DOI:10.19651/j.cnki.emt.2107445

车载控制器 FOTA 固件安全多重校验方案*

武 恪[1,2] 李超超[1,2] 杨兴达[1] 方 菱[1]
(1.中国科学院合肥物质科学研究院 合肥 230031;2.中国科学技术大学 合肥 230026)

摘 要: 固件空中升级(FOTA)是一种利用无线通信实现电子控制单元软件升级的技术。FOTA 在汽车电子控制器中的大规模应用,使得汽车控制系统面临来自于公共网络中的安全威胁日益增多,而目前主流的 FOTA 方案着重关注固件从服务端到汽车端的远程传输,车内固件的安全性处理仍是薄弱环节。本文提出一种固件安全多重校验方案,服务端通过基于 ECC 的数字签名算法签名固件得到两个校验码,分别用于远程传输以及车内处理流程中的固件完整性和身份验证,以保证 FOTA 全流程的安全。实验结果表明所提的方案可以很好的识别固件远程传输和存储时篡改的风险,增加的时间成本仅约为 5%,同时与使用 RSA 算法实现的方案相比,同等安全条件下,还具有验签速度快、占用存储资源少等优点。

关键词: 固件空中在线升级;数字签名;身份验证;固件完整性
中图分类号: TP309 **文献标识码:** A **国家标准学科分类代码:** 520.1060

Multiple check scheme of the security of vehicle-mounted controller FOTA

Wu Ke[1,2] Li Chaochao[1,2] Yang Xingda[1] Fang Ling[1]
(1. Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei 230031, China;
2. University of Science and Technology of China, Hefei 230026, China)

Abstract: Firmware Over-The-Air (FOTA) is a technology which uses wireless communication to upgrade the software of electronic control unit. FOTA has been applied in a large scale in automobile electronic control devices, as a result, the vehicle control system is faced with increasing security threats from the public network. However, the present mainstream FOTA schemes merely focus on the remote transmission from the server side to the vehicle side, and the security handling of the in-vehicle firmware is still a weak link. This thesis proposes a multi-check scheme of firmware security, the server signs the firmware through the Elliptic Curve Cryptography-based digital signature algorithm, so as to get two check codes, which are respectively used for remote transmission and check of firmware integrity and identity in-vehicle handling process, thereby ensuring full-process security of FOTA. The experimental results show that the scheme proposed in this thesis can well identify the risk of tampering during remote transmission and storage of firmware, and the time cost increased is only about 5%. Meanwhile, compared with the scheme realized by using RSA, this scheme is advantaged by fast speed of signature verification and small occupancy rate of resources.

Keywords: firmware over-the-air; digital signature; identity; firmware integrity

0 引 言

随着信息技术与汽车产业的不断融合,网络互联和智能化已成为汽车产业发展的必然趋势[1]。汽车固件空中升级(firmware over-the-air,FOTA)是通过移动通信接口,对汽车电子控制单元(electronic control unit,ECU)固件进行空中升级优化,从而无需更换硬件和人工干预,即可完成性能优化和功能升级的技术[2]。FOTA 能解决低成本软件故障、软件风险应急响应、漏洞修复等潜在问题,同时也在新功能导入和用户体验感提升等方面发挥重要作用[3-4]。FOTA 在给汽车软件升级带来便利的同时,也使得汽车暴露在公共网络中,既增加了额外的攻击面,也增加了恶意软件从远程站点感染车辆的机会,其中以信息篡改、病毒入侵、恶意代码植入等手段对网联汽车进行网络攻击而引发的汽车安全问题日益严峻[5]。

具体来说,在 FOTA 升级过程中固件的处理需要经历

收稿日期:2021-07-30
* 基金项目:安徽省重点研究与开发计划项目(202004a05020041)资助

车外网络传输、车内存储、转发及安装等多个过程，每个过程都可能存在攻击的窗口。如在车载终端下载固件包的传输流程中，攻击者可利用网络攻击手段（例如中间人攻击），将篡改伪造的升级包发送给车载终端。在终端固件包安装的过程中，攻击者还可能通过替换存储部件的方式将固件包替换，如果在终端升级流程中同时缺少验证机制或者现有的验证机制覆盖不全面，那么被篡改的升级包即可顺利完成升级流程，攻击者将达到篡改系统、植入后门的目的。

