https://doi.org/10.33778/kcsa.2019.19.5.037

영상보안시스템에서의 데이터 보호를 위한 ECC(Elliptic Curve Cryptography) 연산알고리즘 비교분석

김종민*, 추현욱**, 이동휘***

요 약

기술의 발전을 통해 기존 영상보안시스템들이 아날로드 방식의 CCTV에서 네트워크 기반 CCTV로 교체되어지고 있다. 이러한 기술 변화로 인해 네트워크를 이용한 도청 및 해킹에 대한 공격이 발생하게 되면 영상정보유출로 인해 개인 및 공공기관에 대한 피해는 막대하다 할 수 있다. 따라서 이러한 피해발생을 해결하기 위해 본 논문에서는 데이터 통신 과정에서 영상정보를 보호할 수 있는 ECC(Elliptic Curve Cryptography) scalar multiplication algorithm들을 비교 분석하여 영상시스템에서 최적화된 ECC scalar multiplication algorithm을 제안하고자 한다.

A Comparative Analysis on ECC(Elliptic Curve Cryptography) Operation Algorithm for Data Protection in Video security System

Jongmin Kim*, Hyunwook Choo**, DongHwi Lee***

ABSTRACT

Video security systems change from analog based systems to network based CCTVs. Therefore, such network based systems are always exposed not only to threats of eavesdropping and hacking, but to personal damage or public organizations' damage due to image information leakage. Therefore, in order to solve the problem, this study conducts a comparative analysis on proposes the optimal ECC(Elliptic Curve Cryptography) scalar multiplication algorithms for image information protection in data communication process and thereby proposes the optimal operation algorithm of video security system.

Key words : ECC, ECDH, ECDSA, calar multiplication algorithm

접수일(2019년 12월 00일), 수정일(1차: 2019년 12월 00일),
게재확정일(2019년 12월 0일)

* 경기대학교/융합보안학과 교수
** 이지스마트팜/개발팀장
*** 동신대학교/융합정보보호학과 교수(교신저자)

1. 서 론

영상보안시스템은 CCTV 중심의 폐쇄형 구조에서 오픈망을 사용하는 IP 기반의 개방형 구조로 발전하고 있다[1]. 이러한 CCTV 등의 영상관제시스템은 방범기능 외에도 지능형 교통망 체계 (ITS : Intelligent Traffic System)와 연계되거나 소방, 경찰 등과도 밀접한 연관을 가지고 운영되고 있다. 그러나 현재의 CCTV 장치 등은 보안이 매우 취약한 상태에서 관리되고 있기 때문에 악의적인 공격으로 CCTV의 순기능이 운영되지 않을 수 가능성이 있으며, 특히 국가 안보나 국민의 안전의 문제와 연결되는 국가기반의 관제 시스템이 해킹되는 경우, 해당 시스템의 동작불능, 오작동, 등과 같은 문제가 발생함으로써 심각한 결과를 초래한다[2][3][4].

기존의 CCTV 영상보안 시스템은 CCTV 카메라로부터 Server로 데이터가 전송되면 Server는 영상데이터를 압축 알고리즘을 통해 영상데이터를 압축해 관제센터로 데이터를 전송한다. 이러한 기존의 방법에서 Server에서 관제센터로 전송되는 구간에서 데이터 조작, 파기 등의 위험에 노출되게 된다.

따라서 본 논문에서는 데이터 통신 과정에서 영상정보를 보호할 수 있는 ECC(Elliptic Curve Cryptography) scalar multiplication algorithm들을 비교 분석하고자 한다.

2. 관련연구

2.1 영상보안시스템

영상보안시스템은 영상정보를 특정의 목적으로 특정 사용자에게 전달해 주는 시스템으로 가정, 학교, 회사, 공공기과 등 산업 전방에 사용되어지고 있다. 이렇게 저장된 개인정보나 기업정보들은 범죄예방, 범죄에 대한 증거, 교통정보제공 등의 중요한 정보로 사용되고 있으며, 영상데이터 관리의 중요성이 부각되고 있다..

