한국산학기술학회논문지
Vol. 8, No. 5, pp. 1069-1075, 2007

멀티미디어 통신망의 네트워크 보안을 위한 새로운 BESA 알고리즘 개발 및 설계

박형근[1*], 이승대[1], 김선엽[2]

Development and Design of New BESA Algorithm for Network Security in Multimedia Communication

Hyoung-Keun Park[1*], Seung-Dae Lee[1] and Sun-Youb Kim[2]

요 약 본 논문에서는 암/복호화의 동시수행, 가변되는 입력데이터들에 대한 라운드 횟수 결정, 별도의 키 생성알고리즘 없이 결정된 라운드 횟수에 대한 키 생성, 인증기능 등의 특징을 갖는 새로운 블록암호알고리즘인 BESA 암호화 알고리즘에 대한 연구를 수행하였다.

Abstract New BESA cryptographic algorithm is suitable network environment and wire/wireless communication network, on implement easy, security rate preservation, scalable & reconfigurable. Though proposed algorithm strengthens security vulnerability of TCP/IP protocol and keep security about many user as that have authentication function in network environment, there is important purpose. So that new BESA cryptographic algorithm implemented by hardware base cryptosystem and en/decryption is achieved at the same time, composed architecture.

Key Words : network security, DES, AES, cryptographic algorithm

1. 서론

멀티미디어 통신망은 보다 많은 기능을 부가하여 통합하는 시스템으로 진화하고 있으며 더욱 복잡한 보안 분야의 발전을 요구하게 된다. 특히 상호 관계를 지향하는 네트워크 환경 플랫폼은 다양한 프로토콜과 서비스가 요구된다. 이러한 시스템의 발달은 OS 자체에 대한 보안 패치의 S/W적인 업그레이드와 더불어 대용량, 실시간 처리를 요구하는 환경이 형성되었다[1]. 따라서 본 연구에서는 암/복호화의 동시수행, 가변되는 입력데이터들에 대한 라운드 횟수 결정, 별도의 키 생성알고리즘 없이 결정된 라운드 횟수에 대한 키 생성, 인증기능 등의 특징을 갖는 새로운 블록암호알고리즘인 BESA(Block Encryption Symmetric Algorithm) 암호화 알고리즘에 대한 연구를 수행하였다.

이 논문은 산학협동재단 학술연구비지원에 의하여 연구되었음

[1]남서울대학교 전자공학과
[2]남서울대학교 정보통신공학과
*교신저자: 박형근(phk315@nsu.ac.kr)

2. 대칭형 암호알고리즘

대칭형 암호방식에서는 암호키와 복호키가 동일하며, 두 개체가 같은 키를 공유하면서 하나의 키를 사용하여 암호화하고 복호화 한다. 또한 알고리즘 자체의 구조적 개념을 이용하여 암/복호화를 수행하기 때문에 수행 속도가 비대칭형 방식에 비하여 매우 빠른 장점을 가진다[2].

그러나 많은 수의 사용자가 네트워크 환경에 연결되어 있을 경우 키 관리 및 보관, 유지 등의 공유 문제와 키 자체를 상대방에게 안전하게 전송해야 하는 전달방법적인 문제가 있다. 대칭형 암호방식은 비대칭형에 비하여 월등히 처리 속도가 빠르고 구현상의 복잡도가 낮아 구현하기에 용이하기 때문에 매우 유용한 암호화 방식이다[3].

블록암호방식의 하나인 DES(Data Encryption Standard)는 1977년 미 연방 표준으로 채택된 후 세계표준으로 사용되어 왔다. 그러나 현재 DES 암호알고리즘은 안전도에 문제가 많아 DES를 다중 연결하여 안전성을 보장받도록 하는 3중 DES가 일반화되어 사용되고 있다[4][5].

기존 블록암호는 대부분 64비트 블록크기와 56비트의 키를 지원하고 있으나 64비트 블록길이는 블록암호를 이

용한 MAC(message authentication code)의 응용 등에서 충분한 안전도를 제공하지 못하며 64비트 정도의 키 길이로는 전수 키 검사에 대한 충분한 저항성을 제공하지 못한다는 것이 일반화되어 있다. 따라서 AES(Advanced Encryption Standard)의 기본 요구조건의 하나는 128비트 블록길이에 128, 192, 256비트의 키를 지원하도록 하는 것이다[5].

