通信情報保護學會論文誌
第 1 卷, 第 1 號, 1991. 12

GF(2^m)의 정규기저를 사용한 D-H형 공용키이분배 시스템†

이창순* · 문상재**

A D-H type Public Key Distribution System using a Normal Basis in GF(2^m)

Chang-Soon Lee and Sang-Jae Moon

요 약

여러 Diffie-Hellman형 공용키이분배 프로토콜들의 문제점을 비교 고찰하고, 개선된 공용키이분버 프로토콜을 소개한다. 이 프로토콜을 GF(2^m)의 정규기저를 사용하여 소프트웨어적으로 구현하여 시뮬레이션하였다. GF(2^m)에서의 승산을 고속으로 할 수 있는 정규기저를 효과적으로 발굴하는 전산 프로그램도 개발하였다.

Abstract

Several variants of the Diffie-Hellman public key distribution are examined, and a simple and relatively secure public key distribution protocol is introduced. Using a normal basis of GF(2^m), this protocol is implemented, and simulated in software. A program is developed, whereby a normal basis is effectively searched for fast multiplication in GF(2^m).

1. 서 론

Modulus 멱승(exponential)을 이용한 단일방향(one way) 함수를 이용한 대표적인 공용키이 암호법으로는 D-H(Diffie-Hellman) 방법[1]과 RSA[2] 방법 등이 있다. 이러한 공용키이 암호법을 이용하면 관용 암호법에 비해 두 통신자의 인증, 세션키이의 생성 및 키이관리 등의 기능을 효과적으로 수행할 수 있다[3]. RSA 방법은 합성수 modulus 멱승을, 그리고 D-H 방법은 유한체 GF(q), q는 소수 혹은 소수의 멱, 에서 멱승을 이용한다. 전자에서는 통신자마다 서로 다른 modulus를 가지는

† 이 논문은 1988년도 문교부지원 한국학술진흥재단의 자유공모과제 학술 연구비에 의하여 연구되었음.
* 대구공업전문대학
** 경북대학교 전자공학과

반면에 후자는 모든 가입자들이 동일한 유한체를 사용하므로 실제적인 구현에 효과적이다.

D-H 방법을 구현하는데서도 유한체의 정규기저(normal basis)를 이용하면 고속으로 멱승을 할 수 있다[4,5]. 이는 정규기저를 사용한 멱승에서 대부분의 연산과정이 순환이동(cyclic shift)으로 수행되고 나머지 일부분의 연산만이 곱셈으로 처리되기 때문이다. GF(q)의 승산기를 하드웨어적으로 구현하기 위해서는 q를 2의 멱수로 하는 것이 적합하다. 또한 고속초리를 위해 승산기내의 EX-OR gate 수가 최소가 되는 정규기저의 사용이 요구된다[8,9].

이러한 정규기저의 발굴에는 일반적으로 많은 시간이 필요하므로 보다 효과적인 발굴 알고리듬의 개발이 요구된다. Peterson은 GF(2^m)에서 m≤16까지 기약다항식, 원시다항식 및 정규기저 유무에 대하여 분류하였다[10]. m≤15까지 원시다항식 및 정규기저를 갖는 원시다항식의 갯수를 조사한 연구등도 있다[11-16]. 그리고 Vanstone 등은 어떤 특정한 조건들을 가진 m에 대하여 최적 정규기저(optimal normal basis)를 구하였다[9].

본 논문에서는 유한체에서의 멱승을 사용하는 여러 D-H형 공용키이분배 프로토콜들의 문제점을 비교 고찰하고, 개선된 공용키이분배 프로토콜에 대해 알아본다. 개선된 알고리듬을 GF(2^m)의 정규기저를 사용하여 소프트웨어적으로 구현한다. 여기에 고속 멱승을 위한 정규기저의 발굴이 요구되는데 이를 위한 효과적인 발굴 프로그램을 개발한다. 개발된 프로그램으로 구한 고속 정규기저로 구현한 개선된 공용키이분배 시스템을 시뮬레이션 한다.

