智能网联车辆空中下载系统精益化设计

柳旭　周时莹　范玲玲　李长龙　周锐

（1.中国第一汽车股份有限公司研发总院，长春 130013；2.高端汽车集成与控制全国重点实验室，长春 130013）

【摘要】为提高现有远程升级系统的高并发连接能力、自动化程度及下载升级效率，利用云端微服务化、基于整车基线的自动升级路线策略、独立下载通道、轻量化远程参数修改等技术对空中下载（OTA）系统方案进行精益化设计，量产车型的实车功能及性能试验结果表明：该方案满足百万辆级车辆的并发使用需求，升级稳定性接近100%。

关键词：空中下载　软件基线　任务自动化　独立下载

中图分类号：U461　　**文献标志码：**A　　**DOI：** 10.20104/j.cnki.1674-6546.20220131

Lean Design of Intelligent Connected Vehicle OTA System

Liu Xu, Zhou Shiying, Fan Lingling, Li Changlong, Zhou Rui

(1. Global R&D Center, China FAW Corporation Limited, Changchun 130013; 2. National Key Laboratory of Advanced Vehicle Integration and Control, Changchun 130013)

【Abstract】In order to improve the high concurrency connectivity, degree of automation and efficiency of downloading upgrading, this article uses cloud-based microservice, automatic upgrade path strategy based on vehicle baseline, independent download channel and lightweight remote parameter correction to make lean design for Over-The-Air (OTA) system scheme. Function and performance test results of mass-produced vehicles show that this scheme satisfies the demand of concurrent use for vehicles with production capacity of one million unit, with upgrade stability nearly reaching 100%.

Key words: Over-The-Air (OTA), Software baseline, Task automation, Independent download

【引用格式】柳旭, 周时莹, 范玲玲, 等. 智能网联车辆空中下载系统精益化设计[J]. 汽车工程师, 2024(6): 1-7.

LIU X, ZHOU S Y, FAN L L, et al. Lean Design of Intelligent Connected Vehicle OTA System[J]. Automotive Engineer, 2024(6): 1-7.

1　前言

以自动驾驶为代表的车辆核心软件功能愈发复杂，面对巨额的开发成本投入，软件价值商业化势在必行[1-2]。基于远程软件升级、软件个性化定制等需求的空中下载（Over-The-Air，OTA）技术应运而生。

国内外车企针对企业级OTA系统、云端搭建、升级流程、信息安全、升级可靠性等方面已开展了大量的开发工作。文献[3]对某公司初代OTA系统的架构组成、升级策略、差分原理、实车测试用例及参数指标进行了阐述；文献[4]介绍了信息安全原理及在OTA下载过程中的具体应用流程；文献[5]介绍了目前国内外OTA相关法规制定情况及其具体要求。由此可见，汽车行业、企业及国家监管部门均在快速开展OTA技术研发及规范工作。

本文针对目前主流的整车OTA技术提出一种精益化设计方案，主要进行OTA平台系统的高并发、自动化任务发布、下载升级速率提升等方面研究，阐述OTA实际项目开发及使用过程中亟待解决的问题，并给出合理的设计方法和解决方案，最后验证方案的可执行性。

2　OTA架构演进

OTA技术是通过4G、5G、WIFI等网络介质从云端服务器下载车辆软件更新包并在车端完成软件替换更新的无线升级技术[6]，采用典型的客户端/服务端（Client/Server）架构，由车端、管端、云端3个部分组成。OTA云端系统及其外围服务器集群为车辆提供

全面的软件升级服务，涵盖车辆管理、软件包管理、升级流程协调等。车云链路确保通信过程的安全性和可靠性，车辆作为服务请求的终端接入这一系统，可精准地向特定的云端服务器发起升级服务的请求[7-8]。

2.1 OTA 系统基本硬件组成

OTA 系统的基本硬件组成如图 1 所示。云端系统硬件主要包括 OTA 应用服务器群、汽车远程服务提供商（Telematics Service Provider，TSP）车辆数据服务器、公钥基础设施（Public Key Infrastructure，PKI）证书服务器、内容分发网络（Content Delivery Network，CDN）第三方应用服务器[9]。车端系统硬件主要包括网络连接电子控制单元（Electronic Control Unit，ECU）、网关控制器、交互控制器等，各控制器之间通过车载总线，如以太网、CAN、CANFD 等进行数据及信号传输[10]。