AES는 128비트 블록암호알고리즘으로 3중 DES보다 안전하고, H/W나 S/W로 효율적인 구현이 가능하고 로얄티가 없어야 한다는 조건이 제시되었다. 2000년 최종 AES 암호알고리즘으로 선정된 Rijndael 암호알고리즘은 암호 키의 길이가 128, 192, 256 비트의 경우와 데이터 블록의 길이가 128, 192, 256 비트의 경우로, 데이터 블록 크기 Nb와 키 크기 Nk의 조합이 가능하다[6].

Hash 알고리즘은 단방향함수로서 임의의 길이를 갖는 메시지를 입력으로 고정된 길이의 Hash값 또는 Hash 코드를 출력하는 함수이다. Hash함수 h는 임의의 길이의 문자열을 고정된 길이를 갖는 n 비트 문자열로 대응시킨다.

또한 Hash함수는 가변 입력값에 대하여 고정된 출력값을 가지는 함수이므로 Hash함수를 처리하기 위해서는 Hash함수 이전에 데이터들에 대한 전처리 과정을 거쳐야 한다. 즉 고정된 정확한 비트 열을 미리 설정한 후 Hash 함수를 구동시켜야 한다.

3. BESA 암호화 알고리즘

기존 대칭형 암호알고리즘은 인증기능이 없기 때문에 별도의 인증기능을 가진 Hash함수와 함께 non-security channel에 데이터를 전송한다. 이러한 결과로 암호문 자체 또는 Hash함수가 오염되었을 경우 이를 검출할 방법이 없기 때문에 상대방에 대한 신뢰성이 보장될 수 없다. 또한 네트워크 환경의 발달로 인하여 사용자 수가 증가됨에 따라 키 관리 및 분배가 용이하지 못하다는 단점을 가지고 있다. 그러나 대칭형 암호알고리즘이 인증기능을 포함하게 된다면 별도의 키 관리가 필요없으므로 키에 관련된 문제점이 해결될 수 있다.

그러므로 본 논문에서는 대칭형 기반 암호화 알고리즘이면서 인증기능을 포함하는 기능을 삽입함으로서 이러한 문제점을 해결하였다. 또한 바이트 연산은 비트 연산에 비하여 비도 유지 및 증대에는 매우 좋은 특성을 나타냈지만 구현상의 어려움, 복잡도 증가 등으로 인하여 오히려 효율성은 떨어지므로 비도 유지를 위하여 Feistel 구조와 SPN(Substitution and Permutation Network) 구조를 가진 라운드 연산을 사용하였다. 그 결과 비도의 증대 및 암/복호화 수행시 매우 간편한 연산을 제공하여 암호시스템의 효율성을 증대시킬 수 있었다. 또한 정해진 라운드 횟수에 대하여 가변적으로 라운드를 결정하기 위하여 그림 1과 같이 데이터 매칭 여부를 탐색하도록 하였다.

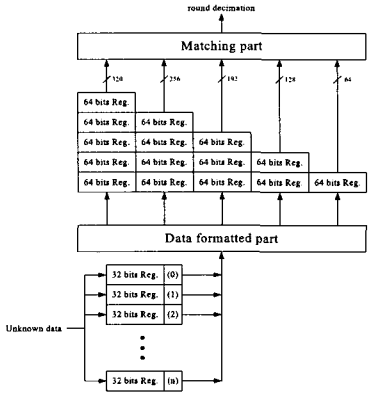


그림 1. 라운드 횟수 결정

본 연구에서 제안한 암호화 알고리즘은 암호화 및 복호화의 동시 수행이 가능하며 암호화를 위해 암호 연산부로 데이터가 유입되기 전 입력데이터는 RD(Round Decimation) 변환을 수행하도록 하였다. RD 변환은 RD-I, RD-II로 구별되며 그림 2와 같다.