2. D-H형 키이분배 프로토콜

D-H 방법의 원형에서 두 통신자 A와 B 사이의 공통키이를 생성하는 과정은 다음과 같다. 첫째 가입자 A가 비밀 불규칙 정수 X_A, $1<X_A<q-1$,를 발생시키고, 공개할 $Y_A=a^{X_A} \bmod q$를 계산한다. 또한 통신할 가입자 B도 비밀 불규칙 정수 X_B, $1<X_B<q-1$,를 발생시키고, 공개할 $Y_B=a^{X_B} \bmod q$를 계산한다. 둘째 두 통신자는 일반적인 전송로를 통해 서로 Y_A와 Y_B를 주고 받는다. 혹은 공개하되 신뢰되도록 한다. 세째 두 통신자 각각 받은 Y_B와 Y_A에 비밀 정수 X_A와 X_B를 멱승하여 공통키이 $K_{AB}=a^{X_A \cdot X_B} \bmod q$를 생성한다.

두 통신자는 K_{AB}를 세션키이로 사용하여 정보데이타를 암호화한다. 이 키이는 어느 한사람에 의하여 특정한 형태로 변형될 수 없음을 알 수 있다. Y_B는 B의 비밀키이 X_B에 의해서만 생성되므로 키이 K_{AB}에 의하여 통신이 이루어진다면 가입자 B의 인증문제가 해결된다. 그러나 D-H 방법을 변형하지 않고 그대로 공통키이 K_{AB}를 암호화에 사용한다면 다음의 문제들이 발생한다. 공통키이를 발생시킬 때마다 같은 키이가 생성되므로 모든 세션키이들이 동일하다. 따라서 한번만이라도 세션키이가 불법 유출되면 더 이상 두 통신자간의 비밀 정보교환을 불가능하다. 또한 세션키이 노출여부를 정확히 알 수 없으므로 통신자들은 암호문이 분석되고 있음을 판단할 수 없다. 이를 방지하기 위하여 정기적으로 비밀 및 공개키이를 변환시키는 방법을 사용할 수도 있지만 통신망 가입자가 많을 경우 키이 관리에 어려움이 많다. 그 대책으로는 매번 서로 다른 세션키이를 생성시키는 방법이 이용될 수 있다. 다음으로 키이관리 센터를 완전히 신뢰할 수 없을 경우에는 키이관리 센터에서는 비밀 및 공개키이를 이용하여 모든 세션키이를 생성할 수가 있다. 따라서 이 센터가 모든 암호문을 복호할 수 있게 되고, 또한 불법 침입자가 키이관리 센터로부터 필요한 비밀키이를 불법채취하여 암호문을 분석할 수 있다.

D-H 방법에서 생성된 K_{AB}에 시간 변수 i를 도입한 $(K_{AB})^i$을 세션키이로 사용하면 세션키이가 다를 수 있다. 그러나 여기서 다른 시간 변수 i+j에 대한 $(K_{AB})^{i+j}$와 $(K_{AB})^i$를 안다면 곧 K_{AB}가 해독될 수 있는 문제점이 있다.

다른 방법으로는 두 통신자가 각각 임의의 불규칙 변수 R_A와 R_B를 발생시켜 교환하여 세션키이로 $K'=(K_{AB})^{R_A} \cdot (K_{AB})^{R_B}$를 사용할 수 있다. 이 경우에

도 K′만 알 수 있으면 제삼자 C가 적법한 통신자 B로 가장해서 A와 K′를 세션키이로 공유 할 수 있다. 즉 적법 통신자 A가 세션키이를 생성하기 위하여 B에게 R'_A를 보내면 이때 B로 가장한 C는 $R_A+R_B-R'_A$를 응답한다. 그러면 C가 알고 있는 K′를 생성하여 사용하게 되는 경우가 발생하게 된다. 이의 대책으로는 세션키이의 재사용 여부를 판별할 수 있는 프로토콜을 포함시키든지, 상호교환되는 불규칙 변수 R_A와 R_B를 다른 가입자가 모르게 변형시켜 교환하는 방법이 있을 수 있다.

