情報保護學會論文誌
第19卷 第6號, 2009. 12

차량 애드혹 네트워크 환경에서 효율적인 메시지 인증 기법*

유 영 준,† 이 준 호, 이 동 훈‡
고려대학교 정보경영공학전문대학원

Efficient Message Authentication Scheme for VANET*

Young Jun Yoo,† Jun Ho Lee, Dong Hoon Lee‡
Graduate School of Information Management and Security, Korea University

요 약

차량 애드혹 네트워크 환경에서 차량들은 교통 정보를 비롯한 다양한 서비스를 네트워크 인프라를 통해 제공 받을 수 있으며 운전자의 생명과 직결되는 차량의 운행 정보들을 빈번한 통신으로 상호 교환한다. 따라서 운전자의 편의와 안전을 위해서 송수신 되는 정보들을 효율적이고 안전하게 전송하는 프로토콜의 연구는 계속되어 왔다. 그 중 TSVC는 TESLA를 기반으로 설계되어 전송과 연산의 효율성을 보장 하지만 수신된 메시지의 검증이 일정시간이 지난 후에 이루어져 시간적인 지연을 가진다. 그러므로 시간에 민감한 메시지들의 전송에 TSVC를 적용하는 것은 적절하지 않다. 본 논문에서는 안전한 통신과 차량의 익명성을 보장하며 메시지 검증에 지연을 최소화하는 효율적인 메시지 인증 기법을 제안한다. 제안하는 기법은 시간에 민감한 메시지들의 전송에 적합하며, 서비스거부 공격에도 강건하다.

ABSTRACT

In VANET, each vehicle can obtain traffic information from other vehicles or infrastructure, and they frequently exchange life-critical safety message. Therefore, it is necessary among vehicles to establish a secure channel for keeping the driver's safe and protecting the channel against several attack challenges. TSVC is a representative scheme which needs low communication and computation to be performed. But, there is a delay when verifying the messages because it is designed based on TESLA. Thus, it is not acceptable to use TSVC for sending the time-critical messages. In this paper, we propose a novel message authentication scheme which reduces a delay for the verification of messages. Therefore, the proposed scheme can be suitable to transmitting time-critical messages. Furthermore, the scheme supports to privacy preservation and can robust against DoS attacks.

Keywords: VANET, Vehicle Ad-hoc Network, Message Authentication, V2V

I. 서 론

차량 애드혹 네트워크는 지능형 차량 기술의 진보와 더불어 발전하고 있는 통신 기술로써 이동형 기기들의 네트워크 인프라 기술인 MANET(Mobile Ad hoc NETwork)의 한 분야이다[1]. 최근의 무선 네트워크 기술의 발전은 차량이 이동을 하면서도 교통 정보, 지리 정보뿐만 아니라 도로의 상태 정보, 사고 발생 정보 등을 차량 간에 공유하는 것을 가능하게 하였다. 이런 환경에서 차량 애드혹 네트워크 환경을 통해 전송되는 다양한 정보들은 운전자들에게 편리함을 제공할 뿐만 아니라 운전자의 생명 역시 보장하는 수

접수일(2009년 9월 25일), 게재확정일(2009년 10월 30일)
* 본 연구는 지식경제부 및 정보통신연구진흥원의 IT산업원천기술개발사업의 일환으로 수행하였음.
(2009-F-056-01, Car-헬스케어 보안 기술개발)
† 주저자, iamakira@hanmail.net
‡ 교신저자, donghlee@korea.ac.kr

단이 될 것으로 기대된다[2]. 따라서 차량 간에 송수신되는 정보들이 악의적인 공격자에 의해 잘못된 정보로 변경되거나, 필요한 정보가 은닉되어 다른 차량에 전달되지 않는다면, 이런 정보들에 의존하는 여러 차량들은 큰 사고에 닥칠 수 있다. 이러한 사고의 위험을 미연에 방지하기 위해서는 수신자가 각 메시지에 포함되어 있는 정보를 신뢰할 수 있어야 하고, 송신자는 전송하는 정보의 신뢰성을 제공할 수 있는 방법이 요구된다[3].

각 메시지의 안전한 통신을 위해서는 메시지의 무결성 뿐만 아니라 효율적인 전송 및 계산이 요구되며, 특히 긴급 메시지의 경우에는 멀티-홉 브로드캐스트 통신을 통하여 전송되기 때문에 주기적인 메시지에 비하여 전송 및 계산단계에서 효율성에 더욱 큰 영향을 받게 된다.

차량 애드혹 네트워크에서의 통신은 해당 모듈의 통신 대상에 따라 크게 두 가지로 구분되며 각 차량에 탑재된 OBU들을 통한 차량간의 통신을 V2V (Vehicle-to-Vehicle) 통신이라 하며, OBU와 도로에 위치한 RSU를 통한 인프라와의 통신을 V2I(Vehicle-to-Infrastructure) 통신이라고 한다[18]. 두 통신기법은 인프라의 통신 참여 유무에 따라 통신 방법에서도 차이점이 존재하며, 각 통신별로 요구되는 보안 요구사항에서도 분명한 차이가 존재한다. 예를 들어 V2V환경에서는 V2I환경과 달리 신뢰할 수 있는 기관이 없기 때문에 전송 차량 스스로 자신을 인증해야 되는 기법이나 확률적인 인증 기법들이 요구된다[5,7,9].

최근 송·수신되는 메시지를 통해 해당 차량의 위치정보가 노출되는 것을 방지하기 위한 프라이버시 보장 기법에 대한 연구가 활발히 진행 중에 있다. 차량의 고유한 아이디를 이용한 메시지 인증과정은 프라이버시가 보장되지 않기 때문에 익명성(Anonymity)을 제공하면서 거짓 메시지에 대해서는 근원 차량의 아이디를 찾을 수 있는 추적성(Traceability) 역시 만족하는 인증 과정이 필요하다. 하지만 익명성과 추적성은 상반된 보안요구사항이므로 이를 동시에 만족하는 기법을 설계하는 것은 쉬운 일이 아니다. 또한 위에 언급된 사항들을 만족하면서 전송과 계산 효율성이 높은 시스템을 설계하는 것도 어려운 문제이다. 그러므로 최근의 연구들은 보안요구사항, 프라이버시 보장과 함께 효율적인 전송 및 계산 오버헤드를 제공하는 관점에 초점을 두고 진행되고 있다.