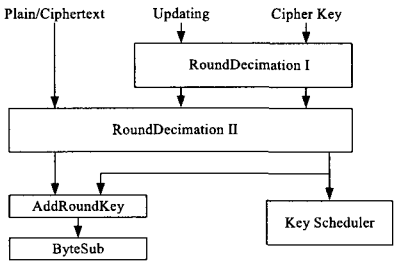


그림 2. RD-I, RD-II 구성

그림 2에서 Updating 정보는 seed 값으로 취급하며 Cipher key는 대칭형 암호알고리즘에서 사용되는 비밀키를 나타낸다. Updating 정보와 비밀키 정보는 RD-I에서 8비트를 기준으로 64비트씩 256비트 데이터들이 단순 bitwise-XOR 연산을 수행한다. 이 변환은 식 (1)과 같고 수행 결과값들 중 일부는 RD-II로, 일부는 키 스케줄러의 입력으로 사용된다.

RD-I 변환은 비선형 S-box를 이용하는 기존 블록암호 알고리즘과는 별개로 선형성을 강조하기 위하여 *modulo* 연산을 수행하며, 선형성을 가지므로 역변환이 가능하다. 이때 선형성을 강조하여 선형동작만을 수행하도록 하면

외부로부터의 공격에 쉽게 노출될 수 있으므로 식 (2)와 같은 아핀(affine) 변환을 $GF(2^8)$상에서 수행하도록 한다.

$$RD-I = Up_8(256) \oplus K_8(256) \quad (1)$$

$$\begin{aligned} b_i &= b_{(i+o)\bmod 8} \oplus b_{(i+e)\bmod 8} \oplus c_i \\ b_j &= b_{(j+o)\bmod 8} \oplus b_{(j+e)\bmod 8} \oplus c_i \end{aligned} \quad (2)$$

RD-II 변환은 평문 또는 암호문과 RD-I의 결과값을 bitwise-XOR 연산을 수행한 후 substitution 블록으로 출력한다. 출력된 값은 64비트씩 입력을 받아 1:1 비율로 치환을 수행하게 된다. 식 (3)은 이와 같은 변환에 대한 식이며, R_{sub0}는 첫 번째 substitution 블록, rd_0는 R_{sub0}의 redundancy이다.

$$\begin{aligned} R_{sub0} &= sub_0 \ \& \ rd_0 \\ R_{sub1} &= 8 \ll R_{sub_0} \ \& \ rd_1 \\ R_{sub2} &= 8 \ll R_{sub_1} \ \& \ rd_2 \\ R_{sub3} &= 8 \ll R_{sub_2} \ \& \ rd_3 \end{aligned} \quad (3)$$

RD 변환을 거치면 암호 연산부에서 실제적인 암호화 과정을 수행하게 되며, 여기에서 암호화 및 복호화가 동시 수행이 가능하다. 이러한 암호 연산부는 AddRound, ByeSub, DiaMat 변환으로 구성한다.

3.1 AddRound 변환

AddRound 변환은 RD로부터 산출된 데이터 R_{sub}와 식 (4)에서 산출된 K_{seed}를 식 (5)와 bitwise-XOR 연산을 수행하여 라운드에 대한 정보를 암호화 과정에 합하는 변환을 수행한다. 이때 연산은 그림 3과 같이 바이트가 기본이 되어 수행된다.

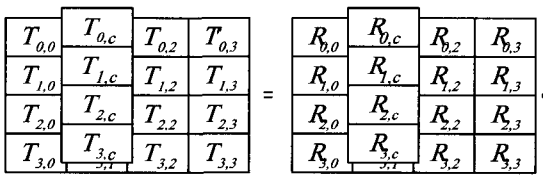


그림 3. AddRound 변환

$$\begin{aligned} K_{seed} &= Nr_{Rd_i\ condition} \otimes rw(RD-I) \\ &= \begin{bmatrix} none \\ Rd_0 \\ Rd_1 \\ Rd_2 \\ Rd_3 \end{bmatrix} \otimes [Up_8(256) \oplus K_8(256)] \end{aligned} \quad (4)$$

$$T_{AR} = R \quad (5)$$

식 (4)에서 $rw(RD-I)$는 식 (2)의 RD-I 변환 결과값에 대하여 redundancy를 구하고 RD-I redundancy에 대한 weight를 구한 값이다. 식 (5)에서 T_{AR}은 AddRound 변환을 의미하며 T_{AR}은 R_{sub_i}와 K_{seed}를 이진합을 이용하여 구한다.

