무선센서네트워크에서의 NTRU에 기반한 키 교환 스킴

정회원 구 남 훈*, 준회원 조 국 화*, 정회원 고 병 환*, 권 순 학*°

An NTRU-based Key Agreement Scheme for Wireless Sensor Networks

Namhun Koo* *Regular Member*, Gook Hwa Jo* *Associate Member*,
Byeonghwan Go*, Soonhak Kwon*° *Regular Members*

요 약

연산 과부하로 인해 대부분의 연구에서는 무선센서네트워크에서 공개키 암호방식의 사용은 힘들다고 여겨져 왔다. 하지만 최근의 일부 연구에서는 키 사이즈, 데이터 사이즈, 연산 시간, 전력 소비 등이 다른 공개키 암호들에 비해 적은 공개키 암호방식을 무선센서네트워크에 적용할 수 있다는 결과가 있다. NTRU 암호방식은 무선센서네트워크에서 사용될 수 있는 공개키 암호방식 중의 하나로 꼽힌다. 하지만 실제로 무선센서네트워크에 적용할 수 있는 NTRU에 기반한 효율적인 키 교환 스킴은 많지 않다. 이 논문에서는 무선센서네트워크에서 사용할 수 있는 NTRU에 기반한 효율적인 키 교환 스킴을 제안한다.

Key Words : NTRU, NTRU Cryptosystem, NTRUSign, Key Agreement Scheme, Wireless Sensor Networks

ABSTRACT

Because of heavy computational overheads, the use of public key cryptosystem in Wireless Sensor Networks seems unfeasible. But some recent researches show that certain public key cryptosystem can be used in WSN, in which the key and data size, power consumption is relatively small. The NTRU cryptosystem is suggested as one of the candidates of public key cryptosystems which can be used in wireless sensor networks. In this paper, we propose an efficient key agreement scheme using NTRU and we show that it can be used in wireless sensor networks.

Ⅰ. 서 론

무선통신과 전자기술의 발전으로 인해 저소비, 저전력, 다기능 센서 노드의 개발이 가능해졌다. 이 극소형 센서 노드는 센싱, 데이터 수집, 통신 부품 등을 통하여 무선센서네트워크를 구현할 수 있게 하였으며 유비쿼터스 서비스의 핵심 기술로 각광받고 있다. 이러한 기술은 농장 모니터링, 스마트 오피스, out-of-tolerance 환경 조건의 감지, 의학적 보호, 스마트 유니폼 등의 광범위한 영역에 적용될 수 있다.

하지만, 무선센서네트워크는 센서 노드와 센서 노드 사이 혹은 센서 노드와 베이스스테이션 사이에 많은 통신을 요구하기 때문에, 이러한 통신들은 서비스 거부나 재생 공격 등의 공격을 받을 수 있다.

무선센서네트워크에서의 보안 서비스는 연산과 전력소비의 제한으로 인하여 많은 연산을 갖는 공개키 암호방식의 적용을 제한하였다. 따라서 대부분의 연구결과들은 센서네트워크에서의 대칭키 암호방식에 대해

※ 이 논문은 2007년도 정부재원(교육인적자원부 학술연구조성사업비)으로 한국학술진흥재단의 지원을 받아 연구되었음. (KRF-2007-313-C00006)
* 성균관대학교 수학과(komaton@skku.edu, achimheasal@nate.com, kobhh@naver.com, shkwon@skku.edu) (° : 교신저자)
논문번호 : KICS2009-09-383, 접수일자 : 2009년 9월 2일, 최종논문접수일자 : 2010년 4월 14일

초점을 맞추었다. 하지만 대칭키 암호방식에 기반한 이상적인 키 분배 스킴이 없기 때문에, 각 센서 노드의 키는 배치되기 전에 선분배되어야 한다. 또한 대칭키방식을 이용하면 센서 노드들의 키 교환이 쉽지 않다.

Gura[1] 등은 무선센서네트워크에서 공개키 암호방식을 적용하는 것이 가능하다는 것을 보였다. Gaubatz[2] 등은 무선센서네트워크에 적용될 수 있는 공개키 암호방식의 후보로 NTRU 암호방식을 제안했다. 그리고 NTRU는 서명 스킴을 제외하고는 센서네트워크에 적합하다.

NTRU 암호방식[3]은 다항식 환에서의 공개키 암호방식으로 그 안전성은 래티스에서의 Closest Vector Problem에 기반한다. NTRU 암호방식이 소개된 후, NTRUSign이라 불리는 NTRU 래티스를 이용한 전자서명 스킴이 제안되었다. Sliding Window 방법을 사용하여 NTRU 연산을 향상시키는 최근의 연구[4]가 있다.

