Journal of The Korea Institute of Information Security & Cryptology(JKIISC) ISSN 1598-3986(Print)
VOL.23, NO.4, August 2013 ISSN 2288-2715(Online)
http://dx.doi.org/10.13089/JKIISC.2013.23.4.593

ECU간 기기인증을 위한 HB-Family 경량인증기법의 적용 방법*

김 태 수,† 김 효 승, 우 사 무 엘, 이 동 훈‡
고려대학교 정보보호대학원

An Implementation Method of HB-Family Light-weight Authentication for Device Authentication between ECU*

Tae Su Kim,† Hyoseung Kim, Samuel Woo, Dong Hoon Lee‡
Graduate School of Information Security, Korea University

요 약

현대 차량 내부 네트워크는 ECU라고 불리는 소형 전자제어 장치로 구성되어 있다. 과거에는 주행 중인 차량의 내부 네트워크에는 접근할 수 있는 방법이 없었고, 따라서 차량 내부 네트워크 폐쇄적인 환경으로 인식되었으며 이로 인하여 내부 네트워크를 구성하고 통신하는 기기간의 인증기법이 존재 할 필요가 없었다. 하지만 현재 통신기술이 발전함에 따라 차량 내부 네트워크에 접근할 수 있는 다양한 방법이 등장하였고, 이로 인하여 발생할 수 있는 차량내부 네트워크를 구성하는 ECU간의 기기인증 문제에 대하여 관심이 집중되고 있다. HB-Family 기법은 RFID 환경에서 대표적인 경량인증기법이며, RFID 환경은 차량 내부 네트워크와 비슷한 제약사항을 가지고 있다. 따라서 본 논문에서는 ECU의 약한 연산처리 능력과 CAN 프로토콜의 제한적인 메시지 전송량을 고려하여 효율적인 기기 인증을 수행하기 위해 HB-Family 경량인증기법을 차량 내부 CAN에 적용하는 방법을 제안한다. 제안하는 방법을 적용한 인증기법의 가용성과 효율성을 평가하기 위해 DSP-F28335 Device기반의 성능평가를 수행한 결과 실험 환경에 따라 수행속도를 최소 10%에서 최대 36%의 속도를 향상 시킬 수 있었으며, 이를 차량 내부 네트워크의 다양한 측면에서 분석한다.

ABSTRACT

The In-Vehicle-Networking(IVN) of modern cars is constituted by an small electronic control device called ECU. In the past, there was no way to be able to access the IVN of a driving car. so IVN has been recognized as a closed environment so there is no need to exist authentication protocol between devices which are to configure the internal network and to communicate with other devices. However, constant improvements made it possible to access the IVN in many different ways as the communication technology evolves. This possibility created a need for device authentication in IVN. HB-Family are representative authentication schemes in RFID environment which has similar restrictions to IVN. In this paper, we propose an implementation method of HB-Family for device authentication between ECU considering ECU has low computing power and the message field of CAN protocol has restricted size of 8 bytes. In order to evaluate the efficiency and availability of the authentication schemes adopted our method, we have evaluated the performance based on DSP-28335 device. Further, it was possible to improve the efficiency rate of at lest 10%, up to 36%, and we then analyze this result in various aspects of the IVN.

Keywords: Authentication, Light Weight Authentication, HB-Family, ECU, CAN

접수일(2013년 4월 26일), 수정일(2013년 7월 10일), 게재확정일(2013년 8월 1일)
* 이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. 2010-0029121).
† 주저자, guimonica@korea.ac.kr
‡ 교신저자, donghlee@korea.ac.kr(Corresponding author)

I. 서 론

초창기의 자동차는 변속기나 연료분사장치 등 대부분의 장치가 기계식이었다. 1980년대에 전자식 연료분사장치가 등장하였고, 자동변속기와 ABS(Anti-lock Braking System)등이 개발되면서 차량의 전자화가 진행되었으며, 2000년대에는 엔진구동, 제동, 조향장치 등 차량의 대부분의 영역을 전자장비가 제어하게 되었다. 이러한 환경에서 주행 중인 자동차의 내부 네트워크에 접근할 수 있는 방법이 없었기 때문에 기존의 차량 내부 네트워크는 암호시스템과 인증기법이 필요 없는 폐쇄적인 환경으로 인식되었다. 하지만 최근에는 OBD-Ⅱ(On-Board Diagnostic 2) 단자를 통한 연결이나 V2I(Vehicle to Vehicle), V2V(Vehicle to Infrastructure)와 같은 통신기술이 발전함에 따라서 차량 내부 네트워크는 더 이상 폐쇄적인 환경이 아니라 외부에서 접근이 가능한 환경이 되었다. 실제로 커넥티드 카(connected car) 패러다임을 적용하여 출시된 최신 자동차 모델들은 스마트폰으로 차량 내부 네트워크와 연결하여 각종 기능을 이용할 수 있다. 따라서 이제 차량 내부 네트워크는 더 이상 폐쇄적인 안전한 채널이 아니며, 앞으로는 외부에서의 접근이 더욱 쉬운 개방된 환경이 될 것이다. 이러한 환경에서 차량 내부 네트워크를 이용하여 전송하는 데이터(Data)들은 악의적인 공격자에게 쉽게 노출 될 수 있으며, 악용될 소지가 있다.

자동차에 첨단 IT 기술을 효율적으로 적용하기 위해 다양한 종류의 ECU(Electronic Control Unit)들이 사용되고 있다[1]. ECU의 사용은 1980년대 도입 당시부터 수요가 꾸준히 증가하였으며 최근에 생산되는 고급 차량의 경우 70여개 이상의 ECU들이 탑재되어있다[2]. 차량 내부에 탑재되는 ECU의 개수의 증가로 효율적인 통신을 하기 위해서 CAN(Controller Area Network), LIN(Local Interconnect Network), FlexRay, MOST(Media Oriented Systems Transport)등의 통신프로토콜을 이용하여 서로 통신을 수행한다.[3] 이 중 BOSCH사에서 개발 한 CAN 프로토콜은 엔진이나 브레이크등과 같이 차량의 운행에 큰 영향을 미치는 장치들을 제어하는 ECU 간 통신에 핵심적인 역할을 한다[4]. 이러한 CAN 프로토콜은 국제표준화기구(ISO)와 자동차 엔지니어협회(SAE)에 의해 국제표준화[5,6,7,8]로 등록되어 있지만, 현재 CAN 프로토콜에는 아무런 보안 기법이 적용되어 있지 않다. 이에 따라 최근 여러 연구에서 CAN의 취약점을 이용한 공격 모델이 제시되었으며, 이러한 취약점을 막기 위한 연구들이 진행되고 있지만 아직까지 성과가 미비한 실정이다[1,9]. 자동차가 일상생활 전반에 밀접한 연관을 맺고 있고, 자동차의 운행이 차량 내부의 ECU와 같은 전자 부품들에 의해 수행되기 때문에, 이러한 암호 및 인증 시스템의 부재는 차량 이용자에게 매우 치명적인 위협이 될 수 있다. 그러므로 이제는 차량 내부 네트워크 또한 일반적인 통신환경과 마찬가지로 데이터의 암호화가 필요하고 암호화된 데이터를 서로 이용하기 위해서는 데이터를 송·수신하는 기기간의 인증이 선행 되어야 한다.

일반적인 인증기법의 경우, 대칭키 방식 암호화 방식 혹은 비대칭키 암호화 방식 또는 해시 함수를 이용하여 인증기법을 설계한다. 이 때, 수학적인 어려움에 기반 하는 비대칭키 암호화 방식의 경우 연산량이 매우 큰 단점이 존재하고, 해시 함수 또한 마찬가지의 문제가 발생한다. 대칭키 암호화 방식을 이용한 인증기법의 경우 앞의 두 가지 기법보다는 연산량의 측면에서 효율적이지만, 차량 내부 네트워크는 매우 제약적인 내부 저장 공간과 연산 능력을 가진 기기들로 구성되어 있으며, 한 번에 전송할 수 있는 데이터의 크기도 매우 작다. 따라서 이과 같은 환경에서는 일반적인 인증기법을 사용 할 수 없으며, 매우 경량화 된 인증기법을 사용해야 한다. 하지만 현재까지 차량 내부 네트워크에 적용하기 위한 인증기법에 대한 연구는 뚜렷한 결과가 나타나지 않았다.