Yacovi[17]는 보안도가 높은 D-H형 키이 분배 프로토콜을 제시하였다. 두 통신자 A와 B는 각각 비밀키이 X_A와 X_B 그리고 공개 키이 Y_A와 Y_B를 가지고 있다. 세션키이를 생성하기 위하여 A와 B는 각각 불규칙 임의의 정수 R_A와 R_B, $0<R_A$, $R_B<q$,를 발생시키고 $W_A=R_A+X_A$와 $W_B=R_B+X_B$를 서로 교환한다. 최종적으로 각 통신자는 K_{AB}(A의 경우 $K_{AB}=(a^{W_B}Y_B^{-1})^{R_A}=a^{R_AR_B} \bmod q$)를 공통키이로 생성하여 사용한다. 여기서는 독립적이고 또한 서로 다른 세션키이가 생성되므로 D-H 방법을 원형대로 적용했을 때의 첫번째 문제가 해결된다. 그러나 두 번째 문제는 해결되지 않는다. 왜냐하면 키이관리센터에서 공개적으로 교환하는 W_A에서 X_A를 빼면 R_A를 알 수 있고 같은 방법으로 R_B도 계산할 수 있기 때문이다. Yacobi의 프로토콜에서 공개키이 a^{S_A}와 a^{S_B} 대신에 a^{-S_A}와 a^{-S_B}를 각각 공개하면 역수 계산이 없어져 보다 개선될 수 있다.

Yacovi의 방법보다 계산이 간편하고, 또한 해결하지 못한 문제를 해결할 수 있는 프로토콜을 소개한다[18]. 그 전체 구성은 3단계로 이루어지며 개괄하면 다음과 같다.

첫째 두 통신자 A와 B는 각각 비밀키이 S_A와 S_B를 발생하고 P_A와 P_B를 공개한다. 여기서 $P_A=a^{S_A}$이고 $P_B=a^{S_B}$이다.

둘째 A와 B는 각각 불규칙 정수 R_A와 R_B를 발생시켜 $X_A=a^{R_A}$와 $X_B=a^{R_B}$를 구한 후 $Z_A=X_A\cdot K_{AB}$와 $Z_B=X_B\cdot K_{AB}$를 계산하여 공개된 전송망으로 서로 교환한다. 여기서 $K_{AB}=(P_A)^{S_B}=(P_B)^{S_A}$이다.

세째 두 통신자는 각각 동일한 키이 $Z_{AB}=(Z_B\cdot K_{AB}^{-1})^{R_A}=(Z_A\cdot K_{AB}^{-1})^{R_B}=a^{R_AR_B}$를 얻는다.

위의 프로토콜에서는 각각의 R_A와 R_B만 같지 않으면 생성되는 모든 키이들이 서로 다르다. 또한 S_A와 S_B가 노출되어도 세션키이는 분석되지 않는다. Yacobi 방법에서는 비밀키이를 알면 불규칙 정수를 즉시 알아낼 수 있지만, 여기서는 멱승하여 전송하기 때문이다.

제 4장에서는 위의 프로토콜에 정규기저를 사용하여 소프트웨어로 구현하여 컴퓨터 시뮬레이션을 한다.

3. 정규기저 발굴

정규기저를 발굴하기 위하여는 먼저 GF(2^m)를 구성할 수 있는 기약/원시다항식을 구한 후 그 다항식이 정규기저를 갖는지를 조사하여야 한다. 또한 그 정규기저의 승산행렬에서의 1의 수도 조사되어야 한다. 이 과정은 많은 시간이 요구되므로 고속으로 처리할 수 있는 알고리듬이 요구된다. 소개하는 알고리듬은 conjugates를 이용하여 minimal 다항식을 구하는 방법으로 기약다항식을 발굴한다. 이 방법은 GF(2^m)에서 conjugates를 이루는 원소들의 다항식만 쉽게 구할 수 있으면 기약다항식의 고속 발굴 작업에 이용될 수 있다[19]. 따라서 2의 지수승에 해당하는 각 원소를 먼저 구한후 이 원소들을 이용하여 필요한 conjugate 원소들을 직접 구한다. GF(2^m)=$\{0, 1, \alpha, \alpha^2, \cdots, \alpha^{m-1}\}$에서 임의의 원소 α^i의 기약다항식을 구하는 과정은 다음과 같다.

첫째 모든 원소 중에서 $0, 1, \alpha, \alpha^2, \cdots, \alpha^{2m-2}$까지의 원소를 구한다.

