Type은 기존 IKEv2의 Payload 형식 및 헤더와 같다. 하지만 기존에 있는 SPI(Security Parameter

Index)와 Notification Data가 Supported Compression Algorithm으로 변경되었다.

Next Payload	C	RESERVED	Payload Length
Protocol ID(=0)	SPI Size(=0)		Notify Message Type
Supported Compression Algorithms			

그림 8. 압축된 Subsequent의 페이로드
Figure 8. Compressed Payload in Subsequent

Next payload, C, RESERVED, Payload Length는 기존 IKEv2와 같다. IKEv2와 다른점은 다음과 같다.

(1) Protocol ID : Protocol ID는 모두 0으로 지정하여 압축되어 있는 Payload라 인식하여 무시된다.

(2) SPI Size : SPI Size 또한 0으로 지정되며 SPI Size를 나타내진 않는다. 사실상 무의미한 파라미터를 갖고 있다.

(3) Notify Message Type : IANA에 지정된 값이 설정된다. INVALID_COMPRESSION_ALGORITHM에서 할당된 값이 Notify Message Type으로 설정되며 2byte의 크기를 갖는다.

(4) Supported Compression Algorithms : 1byte는 IKEv2-IANA 의해 IKEv2 Notification IPCOMP Transform IDs에서 관련된 압축 알고리즘들의 ID 값들을 나타내게 된다. 이 값은 [표 1]에 나타나 있는 압축 알고리즘이다.

2.2.3 기존 방식 분석

Smyslov's Draft의 큰 문제점은 기존 IKEv2와의 호환이 어렵다는 점이다. Draft에서 호환에 대해 언급을 하지만 자세한 사항까지의 내용을 담고 있지는 않다. 실제로 Draft에서는 압축 IKEv2는 다른 확장된 IKEv2와 대부분 호환된다고 하지만 몇몇 특별처리가 필요한 부분이 있다고 언급하고 있다.

(1) Interaction with IKEv2 Fragmentation

IKEv2 단편화시 페이로드 전체를 압축해야 한다. 압축하기 전에 단편화를 진행할 경우 기존 IKEv2와의 호환이 저하될 수 있다. 또한 단편화의 패킷을 통해 압축된 패킷을 확인해서는 안되며 반대의 경우도 불가능하다고 언급한다.

(2) Interaction with IKEv2 Resumption

IKEv2 Resumption 은 IKE_SA 가 실패할 경우 재시도하는 메커니즘이다. 이 경우 새롭게 정의되는 IKE_SESSION_RESUME은 이전에 IKE_AUTH exchange에서 생성한 정보를 이용하여 IKE_SA를 새롭게 생성한다. 만약 압축을 진행할 경우 Flag 내에 저장이 되어있거나 이전에 협상하였던 알고리즘이 IKE_SA에 있어야 한다.

(3) Interaction with IKEv2 Redirect

IKEv2의 재전송은 응답자가 개시자와 다른 Host에게 재전송하는 것을 의미한다. 재전송은 이미 생성된 IKE_SA와 나중에 생성된 IKE_SA_INIT exchange를 이용하여 2개를 각각 전송하는 것이 가능하다. IKEv2 재전송의 Notification은 IKE_SA_INIT에서 압축된 페이로드와 다르게 다른 형태로 패킷이 존재해야한다. 이러한 이유는 응답자가 압축해제에 따른 추가적인 자원 소모를 줄이면서 재전송을 결정할 수 있도록 하기 위함이다.

위에서 보는 사항과 같이 특별한 부분에서는 메커니즘이 따로 필요하기 때문에 메커니즘에 대응하지 못하는 기존의 IKEv2 경우 호환이 안될 수 있다. 또한 기존 IKEv2와 다르게 단계 내의 단계가 생략되므로 이러한 방법이 적용이 안될 경우 큰 문제점이 발생 할 수 있다.

응답자의 Initial Exchange 메시지의 경우 기존 경량화 IKEv2와는 다르게 인증서의 부분이 필요할 수도 있으며 안에 파라미터의 값들이 달라지게 된다. 이는 파라미터에 따른 매칭 테이블이 존재하지 않으면 값의 의미를 알아내기 힘들며 IPsec을 위한 IKE 가 진행되지 않게 된다.

