软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software [doi: 10.13328/j.cnki.jos.007196]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

智能网联汽车的车载网络攻防技术研究进展*

陈博言[1,2,4], 沈晴霓[1,2,4], 张晓磊[2,3,4], 张 鑫[1,2,4], 李 聪[2,3,4], 吴中海[1,2,3,4]

[1](北京大学 软件与微电子学院, 北京 102600)

[2](软件工程国家工程研究中心 (北京大学), 北京 100871)

[3](北京大学 计算机学院, 北京 100871)

[4](高可信软件技术教育部重点实验室 (北京大学), 北京 100871)

通信作者: 沈晴霓, E-mail: qingnishen@ss.pku.edu.cn; 吴中海, E-mail: wuzh@pku. edu. cn

摘　要: 随着人工智能和 5G 技术在汽车行业的应用, 智能网联汽车应运而生, 它是一种由众多来自不同供应商的电子控制单元 (ECU) 组成的复杂分布式异构系统, 通过以 CAN 为代表的车载网络协议交互协同控制各 ECU. 然而, 攻击者可能通过各种接口攻击智能网联汽车, 渗透到车载网络, 再攻击车载网络及其各组成部分如 ECU. 因此, 智能网联汽车的车载网络安全成为近些年车辆安全研究的焦点之一. 在介绍智能网联汽车整体结构、ECU、CAN 总线和车载诊断协议等基础之上, 首先总结了目前车载网络协议的逆向工程技术进展, 逆向工程的目标是获取汽车行业通常不公开的车载网络协议实现细节, 也是实施攻击和防御的前提条件. 然后从车载网络攻、防两个角度展开: 一方面概括了车载网络攻击向量及主要攻击技术, 包括通过物理访问和远程访问方式实施的攻击技术, 以及针对 ECU 和 CAN 总线实施的攻击技术; 另一方面, 讨论了车载网络现有的防御技术, 包括基于特征工程和机器学习方法的车载网络入侵检测和基于密码学方法的车载网络协议安全增强技术. 最后展望了未来的研究方向.

关键词: 智能网联汽车; 车载网络; 逆向工程; 入侵检测; 协议安全增强

中图法分类号: TP393

中文引用格式: 陈博言, 沈晴霓, 张晓磊, 张鑫, 李聪, 吴中海. 智能网联汽车的车载网络攻防技术研究进展. 软件学报. http://www.jos.org.cn/1000-9825/7196.htm

英文引用格式: Chen BY, Shen QN, Zhang XL, Zhang X, Li C, Wu ZH. Research Progress on Attacks and Defenses Technologies for In-vehicle Network of Intelligent Connected Vehicle. Ruan Jian Xue Bao/Journal of Software (in Chinese). http://www.jos.org.cn/1000-9825/7196.htm

Research Progress on Attacks and Defenses Technologies for In-vehicle Network of Intelligent Connected Vehicle

CHEN Bo-Yan[1,2,4], SHEN Qing-Ni[1,2,4], ZHANG Xiao-Lei[2,3,4], ZHANG Xin[1,2,4], LI Cong[2,3,4], WU Zhong-Hai[1,2,3,4]

[1](School of Software and Microelectronics, Peking University, Beijing 102600, China)

[2](National Engineering Research Center for Software Engineering (Peking University), Beijing 100871, China)

[3](School of Computer Science, Peking University, Beijing 100871, China)

[4](High Confidence Software Technology (Peking University), Ministry of Education, Beijing 100871, China)

Abstract: As artificial intelligence and 5G technology are applied in the automotive industry, the intelligent connected vehicle came into being. It is a complex distributed heterogeneous system composed of a large number of electronic control units (ECUs) from different suppliers and collaborates to control each ECU through the in-vehicle network protocol represented by CAN. However, an attacker could attack an intelligent connected vehicle through a variety of interfaces to penetrate the in-vehicle network, and then attack the in-vehicle network and its components such as ECU. Therefore, in-vehicle network security for intelligent connected vehicles has become one of the

* 基金项目: 国家自然科学基金 (61672062)
收稿时间: 2023-07-18; 修改时间: 2023-11-18; 采用时间: 2024-04-01

focuses of vehicle security research in recent years. On the basis of introducing the structure of intelligent connected vehicle, ECU, CAN bus and on-board diagnostic protocol, this study first summarizes the research progress of reverse engineering technology for in-vehicle network protocols. The reverse engineering technology aims to obtain the implementation details of in-vehicle network protocols that are usually not disclosed in the automotive industry. It is also a prerequisite for the implementation of in-vehicle network attack and defense. The remaining part is developed from two angles of attack and defense. On the one hand, the attack vectors and main attack technologies of in-vehicle network are summarized, including the attack technologies implemented through physical access and remote access, as well as the attack technologies implemented against ECU and CAN bus. On the other hand, the existing in-vehicle network defense technologies are discussed, including the intrusion detection technology based on feature extraction and machine learning methods, and the security enhancement technology of in-vehicle network protocols based on cryptographic approaches. Finally, the future research direction is prospected.

Key words: intelligent connected vehicle (ICV); in-vehicle network (IVN); reverse engineering; intrusion detection; protocol security enhancement

智能网联汽车是一个庞大而复杂的系统, 由数量众多的电子控制单元 (electronic control unit, ECU) 和传感器等部分组成. 各种 ECU 均连接到车载网络 (in-vehicle network, IVN), 并通过车载网络传输数据. 车载网络协议中 CAN (control area network) 凭借其高性能、高容错、低成本等特点, 成为应用最为广泛的车载网络标准. 但随着智能网联汽车技术的发展, 以 CAN 为代表的车载网络协议开始不断暴露出一系列安全问题.

安全是汽车行业关注的焦点. 智能网联汽车中软件的复杂度逐渐增大, 与外部连接的接口逐渐增多, 使得车载网络的攻击向量不断增加. 黑客可能通过物理访问 (例如 OBD-II 接口) 或者无线访问 (例如 WiFi) 方式攻击智能网联汽车, 入侵车载网络, 进一步通过车载网络向 ECU 发送指令控制汽车, 威胁驾驶者生命安全. 近些年学术界已经验证类似攻击的存在, 例如 2015 年研究者在 Black Hat 大会上演示了远程攻击并控制正在路上行驶的 Jeep Cherokee 汽车, 该攻击利用了车载娱乐系统中的漏洞, 向车载网络注入恶意消息, 使汽车转弯偏离道路造成事故[1]. 腾讯科恩实验室设计了一系列针对特斯拉汽车的攻击, 包括通过蜂窝网络远程入侵 CAN 总线, 从而执行恶意功能的 Free-fall 攻击[2], 以及攻击 OTA 升级过程的 Over-the-air 攻击[3]. 由此可见, 智能网联汽车安全的关键之一是车载网络安全.

车载网络协议一般是闭源的, 对于研究者而言是一个黑盒, 因此研究者需要对车载网络协议进行逆向工程, 分析车载网络中消息的格式和语义等信息, 这对车载网络的攻击和防御都有较大意义. 本文将车载网络协议逆向过程归纳为两个阶段, 第 1 个阶段为载荷令牌化 (tokenization), 将 CAN 数据帧切分为多个 CAN 信号, CAN 信号是描述车辆行为的最小单元, 表示车辆功能相关的实时信息, 如车门状态、当前速度等; 第 2 个阶段为令牌解析, 即分析 CAN 信号的格式和包含的信息语义. 本文按照这两个阶段讨论和总结车载网络协议逆向相关工作.

对于车载网络攻击, 研究者通过理论分析和实验等方法总结了车载网络的攻击面, 证明攻击者不仅可以通过物理访问入侵车载网络, 也可以通过无线访问入侵车载网络. 基于智能网联汽车攻击面的分析, 本文将攻击者分为本地攻击者和远程攻击者, 并从两个维度对车载网络攻击进行分类. 首先根据访问方式进行分类, 将车载网络攻击分为本地攻击者可实施的物理访问攻击, 以及远程攻击者可实施的无线访问攻击, 并根据具体攻击入口 (例如 OBD-II 接口、WiFi 等) 对两类攻击进行细分. 攻击者通过物理或者无线访问入侵车载网络后, 一般需要攻击车载网络及其连接的其他部分如 ECU 等攻击目标, 执行恶意功能. 进一步地, 本文根据最终攻击目标, 将车载网络攻击归纳为两类, 针对 ECU 的攻击和针对 CAN 总线的攻击, 并对两类攻击进行详细介绍.

为应对日益严峻的车载网络攻击, 研究者提出了多种车载网络防御方案. 本文将其归纳为两类: 第 1 类是车载网络入侵检测技术, 其具备的主动性和实时性等特点非常适合车载网络. 研究者采用不同技术, 通过入侵检测的方法防御各种类型的车载网络攻击. 近些年研究者不断优化相关工作, 提高车载网络入侵检测技术的性能和准确度. 本文根据使用的技术, 将其分为基于特征工程的入侵检测与基于机器学习的入侵检测. 第 2 类是车载网络协议安全增强技术. 针对车载网络协议的安全缺陷, 研究者提出了两种方案, 一种通过轻量级密码学方案, 增强通信的机密性; 另一种利用高效的 HMAC (Hash-based message authentication code), 增强车载网络的消息完整性和身份鉴别能力.

本文第 1 节介绍基础知识. 第 2 节总结车载网络攻防研究方向. 第 3 节根据车载网络协议逆向的两个过程分析车载网络协议的逆向研究. 第 4 节根据攻击入口和攻击目标, 对车载网络攻击技术进行分类讨论. 第 5 节总结车载网络防御技术研究. 第 6 节总结全文, 并对车载网络攻防技术未来研究进行展望.

1 基础知识

本节对智能网联汽车的车载网络攻防研究涉及的相关基础知识进行介绍, 包括智能网联汽车的整体软硬件架构和车载网络的相关概念和基本知识.

1.1 相关术语

智能网联汽车存在很多专业术语, 表 1 总结了智能网联汽车的车载网络攻防研究中需要了解的相关术语.

表 1 车载网络攻防研究相关术语

中文名称	英文全称和缩写	含义
智能网联汽车	intelligent connected vehicle (ICV)	车联网与智能车的有机联合, 搭载先进的车载传感器、控制器等装置, 并融合现代通信与网络技术, 实现车与人、路、后台等智能信息交换共享, 最终可替代人来操作的新一代汽车
电子控制单元	electronic control unit (ECU)	用于控制车辆电器元件的控制单元, 包含了硬件及其配套的软件系统. 当前一台汽车一般包含几十或上百个ECU, 用于控制动力系统如发动机、刹车, 车辆内部环境如空调等相关子系统
车载网络	in-vehicle network (IVN)	连接车辆内部各个ECU的网络, 主要包括CAN、LIN和FlexRay等
控制器局域网	control area network (CAN)	应用最广泛的车载网络协议, 分为物理层、数据链路层和应用层, 连接车内各ECU, 是一种广播的协议, 不区分发送者和接受者, 消息可以传输给每个连接在CAN总线上的ECU
车载诊断协议	on-board diagnoistic (OBD) protocol	用于ECU故障报警和处理的相关协议, 一般ECU在开发时都会进行相应的诊断协议开发, 用于在ECU出现故障时报错
车载诊断工具	on-board diagnoistic tool	一种车辆维修人员使用的工具, 通过连接车辆的OBD-II接口获取车辆的诊断信息
OBD-II接口	OBD-II port	用于连接车载诊断工具的16针接口, 一般位于驾驶台下方
参数标识	parameter identification (PID)	利用车载诊断工具, 可以向OBD-II接口注入参数标识(PID), 请求CAN数据流和故障码, 或者写入CAN消息
T-Box	telematics box (T-Box)	T-BOX作为无线网关, 通过4G/5G远程无线通信、GPS卫星定位等功能, 为整车提供远程通信接口, 提供包括行车数据采集、行驶轨迹记录、车辆故障监控、车辆远程控制等服务
入侵检测系统	intrusion detection system (IDS)	是一种积极主动的网络防御技术, 通过对网络数据流进行分析, 判断是否出现网络攻击, 在发现可疑行为时发出警报

1.2 智能网联汽车整体结构

在过去的几十年里, 汽车硬件结构不断复杂, 其软件规模几乎从零增长到数千万行代码, 这些代码分布在众多电子控制单元 (ECU) 上. ECU 是智能网联汽车的基本组成单元, 可以理解为控制不同子系统的微型电脑, 包括安全关键子系统 (如发动机控制、刹车控制) 和非安全关键子系统 (如车载娱乐系统). 现代汽车的许多功能都需要跨 ECU 进行复杂的通信和交互. 例如, 刹车控制系统的 ECU 需要监测驾驶者踩刹车踏板的操作, 当车辆刹车时, 刹车控制系统的 ECU 通过车载网络与车灯控制相关 ECU 交互, 车灯控制相关 ECU 亮起刹车灯.

随着智能网联汽车的出现, ECU 的软硬件功能更加丰富. 这些 ECU 提供较为完整的类 Unix 环境, 并且与 GPS 等传感器设备相结合. 在一些早期车辆中, 使用一次性设计的双边物理线连接不同的 ECU, 来完成 ECU 与 ECU 之间、传感器与 ECU 之间的交互. 该方法可扩展性和通用性较弱, 会带来极大的布线开销和交互复杂性. 这促使汽车制造商和供应商对传输总线上的车载网络协议进行标准化, 提出了控制器局域网 (control area network, CAN) 协议等. 其中, 工业界应用最为广泛的车载网络协议是 CAN 协议[4].

一个典型的智能网联汽车整体结构及其主要威胁如图 1 所示. 其基本组成单元是控制车辆子功能的 ECU, 位于中心的是一个中央网关, 中央网关负责连接车内各个子网络. 车载网络包括 CAN、FlexRay、LIN 和 MOST, FlexRay 一般用于对传输速率要求较高的车载网络, 例如连接雷达和传感器的子网络, MOST (media oriented systems transport) 一般服务于多媒体传输, 用于连接车载娱乐系统. LIN (local interconnect network) 的传输速率最慢, 但是成本最低.

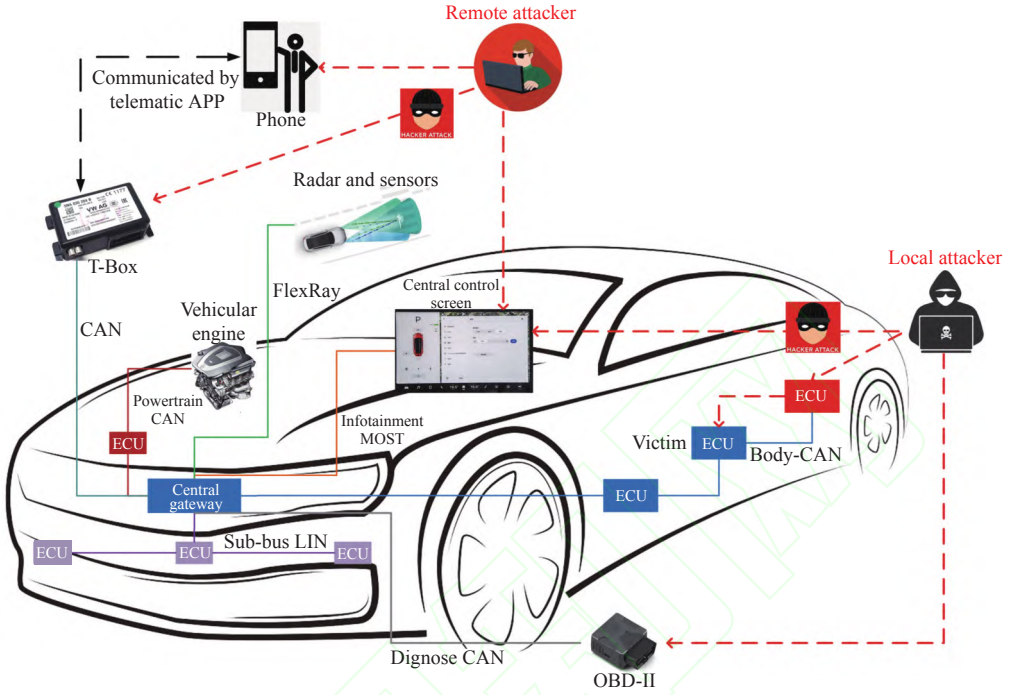

图 1 智能网联车整体结构及其主要威胁示意图

目前车载网络攻防的研究大多基于 CAN. CAN 是业界普遍采用的车载网络协议, 其中的数据包被广播传送到所有节点, 每个 CAN 数据包只有 CAN ID, 没有发送和接收地址, 各节点根据 CAN ID 决定是否接收该数据包. 车载网络是智能网联汽车的控制中枢, 负责不同 ECU 上软硬件的交互. 车载网络的安全对智能网联汽车的网络安全、功能和信息安全都至关重要, 图 1 展示了智能网联汽车面临的本地攻击者 (local attacker) 和远程攻击者 (remote attacker) 带来的主要威胁. 下一小节将详细介绍 ECU 和车载网络协议.

1.3 ECU 和车载网络协议

1.3.1 ECU

ECU 通常包括硬件微控制器单元 (micro control unit, MCU) 以及相应的软件应用. ECU 的应用从传感器读取输入, 将输出写入微控制器单元执行相应操作. 图 2 所示的 ECU 的通信协议栈由应用层、数据链路层组成, ECU 连接 CAN 总线与其他 ECU 进行交互. CAN 协议与其他网络协议一样采用分层设计, CAN 的信息传输量较少, 对实时性要求较高, 其连接方式相对较简单. 因此 CAN 协议只采用了 OSI 7 层通信模型中的 3 层, 在网络下层只采用了物理层和数据链路层, 而在上层只有应用层[4]. 图 2 展示了 CAN 分层架构和 ECU 通信过程.

如图 2 所示, 运行在 ECU 的应用创建一个 CAN 消息, 将此消息传输到数据链路层, 该层将添加各种控制和完整性字段以生成一个 CAN 帧 (CAN frame), 该 CAN 帧通过物理层以比特流形式串行传输. 为了接收消息, 接收端 ECU 在数据链路层解析和验证 CAN 帧, 然后传递给应用层. 假设图 2 中 ECU A 为监测刹车脚踏板的 ECU, ECU B 为车灯控制 ECU. 运行在 ECU A 应用层的软件在监测到驾驶者踩刹车时, 生成通知刹车灯亮的 CAN 消息, 通

过数据链路层的 CAN 硬件 (例如 CAN 发射器) 发送 CAN 帧, 在物理层的 CAN 总线上发送比特流到 ECU B, 然后 ECU B 再依次通过数据链路层和应用层, ECU B 的应用程序解析 CAN 消息, 并亮起刹车灯.