둘째 $\alpha^{2^0}, \alpha^{2^1}, \alpha^{2^2}, \cdots, \alpha^{2^{m-1}}$를 구한다. ($\alpha^{2^k}=\alpha^{2^{k-1}}\cdot\alpha^{2^{k-1}}$) 관계를 이용하면 쉽게 구할 수 있다.

세째 $\alpha^i=\alpha^{l_{r-1}2^{r-1}+l_{r-2}2^{r-2}+\cdots+l_12^1+l_02^0}$ (l_j=GF(2))로 표현 가능하므로 둘째에서 구한 α^{i2^k}를 이용하여 α^i를 구한다.

표 3.1 기약다항식 및 원시다항식에 대한 조사 결과

m		TNO	TNB	PMB	NNB
2	기 약 다 항 식	1	1	7	3*
	원 시 다 항 식	1	1	7	3*
3	기 약 다 항 식	2	1	15	5*
	원 시 다 항 식	2	1	15	5*
4	기 약 다 항 식	3	2	37	7*
	원 시 다 항 식	2	1	31	9
5	기 약 다 항 식	6	3	67	9*
	원 시 다 항 식	6	3	67	9*
6	기 약 다 항 식	9	4	163	11*
	원 시 다 항 식	6	3	163	11*
7	기 약 다 항 식	18	7	345	19
	원 시 다 항 식	18	7	345	19
8	기 약 다 항 식	30	16	651	21
	원 시 다 항 식	16	7	651	21
9	기 약 다 항 식	56	21	1563	17*
	원 시 다 항 식	48	19	1563	17*
10	기 약 다 항 식	99	48	3777	19*
	원 시 다 항 식	60	29	3023	37
11	기 약 다 항 식	186	93	6435	21*
	원 시 다 항 식	186	93	6435	21*
12	기 약 다 항 식	335	128	17777	23*
	원 시 다 항 식	144	52	14747	41
13	기 약 다 항 식	630	315	32231	45
	원 시 다 항 식	630	315	32231	45
14	기 약 다 항 식	1161	448	71403	27*
	원 시 다 항 식	756	291	71403	27*
15	기 약 다 항 식	2182	675	151265	45
	원 시 다 항 식	1800	562	151241	53
16	기 약 다 항 식	4080	2048	336657	85*
	원 시 다 항 식	2048	1017	336657	85*
17	기 약 다 항 식	7710	3825	677253	81
	원 시 다 항 식	3855	3825	677253	81
18	기 약 다 항 식	14532	5376	1777777	35*
	원 시 다 항 식	7776	2870	1763717	87
19	기 약 다 항 식	27594	13797	3204523	117
	원 시 다 항 식	27594	13797	3204523	117
20	기 약 다 항 식	52377	24576	7154113	63
	원 시 다 항 식	24000	11255	6363235	73
21	기 약 다 항 식	99858	27783	16561061	95
	원 시 다 항 식	84672	23597	15040735	125
22	기 약 다 항 식	190557	95232	35362245	63
	원 시 다 항 식	120032	59986	31711221	81
23	기 약 다 항 식	364722	182183	64200721	45*
	원 시 다 항 식	356960	178259	64200721	45*
24	기 약 다 항 식	698870	262144	157401651	105
	원 시 다 항 식	276480	103680	172566533	171
25	기 약 다 항 식	1342176	629145	322317037	93
	원 시 다 항 식	1296000	607522	322317037	93
26	기 약 다 항 식	2580759	1290240	715600067	51*
	원 시 다 항 식	1719900	859849	715600067	51*

TNO : 전체 다항식 수
TNB : 정규기저를 갖는 다항식 수
PMB : 최적(최소) 정규기저를 갖는 다항식
NNB : 최적(최소) 정규기저에서의 1의 수
* : 최적 정규기저

네째 α^{i2^1}은 $\alpha^i \cdot \alpha^i = \alpha^{i2^1}$로부터 구한다. 마찬가지로 α^{i2^2}는 $\alpha^{i2} \cdot \alpha^{i2} = \alpha^{i2^2}$로부터 구한다. 계속해서 $\alpha^{i2^{m-1}}$까지 구한다.

다섯째 네째까지에서 구한 각 원소들을 식을 전개하여 i(α^i)를 구한다. 어떤 기약다항식이 원시다항식인지의 여부는 다음 정리 3. 1를 이용하여 판별한다.

