

Implementation of Vehicle ECU Firmware Over The Air (OTA) Update Performance Reliability Evaluation Technology

Hyoduck Seo[†]/Researcher · Soosung Lee/Researcher · Ohhun Kwon/Researcher · Oeon Kwon/Researcher · Sungho Kim/Researcher · Jangkyu Yun/Researcher
Dept. of Research Development, Gyeongbuk Institute of IT Convergence Industry Technology (GITC)

차량용 ECU 펌웨어 무선 업데이트 성능 신뢰성 평가 기술 구현

서효덕[†]/연구원 · 이수성/연구원 · 권오훈/연구원 · 권오연/연구원 · 김성호/연구원 · 윤장규/연구원
(재)경북IT융합산업기술원 연구개발부

Abstract

This paper proposes a method for implementing a reliability evaluation technology for Over The Air (OTA) update performance of ECU firmware for vehicles. OTA refers to self-updating various tasks such as improving, adding, and correcting errors in a wireless environment. In addition, the OTA method is delivered remotely from a cloud-based server to a vehicle through a cellular or Wi-Fi connection. It is a similar process to updating a smart phone or laptop. However, OTA technology is specialized in smart devices such as smart phones, so performance reliability and authorized verification for vehicle application are insufficient. Therefore, in this paper, we propose a OTA update performance reliability evaluation and method to present standardization for improving reliability of OTA update performance for vehicles and performance inspection of OTA update technology for vehicles. In addition, it is expected that the proposed technology can be used to serve as a catalyst for the dissemination and activation of OTA update technology for vehicles.

Keywords : Over The Air, Firmware, Vehicle ECU, Reliability

1. 서 론

1.1 무선 업데이트 (Over The Air)

Over The Air (OTA)란, 무선통신망에서 필요한 데이터를 무선 채널을 통해 해당 기기로 전송하여, 기기

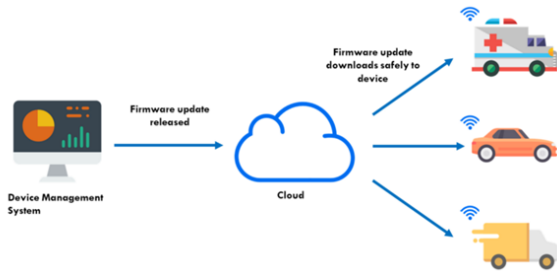
[†] To whom correspondence should be addressed.

Corresponding Author: hdseo@gitc.or.kr

©2022 The Korean Institute of Plant Engineering

Received 31 October 2022; Revised 4 November 2022; Accepted 11 November 2022

의 환경 설정, 소프트웨어 (Software) 등을 갱신하는 방법으로 정의 할 수 있다. OTA는 원격의 한 송신기에서 해당되는 기기 전부에 데이터를 전송하면, 각 기기에서 수신된 데이터를 사용하여 필요한 과정을 수행한다. OTA로 인해 휴대폰, 셋톱박스, 와이파이 (Wi-Fi) 기기, 기지국, 자동차 등 사용자 기기를 개별적으로 관리하지 않아도 되며, 실시간으로 새로운 버전의 소프트웨어 (Software), 펌웨어 (Firmware) 등을 갱신시켜 최신 상태로 유지 할 수 있다[1].



[Figure 1] Over The Air (OTA) Concept

[Figure 1]은 OTA의 개념을 도식화 한 것이다. 앞서 이야기 했듯이, OTA는 새로운 소프트웨어 (Software), 펌웨어 (Firmware), 설정 등을 휴대폰, 셋톱박스 등의 장치에 무선으로 배포하기 위한 방식이다[2]. OTA 기술은 공간적 시간적인 제약 없이 필요할 때 언제든지 소프트웨어 (Software)나 펌웨어 (Firmware)의 기능 및 오류를 개선할 수 있는 장점이 있다. 그리고 OTA는 업데이트 방식에 따라 종류를 나눌 수 있다[3].

