AUTOCRYPT

Automotive KMI

자동차 환경의 키 관리

심상규 (Ph.D / CTO / 부사장)

목차(Table of Contents)

서론

유럽을 시작으로 세계 주요 국가들이 자동차 사이버보안을 의무화하고 있다. 자동차에 사이버보안을 적용하기 위해 필수적으로 해야 하는 활동들을 의무로 규정하는 것이 UN Regulation 155이고, UN R155의 제정 이후 유럽연합과 세계 주요 국가들이 UN R155를 참조하여 각 국가의 법을 제정했다. UN R155가 정의하는 자동차 사이버보안의 What은 ISO/SAE 21434를 따르는 How로 구체화 되었다. UN R155와 ISO/SAE 21434는 자동차 사이버보안을 위해 완성차 제조사가 해야 할 활동들을 정의하고 활동의 구체적인 방법들을 정의하고 있으나, 자동차의 사이버보안을 위해 어떤 보안 수단을 어떻게 적용해야 할지를 다루지는 않는다.

자동차에 존재할 것으로 예상되는 보안 위협(Threat)은 보안 위험(Risk)으로 정의되고 다루어져야 하는데, 제거할 수 없는 위험은 위험의 발생을 방지하기 위해 대응 수단을 적용하여야 한다. 보안 위험을 방지하거나 탐지하는 대응 수단은 보안 기능(Feature)으로 설계, 구현, 적용되어야 한다. UN R155와 ISO/SAE 21434는 보안 위협을 평가(assessment)하고 보안 위험에 대한을 분석(analysis)을 수행한 후 위험에 적절한 대응할 것을 명시하지만, 적절한 대응이 무엇인지, 적절한 대응 수단을 어떻게 구현할지는 이 문서들의 범위 밖이다. 그렇다면, 보안 위협과 위험에 대응할 수 있는 보안 기능들을 어떻게 확보할 것인가.

보안 기능의 대부분은 암호 알고리즘들을 기반으로 구현된다. 암호 알고리즘들을 체계적 으로 엮어서 보안 기능을 완성한다. 암호 알고리즘들은 국제 표준으로 공개되고, 암호 알고리즘들을 체계적으로 엮어서 보안 기능을 구성하는 방법도 비밀이 아닌 경우가 거의 대부분이다. 암호 알고리즘이 비밀이 아니고, 보안 기능의 구성이 비밀이 아니기 때문에 우리가 비밀을 유지하면서 보안성 수준을 높일 수 있는 방법은 암호 알고리즘을 구동하는데 사용되는 암호키 뿐이다. 암호 알고리즘에 사용되는 암호키가 비밀로 잘 관리되고, 적절히 사용되어야 암호 알고리즘에서 차곡차곡 쌓아 올린 보안 기능이 보안성을 갖추게 된다.

자동차 환경에서 사이버보안의 높은 수준을 확보하기 위해 다양한 보안 기능들이 구현, 적용되고 있지만 많은 이들이 암호키의 중요한 의미에 대해 인식이 부족하다. 암호키를 안전하게 관리하는 것은 힘들게 구현한 보안 기능이 제대로 역할을 하느냐, 아니냐의 큰 문제이다. 본 백서는 암호키의 안전한 관리를 위한 키 관리 개념과 자동차 환경에서 필요한 키 관리 기술들을 살펴본다.

2. 사이버보안과 암호키

2.1 사이버 보안 솔루션의 구현

자동차를 비롯한 많은 분야에서 사이버보안이 필수 요소로 받아들여지고 있다. 모든 기기들이 네트워크에 연결되고, 소프트웨어의 비중이 늘어나고, 서비스와 연계한 활용이 늘어나면 사이버보안의 필요성은 더욱 증대될 수 밖에 없다. 특히, 스마트폰 처럼 자동차가 스마트기기로 변모하는 SDV 시대에 자동차의 사이버보안은 무엇보다 중요한 요소가 아닐 수 없다. 기존의 자동차에서 소프트웨어는 하드웨어에 종속적인 부품으로 인식되었지만, SDV 시대에서 소프트웨어는 하드웨어에 독립적인 부품으로 인식된다. 소프트웨어가 하드웨어로부터 독립되면 임의의 소프트웨어가 특정 하드웨어에서만 동작하는 것이 아니라 다른 하드웨어에서도 동작할 수 있게 된다. 많은 보안 위협이 소프트웨어의 취약점을 악용하여 발생하기 때문에, 보안 취약점을 내포하고 있는 소프트웨어는 특정 모델의 차량에만 영향을 미치는 것이 아니라 소프트웨어가 구동될 수 있는 모든 차량 모델에 영향을 미치게 된다.

Figure 2-1 | SDV 환경의 소프트웨어

사이버보안을 적용하는 것은, 적용 환경에서 발생할 수 있는 보안 위험(Security Risk)을 분석하고, 분석된 보안 위험에 대응할 방안을 설계하는 것에서 시작한다. 시스템 아키텍처를 설계하는 대부분의 아키텍트는 하드웨어에서 소프트웨어와 서비스에 이르는 여러 계층을 정의하고, 보안을 독립적인 하나의 요소로 정립하는 경향이 있다.

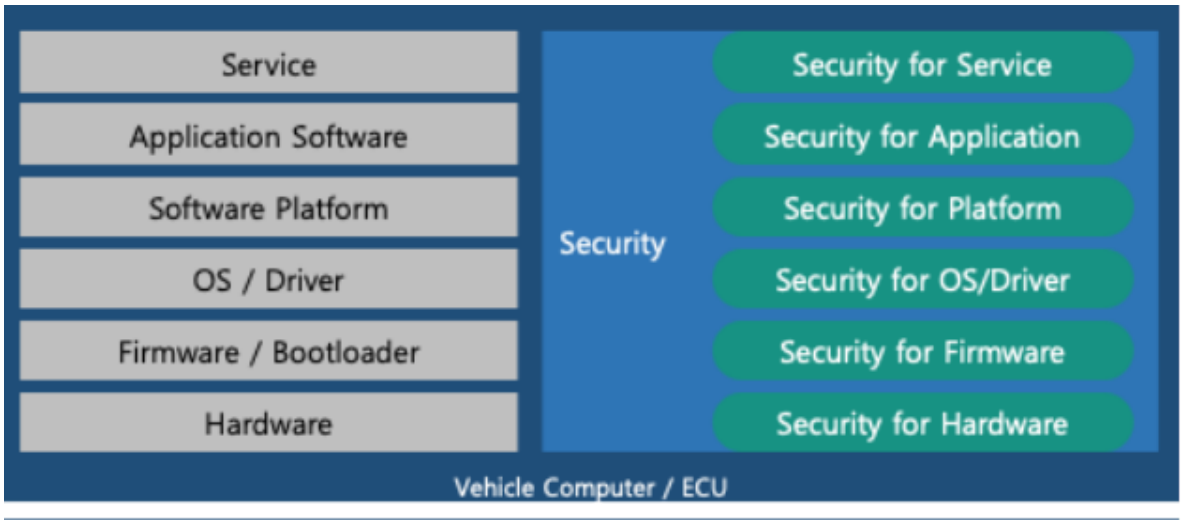

Figure 2-2 | 전장 제어기의 시스템 아키텍처 예시

보안은 하드웨어에서 소프트웨어와 서비스에 이르는 여러 계층과는 별개로 동작해야 하는 것은 옳지만, 보안이 시스템의 어느 계층을 보호하는가에 따라 보안도 제 각각의 특성을 가질 수 있다. 예를 들어, OS를 보호하기 위한 보안과 어플리케이션 소프트웨어를 보호하기 위한 보안이 동작하는 방식과 구조는 다르다. 시스템에 보안을 적용하여 시스템을 온전히 보호하기 위해서는 시스템의 여러 계층들 마다 적절한 보안 기술을 적용해야 한다. 시스템 아키텍처에서 이러한 개념을 구조화해서 설계하기란 쉬운 문제가 아니다. 이런 이유로 대부분의 아키텍트는 보안을 하나의 상자로 정의하곤 하지만, 상자를 열면 상자 속에 여러 구성물이 있을 수 밖에 없다. 보안이 하나의 공구 상자라면, 해결하고자 하는 문제에 특화된 여러가지 공구들이 공구 상자 안에 담겨 있다.

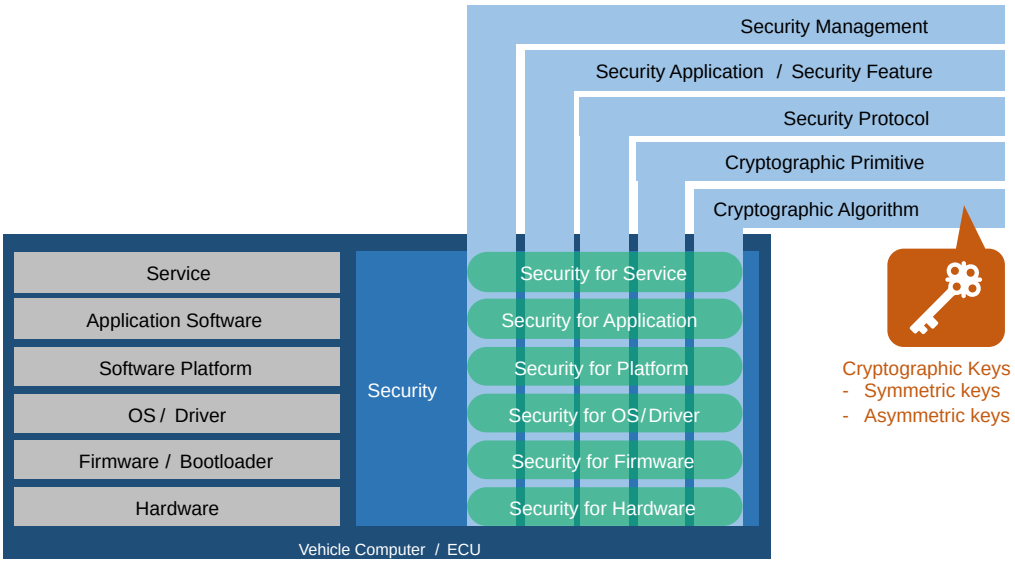

Figure 2-3 | 보안 기술을 구현하기 위한 계층 구조

시스템의 어느 계층을 보호하는 '공구'냐에 따라서 공구의 모양새는 다를 수 있지만, 공구를 만드는 방법은 유사하다. 시스템의 특정 계층을 보호하기 위해 필요한 것은 보안 응용(Security Application) 또는 보안 기능(Security Feature)이다. 새롭게 개발하고자 하는 시스템이 외부 통신을 사용하고 외부 통신을 보호하기 위해 TLS가 필요하다면, 'TLS를 사용한 외부 통신 보안'이 하나의 보안 응용 또는 보안 기능이라 정의할 수 있다. TLS를 구현하자면 TLS Client와 TLS Server의 역할이 있고 두 개체가 여러 단계에 걸쳐서 단계 별로 수행해야 하는 처리 과정이 있다. TLS Client가 어떤 연산 처리를 해서 TLS Server에게 전송하면, TLS Server는 수신한 데이터를 사용하여 수행해야 하는 연산 처리 과정이 있다. 마찬가지로, TLS Server가 연산 처리한 데이터를 TLS Client에게 전송하면, TLS Client는 수신한 데이터를 사용하여 수행해야 하는 연산 처리 과정이 있다. 이러한 처리 과정들은 보안 프로토콜(Security Protocol)이다.

