서버환경에서의 LEA 암호 알고리즘 구현 및 성능분석

윤채원*, 이재훈**, 이옥연°

*국민대학교 금융정보보안학과

Analysis of Implementation and Performance of LEA Algorithm for Server Environment

Chae-won Yun*, Jaehoon Lee**, Okyoen Yi°

*Dept. of Financial Information Security, Kookmin University

chwyun91@kookmin.ac.kr, guderian88@kookmin.ac.kr, oyyi@kookmin.ac.kr

요 약

최근 서버는 다양한 응용서비스를 제공하기 때문에 수많은 데이터를 빠르게 처리할 수 있는 능력이 요구되고 동시에 보안의 필요성이 강조되고 있다. 서버 보안에는 국제 표준 암호 알고리즘인 AES 암호 알고리즘이 주로 사용된다. AES 암호 알고리즘은 암호학적 안전성과 성능 우수성을 인정받아 여러 나라에서 활용되고 있다. 국내에서는 2004년 ARIA 암호 알고리즘 개발을 시작으로, 최근 2012년 LEA 암호 알고리즘이 개발되었고 암호학적 안전성 또한 인정받았다. 본 논문에서는 다양한 서버환경에서 국제 표준 AES 암호 알고리즘, 국내표준 ARIA 암호 알고리즘과의 성능비교분석을 통해 LEA 암호 알고리즘의 우수성을 보이고자 한다.

ABSTRACT

With recent growing of application service, servers are required to sustain great amount of data and to handle them quickly: besides, data must be processed securely. The main security algorithm used in security services of server is AES(Advanced Encryption Standard – 2001 published by NIST), which is widely accepted in the world market for superiority of performance. In Korea, NSRI(National Security Research Institute) has developed ARIA(Academy, Research Institute, Agency) algorithm in 2004 and LEA(Lightweight Encryption Algorithm) algorithm in 2012. In this paper, we show advantage of LEA by comparing performance with AES and ARIA in various servers.

키워드

AES, ARIA, LEA, Server Encryption, Blockcipher, Cycle per bytes

Ⅰ. 서 론

IT 기술이 발달함에 따라, 사회는 보이는 곳은 물론 보이지 않는 곳까지 IT 기술이 많이 활용되고 있다. 인터넷뱅킹이나 스마트폰의 사용이 대표적인 예이다. 발달하는 IT 사회에서 통신이나 저장에 처리되는 데이터의 크기나 양이 매우 증가했고, 활용되는 환경도 모바일, 소형장비부터 대용량 서버까지 매우 다양해졌다. 급증하는 IT기술로, 각 환경에서는 많은 사용자가 증가하고 많은 데이터의 양을 처리하는 것이 가능해졌지만, 보안적인 문제점이 계속해서 발생하면서 사용자나 관리자 모두 보안의 필요성을 인식하게 되었다. 그러나 사용자의 입장에서는 보안의 필요성은 알지만, 기존에 제공받던 서비스가 보안이 적용되어 속도가 느려지는 것은 원하지 않는다. 이러한 사용자의 입장을 만족할 수 있도록 각 환경에서는 속도의 차이가 발생하지 않는 보안서비스를 제공해야 한다.

특히, 빅데이터를 처리하는 환경이나 클라우드 환경의 경우, 크고 많은 데이터를 처리하는 서버에서의 역할이 중요하다. 서버는 사용되는 용도에

따라 그 종류가 다양하지만, 어떤 서버이든지 기존에 사용자에게 제공되던 기능과 속도는 크게 차이가 없도록 하면서 보안이 적용되도록 해야 한다. 이를 만족하기 위해, 서버가 연산 처리능력이 매우 좋거나, 데이터를 암호화하는데 쓰이는 알고리즘의 속도가 매우 빠른 경우를 고려할 수 있다. 서버의 연산 처리 능력이 매우 좋다면 좋겠지만, 모든 서버가 해당되는 것은 아니므로, 대부분은 암호 알고리즘의 속도가 적게 걸리기를 기대한다.

지금까지는 많은 서버에서 국제 표준으로 지정되어 있는 AES 암호 알고리즘이 전세계적으로 많이 사용되어 왔다. AES 암호 알고리즘은 블록암호 알고리즘이어서 공개키암호 알고리즘보다 속도가 빠르며 안전성 또한 검증 받았다. 또한, 최적화도 많이 이루어졌고 이에 관한 많은 논문들이 있다.