메시지 인증과정에서 익명성을 어떻게 보장하느냐의 관점에서 각 연구들은 크게 두가지 분류로 나눠진다. Chaum 등은 차량 애드혹 네트워크 환경에 하나의 그룹 공개키로 검증이 가능하고 그룹 구성원 각각은 다른 키로 서명을 하는 그룹 서명 기법을 적용한 컨셉을 제시하였다[6]. Boneh 등은 차량 애드혹 네트워크에 적합한 그룹 서명기법을 제안하였고[7], Lin 등에 의하여 처음 차량 애드혹 네트워크 환경에 적용된 기법이 제안되었다[8]. 그룹 서명 방식은 다량의 익명 아이디를 저장할 필요가 없고, 특정 기간마다의 아이디 업데이트도 필요가 없다. 하지만 그룹 서명은 타원곡선 환경의 연산으로 구성되어져 있기 때문에 기본적으로 메시지 검증 단계에서 많은 계산 오버헤드가 발생한다. 반면 Raya 등은 익명성을 보장하기 위하여 다량의 익명 아이디를 포함하는 셋(Pseudo ID set)을 사용하는 기법을 처음으로 제안하였다[9]. 각 차량은 랜덤하게 생성된 익명 아이디 셋을 초기단계에서 발급을 받은 후에 일정 간격으로 익명 아이디를 바꿔가면서 익명성을 유지하게 된다. 하지만 이와 같은 방법은 익명 아이디 셋의 저장을 위한 별도의 저장 오버헤드가 발생하게 되고, 익명 아이디 셋에 대한 폐지 목록의 업데이트가 주기적으로 이뤄져야 한다는 단점을 가진다. 또한 사용하는 ID가 일정간격으로 바뀌기 때문에 특정시간마다 익명 아이디를 재 발급받아야 된다. Zhang등이 제안한 TSVC는 이와 같은 방법을 통하여 프라이버시를 보장하며 메시지를 인증하는 대표적인 기법이다[10]. TESLA를 기반으로 제안된 TSVC는 계산 및 전송 효율성이 높은 반면 메시지 검증에 일정 시간의 지연현상이 존재하기 때문에 TESLA에서 언급되는 기본적인 취약점 외에도 긴급 메시지 전송 시에는 적용이 불가능하다는 치명적인 단점이 존재한다[11].

본 논문에서 제안하는 기법은 다음과 같은 공헌을 한다. 첫째, 메시지 검증단계에서의 지연현상을 최소화 하여 주기적 메시지뿐만 아니라 긴급 메시지에도 적용 가능하도록 하였으며, 둘째, RSU와의 통신이 없어 메시지 전송 효율성을 보장한다. 셋째, 메시지 인증을 위한 서명 단계를 최소화하여 수신자의 메시지 검증 단계에서 계산 효율성을 향상하면서 모든 메시지에 대한 부인방지(Non- repudiation)를 보장한다. 마지막으로 익명 아이디에 기반을 둔 기존의 기법들과 비교해서 공개키와 개인키 쌍의 사용 횟수를 줄임으로서 송신 차량의 재 인증 요구 횟수를 최소화 할 수 있다.

II. 관련 연구 및 보안 요구사항

2.1 관련 연구

2.1.1 TSVC 프로토콜

TSVC 프로토콜은 효율성을 위하여 센서 네트워크에서 주로 사용되는 TESLA 프로토콜을 기반으로 둔 차량 애드혹 네트워크에서의 메시지 인증 기법이다. TESLA는 효율적이고 메시지 손실에 안전한 대표적인 브로드캐스트 인증 프로토콜이다[11]. 메시지 송신자는 서명 값 대신 MAC(Message Authentication Code)값을 전송하기 때문에 전송 효율성이 뛰어나며, 수신자는 서명검증 대신 MAC연산을 통하여 메시지를 검증할 수 있기 때문에 검증단계에서의 계산 효율성 또한 뛰어나다. 이러한 장점 때문에 낮은 연산 계산 능력과 전력을 통하여 통신이 이뤄지는 센서 기반의 네트워크 환경에서 주로 사용된다.

즉, TESLA기반의 TSVC 역시 전송과 계산과정에서 효율성을 지니고 있지만 정확한 키 노출 타이밍을 요구하기 때문에 송신자와 수신자는 시간 동기화가 이루어져 있어야 한다는 가정이 필요하다. 또한 송신자는 일정 시간 이후 자신이 사용한 키를 공개하기 때문에 부인방지를 만족하지 못하고, 메시지를 받음과 동시에 검증과정을 수행하지 못하여 키 노출 시간 동안 해당 메시지를 임시로 저장하고 있어야 하기 때문에 DoS(Denial of Service) 공격에도 취약하다. 결정적으로 TSVC는 수신자의 메시지 인증을 위하여 키 노출 시간만큼의 지연 현상이 존재하기 때문에 긴급메시지에는 적합하지 않다.

2.2 시스템 위협요소

본 논문에서는 V2V에서의 안전한 메시지 인증 기법을 제안한다. V2V 통신환경에서 공격자는 도청(Eavesdropping), 의미 없는 메시지의 전송(Bogus Message Attack), 메시지 변조(Message Modification), 재생공격(Replay Attack), DoS 공격 등을 시도하여 다른 차량의 메시지인양 위조하고자 한다. 또한 메시지를 통해 차량의 특정 정보를 얻어 차량의 이동 경로 등을 파악하려 한다.

2.3 보안 및 프라이버시 요구사항

본 논문에서 제안하는 메시지 인증 프로토콜은 2.2절에서 제시한 위협요소에 안전한 메시지 통신을 위하여 다음과 같은 요구사항들을 기본적으로 필요로 한다.

2.3.1 객체 인증 및 메시지 무결성 (Entity Authentication and Message Integrity)

위조 공격으로부터 안전하기 위해서 수신자는 해당 메시지가 정당한 사용자로부터 전송된 것인지 확인할 수 있어야 한다. 또한 메시지 변조를 파악하기 위해서 수신자는 전송받은 메시지에 대한 무결성을 확인할 수 있어야 한다.

