JKIICE Journal of the Korea Institute of Information and Communication Engineering

한국정보통신학회논문지 Vol. 22, No. 11: 1554~1561, Nov. 2018

DDS Security 성능 향상을 위한 메시지 암호화 기법 연구

한재훈*

Message Encryption Methods for DDS Security Performance Improvement

Jae-Hoon Han*

Naval R&D center , HanwhaSystems, Gumi-City, 39376 Korea

요 약

본 논문은 실시간 통신 미들웨어인 DDS에 대해 알아보고, DDS 보안통신의 성능을 향상하기 위한 방법을 제시한다. DDS는 OMG(Object Management Group)에서 지정한 통신 미들웨어 표준이다. OMG는 최근 발생하는 보안이슈들에 대응하기 위해 DDS Security 표준을 지정하였다. DDS의 보안의 성능은 기밀성의 유지와 전송 속도를 고려해야 한다. 기밀성 측면에서 현재 DDS Security 표준의 암호화 알고리즘인 AES-GCM은 강력한 암호화 알고리즘이지만 메시지 인증관련 부분에 약점이 존재한다. 속도 측면에서 보안기능을 위한 연산 부하는 실시간성이 요구되는 시스템에서 DDS를 사용하는데 제약사항이 된다. DDS의 보안기능을 개선하기 위해서는 AES-GCM보다 빠르고 암호화 강도가 높은 알고리즘이 필요하다. 본 논문에서는 이러한 요구사항을 충족하기 위해 AES-OCB 알고리즘을 적용한 DDS 메시지 암호화 방법을 제안하고 DDS와 전송성능을 비교해 최대 12%의 성능개선을 확인하였다.

ABSTRACT

This paper surveys the DDS, a real - time communication middleware, and proposes ways to improve the DDS secure communication performance. DDS is a communication middleware standard by the OMG. The OMG has released the DDS Security standard to resolve the security issues. The security performance of DDS can be considered into transmission speed and confidentiality. In terms of confidentiality, AES-GCM, currently the encryption algorithm specified by DDS Security, is a very strong encryption algorithm, but there are well known weaknesses associated with authentication. In terms of speed, The computational load for the security function is a restriction to use DDS in systems which requires real-time performance. Therefore, in order to improve the DDS security, algorithms that are faster than AES-GCM and strong in encryption strength are needed. In this paper, we propose a DDS message encryption method applying AES-OCB algorithm to meet these requirements and Compared with the existing DDS, the transmission performance is improved by up to 12%.

키워드 : DDS, DDS Security, 암호화, AES-GCM, AES-OCB

Keywords : DDS, DDS Security, Cryptography, AES-GCM, AES-OCB

Received 20 September 2018, **Revised** 8 October 2018, **Accepted** 29 October 2018

*** Corresponding Author** Jae-Hoon Han(E-mail:jh.han@hanwha.com, Tel:+82-054-640-8887)
Naval R&D center, HanwhaSystems, Gumi-City, 39376 Korea

Open Access http://doi.org/10.6109/jkiice.2018.22.11.1554 print ISSN: 2234-4772 online ISSN: 2288-4165

Ⅰ. 서 론

DDS(Data Distribution Service)는 OMG(Object Management Group)에서 표준화한 통신미들웨어 API (Application Programming Interface)이다. DDS는 분산 환경에서 실시간 처리를 위한 발간-구독 방식의 통신을 지원하며, 성능, 확장성 그리고 가용성 측면에서 강점 있다. 이러한 강점을 기반으로 현재 국방, 교통, 의료 분야를 포함한 다양한 산업시스템에 활용되고 있다[1].

DDS가 주로 활용되는 국방 혹은 사회 기간사업에 대한 보안 위협은 지속적으로 증가하고 있다.[2] 기존의 DDS는 자체 보안기능이 없어 동일 네트워크 도메인(Network Domain)에 포함된 참여자(Participant)라면 누구나 통신 패킷의 감청, 위조 또는 변조 등이 가능한 취약점이 있다.[3] OMG는 이러한 보안 위협에 대응하기 위해 최근 DDS Security표준을 지정하였다[3]. DDS Security는 인증(Authentication), 접근제어(Access Control), 암호화(Cryptographic)그리고 로깅(Logging) 기능으로 구성된다. 인증과 접근제어는 DDS 통신 시작 전 상호간의 인증과 접근권한을 확인하며 암호화 기능은 DDS 통신의 기밀성과 무결성을 보증한다. [4]

