NIST 경량암호 공모전 TinyJAMBU의 기능적 고찰

권혁동* 엄시우** 심민주** 서화정***

*한성대학교 정보컴퓨터공학과 (대학원생)
** 한성대학교 IT융합공학부 (대학원생)
***한성대학교 IT융합공학부 (조교수)

A study of the NIST lightweight cryptography contest TinyJAMBU

Hyeok-Dong Kwon* Si-Woo Eum** Min-Joo Sim** Hwa-Jeong Seo***

*Hansung University, Dept of Information Computer Engineering (Graduate student)
**Hansung University, Dept of IT convergence Engineering (Graduate student)
***Hansung University, Dept of IT convergence Engineering (Assistant Professor)

요 약

IoT 기술의 발전에 따라 경량암호 기술의 중요성이 대두되었다. 이런 흐름에 따라 NIST에서는 경량암호 공모전을 개최하였다. 공모전의 목표는 제한된 환경 상에서 안전하고 효과적으로 사용할 수 있는 경량암호 알고리즘을 표준화하는 것이다. 2022년에는 최종 라운드가 진행 중에 있으며, 10개의 알고리즘이 진출하였다. 본 논문에서는 Finalist 진출 알고리즘 중 하나인 TinyJAMBU에 대한 기능적인 고찰을 한다.

I. 서론

IoT(Internet of Things) 기기는 소형 센서 노드들과 같이 초소형 디바이스로, 인터넷 상에 연결되어 다른 기기들과 통신하며 윤택한 생활을 가능케 한다. 이러한 기기들은 제한된 하드웨어 환경을 가지는데, 일반적인 암호 알고리즘은 많고 복잡한 연산을 지니기에 IoT 기기와 같은 극한의 환경에서 원활한 구동이 어렵다.

경량암호(Lightweight cryptography)는 제한된 환경 상에서 안전한 암호화를 제공할 수 있도록 제안된 암호 알고리즘이다. 따라서 안전한 사물 인터넷 환경을 조성할 수 있는데 매우 중요한 역할을 하고 있다.

이러한 흐름에 따라 NIST(National Institute of Standards and Technology)에서는 경량암호 표준화를 위한 공모전을 개최하였다. 해당 공모전의 취지는 안전한 사물 인터넷 환경을 제공하기 위해 경량암호 표준을 설립하고자 2019년부터 공모전 1라운드가 시작되었다.

본 논문에서는 NIST 경량암호 공모전에 대한 내용과 해당 공모전의 Finalist 중 하나인 TinyJAMBU에 대한 기능적인 부분에 대해서 탐구한다. 본 논문의 구성은 다음과 같다. 2장에서 NIST 경량암호 공모전의 현황에 대해 알아본다. 3장에서 NIST 경량암호 공모전의 Finalist인 TinyJAMBU에 대해서 확인한다. 4장에서 본 논문의 결론을 맺는다.

II. NIST 경량암호 공모전

NIST에서는 경량암호의 필요성에 따라 표준화 작업이 필요할 것이라 예측하고 2018년 공모전 개최를 공표했다. NIST는 평가 기준으로 112-bit 이상의 보안강도, 다양한 플랫폼 상에서의 효율적인 구현 가능성, 낮은 오버헤드, 작은 암호문 크기, 부채널 공격 내성, 평문과 암호문 쌍의 숫자 제한이 있음을 밝혔다[1].

공모전 발표 이후 57개의 작품이 출품되었고 Round 1 후보로 56개의 작품이 선정되었다. 이

후 2019년 Round 2 후보 알고리즘으로 32종의 알고리즘이 진출하였다. 2021년에는 Finalist 알고리즘으로 총 10 종류의 알고리즘이 선정되었으며 해당 알고리즘들을 동작 방식에 따라 구분하면 표 1과 같이 나열할 수 있다.

Table. 1. The finalists of NIST lightweight cryptography standardization.

Core function	AEAD and Hasing	AEAD only
Permutation	ASCON, PHOTON-Beetle, SPARKLE, Xoodyak	Elephant, ISAP
Block Cipher	-	GIFT-COFB, TinyJAMBU
Tweakable Block Cipher	-	Romulus
Stream Cipher	-	Grain-128AEAD

NIST에서는 기본적인 요구사항으로 AEAD(Authenticated Encryption with Associated Data)를 제공할 것을 제시했고, 추가적으로 Hashing을 지원할 수 있게 하였다. 라운드를 진행하며 탈락한 알고리즘들은 특정 공격에 대한 취약점이 밝혀져서 탈락하게 되었다. 대표적으로 Forgery attack은 입력 값 쌍이 다르더라도 같은 인증 값이 생성되는 공격으로 인증 과정을 회피할 수 있다[2].

III. TinyJAMBU

TinyJAMBU는 CAESAR 경진대회에서 제안된 JAMBU[3]의 변형이다. JAMBU는 CAESAR에서 제안된 알고리즘 중에 가장 작은 블록 크기를 가지는 알고리즘이었다. 이후 JAMBU를 변형한 TinyJAMBU가 제안되었다.

TinyJAMBU는 블록암호 기반의 내부 연산을 지원하며 키 크기는 128-bit, 192-bit, 256-bit의 세 종류를 지원한다. 공통적으로 논스는 96-bit를 사용한다.