近几年针对联网汽车的安全攻击事件频发，2015 年攻击者先利用 Linux 系统漏洞对克莱斯勒的 Jeep 汽车发起攻击[6]，劫持了该车的车载信息娱乐系统，获得了远程访问汽车的能力。随后又对汽车控制器局域网络（controller area network，CAN）控制器的固件进行了逆向分析，破解了车内 CAN 总线的通讯协议和数据格式，并对固件进行了修改，由于该控制器固件更新机制中缺乏固件的完整性和真实性验证措施，最终成功利用这一漏洞向控制器远程发送恶意软件并将其成功的安装，控制了该车的 CAN 总线，之后攻击者远程向汽车发送控制指令，启动了车上的各种功能，包括减速、关闭发动机、制动或让制动失灵，严重威胁司乘人员的生命安全。该事件导致克莱斯勒公司不得不召回多达 140 万辆相关型号的汽车，造成的经济损失高达数亿美元。

针对 FOTA 升级过程中固件的安全防范机制，学者们进行了一定的研究，文献[7]中提出了一种基于哈希函数的智能汽车安全无线固件更新协议（SFOTA），能够确保汽车下载固件的完整性和真实性，但固件在被安装到只读存储器（read-only memory，ROM）之前缺乏防篡改校验机制，固件仍有可能会被攻击者篡改。文献[8]中提出了一种 ECU 更新框架，利用硬件虚拟化技术，分别在控制系统和功能系统中对下载的固件进行验证。该框架能够在固件安装 ECU ROM 之前校验其完整性，但无法验证固件的来源，即无法验证固件的真实性。文献[9]提出了一种基于区块链的去中心化智能车辆软件升级架构，该架构能够确保升级流程中软件包的完整性和真实性，但该方案对硬件平台资源（包括内存和计算能力）要求较高，在汽车领域应用存在局限性。文献[10]中提出在随机时间间隔内发送两次同分组软件包的方式，作为服务器与汽车间传输过程中软件完整性验证的依据，但该方法要求 ECU 内存足够大，至少需要两倍的应用内存，适用范围同样有限，此外该方法也无法验证固件的真实性。

为此本文提出了一种固件安全多重校验方案，该方案在服务端通过数字签名技术得到两个校验码，分别用于固件包从车外到车内的传输过程和固件包在车内处理流程的完整性和真实性校验。它既能确保车载端从可信的门户下载固件，也能确保只有正确的固件才被安装到 ECU 的 ROM 中，另外也能保证在非更新时引导加载系统仅加载运行合法的固件。从实验结果来看，该方案能够在 FOTA 各处理阶段有效的识别恶意软件，并及时中止升级流程，降低车辆被入侵的机率。同时，该方案使用公钥加密体制中的椭圆加密算法（elliptic curve cryptography，ECC）实现，与同样被广泛使用的 RSA（rivest-shamir-adleman，RSA）加密算法相比，同等安全条件下 ECC 的秘钥和生成的签名长度更短、计算速度更快、占用的存储资源更少，同时增加的升级时间成本仅约 5%。

1　FOTA 概述

本节简要描述典型的 FOTA 架构、升级流程，及在该流程中固件的安全校验需求。

1.1　FOTA 架构及升级原理

车载控制器 FOTA 升级系统包括远程服务端和汽车端两部分，如图 1 所示，主要由 4 个实体组成：软件供应商、原始设备制造商（original equipment manufacturer，OEM）、车载信息通讯单元（telematics BOX，T-BOX）和目标 ECU[11]，其中软件供应商和原始设备制造商位于远端。汽车为近端，包括 T-BOX 和众多车载 ECU。软件供应商主要向原始设备制造商提供 ECU 的技术维护和固件更新包的开发。随着各种新型技术在汽车领域的快速应用，汽车 ECU 也随之快速增加，目前一辆汽车中的控制器数量可能超过上百。因此一辆汽车的软件供应商也是众多的。OEM 作为汽车制造商，其承担车辆的软硬件售后和维护，它将作为汽车控制器固件升级的发起者。车载信息通讯单元作为汽车与外界网络交换数据的枢纽，使汽车既能上传设备状态信息，也能接收固件、指令等信息。目标 ECU 是需要进行固件更新的电子控制单元。

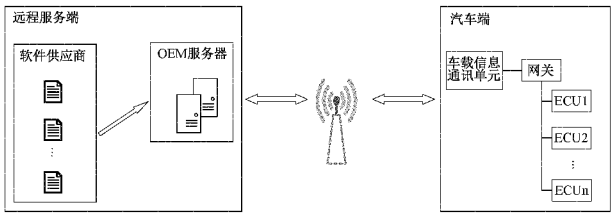

图 1　车载 FOTA 系统架构示意图

FOTA 升级的整体流程：当 ECU 需要固件升级时，软件供应商将新版固件及相关数据打包一并提供给 OEM；OEM 将其部署在自己的服务器中，启动固件 FOTA 升级服务，车载信息通讯单元下载固件包及其相关数据、存储并转发到目标 ECU；ECU 解包并将新版固件安装到 ROM 区域，安装成功后运行新版固件，结束 FOTA 升级服务。