키 교환 스킴은 Diffie와 Hellman[5]에 의해 처음으로 제안되었으며, 그 안전성은 이산대수 문제의 어려움에 기반한다. Joux[6]는 겹선형 페어링을 이용한 3자간 1라운드 키 교환 스킴을 제안하였다. 그 후에, 페어링 방법에 기반한 많은 키 교환 스킴이 제안되었다. NTRU에 기반한 키 교환 스킴은 Jiang과 He에 의해 제안되었다[7]. 하지만 이 스킴은 무선통신 사용자와 네트워크 서버 사이의 키 교환을 제공하기 때문에 이 스킴을 그대로 무선센서네트워크에 적용하기는 쉽지 않다. 따라서, 본 논문에서는 기존 스킴에 비하여 무선센서네트워크의 센서 노드에 적용하기에 더 적합한 NTRU에 기반한 새로운 키 교환 스킴을 제안한다. 제안된 스킴은 기존의 스킴들보다 더 효율적이고 안전하므로, 자원제약적인 센서 노드에 더 적합한 키 교환 스킴이다.

본 논문의 나머지 부분은 다음과 같다 : 2장에서는, NTRUEncrypt 암호 방식과 NTRUSign 서명 스킴이 설명된다. 제안된 키교환 스킴이 3장에서 소개 되며, 키 교환 스킴의 보안요건들에 소개와 제안된 스킴의 안전성에 대한 증명이 4장에서 진행된다. 5장에서는 제안된 스킴을 NTRU 암호방식에 기반한 다른 키 교환 스킴들과 비교한다. 마지막으로 6장에서 결론짓는다.

Ⅱ. NTRU 암호 시스템

2.1 NTRU 공개키 암호

NTRUEncrypt는 Hoffstein[3]에 의해 처음으로 제안된 공개키 암호이다. 비밀키와 공개키는 환 $R=\mathbb{Z}[x]/(x^N-1)$안에 속한다. *는 R에서의 곱셈 연산을 뜻하며, star multiplication이라고 부른다. $f=\sum_{i=0}^{N-1} f_i x^i$, $g=\sum_{i=0}^{N-1} g_i x^i \in R$에 대해서 덧셈은 $f+g=\sum_{i=0}^{N-1}(f_i+g_i)x^i$로 정의되며, 곱셈은 다음과 같이 정의된다.

$$m = f*g = \sum_{i=0}^{N-1} m_i x^i$$
$$\text{여기서 } m_k = \sum_{k \equiv i+j(\mathrm{mod}N)} f_i g_i \tag{1}$$

사실, 위의 덧셈과 곱셈의 정의는 환 R에서의 일반적인 덧셈과 곱셈이다.

1) 파라미터 선택 : NTRU의 주요 공개 파라미터는 N, p, q이다. N은 R의 다항식의 차수와 관련이 있다. p와 q는 암호화와 복호화 중에 모듈 연산과 관련이 있다. p와 q는 소수여야 하지는 않지만, 서로소여야 한다. 무선센서네트워크에 추천되는 파라미터는 (N,p,q)=(167,3,128)이다.

2) 키 생성 : 먼저 작은 계수를 갖는 N-1차의 두 개의 다항식 $f,g \in R$을 선택한다. f는 mod p와 mod q상에서 역원이 존재해야 한다. 이 역원을 f_p, f_q라고 쓰기로 한다. f는 어떤 다항식 F에 대해서 $f=1+pF$의 형태로 선택하여 $f_p=1$이 되도록 할 수 있다. f, g, f_p, f_q는 비밀로 저장된다. 공개키 h는 다음과 같이 계산된다.

$$h = f_q * g \,(\mathrm{mod} q) \tag{2}$$

3) 암호화 : 메시지 $m \in R$에 대해서, 임의의 다항식 $r \in R$을 선택한다. 메시지 m에 대한 암호문 e는 다음과 같이 계산된다.

$$e \equiv pr * h + m \,(\mathrm{mod} q) \tag{3}$$

4) 복호화 : 암호문 e에 대해서 먼저 다음을 계산한다.

$$a \equiv f * e \,(\mathrm{mod} q) \tag{4}$$

복호화 실패를 방지하기 위해서 a의 계수들은 $[-q/2, q/2]$안에 있어야 한다. 그리고, a를 이용하여 암호문 e에 해당되는 평문 m'를 다음과 같이 계산한다.

$$m' \equiv a * f_p \,(\mathrm{mod} p) \tag{5}$$

5) 복호화 성립 설명 : (4)에서 계산된 다항식 a는 다음과 같이 나타낼 수 있다.

$$\begin{aligned} a &\equiv f^*e \equiv f^*(pr^*h+m)(\mathrm{mod}q) \\ &\equiv f^*pr^*h+f^*m(\mathrm{mod}q) \\ &\equiv f^*pr^*(f_q{}^*g)+f^*m(\mathrm{mod}q) \\ &\equiv f^*f_q{}^*pr^*g+f^*m(\mathrm{mod}q) \\ &\equiv pr^*g+f^*m(\mathrm{mod}q) \end{aligned} \quad (6)$$

*a*를 mod *p*로 줄이면, 다항식 *f*m* (mod *p*)를 얻고, f_p를 곱하면 $f_p{}^*f^*m \equiv m$ (mod *p*)를 얻는다.

2.2 NTRUSign

NTRUSign은 NTRU 래티스에 기반하는 전자 서명이다. 2001년에 NTRU Signature Scheme(NSS)가 제안되었지만[8], NSS의 서명을 위조하여 사본 공격이 가능함이 증명되었다[9]. 이후 NSS의 결점을 극복하기 위하여 NTRU를 이용한 새로운 서명 스킴 NTRUSign[10]이 제안되었다. NTRUEncrypt는 NSS와 같은 키 생성 알고리즘을 갖고 있지만, NTRUSign은 서명 키로 래티스의 기저 원소를 사용하기 때문에 복잡한 키 생성 알고리즘을 가지고 있다. NTRUSign의 안전성은 APPR-CVP(The Approximate Closest Vector Problem)의 어려움에 기반한다.