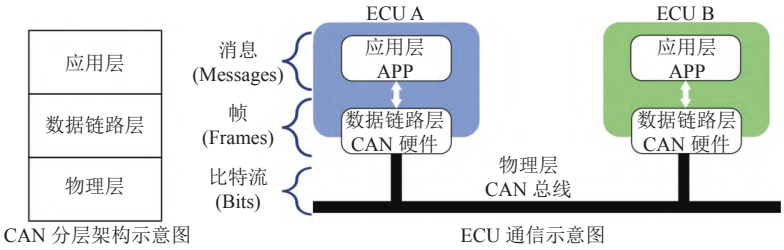

图 2　CAN 分层架构和 ECU 通信示意图

1.3.2　CAN 协议

CAN 物理层定义了 CAN 总线的物理连接、电气信号和传输速率等方面的规范. CAN 总线物理层采用差分信号传输, 使用两根线分别传输高电平和低电平信号, 以提高抗干扰能力. 这两根线分别称为 CAN_H 和 CAN_L 线, 它们之间的电压差表示数据位的取值. 当 CAN_H 高电平, CAN_L 低电平时, 表示逻辑位为 1; 当 CAN_H 低电平, CAN_L 高电平时, 表示逻辑位为 0[4].

CAN 数据链路层实现 CAN 总线上的可靠数据传输和通信, 提供高效性、实时性和可靠性的保证, 数据的基本组织形式是 CAN 数据帧. 图 3 描述了 CAN 2.0a 数据帧的结构, 这是 CAN 总线最常用的数据帧类型.

图 3　CAN 数据帧结构

图 3 中, 从左到右, CAN 数据帧包含了如下较为重要的字段[4].

(1) CAN ID: CAN 是一个多主控的、基于消息的广播总线. CAN 帧可以不包含任何有关其源或目标 ECU 的任何信息, 每个 CAN 帧都携带唯一的消息标识符 (CAN ID), 较低的 CAN ID 具有较高的优先级. 在 CAN 2.0a 规范中 CAN ID 是 11 位, 因此允许多达 2 048 个不同的 CAN ID. 其数值越小优先级越高, 因此 CAN ID 为 0 的优先级最高, CAN ID 为 0 的 ECU 总是可以抢占 CAN 总线并发送消息. 当多个 ECU 节点同时发送消息时, 具有较高优先级的帧将优先被发送, 其他帧需要等待. 然而, 当多个节点同时发送消息时, 可能会发生冲突. CAN 总线采用了一种错误检测和恢复机制, 节点会侦测到发送错误并在一个时间段后重新发送消息.

(2) DLC: 该字段长 4 位, 指定消息的数据字段中的字节数.

(3) Data: 包含实际消息数据, 也被称为有效载荷 (payload). 根据 DLC 字段的值, 该字段包含 0–8 个字节的数据. 有效载荷由一个或多个信号 (signal) 组成, 信号是 ECU 传输的信息, 例如车速、油门. 发送给变速器控制模块的一条 CAN 消息可以同时包含车速和发动机转速信号. 这些信号的格式、语义是由相关厂商定义的, 一般不开源.

CAN 应用层定义了消息的格式以及消息的发送和接收过程, 基本传输单位是 CAN 消息. 一些其他协议在 CAN 应用层上实现, 完成具体的应用功能, 例如车载诊断协议.

1.3.3　车载诊断协议

车载诊断协议允许专业人员通过工具与车载网络进行交互, 能够快速准确地判断车辆某个 ECU 或者控制器的故障以及原因, 从而为车辆维修提供可靠的依据. 在车辆 ECU 的研发过程中, 相关开发者都会进行相应的诊断开发, 可以在 ECU 发生故障时及时上报, 在必要时点亮故障灯[5]. 统一的车载诊断协议可以大大降低各个供应商的开发成本. 因此, 汽车行业形成了车载诊断协议, 规定汽车在线诊断的统一标准. CAN 协议定义了传输层、数据链路层及物理层, 而车载诊断协议则定义了 CAN 协议之上的应用层. 目前汽车行业车载诊断协议的主要标准是

KWP 2000 (keyword protocol 2000) 协议和 UDS (unified diagnostic services) 协议[5].

图 1 中, OBD-II 接口通过中央网关连接到车载网络中, 车载诊断工具可以连接到汽车的 OBD-II 接口, 通过车载网络向 ECU 发送诊断请求消息, 然后接收 ECU 发送来的诊断回复消息, 并解析消息将相关结果显示给维修人员. 一些车载诊断工具可以通过发送参数标识 (parameter identification, PID) 的方法, 向 ECU 发送控制信息, 维修人员利用该方法, 可以通过车载诊断工具, 向 CAN 总线发送参数标识消除特定故障码.

2 车载网络攻防研究方向

智能网联汽车可以理解为由众多不同功能的 ECU, 通过车载网络连接, 兼具消息收集、用户交互、车辆控制等功能的复杂系统, 具有分布性、异构性和闭源等特点. 车载网络是智能网联汽车的中枢, 其安全研究逐渐引起学术界和工业界关注[6,7]. CAN 是车载网络应用最广泛的协议, 相关研究大多聚焦于 CAN[8,9]. 本文从车载网络协议逆向、车载网络的攻击和防御技术 3 个角度整理了相关研究工作.

本节作为全文的纲要, 总结车载网络攻防技术的研究方向.

• 对于车载网络协议逆向, 虽然 CAN 数据帧的整体格式是已知的, 但是其有效载荷的具体格式和语义是闭源的. 在 2015 年针对 Jeep 汽车的攻击中, 研究者必须对 CAN 消息进行人工逆向, 以获得对车辆的远程控制. 这是一项繁琐且不可扩展的工作[1]. 对于黑客和研究者, 均需要了解 CAN 消息的格式和语义, 这就需要逆向工程的相关技术. CAN 消息帧的有效载荷由多个信号组成, 每个信号表示一个具体的语义 (例如发动机转速), 这些信号的格式和语义均不开源. 车载网络协议逆向的目标是在 CAN 数据帧中定位信号的位置, 将载荷切分为一个个令牌 (token), 该过程称为载荷令牌化 (tokenization). 然后对每个信号的格式和语义进行逆向, 该过程称为令牌解析 (translation)[10]. 本文按照载荷令牌化[11−17]和令牌解析[13−15,18−21]这两个阶段, 总结当前车载网络逆向相关研究.

• 对于车载网络攻击, 以 CAN 为代表性的车载网络, 其在设计之初就缺乏相应的安全性设计, 其机密性保护不足, 且缺乏完整性和身份鉴别等安全机制. 随着智能网联汽车技术的发展, 针对车载网络的黑客攻击已成为真实迫切的威胁, 许多工作证明了攻击者通过物理访问和无线访问对车载网络的攻击是可行的. 本文首先总结了车载网络存在的攻击向量, 根据访问方式将攻击者分为两类, 然后再根据具体攻击入口进行划分. 第一类攻击者是本地攻击者, 该类攻击者可以通过物理访问如 OBD-II 接口、USB 接口等[22,23]入侵智能网联汽车; 远程攻击者可以通过无线访问入侵智能网联汽车, 其可以利用的攻击入口包括传感器[24−35](例如摄像头)、近程通信接口[36−40](例如蓝牙) 和远程通信接口[3,8,9,41](例如蜂窝网络). 本地攻击者和远程攻击者入侵智能网联汽车后, 要形成完整的攻击链, 其最终目标是干扰和控制汽车, 大多需要入侵车载网络及其组成部分. 进一步地, 本文根据其攻击目标, 将车载网络攻击分为针对 ECU 的攻击[2,3,42]和针对 CAN 总线的攻击[43−47], 并对各类攻击进行了详细讨论.

• 对于车载网络防御, 研究者主要提出了两类车载网络防御技术. 一类是车载网络入侵检测技术. 入侵检测是一种主动的防御技术, 可以主动感知攻击者的攻击行为, 并报警或阻止攻击者的攻击行为. 与传统网络相比, 车载网络具有不同的特点, 例如消息广播、高实时性要求等, 研究者针对车载网络设计了针对性的入侵检测方案, 从而抵御攻击者对车载网络的特定攻击. 本文根据采用的技术不同, 将车载网络入侵检测工作分为两大类: 一类是基于特征工程的入侵检测[48−62], 另一大类是基于机器学习的入侵检测[63−69]. 另一种车载网络防御是车载网络协议安全增强技术. CAN 协议存在很多安全缺陷, 研究者尝试对 CAN 等车载网络协议进行安全增强, 从而防御某些攻击. 本文重点关注两类工作: 基于轻量级密码学的通信机密性增强[70−74], 以及基于高效 HMAC 的身份鉴别和消息完整性增强[75−81].

相比于传统网络安全问题, 车载网络安全问题导致的后果更为严重, 可能威胁驾驶者的生命安全, 随着智能网联汽车技术的迅猛发展, 车载网络暴露的攻击面越来越大. 网络空间安全和智能汽车领域顶级会议和期刊 (S&P、CCS、USENIX Security、NDSS、IV、TDSC、TIFS、TITS、TVT、软件学报) 上不断涌现车载网络安全的相关文章. 本文梳理了近 10 年车载网络攻防研究的代表性工作, 表 2 展示了车载网络攻防研究方向、研究内容、主要工作、主要方案类型和代表工作.

表 2 车载网络攻防相关研究工作

研究方向	研究内容	主要工作	主要方案类型	代表工作
车载网络协议逆向	载荷令牌化阶段 (tokenization)	识别并切分有效载荷中的每个令牌	(1) 基于组合优化的载荷令牌化 (2) 基于比特翻转率的载荷令牌化	(1) ACCT[13] (2) LibreCAN[15]
	令牌解析阶段	对令牌的格式、语义进行分析	(1) 基于PID注入的令牌解析 (2) 基于语义分类的令牌解析 (3) 基于配套应用的令牌解析	(1) CAN-D[14] (2) AutoCAN[20] (3) DP-Reverser[21]
车载网络攻击技术	攻击向量	分析和评估车载网络可能的攻击向量	(1) 车载网络攻击入口分析 (2) 车载网络安全风险评估	文献[6−9]
	根据攻击入口分类	物理访问攻击	(1) 通过OBD-II接口的攻击 (2) 通过USB/CD等接口的攻击	(1) Plug-N-Pwned[21] (2) DeepSniffer[22]
		无线访问攻击	(1) 通过传感器的攻击 (2) 通过近程通信的攻击 (3) 通过远程通信的攻击	(1) AttrackZone[27] (2) Antonioli等人[36] (3) Free-fall[2]
	根据攻击目标分类	针对ECU的攻击	(1) ECU伪装攻击 (2) 重刷攻击	(1) Cloaking[42] (2) Over-the-air[3]
		针对CAN总线的攻击	(3) CAN重放攻击 (4) 总线关闭攻击 (5) 消息篡改攻击 (6) 消息注入攻击	(3) Fröschle等人[43] (4) CANnon[46] (5) CANflict[47] (6) Miller等人[1]
车载网络防御技术	车载网络入侵检测技术	采用不同技术对车载网络攻击进行检测	(1) 基于特征工程的入侵检测 (2) 基于CAN消息的入侵检测	(1) Viden[48] (2) PercepGuard[69]
	车载网络协议安全增强技术	对CAN协议的机密性、完整性和身份鉴别能力进行增强	(1) 基于轻量级密码学的通信机密性增强 (2) 基于高效HMAC的消息身份鉴别和消息完整性增强	(1) SENECAN[73] (2) CANTO[81]

3 车载网络协议逆向

汽车行业中各类厂商, 出于各种原因 (例如商业保密要求), 一般不会公开其车载网络的相关协议实现细节. 因此, 研究者即使获得原始的 CAN 总线数据流, 在没有一定逆向研究的前提下, 很难推测出消息的具体语义. 此外, 攻击者要向 CAN 总线上注入恶意消息, 首先需要了解消息的格式和语义. 对车载网络攻击进行防御, 也需要了解消息的格式和语义, 从而检测消息流中的恶意消息. 因此, 要对车载网络进行安全研究, 车载网络协议的逆向是研究者很难绕开的话题. 近些年, 研究者利用多种技术, 对车载网络协议进行逆向.

3.1 协议逆向过程

CAN 信号是描述车辆行为的有效载荷块. 每个信号都表示车辆功能相关的实时信息, 如车门状态、当前速度等. CAN 信号是车载网络协议逆向的基本单位, 作为一个令牌 (token). 令牌是车载网络协议逆向的基本单位, 主要包括如下几种类型[10].

(1) 物理值 (Physical): 一般用于表示实时的车辆动态, 例如车速、转弯角等.

(2) 状态值 (Status): 表示一组有限的状态, 例如车门打开/关闭.

(3) 计数器 (Counters): 在特定范围内作为循环计数器的信号.

(4) 校验码 (Checkcodes): 有效载荷还可以包含额外的校验码, 通常作为有效载荷中的最后一个信号.

车载网络逆向输入是原始的 CAN 数据流, 获取 CAN 数据流最常用的方法是将 CAN 数据记录仪通过 OBD-II 接口连接到 CAN 总线上. CAN 数据记录仪输出一个包含 CAN ID、DLC 和有效载荷数据等字段的文本文件. 此外, CAN 数据记录仪还可以通过 OBD-II 接口向 CAN 总线写消息

车载网络逆向的输出是 DBC (database CAN) 格式的文件. DBC 文件是一种表示 CAN 信号信息的可读文件, 包含了 CAN 信号的位置、语义和格式以及取值范围等信息[10]. 一些 CAN 分析软件如 CANtact、CANalyst-II[11]

可以将 DBC 文件作为输入, 输出可视化的 CAN 信号视图. 然而原始的 DBC 文件由 OEM 厂商持有, 不对外公开.

一些研究者对 CAN 总线进行了人工逆向, Currie 等人[12]利用 OBD-II 接口提取 CAN 消息流, 然后观察对车辆执行的不同操作时, 不同字节如何随时间的变化来解释这些消息, 并重放这些消息来操作相应的功能. 该方法具有很大的局限性, 需要事先了解需要逆向的车辆的具体细节, 不具备通用性. 人工逆向是一个耗时和繁琐的过程, 且无法扩展. 因此, 一些研究者对 CAN 总线进行了半自动化和自动化的逆向工作.

车载网络协议逆向的目标是定位 CAN 信号在 CAN 数据帧中的位置, 将 CAN 数据帧切分为 CAN 信号, 该过程称为载荷令牌化 (tokenization); 然后再分析每一个 CAN 信号的语义和格式, 该过程称为令牌解析. 图 4 展示了 CAN 逆向过程和 DBC 文件的格式.

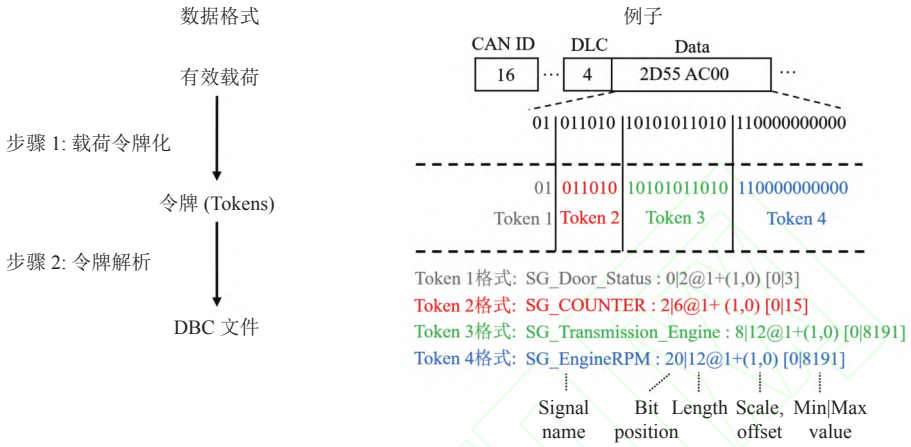

图 4 CAN 逆向过程示意图[10]

图 4 中, CAN 逆向过程的输出是 DBC 文件, 其一条信息可以表示一个 CAN 信号的格式, 从左到右依次为信号名、比特位、信号长度、精度、取值范围, 例如图 4 中 DBC 文件 Token 4 的格式为: SG_EnginePRM 表示该信号名为发动机转速信号, 20|12 表示该信号位于第 20 个比特位, 信号长为 12 位, (1, 0) 表示精度为 1, 偏移为 0, [0|8191] 代表其取值范围为 0–8 191. 下面分别介绍载荷令牌化和令牌解析的相关研究工作.

3.2 载荷令牌化

载荷令牌化的目标是识别 CAN 帧的有效载荷中的每个 CAN 信号. 相关方法可以分为两类: (1) 基于组合优化的方法, 从一组可能的令牌组合中提取分数最大的令牌集, 或者 (2) 基于比特翻转率 (bit filp rate, BFR) 的方法, 从有效负载序列中计算 BFR 数组, 并扫描它以根据值的下降确定令牌的边界.

(1) 基于组合优化的载荷令牌化

基于组合优化的方法设置一个组合优化分数函数, 在令牌组合候选集中选择使该分数函数最大的令牌切分组合. ACTT[13]提出一种从 CAN 消息中提取信号的简单算法, 并使用 OBD-II 接口的参数标识 (PID) 对其进行标记. 该算法首先在车辆在行驶过程中识别通过 OBD-II 接口注入 PID 请求触发的目标轨迹部分, 并获得令牌的时间序列. 然后在所有位上计算可能的令牌组合. ACTT 使用线性回归计算候选集中每个令牌的时间序列与车辆诊断协议 ID 时间序列之间的适应度分数, 并基于动态规划的调度算法输出一组不重叠的令牌, 使适应度分数最大化. 但是该工作只考虑了由不连续的常数位集合组成的信号, 无法处理由连续常数位集合组成的信号.

德国 CISPA 的研究者提出了用于汽车隐私和逆向分析的工具 AutoCAN[20], 该工具包含一个 tokenizer. 对每个 CAN ID 对应的消息, 该 tokenizer 从有效负载的第 1 个比特开始扫描, 在每次迭代中, 算法通过验证其属性是否与任何信号类别 (物理值、计数器、校验值、状态值) 的属性匹配来评估 CAN 信号的属性. 这是第 1 个通过单向扫描有效载荷来迭代评估连续比特序列的方法. 与以前的工作相比, 大大减少了计算开销.