1.2　固件包的校验需求

固件在 FOTA 升级服务中按照流程的走向可归纳为传输和存储两方面，其流经的实体如图 2 中带箭头虚线所示。

1）传输

FOTA 的固件传输由车外网传输和车内网传输两部分组成，即图 2 中的②和④；车外网传输是指车载通讯单元

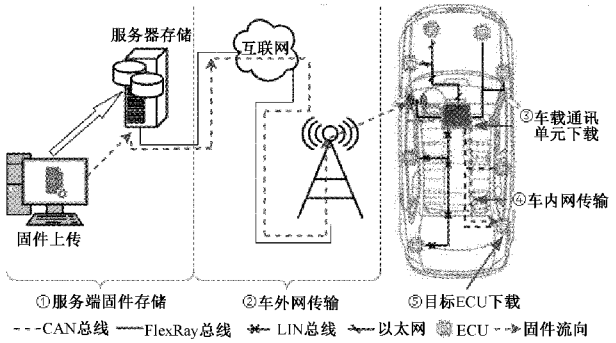

图 2　FOTA 中固件处理流向图

从 FOTA 服务器下载固件更新包流经的通路和各种通讯实体(如光缆、蜂窝基站等)。车内网传输是指目标 ECU 从车载通讯单元下载固件更新包流经的通路,主要是由 CAN、局域互联网络(local interconnect network,LIN)、FlexRay、车载以太网等总线组成车载局域网,图 2 虚线为使用 CAN 总线向目标 ECU 传输固件。

2)存储

FOTA 的存储主要由服务器存储、车载通讯单元存储和目标 ECU 存储 3 部分组成,即图 2 中的①③⑤。服务器存储是指 OEM 将固件更新包上传至 FOTA 服务器,供目标车辆下载存储和安装的过程。车载通讯单元是汽车与外界信息交互的通讯接口,其与 FOTA 控制平台通过查询的方式确定汽车中是否有 ECU 需要更新固件,当有 ECU 需要固件更新服务时,其需要将固件先缓存下来,再转发给目标 ECU。目标 ECU 的存储包括固件的下载和固件的运行。

在整个升级流程中固件有两种数据形式,即固件明文和固件密文。通常,为了防止攻击者通过监听、拦截等方式获取固件内容,固件需要被加密为密文,只有在安装和运行固件时才以明文形式存在。由于汽车功能的多样性,车载 ECU 的性能(计算能力和内存容量等)并不统一,针对不同性能的 ECU 需要实行不同的更新策略,如向低性能 ECU 提供固件更新服务时,由于 ECU 自身资源(如内存、计算能力等)的限制,为了加快升级的进度,减少系统处于升级状态的时间,一般将固件的解密操作放在车载信息通讯单元中,所以在此场景下,图 2 中的④⑤固件皆是以明文形式存在。

以上的各个升级流程阶段都可能存在攻击的窗口,例如在车外网和车内网传输时,使用中间人攻击,篡改固件包,若该固件被成功的安装,ECU 将无法执行正常的功能。此外,固件在车载信息通讯单元和 ECU 存储时,可以更换存储器暴力破解固件信息,篡改并将其替换掉。因此,为了预防升级过程中可能存在的篡改攻击,在 FOTA 系统中有以下固件包防篡改校验需求:

(1)车载信息通讯单元接收下载固件时,由于车外互联网是一个不可信的网络系统,固件在传输过程中存在被篡改的风险,所以需要在车载信息通讯单元中增加校验机制,防止非法数据源。

(2)目标 ECU 接收固件时,由于当前汽车车内网络协议(例如 CAN、LIN 等)在安全性(无数据源识别、无加密机制)设计方面存在缺陷,固件在不可信网络传输过程中存在被篡改的风险,所以需要在 ECU 接收固件时增加校验机制,防止非法数据源。

(3)固件在控制器存储过程中,同样存在被篡改的风险,例如在通讯单元中,攻击者可通过内存硬件的替换,将合法的固件替换为恶意的,目标 ECU 中同样存在这种篡改风险,因此也需要在新固件安装后,加载运行时验证其合法性。

2　固件安全多重校验方案

本节介绍一种固件安全多重校验方案,在新固件发布前使用数字签名技术计算出两个校验码,分别用于固件明文阶段和密文阶段的防篡改校验。该方案能有效抵御固件在 FOTA 服务传输和存储过程中受到的篡改攻击,并验证固件的完整性和真实性。