최근에는 이러한 영상보안시스템의 안전한 관리를 위해 다양한 기술이 도입되고 있으며, 해킹방지를 위해 전용망 구축, 데이터 전송상에서의 암호화 기술 도입 등을 하고 있다.

대부분의 영상 암호화 방법에는 RSA, AES와 같은 방식을 적용하고 있다.

공개키 기반의 AES의 경우 처리속도와1286~256비트 키를 적용가능하기 때문에 보안성 부분에서 뛰어나며 공개된 알고리즘이라 누구나 사용가능하다. 하지만 이렇게 공개된 치를 사용함으로써 키 전달과정에서 키 정보가 노출될 가능성이 존재하게 된다. 키 전달과정에서의 문제를 해결하기 위한 방식이 비대칭형 암호화이며, RSA가 이에 해당한다. RSA는 AES보다 보안성으로 안전하지만 텍스트 데이터 암호화를 영상에 직접 적용을 하기 때문에 계산량이 많아짐으로써 암호화시 걸리는 시간이 증가하기 때문에 영상보안시스템에 적용시키기에는 한계점이 있다.

2.2 타원곡선 알고리즘

타원곡선 알고리즘은 각각 1985년 Miller와 Koblitz가 독립적으로 제안한 공개 키 기반의 암호 알고리즘으로 유한체상에서 정의되는 타원곡선이산대수문제의 어려움에 근거하여 강력한 보안성을 제공하는 알고리즘으로 현재까지도 타원곡선 이산대수문제를 효과적으로 공격할 수 있는 방법이 발견되고 있지 않아 다른 공개 키 알고리즘에 비해 짧은 키 크기로도 동등한 보안성을 보장할 수 있으며 짧은 키 길이로 인하여 타 공개 키 알고리즘보다 오버헤드나 연산속도로부터 비교적 자유로울 수 있음으로 인하여 하드웨어의 자원이 제약된 모바일 단말기기 또는 스마트카드 같은 분야에서 크게 각광받고 있다[5][6].

<표 1> 보안강도에 따른 공개 키 암호 알고리즘의 분류[7]

보안강도 (bit)	인수분해문제 (bit)	이산대수문제(bit)		타원곡선 (bit)
		공개키	개인키	
80	1,024	1,024	160	160
112	2,048	2,048	224	224
128	3,072	3,072	256	256
192	7,680	7,680	384	384
256	15,360	15,360	512	512

2.2.1 EC-ElGamal

타원곡선은 자체적으로 암호연산을 수행할 수 없

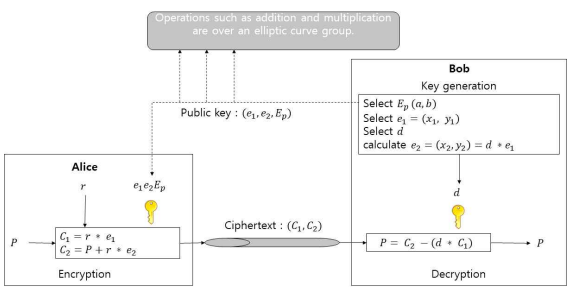
기 때문에 DH, DSA, El gamal 같은 다른 공개키 암호기법을 타원곡선 상에서 수행하는 방식을 적용하고 있다[8][9].

<표 2> Elgamal 암호기법을 이용하는 타원곡선 암복호화 parameter

	비밀 키로 충분히 큰 소수
r	암호화에 사용되는 충분히 큰 소수
e	첫 번째 공개 키로 기본 점
e_2	두 번째 공개 키로 $dG = e_2$
n	G의 위수
C_1	첫 번째 암호문
C_2	두 번째 암호문
P	평문(Plain Text)