응답자 뿐만 아니라 Initial Exchange에서의 개시자의 메시지의 경우도 문제점이 발생 할 수 있다. 기존 경량화 IKEv2 에서는 1단계로 진행되지만 실

제로 지금 사용하는 IKEv2의 경우는 2단계로 진행하게 된다.

3. 제안 프로토콜

제안 경량화 IKEv2 프로토콜은 기존 경량화 IKEv2 프로토콜과는 다르게 기존 IKEv2와 호환이 가능하도록 연구를 진행하였다. 기존에 사용하지 않는 RESERVED 부분을 사용안하도록 하여 압축을 증가시켰으며 1절에서 살펴보았던 경량 IPsec과 마찬가지로 Generic Header 부분을 제외한 모든 Payload 부분을 압축하도록 하였다.

3.1 IKEv2 Header의 압축

3.1.1 IKE_SA SPI 필드의 압축

6LoWPAN에서 SPI 부분의 압축 방식을 응용하여 IKE Header의 SPI를 압축진행 하였다. SPI필드 앞에 별도의 2bit의 Index 필드를 만들어 비트 패턴에 따라 SPI 필드의 크기를 가변적으로 나타날 수 있도록 하였다. 압축을 위한 비트 패턴에 따른 SPI의 크기는 다음의 [표 2]에 나타나있다.

표 2. 비트 패턴에 따른 SPI 필드의 크기
Table 2. SPI Field Size by bit pattern

Bit Pattern	SPI Field Size
00	8 bit
01	16 bit
10	24 bit
11	32 bit

경량 IPsec의 SPI 크기와 같기 때문에 32bit를 이용하여 압축을 진행한다. 비트가 00일 경우 8bit의 SPI가 나타나며 나머지 24bit는 삭제된다. 비트가 01일 경우 16bit의 SPI가 나타나며 나머지 16비트는 삭제된다. 비트가 10일 경우 8bit가 삭제되며 비트가 11일 경우 모든 SPI가 나타나게 된다.

3.1.2 Minor Version Field의 삭제

IKEv2의 관련된 RFC 중 RFC 4306에는 MnVer (Minor Version) 필드의 설명에 관해 다음과 같이 기술되어 있다.

'Minor Version(4 bits) - Indicates the minor version of the IKE protocol in use, Implementations based on this version of IKE MUST set the Minor Version to 0. They MUST ignore the minor version number of received messages.'

송신자는 Minor Version을 생성하여도 패킷을 수신한 수신측은 Minor Version을 무시하게 된다. 이러한 방법에 따라 압축 진행을 위해 Minor Version 필드를 삭제한다.

3.1.3 Exchange Type Field의 압축

기존 IKEv2에서는 Exchange Type은 0에서부터 255까지의 값을 갖고 있다. 0에서 33까지의 값은 RESERVED이며 34번은 IKE_SA_INIT, 35번은 IKE_AUTH, 36번은 CREATE_CHILD_SA를 나타내며 나머지 38번부터 255번까지는 RESERVED 부분으로 IANA 지정한 부분과 Private Use에 따라 나누어 진다.

효율적인 압축을 위해 사용하지 않는 0부터 33까지의 RESERVED 부분과 38부터 255까지의 RESERVED TO IANA과 RESERVED FOR PRIVATE USE부분을 삭제하여 필요한 34번, 35번, 36번, 37번을 사용한다. 최종적으로 4가지의 값을 갖게 되며 이는 2bit를 이용하여 충분히 나타낼 수 있게 된다.

3.1.4 Flag Field의 압축

Flags필드는 X, I, V, R, X 순으로 구성된 1byte 필드로 첫 X는 3bit의 Reserved를 나타내며 I의 값은 1 bit의 개시자을 나타낸다. V의 값은 1bit의 IKEv2 의 버전을 나타내며 R의 값은 1bit의 응답을 나타내며 마지막으로 X의 값은 2bit의 Reserved를 나타낸다.