2.1　方案设计

第 2 节分析了由汽车、汽车制造商和软件供应商组成的分布式系统需要哪些固件防篡改校验需求。就目前而言,篡改验证对于 FOTA 尤其具有挑战性,在将固件刷入设备之前需要一个完善的验证机制来确保固件的合法性。

本文的方案由两部分组成:软件供应商和汽车制造商组成的服务端,以及车内众多控制器组成的汽车端。服务端生成固件校验码,用于汽车端的校验。

1)服务端生成校验码

固件明文由软件供应商开发并提供给 OEM 部署到 FOTA 服务平台。为了在安装阶段使 ECU 具备验证固件身份的能力,避免恶意软件被成功的安装在 ECU 中,汽车端的每个 ECU 都拥有自己的一对公私钥用于固件的身份验证,其中 ECU 的公钥 PuK_{ECU} 可以通过软件编码的形式将其存储在相关 ECU 中,而与之对应的 ECU 私钥 PrK_{ECU} 必须被软件供应商小心保存。在软件供应商开发完成固件后,软件供应商使用摘要算法对固件明文 $FW_{plaintext}$ 计算得到摘要值 MD_{ECU},并使用 ECU 的私钥 PrK_{ECU} 对 MD_{ECU} 加密得到数字签名 SIG_{ECU}。软件供应商将 ECU 的固件明文 $FW_{plaintext}$ 和明文的数字签名 SIG_{ECU} 一并提供给 OEM,OEM 将固件明文 $FW_{plaintext}$ 通过对称加密算法计算得到固件密文 $FW_{ciphertext}$,并使用摘要算法计算得到固件密文的摘要值 MD_{OEM},再使用自己的私钥 PrK_{OEM} 将密文摘要值 MD_{OEM} 加密得到 OEM 的数字签名 SIG_{OEM},OEM 将固件密文数字签名 SIG_{OEM}、固件密文 $FW_{ciphertext}$ 和固件明文的数字签名 SIG_{ECU} 存储于服务器中,供相关车辆升级时下载。

2)汽车端验证校验码

汽车端固件校验的流程主要由车载通讯单元下载阶段

校验、车载通讯单元固件转发阶段校验、ECU 下载缓存阶段校验、ECU 安装阶段校验和 ECU 运行阶段校验 5 部分组成。

算法 1. 车载通讯单元下载阶段固件校验算法

输入：PuK_{OEM}，SIG_{OEM}，$FW_{ciphertext}$

输出：固件校验结果

1. $Hsah(FW_{ciphertext})\rightarrow MD_1$；
2. $Decrypt(PuK_{OEM},SIG_{OEM})\rightarrow MD_{OEM}$；
3. IF $MD_1=MD_{OEM}$
4. 　固件校验结果＝进入固件转发阶段；
5. ELSE
6. 　固件校验结果＝中止升级；
7. END IF
8. return 固件校验结果.

在车载通讯单元下载阶段，固件包相关数据缓存完成后，计算得到固件密文的摘要 MD_1，再使用硬编码在其内存中的 OEM 的公钥 PuK_{OEM} 将下载的固件密文签名 SIG_{OEM} 解密，得到 OEM 计算的固件密文 MD_{OEM}。通过比较 MD_1 和 MD_{OEM} 的结果，以确定固件在互联网传输过程中是否被篡改。若相等，则固件是完整的，固件源是真实可信的，进入固件转发阶段。反之，则固件是非法或不完整的，中止升级服务。

