Historia Mathematica Vol. 11, No. 2, 1998, 8-16

암호학의 발달과 최근 동향*

강남대학교 수학과 **이희정**

Abstract

We survey the development of cryptography from the ancient to the present as a mathematical point of view, and recent results of public key cryptosystems.

0. 암호학과 그의 발달

0.1 암호학이란

암호학(cryptology)은 암호 구현법(cryptography)과 암호 해독법(cryptoanalysis), 두 부분으로 나눌 수 있는데 송신자가 허가되지 않은 제3자는 해독하기 어려운 형태로 평문을 변형하고(암호화) 또 이를 수신자는 해독 가능한 형태로 변환하는(복호화) 방법들을 암호 구현법이라고 하고 허가 받지 않은 제3자가 가로챈 정보를 해독하려는 모든 방법을 암호 해독법이라고 한다. 이 논문에서는 암호 해독법에 관한 것은 언급을 하지 않고 암호 구현법을 중심으로 발달과정을 살펴보려고 한다. 암호학은 수학적 이론이 기초가 되었으나 단순히 수학만으로는 가능하지가 않고 컴퓨터의 발달로 인하여 전산학이 중요한 역할을 하게 되었으며 통신수단의 발달 등으로 전자공학, 통계학 등 여러 분야가 공조함으로써 가능한 학문으로 발전하였다. 본 논문에서는 암호학의 이론적 바탕이 되는 수학 이론을 중심으로 암호학의 발달과정을 살펴보고 이러한 과정을 통하여 암호학에서의 수학 역할의 발전을 기대해본다.

0.2 암호학의 발달 단계

암호학의 발달은 시대적으로 크게 3단계로 구분할 수 있다.

제1단계는 고대부터 19세기말까지라고 간주한다.

가장 오래된 암호방식으로는 기원전 400년경 고대 희랍인들이 사용한 Scytale 암호 시스템으로 곤봉에 종이(papyrus)를 감아 평문을 횡으로 쓴 다음 종이를 풀면 평문의 각 문자는

* 본 논문은 1998년 강남대학교 교내연구비 지원에 의하여 수행되었음.

재배치되어 평문의 내용을 알 수 없게 된다. 수신자는 송신자가 사용한 똑같은 크기의 곤봉에 종이를 감고 횡으로 읽으면 평문을 얻을 수 있게 된다. 단순한 문자대입을 통하여 최초로 사용된 암호 구현법은 카이사르(Julius Caesar)에 의한 것으로 영어의 알파벳을 0에서 25까지 치환한 후, 메시지 P를 $C \equiv P+3 \bmod 26$으로 변경시켜서 보낸 것이다. 일반적으로 언어의 문자나 기호를 숫자로 치환한 후, $(a, n)=1$, $0 \le b \le (n-1)$에 대한 아핀변환(affine transformation) $C \equiv aP+b \bmod n$을 이용하여 암호화시키고 수신자는 $a^{-1}(C-b) \equiv P \bmod n$을 통하여 암호문을 복호화하는 암호체계를 이용한 시대이다. $a^{-1} \bmod n$은 a와 n이 서로 소이므로 존재한다. 이처럼 최초의 암호 체계는 합동식에 그 이론적 근거를 두고 있다. 그러나 이와 같은 암호 체계는 통계적으로 나타나는 평문의 특성상 문자의 빈도수를 살펴봄으로써 a와 b를 찾아낼 수 있기 때문에 안전하지가 못하다. 제1단계에서는 전달매체가 주로 사람이거나 비둘기여서 암호문의 누출이 상대적으로 어려웠다. 그러나 사용된 암호 구현법은 비밀키 암호 구현법으로 각자 키의 보관에 신경을 써야 했다.