CAN-D[14]提出了一种基于组合优化的载荷令牌化方法, 该方法考虑了每个比特翻转的条件概率, 以及后面两

个比特的条件翻转概率之差, 并构建出一个组合优化函数. 一旦计算出有效载荷中每个位作为信号边界的概率, 该算法提取所有可能有效令牌的集合. 该组合优化算法的目的是, 在将有效载荷划分为多 CAN 信号和为同一 CAN 信号分配多位之间找到最优平衡, 避免陷入局部最优.

(2) 基于比特翻转率的载荷令牌化

该方法的基本思想是, 不同 CAN 信号之间的比特翻转率很可能不同, 一个 CAN 信号内的比特翻转率接近. TANG[16]和 READ[17]同时提出了基于比特翻转率的载荷令牌化方法. 比特位翻转对应于负载中位状态的变化, 从 0 到 1 或反之亦然. 对于每一个 CAN ID 序列, 计算一个比特位翻转率数组 $B = [b_1, b_2, \ldots, b_n]$, 其中 n 是 CAN 帧有效载荷的长度, b_i 代表了第 i 个比特的翻转率. 然后从左到右扫描 BFR 数组, 查找在 b_i 和 b_{i+1} 之间的显著下降的值. 该边界值很可能是两个信号之间的边界. 尽管 TANG 和 READ 是载荷令牌化的早期工作, 但是其完成度较高. 后续大多数基于比特翻转率的载荷令牌化工作都基于该方法并进行了调整和扩展. LibreCAN[15]扩展了该算法, 将连续两个比特翻转率的下降表示为百分比, 从而可以以更高的精度确定最佳阈值. 在 LibreCAN 中, 作者定义了两个指标 CE/TE 和 CE/TBDC, 用于载荷令牌化算法. CE/TE 是正确标记的令牌数 (CE) 与提取的总令牌数 (TE) 之间的比率, 用于衡量精度. CE/TBDC 是正确提取的令牌数量 (CE) 与 DBC 文件中与目标车辆相关的信号总数 (TDBC) 之间的比率, 用于衡量覆盖率.

3.3 令牌解析

经过载荷令牌化过程, CAN 信号在有效负载中的位置已知, 但其语义含义和格式尚未被翻译. 令牌解析阶段将解析 CAN 信号的格式和语义, 输出 DBC 文件. 令牌解析相关工作可以分为如下 3 类.

(1) 基于 PID 注入的令牌解析

基于 PID 注入的令牌解析, 通过向 OBD-II 接口注入参数标识 (PID), 并观察 CAN 数据流的变化, 分析 CAN 信号的语义. 这是很多令牌解析的基础方法, 一些其他类别的工作也需要结合基于 PID 注入的令牌解析方法.

ACTT[13]在翻译阶段, 采用了基于 PID 注入的方法, 并计算提取的令牌和诊断消息之间的线性相关性, 从而提高翻译的准确度. CAN-D[14]同样采用了基于 PID 注入的令牌解析方法, 而且是第一种可以解码信号正负的翻译方法. 其方法基于以下假设: 如果信号包含正负值, 则当它从正值变为负值时, 其两个最高有效位都会从 0 翻转到 1, 反之亦然. 其背后的逻辑是, 有符号的 CAN 信号在两个最高有效位的值中具有连续性.

基于 PID 注入的令牌解析方法可以用来准确地翻译一些对车辆功能至关重要的信号. 由于车辆诊断协议响应信号的格式是公开的, 一旦 CAN 信号的语义被正确识别, 其格式信息也被获取. 该方法的主要局限性是并非所有的信号都可以通过 PID 触发, 特别是那些与汽车动力系统无关的信号.

(2) 基于语义分类的令牌解析

该方法利用语义相似的 CAN 信号组的内在属性, 以及这些组之间语义的联系解析 CAN 信号. 基于语义分类的令牌解析需要对信号的性质有深刻的理解, 以及将这些知识转化为软件表示的能力. 因此该方法需要操作人员负责执行搜索信号性质的操作, 以收集令牌解析过程的数据. 因此大多数基于语义分类的令牌解析工作都是半自动化的.

LibreCAN[15]是第一个可以逆向 CAN 总线车辆运动消息和车身信息的 CAN 逆向工程框架. LibreCAN 通过归一化互相关 (normalized cross correlation), 通过在提取的 CAN 信号、PID 和 GPS 数据之间寻找对应关系进行令牌解析. 在得到信号的语义信息后, LibreCAN 在信号的原始值和真值数据之间执行线性回归, 以解码 CAN 信号的精度和偏移等信息. LibreCAN 将逆向 CAN 消息所需的时间减少到 40 min 左右, 并在 4 种真实的车辆上进行了测试, 均显示出了较为良好的结果. 然而 LibreCAN 既不能处理通过应用层协议传输的长消息, 也不能发现字段和信号之间的非线性关系.

AutoCAN[20]通过查找令牌之间的语义相关性实现令牌解析. 其设计原理是, CAN 信号代表了现实世界中的事件, 这些事件受物理定律的约束. 在翻译过程中, AutoCAN 通过计算 CAN 信号与关联真实数据的相关性来解码最初的部分令牌, 这些真实数据通过外部传感器获取. 然后通过抽象语法树迭代提取一些数学公式, 并应用于转换其

余的令牌, 找到这些转换的输出与已知信号之间的相关性. 例如, 由于速度等于加速度的积分值, 可以通过计算每个令牌的积分, 并识别积分值与当前车速的关联性, 翻译加速度相关的 CAN 信号. AutoCAN 分析显示, 汽车制造商可以跟踪汽车 GPS 位置、车内人员数量、门和空调的使用统计数据, 此外研究者还发现 OEM 植入了远程关闭汽车或在司机超速时警报的功能. Buscemi 等人[19]研究了基于语义分类的逆向工程流程的优化, 作者建议收集多个短 CAN 信号序列, 而不是记录一个长 CAN 信号序列, 有助于提高翻译的准确度.

(3) 基于配套应用的令牌解析

智能网联汽车的配套应用 (companion app) 是运行在移动端和桌面端的应用程序, 可以与车载网络进行交互. 运行在车主手机的车控应用是一类配套应用, 一般通过蜂窝网络与汽车 T-Box 通信, 向车载网络发送指令控制汽车相关功能. 另一类配套应用运行在车载诊断工具上, 通过 OBD-II 接口与车载网络发送消息. 基于配套应用的令牌解析借助配套应用与车载网络交互, 翻译 CAN 信号.

CANHunter[18]是第一个通过使用车载配套应用程序自动化 CAN 逆向工程的工具. 作者将这些应用程序分为车载娱乐应用和车载诊断工具应用程序. 车载诊断工具应用程序通常由售后公司 (例如保险公司) 开发, 需要一个连接到 OBD-II 接口的车载诊断工具. 由于每个应用程序的逻辑和架构不同, 因此 CANHunter 解决的第一个挑战是找到向 CAN 总线发送请求的模块并了解其功能. CANHunter 使用了静态分析中程序切片的技术定位 CAN 总线发送请求的模块. 此外, CANHunter 提出了一种动态强制执 CAN 命令的方法, 可以提取车载诊断工具应用程序发送的所有消息. CANHunter 使用 3 个车载娱乐系统应用和 104 个车载诊断工具应用, 识别和解码超过 150 000 个 CAN 信号的语义. 然而, 这些应用程序大多数是为想要从车辆中提取普通信息的驾驶员设计的, 因此可获得的大多数 CAN 信号都是对 PID 的响应, 其文档已经公开. 此外 CANHunter 也不提供 CAN 信号的格式.

大多数车载网络协议逆向工作主要针对 CAN 协议, 这些工作无法直接应用于车载诊断协议. 针对上述问题, Yu 等人提出了一种车载诊断协议逆向框架 DP-Reverser[21], 可以对 KWP 2000 和 UDS 协议进行了逆向. 作者在 18 个车辆上使用了 4 种专业的诊断工具, 进行了较大范围的实验. DP-Reverser 本质上是对车载诊断协议人工逆向的模拟, 其易用性、可扩展性以及效率仍存在不足. 目前针对车载诊断协议的逆向工作目前相对较少, 安全界需要一种更加易用高效的诊断协议逆向方案, 这也是未来的一个研究方向.

3.4 小 结

车载网络逆向主要分为 3 个步骤, 首先需要进行数据收集, 然后需要进行载荷令牌化, 将 CAN 消息中有效载荷切分为最小的语义单元-CAN 信号, 相关方案分为两类. 表 3 总结了载荷令牌化相关工作的方案类型、算法涵盖的信号种类以及复杂度.

表 3 载荷令牌化相关工作和算法总结

方案类型	相关工作	涵盖信号种类				复杂度
		物理值	状态值	计数器	校验码	
基于组合优化	ACTT[13]	√	⊗	√	√	$O(N^2)$
	AutoCAN[20]	√	√	√	√	$O(N\log N)$
	CAN-D[14]	√	√	√	√	—
基于比特位翻转率	READ[17]	√	√	√	√	$O(N)$
	TANG[16]	√	⊗	⊗	√	$O(N^2)$
	LibreCAN[15]	√	√	√	√	$O(N)$

车载网络逆向的最后阶段是令牌解析, 该阶段分析每个 CAN 信号的格式、语义等信息, 表 4 总结了不同令牌解析相关工作的方案类型、自动化程度、软硬件要求和所需时间.

车载网络协议一般闭源, 研究者只能通过间接的方法和工具进行逆向工作, 其准确性、通用性和效率仍存在很大问题. 在保证汽车厂商保密需要的前提下, 研究者和相关厂商需要密切合作, 从而降低研究者了解车载网络协议的门槛.

表 4 令牌解析相关工作和算法总结

方案类型	相关工作	自动化程度	软硬件要求	所需时间
基于PID注入	ACTT[13]	完全自动化	从OBD-II接口进行PID注入	20 min左右
	CAN-D[14]	完全自动化	从OBD-II接口进行PID注入	4 min左右
基于PID注入和语义分类	LibreCAN[15]	半自动化	GPS数据, 从OBD-II接口进行PID注入, 需要操作员根据要求进行人工数据收集	40 min左右
基于语义分类	AutoCAN[20]	完全自动化	GPS数据	小时级别
	Buscemi等人[19]	半自动化	GPS数据, 需要操作员根据要求进行人工数据收集	5 min左右
基于配套应用	CANHunter[18]	完全自动化	配套应用, 主要为手机安装的车控APP	几分钟到几小时不等
	DP-Reverser[21]	完全自动化	配套应用, 摄像头, 机械臂	分钟级别

注: 所需时间是根据论文采用的数据和实验平台得出的, 仅供参考

4 车载网络攻击技术

智能网联汽车的发展带来了一系列新的潜在安全风险. 本节首先概述智能网联汽车的车载网络攻击, 对其攻击向量进行整体建模, 将攻击者分为本地攻击者和远程攻击者, 两类攻击者分别可以通过物理访问, 近距离无线访问和远距离无线访问入侵车载网络. 然后分别根据攻击入口和攻击目标分类介绍车载网络攻击相关研究, 最后进行小结.

4.1 概 述

本文主要从攻击入口和攻击目标两个维度进行威胁建模, 梳理当前车载系统攻防研究相关工作. 针对车载网络的攻击研究可以大致分为两个阶段. 第 1 阶段 (2010–2015 年), 研究者重点关注通过物理访问连接汽车并入侵车载网络, 例如通过 OBD-II 接口连接车载网络, 或者物理访问 ECU 并向车载网络发送恶意消息; 第 2 阶段 (2015 年至今), 2015 年 Miller 等人[1]在 BlackHat 大会上公开了针对 Jeep 汽车的远程攻击, 黑客通过漏洞远程控制车辆转向导致事故, 研究者开始关注通过远程访问对智能网联汽车进行攻击. 这些攻击在最后阶段大多需要入侵车载网络, 从而控制车辆, 该阶段最有可能造成严重危害并威胁乘客生命安全.

综上所述, 车载网络可能遭受两类攻击者攻击.

• 本地攻击者 (local attacker). 这类攻击者可通过物理访问的方式入侵车载网络, 例如通过物理连接 ECU, 或者利用车载诊断工具连接 OBD-II 接口入侵车载网络. 车载网络缺乏身份鉴别和完整性验证机制是这类问题的原因之一, 本地攻击者的破坏能力和防御难度较大, 但攻击实现的条件较高.

• 远程攻击者 (remote attacker). 这类攻击者通过智能网联汽车暴露的远程接口入侵车载网络, 远程接口可以分为近程通信接口如蓝牙, 以及远程通信接口如蜂窝网络等. 用户可以使用车控 APP (例如 Tesla APP) 操控汽车, 通过 5G 移动蜂窝网络、蓝牙等通信方式与车辆进行交互, 远程攻击者同样可以利用相同的接口入侵智能网联汽车, 例如攻击者入侵用户的手机并通过车控 APP 恶意解锁汽车.

图 1 展示了车内系统的主要威胁, 包括本地攻击者以及远程攻击者. 二者的共同点是最终都渗透车载网络, 尝试注入恶意消息, 破坏车辆正常功能. 车载网络安全研究的第二阶段中, 研究者逐渐重视车载网络的远程攻击, 攻击者可以先攻击车主的手机, 通过车控 APP 的相关接口向汽车发送恶意指令, 也可以通过 WiFi 和蜂窝网络远程入侵车载娱乐系统或者 T-Box, 获取 root 权限后, 再通过一系列攻击路径向 CAN 总线注入恶意信息. 随着智能网联汽车外部接口逐渐增多, 远程攻击实施的成本和难度不断降低, 破坏性不断增大. 但是远程攻击者的专业背景要求较高, 需要攻击者对车载网络协议如 CAN 消息格式有一定了解.

自 2010 年以来, 随着各类汽车软件漏洞和黑客攻击事件的不断曝光, 研究者开始逐渐重视智能网联汽车的安全研究. 安全研究的首要任务之一就是对被攻击目标的脆弱性进行全面分析, 然后总结出可能的攻击面和攻击向量, 这对车载网络安全研究也同样适用. 研究者通过对车载网络协议进行逆向, 了解掌握相关实现细节, 并通过理论分析、实验等方法对车载网络的脆弱性进行分析. 相关文献 [6–9] 对智能网联汽车的攻击面进行了分析和总结,

他们通过实验的方法验证并总结了攻击车载网络的访问方式.

(1) 物理访问: 本地攻击者可以直接物理访问汽车某些接口, 或者利用一些外部设备, 间接物理访问汽车的外部接口, 例如 OBD-II 接口、USB、CD 播放器等.

(2) 短距离无线访问: 远程攻击者可以通过蓝牙、远程无钥匙进入系统等接口入侵汽车, 此外可以通过干扰传感器, 执行传感器欺骗攻击, 使自动驾驶系统做出错误决策.

(3) 远距离无线访问: 远程攻击者通过广播通道, 例如 GPS, 或者可寻址通信通道, 例如蜂窝网络入侵智能网联汽车, 进一步渗透车载网络.

本节首先对智能网联汽车的攻击进行整体建模, 将攻击者分为本地攻击者和远程攻击者, 其访问汽车的方式分别为物理访问和无线访问. 然后再根据攻击入口的不同, 将车载网络攻击分为通过物理接口的攻击, 通过传感器的攻击, 通过近程通信的攻击和通过远程通信的攻击. 结合相关文献, 具体分类和总结如表 5 所示.

表 5 根据攻击入口对车载网络攻击技术分类

访问方式	攻击入口	攻击方法	代表工作
物理访问攻击	通过物理接口的攻击: (1) 通过OBD-II接口攻击 (2) 通过USB/CD的攻击	(1) 通过OBD-II接口进行数据篡改、消息窃听或者拒绝服务攻击 (2) 通过USB接口进行数据篡改或者侧信道攻击	(1) Plug-N-Pwned[23] (2) DeepSniffer[22]
远程访问攻击	通过传感器的攻击: (1) 摄像头欺骗攻击 (2) 雷达传感器欺骗攻击	(1) 对视觉传感器的输入进行扰动 (2) 干扰激光雷达、毫米波雷达的输入	(1) AttrackZone[27] (2) mmSpoof[35]
	通过近程通信的攻击: (1) 车载蓝牙攻击 (2) 无钥匙进入系统攻击	(1) 利用蓝牙协议漏洞攻击车载蓝牙系统 (2) 对无钥匙系统进行攻击	(1) Antonioli等人[36] (2) Xie等人[40]
	通过远程通信的攻击: (1) WiFi攻击 (2) 蜂窝网络攻击	(1) 通过WiFi入侵汽车 (2) 通过蜂窝网络攻击汽车T-Box, 或通过受害者手机攻击汽车	(1) Over-the-air[3] (2) Free-fall[2]

进一步地, 根据攻击目标不同, 将车载网络攻击总结如表 6 所示.

表 6 根据攻击目标对车载网络攻击分类

攻击目标	攻击技术	具体攻击方法	代表工作
ECU	ECU伪装攻击	冒充正常的ECU发送消息	Cloaking[42]
	重刷攻击	通过物理连接或者OTA等方式修改ECU上的固件或者软件	Over-the-air[3]
CAN总线	CAN重放攻击	重放车载网络上的消息	Fröschle等人[43]
	CAN总线关闭攻击	利用CAN协议相关漏洞, 使CAN总线进入总线关闭状态	CANnon[46]
	CAN消息篡改攻击	篡改CAN总线上的消息	CANflict[47]
	CAN消息注入攻击	向CAN总线注入恶意消息	Miller等人[1]

攻击者通过上述攻击入口渗透进智能网联汽车系统后, 如果要进一步操控汽车并执行恶意操作, 就需要通过 CAN 总线向特定 ECU 发送特定指令. 如表 5 所示, 攻击者均可以通过上述攻击入口获取 CAN 总线的部分或者全部控制权, 从而完全控制车辆. 由此可见, CAN 总线的安全是智能网联汽车安全的底线, 本文列举了部分黑客入侵车载网络的具体方式.

(1) 将黑客设备连接到车辆诊断维修接口 (OBD-II), 获得与汽车 CAN 总线的直接物理访问.

