情 報 保 護 學 會 誌
第19卷 第4號, 2009. 8

차량네트워크를 위한 프라이버시 보장 인증 기술 동향 분석

유 영 준*, 김 윤 규, 김 범 한, 이 동 훈

요 약

차량네트워크(VANET)는 이동형 에드 혹 네트워크의 가장 유망한 응용환경으로 인식되어 지고 있다. 특히, 차량 간의 안전주행 통신인 V2V의 경우 운전자의 안전을 위한 통신기술로 주목받고 있다. 안전주행을 위해서 V2V에서 송수신되는 메시지는 다양한 네트워크 공격을 막기 위해서 반드시 인증이 되어야 하는 반면 운전자의 위치 프라이버시를 보호하기 위해서는 익명성이 보장되어야 한다. 이러한 보안 속성은 V2V 통신만의 고유한 성질로써, 현재 인증과 프라이버시를 동시에 보장하기 위한 인증기술에 대한 연구가 활발히 진행되고 있다. 본 고에서는 프라이버시를 보장하는 V2V 인증 프로토콜들을 분석하고 보안 및 효율성 관점에서 각 프로토콜을 비교분석한다.

Ⅰ. 서 론

지능형 차량네트워크(VANET, Vehicular Ad Hoc Network)는 MANET(Mobile Ad Hoc Network)의 가장 유망한 애플리케이션으로 인식되고 있다. 지식경제부 5대 주력산업에 지능형자동차 분야도 포함되는 등 국내에서도 지능형 차량네트워크에 대한 연구가 점차 진행되고 있다. 지능형 차량네트워크는 통신 대상과 환경에 따라 크게 V2V (Vehicle-to-Vehicle) 통신과 V2I (Vehicle-to-Infrastructure) 통신으로 나뉜다[1]. V2V 통신은 안전주행메시지 교환을 위한 차량 간의 통신만을 말하며, V2I는 차량과 각 도로에 설치되어 있는 RSU (Road Side Unit)간의 통신을 말한다.

지능형 차량네트워크의 다양한 특성들 가운데 이동성(Mobility)과 자체구성(Self-Organization) 특성들은 MANET의 본질을 잘 나타낸다. 하지만, 이러한 특성들은 악의적인 공격자로 하여금 지능형 차량네트워크 통신의 안전을 쉽게 위협하기도 한다. 특히, V2V 통신에서의 안전성은 내부 네트워크의 참여자 혹은 RSU에 의해 브로드캐스트 되는 정보에 전적으로 의존하기 때문에 어떠한 정보도 운전자의 안전에 치명적이게 된다. V2V 통신이 제공해야 하는 다양한 보안 기능들 가운데 2가지의 상충되는 개념들이 존재한다. 바로 보안과 프라이버시이다. 즉, 운전자의 안전을 증가시키기 위해서는 참여자의 신원이 프라이버시를 위해 감추어 져야 하는 반면 다른 한 쪽에 있는 신원이 인증되어야 한다[12-13]. 또한, 차량의 서로 다른 위치 정보들은 프라이버시를 위해 연결되어선 안 되지만 긴급 상황 혹은 분쟁을 해결하기 위한 사고의 재구성을 위해 신원의 추적이 가능해야 한다. 이처럼 V2V 통신은 전통적인 무선통신 시스템과는 달리 서로 상충되는 어려운 기술적 해결을 요구하고 있다.

이러한 보안 요구사항과 함께 가용성(Availability)도 보안과 관련하여 매우 중요한 고려사항이다. V2V 통신에서 각 차량은 약 300ms 마다 안전주행메시지(Safety Message)를 브로드캐스트 하게 된다. 예를 들어, 통신 반경안에 평균적으로 50대의 차량이 있을 때, 3초 동안의 암호연산을 고려해 보자. 각 차량은 메시지 전송을 위해 10개의 전자서명을 생성하게 되고 490개의 메시지를 수신하여 각각에 대한 전자 서명을 검사해야 한다. 이처럼 V2V 인증 기법은 빈번한 암호처리를 위해 서비스 고유의 최소 요구사항을 만족하도록 효율적으로 설계되어야 한다.

현재 까지 2종류의 보안기법들이 제안되고 있다. 첫

* 고려대학교 정보경영공학전문대학원(iamakira@hanmail.net)

번째는 인증과정이 순수 차량 간의 통신 및 연산에만 의존하는 V2V 인증 기법이다. 인증, 익명성, 추적성, 비연결성, 공모방지 등의 V2V 보안요구사항을 보장하기 위해 그룹서명(Group Signature)이 필수 요소기술로 여겨지고 있으며, 이를 기반으로 다양한 프로토콜이 제시되고 있다. 이러한 프로토콜의 단점은 그룹서명의 단점을 그대로 계승하여 통신 메시지의 오버헤드가 매우 높고, 계산 효율성이 낮다는 문제점을 가지고 있다. 두 번째 종류의 인증기법은 RSU(Road Side Unit)와 같은 네트워크 인프라를 기반으로 한 인증기법이다. 일반적으로 이러한 인증기법들은 상대적으로 매우 효율적으로 연산이 가능하고, 비교적 짧은 크기의 인증메시지를 생성하는 장점을 가지고 있으나, RSU와 차량 간의 추가적인 통신을 필요로 하며 RSU가 공격자에게 노출될 경우 치명적인 보안위협이 생길 수 있다는 단점들을 가지고 있다.