제2단계는 20세기 초 전신기가 일반화된 시점으로부터 40년대 말까지로 분류된다. 평문을 적당한 길이로 나누어 'block'으로 암호화하는 polygraphic ciphers를 고안해냈는데 이는 1930년경 Hill에 의해서 만들어졌다.

$$
\begin{aligned}
C_1 &\equiv a_{11}P_1 + a_{12}P_2 + \cdots + a_{1r}P_r \bmod n \\
C_2 &\equiv a_{21}P_1 + a_{22}P_2 + \cdots + a_{2r}P_r \bmod n \\
&\vdots \qquad\qquad\quad \vdots \\
C_r &\equiv a_{r1}P_1 + a_{r2}P_2 + \cdots + a_{rr}P_r \bmod n
\end{aligned}
$$

즉, $C \equiv AP \bmod n$이다. A는 $r \times r$ 행렬이고, $\det A \neq 0$이며 $C=(C_1, C_2, \ldots, C_r)^t$, $P=(P_1, P_2, \ldots, P_r)^t$이다. 복호화하기 위해서는 $A^{-1}C \equiv P \bmod n$을 하면 된다. 이와 같이 역행렬이 존재하는 행렬을 찾아서 암호화하여야 한다. 그러나 행렬을 이용하여 문자 하나 하나가 아닌 다발로 암호화하여도 그런 것들의 빈도수를 살펴봄으로써 암호 해독이 가능하다. 근대 암호학의 기초가 된 논문은 1920년 Freidman이 발표한 "일치 반복률과 암호응용"과 C.E. Shannon의 "비밀 시스템의 통신이론" 등이다. 근대 암호학에서는 주로 복잡한 기계를 이용하여 암호 알고리즘을 실현하였는데 암호 해독을 위해서는 엄청난 계산량이 요구되므로 그 당시로는 안전한 암호 구현법이었다. 근대 암호학의 발달은 두 차례의 세계대전이 계기가 되었다. 제2차 세계대전 중 암호 시스템을 공격하기 위해서 초기 컴퓨터의 일종인 Colossus가 개발되었다는 사실로 미루어 컴퓨터의 발달이 암호 해독과 관련이 있음을 엿볼 수 있다.

제3단계는 1940년대 말 C.E. Shannon이 "암호학의 수학적 배경"이란 논문을 발표한 시점을 현대 암호학의 시점으로 보고 있으나 전자 산업의 획기적인 발달로 인한 각종 이론을 실현 가능하게 된 70년대 초를 실질적인 현대 암호학의 시점으로 간주하고 있다. 현대 암호학은 수학뿐만이 아니라 전자계산학, 전자공학, 통계학 등 여러 분야와 관련을 맺고 있다. 현대 암호학은 관용 암호 시스템과 공개키 암호 시스템으로 나눌 수 있는데 현재 가장 널리 사용되고 있는 관용 암호 시스템은 DES(Data Encryption Standard)로 1997년 미국 상무성 표준국(NBS)에서 미국 표준 암호 알고리즘으로 채택한 것이다. DES가 나오기까지는

Feistel 알고리즘, 1975년 H. Feistel이 설계한 NDS(New Data Seal), Shannon의 이론을 최초로 적용한 1973년 Feistel이 설계한 Lucifer 시스템 등이 있었다. DES는 블록 암호 시스템으로 블록 암호 시스템이란 고정된 크기의 입력 블록을 고정된 출력 블록으로 변형하는 암호 알고리즘에 의해 암호화 및 복호화를 하는 암호 시스템이다. DES를 실현하는 중요한 기법은 치환, 대치, 그리고 키 스케줄이다. 그 외 FEAL(Fast Data Encipherment Algorithm)은 1987년 일본 NTT에서 DES를 개량하여 설계한 알고리즘으로 Feistel 알고리즘에 근간을 두고 있으며 1988년 시험용이 생산된 이후 표준화를 시행중이다.