3.2 ByteSub 변환

ByteSub 변환은 바이트들을 기본 연산으로 사용하여 substitution을 수행하는 기능블록이며, w, x, y, z의 기본 바이트 치환 변환 모듈을 이용하여 pseudo-random 변환을 한다. 이때 식 (6)과 같이 w가 기본이 되고 행과 열을 이동하여 x와 y축에 대칭인 x, y가 형성되며 w는 $y=x$ 변환하여 z를 형성한다.

$$\begin{aligned} w &\Leftarrow w \\ x &\Leftarrow w \mid x-axis \\ y &\Leftarrow w \mid y-axis \\ z &\Leftarrow w \mid y=x \end{aligned} \quad (6)$$

그림 4에서와 같이 입력되는 데이터의 크기를 파악하여 w, x, y, z에 대한 각각의 치환을 수행한 후 입력 데이터 크기에 맞는 출력값을 산출한다.

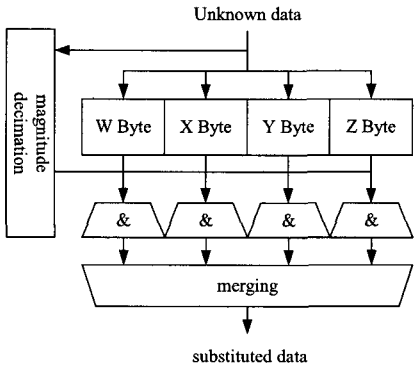


그림 4. ByteSub 변환

3.3 DiaMat 변환

DiaMat 변환은 Diagonal Matrix로서 행과 열을 동시에 이동시키며 연산을 수행하는 변환으로서 암/복호화가 동시에 가능할 수 있도록 하는 연산부이다.

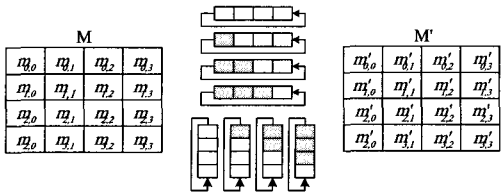


그림 5. DiaMat 변환

그림 5에서와 같이 기본 바이트들의 첫째 행을 제외한 나머지 행들을 각각 서로 다른 offset으로 byte 단위 cyclic rotation을 통해 위치를 교환하는 변환이다. 또한, 전체 바이트들의 row, column들을 $GF(2^8)$상에서의 다항식들 $a(x)$, $b(x)$로 정의하고, rotation 결정횟수인 rotation offset들을 고정된 다항식 $c(x)$로 정의하면 식 (7)과 같은 연산이 가능하며, $c(x)$는 식 (8)과 같다.

$$a(x) \otimes c_r(x) = \mathrm{mod}x^4 + 1 \quad (7)$$
$$b(x) \otimes c_c(x) = \mathrm{mod}x^4 + 1$$

$$c_r(x) = \{03\}x^3 + \{02\}x^2 + \{01\}x + \{01\} \quad (8)$$
$$c_c(x) = \{30\}x^3 + \{20\}x^2 + \{10\}x + \{10\}$$

이러한 암호변환은 BESA 암호알고리즘의 암호화와 복호화를 동시에 수행하도록 함으로서 암/복호화 시간이 절약될 뿐만 아니라 시스템 IP화가 가능하여 다른 플랫폼에 암호시스템을 적용할 경우 시스템의 성능을 향상시킬 수 있다. 또한 비대칭형 암호알고리즘은 네트워크 환경에서 키 관리 및 분배, 인증기능이 강화되었지만, 처리시간이 너무 길고 하드웨어의 구현이 어렵다.