2.3 NTRUEncrypt와 NTRUSign의 안전성

NTRU 암호방식의 안전성은 주어진 래티스에서의 길이가 짧은 벡터 찾는 문제(CVP: closest vector problem)에 기반한다. CVP 문제가 NP-complete 문제 중의 하나임은 1981년 Brickell등에 의해 알려졌다.

정의 1. [래티스]

$\{b_1, b_2, \ldots, b_n\}$을 $\mathbb{R}^n$에서 일차독립인 벡터들의 집합이라고 하자. 벡터들의 집합

$$L = \{\mathrm{z} | \mathrm{z} = \sum_{\mathrm{i}=0}^{\mathrm{n}} \mathrm{c_i b_i}, \mathrm{c_1}, \mathrm{c_2}, \ldots, \mathrm{c_n} \in \mathbb{Z}\}$$

을 *n*차원 래티스라고 한다. 집합 $\{b_1, b_2, \ldots, b_n\}$을 래티스 *L*의 기저라고 한다.

문제 1. [The Closest Vector Problem(CVP)]

주어진 벡터 $r \in \mathbb{R}^n$에 대하여 다음을 만족하는 래티스 벡터 $v \in L$을 찾아라.

$$\|r-v\| = \min_{z \in L} \|r-z\|$$

문제 2. [The Approximite Closest Vector Problem(APPR-CVP)]

주어진 벡터 $r \in \mathbb{R}^n$에 대하여 충분히 작은 $N \in \mathbb{R}$에 대하서 다음을 만족하는 래티스 벡터 $v \in L$을 찾아라.

$$\|r-v\| \leq N$$

정의 2. [NTRU 래티스]

$h = \sum_{i=0}^{N-1} h_i x^i$를 NTRUEncrypt의 공개키라고 하자. 그러면 집합

$$\begin{aligned} &(0,0,0,\ldots,0,1,\mathrm{h_{N-1}},\mathrm{h_{N-2}},\mathrm{h_{N-3}},\ldots,\mathrm{h_1},\mathrm{h_0}), \\ &(0,0,0,\ldots,1,0,h_{N-2},h_{N-3},h_{N-4},\ldots,h_0,h_{N-1}), \\ &\quad\ldots \\ &(0,1,0,\ldots,0,0,h_0,h_{N-1},h_{N-2},\ldots,h_2,h_1), \\ &(0,0,0,\ldots,0,0,q,0,0,\ldots,0,0), \\ &(0,0,0,\ldots,0,0,0,q,0,\ldots,0,0), \\ &\quad\ldots \\ &(0,0,0,\ldots,0,0,0,0,q,\ldots,0,0), \end{aligned}$$

는 차원이 2*N*인 래티스를 생성한다. 이 래티스를 NTRU 래티스라고 한다.

위의 래티스를 다음과 같이 쓸 수 있으며,

$$(\mathrm{u},\mathrm{v}) \in \mathrm{R} \times \mathrm{R} \mid \mathrm{u}^*\mathrm{h} \equiv \mathrm{v}(\mathrm{mod}q)$$

(*f*,*g*)는 위 래티스에 포함된다. 따라서 NTRU 래티스에서 *r*=0일 경우의 CVP를 풀어낸다면, NTRUEncrypt의 비밀키인 *f*, *g*를 찾아낼 수 있다. 하지만 높은 차원에서의 CVP는 여전히 풀기 어렵다. CVP와 APPR-CVP를 완전히 풀어낼 수 있는 다항식 시간 알고리즘은 아직까지 발견되지 않았다. 그러나 차수가 작은 경우엔 LLL 또는 L^3 알고리즘[11]을 이용하여 주어진 래티스의 기저벡터로부터 길이가 짧은 벡터를 찾을 수 있다. 또한 L^2 알고리즘[12]은 부동소수점 연산에 기반하며 엔트리가 많을 경우에는 L^3 알고리즘보다 더 효율적이다. 하지만, 주어진 래티스의 차원이 충분히 크면, 위의 알고리즘들은 가장 짧은 벡터를 찾아내지 못하며, 따라서 NTRU 암호 방식은 여전히 안전하다고 여겨지고 있다.

Ⅲ. 제안된 스킴

이 절에서는 무선센서네트워크에서의 두 센서 노드 사이에서의 키 교환 스킴을 제안한다. 무선센서네트워

크에서는 베이스스테이션을 신뢰된 키 분배 기관으로 선정할 수 있으므로 제안된 스킴에서는 베이스스테이션을 신뢰된 기관으로 가정한다. 스킴에서 사용할 기호들은 다음과 같다.