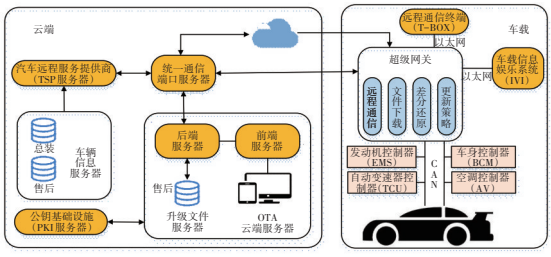

图 1　OTA 系统硬件组成

2.2 OTA 系统基本软件组成

OTA 云端系统按照云服务架构可拆解成如图 2 所示的模块：基础设施即服务（Infrastructure-as-a-Service，IaaS）平台层提供基本的系统级资源，包括 Kafka 类调度中心、各类型数据库、对象存储（Object Storage Service，OSS）、数据缓存等；平台即服务（Platform-as-a-Service，PaaS）层划分为核心业务服务、通用服务、安全服务 3 个模块，其中核心业务模块为 OTA 的主逻辑，可按功能划分为通信、下载、车辆管理、软件管理、升级管理、统计等服务；软件即服务（Software-as-a-Service，SaaS）应用层为面向专业人员的云端 Web 及向外可扩展的信息接口[11]。

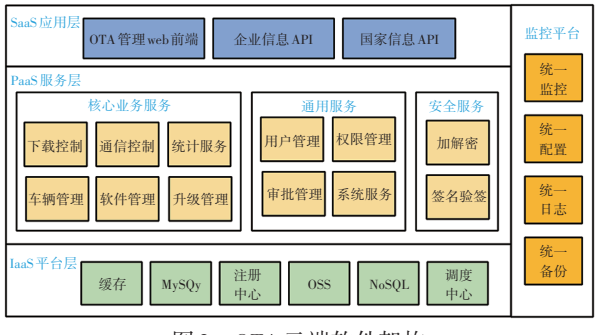

图 2　OTA 云端软件架构

OTA 车端按照模块化设计，包含以下基本组件：下载管理器组件（DMclient），主要功能为下载、验签、加密、安全认证、版本管理、差分还原；升级代理（Update Agent，UA）组件，用于收集并保存车辆状态信息，辅助 OTA 策略的实施[12]，该组件触发刷写脚本应用程序接口（Application Programming Interface，API）完成 ECU 的刷写脚本调用，获取 ECU 的升级进度和结果，同时收集整车各 ECU 的软件版本；人机交互代理（Human Machine Interface Agent，HMI Agent）组件用于处理整个升级过程中所有 OTA 相关状态的显示及用户操作信号的反馈；系统代理（System Agent）组件用于 OTA 升级过程中车辆电源控制、车辆状态判断、中央网关（Central Gateway，CGW）的睡眠接口调用以及车辆起动状态获取[13]。

同时，OTA 组件的正常运行依赖于部署节点的底层能力，包括刷写相关、条件相关、通用信息相关、交互相关的底层功能接口。车端软件架构如图 3 所示。

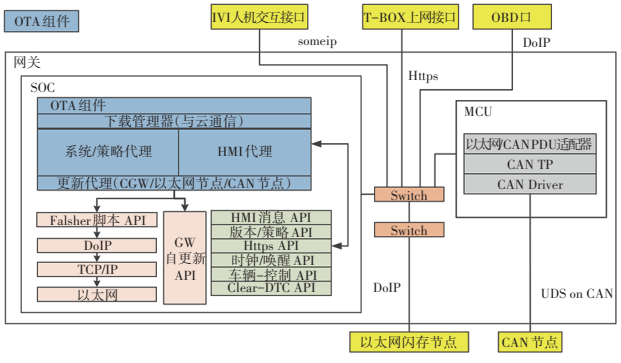

图 3　OTA 车端软件架构

3　OTA 系统精益化设计

OTA 精益化设计以现有硬件为基础，优化现有软件架构及功能逻辑，主要包括基于整车基线的自动任务引擎、独立下载技术和远程配置技术 3 个方面。

不同于基础 OTA 逻辑的单 ECU 软件版本管理的维度，车端整车软件版本管理组件用于整车版本收集及整车基线判断，云端软件包管理以整车软件为逻辑集合，利用独立下载技术实现大数据量级 ECU 直接访问 CDN 的方法，缩减原有的车内传输过程。同时，OTA 云端下发的整车软件包中携带 ECU 配置信息，可通过车端的配置模块实现 ECU 功能的开启和关闭。云端任务触发引擎按照整车基线设置预定路线，单个车辆终端上报状态后，自动确定及下发任务。OTA 精益化设计方案如图 4 所示。