정리 3.1 원시다항식의 근은 2^m-1의 order를 가지며 모든 근들의 order는 동일하다.

즉 기약다항식의 근 $\alpha^i \in GF(2^m)$는 알고 있으므로 그 근의 order를 구하여 원시다항식 여부를 판별하는 방법이다. 임의의 근 α^i의 order는 식 (3.1)과 같다.

$$\alpha^i\text{의 order } e = \frac{2^m-1}{\gcd(i,\ 2^m-1)} \qquad (3.1)$$

1) $\gcd(i, 2^m-1)=1$면 본 기약다항식은 원시다항식이다.

2) $\gcd(i, 2^m-1)=k$ ($k\neq1$ 정수)면 본 기약다항식은 원시다항식이 아니다.

위와 같은 방법으로 원시다항식을 판별하는데는 원시원의 지수승 i와 2^m-1의 최대공약수 즉 $\gcd(i, 2^m-1)$만 계산하면 된다. 그리고 본 연구의 알고리듬에서는 i와 2^m-1은 처음부터 2진법으로 표현되어 있으므로 2진 Euclidean 알고리듬을 적용할 수 있어 m의 값이 큰 경우에도 빠른 속도로 최대공약수를 구할 수 있다.

소수 p와 정수 m>0에 대해 GF(p) 상에서 (α^{2^0}, α^{2^1}, α^{2^2}, …, $\alpha^{2^{m-1}}$)가 GF(p^m)의 기저가 되면 이를 정규기저라 한다. 그리고 그 다항식의 근들이 서로 일차독립이면 정규기저가 존재한다. 즉 다음 식 (3.2)에서 행렬 A의 역행렬이 존재하면 GF(2^m)내의 임의의 원소는 정규기저를 사용하여 표현될 수 있다.

$$\begin{bmatrix} \alpha^{2^{m-1}} \\ \alpha^{2^{m-2}} \\ \cdot \\ \cdot \\ \alpha \end{bmatrix} = A \cdot \begin{bmatrix} \alpha^{m-1} \\ \alpha^{m-2} \\ \cdot \\ \cdot \\ \alpha \end{bmatrix} \qquad (3.2)$$

정규기저로 표현된 어떤 원소를 $b_0\alpha^{2^0}+b_1\alpha^{2^1}+b_2\alpha^{2^2}+\cdots+b_{m-1}\alpha^{2^{m-1}}$라 하면 이 원소의 자승은 $b_{m-1}\alpha^{2^0}+b_0\alpha^{2^1}+b_1\alpha^{2^2}+b_2\alpha^{2^3}+\cdots+b_{m-2}\alpha^{2^{m-1}}$가 되어 실제로 각 항의 계수를 오른쪽 순환이동 시킨 것과 같다.

표 3.1은 각 m의 값에 따른 각종 결과를 나타낸 것이다. 다항식의 표현은 8진법으로 표시되었으며 예로 m=7에서의 345는 그 다항식의 각 항의 계수가 (11100101)이며 실제 다항식은 $x^7+x^6+x^5+x^2+1$이다.