암호화 기능은 DDS 통신 중 상호간에 주고받는 메시지에 대한 암호화 연산을 통해 메시지의 기밀성을 유지한다. 이때 암호화 연산으로 인해 필연적으로 처리지연시간이 발생하며[5], 지연으로 인한 시스템 성능 저하가 증가하여 DDS의 장점 중 하나인 실시간성을 저해하는 요소로 작용한다. 그렇기에 암호화 기능 적용 시 DDS의 성능향상을 위해 신속한 암호화 연산이 필요하다. 또한 현재 DDS 암호화에 사용되는 알고리즘인 AES- GCM은 NIST 표준으로 지정되어 널리 사용되고 있지만 특정 상황에서 기밀성을 보장할 수 없는 기능적 취약점을 내포하고 있다. [6] 그렇기에 본 연구에서는 속도와 기밀성 측면에서 AES-GCM에 비해 장점을 가지는 AES-OCB 알고리즘을 적용한 DDS를 제안한다.

이후 단락에서는 DDS와 DDS Security에 대해 분석한 후 현재 DDS Security의 메시지 암호화 알고리즘인 AES-GCM과 새롭게 제안한 AES-OCB대해 분석하였다. AES-OCB를 적용한 DDS Security와 기존의 DDS의 성능 비교를 통해 AES-OCB를 적용한 DDS Security의 장점을 도출하였다.

Ⅱ. 관련 연구

2.1. DDS

DDS는 발행(publish)/구독(subscribe) 모델에 기반한 실시간 데이터 통신 미들웨어 표준이며, 분산 환경에서 데이터 중심 프로그램 모델에 대한 신뢰성을 제공하고 각 노드간의 실시간 통신을 지원한다[1]. DDS는 네트워크에 참여하는 애플리케이션의 위치 혹은 존재와 무관하게 상호간 데이터를 교환하여 네트워크 프로그래밍을 단순화해 분산 어플리케이션 설계 및 구현을 단순화시킬 수 있는 장점을 가지고 있다. DDS 표준은 DDS API 표준을 서술한 DCPS(Data Centric Publisher/Subscriber) [7]와 네트워크 계층 통신 프로토콜을 서술한 RTPS(Real-Time Publish-Subscribe)[8]로 구성된다. DCPS는 발행/구독 모델 기반의 데이터 교환 기능에 대한 인터페이스 표준을 정의한다.

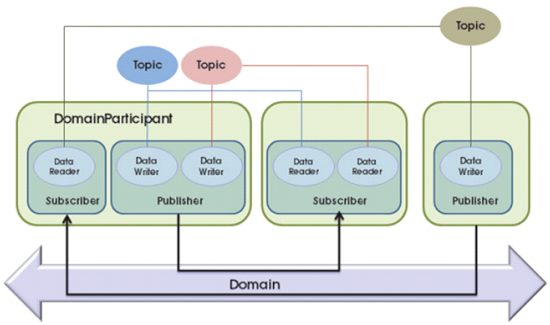

Fig. 1 System architecture DDS middleware

그림 1은 DDS 미들웨어의 통신 구조를 나타내고 있다. 발행자(Publisher)는 전송할 데이터를 생성하고 배포하는 기능을 제공하기 위해 하나 이상의 발간개체(DataWriter)를 생성해 데이터를 배포하고 구독자(Subscriber)는 발간개체와 대응하는 구독개체(DataReader)를 생성해 데이터를 수신한다. 이때, 발행자와 구독자는 동일한 DDS 네트워크 도메인(Domain)에 참여한 상태여야 하고 데이터는 DDS의 토픽(Topic)이라는 개념으로 정의된다. 또한 DDS 표준은 토픽 데이터 전달의 신뢰성과 실시간을 위하여 여러 QoS(Quality Of Service) 설정을 제공하고 있다. RTPS(Real Time Publish/Subscribe)는 실제 메시지 전송과 수신을 담당하는 데이터 전송 프로토콜로서 UDP(User Datagram Protocol)/IP와 같은 전

송계층 위에서도 동작 가능하도록 설계되어있다. RTPS에서는 DCPS에서 정의한 발행자, 구독자 객체가 실제 통신하기 위해 필요한 디스커버리(discovery), 데이터 코딩 방식, 메시지 포맷 및 교환방식, 전송 절차 등에 대한 사항을 정의하고 있다[8].