2.3.2 프라이버시 보장(Privacy Preservation)

프라이버시를 보장하기 위해서 각 차량은 익명성(Anonymity)과 비연결성(Unlinkability)을 만족해야 한다. 익명성이란 공격자가 차량의 통신을 통해서 해당 차량의 고유한 정보를 감지할 수 없어야 한다는 성질이다. 이 때 해당 차량의 고유한 정보는 알 수 없지만 여러 통신을 통하여 임의 차량의 경로를 추적할 수 있다면 이는 비연결성이 만족되지 않는다고 한다. 따라서 각 차량은 서론에서 언급한 방법을 통해 익명성을 유지하는 기법을 사용하여 공격자에게 차량의 경로를 추적당하지 않도록 하여야 한다. 또한 특정한 정보를 통하여 차량의 이동경로를 파악할 수 없도록 비연결성을 만족해야 한다.

2.3.3 추적성(Traceability)

차량에게 각 차량 고유의 값을 생성해주는 TA(Trust Authority)와 같은 키 생성 센터는 사고가 발생하거나 사용자가 악의적인 행동을 했을 때의 책임을 부여하기 위하여 추적성을 갖추어야 한다. 이를 위하여 키 생성 센터는 발급하는 모든 값에 대하여 일정시간 저장하고 사고발생시 증거로서 제출될 수 있도록 유지하여야 한다.

2.3.4 효율성(Efficiency)

차량의 이동 속도가 빠르고, 특히 시내의 경우에는 일정범위 내의 차량 밀집도가 높기 때문에 각 차량들은 일정시간동안 많은 수의 메시지를 검증하게 된다.

때문에 메시지 손실의 최소화를 위해서는 효율적인 전송 오버헤드와 검증단계에서의 계산 효율성이 요구된다.

III. 제안하는 기법

3.1 시스템 모델

제안하는 기법은 메시지를 보내려는 송신 차량과 해당 메시지를 받는 여러 대의 수신 차량들로 구성된 환경에서 가정된다. 메시지를 보내고자 하는 차량은 자신의 통신 범위내의 차량이 참여할 때마다(혹은 나갈 때마다) 유동적으로 변하는 그룹을 형성하고 그 그룹의 리더로서 역할을 한다. 주위 차량의 범위 안에 중복되어 속하게 될 경우 차량은 여러 그룹에 중복된 그룹원으로서 메시지를 전송받게 된다. 각 차량은 브로드캐스트 통신을 통해 메시지를 그룹 내 다른 차량들에게 전송하게 된다. 그룹 내부에 새로운 멤버가 참가하게 되는 경우를 위해 각 송신 차량은 초기 인증 메시지를 주기적으로 전송한다.

최초 통신을 위해 생성되는 서명은 신뢰기관인 키 생성 센터(Key Generation Center)로부터 발행되어 각 차량이 저장하고 있는 익명 아이디와 그에 해당하는 공개키, 개인키 그리고 공개키가 포함된 인증서에 의하여 유효하게 된다.

메시지 송신 차량은 최초 메시지와 함께 매 세션마다 생성하는 세션 공개키를 포함하여 서명 값을 생성한 후 이 값을 각 차량에 전송한다. 이 후 진행되는 메시지 전송과정에서 송신자는 최초 서명 값에 포함되는 세션 공개키에 해당하는 세션 개인키를 통하여 메시지 인증 값을 생성하게 된다.

3.2 제안하는 기법

제안하는 기법은 일정 시간 이후 키를 노출하는 방식에서 필연적으로 발생하는 지연현상을 최소화하기 위해서 이전 단계에 미리 세션 키를 생성하여 숨겨두는 방식을 활용하였다. 다음 세션에 사용되는 공개키를 미리 수신자에게 전송하기 때문에 수신자는 각 세션의 메시지에 대해 검증 과정을 바로 진행할 수 있게 된다.

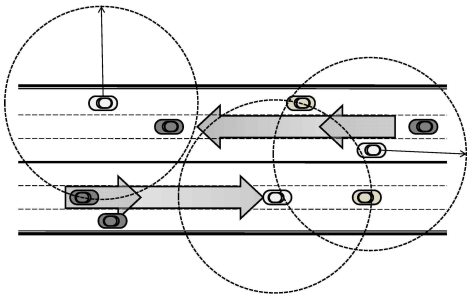
제안하는 기법은 크게 4단계로 구성된다. 차량 그룹 설정, 파라미터 설정, 인증 메시지 전송 및 검증, 마지막으로 일괄 검증과정으로 구성된다. 우선, 각 차량은 메시지를 전송하기 위해 자신의 그룹 영역을 설정하는 단계를 수행한다. 파라미터 설정단계에서는 메시지의 인증 값을 생성하기 위해 여러 공개 값들과 비밀 값들을 생성하고 송신자의 그룹 영역내의 차량들과 공유하게 된다. 이 후 송신자는 메시지를 전송할 때 인증 값을 생성하여 메시지와 함께 그룹원들에게 브로드캐스트 하게 되고, 수신자는 검증단계를 통하여 메시지를 확인하게 된다. 마지막으로 수신자는 한 번에 여러 메시지를 검증할 수 있는 일괄 검증과정을 수행할 수 있다.

3.2.1 차량 그룹 설정

각 차량에 의하여 생성되는 그룹은 [그림 1]와 같은 형태를 이루게 된다. 생성된 그룹은 각 차량의 이동성과 제한된 통신 반경 때문에 역동적으로 생성된다. DSRC[4]에 의하면 각 차량의 통신반경은 일반적으로 250m ~ 1000m이기 때문에, 각 차량의 이동속도에 따라 몇 초에서 길게는 몇 분 동안 그룹관계를 유지하게 된다[10]. 각 차량의 그룹 참가 행위가 빈번히 발생하게 된다면 그룹의 생성 과정 역시 빈번히 발생될 수 있지만, 대대수의 그룹 멤버들은 비교적 긴 시간을 함께 그룹 관계를 유지하며 이동한다고 가정되기 때문에 실제 그룹의 재설정은 빈번하지 않다[10].