RFID는 라디오 주파수를 이용하여 크게 리더(Reader)와 태그(Tag)의 양방향 통신으로 구성되어 있으며, 이 중 수동형 태그는 전력 장치가 존재하지 않기 때문에 기능적으로 매우 제한되어 있다. 이러한 RFID의 제한적인 환경은 차량 내부 네트워크와 유사한 제한 조건이며 이러한 조건으로 인하여 큰 연산량을 처리하고 많은 수의 게이트를 필요로 하는 블록암호나 해시 함수 등의 사용을 어렵게 한다. 이로 인하여 수동형 태그를 사용하는 RFID 환경에서는 매우 경량화 된 인증프로토콜에 대한 연구가 진행되고 있다. 2001년 Hopper와 Blum은 LPN문제의 어려움을 이용하여 경량 휴먼 인증 프로토콜인 HB[10]를 제안하였으며, 2005년 Juels 와 Weis는 RFID 태그와 인간의 연산능력의 유사성을 이용하여 HB를 RFID 시스템에 적용하고, 능동 공격에 대하여 안전한 HB+[11]를 제안하였다. 이후로 HB+를 개선하여 다양한 인증 프로토콜들이 발표되었으며, 이렇게

HB 타입의 구조를 가진 프로토콜들을 HB-family[10,11,12] 기법이라 한다. 이러한 HB-family 기법의 프로토콜은 bit 단위의 AND, OR, XOR과 같이 매우 효율적인 논리 연산을 이용하여 구현되었으며, 해당 프로토콜의 안전성을 보장하기 위해서 LPN(Learning Parity with Noise)문제를 기반으로 하여 설계되었다.

본 논문에서는 CAN 프로토콜에서 기기간의 인증을 위한 인증기법을 사용하기 위해 차량 내부 네트워크와 유사한 제약사항이 있는 RFID 환경의 경량인증 프로토콜인 HB-Family 기법을 차량 내부 네트워크에 적용하는 방법을 제안한다. 차량 내부 네트워크는 내부 저장 공간이 작고, 약한 연산능력을 가진 제한적인 환경을 가지고 있다는 점에서 RFID 환경과 공통점이 있다. 하지만 CAN 통신 환경에서는 RFID 환경보다 빠른 최대 1Mbps의 속도로 통신이 가능하기 때문에 보다 더욱 제약적인 RFID 환경에서 잘 적용한 경량 인증 프로토콜을 차량 내부 네트워크 환경으로 이식할 경우 뛰어난 성능을 보여줄 것으로 예상할 수 있다. 이에 따라 본 논문에서는 HB-Family 기법을 차량 내부 네트워크의 특수한 환경에 맞게 효율적으로 변형시키는 방법을 크게 두 가지로 제안하며, 실제 ECU를 이용한 구현을 통한 실험을 하여 제안 방법을 적용시킨 HB-Family 기법의 수행 결과를 분석한다.

본 논문의 구성은 다음과 같다. 2장에서는 기법을 적용하기 위한 환경인 CAN에 대한 소개, HB-Family 기법에 대한 배경 지식, 공격모델과 관련연구를 소개하고, 3장에서는 HB-Family 경량인증 기법을 CAN에 적용하기 위한 방법을 제안한다. 4장에서는 성능 평가를 위한 실험 환경과 실험한 결과, 그리고 이에 따른 분석 결과를 제시하고, 5장에서는 성능 평가 결과를 토대로 결론을 서술한다.

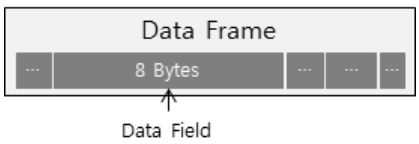
II. 배경 지식

본 장에서는 제안 하는 방법의 배경 지식이 되는 차량 내부 환경의 CAN 프로토콜과 RFID 환경의 HB-Family 경량인증 기법에 대하여 설명한다.

2.1 CAN

초기에 자동차는 제동 및 가속 등이 모두 물리적인 페달이나 체인 등으로 제어할 수 있었다. 현재 산업의 발전과 기술의 발전에 편승하여 차량 또한 1970년대 이후 고성능화, 지능화와 함께 차량 내부에서 차량의 운행에 영향을 주는 많은 수의 ECU가 탑재되었다. 이러한 ECU를 제어하기 위해 다양한 네트워크 통신 방법들이 존재하고, 특히 차량 내부 네트워크는 실시간 제어능력이 중요시되기 때문에 높은 신뢰성을 갖는 시리얼 버스 시스템(serial bus system)인 CAN은 여러 가지 차량 제어 프로토콜 중 사실상 표준으로 사용되고 있다[13,14]. CAN 통신은 단일(single) 또는 이중(dual) 트위스트 쌍(twist pair)으로 연결되어 있고, 메시지 전송 시 브로드캐스트(broadcast) 방식을 사용하며 이에 따라 하나의 ECU의 메시지를 모든 ECU들이 받아 볼 수 있다. CAN 통신을 이용하여 메시지 전송을 하기 위해서는 데이터 프레임, 리모트 프레임, 에러 프레임, 오버로드 프레임의 총 4개의 프레임이 사용된다. 실질적인 데이터는 데이터 프레임을 이용하여 전송된다. 하지만 [그림 1]에서 보는 것과 같이 CAN 통신 시 데이터 필드에는 최대8byte의 메시지를 담을 수 있기 때문에 더 큰 크기의 메시지를 전송하기 위해서는 해당 메시지를 8byte단위로 나누어 전송할 필요가 있다.

이러한 CAN 통신은 여러 가지 취약점이 존재한다. K. Koscher 등은 CAN의 취약점을 분석하였으며[1]. 분석결과에 따르면 CAN 통신은 데이터 프레임 내부 구성의 문제점으로 인한 취약점을 갖고 있다. 데이터 프레임을 구성하는 총 8개의 필드에는 메시지를 송신하는 ECU를 인증할 수 있는 필드가 존재 하지 않는다. 기존에 CAN에 대한 공격 방법과 실제 차량으로 CAN의 취약점을 이용한 실험이 진행되었으며[1,9] 대표적으로 차량의 특정 ECU를 이용하여 차량의 동작을 수행하는 실험 등이 진행되었다.



[그림 1] CAN 데이터 프레임과 데이터 필드

2.2 HB-Family 경량인증 기법

2.2.1 N.J.Hopper와 M.Blum의 프로토콜(HB)

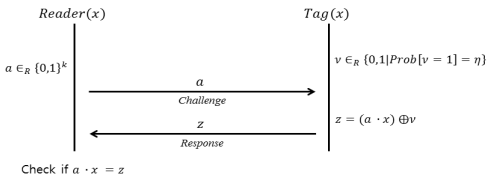
HB의 인증 과정은 아래와 같이 2단계로 구성된 라운드를 r번 진행한다[그림 2].

- *Challenge* 단계 :

리더는 k길이를 가지는 a를 랜덤하게 생성하여 태그에게 전송한다.

- *Response* 단계 :

태그는 바이너리 내적 연산과 XOR 연산을 이용하여 $z=(a \bullet x)\oplus\nu$를 계산한 뒤, 리더에게 전송한다.



〔그림 2〕 HB인증 프로토콜

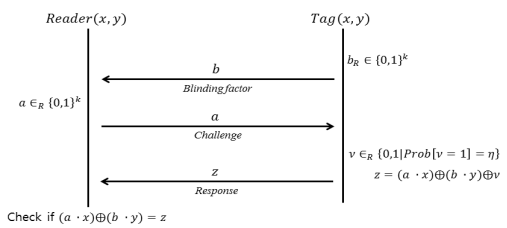
검증 : 한 라운드가 진행될 때마다 리더는 z가 $z=a \bullet x$을 만족하는지 확인한다. r라운드를 수행한 후, 정당하지 않은 응답의 수가 임계값 ηr이하이면 정당한 태그로 인정한다.

HB에서는 서로 다른 t개의 질의 값을 각각 여러 번 보내 노이즈가 포함되지 않은 정당한 응답 값을 t개 얻은 후, 가우시안 소거법을 이용하면 비밀 값 x를 쉽게 계산 할 수 있다.

2.2.2 A.Juels와 S.A.Weis의 프로토콜(HB+)

HB+의 인증 과정은 아래와 같이 3단계로 구성된 라운드를 r번 진행한다〔그림 3〕.