보안 프로토콜의 한 단계를 처리하기 위해 필요한 연산들은 인증, 접근제어, 전자서명, 암호화 등을 결합하여 이루어진다. 이러한 기능들은 암호학적 근원(Cryptographic Primitive)이다. 암호학적 근원 기능들을 구현하는 데에는 암호 알고리즘(Cryptographic Algorithm)이 필요하다. 예를 들어, 전자서명은 메시지 해쉬(Hash), 난수 생성, 전자서명 생성 알고리즘, 서명 키에 대한 접근, 전자서명 데이터의 인코딩(encoding) 등의 조합으로 구성된다. 추가로 보안 관리(Security Management)도 중요한 요소이다. 보안 응용을 구현했더라도 구현된 보안 기능을 운영하기 위한 활동과 시스템의 보안 수준을 일정 수준 이상으로 지속시키기 위한 활동 등이 필요하다.

2.2 키 관리의 중요성

시스템에 특정 계층을 보호하는 보안을 추가하기 위해서, 암호 알고리즘(Cryptographic Algorithm), 암호학적 근원(Cryptographic Primitive), 보안 프로토콜(Security Protocol), 보안 응용(Security Application)과 보안 관리(Security Management)까지 구성되어야 한다. 이들의 일부는 하드웨어로 구현되기도 하지만, 대부분은 소프트웨어로 구현된다. 이들의 대부분을 표준화 하거나, 일부를 표준화 하는 경우도 있다.

보안 응용과 보안 기능을 구현하는 것은 암호 알고리즘부터 차근차근 쌓아 올려야 건물을 만드는 것과 같다. 암호 알고리즘은 표준으로 정의된 것들만 사용하기 때문에 암호 알고리즘의 구현은 표준과의 정합성이 기본이다. 구현된 암호 알고리즘이 일정 수준 이상의 처리 성능을 갖춰야 할 뿐만 아니라, 암호 알고리즘 구현물이 보안적으로 취약점을 내포하지 않도록 하는 것이 필요하다. 하지만, 이런 것들은 구현 상의 문제일 뿐이다.

암호 알고리즘이 잘 구현되었더라도 암호 알고리즘이 사용할 암호키를 관리하는 것은 별개의 중요한 문제이다. 우리가 사용하는 컴퓨터에 보안을 강화할 보안 소프트웨어를 설치했다고 가정해 보자. 이 보안 소프트웨어의 관리를 위해 비밀번호가 있는데, 비밀번호를 외우는 것이 어려우니 컴퓨터 화면 옆에 메모를 만들어 붙여 두었다. 보안 소프트웨어 덕택에 보안은 강화되었고, 비밀번호가 필요할 때마다 메모를 확인하면 되니 사용 편의성도 확보 되었다. 하지만, 메모에 적어둔 비밀번호가 유출된다면? 막강한 보안 기능을 제공하는 보안 소프트웨어라도 비밀번호 유출 앞에서는 무용지물이다.

암호 알고리즘도 마찬가지이다. 암호 알고리즘의 키를 잘 관리하는 것이 핵심이다. 암호 알고리즘이 대칭키(Sysmmetric Key)를 사용하냐, 비대칭키(Asymmetric Key)를 사용하느냐의 차이가 있으나, 키가 유출되면 암호 알고리즘은 제 역할을 수행할 수 없다. 암호 알고리즘은 표준으로서 공개되어 있는 것이기 때문에 동일한 암호 알고리즘이 구현된 시스템 간의 구분을 제공하는 것은 암호키 뿐이다. 적용되어 있는 보안 기능은 암호 알고리즘들의 조합이고, 암호 알고리즘들의 암호키는 암호키를 알고 있는 특정한 기기 혹은 사용자만이 특정 보안 기능을 수행할 수 있음을 증명하는 근거가 된다. 내 컴퓨터에 로그인할 수 있는 사람은 내 컴퓨터 계정의 로그인 패스워드를 아는 사람이고, 도어락이 설치된 문을 정상적으로 통과할 수 있는 사람은 키를 소유하고 있거나 비밀번호를 아는 사람이다. 비밀번호가 유출되면, 로그인을 허용하지 않은 사람이 로그인을 할 수 있고 출입이 허가되지 않은 사람이 출입을 할 수 있게 된다.

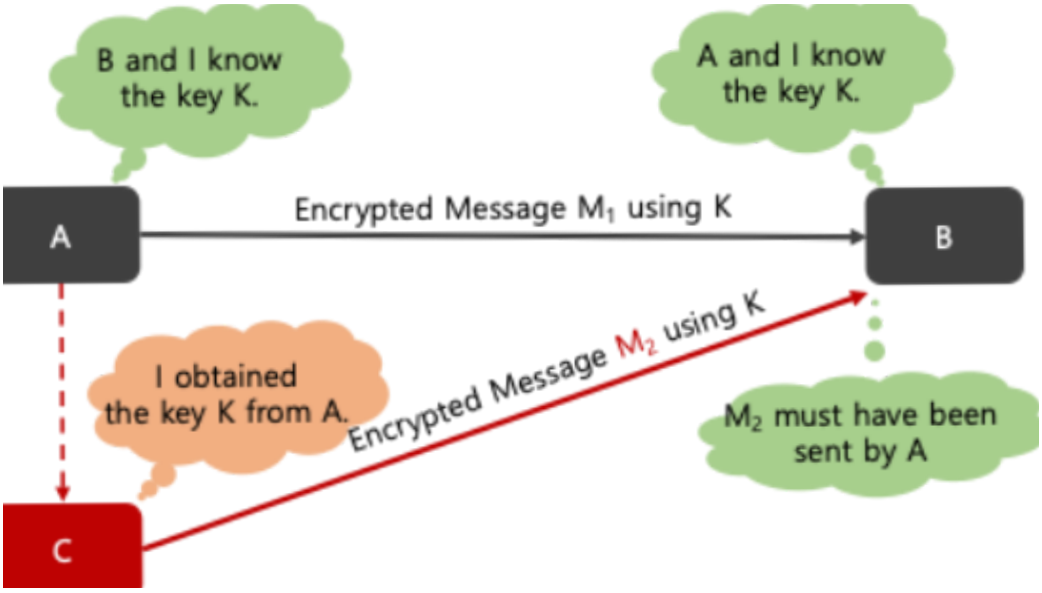

Figure 2-4 | 암호키 유출 피해의 사례

자동차, IoT기기, 컴퓨터, 클라우드 등에서 암호 알고리즘을 사용하여, 사용자 인증, 기기 인증, 통신 채널의 보안, 컴퓨팅 자원에 대한 접근 제어 등을 제공하고 있다. 기기 A와 기기 B가 암호키 K를 공유하고 있다고 가정하자. 기기 A와 기기 B는 암호키 K를 사용하여 암호화 된 데이터를 주고 받음으로써 기기 A와 기기 B는 상대가 누구인지 알 수 있다. 하지만, 기기 A가 보유한 암호키 K가 기기 C에게 유출된 상황을 생각해 보자. 기기 C는 새롭게 취득한 암호키 K로 암호하된 데이터를 기기 B에게 보낼 수 있고, 기기 B는 수신한 데이터를 정상적으로 생성할 수 있는 것은 기기 A 뿐이니, 기기 A가 보낸 데이터라고 "당연하게" 여길 것이다. 데이터를 보낸 것은 기기 C이지만, 기기 B는 기기 A와 기기 C를 구분해 낼 수 없다. 암호키가 유출되었기 때문에 발생하는 이런 문제는 가장 단순한 Impersonation의 한 사례이다.

3. 암호키의 관리

3.1 기존의 키 관리 방법들

우리는 다양한 온라인 서비스를 사용하고 있고, 서비스마다 등록된 사용자 계정을 가지고 있다. 서비스에 로그인하기 위해서는 등록된 사용자 계정과 비밀번호를 알아야 한다. 서비스의 수가 많아짐에 따라 계정과 비밀번호를 관리하는 것도 어려워지고 있다. 가장 쉬운 방법은 모든 서비스에서 사용하는 사용자 계정과 비밀번호를 통일시키는 것이다. 가입하는 서비스가 늘어나도 하나의 계정과 비밀번호로 모든 서비스 사이트에 로그인할 수 있으니 편리하다. 하지만, 치명적인 문제가 있다. 만약, 사용하는 서비스 사이트 중의 하나에서 사용자 계정과 비밀번호가 유출된다면, 내가 사용하는 다른 서비스 사이트의 로그인 정보가 모두 유출되는 것과 같다. 보안 전문가들은 서비스 사이트마다 서로 다른 비밀번호를 사용할 것을 권장하는 이유이다.

기기들도 다른 기기와 연결하거나 온라인 서버에 연결하기 위해 비밀 정보를 사용한다. 사람은 비밀 정보를 외울 수 있어야 하기 때문에 비밀번호를 사용하지만, 기기는 암호키를 비밀 정보로 사용한다. 자동차에는 많은 ECU(Electronic Control Unit)이 탑재되고, ECU의 보안 확보를 위해 암호키를 사용한다.