算法 2. 车载通讯单元转发阶段固件校验算法

输入：PuK_{OEM}，PuK_{ECU}，SIG_{OEM}，$FW_{ciphertext}$

输出：$FW_{pliantext}$，固件校验结果

1. IF 立即转发固件包
2. 　IF 高性能 ECU
3. 　　向目标 ECU 转发 $FW_{ciphertext}$、SIG_{OEM} 和 SIG_{ECU}；
4. 　　固件校验结果＝进入 ECU 下载缓存阶段；
5. 　ELSE
6. 　　$Symmetric\_Decrypt(FW_{ciphertext})\rightarrow FW_{pliantext}$；
7. 　　向目标 ECU 转发固件明文 $FW_{pliantext}$ 和 SIG_{ECU}；
8. 　　固件校验结果＝进入 ECU 安装阶段；
9. 　END IF
10. ELSE 择机转发
11. 　$Hash(FW_{ciphertext})\rightarrow MD_2$；
12. 　$Decrypt(PuK_{OEM},SIG_{OEM})\rightarrow MD_{OEM}$；
13. 　　IF $MD_2=MD_{OEM}$
14. 　　IF 高性能 ECU
15. 　　向目标 ECU 转发 $FW_{ciphertext}$、SIG_{OEM} 和 SIG_{ECU}；
16. 　　固件校验结果＝进入 ECU 下载缓存阶段；
17. 　　ELSE
18. 　　$Decrypt(PrK_{ECU},FW_{ciphertext})\rightarrow FW_{pliantext}$；
19. 　　向目标 ECU 转发 $FW_{pliantext}$ 和 SIG_{ECU}；
20. 　　固件校验结果＝进入 ECU 固件安装阶段；
21. 　　END IF
22. 　ELSE 固件校验结果＝中止升级服务；
23. 　END IF
24. END IF
25. return $FW_{pliantext}$，固件校验结果.

车载通讯单元在完成固件下载和校验通过后，将进入固件转发阶段，由于目标 ECU 性能的差异，其固件升级策略也并不相同。对于性能高、硬件资源丰富的 ECU，一般是将固件缓存下来，之后再进行安装。而低性能 ECU 一般是边接收边安装固件。

车载通讯单元还需要根据汽车当前的运行状态选择转发策略，如果汽车当前的运行状态满足转发条件（电量、档位等）则立即将固件转发给目标 ECU，若转发的对象是高性能 ECU，则进入 ECU 固件下载阶段。若转发的对象是低性能 ECU，则需要将固件密文 $FW_{ciphertext}$ 通过对称加密算法解密得到固件明文 $FW_{plaintext}$，再将其转发给目标 ECU，且目标 ECU 进入安装阶段。

若汽车当前的运行状态不满足转发条件，则在条件满足后再进行固件的转发，其转发策略与立即转发的方式一样，但由于未立即转发固件，所以固件更新包有被篡改（如：硬件内存的替换等）的风险，因此在转发固件前，仍需要进行一次固件合法性校验，即通过计算固件得到固件摘要 MD_2，并通过与解密固件密文 SIG_{OEM} 得到的 MD_{OEM} 进行比较，若一致，则进入 ECU 下载缓存阶段，否则中止升级服务。

算法 3. 目标 ECU 下载缓存阶段固件校验算法

输入：PuK_{ECU}，$FW_{ciphertext}$

输出：固件校验结果

1. $Hash(FW_{ciphertext})\rightarrow MD_3$；
2. $Decrypt(PuK_{OEM},SIG_{OEM})\rightarrow MD_{OEM}$；
3. 　IF $MD_3=MD_{OEM}$
4. 　　进入固件安装阶段；
5. 　ELSE 中止升级服务；
6. 　END IF
7. return 固件校验结果.

将固件密文 $FW_{ciphertext}$ 转发给高性能 ECU 后，需要进行合法性和完整性校验，确保固件在车内网传输时未被篡改。其校验过程为：首先计算缓存固件的摘要 MD_3，并使用硬编码在 ECU 中的 OEM 公钥 PuK_{OEM} 解密固件密文 $FW_{ciphertext}$ 的数字签名 SIG_{OEM} 得到固件密文的摘要 MD_{OEM}，若 MD_3 与 MD_{OEM} 相同，则进入 ECU 安装阶段。反之，则中止 FOTA 升级服务。

算法 4. 目标 ECU 安装阶段固件校验算法

输入：PuK_{OEM}，SIG_{OEM}，$FW_{ciphertext}$

输出：固件校验结果

1. IF 高性能 ECU
2. 　$Symmetric\_Decrypt(FW_{ciphertext})\rightarrow FW_{pliantext}$；

3.　$Hash(FW_{pliantext})\rightarrow MD_4$;
4.　$Decrypt(PuK_{ECU},SIG_{ECU})\rightarrow MD_{ECU}$;
5.　IF $MD_4=MD_{ECU}$
6.　　安装固件到 ECU ROM;
7.　　固件校验结果=进入目标 ECU 运行阶段;
8.　ELSE 固件校验结果=终止升级服务;
9.　END IF
10. ELSE
11.　　$Hash(FW_{pliantext})\rightarrow MD_5$;
12.　　$Decrypt(PuK_{ECU},SIG_{ECU})\rightarrow MD_{ECU}$;
12.　　IF $MD_5=MD_{ECU}$
13.　　固件校验结果=进入固件转发阶段;
14.　　ELSE 中止升级服务;
15.　　END IF
16. return 固件校验结果.