- *Blinding* 단계 : 태그는 k길이를 가지는 b를 랜덤하게 생성하여 리더에게 전송한다.
- *Challenge* 단계 : 리더는 k길이를 가지는 a를 랜덤하게 생성하여 태그에게 전송한다.
- *Response* 단계 : 태그는 바이너리 내적 연산과 XOR 연산을 이용하여 $z=(a \bullet x)\oplus(b \bullet y)\oplus\nu$를 계산한 뒤, 리더에게 전송한다.



〔그림 3〕 HB+인증 프로토콜

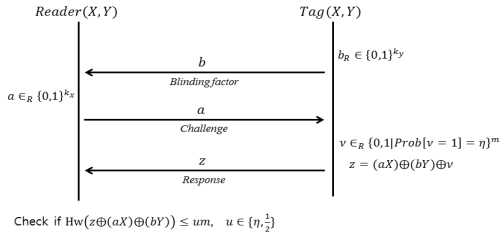
검증 : 한 라운드가 진행될 때마다 리더는 z가 $z=(a \bullet x)\oplus(b \bullet y)$을 만족하는지 확인한다. r라운드를 수행한 후, 정당하지 않은 응답의 수가 임계값 ηr이하이면 정당한 태그로 인정한다.

HB+에서는 HB와 달리 *Blinging* 값 b로 인하여 같은 질의를 여러 번 반복하여도 노이즈가 포함되지 않은 정당한 값을 구할 수 없으므로 행렬식을 구성할 수 없다. 하지만 HB+는 중간자 공격의 한 종류인 GRS-중간자 공격에 취약함이 밝혀졌다〔15〕.

2.2.3 H.Filbert 등의 프로토콜(HB#)

HB#〔12〕의 인증 과정은 아래와 같이 3단계로 구성된 라운드를 1번 진행한다〔그림 4〕.

- *Blinding* 단계 : 태그는 k_y길이를 가지는 b를 랜덤하게 생성하여 리더에게 전송한다.
- *Challenge* 단계 : 리더는 k_x길이를 가지는 a를 랜덤하게 생성하여 태그에게 전송한다.
- *Response* 단계 : 태그는 $z=(aX)\oplus(bY)\oplus\nu$를 계산한 뒤, 리더에게 전송한다. 여기서 ν는 m bit 길이의 노이즈 벡터이다.



〔그림 4〕 HB#인증 프로토콜

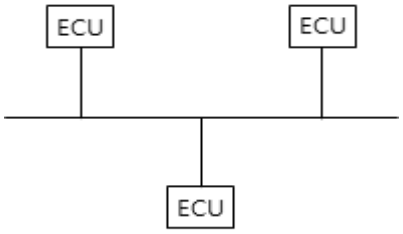
검증 : 리더는 0과 $\frac{1}{2}$ 사이에서 η를 선택하여 임계값을 um으로 설정한 뒤, $z\oplus aX\oplus bY$의 해밍 웨이트(hamming weight) 값이 임계값보다 작으면 정당한 태그로 인증한다.

HB#에서는 GRS-중간자 공격으로 a의 단일 bit를 변경하여도 비밀 값의 여러 bit에 영향을 주기 때문에 결과적으로 비밀 값에 대한 정보를 얻을 수 없다. 또한〔16〕에서 구체적인 임계파라미터 범위를 제시하지 않은 반면 HB#에서는 오거부율을 고려한 구

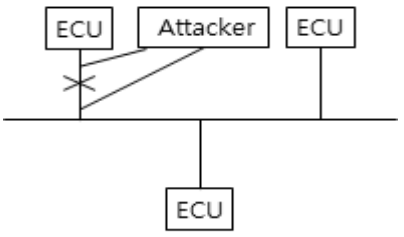
체적인 임계값 um을 설정하였다. 하지만 여전히 중간자 공격에 취약함이 밝혀졌다[17].

2.3 HB-Family에 대한 공격 모델

인증 프로토콜의 공격 유형은 수동 공격, 능동 공격, GRS-중간자 공격, 중간자 공격의 총 4가지로 구분된다[11,12,15,16]. 수동 공격이 공격자의 능력이 가장 약한 공격 형태이며, 중간자 공격이 공격자의 능력이 가장 강한 공격이다. 차량 내부 CAN은 [그림 5]와 같은 유선 네트워크 환경으로 구축된다. 공격자가 중간자 공격을 시도하기 위해서는 ECU간 연결되어있는 통신 회선을 절단하여 [그림 6]과 같이 네트워크 회선을 재구성 하여 물리적으로 ECU간의 통신을 방해해야만 가능하다. 하지만 이와 관련된 연구[18]에 의하면 차량 내부에 물리적인 공격을 하는 것은 공격자가 우선 차량을 점유하였다는 매우 강력한 가정이 필요하며, 논리적인 공격에 비하여 실현 가능성과 효율성이 매우 떨어지므로 고려하지 않는다. 또한 2.1절에서 설명한 것과 같이 CAN 통신은 브로드캐스트방식을 사용하기 때문에 중간자공격을 하기 위한 메시지 드롭(message drop)이 불가능 하다. 이에 따라서 중간자 공격의 일종인 GRS-중간자 공격과 중간자 공격은 차량 내부 CAN에서는 수행될 수 없다. 따라서 해당 환경에서 쉽게 수행 가능한 인증 프로토콜에 대한 공격 유형은 수동 공격과 능동 공격으로 2가지가 존재한다. 2.2절에서 소개한 것과 같이 HB는 수동공격에서는 안전하지만 능동공격에서는 안전하지 않다. 하지만 HB+, HB#은 최소 능동공격에 안전한 기법들이다[12,16,17]. 추가적으로 본 논문에서는 차량 내부 CAN에 대한 DDoS 공격은 고려하지 않는다.



[그림 5] CAN 통신상의 ECU간의 연결 형태



[그림 6] CAN 통신에서 중간자 공격을 하기 위한 공격 모습

2.4 관련 연구

T. Hoppe등의 연구[19]에 따르면 차량 내부 네트워크와 차량 내부와 외부 기기간의 통신 네트워크를 포함하는 차량 네트워크(vehicular network)의 안정성을 위해서는 크게 기기인증, 메시지 암호화 및 무결성 보장, 침입탐지 시스템의 세 가지 단계가 필요하다. 메시지 인증과 같이 메시지의 무결성을 보장하기 위한 연구는 D. K. Nilsson등의 연구[20]와 같이 진행되었다. 해당 연구에서 제안하는 DDA(Delayed Data Authentication)기법은 메시지 재전송공격을 막기 위해 MAC(Message Authentication Code)을 사용한다. 이때 CAN 메시지 구조에서 가용할 수 있는 영역이 부족함을 지적하고 CRC필드를 MAC필드로 대체하는 기법을 제안하였지만 DDA기법은 전송된 4개의 메시지 별로 64bit MAC을 생성하고 16bit로 분할 한 후 다음번에 전송하게 될 4개의 CAN 메시지에 적재하여 이전에 전송한 4개의 메시지의 MAC값의 인증작업을 마치기 때문에 상당히 긴 지연시간이 발생한다. 보통 수 ms이내로 전송 받은 CAN 메시지에 대한 처리가 필요한 자동차 내부 네트워크에서 DDA기법을 사용할 경우 최소 80ms이상의 인증지연 시간이 발생하기 때문에 자동차환경에 DDA기법을 적용하는 것은 사실상 불가능하다. 일반적으로 자동차 내부 CAN 버스시스템에서 통신에 참여한 ECU들 중 데이터 송신 주기가 가장 짧은 경우는 10ms마다 데이터를 송신하는 경우이며, 이를 바탕으로 DDA기법을 사용할 경우 최소 80ms이상의 인증지연 시간이 발생하기 때문이다. 위에서 언급한 것과 같이 DDA기법은 가용성적인 측면에서 문제가 있기 때문에 차량 내부 ECU간의 기기인증에 사용하는데 무리가 있다. 이처럼 차량 내부 네트워크를 이용하는 메시지인증에 대한 연구는 진행되고 있지만, 현재 차량 내부 ECU간 혹은 내부 ECU와 외부 CE간의 기기인증에 관한 연구는 진행되고 있지 않다. 이에 본 논문은 vehicular network 의 첫 번째 단계인 기기인증을 하기 위해서 기존의 경량인증 기법인 HB-Family의 사용을 제안하면서, 차량 내부 CAN

의 데이터 프레임을 효율적으로 사용하는 방법을 제시하여 가용성 측면에서 효율적으로 기기인증을 할 수 있는 방법을 제안한다.