본 논문에서는 V2V 인증시스템이 갖추어야 할 보안 모델을 정의하고, 프라이버시를 보장하는 V2V 인증 기법을 분류하여 연구 동향을 살핀다. 또한, 제시된 연구 결과에 대한 분석을 수행하고 향후 연구에 대한 방향을 제시한다.

Ⅱ. 보안 모델

본 장에서는 프라이버시를 보장하는 V2V 인증 시스템을 위한 보안 요구사항을 정의한다. 모든 V2V 통신은 DSRC와 같은 무선 통신을 이용하여 데이터를 주고받으므로 Bogus information 공격, DoS 공격, 통신 방해, 재생 공격(replay attack), 위조 공격, 및 ID 노출 공격 및 차량 추적 등의 여러 가지 위험에 노출되어 있다[11]. 이와 같은 위협을 막기 위해 V2V 통신프로토콜이 준수해야할 보안요구 사항과 프로토콜의 보안기능이 원활히 운용되기 위해 필요한 요구사항은 다음과 같다.

인증(Authentication): 수신자는 전송된 메시지가 정당한 사용자로부터 생성된 메시지임을 검증할 수 있어야 한다. 전통적인 공개키기반 인증시스템과는 달리 익명성 지원을 위해 정당한 사용자를 시스템의 신뢰기관에 등록되어 있는 탈퇴되지 않은 정당한 키를 소유한 사용자를 말한다. 즉, 인증의 개념이 식별(Identification)을 포함하지는 않는다.

익명성(Anonymity): 공격자가 V2V 메시지들을 캡쳐하더라도 이러한 메시지들로부터 송신자에 대한 어떠한 신원 정보도 알 수 없어야 함을 의미한다. Bellare 등[5]이 정의 한 바와 같이 익명성은 임의의 두 메시지가 같은 송신자로부터 생성된 것인지 확인할 수 없어야 한다는 성질인 비연결성(Unlinkability)을 포함한다. 즉, 수신된 메시지들로부터 송신자의 신원을 알 수 없을 뿐 아니라, 임의의 두 개의 메시지가 같은 사람으로부터 생성된 것 인지 확인될 수 없어야 함을 의미한다.

추적성(Traceability): 사고 발생 시 혹은 공격자의 출현으로 인한 피해가 발생했을 때를 대비하여 필요 시 제 3의 기관이 개입하여 특정 차량에 대한 추적을 가능하게 하는 성질이다. 이 때, 제 3의 기관이 신뢰기관이 아니라면 심각한 프라이버시 침해 문제를 야기할 수 도 있다.

가용성(Availability): 인증에 필요한 암호연산 및 통신 오버헤드가 정상적인 V2V 서비스를 저해하면 안된다는 성질이다. 브로드캐스트 통신이라는 V2V의 특성상 송신메시지의 서명보다 수시메시지의 검증과정이 압도적으로 더 많은 연산을 요구한다. 만약, 특정 메시지의 인증에 대한 연산을 수행하고 있을 때, 다음 메시지의 인증에 대한 연산이 요청되면 다음 메시지는 처리되지 못하고 삭제(Drop)되게 된다. 따라서 통신반경 안의 차량의 밀도와 차량의 이동성을 고려하여 인증에 소요되는 오버헤드를 계산하였을 때 V2V 서비스가 요구하는 최소 요구조건을 만족해야 한다.

Ⅲ. 프라이버시를 보장하는 V2V 인증기법 연구 동향

V2V의 인증은 크게 V2V 인증과 네트워크 인프라 기반 V2V 기반 인증으로 나눌 수 있다. 전자의 경우는 인증과정에서 제 3의 개체의 개입이 없어도 차량 스스로 메시지의 인증을 처리할 수 있는 프로토콜을 말하며, 후자의 경우 인증을 위해 RSU와 같은 개체가 개입을 하여 인증을 처리하는 프로토콜을 말한다. 명확하게 전자의 경우가 보다 간단하고 강력한 프로토콜로 보이나

상대적으로 오버헤드가 매우 높은 그룹서명과 같은 암호알고리즘을 필요로 한다. 반면, 후자의 경우 RSU의 적극적인 개입으로 매우 효율적인 인증을 가능하게 해 주지만, 추가적인 통신(RSU-차량)을 요구하며, RSU가 공격에 노출되었을 때 심각하게 보안이 취약해 질 수 있다. 본 장에서는 두 종류의 인증기법에 대해 자세히 살펴보도록 한다.

3.1 V2V 인증

3.1.1 GSIS: Secure Vehicular Communications with Privacy Preserving

3.1.1.1 기법의 기본 소개

GSIS는 2007년 Lin 등에 의해 그룹 서명기법에 기반을 두어 제안된 기법이다[10]. 이 때 공격자는 전송되는 모든 메시지에 대해서 도청이 가능하기 때문에 메시지의 기밀성(Confidentiality)에 대해서는 고려하지 않는다. 즉, 공격자를 포함한 메시지 수신자 모두가 메시지에 대한 검증과정을 수행할 수 있다고 가정하며, 메시지의 무결성(Integrity)과 인증(Authentication), 부인방지(Non-repudiation)에 목적을 두고 있다. 이 기법은 OBU간의 통신인 V2V환경과 OBU와 RSU와의 V2I 환경에서의 메시지 전송 방법을 구분하여 제안한다.