<Table 1> Type of Over The Air (OTA)

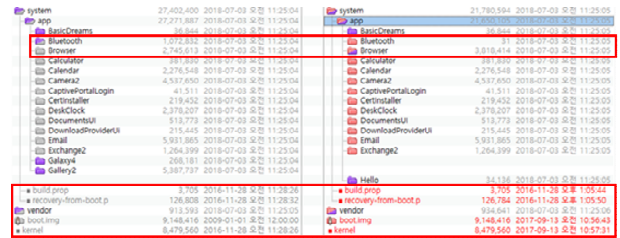
	Full updates	Incremental updates
Method	How to reinstall the whole with the new firmware	How to download and update only changes in content
Notice	The device downloads the entire firmware and provides space for storage	Additional programs are needed to create changes and use them to update to new versions

<Table 1>은 OTA의 종류에 따른 방식과 유의사항

에 나타내고 있다. OTA는 업데이트 방식에 따라 Full updates와 Incremental updates로 구분할 수 있다[4].

Full updates는 디바이스 이전 상태에 관련해서, 새로운 펌웨어 (Firmware)로 전체를 재설치 하는 방식이다. 따라서 디바이스는 전체 펌웨어 (Firmware)를 다운로드하여 저장할 수 있는 공간이 필요하다[5].

Incremental updates는 기존 펌웨어 (Firmware)와 새로운 펌웨어 (Firmware)와의 변경점만 다운로드하여 업데이트하는 방식이다. Full updates 방식에 비해 작은 저장 공간을 사용하는 장점이 있으나, 변경점만 다운로드하여 업데이트하는 방식이기 때문에 변경점을 생성하고, 이를 이용하여 새로운 버전으로 업데이트하는 프로그램들이 추가적으로 필요하다[6].



: Changes made to conventional Firmware and new Firmware

[Figure 2] Creating changes for Incremental update

[Figure 2]는 Incremental updates를 위한 변경점 생성을 나타낸 것이다. 기존 펌웨어 (Firmware)와 새로운 펌웨어 (Firmware) 변경점을 추출하여 기존 펌웨어 (Firmware)의 파일 추가, 삭제, 속성 변경 등을 적용하여 OTA를 진행하여 기존 펌웨어 (Firmware)에서 새로운 펌웨어 (Firmware)로 적용된다. Full updates에 비해 소요시간이나 사용하는 데이터 용량은 작지만 부분 변경이 가능하도록 변경점 생성 및 관리 프로그램이 필요하다[7]. <Table 1>과 같이. OTA는 방식에 따라 특징이 달라지며, 적용하는 디바이스나 환경에 따라 유연하게 적용이 가능하며, 무선통신이 가능한 지역에서는 시간이나 환경에 구애 받지 않고 관리 및 업데이트 가능한 장점이 있다.

1.2 차량용 ECU 펌웨어 무선 업데이트

차량의 기능 및 ECU 펌웨어 (Firmware)의 성능 및 오류 개선과 업그레이드를 위해서는 기존에는 서비스 센터나 정비소에서 시간과 날짜를 예약하고 직접 찾아가서 성능 및 오류 개선 그리고 업데이트들을 진행하였다. 하지만 기존 방식은 운전자의 시간적인 손실과 공간적인 제약을 주는 단점이 있다. 또한, 운전자가 사용하지 않는 기능이나 개선 및 업데이트 내역에 대해서 전달 받지 못하는 경우, 최신버전의 펌웨어나 기능을 활용할 수 없는 문제점도 존재한다[8].

이와 같이, 기존 차량 기능 및 ECU 펌웨어 (Firmware) 업데이트 방식은 시간 및 공간의 제약이 있으며, 업데이트 내용을 알 수가 없으면 활용할 수 없는 단점으로 차량 업데이트에 대한 접근성을 낮추고 있다. 하지만 차량용 무선 업데이트 기술은 기존 업데이트 방식의 문제점을 해결할 수 있다. 그 이유는 차량 제조사에서 개선된 소프트웨어 (Software)나 펌웨어 (Firmware)를 제공하면, 운전자가 별다른 조치 없이 업데이트 여부를 알림 받고 무선 업데이트가 바로 진행되어, 기존 방식의 시간과 날짜를 예약하고 직접 찾아가는 경우가 없어지고, 무선 업데이트가 가능한 환경에서 언제 어디서든지 업데이트가 진행된다. 그래서 차량용 무선 업데이트 기술은 편리성으로 인해 제조사뿐만 아니라, 많은 운전자들도 무선 업데이트 방식을 선호하고 있다[9].