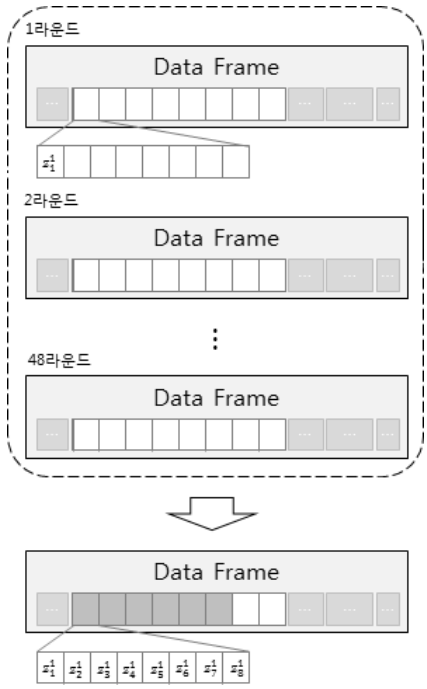
III. 제안하는 적용방법

본 장에서는 차량 내부 네트워크의 ECU간 인증이 없는 취약성을 이용한 공격을 막기 위해서 차량 내부 네트워크에 HB-Family 경량인증기법을 환경에 적절하게 적용하는 방법을 제안한다.

실제로 ECU간 CAN 통신을 수행할 때 데이터를 전송할 수 있는 데이터 필드는 8byte이다. 하지만 HB-Family 경량인증기법의 *Blinding* 단계에서 생성하는 b와 *Challenge* 단계에서 생성하는 a는 8byte로 나누어지지 않으므로 마지막 전송 단계에서 데이터 필드에 빈 공간이 생기게 된다. 또한 *Response* 단계에서 계산하는 z는 8byte 크기 중 극히 일부만 사용하여 전송하므로 실제 차량 내부 네트워크에 HB-Family 기법을 그대로 적용시키는 것은 비효율 적이다. 이에 따라 HB-Family 기법을 차량 내부 네트워크에 맞게 효율적으로 개선하고, ECU간의 인증 속도를 증가시키기기 위해서 다음과 같이 크게 두 가지 방법을 제안한다.

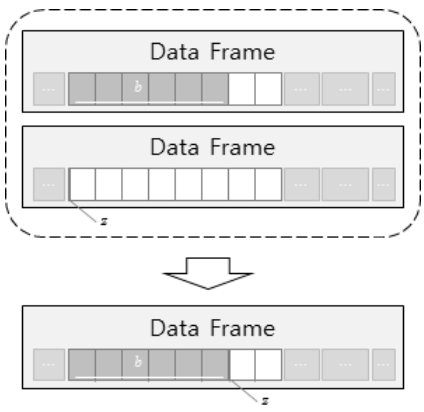
먼저 각각의 ECU의 역할에 따라 구분을 하기 위해서 RFID 환경의 리더와 같이 검증자 역할을 하는 ECU를 ECU_V, 태그와 같이 증명자 역할을 하는 ECU를 ECU_P로 표시한다.

- 첫 번째 방법〔그림 7〕은 여러 라운드를 한 라운드로 줄여서 동시에 수행하는 방법이다. 이 방법을 이용하면 마지막에 ECU_P가 ECU_V에게 보내는 z값의 전송 횟수를 줄일 수 있다. 특히 HB와 HB+는 기기 간 인증을 하기 위해서 인증 기법을 여러 라운드 동안 진행하여야한다. 이 때, ECU_P로부터 ECU_V에게 보내지는 z값의 크기는 1bit이고, 이 1bit크기의 z값을 전송받기 위해 수행하는 라운드 수(r번)만큼의 통신 과정(r번)이 필요하다. 하지만 ECU_P가 매 라운드마다 1bit의 z를 전송하는 것이 아니라 모든 라운드가 다 완료가 된 후에 최종 라운드 수 bit(r bit) 만큼의 z값을 라운드 순서대로 벡터형태로 ECU_V에게 전송하는 방법을 이용하여 데이터 전송 횟수를 줄일 수 있다.



〔그림 7〕 라운드 수를 줄이는 개선 방법 예시

- 두 번째 방법〔그림 8〕은 인증프로토콜에 사용되는 단계를 줄이는 방법이다. HB+와 HB#은 총 3단계로 구성되어있다. 이때, a와 b는 독립적으로 생성되므로 순서가 바뀌어도 무관하므로 *Blinding* 단계와 *Challenge* 단계의 순서를 변경하고, *Blinding* 단계와 *Response* 단계를 동시에 수행하여 b와 z를 함께 보내는 방식을 선택하면 총 두 번의 단계로 기법을 수행할 수 있으며, 데이터를 전송하는 횟수를 줄일 수 있다.



〔그림 8〕 수행 단계를 줄이는 개선 방법

첫 번째 방법을 사용하였을 경우, HB와 HB+인 증기법에서 최대 64라운드를 수행하여 얻은 64bit의 z값을 한 번에 전송 할 수 있다. 그리고 두 번째 방법을 사용하였을 경우 데이터 필드의 빈 공간 없이 효율적으로 데이터를 전송할 수 있으며, 3단계 과정을 2단계로 줄여 ECU 통신 간 대기 시간 또한 절약 할 수 있다. 따라서 첫 번째와 두 번째 방법을 적절하게 적용한다면 HB-Family 기법의 연산량은 기존과 동일하지만 전송횟수 및 대기 시간을 줄여서 전체적인 수행속도의 증가를 기대할 수 있다. 제안 방법을 적용하여 개선한 HB-Family 기법의 인증프로토콜은 다음과 같다.

3.1 제안하는 방법을 적용한 HB

HB는 마지막 *Response* 단계의 z값이 1bit이기 때문에 첫 번째 방법을 사용하고, 기법 상 *Blinding* 단계가 없는 총 2단계로 구성되어 있기 때문에 두 번째 방법은 사용하지 못 한다. 최종적으로 첫 번째 방법을 사용하여 개선시킨 HB는 다음과 같다.

- *Challenge* 단계 : ECU_V는 $r\times k$길이를 가지는 bit벡터A를 랜덤하게 생성하여 ECU_P에게 전송한다.
- *Response* 단계 : ECU_P는 바이너리 내적 연산과 XOR 연산을 이용하여 $r$bit의 $z=(Ax)\oplus\nu$를 계산한 뒤, ECU_V에게 전송한다. 여기서 ν는 r bit 길이의 노이즈 벡터이다.

검증 : ECU_V는 0과 $\frac{1}{2}$ 사이에서 η를 선택하여 임계값을 um으로 설정한 뒤, $z\oplus Ax$의 해밍 웨이트(hamming weight) 값이 임계값보다 작으면 정당한 ECU_P로 인증한다.

3.2 제안하는 방법을 적용한 HB+

HB+는 마지막 *Response* 단계의 z값이 1bit이기 때문에 첫 번째 방법을 사용하고, 기법 상 총 3단계로 구성되어 있기 때문에 두 번째 방법도 사용할 수 있다. 최종적으로 두 방법을 사용하여 개선시킨 HB+는 다음과 같다.

- *Challenge* 단계 : ECU_V는 $r\times k$길이를 가지는 bit벡터A를 랜덤하게 생성하여 ECU_P에게 전송한다.
- *Blinding* & *Response* 단계 : ECU_P는 $r\times k$ 길이를 가지는 bit벡터B를 랜덤하게 생성하여 바이너리 내적 연산과 XOR 연산을 이용하여 r bit의 $z=(Ax)\oplus(By)\oplus\nu$를 계산한 뒤, ECU_V에게 전송한다. 여기서 ν는 $r$bit 길이의 노이즈 벡터이다.

검증 : ECU_V는 0과 $\frac{1}{2}$ 사이에서 η를 선택하여 임계값을 um으로 설정한 뒤, $z\oplus Ax\oplus By$의 해밍 웨이트(hamming weight) 값이 임계값보다 작으면 정당한 ECU_P로 인증한다.

3.3 제안하는 방법을 적용한 HB#

HB#의 경우 위의 두 기법과는 다르게 1라운드만으로 인증을 수행할 수 있기 때문에 첫 번째 방법을 이용하여 라운드를 줄일 수 없고, 총 3단계의 인증 과정을 2단계로 줄이는 두 번째 방법을 사용하며, 해당 방법을 적용한 결과는 다음과 같다.