2.2. 기존 DDS 암호화 방법 분석

DDS 보안 표준은 2016년 버전 1.0 표준 문서를 통해 발표되었으며 아래 다섯 가지의 보안기능을 제공한다.[4]

1. 통신 기밀성 보장
2. 통신 무결성 보장
3. 도메인 참여자에 대한 상호 인증
4. 도메인 참여자에 대한 상호 접근 권한 확인
5. 부인봉쇄

DDS 보안 표준은 위의 정보 보안 기능 제공을 위해 인증, 접근제어, 암호화, 로깅 기능을 제공한다. 이러한 기능들은 기존에 사용하던 DDS 어플리케이션의 변경 없이 동작하기 위해 SPI(Service Plug-in Interface)형태로 설계되었다. DDS Security 표준에 따른 네트워크 통신은 데이터 교환 전 상호간의 인증과 권한확인 절차를 수행한다. 인증이란 자신이 가지고 있는 인증서 교환을 통해 상호간의 권한을 확인/증명하는 과정이며 이 과정에서 데이터 통신에 사용할 비밀 키(Secret Key)를 각자 생성 후 공유한다.

다음으로 접근 권한 확인을 위해 상호간의 송/수신 권한과 도메인의 보안 정책을 확인한 후 비인가자의 통신참여를 제한한다. 인증과 권한확인이 완료된 후 데이터 교환을 시작하게 된다. 전송되는 메시지는 암호화를 통해 보호한다. DDS Security는 128bit와 256bit 암호화 키를 이용해 메시지를 암호화/ 복호화 하는 것을 정의하고 있으며 메시지 암호화 기능은 페이로드(Payload) 암호화, 서브메시지(Submessge) 암호화, RTPS 메시지(RTPS message) 암호화 3단계로 이루어진다.

그림 2는 DSS의 RTPS 메시지 암호화 결과를 보여준다. 각 암호화 과정은 메시지를 암호화 해 암호화된 데이터를 생성하고 해더(Header)와 태그(Tag)를 삽입해 암호화 정보와 무결성을 확인한다. 이때 각 암호화 방법은 각 단계별로 독립적으로 이루어지며 사용자는 제공

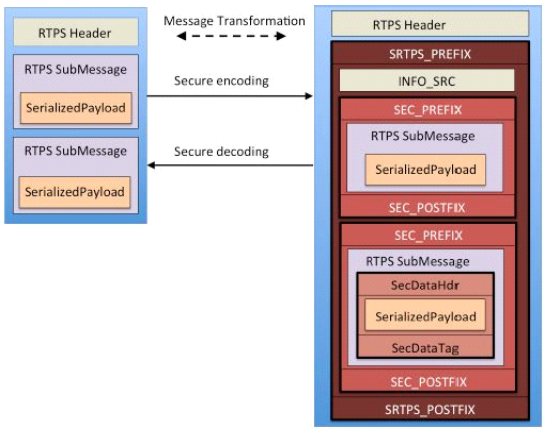

Fig. 2 DDS RTPS Message transformation[4]

되는 QoS를 통해 적합한 암호화 방법을 결정할 수 있다.

메시지 암호화 알고리즘은 AES(Advanced Encryption Standard)의 파생 알고리즘 중 하나인 AES-GCM 사용된다. AES는 NIST 표준에 지정된 차세대 표준 대칭키 암호화 방식이며 효율, 보안, 성능 구현 등의 관점에서 기존 표준인 DES에 비해 탁월한 성능을 보인다.[9] AES-GCM 운영모드는 AES 알고리즘에 기반해 이진 Galois체 상에서 정의된 GHASH함수를 이용하여 기밀 데이터의 인증을 보장하는 운영모드로서 메시지의 기밀성과 인증 기능을 제공 한다.[10] AES-GCM은 NIST 표준으로 지정되어 시스템의 암호화 과정에서 사용되고 있지만 Ferguson와 Saarinen의 연구에 따르면 암호화의 기밀성이 깨질 수 있는 가능성이 존재한다. 이러한 공격 가능성은 태그의 길이가 128bit일 때 최소화되지만 완전히 사라진다고 할 수 없다.[6]

메시지 암호화 과정은 필연적으로 지연시간을 발생시킨다. DDS는 실시간 통신 미들웨어이므로 지연시간을 최소화가 통신 성능의 지표로 작용한다. 본 논문에서는 AES-GCM보다 빠른 암호화 연산속도를 가지며 암호화 공격에 대해 기밀성 유지에 효과적인 AES-OCB 암호화 알고리즘을 DDS에 접목해 AES-GCM기반의 DDS 통신의 단점을 보완하고 성능측정을 통해 개선 사항을 증명할 것이다.

