형태보존암호를 이용한 DTLS 프로토콜 기법 제안

Method for DTLS protocol Using Format-preserving Encryption

저자
(Authors)

김시온, 김수현, 정다윗, 이상현, 이선영
Kim Si On, Kim Soo Hyun, Jeong Da Wit, Lee Sang Hyeon, Sun-young Lee

출처
(Source)

한국통신학회 학술대회논문집 , 2021.2, 853-854 (2 pages)
Proceedings of Symposium of the Korean Institute of communications and Information Sciences , 2021.2, 853-854 (2 pages)

발행처
(Publisher)

한국통신학회
Korea Institute Of Communication Sciences

URL

http://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE10547802

APA Style

김시온, 김수현, 정다윗, 이상현, 이선영 (2021). 형태보존암호를 이용한 DTLS 프로토콜 기법 제안. 한국통신학회 학술대회논문집, 853-854.

형태보존암호를 이용한 DTLS 프로토콜 기법 제안

김시온, 김수현, 정다윗, 이상현, *이선영
순천향대학교

gu100433@sch.ac.kr, imsh0314@naver.com, djung0605@gmail.com,
cpd4268@naver.com, *sunlee@sch.ac.kr

Method for DTLS protocol

Using Format-preserving Encryption

Kim Si On, Kim Soo Hyun, Jeong Da Wit,
Lee Sang Hyeon, *Sun-young Lee
Dept of information security Soonchunhyang Univ

요 약

COVID-19 으로 인한 팬데믹 사태를 겪으며 일상의 대부분이 비대면으로 전환되었다. 이로 인하여 Zoom 과 같이 실시간 화상 및 음성 공유가 가능한 프로그램이 주목을 받게 되었다. 실시간 통신 프로그램은 대부분 UDP 프로토콜을 사용하여 통신을 진행하는데 이때 정보를 보호하기 위해 오픈소스 형태의 DTLS 를 같이 사용하여 보안성을 높인다. 본 논문에서는 DTLS 프로토콜의 Feeder 가 Host 로부터 전달받은 데이터를 형태보존암호를 사용하여 암호화하는 기법을 기술한다. 형태보존암호를 사용하여 서버의 자원소모를 줄여 사용자에게 할당하는 자원을 늘릴 수 있도록 프로토콜을 설계하는 것을 제안한다.

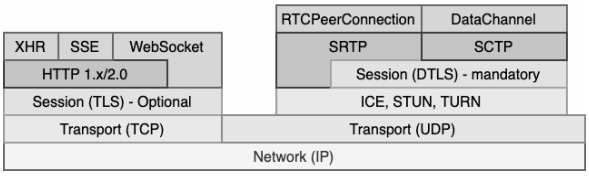
Ⅰ. 서 론

전세계적인 팬데믹 사태에서 모바일, PC 를 가리지 않고 실시간 비대면을 요구하는 상황이 생기며 Zoom 과 같은 프로그램에 관심이 높아졌다. 실시간 화상, 음성 프로그램의 이용자가 늘어나며 보안에 대한 관심이 높아졌다. 과거 대비 오늘날 실시간 비대면 화상, 음성 채팅 프로그램을 통한 공격이 지속적으로 증가하는 추세이다. 이에 따라 기존에 사용하던 UDP 프로토콜의 속도를 유지하며 정보를 보호하기 위해 DTLS 프로토콜을 사용하게 되었다.[1] DTLS 에 사용되는 암호화 알고리즘은 대부분 AES 혹은 RSA 를 사용한다.[2] 이때 DTLS 를 통해 암호화되는 정보들의 보안 강도를 유지하며 서버 측의 자원 소모를 개선하고자 형태보존암호 알고리즘을 이용한 기존 DTLS 프로토콜의 개선에 대해 제안한다.