X의 값을 제외한 I, V, R 필드의 값은 IKEv2에서 사용되는 영역이기 때문에 삭제할 수 없으나 Reserved 부분인 X의 값은 IKEv2에서 사용되지

않기 때문에 삭제를 통해서 Flag 부분의 압축이 가능하다.

I, V, R의 표시 방법은 I는 3bit, V는 4bit, R은 5bit로 나타내기 때문에 Reserved 부분을 삭제하여 나타내어도 최대 4bit의 Flag 부분을 사용하여야 한다. 만약 2bit로 압축을 진행할 경우 기존 IKEv2와의 호환이 어려울 수 있기 때문이다.

3.1.5 제안하는 압축 IKEv2 Header

앞의 사항을 적용한 최종 압축 제안 헤더는 개시자 SPI 필드와 응답자 SPI 앞에 Index 필드가 추가되며 Minor Version 필드가 삭제된다. 그리고 Exchange Type 필드와 Flags 필드의 크기가 압축된다. 압축된 IKEv2 Header는 [그림 9]과 같다.

그림 9. 제안하는 압축 IKEv2 헤더
Figure 9. Proposed IKEv2 Header using Compression

3.2 IKEv2 Generic Payload Header의 압축

IKEv2에서는 기본 IKEv2 Header 후 뒤에 나오는 Payload들의 형식을 나타내기 위한 Generic Payload Header가 나타나게 된다. Generic Payload Header는 Payload의 Type과는 무관하게 Payload 앞에 나오는 헤더이다. Generic Payload Header의 압축 및 헤더를 이용한 Payload 압축을 진행한다. [그림 10]은 Generic Payload Header를 나타낸다.

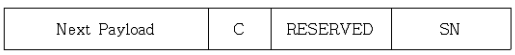

그림 10. 포괄적인 페이로드의 헤더
Figure 10. Generic Payload Header

3.2.1 Reserved 필드의 삭제

Generic Payload Header의 RESERVED 필드는 실질적으로 사용되지 않는 필드이다. 경우에 따라 사용을 할 수도 있지만 IANA에서는 이 부분에 대해 언급한 문서가 없기 때문에 삭제해도 큰 문제가 없다. 따라서 RESERVED 부분을 삭제하여 Header의 크기를 축소하였다.

3.2.2. Next Payload 필드의 압축

기존 IKEv2의 Next Payload 필드는 뒤에 오는 페이로드 타입을 정의하는 필드로 1byte로 구성되어 있다. 하지만 IKEv2에서 실질적으로 사용하는 타입의 수는 No Next Payload를 포함하여 17가지로 한정되어 있다.

Generic Payload Header의 Critical 필드는 뒤에 올 페이로드를 스킵 여부의 나타내는 필드이므로 No Next Payload를 사용할 필요가 없고 1부터 32의 값인 Reserved와 49부터 127까지의 값인 RESERVED IANA 그리고 238부터 255까지의 값인 Private Use는 사용이 제한적이므로 삭제가 가능하다. 따라서 Next Payload 필드에서 4bit를 사용하면 실질적으로 사용하는 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48값을 가지는 16가지의 타입을 모두 정의할 수 있다.

3.2.3 Compression Algorithm Field

Smyslov's Draft에서도 압축 알고리즘 사용하여 1byte 비트 패턴의 압축 알고리즘을 사용한다. 경량화 IPsec에서도 언급한 것처럼 사용하지 않는 알고리즘을 제외한 [표 3]와 같은 알고리즘을 이용하여 Payload 압축을 진행한다.

표 3. 페이로드 압축을 위한 사용 알고리즘
Table 3. Payload Compress Algorithm in Proposed Protocol

Value	Compression Type	Reference
0	No Compression	-
1	IPCOMP_DEFLATE	RFC2394
2	IPCOMP_LZS	RFC2395
3	IPCOMP_LZJH	RFC3051

3.2.4 압축 Generic Payload Header

압축 Generic Payload Header는 앞에서 언급한 사항과 같이 Next Payload의 압축, CA (Compression Algorithm)이 사용된다. [그림 11]는

제안 압축 Generic Payload Header Format을 나타낸다.

