ID 기본 암호 시스템과 그 실현

*박영옥, °염흥열, *이만영
*한양대학교 전자통신공학과, °순천향대학교 전자공학과

ID-Based Cryptosystem and Its Implementation

*Young Ok PARK, °Heung Youl YOUM, *Man Young RHEE
*Dept. of Electronic Communication Eng. Hanyang Univ.
°Dept. of Electronics Eng. Sooncheonhyang Univ.

요약

유한체상의 이산 대수 문제에 안전성의 기반을 둔 ID 기본 암호 시스템에 대해 연구하고 실제적인 예로써 시스템의 기능을 실현한다. 기반이 되는 암호 및 서명 방식으로는 ElGamal의 시스템을 이용한다. 그리고 사용자들의 공모로 인한 시스템에의 공격 형태를 분석하여 이 시스템의 안전성에 관한 조건을 유도한다.

1. 서론

암호학은 데이타의 보안과 인증을 목적으로 1970년대 이후 급속히 발전해왔다. 특히 1976년, Diffie와 Hellman이 공개 키 암호 시스템의 개념을 제창하면서 부터 수 많은 알고리듬이 발표되고 또 해독되었다. 이러한 암호 시스템의 안전성에 있어서 현대 대부분의 암호 시스템은, 암호해독자가 이용할 수 있는 정보량이 충분하여 언젠가는 암호를 풀 수는 있지만 그 과정이 복잡하고 시간이 많이 요구되어 경제적으로 불합리한 경우의 계산적 안전(computational secrecy)을 기반으로 하고 있다.

본 논문에서도 유한체 내에서의 이산 대수(discrete logarithm) 계산이 어렵다는 것을 이용하여, 1984년 Shamir가 제안한 ID 기본 암호시스템(ID-based cryptosystem)의 개념을 도입한 암호 시스템에 관해 연구한다. 이 시스템은 ElGamal의 공개 키 암호 시스템을 근간으로 구성되며, 실제 응용할 수 있도록 안전성이 보장되는 큰 수를 적용하여 구체적인 시스템의 기능을 실현한다. 또한 가능한 공격의 형태를 분석하여 이 시스템의 타당성을 보이고 제약점을 유도한다.

2. ID 기본 암호 시스템

2 - 1 개요

ID 기본 암호 시스템은 이름, 주소, 등록 번호 등의 조합으로 생성되어 누구에게나 공개되어 있는 ID(identity)를 기본 요소로 하여 사용자 간의 비밀 통신이나 서명문

인증을 구현할 수 있도록 제안된 방식이다. 이 방식은 기존의 관용 키 시스팀이나 공개 키 시스팀에서 키 분배, 관리, 키 디렉토리 유지 등에 어려움이 따랐던 점을 개선한 것으로, 통신중 제 삼자가 개입하거나 상호의 비밀 키를 교환하거나 각 사용자의 공개 키를 보관하기 위한 디렉토리를 유지할 필요가 없다. 그리고, 주로 공개 키 암호 시스팀을 기반으로 하여 설계되며 시스팀의 구성을 위해서는 모든 사용자가 신뢰할 수 있는 키 센터(trusted key generation center)가 있어야 한다. 키 센터는 전체 시스팀에 필요한 모든 제원(parameter)들을 만들어 내고 사용자가 망에 가입할 때 그 사람의 비밀 키를 만들어 스마트 카드 등의 형태를 통해 비밀리에 전달해 준다. 결국 키 센터는 모든 사용자에게 비밀 키를 생성해 주고 일단 망이 생성된 후에는 존재할 필요가 없다.