3.2.2 파라미터 설정

파라미터의 설정에 앞서, 모든 차량들은 익명 아이디 PID_i를 기반으로 둔 공개키가 포함된 인증서 $CERT(PID_i)$와 개인키를 초기 키 생성 센터로부터 발급받아 각자 저장하고 있다고 가정한다[12]. 익명 아이디 PID_i와 그에 해당하는 인증서는 [12]에서와 같은 형태를 지니고 있다. 인증서를 발급하는 기관은 추적성을



[그림 1] 차량 그룹 설정 -
각 차량별 해당하는 범위만큼의 그룹을 설정한다

[표 1] 시스템 파라미터

표기법	설 명
V_i	차량 i
M_i	i번째 메시지
PID_i	차량 V_i의 익명 아이디
$CERT_i$	차량 V_i의 익명 인증서
g	그룹 Z_p*의 생성원
SK_i	차량 V_i의 개인키 (q-bits)
PK_i	개인키 SK_i에 대한 차량 V_i의 공개키 (p-bits)
r_i	i번째 메시지에 사용되는 개인키 (q-bits)
pk_i	i번째 개인키 r_i에 대한 공개키 (p-bits)
σ_i	차량 V_i의 익명 아이디에 기반을 둔 서명값
F_i	i번째 메시지에서 생성된 메시지 인증값
$H(.)$	SHA-1과 같은 안전한 일방향성 해시 함수

위하여 반드시 익명 아이디 PID_i, 해당 공개키와 인증서 $CERT(PID_i)$, 즉 실제 아이디와의 확인이 가능한 정보를 함께 저장하고 있다. 각 키 쌍들은 실제 몇 초의 짧은 시간동안만 사용된다. 이 단계에서 송신자는 여러 가지 보안 파라미터와 공개 키들을 그룹 멤버들과 공유한다. 각 기호에 대한 표기법은 [표 1]과 같다.

- p와 q는 $q|(p-1)$를 만족하는 소수이다. (일반적으로 q는 160bits, p는 512bits)
- g는 Z_p로부터 랜덤하게 생성되는 생성자로서 오더는 q이다.
- SK_i는 차량 V_i의 개인키로서 일정 기간마다 Z_q에서 랜덤하게 생성된다.
- PK_i는 개인키 SK_i에 해당하는 공개키로서 $g^{sk_i^{-1}}$와 같이 생성된다.
- r_i는 각 세션 i에 해당하는 개인키로서 Z_q*에서 랜덤하게 생성된다.
- pk_i는 세션 개인키 r_i에 해당하는 세션 공개키로서 g^{r_i}와 같이 생성된다.

차량 V_i의 그룹 내로 참가하게 되는 차량에게 그룹 리더 V_i는 공개 파라미터 (p, q, g)와 함께 공개키 PK_i를 함께 공유한다.

3.2.3 인증 메시지 전송 및 검증

본 기법에서의 인증 메시지는 [그림 2]와 같이 크게 2가지 형태로 구분되어 생성된다. 각 메시지는 송신자가 인증과정을 위하여 ECDSA 서명 값을 포함하고 있는지 그렇지 아니한지로 구분된다. 그룹 생성 후 첫 번째 메시지는 ECDSA 서명값을 포함하여 인증 메시지를 전송하고, 이 후의 메시지는 서명값을 포함하지 않고 인증 메시지를 전송하게 된다. 제안하는 프로토콜은 일정 시간에만 유지되며, 미리 정해진 시간(수 초)이 지나면 다시 첫 번째 메시지 전송 단계를 진행한다. 즉, 서명과 함께 재 생성된 개인키 SK_i에 해당하는 공개키 PK_i를 다시 공유하게 된다.

송신자		**하나 이상의 수신자들**
(1) 첫 메시지 전송 1. 세션 개인키 $r_2 \leftarrow \mathbb{Z}_q^*$ 생성 2. r_2에 해당하는 공개키 값 $pk_2 \leftarrow g^{r_2}$ 계산 3. 서명 $\sigma_i \leftarrow sign(PID_i, M_1 pk_2)$ 계산 4. $D_1 = < PID_i,\ M_1,\ pk_2, \sigma_i,\ CERT_i(PID_i) >$	D_1 →	(2) 서명 검증 5. $True$ or $False \leftarrow verify(PID_i, M_1, pk_2, \sigma_i)$ $True$이면, 메시지 M_1 승인 $False$이면, 해당 데이터 D_1 삭제 6. 다음 메시지의 검증을 위해 pk_2 저장
.....		.....
(3) n번째 메시지 전송 1. 세션 개인키 $r_{n+1} \leftarrow \mathbb{Z}_p^*$ 생성 2. r_{n+1}에 해당하는 공개키 값 $pk_{n+1} \leftarrow g^{r_{n+1}}$ 계산 3. $F_n \leftarrow SK_i \cdot (r_n \cdot H(M_n \| PID_i \| pk_{n+1}) + r_{n+1})$ 4. $D_n = < PID_i,\ M_n,\ pk_{n+1}, F_n >$	D_n →	(4) 메시지 검증 5. $n-1$번째 저장한 공개키 pk_n을 이용하여 검증 $PK_i^{F_n} ?= (pk_n)^{H(M_n \| PID_i \| pk_{n+1})} \cdot pk_{n+1}$ $True$이면, 메시지 M_n 승인 $False$이면, 해당 데이터 D_n 삭제 6. 다음 메시지의 검증을 위해 pk_{n+1} 저장

[그림 2] 제안하는 프로토콜의 메시지 인증 생성 및 검증

첫 번째 메시지 인증 값의 생성 단계는 다음과 같다.

1. 송신자 S는 랜덤 값 r_2를 Z_q 에서 생성한 후 이를 2번째 메시지의 인증을 위한 개인키로 사용한다.
2. 랜덤하게 생성된 개인키 r_2에 대한 공개키를 생성하게 위하여 송신자 S는 $pk_2 = g^{r_2}$ 를 계산한다.
3. 초기 서명을 위하여 서명함수 $sign$를 이용하여 서명값 σ_i를 생성한다.
4. 최초 서명 값을 포함한 송신자의 메시지 전송 값은 다음과 같다.

$$D_1 =< PID_i,\ M_1,\ pk_2, \sigma_i,\ CERT_i(PID_i) > \quad (1)$$

첫 번째 메시지를 받은 경우, 수신자는 우선 송신자 S의 인증서를 확인한다. 인증서가 정당하다고 확인되면 아래와 같은 과정을 통하여 메시지를 검증하게 된다.