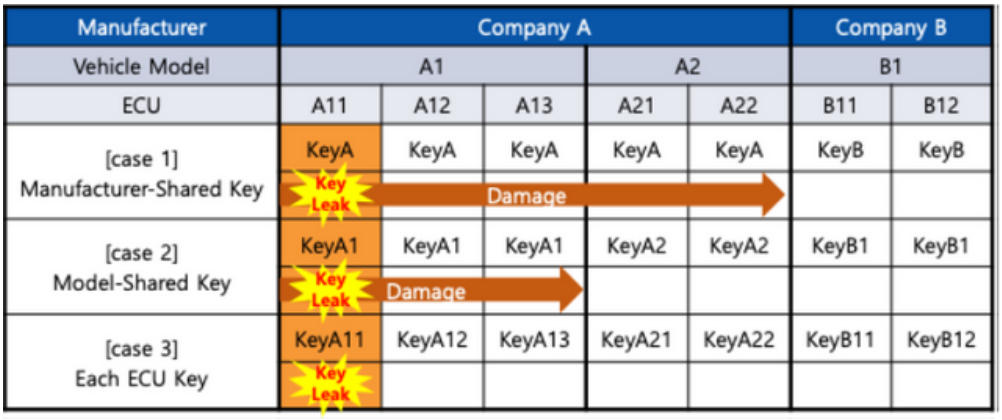

Figure 3-1 | 자동차 내부의 암호키 사용 예시

자동차의 많은 ECU에 사용되는 암호키를 관리하는 것은 어려운 문제이다. 가장 손쉬운 방법은 특정 제조사가 생산하는 모든 차량의 모든 ECU에 동일한 키를 사용하는 것이다. 이 경우, 제조사가 사용하는 키가 유출되면 피해가 모든 차량에 미칠 수 있다. 암호키를 조금 더 구분해서, 차량 모델 별로 암호키를 공유하는 방법을 고려해 볼 수 있다. 특정 차량 모델 안에 탑재되는 모든 ECU들이 같은 암호키를 공유하는 방식이다. ECU들 중에 하나의 ECU에 저장되어 있는 암호키가 유출되는 경우, 그 피해는 해당 차종의 모든 차량에 미치지만, 다른 차종으로 확산되지는 않는다. 그 보다 보안 강도를 높이자면, 같은 차종의 차량 모델에 포함되는 ECU들 간에도 다른 키를 사용하는 것이다. 특정 ECU가 사용하는 암호키가 유출되더라도 다른 ECU들의 암호키는 계속 사용 가능하기 때문에 앞에서 언급된 방식들 보다는 보안 강도가 높아진다. 하지만, 이 방식 마저도 보안에 문제가 없는 것이 아니다. 하나의 차량 모델이 양산되기 시작하면 많은 물량이 생산되기 때문에 동일 모델에 포함된 특정 제어기가 동일한 암호키를 고정적으로 사용하고 있다면 피해가 차량 한 대로 그치지 않는다.

3.2 암호키 관리의 기본 법칙

키 관리의 중요성은 자동차에만 해당하는 문제가 아니다. 모든 IT기기, IoT기기, 서버 시스템, 클라우드, 네트워크 등에 이르기까지 모든 보안은 암호학적 알고리즘을 이용하여 구현되기 때문에 암호학적 알고리즘의 구동에 사용되는 암호키의 안전한 관리는 중요한 요소로 인식되어 왔다. 암호키를 안전하게 활용하기 위해서는 중요한 원칙들이 지켜져야 한다.

3.2.1 암호키를 사용하는 주체의 최소화

암호키는 암호키를 알고 있는 주체만이 가능한 연산을 가능하게 함으로써 해당 주체의 자격을 수용하는 신뢰 근거가 된다. 이러한 근거는 암호키를 알고 있는 주체가 적어야 그 효용성이 높아지며, 암호키의 유출에 대한 대응 효과도 높아진다. 동일한 암호키가 10개의 시스템에 저장되어 있다면, 10개의 시스템들 중의 하나와 나머지 9개의 시스템을 구분 짓는 것이 어렵고, 암호키를 부정한 방법으로 획득한 11번째 시스템의 진위 여부를 판단하는 것도 쉽지 않다. 가장 바람직한 경우는 특정 암호키를 아는 주체가 하나이거나 최대 둘인 경우이다. 시스템이 암호화하여 저장한 데이터를 복호화하여 읽는 경우는 암호키를 아는 주체가 하나이다. 시스템이 개인키와 공개키 쌍을 사용하는 경우, 개인키를 아는 시스템이 하나만 존재하는 것이 바람직하다. 고유한 대칭키를 사용하여 보안 통신을 주고 받는 송신자와 수신자는 암호키를 알고 있는 주체가 둘인 경우에 해당한다

3.2.2 암호키의 유일성(Uniqueness)

주체 A와 주체 B가 256비트 길이의 서로 다른 암호키를 사용하더라도, 주체 A의 암호키와 주체 B의 암호키가 우연히 동일하면 전체 시스템의 보안성에 심각한 위험을 초래한다. 암호키를 생성할 때 난수 생성기를 사용하는 것이 일반적이기 때문에 암호키를 생성하는 데에 사용되는 난수 생성기의 보안성이 매우 중요하다. 주체들 간에 사용하는 암호키가 서로 달라야 하며, 동일한 주체가 암호키의 용도에 따라서도 암호키가 달라야 한다. 특히, 동일한 암호키를 데이터 암호화/복호화와 개체 인증에 사용하는 것은 위험하다. 가령, 개인키와 공개키 쌍을 사용하는 주체가 동일한 키 쌍으로 데이터 암호화와 전자서명을 사용하는 것은 절대적으로 위험하다. 암호키는 키를 사용할 주체 마다, 암호키가 사용되는 용도에 따라 유일성을 갖도록 해야 한다.

3.2.3 암호키의 안전한 저장과 접근제어

암호 알고리즘을 사용하여 통신에서 인증을 적용한다면, 암호 알고리즘의 암호키를 소유하고 있음을 증명함으로써 신원을 인증할 수 있게 된다. 암호화된 데이터를 암호 알고리즘으로 복호화 한다면, 암호 알고리즘의 암호키를 소유하고 있어야 데이터를 올바르게 복호화할 수 있다. 암호키의 소유를 증명하는 것은 암호 알고리즘의 정상적인 동작을 통해 보안 응용이 기능을 할 수 있도록 해준다.

암호키의 소유 증명은 암호키를 안전하게 저장하고, 저장된 암호키에 소유가 허용되지 않는 주체가 접근하지 못하도록 하는 것을 요구한다. 암호키가 안전하게 저장되지 못해서 다른 주체가 손쉽게 암호키를 획득하거나, 안전하게 저장되어 있더라도 암호키에 대한 접근 제어가 느슨하다면 다른 주체가 암호키에 접근할 가능성이 존재한다.

3.2.4. 암호키의 적절한 갱신

온라인 서비스 사이트에 로그인을 하면, 비밀번호 변경을 권장하는 안내를 볼 수 있다. 비밀번호를 변경하지 않고 오랫 동안 사용하면 비밀번호 노출의 우려가 높아지기 때문에 보안 강도가 저하될 수 있다. 암호키도 마찬가지로 정기적인 갱신을 하여야 한다. 암호키가 사용되는 암호 알고리즘은 수학적 연산의 복잡한 조합이고 수학 연산이 조합되는 방식과 상세한 구조가 표준으로 정리되어 있으며, 다양한 오픈소스들도 존재한다. 암호 해독을 시도하는 공격자는 암호 알고리즘을 구동할 수 있고, 적절한 암호키를 알지 못할 뿐이다. 동일한 암호키를 사용하는 암호 알고리즘을 대상으로 암호키를 추론하는 다양한 공격 기법들이 존재하고, 동일한 암호키를 사용하는 평문과 암호문의 조합을 대량으로 확보하면 사용된 암호키에 대한 정보를 획득할 가능성이 높아진다. 이러한 이유로 암호키를 적절한 기간 마다 갱신하는 것이 필요하고, 암호키 갱신에 관한 정책을 키 관리 정책에 반영해야 한다.

3.2.5. 암호키의 보안정책 운영

일상 생활에서 우리는 다양한 공산품을 사용하고 있는데, 모든 공산품에는 사용법과 취급법, 주의 사항 등이 명시되어 있다. 해당 물품을 적절하게 사용하지 않고 남용하거나 오용한다면 사용자에게 피해를 주거나 제품을 손상시킬 수 있기 때문이다. 암호키도 암호키 마다 각각의 사용법과 취급법, 주의 사항 등이 명시되어 있어야 한다. 암호키를 적절하게 사용하지 않고 남용하거나 오용한다면 시스템의 보안성을 크게 손상시킬 수 있다. 암호키의 사용법, 취급법, 주의 사항 등을 명시하는 것을 암호키의 보안정책이라 한다. 암호키의 암호키의 보안정책은 보안정책이 적용되어야 하는 암호키의 범위를 지정하고, 적용 범위 내의 암호키를 어떻게 다루어야 하는지를 명시한다. 암호키의 보안정책은 신뢰할 수 있는 주체에 의해서 생성되고, 암호키의 이용 상태와 새로운 보안 취약점 정보 등을 분석하여 보안정책을 갱신할 필요가 있다면 빠르게 수정하여 배포해야만 한다. 모든 암호키는 보안정책에 따라 생명주기 동안 관리되어야 하기 때문에 암호키의 보안정책을 수립하고 운영하는 중앙의 신뢰 주체에 의해 중앙 관리되어야 한다.

3.2.6 암호키의 생명주기 관리

암호키도 수명을 가진다. 암호키는 생성됨으로써 생명주기를 시작하여 폐기됨으로써 생명주기를 마친다. 생명주기 동안에 암호키는 저장되고, 조회되고, 암호 알고리즘에 사용되고, 갱신되고, 사용 중지 되기도 하고, 삭제되었다 복구되기도 하는 등의 많은 일이 발생할 수 있다. 암호키의 생명주기 동안 발생하는 모든 과정은 철저하게 보안정책에 따라 이루어져야 한다. 또한, 어떤 암호키가 생명주기 중 어느 과정에 있는지 관리되어야만 한다. 가령, 사용 중지된 암호키가 사용되는 것은 시스템의 보안에 큰 위협이 된다. 암호키의 생명주기 동안 일어나는 모든 과정은 해당 과정을 처리할 수 있는 권한을 가진 적절한 주체에 의해서만 시행되어야 한다. 예를 들어, 암호키의 삭제 권한이 없는 주체가 암호키를 삭제하는 것은 아주 큰 문제이다.

Figure 3-2 | 암호키의 생명주기

3.2.7 중앙화된 제어와 감사

암호키는 암호키 보안정책을 정하고, 암호키의 생명주기 동안 발생하는 모든 과정이 보안정책에 따라 이루어져야 한다. 보안정책은 필요한 여러 곳에 배포될 수 있지만, 보안정책을 생성하는 것은 하나의 신뢰주체만 수행하여야 한다. 보안정책이 여러 주체에 의해서 생성되거나 수정되면 보안정책 간에 상충하는 요소를 감지하기 어렵거나, 상충 요소를 감지하더라도 수정하기 어렵다. 부득이하게 보안정책을 생성하는 주체를 추가해야 한다면, 중앙의 보안정책 관리를 따르는 하위의 보안정책 관리를 만들어야 한다. 이 방법은 암호키의 관리를 일원화하여 잘못된 보안정책에 의해 시스템 전체의 보안 수준이 저하되는 것을 방지할 수 있다. 또한, 암호키의 생명주기 동안 발생하는 모든 과정이 보안정책에 따라 올바르게 진행되었는지를 감사할 수 있어야 한다. 감사 내용이 적절하고 보안정책에 부합하지 않는 점이 없음을 최종적으로 판단하는 것도 중앙의 보안정책 관리 주체여야 한다.

3.2.8. 관련 법제와 표준의 준수

암호 알고리즘은 다양한 기기나 서버 시스템에서 사용되고 있다. 자동차, 의료기기, IoT기기, 산업제어 시스템 등의 용도에 따라서 사이버보안 관련 법제와 표준이 존재한다. 이들 법제와 표준이 암호 알고리즘의 종류와 암호키 길이를 명확하게 정의하는 경우도 있지만, 거의 모든 법제와 표준은 암호키의 사용과 관리에 대한 요건을 규정한다. 암호키의 사용과 관리에 대한 요건을 준수하였는가는 정량적인 평가로 이어질 수 있어서, 암호키의 관리는 해당 제품의 관련 법제와 표준에 부합하여야 한다.