Next Payload	CA	C	Payload Length
4 bit	2 bit	1 bit	16 bit

그림 11. 제안하는 포괄적인 페이로드의 헤더
Figure 11. Proposed Generic Payload Header

4. 성능 평가

4.1 성능 평가 환경

IoT 환경을 구성하기 위해 IoT Node와 IoT Gateway 그리고 IPv6 Device로 환경을 구성하였다. IoT Node와 IoT Gateway는 IEEE 802.15.4의 환경을 위해 무선 네트워크를 사용하며 IoT Gateway와 IPv6 Device는 기존 IPv6 네트워크를 사용한다.

그림 12. 성능 평가 환경
Figure 12. Performance Evaluation Environment

IoT Node는 Raspberry Pi를 이용하여 구성하였으며 IoT Gateway와 IPv6의 Device는 일반적인 PC를 사용하여 구성하였다.

통신환경은 IoT Node와 IoT Gateway는 Bluetooth 4.0 위에 6LoWPAN 프로토콜을 이용하여 구성하였으며 거리는 약 3m를 두어 성능평가를 진행하였다. IoT Gateway와 IPv6 Device는 IPv6를 이용한 기존 네트워크 환경을 이용하여 구성하였다.

4.1.1 무선 네트워크 환경

6LoWPAN을 이용한 IEEE 802.15.4 기반의 무선 네트워크 구성을 위해 통신은 Bluetooth 4.0 (BLE 4.0)을 이용하여 구성하였다. Linux 상에서 Bluetooth를 설정하기 위해서는 Bluez 패키지가 필요하며'bluetooth','6lowpan','bluetooth_6lowpan'모듈을 실행해야한다.

4.1.2 유선 네트워크 환경

Gateway에서는 IoT Node에서 전송한 Pakcet을 IPv6 Device에 전송해야 하기 때문에 IPv6 주소를 할당하고 라우팅을 설정해야 한다.

4.1.3 Strongswan

Strongswan은 Linux 상에서 IPsec을 구현할 수 있도록 지원하는 패키지이다. IPsec의 AH, ESP 뿐만 아니라 전송모드 및 터널모드도 설정이 가능하며 IKE 와 IKEv2 프로토콜도 지원이 가능하다.

4.1.4 Zlib

Zlib는 압축 알고리즘을 사용하기 위한 라이브러리이다. 성능 평가를 위해 Compress API를 사용하였으며 압축 해제를 위한 API는 Uncompress를 사용하였다.

4.2 성능 평가

기존 경량화 IKEv2는 단계 생략 및 호환성 문제로 비교 · 분석에 대해 어려움을 갖고 있다. 기존의 IKEv2와 제안 경량화 IKEv2의 단계별 메시지 크기 및 전체 단계의 소요시간을 비교 · 분석하였다.

4.2.1 Initiator Request

처음 Initiator Request의 패킷에 대해 기존 IKEv2와 제안 경량화 IKEv2의 패킷 크기를 확인해보았다.

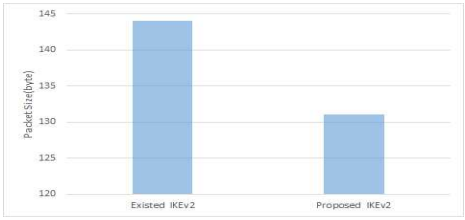

그림 13. 개지사의 요청 패킷의 크기
Figure 13. Initiator Request packet size

기존 IKEv2는 144byte이며 압축을 진행한 제안 경량화 IKEv2는 131byte로 13byte의 크기 차이를 나타냈다. Payload 부분만 살펴 볼 경우 기존 IKEv2는 40byte이며 제안 경량화 IKEv2는 34byte로 압축으로 인해 6byte가 감소되었다.

4.2.2 Responder Reponse

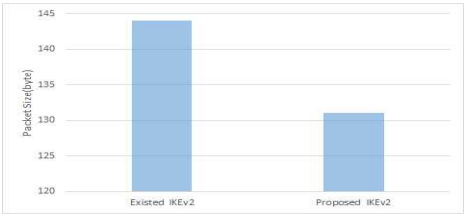

그림 14. 응답자의 응답 패킷의 크기
Figure 14. Responder Response packet size

Initiator Request와 마찬가지로 응답 메시지이기 때문에 기존 IKEv2는 약 144byte, 제안 경량화 IKEv2는 131byte를 나타냈으며 Payload 부분도 IKEv2는 40byte, 제안 경량화 IKEv2는 34byte의 크기로 확인 되었다.