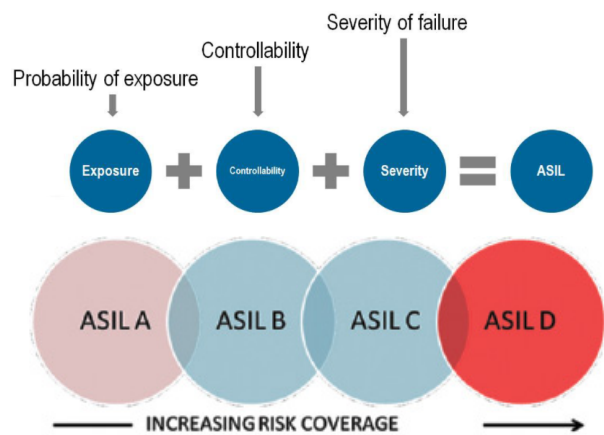
하지만 차량용 무선 업데이트 기술은 제조사와 많은 운전자들이 선호하지만 기술 상용화에 대한 어려움이 있다. 그 이유는 무선 업데이트 기술이 활용되고 있는 스마트 디바이스 업데이트 방식과 달리 주행 환경이라는 특수함이 있기 때문이다. 주행 중 차량용 ECU 펌웨어 (Firmware)에서 문제가 발생한다면, 운전자 및 탑승자의 생명과 직결되는 문제가 발생한다. 현재 차량용 무선 업데이트는 최신 펌웨어 (Firmware)나 기능을 제공하고 있지만 주행에 직접적인 영향을 주는 것들이 아닌 차량의 엔터테인먼트 혹은 부수적인 기능 업데이트

트에 국한되고 있다.

그래서 본 논문에서는 이러한 어려움을 개선하고 차량용 무선 업데이트 기술이 상용화가 될 수 있도록 차량용 ECU 펌웨어 무선 업데이트 성능 신뢰성 평가 기법을 제안한다. 제안 평가 기법의 결과를 기반으로 차량 공인 검사 기관에서도 활용할 수 있는 차량용 무선 업데이트 기술에 대한 표준을 제시한다[10]. 또한, 제안하는 평가 기법을 활용하여 공인 검사기관에서 활용할 수 있도록 테스트 베드와 설비를 제안하고 제안하는 평가 기술에 대한 표준과 신뢰성 그리고 범제화가 될 수 있는 가교 역할을 기대할 수 있다.

2. 본 론

2.1 자동차 소프트웨어 등급 정의



[Figure 3] Automotive Safety Integrity Level (ASIL) based on ISO26262

[Figure 3]은 ISO26262 기반 자동차 안전 무결성 수준을 나타낸 것이다. 자동차 소프트웨어의 종류는 매우 다양하며, 자동차의 기능이나 고도화가 될수록 계속해서 새로운 기능들의 소프트웨어들이 개발 될 것이다. 하지만 관리기관에서 모든 소프트웨어 기능들에 대해서 확인 및 검증하기에는 현실적으로 어려우며, 운영 및 관리에 대한 부담감이 증가하고 있다. 그 결과 자동차 소프트웨어의 중요도에 따라 등급을 구분하여 관리

방법을 체계화 할 필요가 있다[11].

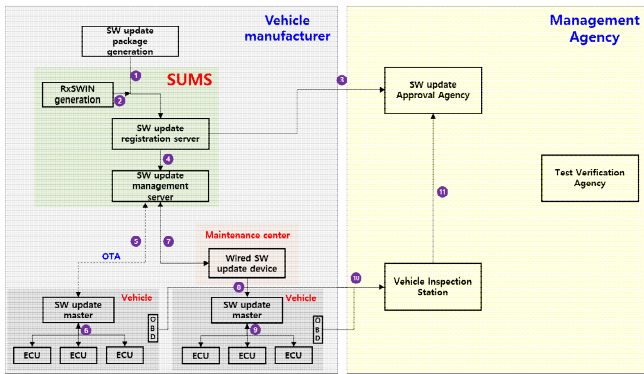
그래서 본 논문에서는 ISO26262 기반 자동차 안전 무결성 수준에 따라 등급을 구분하고자 한다. 자동차 소프트웨어 업데이트 신고/승인을 위한 등급을 안전을 기준으로 3단계로 정의한다.