- *Challenge* 단계 : ECU_V는 k_x길이를 가지는 a를 랜덤하게 생성하여 ECU_P에게 전송한다.
- *Blinding* & *Response* 단계 : ECU_P는 k_y길이를 가지는 b를 랜덤하게 생성하여 $z=(aX)\oplus(bY)\oplus\nu$를 계산한 뒤, b와 z를 ECU_V에게 전송한다. 여기서 ν는 $m$bit 길이의 노이즈 벡터이다.

검증 : ECU_V는 0과 $\frac{1}{2}$ 사이에서 η를 선택하여 임계값을 um으로 설정한 뒤, $z\oplus aX\oplus bY$의 해밍 웨이트(hamming weight) 값이 임계값보다 작으면 정당한 ECU_P로 인증한다.

IV. 성능 평가

본 장에서는 차량 내부 네트워크 환경보다 제약적인 RFID환경에 잘 적용되어있는 HB-Family 경량인증기법을 차량 내부 네트워크 환경으로 이식 하여 구현 한 후, 실험 결과를 분석하여 성능평가를 수행한다.

4.1 실험 환경

실험에 앞서서 각 기법의 파라미터들의 값을 결정하기 위해서 H. Gilbert등의 연구 결과[12]를 참고하였다. HB-Family 기법의 안전성은 비밀 값의 bit 수에 의존하며, 80bit 비도의 안전성을 주기 위해서는 HB, HB+에서는 각각 비밀 값의 크기가 224bit가 필요 하다. 이때, HB+는 두 개의 비밀 값(x, y)이 필요하기 때문에 비밀 값의 총 크기는 448bit이다. HB#에서는 행렬을 생성할 수 있는 두 개의 벡터가 필요하고, 각각의 벡터는 224bit와 80bit의 크기이기 때문에 304bit 크기의 비밀 값이 필요하다. 세 가지 기법 모두 오류 발생률(η)는 0.25로 동일하다. 이 값도 위의 연구 결과[12]를 참고하여 설정하였다. 라운드 수는 80으로 설정하였으며, 이는 H. Gilbert등의 연구 결과[12]에 따라 허용가능 한 오거부율(FRR : False Reject Rate) 및 오허용율(FAR : False Accept Rate)를 주기 위함이다. 실험의 정확도를 높이기 위해서 각 기법 별로 100000회 상호인증 과정을 수행 후 평균 수행시간을 분석하였다. 시뮬레이션 수행과 관련된 자세한 설정 값은 다음과 같다.

[표 1] HB-Family 기법의 파라미터 설정 값

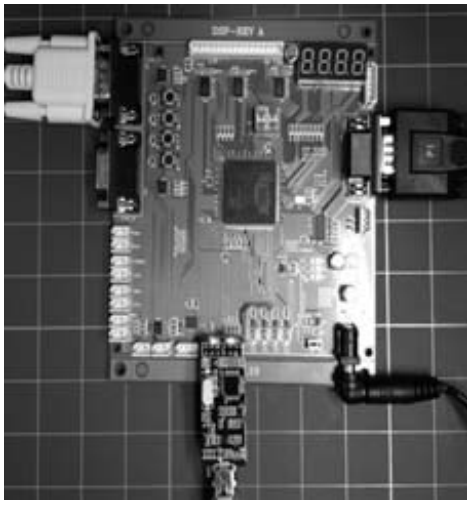
HB	라운드 수(r)	80
	비밀 값(x)의 bit수	224
	오류 발생률(η)	0.25
	전체 수행 횟수	100000
HB+	라운드 수(r)	80
	비밀 값(x, y)의 bit수	448
	오류 발생률(η)	0.25
	전체 수행 횟수	100000
HB#	라운드 수(r)	1
	비밀 값(X, Y)의 bit수	304
	오류 발생률(η)	0.25
	전체 수행 횟수	100000

제안메커니즘의 성능평가를 위해 Texas Instruments사의 F28335 DSP Chip을 탑재한 ECU를 이용하여 제안 메커니즘을 Firmware로 구현한 후 각 인증 기법의 성능 평가를 진행하였다. 사용된 기기 및 환경에 대한 상세한 설정 값은 다음과 같다.

- *컴파일러 :* Code Composer Studio v3.3
- *Micro Controller Unit:* TI F28335 DSP Chip
  CPU Clock Rate - 150 ~ 30 MHz
  Flash - 4Mb(256K×16),
  SRAM - 544Kb(34K×16)

제안 메커니즘의 구현 및 성능평가는 Texas Instruments사의 DSP F28335 Chip 기반의 임베디드 디바이스를 이용하여 수행하였으며, DSP F28335 Chip은 CPU Clock Rate를 30에서 150까지 조절이 가능하다. DSP F28335 Chip의 내부 저장 공간은 256K 크기의 Falsh memory가 16개, 34K 크기의 SRAM이 16개를 가지고 있다. 컴파일러로 사용한 Code Composer Studio v3.3(CCS)는 Texas Instruments사의 임베디드 프로세스들에 대한 통합 개발 환경을 제공한다. CCS는 임베디드 응용 프로그램을 개발하고 디버깅하는데 사용되는 도구들로 구성되어있다.

위의 실험 환경을 토대로 실험은 2장에서 소개한 HB-Family 기법과 3장에서 제안한 방법을 적용하여 차량내부 네트워크에 개선시킨 HB-Family 기법에 대해서 수행하였으며, 위의 구현 환경을 토대로 CPU의 성능을 150MHz, 120MHz, 90MHz, 60MHz의 4가지로 구성하여 실험을 진행하였다. 정확한 결과를 얻기 위해서 100000회 정도 수행하였다.(TI사의 F28335 DSP Chip은 CPU Clock Rate을 150MHz부터 60MHz까지 조절 가능하다.) 실험에 사용한 F28335 DSP Device의 실제 모양은 다음과 같다.



[그림 9] F28335 DSP Device

4.2 실험 결과

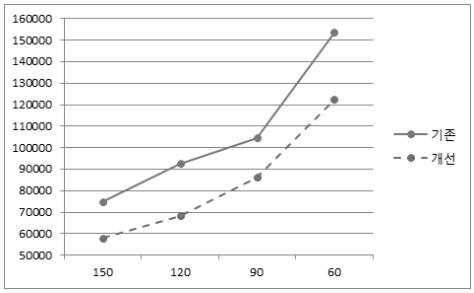
본 실험의 결과는 크게 경량인증 기법의 수행시간과 연산량 그리고 저장량의 세 부분으로 구분하여 실제 실험 결과를 토대로 결과를 서술한다. 아래 표시된 그래프 중 기존으로 표시한 실선 그래프는 2장에서 소개한 HB-Family 기법의 수행 시간을 의미 하며, 점선 그래프는 3장에서 제안한 방법을 적용하여 개선한 HB-Family 기법의 수행 시간을 의미 한다.

4.2.1 수행 시간

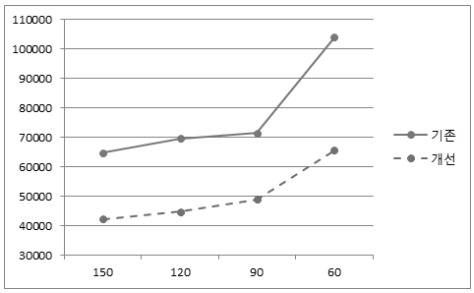
- *HB*

〔표 2〕 ECU_V와 ECU_P의 HB의 수행 시간(μs)

CPU 성능(MHz)		150	120	90	60
ECU_V	기존	74978	92642	104565	153539
	개선	58035	68382	86241	122445
	향상률	22.6%	26.2%	17.5%	20.3%
ECU_P	기존	64840	69645	71536	104004
	개선	42271	44868	48970	65600
	향상률	34.8%	35.6%	31.5%	36.9%



〔그림 10〕 ECU_V의 HB의 수행 시간(μs)

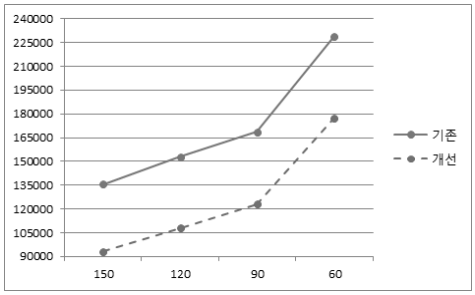


〔그림 11〕 ECU_P의 HB의 수행 시간(μs)