4.2.3 IKE_SA_INIT Initiator Request

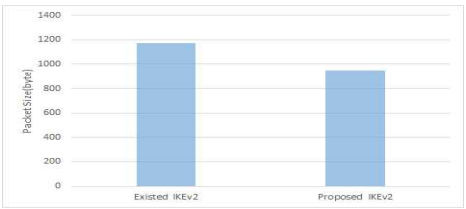

그림 15. IKE_SA_INIT에서의 개시자 요청 패킷의 크기
Figure 15. IKE_SA_INIT Initiator Request packet size

기존 IKEv2의 패킷 크기는 1172byte이며 압축을 진행한 제안 경량화 IKEv2의 패킷 크기 945byte로 227byte의 크기 차이를 나타냈다. Payload 부분만 살펴 볼 경우 기존 IKEv2는 626byte이며 제안 경량화 IKEv2는 406byte로 220byte가 압축이 진행되었다.

4.2.4 IKE_SA_INIT Responder Response

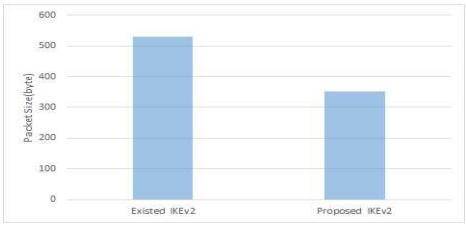

그림 16. IKE_SA_INIT에서의 응답자 응답 패킷의 크기
Figure 16. IKE_SA_INIT Responder Response packet size

기존 IKEv2의 패킷 크기는 529byte이며 압축을 진행한 제안 경량화 IKEv2의 패킷 크기 352byte로 177byte의 크기 차이를 나타냈다. Payload 부분만 살펴 볼 경우 기존 IKEv2는 385byte이며 제안 경량화 IKEv2는 215byte로 압축으로 인해 170byte가 감소되었다.

4.2.5 IKE_AUTH Initiator Request

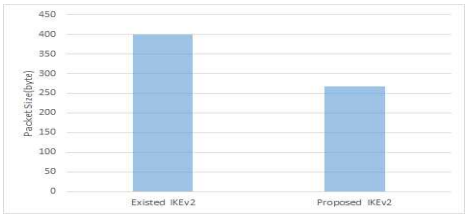

그림 17. IKE_AUTH에서의 개시자 요청 패킷의 크기
Figure 17. IKE_AUTH Initiator Request packet size

기존 IKEv2의 패킷 크기는 400byte이며 압축을 진행한 제안 경량화 IKEv2의 패킷 크기 267byte로 133byte의 크기 차이를 나타냈다. Payload 부분만 살펴 볼 경우 기존 IKEv2는 296byte이며 제안 경량화 IKEv2는 170byte로 압축

으로 인해 126byte가 감소되었다.

4.2.6 IKE_AUTH Responder Response

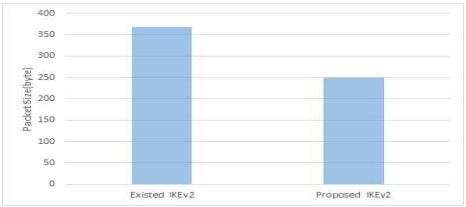

그림 18. IKE_AUTH에서의 응답자 응답 패킷의 크기
Figure 18. IKE_AUTH Responder Response packet size

기존 IKEv2의 패킷 크기는 368byte이며 압축을 진행한 제안 경량화 IKEv2의 패킷 크기 249byte로 119byte의 크기 차이를 나타냈다.

5.2.1 ~ 5.2.6 까지의 패킷 크기 비교를 정리하면 다음 [표 4]과 같다.