공개키 암호 구현법은 기존의 관용 암호 시스템이 비밀키를 보관해야 하는 어려움이 있고 또한 다자간의 정보를 공유하기 위해서는 너무나 많은 비밀키를 유지해야 하는 어려움을 해소하기 위해서 개발해 낸 암호 구현법이다. 이는 암호화 과정에 사용될 키를 공개하고 단지 복호화할 키만을 각자가 비밀리에 갖고 있도록 하는 암호 구현법으로 기존의 암호 체계와는 획기적으로 발전된 상태라고 할 수 있다. 미래의 암호 체계란 공개키 암호 시스템을 일컫는다고 해도 과언이 아닐 것이다. 이는 함정이 있는 일방향 함수(Trapdoor One Way Function)에 이론적 근거를 두고 있다. 일방향 함수(One Way Function)란 한쪽 방향은 구하기 쉬우나 반대 방향은 찾기가 어려운 함수를 말한다. 그러나 적절한 방법이 있어서 허가된 자는 반대 방향의 값도 쉽게 찾을 수 있는 것을 함정이 있는 일방향 함수라고 한다. 즉, 허가된 통신자들은 암호 구현과 해독이 쉬우나 제3자(intruder)는 설령 방법은 알아도 값을 찾아내기 어렵도록 하는 것을 뜻한다. 공개키 암호 구현의 효율성과 안전성을 향상시키기 위한 노력이 활발히 이루어지고 있는데 수학자들의 입장에서는 얼마나 효율적인 함정이 있는 일방향 함수를 찾아낼 수 있는가에 노력을 기울여야 할 것이다. 물론 수학 외의 여러 분야에서도 효율성과 안전성을 높이기 위해서 빠른 알고리즘과 통신수단 등을 개발하고 있다. 예를 들어 빠른 곱셈을 수행하기 위해서 유한체상에서의 optimal normal bases를 찾거나 optimal이 존재하지 않을 경우에 low complexity bases를 찾는 연구, 또 빠른 역원계산에 관한 연구 등 효율성을 높이려는 연구들이 활발히 연구되고 있고 하드웨어상의 개발도 활발하며, 최근에 최고의 관심을 끌고 있는 타원곡선상의 효과적인 수행을 위한 여러 계산상의 연구도 활발히 이루어지고 있다. 공개키 암호 구현법은 소인수분해의 어려움이나 이산대수문제 등 여러 이론들이 바탕이 되었고 최근에는 타원곡선과 격자줄임(lattice reduction) 등을 이용한 연구들이 활발히 진행되고 있다. 이러한 공개키 암호 구현법들을 실질적으로 '현대 암호학'이라고 말할 수 있다.

1. 공개키 암호 구현법의 발달과정

1.1 이산대수문제를 근거한 암호 구현법

공개키 암호 구현법은 1976년 W. Diffie와 M.E. Hellman이 "New Directions in Cryptography"에서 비밀키를 공유하기 위해서 공개키 암호 시스템이란 개념을 최초로 제시하면서 나타났다. Diffie-Hellman의 키 분배방식은 이산대수문제를 이용하여 사용하는 방식으로 암호 구현법이라기보다는 비밀키의 공유를 위해서 제안해낸 방법이다. 이산대수문제란

유한체 GF(p)상에서(p:소수), g를 원시근이라 할 때 $g^x \equiv y \bmod p$에서 x를 알면 y를 구하기 쉽지만 y를 알고 $x = \log_g y$를 구하기는 어렵다는 데 그 근거를 둔다. 이산대수문제는 Diffie-Hellman의 키 분배 방식, ElGamal 암호 시스템, The Massey-Omura 암호 시스템 등에 활용되었는데 Diffie-Hellman의 키 분배방식은 p와 g를 공개하고 사용자 A, B가 각각 a와 b를 비밀키라고 했을 때 A는 g^a, B는 g^b를 공개한다. 그러면 A와 B는 $K=(g^a)^b=(g^b)^a=g^{ab} \bmod p$를 공유하게 된다. 이때 a나 b를 모르고 K를 찾아내는 것은 쉬운 일이 아니다. ElGamal은 Diffie-Hellman의 키 분배방식을 이용하여 공개키 암호 구현법을 제안하였는데 A는 유한체상의 한 원소, g(generator가 아니어도 됨)와 $0<a<q-1$인 임의의 원소 a를 선택하여 a는 비밀키로 보관하고 g^a를 공개한다. 메시지를 A에게 보내고자 하는 B는 임의의 k를 선택하여 g^k를 계산한 후 메시지 P에 $(g^a)^k$를 곱한 후 g^k와 $P(g^a)^k$를 보낸다. A는 보내온 g^k와 a를 이용하여 $(g^k)^a$를 구한 후 $P(g^a)^k$에서 $(g^k)^a$로 나누어 메시지를 복호화한다. Massey-Omura 암호 시스템은 유한체 F_q를 설정한 후 q-1과 서로 소인 e를 임의로 정한다. 이때 $ed \equiv 1 \bmod q-1$인 d를 구한다. 이와 같이 A와 B는 각각 e_A, d_A, e_B, d_B를 준비한다. A는 메시지 P에 P^{e_A}를 구하여 B에게 보낸다. 이때 P^{e_A}는 B에게는 알 수 없는 메시지이다. B는 다시 보내온 P^{e_A}에다가 e_B를 곱하여 $(P^{e_A})^{e_B}$를 A에게 보내면 A는 d_A를 이용하여 $(P^{e_A})^{e_B d_A}=P^{e_B}$를 B에게 보낸다. 이때 B는 $P^{e_B d_B}=P$로 복호화할 수 있다.