<표 3> Elgamal암호기법을 이용하는 타원곡선 암복호화 과정

1단계	Alice는 Bob에게 비밀통신을 위한 공개키를 요구
2단계	Bob은 적절한 $GF(p)$, n, a, b, d를 선택하여 e_1, e_2를 생성한 후 d는 간직하고 e_1, e_2를 Broadcast함
3단계	Alice는 Bob의 공개 키 e_1, e_2를 이용해 적절한 수수 r과 메시지 P에 적합한 타원곡선 (x, y)를 선택해 C_1, C_2를 생성해 Bob에게 전달
4단계	Bob은 C_1, C_2와 비밀 키 d를 이용해 평문 P를 복호화

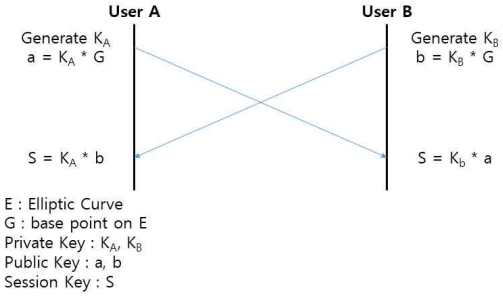


[그림 1] Elgamal 암호기법을 이용한 타원곡선 암복호화[10]

2.2.2 ECDH(Elliptic Curve Diffie-Hellman)

ECDH는 DH(Diffie-Hellman) 키 교환 알고리즘에 타원 곡선 암호 방식인 ECC(Elliptic Curve Crypto- graphy) 을 적용한 키 교환 알고리즘이다. ECC 방식을 사용함으로써 기존의 DH 알고리즘과 같은 키 길이에 비해 더 강한 안전성을 보장한다[11].

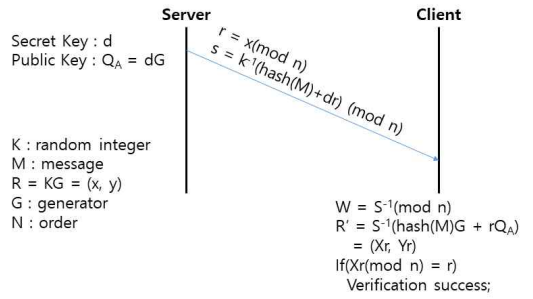
사용자 A와 B는 각각의 개인키 K_A와 K_B를 생성한다. 타원 곡선 상의 기점을 G라고 가정하였을 때, 사용자 A는 자신의 개인키 K_A와 G를 이용하여 공개키 a를계산하고, 사용자 B는 자신의 개인키 KB와 G를 이용하여 공개키 b를 계산한다. 사용자 A와 B는 자신의 공개키 a와 b를 서로에게 전송한다. 사용자 A는 사용자 B로부터 공개키 b를 수신하고 자신의 개인키 K_A와 계산하여 세션 키 S를 얻는다. 마찬가지로 사용자 B는 사용자A로부터 공개키 a를 수신하여 자신의 개인키 K_B와 계산하여 사용자 A와 사용자 B만 아는 공통의 세션 키 S를 얻는다. [그림 1]은 ECDH 키 교환 알고리즘을 이용하여 세션 키를 생성하는 과정을 나타낸다.



[그림 2] Session key generation process using ECDH[12][13]

2.2.3 ECDSA(Elliptic Curve DSA)

ECDSA는 DSA(Digital Signature Algorithm)에 ECC 방식을 적용한 알고리즘이다[12][13][14][15][16]. [그림 2]는 송신자가 ECDSA를 이용하여 전자 서명을 생성하고, 수신자가 전자 서명을 검증하는 과정을 나타낸다.

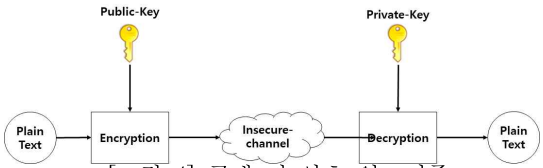


[그림 3] ECDSA digital signature generation and verification process

2.3 기존 공개키 알고리즘

공개 키 알고리즘은 1976년 Whitfield Diffie와 Martin Hellan과 Ralph M erkle이 처음 소개한 기술로써 정수론에 기반을 두고 있다. 공개 키 알고리즘은 대칭 키 알고리즘과는 다르게 암호화, 복호화에 쓰이는 키가 서로 다르며 이는 공개 키, 비밀 키로 구분된다. 대표적인 알고리즘으론 RSA, DSA, ECC등이 있으며 데이터의 기밀성, 인증, 부인방지를 보장한다. 공개 키 알고리즘은 데이터의 암호화에 사용되는 키와 복호화에 사용되는 키를 별도로 구분하기 때문에 대칭 키 알고리즘에서 제공하지 못하는 키 분배와 관리를효율적 제공할 수 있다[17].

