http://dx.doi.org/10.3745/KTSDE.2013.2.11.747

An Elliptic Curve Cryptosystem based on Trust and RBAC to Reduce Security Overhead in Sensor Networks

Hyojin Kim† · Ho-Hyun Park††

ABSTRACT

It is important to reduce unnecessary overhead in sensor network using battery. In addition encryption is important because of necessity of security. Since unavoidable overhead occurs in case of encryption, security and overhead are in trade-off condition. In this paper, we use a concept called trust to reduce the encryption overhead. We reduce overhead by controlling encryption key sizes while maintaining the security level where high and low trust nodes are mixed. We simulated and compared normal encryption and trust value based encryption. As a result, the latter has lower execution time and overhead. If we define a standard of trust levels considering purpose and circumstances of real network, we can use constrained resources efficiently in sensor network.

Keywords : Trust, Elliptic Curve, RBAC, Sensor Network, RC6

센서네트워크의 보안 오버헤드를 줄이기 위한 신뢰와 RBAC 기반의 타원곡선암호

김 효 진† · 박 호 현††

요 약

배터리를 사용하는 센서 네트워크에서는 불필요한 오버헤드를 줄이는 것이 중요하다. 또한 보안의 필요성으로 인해 암호화 역시 중요하다. 하지만 암호화의 경우 어쩔 수 없는 오버헤드가 발생하는 데 보안과 오버헤드는 트레이드오프 관계에 있다. 본 논문에서는 암호화시 추가되는 오버헤드를 줄이기 위하여 신뢰값(Trust)라는 개념을 암호화에 사용하고 신뢰도가 높은 경로와 신뢰도가 낮은 경로 이용시 암호화에 사용되는 키 크기의 조절을 통해 보안 수준은 유지하면서 오버헤드는 줄이는 방법을 시도하였다. 시뮬레이션을 통해 일반적인 암호화와 신뢰값을 고려한 암호화를 비교하였고 그 결과 신뢰값을 고려하는 경우가 총 실행시간도 적고 오버헤드도 적었다. 실제 네트워크에서 구성 목적이나 환경 조건을 고려하여 보안 수준을 충족하는 신뢰값 기준을 정한다면 센서 네트워크에서 제한된 리소스를 효율적으로 사용할 수 있을 것이다.

키워드 : 신뢰, 타원 곡선, 역할기반접근제어, 센서네트워크, RC6

1. 서 론

현재 무선 센서 네트워크는 의료, 에너지 관리, 의료시설, 스마트 미터링(Smart Metering), 홈 네트워크, 빌딩 자동화, 산업 자동화, 농업 등 많은 분야에서의 사용을 목적으로 개발이 되고 있다.

이렇게 활용 분야가 다양해짐에 따라 수집되는 정보의 보안에 대한 중요성이 커지고 있다. 하지만 무선 센서 네트워크는 유선 네트워크에서 적용되는 보안 기법을 그대로 사용할 수 없는 데 그 이유는 유선과는 다른 무선의 개방성과 무선 센서 네트워크에서 사용하는 노드의 성능 제약 때문이라고 할 수 있다.

일반적으로 보안 수준(Security level)을 높이기 위해서는 암호화키의 크기를 늘리는 방법이 있으나 센서 네트워크에서는 휴대성을 필요로 하는 노드가 배터리를 사용함으로 인해 성능상 한계가 있고 대역폭에 제한이 있기 때문에 좀 더 효율적인 보안이 필요하다.

본 논문에서는 신뢰값(Trust)을 기반으로 하여 역할기반 접근제어(RBAC: Role-Based Access Control)와 보안 알고리즘 중 키교환 프로토콜인 타원 곡선 디피-헬만(ECDH: Elliptic Curve Diffie-Hellman) 키교환 프로토콜, 전자서명 기법인 타원 곡선 전자 서명 알고리즘(ECDSA: Elliptic

※ 이 논문은 2012년도 교육과학기술부의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2012-0007712).
† 정 회 원 : 중앙대학교 전자전기공학부 교수
†† 정 회 원 : LG전자 연구원
논문접수 : 2013년 3월 15일
수 정 일 : 1차 2013년 5월 2일
심사완료 : 2013년 5월 17일
* Corresponding Author : Ho-Hyun Park(hohyun@cau.ac.kr)

Curve Digital Signature Algorithm), 대칭키 암호화 알고리즘인 RC6(Rivest Cipher 6)를 통해 센서 네트워크에서의 보안과 관련된 전반적인 프레임워크를 제시하였다.

본 논문의 2절에서는 신뢰 모델, 역할기반접근제어 그리고 ECDH, ECDSA, RC6에 대해 살펴보고 3절에서는 2절에서 소개한 내용을 바탕으로 기존의 방식에 신뢰 개념을 추가시킨 암호화방식을 제시한다. 그리고 4절에서는 신뢰기반의 RBAC과 신뢰기반의 암호화를 통해 구성된 네트워크의 오버헤드를 시뮬레이션하고 5절에서 결론을 맺는다.

2. 관련 연구

보안과 오버헤드는 트레이드오프 관계라고 할 수 있는데 평문을 그대로 보내는 경우에 비해 암호화를 할 경우에는 암호화시 생기는 계산적 오버헤드뿐만 아니라 전자서명을 추가할 경우 전자서명 만큼의 추가 데이터 오버헤드가 생기게 된다. 그리고 이 오버헤드는 보안 수준을 높일수록 더 커지게 된다[1,2].

본 절에서는 프레임워크 구성에 사용되는 신뢰를 기반으로 한 RBAC 모델과 키교환 프로토콜인 ECDH, 전자서명 방식인 ECDSA 그리고 대칭키 알고리즘인 RC6에 대해서 알아본다.

2.1 신뢰 모델

신뢰 모델(Trust Model)은 노드의 신뢰도를 신뢰값이라는 수치로 표현하는 것을 말한다.

센서 네트워크에서 신뢰성은 보안 외에도 센서 노드와 라우터의 하드웨어와 소프트웨어의 안정성, 센서 네트워크가 설치된 곳의 주변 환경, 라우팅을 포함한 네트워크 프로토콜의 종류, 주변 통신환경, 센서 노드의 수명 등의 다양한 요소에 의해 영향을 받는다[3]. 그러나 본 논문에서는 이 신뢰값을 보안 수준 결정에 활용하여 오버헤드는 줄이면서 데이터 신뢰성을 유지하는 방법에 초점을 맞추고자 한다.

