

硕士学位论文

车内 CAN 总线入侵检测算法研究

**RESEARCH ON IN-VEHICLE CAN BUS
INTRUSION DETECTION ALGORITHM**

关亚东

哈尔滨工业大学

2019 年 6 月

国内图书分类号：TP309
国际图书分类号：621.3

学校代码：10213
密级：公开

工程硕士学位论文

车内 CAN 总线入侵检测算法研究

硕 士 研 究 生：关亚东

导 师：金显吉 助理研究员

副 导 师 李中伟 副教授

申 请 学 位：工程硕士

学 科：电气工程

所 在 单 位：电气工程及自动化学院

答 辩 日 期：2019 年 6 月

授予学位单位：哈尔滨工业大学

Classified Index: TP309

U.D.C: 621.3

Dissertation for the Master Degree in Engineering

**RESEARCH ON IN-VEHICLE CAN BUS
INTRUSION DETECTION ALGORITHM**

Candidate:	Guan Yadong
Supervisor:	Assistant researcher Jin Xianji
Vice-supervisor	Associate Prof. Li Zhongwei
Academic Degree Applied for:	Master of Engineering
Speciality:	Electrical Engineering
Affiliation:	School of Electrical Engineering and Automation
Date of Defence:	June, 2019
Degree-Conferring-Institution:	Harbin Institute of Technology

摘 要

随着汽车产业的蓬勃发展，车内 CAN 通信安全面临着越来越严峻的挑战。提高车内 CAN 通信安全性，对保证车联网安全以及司乘人员行车安全意义重大。目前，车内通信网络入侵检测技术是确保车内 CAN 通信安全而采用最广泛的技术之一，其通过实时监视 CAN 总线上的报文传输情况，在发现异常报文时进行告警。目前对于车内 CAN 总线的入侵检测算法的研究还不够成熟，检测算法存在漏检、误检及算法难以实现等问题。为解决以上问题，本文在分析车内 CAN 总线通信特点及各类攻击特征的基础上，提出了基于报文周期特性的自适应入侵检测算法与基于 DACHE 特征的入侵检测算法。

首先，研究了车内通信网络架构，对车内 CAN 通信协议进行了解析；分析了车内 CAN 通信存在的脆弱性及其面临的安全威胁，并确定了入侵检测算法的检测特征与评价指标。

其次，在分析实际车内 CAN 报文周期特征的基础上，针对注入型与中断型攻击，研究了基于报文周期特性的入侵检测算法；考虑到不同 ID 类型的报文具有不同的周期变化特性，分析了周期变化大小的影响因素；提出了基于报文周期特性的自适应入侵检测算法，分析了不同周期特性下的自适应检测阈值对检测精度的影响，给出了自适应检测阈值的计算方法。基于搭建的车内 CAN 通信模拟平台验证了所提自适应入侵算法的有效性。

然后，为检测伪造与重放攻击，给出了基于数据场汉明距离的入侵检测算法，分析了不同情况下该算法的检测精度并给出了改进思路；提出了基于 DACHE 特征的入侵检测算法，并以 BP 神经网络作为该算法的分类模型；在数据预处理过程中，针对数据类别不平衡的问题提出一种欠采样的数据选择方法；在神经网络的训练过程中，通过向网络参数迭代公式中加入动量项与自适应学习率，实现了经典 BP 算法的优化；基于 Python 进行了模型搭建、训练与测试。

最后，分析了一种前两种算法都无法有效检测的攻击方式——Bus-off 攻击，研究了其攻击原理、攻击过程、实现条件及具体实现方法；研究了攻击实现过程中恶意报文与被攻击报文的两种同步方式；分析了 Bus-off 攻击特征并提出了针对该类型攻击的检测方法；基于车内 CAN 通信模拟平台实现了 Bus-off 攻击及其入侵检测。

关键词：车内 CAN 总线；入侵检测；周期检测；DACHE 特征；Bus-off 攻击

Abstract

With the booming development of the automotive industry, especially the Internet of Vehicles, the safety of CAN communication in vehicles is facing more and more severe challenges. Improving the safety of CAN communication in vehicle is of great significance to ensure the safety of vehicle network and the safety of the passengers. It can also protect the healthy development of the automobile industry. At present, in-vehicle intrusion detection technology is one of the most used technologies for ensuring the safety of CAN communication in vehicle. It monitors CAN message transmission in the communication network in real time, and alerts the insiders when suspicious messages are found. At present, the research on the intrusion detection algorithm of the CAN bus in vehicle is not mature enough. The detection algorithm has problems such as missed detection, misdetection and difficulty in realizing the algorithm. In order to solve the above problems, based on the characteristics of CAN bus communication and various attack characteristics, this paper proposes an adaptive intrusion detection algorithm based on packet period characteristics and an intrusion detection algorithm based on packet data field features.

Secondly, based on the analysis of the characteristics of the actual CAN message period in vehicle, an intrusion detection algorithm based on the characteristics of the message period is proposed for the injection attack and the interrupt attack. Then the algorithm is improved. According to the different periodic variation characteristics of different ID packets, an adaptive intrusion detection algorithm based on packet periodicity is proposed. The effect of adaptive detection threshold on detection accuracy under different periodic characteristics is analyzed. A determination algorithm for the adaptive detection threshold is derived. The algorithm is implemented based on software and hardware.

Then, in order to detect forgery and replay attacks, an intrusion detection algorithm based on Hamming distance of data field is proposed. The detection accuracy under various conditions is analyzed in detail and the improvement ideas are given. A new input feature, DACHE feature and new under-sampling method of message data, is designed. And BP neural network is selected as the classification model for training and detection. The classical BP algorithm is optimized by adding momentum terms and adaptive learning rate, and the model is built, trained and tested based on Python.

Finally, a new attack method for the in-vehicle CAN network, Bus-off attack, is proposed. We study attack principle, attack process, implementation conditions and

implementation methods. Two methods of synchronizing malicious packets and attacked packets during attack implementation are mainly studied. Based on this, we analyze the characteristics of Bus-off attack and propose detection method for this type of attack. Based on STM32, Bus-off attack and detection experiments are implemented.

Keywords: In-vehicle CAN bus, intrusion detection, period detection, DACHE feature, Bus-off attack

目 录

摘 要	I
Abstract	II
第 1 章 绪论	1
1.1 课题背景及研究的目的和意义	1
1.2 国内外研究现状	2
1.2.1 车内 CAN 总线标准国内外研究现状	2
1.2.2 车内 CAN 总线入侵检测算法国内外研究现状	3
1.3 本文主要研究内容	6
第 2 章 车内 CAN 通信网络架构及威胁分析	8
2.1 引言	8
2.2 车内通信网络架构与 CAN 通信协议分析	8
2.2.1 车内通信网络架构	8
2.2.2 车内 CAN 通信协议 SAE J1939 解析	9
2.3 车内 CAN 总线脆弱性与威胁分析	12
2.3.1 车内 CAN 总线脆弱性分析	12
2.3.2 车内 CAN 总线面临的安全威胁分析	13
2.4 入侵检测算法的检测特征与评价指标	15
2.5 本章小结	16
第 3 章 基于报文周期特性的自适应入侵检测算法	17
3.1 引言	17
3.2 正常车内 CAN 报文周期特征获取与分析	17
3.3 攻击特征分析与检测方法	19
3.3.1 注入型攻击时间特征分析与检测方法	19
3.3.2 中断型攻击时间特性分析与检测方法	20
3.3.3 算法检测精度分析	21
3.4 基于报文周期特性的自适应入侵检测算法	23
3.4.1 原有检测阈值的优化	23
3.4.2 基于报文周期特性的自适应入侵检测算法设计	26
3.5 算法试验验证	28
3.5.1 试验设计	28
3.5.2 试验结果与分析	29

3.6 本章小结	30
第 4 章 基于报文数据场特征的入侵检测算法	31
4.1 引言	31
4.2 基于数据场汉明距离的入侵检测算法.....	31
4.2.1 检测原理.....	31
4.2.2 检测准确率分析与改进	33
4.3 基于 DACHE 特征的入侵检测算法.....	34
4.3.1 DACHE 特征	34
4.3.2 分类模型.....	35
4.4 算法实现与仿真	37
4.4.1 仿真环境与数据集	37
4.4.2 数据预处理.....	37
4.4.3 模型建立与仿真	39
4.5 算法优化与仿真.....	41
4.5.1 算法存在的不足	41
4.5.2 算法优化.....	42
4.5.3 仿真结果与分析	44
4.6 本章小结	45
第 5 章 Bus-off 攻击及其检测算法研究.....	47
5.1 引言	47
5.2 Bus-off 攻击原理与实现条件	47
5.2.1 前提假设.....	47
5.2.2 攻击发生原理	47
5.2.3 攻击发生条件	48
5.3 Bus-off 攻击过程分析.....	49
5.4 Bus-off 攻击实现方法分析.....	51
5.5 Bus-off 攻击试验	55
5.5.1 试验设计	55
5.5.2 试验结果.....	56
5.6 Bus-off 攻击特征分析与检测	61
5.7 本章小结	62
结 论	63
参考文献	65
攻读硕士学位期间发表的论文及其它成果.....	68

哈尔滨工业大学学位论文原创性声明和使用权限.....	69
致 谢	70

第 1 章 绪论

1.1 课题背景及研究的目的和意义

随着汽车产业的迅猛发展，2018 年我国汽车保有量已达 2.4 亿辆，我国已成为汽车产销量全球第一、保有量全球第二的国家，预计 2019 年我国汽车保有量将超越美国成为全球第一。目前，汽车产业正朝着电动化、网联化以及智能化的方向发展^[1]。近年来，随着车联网、无人驾驶、智能交通，以及车载移动通信技术和计算机网络的不断发展，汽车与外部的信息交互的方式也更加多种多样，信息交互的频率也越来越高。2017 年，我国有关部门发布了《汽车产业中长期发展规划》，将智能网联汽车^[2]作为我国汽车产业发展战略目标。智能网联汽车是将车内网络与互联网、其他汽车以及智能交通基础设施连接在一起，形成一个融合网络，从而使得汽车能够实现更加多样而强大的功能。

汽车特别是网联汽车行业的飞速发展为用户带来了极大的舒适和便捷，与此同时，随着车内通信网络开放性的提高，黑客对于车内通信网络的攻击路径也越来越多。CAN 是目前最主要车内通信方式，其相当于汽车的神经网络，汽车内的绝大多数 ECU（Electronic Control Unit，电子控制单元）均基于 CAN 总线进行通信。但是，CAN 总线本身的安全防护机制比较脆弱。近几年各类汽车信息安全事件频发，且各汽车厂商对车内通信安全问题的认识还明显不足。2015 年 Miller 和 Valasek 远程入侵了 Jeep 自由光的车载娱乐系统，并获取了车内 CAN 总线的读写权限与汽车控制权，进而向变速器与方向控制系统发送错误指令，导致该车偏离驾驶方向并冲入路边斜坡^[3]，此事件导致克莱斯勒公司召回了近 140 万辆汽车。德国汽车协会（Verband der Automobilindustrie，VDA）于 2015 年 2 月发布报告称：几乎所有宝马品牌的车型都存在着严重的通信安全漏洞，黑客利用该漏洞可在数分钟之内远程开启车门^[4]。2016 年，腾讯科恩实验室的研究人员通过无线方式入侵了一辆特斯拉 Model X，并取得了车内若干 ECU 的控制权，且研究人员声称可以破解全球任何一辆特斯拉汽车。甚至美国一位年仅 14 岁的黑客只花费 15 美元就制作了一个电子遥控自动通信装置，为车内 CAN 总线安装了一个后门。2018 年网联汽车遭受黑客恶意攻击的数量比 3 年前增加了近 6 倍^[5]。针对车内 CAN 总线进行的攻击轻者可干扰正常驾驶，或使车辆无法正常行驶，瘫痪道路交通；重者甚至可造成车毁人亡，杀人于无形。

无论黑客通过何种方式对车内通信网进行攻击，其最终落脚点都是干扰或改变 CAN 总线上正常报文的发送，从而人为制造车辆异常状态。因此，保证车

内 CAN 通信安全是确保车辆通信安全最后也是最重要的一道防线^[6-7]。

入侵检测系统^[8] (intrusion detection system, IDS) 是目前针对车内 CAN 通信安全研究最广泛的安全措施之一, 其通过实时监视 CAN 总线上的报文传输, 在发现可疑的报文时对车内人员进行告警提示或采取主动应对措施。在 CAN 总线上部署入侵检测节点无需对总线上已有的 ECU 进行改动, 只需对其中某些节点进行改动或增加一个专用的安全检测节点, 并且不会对原有 CAN 通信造成影响。

传统的入侵检测技术主要基于 TCP/IP 网络进行安全检测和防御, 而针对车内 CAN 总线的入侵检测算法研究较少, 且由于车内 CAN 总线的特殊的通信机制与通信环境, 因此将应用于互联网的通用入侵检测技术直接用于车内入侵检测的效果并不理想。

综上所述, 车内通信网络特别是 CAN 网络的安全面临着越来越严峻的挑战, 而当前对于车内 CAN 网络的入侵检测算法仍未成熟。因此, 需首先对车内 CAN 总线面临的安全威胁进行分析, 在此基础上研究并实现相应检测算法, 可及时发现入侵行为与入侵造成的车辆故障, 确保司机及乘客的生命财产安全, 保证道路交通安全。从长远来看, 对于车内 CAN 通信入侵检测技术的研究对于整个车联网的安全也意义重大, 也为汽车产业健康发展保驾护航。

1.2 国内外研究现状

1.2.1 车内 CAN 总线标准国内外研究现状

国际标准化组织 (International Organization for Standardization, ISO) 与美国汽车工程协会 (Society of Automotive Engineers, SAE) 对车内 CAN 通信制订了一套完整的标准体系, 如表 1-1 所示。

ISO11898 标准和 ISO11519-2 标准详细规定了 CAN 通信的物理层与数据链路层的设计规范, 其中 ISO11898 与 ISO11519-2 分别是针对通信速率为 125kbps~1Mbps 的高速 CAN 与 125kbps 以下的低速 CAN 而制定的标准。

美国汽车工程协会针对不同的应用环境, 制定了 SAE J1939、SAE J2284 以及 SAE J2411。SAE J1939 是专门针对商用车的 CAN 通信标准, 主要用于商用车、轨道机车等中重型道路车辆, 目前该标准也用于制定电动汽车内的 CAN 通信协议。SAE J2284 用于汽车动力、传动系统等高速网络; SAE J2411 用于汽车车身系统等低速网络。另外, 不少各小型汽车生产商出于行业保密与公司利益, 采用各自公司制定的应用层协议。国内对于车内 CAN 通信标准的研究与制定工作很少, 各汽车生产商大都采用国际标准或参照国际标准进行制定。

表 1-1 CAN 相关标准

标准号	标准名称	说明
ISO11898	《道路车辆 控制器局域网络》	规定了数据链路层和物理层，用于 125kbps-1Mbps 的高速 CAN。
ISO11519-2	《道路车辆 低速串行数据通信》	125kbps 以下的低速 CAN 通信标准。
SAE J1939	《商用车控制系统局域网络通信协议》	主要用于中重型车辆，以及部分电动汽车的 CAN 网络的应用层标准。
SAE J2284	《适用于 500kbps 车辆应用的高速 CAN》	用于汽车动力、传动系统等高速网络。
SAE J2411	《用于车辆的单线 CAN 网络》	用于汽车车身系统等低速网络，通信速度：33.3kbps、83.3kbps。

1.2.2 车内 CAN 总线入侵检测算法国内外研究现状

(1) 国外研究现状

国外对于车内 CAN 通信网络信息安全研究较早。早在 2010 年，Koscher 等学者首次提出并证明了通过物理访问向车内 CAN 总线注入报文即可控制车辆功能的可行性^[9]。他们分析了车内 CAN 通信报文并找到了控制车辆命令的报文，通过注入恶意 CAN 报文即可使得各种部件发生故障。2011 年，Checkoway 等人提出可通过多种间接方式访问车内 CAN 网络^[10]，如娱乐系统（包括 CD 播放器或外部数字多媒体端口）。由于娱乐系统与车内 CAN 网络相互连接，黑客可通过受感染多媒体系统向车内 CAN 网络发送恶意报文。这些攻击的局限性在于需要满足某些物理条件才能实现。2015 年，白帽黑客 Miller 和 Valasek 首次演示了利用无线方式在未改装车辆上接入车内 CAN 网络的方法^[11]。他们利用连接到蜂窝网络的远程通信设备的漏洞从而远程发送恶意 CAN 报文。2016 年 Charlie Miller 和 Chris Valasek 演示了通过物理连接将目标 ECU 设为维护模式并利用另一 ECU 发送假命令从而控制了汽车方向盘与刹车。随着越来越多的车内通信网络漏洞被发现，针对车内 CAN 通信的入侵检测技术的研究也得到不少学者的关注。

文献[12]首次提出了车内入侵检测系统的概念与特点。根据检测特征的不同，目前的研究主要分为四类，分别为：报文的周期性特征、电气特征、报文数据场特征、报文周期与数据场构成的混合特征。

由于车内大多数报文都具有周期性，因此部分学者将报文周期特性作为检测特征，其主要是基于周期统计、信息熵以及多元时间序列分析等方法。文献[13]采用检测报文周期的改变从而检测入侵，文献[14-15]提出了基于信息熵的入侵检测方法。然而基于报文频率与信息熵的 IDS 无法检测到模拟正常报文发

送频率从而发送恶意报文的攻击，如 Greenberg 就成功绕过了基于报文频率的 IDS 进而实现了对车内 CAN 网络的攻击^[11]。Lee 等人提出一种新型的远程帧的车内网络入侵检测系统，其通过测量数据帧和远程帧之间的延迟从而来判断总线上是否发生了入侵^[16]。然而，文献[15]证明了基于时钟偏差的检测机制易受到隐藏式攻击^[17]，黑客可通过模仿正常节点的发送时序从而进行攻击，这表明基于时序的入侵检测系统仍存在脆弱性。

部分学者提出可通过检测并识别 CAN 总线上的报文电气特性从而识别恶意报文。Murvay 和 Groza^[18]提出正常情况下，各报文的发送节点相对固定，通过检测总线上电压信号特征可识别 ECU 与其发送的正常报文，但由于其未考虑到仲裁过程时的信号叠加，因此其方法识别的准确率低。Cho 和 Shin^[19]提出了一种基于时钟的 IDS，该检测方法利用了每个 ECU 都具有独特时钟偏差的特性，该方法优点是无需对现有 ECU 进行任何修改。但之后 Sagong 等人通过实验证明了该算法无法检测到通过模拟正常 ECU 的时钟偏差而进行的攻击。因此，以电气特性作为检测特征的检测方法也不够理想。