통보성 검사단계 (Level 1)는 안전에 무관한 기능으로 제조사가 관리기관에 통보 후 별도의 승인절차 없이 SW업데이트를 진행한다.

검토성 검사단계 (Level 2)는 안전에 간접적으로 영향을 미치는 기능으로 관리기관에서 SW업데이트에 대한 내용을 검토 후 승인하며, 별도의 시험/검증 절차는 필요하지 않는다.

검증성 검사단계 (Level 3)는 안전에 직접적으로 영향을 미치는 기능으로 관리기관 및 전문 시험기관에서 해당 SW에 대한 시험/검증 후 결과에 따라 승인절차를 진행한다.

2.2 등급별 무선 업데이트 시나리오 정의



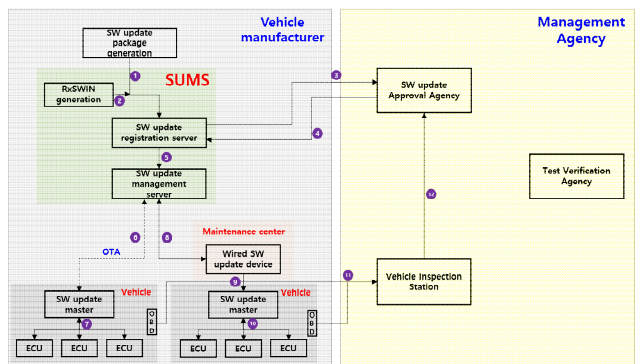
[Figure 4] Notification Level (Level 1) software update Diagram

[Figure 4]는 통보성 검사단계의 무선 업데이트 시나리오를 정의한 것이다. 소프트웨어 업데이트를 위해 생성된 패키지를 소프트웨어 등록 서버에 업로드 진행한다. 업데이트할 소프트웨어에 대한 RxSWIN를 생성하고, 소프트웨어 등록 서버에 업로드 진행한다. 등록된 소프트웨어 업데이트에 대한 내용을 승인기관에 통보

한다. 그리고 등록된 소프트웨어 패키지를 소프트웨어 업데이트 관리 서버로 전송 및 차량에 업데이트하기 위한 준비를 마친다.

무선 업데이트를 통해 차량의 소프트웨어 업데이트 마스터에 소프트웨어 패키지 파일을 전송하며, 소프트웨어 업데이트 마스터에서, 해당 ECU에 소프트웨어를 업데이트를 수행한다. 유선 소프트웨어 업데이트 장치로 업데이트할 소프트웨어 패키지 전송한다. 유선 소프트웨어 업데이트 장치에서 업데이트할 소프트웨어 패키지를 소프트웨어 업데이트 마스터로 전송하며, 소프트웨어 업데이트 마스터에서, 해당 ECU에 소프트웨어를 업데이트를 수행한다.

자동차 검사소에서 검사차량의 OBD를 통하여 소프트웨어에 대한 정보 (RxSWIN)를 수집하며, 검사가 완료된 차량의 RxSWIN 정보를 승인기관으로 송신하여, 등록된 소프트웨어가 맞는지 확인 및 검증을 진행한다.



[Figure 5] Review Level (Level 2) software update Diagram

[Figure 5]는 검토성 검사단계의 무선 업데이트 시나리오를 정의한 것이다. 소프트웨어 업데이트를 위해 생성된 패키지를 소프트웨어 등록 서버에 업로드 진행한다. 업데이트할 소프트웨어에 대한 RxSWIN를 생성하고, 소프트웨어 등록 서버에 업로드를 한다. 통보성 검사단계 시나리오와 달리 검토성 검사단계에서는 등록된 소프트웨어 업데이트에 대한 내용을 승인기관에 승인 요청을 진행한다. 승인기관에서 해당 소프트웨어 업데이트 건에 대한 내용을 검토 후 승인 또는 거절하고,

필요시 보완사항을 요구하며, 그에 따라 소프트웨어 수정 혹은 보완이 필요하다. 그 후, 승인기관으로부터 해당 건에 대해 승인이 된 경우 등록된 소프트웨어 패키지를 소프트웨어 업데이트 관리 서버로 전송 및 차량에 업데이트하기 위한 준비 진행한다. 이후 절차는 통보성 검사단계의 무선 업데이트 시나리오와 동일하게 진행된다.