따라서 본 연구에서는 대칭형 기반 암호알고리즘이지만 Hash함수와 MAC, MDC(Manipulation Detection Code)를 혼합하여 사용함으로서 기존 인증기능을 강화하였으며, 이러한 특징을 가진 알고리즘의 전체 블록도는 그림 6과 같다. 그림 6에서 입력 기지 데이터와 Updating 정보, 비밀키 정보를 RD-I, RD-II에서 전처리를 수행하고 AddRound, ByteSub, DiaMat 블록에서 암호화를 수행한다. 이때 전처리 연산은 알고리즘이 수행할 라운드 정보를 만들어내며 암호 연산부는 라운드 정보에 의하여 9회까지는 기본적인 라운드를 수행하고 10회에서 14회까지 설정된 라운드 횟수에 의하여 다양한 라운드 연산을 수행한다.

또한 RD-I, RD-II 연산부와 암호 연산부등과 더불어 MDC, MAC에 해당하는 Hash값을 산출한다. 이러한 Hash값의 결과는 별도의 인증채널을 확보할 필요 없이 BESA 출력만을 이용하여 인증기능을 수행하게 된다.

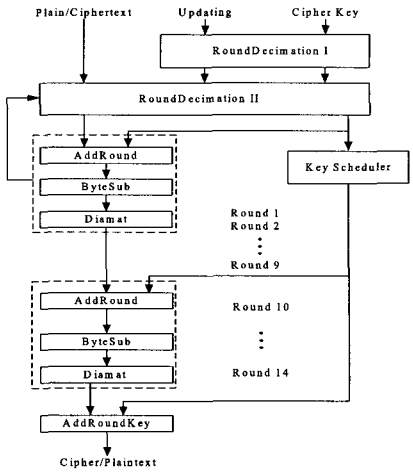


그림 6. BESA 암호알고리즘

4. BESA 암호시스템 설계 및 분석

암호알고리즘은 구현방법에 따라 시스템 성능이 매우 달라진다. 일반적으로 하드웨어 구현방식을 선택하는 주요한 이유는 처리시간이다. 대칭형인 경우 encryption rate은 100Mbps 정도 하드웨어가 빠르고 비대칭인 경우 10msec 정도 우수한 특성을 가진다. 또한, 하드웨어 기반 암호시스템은 물리적인 보안성을 제공함으로서 네트워크 환경에서 외부로부터의 Spyware등과 같은 해킹방법으로부터 매우 안전한 안전성을 제공한다. 그러므로 평균적으로 소프트웨어에 비하여 하드웨어 기반이 속도 면에서 10^6정도 우수하다. 본 연구에서 제안한 BESA 암호알고리즘은 reconfigurable하며 flexibility한 특성을 가질 수 있도록 시스템을 설계하였고, 구현상의 용이성 및 IP의 재사용을 위해 3개의 블록으로 구별하여 설계하였다.

4.1 전처리 및 후처리부 설계

암호 연산부가 동작하기 위해서는 암호 연산부로 유입되는 기지 입력에 대한 formatting 과정이 필수적이다. 그러므로 전처리부에서는 입력정보에 대하여 데이터들을 32비트를 하나의 바이트로 정의하여 32비트가 최소단위로 동작하며, RD-I 변환과 RD-II 변환에 대한 기능블록이 포함된다. 이와 동시에 전처리부는 SSP를 수행할 수 있도록 하였으며 이로 인하여 인증기능에 관련된 무결성 가지를 지원할 수 있도록 하였다.

그림 7은 RD-I 블록으로 Updating 정보와 비밀키 정보를 modulo-2 연산을 수행하는 블록이며, 이 연산의 결과값들 중 일부는 RD-II로, 일부는 키 스케줄러의 입력으로

사용된다. RD-I 블록은 비선형 함수인 S-box를 사용하여 trapdoor로부터 안전성을 보장받기 보다는 선형성을 강조하고, 외부로부터의 공격에 쉽게 노출될 수 있는 단점을 보완하기 위하여 아핀(affine) 변환을 $GF(2^8)$상에서 수행하도록 한다.