4. 공용키이분배 시스템의 시뮬레이션

본 시뮬레이션에서는 3절에서 소개한 공용키이분배 프로토콜을, 정규기저를 사용하여 소프트웨어적으로 구현하여 공통 세션키이를 생성하는 과

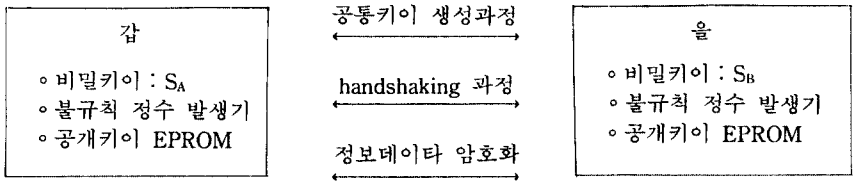


그림 4. 1. 시뮬레이션 과정 선도

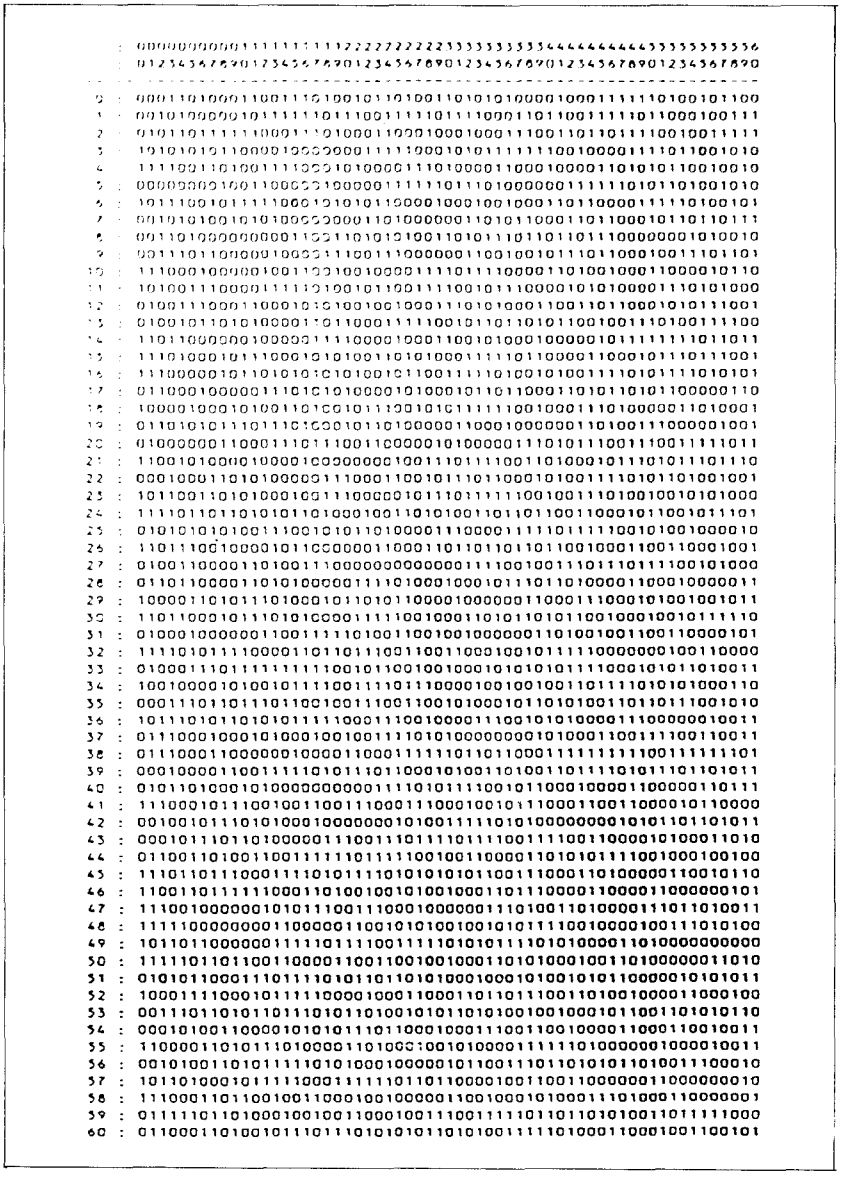


그림 4. 2. GF(2^{61})의 승산행렬

정으로 3절 후반부의 첫째 과정에서 세째 과정까지에 해당한다. 본 시뮬레이션에서는 편이상 Mersenne 소수인 m=61를 선택하여 유한체 GF(2^{61})을 사용한다.

시뮬레이션에 사용한 GF(2^{61})에서의 승산기와 공용키이 생성 알고리듬은 C-언어로 구현하였으며 32 bit CPU Concrrent Computer MC6600에서 수행하였다.

그림 4. 1은 두 통신자 갑순이(이하 갑이라 칭함)와 을돌이(이하 을이라 칭함)간에 이뤄지는 시뮬

레이션 과정을 나타낸 것이다.

그림 4. 2는 GF(2^{61})의 정규기저를 갖는 기약다항식(이 경우는 원시 다항식) $p(x)=x^{61}+x^{60}+x^{58}+x^{55}+x^{53}+x^{50}+x^{48}+x^{45}+x^{43}+x^{40}+x^{38}+x^{35}+x^{33}+x^{30}+x^{28}+x^{25}+x^{23}+x^{20}+x^{18}+x^{15}+1$ 사용하여 구한 승산행렬이다.