部分学者提出黑客在进行攻击时 CAN 报文数据场内容会出现异常，通过对报文数据场内容进行检测可识别入侵行为，所采用算法主要包括机器学习^[20]、深度学习^[21]和回归学习^[22]。文献[23-25]提出了基于神经网络的 CAN 总线入侵算法，虽然其检测率高，但该方法进行检测的数据源十分简单，与实际车内 CAN 通信环境有很大差距，且车内硬件环境也难以满足这些算法的运行需求，因此其并未真正用于实际的车内网络。文献[26]中提出了基于隐马尔可夫模型的检测方法。文献[27]提出了一种更形式化的基于有限状态自动机的入侵检测方法，其仍然只能检测到某些类型的攻击，且存在误报与漏报情况。

由于单独对报文周期和报文数据场内容进行检测都存在一定局限性，因此有学者提出了将报文周期与内容构成的混合特征作为检测算法的输入特征。文献[28]提出还需对数据段进行检测才能弥补基于时序的入侵检测系统的不足，在此基础上提出了一种基于 Bloom 过滤器的入侵检测算法，其同时考虑到了报文周期性与报文数据场内容，通过基于 SAE J1939 的 CANoe 仿真以及实际车辆的测试表明，该算法具有良好效果，该算法对模糊攻击检测效果较好，而若黑客利用符合该型号车辆的车内 CAN 通信应用层标准的伪造报文进行攻击，则该检测算法的检测效果欠佳。

（3）国内研究现状

国内对于车内 CAN 总线通信安全与相应的入侵检测的研究起步较晚但发展很快，无论是企业界还是学术界都认识到了车内 CAN 总线通信安全的必要性与重要性并开始进行许多研究。

2016 年, 电子科技大学联合中国互联网应急响应中心、国内外多家汽车厂商以及奇虎 360 为代表的网络安全公司共同成立了车联网络安全委员会。该委员会于次年 2 月发布了《车联网络安全防护指南细则》^[29]。2016 年 11 月, 我国首个专门从事汽车以及车联安全防护的跨行业合作机构——车联安全中心成立, 其发布了 CANPICK、汽车卫士等汽车安全工具。

2017 年, 中国信通研究院发布《车联网络安全白皮书》^[30], 该文件系统分析了车联目前的安全现状与其面临的信息安全威胁, 并探讨了相应的防护策略, 并且对未来车联网络安全的发展前景进行了展望, 以此促进车联的安全健康发展。奇虎 360 公司 Unicorn Team 团队发布了针对 CAN 总线入侵检测的 CANS see 系统^[31], 该检测系统通过机器学习方法检测采集到的 CAN 报文数据是否存在异常。

2018 年 1 月百度于发布了 Apo1lo2.0 系统^[32], 并基于该系统展示了其为保障车内网络通信安全所取得的成果, 其中就包括车内入侵检测与防御系统, 该系统采用了多种车内 CAN 总线的入侵检测与防御技术, 如 CAN 指令的智能识别与过滤技术, 识别特定风险控车指令技术等。

吉林大学的于赫^[33]提出了基于信息熵的 CAN 总线异常检测算法, 通过得出不同 ID 报文的信息熵并与正常情形下的信息熵基线进行对比从而进行检测; 于赫还以 CAN 报文数据场内容为检测特征, 提出了基于 CART 决策树的入侵检测算法。该研究存在的不足是其未研究攻击的具体实现方法以及实现的可行性。同课题组的闫鑫^[34]通过仿真表明基于 Renyi 熵的异常检测方法对于检测车内网络的洪泛攻击与重放攻击具有较好效果。戚琦^[35]专门针对车内 CAN 总线的认证泛洪攻击与报文泛洪攻击提出了非参数 CUSUM 算法。文献[36]根据 CAN 网络数据包结构的特点, 提出了一系列攻击方法, 其中包括 CAN 总线的外部接入、报文逆向解析等方法, 并在此基础上以报文 ID 为检测特征提出基于信息熵的入侵检测算法, 以数据场内容为检测特征提出了基于支持向量机的入侵检测算法。

电子科技大学的曾凡^[37]设计了位级别规则和字节级别规则用于入侵检测并实现了完整的入侵检测引擎。该文献的不足之处在于所设计入侵检测系统未经过实际车内 CAN 通信数据集的检验。

北京邮电大学的曾润^[38]通过将车内 CAN 报文分为时间触发报文和事件触发报文, 其针对两类报文分别提出了基于发送时间间隔与 C4.5 决策树算法的入侵检测算法。

成都信息工程大学的吴贻淮^[39]将神经网络算法引入到车内 CAN 报文入侵检测中, 设计了基于 PCA-BP 神经网络的 CAN 数据包发送频率检测算法以及基

于 GA-RBF 神经网络的车内网络数据关联性检测算法并证明了算法有效性。

同时，上述文献所提入侵检测算法无法检测到某些类型的攻击，如与正常报文同步注入且符合正常报文数据格式的伪造攻击。且上述部分文献中提出的算法也仅仅是通过仿真与汽车通信模拟平台来实现，缺乏在真实车内 CAN 通信环境中的验证。

总体来说，国内的企业与学术界对于车内 CAN 通信安全的研究已取得了一系列成果，其中不少成果已经应用于实际汽车并且效果良好，但是大部分成果都是基于国外学者的研究基础上进行的改进，车内 CAN 通信安全仍然需国内专家和学者持续研究和探索。

1.3 本文主要研究内容

本文拟针对车内 CAN 总线面临的信息安全威胁，研究并提出三种入侵检测算法，主要研究内容包括：

(1) 研究车内通信网络架构，对车内 CAN 通信协议进行解析；分析车内 CAN 总线存在的脆弱性，研究针对车内 CAN 总线攻击的途径、类别及造成后果；确定入侵检测算法技术需求与评价指标。

(2) 分析实际车内 CAN 报文周期特征，针对注入型与中断型攻击，给出一种基于报文周期特性的入侵检测算法；对该算法进行改进，针对不同 ID 报文的周期变化特性，提出一种基于报文周期特性的自适应入侵检测算法；分析不同周期特性下自适应检测阈值对检测精度的影响，并给出自适应检测阈值的确定方法；利用 CANoe 搭建车内 CAN 通信仿真环境，基于 Python 实现所提自适应入侵算法并进行仿真验证；利用 STM32F407ZGT6 搭建车内 CAN 通信模拟平台，采用 C 语言实现所提自适应入侵算法并进行试验验证。

(3) 针对伪造与重放攻击，提出了一种基于数据场汉明距离的入侵检测算法，详细分析各种条件下的检测精度并给出改进思路；提出基于 DACHE 特征的入侵检测算法，选择 BP 神经网络作为分类模型；针对数据类别不平衡的问题，提出一种欠采样的数据选择方法；通过向网络参数迭代公式中加入动量项与自适应学习率，实现对经典 BP 算法的优化；基于 Python 进行模型搭建、训练与测试。

(4) 给出一种针对车内 CAN 总线的攻击类型——Bus-off 攻击。研究 Bus-off 攻击原理、攻击过程、攻击实现条件与方法、攻击实现过程中恶意报文与被攻击报文的两种同步方式；分析 Bus-off 攻击特征并提出一种针对该类型攻击的入侵检测方法；基于 STM32F407ZGT6 实现 Bus-off 攻击及其入侵检测。

本文的组织结构如图 1-1 所示。

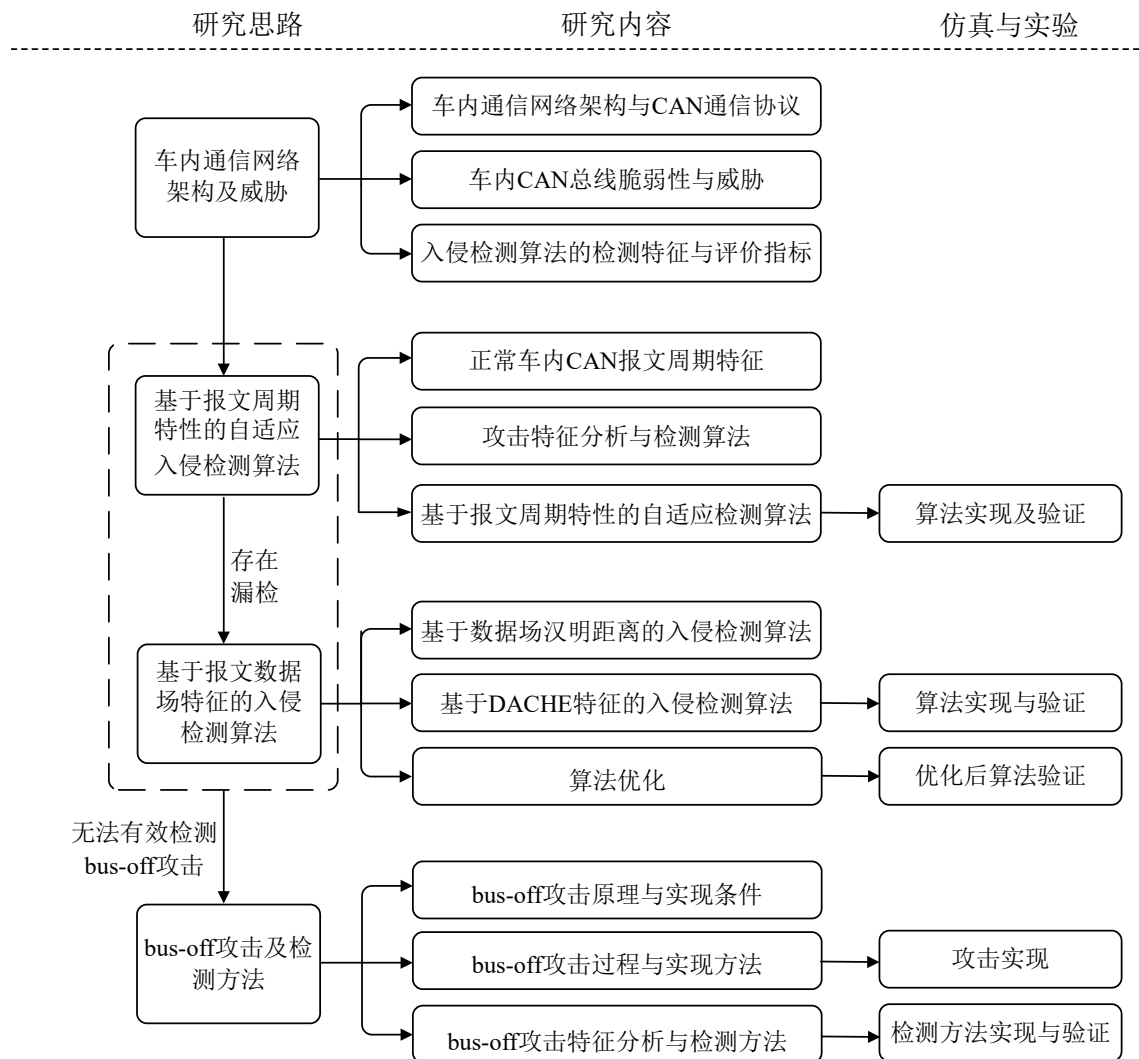


图 1-1 本文组织结构

第 2 章 车内 CAN 通信网络架构及威胁分析

2.1 引言

要设计车内 CAN 总线入侵检测算法首先需明确车内 CAN 通信特点、脆弱性及其面临的威胁类型与特征。因此，本章从车内通信网络架构与具有代表性的车内 CAN 通信协议——SAE J1939 协议等方面入手，进行车内 CAN 总线的脆弱性分析，研究各种针对车内 CAN 网络的攻击的实现途径及各类攻击造成的后果，确定了入侵检测算法的检测特征与评价指标。

2.2 车内通信网络架构与 CAN 通信协议分析

2.2.1 车内通信网络架构

不同品牌汽车的车内通信网络架构都存在差别，图 2-1 为某品牌车型的车内通信网络架构，其主要包括以下部分：

（1）CAN 网络。CAN 网络为车内网络最主要的子网络，其主要包含三部分：1）动力总成系统网络，其波特率为 500kbps，其挂载的 ECU 主要有变速箱电控系统、电子控制制动系统及发动机管理系统等；2）车身控制系统网络，其波特率为 100kbps，其挂载的 ECU 包括安全气囊控制系统、车身控制系统等；3）底盘控制系统网络，其通信波特率也为 500kbps，其挂载的 ECU 包括动力转向系统、胎压监测系统、ABS 系统等。

（2）网关。网关为车内网络信息交互中心，可用于转发与其相连的各子网络的报文数据。

（3）OBD-II 接口。OBD-II 接口（On-Board Diagnostics，车载诊断接口）是外部设备通过物理方式接入到车内网络的主要接口，专业维修人员通过专用设备接入该接口进而实时监控汽车运行状态并做出诊断。

（4）MOST 总线、TBox/IVI 等。MOST 总线为面向媒体的系统传输总线，其数据传输速度可达 24.8Mbps，其主要用于车内娱乐系统。车载联网设备（Telematics Box，TBox）与网关相连，车内网可通过该设备接入移动互联网。对于某些车型，其网关可直接与互联网相连。

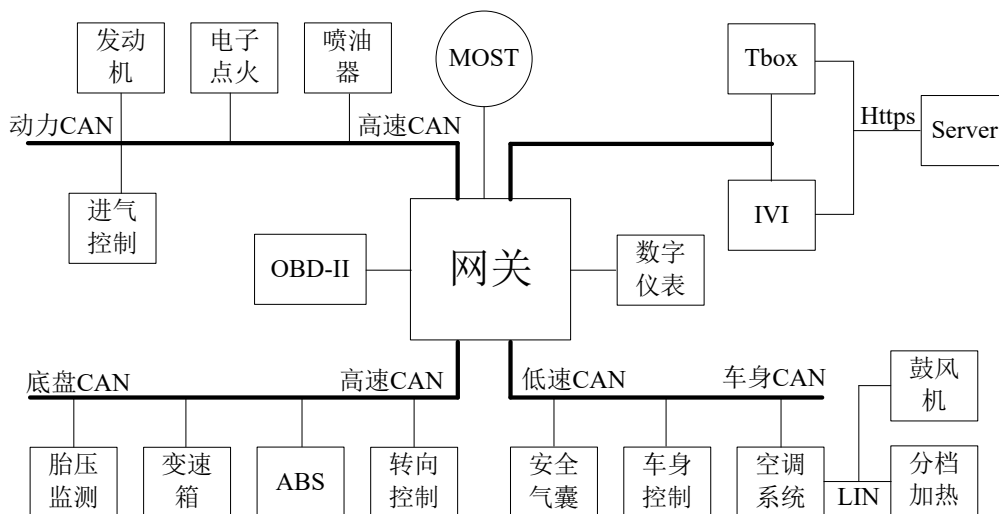


图 2-1 车内通信网络架构

2.2.2 车内 CAN 通信协议 SAE J1939 解析

SAE J1939 主要对商用车内 CAN 通信数据链路层以及应用层内容进行了规定。该协议主要针对扩展帧格式报文进行了定义，并且指定了报文的具体结构、报文类型与内容。以下具体分析该协议对数据链路层以及应用层的规定

(1) 分组传输机制

扩展数据帧数据场长度为 8 字节，SAE J1939 规定网络中大部分数据都以参数的形式进行传递，而通常一个参数用不了 8 字节，为便于报文管理，同时充分利用每帧报文所能携带的 64bit 的数据空间，SAE J1939 协议将各参数进行打包并分组传输。

SAE J1939 协议为每个参数都分配了唯一的索引值，即 SPN (Suspect Parameter Number, 可疑参数编号)，而后根据各参数之间的关联性与发送频率的不同，将各 SPN 进行分组，并为每个组分配唯一的索引编号，即参数组编号 (PGN)。CAN 扩展数据帧、PDU (Protocol Data Unit, 协议数据单元) 以及 PGN 以及三者之间的对应关系如图 2-2 所示。

PGN 长度为 24 位，包括以下部分：填充位 (6 位)、保留位 R (1 位)、数据页位 DP (一位)、PDU 格式域 PF (8 位) 以及组扩展域 PS (8 位)。PF 用于设置 PDU 的两种格式类型：PDU1 与 PDU2。如表 2-1 所示，若 $PF < 240$ 时，PS 置为 0，采用 PDU1 格式，用于向特定目标地址的发送；若 $PF \geq 240$ ，PS 置为组扩展值，采用 PDU2 格式，用于向全局地址的发送，此时 PF 的低四位与 PS 共 12 位规定了共 4096 个参数组。