무선 업데이트를 통해 차량의 소프트웨어 업데이트 마스터에 소프트웨어 패키지 파일을 전송하며, 소프트웨어 업데이트 마스터에서, 해당 ECU에 소프트웨어를 업데이트를 수행한다. 유선 소프트웨어 업데이트 장치로 업데이트할 소프트웨어 패키지 전송한다. 유선 소프트웨어 업데이트 장치에서 업데이트할 소프트웨어 패키지를 소프트웨어 업데이트 마스터로 전송하며, 소프트웨어 업데이트 마스터에서, 해당 ECU에 소프트웨어를 업데이트를 수행한다.

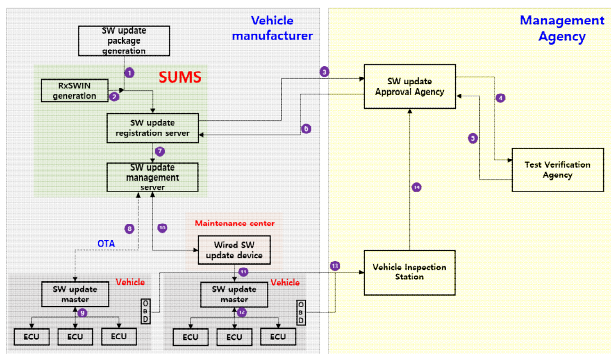
자동차 검사소에서 검사차량의 OBD를 통하여 소프트웨어에 대한 정보 (RxSWIN)를 수집하며, 검사가 완료된 차량의 RxSWIN 정보를 승인기관으로 송신하여, 등록된 소프트웨어가 맞는지 확인 및 검증을 진행한다.

사단계와 동일하게 등록된 소프트웨어 업데이트에 대한 내용을 승인기관에 승인 요청을 진행한다. 하지만 검증성 단계에서는 승인 요청이후 한 단계를 더 정의하게 된다. 해당 소프트웨어 업데이트 건에 대해 전문 시험기관에 의뢰를 진행한다. 시험기관은 해당 소프트웨어에 대한 검증 후 결과를 승인기관에 송부한다. 그 결과에 따라 승인기관에서 해당 소프트웨어에 대한 시험 결과에 따라 승인 또는 거절한다. 승인기관으로부터 해당 건에 대해 승인이 된 경우 등록된 소프트웨어 패키지를 소프트웨어 업데이트 관리 서버로 전송 및 차량에 업데이트하기 위한 준비 진행한다. 이후 절차는 통보성 검사단계와 검토성 검사단계의 무선 업데이트 시나리오와 동일하게 진행된다.

무선 업데이트를 통해 차량의 소프트웨어 업데이트 마스터에 소프트웨어 패키지 파일을 전송하며, 소프트웨어 업데이트 마스터에서, 해당 ECU에 소프트웨어를 업데이트를 수행한다. 유선 소프트웨어 업데이트 장치로 업데이트할 소프트웨어 패키지 전송한다. 유선 소프트웨어 업데이트 장치에서 업데이트할 소프트웨어 패키지를 소프트웨어 업데이트 마스터로 전송하며, 소프트웨어 업데이트 마스터에서, 해당 ECU에 소프트웨어를 업데이트를 수행한다.

자동차 검사소에서 검사차량의 OBD를 통하여 소프트웨어에 대한 정보(RxSWIN)를 수집하며, 검사가 완료된 차량의 RxSWIN 정보를 승인기관으로 송신하여, 등록된 소프트웨어가 맞는지 확인 및 검증을 진행한다.

각 단계별 무선 업데이트 시나리오는 절차상으로 동일하나 단계에 따라 관리기관의 승인 절차와 시험기관의 시험 절차가 추가되면서 무선 업데이트의 성능 신뢰성 향상을 목표로 시나리오를 정의했다. Level 1~3까지의 과정은 앞서 정의한 자동차 안전 무결성 수준 (ASIL) 등급에 따라 ASIL 등급 QM은 통보성 검사단계, A, B는 검토성 검사단계, C, D는 검증성 검사단계로 정의한다.