4. 자동차 키 관리 체계

4.1 자동차 환경의 암호키

자동차 환경에서 암호키는 자동차 내부의 ECU가 사용하는 것 뿐만 아니라, 자동차와 연결되는 외부 기기나 백엔드 서버가 사용하는 암호키도 존재한다. 자동차 내부의 ECU가 사용하는 암호키는 자동차가 외부 통신을 위해 사용하는 암호키와 자동차의 내부 네트워크를 위해 사용하는 암호키가 있을 수 있다. 이와는 별개로, ECU의 시스템을 보호하기 위해 사용하는 암호키가 사용된다. 자동차의 외부 통신을 위해 사용되는 암호키는 외부 통신의 종류나 용도에 따라 복수의 암호키가 사용될 수 있다. 자동차의 내부 네트워크를 위해 사용되는 암호키도 자동차 내부 네트워크의 종류와 적용 구간에 따라 서로 다른 암호키가 동시에 사용될 수 있다.

Owner	Usages of Key	Key Type	
		Symmetric	Asymmetric
Network Server	TLS Handshaking		O
Code Signing Server	Security for OTA update		O
PKI Infra Server	Certificate Management		O
Security Operation Server	Authenticity & Confidentiality for Security Operation Server		O
Application Server	Security for Connected Service	△	O
ECU with Ext. Connectivity	TLS Handshaking (mutual TLS)		
ECU with Ext. Connectivity	Security for Connected Service	△	O
ECU	Master Key for Vehicle E/E System	△	O
ECU	Master Key for ECU System	△	O
ECU	Key for OnBoard Communication	O	△
ECU	System Security of ECU (secure boot, secure flash, secure access, …)	O	O
External Device	Master Key for Device System	△	O
External Device	Security for Connected Service	△	O
External Device	System Security of Device (secure boot, secure flash, secure access, …)	O	O

Figure 4-1 | 자동차 환경의 암호키 예시

위의 예시(Figure 4-1)는 자동차 환경에서 사용될 수 있는 다양한 암호키의 예를 보여준다. 암호키는 대칭키 방식이거나 비대칭키 방식으로 적용될 수 있다. 위의 예시에서 △표시는 해당 방식이 사용될 수 있으나, 권장되지 않는다는 의미이다. 예를 들어, 자동차와 외부 온라인 서비스 백엔드 간의 연결을 위한 인증에서 비대칭키 방식을 사용하는 것이 일반적이나 대칭키 방식의 인증을 사용하는 것도 가능하다는 것을 의미한다.

자동차 환경에서 사용되는 암호키의 종류가 다양하게 존재하지만, 모든 암호키를 자동차 제조사가 관리하지는 못한다. 자동차가 V2X 통신을 통해 전기차 충전, 디지털 키, 홈 IoT 제어 등의 서비스를 사용한다면, 이들 서비스와 사용되는 암호키는 자동차 제조사가 모든 것을 관리할 수는 없다. 한편, 범용의 자동차 진단 기기나 운전자의 스마트폰에 설치되는 암호키는 자동차 제조사의 관리 범주에 속하기도 한다.

4.2 자동차 키 관리의 요구사항

4.2.1 OEM에 의한 중앙 관리 (Centralized Control by OEM)

자동차 산업에서 완성차 제조사인 OEM의 역할이 매우 중요하다. 특히, 자동차 사이버보안을 규정한 UN Regulation 155는 자동차에 사이버보안을 적용하는 것에 대한 책임을 OEM에 부여하고 있다. 유럽을 비롯한 여러 주요 국가가 제정한 관련 법제들도 자동차의 사이버보안을 확보해야 하는 의무 주체를 OEM으로 규정하고 있다.

자동차 환경에서 사이버보안을 완벽하게 확보하는 것은 매우 어려운 일이며, 암호키를 관리하는 것도 매우 어려운 일이다. 모든 암호키를 OEM이 관리하면 키 관리 문제가 용이해질 수 있지만, 복잡한 자동차 산업 구조에서 모든 암호키를 OEM이 관리하는 것은 쉽지 않다. 부품 공급사가 관리하는 암호키가 있을 수 있고, 자동차와 연결되는 외부 기기를 제조하는 기기 제조사가 관리하는 암호키가 있을 수도 있다. 자동차에 직접, 혹은 간접적으로 자동차의 사이버보안에 영향을 미치는 암호키는 OEM이 수립한 암호키 보안정책에 부합하도록 운영하여 자동차의 사이버보안에 대한 의무 책임을 OEM이 완수할 수 있어야 한다.

4.2.2 암호키의 생명주기 관리(Key Lifecycle Management)

한 대의 자동차 내부에는 많은 ECU가 탑재되고, 하나의 ECU 안에도 복수 개의 암호키가 사용될 수 있다. OEM 입장에서는 엄청난 수의 암호키를 관리해야 하는 어려움이 있다. ECU 내부에 저장되는 암호키는 ECU의 생산 과정에서부터 관리되는 것이 바람직하지만, 이 역시도 쉬운 문제가 아니다. OEM은 많은 부품 공급사와 협력해야 하고 부품 공급사들은 기업의 규모와 보유한 보안 인프라의 수준에 따라 암호키를 관리할 수 있는 수준에 차이가 존재하기 때문에 일괄적인 암호키 관리를 적용하는 것은 어렵다. 자동차에서 사용되는 많은 암호키들의 생애주기를 관리하더라도 자동화하여야 한다. 자동차에서 사용되는 암호키의 수는 굉장히 많고, 각각의 암호키는 각각의 생애주기에서 서로 다른 상태에 놓여 있을 수 있다. 암호키의 생애주기 동안 발생하는 작업마다 사람의 수작업이나 관리자의 승인이 필요하다면 자동차에서 가용성을 확보하기 어렵다. 자동차의 암호키를 포함한 모든 부품은 항상 가용성이 확보되어 있어야만 운행에 문제가 발생하지 않기 때문이다.

4.2.3 암호키 사용의 세분성과 다양성 제공(Providing Granularity and Variety of Key usages)

자동차가 사용하는 암호키는 ECU와 용도에 따라 다양하다. 암호키가 사용되는 암호 알고리즘과 암호키의 크기도 다양하다. OEM은 다양하고 많은 암호키를 관리할 수 있도록 키 관리 체계를 갖춰야 한다. 같은 ECU가 같은 용도를 위해 암호키를 사용하는 경우라도, 용도를 세분화해야 하는 필요가 있다면 암호키를 구분해서 관리하는 것도 필요하다.

암호키는 관리의 대상이기도 하지만, 활용의 대상이기도 하다. 암호키가 활용되기 위해서는 암호 알고리즘과 암호키의 저장소가 필요하다. 자동차가 다양한 암호키를 용도별로 사용할 수 있기 위해서는 암호 알고리즘과 암호키 저장소를 위한 지원이 필요하다. 256 비트 길이의 암호키로 암호 알고리즘을 사용하는 ECU가 보안 강도를 높이기 위해 384 비트 길이의 암호키를 사용하기 위해서는 그에 맞춘 준비와 대응이 필요하다.

자동차의 생명주기는 IT 기기의 생명주기에 비해 아주 길다. 자동차의 긴 생명주기 동안에 치명적인 보안 취약점이 발견되어 암호 알고리즘이나 암호키의 사용을 조정해야 하는 가능성은 얼마든지 있을 수 있다.

4.2.4 유연한 보안정책(Flexible Security Policies)

암호키는 체계를 갖춰서 관리되어야 하고, 관리를 위해서는 보안정책이 있어야 한다. 자동차의 암호키 보안정책은 OEM이 중앙 관리를 하는 것이 올바르고, 자동차와 연결되는 외부 기기의 제조사나 온라인 서비스를 제공하는 주체도 자동차의 암호키 보안정책을 감안하여야 한다. 이러한 점은 기존의 IT 환경에서 필요한 암호키 보안정책과 다를 바 없다. 자동차 환경의 암호키 보안정책이 가져야 하는 중요한 것은 유연함이다.

자동차에서 사용되는 다양한 제어기와 다양한 용도에 사용되는 다양한 암호키를 관리하는 내용을 담기 위해 보안정책이 유연하여야 한다. 다른 유연함은 자동차의 긴 생명주기 때문이다. 자동차를 생산한 이후에 새로운 보안 취약점이 발견되었다고 생각해보자.

새로운 보안 취약점에 대처하기 위해 자동차의 암호키 보안정책을 갱신해야 한다. 보안 정책 안에 새로운 항목을 추가해야 하는데, 보안정책 내에 추가된 항목을 ECU가 대응할 수 있어야 한다. 보안정책의 유연함은 보안정책 자체의 유연함과 더불어 보안정책을 집행하는 ECU도 내부 시스템의 유연함을 갖춰야 한다.

4.2.5 암호키 사용에 대한 접근권한 관리(Access Rights Management toward Key Usages)

자동차의 내부는 많은 ECU들이 복잡하게 연결되어 있다. 하나의 ECU에는 다양한 암호키 사용 용도에 따라 많은 암호키가 저장된다. ECU에 저장되어 있는 암호키는 적합한 사용 용도에만 사용되어야 하고, 암호키 사용을 위해 접근하는 주체에 대한 접근제어도 필요하다. 암호키의 지정된 사용 용도를 벗어나 암호키를 사용하는 것은 보안에 심각한 문제를 일으킬 수 있고, 암호키에 접근하는 것이 허용되지 않은 주체가 암호키에 접근하는 것도 엄청난 문제를 일으키는 원인이 된다.

암호키에 접근하는 주체가 올바른 대상이고, 접근을 요청하는 암호키의 지정된 사용 용도와 암호키에 접근하는 주체가 사용하고자 하는 용도가 동일한지에 대한 확인도 필요하다. 임의의 암호키에 대한 접근권한은 보안정책으로 기술하여 관리할 수도 있다. 접근 권한 관리를 기반으로 접근제어를 실행하는 것은 OS 내의 권한 관리로 구현될 수 있거나, 응용 소프트웨어의 내부에 컴포넌트 형태로 구현될 수 있고, 그 외에도 다양한 방법이 있을 수 있다. ECU 내에 저장되는 암호키의 위치와 암호키를 활용하는 주체들의 종류에 따라 구현 방식을 선택할 수 있다.

4.2.6 암호키의 안전한 저장(Secure Storing Keys)

암호키를 안전하게 저장하는 것은 무엇보다 중요하다. ECU 내부에 암호키를 안전하게 저장하기 위해서 HSM(Hardware Security Module)을 사용하거나, ARM core 기반의 프로세스에서 TrustZone 기술을 사용할 수도 있다.