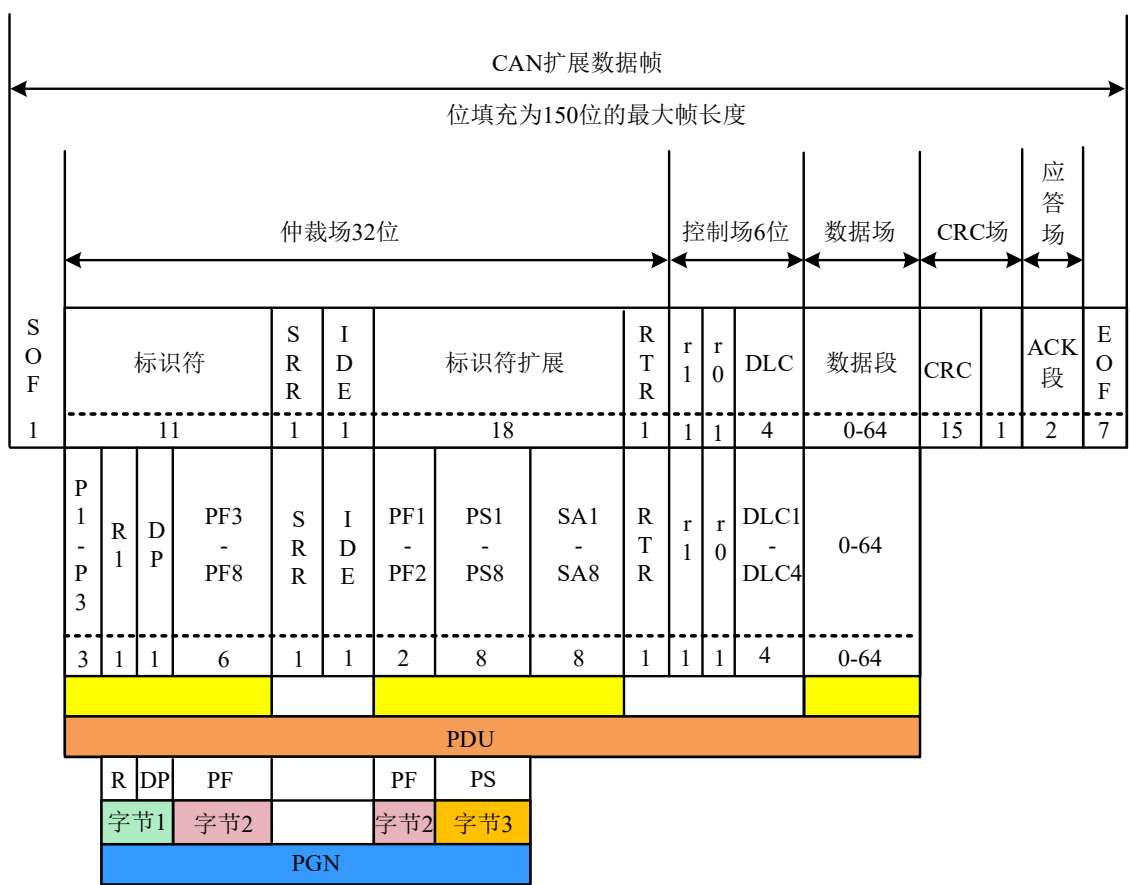


图 2-2 CAN 扩展帧、PDU 与 PGN 的对应关系

表 2-1 两种格式 PDU 的特点

	PDU 格式 (PF) 域	特定 PDU (PS) 域
PDU1 格式	0-239	目标地址 DA
PDU2 格式	240-255	组扩展

PDU1 与 PDU2 可用的参数组编号的总数目可用下式表示:

$$(240 + (16 \times 256)) \times 2 = 8672 \quad (2-1)$$

通过采用两种协议数据单元格式,不但能实现与特定地址的通信,还可提供足够数量的报文 PGN 组合。

(2) 参数格式定义

SAE J1939 规定了两种数据类型:测量数据与状态数据。测量数据的值表示 ECU 对某参数进行测量后而返回的值。根据测量数据值的连续性来划分,测量数据的值包括连续值与离散值。SAE J1939 规定 ECU 在执行动作后无需返回动作已完成的确认信息,因此通过状态数据来表示 ECU 在执行动作指令后产生的结果,因此状态数据也可称为控制数据。例如:某状态数据的值表示车内某

个电磁线圈被激活，无需对该线圈进行测试以确认其被激活。状态类型的数据有：自动巡航是否在运行、发动机制动是否启动等。spn2357 为车辆工作灯是否开启的数据，其数据类型为状态数据；而 spn2357 为车辆工作灯的当前状态，其数据类型为测量数据。

表 2-2 与 2-3 分别表示连续测量值与离散测量值的取值范围，表 2-4 为控制指令的取值范围。

表 2-2 传输信号的数值范围

范围名称	1 字节	2 字节	ASCII
有效信号	00 ₁₆ 到 FA ₁₆	0000 ₁₆ 到 FAFF ₁₆	01 ₁₆ 到 FE ₁₆
特定参数指示	FB ₁₆	FB00 ₁₆ 到 FBFF ₁₆	无
保留给将来指示使用的范围	FC ₁₆ 到 FD ₁₆	FC00 ₁₆ 到 FDFF ₁₆	无
错误指示	FE ₁₆	FExx ₁₆	00 ₁₆
不可用	FF ₁₆	FFxx ₁₆	FF ₁₆

表 2-3 离散参数的数值范围（测量值）

范围名称	数值（两位）
禁止（关闭，非运行等）	00
启动（打开，正在运行等）	01
错误指示	10
不可用	11

表 2-4 控制命令的数值范围（状态值）

范围名称	数值（两位）
用于停止功能的命令（关闭等）	00
用于启动功能的命令（打开等）	01
保留	10
无关紧要/无动作	11

表 2-2 与 2-3 中的错误指示值用于指示功能模块发生错误，即若某组件发生故障后导致其无法发送对应参数数据，则采用表中的错误指示值作为该参数的值进行发送。若发送“不可用”数值则表示模块发送的报文中所包含的参数在该模块中不支持或不可用。若测量或计算出来的数值超出了规定范围，那么应用最小或最大参数值进行传输。

SAE J1939 定义了车内 CAN 报文中仲裁场与数据场中的各个参数，包括参数组编号（PGN）与可疑参数号码（SPN），数据场各数据的类型、长度、范围与分辨率。可疑参数号码为某项具体参数所对应的索引，在进行故障诊断时通过该索引即可得到对应参数值，如方向盘角度参数的 SPN 为 1807。对于连续性

参数，其物理数值与总线数值的对应关系为：物理值=总线数值×分辨率+偏移量。

2.3 车内 CAN 总线脆弱性与威胁分析

2.3.1 车内 CAN 总线脆弱性分析

虽然 CAN 通信是一种高效的通信方式，但 CAN 协议在设计之初未充分考虑安全性这一重要因素。在目前车内通信环境下，由于各种外部通信方式的接入，使得车内 CAN 总线的开放性越来越高，因此其也暴露出越来越多的脆弱性：

(1) CAN 本身基于过滤器的接受机制存在脆弱性。通过配置 CAN 节点的过滤器可接受来自特定节点的报文，同样地，黑客利用该特性通过配置某节点的过滤器可收到总线上的所有报文，继而可对报文协议的进行逆向分析。

(2) 缺乏数据保护机制。CAN 协议中仅有 CRC 查错校验功能，而无任何数据加密、消息认证码、消息摘要以及数字签名等信息安全防护机制，无法保障报文数据的保密性、真实性与不可抵赖性。原因有两点：一是报文数据场长度有限制，其最长仅为 8 字节，因此应用输出长度超过 20 字节的加密哈希函数的方法是不现实的；二是受限 ECU 的处理能力，若对数据进行加解密则会影响报文实时性。

(3) 缺乏针对拒绝服务攻击的保护机制。CAN 报文的发送严格遵循基于报文仲裁场优先级的竞争发送机制。若黑客通过某一 ECU 以极高频率向总线上发送最高优先级的报文，这会使得其他较低优先级报文被长时间阻塞，导致某些 ECU 无法正常运行。不仅如此，由于 CAN 具有特殊的错误处理机制，黑客可诱使 ECU 使其认为自身出现错误，从而使该节点主动进入脱离总线状态而拒绝服务。

(4) 缺乏数据源认证机制。

CAN 节点收到报文时无法分辨该报文的发送方是否是虚假的。这使得恶意节点可向总线上发送任意伪造 ID 的报文，但总线上的其他正常节点无法察觉到恶意节点的存在。

(5) CAN 控制器的脆弱访问与配置方式。

目前主流的 CAN 控制器 MCP 2515 和 SJA 1000 在一些模式下都允许通过软件命令修改控制器配置，例如 SJA 1000 在 Basic CAN 模式下，若复位请求位被置高，那么可访问修改接受屏蔽寄存器 AMR。而对于带有 MCP2515 CAN 控制器的 ECU，不仅可以在通电或重置时进入配置模式，还通过串行外围接口

(SPI) 通过用户指令进入配置模式。用户通过 SPI 即可读取/写入 CAN 控制器寄存器。因此, 利用用户级功能黑客就可通过软件命令进入配置模式修改原有数据。

2.3.2 车内 CAN 总线面临的安全威胁分析

(1) 攻击途径

由于汽车开放性越来越高, 因此黑客可通过许多途径入侵车内 CAN 通信网络, 与 CAN 总线相连的 ECU、OBD 接口以及车载终端设备都可作为入侵 CAN 总线的途径。黑客通过这些途径得到总线的任意读写权限后, 便可进一步实施各种攻击。具体来说, 这些攻击途径包括无线网络(蜂窝网络、蓝牙、射频、V2V/V2I)、诊断接口(OBD, OBD-II)、娱乐系统等。其中 V2V 通信指车辆之间基于无线的数据通信, V2I 通信指车辆与基础设施基于无线的数据通信。目前各种 OBD 工具被广泛使用, 用户利用这些工具可在移动设备上检测车辆运行状态、分析车辆数据与故障诊断, 甚至远程控制车门、车内空调等, 同样, 这些实用的工具也可能被黑客利用而成为其侵入车内 CAN 总线的重要途径, 可能也会有不怀好意的汽车维修人员利用诊断接口侵入车内 CAN 总线。黑客通过 ECU、OBD 接口或车载终端设备等途径接入车内网络后, 其可通过修改 CAN 控制器中的数据从而实现进一步攻击。

(2) 攻击类型

针对车内 CAN 总线的攻击类型如图所示。根据不同攻击在攻击过程中对车内 ECU 造成的影响程度, 可将攻击分为两大类: 弱攻击和强攻击, 相应地, 受这两种攻击影响的 ECU 分别为弱受损 ECU 和完全受损 ECU。可通过弱攻击, 黑客可使弱受损 ECU 停止发送某些报文或使其处于仅监听模式, 但黑客无法通过该弱受损 ECU 向 CAN 总线注入任何伪造的报文。而对于完全受损的 ECU, 黑客可完全控制该 ECU 并访问其内存数据, 在这种情形下, 黑客具有更多攻击方式, 不仅可使受损 ECU 停止发送报文或使其处于仅监听模式, 还可通过重新配置 ECU 向 CAN 总线注入任意报文扰乱 CAN 通信, 由于此时黑客可访问存储在其内存中的任何数据, 因此即使在 ECU 中内置了预防性安全机制(如 MAC), 黑客也通过禁用这些安全措施从而使其失效。

更具体地, 弱攻击主要包括报文窃取与恶意中断。由于任何总线上的节点都可以获取总线上传输的报文, 黑客入侵到某节点后即可监控并记录 CAN 总线上的报文, 从而进行报文解析, 以便于进行后续进一步的攻击。若黑客中断某些报文发送, 则相关 ECU 无法收到有效报文。

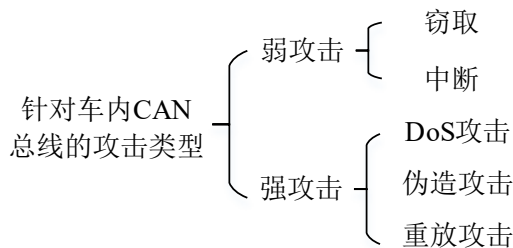


图 2-3 针对车内 CAN 总线的攻击类型

强攻击包括 DoS^[40]、伪造、重放等攻击。DoS 攻击即为恶意节点以极高速率向 CAN 总线注入大量高优先级报文，占用总线资源，从而使得其他较低优先级的正常报文无法传输。伪造攻击即为将伪造报文通过恶意节点发送至总线。其包括的攻击类型如表 2-5 所示。重放攻击为黑客首先窃听并记录某正常 CAN 报文，之后在其他任意时刻将该报文进行重放。重放的报文不仅可以为车内 CAN 报文，而且包括从车外发送到车内的 CAN 报文，如有关远程开启车门命令的报文。

表 2-5 伪造攻击的类型

伪造攻击类型	描述
Fuzzing 攻击 ^[41]	向总线注入随机 ID 与随机数据的报文，干扰或污染正常通信。黑客可通过 Fuzzing 攻击破解车辆私有协议。
伪造畸形报文	向总线注入违反 CAN 总线协议规范的 CAN 报文，可能引起系统错误。
伪造诊断报文	向总线上中注入诊断报文，从而采集信息或控制汽车。在汽车行驶过程中，注入诊断报文是一种很危险的行为，极易导致汽车失控，导致安全事故的发生。

（3）攻击造成影响

黑客通过弱攻击可窃听车内 CAN 通信报文造成敏感数据泄露，或造成 CAN 通信中断；而黑客通过强攻击不仅可造成弱攻击造成的问题，还可能瘫痪车内 CAN 通信，造成操作失灵；或人为造成车内异常状况，干扰驾驶员的正常驾驶，影响司乘人员乘车舒适度，甚至造成车辆在行驶中的非正常急转弯或刹车，大大增加发送交通事故的几率，严重威胁司乘人员的生命安全。

表 2-6 为目前车内最易遭受黑客入侵的部分 ECU 以及这些 ECU 受攻击后造成的后果，这些 ECU 的正常工作对车内正常通信、车辆安全以及车内人员的安全十分关键，如安全气囊 ECU、转向与刹车 ECU 等。表中 ADAS（Advanced Driving Assistant System，高级驾驶辅助系统）的主要作用为收集车辆周围的环境数据并进行物体的侦测与辨识，并及时向驾驶员发出告警信息，确保车辆的安全行驶。TPMS 为轮胎压力监测系统（Tire Pressure Monitoring System），其

作用为采集胎压、胎温等数据并进行实时显示，并在轮胎出现异常时提醒驾驶者。

表 2-6 部分易受攻击的 ECU 以及受攻击后的后果

易受攻击的 ECU	可能造成的后果
引擎与传动控制 ECU	遥控车辆，使车辆无法启动，使行进中的车辆失控，导致事故。
转向与刹车控制 ECU	遥控车辆，使车辆转向与刹车失效，导致事故。
安全气囊控制 ECU	安全气囊误动或拒动，无法保证司机安全
车灯控制 ECU	恶意开启车灯或使车灯无法正常开启，影响交通安全
ADAS ECU	使车内传感器异常报警或无法正常报警，令司机误判
TPMS ECU	使胎压监测出现异常报警或无法正常报警，影响行车安全

CAN 总线为 ECU 之间提供了高效、可靠、经济的通信链路。然而，CAN 总线没有足够的安全特性来保护自己免受内部或外部攻击。

2.4 入侵检测算法的检测特征与评价指标

(1) 入侵检测算法的检测特征选取

已有研究中所提入侵检测算法均只能检测到特定类别的攻击，且都存在一定的几率检测错误，因此可将多种检测算法结合，使其优势互补，这样一方面不仅可检测到尽可能多种类型的攻击，另一方面可降低检测错误率。本文分别将报文周期与数据场内容作为检测特征并设计两种检测算法。两种算法可检测到绝大多数攻击，但理论上由于 CAN 协议的自身特性，仍存在少数攻击无法被检测到，因此需要针对这些攻击设计相应的检测算法。

(2) 检测算法评价指标

入侵检测算法的评价指标主要包括检测精度、数据处理性能、自身安全性与对原系统的影响程度。

检测精度是入侵检测算法最基础也是最重要的评价指标。入侵检测系统对数据进行检测得到的结果包含四类，如下表 2-7 所示。对于入侵检测系统来说，需重点关注其误报与漏报情况，因此可将假阳性率（False Positive Rate, FPR）与假阴性率（False Negative Rate, FNR）作为 IDS 检测精度指标，在本文中假阳性率与假阴性率分别可称为误报率与漏报率，二者的计算公式为式（2-2）与式（2-3）。

$$FNR = \frac{FN}{TP + FN} \quad (2-2)$$

$$FPR = \frac{FP}{FP + TN} \quad (2-3)$$

表 2-7 检测结果分类表

检测类别	描述
TP (True Positive, 真阳性)	检测结果为异常, 实际也为异常
TN (True Negative, 真阴性)	检测结果为正常, 实际也为正常
FP (False Positive, 假阳性)	检测结果为异常, 但实际为正常
FN (False Negative, 假阴性)	检测结果为正常, 但实际为异常

数据处理性能指入侵检测系统处理数据报文的速度。若设计的检测算法复杂度过高, 以至于超过处理器处理能力, 则会对检测效果造成影响, 甚至影响原有系统的性能。考虑到车内 CAN 通信入侵检测系统的实现环境为嵌入式系统, 其计算资源与存储资源不足, 本身处理大量数据的能力不高, 因此要尽可能在保证检测精度的前提下简化算法。

IDS 还需保证自身的信息安全, 以防止被黑客绕过或禁用, 否则检测出的结果不可信, 同时车内加入的 IDS 系统也不应妨碍原有系统的正常工作, 为确保这两点, 可在原 CAN 网络专门加入一个专用的检测节点, 且需采取专门的安全措施进行安全防护, 其他节点与上层网络不能禁用该节点, 也不能直接对该节点的内存进行读写。本文研究的车内 CAN 通信入侵检测系统主要侧重于优化检测精度的研究, 暂不考虑对于检测系统自身安全与对原系统的影响的研究。

2.5 本章小结

本章在研究车内通信网络架构的基础上, 重点分析了车内 CAN 网络结构与典型的车内 CAN 通信协议——SAE J1939 协议, 分析了协议对数据帧仲裁场的规定以及数据场中的参数格式。在此基础上进行了车内 CAN 总线脆弱性分析, 其脆弱性主要包括四个方面: CAN 本身基于过滤器的接受机制存在脆弱性; 缺乏数据保护机制; 缺乏针对拒绝服务攻击的保护机制; 缺乏数据源认证机制。针对车内 CAN 总线的攻击途径包括无线与有线两种, 具体攻击包括窃取、中断 DoS、伪造及重放。以报文周期与报文数据场内容作为入侵检测算法的检测特征, 并确定了检测算法的评价指标。

第 3 章 基于报文周期特性的自适应入侵检测算法

3.1 引言

如上一章所述，车内 CAN 总线面临的攻击类型主要包括：窃取、中断、DoS、伪造、篡改、重放。对于窃取攻击的检测难度很高，但窃取攻击不会直接对 CAN 通信本身造成影响，且黑客在进行窃取攻击之后往往伴随其他攻击。因此，本章着重研究对于中断、DoS、伪造、重放攻击的检测。

与传统的网络安全入侵检测算法不同，车辆入侵检测算法由于电子设备 ECU 计算能力的限制，需要使用轻量级的入侵检测算法。由于车内绝大多数 CAN 报文都为周期性报文，因此本章主要考虑针对周期性报文的攻击和入侵检测，本章在分析实际车内 CAN 报文周期特性的基础上，提出了一种基于报文周期特性的自适应入侵检测算法。