대표적인 공개 키 알고리즘 RSA는 매우 큰 수의 인수분해(Integer facto rization) 문제를 기본 원리로 하여 강력한 보안성을 제공하는 알고리즘으로1024-bit에서 AES의 80-bit에 해당되는 보안성을 제공한다. 이러한 점은 공개 키 알고리즘의 단점을 보여주는데 키 교환 문제나 키 탈취에 대해 대칭키 알고리즘보다 안전하지만 충분한 보안성을 확보하기 위해선 키 길이가 커야하고 이는 곧 연산속도의 저하와 많은 하드웨어 자원 소모 등의 문제를 야기한다[18].



[그림 4] 공개 키 암호 알고리즘

2.4 기존 ECC와 기존 암호 알고리즘 비교

ECC 알고리즘은 RSA/DSA에 비해 여러 가지 장점을 가지고 있다. 첫째, 에너지 효율성 및 키 사이즈가 작고 서명의 길이가 짧다는 장점이 있어 IC카드, 무선기기 등에 적용이 가능하다. 둘째, RSA 암호 알고리즘 보다 구현이 용이하며, 보안 강도면에서도 효율성이 높다. 셋째, 타원곡선 상에서 군(Group)을 정의할 수 있고, 다양한 타원곡선을 활용해 다양한 암호 알고리즘 설계가 가능하며, 이산대수 문제의 어려움에 근거를 두고 있어 안전한 암호 알고리즘을 설계할 수 있다[19].

이와 같은 ECC 암호 알고리즘의 장점은 시스템의 부하를 줄이면서도 보안 강도를 높일 수 있다.

<표 4>는 RSA 암호 알고리즘과 ECC암호 알고리즘의 안정성을 비교한 것이다.

<표 4> 암호 알고리즘 안정성 비교[20]

Time to break in MIPS years	RSA Key size (bit)	ECC key size (bit)	RSA/ECC (key size ratio)
10^4	512	106	5 : 1
10^8	768	132	6 : 1
10^{11}	1,024	160	7 : 1
10^{20}	2,048	210	10 : 1
10^{78}	21,000	600	35 : 1

타원곡선 암호 알고리즘은 일반적으로 RSA보다 비교적 높은 안정성 같지고 있으며, 보안효율을 높이기 위한 암호 설계 시 유용하게 사용 가능하다.

3. Scalar Multiplication Algorithm [17, 21]

3.1 Double-and-add Algorithm

double-and-add Algorithm은 타원곡선 알고리즘의 가장 기본적이며, 가장 많이 사용된 방식이다. 알고리즘의 중요한 점은 정수 k를 이진수로 표현한 뒤 연산을 수행하는 방법으로 자리 수를 넘어갈 때 점의 두 배 연산을 수행하고 비트를 검사

하여 1일 경우 기본 점 P와 점의 덧셈 연산을 추가로 수행해준다. 비트가 0일 경우는 추가적인 연산 없이 다음 비트를 판별한다. 이 알고리즘은 k-bit일 때 평균 1.5k번의 점의 덧셈과 점의 두 배 연산이 필요하다.