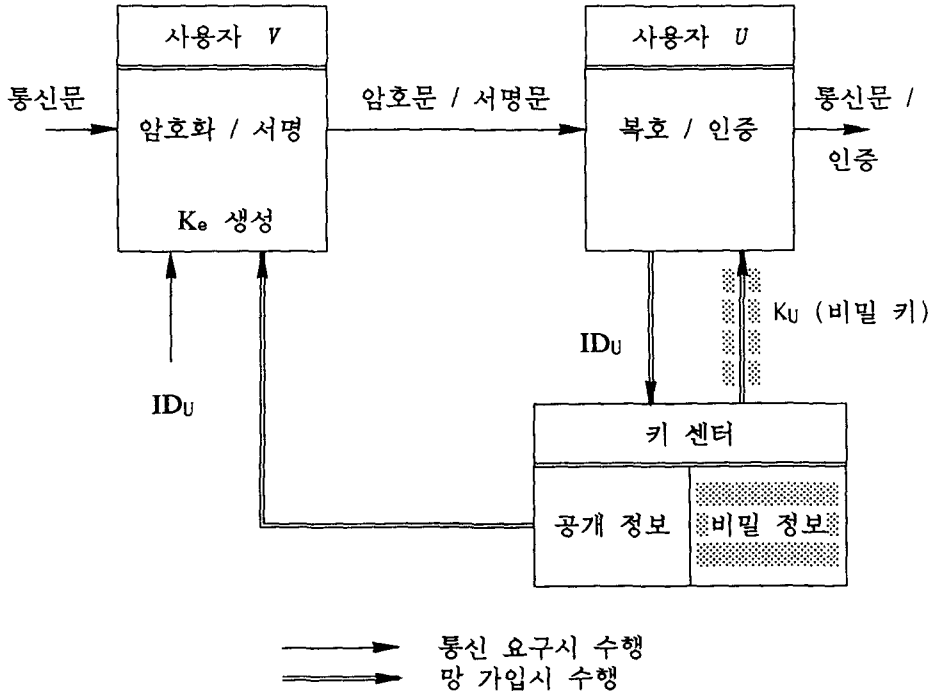

그림 1 ID 기본 암호 시스팀

2 - 2 키 센터의 시스팀 제원 생성

키 센터의 시스팀 제원 생성 과정은 다음과 같다.

(1) ID

사용자의 ID는 그 사람의 이름, 망 주소, 등록 번호 등을 조합하여 다음과 같은 k 차의 이진 계열(binary sequence)로 만든다.

$$ID = (x_1, x_2, \ldots, x_k), \quad x_i \in \{0,1\}, \quad 1 \le i \le k \qquad (2.1)$$

(2) EID

키 센터는 비밀 키 생성을 위해 단방향 함수 f를 응용하여 ID를 보다 더 긴 이진 계열 EID로 확장시킨다. 여기서는 단방향 함수로서 RSA 암호 시스팀을 이용한다. 우선 두개의 큰 소수 q_1과 q_2를 선택하여 그 곱을 N 이라 하고 gcd (e, $\phi(N)$) = 1 을 만족하는 e를 선택한다. 이진 계열 ID를 십진수 dID로 전환한 뒤 $dEID \equiv (dID)^e \pmod N$ 를 계산하고 dEID를 다시 n차 이진 계열 EID로 바꾼다.

$$EID = (x_1', x_2', \ldots, x_n'), \quad x_i' \in \{0,1\}, \quad 1 \le i \le n \qquad (2.2)$$

이 때, n은 N을 이진수로 표현했을 때의 비트수가 되고, $k < n$ 이어야 한다. e와 N값은 모든 사용자에게 공개된다.

(3) 비밀 정보

키 센터는 매우 큰 소수 p를 선택하고 n차 벡터인 비밀 정보 A를 만든다. A는 각 사용자의 비밀 키 생성에 사용되는데 서로 다른 가입자가 같은 비밀 키를 갖게 되지 않도록 A의 모든 원소는 서로 달라야 한다. 그러한 A를 생성하기 위해 Merkle-Hellman 의 암호 시스팀을 응용할 수 있다. 먼저 키 센터는 다음식을 만족하는 초증가성 계열 (superincreasing sequence) A'을 구한다.

$$A' = (a_1', a_2', \ldots, a_n'), \qquad \sum_{i=1}^{n} a_i' < p-1 \qquad (2.3)$$

그 다음, gcd(w, p-1) = 1 을 만족하는 w를 선정하고 다음 식을 이용하여 비밀 정보 A를 계산한다.

$$a_i \equiv a_i' w \pmod{p-1}, \quad a_i \in Z_p^*, \quad 1 \le i \le n, \qquad (2.4)$$
$$A = (a_1, a_2, \ldots, a_n)$$

(4) 공개 정보

Z_p^*의 원시근이 되는 α를 선택한 후 다음 식에 따라 공개 정보 B를 만든다.

$$b_i \equiv \alpha^{a_i} \pmod p, \quad 1 \le i \le n \qquad (2.5)$$
$$B = (b_1, b_2, \ldots, b_n)$$

(5) 사용자의 비밀 키

키 센터는 비밀 정보와 사용자의 EID를 벡터적(vector product)하여 그 사용자의 비밀 키를 만든다. 예를 들어 사용자 U의 비밀 키는

$$K_U \equiv A \cdot EID_U \pmod{p-1} \equiv \sum_{i=1}^{n} a_i \cdot x_{Ui}' \pmod{p-1} \qquad (2.6)$$

이 된다. 이렇게 생성된 K_U는 매우 안전한 채널을 통해 사용자 U에게 전달된다.

2 - 3 ElGamal 암호 시스팀을 기반으로 한 ID 기본 암호 시스팀

키 센터가 생성한 제원들을 이용하여 사용자 V가 통신문 M을 사용자 U에게 암호문 (C_1, C_2)로 암호화하여 전달하고, U가 이를 복호하는 과정에 대해 기술한다.

<u>암호화 과정</u>

(1) 수신자 U의 ID_U로부터 EID_U를 계산한다.

(2) 키 센터의 공개 정보 B와 EID_U를 이용하여 암호화 키 K_{eU}를 구한다.

$$K_{eU} \equiv \prod_{i=1}^{n} b_i^{x_{Ui}'} \pmod p \tag{2.7}$$

이 때, $b_i \equiv \alpha^{a_i} \pmod p$ 이므로

$$K_{eU} \equiv \prod_{i=1}^{n} \left(\alpha^{a_i}\right)^{x_{Ui}'} \equiv \alpha \cdot \exp\left(\sum_{i=1}^{n} a_i \cdot x_{Ui}'\right) \equiv \alpha^{K_U} \pmod p \tag{2.8}$$

로 된다.