3.2 正常车内 CAN 报文周期特征获取与分析

有两种方式可获取含有某 ID 报文的周期性特征：通过相邻报文的发送时刻做差或接收时刻做差。一方面，若报文发送节点通过记录 CAN 报文的发送时刻从而获取报文发送的周期特性并进行入侵检测，则需要对所有 ECU 重新编程，且运行检测系统会影响节点的原有功能。车内 CAN 报文数据场无法携带报文发送时间戳，接收节点亦无法获取报文的发送时刻。因此，无法通过检测报文发送时刻以获取报文周期信息。另一方面，报文接收节点通过记录 CAN 报文的接收时刻可获取报文发送的周期特性从而进行入侵检测。考虑到 CAN 报文的接收机制，可在总线上添加一个专用节点用于接收所有报文、记录接收报文的时间并进行检测，这样可避免对原有系统进行大规模更改，同时也不会对总线上的其他节点的正常通信造成影响。因此选用第二种方式获取报文的周期性特征。

实际车内 CAN 总线中报文的周期存在波动，而 CAN 报文的周期变化主要取决于两个因素：总线负载与报文优先级。总线负载越大，报文优先级越低，则报文越有可能进入发送等待状态，从而报文的周期特征变化越大。实际车内 CAN 总线上每秒约 2000 条报文，总线负载较大，因此接收到报文的周期会发生一定变化。下面结合图 3-1 详细分析。

如图 3-1，由节点 A 发送给节点 B 标识符为 ID_j 的报文 M ， M 的发送周期为 T 。理想情况下，A 在 t_{i-1} 、 $t_{i-1}+T$ 、 $t_{i-1}+2T$ 时发送报文 M 。忽略总线上的信

号传输延时，则 B 在 t_{i-1} 、 $t_{i-1}+T$ 、 $t_{i-1}+2T$ 时接收报文。假设在实际情况下发送节点在发送第 $i-1$ 与 i 条报文前总线已被占用而在发送第 $i+1$ 条报文前总线空闲，因此第 $i-1$ 与第 i 条报文的发送时间延后，同理 B 节点接收第 $i-1$ 与第 i 条报文的时间也延后，B 接收报文的时间点分别为 t'_{i-1} 、 t'_i 与 t'_{i+1} 。

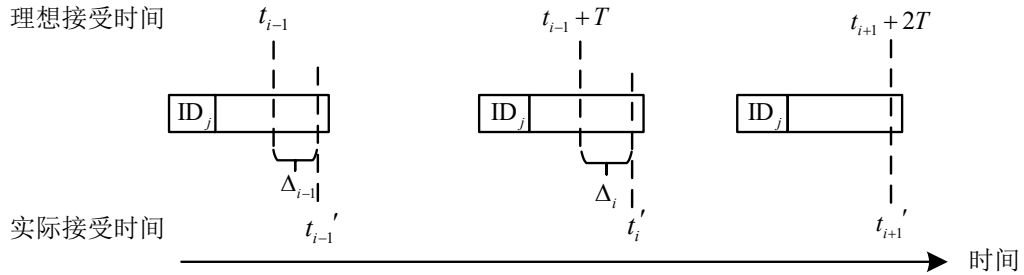


图 3-1 报文理想接收时间与实际接收时间对比图

由于报文发送时刻由主控制器的定时器决定，因此第 $i-1$ 与 $i+1$ 条报文的发送时刻不受前一条报文发送延迟的影响，而仅取决于发送时总线状态，同时忽略报文在总线上的传输延迟，因此报文接收时刻也不受前一条报文接收延迟的影响。因此对于实际接收第 $i-1$ 、 i 与 $i+1$ 条报文的时刻分别为

$$t'_{i-1} = t_{i-1} + \Delta_{i-1} \quad (3-1)$$

$$t'_i = t_i + T + \Delta_i \quad (3-2)$$

$$t'_{i+1} = t_{i+1} + 2T + \Delta_{i+1} \quad (3-3)$$

式中 $\Delta_{i-1}, \Delta_i, \Delta_{i+1}$ 为由于总线被占用而延迟发送的时间，因此 $\Delta_{i-1}, \Delta_i, \Delta_{i+1} \geq 0$ 。对于图 3-1 中， $\Delta_{i+1} = 0$ 。

由式 (3-1) 与式 (3-2)，可得第 i 与 $i-1$ 条报文的接收时间差 s_{i-1} 即式 (3-4)，由式 (3-2) 与式 (3-3)，可得第 $i+1$ 与 i 条报文的接收时间差 s_i 即式 (3-5)。

$$s_{i-1} = t'_i - t'_{i-1} = T + \Delta_i - \Delta_{i-1} \quad (3-4)$$

$$s_i = t'_{i+1} - t'_i = T + \Delta_{i+1} - \Delta_i = T - \Delta_i \quad (3-5)$$

由式 (3-4) 可知， s_{i-1} 的变化取决于 $\Delta_i - \Delta_{i-1}$ ，设 $\Psi_{i-1} = \Delta_i - \Delta_{i-1}$ ， Ψ_{i-1} 表示第 i 与 $i-1$ 条报文的接收时间间隔相对理想周期 T 的偏差。由于 $\Delta_{i-1}, \Delta_i, \Delta_{i+1} \geq 0$ ，因此当 $\Delta_{i-1} = 0$ 且 Δ_i 取最大值 Δ_{\max} 时， Ψ_{i-1} 取最大值 Ψ_{\max} ，因此 s_{i-1} 取最大值；若 $\Delta_i = 0$ 且 Δ_{i-1} 取最大值 Δ_{\max} 时， Ψ_{i-1} 取最小值 Ψ_{\min} ， s_{i-1} 也取最小值。

从实际车内采集得到的四种报文 A、B、C、D 的接收周期随时间变化的散点图分别如图 3-2 中 (a) (b) (c) (d) 所示。报文 A、B、C 的 ID 分别为 0x081、0x2a0 与 0x545，报文周期均为 10ms；而报文 D 的 ID 为 0x220，其为非周期性

报文。本章仅考虑周期性报文。报文 A、B、C 优先级为 0、2、5，其 Ψ_{\max} 分别为 1.4ms、3.2ms、3.8ms，由于存在对称性，报文 A、B、C 的 Ψ_{\min} 分别为 -1.4ms、-3.2ms、-3.8ms，由图可知，报文周期变化幅值大小的主要影响因素即优先级，优先级越低，则该报文的 Ψ_{\max} 越大，且 Ψ_{\min} 越小。

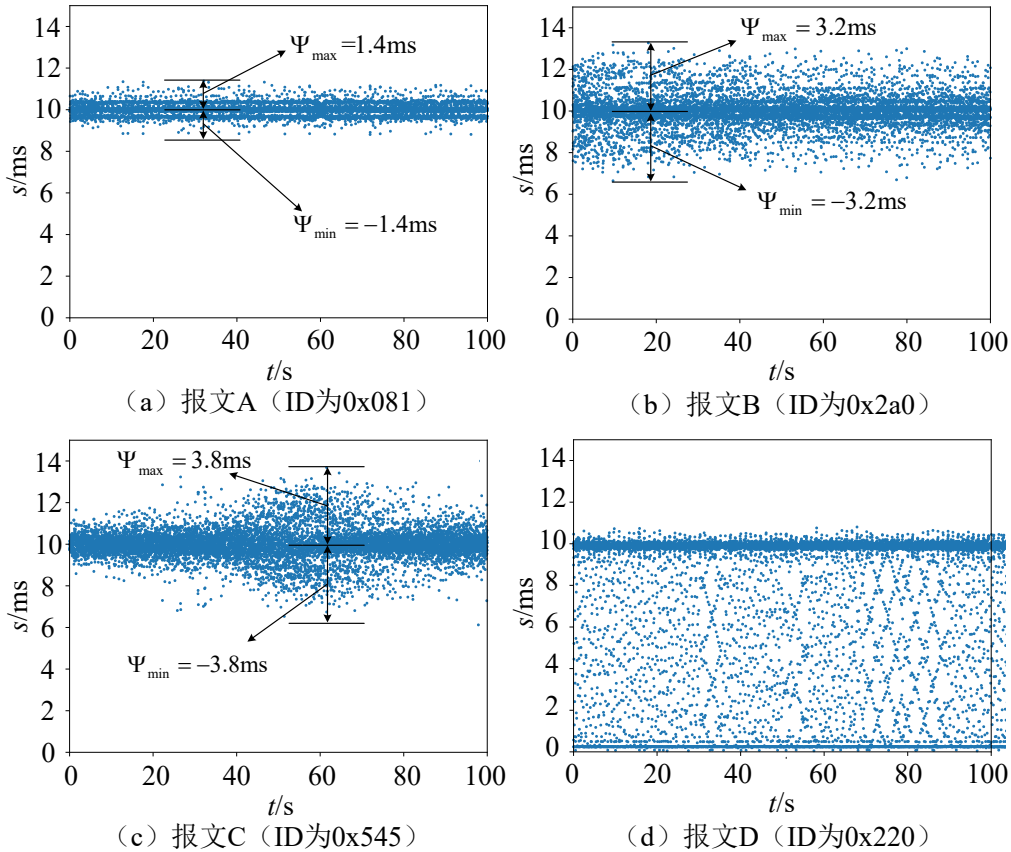


图 3-2 报文周期随时间变化散点图

3.3 攻击特征分析与检测方法

本章着重研究对伪造、DoS、重放与中断攻击的检测，因此以下首先分析这四种攻击下总线上报文时间特性变化情况。为便于分析，按照攻击过程的不同，这四种攻击可分为两种类型：注入型攻击和中断型攻击。其中注入型攻击包括伪造、Dos、重放攻击，其特点是在总线上注入了恶意报文。下面分别分析注入型攻击与中断型攻击的特征及其检测方法。

3.3.1 注入型攻击时间特征分析与检测方法

注入型攻击报文有两种具体形式：一种是注入 CAN 诊断报文，另一种是注

入标准报文。一般来说，当汽车在行驶状态时不应出现诊断报文。若在行驶过程中出现诊断报文，则显然是汽车受到攻击或系统发生故障，因此主要考虑将标准报文作为注入报文。当黑客试图进行报文注入攻击时，即使其发送的报文内容都合法，但总线上通信报文的正常周期会发生变化，以下分析发生注入攻击时，受攻击报文的周期变化情况。

黑客注入伪造报文时，由于车内正常 ECU 仍在不断发送被伪造的正常报文。最终，总线上该 ID 报文速率可以增加到两倍以上，而在进行 DOS 攻击的情况下，恶意节点发送报文的速率通常为原来的 20-100 倍。具体分析如图 3-3 所示，设节点 A 向 B 发送的正常报文 M 的标识符为 ID_j ，报文发送周期为 T ，假设 M 发送前总线空闲，B 在 t_i 与 $t_i + T$ 时收到由 A 发送的正常报文。黑客向总线注入一条标识符为 ID_j 的恶意报文时，该报文在 t_e 时被 B 接收。

由图可得式 (3-6)：

$$\Delta_e = t_e - t_i < \frac{T}{2} \quad (3-6)$$

即 B 接收到报文周期减少为 T 的一半以下，因此可将 $T/2$ 作为检测阈值，即当相邻报文的时间差变为正常报文周期的一半以下时，即可判定发生了注入攻击。

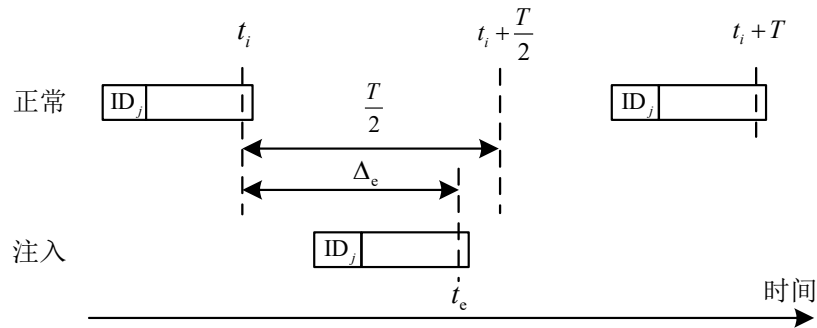


图 3-3 报文理想接收时间与实际接收时间对比图

3.3.2 中断型攻击时间特性分析与检测方法

黑客对某报文 M 进行中断攻击，存在以下两种情况：（1）仅中断若干条报文 M 的发送，之后恢复正常发送；（2）永久中断该报文的发送。无论是哪种情况， M 的接收时间均会大大延长，同时考虑到正常报文周期波动，结合图 3-4，因此可将检测阈值 λ 需满足式 (3-7)：

$$\lambda - T > \Psi_{\max} \quad (3-7)$$

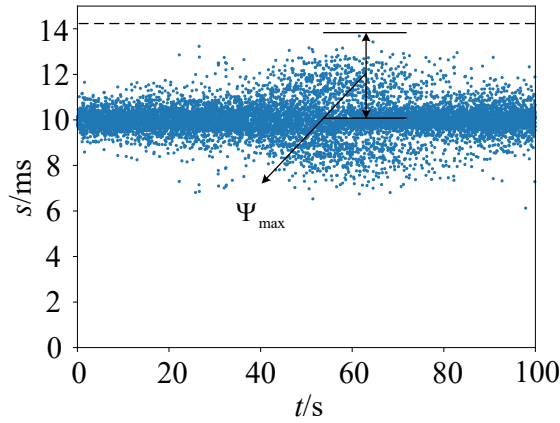


图 3-4 某报文正常状态下周期随时间变化图

3.3.3 算法检测精度分析

由 3.2 节分析可知，实际车内 CAN 报文周期大小会发生变化，在这种情况下，若将检测注入攻击的阈值设为 $T/2$ ，则可能出现异常报文的漏报情况。下面分析报文周期发生变化时检测算法在检测注入攻击时的漏报与误报情况。

(1) 漏报分析

首先分析漏报情况，如图 3-5 所示， t_i 为第 i 次发送报文时无发送延迟情况下的发送时刻， t'_i 为实际发送时间，若在无发送延迟的情况下 B 应在 $t_i + T$ 时接收到第 $i+1$ 条报文，但本例中 $s_i = t_{i+1}' - t'_i$ 取最大值，即 Δ_i 取最小值 0 且 Δ_{i+1} 取最大值 Δ_{\max} 因此有：

$$t_i = t'_i \quad (3-8)$$

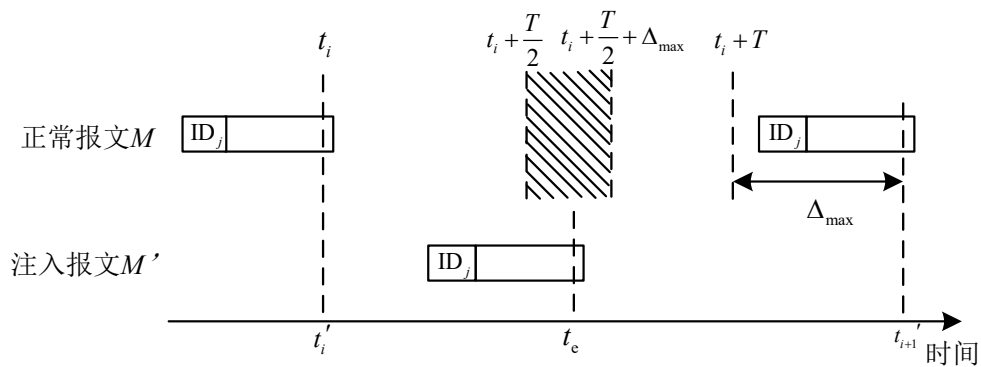


图 3-5 注入攻击发生时报文接收时间图

由于检测值设为 $T/2$ ，而节点 B 收到 M' 的时刻为 t_e ，若 t_e 满足式 (3-9)：

$$t_i + \Delta_{\max} + \frac{T}{2} > t_e > t_i + \frac{T}{2} \quad (3-9)$$

由式 (3-8) 和式 (3-9) 可得式 (3-10) 和式 (3-11):

$$t_e - t_i' = t_e - t_i > \frac{T}{2} \quad (3-10)$$

$$\begin{aligned} t_{i+1}' - t_e &= t_i + T + \Delta_{\max} - t_e \\ &> t_i + T + \Delta_{\max} - t_i - \frac{T}{2} - \Delta_{\max} = \frac{T}{2} \end{aligned} \quad (3-11)$$

由式 (3-10) 和式 (3-11) 可知, 被注入的报文 M' 与其前后相邻的正常报文 M 的时间间隔均大于检测阈值 $T/2$, 因此在这种情况下, 注入攻击无法被检测到, 即当接收时间落在图 3-5 的阴影中时就有可能无法检测到注入型攻击, 因此存在漏报。

(2) 误报分析

下面分析在报文时间间隔最短时是否会发生误报, 若在这种情况下都不发生误报, 则在其他情况下也不会有误报。如图 3-6 所示, 在图中 t_i 与 $t_i + T$ 分别为假设第 i 与 $i+1$ 次无发送延迟情况下的发送时刻, 其中 t_i' 与 $t_i' + T$ 分别为第 i 与 $i+1$ 次发送时的实际发送时刻, $s_i = t_{i+1}' - t_i'$ 。令 s_i 取最小值, 则 Δ_i 取最大值 Δ_{\max} 且 Δ_{i+1} 取最小值 0。

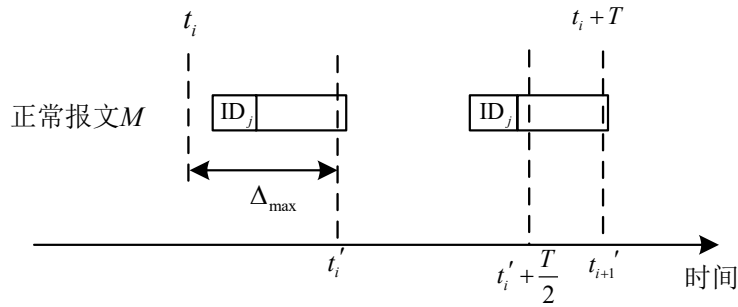


图 3-6 注入攻击发生时报文接收时间图

由于有 $\Delta_i = \Delta_{\max}$ 与 $\Delta_{i+1} = 0$, 因此分别有

$$t_i' = t_i + \Delta_{\max} \quad (3-12)$$

$$t_{i+1}' = t_i + T \quad (3-13)$$

不发生误报的充分必要条件如式 (3-14):

$$t_i' + \frac{T}{2} < t_{i+1}' \quad (3-14)$$

由式 (3-12)、(3-13), 则 (3-14) 可等价为式 (3-15):

$$T > 2\Delta_{\max} \quad (3-15)$$

在实际的车内 CAN 通信网络中, 周期性报文的周期变化量与周期的关系符合式 (3-15), 因此在这种情况下不发生误报。

综上所述可知，若将检测阈值设为 $T/2$ ，那么有可能发生漏报，但不发生误报。若黑客专门针对 Δ_{\max} 较大的报文进行注入攻击，并以特定的时刻发送到总线，则算法漏检率会大大提高，因此需对原算法进行改进。