3.1.1.2 OBU간의 통신: V2V통신

1) 시스템 설정

각 그룹 내의 그룹 매니저는 크게 그룹 멤버 매니저(Membership Manager: MM)와 추적 매니저(Tracing Manager: TM)로서의 2가지 역할을 수행한다. 그룹 공개키와 개인키를 각 차량에 발급해 주는 TRC (Transportation Regulation Center)는 MM의 역할을 수행하고, 각 메시지의 실제 ID를 추적하기 위하여 신뢰되는 기관은 TM으로서의 역할을 수행하게 된다.

메시지 전송 단계에 앞서, 시스템 환경 설정 단계를 실행한다. TM은[3],[4]과 같은 시스템 파라미터들을 다음과 같이 생성한다.

○ $\mathbb{G}_1$, $\mathbb{G}_2$: 위수 p에 대해서 생성자 g_1과 g_2를 갖는 곱셈 순환군

○ Ψ: $\Psi(g_2)=g_1$을 만족하는 완전 동형사상

○ h, h_0: 랜덤 선택 $h \leftarrow \mathbb{G}\backslash 1_{\mathbb{G}_1}$, $h_0 \leftarrow \mathbb{G}\backslash 1_{\mathbb{G}_2}$

○ u, v: $\mathbb{Z}_p^*$에서 랜덤하게 선택된 ξ_1,ξ_2에 대하여 $u^{\xi_1}=v^{\xi_2}=h$를 만족하는 $\mathbb{G}_1$의 원소값

○ h_1,h_2: 각각 $h_1=h_o^{\xi_1}$, $h_2=h_o^{\xi_2}$를 만족하는 $\mathbb{G}_2$의 원소값

TM은 추적성을 위하여 $gmsk_t=(\xi_1,\xi_2)$를 저장하고, MM에게 다음과 같은 시스템 파라미터들을 전송한다.

$(\mathbb{G}_1,\mathbb{G}_2,\mathbb{G}_T,g_1,g_2,g,p,\Psi,\hat{e},u,v,h,h_0,h_1,h_2)$

MM은 자신의 개인키 Υ을 $\mathbb{Z}_p^*$상에서 랜덤하게 선택한 후 시스템 파라미터 $w=P_{pub}=g_2^{\Upsilon}$을 설정한다. 또한 안전한 해시 함수 H: $\{0,1\}^* \rightarrow \mathbb{Z}_p^*$과 H_1 : $\{0,1\}^* \times \mathbb{G}_T \rightarrow \mathbb{Z}_p^*$를 선택한다. MM은 최종적으로 시스템 파라미터 $param$과 그룹 공개키 gpk를 다음과 같이 설정한다.

$$param=\begin{pmatrix}\mathbb{G}_1,\mathbb{G}_2,\mathbb{G}_T,g_1,g_2,g,p,\Psi,\hat{e},\\ H,H_1,P_{pub},u,v,h,h_0,h_1,h_2\end{pmatrix}$$

$$gpk=(g_1,g_2,g,w)$$

2) OBU간의 메시지 인증 프로토콜

그룹 멤버 등록: 등록 단계 동안 MM은 ID_i의 차량 i에 대한 개인키 $gsk[i]=(A_i,x_i)$를 다음과 같이 발급한다.

$$x_i \leftarrow H(\Upsilon, ID_i) \in \mathbb{Z}_p^*$$

$$A_i \leftarrow g_1^{1/(\Upsilon+x_i)}$$

이때 MM은 사고발생시의 추적을 위하여 (A_i,ID_i)을 기록한다.

메시지 서명: 차량 i는 MM으로부터 받은 그룹 공개키 gpk와 개인키 (A_i,x_i)를 이용하여 다음과 같은 과정을 통하여 메시지 M에 대한 서명과정을 수행한다.

- 랜덤 값 $\alpha,\beta \leftarrow \mathbb{Z}_p^*$을 선택
- A_i의 암호화 값과 (T_1,T_2,T_3), δ_1과 δ_2을 계산

$$T_1 \leftarrow u^{\alpha},\ T_2 \leftarrow v^{\beta},\ T_3 \leftarrow A_i h^{(\alpha+\beta)},$$

$$\delta_1 \leftarrow x_i\alpha,\ \delta_2 \leftarrow x_i\beta$$

- 서명 블라인드 값 r_α, r_β, r_{x_i}, r_{δ_1}, r_{δ_2}을 $\mathbb{Z}_p^*$에서 랜덤하게 선택
- R_1, R_2, R_3, R_4, R_5값을 아래와 같이 계산

$R_1 \leftarrow u^{r_\alpha}$, $R_2 \leftarrow v^{r_\beta}$,

$R_3 \leftarrow e(T_3,g_2)^{r_{x_i}} \cdot e(h,w)^{-r_\alpha - r_\beta} \cdot e(h,g_2)^{-r_{\delta_1}-r_{\delta_2}}$

$R_4 \leftarrow T_1^{r_{x_i}} \cdot u^{-r_{\delta_1}}$, $R_5 \leftarrow T_2^{r_{x_i}} \cdot u^{-r_{\delta_2}}$

- 앞서 계산한 값들을 이용하여 c값과 서명 원소 s_α, s_β, s_{x_i}, s_{δ_1}, s_{δ_2}를 계산

$c \leftarrow H(M, T_1, T_2, T_3, R_1, R_2, R_3, R_4, R_5) \in \mathbb{Z}_p^*$

$s_\alpha = r_\alpha + c\alpha$, $s_\beta = r_\beta + c\beta$,

$s_{x_i} = r_{x_i} + cx_i$, $s_{\delta_1} = r_{\delta_1} + c\delta_1$,

$s_{\delta_2} = r_{\delta_2} + c\delta_2$

위의 계산 결과 메시지 M에 대한 차량 i의 서명값은 다음과 같다.

$$\sigma \leftarrow (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_{x_i}, s_{\delta_1}, s_{\delta_2})$$

메시지 검증: 메시지 수신 차량은 우선 해당 서명값이 정당한 시간 내에 작성되어 전송되었는지를 검증한다. 정당하다고 판단될 경우에는 전송받은 파라미터를 이용하여 $\tilde{c}$값을 아래와 같이 계산한다.