<표 5> Double-and-add Algorithm

Input : Binary representation of and
Output : $kP = (x, y)$

1. $R \leftarrow P$
2. $For\ i = n-2$ to 0 do
   2.1 $R \leftarrow 2R$ (Doubling)
   2.2 $If\ k = 1$ then $R = R + P$ (Addition)
   2.3 $i \leftarrow i + 1$
3. return $kp \leftarrow R$

3.2 NAF(Non Adjacent Form) Algorithm

NAF Algorithm은 double-and-add Algorithm에서 0이 많으면 많을수록 연산의 성능이 향상된다는 점을 이용하는 알고리즘으로 NAF 변환 알고리즘을 이용해 정수 k를 1, 0, -1의 2진수로 변환 후 변형된 double-and-add Algorithm으로 연산을 하는 방법이다. NAF알고리즘에서 정수 7은 <표 6>과 같은 표현으로 나타낼 수 있는데 여기서 (1 0 0 -1)은 1의 개수를 충분히 줄여준 표현이다. NAF를 이용한 스칼라 곱셈 연산은 기본적으로 double-and-add Algorithm과 동일하고 -1의 연산은 P의 역원 (x, -y)을 더해주는 방식으로 연산이 수행된다[17][21].

<표 6> NAF의 정수 7 표현

NAF(7)의 표현	10진수에서의 표현
$(0\ 1\ 1\ 1)_2$	4 + 2 + 1 = 7
$(1\ 0\ -1\ 1)_2$	8 - 2 + 1 = 7
$(1\ -1\ 1\ 1)_2$	8 - 4 + 2 + 1 = 7
$(1\ 0\ 0\ -1)_2$	8 - 1 = 7

NAF의 기본 원리는 정수 k의 이진 값에서 1을 인접하지 않게 배치하여1의 개수를 줄여나가 double-and-add Algorithm을 수행할 때에 점의 덧셈 연산 횟수를 줄이고 다시 정확한 수로 조합하는데 있다.

<표 7> nteger to NAF(k) Algorithm

Input : Positive Integer k
Output : $NAF(k)$

1. $i \leftarrow 0$
2. While k 1 do
   2.1 $If\ k$ is odd then $k_i \leftarrow 2 - (kmod4),\ k \leftarrow k - k_1$
   2.2 Else $k_i \leftarrow 0$
   2.3 $k \leftarrow k/2,\ i \leftarrow i + 1$
3. return $NAF(k)$

<표 8>NAF를 이용한 스칼라 곱셈 연산 알고리즘

Input : $NAF(k),\ P$
Output : kP

1. $R \leftarrow P$
2. $For\ i = n-2$ to 0 do
   2.1 $R \leftarrow 2R$
   2.2 $If\ k_i = 1$ then $R \leftarrow R + P$
   2.3 $If\ k_i = -1$ then $R \leftarrow R - P$ where $-P = (x_p, -y_p)$
   2.4 $i \leftarrow i - 1$
3. return $kp \leftarrow R$

3.3 Sliding Window Algorithm

Sliding Window Algorithm은 점의 덧셈 연산을 줄임으로써 연산의 효율성을 갖는 알고리즘으로 기존 NAF(k)를 이용하며 최대 w만큼의 그룹연산을 통해 점의 덧셈 연산을 최소화 한다. w가 커질

수록 그룹연산으로 인하여 점의 덧셈이 감소하지만 이 경우 미리 구성하는 i ∈ {1,3, 5,..., 2w-1-1} P가 증가하고 그룹연산의 과정이 복잡하기 때문에 적절한 w를 선정하는 것이 중요하다.

<표 9>Sliding Window Algorithm

Input : Window width , $AF(k) = \sum_{i=0}^{l-1} k_i 2^i$, P
Output : kP

1. Compute $P_i = iP$ for $i \in \{1, 3, ..., 2(2^w-(-1)^w)/3-1\}$
2. $Q \leftarrow$, $i \leftarrow l - 1$
3. Whlie i 0 do
 3.1 *If* k = 0 then $t \leftarrow 1$, $u \leftarrow 0$
 3.2 Else : find the largest t w such that $u \leftarrow (k_i,...,k_{i-t+1})$ is odd
 3.3 $Q \leftarrow 2^t Q$
 3.4 If *IF* u 0 then $Q \leftarrow Q + P_u$
 3.5 Else if u 0 then $Q \leftarrow Q - P_u$
 3.6 $i \leftarrow i - t$
3. return $kp \leftarrow Q$