유한체상에서의 이산대수문제를 풀 수 있는 알고리즘(Silver, Pohlig, Hellman에 의해서 고안됨)이 있는데 이는 유한체 F_q의 q-1의 소수약수가 작을 때만이 가능하고 소수약수가 크면 시간이 오래 걸려서 효율성이 없다. 그 외 The index-calculus algorithm 등 이산대수문제를 풀 수 있는 알고리즘이 있지만 결론적으로 이산대수문제는 소인수분해만큼의 어려움이 있어서 안전하다고 생각할 수 있다.

1.2 소인수분해의 어려움에 근거한 암호 구현법

소인수 분해의 어려움이란 서로 다른 두 소수의 곱에서 각각의 소수를 찾아내기가 용이하지 않음을 뜻한다. 숫자가 커지면 과연 그 수가 소수인지를 판별하는 것도 용이하지가 않다. 이런 점을 이용하여 1978년 Rivest, Shamir, Adleman은 논문 "A Method for Obtaining Digital Signatures and Public Key Cryptosystem"에 새로운 암호 구현법을 제안하였는데 이들의 이름을 따서 RSA 암호 구현법이라고 한다. RSA 암호 시스템, Rabin 암호 시스템, William 암호 시스템 등이 소인수분해의 어려움을 활용하고 있는데 이론적 근거는 다음과 같다. $n=pq$이고 p와 q는 서로 다른 소수이며, $\varphi(n)=(p-1)(q-1)$은 오일러의 파이(phi) 함수라고 하자. 이때, $(e, \varphi(n))=1$인 e를 선택한 A는 n과 e를 공개한다. p, q, mod $\varphi(n)$에 대한 e의 역원 x를 비밀키로 갖는다. 메시지 M을 A에게 보내고자 B는 A의 공개된 키 n, e를 이용하여 $C \equiv M^e \bmod n$로 암호화하여 A에게 보낸다. A는 e의 역원 x를 구하여 $C^x \equiv (M^e)^x = M^{ex} = M^{\varphi(n)k+1} \equiv M^1 \equiv M \bmod n$으로 복호화할 수 있다. 이때 intruder가 정보를 가로채기 위해서는 법 $\varphi(n)$에 대한 e의 역원 x를 구하여야 하는데 그러기 위해서는 $\varphi(n)=(p-1)(q-1)$을 알아야 한다. $\varphi(n)$을 알면 p, q를 알 수 있고 p, q를 알면 $\varphi(n)$을 쉽게 찾을 수 있다. 그러므로 e의 역원을 찾는 문제는 결국 $n=pq$에서 p, q를 찾는 문제와 같아진다. 암호 해독상에서 뿐만이 아니라 암호 구현을 위해서도 어느 정도 크기의 소수를 사용하는 것이 안전하고 또 효율성이 있는지