Ⅱ. 관련 연구

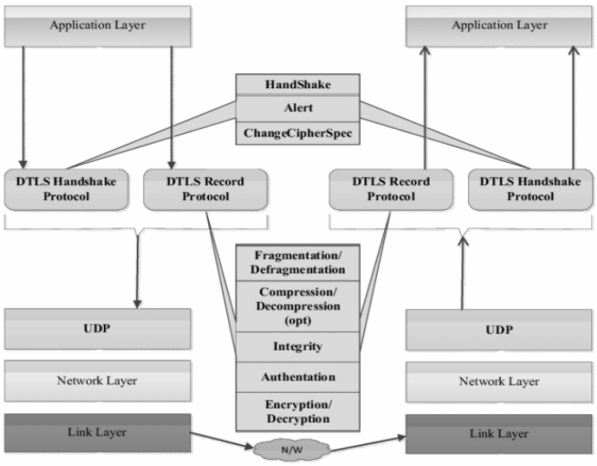
2.1 DTLS 프로토콜

SSL(Secure Socket Layer)과 TLS(Transport Layer Security)는 공개키와 개인키를 교환하여 보안 세션을 생성하여 통신을 암호화하는 방식을 사용하고, 웹 서버와 사용자의 웹 브라우저 간 통신을 암호화하는데 사용되는 프로토콜이다. DTLS(Datagram Transport Layer Security)는 이러한 SSL, TLS 기술을 토대로 만들어졌기에 유사한 보안 강도를 제공하고 있으며, 전송 계층의 TCP(Transmission Control Protocol) 프로토콜에 보안성을 제공해주는 TLS 프로토콜을 UDP(User Datagram Protocol)에 적용 가능하게 해주는 UDP 를 위한 보안 프로토콜이라고 할 수 있다. [1][3]



[그림 1] 프로토콜 스택

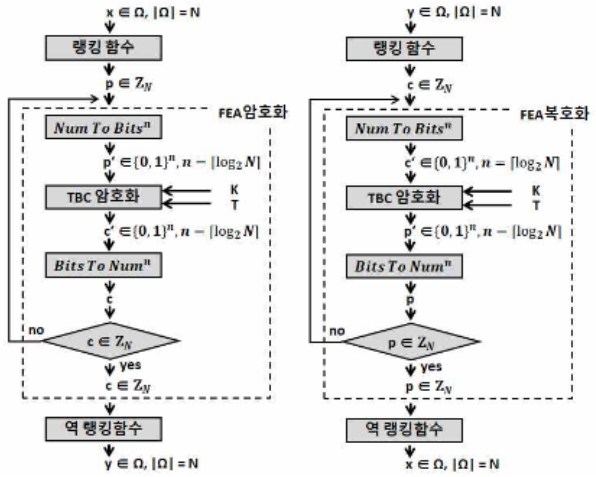
전송 계층 보안(TLS)은 신뢰할 수 있으며 재전송 동작으로 인해 실시간 응용 프로그램에 부적절한 지연이 발생하지만, 데이터그램 전송 계층(DTLS)은 단일 데이터그램 채널을 통해 대량 데이터 전송 및 완전한 키 협상을 제공함으로써 실시간 통신을 확보하기 위한 애플리케이션에 적합하므로 짧은 시간에 많은 패킷을 전달해야 하는 오디오/화상 회의를 위해 사용되고 있다.[1][4] DTLS 는 두 가지 주요 프로토콜로 핸드 쉐이크 프로토콜 및 레코드 프로토콜로 구성되어 있어 암호화 알고리즘 및 매개 변수를 협상, 엔드 포인트를 인증, 세션 키를 계산하기 위한 키 링 자료를 설정하는 메커니즘을 제공한다.[3] DTLS 를 통하여 데이터그램을 주고받는 양쪽 엔드 포인트의 중간 네트워크에서, 전송 중인 데이터를 훔쳐보거나 위조하는 등의 악성 행위를 막을 수 있다.[2]



[그림 2] 핸드 쉐이크 및 레코드 프로토콜의 작업 메커니즘

2.2 형태보존암호

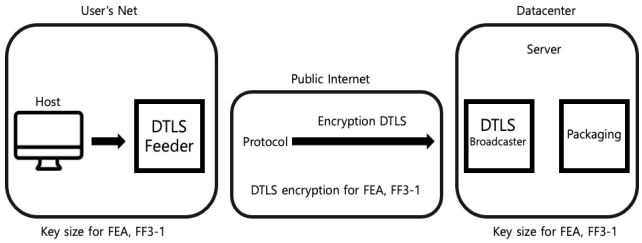
(Format-Preserving Encryption) 형태보존암호는 1997 년 처음 제안되었으며 현재는 다양한 모드를 가지고 있다.[5] 크게는 3 가지 타입으로 나뉘며 FF1, FF3-1 과 FEA 가 있으며 형태보존암호는 암호화 시 평문의 데이터 타입과 길이를 그대로 유지하는 암호화 알고리즘이다. 이때 형태보존암호는 일반적인 암호와 다르게 트윅(Tweak)이라 불리는 부가정보를 적용하여 암호화를 진행하며 Feistel 구조와 유사한 구조를 지녀 안전성을 가진다.[5][6][7] 또한 추가적으로 보안 부담과 시스템의 개발 유지에 필요한 비용이 큰 토큰화와 기존 블록 암호 대비 유용하다.[8]