TinyJAMBU의 핵심적인 연산은 keyed permutation으로, 모든 과정에서 keyed permutation을 반복적으로 연산하는 단계가 포함된다. keyed permutation의 가장 큰 특징은 특별한 키 스케줄 과정이 없다는 것이다. 즉, 다른 암호 알고리즘은 입력 받은 비밀키에서 연산에 사용할 라운드 키 값을 도출한다. 반면에 TinyJAMBU는 라운드 키 생성 과정을 거치지 않고 비밀키를 그대로 사용하는 특징이 있다.

keyed permutation 연산을 위해서 내부적으로 그림 1과 같은 128-bit 기반의 Nonlinear Feedback Shift Register(NLFSR)을 사용한다.

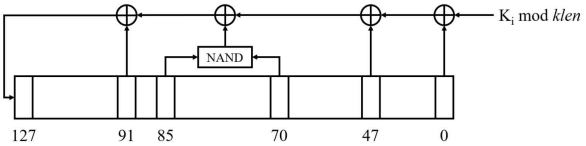

Figure. 1. Structure of Nonlinear Feedback Shift Register.

keyed permutation은 P_n으로 지칭되며, 이때 n은 permutation의 횟수인 라운드를 의미한다. permutation 과정은 표 2와 같은 알고리즘을 따라서 진행된다.

Table. 2. The keyed permutation P_n process.

```
StateUpdate(S, K, i):
    feedback = s0⊕s47⊕(~(s70&s85))⊕s91⊕k(i mod klen)
    for j from 0 to 126: sj = sj+1
    s127 = feedback
end
```

TinyJAMBU는 Initialization, Processing AD(Associated Data), Encryption/Decryption, Finalization, 그리고 Verification 과정을 지닌다. 이때 각 단계별로 keyed permutation을 진행하는 횟수에 차이를 둔다. 각각의 TinyJAMBU 별로 keyed permutation의 반복 횟수는 표 3에 정리되어 있다.

Table. 3. The list of number of permutation.

Key length	128	192	256
Initialization key setup	1024	1152	1280
Inititialization nonce setup	640	640	640
Processing AD	640	640	640
Encryption/ Decryption	1024	1152	1280
Finalization	1024, 640	1152, 640	1280, 640
Verification	1024, 640	1152, 640	1280, 640

TinyJAMBU는 비밀키가 기기 내부에 저장된 상태에서도 안전한 환경을 제공하게 설계되었다. 따라서 공격자가 CPA(Chosen Plaintext Attack) 상황에서 공격을 시도해도 비밀키를 복구할 수 없도록 설계되었다. 이런 특징으로 인해 TinyJAMBU는 비밀키 자체를 저장해두고 사용하는 하드웨어 상에서 사용하기 유리한 암호 알고리즘이다. 또한 NLFSR은 하드웨어 구현 비용이 적기 때문에 하드웨어 친화적인 암호로 평가된다[5].

IV. 결론

본 논문에서는 NIST 경량암호 공모전의 Finalist 중 하나인 TinyJAMBU에 대한 내용을 간략히 살펴보았다. TinyJAMBU는 키 스케줄이 없는 특이한 형태로 동작하고 다른 암호에 비해 매우 작은 소스코드의 크기 및 하드웨어 친화적인 암호이다. 하지만 안전성 부분에서 TinyJAMBU의 차분 분석 및 선형 분석 결과 안전성이 떨어진다는 평가가 있었고, 그에 따라 Finalist 진출의 수정판 에서는 keyed permutation 384회 사용 부분이 640번으로 수정되었다[6].

이와 같이 TinyJAMBU는 장점이 많은 암호 알고리즘 및 인증 알고리즘이지만 지속적인 연구를 통해 적합성 여부를 판단하며, 아울러 효과적인 소프트웨어 구현을 위한 최적 구현 연구도 진행할 수 있다.

V. Acknowledgment

이 성과는 2022년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. NRF-2020R1F1A1048478, 100%).

[참고문헌]

[1] D.S.Milojicic, V.Kalogeraki, R.Lukose,K.Nagaraja, J.Pruyne, B.Richard, S.Rollins and Z.Xu, Peer to Peer Computing, HP Laboratories Palo Alto HPL-2002-57, March, 2002.

[2] H.J.Kim, J.H.Park, H.D.Kwon, and H.J.Weo, “A trend of NIST cryptography standardization contest,” *Review of KIISC*, 30(6), 117-123, 2020.

[3] H.Wu, and T.Huang. “JAMBU lightweight authenticated encryption mode and AES-JAMBU,” *CAESAR competition proposal*, 2014.

[4] H.Wu, and T.Huang. “TinyJAMBU: A family of lightweight authenticated encryption algorithms (version 2),” *Submission to the NIST Lightweight Cryptography Standardization Process*, 2021.

[5] S.J.Baek, Y.G.Jeon, H.G.Kim, and J.S.Kim, “Technology trend of NIST Lightweight Cryptography Competition,” Review of KIISC, 30(3), 17-24, 2020.

[6] D.Saha, Y.Sasaki, D.Shi, F.Sibleyras, S.Sun, and Y.Zhang, “On the security margin of TinyJAMBU with refined differential and linear cryptanalysis,” *IACR Transactions on Symmetric Cryptology*, 152-174, 2020.