(2) 通过受感染的第三方设备, 例如存在漏洞的第三方 ECU, 或者车载诊断工具进入 CAN 总线.

(3) 通过智能网联汽车与互联网的无线通信、蓝牙等实现与 CAN 总线交互.

(4) 通过 OTA (update over the air) 无线更新服务接口入侵 CAN 总线.

攻击者入侵车载网络后, 可能需要进一步攻击车载系统其他部分, 形成完整的攻击链, 从而执行特定的恶意功能, 本文列举了部分攻击者入侵 CAN 总线后, 可能执行的一些恶意功能.

(1) 数据窃取, 汽车中有些设备可以收集驾驶者相关信息, 通过入侵这些设备可以实现数据窃取.

(2) 控制车辆, 攻击者发送恶意 CAN 数据包破坏车辆驾驶行为.

(3) 数据欺骗, 向驾驶员或自动驾驶系统反馈一些错误信息, 导致错误的驾驶决策.

4.2 攻击入口

表 5 总结了车载网络的攻击入口, 大部分针对智能网联汽车的攻击, 如果黑客需要进一步控制车辆功能, 扩大攻击的影响范围, 均需要对车载网络进行攻击. 本节根据攻击入口和攻击方式, 对车载网络攻击进行分类.

4.2.1 物理访问攻击

本地攻击者可以通过物理访问方式连接车载网络. 其中, OBD-II 接口是一种汽车故障检测和诊断接口, 通过 OBD-II 接口可以获取车辆的实时和历史数据. 然而, 由于其设计之初并没有考虑安全问题, 因此 OBD-II 接口存在着一些潜在的安全漏洞, 攻击者可以利用这些漏洞进行攻击, 引起车辆的安全风险. 此外, 车载娱乐系统一般配备 USB 接口, 用于系统与外部设备的连接. 一旦攻击者获得了这些物理接口的访问权限, 就可以执行进一步的攻击. 通过 OBD-II 接口或 USB 接口等物理接口, 可以执行数据篡改、信息窃取、拒绝服务和侧信道攻击[22].

Plug-N-Pwned[23]提出了一种针对车载诊断工具的攻击框架, 该框架可以自动化地扫描和分析车载诊断工具, 检测和利用其中的安全漏洞, 其包括 3 个主要模块: 扫描模块、攻击模块和渗透测试模块. 扫描模块可以自动识别车载诊断工具, 并获取其基本信息和特征; 攻击模块可以利用已知的安全漏洞对车载诊断工具进行攻击, 例如发送特定的命令和数据包来控制车辆的各种系统; 渗透测试模块可以模拟攻击者的行为, 对车载诊断工具进行全面的安全测试和评估. 攻击者可以利用该攻击实现车载网络的数据篡改和信息窃听.

4.2.2 无线访问攻击

远程攻击者可以对智能网联汽车实施非接触式攻击. 本节将其分为通过传感器的攻击、通过近程通信的攻击和通过远程通信的攻击.

• 通过传感器的攻击

智能网联汽车需要不同功能的传感器来实时获取周围环境信息, 例如摄像头、激光雷达和毫米波雷达等, 这些传感器的测量结果将作为自动驾驶系统的输入, 其准确性对于车辆的行驶安全至关重要. 传感器欺骗攻击发生在自动驾驶的信息收集阶段, 攻击者通过使传感器获得错误的测量值, 使自动驾驶系统做出错误判断, 其最终结果是向车载网络中注入恶意消息, 使车辆产生恶意行为. 传感器种类众多, 本文重点关注两类攻击, 一类是针对视觉传感器尤其是摄像头的欺骗, 另一类是针对雷达传感器的欺骗攻击.

(1) 摄像头欺骗攻击

摄像头是自动驾驶系统进行环境感知、即时定位与地图构建 (simultaneous localization and mapping, SLAM) 的最重要传感器. ICSL 攻击[24]探索了利用人类不可见的红外光进行摄像头欺骗攻击. 该攻击发现不可见的红外光可以成功触发图像传感器, 而人眼无法感知红外光. 此外, 红外光在相机中呈现洋红色, 与环境可见光触发不同的像素点, 可以作为 SLAM 过程中的关键点, 从而诱导自动驾驶系统做出错误决策. 作者演示了利用 ICSL 攻击可以产生虚假的红绿灯, 生成虚假的不可见的障碍物. 作者在特斯拉 Model 3 和企业级自动驾驶平台上进行了实验评估, 验证了 ICSL 攻击的有效性. 证明其会带来严重的安全问题.

Jing 等人[25]对自动驾驶汽车的车道检测模块进行攻击, 通过对输入图像进行微小的扰动, 可以欺骗车道检测系统, 使其错误地识别车道线或在检测到车道线时产生误报. 攻击者可以使用此方法将车辆从其预定路径偏离. 作者通过设计多种不同类型的扰动模式, 展示了攻击的有效性.

自动驾驶系统在视觉感知中依赖目标检测和目标跟踪. Zhao 等人[26]提出针对目标检测器的物理对抗攻击, 能够实现远距离 (>20 m)、宽角度以及不同光线与背景下的高鲁棒性对抗样本攻击. AttrackZone[27]提出了一种新的物理可实现的目标跟踪器劫持攻击. 该攻击利用三维点云数据构建攻击区域, 然后使用孪生候选区域生成网络 (siamese region proposal network) 的热图生成物体的边界框, 导致目标追踪器识别出错误的信息, 例如虚假的车辆或者行人. 实验评估显示, AttrackZone 在 92% 的时间内实现了攻击目标, 平均只需要 0.3–3 s.

基于深度估计的避障技术已被广泛应用于自动驾驶系统中. 它通常依靠摄像头自动检测障碍物并做出驾驶决策, 例如, 在识别障碍物并在前方几米处停车. Zhou 等人[28]研究了用于避障的基于立体视觉的深度估计算法的安全风险, 提出了一种欺骗基于深度感知的避障系统的远程攻击 DoubleStar. 其包括两种攻击形式: 光束攻击和球攻击, 它们分别利用投射光束和透镜耀斑球来造成错误的深度感知. 实验结果表明, DoubleStar 在夜间可创建 15 m 的假深度, 白天可创建 8 m 的假深度, 从而使自动驾驶系统识别虚假的障碍物.

(2) 雷达传感器欺骗攻击

激光雷达通过发射激光并捕获反射来计算与障碍物的距离, 是智能网联汽车重要的传感器之一. 激光雷达可获取周围环境的三维点云数据, 并用于感知和识别障碍物、道路标志等关键任务. Cao 等人[29]首次对激光雷达的感知过程进行了安全研究. 作者将激光雷达欺骗攻击作为威胁模型, 并将攻击目标设定为欺骗目标车辆前方有障碍物. 研究发现, 简单应用激光雷达欺骗不足以实现该目标, 因为其受到基于机器学习的目标检测过程的限制. 该论文主要有两个贡献, 首先, 作者通过实验分析了激光雷达欺骗攻击的能力, 并设计了基于全局空间变换的方法来对这种能力进行数学建模. 其次, 作者确定了直接使用优化方法攻击的固有局限性, 并设计了一种将优化和全局采样相结合的算法, 最终能够将攻击成功率提高到 75% 左右.

Zhu 等人[30]提出了一种新的攻击框架, 攻击者可以借助该框架识别物理空间中的几个对抗位置, 通过在这些位置周围放置具有反射表面的任意物体作为对抗样本, 可以欺骗激光雷达感知系统. 作者证明仅使用两架商用无人机就可以轻松执行所提出的攻击. Sun 等人[31]提出了一种针对激光雷达的黑盒对抗攻击, 这种攻击不需要对目标系统有先验知识, 也不需要访问模型结构和权重, 只需要在输入数据上添加微小的扰动即可欺骗模型. 该攻击能够欺骗大多数现有的基于激光雷达的物体检测和跟踪算法, 并在现实场景中进行了验证.

Cao 等人[32]提出了一种针对激光雷达的传感器欺骗攻击, 称为物理消除攻击 (physical removal attacks, PRA). 该攻击通过一种人眼不可见的方法, 采用基于激光的欺骗技术, 在传感器层面选择性地去除真实障碍物的激光雷达点数据, 然后将其用作自动驾驶系统的输入. 这些关键的激光雷达信息的消融会导致自动驾驶障碍物探测器无法识别和定位障碍物, 从而导致自动驾驶汽车做出危险的自动驾驶决策. 作者在 3 种流行的自动驾驶平台 (Apollo, Autoware, PointPillars) 上进行验证, 以 92.7% 的成功率去除了 90% 的目标障碍物点.

虽然之前的攻击已经证明了操纵三位点云来欺骗目标探测器的可行性, 但是均未实现物理层面的欺骗. PLA-LiDAR[33]攻击使用激光注入对抗点云, 实现了对基于激光雷达的三维目标检测器的物理欺骗. 该攻击使用了一种新开发的激光收发器, 可以注入多达 4 200 个点, 这是以前工作的 20 倍, 并且可以通过测量受害激光雷达的扫描周期来调度欺骗激光信号. 该工具通过一种将点云坐标转换为控制信号的方法, 以及一种结合激光雷达物理约束和攻击能力的对抗性点云优化算法, 可以将欺骗三维点云物理注入到受害激光雷达中.

相较于摄像头和激光雷达, 毫米波雷达具有更高的精确度, 这对于自动驾驶至关重要. Sun 等人[34]对毫米波感知模块的安全进行了深入研究, 通过欺骗毫米波传感模块来控制受害者, 包括在任意位置添加假障碍物和伪造现有障碍物的位置. 作者构建了 5 个真实世界的攻击场景, 以欺骗受害者汽车, 迫使其做出危险的驾驶决策, 从而导致致命的车祸. 作者使用基于林肯 MKZ 的自动驾驶测试平台进行了实验, 研究了各种攻击场景的安全影响, 验证了攻击者可以通过欺骗毫米波雷达模块控制受害者车辆, 危及自动驾驶的安全性.

之前的针对毫米波雷达的攻击, 需要攻击者对毫米波有较强的先验知识, 而且需要攻击者和受害者之间进行消息同步, 降低了这些攻击的有效性. mmSpoof[35]提出了一种新的毫米波雷达欺骗攻击, 该攻击不需要任何先验知识, 也不需要攻击者和受害者进行同步, 使用基于反射阵列 (reflect array) 的物理设备, 该设备通过调制反射雷达信号来欺骗受害者的雷达.

• 通过近程通信的攻击

智能网联汽车需要结合蓝牙、RFID 等近程通信技术完成相应的功能, 例如无钥匙进入等功能, 这些近程通信接口也为攻击者带来了新的攻击面, 本文将通过近程通信的攻击分为如下两类.

(1) 车载蓝牙攻击

一些研究者已经充分论证了蓝牙的安全威胁, 然而针对车载蓝牙攻击的研究相对较少. Antonioli 等人[36]系统

评估了车载蓝牙协议层的安全风险, 作者在 5 个常见的车载娱乐系统和 3 个真实车辆上测试了两个蓝牙攻击 KNOB 攻击[37]和 BIAS 攻击[38], 这两个攻击结合可以伪装蓝牙设备. 实验发现所有的测试设备均易受这两个攻击, 证明了当前汽车的蓝牙协议普遍版本过时且缺乏安全特性.

(2) 无钥匙进入系统攻击

无钥匙进入系统通过无线射频技术 (RFID) 或蓝牙技术实现车辆的解锁和上锁功能, 使车主无须金属钥匙即可打开汽车. 针对传统汽车的无钥匙进入系统, 已经有攻击者通过无线中继攻击恶意解锁并盗窃车辆的例子. 随着车辆智能化程度不断提高, 其暴露的攻击面不断增加. Garcia 等人[39]对多个车型的无钥匙进入系统进行研究, 发现这些系统存在严重的安全漏洞. 作者通过分析现有的无线信号传输协议和加密机制, 揭示了攻击者进行无钥匙进入系统入侵的方式. 包括无线信号重放攻击、暴力破解攻击和中间人攻击等. 作者通过实验证明, 这些攻击方法可以使攻击者绕过无钥匙进入系统的安全防护措施, 成功地进入车辆并实施非法活动, 例如启动车辆、解锁车门等. Xie 等人[40]对特斯拉的无钥匙进入系统进行了攻击, 特斯拉会随车附送两张 RFID 卡片, 车主通过认证后会给手机下发一个 token, 车主就可以通过手机解锁汽车. 作者通过一张智能卡模仿了整个交互协议, 并且利用了特斯拉的高开锁延迟的漏洞, 通过的蓝牙中继攻击解锁汽车.

• 通过远程通信的攻击

智能网联汽车需要通过蜂窝网络等远程通信技术完成相应的功能, 使驾驶者可以远程操控汽车, 但这也给远程攻击者提供了较大的攻击面, 本文将通近程通信的攻击分为如下两类.

(1) WiFi 攻击

智能网联汽车丰富的联网特性在为使用者带来便利的同时, 也引入了很大的风险. WiFi 是智能网联汽车进行 OTA 升级的重要入口, 因此也往往被攻击者利用. 在针对特斯拉 OTA 升级的 Over-the-air 攻击[3]中, 攻击者首先伪造特斯拉 4S 店的 WiFi 热点, 特斯拉汽车会自动连接该 WiFi, 然后利用浏览器的漏洞进行进一步的攻击, 成功篡改了特斯拉的软件更新包, 将恶意代码注入车辆, 可以远程获得车辆的使用权, 在业界产生较大影响.

(2) 蜂窝网络攻击

车主可以通过手机远程连接并操控智能网联汽车, 如果车主手机被黑客操控, 黑客可以控制车控 APP, 通过蜂窝网络连接并访问汽车[8]. 腾讯科恩实验室的研究者演示了如何利用移动端 Tesla APP 的漏洞, 通过蜂窝网络入侵 CAN 总线, 进而完全操控特斯拉汽车[2,41]. 此外, 智能网联汽车的娱乐系统也需要通过 5G 蜂窝网络访问联网资源. 蜂窝网络通信依赖于车辆的 T-Box, 然而相关研究表明, T-Box 可能存在安全漏洞, 远程攻击者如果可以获取 T-Box 系统的管理员权限, 则可以向车载网络发送任意消息, 执行恶意功能[9].

4.3 攻击目标

攻击者通过物理访问或无线访问的方式入侵车载网络后, 其最终目标是攻击车载网络及其组成部分如 ECU, 进而控制车辆, 执行恶意功能. 根据攻击目标进行分类, 车载网络攻击的最终目标可能是 ECU 或者 CAN 总线, 相当于网络中的终端和网络总线. 车载网络攻击往往是一个攻击链, 一些最终针对 CAN 总线的攻击中间会利用 ECU, 所以本节根据其最终的攻击目标是破坏 ECU 还是 CAN 总线进行划分, 下面分别介绍这两类攻击.

4.3.1 针对 ECU 的攻击

针对 ECU 的攻击具体可分为如下两类.

(1) ECU 伪装攻击

由于 CAN 总线具有广播的特性, 如果攻击者可以访问 CAN 总线, 则攻击者可以获取总线上传输的所有消息, 因此攻击者可以学习 ECU 的行为方式, 包括标识 ECU 身份的 CAN ID, 传输速率, CAN 消息有效载荷的范围. 因为 CAN 消息中不包含收发者的相关信息, 所以攻击者通过构造相同的信息, 可以冒充正常的 ECU 发送消息. 一些车载网络入侵检测系统可以识别基本的 ECU 伪装攻击.

Cloaking 攻击[42]是一种新的 ECU 伪装攻击, 旨在绕过基于时钟信号特征的入侵检测系统. Cloaking 攻击通过两个攻击者控制的 ECU 节点配合完成, 分为两个步骤. 首先, 攻击者控制的节点 A 向攻击者控制的节点 B 发送一个需要掩盖的异常消息, 节点 B 将其时间戳设置为比实际时间早一段时间. 然后, 节点 B 将经过调整的消息重新

发送到总线上, 而节点 A 在接收到消息后再次将其时间戳设置为比实际时间晚一段时间, 并将其发送到 CAN 总线上. 这样, 被掩盖的异常消息看起来就像是由两个普通的节点发送的正常消息.

(2) ECU 重刷攻击

ECU 重刷 (reflashing) 攻击指攻击者试图在智能网联汽车中修改 ECU 上的软件或固件. 这种攻击可以通过直接物理接入 ECU, 或者通过网络连接实现, 例如使用钓鱼攻击或远程漏洞利用等技术, 获取 ECU 的访问权限. 攻击者可能会在 ECU 上植入恶意代码, 使其执行任意操作, 例如停用安全功能, 篡改控制系统, 或将车辆控制权转移给攻击者. ECU 重刷攻击是一种危险的攻击, 因为它可以长期影响汽车的安全性和性能. 攻击者可以利用这种攻击远程控制汽车, 从而危及乘客的生命安全.

腾讯科恩实验室在 Blackhat 大会上公开了针对特斯拉的 Free-fall 攻击[2], 该攻击通过利用一系列漏洞后, 绕过 ECU 固件的完整性检查, 对网关 ECU 进行 ECU 重刷攻击, 修改车载网络网关的固件, 从而可以在 CAN 总线上发送恶意消息.

4.3.2 针对 CAN 总线的攻击

攻击者可以通过各类攻击入口, 例如 T-Box 漏洞入侵车载网络, 利用 CAN 总线的安全缺陷, 最终发动针对 CAN 总线的攻击, 具体可分为如下 4 类攻击.

(1) CAN 重放攻击

攻击者进行 CAN 重放攻击的目标是尝试捕获 CAN 总线上的有效消息, 不加修改或者添加恶意的 payload 后再重新发送该消息, 从而伪造为正常 CAN 消息. 由于 CAN 信息中包含重要的实时信息, 因此 CAN 消息的延迟或不可用将影响整个 CAN 总线, 影响车载网络的可用性. 因此需要快速检测 CAN 重放攻击[43].

(2) CAN 总线关闭攻击

CAN 总线关闭 (CAN bus-off) 攻击是针对 CAN 总线的一类特殊的拒绝服务 (DoS) 攻击, 其主要目的是使 CAN 总线进入总线关闭状态, 从而使整个 CAN 总线上的通信受阻. 攻击者通常会向 CAN 总线发送大量的错误消息, 例如不合法的帧 ID、不合法的数据格式等, 这些错误消息会使 CAN 控制器错误地认为总线上发生了严重的通信错误, 从而触发总线关闭机制, 导致整个 CAN 总线上的通信中断. CAN 总线关闭攻击会对智能网联汽车的正常行驶造成严重影响, 使车辆失去控制, 从而带来巨大的安全风险.