본 논문에서의 신뢰값은 노드의 배터리 상태, 현재 위치, 주변 상황 등 다양한 요소를 이용하여 계산한다. 만약 신뢰(Trust), 비신뢰(Distrust)로만 이루어진 신뢰 모델을 생각해 보면 어떤 노드에 대해 충분한 정보를 가지고 있을 경우, 그 노드의 신뢰여부를 판단하는 것은 어렵지 않다.

하지만 만약 노드에 대한 정보가 충분하지 않은 경우에는 신뢰여부를 판단하는 것이 큰 의미가 없게 된다. 따라서 신뢰와 비신뢰에 불확실성(Uncertainty)이라는 파라미터를 추가하여 신뢰 수준(Trust level)을 판단하는 데 있어 정확성을 높이도록 한다.

신뢰(Trust), 비신뢰(Distrust), 불확실성(Uncertainty)의 관계는 식 (1)과 같다.

$$T+D+U=1, \quad T, D, U \in [0,1] \tag{1}$$

세 파라미터의 합은 1이며, 각 파라미터의 범위는 0~1사이이다. 이 세 파라미터를 ω라는 여론(opinion)으로 표현한다. 이 여론은 삼각형으로 표현이 가능한데 예를 들어 $\omega = \{0.7, 0.1, 0.2\}$를 삼각형으로 표현하면 Fig. 1과 같다[4].

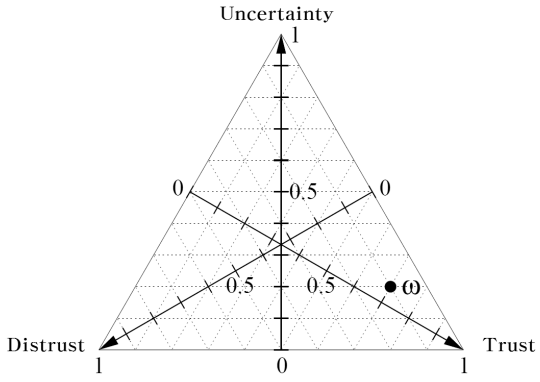


Fig. 1. Opinion Triangle

2.2 RBAC 모델

RBAC(Role based access control)은 사용자에 따라 역할(Role)을 부여하고 그 역할에 따라 권한(Permission)을 부여한다. 기존의 그룹과 역할의 가장 큰 차이점은 그룹은 사용자들의 집합이지만 권한의 집합은 아닌데 비해 역할은 사용자의 집합이면서도 다른 한편으로는 권한의 집합이라는 것이다[5,6].

RBAC은 핵심(Core) RBAC($RBAC_0$), 계층(Hierarchical) RBAC($RBAC_1$), 제한(Constraint) RBAC($RBAC_2$) 그리고 이 셋의 기능이 합쳐진 통합(Consolidated) RBAC($RBAC_3$)으로 구분할 수 있으며 이들의 관계는 Fig. 2와 같다[5].

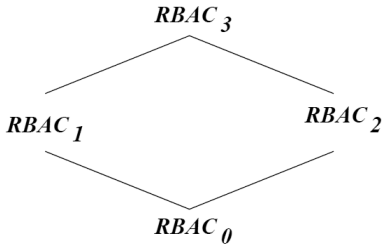


Fig. 2. The relation among RBAC models

$RBAC_1$은 핵심 RBAC에 계층구조를 더한 모델이고 $RBAC_2$는 권한과 함께 제한을 부여하는 모델이다. 그리고 이 두 모델을 통합한 모델이 $RBAC_3$이다.

1) 핵심 RBAC

RBAC의 구성요소는 아래와 같고 Fig. 3과 같이 표현할 수 있다[6]. Fig. 3에서 사용자(Users), 세션(Sessions), 역할(Roles)은 집합을 나타내고, PA(Permission assignment),

UA(User assignment)는 두 집합 간의 관계(Relation)를 나타낸다. 두 집합 간의 관계는 카티전 곱(Cartesian Product)의 진부분집합으로 표현된다. 화살표가 양쪽으로 표시된 내용은 다대다(many-to-many) 관계를 뜻한다.

- 사용자(Users)
  - 사용자는 한 명의 사용자에 대응
- 세션(Sessions)
  - 사용자와 여러 개의 역할(Role)로 구성
- 역할(Roles)
  - 역할이 수행 가능한 권한으로 구성
- 권한(Permissions)
  - 실행 가능한 연산의 집합
- 객체(Objects)
  - 시스템 자원 (예, 파일, 데이터베이스)
- PA(Permission assignment)
  - $PA \subseteq PERMS \times ROLES$, 역할과 권한의 다대다 관계
- UA(User assignment)
  - $UA \subseteq USERS \times ROLES$, 사용자와 역할의 다대다 관계

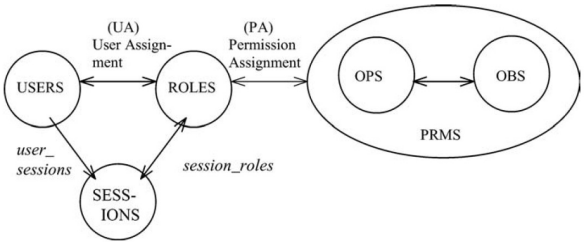


Fig. 3. Core RBAC

2) 계층 RBAC

기존의 핵심 RBAC에 역할의 상속관계를 나타내는 역할계층(RH: Role hierarchy)이라는 요소가 추가된 것으로 그 관계는 다음과 같다.

- RH(Role hierarchies)
  - $RH \subseteq ROLES \times ROLES$, 역할 간의 상속 관계

역할 계층은 역할간의 상속관계를 정의한다. 어떤 역할 r_2이 수행하는 내용이 r_1에서도 수행하는 내용인 경우 r_2는 r_1에 상속관계이다. 또는 $r_1 \geq r_2$라고 한다.

3) 제한 RBAC

핵심 RBAC에 역할간의 충돌을 방지하는 의무분할(Separation of Duty)관계를 추가한 것이다. 그리고 제한하는 위치에 따라 정적의무분할(Static Separation of Duty)과 동적의무분할(Dynamic Separation of Duty)로 구분된다.