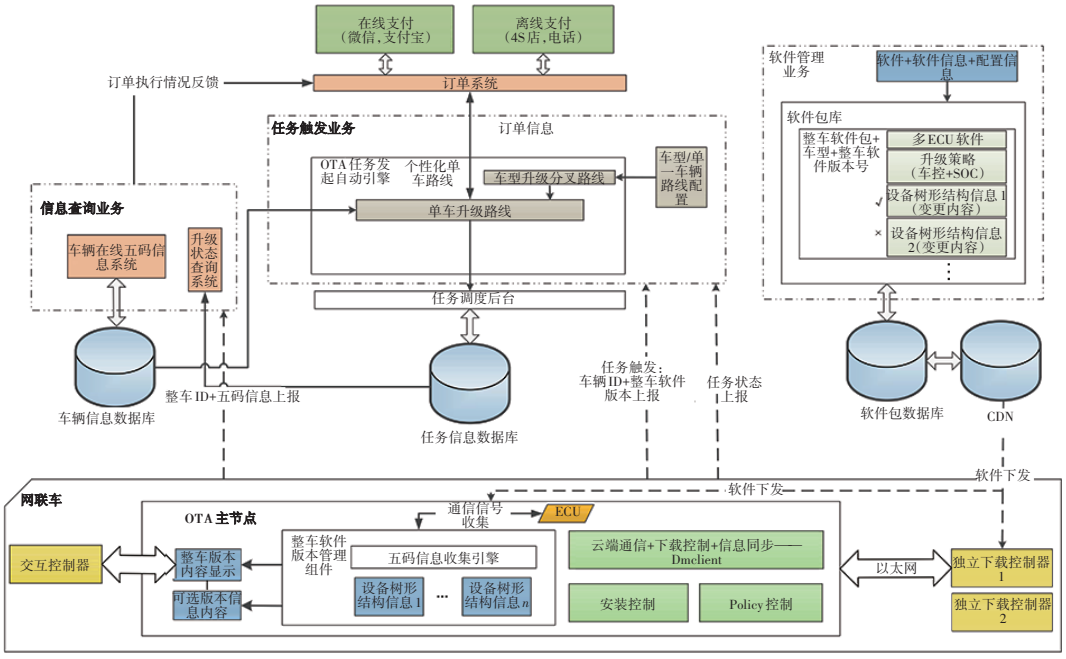

图4　OTA软件架构精益化设计方案

3.1　OTA云端微服务架构

单体式架构中多个服务通过共享同一数据库，将应用程序整合成一个单一的、不可分割的单元，因此所有的服务、组件和模块都是紧密耦合的，这将导致车辆并发能力偏弱、云端Web管理界面卡滞、云端部署任务及软件包上传时间过长等问题。同时，由代码依赖、数据共享等架构特性带来的强耦合性会导致系统功能变更困难。

微服务架构将单一应用程序划分为微小的服务，各服务之间互相协调、配合，为用户提供最终价值。各服务独立运行，并采用轻量级的通信机制互相协作，围绕具体业务进行构建，并且能够被单独分配到生产环境中，同时，各微服务拥有独立的数据库[14]。微服务核心功能机制如图5所示。

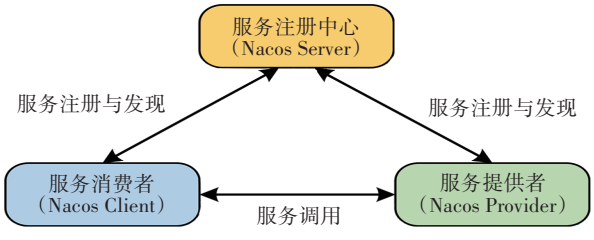

图5　按需申请的服务注册中心

3.2　基线控制及云端任务路线

用户较为关注汽车的整车级功能，但整车制造商（Original Equipment Manufacturer，OEM）更关注ECU的模块化设计，其基线存在控制器版本不受控、新功能通知滞后、企业运营及宣传通道受限等弊端。