(3) $0 \le r \le p-2$ 의 구간에 있는 임의의 수 r을 선택한다.

(4) 사용자 V가 전달하고자 하는 통신문 M은 $0 \le M \le p-1$ 이어야 하며 그에 해당하는 암호문 (C_1, C_2)는 다음 식으로써 계산한다.

$$\begin{aligned} C_1 &\equiv \alpha^r \pmod p \\ C_2 &\equiv M \cdot (K_{eU})^r \pmod p \end{aligned} \tag{2.9}$$

<u>복호 과정</u>

(1) (C_1, C_2)를 수신한 U는 자신의 비밀 키 K_U로 다음을 계산한다.

$$C_1' \equiv (C_1^{K_U})^{-1} \equiv (\alpha^r)^{-K_U} \pmod p \tag{2.10}$$

(2) C_1'과 C_2로부터 M을 복구한다.

$$\begin{aligned} (C_1') \cdot C_2 &\equiv (\alpha^r)^{-K_U} \cdot M \cdot (K_{eU})^r \\ &\equiv (\alpha^r)^{-K_U} \cdot M \cdot (\alpha^{K_U})^r \equiv M \pmod p \end{aligned} \tag{2.11}$$

이 암호 시스팀은 암호문이 통신문의 두배가 되므로 효율면에서 단점을 가진다. 그러나, 공개 정보로부터 비밀 정보를 구하는 것은 $b_i \equiv \alpha^{a_i} \pmod p$ 에서 a_i를 구하는 이산 대수 문제이므로 안전이 보장될 수 있다.

3. ID 기본 암호 시스팀의 실현

암호 시스팀에 사용되는 수 들은 시스팀의 안전성 보장을 위해 천문학적인 크기를 갖는다. 그러나, 컴퓨터의 계산능력 향상과 앨고리듬의 개선으로 보다 짧은 시간 내에

보다 큰 수에 대한 이산 대수 계산이나 인수 분해 계산이 이루어지고 있기 때문에 그 능력 이상의 수를 채택해야만 시스템이 안전하다고 볼 수 있다. LaMacchia와 Odlyzko는 $p \approx 10^{100}$ 인 Z_p^*에서의 이산 대수 문제를 푸는 것이 가능하다고 밝힌 바 있다. 따라서 본 논문에서는 $p \approx 10^{165}$ 인 수를 채택하여 ID 기본 암호 시스템을 구현해 본다. 이에 필요한 연산은 C-language와 8086 assembler를 사용하여 IBM PC 386에서 실행하였다.

3 - 1 키 센터의 기능 실현 예

먼저 기본 제원들을 구한 예를 보이고 이어 그 밖의 제원들을 제시한다.

(1) p ($\approx 10^{165}$) : 소수

p를 만들기 위해서 우선, 예정한 크기(약 10^{165})로 임의의 수를 만든 뒤 그 수의 소수성을 판정한다. 본 연구에서는 소수 판정을 위해 Solovay-Strassen의 방법을 채택하였다. 다시 말하면, p보다 작은 수 중 k개를 임의로 골라 a라고 한 뒤

$$\gcd(a, p) = 1 \text{ 이고 } a^{(p-1)/2} \equiv \left(\frac{a}{p}\right) \pmod{p} \qquad (3.1)$$

인지 확인하여 모두 맞으면 p를 소수로 인정한다. 만약 선택한 모든 a에 대해서 위의 두 조건을 만족하고도 소수가 아닐 확률은 2^{-k} 이하이므로 k를 적절히 정했다면 p를 소수로 받아들일 수 있다. 그러나 p-1의 소인수들 가운데 매우 큰 소수가 적어도 하나는 있어야 하기 때문에, 본 논문에서는 $k = 20$ 으로 하여 먼저 $10^{83}(\approx 2^{280})$ 정도의 소수 f를 만들고 $p = f \cdot Q + 1$ 의 식으로 p를 만들어 p가 소수인지를 다시 판정했다. 이렇게 하여 만든 소수 f와 p는 다음과 같다.

$$\begin{aligned} f = {} & 30,379,094,892,308,063,530,957,925,674,071,409,186, \\ & 848,392,822,883,766,461,848,106,929,536,922,569,640, \\ & 759,219 \end{aligned} \qquad (3.2)$$

$$\begin{aligned} p = {} & 106,275,326,779,898,375,514,837,219,519,397,126,420, \\ & 491,508,645,703,415,499,109,079,653,831,352,909,870, \\ & 772,153,140,203,616,768,858,371,060,561,567,707,969, \\ & 863,965,972,471,954,444,941,748,256,319,214,505,078, \\ & 385,795,297 \end{aligned} \qquad (3.3)$$

(2) α : Z_p^*의 원시근

어떤 소수 p에 대해 Z_p^*의 원시근은 mod p에 대해 차수가 p-1 이다. 그러므로 임의의 수 α에 대해, p-1의 인수들과 그 인수들의 각 조합을 t라 하고 $\alpha^t \equiv 1 \pmod{p}$ 가 되는지 살펴보아야 한다. 모든 경우의 t에 대해 $\alpha^t \not\equiv 1 \pmod{p}$ 이어야만 α는 원시근이 된다. 본 연구에서는 α를 다음의 정리 1을 만족하는 임의의 수로 정한 뒤 시스템 제원으로 해본다.