3.4 基于报文周期特性的自适应入侵检测算法

3.4.1 原有检测阈值的优化

基于 3.4.1 的分析，若检测阈值的最小值 μ_{\min} 为

$$\mu_{\min} = \frac{T + \Delta_{\max} - \Delta_{\min}}{2} = \frac{T + \Psi_{\max}}{2} \quad (3-16)$$

当 $\mu > \mu_{\min}$ 时，即使出现 3.4.1 节中图 3-6 的情况也不会出现漏报。

另一方面，为确保不出现误报，检测阈值也不能无限大，因此需要得到检测阈值的最大值 μ_{\max} 。

与 3.4.1 节中图 3-7 的情况类似，若在阈值为 μ_{\max} 下两报文时间间隔最小时都不出现误报，那么在 $\mu < \mu_{\max}$ 且时间间隔非最小的情况下也必然不会出现误报，因此下面分析两报文时间间隔最小时都不出现误报的阈值 μ 。

如图 3-7 所示， t_i 为报文第 i 次发送时无发送延迟情况下的发送时刻， t'_i 为实际发送时间，若在无发送延迟的情况下 B 应在 $t_i + T$ 时接收到第 $i+1$ 条报文， $s_i = t'_{i+1} - t'_i$ 取最小值，即 Δ_i 取最大值 Δ_{\max} 且 Δ_{i+1} 取最小值 0，因此有

$$s_i = t'_{i+1} - t'_i = t_i + T - t_i - \Delta_{\max} = T - \Delta_{\max} \quad (3-17)$$

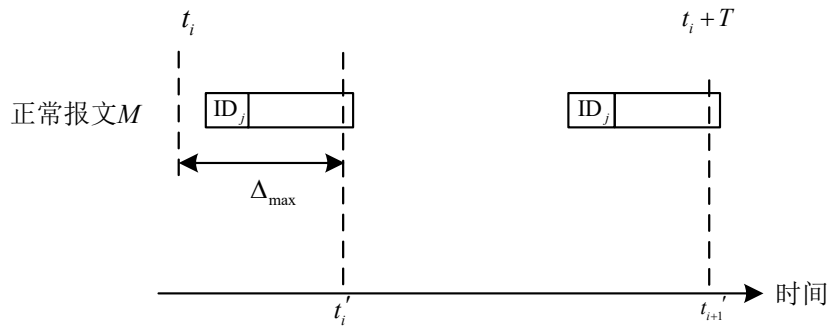


图 3-7 正常状况下报文接收时间图

若不发生误报，需满足

$$T - \Psi_{\max} > \mu \quad (3-18)$$

将 μ 的最大值表示为 μ_{\max} ，则有

$$\mu_{\max} = T - \Psi_{\max} \quad (3-19)$$

由于误报与漏报的情况是分开分析的， μ_{\max} 不一定大于 μ_{\min} ，下面进行分类讨论。

(1) 若 $\mu_{\max} > \mu_{\min}$ ，则由式 (3-16) 与式 (3-19) 可得

$$T - \Psi_{\max} > \frac{T + \Psi_{\max}}{2} \quad (3-20)$$

化简得

$$T > 3\Psi_{\max} \quad (3-21)$$

即满足式 (3-21) 时，将阈值 μ 设为 $\mu_{\max} > \mu > \mu_{\min}$ ，则检测系统不会出现误报与漏报。

图 3-8 为 $T > 3\Psi_{\max}$ 下检测错误率（误报率与漏报率）随检测阈值变化曲线，由式 (3-20) 可知 $T/2 < \mu_{\min}$ ，因此当 $\mu = T/2$ 时会存在漏报，与 3.4.1 中的分析一致；将阈值 μ 设为 $\mu_{\max} > \mu > \mu_{\min}$ ，检测系统既不会有误报也不会有漏报。

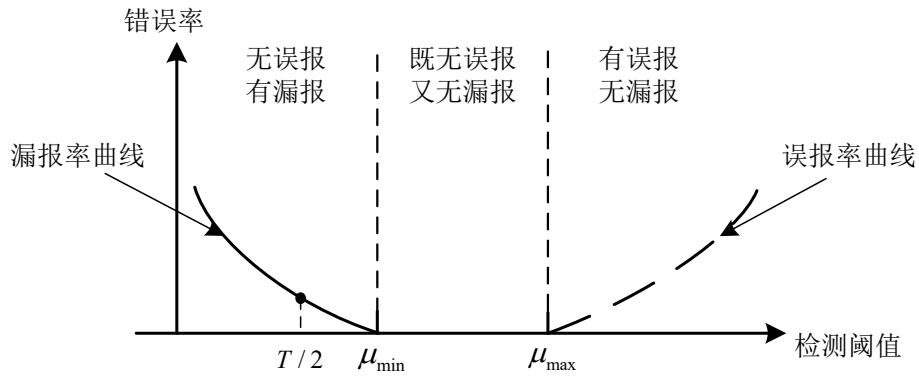


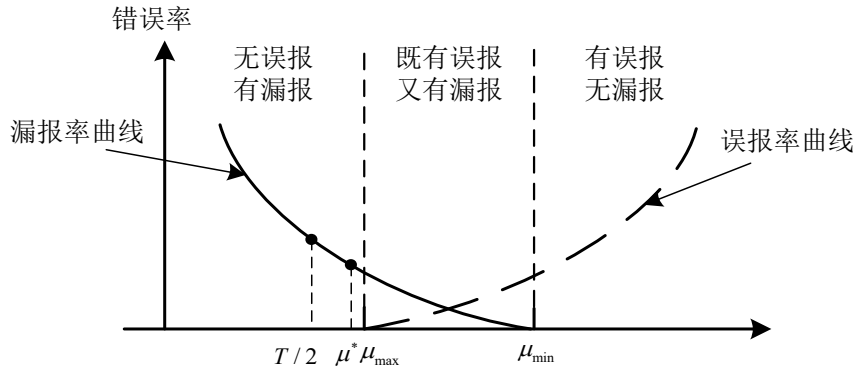
图 3-8 $T > 3\Psi_{\max}$ 时检测错误率随检测阈值变化曲线

(2) 若 $\mu_{\min} > \mu_{\max}$ ，则由式 (3-16) 与式 (3-19) 得

$$T < 3\Psi_{\max} \quad (3-22)$$

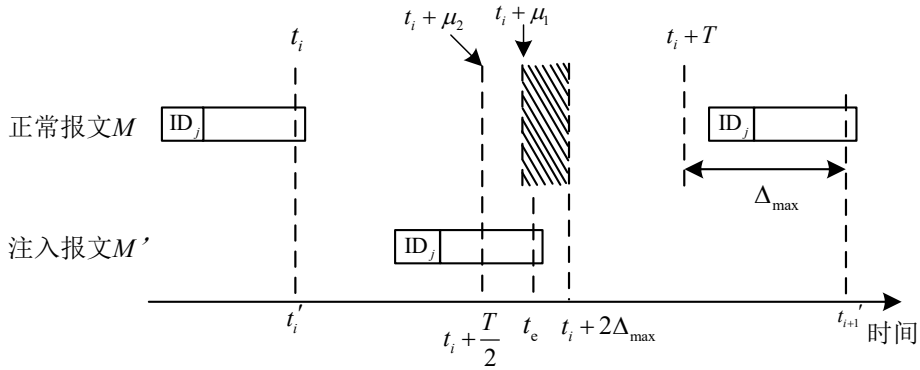
图 3-9 即为 $T < 3\Psi_{\max}$ 下检测错误率（误报率与漏报率）随检测阈值变化曲线，由式 (3-15) 与 (3-19) 可知 $T/2 < \mu_{\max}$ ，因此当 $\mu = T/2$ 时会存在漏报，与 3.4.1 中的分析一致。由图 3-9 可知，若检测阈值取 $\mu < \mu_{\max}$ ，则不会出现误报但可能有漏报；若检测阈值取 $\mu > \mu_{\min}$ ，则不会出现漏报但可能出现误报；若阈值 μ 的取值范围为 $\mu_{\min} > \mu > \mu_{\max}$ ，则检测系统既有误报也有漏报。

考虑到非 DoS 攻击下黑客注入的恶意报文数量远小于正常报文数量，因此选择无误报而存在小概率漏报情况下的检测阈值更合适，因此 μ 可取略小于 μ_{\max} 的值，如图 3-9 中的 μ^* 。


 图 3-9 $T < 3\Psi_{\max}$ 时检测错误率随检测阈值变化曲线

为便于理解，下面对比分析在 $T < 3\Psi_{\max}$ 时将检测阈值分别取 $\mu_1 = \mu_{\max}$ 下与 $\mu_2 = T/2$ 时的漏报率。

如图 3-10 所示， t_i 为报文第 i 次发送时无发送延迟情况下的发送时刻， t'_i 为实际发送时间，若在无发送延迟的情况下 B 应在 $t_i + T$ 时接收到第 $i+1$ 条报文， $s_i = t'_{i+1} - t'_i$ 取最大值，即 Δ_i 取最小值 0 且 Δ_{i+1} 取最大值 Δ_{\max} ，因此有：


 图 3-10 检测阈值分别取 $\mu_1 = \mu_{\max}$ 与 $\mu_2 = T/2$ 时的漏报率分析图

当 $\mu_1 = T + \Psi_{\min} = T - \Delta_{\max}$ ，且有

$$T < 3\Psi_{\max} \quad (3-23)$$

则有

$$t_i + \mu_1 < t_i + 2\Delta_{\max} \quad (3-24)$$

即如图 3-10 中的相对时间所示，当注入报文的接收时间 t_e 满足

$$t_i + \mu_1 < t_e < t_i + 2\Delta_{\max} \quad (3-25)$$

则有式 (3-26)：

$$t'_{i+1} - t_e > t_i + T + \Delta_{\max} - (t_i + 2\Delta_{\max}) = T - \Delta_{\max} = \mu \quad (3-26)$$

又因为：

$$t_e - t_i' > t_i + \mu_1 - t_i = \mu_1 \quad (3-27)$$

由式（3-26）与式（3-27）可知，当注入报文的接收时间 t_e 满足式（3-25）时，检测系统无法检测到注入攻击，检测系统出现漏报。漏报的时间区间为：

$$t_i + 2\Delta_{\max} - (t_i + \mu_1) = 3\Delta_{\max} - T \quad (3-28)$$

而之前当 $\mu_2 = T/2$ 时漏报的时间区间为 Δ_{\max} ，目前车内周期性报文的周期均满足式（3-29）：

$$T > 2\Delta_{\max} \quad (3-29)$$

因此由式（3-29），可得式（3-30）：

$$3\Delta_{\max} - T < \Delta_{\max} \quad (3-30)$$

即将周期检测阈值设为 $\mu_1 = T - \Psi_{\max}$ 的漏检率更低。

综上所述，得出以下结论：

若报文 $T > 3\Psi_{\max}$ ，则将阈值设为 $\mu_{\min} < \mu < \mu_{\max}$ 时检测系统即不会出现误报也不会出现漏报。

若报文 $T < 3\Psi_{\max}$ ，无论阈值取何值检测系统都会出现误报或漏报，而可将阈值 μ 设为略小于 μ_{\max} 的值 μ^* 时检测系统不会出现误报，但有极低的概率出现漏报。

3.4.2 基于报文周期特性的自适应入侵检测算法设计

基于报文周期特性的自适应入侵检测算法主要包括两部分：（1）自适应检测阈值标定算法；（2）检测算法。

（1）自适应检测阈值标定算法

下面以标识符为 ID_j 的报文为例，结合图 3-9 说明自适应检测阈值标定算法流程。首先检测节点接收标识符为 ID_j 报文并记录报文接收时间戳，计算出 n 个接收周期 T_i ，其中 $i=1,2,\dots,n$ ；计算 T_i 的均值 T_{mean} 与最大值 T_{\max} ，令 $\Psi_{\max} = T_{\max} - T_{\text{mean}}$ ；利用由 3.4.2 中所得结论确定注入型攻击检测阈值 μ ；利用 3.3.2 中所得结论确定中断型攻击的检测阈值 λ ；为防止误报，设定检测阈值时需额外加入少量时间裕度。

（2）检测算法

在完成自适应检测阈值的标定后即可进行检测。同样以标识符为 ID_j 的报文为例，结合图 3-10 说明检测算法流程。首先进行初始化 $i=0$ ，记录第 i 条报文的接收时间 t_i ；而后进行判断，若未收到第 $i+1$ 条报文，则直接判定为报文中

断发送，而若接收到第 $i+1$ 条报文，则记录第 $i+1$ 条报文的接收时间 t_{i+1} ；进而计算出第 i 与 $i+1$ 条报文之间的时间 Δ_i ；若 $\Delta_i < \mu$ ，则发出注入攻击警报，否则进入中断检测，进入中断检测后，若 $\Delta_i > \lambda$ ，则发出中断发送警报，接下来进入循环继续检测，直至系统关机。

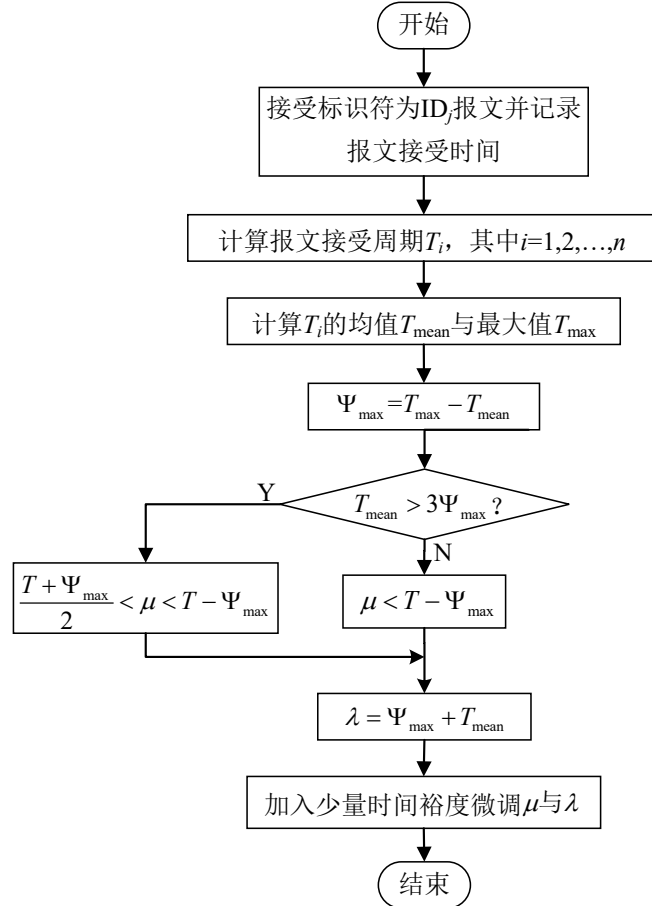


图 3-9 自适应检测阈值标定算法流程图

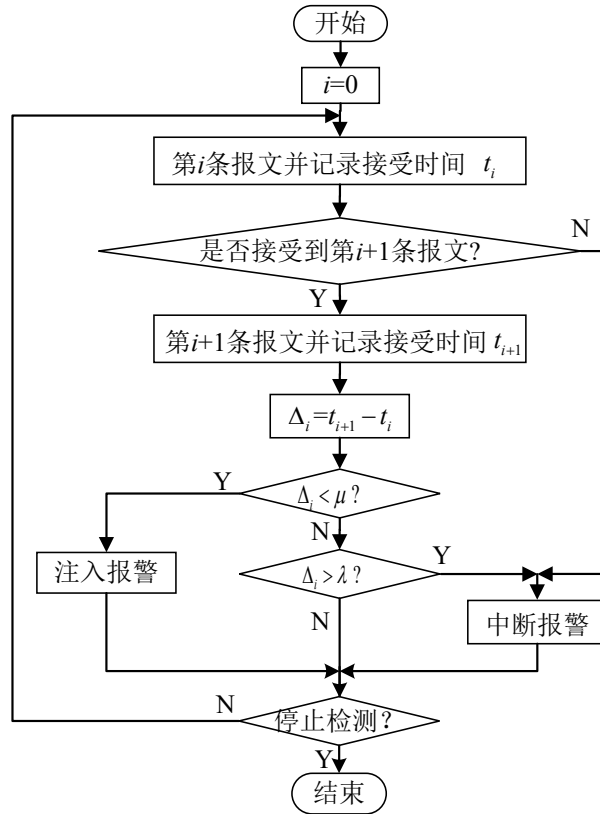


图 3-10 检测算法流程图

3.5 算法试验验证

3.5.1 试验设计

为验证所提算法有效性, 采用三个 CAN 通信节点模拟车内 CAN 通信, 如图 3-11 所示, 各节点以 ST 公司的 STM32F407ZGT6 为主控芯片, 该芯片采用 Cortex M4 内核, 主频为 168MHz, 集成了 CAN 控制器, 数据通过输出引脚发送给开发板的 CAN 收发芯片 TJA1050, 继而将报文发送至总线。

由于该模拟平台节点数较少, 因此总线负载小。为模拟实际车内 CAN 总线较高负载时的报文发送延迟, 在各节点的原有报文发送周期的基础上加入随机数 Δ ($0 < \Delta < 1.9$)。正常情况下 A 以 $(10+\Delta)$ ms 为周期向 B 发送报文 M_1 , 以 $(10+2\Delta)$ ms 为周期向 B 发送报文 M_2 。

(1) 针对注入攻击的检测算法验证

C 为恶意节点, 该节点随机向总线中分别注入 10000 条伪造的报文 M_1' 与 M_2' 。根据所提基于报文周期特性的自适应入侵检测算法, 报文 M_1 的注入攻击检测阈值 μ_1 应满足式 (3-33):

$$5.95\text{ms} = \frac{10+1.9}{2}\text{ms} < \mu_1 < (10-1.9)\text{ms} = 8.1\text{ms} \quad (3-33)$$