소수판별법과 소인수분해에 관한 관심이 고조되었다. 충분히 큰 수가 소수인지를 판별하는 데 사용되는 용어나 방법의 수학적 이론은 페르마와 오일러의 정리에 근거한다. 정수 n이 소수일 때 $a^n \equiv a \bmod n$이다. 그러므로 위의 합동식을 만족하지 않는 a를 찾아내면 그것이 합성수임을 알 수 있다. 그러나 n이 합성수임에도 위의 합동식을 만족하는 경우가 있는데 이러한 수를 a에 대한 pseudoprime이라고 한다. 이때에는 $b^n \not\equiv b \bmod n$인 b를 찾으면 된다. 그러나 불행스럽게도 어떠한 a에 대해서도 위의 합동식을 통과하는 합성수가 있다. 이러한 수를 Camichael 수라고 한다. Camichael 수는 무한히 많다는 가정이 있으나 아직 증명되지는 않았다. 소수를 찾는 또 다른 방법으로 밀러의 판정법(Miller's test)이 있다. 이것은 모든 소수는 밀러의 판정법을 통과한다는 사실에 근거한다. 그러나 어떤 합성수는 밀러의 판정법도 통과하는데 이러한 수를 strong pseudoprime(based to b)이라고 한다. 드물기는 하지만 역시 strong pseudoprime이 무한히 많고 특히 2에 대한 strong pseudoprime은 무한히 많다는 것이 증명되었다. 그러나 다행스럽게도 Rabin's Probabilistic Primality Test에 의하면 합성수 n이 n보다 작은 k번의 밀러의 판정법을 모두 통과할 확률은 $(1/4)^k$보다 작다고 한다. 그러므로 deterministic하게 판별할 수는 없지만 확률적인 방법으로는 소수를 찾는 것이 가능하다. 결론적으로 Riemann hypothesis(모든 양의 합성수 n에 대하여 밀러의 판정법에 실패할 base $b(<2\log_2 n)^2$)가 존재한다)가 사실이라면 양의 정수 n이 소수인지를 판별하는 데 $O(\log_2 n)^5$) bit operations만큼 걸리는 알고리즘이 존재한다는 정리가 있다. 이외에 오일러의 파이 함수와 이차잉여류를 이용한 소수 판별법, 원시근을 이용한 소수 판별법이 있고 메르센 수의 소수 여부를 판별하는 Lucas-Lehmer test, Lucas 수열을 이용한 소수 판별법 등이 있다. 소수를 판별하는 연구는 끊임없이 이루어지고 있다. 정수 n을 소인수분해하기 위해서는 이미 알고있는 10000보다 작은 모든 소수를 이용하여 소인수분해를 해보고 10^{15}까지의 중간정도 크기의 수는 Pollard Rho Method를 이용하고 그 후에는 quadratic sieve 또는 elliptic curve method 등을 이용한다. Monte Carlo method(Pollard Rho Method)란 합동식을 이용한 것이다. 이에 대한 설명은 생략하기로 한다. 또 다른 방법으로는 페르마 소인수분해법, $\sqrt{n}$의 연분수 전개법을 이용한 인수분해, H.W. Lenstra에 의해서 고안된 타원곡선을 이용한 소인수분해법 등이 있다.

1.3 Knapsack을 이용한 공개키 암호 구현법

Knapsack 공개키 암호 시스템은 1978년 Merkle과 Hellman이 처음 제안했다. 이 시스템은 쉽게 문제를 해결할 수 있는 수열(superincreasing sequence)을 선택한 후 이를 위장하여 해결하기 어려운 형태(선형적 변환)의 수열을 공개키로 공개하는 방법이다. Knapsack 공개키 암호 시스템으로는 Merkle-Hellman 외에 Lu-Lee시스템, algebraic coding theory를 이용한 Niederreiter 시스템, Goodman-McAuley 시스템, Pieprzyk 시스템, Chor-Rivest 시스템 등이 있다. 그러나 Shamir에 의해서 안전성에 의문이 제기되었고 Chor-Rivest시스템이 현재까지는 비교적 안전하다고 간주되고 있으나 Schnorr와 Hrner의 lattice reduction을 이용한 공격에 의하여 위협을 받고 있다.