암호키 중에는 사전에 저장되어 있던 값이 아니라, 사전에 저장되어 있던 값들을 활용하여 새롭게 만들어지는 암호키도 있다. 비대칭키 기반의 키교환 프로토콜을 통해 생성되는 세션키가 이런 사례에 해당한다. 이러한 키들은 특별한 저장소가 존재하지 않지만 일정 시간 동안 지속적으로 사용되어야 하기 때문에 안전한 관리가 필요하다.

암호키의 안전한 저장이 ECU 내부 저장소에만 해당하는 것은 아니다. ECU의 생산 과정에서 복수의 ECU에 암호키를 설치하기 위해 암호키를 임시 저장하는 기기가 존재한다면, 해당 기기도 암호키를 안전하게 저장해야 하는 의무 대상에 해당한다. 자동차와 연결하는 외부기기들 안에도 다양한 암호키가 사용될 수 있어, 외부기기들의 암호키도 안전하게 저장되어야 한다.

4.2.7 암호키 사용의 모니터링과 감사(Monitoring & Auditing Key Usages)

자동차 내부에서 암호키가 어떻게 사용되는지 모니터링하는 것은 중요하지만, 자동차에 모니터링을 적용하는 것은 쉽지 않다. 자동차 내부에는 많은 ECU가 존재하지만, 모든 ECU가 모니터링 서버에 접속 가능한 것은 아니기 때문이다. 모든 ECU가 모니터링 서버에 연결될 수 있다고 하더라도, 자동차의 외부 네트워크 가용성 측면에서 효율적인 방법은 아니다. 자동차 내부의 암호키 사용 상태를 파악하기 위해, 자동차 내부 ECU들이 암호키 사용 상태를 전송하면 자동차 내부에서 수집하는 ECU가 필요하다. 자동차 내부의 암호키 사용 상태를 수집한 ECU는 텔레매틱스(telematics) ECU 등을 통해서 완성차 제조사의 모니터링 서버로 모니터링 정보를 전송할 수 있다. 모니터링 서버는 암호키의 사용 상태 정보들을 암호키 보안정책과 비교하여 암호키 사용이 적절하지 못한 사례를 탐지할 수 있다.

완성차 제조사는 암호키의 생애주기에서 발생하는 모든 사건들이 보안 정책에 위배되지 않고 적절히 이루어졌는지 검토하는 감사 체계를 구축하고 운영함으로써 자동차의 사이버보안이 적절하게 관리되고 있는지 확인할 수 있다.

4.2.8. 관련 규정의 준수(Compliance with Regulations)

자동차 사이버보안에서 UN R155는 중요한 규제이지만, UN R155가 자동차 사이버보안의 모든 측면을 모두 담고 있는 것은 아니다. 특히, 자동차가 외부와 연결하는 서비스에서는 해당 서비스에 특화된 사이버보안 규정이 존재할 때가 많이 있다. 자동차가 통신사의 이동통신 네트워크를 사용하는 텔레매틱스 서비스를 탑재하고 있다면, 자동차의 텔레매틱스 ECU는 셀룰러 네트워크 접근 암호키의 관리를 위해 통신사의 보안 가이드라인을 따라야 한다. 전기차가 PnC(Plug & Charge) 기반의 충전 서비스를 사용한다면, PnC 충전 서비스를 위한 사이버보안 기준을 만족해야 한다. 자동차가 차량 내 결제(In-Car Payment)를 제공한다면, 결제 서비스가 요구하는 사이버보안 기준을 만족해야 한다.

자동차에 외부 연결성이 많아지고, 소프트웨어의 비중과 데이터 활용의 중요성이 높아지면서 UN R155 외의 다양한 규제들을 참조하는 경우가 늘어나고 있다. CRA(Cyber Resilience Act), NIS(Network and Information System), RED(Radio Equipment Directive), Data Act 등의 유럽 규제들은 자동차에 특화하여 제정된 것은 아니지만, 자동차나 ECU에 영향을 미치고 있어서 충분한 고려가 이루어져야 한다.

4.2.9. 공급망 전반의 통합(Integration across Supply Chains)

자동차 산업은 다른 산업들에 비해서 복잡한 공급망을 가지고 있는 것으로 유명하다. 자동차를 완성하기 위해 필요한 부품들이 그만큼 많기 때문이다. 공급망의 일부 과정에서 작은 보안 취약점이라도 포함되면 완성된 자동차도 보안 취약점을 가질 수 밖에 없어서, 자동차의 사이버보안 관리를 공급망과 함께 관리되어야 한다. 암호키 관리는 사이버보안의 가장 핵심적인 요소들 중의 하나라서 공급망과 함께 관리되어야 한다. 어떤 부품 공급사가 생산한 ECU에 어떤 암호키가 어떤 방법으로 설치되어 사용되는지 관리되어야 한다. 암호키가 사용되는 과정에서 발견된 보안 문제는 ECU의 펌웨어 업데이트나 소프트웨어 업데이트로 보완되어야 한다.

자동차의 공급망은 수많은 부품들이 자동차로 조립되어 소비자에게 판매됨으로써 끝나지만, 암호키 관리는 그 이후에도 관리가 되어야 한다. 자동차를 운행하는 소비자가 사용하는 서비스들에 따라 새로운 암호키가 설치되고 갱신될 수 있기 때문이다. 완성차 제조사는 소비자가 자동차를 운행하는 동안에 발생하는 암호키의 생명주기를 모니터링하여 공급망의 연장선에서 관리하여야 한다.

4.3 키 관리 흐름

차량 내 ECU에서 사용되는 암호키는 암호키를 초기 설정하는 단계, 서비스에 암호키를 사용하기 위한 준비 단계와 서비스에 암호키를 사용하는 단계로 구분할 수 있다. 이 모든 과정은 암호키의 생애주기 관리 안에서 이루어져야 하고, 수립된 보안정책을 따라야 한다.

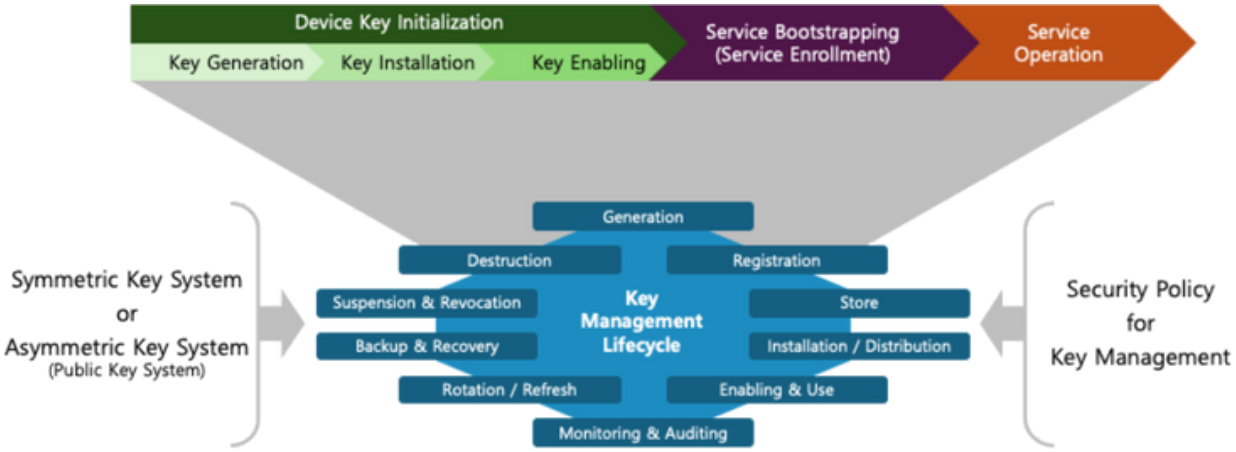

Figure 4-2 | 자동차 환경의 키 관리 흐름

4.3.1 Device Key Initialization

부품 공급사가 ECU의 하드웨어를 생산하고, 펌웨어와 소프트웨어를 설치 완료하더라도 ECU는 내장된 암호 알고리즘을 구동하기 위한 준비가 되어 있지 않다. ECU가 사용할 암호키가 설치되어야 ECU는 다른 ECU와 연결하거나, 자동차 외부와 연결할 수 있다. ECU 내부의 시스템 보호를 위해 암호 알고리즘을 구동하더라도 암호키가 설치되어야 한다. 이 과정에서 설치되는 암호키는 ECU의 마스터 키로서 ECU의 다른 암호키를 발급받기 위한 기반으로 사용될 수 있다. 혹은, 서비스를 위한 임시 암호키로 설치되고 Service Bootstrapping 과정에서 서비스를 위한 암호키로서 갱신할 수도 있다. 기기에 암호키를 설치하여 기기의 초기화를 마치는 이 과정은 키 생성, 키 설치, 키 활성화의 세부 단계로 구성되며, 4.4절에서 설명하는 6가지 모델들 중의 하나로 구현될 수 있다.

4.3.2 Service Bootstrapping

이 과정은 ECU나 자동차가 서비스를 본격적으로 사용하기 위해 필요한 준비 과정이다. 서비스를 제공하는 백엔드 서버에 ECU가 등록되는 과정일 수도 있다. ECU가 사용하는 서비스는 자동차의 ECU와 외부 개체를 연결하기 위해 인증 인프라를 서비스 별로 가지게 된다. Service Bootstrapping은 ECU를 서비스 인증 인프라에 등록함으로써 ECU가 가지고 있는 암호키를 사용하여 서비스를 사용할 수 있도록 해주거나, 서비스를 위해 사용하는 별도의 암호키를 추가로 발급 받을 수 있도록 해준다. 이를 위해서 ECU나 자동차는 서비스 인증 인프라의 인증을 통과해야 하는데, 이 과정은 ECU가 앞의 과정(4.3.1절)을 통해 사용 허가된 암호키를 활용하여 진행할 수 있다. 서비스 인증 인프라는 완성차 제조사의 도움을 받아 ECU가 보유한 암호키가 유효한지 검증하여야 한다.

4.3.3 Service Operation

ECU가 서비스 백엔드를 통해 서비스를 지속적으로 사용하는 단계이다. 이 단계에서 보안정책이 설정한 바에 따라 일정한 주기로 새로운 서비스 암호키로 갱신해야 할 수 있다. 서비스의 잘못된 사용으로 인해 암호키를 재발급 받는 과정이 발생할 수도 있다. 이러한 모든 동작은 암호키의 보안정책에 따라서 안전하게 이루어져야 한다.

4.4 암호키의 초기 설정 모델

ECU 내부에 암호키를 초기 설정하는 것에는 다양한 방법이 있을 수 있다. 아래의 6가지 방법은 암호키를 초기 설정하는 모든 방법을 대변하는 것이 아니라, 자동차 산업계에서 사용되는 대표적인 방법들을 설명한다. 다음의 모델들은 서로 다른 장점과 단점을 가지고 있고, 적용을 위해 필요한 환경이 다르기 때문에 완성차 제조사와 부품 공급사의 환경에 따라 선택해야 한다. 완성차 제조사 입장에서는 하나의 부품 공급사와만 협력하는 것이 아니기 때문에 협력하는 부품 공급사에 따라 서로 다른 모델을 채용하는 방법도 고려하여야 한다.