- $R=\mathbb{Z}[x]/(x^N-1)$: NTRU 환
- BS : 베이스스테이션
- f_A, g_A : A의 비밀키(long-term key)
- h_A : A의 공개키(long-term key)
- f_A^{-1} : f_A의 R mod q에서의 역원
- $E_{PK\_A}(M)$: 메시지 M을 A의 공개키를 이용하여 대칭키 암호방식을 이용하여 암호화한 암호문
- $MAC_A(M)$: 메시지 M에 대한 메시지 인증 코드(Message Authentication Code; 대칭키 암호 방식을 이용하여 생성)
- k_A : 메시지 인증 코드를 생성하는데 사용되는 대칭키 암호 방식에서의 비밀 키
- S(M) : NTRUSign을 이용하여 생성한 메시지 M에 대한 서명
- ID(A) : q 비트인 A의 ID

제안된 스킴은 크게 Setup, Request, KeyAgreement 등의 3개의 과정으로 나눌 수 있다. Setup 과정에서는 베이스스테이션은 서명 스킴인 NTRUSign에서 사용될 자신의 공개키와 개인키를 설정하고, 각 센서 노드는 NTRU에서 사용할 자신의 공개키와 개인키를 생성한다. Request 과정에서는 키 교환을 원하는 노드들이 베이스스테이션에게 자신이 특정 노드와 키 교환을 하기 원한다는 사실을 알려 베이스스테이션의 중재를 요청한다. KeyAgreement 과정에서는 각 노드는 교환하고자 하는 세션키를 생성하게 된다. KeyAgreement 과정 후, 각 노드는 자신의 세션키를 제거한다. 각 덧셈 연산과 곱셈 연산은 NTRU 환에서 정의된 덧셈과 곱셈(star multiplication)을 사용하였다.

Setup

- 베이스스테이션은 NTRUSign에서 사용할 개인키와 공개키, 그리고 메시지 인증 코드에서 사용될 개인키를 생성한다. 각 노드들은 자신의 공개키와 개인키(long-term)를 이미 분배받았다고 가정한다. 각 키들은 베이스스테이션과 해당 노드만 공유하기 때문에 다른 노드는 이를 알 수 없다.
- 노드 A는 개인키로 두 다항식 f_A, g_A를 선택하고 이를 이용하여 공개키인 h_A를 계산한다. 노드 B 역시 마찬가지로 f_B, g_B를 선택한 다음 공개키인 h_B를 계산한다. f_A와 f_B는 반드시 mod q 상에서 역원을 가져야 한다.
- 각 노드는 자신의 공개키를 베이스스테이션에 전송하고 베이스스테이션은 이를 저장한다.

Request

- 노드 A가 노드 B와의 키 교환을 원한다면, 자신의 공개키와 상대의 ID를(h_A, ID(B)) 베이스스테이션에 전송한다.
- 베이스스테이션은 상대방 노드의 공개키를 각 노드에게 전송해준다.

KeyAgreement

- 노드 A는 임의로 다항식 r_A를, 노드 B는 임의로 다항식 r_B를 선택한다.
- 노드 A는 $K_A=E_{PK_B}(r_A*f_A^{-1})$와 $M_A=MAC_A(r_A*f_A^{-1})$를 계산한다. 마찬가지로 노드 B는 K_B, M_B를 계산한다.
- 노드 A와 B는 각각 m_A와 m_B를 베이스스테이션에 전송한다. 여기서 $m_A=(K_A, M_A, ID(A))$, $m_B=(K_B, M_B, ID(B))$이다.
- 베이스스테이션은 메시지 인증코드인 M_A, M_B를 복호화하여 $r_A*f_A^{-1}$, $r_B*f_B^{-1}$값을 알아낸다. 그 뒤, 이를 각 노드의 공개키로 암호화하여 K_A, K_B와 각각 같은지를 확인한다. 만약에 같다면 각 메시지에 대한 인증서를 발급하는데, 노드 A에 대한 인증서로 Cert(A)=(ID(A), K_A, S(ID(A), K_A))를 발급하여 노드 B에게 전송한다. 마찬가지로 노드 B에 대한 인증서를 발급하여 노드 A에게 전송한다.
- 노드 A는 베이스스테이션의 서명을 확인한 후, K_B를 복호화하여 $r_B*f_B^{-1}$를 알아낸다. 마찬가지로 노드 B는 $r_A*f_A^{-1}$를 알아낸다.
- 노드 A와 B는 공유키로서 $K=r_A*f_A^{-1}+r_B*f_B^{-1}$ 를 사용한다.
- 세션 종료 후 각 노드는 보안성을 위해 세션키 생성에 사용된 정보들을 모두 제거한다.

Ⅳ. 안전성 분석

이 섹션에서는 제안된 스킴의 안전성에 대하여 분석한다. 먼저 공개키 암호방식에서의 키교환 스킴의

안전성의 기본적인 요구조건을 간단하게 소개한다[13, 14]. 스킴들이 무선센서네트워크같은 열린 네트워크에서 사용되기 때문에, 모든 응용에 대해 기본적인 안전 목표들이 중요하다. 다른 요구조건들도 중요하지만 여기에서 언급된 것들보다는 덜 중요하다.

4.1 기본적인 안전 요구조건들

A와 B를 스킴을 정확하게 이행하는 두 실체라고 가정하자.

1) Implicit key authentication : 특정 상대 B를 제외한 다른 실체가 특정 키의 값을 알아낼 수 없을 경우에 그 키교환 스킴은 B에서 A로의 implicit key authentication을 제공한다고 한다.