본 논문에서는 국내에서 ARIA에 이어 최근 새롭게 제시한 새로운 암호 알고리즘 LEA를 소개하고, 국제 표준 알고리즘인 AES, 국내 표준 알고리즘인 ARIA와 속도를 비교하여, LEA 암호 알고리즘의 우수성을 보이고자 한다.

본 논문의 구성은 2장에서 AES, ARIA, LEA 알고리즘의 개요 및 비교에 대해 설명하고, 3장에서 실험 서버환경을 설명하고 각 서버에서의 알고리즘 비교 결과를 정리 및 분석하고, 4장에서 결론으로 끝맺는다.

Ⅱ. 관련 연구

AES 암호 알고리즘은 국제 표준으로서 전 세계적으로 많이 사용되고 있고, ARIA 암호 알고리즘은 국내 표준으로서 공공기관에 많이 활용되고 있다. 또한, LEA 암호 알고리즘은 두 암호 알고리즘과 동일한 키 길이를 가지므로, 동일한 보안 강도를 제공한다, 이번 장에서는 세 알고리즘을 소개한다.

2.1 AES [1]

AES(Advanced Encryption Standard)는 알고리즘은 훌륭하지만 키 길이로 인하여 안전성에 문제가 발생한 DES를 대체하기 위해 미국 국립 표준 기술 연구소(NIST)에서 전 세계적으로 공모한 결과, Daemen과 Rijmen에 의해 개발되어 2000년 10월에 채택된 국제 표준 알고리즘이다. AES 암호 알고리즘은 많은 알려진 공격에 대한 저항력이 강하고 다양한 플랫폼에서 활용될 수 있어서, 전 세계적으로 가장 널리 이용되고 있다.

AES 알고리즘은 SPN(Substitution-Permutation Network) 구조의 블록암호로서, 입력 평문의 길이는 128bit로 고정이고, 암호화에 사용되는 라운드 키의 길이는 요구되는 안전성의 정도에 따라 128bit, 192bit, 256bit 중에서 선택이 가능하다. 선택한 키의 길이에 따라 10, 12, 14 라운드가 적용되며, 각 라운드는 SubBytes, ShiftRows, MixColumns, AddRoundKey 의 독립적인 네 함수로 구성되어 있다.

함수들 중에서 SubBytes 와 MixColumns는 유한체 연산을 기반으로 만들어졌기 때문에 구현하기에 따라 속도차이가 크게 발생한다.

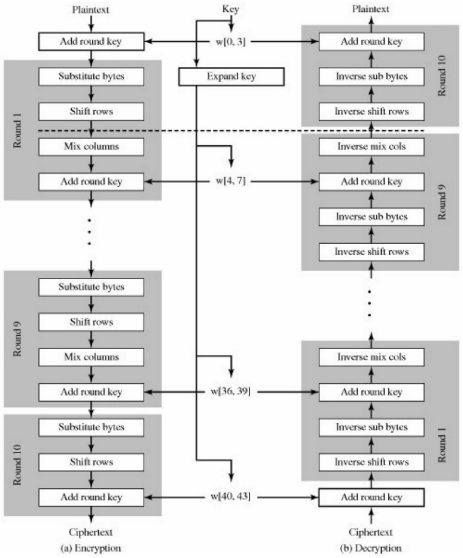

그림 1. AES 암복호화 과정

AES 암호 알고리즘은 유한체를 연산하는 방식으로 구현할 경우, 속도가 매우 느려질 수 있으나, LUT(Look-Up Table) 방식으로 구현할 경우 매우 빠르게 동작한다. 하지만, 이 경우 미리 Table을 구성해야 하므로 많은 메모리가 요구되어, 작은 메모리를 갖는 환경에서는 어려움이 있다는 단점이 있다.

2.2 ARIA [2]

ARIA(Academy, Research Institute, Agency)는 네트워크 기반의 전자정부 시스템을 비롯한 정보보호 환경을 대비하여 국가보안기술연구소에서 개발된 국가 암호 알고리즘으로, 2004년 국가 표준으로 지정되었다. ARIA 암호 알고리즘은 스마트 카드 등의 초경량 환경 및 고성능 서버 환경에서 장점을 가지고 있다.

ARIA 암호 알고리즘은 ISPN(Involution SPN) 구조를 가지므로, 별도의 복호화 알고리즘 없이 복호화를 할 수 있는 장점이 있다. AES와 마찬가지로 128bit, 192bit, 256bit 선택가능하며, 키의 길이에 따라 12, 14, 16 라운드가 적용된다. 각 라운드는 짝수라운드와 홀수라운드에 따라 구성이 약간 다르지만 전체적으로는 치환계층, 확산계층의 과정을 거친다.