5. 송신 데이터에 포함되어 있는 서명 값 σ_i을 확인한다.
6. 서명 값 σ_i가 정당하다면, 메시지 M_1을 승인하고, 다음 메시지의 검증을 위해 2번째 세션 키 pk_2를 저장하게 된다.

n 번째 메시지 인증 값의 생성 단계는 다음과 같다. $(n \geq 2)$

1. 송신자 S는 랜덤 값 r_{n+1}를 Z_p에서 생성한 후 이를 n번째 메시지의 인증을 위한 개인키로 사용한다.
2. 개인키 r_{n+1}에 대한 공개키 $pk_{n+1} \leftarrow g^{r_{n+1}}$를 계산한다.
3. 개인키 SK_i와 세션 개인키 r_n, 다음 세션을 위한 개인키 r_{n+1}등을 이용하여 n번째 메시지 M_n에 대한 메시지 인증 값 F_n을 다음과 같이 생성된다.

$$F_n =< SK_i \cdot (r_n \cdot H(M_n \| PID_i \| pk_{n+1}) + r_{n+1}) > \mod q \quad (2)$$

4. 송신자 S는 n번째 메시지에 대해서 다음과 같은 데이터를 전송하게 된다.

$$D_n =< PID_i,\ M_n,\ pk_{n+1}, F_n > \quad (3)$$

위에 언급된 바와 같이 메시지는 초기 메시지와 이 후의 메시지로 구분되어지기 때문에 수신자 역시 초기 메시지를 검증할 때와는 다른 검증 과정을 수행하게 된다.

n번째 메시지에 대해서, 수신자는 다음과 같은 과정을 통하여 검증과정을 진행한다.

5. 수신자는 메시지 인증값 F_n에 대해서 다음과 같은 검증과정을 진행한다.

$$PK_i^{F_n} \stackrel{?}{=} (pk_n)^{H(M_n \| PID_i \| pk_{n+1})} \cdot pk_{n+1} \mod p \quad (4)$$

위의 검증식은 다음과 같은 과정을 통하여 진행된다.

$$\begin{aligned} PK_i^{F_n} &= (g^{SK_i^{-1}})^{F_n} \mod p \\ &= (g^{SK_i^{-1}})^{SK_i \cdot (r_n \cdot H(M_n \| ID \| pk_{n+1}) + r_{n+1})} \mod p \\ &= g^{r_n \cdot H(M_n \| PID_i \| pk_{n+1}) + r_{n+1}} \mod p \\ &= g^{r_n \cdot H(M_n \| PID_i \| pk_{n+1})} \cdot pk_{n+1} \mod p \\ &= pk_n^{H(M_n \| PID_i \| pk_{n+1})} \cdot pk_{n+1} \mod p \end{aligned} \quad (5)$$

6. 메시지 인증값 F_n이 정당하다면, 수신자는 메시지 M_n을 승인하고, 다음 세션의 검증을 위해 $n+1$번째 세션 키 pk_{n+1}를 저장하게 된다.

3.2.4 일괄 검증

본 기법에서는 수신자가 전송받은 메시지들을 한 번에 검증할 수 있는 일괄 검증을 제공한다. 이러한 일괄 검증 과정은 메시지를 전송한 송신자에 따라 약간의 차이를 보인다. 일괄 검증 과정은 일괄 검증을 수행할 때 대상이 되는 메시지가 모두 동일한 송신자에 의한 것인지 그렇지 않은지의 여부에 따라서 크게 2가지로 나눠진다.

- 동일한 송신자에 의해 전송받게 된 메시지를 검증하게 될 때의 일괄 검증 과정은 다음과 같다. 아래 식은 메시지 M_n과 M_{n+1}에 해당하는 인증값 F_n, F_{n+1}를 이용하여 검증과정을 진행하게 된다.

$$\begin{aligned} PK_i^{F_n + F_{n+1}} &= (g^{SK_i^{-1}})^{F_n + F_{n+1}} \\ &= (g^{SK_i^{-1}})^{SK_i \cdot (r_n \cdot H(M_n \| PID_i \| pk_{n+1}) + r_{n+1})} \\ &\quad \cdot (g^{SK_i^{-1}})^{SK_i (r_{n+1} \cdot H(M_{n+1} \| PID_i \| pk_{n+2}) + r_{n+2})} \\ &= g^{(r_n \cdot H(M_n \| PID_i \| pk_{n+1}) + r_{n+1})} \\ &\quad \cdot g^{(r_{n+1} \cdot H(M_{n+1} \| PID_i \| pk_{n+2}) + r_{n+2})} \\ &= pk_n^{H(M_n \| PID_i \| pk_{n+1})} \cdot pk_{n+1}^{H(M_{n+1} \| PID_i \| pk_{n+2})} \\ &\quad \cdot pk_{n+1} \cdot pk_{n+2} \end{aligned} \quad (6)$$

- 각기 다른 송신자에 의해 전송받게 된 메시지를 검증하게 될 때의 일괄 검증 과정은 다음과 같다. 아래 식은 V_i의 메시지 M_m과 V_j의 메시지 M_n에

해당하는 인증값 F_m과 F_n를 이용하여 검증과정을 진행하게 된다.

$$\begin{aligned} & PK_i^{F_m} \cdot PK_j^{F_n} = (g^{SK_i^{-1}})^{F_m} \cdot (g^{SK_j^{-1}})^{F_n} \mod p \\ & = (g^{SK_i^{-1}})^{SK_i \cdot (r_m \cdot H(M_m \| PID_i \| pk_{m+1}) + r_{m+1})} \\ & \quad \cdot (g^{SK_j^{-1}})^{SK_j \cdot (r_n \cdot H(M_n \| PID_j \| pk_{n+1}) + r_{n+1})} \mod p \\ & = g^{(r_m \cdot H(M_m \| PID_i \| pk_{m+1}) + r_{m+1})} \\ & \quad \cdot g^{(r_n \cdot H(M_n \| PID_j \| pk_{n+1}) + r_{n+1})} \mod p \\ & = pk_m^{H(M_m \| PID_i \| pk_{m+1})} \cdot pk_n^{H(M_n \| PID_j \| pk_{n+1})} \\ & \quad \cdot pk_{m+1} \cdot pk_{n+1} \mod p \end{aligned} \tag{7}$$

각기 다른 차량의 메시지들을 일괄 검증할 경우에는 메시지의 수가 n개일 때 $2n$만큼의 지수 연산과 $2n+1$만큼의 곱셈 연산이 사용된다. 즉, 일괄 검증의 효율성이 오히려 떨어지게 된다. 하지만 동일 차량의 메시지들을 일괄 검증하는 경우에는 $n+1$번의 지수 연산과 $2n+1$번의 곱셈 연산을 사용하게 되어 단일 검증 과정보다 효율적이다. 예를 들어 10개의 메시지를 일괄 검증할 경우에는 단일 연산보다 약 2배의 계산 오버헤드를 줄일 수 있다.