Ⅲ. AES-OCB 기반 암호화 기법

3.1. AES-GCM의 구조 및 취약점 분석

AES-GCM은 가장 보편화 되어있는 인증 암호(authenticated Encryption)중의 하나로 AES 알고리즘에 기초해 설계되었으며 NIST(SP 8000-38D) 표준으로 지정되었다. AES-GCM은 암호화되는 데이터에 기밀성과 무결성을 제공한다. 기밀성은 CTR모드(Counter Mode)를 통해 제공되며, 무결성은 GHASH 함수를 통해 제공된다. 이때 Galois Field (GF)의 곱셈연산과 AES암호 연산은 병렬로 처리되어 고성능 구현이 가능하다. GCM 알고리즘의 암호화 과정은 그림 3과 같이 이루어진다.[10].

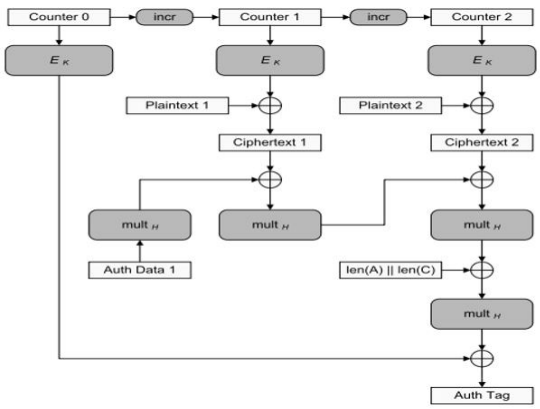

Fig. 3 AES-GCM mode work[10]

마스터키가 입력된 후, 128비트 길이의 블록암호 알고리즘으로 암호화되고, 그 결과를 해시키(hash key)로 사용한다. 해시키 생성이 완료된 후 추가인증데이터(AAD: Additional Authentication Data)가 입력되고, AAD 블록 길이에 따라 곱셈연산을 수행한다. 이후 AES 알고리즘에 의해 CTR 모드 암호화 연산이 수행된다. CTR 연산의 결과로 출력되는 암호문과 이전 GF 곱셈연산결과 값이 XOR 연산된 후, GF 곱셈연산이 수행된다. AES-GCM 암호화의 무결성 증명은 GHASH 함수를 통해 이루어지며 이 과정에서 다항식 표기법(polynomial notation)을 이용해 각 암호문 블록을 해시키와 곱해 빠른 연산을 지원한다.

AES-GCM의 특징으로 암호화/복호화 연산을 병렬화를 할 수 있다. 그 결과 서로 다른 평문 블록이나 암호문 블록을 개별적으로 암호화 또는 복호화 할 수 있다. 하지만 태그 계산과정은 병렬화 할 수 없는 단점이 존재한다.

이렇듯 AES-GCM은 매우 강력한 암호화 알고리즘이지만 인증과 관련한 취약점이 존재한다[11]. 그중 하나는 공격자가 메시지 태그를 많이 알게 되면 위조의 성공확률이 높아지고 비밀 키를 복원할 수 있는 가능성이 존재하는 것이다[6]. 태그의 길이가 n비트이고 공격자가 2^m 개의 블락(block)의 태그를 알고 있을 때 위조 성공확률은 $1/\ 2^n$이 아니라 $2^m/2^n$이 된다. 예를 들어 48bit의 태그를 사용하는 경우, 약 4GB 분량의 메시지와 이에 대한 태그 값을 알고 있는 공격자가 위조에 성공할 확률은 $1/\ 2^{48}$이 아니라 $2^{28}/2^{48}$이 된다.[12] 이러한 약점은 128비트의 인증 값을 사용할 때 현저히 감소한다. 하지만 확률이 줄어들 뿐 공격자가 확보한 메시지 태그의 크기가 지속적으로 증가하면 메시지 위조 가능성 또한 증가하게 된다. 일반적인 응용프로그램에서는 메시지 위조 가능성 취약점이 크게 부각되지 않지만 한 번 작동되면 수개월 이상 실행되고 초당 수백회의 메시지 통신이 이루어지는 국방 전투체계, 교통 통제시스템, 항공 통제 시스템 등 대규모 시스템의 DDS 통신에서는 이러한 취약점으로 인해 보안의 위협이 증가한다. 그렇기에 DDS에 보안기능을 적용할 때 AES-GCM을 보완할 수 있는 추가적인 암호화 알고리즘이 필요하다.