정리 1 p가 소수이고 α가 Z_p^*의 원시근이면

$$\alpha^{(p-1)/2} \equiv -1 \pmod{p} \tag{3.4}$$

이다.

[증명] α가 원시근이라 하고 Z_p^*의 원소들을 모두 곱하면

$$1 \cdot 2 \cdot 3 \cdots (p-1) \equiv \alpha\,\alpha^2\alpha^3 \cdots \alpha^{p-1} \pmod{p} \tag{3.5}$$

$$(p-1)! \equiv \alpha^{(p-1)/2} \pmod{p} \tag{3.6}$$

이다. Fermat의 정리에 의해 $\alpha^p \equiv \alpha \pmod{p}$ 이고, Wilson의 정리에 의해 $(p-1)! \equiv -1 \pmod{p}$ 이므로 식 (3.6)은

$$-1 \equiv \alpha^{(p-1)/2} \pmod{p} \tag{3.7}$$

가 된다. (증명끝)

원시근을 채택하는 것은 비밀 정보로부터 공개 정보를 구할 때 일대일 대응이 되기 위한 이유이므로 임의로 선택한 α로 일대일 대응이 가능한 경우라면 α가 원시근이 아니더라도 무리없이 시스팀을 구성할 수 있다. 여기서는 $\alpha = 5$ 로 하였다.

(3) w : gcd(w, p-1) = 1

w는 식 (2.5)에 의해 초증가성 계열로부터 비밀 정보를 만드는데 사용된다. 이 때, 비밀 정보의 원소들이 모두 비슷한 크기를 갖기 위해서는 w를 p에 가까운 크기로 해야 한다. 따라서 p 정도 크기의 수를 발생시켜 gcd(w, p-1) = 1 을 만족하는지 확인한다. 두 수의 최대공약수는 Euclid의 앨고리듬으로 구할 수 있다. 이렇게 구한 w 는 다음과 같다.

$$\begin{aligned} w = {} & 57,586,568,961,070,745,603,968,966,522,877,458,062, \\ & 880,186,288,580,363,060,678,974,288,930,954,786,538, \\ & 847,915,235,690,877,086,327,292,752,345,084,347,041, \\ & 098,790,887,974,973,034,082,774,553,012,993,727,357, \\ & 937,588,691 \end{aligned} \tag{3.8}$$

(4) q_1, $q_2(\approx 10^{80})$: 소수, N, $\phi(N)$

q_1과 q_2는 소수이므로 앞서 기술한 방법으로 구하며, $N = q_1 \cdot q_2$ 이고 $\phi(N) = (q_1-1)(q_2-1)$ 인 것을 이용하여 N과 $\phi(N)$을 만들 수 있다.

$$\begin{aligned} q_1 = {} & 5,841,709,580,226,467,726,160,203,139,301,612,255, \\ & 509,959,915,365,990,530,265,040,363,850,795,715,093, \\ & 579,093 \end{aligned} \tag{3.9}$$

$$\begin{aligned} q_2 = {} & 96,939,046,188,131,698,153,249,921,290,481,507,536, \\ & 878,381,731,935,002,307,462,646,553,435,507,651,467 \end{aligned} \tag{3.10}$$

$$\begin{aligned} N = {} & 566,289,754,815,224,988,769,873,466,726,379,381,532, \\ & 696,702,264,249,003,550,891,168,779,437,750,143,788, \\ & 591,765,391,049,045,088,179,053,751,809,701,517,554, \end{aligned} \tag{3.11}$$

798,434,287,313,651,801,197,088,907,903,923,841,979,
431

$\phi(N)$ = 566,289,754,815,224,988,769,873,466,726,379,381,532,
696,702,264,249,003,550,891,168,779,437,750,143,782,
749,958,871,776,389,230,320,697,362,586,798,780,537, (3.12)
301,640,539,591,186,533,849,262,410,554,773,240,748,
872

여기서 N은 이진수로 528 비트의 수이다.

(5) e : gcd(e, $\phi(N)$) = 1

w를 구하던 것과 마찬가지로 임의의 수를 만든 뒤 Euclid 앨고리듬을 이용하여 gcd(e, $\phi(N)$) = 1 인지를 확인한다.

e = 3,406,592,165,502,205,275,452,926,636,956,391,673,
401,796,322,508,184,116,507,428,895,001,545,893,165, (3.13)
912,509

이상의 기본 제원들을 가지고 그 밖의 시스팀 제원들을 만든다.