4.1 공통키이 생성 과정

• 첫째 과정 :

각 가입자는 비밀키이 S_i를 만들고, 그림 4. 2의 승산기로 공개키이 $P_i=(a)^{Si}$를 구한다. 여기서 i는 가입자를 지칭한다. 예로써, $S_A=49=2^5+2^4+1$이면, $a^{49}=R^5(a)\cdot R^4(a)\cdot a$이 된다. 여기서 원시원 a는 임의로 선택되었으며, 그 역수는 $a^{-1}=a^m-2$을 이용하여 구한다. 이렇게 구하여 표 4.1에 본 시뮬레이션에 사용할 공개키이를 예시하였다. 이는 모든 가입자에게 공개된다.

표 4.1 공개키이 EPROM*

가입자 명	공개키이
갑(갑순이)	$700137241688796247_{(10)}$
을(을돌이)	$588801269445576691_{(10)}$
병(아무개)	·
·	·
·	·

* 갑의 비밀키이 $S_A=510131$
을의 비밀키이 $S_B=480312$
원시원 $a=174D6914D4D3A8A5_{(16)}$

• 둘째 과정 :

갑과 을은 불규칙 정수 발생기를 사용하여 R_A와 R_B를 발생한다.

$R_A=3704794018$(비밀)

$R_B=5013483$(비밀)

다음은 $X_A=a^{R_A}$, $X_B=a^{R_B}$, 및 $X_{AB}=(P_A)^{S_B}=(P_B)^{S_A}$ 구한후

$X_A=9920FC098CB24C_{(16)}$ (비밀)

$X_B=97873D6AC96436E_{(16)}$ (비밀)

$K_{AB}=2A73A2D5A436E10_{(16)}$ (비밀)

$Z_A=X_A\cdot K_{AB}$와 $Z_B=X_B\cdot K_{AB}$를 구한다. 그 결과는

$Z_A=127CF8AB813E646C_{(16)}$ (공개)

$Z_B=1B706388F8461034_{(16)}$ (공개)

이다. 이를 공개된 전송망으로 서로 교환한다.

• 세째 과정 :

갑은

$Z_{AB}=(Z_B\cdot K_{AB}^{-1})^{R_A}=1787EE848F599B84_{(16)}$

을 얻고, 을도 같은 방법으로 Z_{AB}를 얻어 공통키이를 생성한다.

4.2 고 찰

GF(2^{61})에서 수행한 컴퓨터 시뮬레이션을 통하여, 유도된 정규기저 승산기와 제안된 키이 분배 프로토콜이 정확하게 작동함을 확인하였다. GF(2^{61})외에 GF(2^{31})에 대해서도 수행하여 확인하였다. 여기에 사용된 원시다항식은 $p(x)=x^{31}+x^{30}+x^{26}+x^{25}+x^{24}+x^{23}+x^{21}+x^{19}+x^{18}+x^{17}+x^{15}+x^{14}+x^{12}+x^{10}+x^8+x^7+x^5+x^3+x^2+x+1$이다.

보안을 충분히 유지하기 위하여 m>1000이 바람직하다. 이 경우 고속 승산을 위하여, 1의 수가 최저인 승산행렬을 갖는 정규기저를 발굴하여 사용해야 한다. 그리고 채택된 관용 암호시스템에 따라서 생성된 공통키이 Z_{AB}를 적절한 형태로 변환하여 관용 암호시스템의 키이로 사용해야 할 것이다.

5. 결 론

구현한 D-H형 공용키이 분배 프로토콜은 키이 관리 센터에서 통신자들의 비밀키이를 알고 있다 하더라도 세션키이는 알 수 없기 때문에 기존의 여러 D-H형 공용키이 분배 프로토콜 보다 안전하다. 이를 고속으로 처리하기 위한 정규기저의 발

굴을 위한 전산 프로그램도 개발하였다. 개발된 알고리듬은 conjugates를 이용하는 minimal 다항식을 구하는 방법으로 기약다항식을 찾은 후 정규기저를 발굴한다. 이 알고리듬에서는 식의 전개에 필요한 벡터들의 다항식을 직접 구하므로 쉽게 기약다항식을 찾을 수 있어 고속으로 정규기저를 발굴할 수 있다. GF(2^{61})의 한 정규기저를 사용하여 공유키이 분배 프로토콜을 소프트웨어적으로 구현한 후 시뮬레이션하여 그 동작 과정을 확인하였다.