3.2. AES-OCB 소개

AES-OCB(offset codebook)는 필립 로거웨이(Phillip Rogaway) 교수가 개발한 AES 기반의 블록 암호 알고리즘이다. 현재 OCB는 OCB1 과 OCB2를 거쳐 OCB3가 개발된 상태이며, OCB3(이하 OCB)에는 기존에 제공하지 않았던 메시지 태그를 지원함으로써 기밀성과 무결성을 지원하며 GCM과 같은 블록 인증 암호화 알고리즘들과 비교하면 빠른 성능을 가진다[13]. 이러한 장점에도 OCB는 2013년까지 특허권자인 로거웨이로부터 사용권을 얻어야 해 사용에 제한이 있었지만 현재는 비 군사 목적의 응용에 한해 무료 라이센스를 제공하고 있다[14]. 또한 2020년이면 OCB의 특허가 만료되어 현재보다 더욱더 다양한 분야에서 응용될 것으로 예상된다. OCB의 작동방식은 아래의 그림 4와과 같다.

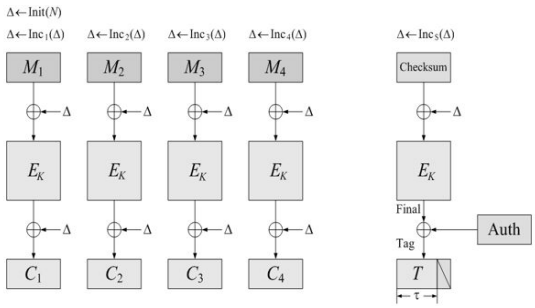

Fig. 4 AES-OCB mode work[13]

OCB 암호화는 각 평문 블록을 메시지(M)을 암호화 키의 크기로 나누어 각각을 XOR연산을 통해 암호화 논스(nonce)에 의존하는 임의의 값이 된다. 연산이 진행되면서 처리된 블록마다 논스 값은 증가된다. OCB인증의 경우 OCB는 모든 평문을 XOR로 결합하고 태그(T)를 계산한다. 이때 Δ는 마지막 평문 블록을 암호화 할 때 계산된 오프셋(offset) 값이 적용된다. AES-GCM과 동일하게 OCB도 AAD를 지원하며 주어진 AAD를 일련의 블록으로 취급해 처리한다. [13][15] 연산과정에서 암호문으로부터 인증 값을 계산하는 AES-GCM과 달리 OCB는 평문자료의 Checksum값을 이용해 태그를 계산한다. 그렇기에 암호화와 인증을 동시에 수행할 수 있는 장점을 가진다[12].

3.3. AES-OCB 알고리즘 적용

DDS 표준이 정의한 암호화 SPI의 구조는 그림 5와 같다. 암호화SPI는 CryptoFactory, CryptoKeyExchange 및 CryptoTransform 모듈로 구성되며 메시지의 암호화/복호화는 CryptoTransform의 내부의 API(Application Programming Interface)가 담당한다. 이때 사용자가 설정한 암호화 방법은 QoS에 따라 암호화 모듈로 전달되고, 각 메시지가 암호화될 때 연관된 암호화핸들(CryptoHandle)을 통해 CryptoTransform의 내부 API로 전달된다.