IV. 제안하는 기법의 분석

4.1 보안 요구사항 분석

제안하는 기법은 2.3절에서 언급한 보안 요구사항을 다음과 같이 만족한다.

4.1.1 객체 인증 및 메시지 무결성

제안하는 기법에서 사용되는 송신자 S의 공개키와 개인키의 안전성은 이산대수 문제에 기반을 둔다. 즉, 공개키 PK_i에서 개인키 SK_i를 구하는 문제는 $PK_i = g^{SK_i^{-1}}$과 같이 공격자가 공개키 PK_i와 생성자 g를 알고 있더라도 SK_i를 구할 수 없다. 그러므로 식 (8)과 같이 공격자가 SK_i를 알지 못한다면 송신자 S의 정당한 서명을 위조해 낼 수 없고, 수신자는 이전 세션을 통해 전송받은 pk_n을 통해 검증 여부를 확인할 수 있다.

$$\begin{aligned} & PK_i^{F_n'} = (g^{SK_i^{-1}})^{(SK_i' (r_n \cdot H(M_n \| PID_i \| pk_{n+1}) + r_{n+1})} \\ & \neq (pk_n)^{H(M_n \| PID_i \| pk_{n+1})} \cdot pk_{n+1} = PK_i^{F_n} \mod p \end{aligned} \tag{8}$$

또한 주기적으로 생성되는 인증 메시지마다 개인키 SK_i는 랜덤하게 재 생성되기 때문에 512bits의 크기만으로도 충분히 안전하다. 이산대수 문제의 안전성은 [13]에서 증명되어 있다.

메시지 전송 때마다 송신 차량은 다음 메시지의 인증 과정에 사용될 세션 개인키 sk_i에 해당하는 세션 공개키 pk_i를 수신자에게 넘겨주게 되는데, 공개키 pk_i를 통해 개인키 sk_i를 알아내는 것은 앞서 언급된 키 쌍과 같은 이산대수 문제에 기반을 둔다. 따라서 개인키 sk_i를 알고 있는 정당한 차량이 아니라면 전 단계에 주어진 공개키 pk_i를 통해 검증식을 통과할 수 있는 서명을 생성할 수 없게 된다. 또한 각 개인키 sk_n은 매번 랜덤하게 생성되기 때문에 이전의 개인키 sk_{n-1}값은 이후 개인키 sk_n에 어떤 영향도 주지 못한다.

4.1.2 프라이버시 보장

제안하는 기법에는 서로 다른 세 가지 형태의 공개키와 개인키 쌍이 존재한다. 그 중 하나는 익명 아이디를 이용하여 신뢰받는 기관으로부터 인증을 받은 인증서에 기반을 둔 공개키와 개인키 쌍이다. 이 키 쌍들은 익명성을 보장하며 이 후 진행되는 세션에서 메시지의 안전성을 보장하기 위하여 ECDSA와 같은 서명 알고리즘으로 활용된다. 두 번째는 일정 기간마다 재 생성되는 공개키 쌍이다. 해당 키 쌍은 특정 시간 후 랜덤하게 생성되기 때문에 송신자에 대한 일체의 정보를 포함하고 있지 않고 있다. 세 번째 매 세션마다 랜덤하게 재 생성되는 키 쌍 역시 송신자에 대한 어떠한 정보를 담고 있지 않다. 즉, 첫 번째 서명 알고리즘의 생성에 사용되는 인증서의 정보에서 어떠한 사용자 정보도 노출되지 않기 때문에 본 기법은 프라이버시를 보장한다.

또한, 제안하는 기법은 차량의 연결성을 이용한 위치 추적을 막기 위해서 차량 V_i에서 사용하는 PID_i 역시 PID_{i-1}나 이 후에 사용되는 PID_{i+1}과는 아무런 관계가 없는 랜덤한 값을 사용한다. 따라서 PID를 발급하는 신뢰받는 기관을 제외한 그 누구도 이 전이나 다음 PID의 관계를 계산할 수 없다. 제안하는 기법에서는 최초 인증 이후 정해진 시간이나 횟수만큼 동일한 PID를 기반으로 메시지를 인증하기 때문에, 해당 시간만큼은 비연결성이 보장되지 않는다. 그러므로 재 인증을 위한 시간과 비연결성이 보장되지 않는 시간에는 효율성과 프라이버시 보장간의 tradeoff가 있다. 하지만, 하나의 PID는 길어야 수 초 의 시간동안만 사용되

기 때문에 결과적으로 공격자가 특정 차량의 연결성을 발견하는 데에는 한계가 있다.

4.1.3 추적성

초기 서명 알고리즘을 위하여 공개키 쌍을 생성 후 신뢰받은 기관에서 인증 받게 되는 인증서는 프라이버시를 보장하기 위하여 익명 아이디 PID_i를 기반으로 작성된다. 이때 해당 기관이 발급하는 인증서의 모든 아이디에 대한 정보를 안전한 장소에 저장하게 된다. 즉, 익명 아이디를 통하여 익명성을 보장받을 수 있지만 발급 기관만은 이 익명 아이디 PID_i와 인증서 $CERT(PID_i)$를 자신이 저장한 정보와 비교하여 실제 아이디를 확인할 수 있다. 즉, 해당 익명 아이디 PID_i와 이에 쓰이는 인증서를 통하여 실제 차량의 아이디를 추적할 수 있다. 이를 통하여 사고 발생 시 해당 익명 아이디가 실제 어떠한 인원인지를 추적할 수 있다.

4.2 효율성 분석

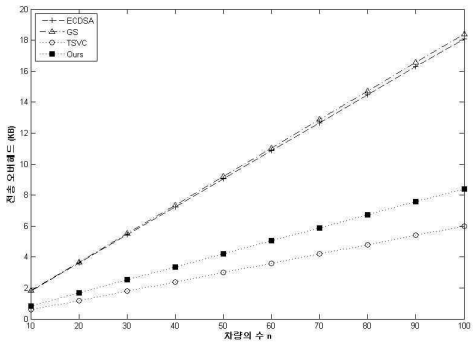
4.2.1 시뮬레이션 환경

제안하는 기법의 효율성 분석을 위하여 JAVA환경 기반의 JIST/SWANS과 같은 시뮬레이터를 사용한다[14]. 각 차량은 1000m × 30m의 시뮬레이션 환경 내에 랜덤하게 배치하도록 한다. 각 차량은 200m반경의 통신반경을 갖고 있으며, 300ms마다 주기적으로 메시지를 전송한다. 이 때 각 차량은 원홉 브로드캐스트를 통하여 메시지를 전송하며, 시뮬레이션의 총 시간은 10초로 제한한다.