2) Key confirmation : 상대방 B가 특정한 키를 가졌다고 A가 확신할 수 있을 때, 그 키교환 스킴은 B에서 A로의 key confirmation을 제공한다고 한다.

3) Explicit key authentication : 어떤 키교환 스킴이 implicit key authentication과 key confirmation을 제공할 때 그 키교환 스킴이 explicit key authentication을 제공한다고 한다.

4.2 다른 보안 요구조건들

키교환 스킴에서 요구되는 몇가지 요구조건들 또한 규명되어야 한다. 비슷한 방법으로, A와 B는 정직한 존재라고 하자.

1) Key-compromise : A의 long-term 키가 누출됐다고 가정하자. 그러면 공격자는 A인 것처럼 행동할 수 있다. 하지만 A 이외의 다른 실체의 정보들에게는 영향을 미치지 않는다.

2) (Perfect) forward secrecy : 키교환을 하는 실체 중에 전체가 아닌 일부의 long-term 키가 유출되었더라도 세션키에 대한 안전성에 영향을 미치지 않으면 그 스킴은 (partial) forward secrecy를 만족한다고 한다. 만약 키교환을 하는 실체 모두의 long-term 키가 유출되었더라도 세션키에 대한 안전성에 영향을 미치지 않을 때 그 스킴은 perfect forward secrecy를 만족한다고 한다.

3) Unknown key-share secure : 실체 A와 B의 키교환이 A가 모르는 상황에서 일어날 수는 없다는 것을 말한다.

4) Known key secure : 공격자가 이전에 교환되었던 세션키들을 알게 되더라도 현재 교환되고 있는 세션키들에 영향이 없다는 것을 말한다.

5) No key control : A와 B 모두 세션이 끝나기 전에 세션키를 결정할 수 없다는 것을 말한다.

다음으로 앞에서 소개한 보안조건들을 주어진 스킴이 만족하는지를 증명한다. 이를 증명하기 위해서 먼저 공격자의 수위를 소개한 뒤에 안전성을 분석한다.

- ▲ 공격자는 베이스스테이션과 각 노드 사이에 전송되고 있는 모든 정보를 볼 수 있다.
- ▲ 공격자는 정당한 개인키를 생성할 수 있지만 아마도 각 노드의 것들과 같지는 않을 것이다.
- ▲ 공격자는 이전에 공유된 세션키들을 알 수 있다.
- ▲ 공격자는 베이스스테이션과 각 노드 사이에 전송되고 있는 정보를 위조하려는 시도를 할 수 있다.

4.3 제안된 스킴의 보안성

노드 A와 노드 B의 대칭성 때문에 노드 B의 경우만 다루어도 충분하다. 각 노드는 Random Oracle을 가지고 있어 비밀키나 MAC 키, 그리고 세션키가 임의로 생성된다고 가정하자.

1) Implicit key authentication : 만약 누군가가 노드 B로부터 직접 알아내지 않고 K를 알아내고자 한다면 먼저 노드 B의 개인키를 알아야 한다. 다른 노드들은 역시 정당한 비밀키를 가지고 있지만, B의 세션키와 관련되어있지 않기 때문에 $r_B*f_B^{-1}$ 값을 알아낼 수 없다. 공격자들은 역시 이를 암호화한 정보인 K_A, K_B, M_A, M_B를 알아낼 수 있지만, 이를 복호화하기 위해서는 역시 복호화키가 필요한데 이를 알고 있지 않으므로 역시 $r_B*f_B^{-1}$를 알아내는 것은 불가능하다. 공격자가 세션키나 비밀키, 혹은 MAC 키를 위조하려는 시도를 할 수 있지만, 이 키들이 임의로 선택되었기 때문에 위조할 수 있을 확률은 무시할 만하다. 따라서 공격자는 노드 A와 B의 비밀키를 모르고서는 세션키를 알아낼 수 없다.

2) Key confirmation : KeyAgreement의 첫 번째 라운드가 끝나면, 베이스스테이션은 각 노드로부터 온 메시지가 정말 그 노드로부터 왔는지를 메시지 인증코드를 통해서 확인한다. 그 후 베이스스테이션은 각 노드의 메시지에 대한 인증서를 발급한 후 상대방 노드에게 전달하게 된다. 베이스스테이션이 신뢰된 키분배 기관이라 가정했으므로, 각 노드는 그 메시지가 정확히 상대방 노드로부터 베이스스테이션을 통하여 온 것임을 확신할 수 있다. 그리고 베이스스테이션에서 온 서명이 맞는 것으로 규명된다면, 서명 스킴의 안전성에 의해 각 노드는 보내고자 하는 메시지가 상대방에게 제대로 갔는지를 확인할 수 있다.

3) Explicit key authentication : 위의 두 가지 조건을 만족하므로 이 조건은 성립한다.

4) Key-compromise secure : 공격자가 다른 노드 C의 비밀키 f_C를 알아냈다고 하자. 하지만 각 노드의 비밀키는 임의로 생성되었기 때문에 비밀키들 사이에는 연관성이 없다. 따라서 f_C를 이용하더라도 각 노드의 비밀키를 찾는 문제의 어려움은 줄어들지 않는다.

