2001년 한국정보처리학회 추계 학술발표논문집 제8권 제2호

Pkc128 블록 암호 알고리즘

김 길 호*, 조 경 연*
*부경대학교 전자컴퓨터정보통신공학부
e-mail:vnlqpcdd@hanmail.net

Pkc128 block cipher algorithm

Gil-Ho Kim*, Gyeong-Yeon Cho*
*Dept. of Electronics, Computer and Telecommunication Engineering, Pukyong National University

요 약

본 논문에서는 데이터 의존 회전 기법과 프로그램 셀룰라 오토마타 기법을 사용한 블록 암호 알고리즘인 가칭 Pkc128(PuKyong Code 128) 암호 알고리즘을 제안한다. 제안한 암호 알고리즘의 블록 크기는 128 비트이고, 키의 크기는 128 비트 이상 가변이며 Feistel Network 구조를 취하였다. 제안한 알고리즘의 안전성을 검정하기 위하여 출력 스트림에 대한 통계적 검정을 실시하였다. 그 결과 16 회전 시에 모든 검정과정을 통과하여 제안된 알고리즘이 통계적으로 안전함을 확인하였다.

1. 서론

1977년 IBM이 개발하고, 미국 정부에 의해 수정되어 미국 정부의 암호 표준으로 채택된 DES(Data Encryption Standard)는 최근까지 널리 사용되고 있는 암호 알고리즘이다. 그러나 최근 컴퓨터 계산 능력과 암호 해독 기술의 발달로 인해 DES가 해독되는 등 보안, 관리상의 취약점 및 문제점이 발견되었다. 이에 미국 국립 표준 기술연구소(NIST, National Institute Standards & Technology)에서는 새로운 표준 블록 암호 알고리즘을 선정하기 위해 1997년 초 새로운 표준 암호 알고리즘(AES, Advanced Encryption Standard)선정 프로젝트를 발표했다. NIST는 차세대 표준 암호 알고리즘으로 128비트 블록 암호 알고리즘, 다양한 길이의 키(128, 192, 256비트)를 사용 할 수 있고, 취약키를 갖지 않으며, 소프트웨어와 하드웨어 상에서 효율적이며, 스마트 카드 상에서도 동작 할 수 있는 알고리즘 등의 기준을 제시했다. AES 선정 프로젝트는 전 세계적으로 후보 알고리즘을 공모하였고, 여러 차례 평가에 의해 5개의 후보 알고리즘(Rijndael, Twofish, MARS, RC6, Serpent)을 선정하였다. 5개의 후보 중 NIST 자체평가와 전 세계적인 공개 검증을 통해 지난 2000년 10월에 최종적으로 Rijndael이 선정되었다[1].

그런데 Rijndael은 소프트웨어 및 하드웨어 구현 시에 속도가 느린 문제점을 가지고 있다. 한편 RC6는 데이터 의존 회전 기법과 곱셈을 사용한 알고리즘으로 하드웨어로 구현 시에 속도가 뛰어난 장점을 가지지만, 곱셈기 구현에 많은 전자회로를 필요로 한다[2,3]. 따라서 이들 알고리즘들은 스마트카드 등 소규모 하드웨어로 암호 기능을 구현하여야 하는 응용분야에 적합하지 않다.

이러한 문제점을 해결하기 위해서 본 논문에서는 데이터 의존 회전 기법과 프로그램 셀룰라 오토마타 [4] 기법을 사용한 가칭 Pkc128(PuKyong Code 128) 블록 암호 알고리즘을 제안한다.