2.3 ECC

타원 곡선 암호(Elliptic Curve Cryptosystem)는 유한체 F_p상에서 타원 곡선위의 점들 사이의 연산에서 정의되는 이산대수 문제(Discrete Logarithm Problem)의 어려움을 기반으로 하는 암호 방식이다[7]. 여기서 이산대수 문제란 유한체 F_p의 부분집합인 순환군 $F_p^* = g, g^2, g^3, \cdots, g^{p-1}$에서 $y \in F_p^*$일 때 $y = g^x$를 만족하는 x $(1 \leq x \leq |F_p^*|)$를 구하는 문제를 말한다. 유한체 F_p에서의 타원 곡선의 도메인 파라미터는 p, a, b, G, n, h가 있는데 p는 유한체 F_p내에서 정의된 소수이며 a와 b는 $y^2 \bmod p = x^3 + ax + b \bmod p$ 에서 정의되고 G는 암호화 연산을 위해 필요한 생성원(generator) 점(x_G, y_G)이다. 그리고 n은 타원 곡선의 차수(order)를 나타내고 h는 타원 곡선 위의 점 개수를 n으로 나눈 값을 의미한다[8].

1) 타원 곡선 위에서의 연산

우선 $p > 3$인 유한체 F_p상에서의 타원 곡선 E 는 식 (2)와 같이 정의한다.

$$y^2 = x^3 + ax + b \tag{2}$$

여기서 a, b는 F_p의 원소이고 $4a^3 + 27b^2 \not\equiv 0 (\mathrm{mod} p)$의 조건을 만족하여야 한다. 타원곡선 E 위의 모든 점은 유한체 F_p로 구성된다.

- 덧셈 연산(Addition)

타원 곡선 E 위의 서로 다른 두 점을 더한 것을 말하며 연산 결과 역시 타원 곡선 위의 점이 된다. 만약 타원 곡선 E 위의 두 점 $P = (x_1, y_1)$, $Q = (x_2, y_2)$을 더한 결과를 $R = (x_3, y_3)$이라 한다면 이를 그림으로 나타내면 Fig. 4와 같다.

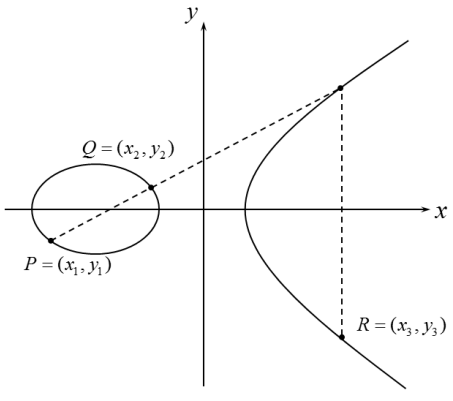


Fig. 4. $P + Q = R$ Addition

계산식은 식 (3)과 같이 나타낼 수 있다.

$$x_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)^2 - x_1 - x_2$$
$$y_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x_1 - x_3) - y_1 \quad (3)$$

• 두배 연산(Doubling)

$P=(x_1, y_1)$을 두 배하여 얻은 결과를 $R=(x_3, y_3)$이라 하고 이는 타원 곡선 위의 점 P에서 접선(Tangent)을 그어 만나는 점과 x축 대칭인 점을 말한다. 이를 그림으로 나타내면 Fig. 5와 같고 식으로 표현하면 식 (4)와 같다.

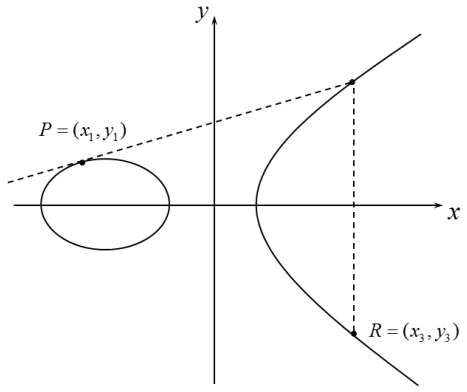


Fig. 5. $P+P=R$ Doubling

$$x_3 = \left(\frac{3x_1^2 + a}{2y_1}\right)^2 - 2x_1$$
$$y_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x_1 - x_3) - y_1 \quad (4)$$

타원 곡선에서 $R=xP$ 와 같은 연산의 경우 덧셈과 두 배 연산을 이용하여 계산할 수 있다.

2) ECDH

ECDH(Elliptic Curve Diffie Hellman)는 키교환 프로토콜로 두 노드가 대칭키 암호화 알고리즘을 사용하기 위해 필요한 공통된 비밀키를 생성하는 역할을 한다[7].

우선 디피-헬만(Diffie Hellman) 키교환 프로토콜은 이산대수 문제를 기반으로 한다. A와 B사이에서의 키교환 과정을 살펴보면 우선 큰 소수 p와 생성원 g는 공개되어 모두가 알고 있다고 가정한다. A는 $1 \sim p-1$사이의 수를 개인키 a 로 선택하고 공개키 $K_a = g^a \bmod p$ 를 계산하여 B에게 전송한다. B역시 A와 같이 $1 \sim p-1$사이의 수 b를 개인키로 선택하고 공개키 $K_b = g^b \bmod p$를 A에게 전송한다. 각각의 공개키를 수신한 A, B는 마지막으로 $K=(K_b)^a \bmod p$, $K=(K_a)^b \bmod p$를 계산하여 공통된 비밀키를 생성하게 된다[9].

하지만 이산대수 문제를 기반으로 한 디피-헬만 키교환 프로토콜은 안전수준을 높일수록 계산량이 지수 함수 수준으로 커지는 문제가 있다. 이에 비해 타원 곡선 암호의 경우 그 증가폭이 적어 안전수준을 높이면서 기존의 디피-헬만 프로토콜보다 계산량을 줄일 수 있다는 장점이 있다.

타원곡선 디피-헬만 키교환 프로토콜에서는 기존의 디피-헬만 프로토콜에서 사용하던 이산대수 문제인 $y=g^x$ 대신에 타원곡선 이산대수 문제(ECDLP)인 $Q=xG$를 사용한다. 키교환 과정을 그림으로 나타내면 Fig. 6과 같다.