$\tilde{c} \leftarrow H(M, T_1, T_2, T_3, \widetilde{R_1}, \widetilde{R_2}, \widetilde{R_3}, \widetilde{R_4}, \widetilde{R_5},)$

이때, 각 $\widetilde{R_i}$값은 다음과 같이 계산된다.

$\widetilde{R_1} \leftarrow u^{s_\alpha}/T_1^c$, $\widetilde{R_2} \leftarrow v^{s_\beta}/T_2^c$,

$\widetilde{R_3} \leftarrow e(T_3,g_2)^{s_{x_i}} \cdot e(h,w)^{-s_\alpha - s_\beta} \cdot e(h,g_2)^{-s_{\delta_1}-s_{\delta_2}} \cdot (e(T_3,w)/e(g_1,g_2))^c$

$\widetilde{R_4} \leftarrow T_1^{s_{x_i}} \cdot u^{-s_{\delta_1}}$, $\widetilde{R_5} \leftarrow T_2^{s_{x_i}} \cdot v^{-s_{\delta_2}}$

생성된 $\tilde{c}$값과 서명값에 포함되어 있는 c값이 같다면 수신자는 해당 메시지가 정당하다고 판단한다.

메시지 추적: TM은 초기단계에 저장해 놓은 (A_i, ID_i)를 이용하여 각 메시지에 해당하는 서명값에서 해당 차량의 실제 아이디를 확인할 수 있다.

- 서명값이 정당한지 확인
- $A_i \leftarrow T_3/(T_1^{\xi_1} \cdot T_2^{\xi_2})$를 계산
- 저장하고 있는 리스트에서 A_i에 해당하는 ID_i를 확인

그룹 멤버 폐기(revocation): 이 기법은 효율적인 멤버 폐기 과정을 위하여 기존의 대표적인 2가지 폐기 방법을 혼합하여 사용한다. 하나는 폐기 목록에 해당하는 모든 공개키와 비밀키 쌍을 포함하여 정당한 멤버를 구분하는 방법이며, 다른 방법은 특정 검증자만이 폐기 과정을 수행할 수 있는 VLR(Verifier-Local Revocation)[1][3]이다. VLR은 차량 수가 적을 때에는 앞선 방법보다 효율적이지만 폐기되는 차량 수에 대한 검증시간이 선형적으로 증가하기 때문에 일정 수 이상의 차량에 대해서는 비 효율적이다. 때문에 이 기법에서는 미리 임계값 T_γ를 설정하여 해당 수보다 낮은 차량에 대한 폐기 목록인 경우에는 VLR 방식을 적용하고 임계값보다 많은 수의 차량을 대상으로 하는 경우에는 폐기 목록에 모든 그룹 멤버의 개인키를 저장하여 업데이트하는 방식을 적용한다.

3.1.1.3 OBU와 RSU간의 통신

RSU의 익명성을 보장할 필요가 없기 때문에 각 RSU의 메시지는 잘 알려져 있는 ID기반의 서명 기법[2]을 사용하여 각 차량에게 전송한다. 해당 기법을 통하여 공개키의 업데이트 및 폐기 목록 연산의 오버헤드를 크게 줄일 수 있으며, 인증서의 관리역시 간단해 진다.

개인키 생성: MM은 RSU i의 고유한 ID_i에 대하여 다음과 같은 과정으로 개인키 S_{ID_i}를 발급한 후 RSU에게 안전한 채널을 통하여 전송한다.

$$S_{ID_i} \leftarrow g_1^{1/(\Upsilon + h(ID_i))}$$

이때 RSU의 고유한 ID_i는 [표 1]과 같은 형태를 지니고 있다.

[표 1] RSU의 ID_i의 포맷

시리얼 번호	물리적 위치 정보	ID 형태

메시지 서명: RSU는 메시지 M에 대하여 다음과 같은 과정을 통하여 서명값 $\sigma = (h_\sigma, S_\sigma)$을 계산한다.

- 랜덤값 선택 $x \leftarrow \mathbb{Z}_p^*$ 그리고 $r \leftarrow g^x \in \mathbb{G}_T$ 계산
- $h_\sigma \leftarrow H_1(M,r) \in \mathbb{Z}_p^*$ 계산
- $S_\sigma \leftarrow S_{ID_i}^{x+h_\sigma} \in \mathbb{G}_1$ 계산

메시지 검증: 메시지 수신 차량은 우선 해당 메시지를 송신한 RSU의 물리적 위치정보를 확인하여 정당한 지역의 RSU인지를 검증한다. 메시지의 ID형태와 RSU의 ID내에 포함된 ID형태를 비교한다. 또한 정당한 시간에 작성된 메시지인지의 여부 역시 확인한다. 위의 3가지 사항에 대해서 정당하다고 판단될 경우에는 아래와 같은 과정을 통하여 서명값을 검증한다.

$$\widetilde{h_\alpha} \leftarrow H_1\left(M, e(S_\sigma, g_2^{H(ID_i)} \cdot P_{pub}) g^{-h_\sigma}\right)$$

생성된 $\widetilde{h_\alpha}$값과 서명값에 포함되어 있는 h_α값이 같다면 수신자는 해당 메시지가 정당하다고 판단한다.