그림 8은 RD-II 기능블록이며, RD-I의 결과값을 bitwise-XOR 연산을 수행한 후 Substitution 블록으로 출력한다. 출력된 값은 64비트씩 입력을 받아 64비트씩 1:1 비율로 치환을 수행하게 된다.

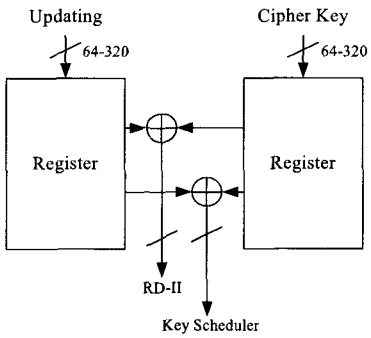


그림 7. RD-I 기능블록

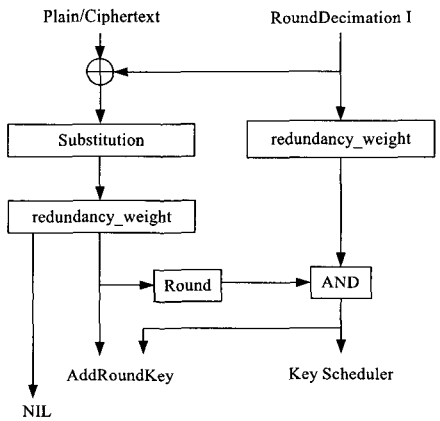


그림 8. RD-II 기능블록

BESA 암호시스템의 최종단에 구성되는 후처리부에서는 64, 128, 192, 256비트 등과 같은 가변 데이터들을 32비트의 고정된 데이터로 변환하는 기능과 동시에 암호화 연산부의 데이터와 인증데이터를 combining하는 기능을 수행한다.

4.2 Linearity 프로세서 설계

BESA 암호시스템에서 실제적인 암호화를 수행하는 부분인 Linearity 프로세서 블록인 암호화 연산부는 암호화와 복호화를 동시에 수행할 수 있도록 역변환이 가능한 선형성을 강조한 블록이다.

AddRound 변환은 데이터와 라운드 정보를 이용하여 암호화를 수행하는 블록으로 그림 9와 같이 RD-I에 대한 redundancy의 weight값과 라운드 횟수를 결정짓는 조건인 R_d에 대하여 modulo 곱을 수행한 후 R_{sub}와 bitwise-XOR 연산을 수행한다. 또한, 하드웨어 구현시 시스템 효율성을 증가시키기 위하여 곱셈부분은 Montgomery 알고리즘을 사용하였다.

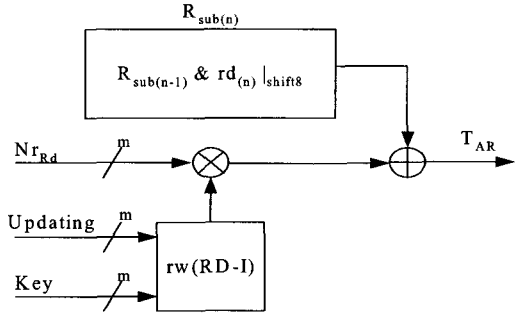


그림 9. AddRound 변환블록

ByteSub 변환을 수행하기 위해 pseudo-random 변환을 이용하여 w, x, y, z의 기본 바이트를 치환한다. 이때 사용되는 RNG(Random Number Generator)는 식 (9)와 같은 생성다항식으로 rng_c, rng_x을 사용하며, 변환블록은 그림 10과 같다. 또한, 암/복호화 기능이 하나의 시스템에서 동작하도록 Feistel 구조와 같은 대칭성을 가지도록 설계하였다.

$$rng_c(x) = x^{16} + x^{13} + x^{12} + x^{11} + x^7 + x^6 + x^5 + x^4 + 1 \quad (9)$$
$$rng_x(x) = x^{16} + x^{14} + x^{10} + x^9 + x^8 + x^6 + 1$$