CAN 总线有一套错误检测和恢复机制, 所有的 ECU 都可以检测错误, 检测出错误的单元会立即通知其他所有单元, 正在发送消息的 ECU 一旦检测出错误, 会强制结束当前的发送. 强制结束发送的单元会反复重发此消息, 直到成功发送为止. ECU 总是处于如下 3 种状态之一. 第 1 种是主动错误状态: ECU 可以正常参与 CAN 总线通讯的状态, 并可以在 ECU 检测出错误后, 输出一个主动错误标志, 并主动通知其他 ECU. 第 2 种是被动错误状态: ECU 易引起错误的状态, 处于此状态的单元虽能参加通讯, 但为不妨碍其他 ECU 通讯, 该 ECU 接收时不能积极发送错误通知. 第 3 种是总线关闭状态: ECU 不能正常参与 CAN 总线通讯的状态, 信息的发送接收都被禁止.

上述 3 种状态与 ECU 的发送错误计数 (transmit error counter, TEC) 和接收错误计数 (receive error counter, REC) 的值相关, TEC 统计一个 ECU 出现发送错误的次数, 而 REC 统计一个 ECU 出现接收错误的次数, 状态机根据 TEC 和 REC 的计数值决定进入何种状态. 3 种状态的转换关系如图 5 所示.

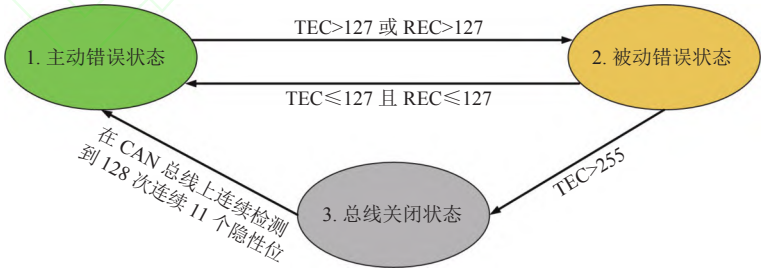

图 5 ECU 状态切换图

当 ECU 检测到错误后, 会发送带主动错误标志的错误帧, 当该 ECU 错误帧发送过多时, 会导致 TEC 或 REC 大于 127, 该 ECU 切换到被动错误状态. 当 CAN 总线恢复正常后, TEC 和 REC 开始减少, 直到 TEC 和 REC 均小于等于 127 后, ECU 恢复到主动错误状态. 当 TEC 累积到大于 255 后, 该 ECU 进入总线关闭状态.

由此可见, CAN 总线具有较强的容错性和鲁棒性. 但是这种容错机制并不是为防御针对 CAN 总线攻击而设计的, 而且 CAN 总线的错误控制机制本身存在缺陷. Cho 等人[44]对 CAN 总线的错误处理机制进行了研究, 发现 CAN 总线的错误处理机制可能被攻击者利用, 执行一系列攻击. 攻击者可以利用 CAN 总线错误处理的时间间隙, 伪造出现错误的 CAN 数据帧的 ID, 注入一些恶意信息, 在 CAN 总线处理完错误后会转发该恶意信息.

Serag 等人[45]通过详细分析 CAN 总线的错误处理机制, 发现了一些新的安全问题. 作者开发了一种测试工具 CANOX, 可以监视 CAN 节点在不同总线和错误条件下的行为, 并标记导致节点出现意外行为的条件. 基于 CANOX 发现的 CAN 总线的漏洞, 提出了扫描后打击攻击, 允许攻击者在没有车辆先验知识的情况下, 映射车辆的 CAN 总线, 识别安全关键的 ECU, 迅速将其关闭, 并持久地阻止其恢复. Kulandaivel 等人[46]认为现有工作不能同时做到远程、隐形和可靠, 因此很有可能被入侵检测系统识别. 因此, 作者提出了一种新的攻击 CANnon[46], 利用现代汽车微控制器单元的外围时钟门控特性, 远程攻击者可以通过纯软件方法冻结 ECU 的输出, 并在任何时间在 CAN 总线上插入任意位. 由于 CANnon 攻击产生的错误模式与自然错误难以区分, 而且不需要插入消息, 现有的入侵检测系统检测是困难的. 作者在两个真实车辆上演示了该攻击, 并给出了相应的防御措施.

(3) CAN 消息篡改攻击

攻击者可能对汽车 CAN 总线上传输的数据进行篡改, 发送被篡改过的数据包来干扰车辆的运行, 执行恶意功能, 例如修改车速、燃油水平等信息, 影响车辆的正常运行. 攻击可既可以通过物理方式, 直接连接 CAN 总线, 并利用相关工具 (例如车载诊断工具) 发送伪造的数据包; 也可以通过远程访问的方式, 入侵汽车中的网络设备, 如 T-Box、娱乐系统等, 然后利用漏洞来控制 CAN 总线的通信, 篡改 CAN 消息, 实现对车辆的控制[43].

CANflict[47]使用一种纯软件方法, 使攻击者从未经修改的微控制器在数据链路层发起 CAN 消息篡改攻击. CANflict 能够绕过最新的入侵检测机制, 验证了远程攻击 ECU 实现 CAN 消息篡改攻击是可行的, 并可以攻击同一 CAN 总线上的其他 ECU.

(4) CAN 消息注入攻击

攻击者可以通过其控制的恶意 ECU 向 CAN 总线注入恶意消息, 执行 CAN 总线消息注入攻击. CAN 总线消息注入攻击的目标是改变 CAN 帧的顺序、消息的频率、CAN 帧的数量或者消息的有效载荷, 执行恶意功能. 在 CAN 总线消息注入攻击中, 攻击者会向 CAN 总线发送伪造的 CAN 消息, 这些消息可能会模拟来自其他合法设备的消息. 这可以导致其他设备产生错误的行为, 例如关闭引擎, 恶意转向或者刹车. 在 Black Hat 大会上对 Jeep 汽车的攻击中, 攻击者最后通过向车载网络注入恶意消息, 使汽车恶意转向[1]. 攻击者可以通过正常的 CAN 消息实现消息注入攻击. 一种常见的方法是使用 CAN 总线相关工具, 例如 CANtact、CANalyst-II 等[11]向 CAN 总线发送定制的 CAN 消息完成攻击. 另外一种方法是物理访问 CAN 总线, 例如通过 OBD-II 接口注入 CAN 消息.

4.4 小 结

本节总结了智能网联汽车主要的攻击入口, 并分析了攻击者的访问方式、最终攻击目标、攻击开销和攻击后果, 总结如后文表 7 所示.

随着智能网联汽车的智能化和互联互通性的增强, 车载网络攻击的风险也在增加. 目前的研究已经揭示了许多潜在的攻击向量. 这些研究为车载网络防御技术研究提供了重要依据.

5 车载网络防御技术

为应对智能网联汽车的车载网络层出不穷的攻击, 研究者提出了一系列车载网络防御工作. 本节首先讨论车载网络防御技术的目标和挑战, 分析车载网络防御和攻击之间的逻辑联系. 然后分别介绍车载网络入侵检测和车载网络协议安全增强工作, 最后进行小结.

表 7 车载网络攻击总结

攻击入口	访问方式	攻击目标	攻击开销	攻击后果	引用
OBD-II接口	物理访问	ECU CAN总线	小	数据篡改、信息窃取、拒绝服务、入侵车载网络, 注入恶意消息等	Aliwa等人[9]
车载诊断工具	物理访问 短距离无线访问	ECU CAN总线	小 中		Plug-N-Pwned[23]
USB	物理访问	ECU 车载娱乐系统	中	数据篡改、信息窃取、拒绝服务、侧信道攻击、入侵车载网络	Koscher等人[6]
传感器	物理访问 短距离无线访问 长距离无线访问	ECU	中 中 大	欺骗传感器使自动驾驶系统产生错误决策, 向车载网络注入错误或恶意消息	AttrackZone[27] PLA-LiDAR[33] mmSpoof[35]
ECU	物理访问 短距离无线访问 长距离无线访问	ECU CAN总线	中 中 大	重放攻击、消息篡改、消息注入、拒绝服务和ECU伪装攻击等	Fröschle等人[43] CANnon[46] CANflict[47]
蓝牙	短距离无线访问	ECU	中	窃取车辆、入侵车载网络、ECU伪装攻击	Antonioli等人[36]
无钥匙进入系统	短距离无线访问	ECU	中	窃取车辆	Xie等人[40]
WiFi	短距离无线访问	ECU	中	通过OTA重刷ECU固件、入侵车载网络	Over-the-air[3]
蜂窝网络	长距离无线访问	ECU	大	窃取车辆、通过OTA重刷ECU固件、入侵车载网络	Free-fall[2]

5.1 目标和挑战

车载网络防御技术的目标是抵御不同能力攻击者从各个层次发起的不同种类的车载网络攻击. 攻击产生的根本原因是车载网络自身存在安全缺陷, 这些安全缺陷包括: 车载网络普遍缺乏消息加密和认证机制, 例如 CAN 总线具有广播的特性, 这也导致其容易遭受 CAN 重放攻击; CAN 总线还采用了基于 CAN ID 的优先级机制, 这也导致了其容易受到 CAN 总线关闭攻击[9]. 图 6 展示了 CAN 总线安全缺陷与车载网络防御工作之间的关系.

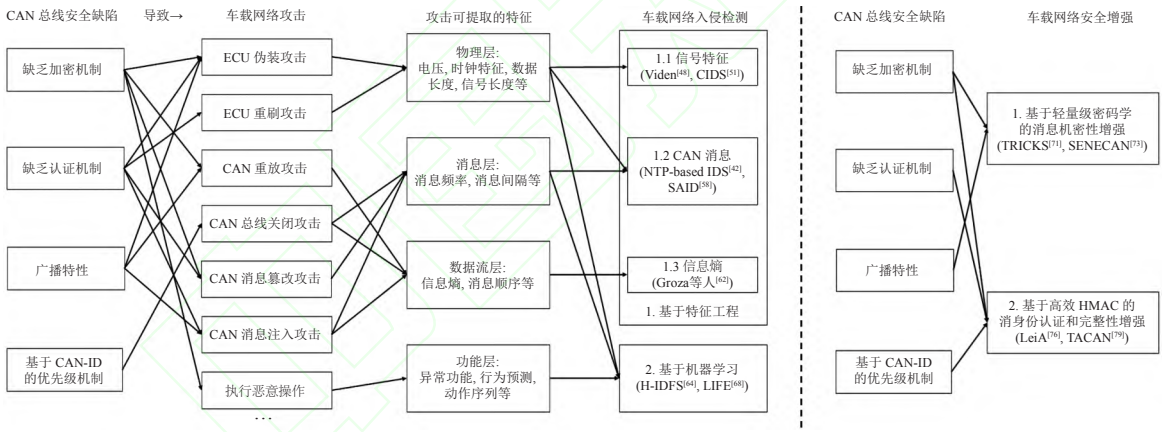

图 6 CAN 总线安全缺陷和车载网络防御相关工作的联系

入侵检测系统是网络防护的重要技术手段之一. 该技术对网络进行实时监视, 并且通过多种方法识别网络是否被攻击者入侵, 在发现可疑行为时发出警报或者主动采取措施. 入侵检测技术是一种主动积极的防护技术, 且实时性相对较强, 因此入侵检测技术非常适合车载网络的安全防护.

车载网络标准在设计时, 主要考虑其可靠性、实时性, 缺乏相应的安全设计. 最具有代表性的 CAN 总线, 因为其低成本、低延迟、容错率较高等特点得到了业界的广泛认可. 但是标准的 CAN 协议本身具有很多安全缺陷. 为弥补这些缺陷, 研究者提出了一系列车载网络协议的安全增强技术. 为提高 CAN 总线的机密性, 弥补 CAN 总线

缺乏加密机制, 以及消息广播的安全缺陷, 一类方法是基于轻量级密码学的通信机密性增强, 由于车载网络对实时性要求较高, 大多数 ECU 的算力有限, 因此需要采用轻量级的密码学加密方法确保机密性. 为了弥补 CAN 总线缺乏加密机制, 以及基于 CAN ID 的优先级机制的安全缺陷, 另一类方法基于高效 HMAC 的消息身份鉴别和完整性增强, 基于哈希的消息认证码 (Hash message authentation code, HMAC) 可以同时实现消息的身份鉴别和完整性验证, 且开销可以被接受.

然而, 车载网络的防御存在诸多挑战, 主要有如下 4 点.

(1) 与现有协议的兼容性. 修改 CAN 现有的数据字段或添加额外的认证字段与现有 CAN 标准冲突, 并导致与现有车辆使用的 CAN 不兼容. 在某些数据域附加验证信息需要修改报文定义和所有参与的 ECU, 这对于厂商来说是不可以接受的.

(2) 实时性约束. 在 CAN 数据包传送过程中, 过长的延迟可能导致安全问题, 因为它会减慢安全关键功能 (例如刹车和转向) 的响应时间. 因此, 针对车载网络的防御, 都应该小心任何数据包的延迟.

(3) ECU 算力约束. 大多数 ECU 的计算资源十分有限, 因此对防御系统的开销非常敏感.

(4) ECU 源代码约束. 在 ECU 上实现的车载网络防御系统可能存在源代码依赖, 如何克服 ECU 源码闭源的挑战, 也是研究者需要考虑的问题.

5.2 车载网络入侵检测技术

本节将车载网络入侵检测技术分为两大类, 基于特征工程的入侵检测和基于机器学习的入侵检测, 并根据具体方案进行详细介绍和比较.

5.2.1 基于特征工程的入侵检测

CAN 在物理层、网络层等不同层次上的特征不同, 每个 ECU 都可能有其固定特征. 研究者结合 CAN 总线逆向的相关知识, 使用特征工程的方法进行车载网络入侵检测, 具体可分为如下几类.

(1) 基于信号特征的入侵检测工作

ECU 具备一些固定的信号特征, 其中一类是电压特征, 恶意 ECU 无法改变电压这一硬件特征. 因此一些研究者利用 ECU 具有独特的电压特征识别恶意的 ECU. Viden[48]最早通过测量电压值的方法在车载网络中识别恶意 ECU, 首先检查信号的来源是否来自合法的消息发送器, 然后在学习阶段测量 ECU 的电压信号生成电压曲线, 并更新 ECU 的指纹特征. 最后, Viden 使用 ECU 的电压曲线在车载网络中检测到恶意 ECU.

VoltageIDS[49]通过对 CAN 信号的电压波形进行分析来检测异常行为, 可以应对 ECU 伪装攻击和总线关闭攻击, 具有较低的误报率和较高的检测率. VoltageIDS 通过观察一个合法 ECU 和另一个恶意 ECU 发送的相同信号, 来识别恶意 ECU 的电压特征. 此外, VoltageIDS 还可以区分总线关闭攻击和系统错误.

基于电压特征的车载网络入侵检测可以有效检测单个攻击者发起的 ECU 伪装攻击. 然而, Bhatia 等人[50]提出了一种新的 ECU 伪装攻击 DUET, 可以绕过之前基于电压的入侵检测系统, 攻击者控制两个 ECU 来破坏入侵检测系统记录的 CAN 总线电压, 从而绕过防御机制. 该攻击分为两个阶段: (1) 入侵检测系统重新训练模式: 攻击者操作一个受害者 ECU 的电压指纹; (2) 入侵检测系统操作模式: 另一个 ECU 冒充上一个 ECU 操作后的电压指纹. 为了避免 DUET 以及其他 ECU 伪装攻击, 作者提出了一种轻量级入侵检测系统 RAID[50], 使用一种独特的协议语言, 由 CAN 上的所有 ECU 在入侵检测系统重新训练模式期间使用, 保护 ECU 电压指纹不被破坏.

车载网络攻击还具有一些独特的信号特征, 例如时钟特征、信号长度等. CIDS[51]利用时钟信号检测异常的 CAN 帧. 该工作从 CAN 帧中提取时钟偏差作为 ECU 的指纹, 并模拟它们的时钟行为. 利用得到的指纹, 用递归最小二乘算法构建 ECU 时钟行为基准. 基于此基准, CIDS 使用累积和来检测识别错误中的任何异常转移, 因为这是入侵的明显标志. 如果检测到攻击, CIDS 的 ECU 指纹还可以帮助分析攻击的根本原因, 确定哪个 ECU 发动了攻击. 作者在 CAN 总线原型和实际车辆上的实验表明, CIDS 能够有效检测到车载网络攻击.

Scission[52]利用车载网络传输的信号的特征, 例如信号幅度、相位和频率等, 构建了一个信号特征数据库. 然后使用这个数据库来训练一个分类器, 识别车载网络中的数据包的发送者. 如果数据包的发送者被识别为异常, 则

可以将其标记为入侵, 并采取相应的措施进行防御. 从而能够检测异常和系统中受损 ECU 的身份识别. 实验结果表明, Scission 可以检测到来自受损和未受监控的基于 ECU 的攻击.

SIMPLE[53]基于单个 CAN 帧, 通过分析传输帧的物理层信号特征, 例如传输时间、起始位、传输速率等, 可以识别每个 ECU 节点. 该方法不需要修改或添加任何硬件或软件组件, 可以实现无缝集成和部署. 现有方法从整个 CAN 帧的符号中提取信号特征, 虽然具有较高的检测率, 但是其计算开销较大. EASI[54]提出了一种轻量级的方法. 作者发现一个 CAN 帧内的特征只有微小的变化. 因此 EASI 从单个符号生成指纹, 极大减少数据量, 并将识别率提高到 99.98%. 此外, EASI 还证明了使用机器学习算法可以处理综合信号特征. 结果表明, EASI 的计算需求和内存占用分别减少了 142–168 倍. 此外, EASI 在 100 μs 内解决分类问题, 训练时间为 2.61 s.

(2) 基于 CAN 消息的入侵检测

这类工作主要实现在消息层. 其中一种方法是基于网络参数模型的车载网络入侵检测, 通过监测 CAN 消息中的参数, 并分析参数的变化来检测潜在的入侵. 一般分为两步, 首先需要对车载网络中常规的参数进行监测和采集, 可以使用汽车的传感器或 ECU 来采集参数. 然后, 对这些参数进行分析和建模, 以确定正常的参数范围和变化模式. 一旦建立了参数模型, 就可以使用该模型来检测潜在的入侵.