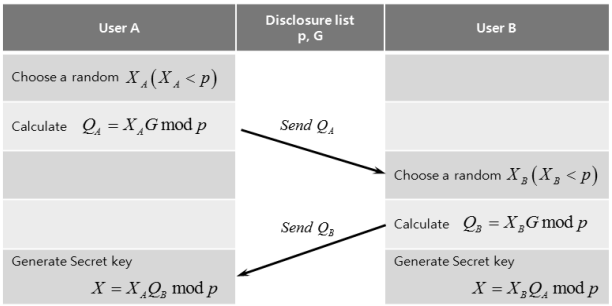


Fig. 6. The process of secret key generation using Elliptic curve Diffie-Hellman Key exchange protocol

타원 곡선 디피-헬만 키교환 프로토콜을 통해 생성한 비밀키를 이용하여 대칭키 암호화 알고리즘으로 메시지를 안전하게 암호화할 수 있다.

3) ECDSA

ECDSA(Elliptic Curve Digital Signature Algorithm)은 전자 서명으로 전송된 메시지의 허위여부, 변조여부를 파악할 수 있고 서명자가 누구인지 확인할 수 있는 기능을 제공한다[8].

타원 곡선 전자 서명에서는 통신을 하려는 두 노드가 타원 곡선의 도메인 파라미터와 서로의 공개키를 알고 있다고 가정한다.

• 서명 생성

A가 메시지 m에 서명을 하기 위해 비밀키 X_A를 사용한다. 서명 과정을 살펴보면 다음과 같다.

a) SHA-1과 같은 일방향 해쉬함수 h를 이용하여 $e=h(m)$ 계산

b) $1 \sim n-1$사이에서 임의의 수 k 선택

c) k를 이용하여 $kG=(x_1, y_1)$계산

d) $r = x_1 \bmod n$을 계산하고 만약 $r = 0$이라면 b) 단계부터 다시 시작

e) $k^{-1} \bmod n$ (k의 modulo n에 대한 역원) 계산

f) $s = k^{-1}(e + X_A r) \bmod n$을 계산하고 만약 $s = 0$이면 b) 단계부터 다시 시작

위 단계로 진행하면 최종적으로 메시지 m에 대한 서명 (r, s)가 생성된다.

• 서명 검증

A가 메시지 m에 서명한 (r, s)를 전송받은 B는 A의 공개키 Q_A와 도메인 파라미터를 이용하여 서명을 검증한다. 검증 과정을 살펴보면 다음과 같다.

a) 우선 전송받은 r, s가 $1 \sim n-1$ 사이의 정수인지 확인

b) SHA-1과 같은 일방향 해쉬함수 h를 이용하여 $e = h(m)$ 계산

c) $w = s^{-1} \bmod n$을 계산

d) $u_1 = ew \bmod n$과 $u_2 = rw \bmod n$을 계산

e) $Y = u_1 G + u_2 Q_A$를 계산하고 만약 $Y = O$ $(Infinity)$라면 서명을 거부(reject)함

f) Y의 x좌표를 이용하여 $v = x_1 \bmod n$을 계산

g) $v = r$이라면 서명 검증

• 서명 검증의 증명

A로부터 전송 받은 서명 (r, s)에서 $s = k^{-1}(e + X_A r) \bmod n$이고 이를 다시 정리하면 $k \equiv s^{-1}(e + X_A r) \equiv s^{-1}e + s^{-1}X_A r \equiv we + wrX_A \equiv u_1 + u_2 X_A \bmod n$이 된다. 그러므로 검증과정에서 $u_1 G + u_2 Q_A = u_1 G + u_2 X_A G = (u_1 + u_2 X_A) G = kG$을 통해 $v = r$을 유도할 수 있다.

4) ECC의 장점

• 복잡성

타원 곡선상에서 덧셈과 두배 연산을 통해 $Q = kP$의 계산을 하는 타원 곡선 암호시스템은 소인수분해를 이용하는 RSA나 이산 대수 문제를 사용하는 디피-헬만, ElGamal에 비해 계산이 더 어렵다[10].

• 키크기

보안 수준을 유지하기 위한 키크기가 다른 공개키 방식에 비해 작다는 장점이 있다. Table 1은 타원 곡선 암호 와 다른 방식을 비교한 내용이다.

• 전력소비

타원 곡선 암호시스템은 계산하는 데 드는 전력소모가 적어 저전력을 필요로 하는 모바일 장치에 적합하다.

Table 1. The comparison of key sizes among three methods

Symmetric key size	Discrete Logarithm Problem Key size for maintaining security level	RSA Key size for maintaining security level	ECC Key size for maintaining security level
56 bit	512 bit	512 bit	112 bit
80 bit	1024 bit	1024 bit	160 bit
112 bit	2048 bit	2048 bit	224 bit
128 bit	3072 bit	3072 bit	256 bit
192 bit	7680 bit	7680 bit	384 bit
256 bit	15360 bit	15360 bit	521 bit

2.4 RC6

RC6는 비밀키를 사용하여 메시지를 블록 단위로 암호화하는 알고리즘으로 간단한 산술 연산과 회전(rotation)으로 이루어진다. RC6는 RC5의 보안 이슈를 해결하기 위해 제안되었는데 RC5는 두 개의 레지스터를 사용한 반면 RC6는 네 개의 레지스터를 사용하며 곱셈과 같은 추가적인 연산을 사용한다. 이로 인해 적은 라운드로도 큰 안전성과 높은 확산(diffusion)을 얻을 수 있다[11-13].

RC6에는 $w/\ r/\ b$의 세가지 파라미터가 있는데 w는 블록 크기를, r은 라운드 수를 뜻하고 b는 암호화키의 크기를 바이트 단위로 나타낸다.

RC6는 크게 키 스케쥴 과정, 암호화 과정 그리고 복호화 과정, 이렇게 세 과정으로 구성된다.

• 키 스케쥴

사용자의 비밀키로부터 각 라운드에 사용될 워드를 생성하는 과정으로 r라운드의 경우 $2r+4$개의 w 비트의 워드를 생성하여 암호화와 복호화시 사용한다.

• 암호화

네 개의 w 비트 레지스터에 평문을 입력으로 받아 암호문을 생성하는 데 암호화를 위해 다음의 6가지 기본 연산을 사용하고 과정을 그림으로 표현하면 Fig. 7과 같다[12].

$a+b$ 덧셈 $(\bmod 2^w)$

$a-b$ 뺄셈 $(\bmod 2^w)$

$a \oplus b$ w 비트 워드의 XOR 연산

$a \times b$ 곱셈 $(\bmod 2^w)$

$a \lll b$ w 비트 워드 a를 왼쪽으로 b 비트만큼 순환 시프트

$a \ggg b$ w 비트 워드 a를 오른쪽으로 b 비트만큼 순환 시프트

• 복호화

복호화는 암호화의 반대방향으로 진행하여 암호문을 입력으로 받아 평문을 구한다.