[Figure 6] Verification Level (Level 3) software update Diagram

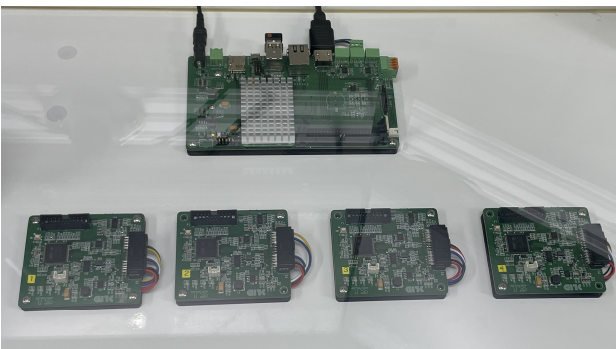
[Figure 6]은 검증성 검사단계의 무선 업데이트 시나리오를 정의한 것이다. 소프트웨어 업데이트를 위해 생성된 패키지를 소프트웨어 등록 서버에 업로드 진행한다. 업데이트할 소프트웨어에 대한 RxSWIN를 생성하고, 소프트웨어 등록 서버에 업로드를 한다. 검증성 검

3. 평가 시나리오 적용 결과

[Figure 7]은 무선 업데이트 시나리오 테스트 프로토타입을 나타낸 것이다. 프로토타입은 실차 주행 단계 전 시나리오 기반 무선 업데이트 성능 및 신뢰성을 확인한다. 실차 주행 단계에서 무선 업데이트 성능 및 신뢰성을 확인할 경우, 탑승자에 대한 안전성을 보장할 수 없어 최대한 프로토타입 Lab 환경에서 성능 및 신뢰성 검증을 마친 후 실차 주행단계에서 검증을 진행한다.



[Figure 7] OTA update scenario test prototype



[Figure 8] OTA transceiver and ECU Module

[Figure 8]은 개발한 차량용 무선 업데이트용 송수신기와 무선 업데이트가 가능한 ECU 모듈을 나타낸 것이다. 실차 주행 단계에서 이뤄지는 모든 과정을 [Figure 8] 프로토타입에서 모든 검사 시나리오를 구현한다. [Figure 9]에서 개발한 무선 업데이트용 송수신기는 기존 차량에 무선 업데이트 송수신 기능이 없는 점

을 고려하여 LTE 상용망 무선 통신기반으로 제작하였으며, 기존 차량에 부착하여 차량 내부의 ECU와 연결을 한다면 무선 업데이트 기능이 추가되는 형식이다. 그리고 4개의 작은 보드는 Lab 환경 시험단계에서의 ECU 모듈을 제작한 것이다. 각각의 ECU들은 Level 1~3 단계별로 검증하기 위해서 Level 전용 검증 ECU와 통합 검증을 위해 모든 Level 검증용 ECU를 포함해서 총 4개 ECU 모듈을 제작했다. 각각의 ECU들은 ASIL 등급 기반으로 헤드라이트 조명 밝기부터 전방 충돌 인식 기능까지 다양한 펌웨어들을 적용하여 통보성 검사 단계부터 검토성 검토단계, 끝으로 검증성 검사단계까지 성능 및 신뢰성 검증을 진행했다.

차량용 무선 업데이트 기술의 성능 평가 및 신뢰성 보장을 위해 체크섬과 해시 함수가 필요하다. 가장 먼저 체크섬은 인텔 실행파일에서 사용하고 신뢰성을 확보한 체크섬 방식을 수행하며, <Table 2>에서 인텔 체크섬을 계산하는 절차를 보여주고 있다.

<Table 2> Calculate Checksum Algorithm

Algorithm 1: Caludate Checksum Procedure

```

input: data, and len
output: csum
// csum is defined as 1byte(unsinged char)
1 csum ← 0xff;
2 for lenindex = 0 to len do
3   | csum ← csum + data[lenindex];
4 end
5 csum ← XOR(csum);
6 return csum;
    
```

<Table 2>에서는 데이터의 신뢰성을 보장하기 위해 0xff를 시작으로 개별 데이터를 1바이트 단위로 더하기 연산을 수행한다. 이후 본 논문에서는 XOR 연산을 수행하여 체크섬을 생성한다.