이 기법은 각 환경에 맞추어 그룹 서명방식과 ID기반 서명방식을 사용하여 익명성을 보장하며 보안 요구사항들을 만족하는 메시지 인증 프로토콜이다.

3.1.2 Efficient and Robust Pseudonymous Authentication in VANET

2007년 Calandriello등은 이전에 제안된 BP(Baseline Pseudonyms) 기법과 GS(Group Signatures) 기법을 조합하여 차량 네트워크에서 메시지 인증에 요구되는 보안 요구사항을 만족시키면서도 효율적인 Hybrid scheme을 제안한다[6]. 이 때 발생하는 오버헤드나 robustness등의 문제점은 최적화 과정을 통해 해결한다.

3.1.2.1 BP (Baseline Pseudonyms)

각각의 차량이 pseudonym set을 사용하는 방법이다. 차량 V의 i번째 pseudonym K_V^i에 대해 CA (Certificate Authority)는 $Cert_{CA}(K_V^i)$를 발급하고, 각 V는 K_V^i에 대응하는 비밀키 k_V^i를 통해 메시지를 사인한다. $\sigma_{k_V^i}(m)$은 V의 i번째 pseudonym과 메시지 m을 말하며 여기서 사용되는 message format은 다음과 같다.

$$m, \sigma_{k_V^i}(m), K_V^i, Cert_{CA}(K_V^i)$$

3.1.2.2 GS (Group Signatures)

각각의 차량 V는 CA에 등록된 안전한 그룹 서명 키 gsk_V를 가지고 서명을 하고, 서명된 값 $\sum^{GAV}(m)$은 그룹 공개키 gpk_{CA}에 의해 각각의 V가 확인할 수 있다. 여기서의 메시지는 다음과 같다.

$$m, \sum^{GAV}(m)$$

3.1.2.3 Hybrid 기법

Baseline Pseudonym scheme과 Group Signature scheme을 조합한 scheme이다. Hybrid scheme에서는 group signature를 생성하는 대신 pseudonym $\{K_V^i\}$를 생성한다. 각각의 V는 그룹 서명키 gsk_V를 사용해 서명하고, 서명된 $\sum^{GAV}(K_V^i)$를 gsk_V를 사용하여 확인한다. $Cert_{CA}^H(K_V^i)$는 CA에게 인가받은 V 스스로 생성한 인증서를 말하며 이 scheme의 message format은 다음과 같다.

$$m, \sigma_{k_V^i}(m), K_V^i, Cert_{CA}^H(K_V^i)$$

3.1.2.4 최적화 과정

오버헤드를 줄이고 Robustness을 향상시키기 위해서 세 가지 최적화 방법을 제안한다.

1) 수신측 V는 앞서 받은 $Cert_{CA(K_V^i)}^H$을 저장시켜놓고, 추후 받은 $Cert_{CA(K_V^i)}^H$가 이미 저장되어 있으면 확인절차를 생략한다.

2) α번의 message마다 m, $\sigma_{k_V^i}(m)$, K_V^i, $Cert_{CA(K_V^i)}^H$를 모두 같이 보내고, 나머지 α-1개의 message에는 $\sigma_{k_V^i}(m)$와 4byte의 keyID만 같이 보낸다. Pseudonym이 바뀌면 m, $\sigma_{k_V^{i+1}}(m)$, K_V^{i+1}, $Cert_{CA(K_V^{i+1})}^H$을 전송한다. 하지만 특히 근접한 차량이나 고속의 차량 간의 통신에서 만약 $m, \sigma_{k_V^{i+1}}(m), K_V^{i+1} Cert_{CA(K_V^{i+1})}^H$을 받지 못한다면 α만큼을 기다려야 한다는 Robust측면에서의 단점을 가지고 있다.

3) K_V^{i+1}가 발생하면 p번의 message마다 m, $\sigma_{k_V^{i+1}}(m)$,

K_V^{i+1}, $Cert_{CA(K_V^{i+1})}^{H}$을 보낸다. 이로 인해 2)에서 보였던 Robust의 문제점을 보완할 수 있다.

Pseudonym과 그룹 서명을 통해 익명성을 제공하여 프라이버시를 보호하고, 만약 OBU가 CA에 등록된 정당한 OBU라면, 자신이 생성한 Pseudonym에 대해 스스로 인증서를 발급할 수 있으므로 CA의 부담을 줄일 수 있다. 하지만 최적화 과정을 통하여 속도향상이 있더라도, 메시지 인증을 위하여 Pseudonym을 사용한 서명을 이용하므로, 주위에 많은 차량이 있을 시 메시지 인증 딜레이가 생길 수 있다.

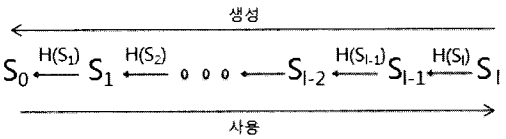
3.2 네트워크 인프라 기반 V2V 인증

3.2.1 TSVC: Efficient and Secure Vehicular Communications with Privacy Preserving)

3.2.1.1 TESLA

2008년 Lin 등이 제안한 TSVC는 효율적인 브로드캐스팅 인증 프로토콜인 TESLA를 차량 네트워크 환경에 적용하여 Hash chain의 각 원소를 메시지 인증을 위한 MAC의 키로 사용한다.