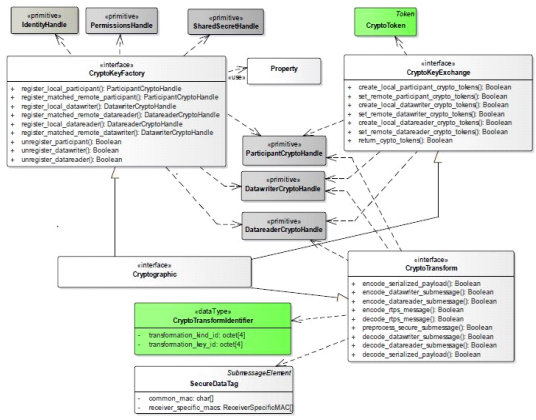

Fig. 5 Cryptographic Plugin Model[3]

기존의 DDS Security에서는 아래 표 1의 5가지(1~5)의 암호화 방법을 지정하고 있다. AES-OCB를 지원하기 위해 추가적으로 4가지(6~9) 암호화 방법을 추가로 정의하고 값을 할당하였다.

Table. 1 Cryptographic SPI TransformationKind

	Transformation Kind	value
1	CRYPTO_TRANSFORMATION_KIND_NONE	{0, 0, 0, 0 }
2	CRYPTO_TRANSFORMATION_KIND_AES128_GMAC	{0, 0, 0, 1 }
3	CRYPTO_TRANSFORMATION_KIND_AES128_GCM	{0, 0, 0, 2 }
4	CRYPTO_TRANSFORMATION_KIND_AES256_GMAC	{0, 0, 0, 3 }
5	CRYPTO_TRANSFORMATION_KIND_AES256_GCM	{0, 0, 0, 4 }
6	CRYPTO_TRANSFORMATION_KIND_AES128_OMAC	{0, 0, 0, 5}
7	CRYPTO_TRANSFORMATION_KIND_AES128_OCB	{0, 0, 0, 6}
8	CRYPTO_TRANSFORMATION_KIND_AES256_OMAC	{0, 0, 0, 7}
9	CRYPTO_TRANSFORMATION_KIND_AES256_OCB	{0, 0, 0, 8}

CryptoTransform의 암호화 API에서 암호화 알고리즘 선택 시 추가된 암호화 방법에 따라 메시지가 암호화될 수 있도록 알고리즘 저장소에 AES-OCB를 추가하였으며, 그림 6과 같이 간략하게 표현할 수 있다. CryptoTransForm 모듈은 내부에 암호화 매니저(CryptoInfoManager)를 통해 각각의 암호화 핸들과 연관된 암호화정보(CryptoInfo)를 저장한다. 암호화 정보에는 자신의 키 재료(KeyMaterial), 암호화 알고리즘 등을 비롯해 암호화에 관련된 정보들이 존재한다.

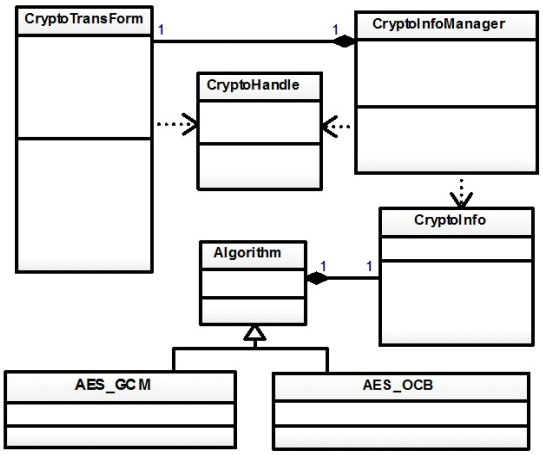

Fig. 6 CryptoTransForm Structure with AES-OCB

그림 6과 같이 AES-OCB 클래스가 알고리즘을 상속받아 외부의 CryptoTransForm의 API의 구조를 수정하지 않고 AES-OCB 알고리즘을 추가하였다.

Ⅳ. 성능비교 및 분석

본 연구에서는 한화시스템에서 기 개발한 Smart DDS를 활용해 AES-OCB 및 AES-GCM알고리즘을 적용하여 메시지 암호화 오버헤드를 측정하고 결과를 분석하였다.

4.1. 시험 환경 제원

알고리즘 시험은 한화시스템사에서 체계 개발 사업에 사용되고 있는 개발 및 운용 환경에서 2대의 PC를 사용해 진행되었다. 시험에 사용된 PC의 자세한 환경 제원은 표 2와 같다.