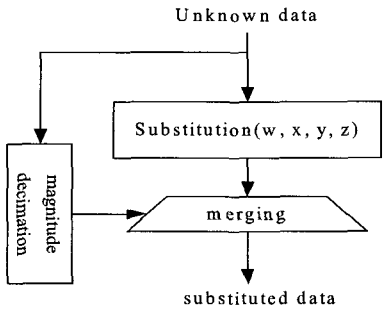


그림 10. ByteSub 변환블록

DiaMat 변환블록은 Diagonal Matrix 형태로 행과 열을 동시에 rotation 시키는 연산을 수행하며, 암호화/복호화가 동시에 가능한 연산부이다. DiaMat 변환은 rotation offset을 행, 열의 위치에 따라 C0~C3까지 변화시키면서 회전시킨다. 이로 인하여 행과 열의 데이터들은 rotation 되고 각각의 데이터들은 행과 열에 대하여 대각방향으로 데이터의 천이가 발생되며, 그림 11과 같다.

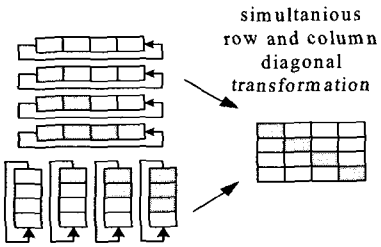


그림 11. DiaMat 변환블록

4.3 BESA 암호프로세서 설계

전/후처리부, Linearity 프로세서를 하나의 시스템 블록으로 구현하였다. 전/후처리부에서 데이터들에 대한 크기는 32 비트로 한정되어 입력되거나 출력되며, 전/후처리부, Linearity 프로세서, 키 생성부 등의 블록들을 제어해 주는 블록은 BESA 암호알고리즘과 별개로 암호시스템 구현시 요구되는 제어신호를 발생한다.

그림 12는 구현된 암호시스템에 대한 모의실험 결과이다. 입력으로 사용된 Updating 정보인 XS 16비트, 비밀키 정보인 XS_mode 32비트, XS와 XS_mode를 입력으로 받는 RD-I 정보인 source, RD-II 정보인 destination과 128비트의 크기를 가지는 평문이 있다. XS는 "1101", XS_mode는 "A0FA09BA"를 인가하였으며 source와 destination은 모두 "000000"의 값으로 설정하였다. 이러한 조건하에서 주어진 평문 128비트에 대해 제어신호들과 BESA 암호알고리즘에 의하여 암호문이 발생되었다.

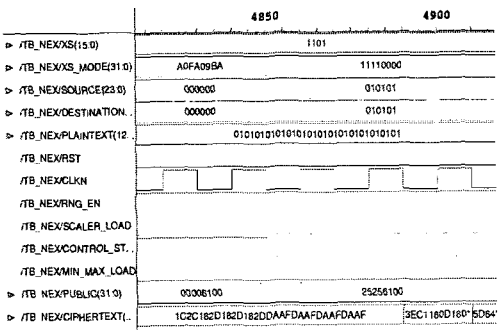


그림 12. BESA 암호시스템 모의실험

기존 암호알고리즘의 경우, 입력조건이 동일하면 일정한 키 패턴과 평문에 대하여 일정한 암호문이 형성된다. 그러므로 암호문에 대한 패턴을 분석하면 키 정보를 얻을 수 있으며 얻어진 키 정보를 바탕으로 평문을 찾아내게 된다. 그러나 본 논문에서 제안한 BESA 암호알고리즘에 의한 BESA 암호시스템의 모의실험 결과, 키 정보가 포함된 라운드 횟수조절로 인하여 평문의 일정패턴에 대하여 상호 독립적인 암호문이 생성됨을 확인하였다.

4.4 시스템 분석

제안된 BESA 알고리즘은 대칭형 블록암호알고리즘으로서 기존 AES의 Rijndael, Twofish 암호알고리즘 등에 비해 네트워크 환경에 보다 잘 적응하며 구현하기 용이하게 만든 알고리즘이며, 기존 알고리즘과의 비교는 표 1과 같다.