다음으로 해시 함수는 OTA 업데이트 시 파일에 대한 위변조를 방지하기 위한 방법으로 활용한다. 본 논문에서는 데이터의 위변조를 방지하기 위해 DES 64 알고리즘을 사용하였다[13]. 다음 수식 1은 DES 64 계

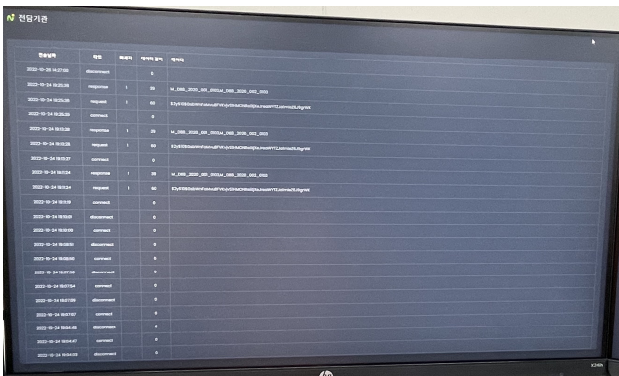
산식을 보여주고 있다.

$$R_i = f(R_{i-1}, K_i) \sum_{i=1}^n R_i \quad (1)$$

R_i 은 i 번째 암호화 주기, K_i 는 i 번째 키 값을 의미한다. 본 논문 결과 파라미터에서는 n 을 16으로 하였다.

이러한 데이터 위변조에 대한 두 가지 방법을 기반으로 다음과 같은 절차를 적용하여 무선 업데이트 성능 및 신뢰성 평가를 수행하였다. 첫 번째, OTA 업데이트를 위해 시스템은 패킷 구조를 구성한다. 두 번째, OTA 업데이트 파일에 대한 원본 바이트 파일을 기반으로 체크섬을 생성한다. 그리고 세 번째는 OTA 업데이트 파일 위변조를 위해 원본 바이트 파일을 해시 함수를 통해 암호화한다. 끝으로, 구조화된 패킷을 ECU 모듈로 전송한다. 이 과정을 거친 무선 업데이트 OTA 결과를 [Figure 9]와 [Figure 10]에서 나타냈다.

본 논문에서는 위와 같은 절차를 기반으로 OTA 데이터 신뢰성 평가를 진행한다.



[Figure 9] OTA update processing

[Figure 9]는 무선 업데이트된 펌웨어 이력을 나타낸 것이다. [Figure 9]와 같이 펌웨어에 변경점이 발생하면 업데이트를 진행하고 변경된 사항과 ASIL 등급에 따라서 단계별 무선 업데이트 검증 시나리오를 적용하여 차량용 ECU 무선 업데이트 성능 및 신뢰성을 검증한

다. 프로토타입에서는 ECU 업데이트를 통해서 변경된 기능들을 [Figure 10]의 컨트롤 패널을 활용하여 기능 검증을 하였으며, 그에 펌웨어들이 정확하게 업데이트가 적용되었음을 확인할 수 있었으며, 탑승자의 안전성이 보장되는 환경에서 차량용 무선 업데이트 성능 및 신뢰성을 검증한다.



[Figure 10] OTA update performance verification control pad

[Figure 10]은 펌웨어 별 성능 검증을 위한 컨트롤 패널을 나타낸 것이다. 컨트롤 패널을 앞서 설명했듯이, 펌웨어 업데이트 이후 성능 검증을 통해 신뢰성 확보 및 기능 검증을 한다.

본 논문에서 제안하는 평가 시나리오 결과를 기반으로 자동차 검사를 주관하는 공인기관에서도 평가 시나리오와 프로토타입 기반의 설비들을 제안하여 차량용 ECU 무선 업데이트 검사 방식으로 표준안으로 제안할 수 있다.

4. 결 론

본 논문은 차량용 ECU 펌웨어 무선 업데이트 성능 신뢰성 평가 기술 구현 방법에 대해서 제안하였다. 본 논문은 자동차 무선 업데이트 기술에 대한 단점들을 보완하고 공인 검사기관의 인증을 통한 기술의 신뢰성을 향상하고 그로 인해, 자동차 무선 업데이트 기술이 확산 및 보급화를 기대한다.