Table. 2 Experiment Target Environment

Division	Description
OS	Windows 7(64bit)
CPU	Intel Core i7-2620M CPU 2.70GHZ
MEMORY	8G-Byte
DDS	Smart DDS
application	STC_TRANSMISSION

4.2. 시험 방법

암호화 오버헤드 측정하기 위해 표 2에서 개발 및 운용환경의 PC중 1대에서 메시지를 주고받는 시간을 측정하였다. 메시지의 크기는 실제 DDS가 운용되는 체계 환경에서 가장 많이 사용되는 토픽 사이즈인 500B(Byte), 1KB(Kilo-byte), 5KB, 10KB 크기의 토픽 메시지를 전송하고 ACK 메시지를 수신하는 시간을 측정해 지연시간(Round-Trip Latency)을 계산하였다.

정확한 지연시간 비교 측정을 위해 10만회 10회 반복 시험을 통해 지연시간을 측정하여 AES-OCB 및 AES-GCM알고리즘 성능을 비교하였다. 또한 측정 프로그램의 프로세스 우선순위를 가장 높게 지정해 다른 응용프로그램으로 인한 컴퓨팅 자원 경쟁요소를 최소화 하여 암호화 연산으로 인한 지연시간이 얼마나 증가되는지를 시험하였다. 대칭키 암호화 알고리즘으로는 AES-128-GCM과 AES-128-OCB을 사용했고, 한화 시스템에서 개발한 지연시간 측정 프로그램인 STC_TRANSMISSION을 제공받아 시험을 수행하였다.

4.3. 시험 결과

AES-OCB 및 AES-GCM알고리즘의 지연시간 측정 결과는 table 3과 같다.

Table. 3 Experiment Result

SIZE	Algor.	Round-Trip Latency(ms)			
		AVG	MAX	MIN	OCB/GCM
500B	OCB	20189.5	22089	19953	0.88 (↑12%)
	GCM	22939.8	23416	21404	
1KB	OCB	22056.8	22293	21902	0.93 (↑7%)
	GCM	23568.4	23900	22682	
5K	OCB	32209.2	32240	32057	1.007 (↓0.7%)
	GCM	31959.7	32217	31325	
10K	OCB	46988.8	46192	47299	1.0004 (↓0.04%)
	GCM	46965.6	46410	47549	

표3에서 나타난 바와 같이 토픽의 사이즈가 500B와 1KB인 경우 AES-GCM에 비해 AES-OCB의 지연시간이 평균값, 최대값 그리고 최소값 모두에서 확연히 감소하는 것을 확인할 수 있다. 하지만 5KB와 10KB의 경우 기존 알고리즘과 유사한 지연시간을 나타낸다. 이러한 결과를 명확히 비교하기 위해 평균값을 기준으로 비교

해 표로 나타내면 그림 7과 같다.

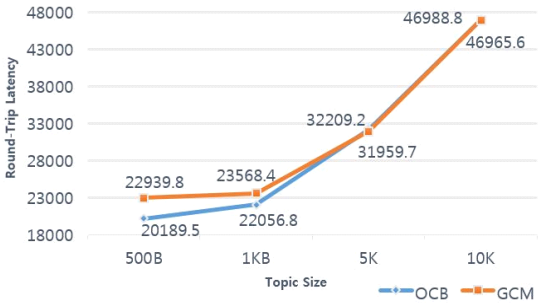

Fig. 7 Average Value Compare Result

그림 7에 나타난 것과 같이 5KB와 10KB 크기의 토픽에서는 기존 알고리즘과 1%내의 오차를 보이며 동일한 성능을 보이는 것을 알 수 있다. 500B와 1KB크기의 토픽에서 각각 12%와 7%의 지연시간 감소가 나타나는 긍정적인 결과를 확인할 수 있었다.

Ⅴ. 결 론

DDS는 규모가 크고 보안 공격이 성공할 경우 파급력이 매우 큰 시스템에서 주로 사용된다. 그렇기에 DDS를 사용하는 시스템에서는 정보 보안에 유의해야 한다. 이에 본 논문에서는 현재의 DDS 보안 기능 적용 시 발생할 수 있는 보안 취약점을 확인해보고 이를 극복하기 위한 새로운 암호화 알고리즘을 제안해 기존의 알고리즘과 지연시간을 측정하고 비교 분석하였다.