TESLA는 Perrig 등이 제안한 브로드캐스팅 인증 프로토콜로 Hash chain [그림 1]을 통해 임의의 랜덤 값(S_l)을 통해 키 배열을 생성 한 뒤, 각 시간 간격에 할당된 키를 사용하여 MAC값을 구한다. TESLA는 각 단계별 생성되는 MAC을 통해 해당 메시지의 인증을 수행한다.



〔그림 1〕 Hash chain 생성 및 사용

3.2.1.2 인증 및 서명 과정

TSVC 인증과 서명 과정은 다음과 같은 과정을 거친다.

- 송신자는 Hash 함수를 사용하여 미리 Hash chain을 생성한다.
- 송신자는 Hash chain의 첫 번째 원소인 h_l을 사용하여 메시지에 대한 MAC값을 생성한 후 P_l을 보낸다.

송신자		수신자
(1) 첫 메시지 전송 • Hash chain 생성 h_1, h_2, • • •, h_n • 메시지 M_l 생성 • MAC 계산 $MAC_{hl}(M_l\|\|T_l)$ • P_l = <PVID, M_l, $MAC_{hl}(M_l\|\|T_l)$, T_l>	P_l →	(2) P_l을 버퍼에 저장
(3) 첫 키 전송 • D초 동안 지연 • 서명(σ) = $Sign_{sks}(h_l, index, T_l')$ • Kr_P_l = <PVID, σ, h_l, index, T_l', CertS>	Kr_p_l →	(4) 검증 • 검증(h_l, index, T_l', σ)?=1 • 검증이 성공하면 • MAC값을 통해 메시지 인증 • $MACh_l(M_l\|\|T_l)$? = $MAC_{hl}'(M_l\|\|T_l)$
.....		.....
(5) j 번째 메시지 전송 • 메시지 M j 생성 • MAC 계산 $MAC_{hj}(M_j\|\|T_j)$ • P_j = <PVID, M_j, $MAC_{hj}(M_l\|\|T_j)$, T_j>	P_j →	(6) P_j을 버퍼에 저장
(7) j 번째 키 전송 • D초 동안 지연 • Kr_P_j = <PVID, h_j, index=j>, (j>1)	Kr_p_j →	(8) 메시지 검증 • 검증 $h_i = H^{j-i}(h_j)$ • 검증이 성공하면 • MAC값을 통해 메시지 인증 • $MAC_{hl}(M_j\|\|T_j)$? = $MAC_{hl}'(M_j\|\|T_j)$

〔그림 2〕 TSVC 프로토콜

- 미리 설정된 키 공개 지연 시간(D) 후에 송신자는 첫 번째 hash chain 원소인 h_1에 전자서명을 한 후, 수신자에게 전달한다. 수신자는 서명이 정당하면, h_1을 자신의 버퍼에 저장한다.
- 수신자는 두 번째 메시지부터 Hash chain의 원소 길이동안 키를 인증하기 위해서 송신자로부터 얻은 정당한 h_1과 Hash chain 성질을 통해 키(h_j)를 인증한다.

TSVC는 처음 메시지를 전송하는 단계에서만 서명을 통해 인증을 하고, 나머지 메시지들은 Hash chain 길이의 기간 동안 간단한 Hash 함수 연산과 MAC 연산을 통해 빠른 인증이 가능하다. 따라서 차량이 많은 상황에서 TSVC 프로토콜은 전자서명에 비해 연산속도가 빠르므로 메시지 손실비율이 적다. 또한 익명 ID를 사용함으로써 익명성도 제공된다.

하지만 일정시간이 지난 후에 키를 공개하므로, 부인방지가 불가능하고, 메시지 인증이 실시간으로 이루어지지 않으므로, 공격자는 임의의 불필요한 메시지를 발생시켜 버퍼에 저장시키는 DoS 공격이 가능하다.

3.2.2 IBV: Identity-based Batch Verification Scheme

2008년 Zhang et al.에 의해 ID-based 서명기법에 기반하며 일괄 검증이 가능한 기법(이하 IBV)이 제안되었다[15]. 이 기법은 위의 두 가지 목적 중 전자에 해당하는 환경을 가정하고 있다. 즉, 많은 수의 차량이 RSU에게 메시지를 전달하여 인증받기 위한 환경을 가정으로 하고 있다. 인증을 위한 서명과 검증과정은 다음과 같은 과정을 거친다.

1) PID 및 개인키 생성단계

차량 V_j는 TA(Trust Authority)를 통해 공유한 공개 파라미터 G, q, P, P_{pub1}, P_{pub2}를 이용하여 PID(Pseudo ID)쌍 (ID_1, ID_2)를 생성한다. G는 P에 의해 생성되는 순환군이며 q는 G의 위수(order)이다. P_{pub1}과 P_{pub2}는 TA가 간직하고 있는 마스터키(s_1, s_2)에 의해 생성된 공개키 쌍이다.

$$ID_1 = r \cdot P$$
$$ID_2 = RID \cdot H(r \cdot P_{pub1})$$

이때 r은 freshness를 위해 생성하는 랜덤값이며, RID는 차량 V_j의 실제 ID정보이다. 개인키 쌍 SK=(SK_1, SK_2)는 다음과 같이 생성된다.

$$SK_1 = s_1 \cdot ID_1$$
$$SK_2 = s_2 \cdot H(ID_1 \| ID_2)$$

2) 서명단계

메시지 M_i을 서명하기 위해서 차량 V_j는 다음과 같이 서명 σ_i을 생성한 후 자신의 PID와 M_i를 함께 포함하여 RSU에게 보내게 된다.