표 4. 패킷의 사이즈 비교 및 압축율 비교
Table 4. Comparison of size and compression rate

	Packet			Payload		
	Before	After	Rate	Before	After	Rate
1	144	131	9	40	34	15
2	144	131	9	40	34	15
3	1172	945	19.3	626	406	35.1
4	529	352	33.4	385	215	44..1
5	400	267	33.25	296	170	42.5
6	368	249	32.3	264	152	42.4

5. 결 론

IoT 통신 환경이 구축되면서 IoT 보안성을 강화하기 위해 IPsec을 연계하기 위한 기존 IKEv2를 경량화 하려는 연구도 있었지만 Phase 내의 단계 삭제 및 파라미터의 호환성 문제로 기존 IKEv2를 이용하는 경우 호환성이 문제가 된다.

본 연구는 기존 경량화 연구들을 분석하여 단점을 도출하여 기존의 인터넷 환경과 호환성을 유지할 수 있도록 연구하였다. 제안 경량화 IKEv2의 경우 호환성 유지를 위해 Phase 내의 단계를 삭제하지 않고 압축 알고리즘을 이용하여 Payload 압축을 진행하며 통신량을 감소시켰다. 제안하는 경량화 IKEv2는 패킷 통신량을 크게 절감할 수 있어 MTU의 사이즈가 작은 IoT 환경에 적용한다면 보안성이 큰 IoT 통신 환경을 구축 할 수 있을 것이다.

REFERENCES

[1] D. Harking and D. Carrel, “The Internet Key Exchange”, *RFC 2409*, 1998

[2] H. J. Um, R. H. Kim and H. Y. Yeom, "Design and Implementation about IKEv2", *Journal of Korean Institute of Communication Sciences*, Vol. 16, No. 3, June 2006

[3] V. Smyslov, “Compression in the Internet Key Exchange Protocol Version 2 (IKEv2) draft-compression-02”, *ELVIS-PLUS*, Sep 2016

[4] G. Montenegro and N. Kushalnagar, J. Hui, D. Culler, “Transmission of IPv6 Packets over IEEE 802.15.4 Networks”, *RFC 4944*, 2007.

[5] J. Hui and P. Thubert, “Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks”, *RFC 6282*, 2011.

[6] Y. H. Kim and K. S. Lee, "IKE version 2 Protocol Standard", *Journal of Korea Institute of Information and Communication Engineering*, July 2003.

[7] S. H. Lim and Y. J. Park, "A study of internet key management protocols : IKE & IKEv2", *Conference of Computing Science and Engineering*, Vol. 30, No 2, 2003.

[8] D. J. Ryu, G. H. Kim and B. N. Noh, "A study on Analysis of Performance in Mobile Security Network with IKEv2", *Conference on Korean Institute of Communication Sciences*, Nov 2005.

[9] William Stallings, “Network Security Essentials’, Pearson, 2013.
[10] S. Kent, “Security Architecture for the Internet Protocol”, *RFC 4301*, 2005.
[11] Zlib Homepage, “http://www.zlib.net”
[12] Strongswan, “https://www.strongswan.org”
[13] BlueZ, “http://www.bluez.org

저자약력

김 홍 성(Hong-Sung Kim) [회원]

- 2010년 2월 : 충남대학교 전기정보통신공학부(학사)
- 2012년 2월 : 충남대학교 컴퓨터공학과(석사)
- 2012년 6월 ~ 현재 : 국가보안기술연구소 연구원

<관심분야> 시스템보안, 디지털포렌식, 해킹

송 인 아(In-A Song) [학생회원]

- 2015년 2월 : 군산대학교 정보통신공학과(학사)
- 2015년 3월 ~ 현재 : 군산대학교 전자정보공학부 석사과정

<관심분야> 네트워크 보안, 네트워크 프로토콜

이 영 석(Young-seok Lee) [종신회원]

- 1992년 2월 : 충남대학교 컴퓨터공학과(학사)
- 1994년 2월 : 충남대학교 컴퓨터공학과(석사)
- 2002년 2월 : 충남대학교 컴퓨터공학과(박사)
- 2002년 3월 ~ 2004년 8월 : 한국전자통신연구원 선임연구원
- 2004년 9월 ~ 현재 : 군산대학교 컴퓨터정보통신공학부 교수

<관심분야> 정보보호, 사물인터넷, 이동컴퓨팅