4.4.1 OEM-Generated

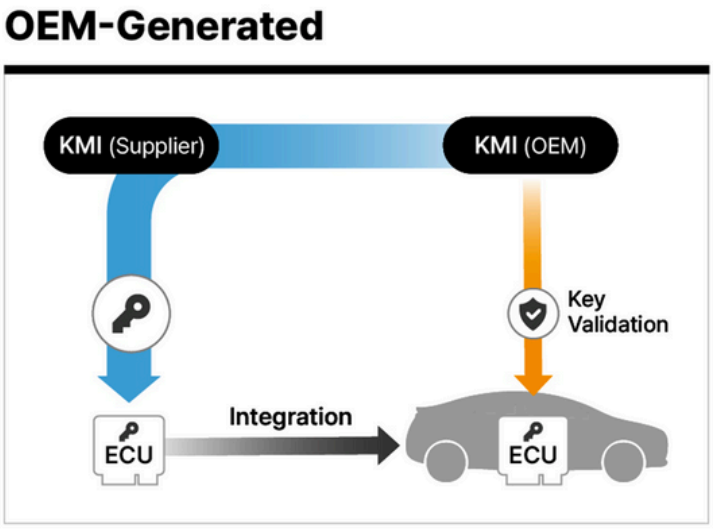

Figure 4-3 | OEM-Generated

부품 공급사는 완성차 제조사가 생성한 암호키를 받아 ECU를 생산하는 마지막 공정에서 암호키를 주입한다. 암호키가 주입된 ECU는 자동차의 부품으로 조립되고, 조립 과정에서 완성차 제조사는 부품 공급사에게 제공한 암호키가 ECU 내에 제대로 설치되어 있는지 검증한다.

이 모델에서 부품 공급사의 암호키 관리 능력이 부족해도 완성차 제조사에게 암호키를 받아서 설치할 수 있기 때문에 구축이 용이한 모델이다. 하지만, 암호키의 정보를 부품 공급사가 알고 있기 때문에 부품 공급사의 내부에서 암호키의 정보가 유출되면 큰 사고로 이어질 수 있다. 특히, 비대칭키 시스템을 사용하는 암호키인 개인키와 공개키 쌍을 이 모델의 방식으로 ECU에 설치하는 것은 비대칭키 시스템의 안전도를 크게 해칠 수 있다.

4.4.2 Supplier-Managed

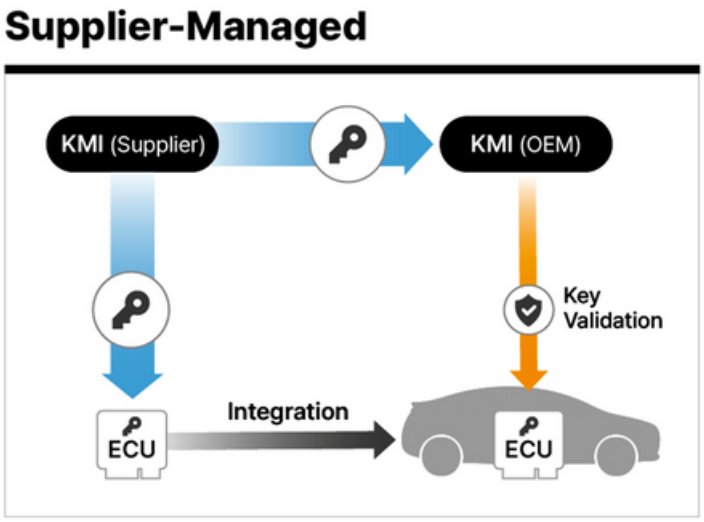

Figure 4-4 | Supplier-Managed

부품 공급사가 암호키를 생성하여 ECU를 생산하는 마지막 공정에서 암호키를 주입한다. 암호키가 주입된 ECU는 자동차의 부품으로 조립되고, 조립 과정에서 완성차 제조사는 부품 공급사로부터 받은 암호키 정보를 사용하여 ECU 내에 암호키가 정상적으로 설치되어 있는지 검증한다.

대칭키 암호시스템에서 사용되는 암호키를 이 모델로 설치하면 부품 공급사와 완성차 제조사가 대칭키인 암호키를 알게 되어 암호키 유출로 인한 보안 위험이 있을 수 있다. 비대칭키 암호시스템을 사용하는 경우, 부품 공급사는 개인키와 공개키 쌍을 생성하여 생성된 암호키를 ECU에 설치하고 공개키만 완성차 제조사에 전달하는 함으로써 완성차 제조사는 ECU 내부의 키를 검증할 수 있지만 개인키를 알 수는 없어 보안 강도를 높일 수 있다.

4.4.3 OEM-Refreshed

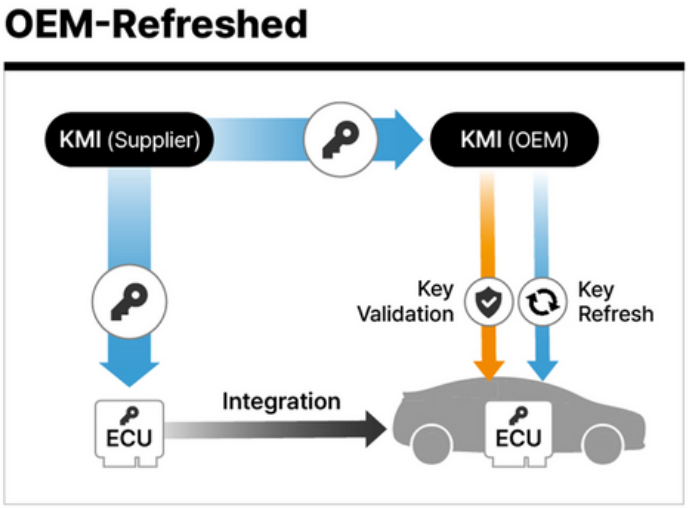

Figure 4-5 | OEM-Refreshed

앞의 Supplier-Managed 모델(4.4.2절)은 부품 공급사가 완성된 자동차 내부 ECU에 설치된 암호키에 대한 정보를 알고 있다는 단점이 있다. OEM이 자동차 암호키 관리를 모두 책임져야 하기 때문에 완성차 제조사 중심의 키 관리 체계를 확보하는데 부족함이 있다. 이를 해소하기 위해 완성차 제조사는 부품 공급사로부터 받은 암호키 정보를 사용하여 ECU 내에 암호키가 정상적으로 설치되어 있는지를 검증한 후 새로운 암호키를 생성하여 ECU에 설치한다.

부품 공급사가 ECU에 처음 설치한 암호키가 대칭키 혹은 비대칭키의 종류에 상관없이 ECU에 새롭게 설치하는 암호키는 대칭키와 비대칭키를 모두 지원 가능하다. 가령, 부품 공급사가 ECU에 대칭키를 설치해서 공급했더라도, 완성차 제조사는 설치된 대칭키를 검증한 후 비대칭키를 새롭게 설치할 수 있다. 반대로, 부품 공급사가 ECU에 비대칭키를 설치해서 공급했더라도, 완성차 제조사는 설치된 암호키의 공개키를 검증한 후 새로운 대칭키를 설치하는 것도 가능하다.

4.4.4 ECU-Generated at Vehicle Production

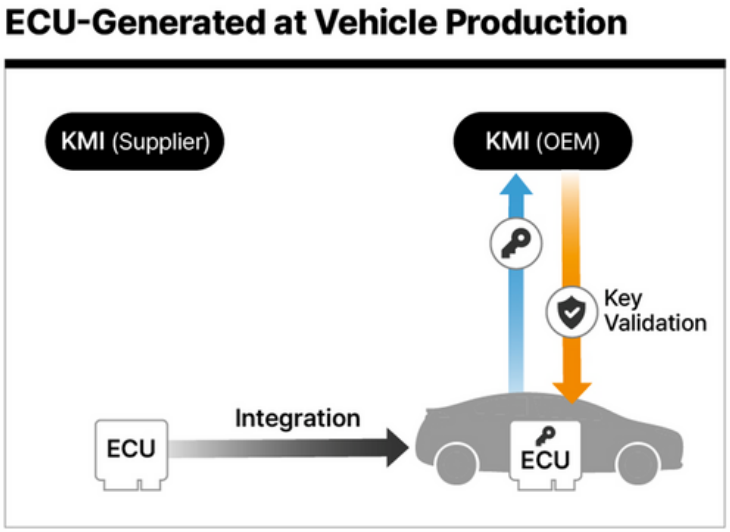

Figure 4-6 | ECU-Generated at Vehicle Production

앞의 세 모델은 부품 공급사가 암호키에 대한 정보를 획득할 수 있거나, 잠시 유효한 암호키 정보를 얻을 수 있어 보안 위험이 존재한다. 특히, 비대칭키 암호시스템에서 사용되는 암호키 중 비밀키를 관리하기에는 보안 강도를 낮추는 문제가 된다.

ECU가 비대칭키 시스템을 사용하는 경우, 부품 공급사가ECU를 생산하는 마지막 공정에서 ECU가 자체적으로 암호키를 생성하도록 명령 신호를 주입한다. 명령 신호를 받은 ECU는 내부에서 개인키와 공개키 쌍을 생성한다. 암호키가 생성된 ECU는 자동차의 부품으로 조립되고, 조립 과정에서 완성차 제조사는 ECU 가 공개키를 제공하도록 명령 신호를 주입하여 ECU의 공개키를 획득한다. ECU가 임의의 메시지에 대해서 정상적으로 전자서명을 생성할 수 있음을 검증하여, ECU가 공개키에 대응하는 개인키를 알고 있음을 확인할 수 있다.

이 모델을 사용하여 ECU에 저장된 개인키는 부품 공급사와 완성차 제조사에게 공유되지 않기 때문에 보안 강도를 높일 수 있다. 또한, 부품 공급사가 별도의 키 관리 체계를 갖추지 않아도 되기 때문에 경제적인 이점이 있다. 하지만, 부품 공급사는 생산한 ECU들 중의 하나가 문제를 일으키면 해당 ECU의 문제를 추적 관리할 수 없는 문제점이 있다.

암호키가 대칭키 시스템에서 사용되는 키인 경우에도 이 모델을 적용할 수 있으나, 보안 강도 측면에서 더 나은 방법은 아니다.