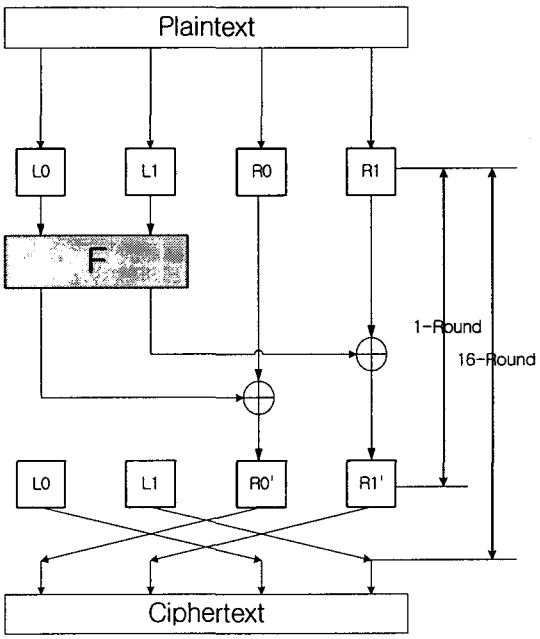
제안한 Pkc128은 Feistel Network 구조를 취하며 128 비트 블록 크기를 가진다. 키는 128 비트 이상 가변 길이를 가지며, 본 논문에서는 128 비트 키로 가정하였다. 제안한 알고리즘의 안전성을 검정하기 위하여 출력 스트림에 대해서 통계적 검정을 실시하였다. 테스트는 FIPS140-1에서 제안한 빈도, 계열, 포커, 런 그리고 자기상관 검정을 수행하였다[5]. 16 회전 시에 검정 결과 모든 검정 과정을 통과하여 알고리즘이 안전함을 확인하였다.

2. Pkc128 블록 암호 알고리즘

Pkc128는 128 비트 대칭키 블록 암호 알고리즘으로 Feistel network 구조이다. 입출력의 크기는 128 비트이며 각 회전마다 2 개의 확장키를 사용하고, 회전 전과 후에 각 2개의 확장키를 사용하므로 전체 확장키 수는 '2 * 회전수 + 2 + 2' 개가 필요하다. 128 비트 평문은 4개의 32 비트 블록으로 나누어 2 개의 블록만 각 회전에서 변경하고 변경하지 않은 2 개의 블록과 논리적 배타합을 취한다.

2.1 암호, 복호 알고리즘

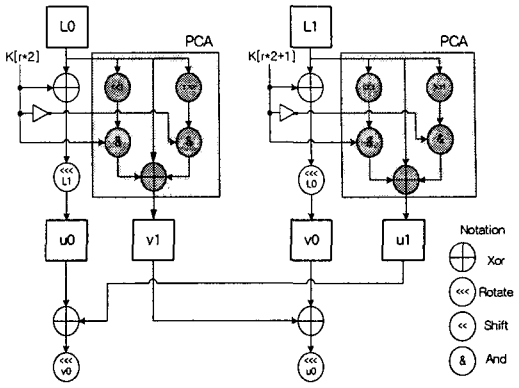
Pkc128 블록 암호 알고리즘의 암호화 과정 한 회전을 (그림 1)에 보인다.



(그림 1) 암호화 과정

128 비트 평문을 입력받아 32 비트 4개의 블록으로 나눈 다음, 레지스터 L0, L1, R0, R1에 각각 저장한다. 한 회전에서 R0와 R1은 F함수를 거쳐온 L0와 L1의 값과 배타적 논리합을 수행하여 값이 변하게되고, L0와 L1은 F함수를 거쳐 R0와 R1을 변화시키기 위한 피 연산자로 사용된다. 한 회전을 수행하고 다음 회전으로 넘어갈 때, L0와 L1은 R0와 R1과 서로 바뀌어서 입력으로 들어가게 된다.

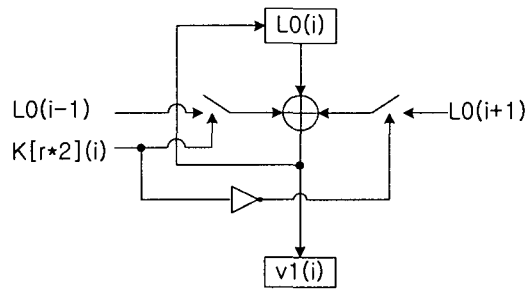
(그림 2)에 F 함수를 보인다.