1.4 타원곡선을 이용한 공개키 암호 구현법

타원곡선을 이용한 공개키 암호 시스템(Elliptic Curve Cryptosystem)은 1985년 N. Kobliz와 V. Miller에 의해서 처음으로 제안되었다. 유한체 위에서 정의된 타원곡선상의 점들은 군(abelian group)

을 이루는데 임의의 점 P와 이를 k배해서 얻는 점 Q에서 P와 Q만을 알고 k를 구하는 것은 유한체상에서의 이산대수문제보다 더 어려움이 있다는 것(EDLP)이 중심논리이다. Diffie-Hellman의 키 분배방식, Elgamal의 공개키 암호 방식, Massay-Omura의 공개키 방식들이 모두 타원곡선상의 시스템으로 전환될 수 있다. 단지 여기서는 메시지를 타원곡선상의 점으로 imbedding하는 문제가 남아 있으나 이것도 deterministic하지는 않지만 probabilistic한 방법들이 제시되고 있다. 타원곡선을 이용한 암호 시스템(ECC)은 우선 타원곡선의 다양한 선택을 통하여 다양한 시스템이 가능하다는 장점이 있다. 1990년 Menezes, Okamoto, Vanstone에 의해서 초특이 타원곡선(Supersingular Elliptic Curve)의 이산대수문제는 subexponential time algorithms이 존재한다는 것을 보였다. 즉, 완전지수복잡도(fully exponential complexity)로 타원곡선 암호 시스템이 깨지기 원한다면 초특이 타원곡선을 피해야 한다. 이러하듯 안전한 암호 시스템을 설계할 수 있다. 또한, 타 암호 시스템과 비교하여 작은 키 길이로 같은 안전도를 유지할 수 있다. 또 다른 이점은 모든 사용자가 같은 기저체를 사용하더라도 각 사용자가 다른 타원곡선을 선택할 수 있기 때문에 체연산을 수행하기 위해 같은 H/W를 사용할 수 있으며 추가적으로 타원곡선을 바꿀 수 있다. 타원곡선 암호 시스템(ECC)은 10여년 전부터 안전도가 타 공개키 시스템보다 효율적이라는 것이 알려졌고 최근 높은 속도의 구현이 가능하게 되었다. 타원곡선들과 유한체상의 타원곡선들이 이루는 군에 관한 여러 가지 사실들, 그리고 구체적인 암호 구현법에 관한 소개는 생략하기로 한다.

타원곡선에 관한 광범위한 연구는 약 150년 전부터 계속되어 왔는데 H.W. Lenstra는 "Elliptic curves and number-theoretic algorithms"에서 타원곡선을 이용하여 기존의 소인수분해보다는 훨씬 효율적인 새로운 소인수분해법을 제시하였고 최근에는 Andrew Wiles가 페르마의 마지막 정리의 증명에서도 중요하게 사용하였다.

1.5 algebraic coding theory를 이용한 암호 구현법

McElice의 공개키 시스템은 대수적 코딩이론을 이용한 것이다. "임의적인 것처럼 보이는" 선형코드로 복호화하는 것은 "완전히 임의적인" 선형코드로 복호화하는 것만큼 어렵다는 전제하에 이용하는 시스템으로 효율성 측면에서 볼 때는 다른 시스템보다 효율적이지만 공개키의 크기가 크다는 점과 전자서명이 제안되지 않았다는 단점이 있다. 동시에 McElieceㅅ스템의 안전성은 아직 광범위하게 연구되지 않았다.