在 ECU 固件安装阶段,ECU 从应用系统切换到引导加载程序,而在切换的过程中同样存在攻击的窗口,因此引导加载程序在启动加载功能前仍然需要固件校验,对于高性能 ECU,首先,将固件密文 $FW_{ciphertext}$ 通过对称加密算法解密为明文 $FW_{pliantext}$,通过对明文 $FW_{pliantext}$ 计算得到摘要 MD_4,使用硬编码在 ECU 中的公钥 PuK_{ECU} 将接收到的固件密文的数字签名 SIG_{ECU} 解密得到摘要 MD_{ECU},通过比较 MD_4 和 MD_{ECU} 的结果确定固件的合法性和完整性。若相同,则将固件安装到 ECU 的 ROM 中,进入 ECU 运行阶段。反之,则中止升级服务。

若是在低性能 ECU 中执行固件安装,首先,在安装过程中计算固件明文 $FW_{pliantext}$ 的摘要 MD_5,使用硬编码在 ECU 中的公钥 PuK_{ECU} 将接收到的固件密文的数字签名 SIG_{ECU} 解密得到摘要 MD_{ECU},通过比较 MD_5 和 MD_{ECU} 的结果确定固件的合法性和完整性。若结果一致,进入 ECU 运行阶段,反之,则中止升级服务。

算法 5. 目标 ECU 运行阶段固件校验算法
输入:PuK_{ECU},SIG_{ECU},$FW_{pliantext}$
输出:固件校验结果
1. $Hsah(FW_{pliantext})\rightarrow MD_6$;
2. $Decrypt(PuK_{ECU},SIG_{ECU})\rightarrow MD_{ECU}$;
3. IF $MD_6=MD_{ECU}$
4.　固件校验结果=加载运行新固件;
5. ELSE 固件校验结果=禁止运行新固件;
7. END IF
8. return 固件校验结果.

在完成以上阶段后,新版固件已经被安装在目标 ECU 中,FOTA 将进入 ECU 运行阶段,此后每次系统复位加载运行新系统前为有效抵御针对固件的篡改攻击,仍需要对系统固件进行合法性和完整性校验,即先对固件明文进行摘要计算,得到固件摘要 MD_6,再使用硬编码在 ECU 中的公钥 PuK_{ECU} 将接收到的固件密文的数字签名 SIG_{ECU} 解密得到摘要 MD_{ECU},比较 MD_6 和 MD_{ECU} 的结果确定固件的合法性和完整性。若结果一致,运行新版固件,反之,则禁止运行新固件。

在整个 FOTA 升级流程中,存在众多的攻击窗口,我们提出的多重固件校验方案覆盖了整个升级流程,能够有效抵御恶意软件的攻击。

2.2　方案实现

本节将介绍固件多重安全校验机制实现相关的内容,包括数字签名算法和摘要算法选择时需要注意的重点。

1)数字签名算法的选择与实现

数字签名算法包括两部分:用于计算哈希值的哈希算法和用于加密哈希值的非对称加密算法[12]。对于非对称加密算法一般是选择 RSA 或者 ECC[13],其签名过程需要使用消息来源者的私钥,因此它可以有效的抵御恶意的篡改攻击,同时也能保证消息发送方的不可抵赖。目前被大量运用的数字签名算法是基于 RSA 的,与 ECC 算法相比 RSA 在实现相同安全等级的情况下消耗更多的计算资源[14-15]。结合车载控制器资源有限的实际情况,并在保证足够安全等级的情况下,我们选择基于 ECC 的数字签名算法　椭圆曲线数字签名算法(elliptic curve digital signature algorithm,ECDSA),权衡到效率和安全,256 位秘钥长度足以满足车载领域数据安全等级的需要,因此本文中使用的是美国国家标准与技术研究院(national institute of standards and technology,NIST)推荐的 256 位椭圆曲线 ECDSA P-256。

2)哈希函数的选择与实现

摘要算法是一种对于任意长度的输入,其输出数据长度都为固定值的算法。常见的摘要算法有循环冗余算法(cyclic redundancy check,CRC)、消息摘要算法 5(message digest 5,MD5)以及安全散列算法[16](secure hash algorithm,SHA-2)。CRC 计算结构简单,但安全性不高。而 MD5 和 SHA-2 计算结构复杂,但安全性更高,由于 MD5 已被证实是不安全的,所以本文选用 SHA-2[13]。

SHA-2 散列函数系列由 NIST 标准化,作为 FIPS180-4 中安全散列标准的一部分[17],它主要由原始哈希函数 SHA-256、SHA-512 和基于原始哈希函数的变体组成。由于数字签名算法本文采用 256 位的秘钥,根据 ECDSA 算法实现的标准,本文需要选择一种生成 256 位哈希值的哈希函数。因此本文选择 SHA-256 将其作为该方案的哈希值计算函数。