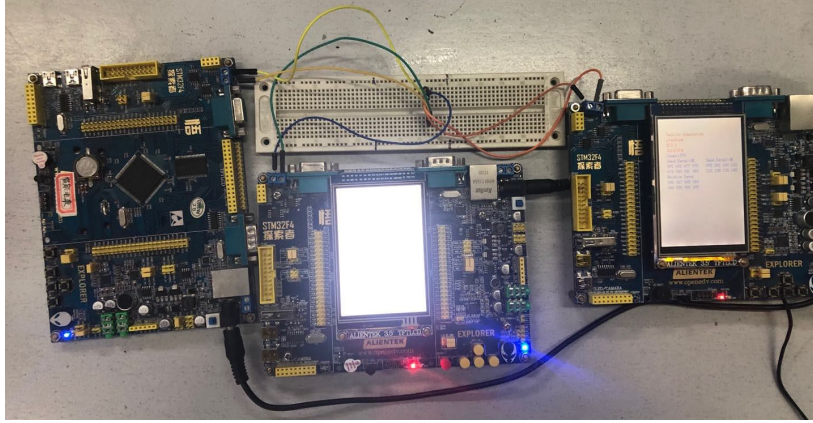


图 3-11 车内 CAN 通信模拟平台

因此将 μ_1 分别设为 5ms、6.5ms、7ms 与 8.5ms，其中 5ms 为不考虑报文周期变化而设的检测阈值，对比不同检测阈值下注入 M_1 时的检测结果。

报文 M_2 的注入攻击检测阈值 μ_2 应满足式 (3-34)：

$$\mu_2 < (10-3.8)\text{ms} = 6.2\text{ms} \quad (3-34)$$

因此将 μ_2 分别设为 5ms、5.8ms、6.5ms，其中 5ms 为不考虑报文周期变化而设的检测阈值，对比不同检测阈值下注入 M_2 时的检测结果。

(2) 针对中断攻击的检测算法验证

不加入 C 节点，仅有 A 与 B 参与通信， M_1 与 M_2 均随机中断发送 10000 次。由之前所提算法，报文 M_1 与 M_2 的中断攻击检测阈值 λ_1 、 λ_2 应满足式 (3-35) 与 (3-36)：

$$\lambda_1 > (10+1.9)\text{ms} = 11.9\text{ms} \quad (3-35)$$

$$\lambda_2 > (10+3.8)\text{ms} = 13.8\text{ms} \quad (3-36)$$

因此将 λ_1 分别设为 11ms、12ms、13ms，对比不同 λ_1 下对 M_1 进行中断攻击的检测效果。因此将 λ_2 分别设为 13ms、14ms、15ms，对比不同 λ_2 下对 M_2 进行中断攻击的检测效果。

3.5.2 试验结果与分析

检测结果如表 3-1 所示，对于注入攻击来说，由于 M_1 的周期 T 满足 $T > 3\Delta_{\max}$ ，将阈值设在 5.95ms 至 8.1ms 区间内不会有误报与漏报；而若将阈值设为 5ms，则出现 45 条漏报的注入报文；若将阈值设为 8.5ms 时，会有 126 条正常报文被

误报。 M_2 的周期 T 满足 $T < 3\Delta_{\max}$ ，将阈值小于 6.2ms 时不会出现误报，但存在漏报，且阈值越小，漏报数越多；若将阈值设为 $T/2$ 即 5ms，则有 67 条漏报报文，漏报较多；若将阈值设为 $T/2$ 即 6.5ms，则有 15 条漏报报文又有 3 条误报报文。因此证明了所提基于报文周期特性的自适应入侵检测算法在检测注入攻击方面比将阈值简单地设为 $T/2$ 而进行检测的算法效果更好。

对中断攻击的进行检测时，将 M_1 的中断检测阈值设为 11ms 时，出现 137 个误报；而将 M_1 中断检测阈值设为 12ms、13ms 时既无误报也无漏报。将 M_2 的中断检测阈值设为 13ms 时，出现 35 个误报；而将 M_2 中断检测阈值设为 14ms、15ms 时既无误报也无漏报，即验证了所提算法对于检测中断攻击的有效性。

表 3-1 注入攻击与中断攻击的检测结果

攻击类型	被攻击报文	T (ms)	Δ_{\max} (ms)	阈值 (ms)	误报数	漏报数
注入攻击	M_1	10	1.9	5	0	45
		10	1.9	6.5	0	0
		10	1.9	7	0	0
		10	1.9	8.5	126	0
	M_2	10	3.8	5	0	67
		10	3.8	5.8	0	21
		10	3.8	6.5	3	15
中断攻击	M_1	10	1.9	11	137	0
		10	1.9	12	0	0
		10	1.9	13	0	0
	M_2	10	3.8	13	35	0
		10	3.8	14	0	0
		10	3.8	15	0	0

3.6 本章小结

本章分析了实际车内 CAN 报文周期特性，提出了一种针对车内网络的基于报文周期的入侵检测算法；由于车内 CAN 总线存在一定的通信负载，且 CAN 报文独特的优先级机制和发送等待机制，造成报文周期存在一定大小的变化，因此分析了周期变化对入侵检测算法检测精度的影响；提出了基于报文周期特性的自适应入侵检测算法，针对周期变化幅度的不同给出了自适应检测阈值的确定规则。试验表明该算法可检测到绝大多数注入攻击与所有中断攻击。

第 4 章 基于报文数据场特征的入侵检测算法

4.1 引言

第 3 章所提算法仅以报文周期作为检测特征，可检测中断攻击与注入型攻击，但该算法在检测注入型攻击中的伪造与重放攻击时仍存在局限性，主要原因有以下两点：（1）检测算法在对周期波动较大的报文进行检测时存在漏报的可能；（2）若黑客在进行伪造或重放攻击时先使某正常 ECU 停止向总线发送报文，而后使另一节点立即以与正常报文相同的周期向总线发送报文，在这种情况下之前所提入侵检测算法也可能无法检测到。

本章将报文数据场内容作为另一种检测特征进行检测。首先根据报文数据场的汉明距离的大小，将报文分为三类，并提出基于汉明距离的入侵检测算法。而后在此基础上进行改进，将报文内容与相邻报文之间的变化同时作为检测特征，即 DACHE 特征，继而提出基于 DACHE 特征的入侵检测算法。采用 BP 神经网络模型进行训练与测试，针对算法易陷入局部最优、收敛速度慢等问题对检测算法进行优化。

4.2 基于数据场汉明距离的入侵检测算法

4.2.1 检测原理

在 CAN 总线中，相同 ID 的两个连续报文数据场之间的汉明距离可用式(4-1)表示：

$$D(m_t, m_{t+1}) = \sum_{i=0}^{63} m_t^i \oplus m_{t+1}^i \quad (4-1)$$

其中 m_t 为第 t 条报文的数据场， m_t^i 为数据场中的第 i 个二进制位； m_{t+1} 为第 $t+1$ 条报文的数据场， m_{t+1}^i 为数据场中的第 i 个二进制位。为便于叙述，下文将 $D(m_t, m_{t+1})$ 简称为汉明距离。

检测思想：在正常情况下，对于每个 ID 的报文而言，由于报文数据场内容的有序性，汉明距离的值无论如何变化都不会超出一个区间，将其称为正常情况下的汉明范围。检测系统可通过学习得到正常的汉明范围并将其作为检测阈值。若注入报文的数据场与具有相同 ID 的相邻报文的数据场之间的汉明距离超出正常汉明范围，则系统检测到攻击。

基于此思想，首先基于公开数据集建立车内部分 ID 报文的正常汉明范围。分别计算数据集中 17 个 ID 的报文的汉明距离与汉明范围，结果如表 4-1 所示。图 4-1 为各 ID 报文汉明范围盒型图，其中 x 轴为报文 ID 序号， y 轴为汉明范围。

表 4-1 报文汉明距离统计表

序号	报文 ID	汉明距离	序号	报文 ID	汉明距离
1	0x517	0	10	0x18f	[0,1,2,...,8]
2	0x370	3	11	0x329	[0,1,2,...,9]
3	0x59b	3	12	0x440	[2,3,4,...,11]
4	0x382	[1,2]	13	0x165	[2,4,6,...,26]
5	0x2b0	[2,4,6,...,10]	14	0x260	[2,3,4,...,16]
6	0x2a0	[2,3,4,...,7]	15	0x316	[0,1,2,...,20]
7	0x4f0	[1,2,3,...,7]	16	0x080	[2,3,4,...,25]
8	0x43f	[0,1,2,...,7]	17	0x545	[0,1,2,...,24]

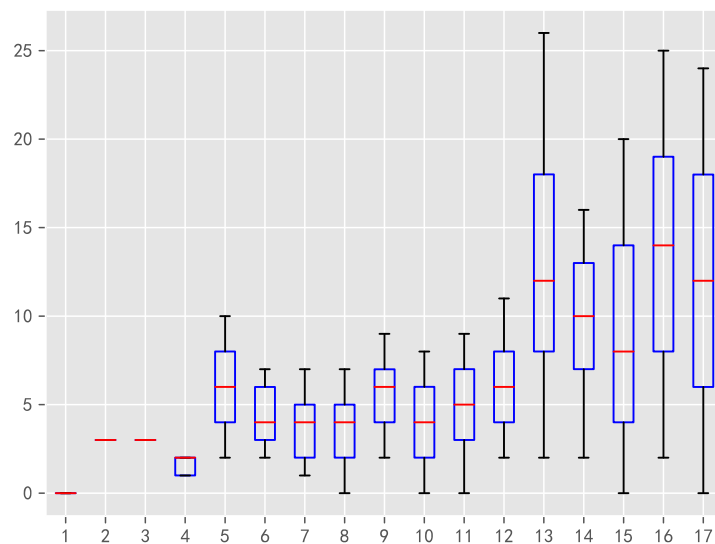


图 4-1 报文汉明范围盒形图

由图 4-1 得到的汉明范围的大小，可将报文分为三类：

- (1) I 类报文，即汉明距离始终不变的报文，其最小汉明距离和最大汉明距离相等，汉明范围为 0，如表 4-1 中 ID 为 0x517、0x370 以及 0x59b 的报文；
- (2) II 类报文。最大和最小汉明距离之间的差别（汉明范围）较小的报文，如表 4-1 中 ID 为 0x2b0 至 0x440 的报文；
- (3) III 类报文。最大和最小汉明距离之间的差别（汉明范围）较大的报文，如表 4-1 中 ID 为 0x165 至 0x545 的报文。

同时由表 1 注意到 ID 为 0x2b0、0x165 的报文的汉明距离均为偶数，因此可在针对这两种 ID 的报文增加判断标准，即如果检测到报文的汉明距离处于正

常的汉明范围内，但其汉明距离不为偶数，这样的报文也视为攻击报文。

4.2.2 检测准确率分析与改进

分析对于伪造和重放攻击的检测效果。由第二章的分析可知，伪造攻击包括 Fuzzing 攻击、伪造畸形报文与伪造诊断报文三类，而车辆正常行驶时 CAN 总线上不应出现诊断报文，对于伪造诊断报文的检测很简单，因此下面主要分析对 Fuzzing 攻击、伪造畸形报文与重放攻击的检测效果。

首先分析针对 Fuzzy 攻击的检测效果。由图 4-1，汉明范围最大的两类报文即 ID 为 0x165、0x545 的报文，其汉明范围为 25，远小于数据场长度 64，在对这两类 ID 的报文进行 Fuzzy 攻击时，由于 Fuzzy 攻击报文数据场内容具有随机性，因此攻击报文的汉明距离会超出检测阈值，会较易检测到这种攻击。而多数报文的汉明范围都小于这两类报文，因此理论上该种检测方法能够很好地检测 Fuzzy 攻击。

其次分析该算法针对伪造畸形报文攻击的检测效果。对于汉明范围为 0 的 I 类报文，只要恶意报文内容有任何改动都可以被检测到，而对于 II、III 类报文，其汉明范围越大，伪造的畸形报文就越有可能检测不到，因此该检测方法对检测畸形报文的效果有限。对于 II、III 类报文而言，改进的方式是通过建立正常报文数据场内容库，检测时将采集到的报文数据场内容与正常报文库进行对比。由于伪造的畸形报文内容包含非协议规定的异常内容，而这些异常内容是正常报文库中不存在的，因此可通过这种方式检出伪造的畸形报文。该方法的不足在于：若通过研究汽车协议以制定正常报文数据场内容库，则需要对车内 CAN 应用层协议进行详细分析；且许多汽车厂商都采用本公司的私有应用层协议，因此需要针对各个公司的各种车型建立正常报文库，所需工作量大。因此可选择利用机器学习算法对正常特征进行学习即可实现正常与异常的分类。

最后分析针对重放攻击的检测效果。首先对于 I 类报文而言，对其进行重放不会对车内 CAN 通信与 ECU 的正常工作造成任何不良影响，因此也无必要对其进行检测。当黑客对 II、III 类报文进行重放时，检测到的报文汉明距离可能超过正常汉明范围，也有可能不会超过正常汉明范围，且汉明范围越大的报文越容易被漏检。因此该检测方法对重放攻击的检测效果有限。

当汽车正常工作并处于某种特定状态时，车内 CAN 报文数据场的变化是有规律的，即当前报文的数据场与上一条 ID 相同报文的数据场对应位的变化存在规律。黑客在进行重放攻击时，虽然该报文内容合法，但重放报文可能不符合正常情况下 ID 相同的相邻报文数据场的变化规律，因此可利用重放攻击的这个特征设计入侵检测算法。例如：黑客通过重放攻击来实施速度欺骗，当前实际

车速为 30km/h，若黑客向总线中注入的重放报文指示车速为 100km/h，虽然该重放报文合法，但由于报文内容超出了正常的变化量，因此可被检测到。

综上所述，对于原有检测算法的改进方法是首先设定两个检测特征：（1）报文数据场内容；（2）被检测报文相对上一次发送的相同 ID 报文的数据场对应的变化量。将这两个特征输入到机器学习模型中进行训练即可实现入侵检测。

4.3 基于 DACHE 特征的入侵检测算法

4.3.1 DACHE 特征

在 4.2.2 分析改进方法的基础上，提出基于 DACHE (data field and change of data field) 特征的入侵检测算法，即将报文数据场及其变化量作为检测特征。检测节点收到报文后提取数据场内容并转化为二进制位，考虑到绝大多数车内 CAN 报文的数据场长度都为 8 字节，本章将这类报文作为检测对象，因此得到 64bit 的数据 M_1 。将该数据场 64 位与上一个检测节点收到的 ID 相同的报文 64 位数据场进行异或运算得到的 64 位数据 M_2 作为报文数据场变化量。将 M_1 与 M_2 进行拼接可得 128 位的特征数据，此即为 DACHE 特征，如图 4-2 即为 DACHE 特征的生成过程。

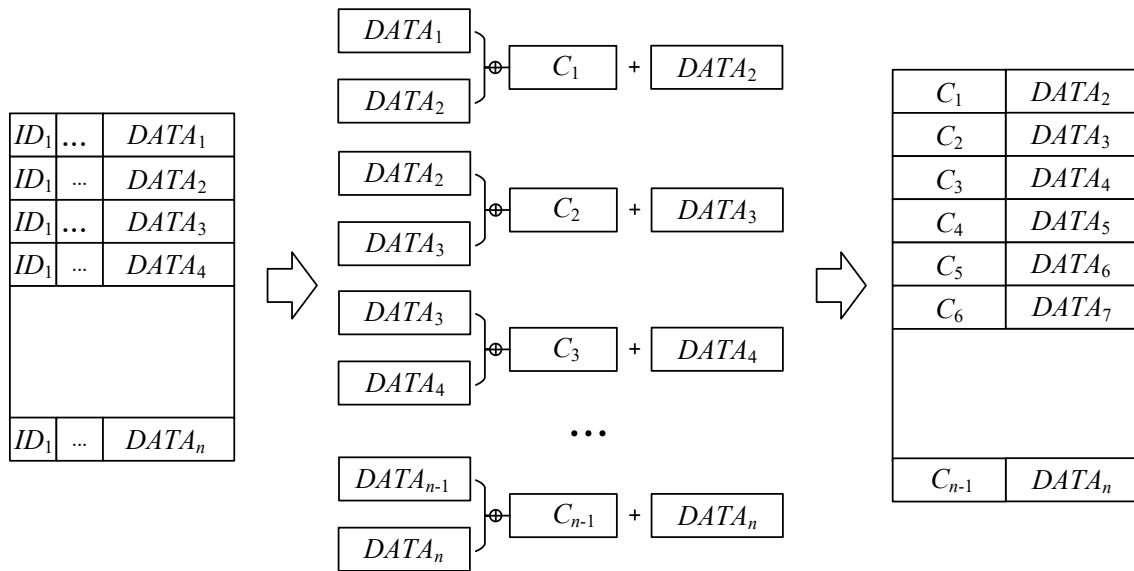


图 4-2 DACHE 特征的生成过程

图中有

$$C_i = DATA_i \oplus DATA_{i+1} \quad (4-2)$$

4.3.2 分类模型

人工神经网络具有很强的自学习、容错与联想记忆能力，并具有高度的并行性，可很好地满足入侵检测算法的功能需要。构建神经网络模型需重点考虑网络拓扑结构、神经元特征以及学习规则。BP 神经网络是目前应用最为广泛的人工神经网络模型之一，其结构简单、分类能力与模式识别能力强，且能以任意精度逼近任何连续函数。因此选择 BP 神经网络（Back Propagation Neural Network, BPNN）^[42]为入侵检测算法的分类模型。

BPNN 是一种多层前馈神经网络，模型中各神经元的权值与偏置由误差反向传播算法不断迭代求得，其拓扑结构如图 4-3 所示，BPNN 的结构主要包括输入层、隐含层与输出层，其中隐含层可为一层或多层。相邻两层间的神经元之间为全连接形式，且每层神经元仅接收上层神经元的输出信号，且每层神经元的输出信号都只影响下层神经元的输入，并且同层神经元之间无连接。