제안하는 기법의 비교 분석을 위하여 기본적인 PKI 기반의 ECDSA서명[15]과 그룹 서명[8]을 함께 비교 분석한다. 또한 TESLA를 적용한 TSVC와 지연현상 및 메시지 손실 정도를 비교 분석한다.

4.2.2 전송 효율성

각 메시지는 ECDSA 서명을 위하여 서명 값과 그에 해당하는 인증서를 포함한 총 181bytes의 추가적인 용량이 필요하다[15]. Boneh 등이 제안한 그룹 서명방식에 의하면 p가 163bits일 때 각 메시지마다 추가적인 184bytes가 필요하다[7]. TSVC에서는 메시지에 해당하는 MAC값과 유효 시간값(Time Stamp), 그리고 키 노출 정보 등의 추가적인 정보를 위해

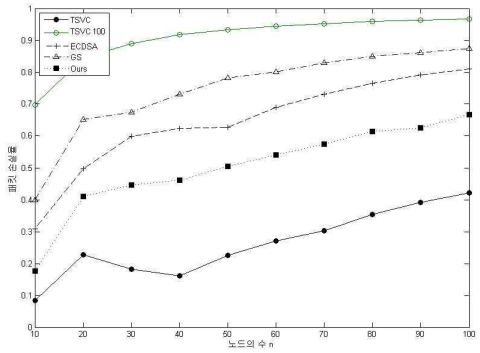


[그림 3] 차량 수에 따른 전송 오버헤드

60bytes의 용량을 필요로 한다. 그러나 본 논문에서 제안하는 기법은 추가적으로 84bytes의 용량만을 필요로 한다. 이때 20bytes는 각 메시지에 해당하는 인증 값을 의미하며 64bytes는 다음 세션을 위해 미리 전송하게 되는 공개키 값의 크기를 의미한다. 즉, [그림 3]과 같이 제안하는 기법은 ECDSA에 비해 약 2.15배, 그리고 그룹 서명 방식에 비해 약 1.94배의 전송 효율성을 갖는다. 이때 공개키 값의 크기는 $\mathbb{Z}_p$의 p값의 크기에 기반을 둔다. 그러므로 p의 크기에 따라 효율성과 안전성의 Tradeoff가 존재하게 된다. 제안하는 기법은 매 세션마다 생성되는 개인키는 랜덤하게 생성되기 때문에, 일정 시간이 지나 새로운 개인키와 공개키가 생성된 경우에는 이전의 비밀/공개키 쌍은 공격자에게 의미없는 값이 된다. 그러므로 본 논문에서 제안하는 기법은 다른 서명 방식에 비하여 p의 크기를 더욱 효율적으로 사용될 수 있다는 장점이 있다.

4.2.3 계산 효율성

PKI 환경 기반의 ECDSA 서명 검증 알고리즘은 메시지 하나 당 총 4번의 지수연산과 1번의 역원 연산이 요구된다[16]. [8]에서 제안된 그룹 서명의 검증 알고리즘은 5번의 페어링 연산과 8번의 곱셈연산, 6번의 지수 연산이 요구된다[17]. 또한 [17]에서 제안된 링 서명방식의 프로토콜은 메시지 검증을 위하여 1번의 페어링 연산과 3번의 지수연산이 요구된다. 즉, 메시지 인증을 위하여 서명 방식을 사용한 경우에는 별도의 지연현상을 발생시키지 않지만 서명된 메시지를 수신자가 검증하기 위해서는 앞서 언급한 바와 같은 계산 오버헤드가 발생한다. 하지만 제안하는 기법은 메시지 하나 당 2번의 지수 연산과 1번의 곱셈 연산으로 검증이 가능하다. 또한 시간에 민감하지 않은



[그림 4] 차량 수에 따른 패킷 손실률

메시지의 경우 같은 차량에게서 전송받은 여러 메시지들은 일괄 검증으로 효율적인 메시지 확인이 가능하다. 제안하는 기법의 계산 효율성은 4.2.4절의 메시지 손실률에도 [그림 4]와 같이 큰 영향을 미친다.

4.2.4 메시지 손실률

메시지 손실률은 각 기법이 실제 환경에 적용될 수 있는지의 여부를 결정하는 중요한 요소가 된다. 메시지 손실률은 메시지 검증에 필요한 계산과 지연 정도, 전송 정도에 의하여 결정된다. 그렇기 때문에 TSVC와의 정확한 비교를 위하여 TSVC를 2가지 환경으로 구분지어 시뮬레이션을 실시하였다. 첫 번째로는 순수한 검증시간만을 고려하여 검증 시간을 결정하고 (TSVC로 표기), 두 번째로는 키 노출 시간을 고려하여 100ms만큼의 지연 시간을 고려하여 검증 시간을 결정하였다(TSVC 100로 표기). TSVC에서 발생하는 100ms의 지연현상을 검증 과정에 추가하여 수신자의 관점에서 지연현상이 얼마나 메시지 손실률에 치명적인지를 비교하기 위하여 TSVC와 TSVC100으로 구분하였다. 패킷 손실률은 각 차량마다 손실한 총 패킷의 수를 수신한 총 패킷의 수로 나눈 비율을 의미한다. TSVC의 경우 [그림 4]와 같이 가장 효율적인 손실률을 보여주지만, 키 노출 시간을 고려하여 계산한 TSVC 100의 경우에는 PKI기반의 ECDSA서명과 그룹 서명보다 좋지 않은 손실률을 보여준다. 그에 반하여 제한하는 기법은 ECDSA서명과 평균 10% 이상의 차이를 보이며, 그룹서명과는 최대 20% 이상의 차이를 보여준다.