(6) 사용자 U의 ID가

원문 : 90501159 park young ok

이진 계열 형태 :

$ID_U = (x_{U1}, x_{U2}, \ldots, x_{Uk})$

= (0,0,1,1,1,0,0,1,0,0,1,1,0,0,0,0,0,0,1,1,0,1,0,1,
0,0,1,1,0,0,0,0,0,0,1,1,0,0,0,1,0,0,1,1,0,0,0,1,
0,0,1,1,0,1,0,1,0,0,1,1,1,0,0,1,0,0,1,0,0,0,0,0,
0,1,1,1,0,0,0,0,0,0,1,1,0,0,0,0,1,0,1,1,1,0,0,1,0,
0,1,1,0,1,0,1,1,0,0,1,0,0,0,0,0,0,1,1,1,1,0,0,1, (3.14)
0,1,1,0,1,1,1,1,0,1,1,1,0,1,0,1,0,1,1,0,1,1,1,0,
0,1,1,0,0,1,1,1,0,0,1,0,0,0,0,0,0,1,1,0,1,1,1,1,
0,1,1,0,1,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)

일 때 k = 240 이고, dID_U를 십육진수로 표현하면

dID_U = 393035303131353920706172 6B20796F756E67206F6B00
00000000000000 (3.15)

이 된다. 따라서

$dEID_U \equiv (dID_U)^e \pmod N$

$$\equiv 5910D176528FE6DC77353955A4144A099E880BF87E358B$$
$$D0CE9AF91B5E49374304FDBFE29D52683E9A4E0E4598E9 \quad (3.16)$$
$$54DACEE1CCE72EF728CDFB008E226EB289366949$$

$EID_U = (x_{U1}', x_{U2}', \ldots, x_{Un}')$

$= (0,1,0,1,1,0,0,1,0,0,0,1,0,0,0,0,1,1,0,1,0,0,0,1,$
$0,1,1,1,0,1,1,0,0,1,0,1,0,0,1,0,1,0,0,0,1,1,1,1,$
$1,1,1,0,0,1,1,0,1,1,0,1,1,1,0,0,0,1,1,1,0,1,1,1,$
$0,0,1,1,0,1,0,1,0,0,1,1,1,0,0,1,0,1,0,1,0,1,0,1,$
$1,0,1,0,0,1,0,0,0,0,0,1,0,1,0,0,0,1,0,0,1,0,1,0,$
$0,0,0,0,1,0,0,1,1,0,0,1,1,1,1,0,1,0,0,0,1,0,0,0,$
$0,0,0,0,1,0,1,1,1,1,1,1,1,0,0,0,0,1,1,1,1,1,1,0,$
$0,0,1,1,0,1,0,1,1,0,0,0,1,0,1,1,1,1,0,1,0,0,0,0,$
$1,1,0,0,1,1,1,0,1,0,0,1,1,0,1,0,1,1,1,1,1,0,0,1,$
$0,0,0,1,1,0,1,1,0,1,0,1,1,1,1,0,0,1,0,0,1,0,0,1,$
$0,0,1,1,0,1,1,1,0,1,0,0,0,0,1,1,0,0,0,0,0,1,0,0,$ (3.17)
$1,1,1,1,1,1,0,1,1,0,1,1,1,1,1,1,1,1,1,0,0,0,1,0,$
$1,0,0,1,1,1,0,1,0,1,0,1,0,0,1,0,0,1,1,0,1,0,0,0,$
$0,0,1,1,1,1,1,0,1,0,0,1,1,0,1,0,0,1,0,0,1,1,1,0,$
$0,0,0,0,1,1,1,0,0,1,0,0,0,1,0,1,1,0,0,1,1,0,0,0,$
$1,1,1,0,1,0,0,1,0,1,0,1,0,1,0,0,1,1,0,1,1,0,1,0,$
$1,1,0,0,1,1,1,0,1,1,1,0,0,0,0,1,1,1,0,0,1,1,0,0,$
$1,1,1,0,0,1,1,1,0,0,1,0,1,1,1,0,1,1,1,1,0,1,1,1,$
$0,0,1,0,1,0,0,0,1,1,0,0,1,1,0,1,1,1,1,1,1,0,1,1,$
$0,0,0,0,0,0,0,0,1,0,0,0,1,1,1,0,0,0,1,0,0,0,1,0,$
$0,1,1,0,1,1,1,0,1,0,1,1,0,0,1,0,1,0,0,0,1,0,0,1,$
$0,0,1,1,0,1,1,0,0,1,1,0,1,0,0,1,0,1,0,0,1,0,0,1)$

이 된다. 이 때 N의 비트수가 528 이므로 $n = 528$ 이다.

(7) 비밀 정보 A를 만들기 위해 $\sum_{i=1}^{n} a_i' < p-1$ 를 만족하는 초증가성 계열을 만들고 이어 식 (2.4)를 이용하여 A를 계산하면 그 각 원소는 표 1과 같다.

(8) 기본 제원으로 선택한 $\alpha = 5$ 와 식 (2.5)로 공개 정보를 구하면 그 원소들은 표 2와 같다.