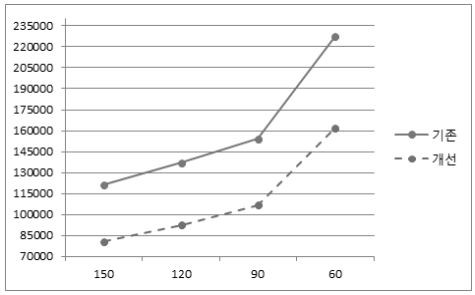
- *HB+*

〔표 3〕 ECU_V와 ECU_P의 HB+의 수행 시간(μs)

CPU 성능(MHz)		150	120	90	60
ECU_V	기존	135614	153049	168627	229134
	개선	93287	107941	123080	177641
	향상률	31.2%	29.5%	27.0%	22.6%
ECU_P	기존	121234	136962	154168	227481
	개선	80742	92345	106879	162150
	향상률	33.4%	32.6%	30.7%	28.7%



〔그림 12〕 ECU_V의 HB+의 수행 시간(μs)

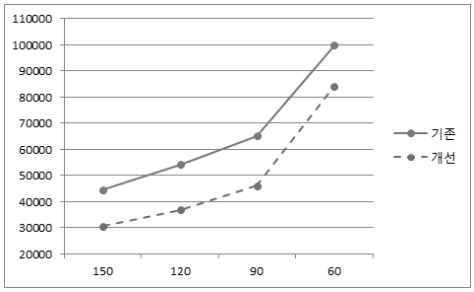


〔그림 13〕 ECU_P의 HB+의 수행 시간(μs)

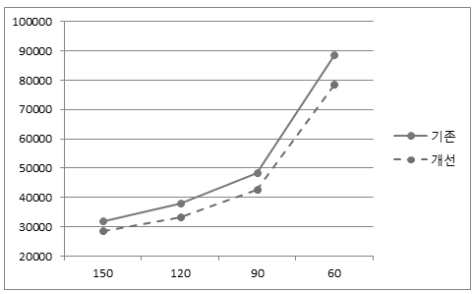
- *HB#*

〔표 4〕 ECU_V와 ECU_P의 HB#의 수행 시간(μs)

CPU 성능(MHz)		150	120	90	60
ECU_V	기존	44467	54175	65117	99828
	개선	30567	36732	46051	84104
	향상률	31.3%	32.2%	29.3%	15.8%
ECU_P	기존	31898	38034	48380	88616
	개선	28503	33295	42591	78509
	향상률	10.6%	12.5%	12.0%	11.4%



〔그림 14〕 ECU_V의 HB#의 수행 시간(μs)



〔그림 15〕 ECU_P의 HB#의 수행 시간(μs)

실험 결과 세 기법과 개선한 기법 모두 CPU성능이 낮아질수록 수행시간이 증가하는 것을 알 수 있으며, 수행시간의 그래프는 단순한 일차함수의 증가형태가 아닌 지수함수 형태로 증가하는 것을 확인 할 수 있다〔그림 10,11,12,13,14,15〕. 또한 ECU_V의 수행시간이 ECU_P의 수행시간보다 오래 걸리는 것을 볼 수 있는데〔표 2,3,4〕 이는 일반적으로 ECU_V가 최종적으로 ECU_P의 진위성 여부를 검증하는 부분이 존재하여 더 많은 연산량이 필요하고, 검증을 위해서 ECU_V가 ECU_P의 응답을 기다려야하기 때문이다.

〔표 5〕 기존의 HB-Family 기법과 개선한 HB-Family 기법의 파라미터 크기와 전송 횟수

기 법	전송 값	전송 횟수	
		기존 기법	개선한 기법
HB	*Challenge*(a)	320	280
	Response(z)	80	2
HB+	*Challenge*(a)	320	280
	Blinding(b)	320	280
	Response(z)	80	2
HB#	*Challenge*(a)	4	4
	Blinding(b)	4	5
	Response(z)	2	

4.2.2 전송량

기존의 HB-Family 기법과 개선한 HB-Family 기법의 전송량은 차이가 없으며, HB는 총 18000bit, HB+는 총 35920bit HB#은 528bit를 전송한다. 하지만 동일한 크기의 값을 전송하더라도 기존 기법과 개선한 기법 사이의 전송 횟수의 차이가 발생하고, 그 차이는 아래 〔표 5〕에서 확인할 수 있다.

4.2.3 저장량

저장량은 기존의 세 가지 인증기법이 필요한 저장량이 각각 차이가 있으며, 비밀 값의 크기에 영향을 받는 것을 알 수 있다. HB는 비밀 값 x(224bit)와 a(224bit), b(224bit), z(1bit)를 저장할 공간이 필요하며, HB+는 비밀 값 x(224bit), y(224bit)와 a(224bit), b(224bit), z(1bit)를 저장해야 하고, HB#은 두 개의 비밀 값 벡터(224bit, 80bit)와 a(224bit), b(224bit), z(80bit)를 저장해야 한다. 3장의 방법을 이용하여 개선한 경우 기존 기법들보다는 더 많은 저장 공간을 요구 한다. 개선한 HB는 비밀 값 x(224bit)와 A(224×80bit), z(1×80bit)를 저장할 공간이 필요하며, 개선한 HB+는 비밀 값 x(224bit), y(224bit)와 A(224×80bit), B(224×80bit), z(1×80bit)를 저장해야 하고, 개선한 HB#은 기존의 HB#과 동일한 값을 저장해야 한다. 각 기법을 수행하기 위해서 필요한 저장 공간은 다음과 같다〔표 6〕.

〔표 6〕 HB-Family 기법의 필요 저장 공간(bit)

	기존 기법	개선한 기법
HB	449	18000
HB+	673	35920
HB#	832	832

ECU_V와 ECU_P모두 동일한 크기의 저장 공간이 필요하며, HB#, HB+, HB의 순으로 많은 저장량을 요구하는 것을 알 수 있다.

HB+는 HB에 비해 더 많은 비밀 값을 사용하고, 더 많은 값들을 전송하고 전송받기 때문에 기존의 기법과 개선한 기법 모두 위의 〔표 6〕에서 확인 할 수 있듯이 더 많은 저장 공간을 필요로 한다. 기존의 HB#의 경우 여러 라운드를 수행하는 HB+의 기법을 한 번에 수행할 수 있도록 비밀 값의 크기가 커지

고 마지막의 인증여부를 확인하는 z의 크기도 크기 때문에 더 많은 저장 공간을 필요로 한다. 하지만 개선한 기법들을 비교해보면 HB#이 다른 기법들을 개선한 것 보다 더 적은 저장 공간이 필요하다는 것을 알 수 있다.

4.3 기법 분석

4.2절의 내용과 같이 기존의 HB-Family 기법을 사용하는 것 보다 제안하는 두 가지 방법을 사용하여 개선한 HB-Family 기법을 각각의 CPU 성능(150MHz, 120MHz, 90MHz, 60MHz)에서 사용하였을 때 수행시간이 최소 10%에서 최대 36% 단축되었다. 동일한 CPU성능에서 실험할 경우 기존의 HB-Family 기법과 개선한 HB-Family 기법 모두 HB#, HB, HB+ 순으로 수행 시간이 적게 걸리는 것을 확인 할 수 있었으며, 저장량 측면에서도 HB#, HB, HB+ 순으로 많은 저장량을 요구하여 HB#이 다른 기법들과 비교했을 때, 수행속도와 저장 공간 측면에서 더 뛰어나다는 것을 확인 할 수 있었다.

4.4 결과 분석

위의 실험 결과를 실제 차량에 사용가능한지 분석하기 위해서 본 절에서는 가용성과 안전성의 두 가지 측면에서 분석한다.

4.4.1 가용성

실제 차량에서 인증기법의 가용성을 판단하기 위해서는 인증을 수행하는데 필요한 시간과 저장량이 매우 중요한 요소이다. 수행 시간은 각 기기의 전송량 및 전송 횟수와 연산량에 의존한다.