4.4.5 ECU-Generated at ECU Production

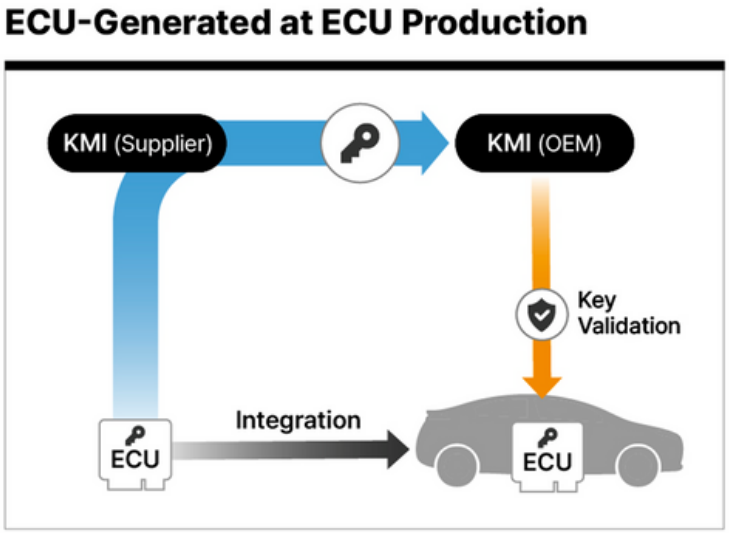

Figure 4-7 | ECU-Generated at ECU Production

ECU가 비대칭키 시스템을 사용하는 경우, 부품 공급사가 ECU를 생산하는 마지막 공정에서 ECU가 자체적으로 암호키를 생성하도록 명령 신호를 주입한다. 명령 신호를 받은 ECU는 내부에서 개인키와 공개키 쌍을 생성하고, 생성된 공개키를 부품 공급사의 키 관리 백엔드 서버에게 제공한다. 부품 공급사는 획득한 ECU 공개키를 완성차 제조사에게 공유하고, 암호키가 생성된 ECU는 자동차의 부품으로 조립된다. 조립 과정에서 완성차 제조사는 부품 공급사로부터 받은 공개키를 사용하여 ECU가 개인키를 보유하고 있음을 검증할 수 있다.

이 모델은 앞선 모델(4.4.4절)과 마찬가지로 ECU의 개인키가 ECU 외부로 공유되지 않아 보안 강도를 높일 수 있다. 앞선 모델은 부품 공급사가 별도의 키 관리 시스템을 보유하고 있지 않지만, 이 모델에서 부품 공급사가 자체의 키 관리 시스템을 갖추고 ECU의 정보와 ECU의 공개키를 관리하기 때문에 ECU의 암호키와 관련한 보안 문제가 발생할 경우 문제를 추적하여 해결하기에 용이한 장점을 가진다.

4.4.6 ECU-Generated with Autonomous Key Update

자동차 내부의 ECU는 암호키를 고정하여 사용하는 것이 아니라 보안정책에 따라 일정한 시간 간격마다 암호키를 갱신하여야 한다. ECU가 암호키를 갱신할 때, ECU는 완성차 제조사의 키 관리 서버와 연결할 필요가 있어 가용성에 문제가 발생할 수 있다. 차량 내 ECU가 암호키를 갱신해야 하는 모든 순간에 네트워크를 통해서 완성차 제조사의 키 관리 서버와 연결될 수 있다는 보장을 할 수 없기 때문이다.

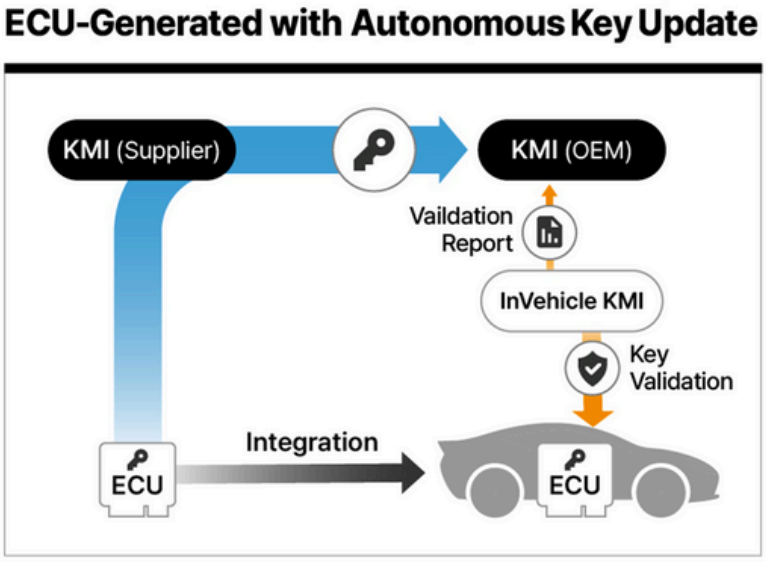

Figure 4-8 | ECU-Generated with Autonomous Key Update

이 모델은 모든 ECU의 키 관리를 담당할 수 있는 키 관리 서버를 자동차 내부에 둠으로써 ECU가 암호키를 갱신할 때 자동차 내부의 키 관리 서버와 연결하여 암호키의 갱신을 진행할 수 있다. 자동차 내부의 키 관리 서버는 네트워크를 통해 완성차 제조사의 키 관리 서버와 연결 가능한 시점에 ECU의 키 관리 정보를 연동함으로써 완성차 제조사가 자동차의 키 관리를 제어할 수 있도록 한다.

자동차 내부의 키 관리 서버는 별도의 ECU로 구성할 수도 있으나, 차량 내 서버의 응용 프로그램이나 게이트웨이의 부가 기능으로 구성할 수도 있다. 비대칭키 시스템에서 완성차 제조사의 키 관리 서버는 CA(Certificate Authority)의 역할을 수행하고, 자동차 내부의 키 관리 서버는 CA의 제어를 받는 Sub-CA의 역할을 수행함으로써 키 관리 체계를 완성할 수 있다.

4.4.7 암호키 초기 설정의 비교

설명한 6가지 모델은 ECU에 처음으로 암호키를 설정하는 방법으로서 대칭키 시스템이나 비대칭키 시스템을 사용하는 암호키에 적용할 수 있다. 뒤의3가지 모델(4.4.4절, 4.4.5절, 4.4.6절)은 암호키가 비대칭키 시스템의 키일 때 보안 강도를 더욱 강화할 수 있는 방법들이다. ECU의 암호키를 초기 설정하는 6가지 모델을 비교하면 아래와 같다.

Case	Key Generation	Key Install	Key Info. Retrieval	Key Validation	Key Refresh
1. OEM-Generated	OEM	Supplier	-	OEM	X
2. Supplier-Managed	Supplier	Supplier	-	OEM	X
3. OEM-Refreshed	Supplier	Supplier	-	OEM	OEM
4. ECU-Generated at Vehicle Production	ECU	-	OEM	OEM	X
5. ECU-Generated at ECU Production	ECU	-	Supplier	OEM	X
6. ECU-Generated with Autonomous Key Update	ECU	-	Vehicle	Vehicle	X

Figure 4-9 | 키 초기 설정의 모델 간 비교

자동차에는 다양한 종류의 ECU가 내장되고, ECU가 사용하는 암호키도 여러가지 종류이다. 완성차 제조사는 ECU를 제공하는 다양한 부품 공급사와 협력해야 하는데, 부품 공급사들의 사이버보안 역량도 차이가 있다. 자동차를 위한 키 관리를 체계적으로 구축하기 위해 완성차 제조사는 하나의 모델을 사용하면 효율적이긴 하나, 실제로는 그럴 수 없다. 다양한 부품 공급사와 암호키 초기 설정 모델을 여러 가지로 혼용해서 사용하여야 한다. 하지만, 여러 모델을 사용하더라도 암호키 관리가 체계적으로 이루어질 수 있도록 암호키의 보안정책을 세우고, 키 사용에 대한 모니터링과 감사를 마련해야 한다.

4.5 커넥티드 서비스의 연결

위의 4.4절에서 설명한 모델들에 의해서, 자동차 내의 ECU는 암호키를 보유하고 있고 완성차 제조사는 ECU의 암호키를 검증을 마쳤다. ECU가 ECU의 내부 시스템을 보호하기 위한 암호키는 별도의 추가 과정이 없어도 암호키를 사용할 수 있다. 하지만, ECU가 자동차 내부의 다른 ECU와 연결하여 암호키를 사용해야 한다면, 암호키를 다른 ECU와 사용할 수 있도록 ECU의 정보와 암호키 정보를 등록하여 다른 ECU들이 정보들을 확인할 수 있도록 해야 한다. ECU가 자동차 외부의 다른 기기나 온라인의 서비스 서버와 연결할 때에도 ECU의 정보, 혹은 자동차의 정보와 암호키 정보를 등록하여 외부 기기나 서비스 백엔드가 확인할 수 있도록 준비해야 한다. 이러한 일련의 과정을 Service Bootstrapping이라 한다.

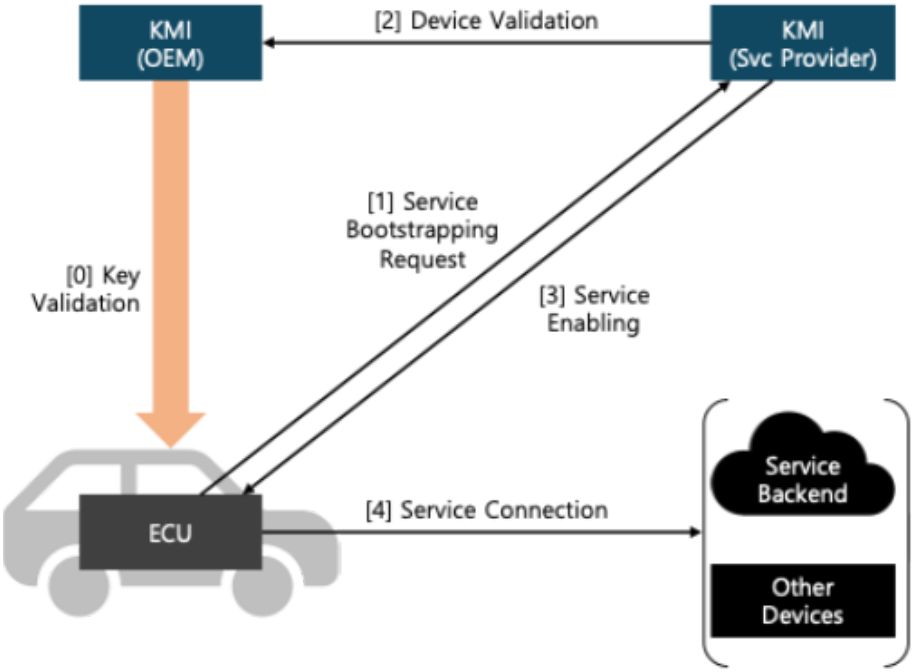

Figure 4-10 | Service Bootstrapping

위의 그림은 제어기가 자동차의 외부에 존재하는 기기나 서비스 백엔드에 ECU가 연결되는 서비스를 위해 ECU에 대해 Service Bootstrapping을 적용하는 개념을 간략하게 보여준다. 서비스가 안전하게 운영되기 위해 서비스를 위한 키 관리 인프라가 존재하고, 이것은 완성차 제조사가 ECU의 암호키 정보를 관리하는 인프라와 구분된다.