(9) 비밀 정보 A와 EID_U로 사용자 U의 비밀 키를 만들면 식 (2.6)에 의해

$$K_U \equiv A \cdot EID_U \pmod{p-1}$$
$$\equiv 460A4420B7776FA70C08263315E31693E569E53E64DA22$$
$$97B5A160BDE85BC7CD13988C2E75F48416CC2AE63A1C08 \quad (3.18)$$
$$A071562A0807DFD0AFF5360616AE75A4B4F05330794D$$

이다.

3 - 2 ID 기본 암호 시스팀의 실현 예

다음 문장을 암호화하여 사용자 U에게 보낸다고 하자.

> Cryptography, indeed, is the only practical means for sending information over an insecure channel, be it telephone line, microwave, or satellite.

암호화될 통신문 M은 $0 \le M \le p-1$ 이어야 하므로 위 문장을 68문자씩 나누어 한블럭으로 하고 한 문자에 8 비트씩 할당한다. 그리고, 각 문자의 ASCII 값을 근거로 하여 통신문을 만들면 표 3과 같다.

암호화 동작

(1) 수신자 U의 ID를 (3.14)라 하고 EID_U를 계산하면 EID_U는 (3.17)과 같다.

(2) 암호화 키 K_{eU} 를 구하면 다음과 같다.

$$\begin{aligned} K_{eU} \equiv\ & 12E4CB5D425F24B3FEFAAFDCEA33E4BE2CDD6EBCAFF51E \\ & 68FA518E9946BEEF74068409BB49C11E22BDA7D03302AC \\ & 593FAE8D58836E42798762F94753717063A0AD998485D \end{aligned} \qquad (3.19)$$

(3) 매 블럭 마다 다른 임의의 r, $0 \le r \le p-2$, 을 선택하여 식 (2.10)으로 암호화하면 표 4가 된다.

복호 동작

(1) U의 비밀 키는 (3.18)과 같으므로 $-K_U$는 다음과 같다.

$$\begin{aligned} -K_U &\equiv (p-1) - K_U \pmod p \\ &\equiv 19268A20FB980659B2405611869FFA56B08A06844664B7 \\ &\quad 51EC870AE2409AA9FAAED62D2E6F6ABAECB1F7B26E3775 \\ &\quad 68E697B122B100D1D844F87458192CC7FC2358E1F6793 \end{aligned} \qquad (3.20)$$

(2) 통신문을 복구한다.

$$\begin{aligned} \text{(블럭 1)}\quad M &\equiv \left[C_1\right]^{-K_U} \cdot C_2 \pmod p \\ &\equiv 726D666F696E6720696E6E6473657220666F7320616E6D \\ &\quad 656C2063617469616370727920 6E6C206F686520746973 \\ &\quad 2C2065646465696E2C20687961706772746F79704372 \end{aligned} \qquad (3.21)$$

$$\begin{aligned} \text{(블럭 2)}\quad M &\equiv 7220206F652C61766F7763726D692C206E656C6965206F \\ &\quad 6E70686C6574657420206962652C20656C6E6E68612063 \\ &\quad 726563757365696E6E20206165726F766E0A696F6174 \end{aligned} \qquad (3.22)$$

$$\text{(블럭 3)}\quad M \equiv 652E69746C6C74657361 \qquad (3.23)$$

이렇게 복호된 통신문을 표 3과 비교하면 일치됨을 알 수 있다.

표 1 비밀 정보

a_1	063FF0BF4F0C22BD179D6B49BD3D320023E2C0D9A924DD00F4D38E4 35ADA9560A0CEB37F9A38CB0EB993F3E5BAC33313DB2E5D2A50FF71 3102E2792F9168E63FA0447CF23
a_2	0C7FE17E9E18457A2F3AD6937A7A640047C581B35249BA01E9A71C8 6B5B52AC1419D66FF3471961D7327E7CB75866627B65CBA54A1FEE2 6205C4F25F22D1CC7F4088F9E46
a_3	18FFC2FD3C308AF45E75AD26F4F4C8008F8B0366A4937403D34E390 D6B6A5582833ACDFE68E32C3AE64FCF96EB0CCC4F6CB974A943FDC4 C40B89E4BE45A398FE8111F3C8C
a_4	1478579771519894 99EA81D931EB6441303561F51C748E8C3EBB512 CB7B4448D8665E60B7AFC5547ADE53E5BFCE3A5B12C5F2621092CA6 43CB3F0FF87724EA8AA40EE9838
a_5	0B6980CBDB93B3D510D42B3DABD89CC2718A1F120C36C39D1595816 B504822A38CBC16259F2EA7613D101BE620915874ABAA8910938A69 434AA9666CDA278DA2EA08D4F90
. . .	. . .
a_{524}	18B7B96E177159D4E23736DF1309D2918B463B3D06DD29247248620 89FB8AEE4A05FCCD20DCBCE537E366D4201A573C13AF1AE44D22D57 3B6385C81FD02F5E78AEE2D4E80
a_{525}	13E8447927D33655A16D95496E15796327ABD1A1E107F8CD7CAFA32 32050F751C0AFE3B2C4CD9978DDB279B22A14F494C8CF9958258BCB 327B36D6BB8C3C757EFFB0ABC20
a_{526}	0A495A8F4896EF571FDA521E242CC7066076FE6B955D981F917E255 82181882C0150117432D12FC39CAA92927AF3F63BE48B6F7ECC48B3 20AA98F3F30456A38BA14C59760
a_{527}	1492B51E912DDEAE3FB4A43C48598E0CC0EDFCD72ABB303F22FC4AB 04303105802A022E865A25F8739552524F5E7EC77C916DEFD989166 415531E7E608AD47174298B2EC0
a_{528}	0B9E3BDA1B4C40085C687003D8B4F05992FB54D628C40702DE17747 266E5BA3885308FDF747ABBE053EFE9781299E601E519FAC9B253E9 3E5E8F1647FD3846BC271C67CA0