3.4 Montgomery ladder

montgomery ladder는 기존 double-and-add Algorithm과 같이 매 비트를 확인하지만 매 루프마다 point doubling과 point addition을 수행한다는 것에서 차이점이 있다. 매번 point doubling과 point addition 수행 시 현재 스칼라(n)에 따라 R_0에 point doubling을 저장할지, R_1에 point point doubling을 저장할지 정해지는데, 0인경우 R_0에 point doubling 을 저징하고, R_1에 point addition을 저장한다. 1인 경우엔 R_1에 point doubling을 저장하고, R_0에 point addition을 저장한다. 이때, 항상 point addition은 point doubling보다 1번 더기저(P)를 더한 상태가 된다 ($|R_0 - R_1| = P$).

<표 10>Montgomery ladder Algorithm

Require : P, d

1. $R_1 = 0$
2. $R_2 = P$
3. for $i = m...0$ do
 3.1 *If* d_i = 0 then
 $R_1 = pointadd(R_0, R_1)$
 $R_0 = pointdouble(R_0)$
 3.2 Else $R_1 = pointadd(R_0, R_1)$
 $R_0 = pointdouble(R_0)$
 3.3 end if
 3.4 end for
4. return R_0

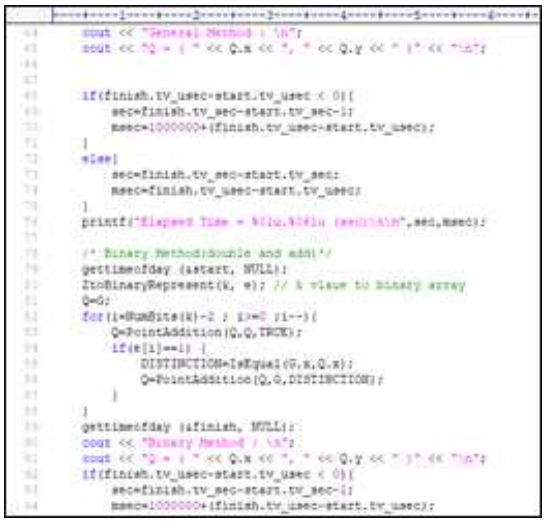
4. 연구결과

Double-and-add Algorithm, NAF(Non Adjacent Form) Algorithm, Sliding Window Algorithm에 대해 구현하여 수행속도를 비교 및 분석하고 가장 효율적인 알고리즘을 도출하였다.

알고리즘 구현의 구성환경 및 구현코드는 다음과 같다.

<표 6> 알고리즘 구현 환경

하드웨어	Intel(R)Core(TM) i7-6700 CPU @ 3.40GHz 3.40 Chz
운영체제	Windows10 Home
프로그래밍 언어	C++
컴파일러	GCC



(그림 1) 구현코드 예제

영상보안시스템에서의 데이터 통신 과정에서 영상정보를 보호할 수 있는 ECC(Elliptic Curve Cryptography) scalar multiplication algorithm(double-and-add Algorithm, NAF(Non Adjacent Form) Algorithm, Sliding Window Algorithm)들의 효율성을 알아보기 위해 수행속도 계산을 통해 비교 및 분석을 진행하였다.

그 결과 <표 7>에서 보듯이 Binary NAF Method가 연구한 알고리즘 중 가장 효율적임을 알 수 있었으며, 그 외 double-and-add Algorithm, NAF(Non Adjacent Form) Algorithm, Sliding Window Algorithm도 암호 시스템에 적용하기에 무리가 없을 것이라 판단되지만, 실제 시스템에서 연산에 대한 프로그램만 쓰는 것이 아니라, 다른 프로그램과 같이 사용되는 것을 감안하여야 한다. 따라서 프로세스 사용률이 낮을수록 좋다는 점에서 Binary NAF Method를 사용해 메모리의 사용량을 최소화해야한다. 이후 IoT 환경에서의 암호시스템을 적용하기 위해 보안 수준과 메모리 효용성이 좋은 상수배 알고리즘의 개선이 불가피해 질 것이며, 장기적인 시각으로 본다면, 이러한 상수배 알고리즘은 앞으로도 많은 연구에 의해 더욱 향상된 성능을 갖는 알고리즘이 개발 되어야 할 것이다.