$$\sigma_i = SK_1^j + h(M_i) SK_2^j$$

3) 검증단계

일반적인 단일 메시지에 대한 검증과정은 다음과 같다.

$$e(\sigma_j, P)? = e(ID_1^j, P_{pub1}) \cdot e(h(M_i) H(ID_1^j \| ID_2^j), P_{pub2})$$

IBV의 장점인 일괄 검증단계는 단일 검증과정에서 각 메시지에 해당하는 서명과 ID 그리고 해시값들을 일괄적으로 더해줌으로서 이뤄지게 된다.

$$e(\sum_i \sigma_j, P)? = e(\sum_i ID_1^j, P_{pub1}) \cdot e(\sum_i h(M_i) H(ID_1^j \| ID_2^j), P_{pub2})$$

IBV는 RSU에서의 계산 효율성을 위해서 일괄 검증 방법이 가능한 기법을 제안하였다. 일괄 검증 방법을 이용하면 많은 수의 메시지를 검증하는 환경에서 눈에 띄는 효율성을 얻을 수 있게 된다[7]. 단일 검증때 3번의 페어링 연산이 필요한 경우 n개의 메시지를 검증하는데 필요한 페어링 연산은 3n번이다. 하지만 [7]에서 제안한 일괄 검증과정을 사용하면 단 3n번의 페어링 연산으로 n개의 메시지를 검증할 수 있게 된다[15]. 때문에 이 기법은 시내와 같이 하나의 RSU당 차량의 밀도가 높은 지역에서 효율성이 더욱 극대화되어진다는 결정적인 장점이 존재한다. 하지만 ID-based 서명의 검증 오버헤드가 상당하다는 근본적인 단점이 있기에 고속도로나 시외지역과 같은 단위 RSU당 해당 차량의 수가 적을 경우에는 오히려 비효율적일 수 있다.

3.2.3 RAISE: RSU-Aided Message Authentication Scheme

RAISE는 IBV 기법과는 다른 목적을 두고 설계된 방법이다. 이 방법에서 RSU는 단지 차량간 통신을 위한 인증 과정을 도와주는 역할을 할 뿐이다[14]. 즉, RAISE는 궁극적으로 차량 간의 통신 인증을 위한 프로토콜이다.

이 기법은 서명의 검증 효율성을 위해서 기존의 PKI 기반의 서명방법이 아닌 대칭키를 사용한 HMAC기반의 인증 프로토콜을 제안한다.

1) 대칭키 설립 단계

차량 V_j와 RSU R_j가 서로 HMAC를 사용하기 위해 필요한 대칭키를 설립하는 과정으로 Diffie-Hellman Key agreement 과정을 이용하여 키 K_j를 설립하게 된다. 이 단계에서는 양방향인증을 위해 서로의 공개키와 인증서를 사용하는 기존의 PKI기반의 서명기법을 이용하고 RSU는 PID를 차량 V_j에게 할당해주게 된다. 추후 추적성을 위해 RSU는 차량에게 전해준 PID와 대칭키 값을 테이블에 저장하게 된다.

2) RSU의 해시 통합 단계(Aggregation)

(1) $V_j \rightarrow R_j$:
$(ID_j \parallel M_i \parallel HMAC(K_j \| ID_j \| M_i))$

(2) RSU R_j 검증단계 :
2-1) ID_j가 테이블에 있는지 확인
2-2) HMAC값을 K_j로 확인

(3) 해시 통합 값 전달 :
RSU는 해시 통합 값 HAggt를 아래와 같이 계산 후 서명과 함께 범위내의 각 차량에게 브로드캐스트한다.

$HAggt = H(ID_1 \| M_1) \| \cdots \| H(ID_n \| M_n)$

3) 검증단계

HAggt 값을 전달받은 차량들은 자신이 차량 V_j로 받은 메시지 $\{ID_j, M_i, H(ID_j, M_i)\}$가 정당한지를 위해서 아래와 같은 검증과정을 거치게 된다.

(1) RSU R_j의 서명 검증

(2) 자신이 받은 메시지의 해시값이 HAggt에 포함되어 있는지 검증

이 기법은 대칭키 기반의 HMAC 인증기법을 사용함으로써 단일 검증과정 역시 효율적이라는 장점이 있다. 하지만 본 논문에서는 소개하지 않았지만 프라이버시 보호를 위해 도입된 k-anonymity 컨셉 역시 문제점이 존재하며, 차량간의 통신을 위해 여러 번 데이터들이 전송되어야 한다는 전송 오버헤드면에서도 문제점이 존재한다. 또한 위의 프로토콜이 차량에 대한 인증이 아닌 메시지 하나에 대한 인증이라는 점 역시 전체적인 관점에서 효율적이지 못하다. 결정적으로 RSU가 메시지를 받은 후 계산하여 브로드캐스트하게 되는 HAggt값은 메시지가 많을수록 그 길이가 길어진다는 단점이 있다.