1.6 Lattice Reduction을 이용한 암호 구현법

1996년 MIT의 Goldreich, Goldwasser, Halevi 등이 "Public-Key Cryptosystems from Lattice Reduction Problems"에서 기존의 문제들과는 다른, 격자줄임(lattice reduction)을 이용한 공개키 암호시스템을 소개하였다. 이는 1996년 M. Ajtai가 "Generating hard instances of lattice problems"에서 만약 한 lattice에서 the shortest non-zero vector를 측정하는(approximating) 것이 가장 나쁜 경우에 어렵다면 이것이 일방향 함수가 될 것이라는 것을 소개하였다. 그러나, Ajtai는 함정이 있는 함수(trapdoor function)를 제시하지 못하였고 따라서 공개키 암호 시스템을 제공하지 못하였다. 위의 세 사람이 Ajtai의 작업을 기초하여 함정

이 있는 함수를 만들어 내었고 따라서 격자줄임을 이용한 공개키 시스템을 개발해 낸 것이다. 이 암호 시스템의 안정성은 lattice에 있는 몇 가지 계산상의 어려운 문제와 관계 있다. 첫째는 The shortest non-zero vector problem(SVP)이다. SVP란 lattice의 기저 B가 주어졌을 때 Euclidean norm이 가장 작은 벡터를 찾는 문제이다. 이는 현재까지는 다항식 시간 알고리즘이 없고 NP-hard인지도 알려져 있지 않다. 그러나 LLL알고리즘, Schnorr에 의해서 개선된 LLL 등은 R^n상의 SVP를 deterministic polynomial-time algorithm으로 찾아낸다. 둘째로, the Closest Vector Problem(CVP)은 lattice에 대한 기저 B와 벡터 v가 주어졌을 때 v에 가장 가까운 lattice상의 벡터를 찾는 문제이다. CVP는 van Emde Boas에 의해서 NP-hard임이 증명되었다. 셋째로, the Smallest Basis Problem(SBP)은 한 lattice에 대한 기저 B가 주어졌을 때 같은 lattice에 대하여 "가장 작은" 기저 B'을 찾는 것이다. "가장 작은"의 의미는 작은 orthogonality defect를 갖는 기저를 뜻하는 데 B를 nonsingular matrix라 할 때 orth-defect(B)= $\prod_{i=1}^{n} \| b_i \| / \det(B)$로 정의한다. (B의 i번째 열의 Euclidean norm을 $\| \hat{b_i} \|$ 하자.) SBP에 대한 다항식 시간 알고리즘은 없고 가장 근사한 것으로 알려진 것은 LLL과 Schnorr 알고리즘이다. GGH의 암호 시스템은 McEliece의 암호 시스템과 매우 흡사한데 small dual orth-defect를 갖는 기저 A를 찾은 후 이를 unimodular transformation에 의하여 high dual orth-defect를 갖는 행렬 B를 만든다. 이때 B와 σ를 공개하고 A^{-1}와 $T(T=B^{-1}A)$를 비밀키로 간직한다. 메시지를 보내려는 송신자는 메시지를 벡터 v로 imbedding한 후 합이 0이고 분산이 σ^2인 error 벡터, e를 만들어 $Bv+e=c$를 계산하여 보낸다. 수신자는 갖고 있는 비밀키 A^{-1}와 T를 이용하여 $T[A^{-1}c]$로 v를 구한 후 $c-Bv=e$를 찾는다.

$$T[A^{-1}c]=T[A^{-1}(Bv+e)]=T[A^{-1}Bv+A^{-1}e]=T[T^{-1}v+A^{-1}e]=v+T[A^{-1}e]$$

여기서 메시지 v를 회복하기 위해서는 $[A^{-1}e]$가 0벡터여야 하는데 그러기 위해서는 초기치 σ가 중요한 역할을 한다. error가 발생하지 않기 위해서는 σ는 A^{-1}의 L_1 norm이 ρ이라 할 때 $1/(2\rho)$보다 작아야한다. 그러나 안전성을 높이기 위하여 σ를 높일 수 있는데 만약 A^{-1}의 각 행의 최대 L_∞ norm이 $\gamma/\sqrt{n}$보다 작다면 복호화할 때 error가 발생할 확률은 $2n\exp(-1/8\sigma^2\gamma^2)$보다 작거나 같다. 결국 초기치 σ와 small dual orth-defect를 갖는 기저가 중요한 역할을 하는데 이를 위해서는 SVP가 관계된다. 따라서 실제 실행을 할 때는 A 와 B의 dual orth-defect 비율이 크도록 할 수밖에 없다. 공개된 기저 B로부터 A를 직접 찾으려는 공격(SBP)도 있을 것이고 c로부터 가까운 lattice point를 찾으려는 노력(CVP)도 있을 것이다. 아예 처음부터 A를 찾으려고 할지도 모른다.(SVP) 그러나 어떠한 노력도 위에서 언급했듯이 어려운 문제이기 때문에 이 시스템이 안전하다고 할 수 있다. dimension n의 크기가 크면 실행시간이 오래 걸리고 n을 작게 하면 안전성이 흔들린다. Eurocrypto '98에서 이 시스템이 공격당하였다는 이야기도 있으나 필자는 아직 확인하지 못하고 있다.