표 1. 알고리즘 비교

	기존 알고리즘	제안된 알고리즘
암호 원리	Feistel, SPN	SPN, sharing
라운드횟수	고정	가변
연산자	XOR, Non-linearity, Modular-2, Addition	XOR, XNOR, Modular-2, Linearity, Addition
인 증	어려움	용이함
처리속도	빠름	빠름
키 관리자	dealer	dealer, joiner
키 관리수	많음	적음

5. 결 론

기존 암호알고리즘의 경우 암/복호 과정에서 미리 설정된 표에 의한 라운드를 수행할 뿐만 아니라 새로운 키 설정과 동시에 수행되므로 시스템의 복잡성을 증가시키고 신호의 병목현상 및 이로 인한 처리시간의 지연을 가져왔다. 따라서 본 논문에서는 라운드 횟수를 입력되는 데이터량의 크기에 의하여 결정하고 TCP/IP 프로토콜에 BESA SSP 암호프로토콜을 적용하여 보안기능을 수행하도록 하였다. 또한 네트워킹 환경에서 N:N 다자간 정보공유를 위하여 인증기능을 가지도록 함으로서 BESA 암호알고리즘이 TCP/IP 프로토콜을 기반으로 동작되는 네트워크 환경에서 많은 사용자들을 수용할 수 있도록 하였다. 그러므로 제안된 알고리즘은 네트워크 환경 특히,

유무선망의 다자간 보안통신에 있어 매우 적합함으로 향후 멀티미디어 응용서비스를 요구하는 플랫폼에 유용할 것이다.

참 고 문 헌

[1] B. Schneier, "Applied Cryptography : Protocols, Algorithms, and Source Code in C", John Wiley & Sons, Inc., New York, USA, 1994.

[2] L. Brown and J. Seberry, "Key scheduling in DES type Cryptosystems", abstract of AUSCRYPT'90, 1990.

[3] L. Brown and J. Seberry, "On the Design of Permutation P in Des Type Cryptosystem", Abstract of AUSCRYPT'90, 1990.

[4] M. Kimberley, "Comparison of Two Statistical Tests for Keystream Sequences", Electronics Letters, Vol. 23, No. 8, pp. 365-366, April 1987.

[5] R. Rueppel, "Stream Ciphers", Contemporary Cryptology : The science of Infor. Integrity, New York, IEEE Pres, pp. 65-134, 1991.

[6] D. R. Stinson, "Cryptography Theory and Practice", Chapman & Hall, CRC, 2002.

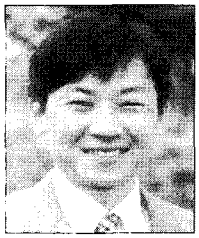
[7] E. Biham and A. Shamir, "Differential Cryptanalysis of DES-like Cryptosystems", Journal of CRYPTOLOGY Vol. 4 No. 1, 1991.

박 형 근(Hyoung-Keun Park) [정회원]

- 1993년 2월 : 원광대학교 전자공학과(공학사)
- 1995년 2월 : 원광대학교 대학원 전자공학과(공학석사)
- 2000년 2월 : 원광대학교 대학원 전자공학과(공학박사)
- 2005년 3월 ~ 현재 : 남서울대학교 전자공학과 전임강사

<관심분야>
반도체회로설계, 고주파 통신용 회로, 마이크로프로세서 응용

이 승 대(Seung-Dae Lee) [정회원]



- 1990년 2월 : 단국대학교 전자공학과 (공학사)
- 1992년 2월 : 단국대학교 대학원 통신공학전공 (공학석사)
- 1999년 8월 : 단국대학교 대학원 통신공학전공(공학박사)
- 1995년 4월 ~ 현재 : 남서울대학교 전자공학과 부교수

<관심분야>
마이크로파 회로해석 및 설계, RF시스템 모델링, 이동통신시스템

김 선 엽(Sun-Youb Kim) [정회원]

- 1993년 2월 : 원광대학교 전자공학과(공학사)
- 1995년 2월 : 원광대학교 대학원 전자공학과(공학석사)
- 2001년 2월 : 원광대학교 대학원 전자공학과(공학박사)
- 2006년 9월 ~ 현재 : 남서울대학교 정보통신공학과 전임강사

<관심분야>
초고주파 통신용 회로, 광통신응용, 이동통신시스템