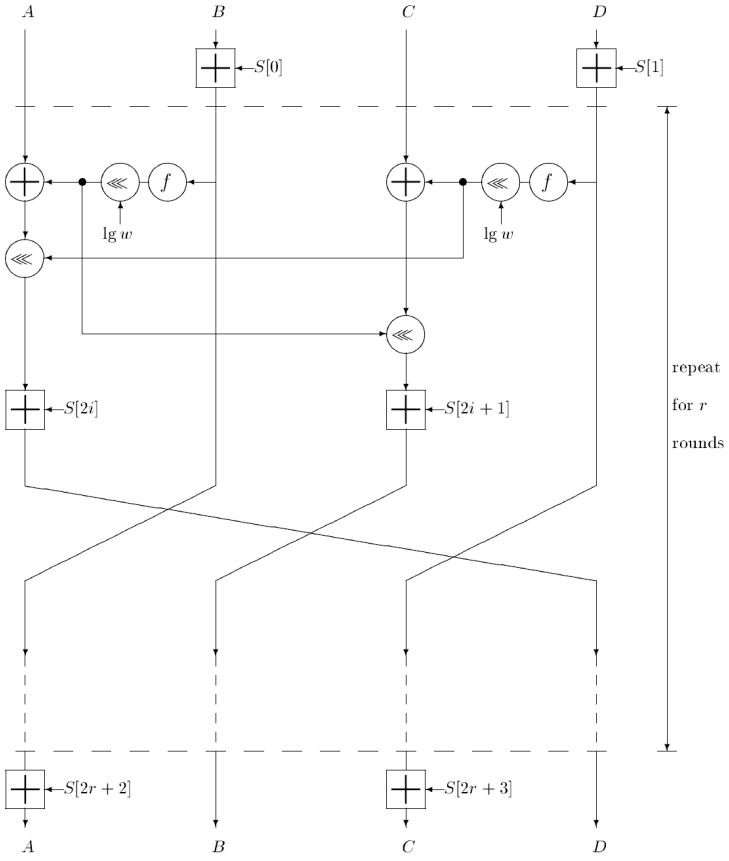


Fig. 7. RC6 encryption $(f(x) = x \times (2x+1))$

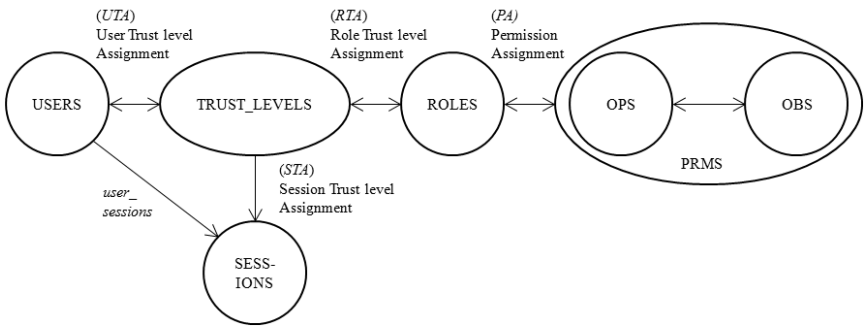


Fig. 8. TRBAC model

3. 제안하는 시스템

배터리를 이용하는 센서 네트워크에서 사용하는 기존 보안 알고리즘이 비교적 안전한 노드, 그렇지 않은 노드 구분 없이 동일한 암호화 수준을 사용하여 일부 경로에서 전체적인 보안 수준 이상의 암호화로 인해 필요이상의 오버헤드가 발생하는 단점이 있다.

본 절에서는 이를 개선하기 위하여 노드의 안정성을 구분하기 위한 신뢰값을 사용하고 보안 알고리즘과 신뢰를 결합시키기 위해 우선 역할기반접근제어(RBAC)에 신뢰라는 개념을 추가하여 신뢰값에 기반한 역할 수행을 하도록 하고 신뢰-역할기반접근제어(TRBAC: Trust-Role Based Access Control)으로 명명한다.

그리고 암호화 알고리즘에서도 신뢰-역할기반접근제어에 따라 역할이 부여된 노드로 이루어진 경로의 전체적인 신뢰도를 이용하여 암호화하는 전체적인 프레임워크를 설계하는 내용을 다룬다.

3.1 신뢰-역할기반 접근제어(TRBAC)

신뢰-역할기반 접근제어(Trust-Role Based Acess Control)는 기존의 역할기반 접근제어방법에서 역할을 부여할 때 노드가 가진 신뢰값(Trust value)를 기준으로 역할을 분배

하도록 하여 기존 방식보다 안전성을 높이는 방법을 말한다.

역할을 신뢰값에 따라 할당하고 노드는 자신의 신뢰값에 따라 그 값에 맞춰 부여된 역할을 수행하여 높은 신뢰를 얻는 노드가 더 중요한 역할을 수행하도록 하고 낮은 신뢰를 얻는 노드는 중요도가 떨어지는 역할을 수행하거나 아예 배제되는 방식으로 Fig. 8과 같이 나타낼 수 있다.

노드가 현재까지 수행한 라우팅 기록, 주변 환경, 신호 세기 등을 이용하여 노드의 신뢰값을 수치화하고 TRBAC에서 정의한대로 신뢰값에 따라 노드의 역할이 분배된다. 그리고 분배된 역할에 따라 라우팅에 참여할 수 있는지 여부 등이 결정된다.

3.2 신뢰기반 타원 곡선 암호 시스템(TECC)

신뢰기반 타원 곡선 암호 시스템(TECC: Trust Elliptic Curve Cryptosystem)은 암호화를 하되 신뢰도를 고려하여 신뢰도가 높은 경로를 이용할 경우와 신뢰도가 낮은 경로를 이용할 경우의 암호화 키크기를 다르게 하여 암호화로 인해 발생하는 오버헤드를 줄이면서 전체적인 보안 수준(Security level)은 유지하는 방법을 말한다.