3　实验验证与安全评估

为了验证固件多重安全校验方案的可行性以及其对系统整体性能的影响,本文建立由车载联网通讯单元 T-BOX、目标 ECU 电源管理系统(battery management system,BMS)以及 FOTA 升级服务器组成的车载控制器

FOTA 实验平台。T-BOX 和 BMS，它们的主控芯片均采用主频为 160 MHz 的 NXP MPC5748G，联网单元使用移远通讯的 EC20 4G 模组，其与主控芯片的通讯速率为 11.2 KB/s，T-BOX 与 BMS 之间通过 CAN 总线通讯，通讯速率为 500 Kbps(62.5 KB/s)。

3.1　性能评估

为了充分验证该方案对 FOTA 系统时效性的影响，我们评估升级时校验次数最多的情况下系统升级所耗费的时间，即 T-BOX 中校验 2 次，BMS 中校验 3 次，计算公式如式(1)所示，实验结果如表 1 所示。

$$T_{fota} = T_{download} + T_{transform} + T_{installation} + T_{verify} \quad (1)$$

式中：T_{fota} 表示 FOTA 升级的总时间(由于服务端计算机性能远强于汽车端，因此签名所消耗的时间可忽略)；$T_{download}$ 表示车载通讯单元 T-BOX 下载固件更新包所需的时间；$T_{transform}$ 表示 T-BOX 通过 CAN 通讯总线向 BMS 转发固件更新包所需的时间；$T_{installation}$ 表示 BMS 安装固件更新包所需的时间；T_{verify} 表示数字签名校验的时间，它包括 T-BOX 中校验固件的时间 T_{verify_Tbox} 和 BMS 中校验固件的时间 T_{verify_Bms}，耗时占比表示校验方案耗时占 FOTA 总耗时的百分比。

表 1　FOTA 系统多重校验耗时表

尺寸/KB	$T_{download}$ / s	$T_{transform}$ / s	$T_{installation}$ / s	T_{verify} / s	耗时占比/%
236	87	114	373	27	4.7
304	114	155	498	41	5.3
395	153	186	623	49	5.1
558	242	281	912	72	5.0

从表 1 中可知，随着固件数据量的增加，哈希值计算时间也随之增加，因此该方案校验的时间也会增加，但校验所用的总时间约占 FOTA 整体升级时间的 5%左右。

为了评估所提方案在资源消耗方面的性能，在保证同等安全等级需求下，我们评估了基于 ECC 的数字签名算法 ECDSA 与目前大量使用的基于 RSA 的数字签名算法在实验平台计算过程中所占内存的大小和验签时间，结果如表 2 所示。

表 2　基于 ECC 与 RSA 的数字签名资源消耗对比表

分类	安全等级	秘钥长度	验签时间/s	签名长度
ECC	128 位	256 位	0.78	256 位
RSA	128 位	3 072 位	3.12	2 048 位

由表 2 可知，基于 ECC 的数字签名算法 ECDSA 在内存占用和验签速度方面都优于目前大量使用的基于 RSA 的数字签名算法，而车载控制器具有计算能力弱、存储容量有限等特点，因此该实现方案在车载控制器领域具有适用性。

3.2　安全有效性验证

本文从 FOTA 的 5 个阶段对固件和校验码进行篡改，以验证该安全方案的有效性，包括篡改固件明文、固件密文、固件明文签名和固件密文签名 4 个对象，实验结果如表 3 所示。

表 3　校验方案有效性验证表

序号	篡改阶段	篡改对象	识别结果
1	T-BOX 下载	篡改固件密文	成功
2		篡改密文校验码	成功
3	T-BOX 转发	篡改固件密文	成功
4		篡改密文校验码	成功
5	BMS 下载	篡改固件密文	成功
6		篡改密文校验码	成功
7	BMS 安装	篡改固件明文	成功
8		篡改明文校验码	成功
9	BMS 运行	篡改固件明文	成功
10		篡改明文校验码	成功

从验证结果来看，该方案能够有效抵御恶意人员针对固件的篡改攻击，能够及时发现 FOTA 升级流程中存在的篡改，并中止升级流程，能有效降低恶意软件借助 FOTA 升级服务被成功安装在汽车 ECU 中的机率，对 FOTA 的安全研究具有实践意义。