3.2.1　整车版本定义

整车版本描述了整车ECU的软件状态，各ECU的软件状态通过软件版本号进行识别，将整车功能的不同状态通过唯一的整车版本号进行区分，如图6所示。整车版本的格式设置为AA OS P S.X.Y<.Z><-T>，各字段定义如表1所示。

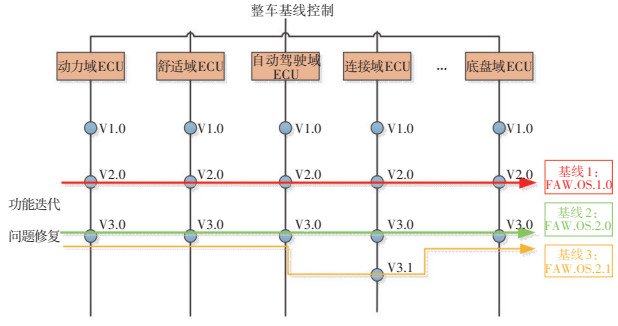

图6　整车版本基线管理

表1　整车版本字段解析

字段	解释
AA OS	商标名称
P	整车软件平台代号（Platform）
S	适配的实车类型（Specify） 包括：骡子车、OTS车、TTO车、量产车
X	主版本-功能新增
Y	次版本-功能小修改
<.Z>	子版本，热修复版本
<-T>	车型分支代号（Type）
示例	FAW OS X 2.1.1-E001

3.2.2 车端整车版本判断模块

ECU通过CAN报文按周期上报ECU硬件及软件版本信息，软件更新涉及软件版本号的变化，刷写流程执行完成后重启，新的软件版本号在第一帧发出。

整车硬件及软件判断依据以xml文本格式记录。针对不同的整车硬件版本，分别描述ECU标识、名称、版本号集合，基本结构如图7所示。整车硬件判断依据通过下线设备灌装入车内，当车辆存在硬件升级的场景时，售后部门重新灌装。整车软件判断依据伴随升级包一同下发到车辆。

```
<VehicleSoftwareVersionList>
    <Version>1.0</Version>
    <VehicleSoftwareVersions>
        <VehicleVSWV>v01.00.01</VehicleVSWV>
        <VHWV_Lists>
            <VHWV_List>
                <VHWV>VHWV001(dipei)</VHWV>
                <ECU_Versions>
                    <ECU_SoftwareVersion>
                        <ECUNAME>GW(联电网关低配)</ECUNAME>
                        <EDID>0x0000</EDID>
                        <Version>["v1.0", "v1.1"]</Version>
                    </ECU_SoftwareVersion>
                    <ECU_SoftwareVersion>
                        <ECUNAME>GW(自主网关低配)</ECUNAME>
                        <EDID>0x0010</EDID>
                        <Version>["v1.0", "v1.1"]</Version>
                    </ECU_SoftwareVersion>
                    <ECU_SoftwareVersion>
                        <ECUNAME>选装件1</ECUNAME>
                        <EDID>0x0100</EDID>
                        <Version>["v1.0", "v1.1"]</Version>
                    </ECU_SoftwareVersion>
                    <ECU_SoftwareVersion>
                        <ECUNAME>选装件2</ECUNAME>
                        <EDID>0x0200</EDID>
                        <Version>["v1.0", "v1.1"]</Version>
                    </ECU_SoftwareVersion>
                </ECU_Versions>
            </VHWV_List>
```

图7 整车软件版本判断依据描述文件结构

整车软件版本判断模块如图8所示，其工作机制为：各ECU通过通信报文发出软、硬件信息；车端版本管理组件收集该信息并存储；整车软件版本引擎按照设备树软件信息，判断存储的版本信息是否相同，若相同则完成整车版本匹配。匹配过程中，由于设备树软件信息较多，按照序号从大到小的顺序依次判断，匹配成功后则停止继续比对。