5) (Partial) Forward secrecy : 노드 A의 비밀키 f_A, MAC 키 k_A를 공격자가 알아냈다고 하자. 공격자는 이미 K_B와 M_A를 복호화하여 $r_B*f_B^{-1}$를 알아낼 수 있다. 하지만 공격자는 K_A와 M_B를 복호화할 수는 없는데, 그 이유는 비밀키들 사이에 관계가 없기 때문이다. 따라서 공격자는 알고 있는 정보를 이용해서 f_B를 찾아내어야만 한다. 하지만, h_B나 $r_B*f_B^{-1}$를 이용해서 f_B를 찾는 것은 CVP(Closest Vector Problem)의 어려움에 의해서 어렵다. 또한 공격자는 이미 f_A를 알고 있으므로 f_A^{-1}을 쉽게 계산해낼 수 있지만, r_A가 임의로 선택되었기 때문에 그것을 이용하여 $r_A*f_A^{-1}$을 계산하는 것은 어렵다. 따라서 공격자는 세션키 K를 계산할 수 없다. 하지만 이 스킴은 perfect forward secrecy하지는 않은데, 그 이유는 공격자가 모든 노드의 세션키를 알아낸다면, $r_A*f_A^{-1}$과 $r_B*f_B^{-1}$ 모두를 알 수 있으므로 $K=r_A*f_A^{-1}+r_B*f_B^{-1}$를 계산해낼 수 있기 때문이다.

6) Unknown key-share : 베이스스테이션이 중간에서 신뢰된 키 분배기관으로 존재하기 때문에 키 교환을 상대방 노드가 모를 수는 없다.

7) Known key secure : 이전 세션키들 K', K'' 등을 알아내더라도, 과정 중에 임의로 선택되는 다항식들이나, 각 노드들의 비밀키, MAC 키 사이에 아무런 관계가 없기 때문에 현재의 세션키인 K를 찾아내는 것은 여전히 어렵다.

8) No key control : 각 노드는 모든 라운드가 끝나야 키를 생성하는데 필요한 정보를 알 수 있다. 예를 들어 노드 A는 $r_B*f_B^{-1}$를 알아내려면, 베이스스테이션의 서명을 받아야 한다. 그 전에 이 정보를 알아내기 위해서는 다른 공격자들과 같은 노력을 해야 하고, 이것은 각 파라미터들의 임의성 때문에 어렵다.

만약 공격자가 세션에 참가하지 않는 노드 C의 비밀키나 MAC 키 등의 비밀정보를 알아낸다면, m_C를 위조할 수 있어 다른 노드를 사칭하면서 네트워크를 공격할 수 있다. 이러한 공격을 방지하기 위해서는 각 노드의 MAC 키를 정기적으로 바꾸어 공격자가 메시지를 위조할 수 없도록 하면 된다.

Ⅴ. NTRU에 기반한 다른 스킴들과의 비교

여기서는 주어진 스킴을 NTRU에 기반한 키 교환 스킴과 비교한다. 현존하는 NTRU에 기반한 키 교환 스킴은 두 가지가 있는데, 한 가지는 Jiang과 He에 의해 제안되었고[7], 다른 하나는 박현미 등에 의해 제안되었다[15]. 이 두 가지 스킴을 편의상 저자의 이름을 따서 JH 스킴, PKCK 스킴이라고 각각 부르기로 한다.

무선센서네트워크는 자원 제약적인 네트워크이기 때문에 노드의 연산의 수가 적을수록 좋다. 즉, 효율적일수록 무선센서네트워크에도 더 적합하다. 여기서는 먼저 스킴들의 연산의 개수를 비교하여 효율성을 비교한 후 안전성에 대해서 비교한다.

5.1 효율성

여기서는 세 스킴의 효율성을 비교한다. 라운드 수와 연산의 개수를 비교한다.

▲라운드의 수

주어진 스킴의 KeyAgreement 과정은 2라운드로 되어있다. 하지만 나머지 두 스킴은 3라운드로 되어있다. 만약에 Request 과정까지 합한다면 주어진 스킴은 4라운드가 되지만, 다른 스킴들 역시 키 교환을 서버에 요청하는 과정을 생략하고 있기 때문에, 주어진 스킴 역시 비슷하게 생각했을 경우 3라운드로 되어있다고 말할 수 있다. 즉, 다른 스킴들과 비교해봤을 때 라운드 수에서는 비슷하다고 할 수 있다.

▲다항식 연산의 수

편의상 R_A를 환 R에서의 덧셈의 개수, R_M을 환 R에서의 곱셈의 개수를 표현한다고 하자.

각 스킴은 각각 다른 서명 알고리즘을 사용한다. 주어진 스킴은 NTRUSign을, JH 스킴은 NSS 알고리즘을, PKCK 스킴은 NTRUSign의 preprint 버전을 사용하여 서명을 한다. 각각의 서명 방법과 NTRUEncrypt에서 사용하는 환 R에서의 연산의 수는 표 1과 같다.