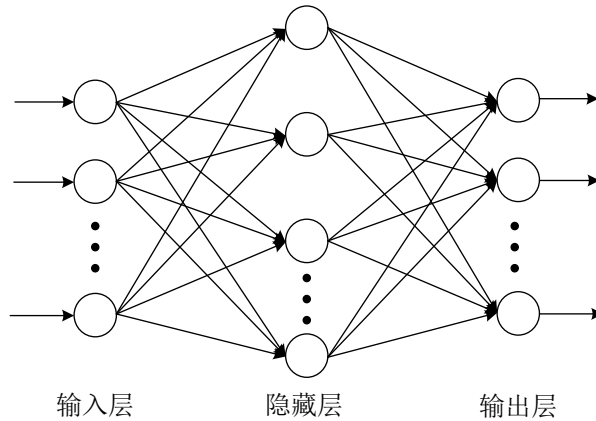


图 4-3 BPNN 结构

BP 算法的流程如图 4-4 所示，每进行一次迭代都包括两个过程：即输入数据的前向传播以及误差的反向传播。且算法每进行一次反向传播都会对神经网络中的权值与偏置进行一次调整。并且每次前向传播后都会计算网络预测值与期望值之间的误差，若误差大小超过预先设定的阈值，则进行误差反向传播以更新网络参数，直到误差小于阈值，算法结束。

（1）前向传播

隐含层第 j 个节点的输出通过式（4-3）来计算：

$$H_j = f\left(\sum_{i=1}^n \omega_{ij}x_i + a_j\right) \quad (4-3)$$

式中 n 表示输入层的节点数量， ω_{ij} 表示隐藏层第 j 个节点对于输入层第 i 个节

点输入值的权值大小， a_j 为隐藏层第 j 个节点的偏置， $f(\cdot)$ 为激活函数。

利用式（4-4）以求得输出层第 k 个节点的输出值：

$$O_k = \sum_{j=1}^l \omega_{jk} H_j + b_k \quad (4-4)$$

式中 ω_{jk} 表示输出层第 k 个节点对于隐藏层第 j 个节点输出值的权值大小， b_k 为输出层第 k 个节点的偏置。

利用式（4-5）计算模型预测值与期望值的误差 E 。

$$E = \frac{1}{2} \sum_{k=1}^m (O_k - Y_k)^2 = \frac{1}{2} \sum_{k=1}^m e_k^2 \quad (4-5)$$

式中 Y_k 为期望输出值， m 为输出神经元节点数量，若 E 的大小不满足误差要求时，即进行误差反向传播。

（2）反向传播

由式（4-3）至式（4-5）可知，神经网络的输出误差 E 是关于网络参数即权值与偏置的函数，若要减少误差 E 就需要调整网络参数。误差反向传播即是将误差 E 作为输入值，由网络输出层到输入层进行反向计算，从而更新权值与偏置以达到减少 E 的目标。计算思路是利用梯度下降法反向计算每层的参数增量，参数的变化方向为误差 E 的负梯度方向，这样即可使误差 E 沿着其负梯度方向不断减少。

神经网络权值与偏置的更新公式分别见式（4-6）与式（4-7）。

$$\begin{cases} \omega_{ij} = \omega_{ij} + \eta H_j (1 - H_j) x_i \sum_{k=1}^m \omega_{jk} e_k \\ \omega_{jk} = \omega_{jk} + \eta H_j e_k \end{cases} \quad (4-6)$$

$$\begin{cases} a_j = a_j + \eta H_j (1 - H_j) \sum_{k=1}^m \omega_{jk} e_k \\ b_k = b_k + \eta e_k \end{cases} \quad (4-7)$$

式中 η 为学习速率。

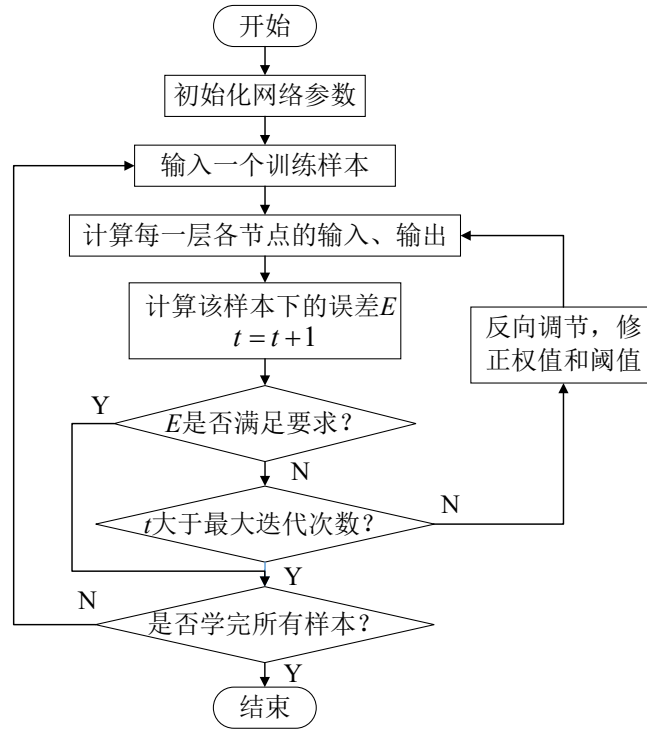


图 4-4 BP 算法流程图

4.4 算法实现与仿真

4.4.1 仿真环境与数据集

仿真所用计算机的 CPU 型号为 Intel Core i5-8265，主频 3.4GHz，内存 8G，操作系统为 Windows10，运用 PyCharm 开发环境，编程语言为 Python，采用谷歌第二代深度学习框架 TensorFlow 搭建神经网络并基于 Tensorboard 实现模型可视化。Tensorflow 是基于数据流编程的数学系统，其灵活性和可延展性让用户可以在各种平台上展开计算，广泛用于各种机器学习算法的编程实现。

数据集采用公开数据集^[43]，该数据集由 Eunbi Seo 等人基于现代索纳塔 YF 采集与构建，其中包括车内正常报文集、伪造的 Gear 报文集与伪造的 RPM 报文集，数据集中对正常报文和伪造的报文做了标记。数据集中的每条数据记录有 12 维特征，主要分为五大类，分别为报文的接收时间、报文 ID、报文长度、数据场内容以及正常或异常标签。下面对伪造的 RPM 报文集进行预处理与检测。

4.4.2 数据预处理

数据预处理包括数据选择与特征计算两步。

(1) 数据选择

首先需对伪造的 RPM 报文集进行选择。该报文集中包含一定时间段内车内 CAN 总线上的所有报文，因此首先需提取所有 RPM 报文记录，RPM 报文的 ID 为 0x043f，因此首先从伪造的 RPM 报文集中提取报文 ID 为 0x043f 的报文，提取后的报文共 897620 条，其中正常报文 739792 条，异常报文 157828 条。

由于提取后的数据集中的正常样本数与异常样本数之比很大，而神经网络的训练对于类别不均衡样本十分敏感，样本因此可能导致 BPNN 分类器对正常样本过拟合，使分类器性能显著下降甚至完全丧失分类能力。可观察到数据集中的正常报文有连续的很多条数据场内容相同的报文，因此可对数据集中的正常样本进行欠采样。

考虑到数据集样本为时间序列样本，且对于其中的某一条报文，要生成 DACHE 特征就需要其数据场与前一条报文数据场进行异或运算，因此欠采样的思想为将其中 G 的部分无效样本舍弃，如图 4-5 所示。

M_1	$DATA_1$	R
M_2	$DATA_2$	T
M_3	$DATA_3$	R
M_4	$DATA_3$	R
M_5	$DATA_3$	R
M_6	$DATA_3$	R
M_7	$DATA_3$	R
M_8	$DATA_4$	R
M_9	$DATA_4$	R
M_{10}	$DATA_5$	T

} 冗余样本

图 4-5 样本集中的冗余样本选择

图 4-6 中有连续的 10 个样本的数据场序号分别为 M_1, M_2, \dots, M_{10} 。R 与 T 分别为样本正常与异常标签， $DATA_1$ 为数据场中的数据，其中第 3 至 7 条报文内容相同均为 $DATA_3$ ，第 8、9 条报文内容均为 $DATA_4$ ，因此有式 (4-8)：

$$M_4 \oplus M_3 = M_5 \oplus M_4 = M_6 \oplus M_5 = M_7 \oplus M_6 \quad (4-8)$$

即 M_5, M_5, M_6 与 M_7 生成的 DACHE 特征均相同，可将 M_5, M_6 与 M_7 作为冗余样本删去，虽然 M_3 与 M_4 相同，但由于：

$$M_3 \oplus M_2 \neq M_4 \oplus M_3 \quad (4-9)$$

即 M_3 与 M_4 生成的 DACHE 特征不同，因此 M_3 与 M_4 作为有效样本保留。基于此思想得到的欠采样算法流程如图 4-6 所示，在遇到若干连续数据场相同的报文时，由图 4-5 可知，需要保留开始的至少两条报文，因此在流程图窗口阈值需 $m \geq 2$ ，在本文中 m 取 2。对原有数据进行欠采样后得 200000 条报

文，其中异常报文 157828 条，正常报文 42172 条。

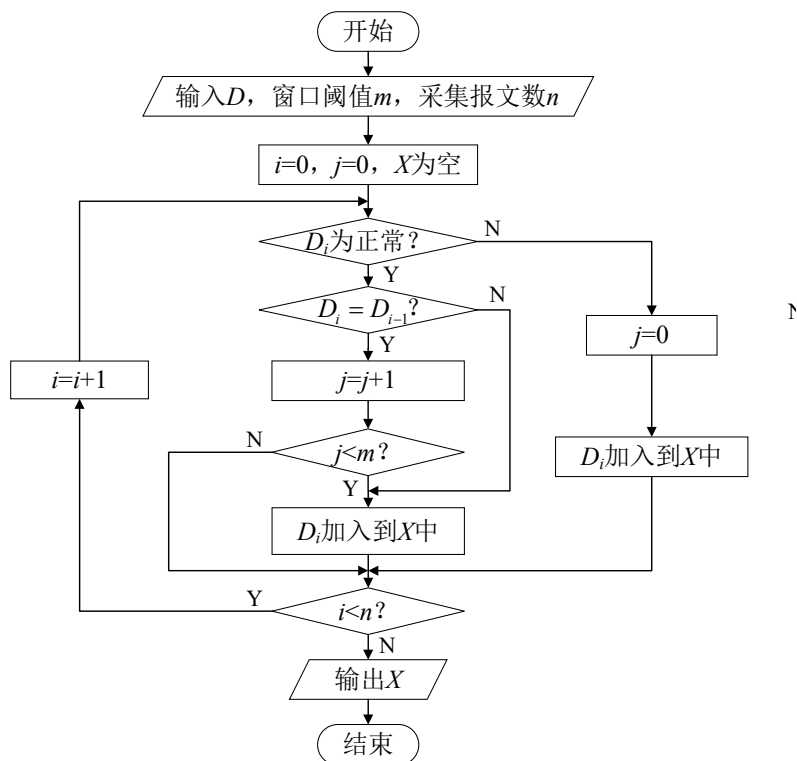


图 4-6 欠采样算法流程图

(2) 特征计算

对每条选择后的数据计算 4.3 中所述的 DACHE 特征。由于数据量很大，在进行特征计算时耗时过长，因此可通过调用 Python 中的 Multiprocessing 模块加速计算过程。在 Python 中虽然可采用 Threading 模块进行多线程运算，但其本质上仍然是一个时间段内只处理一个线程，并且不同线程间切换需要一定时间，因此通过多线程方式对数据进行分块处理的时间可能比正常情况下更长。多核计算可将任务分配给处理器的每一个核心，每一个核拥有单独的运算空间与运算能力，并且每个核心都能同时处理分块后的数据。因此采用多核运算可避免多线程运算的劣势，有效加快特征计算的速度。

在本实验中的计算机为 4 核，因此将 200000 条数据平均分为 4 份，并分配给每个核心进行处理。对比未采用多核运算与采用多核运算耗时，前者耗时 102283s，加速后运算耗时 27644s。

4.4.3 模型建立与仿真

构造基于 BPNN 的入侵检测分类模型，模型由输入层、2 个隐藏层以及输出层组成，其中输入层神经元 128 个、两个隐藏层神经元各 100 个，输出层神

经元 2 个，学习率设为 0.01。训练集与测试集数据比为 7:3。

激活函数选择 ReLU 即线性整流函数，该函数在梯度下降和反向传播的计算时较为高效，同时可避免在误差反向计算时出现梯度消失或梯度爆炸的问题，ReLU 函数见式 (4-8)。上层神经网络输入值 x 进入某个神经元后，该神经元输出值即为 $\max(0, \omega^T x + b)$ ，并将该值传入下一神经元。

$$f(x) = \max(0, x) \quad (4-10)$$

目标函数选择 Softmax 函数，函数公式为式 (4-11) 取负对数得到的损失函数，如式 (4-12) 所示

$$f_j(z) = \frac{e^{z_j}}{\sum_k e^{z_k}} \quad (4-11)$$

$$y_i = -\log\left(e^{z_i} / \sum_{l=1}^p e^{z_l}\right) = -z_i + \log \sum_{l=1}^p e^{z_l} \quad (4-12)$$

式中 s_j 为 j 上的得分， y_i 为真实的类别。

采用该分类模型对训练集进行训练，训练时采用 SGD 优化算法。得到入侵检测分类模型并进行训练。将训练得到的 BPNN 模型对测试集进行测试。采用 Tensorboard 将模型可视化，如图 4-7 所示。

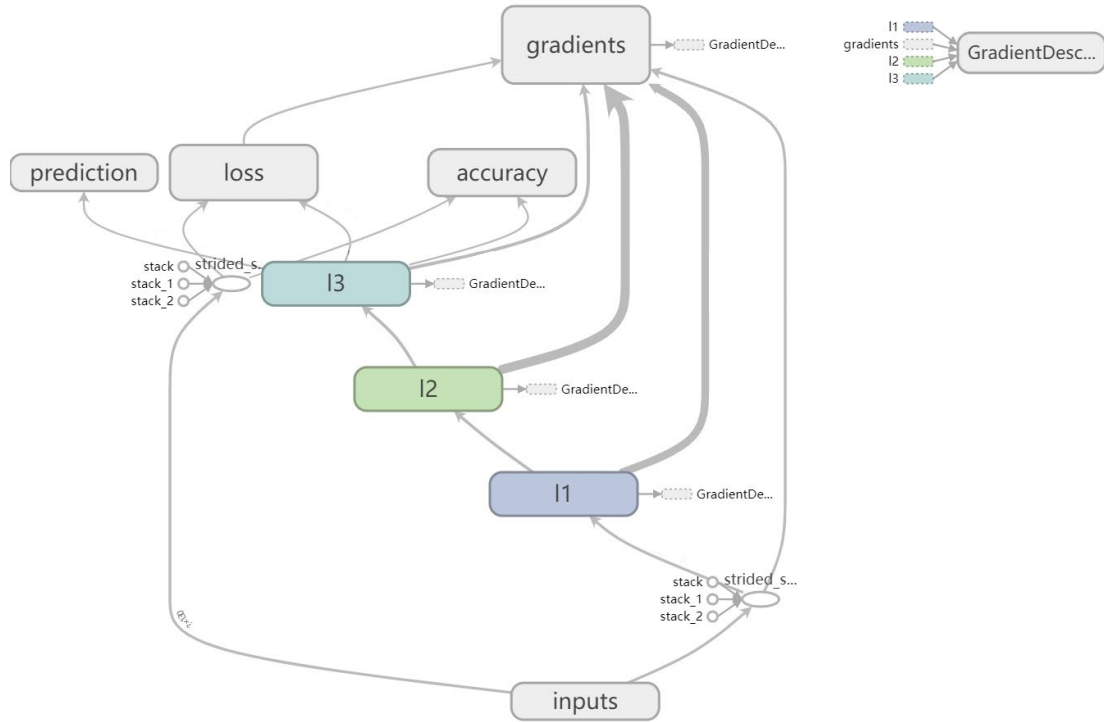


图 4-7 模型结构图

将数据场变化、数据场以及 DACHE 特征分别记为特征 1、特征 2 与特征 3，将三种特征分别输入到模型中进行训练并测试，得到不同输入特征下的准确率随训练次数变化曲线图 4-8。

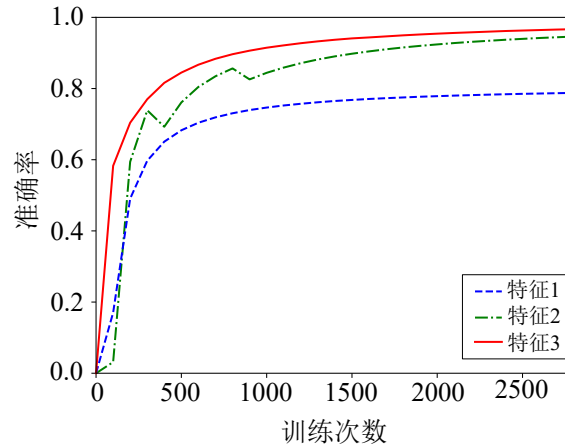


图 4-8 不同输入特征下分类准确率对比图

由图 4-8 可知，由特征 1、2、3 在迭代 2900 次时得到的模型分类准确率为 78.85%、94.8%、96.79%，且特征 3 下的模型训练的收敛速度更快，特征 2 在迭代时其准确率有波动，因此 DACHE 特征为有效的检测特征。

4.5 算法优化与仿真

4.5.1 算法存在的不足

采用 DACHE 特征进行分类后分类准确率可达 96.97%，但通过多次仿真发现有时 BPNN 在训练时会陷入局部最优解，并且有时训练时准确率曲线会出现震荡现象。原因有以下：

BPNN 本质上是一种基于局部搜索的模型，其网络参数的调整方向为局部的梯度方向，因此有陷入局部最优的可能^[44]。并且，BP 神经网络对网络中各神经元参数的初始值较为敏感，采用不同的初始值训练后得到的结果可能大相径庭。

BPNN 在训练时模型参数收敛速度较慢。反向传播本质上采用的是梯度下降算法，而目标函数往往比较复杂，网络参数在训练过程中存在震荡现象，使得训练效率变低。同时，若神经元的输出值接近 1 或 0，那么在训练过程中误差函数的梯度值可能会很小，这也使训练速度下降。并且在训练 BPNN 时，采用静态步长可能使误差函数越过最优解，也会导致震荡现象的发生。

在这种情况下模型分类准确度会大打折扣，因此考虑对原算法进行优化，

解决算法陷入局部最优以及震荡的问题，进一步提高检测准确度。

4.5.2 算法优化

原有的随机梯度下降(SGD)算法每次更新时都仅考虑了一个样本数据点，这样虽加快了训练速度，但同时带来的问题是训练过程中的网络参数值不一定朝着极小值的方向进行更新，这有可能会使训练出现震荡而减慢收敛速度。因此，改进的方法是可使用 MBGD (Mini-Batch Gradient Descent) 算法，该算法每次选取一小部分样本进行计算。适当增加增大单次迭代所用样本数 N 即可提高训练速度， N 越大则误差下降方向越准确，训练过程中网络参数的震荡越小，但 N 过大则易陷入局部最优，本模型中将 N 设为 50。为进一步减少训练震荡，提高收敛速度，还需对算法进一步优化。