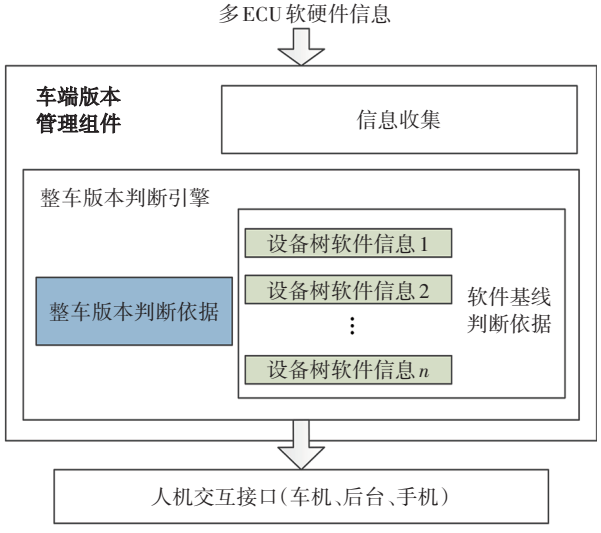

图8 整车软件判断模块

3.2.3 基于整车版本的云端任务路线

不同于繁琐的人工部署任务，基于整车版本的云端任务采用提前预设路线的方式，不同车辆识别码（Vehicle Identification Number，VIN）的车辆上报整车版本后，自动选择后续节点生成恰当的任务[15]。云端任务路线示例如图9所示。

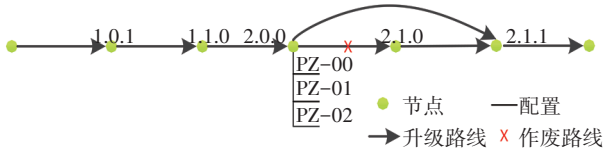

图9 云端任务路线

3.3 独立下载

车内车载远程通信终端（Telematics-BOX，T-BOX）、车载信息娱乐系统（In-Vehicle Infotainment，IVI）及高级驾驶辅助系统（Advanced Driving Assistance System，ADAS）等系统的升级包大小可达GB级，若OTA组件将升级包下载至主节点内，升级包需要从网关传输到目标设备，传输过程耗时较长。

独立下载技术通过将下载信息经由加密信道传输给目标端，目标端直接与云端通信的方式完成下载动作[16]，相关流程如图10所示。

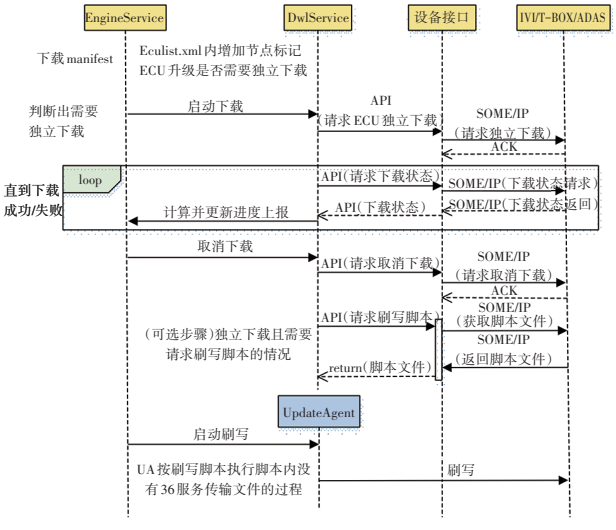

图10 独立下载交互流程

Engine Service和Dwl Service Ext为主节点中OTA组件的2个服务模块，分别负责任务管理和下载管理。独立下载方案中，Engine Service负责同步各软件包的下载相关信息，当判断升级包所属为独立下载ECU后，将下载信息同步给对应ECU。

IVI/T-BOX/ADAS收到下载信息后，直接通过下载通路连接对应服务器，完成数据下载，支持断点

续传。在独立下载过程中，同时响应OTA主节点取消下载、获取下载状态和获取刷写脚本的控制指令。

独立下载和正常下载全部完成后进入安装阶段，仍然由Engine Service统一调用UAt模块完成升级刷写的控制。至此，OTA任务的流程全部完成。

3.4 远程配置

远程配置即通过远程通信的方式对整车的软件功能进行配置，通常是启动和关闭整车的某一功能，可用于开启与关闭整车的基础功能和可销售的软件功能。基础功能通常可免费获取，通过OTA的方式升级软件后，通过远程配置启动与关闭。可销售软件功能在用户购买后可进行单独配置，或者需要先通过OTA平台将软件升级至指定版本再进行配置。