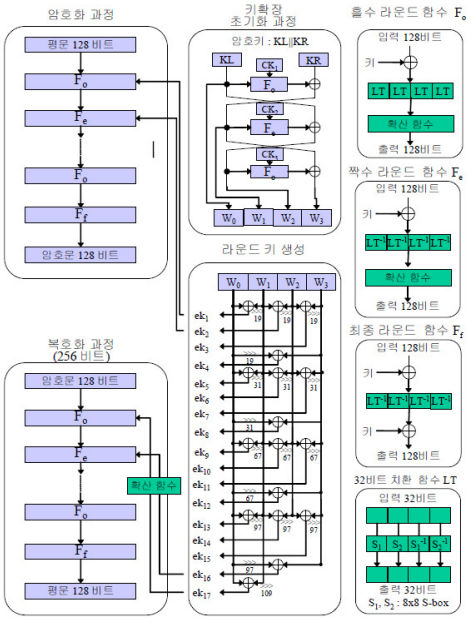

그림 2. ARIA 암복호화 과정

ARIA 암호 알고리즘은 벨기에 루벤 대학으로부터 안전성에 대한 평가를 받았다. 하드웨어 구현 및 8bit 환경에서 뛰어난 효율성을 가지고 있어 IC카드, VPN장비 등 다양한 환경에서 가능하고, 소프트웨어 구현에서도 벨기에 루벤 대학의 효율성 평가에서 AES에 근접하는 성능을 보였다.

그러나, 확산계층의 연산이 16×16의 행렬연산이어서 매우적은 메모리를 가진 장비에서는 어려움이 있다는 한계를 가지고 있다.

2.3 LEA [3]

LEA(Lightweight Encryption Algorithm)는 2012년 국가보안기술연구소에서 개발한 높은 안전성과 우수한 효율성을 제공하는 블록암호 알고리즘이다. LEA 암호 알고리즘은 모바일 기기에서의 저전력 데이터 암호화, 대용량 데이터 서버에서의 고속 데이터 암호화에 적합하도록 설계되어 있다.

LEA 암호 알고리즘은 단순한 라운드의 반복으로 구성되어 있는데, 각 라운드는 단지 ARX(Addition, Rotation, XOR) 연산으로 구성되어 32bit 플랫폼에서 고속으로 동작할 수 있고, 기존 블록 암호들에서 대부분 있던 S-box를 사용하지 않음으로서 경량 구현이 가능하도록 하였다. AES와 마찬가지로 128bit, 192bit, 256bit로 키의 길이를 선택할 수 있고, 각 키의 길이에 따라 24, 28, 32의 라운드 수가 요구된다.

또한, LEA 암호 알고리즘은 블록암호에 대한 모든 공격에 대하여 안전하도록 설계했으며, 국외 전문 연구기관인 벨기에 COSIC 연구소로부터 안전성에 대한 객관적 검증을 받았다[4].

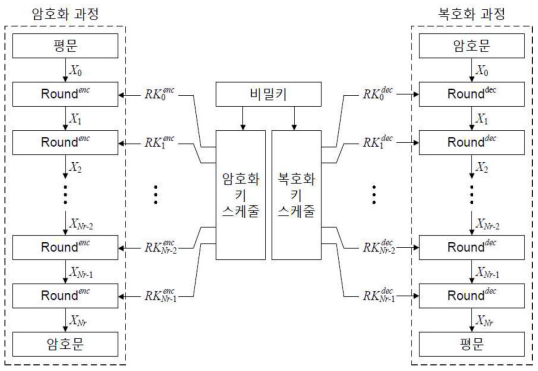

그림 3. LEA 암복호화 과정

Ⅲ. 실험 환경 및 결과

본 논문의 실험은 총 4개 서버로 Windows 2008 R2, Ubuntu 12.04 Server, Radhat 6.5, CentOS 6.5에서 진행했다. Windows 2008 R2는 Visual Studio 2012환경에서 진행했으며 컴파일 옵션으로 -O2를 사용했다. Ubuntu 12.04 Server는 gcc 4.6.3버전에서 커파일 옵션 -O2를 사용했다. Redhat 6.5와 CentOS 6.5는 gcc 4.4.7버전에서 컴파일 옵션으로 -O2를 사용했다. 각 4개의 서버환경에서 C언어로 측정하였으며, 암호 알고리즘 AES는 Oepnssl t-table LUT소스, ARIA는 KISA 참조소스, LEA는 암호포럼 참조소스로 각각 키 생성, 암호화, 복호화, 키 생성+암호화, 키 생성+복호화로 총 5개로 구분하여 CPB(Cycle per bytes)를 측정했다.