제안된 스킴에서의 노드의 역할을 JH 스킴과 PKCK 스킴에서는 사용자(user)가 하기 때문에, 두 스킴의 효율성을 비교하기 위해 제안된 스킴에서의 노드의 연산의 수와 나머지 두 스킴에의 사용자의 연산의 수를 비교한다. 제안된 스킴에서 각 노드는 NTRUEncrypt 암호화와 복호화를 각각 1번씩, 그리고 환 R에서의 덧셈과 곱셈 1번씩, 1번의 대칭키 암호화와 1번의 NTRUSign 서명확인 과정을 연산한다. JH 스킴에서의 사용자는 NTRUEncrypt 암호화와 복호화를 각각 1번씩, 그리고 1번의 대칭키 암호화와, 1번의 NSS 서명, 2번의 NSS 서명확인 연산을 한다. PKCK 스킴의 사용자는 1번의 NTRUEncrypt 암호화와 4번의 환 R에서의 곱셈, 1번의 NTRUSign

표 1. 각 서명 알고리즘과 NTRUEncrypt의 각 과정에서의 연산의 수

서명	R_M	R_A	Hash	Norm
NTRUSign 서명확인	1	1	1	1
NSS 서명	1	1	0	0
NSS 서명확인	3	0	0	0
NTRUSign preprint 서명	1	2	0	0
NTRUEncrypt 암호화	1	1	0	0
NTRUEncrypt 복호화	2	0	0	0

preprint 서명, 1번의 해쉬 연산을 한다. 표 1과 위에서 나열한 연산의 수를 종합하여 표로 나타내면 표 2와 같다.

연산들 중에서 가장 비용이 많이 드는 연산은 환 R에서의 곱셈이다. 그런데 주어진 스킴은 4번의 곱셈을, JH 스킴은 10번의 곱셈을, PKCK 스킴은 7번의 곱셈을 실행하기 때문에, 주어진 스킴이 세 스킴 중에서 가장 효율적으로 키 교환을 한다는 것을 알 수 있다.

표 2. 효율성 비교

스킴	주어진 스킴	JH 스킴	PKCK 스킴
라운드 수	3	3	3
R_M	4	10	7
R_A	2	2	2
Norm 연산	1	0	0
해쉬	1	0	0
대칭키 암호화	1	1	1

5.2 안전성

여기서는 세 스킴들이 안전성 요구조건들을 만족하는 지를 비교한다.

세 스킴에서 사용되는 서명 알고리즘은 각각 다르다. 제안된 스킴은 안전성을 고려하여 NTRUSign을 사용하였다. 반면에, 이전의 스킴인 JH 스킴은 NSS를, PKCK 스킴은 NTRUSign의 preprint 버전을 사용하였다. NTRUSign은 APPR-CVP의 어려움에 의해 안전성이 보장되고 있지만, NTRUSign의 이전에 나온 스킴인 NSS 등은 이미 안전하지 않음이 증명되어 있다.[9,16] 따라서 이 두 스킴에서는 그 서명이 안전하다고 할 수 없으므로, 이를 토대로 각 스킴의 안전성을 따져보면 아래의 표 3와 같다.

JH 스킴과 PKCK 스킴의 서명 알고리즘을 NTRUSign으로 바꾸면 안전성을 강화할 수 있겠지만, NTRUSign, NSS, NTRUSign의 preprint에서의 서명 방식은 구조가 아주 다르기 때문에 적용하기 어렵다. 만약에 두 스킴을 수정하여 적용하더라도 두 스킴에서는 사용자가 서명을 해야 하며, NTRUSign의 서명 과정이 매우 복잡하기 때문에 효율성에 상당한 문제가 될 것이다. 제안된 스킴은 NTRUSign의 서명확인 과정만을 이용하여 효율적이고 안전하다.

표 3. 안전성 비교

스킴	주어진 스킴	JH 스킴	PKCK 스킴
Explicit key authentication	Yes	No	Yes
Key confirmation	Yes	No	No
Key compromise	Yes	Yes	Yes
(Partial) forward secrecy	Yes	No	No
Perfect forward secrecy	No	No	No
Unknown key share	Yes	Yes	Yes
Known key share	Yes	Yes	No
No key control	Yes	No	Yes

Ⅵ. 결 론

NTRU 암호방식은 무선센서네트워크에 적용될 수 있는 가장 효과적이고 안전한 공개키 암호 중 하나이다. NTRUEncrypt를 암호화 알고리즘으로 NTRUSign을 서명 스킴으로 사용할 수 있다.

하지만 이미 제안된 키 교환 스킴들은 무선센서네트워크의 센서 노드에 적용하기에는 적합하지 않다. 이 논문에서는 무선센서네트워크에 적합한 NTRU에 기반한 키 교환 스킴을 제안하였다. 그리고 이 스킴은 이전의 스킴보다 더 안전하고 효율적이므로, 제한적인 환경을 가진 무선센서네트워크에 적용하기에 더 적합하다. 제안된 2자간 키 교환 스킴을 추후에 다자간 키 교환 스킴으로 확장할 수 있다면, 여러 센서 노드 간에 키 교환을 하는 데 응용할 수 있으리라 생각된다.