표 2 공개 정보

b_1	06D6A7CA701238C4A8C51F4185E12A3B6DAD50B04A3850699301DEE 191F63F758299C3A86AEEA36AC2811E687B6987A18FFF7FF9B0A78D 2F6C876B3216D168ED22C3ACEBD
b_2	1B4B3F070AF92FBA695CC8F2331A536ABD1E9A3F81CE80927866DA1 D408B77005C619A7FEE08E70B418441E3FB29C4BC3AEFCAD91BC544 54297B7C805912EDDC337D838F1
b_3	01350FF961BE49ED48D5B45637CF098D04A9C044B093A9C60B2A08F 4EAED9DE62FA1A047CC0640C6F7FF68269636195347AA29023DCA51 57CEB7494425EA9EF8D583D2106
b_4	121D16389777C32DC245E40C01204FC8D07C20469B37D5567D55527 205EE16924F040AB4DA8593E8DA49AEB78411F4BCED52C9C3EBE12A A9B2622AE343D598FAFA10FD7BB
b_5	0A061C96179517695CE680565F46B17D0F27EBAA978114E4C96F0DF 458674C32718CEA45F370FA5AD047377C57B81E181B8DE420138E00 12C2491C6068009F87C072415F5
. . .	. . .
b_{524}	066329D50B2A74FF1578BE13E3DA577C2F77A70EB85CFE2CD122583 4875BE189120FFEA04222AA74E956AE9123A94DE8C23BBDF6CACB21 7DB2BDC0E74075A7B4F7E4A5C9A
b_{525}	1C988F4D7256A84213B1242B2411B90018B83B2CFBAB74A75E47D90 4D468D7754AB79649704116A751CC90FE748DCB24A512A8D1B71874 742B777559FB8FA5DF06C57FF07
b_{526}	17F7B0AF50D35EE1BA1C589DF1FE873F01CE5D89399960E3E601210 965C2F036585D59A076FA37087C27825F3BFCE3C77561B9B7E39B4A B9A822F43ADBED79B8609B5431E
b_{527}	0E9A90DFEE0555C5FD943A1B1E6691A6D4F40B431A81A92EDC7C662 89244F47DDB211A64C257E214AD5CA158D4E032FE664D713D8C04DF 22EDA79898E29E21D4909922160
b_{528}	10D7075981DC4F3D18276159F4411C0D6796AFC7BE4034AA2672B7D 506065E898C17A4F7CF76B7312D7E40217F120304F72A2B508A14E3 833D3BCC2EE368C26F0F48A9579

표 3 통신문

블럭 1	원문	Cryptography,_indeed,_is_the_only_practical_means_for_sending_inform
	통신문	726D666F696E6720696E6E6473657220666F7320616E6D656C20636174696163707279206E6C206F6865207469732C2065646465696E2C20687961706772746F79704372
블럭 2	원문	ation_over_an_insecure_channel,_be_it_telephone_line,_microwave,_or_
	통신문	7220206F652C61766F7763726D692C206E656C6965206F6E70686C65746574202069626 52C20656C6E6E686120637265637357365696E6E20206165726F766E0A696F6174
블럭 3	원문	satellite.
	통신문	652E69746C6C74657361

표 4 암호문

블럭 1	C_1	AE9E6E4EBFD3115B72FFFC2441DD5CD98901773E502C883714BEB9909EF0D5F76EF18DFA22DF9CBBE7A978682A36811D91F556E1F54EB1878132A3832A1ABFD1CCA4151A
	C_2	91348BF792C13D1C1D0DB753A179E3AB02134235E95884E1A16AD734F8E6542567F9419DADD20F681E03323B1D251DF175D0ED4F8C5F717BE8C277CBAF89AEAE218CAEC2
블럭 2	C_1	CCDE7D5383BC240B48E1546572831950OCEBF0E6B66FD779C3C3E063D4BB1008479A8D0BBFC10FF7844EB14C77B9EF16B20C1577942EA1E727D1A2DF2F2B7F3287C142C2
	C_2	161B26798FBB54F7DD0171B80EDA425DFD249E6B35C01D23EB339CCF33688A8A706B8AD003DE3A9001D2F9C3955AF331C76771FA06E96886F9D3F22ADEC4E03A1B369E2B
블럭 3	C_1	9DD9D1926B2A8422FEBD336377CE9E18F52314E880D3E4B721185875E489E3A9961A64C473F6A3E35141164A254B5B4F67841FED1B27AD538C832FB4832AAD357A17660A
	C_2	D10E97F0E2636B04D207A894614E5EB105CA4D94D29829499E7D774B7BD96D65EA937ECBBC0368DF9D5C175AF19F7A0276554B3C5E205AA9EFD9446A39113B694F5CF20A