上文提到的针对 ECU 的 Cloaking 攻击[42], 攻击者操纵 ECU 发送消息的传输时间, 通过添加延迟来使 ECU 发送伪造 CAN 消息, 从而绕过一般的以频率作为参数的入侵检测系统. 为了检测这种新型的 ECU 伪装攻击, 作者也提出了相应的入侵检测方案 NTP-based IDS[42], 通过分析了 Cloaking 攻击的执行情况, 然后对其网络消息特征进行系统建模, 从而识别 Cloaking 攻击. 作者比较 Cloaking 攻击在当前最先进的入侵检测系统和 NTP-based IDS 的识别率, 实验结果表明, NTP-based IDS 具有更高的识别率, 并可以有效检测重放和 ECU 伪装攻击.

CANvas[55]提出了一种车载网络映射和逆向工具, 该工具可以识别消息的发送 ECU 和接收 ECU, 进而发现恶意的 ECU. 然而 CAN 总线具有广播特性, 其消息不包含发送者的信息. 因此 CANvas 使用基于成对时钟偏移跟踪算法分析发送方 ECU 的信息, 并使用强制 ECU 隔离技术查找接收方 ECU. 实验结果表明, CANvas 可以精确识别车载网络中的 ECU.

WINDS[56]设计了一种基于 CAN 消息频率的细粒度低误报率的入侵检测系统, 通过连续小波变换来获取频率分量在时间轴上的确切位置, 并利用它来检测 CAN 总线上的异常, 然后根据不同的时间域执行分析, 从而捕获长时间和短时间内的攻击. WINDS 在商用汽车生成的两个数据集上进行了评估, 实验结果表明即使对于短时间内的攻击, 其也能达到较高的检测率.

基于网络参数模型的入侵检测对计算资源要求较低, 适合车载网络中计算能力受限的场景. 然而, 基于网络参数模型的入侵检测对于消息注入攻击效果较差, 因为消息注入攻击的消息可以是正常消息, 其网络参数不会有任何变化. 对于非周期性的操作检测率很低, 例如车载网络中攻击者发送的开门和关门的消息.

另一类方法是基于消息语义的入侵检测. 其根据消息的语义和车辆状态、外部环境等信息判断消息是否为恶意, 这类方案对消息注入攻击更为有效. 丁男等人[57]提出了一种利用车辆速度以及刹车油门等驾驶行为信息的 CAN 总线防注入攻击方法, 该方法需要利用网络消息的语义得到车辆的驾驶信息, 并分析了车载网络数据注入的特征, 建立了基于速度-行为的朴素贝叶斯分类器, 从而实现基于速度与行为的防注入攻击. 在仿真平台上验证了方案的有效性, 然而该方案只能覆盖 CAN 总线注入攻击, 覆盖的攻击较少, 且未在真实车辆平台上测试.

Xue 等人[58]认为之前的工作无法有效防御更高层的消息注入攻击, 因为消息注入攻击可以通过正常的 CAN 帧来实现, 造成车辆不正常的行驶动态. 因此作者提出了 SAID[58], 这是一个基于车辆动态的入侵检测系统, 主要应对两类攻击, 第一类是功能攻击, 如 DoS 攻击, 不需要攻击者了解车载网络协议; 另一类攻击是状态攻击, 目的是造成车辆不正常的状态和运动, 如侧翻或侧滑. SAID 在网络层监测 CAN 帧和诊断信息, 并通过对应的协议规定判断是否受到功能攻击. 同时获得车辆的动态信息 (例如加速、转弯) 以及车内设备状态信息, 同时考虑消息的语义, SAID 检查它们是否会触发车辆异常状态, 包括危险的车辆运动 (例如翻车、打滑) 和异常的驾驶行为 (如在高速公路上开门). SAID 在仿真平台、机器车和真实车辆上分别进行了测试, 证明了该方法的可行性.

ZBCAN[59]提出一个通用的低开销防御系统. 它使用 CAN 消息字段的零字节. ZBCAN 不使用消息字段、身

份验证消息或计算开销较大的操作, 它单独使用消息定时来防止最常见的车载网络攻击, 包括 CAN 消息注入、CAN 总线关闭、ECU 伪装攻击. ZBCAN 由一个可中断传输的可信 officer 节点和几个安装在 ECU 上的软件代理组成. officer 节点和每个代理同意一个秘密的、无限的、动态生成的帧间空间序列 (inter-frame spaces), officer 节点监控每条消息. 除了将 officer 节点连接到 CAN 总线之外, ZBCAN 不需要任何硬件更改, 实现了完全向后兼容. 此外, 由于 ZBCAN 不使用 CAN 消息字段, 因此它可以与其他的解决方案结合使用.

(3) 基于信息熵的入侵检测

基于信息熵的车载网络入侵检测技术通过对网络流量的熵值进行分析, 来判断是否存在入侵行为. 熵是信息理论中的一个重要指标, 它与随机变量的混乱程度和不确定性有关. 正常情况下车载网络流量的统计熵值应该是稳定的, 而任何不正常的行为都可能导致网络熵值的显著变化.

文献 [60,61] 提出了一系列基于熵的车载网络异常检测方法. 这些方法使用信息熵来度量车载网络流量的不确定性, 通过监测熵的变化来检测车载网络潜在的异常. 实验证明其具有较高的准确性和较低的误报率, 其优点在于不需要先验知识或规则库, 且不依赖于车辆网络消息的内容, 因此可以适用于任何类型的流量, 包括专有消息, 因此可以适用于不同类型的车辆和网络拓扑结构.

然而, 上述方法对于目标为消息内容的攻击则不太有效, 例如 ECU 伪造攻击. 另外, 该技术对于低频攻击也不太敏感, 例如 CAN 消息篡改攻击和 CAN 重放攻击, 因为攻击者每秒只注入少量的数据包, 不会显著增加系统的熵值. 为了解决上述问题, Groza 等人[62]提出了一种使用 Bloom 过滤器的入侵检测方案, 旨在提高入侵检测的效率并减少计算开销. 该方案可以通过消息标识符和数据域的内容验证帧的周期性, 从而有效地检测修改的帧. 该方案通过使用 Bloom 过滤器, 即使攻击者在最佳时间重播帧, 也可以识别该重复帧, 进而检测重放攻击. 此外, 该方案还可以用于检测消息篡改攻击和异常流量等.

5.2.2 基于机器学习的入侵检测

机器学习技术在应用于一些黑盒场景时具备一定的优势, 可以在缺乏先验知识的条件下进行高效的异常检测. 车载网络协议一般是闭源的, 因此机器学习的方法较为适合车载网络的入侵检测, 研究者提出了一系列结合机器学习技术的车载网络入侵检测技术. 基于机器学习的入侵检测技术可以分为如下两类.

(1) 基于传统机器学习的入侵检测

传统的机器学习方法, 包括支持向量机、决策树、随机森林和多层感知机, 被广泛应用于车载网络入侵检测. Olufowobi 等人[63]基于变化点检测技术, 将自适应累积和算法应用于检测 CAN 总线信息流, 通过检测 CAN 总线信息流的统计变化识别车载网络入侵. 此外, 该工作也利用 CAN 的响应时间分析提出了一种基于规范的入侵检测系统. 丁男等人[57]利用了朴素贝叶斯网络分类器, 并结合速度和驾驶行为, 构建了基于速度-行为的朴素贝叶斯分类器, 进而有效检测车载网络注入攻击.

H-IDFS[64]使用支持向量机, 并结合基于窗口的入侵检测和过滤方法, 开发了基于直方图的入侵检测和过滤框架. H-IDFS 首先使用 CAN 流量直方图开发入侵检测模型, 以理解不同 CAN 流量类别的独特结构, 针对不同的网络流量特征, 如 CAN 消息频率、消息长度和消息 ID, 设计了特定的直方图以捕捉不同的攻击. 基于两个数据集的实验结果表明, H-IDFS 可以通过一个窗口准确分类, 并且过滤系统能够从异常窗口中过滤出标准数据包, 正确率超过 95%.

Han 等人[65]提出了一种基于间隔和事件触发的车载网络入侵检测机制, 使用机器学习来识别异常并检测车载网络攻击. 首先基于 CAN 消息定义了 4 种攻击场景, 以了解车载网络上下文中的正常和恶意驾驶数据. 然后, 通过考虑固定的时间窗口, 在它们的统计瞬间分析和测量 CAN 身份的事件触发间隔. 在实际驾驶数据上的实验结果表明, 该方法可以快速识别异常, 并在攻击类型识别、时间和异常检测方面获得更好的性能.

(2) 基于深度学习的入侵检测

与传统的机器学习方法相比, 深度学习避免了复杂的特征提取步骤. 基于机器学习的车载网络入侵检测的主要缺点是计算复杂度较高, 此外还需要大量的数据集来训练模型, 而且有价值的数据集很少, 尤其是有攻击或异常流量的数据集. Taylor 等人[66]使用长短期记忆神经网络 (LSTM) 来预测总线上每个发送者的下一个数据字, 并在发现较大偏差时检测并报告异常, 该方案有较大的误报率.

车载网络通过 T-Box 等设备与外部网络连接, 因此容易受到来自蜂窝网络等远程连接的攻击. CAN-ADF[67]提出了一种 CAN 总线消息层攻击的检测框架, 不仅可以配置各种异常和攻击特征, 还可以配置不同的检测方法, 并提供可视化框架来有效地检测这些攻击和异常, 因此 CAN-ADF 可以结合其他入侵检测系统合并工作. 对于检测器, CAN-ADF 采用了基于动态网络流量特征和循环神经网络 (RNN) 的方法. 通过不同车辆采集的 CAN 数据包进行测试, CAN-ADF 的平均准确率为 99.45%.

传感器欺骗攻击会使传感器无法准确感知周围的驾驶环境, 使传感器捕获的数据可能是错误的, 导致意想不到的后果. 为了解决这个问题, Liu 等人[68]提出了一种入侵检测方案 LIFE, 将激光雷达和图像数据融合用于检测感知误差, 通过对象匹配和对应点技术, 并且使用了 RNN 和 LSTM 等深度学习算法来评估激光雷达和相机图像之间的数据一致性, 采用交叉验证的方法来识别异常. LIFE 可以检测各种传感数据异常, 例如激光雷达欺骗、相机致盲、激光雷达误差等, 从而抵御传感器欺骗攻击.

PercepGuard[69]提出了一种针对传感器攻击的入侵检测系统, 利用被检测对象的时空属性 (如固有的轨迹), 交叉检查轨迹和类别预测之间的一致性. 为了提高对抗防御传感器欺骗攻击的鲁棒性, PercepGuard 将上下文数据 (如车辆速度) 进行上下文一致性验证. 使用递归神经网络 (RNN) 作为序列分类器将轨道分类到对象类中. 实验表明, PercepGuard 平均检测到 96% 的攻击.

5.3 车载网络协议安全增强技术

5.3.1 基于轻量级密码学的通信机密性增强

车载网络普遍缺乏机密性设计, CAN 总线广播的特性, 且缺乏加密机制, 导致了 CAN 总线容易遭受 ECU 伪装攻击、消息篡改攻击等. ECU 的算力较低, 且 CAN 实时性要求较高, 因此无法使用传统开销较大的密码学算法. Siddiqui 等人[70]提出了一种基于硬件的方法为 CAN 总线提供加密和身份验证. 该方法采用专用硬件 (ECU server) 管理 CAN 总线上的所有 ECU, 对 ECU 进行认证和密钥分发. 该方法假设 ECU 在制造过程中注册了密钥. 基于硬件的方法开销较小, 但由于需要对硬件进行修改, 因此在当前的车载网络中无法部署.

CAN 总线具有广播特性, 在 CAN 总线上分享密钥具有较大挑战, TRICKS[71]利用 CAN 总线上的时间触发机制来实现隐蔽的密钥共享. TRICKS 提出了一种基于时钟偏移的方法, 使 CAN 总线上的通信可以在特定时间窗口内进行, 并利用这个时间窗口内的微小时钟偏移来传输密钥信息. 通过在 CAN 总线上模拟微小的时钟偏移, 并在特定时间窗口内进行通信, 能够实现密钥共享和交换, 而不容易被外部观察或检测到.

Musuroi 等人[72]基于美国国家标准与技术研究院 (NIST) 系列的椭圆曲线和 Diffie-Hellman 的 FourQ 曲线评估了密钥交换协议, 并提出了一种与业界要求兼容的 CAN 总线上的组密钥交换协议. 在英飞凌和 ARM 核心处理器平台上的实现结果表明, 密钥交换的计算时间比带宽更关键, 该协议在 CAN 上具有较高的性能. 为了实现 CAN 总线长期有效的密钥分发, SENECAN[73]提出了一种结合水印和有线干扰 (wired jamming) 的解决方案. SENECAN 利用故意干扰和扩频水印来实现机密性和完整性, 并防止重放攻击. 与其他工作相比, SENECAN 不需要修改任何 CAN 协议和系统架构, 但是需要一个额外的 CAN 收发器.

当前的车载网络通信机密性增强方案中, 普遍存在计算开销较大, 延迟高的问题, 阻碍了安全方案的实际应用. 为了平衡性能和安全性, S2-CAN[74]提出了一种基于软件的机密性增强解决方案. 该方案由两个迭代交替的阶段组成: 第 1 个阶段为握手阶段, 该阶段内在 ECU 之间安全共享多个参数; 第 2 个阶段为操作阶段, 该阶段内 ECU 发送轻量级加密后的 CAN 帧, 这些 CAN 帧由接收方 ECU 根据握手期间共享的参数解码. S2-CAN 在保证机密性的同时, 实现了低延迟和低通信开销.

5.3.2 基于高效 HMAC 的身份鉴别和消息完整性增强

基于哈希的消息认证码 (HMAC) 是消息认证的重要方法之一, 哈希函数是一类单向函数, 具有不可逆性. 因此, 消息的发送方可以在发送消息时, 通过一个哈希函数计算该消息的消息摘要, 在发送消息的同时附带消息摘要, 消息的接收方可以根据消息摘要校验该消息是否被修改, 如果消息摘要不一致, 则可判断发送方的消息已经被篡改. 在 CAN 标准中, 没有消息认证机制, 但是每个消息发送方都包含一个唯一标识 CAN ID , 可以根据 CAN ID 产生一个消息摘要, 接收消息的 ECU 再根据该消息摘要进行校验, 从而鉴别消息发送者的身份.

Nilsson 等人[75]最早提出了一种使用复合消息认证码的高效延迟数据认证方法. 因为消息身份验证码是在连续消息的组合上计算的, 并与后续消息一起发送, 从而实现身份延迟验证. 该数据认证可用于检测和恢复车载网络中的注入和篡改攻击. 该方案需要对 CAN 数据包的格式进行修改, 而且存在较高的延迟, 但是其提供了一种基础性的思路. CaCAN[76]使用一个额外的 ECU 硬件, 使用硬件修改的中央监控节点在 CAN 总线上执行整个认证, 称为 HMAC CAN 控制器, 用于检测和丢弃未被授权的数据包, 因此可以有效防御伪造和重放攻击. CaCAN 存在与集中式授权相同的问题, 如果监控节点受损或被移除, 整个网络都会受损. 此外, CaCAN 中的消息没有加密, 且使得 CAN 总线的负载大幅增加, 无法在工业场景中应用.

LeiA[77]是一种基于计数器的身份验证协议, 使用扩展的长度为 29 位的 CAN ID 来包括新鲜度值和用于身份验证的通用 ICMAC 算法. MAC 为 8 字节长, 在单独的 CAN 消息中传输, 使总线负载加倍. 作者并未给出具体的延迟数据. vatiCAN[78]提供向后兼容的发送者认证和消息认证, 通过预安装的密钥计算的 HMAC 对关键 CAN 消息进行检验, 从而防止重放攻击. HMAC 在一个单独的 CAN 消息中发送, 具有不同的 CAN ID. 由于 HMAC 需要占用额外的 CAN 消息发送, 因此其开销相对较大.

针对无钥匙进入系统攻击, hold the door (HODOR)[79]设计了一种针对无钥匙进入系统攻击的防御机制, 利用超高频射频信号的认证机制, 无须任何修改, 即可在现有无钥匙进入系统的认证过程中实现和部署. 实验结果表明在模拟攻击的条件下, 假阳性率为 0.27%, 假阴性率为 0, 取得了较好的效果.

TACAN[80]通过基于到达时间间隔、基于最小有效位和混合通道这 3 种不同的隐蔽通道, 在 CAN 总线上提供 ECU 之间的安全认证, 从而抵御 ECU 伪装和 CAN 消息注入攻击. 此外, TACAN 可以在不修改 CAN 协议和增加额外通信开销的情况下部署. 在雪佛兰 Camaro 2016 和丰田 Camry 2010 数据集上的实验结果表明, TACAN 有效地检测了 CAN 总线攻击, 并在评估比特错误和吞吐量性能参数时取得了更好的结果.

利用 CAN 帧中的定时参数可以创建一个满足认证的隐蔽通道, 避免了在资源受限的车载网络上进行数据加密. 但是这种方案其安全级别有限, 攻击者仍然可以在 CAN 总线上发动不同的攻击. CANTO[81]提出了一种改进的协议, 通过二进制对称 (binary symmetric)、贪心算法和最大公约数等算法循环调度 CAN 帧, 为 CAN 流量建立隐蔽通道, 实现了更高的安全等级, 该协议可以在隐蔽信道上实现更高的数据速率.

5.4 小 结

本节首先总结了车载网络防御技术的目标, 结合车载网络相关安全缺陷和攻击, 对车载网络入侵检测技术进行分类, 并且分析其面临的挑战. 然后分类介绍车载网络入侵检测技术, 最后介绍了车载网络协议安全增强技术. 本节并对不同类别的入侵检测技术进行总结和比较. 根据车载网络入侵检测工作的方案类型、代表工作、防御的攻击、主要优点和局限性等几个维度, 将车载网络入侵检测工作总结为表 8.