不同的整车硬件版本配置软件的方式不同，主要分为广播配置与诊断配置，物理层支持通过CAN网络和车载以太网进行配置。广播配置的优势在于一次广播播报可配置多个相关ECU报文，诊断配置则需要对各ECU进行单独寻址，逐步完成功能的配置[17]。

3.5 OTA安全设计

OTA安全设计主要包括服务器端、管端及车端3个部分。服务器端安全设计如图11所示。

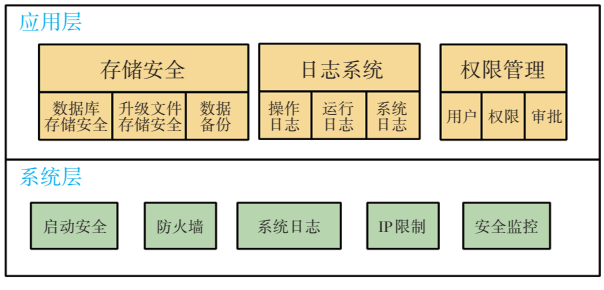

图11 服务器端安全设计

OTA服务器与车端采用超文本传输协议（Hyper Text Transfer Protocol，HTTP），基于私有PKI体系实现双向身份认证，保证通信信道的加密传输。

车端安全设计主要包括升级前置条件控制和升级异常处理。升级前置条件设计内容包括车速、挡位、蓄电池电量、网络通信等，此外，本文特别设计了OTA模式。OTA模式是指OTA升级过程中，各控制器为适应OTA刷写需求进入的特殊状态。OTA任务下载完成开始安装后，主节点会播报“OTA模式”信号，模式信号为状态值，各控制器监测到信号后，进入OTA模式，在该模式下，各控制器执行不同动作，如车身控制器维持电池锁定状态、娱乐主机降低亮度等。

升级异常策略分为重刷和回滚2种处理机制。依赖于可重复刷写的引导加载程序（Bootloader）功能设计，ECU至少需要实现升级异常后可重复刷写；回滚机制包含云端、车内、ECU内回滚等方式，其中ECU内回滚即ECU的a、b面升级，能够彻底解决刷写失败异常的问题，但对于ECU芯片性能、存储空间的高要求，会使硬件成本增加20%~30%，因此需要根据ECU的功能安全等级分级设计。

4 试验验证

4.1 试验环境

针对上述精益化设计内容，进行OTA功能及系统性能的试验验证，以实物台架试验为主，系统仿真为辅。功能测试包括任务自动化、断点续传、独立下载流程、远程配置流程、整车版本判断、软件包篡改、升级前置条件等；性能测试包括云端服务器每秒查询率（Queries Per Second，QPS）、独立下载速度、CAN刷写速度、软件升级成功率、功能配置成功率等。云端测试系统及实物台架如图12、图13所示。

图12 云端测试系统

图13 试验台架

4.2 功能测试

功能测试基本采用云端任务部署、车端任务执行的方式，测试目的及方法如表2所示。以自动化的任务下发为例，测试大致步骤为：设置一条三节点的升级路线进行全网发布；使用实车1进行车云通信；使用实车2首次接入OTA系统进行车云通信（模拟车辆刚下线）。测试发现实车1和实车2皆能完成2次任务串行下发，任务升级成功，整车软件版

本判断正确。通过本次测试,验证了云端升级任务自动化下发以及车辆连续升级功能的实效性。

表2　部分功能测试项

功能	测试目的	测试内容
任务自动化	车端整车软件版本正确判断,云端任务能否按预设下发	1.整车版本预置列表解析逻辑 2.各ECU软硬件版本上报逻辑 3.云端路线下发流程 4.任务相关车云交互协议
断点续传	验证软件下载过程中网络波动、连接中断、切换下载通路等异常工况下,下载包能否继续下载	1.下载过程中断开网关与T-BOX的以太网连接 2.下载过程中使用隔离设备阻止T-BOX天线信号 3.下载过程中4G与WIFI切换 4.下载过程中切断台架整体电源 5.下载过程使车辆进入睡眠模式
独立下载	验证IVI等控制器下载及加密、解密功能,以及与网关间的状态交互	1.独立下载的断点续传 2.升级包解密验签 3.下载状态及进度反馈与网关的SOME/IP交互
远程配置	配置信息远程下发,相关ECU功能能否相应开启关闭	1.配置信息车云交互协议准确性 2.诊断配置流程 3.广播配置流程
软件包篡改	考查能否有效验证软件包的完整性	车云任务同步后,通过篡改网关本地域名解析缓存,使车辆访问伪造服务器下载被篡改软件包
升级前置条件	考查升级前,对车辆处于可升级的安全模式的验证	1.驻车条件:P挡、电子驻车制动(EPB)系统启动、车速、转速等 2.电量条件:蓄电池SOC、电压,动力电池SOC等 3.模式条件:OTA模式下各ECU的功能禁止