(그림 2) F 함수

F 함수는 평문 L0(L1)을 확장키와 배타적 논리합을 취하고, 그 결과를 L0(L1)에 따른 데이터 의존 회전을 수행하여 u0(v0)를 생성한다. 동시에 L0(L1)을 확장키에 의하여 프로그램 셀룰라 오토마타를 수행하여 v1(u1)을 생성한다. 생성한 u0는 u1과, v0는 v1과 각각 배타적 논리합을 수행한다.

그림-3에 프로그램 셀룰라 오토마타 과정을 보인다.



(그림 3) PCA(Programmable Cellular Automata)

복호화 과정은 Feistel network의 기본 성질에 따라서 암호 알고리즘과 동일하며 단지 확장키를 역순으로 적용한다.

2.2 확장키 생성

키는 128 비트이상 가변 길이를 사용한다. 본 논문에서는 RC6에서 사용하는 확장키 알고리즘을 그대로 적용하였다. 표-1에 16 회전시의 확장키 알고리즘을 보인다.

<표 1> 확장키 알고리즘

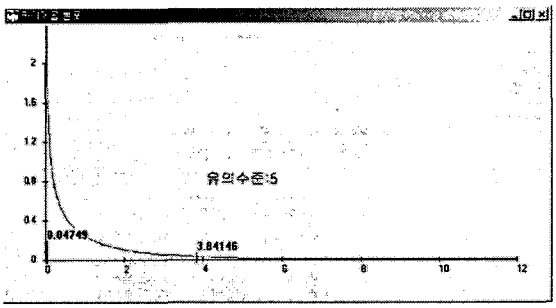
```
/* initialize rkey[36] */
for(rkey[0]=P, i=1; i<36; i++)
  rkey[i] = rkey[i-1] + Q;
/* data-dependent-rotation */
for(A=B=i=j=k=0; k<108; k++, j=j&3)
  {
  A=rkey[i]=ROTL(rkey[i]+A+B,3);
  B=key[j]=ROTL(key[j]+A+B,(A+B));
  i++;
  j++;
  if(i==36) i=0;
  }
```

3. 구현 및 통계 검정

제안한 Pkc 128 블록 암호 알고리즘을 Borland C++ 3.1을 사용하여 소프트웨어로 구현하고, 다음과 같은 환경에서 통계 검정을 수행하였다.

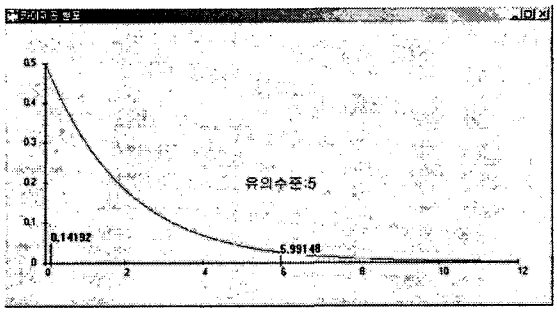
System : Pentium II 366MHz PC
OS : Windows 98
Memory : 192Mbyte
Sample size : 4,378,424 bit

통계 검정은 FIPS(Federal Information Processing Standards) 140-1에서 제안한 빈도 검정(Frequence test), 계열 검정(Serial test), 포커 검정(Poker test), 런 검정(Run test) 그리고 자기상관 검정(Autocorrelation test)의 5 가지 모두를 시행했다.



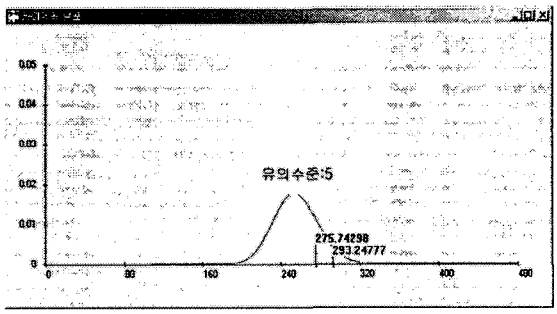
(그림 4) 빈도 검정 결과

(그림 4)는 대상 수열이 0과 1의 수가 고르게 분포하는 지를 검정하는 빈도 검정 결과를 보여 주고 있다.