表 8 车载网络入侵检测工作总结

分类	具体方案	代表工作	防御的攻击	主要优点	局限性
基于特征工程	基于电压特征	Viden[48]	ECU伪装 消息篡改	识别被攻击的ECU	无法检测不规则消息的注入
		Scission[52]	ECU伪装 总线关闭	在未对系统进行修改的条件下检测入侵	ECU伪装攻击检测率较低
	基于时钟和其他特征	CIDS[51]	ECU伪装 消息篡改	找到反复出现的车内指令消息的时间间隔	无法检测不规则消息的注入
		SIMPLE[53]	ECU伪装	较低计算和数据采集成本的单帧攻击检测	未分析其他重要的安全威胁
	基于网络参数模型	NTP-based IDS[42]	重放 ECU伪装	可以基于时间的攻击, 例如Cloaking攻击	引入的噪声较大
		WINDS[56]	重放 ECU伪装 总线关闭	可以识别CAN流量中的行为变化位置	只有当攻击使用消息频率时有效
	基于信息熵	Groza等人[62]	重放 消息篡改	可以有效抵御重放和CAN总线篡改攻击, 且不需要先验知识	误报率相对较高

表 8 车载网络入侵检测工作总结 (续)

分类	具体方案	代表工作	防御的攻击	主要优点	局限性
基于机器学习	基于网络消息语义	Ding等人[57]	消息注入	较早地考虑了消息语义和车辆状态	涵盖攻击较少, 实验不够充分
		SAID[58]	消息注入 总线关闭等	考虑了上层消息语义, 例如车辆状态信息	无法检测策略和规则未涵盖的未知异常
	传统机器学习	H-IDFS[64]	重放 ECU伪装 消息注入	将CAN数据包组装成窗口对流量进行分类, 适配的攻击较多, 实现了过滤机制	效率有待提高, 计算复杂度较高, 不够轻量
		Han等人[65]	重放 消息篡改 总线关闭	通过周期性事件触发间隔检测和识别异常	入侵检测的响应时间较长
	深度学习	CAN-ADF[67]	重放 消息注入 总线关闭	可以配置异常和攻击的特征, 并且与其他入侵检测系统合并工作	攻击特征配置较为复杂
		LIFE[68]	传感器欺骗 ECU伪装 消息注入	可以有效识别和抵御传感器欺骗攻击	效果还有待优化

由表 8 的总结, 当前车载网络入侵检测研究还存在一些局限性, 主要有如下几点.

(1) 现有大多数入侵检测系统仅使用一种类型的车辆数据 (如传感器、CAN 帧或 OBD-II 消息) 来检测异常, 没有充分运用各类信息.

(2) 现有工作大多基于单一的技术或算法, 没有很好地规避单一方法的缺点, 多个方法混合使用, 扬长避短, 可能会达到更好的效果.

(3) 现有工作中大多数没有充分利用汽车的场景信息, 汽车是一个与驾驶者和环境不断交互的系统, 充分利用这些不断变化的场景信息对车载网络的入侵检测工作十分重要.

(4) 受限于时间和成本, 大多数现有工作的实验平台受限, 测试不够充分.

表 9 从目标、方案类型、具体算法、缓解的攻击等方面比较和总结了车载网络协议安全增强工作.

表 9 车载网络协议安全增强工作总结

目标	方案类型	代表工作	主要算法	缓解的攻击	局限性
通信过程中消息的机密性	轻量级密码学算法	TRICKS[71]	基于Diffie-Hellman的加密密钥交换协议	重放 消息注入 节点伪装	多ECU节点的情况下计算开销较高
		SENECAN[73]	数字水印、有线干扰 (wired jamming)	重放 消息注入 消息篡改 ECU伪装	制造商可能需要刷新所有节点的密钥, 以增加其长度或更改加密原语
		S2-CAN[75]	握手阶段在ECU之间安全共享参数; 操作阶段加密发送	重放 消息注入 ECU伪装	缺乏密钥管理机制, 且需要使用负载中的2字节
通信双方的身份鉴别和消息的完整性	高效的哈希消息验证码 (HMAC)	CaCAN[76]	HMAC对称密钥计数器	重放 消息注入 ECU伪装	需要额外的ECU硬件
		LeiA[77]	128位密钥MAC计数器	重放 ECU伪装	需要在CAN帧头部添加16位的计数器
		TACAN[80]	二进制对称信道、贪心算法和最大公约数	重放 CAN总线关闭	需要额外的工具, 覆盖的攻击较少

由表 9 的总结, 目前两个问题阻碍了车载网络协议安全增强技术的应用.

(1) 一些方案与现有 CAN 协议不兼容. 车载网络协议安全增强方案不兼容导致厂商需要重新设计 ECU 的固件, 对于供应商来说, 推出新固件的成本是很高的. 此外, 这样的改动需要不同供应商之间进行大量的协调工作, 以

确保不同 ECU 之间可以互操作性, 对于厂商是很难接受的.

(2) 延迟和开销较大. 虽然这些延迟可能对大多数通用操作系统是可以接受的, 但是车辆操作有严格的实时约束, 较高的延迟和开销很难被接受.

6 车载网络攻防技术研究展望

车载网络在设计之初, 受制于技术、成本等限制, 没有全面考虑未来智能网联汽车出现的攻击问题. 例如, 最具代表性的车载网络 CAN, 虽然其布线成本低, 信号传输速率较高且有一定容错机制, 但其广播、无身份鉴别等特点, 给攻击者暴露了较大的攻击面. 车载网络的攻防与传统网络攻防研究有很多共性问题, 同时也存在很大区别. 本文总结和归纳了车载网络协议逆向, 车载网络攻击和防御技术的研究现状. 如何更好地了解车载网络的漏洞和可能存在的攻击, 并设计更为高效、准确、鲁棒的车载网络防御技术需要进一步的研究. 我们总结了车载网络攻击和防御技术的开放挑战和关键研究问题.

根据本文的总结, 车载网络的攻防研究面临如下几个挑战.

(1) 通用性和性能更强的车载网络协议逆向: 当前的车载网络逆向技术存在通用性不足的问题, 其适配的车型有限. 虽然一些工作在实验中可以取得较好的性能, 但是其性能表现在不同车型上差别较大. 此外, 一些令牌解析的工作是半自动化的, 需要专业人员介入, 其自动化程度有待进一步提高.

(2) 实时性更强的安全系统: 汽车系统需要及时做出相应决策, 因此 ECU 的实时性要求较高. 一些安全攸关的系统往往采用实时操作系统, 但对于实时操作系统的攻防研究还相对较少. 此外, 一些 ECU 算力有限, 因此无法部署时延和开销较高的安全系统.

(3) 向后兼容性的车载网络协议安全增强方案: 很多车载网络协议安全增强工作需要对 CAN 等已经广泛部署的车载网络协议进行更改, 这可能造成安全增强后的车载网络协议与旧款车辆的相关协议不兼容, 不满足兼容性是很多安全方案在工业界无法大范围部署的重要原因之一.

(4) 统一的车载网络安全测试标准和实验平台: 虽然智能网联汽车在发售前都经过严格的网络安全测试, 但目前工业界还缺乏统一的车载网络检测标准. 此外, 当前的研究工作在实验平台、测试数据集和评估指标方面差别较大, 目前学术界缺乏统一的车载网络安全实验标准和平台.

为应对上述研究挑战, 我们认为未来的研究工作应该重点关注以下方面.

(1) 多种方法融合并结合机器学习的车载网络逆向技术

为了提高逆向工程的自动化、鲁棒性和性能, 可以将多种方法结合, 充分发挥不同方法的优势, 并利用机器学习和大模型技术在黑盒场景的优势, 进一步提高方案的通用性. 例如, 基于 PID 注入的令牌解析可用于初步逆向可公开获得的车辆功能, 然后采用基于语义分类和配套应用的方法对第 1 步中遗漏的 CAN 信号进行逆向. 最后结合机器学习尤其是新兴的大模型技术提高方案的通用性, 增强对不同车型的适配性.

(2) 针对车载网络 ECU 实时操作系统的攻击面分析和防御

智能网联汽车的一些 ECU 采用实时操作系统, 例如 QNX[82]、VxWorks[83], 当前缺乏对车载实时操作系统攻击面的全面分析, 对于车载实时操作系统中恶意软件的研究也相对较少. 一些研究表明实时操作系统存在一些独特的攻击, 例如 Cotroneo 等人[83]对 VxWorks 基于时间的隐蔽通道进行了分析. 对于实时操作系统的防御, 纯软件方案可能存在开销和时延较高的问题, 软硬件结合的安全方案是一种思路, 例如 RT-TEE[84]提出了一种针对物理信息系统的可行执行环境, 可以保护关键系统资源不被恶意软件破坏. 并在无人机上进行了测试, 然而 RT-TEE 是否适合智能网联汽车仍存在疑问. 工业界和学术界都迫切需要轻量、高效、鲁棒的车载实时操作系统防御方案.

(3) 多种方法结合的高实时性车载网络入侵检测方案

通过本文的总结和分析来看, 现有大多数入侵检测系统仅使用某一种类型的车辆数据来检测异常, 没有充分结合各类信息如 CAN 数据流、传感器数据和车辆动态信息等. 目前的研究工作大多基于单一的技术或算法, 没有很好地规避单一方法的缺点, 例如基于深度学习的入侵检测可以结合 CAN 消息的语义, 从而提高模型的可解释性

和鲁棒性. 此外, 一些方法的实时性较差, 无法部署于实时性要求较高的 ECU 上. 未来的车载网络入侵检测可以将多个方法混合使用, 提高准确率的同时降低入侵检测系统的性能开销.

(4) 向后兼容的轻量级车载网络协议安全设计

CAN 协议没有内置的身份验证和加密机制. 因此, 研究者一直致力于利用密码学方法弥补该缺陷. 基于硬件的加密方案安全级别较高, 满足智能网联汽车的实时性需求. 然而其成本较高和兼容性不足是其主要问题. 基于软件的加密方法不需要更改 CAN 总线体系结构, 然而其高计算负载和通信开销很难被工业界接受. 因此, 设计轻量级的车载网络安全协议是后续的重要研究方向.

(5) 统一的智能网联汽车的车载网络测试标准和实验平台

虽然目前国家有车载网络安全的指导方针, 2022 年 2 月工业和信息化部发布了《车联网网络安全和数据安全标准体系建设指南》[85]对车载网络安全提供了指导, 其中已发布了《汽车网关信息安全技术要求及试验方法》, 但是大多数车载网络安全标准仍处于规划状态, 例如《车载总线系统网络安全技术要求》仍处于待制定状态, 相关部门和工业界亟需推出更为完善的智能网联汽车的车载网络安全标准. 此外, 学术界也需要推出统一的车载网络安全测试平台, 从而降低实验成本并统一测试指标. 例如, 车载网络协议逆向相关研究工作中, 不同工作采用的实验平台和测试数据集大多不同, 目前缺乏被广泛接受且公开的车载网络逆向测试标准数据集.

7 结束语

本文对车载网络攻击和防御技术进行了分析和总结, 首先介绍了智能网联汽车的整体结构和 CAN 协议, 引出了车载网络安全的重要性. 其次归纳了车载网络协议的逆向的研究工作, 讨论了车载网络攻击向量, 从不同维度对车载网络攻击进行分类. 再次总结了车载网络防御的相关研究工作; 最后, 展望了车载网络安全研究的未来研究方向. 期望通过我们的工作, 能够给研究者提供有益的借鉴与参考, 为智能网联汽车安全研究做出贡献.

致谢 在此, 我们向对本文的工作给予支持和宝贵建议的评审老师和同行表示衷心的感谢!

References:

[1] Valasek C, Miller C. Remote exploitation of an unaltered passenger vehicle. Black Hat USA, 2015, 2015(S91): 1–91.

[2] Nie S, Liu L, Du YF. Free-fall: Hacking Tesla from wireless to can bus. Black Hat USA, 2017, 25: 1–16.

[3] Nie S, Liu L, Du YF, Zhang WK. Over-the-air: How we remotely compromised the gateway, BCM, and autopilot ECUs of Tesla cars. Black Hat USA, 2018. 1–19.

[4] HPL S C. Introduction to the controller area network (CAN). Application Report SLOA101, 2002. 1–17.

[5] Sagstetter F, Lukasiewycz M, Steinhorst S, Wolf M, Bouard A, Harris WR, Jha S, Peyrin T, Poschmann A, Chakraborty S. Security challenges in automotive hardware/software architecture design. In: Proc. of the 2013 Design, Automation & Test in Europe Conf. & Exhibition (DATE). Grenoble: IEEE, 2013. 458–463. [doi: 10.7873/DATE.2013.102]

[6] Koscher K, Czeskis A, Roesner F, Patel S, Kohno T, Checkoway S, Mccoy D, Kantor B, Anderson D, Shacham H, Savage S. Experimental security analysis of a modern automobile. In Proc. of the 2010 IEEE Symp. on Security and Privacy. Oakland: IEEE, 2010. 447–462. [doi: 10.1109/SP.2010.34]

[7] Checkoway S, McCoy D, Kantor B, Anderson D, Shacham H, Savage S, Koscher K, Czeskis A, Roesner F, Kohno T. Comprehensive experimental analyses of automotive attack surfaces. In: Proc. of the 20th USENIX Security Symp. San Francisco: USENIX Association, 2011. 447–462.

[8] Kim K, Kim JS, Jeong S, Park JH, Kim HK. Cybersecurity for autonomous vehicles: Review of attacks and defense. Computers & Security, 2021, 103: 102150. [doi: 10.1016/j.cose.2020.102150]

[9] Aliwa E, Rana O, Perera C, Burnap P. Cyberattacks and countermeasures for in-vehicle networks. ACM Computing Surveys, 2021, 54(1): 21. [doi: 10.1145/3431233]

[10] Buscemi A, Turcanu I, Castignani G, Panchenko A, Engel T, Shin KG. A survey on controller area network reverse engineering. IEEE Communications Surveys & Tutorials, 2023, 25(3): 1445–1481. [doi: 10.1109/COMST.2023.3264928]

[11] Evenchick E. An introduction to the CANard toolkit. In: Proc. of the 2015 Blackhat Conf. 2015.

[12] Currie R. Hacking the CAN bus: Basic manipulation of a modern automobile through can bus reverse engineering. SANS Institute, 2017.

[13] Verma M, Bridges R, Hollifield S. ACTT: Automotive CAN tokenization and translation. In: Proc. of the 2018 Int'l Conf. on Computational Science and Computational Intelligence (CSCI). Las Vegas: IEEE, 2018. 278–283. [doi: 10.1109/csci46756.2018.00061]

[14] Verma ME, Bridges RA, Sosnowski JJ, Hollifield SC, Iannacone MD. CAN-D: A modular four-step pipeline for comprehensively decoding controller area network data. IEEE Trans. on Vehicular Technology, 2021, 70(10): 9685–9700. [doi: 10.1109/TVT.2021.3092354]

[15] Pesé MD, Stacer T, Campos CA, Newberry E, Chen DY, Shin KG. LibreCAN: Automated CAN message translator. In: Proc. of the 2019 ACM SIGSAC Conf. on Computer and Communications Security. London: ACM, 2019. 2283–2300. [doi: 10.1145/3319535.3363190]

[16] Nolan BC, Graham S, Mullins B, Kabban CS. Unsupervised time series extraction from controller area network payloads. In: Proc. of the 88th IEEE Vehicular Technology Conf. (VTC-Fall). Chicago: IEEE, 2018. 1–5. [doi: 10.1109/VTCFall.2018.8690615]

[17] Marchetti M, Stabili D. READ: Reverse engineering of automotive data frames. IEEE Trans. on Information Forensics and Security, 2019, 14(4): 1083–1097. [doi: 10.1109/TIFS.2018.2870826]

[18] Wen HH, Zhao QC, Chen QA, Lin ZQ. Automated cross-platform reverse engineering of CAN bus commands from mobile APPs. In: Proc. of the 2020 Network and Distributed System Security Symp. San Diego: ISOC, 2020. 1–17. [doi: 10.14722/ndss.2020.24231]

[19] Buscemi A, Turcanu I, Castignani G, Crunelle R, Engel T. Poster: A methodology for semi-automated CAN bus reverse engineering. In: Proc. of the 2021 IEEE Vehicular Networking Conf. (VNC). Ulm: IEEE, 2021. 125–126. [doi: 10.1109/VNC52810.2021.9644673]

[20] Frassinelli D, Park S, Nürnberger S. I know where you parked last summer: Automated reverse engineering and privacy analysis of modern cars. In Proc. of the 2020 IEEE Symp. on Security and Privacy. San Francisco: IEEE, 2020. 1401–1415. [doi: 10.1109/SP40000.2020.00081]

[21] Yu L, Liu YY, Jing PF, Luo XP, Xue L, Zhao KF, Zhou YJ, Wang T, Gu GF, Nie S, Wu S. Towards automatically reverse engineering vehicle diagnostic protocols. In: Proc. of the 31st USENIX Security Symp. Boston: USENIX Association, 2022. 1939–1956.

[22] Hu X, Liang L, Li SC, Deng L, Zuo PF, Ji Y, Xie XF, Ding YF, Liu C, Sherwood T, Xie Y. DeepSniffer: A DNN model extraction framework based on learning architectural hints. In: Proc. of the 25th Int'l Conf. on Architectural Support for Programming Languages and Operating Systems. Lausanne: ACM, 2020. 385–399. [doi: 10.1145/3373376.3378460]

[23] Wen HH, Chen QA, Lin ZQ. Plug-N-Pwned: Comprehensive vulnerability analysis of OBD-II dongles as a new Over-the-air attack surface in automotive IoT. In: Proc. of the 29th USENIX Security Symp. USENIX Association, 2020. 949–965.

[24] Wang W, Yao Y, Liu X, Li X, Hao P, Zhu T. I can see the light: Attacks on autonomous vehicles using invisible lights. In: Proc. of the 2021 ACM SIGSAC Conf. on Computer and Communications Security. Virtual Event: ACM, 2021. 1930–1944. [doi: 10.1145/3460120.3484766]

[25] Jing PF, Tang QY, Du YF, Xue L, Luo XP, Wang T, Nie S, Wu S. Too good to be safe: Tricking lane detection in autonomous driving with crafted perturbations. In: Proc. of the 30th USENIX Security Symp. Virtual: USENIX Association, 2021. 3237–3254.