4　结　　论

现代汽车智能化的程度越来越高，汽车制造商迫切需要 FOTA 技术更新车载控制器固件，然而复杂的网络环境使得 FOTA 本身能够成为攻击汽车的工具。本文从多个角度综合分析 FOTA 升级过程中可能存在的篡改风险，如固件远程传输时的篡改、固件存储时的篡改等，并针对这些潜在风险提出了一种基于 ECC 的数字签名算法的固件安全多重校验方案。该方案使用两个校验码即可在升级和运行过程中执行多次校验，将校验机制覆盖整个升级流程。最后通过实验验证多重校验方案对车载控制器软件系统升级的时效性影响很小，仅占整个升级流程时间的 5%左右。本文提出的多重校验方案能够有效防止恶意固件的安装，对 FOTA 安全研究和工程应用具有一定借鉴意义。下一步计划使用硬件加密模块替代软件加密算法，进一步提升方案的校验效率，减少对系统更新时间的影响，增强方案的时效性。

参考文献

[1]　张艳辉，徐坤，郑春花，等. 智能电动汽车信息感知技术研究进展[J]. 仪器仪表学报，2017，38(4)：794-805.

[2]　KARTHIK T，BROWN A，AWWAD S，et al. Uptane：Securing software updates for automobiles [C]. International Conference on Embedded Security in Car，

2016:1-11.

[3] ASOKAN N, NYMAN T, RATTANAVIPANON N, et al. ASSURED: Architecture for secure software update of realistic embedded devices [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2018, 37(11):2290-2300.

[4] 孔雪卉. 基于 C # 的 MCU 程序在线下载系统的设计[J]. 国外电子测量技术,2020,39(5):143-147.

[5] HE K X, WANG C Y, HAN Y Y, et al. Research on cyber security technology and test method of OTA for intelligent connected vehicle[C]. 2020 Inter-national Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), 2020: 194-198.

[6] ZHOU W, YAN J, PENG A N, et al. The effect of IoT new features on security and privacy: New threats, existing solutions, and challenges yet to be solved[J]. IEEE Internet of Things Journal, 2018, 6(2):1606-1616.

[7] MUHANMAD S I, HENDRIK S, YVES R, et al. Secure automotive on-board protocols: A case of over-the-air firmware updates[C]. International Workshop on Communication Technologies for Vehicles, Springer, Berlin, Heidelberg, 2011:224-238.

[8] ABDO A A A W, ZARINA S, MUHAMMAD A I. Framework for software tampering detection in embedded systems[C]. 2015 International Conference on Electrical Engineering and Informatics (ICEEI), IEEE, 2015:259-264.

[9] LUKAS K, PAVEL H, LUKAS V, et al. Software implementation of secure firmware update in IoT concept[J]. Advances in Electrical and Electronic Engineering, 2017, 15(4):626-632.

[10] STEGER M, BOANO C A, NIEDERMAYR T, et al. An efficient and secure automotive wireless software update framework [J]. IEEE Transactions on Industrial Informatics, 2017, 14(5):2181-2193.

[11] AKSHAY C, SUN W Q, AHMAD J, et al. Security enhancement of over-the-air update for connected vehicles[C]. International Conference on Wireless Algorithms, Systems, and Applications, 2018: 853-864.

[12] HALDER S, GHOSAL A, CONTI M. Secure over-the-air software updates in connected vehicles: A survey[J]. Computer Networks, 2020, DOI: 10.101-6/j.comnet.2020.107343.

[13] 刘英杰. 车联网中数据传输安全的关键技术研究[D]. 南京:东南大学,2019.

[14] 陈昱玮,刘毅力,马龙涛,等. 面向变电站的 AES 与 ECC 算法改进及混合加密研究[J]. 国外电子测量技术,2020,39(10):60-65.

[15] 陈思伟,高翠云,沈庆伟,等. 面向智能家居的生理参数密钥加密方法研究[J]. 电子测量与仪器学报,2021,35(3):173-180.

[16] 邓力,周新志. 一种改良安全机制的嵌入式远程升级系统的研究[J]. 电子测量技术,2017,40(8):207-211.

[17] 韩玮. 基于区块链的物联网身份认证算法研究[D]. 南昌:南昌大学,2020.

作者简介

武恪,硕士研究生,主要研究方向为车联网安全、嵌入式系统应用。

E-mail:wke@mail.ustc.edu.cn

李超超,硕士研究生,主要研究方向为嵌入式系统应用。

E-mail:lichaochao@mail.ustc.edu.cn

方菱(通信作者),副研究员,博士,主要研究方向为嵌入式系统测试、形式化方法验证。

E-mail:fangl@hfcas.ac.cn