3.3 비교 분석

[표 2]은 지금까지 살펴본 인증 프로토콜들을 비교분석한 결과이다. 그룹서명을 기반으로 하고 있는 GSIS

〔표 2〕 프라이버시를 보장하는 V2V 인증 기법들의 비교 분석

구분	V2V 인증		네트워크 인프라 기반 V2V 인증		
기법	GSIS	Calandriello 등의 기법	TSVC	IBV	RAISE
핵심 아이디어	· 그룹서명 사용	· 그룹서명 사용 · Pseudonym 사용 · Pseudonym 최적화	· 변형된 TESLA 사용	· 익명 ID-기반 서명 · Batch Verification · 익명 ID를 RSU가 발급	· RSU가 인증 대행
장점	· RSU와의 추가적인 통신이 필요 없음	· RSU와의 추가적인 통신이 필요 없음 · Pseudonym 최적화로 최소의 그룹서명 사용	· 연산 및 패킷 오버헤드 적음	· Batch Verification을 통해 연산효율 높임	· 연산 오버헤드 거의 없음
단점	· 높은 연산 오버헤드 · 큰 메시지 크기	· 비교적 높은 연산 오버헤드 · 비교적 큰 메시지 크기	· 추가적인 통신 필요 · 인증 지연 문제 · RSU 공격에 취약	· 추가적인 통신 필요 RSU 공격에 취약	· 추가적인 통신 필요 · RSU 공격에 매우 · 취약

와 Calandriello 등의 프로토콜은 상대적으로 보안성이 높고 추가적인 통신 오버헤드가 없는 반면에, RSU를 사용하고 있는 프로토콜들은 연산 효율성이 매우 높은 반면에 추가적인 통신을 필요로 하며 RSU 공격에 매우 취약한 단점을 지니고 있다.

Ⅳ. 결 론

본 논문에서는 V2V 통신을 위한 보안 요구사항 들을 제시하고 이를 기반으로 인증 프로토콜 연구 동향에 대한 분석을 수행 하였다. 현재까지 제안된 프로토콜은 각각이 단점을 지니고 있어서 향후에는 V2V 보안요구사항을 준수하는 새로운 형태의 효율적인 전자서명 혹은 RSU 공격에 안전한 네트워크 인프라 기반 인증 프로토콜에 대한 집중적인 연구가 필요하다.

참고문헌

[1] G. Atenies, D. Song and G. Tsudik, "Quasi-Efficient Revocation of Group Signatures," in Proceedings of Financial Cryptography, LNCS 2357, pp. 183-197, 2002.

[2] Paulo S. L. M. Barreto, et al., "Efficient and provably-secure identity-based signatures and signcryption from bilinear maps," Asiacrypt 2005, pp. 515-532, 2005.

[3] D. Boneh, X. Boyen, and H. Shacham, "Short Group Signatures", in Advances in Cryptology-CRYPTO 2004, LNCS 3152, pp. 45-55, Springer-Verlag, 2004.

[4] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," CRYPTO 2001, LNCS 2139, pp. 213-229, 2001.

[5] M. Bellare, D. Micciancio, and B. Warinschi. "Foundations of Group Signatures: Formal Definition, Simplified Requirements and a Construction Based on General Assumptions", in Advances in Cryptology-Eurocrypt 2003, vol. 2656, LNCS, pp. 614-629, 2003.

[6] G. Calandriello, P. Papadimitratos and J. P. Hubaux, Efficient and Robust Pseudonymous Authentication in VANET, in Proc. International Workshop VANET, pp. 19-28, 2007.

[7] J. Camenisch, S. Hohenberger and M. Pedersen, Batch verification of short signatures, in Proceedings of EUROCRYPT, LNCS, Vol. 4514, pp. 246-263, 2007.

[8] M. Gerlach and F. Guttler, "Privacy in VANETs using Changing Pseudonyms-Ideal and Real", in VTC2007, Dublin, Ireland, April 2007.

[9] 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, Dec. 2008.

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

[11] M. Raya, P. Papadimitratos, and J.-P. Hubaux, "Securing vehicular communications", in IEEE Wireless Communications Mag., vol. 13, num. 5, pp. 8-15, Oct. 2006.

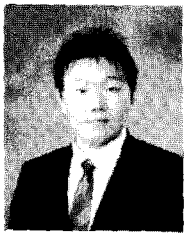
[12] A. Wasef and X. Shen, "PGCV: Privacy Preserving Group Communications Protocol for Vehicular Ad Hoc Networks", in "Proc. IEEE ICC 2008", Beijing, China, May 19-23, 2008.

[13] K. Sha, Y. Xi, W. Shi, L. Schwiebert, and T. Zhang, "Adaptive privacy-preserving authentication in vehicular networks", in Proc. International Workshop on Vehicle Communication and Appliations, Oct. 2006.

[14] C. Zhang, X. Lin, R. Lu and P. -H. Ho. "RAISE: An Efficient RSU-aided Message Authentication Scheme in Vehicular Communication Networks", in IEEE International Conference on Communications (ICC 2008), Beijing, China, May 19-23, 2008.

[15] C. Zhang, R. Lu, X. Lin, P. H. Ho and X. Shen, An Efficient Identity-based Batch Verification Scheme for Vehicular Sensor Networks, IEEE INFOCOM, 2008.

〈著者紹介〉



유 영 준 (Young Jun Yu)
학생회원

2008년 2월: 숭실대학교 수학과 졸업
2008년 3월~현재: 고려대학교 정보경영공학전문대학원 석사과정
<관심분야> 암호프로토콜, VANET, 네트워크 코딩, 응용암호

김 윤 규 (Yungyu Kim)
학생회원

2008년 2월: 명지대학교 컴퓨터공학과 졸업
2008년 3월~현재: 고려대학교 정보경영공학전문대학원 석사과정
<관심분야> 암호프로토콜, VANET, 응용암호, 암호시스템

김 범 한 (Bum Han Kim)
학생회원

2004년 2월: 숭실대학교 수학과 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