1대 1 전송 시험에서는 개발 및 운용환경에서 가장 많이 사용되는 500B, 1KB, 5KB, 10KB 사이즈의 토픽 메시지를 대상으로 지연시간을 측정하였다. 그 결과 토픽의 크기가 5KB이상 커질 경우 기존의 DDS와 거의 비슷한 지연시간을 가지는 것을 확인하였다. 이러한 결과의 원인은 메시지가 커지면서 메모리 할당이나 MTU 설정 등과 같이 복합적인 추가 지연시간이 발생하여 알고리즘변경의 이점이 줄어든 것으로 예상되므로 추가적인 원인 확인이 필요하다. 메시지 토픽의 크기를 500byte로 설정 했을 때 네트워크 지연시간 측면에서 기존 대비 약 12%의 지연시간 감소라는 긍정적인 결과를 얻었다. 이러한 시험 결과는 향후 AES-OCB를 사용하는 DDS의 토픽 크기를 결정하는데 하나의 고려사항이 될 수 있을 것으로 생각된다.

향후 연구로는 크기에 따른 메시지 전송 속도 차이의 원인을 분석하고 현재의 논문에서 진행하지 않았던 멀티케스트(Muliticast)환경에서 암호화 알고리즘 변경으로 인한 성능차이를 확인해, 향후 실제 시스템에 AES-OCB를 적용한 DDS를 사용해 적합성 평가를 진행할 예정이다.

REFERENCES

[1] DDS Portal. What is DDS[Internet]. Available: https://www. omgwiki.org/dds/what-is-dds-3/.

[2] S. H. Ham, and D. W. Park, “Study on Policies for National Cybersecurity,” *Journal of the Korea Institute of Information and Communication Engineering*, vol. 21, no. 9, pp. 1666-1673, Sep. 2017.

[3] T. White, M. N. Johnstone and M. Peacock, “An investigation into some security issues in the DDS messaging protocol,” in *Proceeding of 15th Australian Information Security Management Conference*, Perth, pp. 132-139, 2017.

[4] OMG Std. *DDS Security Version 1.1*, OMG, 2018.

[5] Y. K. Go and C. S. Kim, “Cryptographic Overhead of DDS Security for Naval Combat System Security,” in *Proceeding of the Korean Information Science Society Conference*, Jeju, pp. 1217-1219, 2017.

[6] N. Ferguson.(2005, May). Authentication weaknesses in GCM. *Comments submitted to NIST Modes of Operation Process*[online]. pp. 1–19. Available: https://csrc.nist.gov/csrc/media/projects/block-cipher-techniques/documents/bcm/comments/cwc-gcm/ferguson2.pdf.

[7] OMG Std. *Data Distribution Service for Real-time Systems Version 1.2*, OMG, 2007.

[8] OMG Std, *The Real-Time Publish-Subscribe WireProtocol: DDS Interoperability Wire Protocol Specification Version 2.1*, OMG, 2014.

[9] S. M. Kim, T. M. Chang, H. S. Kim, and M. S. Kang, “Design of High-Speed AES Cipher Processor Using Pipeline Technique,” *Journal of Security Engineering*, vol. 11, no. 2, pp.145-154, Apr. 2014.

[10] D. A. McGrew, and J. Viega, “The security and performance of the Galois/Counter Mode (GCM) of operation,” in *Proceeding of the International Conference on Cryptology in India*, Berlin, pp. 343-355, 2004.

[11] G. Procter and C. Cid, “On weak keys and forgery attacks against polynomial-based MAC schemes,” *Journal of Cryptology*, vol. 28, no. 4, pp. 769-795, Oct. 2015.
[12] J. P. Aumasson, *Serious Cryptography: A Practical Introduction to Modern Encryption*, San Francisco, 2017.
[13] T. Krovetz and P. Rogaway, “The Software Performance of Authenticated-Encryption Modes,” in *Proceedings of the International Workshop on Fast Software Encryption*, Berlin, pp. 306-327, 2011.
[14] OCB Mode. OCB: free licenses[internet]. Available: http://web.cs.ucdavis.edu/~rogaway/ocb/license. htm.
[15] T. Krovetz and P. Rogaway. (2014, May). The OCB authenticated-encryption algorithm, *IETF RFC 7253*[Online], pp. 1-19. Available: https://tools.ietf.org/ html/rfc7253.

한재훈(Han-Jae Hoon)

2013년 2월 부산대학교 정보컴퓨터공학부 공학학사
2015년 2월 부산대학교 정보컴퓨터공학부 공학석사
2015년 12월~현재 한화시스템 해양연구소 연구원
※관심분야 : DDS 미들웨어, DDS Security, 암호화 알고리즘, 데이터 통신