참 고 문 헌

1. W. Diffie and M.E. Hellman, "New directions in cryptography," IEEE Trans. on Inform. Theory, Vol. IT-22, pp.644-654, Nov. 1976.

2. R.L. Rivest, A. Shamir, and L. adleman, "On Digital Signatures and Public Key Crpytosystems," Comm. Ass. Comput. Math., Vol. 21, pp.120-126, Feb. 1978.

3. M.E. Hellman, "An Overview of Public Key Cryptography," IEEE Comm. Society Mag., Vol. 16, No. 6, pp. 24-32, Nov. 1978.

4. National Bureau of Standards, Data *Encryption standard*, U.S. FIFP PUB 46, pp.1-18, 1977.

5. J.L. Massey and J.K. Omura, Patent Application of "Computational Method and Apparatus for Finite Field Arithmetic," 1981.

6. C.C. Wang, T.K. Troung, H.M. Mao, L.T. Deutch, J.K. Omura, and I.S. Reed, "VLSI Architecture for computing Multiplication and Inverse in GF(2^m).," IEEE Trans. on Compu. Vol. C-34, No. 8, pp.709-716, Aug, 1985.

7. S. Pohlig and M.E. Hellman, "An Improved Algorithm for computing Logarithms over GF(p) and Its Cryptographic Significance," IEEE Trans. Inform. Theroy, vol. IT-24, No. 1, 1978.

8. T. Itoh and S. Tsujii, "A Fast Algorithm for Computing Multiplicative Inverses in GF(2^m) Using Normal Bases," Information and Computation, Vol. 78, No. 2, pp.171-177., Aug, 1988.

9. R.C. Mullin, I.M. Onyszchuk, S.a. Vanstone and R.M. Wilson, "Optimal Normal Bases in GF(p^m)," Discrete applied Mathematics, Vol. 22, pp.149-161, 1989.

10. W.W. Peterson and E.J. Weldon, *Error-Correcting Codes*, *New York Wiley*, 1961.

11. 원동호 "GF(2)상의 정규기저를 갖는 원시 다항식 분류에 관한 연구" 정보보호 워크샵, 1989.

12. W. Stahnke, "Primitive Binary Polynomials," Math of Computation, Vol. 27, No. 124, pp.977-980, Oct, 1973.

13. E.J. Watson, "Primitive Polynomials(Mod 2)," Math. Comp., Vol. 16, pp.368-369, 1962.

14. E.R. Berlekamp, "factoring Polynomials Over Finite Fields," The Bell System Technical Journal, Oct. 1967.

15. S. Mossige, "Table of Irreducible Polynomials Over GF(2) of Degree 10 Through 20," Math. Comp., Vol. 26, No. 26, Oct. 1972.

16. R.E. Blahut, *THEORY AND PRACTICE OF ERROR CONTROL CODES*, *Addison Wesley*, 1983.

17. Y. Yacobi and Z. Shmueiy, "On Key Distributions," Proc. CRYPTO '89, pp. 335-346, 1989.

18. 문상재, 이필중 "키이 분배 프로토콜의 제안" 정보보호 워크샵, 1990.

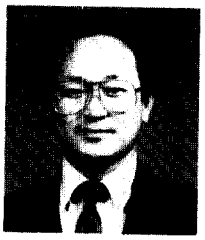
19. 이창순, 백기진, 문상재, "GF(2)상의 최적 정규기저를 갖는 기약다항식의 발굴에 관한 연구," 통신정보합동학술대회 논문집, 제 1 권, pp.36-42, 1991.

□ 著者紹介

李 昌 淳(正會員)

경북대학교 전자공학과(학사)
경북대학교 전자공학과(석사)
경북대학교 전자공학과 박사과정
현재 대구공업전문대학 조교수

주관심분야 : 암호이론 및 컴퓨터 네트워크 등



文 相 在(正會員)

通信情報保護學會誌 創刊號 參照
현재 경북대학교 전자공학과 교수