[26] Zhao Y, Zhu H, Liang RG, Shen QT, Zhang SZ, Chen K. Seeing isn't believing: Towards more robust adversarial attack against real world object detectors. In: Proc. of the 2019 ACM SIGSAC Conf. on Computer and Communications Security. London: ACM, 2019. 1989–2004. [doi: 10.1145/3319535.3354259]

[27] Muller R, Man YM, Celik ZB, Li M, Gerdes R. Physical hijacking attacks against object trackers. In: Proc. of the 2022 ACM SIGSAC Conf. on Computer and Communications Security. Los Angeles: ACM, 2022. 2309–2322. [doi: 10.1145/3548606.3559390]

[28] Zhou C, Yan Q, Shi Y, Sun LC. DoubleStar: Long-range attack towards depth estimation based obstacle avoidance in autonomous systems. In: Proc. of the 31st USENIX Security Symp. Boston: USENIX Association, 2022. 1885–1902.

[29] Cao YL, Xiao CW, Cyr B, Zhou YM, Park W, Rampazzi S, Chen QA, Fu K, Mao ZM. Adversarial sensor attack on LiDAR-based perception in autonomous driving. In: Proc. of the 2019 ACM SIGSAC Conf. on Computer and Communications Security. London: ACM, 2019. 2267–2281. [doi: 10.1145/3319535.3339815]

[30] Zhu Y, Miao CL, Zheng TH, Hajiaghajani F, Su L, Qiao CM. Can we use arbitrary objects to attack LiDAR perception in autonomous driving? In: Proc. of the 2021 ACM SIGSAC Conf. on Computer and Communications Security. Virtual Event: ACM, 2021. 1945–1960. [doi: 10.1145/3460120.3485377]

[31] Sun JC, Cao YL, Chen QA, Mao ZM. Towards robust LiDAR-based perception in autonomous driving: General black-box adversarial sensor attack and countermeasures. In: Proc. of the 29th USENIX Security Symp. USENIX Association, 2020. 877–894.

[32] Cao YL, Bhupathiraju SH, Naghavi P, Sugawara T, Mao ZM, Rampazzi S. You can't see me: Physical removal attacks on LiDAR-based autonomous vehicles driving frameworks. In: Proc. of the 32nd USENIX Security Symp. Anaheim: USENIX Association, 2023.

[33] Jin ZZ, Ji XY, Cheng YS, Yang B, Yan C, Xu WY. PLA-LiDAR: Physical laser attacks against LiDAR-based 3D object detection in autonomous vehicle. In: Proc. of the 2023 IEEE Symp. on Security and Privacy. San Francisco: IEEE, 2023. 1822–1839. [doi: 10.1109/

SP46215.2023.10179458]

[34] Sun Z, Balakrishnan S, Su L, Bhuyan A, Wang P, Qiao CM. Who is in control? Practical physical layer attack and defense for mmWave-based sensing in autonomous vehicles. IEEE Trans. on Information Forensics and Security, 2021, 16: 3199–3214. [doi: 10.1109/TIFS.2021.3076287]

[35] Vennam RR, Jain IK, Bansal K, Orozco J, Shukla P, Ranganathan A, Bharadia D. mmSpoof: Resilient spoofing of automotive millimeter-wave radars using reflect array. In: Proc. of the 2023 IEEE Symp. on Security and Privacy. San Francisco: IEEE, 2023. 1807–1821. [doi: 10.1109/SP46215.2023.10179371]

[36] Antonioli D, Payer M. On the insecurity of vehicles against protocol-level bluetooth threats. In: Proc. of the 2022 IEEE Security and Privacy Workshops. San Francisco: IEEE, 2022. 353–362. [doi: 10.1109/SPW54247.2022.9833886]

[37] Antonioli D, Tippenhauer NO, Rasmussen K. BIAS: Bluetooth impersonation attacks. In: Proc. of the 2020 IEEE Symp. on Security and Privacy. San Francisco: IEEE, 2020. 549–562. [doi: 10.1109/SP40000.2020.00093]

[38] Antonioli D, Tippenhauer NO, Rasmussen KB. The KNOB is Broken: Exploiting low entropy in the encryption key negotiation of bluetooth BR/EDR. In: Proc. of the 28th USENIX Security Symp. Santa Clara: USENIX Association, 2019. 1047–1061.

[39] Garcia FD, Oswald D, Kasper T, Pavlidès P. Lock it and still lose it—On the (in)security of automotive remote keyless entry systems. In: Proc. of the 25th USENIX Security Symp. Austin: USENIX Association, 2016. 929–944.

[40] Xie XY, Jiang K, Dai R, Lu J, Wang LH, Li Q, Yu J. Access your tesla without your awareness: Compromising keyless entry system of model 3. In: Proc. of the 2023 Network and Distributed System Security Symp. San Diego: ISOC, 2023. 1–18. [doi: 10.14722/ndss.2023.24082]

[41] Tesla cars can be stolen by hacking the App. 2016. https://electrek.co/2016/11/23/tesla-hacker-steal-car/

[42] Ying XH, Sagong SU, Clark A, Bushnell L, Poovendran R. Shape of the cloak: Formal analysis of clock skew-based intrusion detection system in controller area networks. IEEE Trans. on Information Forensics and Security, 2019, 14(9): 2300–2314. [doi: 10.1109/TIFS.2019.2895957]

[43] Fröschle S, Stühring A. Analyzing the capabilities of the CAN attacker. In: Proc. of the 22nd European Symp. on Research in Computer Security. Oslo: Springer, 2017. 464–482. [doi: 10.1007/978-3-319-66402-6_27]

[44] Cho KT, Shin KG. Error handling of in-vehicle networks makes them vulnerable. In: Proc. of the 2016 ACM SIGSAC Conf. on Computer and Communications Security. Vienna: ACM, 2016. 1044–1055. [doi: 10.1145/2976749.2978302]

[45] Serag K, Bhatia R, Kumar V, Celik ZB, Xu DY. Exposing new vulnerabilities of error handling mechanism in CAN. In: Proc. of the 30th USENIX Security Symp. USENIX Association, 2021. 4241–4258.

[46] Kulandaivel S, Jain S, Guajardo J, Sekar V. CANNON: Reliable and stealthy remote shutdown attacks via unaltered automotive microcontrollers. In: Proc. of the 2021 IEEE Symp. on Security and Privacy. San Francisco: IEEE, 2021. 195–210. [doi: 10.1109/SP40001.2021.00122]

[47] de Faveri Tron A, Longari S, Carminati M, Polino M, Zanero S. CANflict: Exploiting peripheral conflicts for data-link layer attacks on automotive networks. In: Proc. of the 2022 ACM SIGSAC Conf. on Computer and Communications Security. Los Angeles: ACM, 2022. 711–723. [doi: 10.1145/3548606.3560618]

[48] Cho KT, Shin KG. Viden: Attacker identification on in-vehicle networks. In: Proc. of the 2017 ACM SIGSAC Conf. on Computer and Communications Security. Dallas: ACM, 2017. 1109–1123. [doi: 10.1145/3133956.3134001]

[49] Choi W, Joo K, Jo HJ, Park MC, Lee DH. VoltageIDS: Low-level communication characteristics for automotive intrusion detection system. IEEE Trans. on Information Forensics and Security, 2018, 13(8): 2114–2129. [doi: 10.1109/TIFS.2018.2812149]

[50] Bhatia R, Kumar V, Serag K, Celik ZB, Payer M, Xu DY. Evading voltage-based intrusion detection on automotive CAN. In: Proc. of the 2021 Network and Distributed System Security Symp. Virtual: ISOC, 2021. 1–17. [doi: 10.14722/ndss.2021.23013]

[51] Cho KT, Shin KG. Fingerprinting electronic control units for vehicle intrusion detection. In: Proc. of the 25th USENIX Security Symp. Austin: USENIX Association, 2016. 911–927.

[52] Kneib M, Huth C. Scission: Signal characteristic-based sender identification and intrusion detection in automotive networks. In: Proc. of the 2018 ACM SIGSAC Conf. on Computer and Communications Security. Toronto: ACM, 2018. 787–800. [doi: 10.1145/3243734.3243751]

[53] Foruhandeh M, Man YM, Gerdes R, Li M, Chantem T. SIMPLE: Single-frame based physical layer identification for intrusion detection and prevention on in-vehicle networks. In: Proc. of the 35th Annual Computer Security Applications Conf. San Juan: ACM, 2019. 229–244. [doi: 10.1145/3359789.3359834]

[54] Kneib M, Schell O, Huth C. EASI: Edge-based sender identification on resource-constrained platforms for automotive networks. In: Proc. of the 2020 Network and Distributed System Security Symp. San Diego: ISOC, 2020. 1–16. [doi: 10.14722/ndss.2020.24025]

[55] Kulandaivel S, Goyal T, Agrawal AK, Sekar V. CANvas: Fast and inexpensive automotive network mapping. In: Proc. of the 28th USENIX Security Symp. Santa Clara: USENIX Association, 2019. 389–405.

[56] Bozdal M, Samie M, Jennions IK. WINDS: A wavelet-based intrusion detection system for controller area network (CAN). IEEE Access, 2021, 9: 58621–58633. [doi: 10.1109/ACCESS.2021.3073057]

[57] Ding N, Liang WB, Xu L, Song CX, Tan GZ. Analysis of malicious injection attack on CAN data in in-vehicle network based on driving behavior and velocity. Ruan Jian Xue Bao/Journal of Software, 2017, 28: 1–10 (in Chinese with English abstract). https://www.jos.org.cn/jos/article/abstract/17001?st=search

[58] Xue L, Liu YY, Li TQ, Zhao KF, Li JF, Yu L, Luo XP, Zhou YJ, Gu GF. SAID: State-aware defense against injection attacks on in-vehicle network. In: Proc. of the 31st USENIX Security Symp. Boston: USENIX Association, 2022. 1921–1938.

[59] Serag K, Bhatia R, Faqih A, Ozmen MO, Kumar V, Celik ZB, Xu DY. ZBCAN: A zero-byte CAN defense system. In: Proc. of the 32nd USENIX Security Symp. Anaheim: USENIX Association, 2023. 6893–6910.

[60] Müter M, Asaj N. Entropy-based anomaly detection for in-vehicle networks. In: Proc. of the 2011 IEEE Intelligent Vehicles Symp. Baden-Baden: IEEE, 2011. 1110–1115. [doi: 10.1109/IVS.2011.5940552]

[61] Wang Q, Lu ZJ, Qu G. An entropy analysis based intrusion detection system for controller area network in vehicles. In: Proc. of the 31st IEEE Int'l System-on-chip Conf. Arlington: IEEE, 2018. 90–95. [doi: 10.1109/SOCC.2018.8618564]

[62] Groza B, Murvay PS. Efficient intrusion detection with bloom filtering in controller area networks. IEEE Trans. on Information Forensics and Security, 2019, 14(4): 1037–1051. [doi: 10.1109/TIFS.2018.2869351]

[63] Olufowobi H, Ezeobi U, Muhati E, Robinson G, Young C, Zambreno J, Bloom G. Anomaly detection approach using adaptive cumulative sum algorithm for controller area network. In: Proc. of the 2019 ACM Workshop on Automotive Cybersecurity. Richardson: ACM, 2019. 25–30. [doi: 10.1145/3309171.3309178]

[64] Derhab A, Belaoued M, Mohiuddin I, Kurniawan F, Khan MK. Histogram-based intrusion detection and filtering framework for secure and safe in-vehicle networks. IEEE Trans. on Intelligent Transportation Systems, 2022, 23(3): 2366–2379. [doi: 10.1109/TITS.2021.3088998]

[65] Han ML, Kwak BI, Kim HK. Event-triggered interval-based anomaly detection and attack identification methods for an in-vehicle network. IEEE Trans. on Information Forensics and Security, 2021, 16: 2941–2956. [doi: 10.1109/TIFS.2021.3069171]

[66] Taylor A, Leblanc S, Japkowicz N. Anomaly detection in automobile control network data with long short-term memory networks. In: Proc. of the 2016 IEEE Int'l Conf. on Data Science and Advanced Analytics. Montreal: IEEE, 2016. 130–139. [doi: 10.1109/DSAA.2016.20]

[67] Tariq S, Lee S, Kim HK, Woo SS. CAN-ADF: The controller area network attack detection framework. Computers & Security, 2020, 94: 101857. [doi: 10.1016/j.cose.2020.101857]

[68] Liu JS, Park JM. "Seeing is not always believing": Detecting perception error attacks against autonomous vehicles. IEEE Trans. on Dependable and Secure Computing, 2021, 18(5): 2209–2223. [doi: 10.1109/TDSC.2021.3078111]

[69] Man YM, Muller R, Li M, Celik ZB, Gerdes R. That person moves like a car: Misclassification attack detection for autonomous systems using spatiotemporal consistency. In: Proc. of the 32nd USENIX Security Symp. Anaheim: USENIX Association, 2023. 6929–6946.

[70] Siddiqui AS, Gui YT, Plusquellic J, Saqib F. Secure communication over CANBus. In: Proc. of the 60th Int'l Midwest Symp. on Circuits and Systems. IEEE, 2017. 1264–1267. [doi: 10.1109/MWSCAS.2017.8053160]

[71] Groza B, Popa L, Murvay PS. TRICKS —Time triggered covert key sharing for controller area networks. IEEE Access, 2019, 7: 104294–104307. [doi: 10.1109/ACCESS.2019.2931247]

[72] Musuroi A, Groza B, Popa L, Murvay PS. Fast and efficient group key exchange in controller area networks (CAN). IEEE Trans. on Vehicular Technology, 2021, 70(9): 9385–9399. [doi: 10.1109/TVT.2021.3098546]

[73] Soderi S, Colelli R, Turrin F, Pascucci F, Conti M. SENECAN: Secure key distribution over CAN through Watermarking and Jamming. IEEE Trans. on Dependable and Secure Computing, 2023, 20(3): 2274–2288. [doi: 10.1109/TDSC.2022.3179562]

[74] Pesé MD, Schauer JW, Li JH, Shin KG. S2-CAN: Sufficiently secure controller area network. In: Proc. of the 37th Annual Computer Security Applications Conf. Virtual Event: ACM, 2021. 425–438. [doi: 10.1145/3485832.3485883]

[75] Nilsson DK, Larson UE, Jonsson E. Efficient in-vehicle delayed data authentication based on compound message authentication codes. In: Proc. of the 68th Vehicular Technology Conf. Calgary: IEEE, 2008. 1–5. [doi: 10.1109/VETECF.2008.259]

[76] Kurachi R, Matsubara Y, Takada H, Ueda H, Horihata S. CaCAN-centralized authentication system in CAN (controller area network). In: Proc. of the 14th Int'l Conf. on Embedded Security in Cars, 2014. 10.

[77] Radu AI, Garcia FD. LeiA: A lightweight authentication protocol for CAN. In: Proc. of the 21st European Symp. on Research in Computer Security. Heraklion: Springer, 2016. 283–300. [doi: 10.1007/978-3-319-45741-3_15]

[78] Nürnberger S, Rossow C. –vatiCAN–vetted, authenticated CAN bus. In: Proc. of the 18th Int'l Conf. on Cryptographic Hardware and Embedded Systems. Santa Barbara: Springer, 2016. 106–124. [doi: 10.1007/978-3-662-53140-2_6]

[79] Joo K, Choi W, Lee DH. Hold the door! Fingerprinting your car key to prevent keyless entry car theft. In: Proc. of the 2020 Network and Distributed System Security Symp. San Diego: ISOC, 2020. 1–18.

[80] Ying XH, Bernieri G, Conti M, Bushnell L, Poovendran R. Covert channel-based transmitter authentication in controller area networks. IEEE Trans. on Dependable and Secure Computing, 2022, 19(4): 2665–2679. [doi: 10.1109/TDSC.2021.3068213]

[81] Groza B, Popa L, Murvay PS. CANTO—Covert authentication with timing channels over optimized traffic flows for CAN. IEEE Trans. on Information Forensics and Security, 2021, 16: 601–616. [doi: 10.1109/TIFS.2020.3017892]

[82] Dasari D, Hamann A, Broede H, Pressler M, Ziegenbein D. Brief industry paper: Dissecting the QNX adaptive partitioning scheduler. In: Proc. of the 27th IEEE Realme and Embedded Technology and Applications Symp. Nashville: IEEE, 2021. 477–480. [doi: 10.1109/RTAS52030.2021.00056]

[83] Cotroneo D, De Simone L, Natella R. Timing covert channel analysis of the VxWorks MILS embedded hypervisor under the common criteria security certification. Computers & Security, 2021, 106: 102307. [doi: 10.1016/j.cose.2021.102307]

[84] Wang JW, Li A, Li HR, Lu CY, Zhang N. RT-TEE: Real-time system availability for cyber-physical systems using ARM TrustZone. In: Proc. of the 2022 IEEE Symp. on Security and Privacy. San Francisco: IEEE, 2022. 352–369. [doi: 10.1109/SP46214.2022.9833604]

[85] Ministry of Industry and Information Technology of the People's Republic of China. Guideline for building a standard system for internet of vehicle network security and data security. 2022 (in Chinese). https://www.miit.gov.cn/jgsj/kjs/wjfb/art/2022/art_587f4340697f42c1a0a39271b4872592.html

附中文参考文献:

[57] 丁男, 梁文斌, 许力, 宋彩霞, 谭国真. 基于驾驶行为和速度的车内网 CAN 数据防注入攻击. 软件学报, 2017, 28: 1–10. https://www.jos.org.cn/jos/article/abstract/17001?st=search

[85] 中华人民共和国工业和信息化部. 车联网网络安全和数据安全标准体系建设指南. 2022. https://www.miit.gov.cn/jgsj/kjs/wjfb/art/2022/art_587f4340697f42c1a0a39271b4872592.html

陈博言(1996－), 男, 博士生, CCF 学生会员, 主要研究领域为操作系统安全, 车载系统安全, 权限最小化.

张鑫(2000－), 男, 博士生, 主要研究领域为系统安全, 侧信道攻击.

沈晴霓(1970－), 女, 博士, 教授, 博士生导师, CCF 高级会员, 主要研究领域为操作系统安全, 大数据与云安全, 区块链与隐私计算, 可信计算.

李聪(1990－), 男, 博士, 讲师, CCF 专业会员, 主要研究领域为应用密码学, 隐私计算, 区块链.

张晓磊(1995－), 男, 博士生, 主要研究领域为操作系统, 分布式系统安全, 云计算, 可信计算.

吴中海(1968－), 男, 博士, 教授, 博士生导师, CCF 杰出会员, 主要研究领域为大数据技术, 系统安全, 嵌入式软件.