전송량의 경우 실험환경을 설정할 때, HB, HB+, HB# 세 개의 인증 기법의 비밀 값의 bit수를 각각 224bit, 224bit, 224×80bit로 설정하였기 때문에 전송하는 a와 b의 값은 모두 224bit 이고 z의 경우 각각 1bit, 1bit, 224bit로 설정된다. 하지만 CAN에 사용되는 데이터 필드의 크기는 8byte로 64bit이기 때문에 a 혹은 b 또는 z의 값을 전송할 때, 최소 1번에서 최대 4번의 전송이 필요하다. 따라서 이번 실험에서 위의 경량인증 기법의 한 라운드를 수행하기 위해서는 HB와 HB+의 경우 각각 총 5번과 9번의 전송이 필요하고, HB#은 10번의 전송과정이 필요하다. 한번 인증을 하기 위해서 HB와 HB+는 80라운드를 수행해야 하기 때문에 실제로 인증 시에 필요한 전송횟수는 각각 400번과 720번이다. 반면에 HB#의 경우 한 번의 라운드로 인증과정이 끝나기 때문에 단 10번의 전송만으로 한 번의 인증과정을 수행할 수 있다.

연산량의 경우 HB#이 두 개의 비밀 값을 가지는 벡터를 이용하여 하나의 비밀 행렬을 만드는 간단한 연산이 존재하고, HB#에서는 HB+에서 80번 나눠서하는 연산을 한 번에 수행하므로 두 기법 간의 차이는 비밀 값 행렬을 만드는 정도의 연산량 차이만 존재한다. 물론 기법의 형태 상 HB+의 연산량은 HB보다 많으며, 또한 HB#의 연산량이 HB보다 많지만 CAN에서 사용하는 두 개의 ECU를 이용하여 실제 수행시간을 비교해 봤을 때, HB보다 HB#의 수행시간이 더 빠른 것을 알 수 있었다. 따라서 실제 수행 시간에 가장 큰 영향을 미치는 것은 기법 안에 사용된 연산량이 아닌 전송량과 전송 횟수임을 알 수 있다.

3장에서 제안한 방법을 이용하여 개선한 각 기법은 각각 292회, 562회 9회의 전송을 요구 하며, 이는 기존의 기법들이 각각 400회, 720회, 10회의 송·수신 횟수에 비해서 각각 118회, 158회, 1회의 전송 횟수가 줄어든 것임을 알 수 있다. 4.2. 실험 결과의 〔표 2,3,4〕와 〔그림 10,11,12,13,14,15〕에서 알 수 있듯이 줄어든 송·수신 횟수에 의해서 수행 시간이 감소한 것을 확인할 수 있으며 가장 느린 HB+는 60MHz 세팅에서 약 0.2초에 한 라운드를 수행하는 것을 볼 수 있다〔표 3〕. 가장 많은 저장 공간을 필요로 하는 기법 또한 개선한 HB+이고 35920bit의 저장 공간을 요구한다. 이것은 4490byte로 약 4.38Kb의 크기 이다. 하지만 위의 세 기법이 요구하는 저장 공간의 크기는 실험에 사용한 F28334 DSP Chip의 내부 저장 공간으로 충분히 저장 가능했고, 충분히 저장이 가능한 적은 양임을 알 수 있다. 위의 기법에 필요한 저장 공간 외에 추가적으로 저장 공간이 필요 할 수 있지만 실제 ECU를 이용한 실험에서 저장량이 부족한 상황은 발생하지 않았다.

HB-Family 기법은 기본적으로 오류의 발생확률을 이용하여 정당한 기기라고 인정할 만한 오류율을 갖는 경우 기기인증에 성공하는 방식을 취하고 있다. 하지만 이는 정당한 기기더라도 확률에 의거하여 오류를 주입하기 때문에 인증에 실패할 수 있는 가능성이 존재한다. 즉 앞에서 정의한 파라미터들의 설정 값에 의한 오거부율은 논문 〔12〕에서 확률적인 방법으로

계산이 되어있다. 위의 설정대로 파라미터의 값을 설정한 경우 정상적인 기기의 인증 실패가 약 44%정도 발생한다[12]. 본 논문에서는 실제로 구현상에서도 10000번 정도의 수행을 하는 동안 세 기법의 오거부율이 논문 [12]의 이론적인 결과인 0.44와 유사한 값을 갖는 것을 확인 할 수 있었고 실제로 구현 중 인증의 성공과 실패 횟수는 다음과 같다.

[표 7] HB - Family 기법의 성공, 실패 횟수와 오거부율

기 법	CPU 성능(MHz)	성 공	실 패	오거부율
HB	150	56813	43187	0.43
	120	55875	44125	0.44
	90	55067	44933	0.45
	60	54575	45425	0.45
HB+	150	55102	44898	0.45
	120	56141	43859	0.44
	90	55704	44296	0.44
	60	55228	44772	0.45
HB#	150	54990	45010	0.45
	120	56435	43565	0.44
	90	55914	44086	0.44
	60	55339	44661	0.45

오거부율이 0.44로 나타난다는 것은 실제로 정상적인 ECU라 하더라도 대략 두 번에 한번정도는 인증에 실패하는 경우가 발생한다는 것이다. 따라서 이러한 오거부율을 생각하면 실제로 정상적인 ECU간의 인증에 걸리는 시간은 한번을 수행할 때 걸리는 시간의 두세배 정도로 늘어 날 수 있다. 하지만, 본 연구에서 수행한 구현 결과에서 알 수 있듯이 위의 기법들의 수행시간이 매우 짧기 때문에 큰 문제가 발생하지 않는다.

그러므로 제안하는 방법을 이용하면 HB-Family 기법에서 가장 비효율적인 HB+도 수행 속도와 저장량적인 측면에서는 충분히 차량 내부 네트워크에 적용하여 사용할 수 있는 가용성이 있다.

4.4.2 안전성

본 연구에서 수행한 실험은 4.1절에서 설명한 것과 같이 80bit 비도의 안전성을 주기 위해서 각 기법의 파라미터들의 값을 H. Gilbert등의 연구 결과[12]를 참고하였다. HB-Family 기법의 안전성은 비밀 값의 bit수에 의존하며, 더 높은 안전성을 제공하기 위해서는 비밀 값의 bit수를 늘리면 된다. 이와는 별개로 HB-Family 기법과 제안한 개선 방법을 적용한 HB-Family 기법 모두 여전히 중간자 공격에 안전하지 않은 취약점이 존재한다. 하지만 2.3절에서 설명한 것과 같이 차량 내부네트워크에서는 중간자 공격이 단순한 메시지 도청과는 달리 CAN 통신방식을 사용하는 차량 내부 ECU간에는 실현 가능성이 매우 낮다. 또한 HB-Family 기법은 한 라운드 내 Challenge 단계와 Blinding 단계에서 사용되는 값들이 서로 독립적으로 생성되므로 이 둘의 전송 순서를 바꾸어도 안전성에 영향을 주지 않으며, 라운드 간 사용하는 값들 또한 독립적이기 때문에 각 라운드의 값을 동시에 전송하여도 안전성에 문제가 발생하지 않는다. 즉, 3장에서 제안한 방법을 사용하여도 안전성에 문제가 발생하지 않는다.

V. 결론

본 연구에서는 차량 내부 네트워크에 기기 간 인증을 하지 않는 취약성을 이용한 공격자의 공격에 대응하기 위해서 차량 내부의 제한된 환경과 유사한 RFID 환경에서 사용할 수 있는 경량 인증 기법인 HB-Family 기법을 적용하기 위한 방법을 제안하였다. 제안하는 방법은 실제 ECU의 제한적인 능력과 CAN 통신의 데이터 전송 크기가 8byte라는 것에 맞춰 기법을 개선하는 방법이며, 이를 구현을 통한 실험 결과를 토대로 적용 방법에 대한 가용성과 안전성을 분석하였다. 분석 결과 차량 내부 네트워크에서의 기기인증은 운전자가 의식하지 못할 정도로 빠른 시간 내에 이루어져야 하는데 제안하는 방법을 적용하면 매우 짧은 시간 안에 인증이 성공한다는 것을 알 수 있다. 또한 ECU간의 통신에서도 잘 수행이 되는 것을 확인함으로써 차량 내부 네트워크의 ECU간 기기인증에 HB-Family 기법을 환경에 맞추어 적용하여 이용할 수 있음을 확인 할 수 있다.

향후 차량 내부 네트워크가 발전함에 따라서 차량 내부 네트워크에 특화되어있는 경량 인증 기법을 개발하는 것은 좋은 연구 과제가 될 것이다. 또한 본 논문에서 이용한 HB-Family 기법의 안전성을 높이기 위해서는 더 큰 크기를 갖는 비밀 값을 설정해야 하고 CAN에 사용하는 데이터 필드의 크기가 8byte로 고정되어있다는 것을 감안하면 수행 시간에 가장 영향을 미치는 전송횟수가 늘어날 수 밖에 없다. 따라서 기기간 전송하는 데이터의 크기와 횟수를 줄이면서 안전성을 유지할 수 있는 방법에 대한 연구가 더욱 필요하다.