4.3　性能测试

4.3.1　云端并发能力测试

通过高性能服务器模拟多车连接请求,验证服务器的处理能力及错误率等关键指标。压力机硬件性能指标如表3所示。

表3　压力机指标

条目	性能
操作系统	Centos7.9
IP地址	10.112.54.3
工具	Locust2.1(专业服务器性能测试工具) nmon16m(前端及服务器性能监控)
CPU	Intel Xeon Gold 6278C CPU@ 2.60 GHz四核
空闲内存/GB	4

使用Locust负载测试工具,设定HTTP请求超时时间为10 s,测试样本为10万辆汽车,包括3万辆有任务车辆和7万辆无任务车辆。并发时根据有任务和无任务情况循环测试所有车辆,并发送对应请求。

测试结果为:在5 h内共请求1 399万次,其中失败266万次,在服务端磁盘饱和前,请求次数为960 次/s,错误率小于0.02%;磁盘饱和后,请求次数下降为500 次/s左右,错误率迅速增大。

4.3.2　CAN刷写速度测试

基于CANoe工具的通信访问编程语言(Communication Access Programming Language,CAPL)脚本实现自动化重复测试上位机,针对车门控制器进行重复刷写计时。上位机设置刷写连续帧间隔(STmin)为0,hex文件为200 KB,分别针对总线非刷写报文无负载、5%负载、10%负载情况进行验证,测试数据如表4所示。测试结果表明,随着总线负载的增加,控制器的刷写时间也逐步增加。

表4　CAN刷写速率测试

测试次序	刷写时间/s		
	总线负载率0	总线负载率5%	总线负载率10%
1	40.1	42.2	45.0
2	40.9	42.5	45.1
3	41.5	42.0	45.3
4	42.0	44.1	45.0
5	43.1	44.3	46.1
6	39.9	43.9	45.9

4.3.3　刷写成功率测试

刷写成功率验证分为单件多次重复刷写验证和多件单次刷写验证,刷写成功率测试方法与刷写速度测试一致。刷写成功需具备2个充分条件:对Flash擦除指令,车门控制器给出积极响应;刷写完成后,通过版本号读取服务和通信报文均能够收到目标版本号。单件多次重复刷写验证对控制器进行百次级的压力测试,成功率为100%;多件单次刷写验证使用10个相同版本的车门控制器进行,成功率为100%。

5　结束语

本文针对进行了方案设计,并通过实车台架验证了其可行性,得到以下主要结论:

a. OTA云端采用微服务架构设计,可支撑百万级车辆的业务请求,同时便于后续业务升级;

b.整车基线控制在管理层面可以有效保证车辆的

初始状态，在技术层面更加便于任务自动化的实现；

c. 独立下载能够有效缩减大数据量软件包在车内网络的传输时间，减少ECU安装过程的等待时间；

d. 不同于大数据量的软件更新，远程配置更适用于独立实现轻量级的参数变动。

参 考 文 献

[1] 陈虹, 郭露露, 边宁. 对汽车智能化进程及其关键技术的思考[J]. 科技导报, 2017, 35(11): 52-59.
CHEN H, GUO L L, BIAN N. Reflections on the Process of Automotive Intelligence and its Key Technologies[J]. Science and Technology Herald, 2017, 35(11): 52-59.

[2] PHUNG P H, NILSSON D K. A Model for Safe and Secure Execution of Downloaded Vehicle Applications[C]// IET Road Transport Information and Control Conference and the ITS United Kingdom Members' Conference (RTIC 2010) -Better Transport Through Technology. London: IEEE, 2010.