2. 결론

정보 보호의 필요성은 오래 전부터 인식되어 왔으나 정보를 안전하게 전달할 수 있는 방법이 없다. 따라서 정보가 유출될 수 있다는 가정 하에 허가되지 않은 자는 정보를 알아내기 어렵고 수신자만이 해독하기 쉬운 암호 시스템이 필요하다. 이를 위한 노력은 고대부터 오늘날까지 꾸준히 되어 오고 있는데 오늘날은 전자 상거래 등에서 생길 수 있는 여러 문제들로 인하여 단순히 정보의 전달에 그치지 않고 인증 및 전자서명 등 여러 형태의 암호 시스템 개발이 요구되고 있다. 전자상거래에서 일어나는 전자서명이나 인증에 관해서는 언급하지 않았는데 이는 공개키 암호 구현법을 기초로 하여 전자서명이나 인증을 할 수 있기 때문에 구체적인 방법은 언급하지 않았다. 전자서명이나 인증은 메시지를 보낸 사람이 자신임을 입증하고 또 보낸 후에 자신이 아님을 부정할 수 없도록 하는 방법이다. 또한, 위조나 변경이 가능하지 않도록 하는 방법이다. 공개키 암호 구현이나 전자서명, 인증은 결국 함정이 있는 일방향 함수를 이용하여 가능하므로 수학자의 입장에서는 이들을 찾는 것이 중요하다. 이의 후보 논리로 유한체이론, graph theory, coding theory들이 많은 관심을 끌고 있다. 함정이 있는 일방향 함수가 공개키 암호 구현법에 사용되기 위해서는 임의성(randomness)이 있어야 하고 확률적 가치가 있어야 한다. 문제를 해결하는 데에 임의성이 없거나 확률적 분포가 없다면 일방향 함수라 해도 암호이론에는 가치가 없다. 그 외에 계산량에 관한 고려도 있어야 한다. 오늘날까지 사용되고 있는 일방향 함수들을 살펴보면서 많은 수학자들이 자신의 관심분야에서 일방향 함수를 찾으려는 노력이 있을 때 미래의 정보통신 분야를 다양하고 활발하게 전개하리라 생각된다.

참고 문헌

1. Ajtai, M., "Generating hard instances of lattice problems," *Pro. of 28th annual ACM Symposium on Theory of Computing*, 1996, 99-108.
2. Diffie, W., Hellman, M.E., "New directions in cryptography," *IEEE Transactions on Information Theory*, IT-22, 1976, 644-654.
3. El-Gamal, T., "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. Information Theory*, vol. 31, 1985, 469-472.
4. ETRI, 현대암호학, 1992
5. Goldreich, O., Goldwasser, S., Halevi, S., "Public-key cryptosystems from lattice reduction problems," *Electronic Colloquium on Computational Complexity*, 1996.
6. Koblitz, N., *A Course in Number Theory and Cryptography*, Springer-Verlag, 1987.
7. _____, "Elliptic curve cryptosystem," *Mathematics of Computation*, 48, 1987,

203-209.

8. Lidl, R., Niederreiter, H., *Finite Fields*, Cambridge University Press, 1987.
9. Menezes, A., *Elliptic curve public key cryptosystems*, Kluwer Academic Publishers, 1993.
10. Rivest, Shamir, Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, 21(2), 1978, 120-126.
11. Rosen, K.H., *Elementary Number Theory and its Applications*, Addison Wesley, 1993.