V. 결 론

본 논문에서 제안한 기법은 시간에 민감한 메시지에도 적용할 수 있도록 지연현상을 최소화한 브로드캐스트 메시지 인증을 보장한다. 또한 안전성이 증명된 익명 아이디 및 키 쌍을 미리 공유하는 방식을 적용하여 프라이버시를 보장하면서도 추적성을 제공하고, 이산대수 문제에 기반을 둔 공개키와 개인키 쌍을 이용함으로써 메시지 무결성을 만족한다. 그리고 그룹 서명과 ECDSA와 같은 다른 서명 기법을 사용하는 것과 비교하여 전송 및 계산 오버헤드 측면에서 효율적이며, 지연현상을 최소화하였기 때문에 DoS 공격에도 안전하며, 부인방지의 기능도 제공한다. 익명 아이디를 기반으로 하는 공개키와 개인키 쌍의 사용 횟수를 최소화하기 때문에 해당 차량의 재 인증 기간을 최소 5배까지 유지할 수 있다. 제안하는 기법은 RSU와의 별도 통신 없이 각 차량 간의 통신만으로 메시지 인증을 진행하기 때문에, 전체적인 통신 라운드 수 역시 효율적이다.

참 고 문 헌

[1] 조영준, 이현승, 박남제, 최두호, 원동호, 김승주, “VANET에서의 보안 기술동향,” 정보보호학회지, 19(1), pp. 134-142, 2009년 2월.

[2] J.A. Misener, “Vehicle-Infrastructure Integration (VII) and Satety: Rubber and Radio Meets the Road in California,” Intellimotion, vol. 11, no. 2, pp. 1-3, 2005.

[표 2] 효율성 및 요구사항 비교

구 분	추가 용량	요구 연산량†	지연현상	DoS 공격	부인방지/추적성
ECDSA[15]	181bytes	5e	X	X	O
GS[8]	184bytes	8m+6e+5p	X	X	O
TSVC[10]	60bytes	1ms(MAC검증)	O (100ms)	O	X
제안하는 기법	84bytes	2e+1m	X	X	O

† e: 지수연산, m: 곱셉연산, p: 페어링 연산 (정수환경에서의 연산)

[3] Vehicle Safety Communications Project, "U.S. Department of Transportation, National Highway Traffic Safety Administration," Final Report, Apr. 2006.

[4] DSRC: Dedicated Short Range Communications, http://grouper.ieee.org/groups/scc32 /dsrc/index.html

[5] C. Zhang, X. Lin, R. Lu, P.H. Ho, and X. Shen, "An Efficient Message Authentication Scheme for Vehicular Communications," IEEE Trans. on Vehicular Technology, vol. 57, no. 6, pp. 3357-3368, Nov. 2008.

[6] D. Chaum and E. van Heyst, "Group Signatures," in Adv. in Cryptology - EUROCRYPT, LNCS 547, pp. 257-265, 1991.

[7] D. Boneh, X. Boyen, and H. Shachamst, "Short Group Signatures," in Adv. in Cryptology - CRYPTO, LNCS 3152, pp. 41-55, 2004.

[8] X. Lin, X. Sun, P.H. Ho, and X. Shen, "GSIS: A Secure and Privacy-Preserving Protocol for Vehicular Communications," IEEE Trans. on Vehicular Tech., vol. 56, no. 6, pp. 3442-3456, Nov. 2007.

[9] 유영준, 김윤규, 김범한, 이동훈, "차량네트워크를 위한 프라이버시 보장 인증기술 동향분석," 정보보호학회지, 19(4), pp. 11-20, 2009년 8월.

[10] X. Lin, X. Sun, X. Wang, C. Zhang, P.H. Ho, and X. Shen, "TSVC: Timed Efficient and Secure Vehicular Communications with Privacy Preserving," IEEE Trans. on Wireless Communications, vol. 7, no. 12, pp. 4987-4998, Dec. 2008.

[11] A. Perrig, R. Canneti, D. Song, and J.D. Tygar, "The TESLA Broadcast Authentication Protocol," RSA Cryptobytes, vol. 5, no. 2, pp. 2-13, Summer/Fall 2002.

[12] M. Raya and J.P. Hubaux, "The Security of Vehicular Ad Hoc Networks," in Proc. on Security of ad hoc and sensor networks, Alexandria, VA, pp. 11-21, Nov. 2005.

[13] B.A. LaMacchia and A.M. Odlyzko, "Computation of Discrete Logarithms in Prime Fields," Designs, Codes and Cryptography I, pp. 47-62, May 1991.

[14] R. Barr, Z.J. Haas, R. van Renesse, K. Tamtoro, B.S. Viglietta, C. Lin, M. Fong, and E. Cheung, "SWANS-Scalable Wireless Ad hoc Network Simulator User Guide," v1.0.6, Cornell Univ. Available: http://jist.ece.cornell.edu/, Mar. 2005.

[15] IEEE Standard 1602.9, "IEEE Trial-Use Standard for Wireless Access in Vehicular Environments - Security Services for Applications and Management Messages," July 2006.

[16] American National Standards Institute, "Public Key Cryptography For The Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)," Jan. 1998.

[17] H. Xiong, M. Ripeanu, and Z. Qin, "Efficient and Spontaneous Privacy- Preserving Protocol for Secure Vehicular Communications," arXiv:0909.1590v1, Sep. 2009.

[18] 김성훈, 김범한, 이동훈, "VANET 환경에서 프라이버시를 보호하면서 사고 발생 시 추적 가능한 인증 프로토콜," 정보보호학회논문지, 18(5), pp. 115-124, 2008년 10월.

<著 者 紹 介>

유 영 준 (Young Jun Yoo) 학생회원
2008년 2월: 숭실대학교 수학과 졸업
2008년 3월 ~ 현재: 고려대학교 정보경영공학전문대학원 석사과정
<관심분야> 암호프로토콜, VANET, 네트워크 코딩, 응용암호

이 준 호 (Jun Ho Lee) 학생회원
2004년 2월: 고려대학교 전산학과 졸업
2006년 2월: 고려대학교 정보경영공학전문대학원 석사
2006년 3월 ~ 현재: 고려대학교 정보경영공학전문대학원 박사수료
<관심분야> 암호프로토콜, VANET, USIM 보안, 애드 혹 네트워크, 응용암호

이 동 훈 (Dong Hoon Lee) 종신회원
1983년 8월: 고려대학교 경제학사
1987년 12월: Oklahoma University 전산학 석사
1992년 5월: Oklahoma University 전산학 박사
1993년 3월~1997년 2월: 고려대학교 전산학과 조교수
1997년 3월~2001년 2월: 고려대학교 전산학과 부교수
2001년 2월 ~ 현재: 고려대학교 정보경영공학전문대학원 교수
<관심분야> 암호프로토콜, 암호이론, USN이론, 키 교환, 익명성 연구, PET 기술