참 고 문 헌

〔1〕 N. Gura, A. Patel, A. Wander, H. Eberle,

S.C. Shantz, "Comparing Elliptic Curve Cryptography and RSA on 8-bit CPUs", *Proceeding of CHES 2004*, LNCS 3156, pp. 119-132, 2004

〔2〕 G. Gaubatz, J. Kaps, B. Sunar, "Public Key Cryptography in Sensor Networks-Revisited", *Proceeding of ESAS 2004,* LNCS 3313, pp.2-18, 2004.

〔3〕 J. Hoffstein, J. Pipher, J. H. Silverman, "NTRU: A new high speed public key cryptosystem", *Proceeding of ANTS III, LNCS 1423*, pp.267-288, 1998.

〔4〕 M. Lee, J. Kim, J. Song, K. Park "Sliding Window Method for NTRU", *Proceeding of ACNS 2007*, LNCS 4521, 432-442, 2007.

〔5〕 W. Diffie, M. Hellman "New Directions in Cryptography", *IEEE Transactions on Information Theory*, 22(6), pp.644-654, 1976

〔6〕 A. Joux, "A One round protocol for Tripartite Diffie-Hellman", *Journal of Cryptology (2004)* 17, 263-276, 2004.

〔7〕 J. Jiang, C. He, "A novel mutual authentication and key agreement protocol based on NTRU cryptography for wireless communications", *Journal of Zhejiang University SCIENCE*, 2005 6A(5), pp.399-404, 2005.

〔8〕 J. Hoffstein, J. Pipher, and J. H. Silverman, "NSS: An NTRU Lattice-Based Signature Scheme", *Proceeding of Eurocrypt 2001*, LNCS 2045, pp.211-228, 2001.

〔9〕 C. Gentry, J. Jonsson, J. Stern, M. Szydlo, "Cryptanalysis of the NTRU Signature Scheme (NSS) from Eurocrypt 2001", *Proceeding of Asiacrypt 2001*, LNCS 2248, pp.1-20, 2001

〔10〕 J. Hoffstein, N. Howgrave-Graham, J. Pipher, J. H. Silverman, W. Whyte "NTRUSign: Digital Signatures Using the NTRU Lattice", *Proceeding of CT-RSA 2003*, LNCS 2612, pp.122-140, 2003.

〔11〕 A. K. Lenstra, H. W. Lenstra Jr., L. Lovasz, "Factoring polynomials with rational coefficients", *Mathmatische Annalen*, 261: 513-534, 1982.

〔12〕 P. Q. Nguyen, J. Stern, "Floating-point LLL revisited", *Proceeding of Eurocrypt 2005*, LNCS 3494, pp.215-233, 2005.

〔13〕 H. Lee, K. Ha, K. Ku "ID-based Multi-party Authenticated Key Agreement Protocols from Multilinear Forms", *Proceeding of ISC 2005*, LNCS 3650, pp.104-117, 2005

〔14〕 H. Lee, H. Lee, Y. Lee, "Multi-party Authenticated Key Agreement Protocols from Multilinear Forms", available at http://citeseer.ist.psu.edu/old/lee02multiparty.html

〔15〕 박현미, 강상승, 최영근, 김순자, "NTRU 기반의 이동 통신에서의 인증 및 키 합의 프로토콜", 한국정보보호학회지, 12(3) pp.49-59, 2002.

〔16〕 C. Gentry, M. Szydlo, "Cryptanalysis of the Revised NTRU Signature Scheme", *Proceeding of Eurocrypt 2002*, LNCS 2332, pp. 299-320, 2002.

〔17〕 B. Driessen, A. Poschmann, C. Paar, "Comparison of Innovative Signature Algorithms for WSNs", *Proceeding of ACM Conference on Wireless Network Security*, Alexandria, USA, pp.30-35, 2008.

구 남 훈 (Namhun Koo) 정회원

2007년 8월 성균관대학교 수학과 학사

2009년 2월 성균관대학교 수학과 석사

2009년 3월~현재 성균관대학교 수학과 박사과정

<관심분야> 공개키 암호 시스템, 타원곡선 암호시스템, Pairing 기반 암호시스템, NTRU 암호시스템, USN 보안

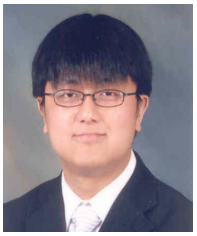
조 국 화 (Gook Hwa Jo) 준회원

2007년 2월 전북대학교 수학과 학사

2009년 3월~현재 성균관대학교 수학과 석사과정

<관심분야>
정수론, 암호학, 타원곡선, 공개키 암호시스템

고 병 환 (Byeonghwan Go) 정회원



2005년 2월 연세대학교 원주캠퍼스 수학과 학사
2007년 2월 성균관대학교 수학과 석사
2007년 3월~현재 성균관대학교 수학과 박사과정
<관심분야> 공개키 암호시스템, 암호시스템 구현, 타원곡선 암호시스템, Pairing 기반 암호시스템, USN 보안

권 순 학 (Soonhak Kwon) 정회원

1990년 2월 KAIST 수학과 학사
1992년 2월 서울대학교 수학과 석사
1997년 5월 Johns Hopkins University 박사
1998년 3월~현재 성균관대학교 수학과, 교수
<관심분야> 정수론, 암호론, Cryptographic Hardware, USN 보안