ECU가 외부 기기나 서비스 백엔드에 연결하기 위해 필요한 자격을 획득하기 위해, ECU는 서비스의 키 관리 인프라에 연결하여 서비스 등록을 요청한다. 이 과정에서 ECU는 4.4절의 초기 설정 모델에 의해서 보유하고 있는 암호키를 활용하여 정상적인 ECU로서의 자격을 증명한다 (Step [1]). 서비스 등록 요청을 받은 서비스의 키 관리 인프라는 ECU가 자격 증명을 위해 사용한 암호키가 유효한 암호키인지를 검증해야 한다 (Step [2]). 이 검증은 서비스의 키 관리 인프라가 완성차 제조사의 키 관리 인프라에게 ECU 암호키에 대한 유효성 검증을 요청하여 그 검증 결과를 수신함으로써 수행할 수 있다. 서비스 등록을 요청한 ECU가 정상적인 암호키를 사용하고 있다면, 서비스의 키 관리 인프라는 서비스 이용을 위해 사용할 수 있는 정보를 ECU에게 제공한다 (Step [3]). 이 과정에서 ECU에게 제공되는 정보는 ECU가 서비스 이용에 사용할 공개키 인증서일 수 있고, 서비스 접속을 위한 비밀 정보일 수 있다. 서비스 이용을 위해 사용할 수 있는 정보를 획득한 ECU는 획득한 정보를 사용하여 외부 기기 또는 서비스 백엔드와 연결할 수 있다 (Step [4]).

Service Bootstrapping 과정은 서비스의 종류나 서비스의 키 관리 체계에 따라 다르지만, 외부 기기나 서비스 백엔드가 ECU를 인증할 수 있도록 필요한 정보를 서비스의 키 관리 인프라에 등록하는 것은 동일하다.

5. SDV와 키 관리

5.1 FoD(Feature on Demand)

SDV를 정의하는 여러가지 표현이 있고, SDV가 갖춰야 한다고 생각되는 여러가지 특징과 기능이 있다. 그 중에서 중요한 기능이 FoD이다. 기존의 자동차가 소비자에게 제공할 수 있는 Feature들이 고정되어 있었지만, SDV는 소비자가 필요로 하는 Feature를 추후에 제공할 수 있다. 여기서 '추후에 제공한다'는 것은 완성차 제조사가 자동차 판매와 함께 소비자에게 제공하기로 약속한 기능을 자동차를 판매하는 시점 보다 후에 제공하는 것을 의미하는 것이 아니다. 소비자의 구매 완료 이후에 소비자의 별도 선택에 의해 기능을 추가할 수 있음을 의미한다. 이러한 형태의 구매는 스마트폰의 모바일 앱 시장에서 익숙하다. 소비자는 스마트폰을 구매한 이후에 본인이 원하는 모바일 앱을 선택적으로 설치할 수 있다.

SDV인 자동차를 구매한 소비자는 원하는 소프트웨어를 자동차에 설치함으로써 원하는 기능을 자동차에 적용할 수 있다. 이 과정에서 설치된 소프트웨어는 별도의 암호키를 필요할 수 있다. FoD 형태로 설치된 소프트웨어가 사용하는 암호키는 자동차의 제조 과정에서 설치되는 키와 달리 소프트웨어 제공사의 요청에 의해 설치되는 암호키이다. 이 암호키도 안전하게 저장되고 안전하게 사용될 필요가 있다. 응용 소프트웨어가 사용하는 암호키가 ECU 내에서 안전하게 저장 및 사용되기 위해서, ECU는 응용 소프트웨어에게 적절한 API(Application Program Interface)를 제공해주어야 한다. 응용 소프트웨어가 암호키의 안전한 저장과 사용을 위해 적합한 API를 적정하게 사용하는지를 감시함으로써 시스템의 보안성을 관리하여야 한다.

5.2 차량 내부의 ROT(ROOT OF TRUST)

자동차가 SDV로 진화하는 것은 자동차의 정체성 변화와 더불어, 자동차 내부의 전장 아키텍처의 진화를 수반한다. 기존의 전장 아키텍처가 수평적 분산 구조였다면, SDV에 적합한 구조는 Zonal Architecture 또는 중앙 집중형 아키텍처이다. 중앙의 자동차 서버(Vehicle Server)가 자율주행을 비롯한 AI 처리를 담당하고, 자동차 어플리케이션 소프트웨어를 구동하는 환경을 제공한다. 자동차의 각 구역을 담당하는 Zone Controller는 주변의 ECU들을 관리하며 중앙의 자동차 서버를 보조한다. 자동차 서버와 Zone Controller들 간에는 고속 통신이 필요한 뿐만 아니라, 안전한 통신이 요구된다. Zone Controller가 다른 Zone Controller와 안전한 통신을 위해 암호키를 사용하는데, 암호키는 일정한 일정한 주기로 갱신을 해야 할 뿐만 아니라 통신 채널에 보안 위험이 탐지될 때에도 암호키를 갱신하는 것이 바람직하다.

Zone Controller가 암호키를 갱신하기 위해 완성차 제조사의 키 관리 백엔드 서버에 해야만 하는 것은 매우 비효율적이다. 암호키를 갱신해야 할 때에 자동차가 완성차 제조사의 온라인 서버와 통신이 가능하다고 보장하는 것은 어렵기 때문이다. 자동차의 주행을 위한 가용성을 확보하기 위해 키 관리의 가용성을 확보하는 것이 중요하고, 이 문제는 자동차 내부에서 독립적으로 동작할 수 있는 키 관리 인프라를 구성하는 것으로 해결할 수 있다.

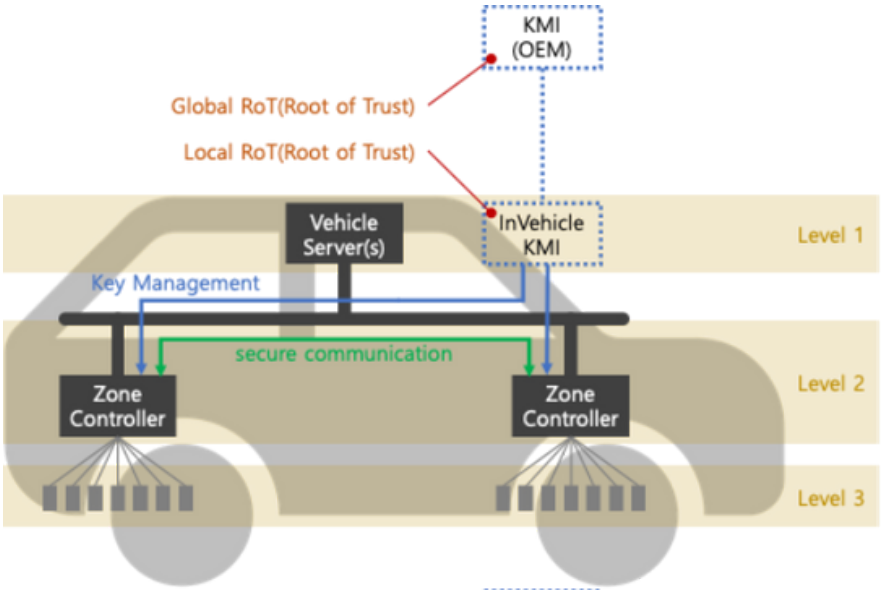

Figure 5-1 | InVehicle KMI(Key Management Infra)

자동차 내부의 키 관리 인프라는 자동차의 Zone Controller 뿐만 아니라 ECU들을 대상으로 키 관리를 제공할 수 있다. Zone Controller나 ECU는 자동차의 외부 통신을 사용하지 않고 내부 통신만으로 키 관리 인프라를 이용할 수 있다. 이로써 자동차 내부의 키 관리에 대한 가용성을 확보하고, 자동차 운행의 가용성을 확보할 수 있다. 자동차 내부 키 관리 인프라는 자동차 내부의 전체 구성 개체(Zone Controller, ECU 등)들 간 신뢰를 제공하는 근간이 되기 때문에 RoT(Root of Trust)가 된다. 자동차 내의 구성 개체 A가 구성 개체 B를 신뢰할 수 있는 것은 구성 개체 A와 구성 개체 B가 공통적으로 RoT를 신뢰하여 가능하다. 내가 가진 신분증을 발행한 기관이 다른 사람에게 발급한 신분증을 통해 상대의 신원을 파악할 수 있는 까닭과 같다. 나의 신분증을 발행한 기관을 믿기 때문에 내가 믿는 기관이 발행한 다른 사람의 신분증도 믿을 수 있는 것이다.

자동차의 내부 키 관리 인프라는 자동차 내부에서 독립적으로 동작함으로써 자동차 내부의 사이버보안이 지속가능한 유연함을 가질 수 있도록 한다. 독립적으로 동작하지만, 완성차 제조사의 키 관리 백엔드 시스템과 연동하기 때문에 완성차 제조사는 Global RoT(Root of Trust)로서 자동차의 내부 키 관리 인프라를 통해 자동차 내부의 보안 상태와 키 관리 상태를 관리할 수 있다.

자동차 내부 키 관리 인프라의 물리적인 위치는 독립된 ECU 형태일 수 있거나, 자동차 서버의 응용 프로그램 형태이거나, 중앙 네트워크의 네트워크 관리 제어기에 존재하는 응용 프로그램 형태일 수도 있다. 자동차 내부 키 관리 인프라를 Local RoT(Root of Trust)로 삼아, 자동차 서버, Zone Controller, ECU 등으로 계층(Level)적인 보안을 확보할 수 있다. 이러한 계층적 구조는 SDV가 복잡한 소프트웨어 구성과 다양한 통신 프로토콜을 수용할 수 있는 구조적 유연함을 제공한다.

6. 결론

보안 기능을 구현하는데 있어서 암호 알고리즘은 중요한 역할을 한다. 암호 알고리즘은 표준이어서 공격자를 포함한 모든 사람들에게 알려져 있기 때문에 암호 알고리즘의 보안성은 암호 알고리즘의 암호키를 어떻게 관리하는가에 달려 있다. IT 환경에서 키 관리의 중요성을 오랫동안 잘 알려져 온 주제이다. 하지만, 자동차 분야에서 키 관리는 초기 상태라 할 수 있다.

자동차 산업에서 요구되는 키 관리는 IT 환경의 키 관리와 유사한 점도 있으나, 자동차에 특화된 측면도 존재한다. 자동차 산업은 일반적인 IT 환경 보다 복잡한 산업 구조를 가지고 있다. 제품으로서 자동차는 다른 제품들 보다 복잡한 구조를 가지고 있고 그 복잡도가 지속적으로 증가하고 있다. 자동차의 제품 수명은 다른 IT 제품들에 비해 절대적으로 길다. 자동차 키 관리는 자동차 산업의 복잡한 공급망에 부합할 수 있어야 하고, 자동차의 긴 제품 수명 동안 사이버보안을 유지하는 첨병이 되어야 한다.


AUTOCRYPT