[3] 王栋梁, 汤利顺, 陈博, 等. 智能网联汽车整车OTA功能设计研究[J]. 汽车技术, 2018(10): 29-33.
WANG D L, TANG L S, CHEN B, et al. Research on the Design of OTA Function for the Whole Vehicle of Smart Networked Vehicles[J]. Automotive Technology, 2018(10): 29-33.

[4] 周奇才, 王奕童, 赵炯, 等. 基于互联网设备的车辆安全空中下载通信协议方案[J]. 汽车技术, 2020(1): 6-11.
ZHOU Q C, WANG Y T, ZHAO J, et al. A Communication Protocol Scheme for Safe Over-the-Air Download of Vehicles Based on Internet Devices[J]. Automotive Technology, 2020(1): 6-11.

[5] 朱云尧, 吴胜男. 国内外智能网联汽车软件在线升级法规分析[J]. 汽车文摘, 2022(10): 6-10.
ZHU Y Y, WU S N. Analysis of Online Software Upgrade Regulations for Domestic and Foreign Intelligent Networked Vehicles[J]. Automotive Digest, 2022(10): 6-10.

[6] 万开明, 洪雷. 车载OTA技术研究[J]. 时代汽车, 2020(11): 10-11.
WAN K M, HONG L. Research on In-Vehicle OTA Technology[J]. Times Automotive, 2020(11): 10-11.

[7] MANSOR H, MARKANTONAKIS K, AKRAM R N, et al. Let's Get Mobile: Secure FOTA for Automotive System[C]// International Conference on Network and System Security. New York: Springer, Cham, 2015.

[8] IDREES M S, SCHWEPPE H, ROUDIER Y, et al. Secure Automotive Onboard Protocols: A Case of Over-the-Air Firmware Updates[C]// International Workshop on Communication Technologies for Vehicles. Heidelberg, Berlin: Springer, 2011.

[9] 程达, 姬东耀. OTA业务的高速下载[J]. 计算机工程与应用, 2005(33): 147-148+204.
CHENG D, JI D Y. High-Speed Download for OTA Services[J]. Computer Engineering and Applications, 2005(33): 147-148+204.

[10] NILSSON D K, SUN L, NAKAJIMA T. A Framework for Self-Verification of Firmware Updates over the Air in Vehicle ECUs[C]// 2008 IEEE Globecom Workshops. New Orleans: IEEE, 2008.

[11] NILSSON D K, LARSON U E. Secure Firmware Updates Over the Air in Intelligent Vehicles[C]// ICC Workshops-2008 IEEE International Conference on Communications Workshops. New Orleans: IEEE, 2008.

[12] 武智, 刘天宇, 贾先锋. 智能网联汽车OTA升级安全设计[J]. 汽车实用技术, 2022, 47(3): 38-40.
WU Z, LIU T Y, JIA X F. OTA Upgrade Safety Design for Smart Networked Vehicles[J]. Automotive Practical Technology, 2022, 47(3): 38-40.

[13] 严娟, 张玉川, 杨鹏翔, 等. 基于以太网OTA远程升级的研究[J]. 上海汽车, 2020(3): 15-18+27.
YAN J, ZHANG Y C, YANG P X, et al. The Study on OTA Remote Updating of Ethernet[J]. Shanghai Automotive, 2020(3): 15-18+27.

[14] VILLAMIZAR M, GARCÉS O, CASTRO H, et al. Evaluating the Monolithic and the Microservice Architecture Pattern Todeploy Web Applications in the Cloud[C]// 2015 10th Computing Colombian Conference (10CCC). Bogota, Colombia: IEEE, 2015.

[15] ZHANG J, LIAO Z, ZHU L. Research on Design and Implementation of Automotive ECUs Software Remote Update[J]. Applied Mechanics and Materials, 2015, 740: 847-851.

[16] MU C Y, SUN L N, DU Z J, et al. Research and Development of Device for Downloading and Updating Software of Product ECU Based on Extended CCP[C]// 2007 2nd IEEE Conference on Industrial Electronics and Applications. Harbin, China: IEEE, 2007: 2865-2869.

[17] SHI G Y, KE Z W, YAN F W, et al. A Vehicle Electric Control Unit Over-the-Air Reprogramming System[C]// 2015 International Conference on Connected Vehicles and Expo (ICCVE). New York: IEEE, 2015.

（责任编辑　王　一）

修改稿收到日期为2024年5月15日。