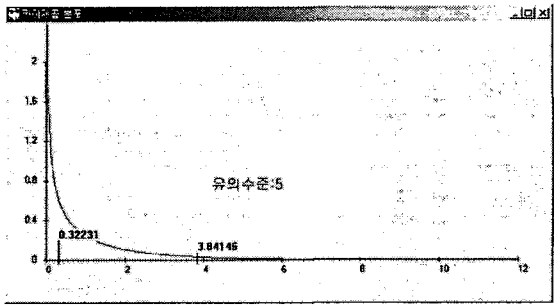
(그림 5) 계열 검정 결과

(그림 5)는 2진 수열에서 한 비트가 그 다음 비트로 전이되어 가는 과정이 랜덤(random) 한가를 검정하는 계열 검정 결과를 보여 주고 있다.



(그림 6) 포커 검정 결과

(그림 6)은 주어진 스트림에서 임의의 m-bit 패턴이 동일하게 나타나는지를 검정하는 포커 검정 결과를 보여 주고 있다.



(그림 7) 자기상관 검정 결과

(그림 7)은 2진 비트 스트림을 임의의 비트만큼 전이시켜 생성한 비트 스트림과 상관관계를 조사하

는 자기상관 검정 결과를 보여 주고 있다.

런 검정은 비트 스트림에서 0이나 1이 연속하여 나타나는 부분을 조사하는 검정이다.

<표 2>에 검정 결과를 요약하여 나타낸다.

<표 2> 통계 검정 결과

test	Threshold value	Result
Frequence	3.84146	0.04749
Serial	5.99148	0.14192
Poker	293.24777	275.74298
Run	24.995	13.610633
Autocorrelation	3.841455338	0.32231
Total bit : 4,378,424		

본 논문에서 제안한 Pkc128 블록 암호 알고리즘은 FIPS 140-1 통계 검정을 전부 통과하여 안전성이 있는 것으로 확인되었다.

4. 결론

본 논문에서는 데이터 의존 회전 기법과 프로그램 셀룰라 오토마타 기법을 사용한 대칭 키 128 비트 블록 알고리즘인 가칭 Pkc128(PuKyong Code 128) 암호 알고리즘을 제안하였다.

제안한 Pkc128 알고리즘의 안전성 검정을 위하여 IBM PC 상에서 FIPS140-1에서 제안한 빈도 검정, 계열 검정, 포커 검정, 런 검정 그리고 자기상관 검정의 5 가지 통계 검정을 수행하였다. 그 결과 모든 통계 검정을 통과하여 Pkc128 암호 알고리즘이 통계적으로 안전함을 확인하였다.

제안한 Pkc128 알고리즘은 하드웨어에 특히 적합한 알고리즘으로 향후 하드웨어로 제작하여 검증하는 것이 연구 과제로 남아 있다.

참고문헌

[1] NIST, "Advanced Encryption Standard Development Effort." http://csrc.nist.gov/encryption/aes.

[2] M. Riaz and H. M. Heys, "The FPGA Implementation of the RC6 and CAST-256 Encryption Algorithm," Proceedings of the 1999 IEEE Canadian Conference on Electrical and Computer Engineering, pp.367-372, May 9-12 1999

[3] Scott Contini, Ronald L. Rivest, M. J. B. Robshaw and Y. L. Yin, "The Security of the RC6 Block Cipher," RSA laboratories report, Aug. 1998

[4] S. Wolfram, "Cryptography with Cellular Automata," Advances in Cryptology: Crypto '85 Proceedings, Lecture Note in Computer Science, vol.218, pp.429-432 (Springer-Verlag, 1986).

[5] FIPS180-1, *Secure hash standard*, Federal Information Processing Standards Publication 180-1, U. S. Department of Commerce/Nist, 1995