<표 7> 타원곡선 연산 알고리즘 수행 속도 비교

$Fq),\ q\ =\ p$	1461501637330902918203684832716283019653785059327					160bit
$y\ =x^3+ax+b$	a=1461501637330902918203684832716283019653785059324, b=618161358937170673988121756987436237099350920727					
본점 $G\ =\ (x,\ y)$	(1168983055804381306122964739888353823030159703303, 591673640518579811944247307252990789873540735386)					
G의 위수 n	1461501637330902918203683057840589624783069764883					160bit
	General method	Binary method	Silding Window method	Binary NAF Method	k	Bits of k
1	0.00010875	0.00004005	0.00005085	0.000038	2^4~ 2^5-1	5
2	0.0018889	0.00010425	0.00010655	0.00009165	2^9~ 2^{10}-1	10
3	0.04960995	0.00014135	0.00014675	0.00012065	2^{14}~ 2^{15}-1	15
4	2.15902355	0.0002207	0.00020710	0.00018035	2^{19}~ 2^{20}-1	20
5	-	0.00039295	0.0003577	0.0003078	2^{29}~ 2^{30}-1	30
6	-	0.00048705	0.00044165	0.0004032	2^{39}~ 2^{40}-1	40
7	-	0.00058075	0.0005755	0.00047795	2^{49}~ 2^{50}-1	50

8	-	0.0007018	0.00061575	0.0005488	$^{9}\sim 2^{60}-1$	60
9	-	0.00669665	0.00077225	0.0007456	$2^{69}\sim 2^{70}-1$	70
10	-	0.0010193	0.00088705	0.0007892	$2^{79}\sim 2^{80}-1$	80
11	-	0.00121805	0.00095895	0.00091845	$2^{89}\sim 2^{90}-1$	90
12	-	0.0011867	0.0010798	0.0009848	$2^{99}\sim 2^{100}-1$	100
13	-	0.00127715	0.00119905	0.0010541	$2^{109}\sim 2^{110}-1$	110
14	-	0.00145975	0.0012922	0.0012211	$2^{119}\sim 2^{120}-1$	120
15	-	0.0014942	0.00127925	0.0012731	$2^{129}\sim 2^{130}-1$	130
16	-	0.00161495	0.00143775	0.0014024	$2^{139}\sim 2^{140}-1$	140
17	-	0.0016242	0.00146975	0.00140155	$2^{149}\sim 2^{150}-1$	150
18	-	0.0018653	0.00176065	0.00160435	$2^{159}\sim n-1$	160

5. 결론

Binary Method와 Binary NAF Method, Sliding Window Method를 중심으로 연구하였다. 효율성은 주로 상수배에 소요되는 수행시간과 연관되며, 또한 메모리의 효율적인 사용과도 관계된다. 이와 같은 효율성에 근거하여 각각의 알고리즘에 대해 수행시간을 비교, 분석하였다. 그 결과 Binary NAF Method가 연구한 알고리즘 중 가장 효율적임을 알 수 있었으며, 그 외 Binary Method와 Sliding Window Method도 암호 시스템에 적용하기에 무리가 없음이 판명되었다. 그러나 이러한 각 알고리즘이 근소한 차이의 수행시간을 나타내었다는 것을 볼 때, 실제적인 사용에서의 차이는 거의 느낄 수 없을 것으로 판단되어지며. 장기적인 시각으로 본다면, 이러한 상수배 알고리즘은 앞으로도 많은 연구에 의해 더욱 향상된 성능을 갖는 알고리즘이 개발 되어야 할 것이다.

참고문헌

[1] 한종욱, 조현숙, “영상보안시스템 기술 동향”, 한국정보보호학회논문지, Vol. 19, No. 5, 2009, pp. 29-37.