[그림 3] FEA 암호화/복호화 과정

Ⅲ. 제안 방법

본 논문에서는 DTLS v1.2 과 형태보존암호 중 FEA 와 FF3-1 알고리즘을 사용하는 것을 전제로 서술한다. 형태보존암호는 다른 암호와 다르게 데이터 타입과 길이를 유지할 수 있으므로 서버 측이 암호화된 데이터를 저장하기 위해 DB 의 데이터 타입을 바꾸지 않아도 되며 추가로 저장 공간을 증설하지 않아도 된다는 점에 주목을 했다. 클라이언트와 서버 측의 변경점이 적다는 점도 이점으로 작용한다. 서비스의 제공자는 DTLS 를 적용했을 때와 마찬가지로 FEA 혹은 FF3-1 이 적용된 DTLS 를 UDP 프로토콜과 함께 사용하며 통신 스트림을 암호화하게 된다. 아래 그림은 영상/음성 촬영과 동시에 클라이언트에서 서버로 데이터를 전송 시 보안을 위한 DTLS 의 실시간 동작 과정을 나타냈다.



[그림 4] DTLS_FEA 동작 예시

호스트를 통해 전달된 데이터가 DTLS Feeder 에게 전송된 후, FEA 혹은 FF3-1 암호화 알고리즘을 사용하여 데이터 센터로 전송한다. 이때 서버에서 복호화를 진행 및 다시 패키징과 트랜스코딩을 진행한다. 패키징 된 정보는 사용자의 디바이스를 통해 서버에서 전송을 하게 된다.

Ⅳ. 결론

형태보존암호를 보안 프로토콜에 적용해 사용자의 스트림을 Feeder 가 전송받아 FEA 혹은 FF3-1 알고리즘을 통한 암호화가 진행되어 데이터 센터로 전달하게 된다. 이 데이터는 서버에서 복호화가 되어 사용자에게 적합한 스트림으로 전달하는 DTLS 의 방식과 의도적으로 동일하게 구상하였다. 형태보존암호를 사용하여 서비스를 제공하는 서버 측에서는 암호화를 통해 늘어나는 데이터의 길이와 달라지는 데이터 타입에 맞게 DB 를 수정해야하는 수고와 자원을 절약할 수 있게 된다. 이렇게 절약된 자원은 사용자에게 더 많은 자원을 할당해 줄 수 있게 된다.

ACKNOWLEDGMENT

이 성과는 2018 년도 정부(교육과학기술부)의 재원으로 한국연구재단의 자원을 받아 수행된 연구임. (No. NRF2018R1D1A1B07047656)

참 고 문 헌

[1] N. Modadugu, E. Rescorla, “The Design and Implementation of Datagram TLS” NDSS, 2003.

[2] 안수현, 김광조, <IoT 환경에 적합한 경량 DTLS 프로토콜 구성 방법>, 《KAIST 정보보호대학원》, 2014.

[3] E. Rescorla, N. Modadugu, "RFC 6347, Datagram Transport Layer Security Version 1.2" IETF, 2012.

[4] Jelena Ćurguz, “Vulnerabilities of the SSL/TLS Protocol” The Sixth International Conference on Computer Science, Engineering and Information Technology, 2016.

[5] Michael Brightwell, H.Smith, "Using Datatype Preserving Encryption To Enhance Data Warehouse Security," 20th National Information Systems Security Conference Proceeding, 1997.

[6] Morris Dworkin. "Recommendation for Block Cipher Modes of Operation: Methods for Format-Preserving Encryption." NIST. 2019.

[7]"Format-Preserving Encryption Algorithm FEA." 한국정보통신기술협회. 2015.

[8] 송지환, 〈민감 정보 암호화에 따른 형태 보존 암호화 기술의 재조명〉, 《소프트웨어정책연구소》, 2016-05-25.