참고문헌

[1] K. Koscher, A. Czeskis, F. Roesner, S. Patel, and T. Kohno, "Experimental security analysis of a modern automobile," Proceedings of the 2010 IEEE Symposium on Security and Privacy, pp. 447-462, May 2010.

[2] R. Charette, "This car runs on code," Online:http: //www.spectrum.ieee.org/feb09/7649,Feb. 2009.

[3] T. Nolte, H. Hansson and L.L. Bello, "Automotive communications-past, current and future," in Proceedings of ETFA(Emerging Technologies and Factory Automation), Sep. 2005.

[4] K.H. Johansson, M. Törngren and L. Nielsen, "Vehicle applications of controller area network," D. Hristu-Varsakelis, W.S. Levine (Eds.), Handbook of Networked and Embedded Control Systems, Springer (2005) ISBN: 0-8176-3239-5

[5] "Road vehicles - Diagnostic communication over Controller Area Network (DoCAN) - Part 1: General information and use case definition," ISO 15765-1, Oct. 2011.

[6] "Road vehicles - Diagnostic communication over Controller Area Network (DoCAN) - Part 2: Transport protocol and network layer services," ISO 15765-2, Nov. 2011.

[7] "Road vehicles - Diagnostics on Controller Area Networks (CAN) - Part 3: Implementation of unified diagnostic services (UDS on CAN)," ISO 15765-3, Aug. 2004.

[8] "Road vehicles -- Diagnostic communication over Controller Area Network (DoCAN) -- Part 4: Requirements for emissions-related systems," ISO 15765-4, Feb. 2011.

[9] D.K. Nilsson and U.E. Larson, "Simulated Attacks on CAN Buses: Vehicle virus," Proceedings of the Fifth IASTED Asian Conference on Communication Systems and Networks (ASIACSN), pp. 66-72, Aug. 2008.

[10] N.J. Hopper and M. Blum. "Secure Human Identification Protocols," in C. Boyd (ed.) Advances in Cryptology - ASIACRYPT 2001, LNCS 2248, pp. 52-66, 2001.

[11] S.A. Weis and A. Juels, "Authenticating Pervasive Devices with Human Protocols," In V. Shoup, editor, Advances in Cryptology: Proceedings of CRYPTO 2005, LNCS 3621, pp. 293 - 308. 2005.

[12] H. Gilbert, M.J.B. Robshaw and Y. Seurin, "HB#: Increasing the security and efficiency of HB+," In: Smart, N.P. (ed.) EUROCRYPT 2008, LNCS 4965, pp. 361 - 378, 2008.

[13] Sato Michicho, "자동차 네트워크 시스템," 성인당, Jan. 2010.

[14] 김강석, "CAN 통신 도청 및 조작을 통한 차량 ECU의 외부위협 가능성 분석," 석사학위논문, 고려대학교, 2011년 2월.

[15] H. Gilbert, M.J.B. Robshaw, and H. Sibert, "An Active Attack Against HB+: A Provably Secure Lightweight Authentication Protocol," IEEE Electronics Letters, vol. 41, no. 21, pp. 1169-1170, Oct. 2005.

[16] J. Katz and J. Shin, "Parallel and Concurrent Security of the HB and HB+ Protocols," Eurocrypt 2006, LNCS 4004, pp. 73-87, 2006.

[17] K. Ouafi, R. Overbeck and S. Vaudenay, "On the Security of HB# against a Man-in-the-Middle Attack," Asiacrypt 2008, LNCS 5350, pp. 108-204, 2008.

[18] S. Checkoway, D. McCoy, D. Anderson, B. Kantor, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner and T. Kohno. "Comprehensive experimental analyses of automotive attack surfaces,"

In D. Wagner, ed., Proceedings of USENIX Security 2011, USENIX, Aug. 2011.

〔19〕 T. Hoppe, S. Kiltz and J. Dittmann "Security threats to automotive CAN networks - Practical examples and selected short-term countermeasures," Reliability Engineering & System Safety, Accepted Manuscript, Available online 5 Jul. 2010, in press.

〔20〕 D.K. Nilsson, U.E. Larson, E. Jonsson, "Efficient In-Vehicle Delayed Data Authentication based on Compound Message Authentication Codes," In: Proceedings of the IEEE 68th Vehicular Technology Conference (VTC2008-Fall), Sep. 2008.

〔21〕 S. Ravi, A. Raghunathan, P. Kocher, and S. Hattangady, "Security in embedded systems: Design challenges," ACM Transactions on Embedded Computing Systems. vol. 3, no. 3, pp. 461 - 491, Aug. 2004.

〔22〕 E.R. Berlekamp, R.J. EcEliece and H.C.A van Tilborg, "On the Inherent Intractability of Certain Coding Problems," Information Theory, vol. 24, no. 3, pp. 384-386. IEEE Transactions, May 1978.

〔23〕 J. Hastad, "Some Optimal Inapproximability Results," J. ACM. vol. 48, no. 4, pp. 798-859, Jul. 2001.

〔24〕 H. Gilbert, "Techniques for Low Cost Authentication and Message Authentication," CARDIS 2000, LNCS 1820, pp. 183-192, 2000.

〔25〕 조아람, 조효진, 우사무엘, 손영동, 이동훈, "CAN 버스 공격에 안전한 메시지 인증 및 키 분배 메커니즘," 정보보호학회논문지, 22(5), pp. 1057-1068, 2012년 10월.

〔26〕 김태수, 김효승, 이동훈, "HB기반 경량인증 기법 증명 모델에 대한 연구," 한국정보보호학회 동계학술대회발표집, pp. 52-55, 2012년 12월.

〔27〕 S. Piramuthu, "HB and Related Lightweight Authentication Protocols for Secure RFID Tag/Reader Authentication," Proceedings of the CollECTeR Europe Conference, Jun. 2006.

〔28〕 J. Katz, J.S. Shin and A. Smith "Parallel and Concurrent Security of the HB and HB+ Protocols," In: Vaudenay, S. (ed.) EUROCRYPT 2006, LNCS 4004, pp. 73 - 87, 2006.

〔29〕 J. Bringer, H, Chabanne and E. Dottax, "HB++: a Lightweight Authentication Protocol Secure against Some Attacks," Proceedings of the IEEE Int'l Conference, Pervasive Service, Workshop Security, Privacy and Trust in Pervasive and Ubiquitous Computing, pp. 28-33, Jun. 2006.

〔30〕 P. Rizomiliotis and S. Gritzalis, "GHB# : A Provably Secure HB-Like Lightweight Authentication Protocol," Proceedings of the 10th international conference on Applied Cryptography and Network Security, LNCS 7341, pp. 489-506, 2012.

〈저자소개〉

김 태 수 (Tae Su Kim) 학생회원
2012년 2월: 서울시립대학교 수학과 학사 졸업
2012년 3월~현재: 고려대학교 정보보호대학원 석사과정
〈관심분야〉 암호프로토콜, 경량인증 프로토콜, IN-Vehicle Security

김 효 승 (Hyoseung Kim) 학생회원
2010년 2월: 고려대학교 수학과 학사 졸업
2010년 3월~현재: 고려대학교 정보보호대학원 석·박사 통합과정
〈관심분야〉 정보보호이론, 암호 프로토콜, 경량인증 프로토콜, 준동형암호

우 사 무 엘 (Samuel Woo) 학생회원
2006년: 단국대학교 컴퓨터과학과 학사 졸업
2006년: (주)EOTECHNICS 근무
2010년: 단국대학교 전자계산학 석사
2011년~현재: 고려대학교 정보보호대학원 박사과정
〈관심분야〉 무선 네트워크 보안, IN-Vehicle Security

이 동 훈 (Dong Hoon Lee) 종신회원
1983년 8월: 고려대학교 경제학사 졸업
1987년 12월: Oklahoma University 전산학과 석사 졸업
1992년 5월: Oklahoma University 전산학과 박사 졸업
1993년 3월~1997년 2월: 고려대학교 전산학과 조교수
1997년 3월~2001년 2월: 고려대학교 전산학과 부교수
2001년 3월~현재: 고려대학교 정보보호대학원 교수
〈관심분야〉 정보보호이론, 암호 프로토콜, USN, 키 교환, 익명성 연구, PET기술