1) 키크기

메시지 전송을 위한 키교환시 경로의 신뢰도(t)를 이용한다. 경로의 신뢰도는 해당 경로에 속한 노드의 신뢰값 중 가장 낮은 값을 말한다. 신뢰도가 높은 경우 낮은 경우에 비해 상대적으로 더 안전하기 때문에 암호화에 사용하는 키크기를 줄여 줄어든 키크기만큼의 오버헤드를 줄이게 된다. 보안 수준은 신뢰도에 따른 보안 수준 및 각 키의 크기는 Table 2 과 같다.

Table 2. Key sizes according to trust values

	t < 0.1	t < 0.3	t < 0.6	t < 1
Security level	-	256 bit	192 bit	128 bit
Secret key	-	256 bit	192 bit	128 bit
Public key	-	521 bit	384 bit	256 bit

신뢰도가 0.1이하인 경우에는 신뢰-역할기반접근제어에서 안전하지 않은 경로라 판단하여 사용하지 않는다.

2) 구성

본 논문에서 구성하는 타원 곡선 암호시스템은 대칭키 알고리즘인 RC6를 통해 암호화를 하고 타원 곡선 디피-헬만 키교환 프로토콜을 통해 RC6에 사용할 비밀키를 교환한다. 그리고 암호화된 메시지의 서명을 위해 타원 곡선 전자 서명 알고리즘을 사용한다.

그리고 신뢰를 기반으로 하기 때문에 여기에 추가적으로 경로 요청시 신뢰도를 받아 그것을 이용하여 암호화에 사용할 키크기를 결정한다. 신뢰기반 타원 곡선 암호시스템의 전체적인 구성을 순서도로 나타내면 Fig. 9와 같다.

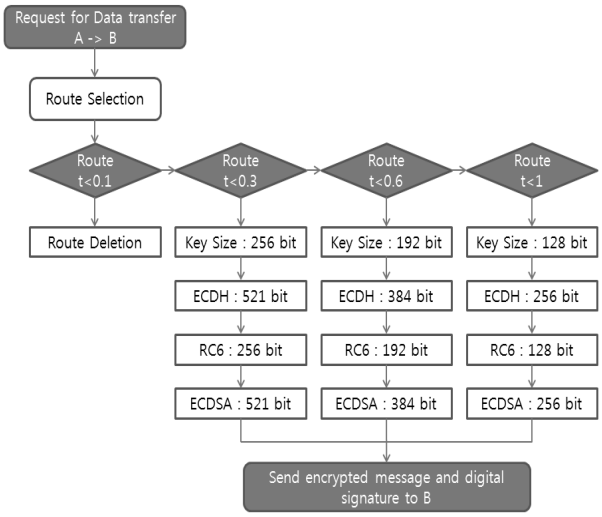


Fig. 9. The configuration of Trust based Elliptic Curve Cryptosystem

4. 시뮬레이션

4.1 시뮬레이션 환경

Visual Studio를 이용하여 C로 코드를 작성하고 시뮬레이션 하였다. 노드에서 노드로 메시지 전송시 걸리는 시간은 센서 네트워크의 특성 상 저속인 경우가 대부분이고, 센서 노드들이 대부분 안정성에 취약하여 노드 간 지연이 많이 발생할 수 있는 점을 고려하여 1024 bit당 0.5초로 가정하였다.

보안 수준을 256 bit로 하는 기존의 타원곡선 암호시스템을 사용하는 경우와 신뢰-역할기반접근제어와 신뢰기반 타원 곡선 암호시스템을 사용하는 경우로 나눠 시뮬레이션하였다. 보안 수준 256 bit는 현재의 암호시스템에서 안정적이고 널리 사용되는 보안 수준이다[8,12]. 따라서 TRBAC 모델에서 신뢰값이 가장 낮은 경우에 256 bit의 보안수준을 할당하였다.

TRBAC 모델을 적용할 수 있는 응용 분야로는 SOS와 같은 긴급구조시스템, 놀이동산의 미아 찾기 등이 있을 수 있다. 이런 응용에서는 네트워크 라우팅 시 유선 게이트웨이까지 일반적으로 2~4 개의 홉 카운트를 가정할 수 있는데 이에 적합한 센서 네트워크의 사이즈를 고려하여 노드 수는 10 개로 설정하였다.

그 외에 실험에 필요한 ECC 및 RC6의 파라미터는 [8,12,14]을 참고하여 결정하였다.

4.2 시뮬레이션 구현

신뢰-역할기반 접근제어에서의 역할부여를 위한 노드의 신뢰값(Trust)는 미리 노드에 0.1~1.0 사이의 수치로 할당하는 것으로 가정하였다. 그리고 성능 및 오버헤드 비교는 신뢰-역할기반 접근제어와 신뢰기반 타원 곡선 암호시스템을 사용한 경우와 신뢰값을 배제한 채 보안 수준을 256 bit으로 유지하기 위해 암호화 키크기를 256 bit로 사용하는 타원곡선 암호시스템을 비교하였다.

1) ECC 파라미터 설정

타원곡선 암호시스템의 보안 수준별 도메인 파라미터 p, a, b, G, n, h는 [14]의 표준화된 파라미터를 사용하는 데 128 bit 보안 수준의 경우 secp256r1을, 192 bit 보안 수준의 경우 secp384r1을, 그리고 256 bit 보안 수준의 경우 secp 521r1을 사용한다.

2) RC6 파라미터 설정

RC6의 파라미터 $w/r/b$의 경우 블록 크기를 나타내는 w는 32 bit를 사용하고 라운드 수를 뜻하는 r은 20, 그리고 암호화키의 크기를 바이트 단위로 나타내는 b는 암호화키 값에 따라 4, 6, 8을 사용한다.

RC6의 경우 4 개의 블록 단위로 암호화되므로 블록 암호 모드로 CBC 모드를 사용하고 평문의 길이를 블록 단위로 맞추기 위한 패딩 기법으로는 PKCS7을 사용한다.