4. ID 기본 암호 시스팀에 대한 공격

비밀 정보 A는 모든 사용자의 비밀 키를 만드는 핵심이 되므로 A가 유출되면 시스팀은 안전은 파괴되고 만다. 그러나 사용자 중 n명 이상이 공모한다면 A는 이산 대수 계산이 아닌 다른 방법으로 유도될 수 있다. 사용자 U의 비밀 키는

$$K_U = \sum_{i=1}^{n} a_i \cdot x_{Ui}' \pmod{p-1}$$
$$= \sum_{i=1}^{n} a_i \cdot x_{Ui}' - Q_U(p-1) \qquad (4.1)$$

이다. $1 \le U \le n$ 에 대해서, 즉 n명의 사용자에 대해서 식 (4.1)은

$$\begin{aligned} \sum_{i=1}^{n} a_i \cdot x_{1i}' - K_1 &= Q_1(p-1) \\ \sum_{i=1}^{n} a_i \cdot x_{2i}' - K_2 &= Q_2(p-1) \\ &\vdots \\ \sum_{i=1}^{n} a_i \cdot x_{ni}' - K_n &= Q_n(p-1) \end{aligned} \qquad (4.2)$$

이 된다. 이를 행렬로 표현하면 다음과 같다.

$$\begin{bmatrix} x_{11}' & x_{12}' & \cdots & x_{1n}' & -K_1 \\ x_{21}' & x_{22}' & \cdots & x_{2n}' & -K_2 \\ \vdots & & & \vdots & \vdots \\ x_{n1}' & x_{n2}' & \cdots & x_{nn}' & -K_n \end{bmatrix} \cdot \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \\ 1 \end{bmatrix} = (p-1) \cdot \begin{bmatrix} Q_1 \\ Q_2 \\ \vdots \\ Q_n \end{bmatrix} \qquad (4.3)$$

$$X \cdot A = (p-1) \cdot Q \qquad (4.4)$$

X는 $n \times (n+1)$ 행렬로 처음 n개의 열은 n명의 서로 다른 사용자의 EID로 되어 있으므로 $\det X \neq 0$ 이다. 따라서

$$X \cdot A = 0 \pmod{p-1} \qquad (4.5)$$

가 된다. 즉 비밀 정보로 이루어진 행렬 A는 n명의 사용자가 공모하여 행렬 X를 만들면 Gauss-Jordan 방법에 의해 구해질 수 있게 된다. 그러므로 비밀 정보 A를 유도 당하지 않으려면 시스팀의 사용자를 n명 이하로 제한할 수 밖에 없다. 앞 장에서 제시한 예의 경우 n = 528 이므로 전체 사용자를 500명 이하로 하는 것이 바람직하다.

5. 결론

이산 대수 문제에 안전성의 기반을 둔 ID 기본 암호 시스팀을 ElGamal의 시스팀을 바탕으로 구성하여 기술하였다. 이 때 사용되는 큰 소수 p에 대해 p-1의 소인수로서 10^{83} 정도 크기의 소수를 하나 채택하였고 p는 10^{165} 정도의 수로 사용하였다. 또한 이를 바탕으로 키 센터의 기능과 암호, 복호 과정을 실현해 보았다. 암호화 과정에서는 2번의 모듈러 멱승산과 최대 n+1번의 모듈러 곱셈이 사용되었으며 복호시에는 각 한번의 감산, 모듈러 멱승산, 모듈러 곱셈이 요구되었다. 그리고, n명 이상의 사용자가 공모할 경우 시스팀의 비밀 정보가 유도되어 안전성이 파괴되므로 사용자 수를 n 이하로 설정해야 하는 근거를 보였다. 그러므로 이 시스팀은 회사나 개별 단체 등 작은 그룹 내의 비밀 통신에 적합함을 알 수 있다.

참고문헌

[1] 최영주, "이산 대수 문제의 비교 분석과 암호론," 데이타 보호기술 Workshop 논문집, pp.21-30, 1991.

[2] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Trans. Inform. Theory*, vol.IT-22, pp.472-492, 1976.

[3] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. Inform. Theory*, vol.IT-31, pp.469-472, 1985.

[4] Hua Loo Keng, *Introduction to Numeber Theory*, New York, Springer-Verlag, 1982.

[5] A. Shamir, "Identity-based cryptosystem and signature scheme," *Advances in Cryptography : Proc. Crypto '84*, New York, Springer-Verlag pp.47-53, 1985.

[6] Henk C.A. van Tilborg, *An Introduction to Cryptography*, Kluwer Academic Publishers, 1988.

[7] S. Tsujii, T. Itoh, and K. Kurosawa, "ID-based cryptosystem using discrete logarithm problem," *Electron. Lett.* vol.23, pp.1318-1320, 1987.