다음 [표 1]은 세 알고리즘의 키 생성과 암호화 복호화를 구별하여 측정한 결과이다. 키 생성의 경우 AES가 가장 빠르며, ARIA, LEA순이다. 암호화와 복호화는 LEA가 가장 빠르며, AES, ARIA순으로 결과가 나왔다.

표 1. 각 암호 알고리즘 CPB 결과 - 1

		KeyGen	Enc	Dec
Windows 2008 R2	AES	8.69	9.81	9.81
	ARIA	10.75	22.69	22.50
	LEA	15.31	4.88	6.34
Ubuntu 12.04 LTS	AES	8.25	8.25	12.38
	ARIA	11.25	18.94	19.12
	LEA	20.62	3.75	5.77
Redhat 6.5	AES	7.00	9.81	12.06
	ARIA	11.88	20.56	20.75
	LEA	18.50	5.06	6.88
CentOS 6.5	AES	7.31	8.62	12.56
	ARIA	11.81	19.31	19.12
	LEA	19.50	3.94	5.88

다음 [표 2]는 세 알고리즘의 키 생성과 암호화, 키 생성과 복호화를 통합하여 측정한 결과이다. 키 생성과 암호화를 통합한 경우 AES가 가장 빠르며, LEA, ARIA순이다. 키 생성과 복호화를 통합한 경우 LEA가 가장 빠르며, AES, ARIA순으로 결과가 나왔다.

표 2. 각 암호 알고리즘 CPB 결과 - 2

		Key+Enc	Key+Dec
Windows 2008 R2	AES	20.19	37.44
	ARIA	34.94	54.63
	LEA	22.88	24.72
Ubuntu 12.04 LTS	AES	17.81	37.69
	ARIA	31.88	47.81
	LEA	26.62	28.64
Redhat 6.5	AES	17.75	37.44
	ARIA	34.38	53.50
	LEA	25.12	27.31
CentOS 6.5	AES	18.38	36.38
	ARIA	32.44	49.69
	LEA	25.12	27.25

Ⅳ. 결 론

AES 암호 알고리즘은 1-20년 가량 되었음에도 여러 공격들에 제시만 되었을 뿐 명확히 알려진 공격법은 없기 때문에, 안전하다고 인정받아 현재 매우 많은 곳에서 사용되고 있다. 그러나, LEA 암호 알고리즘은 세상에 알려진지 2-3년 밖에 되지 않은 신생 알고리즘임에도 불구하고, AES에서 제시된 연관키 공격과 단일키 공격법인 Biclique 공격에 강한 내성을 가진다고 평가되었다. 또한, 연산 크기나 종류가 매우 간단한 경량 암호이므로, 크기가 큰 빅데이터나 많은 데이터의 양을 빠르게 처리하는데 효과적일 것으로 기대된다.

다만, 실험의 결과 1과 결과 2를 보면 LEA의 경우 암호화와 복호화가 빠른 반면 키생성은 기타 다른 알고리즘에 비해 느린 것을 확인 할 수 있다. 따라서 키생성이 빈번하지 않은 경우 키 생성이 선행되고 암호화와 복호화만 동작하면 LEA의 최대 성능을 사용할 수 있으며, 효율적인 방법으로 사용될 것으로 예상된다. 세 알고리즘 모두 참조코드를 이용했으나, 속도 향상을 위해 최적화 또는 병렬처리로 구현 시 속도 차이는 더 클 것으로 기대된다.

참고문헌

[1] Federal Information Processing Standards, “ADVANCED ENCRYPTION STANDARD (AES)”, *FIPS PUB 197*, November 2001.

[2] 지식경제부 기술표준원, “128비트 블록암호 알고리즘 ARIA”, *KS X 1213:2004*, December 2004.

[3] 한국정보통신기술협회 (TTA), “128 비트 블록 암호 LEA”, *TTAK.KO-12.0223*, December 2013.

[4] Security Evaluation of the Block Cipher LEA Final Report, COSIC, July. 2011.

본 연구는 미래창조과학부 및 정보통신기술연구진흥센터의 정보통신·방송 연구개발사업의 일환으로 수행하였음. [10039140 , 스마트 디바이스용 칩(ARM7/9/11, UICC 등)에 최적화된 암호(ARIA, SEED, KCDSA 등)의 국가 인증 모듈 및 배포 체계 개발]