4.3 시뮬레이션 결과

1) TRBAC을 통한 경로 결정

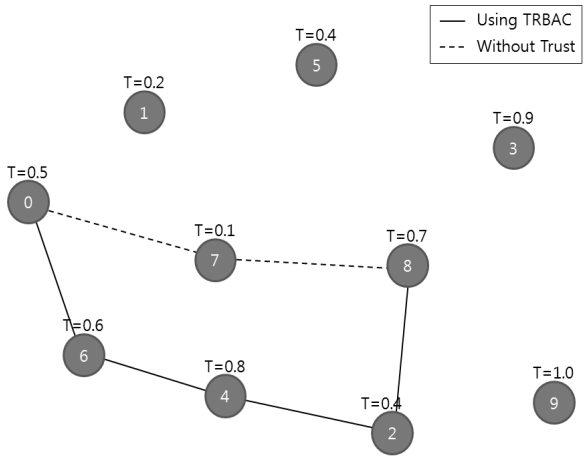


Fig. 10. The route determination through TRBAC

TRBAC에서는 노드의 신뢰값(Trust value)에 기반하여 역할을 부여하므로 신뢰가 0.1이하인 경우는 경로에서 배제한다. Fig. 10은 노드 0에서 노드 8까지의 경로를 신뢰를 고려하지 않는 방식과 신뢰개념을 고려한 TRBAC을 이용한 내용을 그림으로 나타낸 것이다. 신뢰를 고려하지 않는 방식의 경우 신뢰가 0.1인 노드 7을 통한 경로가 최적의 경로라고 판단한다. 하지만 노드 7의 경우 신뢰가 0.1로 낮기 때문에 안전한 노드가 아닐 가능성이 크기 때문에 이 경로는 적절하지 않다.

반면 신뢰를 적용한 TRBAC을 통한 라우팅의 경우 신뢰가 0.1인 노드는 라우팅에서 배제시키므로 주변의 다른 노드를 통한 라우팅 경로를 찾게 된다. 따라서 0-1-5-3-8과 0-6-4-2-8 두 경로 중에서 전체적인 신뢰가 높은 0-6-4-2-8의 경로를 선택하게 된다.

2) 소요시간

구현한 시스템에서 각 모듈별 소요시간은 Table 3과 같다.

Table 3. Execution time of each module

Trust	0.1~0.3	~0.6	~1.0
RC6 key size	256 bit	192 bit	128 bit
EC key size	521 bit	384 bit	256 bit
Doubling(Q=2G)	3 msec	2 msec	< 1 msec
Addition(Q=R+P)	2 msec	1 msec	< 1 msec
Keypair (Q=kG)	1.96 sec	0.75 sec	0.25 sec
ECDH	3.96 sec	1.51 sec	0.5 sec
RC6	1 msec	1 msec	1 msec
ECDSA Signature	2.07 sec	0.76 sec	0.24 sec
ECDSA Verification	3.97 sec	1.45 sec	0.49 sec

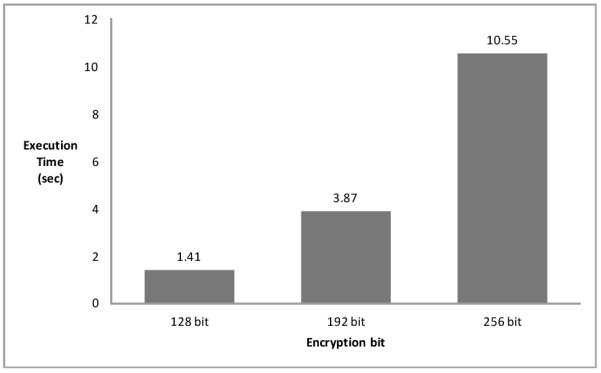


Fig. 11. Execution time message exchanges for each encryption bit size

타원 곡선 암호시스템(ECC)의 암호화 비트별 메시지 교환시 소요시간을 살펴보면 비트수를 늘릴수록 소요시간은 더 큰 폭으로 증가한다. 따라서 신뢰도를 이용하여 암호화키 크기를 줄인다면 네트워크 전체적으로 봤을 때 총 소요시간을 줄일 수 있다는 것을 알 수 있다.

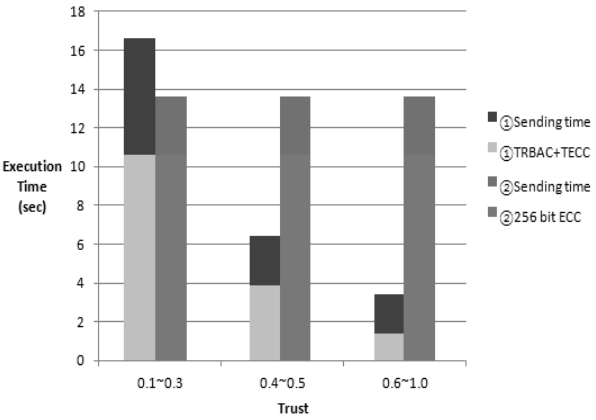


Fig. 12. Execution time along with trust values of nodes

Fig. 12는 4.3.1의 상황에서 노드 7의 신뢰값을 0.1부터 1.0까지 변화시키면서 그에 따른 소요시간을 측정한 결과이고 전송한 메시지는 1024 bit이다. 256 bit ECC의 경우 신뢰를 고려하지 않으므로 암호화 시간이 일정하고 경로 역시 변하지 않으므로 총 소요시간이 일정함을 알 수 있다.

반면 본 논문에서 제시한 TRBAC+TECC의 경우에는 노드의 신뢰가 0.1~0.3 인 경우 0-6-4-2-8의 경로를 택하므로 메시지 전송 시간이 추가되어 소요시간은 일반적인 256 bit 보다 더 걸렸지만 신뢰가 0.4이상이 되면서 0-7-8의 같은 경로를 사용하게 되고 신뢰를 이용해 암호화 키크기를 줄이고 그에 따라 메시지 이외에 추가적인 전송량이 줄어들어 소요시간은 절반이하로 줄어드는 것을 알 수 있었다.

3) 오버헤드

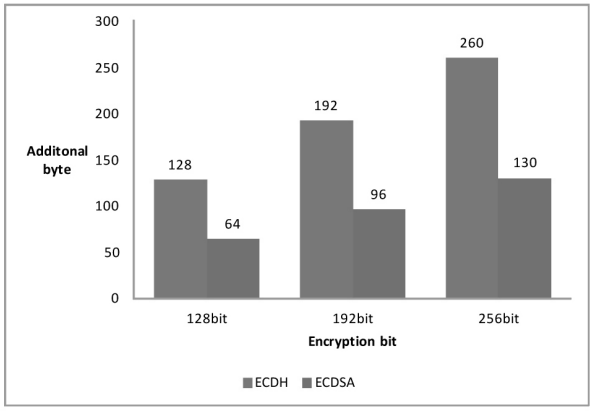


Fig. 13. The additional data for each encryption bit size

타원곡선 디피-헬만 키교환 프로토콜의 경우 통신을 원하는 두 노드만의 비밀키를 생성하기 위해 서로 생성한 키쌍을 전송하게 되어 그만큼의 추가 바이트가 생기게 된다.