또한 본 논문에서 제안하는 차량용 무선 업데이트 기술의 평가 기술을 기반으로 주행 단계에서의 안전성과 신뢰성 보장을 기대할 수 있다. 그리고 본 논문을 통해 제작한 차량용 무선 업데이트 프로토 타입용 검

증 시스템을 고도화하여 차량 무선 업데이트 기술의 표준화 및 지식 재산권 확보도 진행 될 예정이다. 이와 같이 자동차 무선 업데이트 기술은 여러 분야의 기술들과의 융합을 통해 개선하고 발전해 나갈 가능성이 많은 분야라고 생각한다.

Acknowledgement

본 연구는 국토교통부/국토교통과학기술진흥원의 지원으로 수행되었음. (22AMDP-C162334-02 : 자동차 통합보안 안전성 평가기술 개발)

(This work is supported by the Korea Agency for Infrastructure Technology Advancement(KAIA) grant funded by the Ministry of Land, Infrastructure and Transport (Grant 22AMDP-C162334-02).)

References

- [1] J. Bauwens, P. Ruckebusch, S. Giannoulis, I. Moerman and E. D. Poorter, "Over-the-Air Software Updates in the Internet of Things: An Overview of Key Principles," in *IEEE Communications Magazine*, vol. 58, no. 2, pp. 35-41, 2020.
- [2] H. Kexun, W. Changyuan, H. Yanyan and F. Xiyu, "Research on cyber security Technology and Test Method of OTA for Intelligent Connected Vehicle," 2020 International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), pp. 194-199, 2020.
- [3] T. Kim and S. Park, "Compare of Vehicle Management over the Air and On-Board Diagnostics," 2019 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), pp. 1-2, 2019.
- [4] S. Yeasmin and A. Haque, "A Multi-Factor Authenticated Blockchain-Based OTA Update Framework for Connected Autonomous Vehicles," 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall), pp. 1-6, 2021.
- [5] M. Y. Hassan, S. Choudhury and Z. M. Fadlullah, "On Optimal Scheduling of OTA Software Updates for Smart Vehicles Leveraging Fog Computing," 2021 International Wireless Communications and Mobile Computing (IWCMC), pp. 2044-2049, 2021.
- [6] M. Khatun, M. Glaß and R. Jung, "An Approach of Scenario-Based Threat Analysis and Risk Assessment Over-the-Air updates for an Autonomous Vehicle," 2021 7th International Conference on Automation, Robotics and Applications (ICARA), pp. 122-127, 2021.
- [7] A. Ghosal, S. Halder and M. Conti, "STRIDE: Scalable and Secure Over-The-Air Software Update Scheme for Autonomous Vehicles," ICC 2020 - 2020 IEEE International Conference on Communications (ICC), pp. 1-6, 2020.
- [8] Y. Jia, X. Shao, S. Wang, R. Zhai, Q. Li and Y. Wang, "Research on Vehicle OTA Upgrade Technology Based on BSDIFF Difference Algorithm," 2021 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA), pp. 1113-1117, 2021.
- [9] B. Kim and S. Park, "ECU Software Updating Scenario Using OTA Technology through Mobile Communication Network," 2018 IEEE 3rd International Conference on Communication and Information Systems (ICCIS), pp. 67-72, 2018.
- [10] S. Hong, "Ergonomics Approach for Quality Improvement of Automobile Repair Service", *Journal of the Korean Institute of Plant*

Engineering, vol. 19, no. 1, pp. 75-80, 2014.

- [11] S. Choi, "Establishment of Extended Model for Determining and Evaluating ASIL in the ISO 26262 Automotive Functional Safety System", *Journal of the Korean Institute of Plant Engineering*, vol. 22, no. 2, pp. 39-55, 2017.
- [12] S. D. Rihan, A. Khalid, and S. E. F. Osman, "A performance comparison of encryption algorithms AES and DES," *International Journal of Engineering Research & Technology (IJERT)*, vol. 4, no. 12, pp. 151-154, 2015.
- [13] P. Mahajan and A. Sachdeva "A study of encryption algorithms AES, DES and RSA for security," *Global Journal of Computer Science and Technology*. 2013.