[2] Taewoong Seo, Myunggyun Jeong, and Changsoo Kim, “A Study on Vulnerabilities of Monitoring and Control System based on IT Convergence Technology,” The 6th International Conference on Multimedia Information Technology and Applications, pp. 245-247, 2010.

[3] 유지영, 이재일, “신규 IT 서비스 도입 확산을 위한 정보보호,” 한국정보처리학회 정보처리학회지, 제17권, 제2호, pp. 10-17, 2010.

[4] 서태웅, 이성렬, 배병철, 윤이중, 김창수, "CCTV 보안관제 취약성 및 성능 분석", 한국멀티미디어학회 논문지, Vol. 15, No. 1, 2012, pp. 93-100.

[5] William Stallings, “Cryptography and Network Security Principles and Practices,” 2006.

[6] Neal Koblitz, “A course in Number Theory and Cryptography”, 1994.

[7] 한국인터넷진흥원, “암호 알고리즘 및 키 길이 이용 안내서,” 2013.

[8] Christof paar and Jan Pelzl, “Unnderstanding Cryptography a Textbook for Students and Practitioners”, 2009.

[9] Alfred Menezes, Paul C.van and Scott Vanstone, “HANDBOOK of APPLIED CRYPTOGRAPHY,” June, 1996.

[10] Neal Koblitz, “A course in Number Theory and Cryptography”, 1994.

[11] Carlisle Adams and Steve Lloyd, Effective way for security PKI, Seoul, Infobook, 2003.

[12] Seok-ho Kim, “Comparison and analysis on efficiency of scalar multiplication for Elliptic Curve C

ryptosystem," M. S. dissertation, Korea Maritime and Ocean University graduate school, Busan, 2003.

[13] Yeong-ja Kim, "Design and implementation of a security messenger system using elliptic curve cryptosystem," M. S. dissertation, Chung-Ang University Information graduate school, Seoul, 2004.

[14] IETF Std. RFC 6979, Deterministic Usage of the Digital Signature Algorithm (DSA) and Elliptic Curve Digital Signature Algorithm (ECDSA), IETF, T. Pornin, August 2013.

[15] Song-hwan Lim, "The efficiency analysis of ECDSA using improved element algorithm," M. S. dissertation, Dongguk University Graduate School of International Affairs & Information, Seoul, 2000.

[16] IETF Std. RFC 4050, Using the Elliptic Curve Signature Algorithm (ECDSA) for XML Digital Signatures, IETF, S. Blake-Wilson, G. Karlinger, T. Kobayashi, Y. Wang, April 2005.

[17] William Stallings, "Cryptography and Network Security Principles and Practices", 2006.

[18] 정도, "NAF와 Look-Up 테이블을 이용한타원곡선 스칼라 곱셈 가속화", 한밭대학교, 석사학위논문, 2016.

[19] K. J. Ha, C. H. Seo, D. Y. Kim, "Design of validation System for a Crypto-Algorithm Implementation," Journal of the Korea

[20] https://www.keylength.com/en/4/, June 10, 2018.

[21] https://en.wikipedia.org/wiki/Elliptic_curve_point_multiplication#cite_note-guide-1

〔저자소개〕

김 종 민 (Jongmin Kim)
2010년 체육학사
2012년 경호안전학석사
2015년 산업보안학박사
현 재 경기대학교 융합보안학과 초빙교수

email : dyuo1004@gmail.com

이 동 휘(DongHwi Lee)
2007년 경기대학교 정보보호박사
2011년~2012년 University of Colorado Denver, Dept. of Computer Science and Engineering
현 재 동신대학교 에너지융합대학 에너지응용학부 정보보안전공 교수

email : dhclub@dsu.ac.kr

추 현 욱 (HyunWook Choo)
2011년 성균관대학교 바이오메카트로닉스 학사 졸업
2011년~2016년 LG 전자 휴대폰 사업부 근무
현재 이지스마트팜 개발팀장.

email : calon77@gmail.com