타원곡선 전자서명 알고리즘 역시 암호문 전송시 서명한 만큼의 추가 바이트가 생긴다.

Fig. 14는 4.3.1의 상황에서 노드 7의 신뢰값을 0.1부터 1.0까지 변화시켰을 때 추가되는 데이터를 나타낸 것이다.

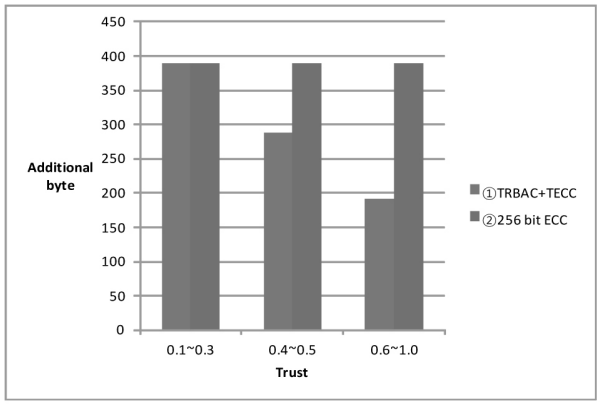


Fig. 14. The additional data along with trust values of nodes

비교해보면 256 bit ECC방식은 신뢰를 고려하지 않기 때문에 일정하지만 신뢰를 고려하는 TRBAC+TECC의 경우에는 신뢰값이 높은 경로를 이용하여 메시지를 전송할 때에는 키크기를 줄여서 사용하므로 추가되는 데이터양이 상대적으로 적다.

5. 결 론

본 논문에서는 암호화시 노드에 신뢰값을 부여하여 전체적인 보안 수준을 고려한 효율적인 키크기 조절을 통해 불필요한 오버헤드 감소를 목표로 하였다. 그러기 위해서 노드에 신뢰값을 부여하여 어느 정도 안전이 검증된 노드들로 이루어진 경로에서는 암호화시 키크기를 줄여서 사용하도록 하여 오버헤드를 줄이면서 전체적인 보안 수준은 유지되도록 설계하였다.

시뮬레이션으로 기본적인 암호화 모드와 신뢰값을 고려하여 키크기를 다르게 하는 TRBAC+TECC 방법으로 나누어 신뢰의 변화에 따른 데이터 전송 시간과 추가 데이터량을 비교하였고, 이를 통해 신뢰를 고려하여 암호화 키크기 변화시켜 암호화로 인해 추가되는 데이터량의 감소를 보였다.

네트워크의 구성 목적과 노드들의 특성에 따라 신뢰값에 영향을 미치는 파라미터를 조절하여 상황에 알맞은 신뢰값을 정의하고 그 뒤 전체적인 보안 수준을 유지하는 범위에서 키크기를 변화하여 암호화를 한다면 센서 네트워크의 리소스를 좀 더 효율적으로 사용할 수 있을 것이다.

참 고 문 헌

[1] F. Zhang, Y. Mu, W. Susilo, “Reducing Security Overhead for Mobile Networks”, AINA’05, IEEE, 2005.

[2] W. Liu, R. Luo, H, Yang, “Cryptography Overhead Evaluation and Analysis for Wireless Sensor Networks”, CMC’09, IEEE, 2009.

[3] S.C. Kim, W.J. Lee, P.J. Jung, “ A Study on the Reliability Type and the Reliability Improvement in Ubiquitous Sensor Networks”, Summer Conference, Vol.31, No.1, The institute of electronics engineers of Korea, 2008.

[4] A. Jøsang, “An Algebra for Assessing Trust in Certification Chains” Network and Distributed Systems Security, 1999.

[5] R.S. Sandhu, E.J. Coynek, H.L. Feinsteink, C.E. Youmank, “Role-Based Access Control Models”, IEEE Computer, Vol.29, No.2, pp.38-47, 1996.

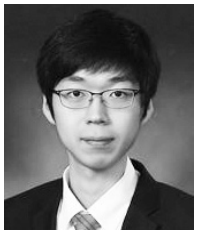
[6] ITI Secretariat, “Standard for Role-Based Access Control”, ANSI BSR INCITS 359, 2004.

[7] C.H. Lim, D.H. Lee, “Elliptic curve crypto algorithm”, TTA journal Vol.80, pp.98-104, 2002.

[8] D. Johnson, A. Menezes, S. Vanstone, “The Elliptic Curve Digital Signature Algorithm (ECDSA)”, International Journal of Information Security, Vol.1, Issue 1, pp.36-63, 2001.

[9] E. Rescorla, “Diffie-Hellman Key Agreement Method, RFC 2631”, IETF Network Working Group, 1999.
[10] M.Y. Malik, “Efficient Implementation of Elliptic Curve Cryptography Using Low-power Digital Signal Processor”, ICACT, 2010
[11] R.L. Rivest, “The RC5 Encryption Algorithm”, MIT Laboratory for Computer Science, 1997.
[12] R.L. Rivest, M.J.B. Robshaw, R. Sidney, and Y.L.Yin, “The RC6 TM Block Cipher”, MIT Laboratory for Computer Science, 1998.
[13] M. Ragab, A. Ismail. S. Farag Allah, “Enhancements and Implementation of RC6 Block Cipher for Data Security”, IEEE Region 10 International Conference on Electrical and Electronic Technology, 2001.
[14] Certicom research, “SEC 2: Recommended Elliptic Curve Domain Parameters”, Certicom Corp, 2000.

박 호 현
e-mail : hohyun@cau.ac.kr
1987년 서울대학교 계산통계학과(학사)
1995년 한국과학기술원 정보통신공학과 (석사)
2001년 한국과학기술원 전산학과(박사)
1987년~2002년 삼성전자 수석연구원
2003년~현 재 중앙대학교 전자전기공학부 교수
관심분야 : 빅데이터, 정보보안, 멀티미디어, USN



김 효 진
e-mail : kingdove@naver.com
2011년 중앙대학교 전자전기공학부(학사)
2013년 중앙대학교 전자전기공학과(석사)
2013년~현 재 LG전자 연구원
관심분야 : 정보보안, USN, Memory Management