(1) 增加动量项修正梯度

对当前梯度下降方向进行修正，在权值与偏置的反向传播公式中加入动量项，动量项表示上一次反向传播过程中的梯度下降方向，其反映了上一次迭代所积累的调整经验。式(4-6)与式(4-7)中隐藏层与输出层权值和偏置公式均可用式(4-13)表示，即

$$\begin{cases} \omega_{i+1} = \omega_i + \eta dW_i \\ b_{i+1} = b_i + \eta db_i \end{cases} \quad (4-13)$$

其中 dW_i 与 db_i 为第 i 次迭代计算隐藏层或输出层的损失函数时进行误差反向传播求得的梯度，加入动量项，即

$$h_i = \beta h_{i-1} + (1 - \beta) dW_i \quad (4-14)$$

$$k_i = \beta k_{i-1} + (1 - \beta) db_i \quad (4-15)$$

其中 h_{i-1} 与 k_{i-1} 分别为损失函数在前 $i-1$ 轮迭代过程中累积的梯度动量，将 i 轮迭代得到的累积梯度动量 h_i 与 k_i 作为更新后的梯度进行权值与偏置更新

$$\omega_{i+1} = \omega_i - \eta h_i \quad (4-16)$$

$$b_{i+1} = b_i - \eta k_i \quad (4-17)$$

在本模型中 β 取 0.9，即表示下一步前进的方向更偏重于历史的下降方向，这样解决 MBGD 优化算法更新幅度摆动大的问题，同时可以使得网络的收敛速度更快。

(2) 自适应改变学习率

考虑到在迭代开始时，由于距离最优解还有一段距离，因此学习率可略大；而在即将到达最优解时，需要将学习率设小些，否则可能迭代时越过最优点。基于此思想，考虑加入自适应学习率。

$$s_i = \gamma s_{i-1} + (1 - \gamma) dW_i^2 \quad (4-18)$$

$$r_i = \gamma r_{i-1} + (1 - \gamma) db_i^2 \quad (4-19)$$

式中 γ 表示梯度累积系数，其表示历史累积梯度对于下次新生成梯度的影响程度， γ 越大，则历史累积梯度对新生成梯度的影响越大。 s_w 与 s_b 分别表示前 $t-1$ 轮迭代过程中损失函数累积的梯度动量，将其作为自适应项对学习率进行调整，即可采用式（4-20）与式（4-21）进行权值与偏置的更新。

$$w_{i+1} = w_i - \frac{\eta}{\sqrt{s_i + \varepsilon}} dW_i \quad (4-20)$$

$$b_{i+1} = b_i - \frac{\eta}{\sqrt{r_i + \varepsilon}} db_i \quad (4-21)$$

为防止分母为零，式中加入了一个很小的数值 ε 来进行平滑。通过自适应地调节学习率大小来修正摆动幅度，可消除摆动幅度大的方向，同时可以使损失函数收敛更快。

将（1）（2）结合起来，即增加动量项修正梯度同时自适应改变学习率可进一步优化训练过程。由式（4-14）、（4-15）得到的修正后的梯度项与式（4-18）、（4-19）得到的自适应步长修正项，即得权值与偏置的更新公式（4-22）、（4-23）

$$w_{i+1} = w_i - \frac{\eta}{\sqrt{s_i + \varepsilon}} h_i \quad (4-22)$$

$$b_{i+1} = b_i - \frac{\eta}{\sqrt{r_i + \varepsilon}} k_i \quad (4-23)$$

式中 $\varepsilon=10^{-8}$ ， $\beta=0.9$ ， $\gamma=0.999$ ，在迭代的初始时刻 $h_0=0$ 、 $k_0=0$ 、 $s_0=0$ 、 $r_0=0$ ，

注意到迭代开始时， $\frac{\eta}{\sqrt{s_i + \varepsilon}}$ 与 $\frac{\eta}{\sqrt{r_i + \varepsilon}}$ 项很大，会导致迭代开始时偏差较大，因此需要进行微调。

$$h_i^c = \frac{h_i}{1 - \beta_i^c} \quad (4-24)$$

$$k_i^c = \frac{k_i}{1 - \beta_i^c} \quad (4-25)$$

$$s_i^c = \frac{s_i}{\sqrt{1 - \gamma_i^c}} \quad (4-26)$$

$$r_i^c = \frac{r_i}{\sqrt{1 - \gamma_i^c}} \quad (4-27)$$

且:

$$\beta_i^c = \begin{cases} \beta & i = 0, 1, 2 \\ 0 & i = 3, 4, \dots \end{cases} \quad (4-28)$$

$$\gamma_i^c = \begin{cases} \gamma & i = 0, 1, 2 \\ 0 & i = 3, 4, \dots \end{cases} \quad (4-29)$$

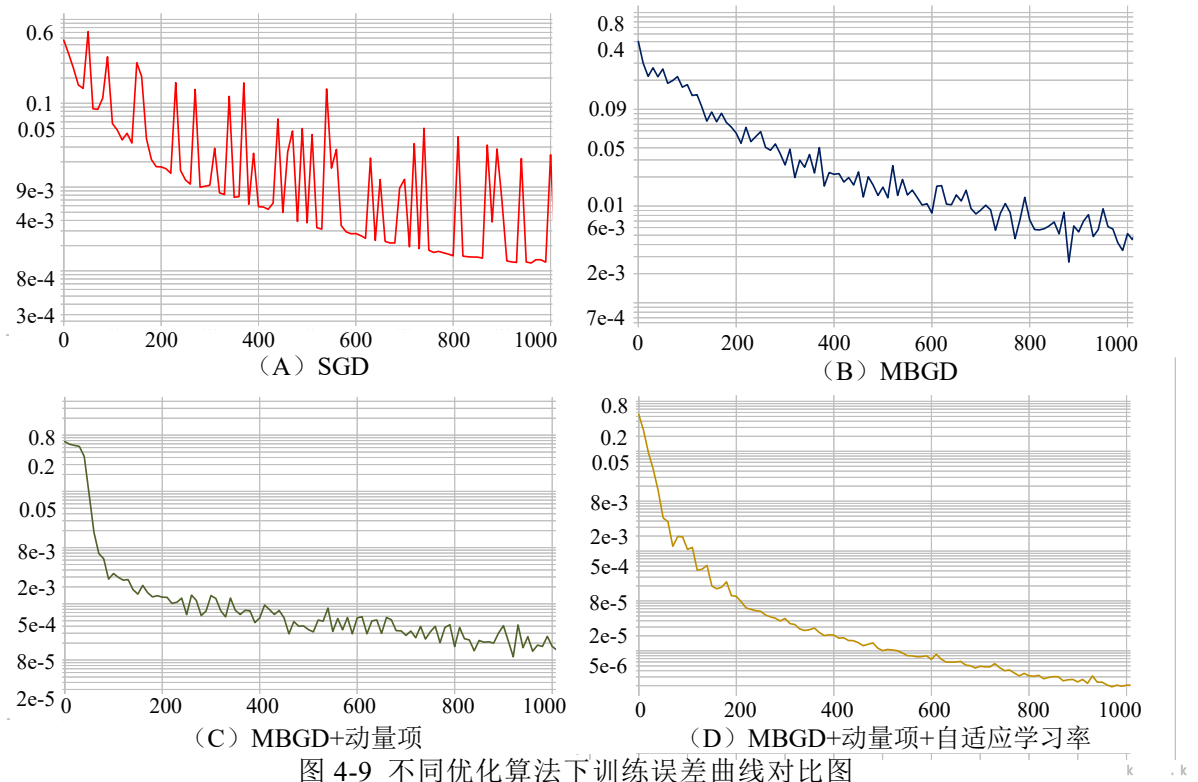
因此权值与偏置公式变为为:

$$w_{i+1} = w_i - \frac{\eta}{\sqrt{s_i^c + \varepsilon}} h_i^c \quad (4-30)$$

$$b_{i+1} = b_i - \frac{\eta}{\sqrt{r_i^c + \varepsilon}} k_i^c \quad (4-31)$$

4.5.3 仿真结果与分析

分别采用 SGD、MBGD、MBGD+动量项、MBGD+动量项+自适应项训练 BP 神经网络，并测试四种情况下分类准确性，四种情况分别记为 A、B、C、D。测试集共 59999 条数据，其中包括 48635 条异常数据与 11364 条正常数据。A、B、C、D 四种情况下模型训练误差曲线如图 4-9 所示。



由图 4-9 可知，SGD 的收敛速度较慢，且存在较大震荡，当采用 MBGD 时震荡减少，但收敛速度与 SGD 相似；C 与 D 在前 200 轮训练时收敛很快，而 D 收敛时更为平稳，且 D 相对 C 的收敛速度更快。

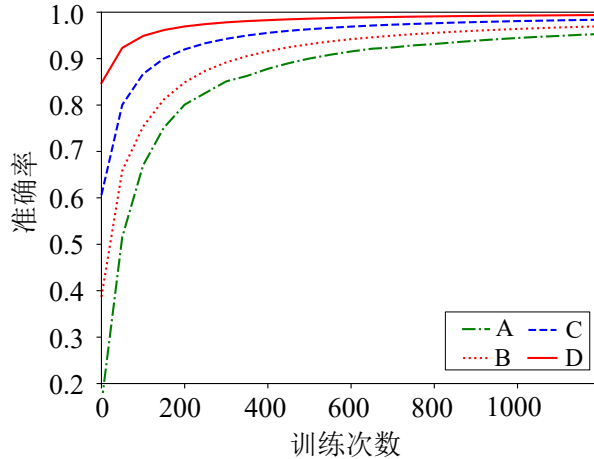


图 4-10 不同优化算法下分类准确率对比图

训练 1500 次后模型的预测情况如表 4-2 所示，A、B、C、D 四种优化算法下训练得到的模型均能够对所有正常数据做出正确分类，而对异常数据均存在一定漏检，其中 A 即 SGD 下的漏检数最多为 715，C 与 D 的漏检数很小且较为相近，漏检数分别为 23 与 34。

表 4-2 检测结果

	TP	FP	TN	TN
A	47920	0	11364	715
B	48474	0	11364	161
C	48612	0	11364	23
D	48601	0	11364	34

综上所述，模型训练时迭代过程中加入动量项调整梯度方向，并加入自适应项调整迭代学习率，收敛速度大大提升，且收敛的稳定性也更好，不易陷入局部最优。

4.6 本章小结

本章以 CAN 报文数据场内容为检测特征，提出了基于汉明距离的入侵检测算法，该算法在检测伪造与重放攻击时存在不足；为弥补该不足，提出了一种新的检测特征 DACHE 特征，将报文内容与前后文变化作为检测特征，采用 BP 神经网络对数据集进行训练；并在此基础上对神经网络训练时的反向传播算法进行了优化，即在迭代过程中加入动量项调整梯度方向，加入自适应项调整迭

代学习率，收敛速度大大提高，且不易陷入局部最优。

总体来说，该算法的优点在于：算法的实现无需提前熟悉车内 CAN 报文格式，仅仅通过获取该品牌车型的 CAN 报文并学习即可实现入侵检测功能，具有很强的通用性；训练速度快；即使注入的恶意报文数量很少，该算法也具有良好的检测效果。仿真结果表明，优化后的神经网络的训练速度大大加快，且训练后的模型具有很好的检测率。

第 5 章 Bus-off 攻击及其检测算法研究

5.1 引言

由于 CAN 通信协议具有独特的错误处理机制,当节点发生错误且错误数量积累达到一定程度时节点会脱离总线。基于此特性,本章提出一种新的针对车内 CAN 通信网络的攻击——Bus-off 攻击,即黑客通过故意频繁触发正常通信节点的错误,使被攻击节点误以为自身出现故障,从而使其脱离总线。该攻击的特征较为特殊,采用第 3、4 章所提算法无法有效检测到该种攻击,因此需要专门研究 Bus-off 攻击的原理、实现过程与攻击特征,并提出针对 Bus-off 攻击的检测方法。

5.2 Bus-off 攻击原理与实现条件

5.2.1 前提假设

为便于分析 Bus-off 攻击,首先做出如下假设:黑客的目的是通过注入尽可能少的报文从而使正常工作的车内 ECU 脱离总线,为达到这个目的,不考虑对手采用其他攻击(如泛洪攻击),原因是泛洪等攻击虽然影响更严重,但实施泛洪时需向总线注入大量报文,易被检测到;黑客可通过各种的攻击面和手段对 ECU 进行远程攻击,从而取得该 ECU 的控制权,并能够利用该 ECU 执行以下恶意操作:通过设置过滤器嗅探到总线上报文,并且可在总线上注入任意带有伪造 ID、DLC 和数据的报文。由于不同汽车制造商甚至不同型号的汽车采用的数据格式都有所不同,因此当黑客对不同的品牌车辆发起攻击时,通过解析报文从而获取报文中的数据格式等信息的难度很高,因此假设黑客不进行报文解析。基于以上假设条件,下面分析 Bus-off 攻击的原理、过程与特征。

5.2.2 攻击发生原理

黑客通过利用 CAN 的以下特性从而实施 Bus-off 攻击:CAN 的错误处理机制会自动使有缺陷或发生错误(即该 ECU 的 $TEC > 255$)的 ECU 隔离,进入到总线脱离状态。具体来说,黑客通过反复注入攻击报文,强迫被攻击 ECU 的 TEC 不断增加,不断欺骗被攻击 ECU 的 CAN 发送器,使其认为自身出现错误,最终触发 CAN 故障从而迫使受攻击节点脱离总线。

5.2.3 攻击发生条件

为便于分析，称攻击节点为 A 节点，受攻击节点为 B 节点，攻击节点发送的伪造报文为 J ，受攻击节点发送的报文为 Z 。要实现这种攻击，需满足以下条件：

- N1: Z 为周期性报文；
- N2: 报文 J 和 Z 的 ID 相同；
- N3: 报文 J 和 Z 同步发送；

N4: 在报文 J 和 Z 的仲裁场外的控制场与数据场至少有一位不同，其中 J 为显性位（0），而 Z 为隐性位（1），且报文 J 和 Z 在该位之前的所有对应位都应相同。

为便于分析 Bus-off 攻击原理，先假设黑客节点发送的报文可以满足以上四个条件，如图 5-1 所示，当总线空闲时，不仅 B 节点在发送报文 Z ，A 节点同时也在发送报文 J ，因此在 CAN 总线上同时有两个节点在发送报文，并且由于 Z 和 J 的报文 ID 相同，因此 Z 和 J 同时赢得总线仲裁，从而在总线上会同时存在报文 Z 和 J 。对于 CAN 通信而言，发送节点在向总线上发送报文的同时也在监听总线上的电平信号，由于攻击报文 J 符合条件 N4，因此在仲裁之后受攻击节点 B 在总线上会监听到与它所传输极性相反的极性。因此，节点 B 会在 A 的强制下触发位错误，从而使其 TEC 增加 8。通过对受攻击节点 B 重复这种攻击，黑客通过节点 A 使 B 的 TEC 不断增加，直到迫使节点 B 进入总线脱离模式并脱离总线。尽管节点 B 在发送其他正常报文时不发生错误，但由于检测到每个错误时 TEC 增加了 8，而每次无错误传输只会使 TEC 减少 1，因此进行重复攻击会使节点 B 的 TEC 迅速增加。

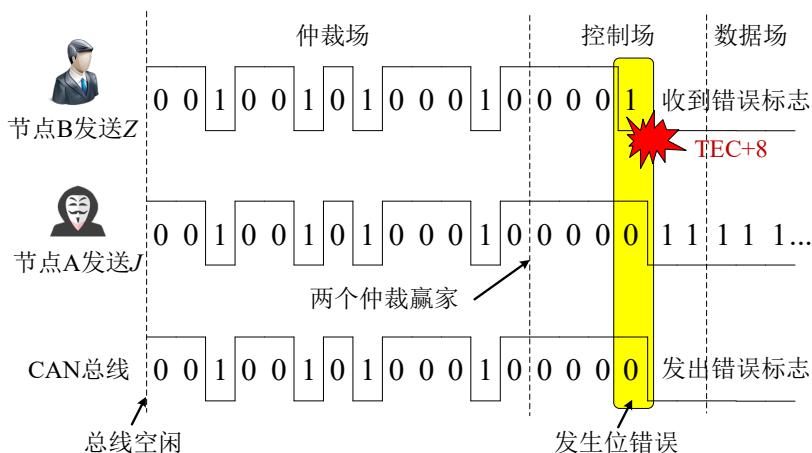


图 5-1 Bus-off 攻击发生条件示意图

5.3 Bus-off 攻击过程分析

CAN 控制器有两种发送模式，分别为自动重发模式与非自动重发模式。在自动重发模式下，若节点在发送报文时出错，则其会在发送错误帧之后立即重新发送之前发送失败的报文；而在非自动重发模式下，若节点在发送报文时出错，则其不会立即重新发送该报文，而是将发送错误的报文丢弃，在下一个发送周期时发送新的报文。

CAN 控制器在自动重发模式与非自动重发模式下，Bus-off 攻击过程略有不同，以下首先分析在自动重发模式下的 Bus-off 攻击。

(1) 自动重发模式下 Bus-off 攻击过程

1) 攻击第一阶段

第一阶段：攻击节点在窃听 CAN 总线上的报文后，将总线上一个正常发送节点作为攻击目标，以下将该正常节点称为节点 B，攻击节点称为 A。而后黑客在节点 B 发送信息的同时通过节点 A 注入攻击报文以增加节点 B 的 TEC。因此，节点 B 发生位错误，并向总线发送错误激活标志，其 TEC 增加 8，如图 5-2 所示。由于错误激活标志由 6 个连续的显性位（000000）组成，因此在攻击节点 A 也会触发一个填充错误或位错误，其 TEC 也增加了 8。在经过错误分隔符和帧间空间 IFS 后，节点 A 和 B 的 CAN 控制器会同时自动重新发送之前发送失败的报文。因此，相同的位错误重复出现，直到两节点同时进入错误认可状态。第一阶段攻击的显著特点是：攻击节点 A 只需注入一条报文，就可强制正常节点 B 成为错误认可状态。

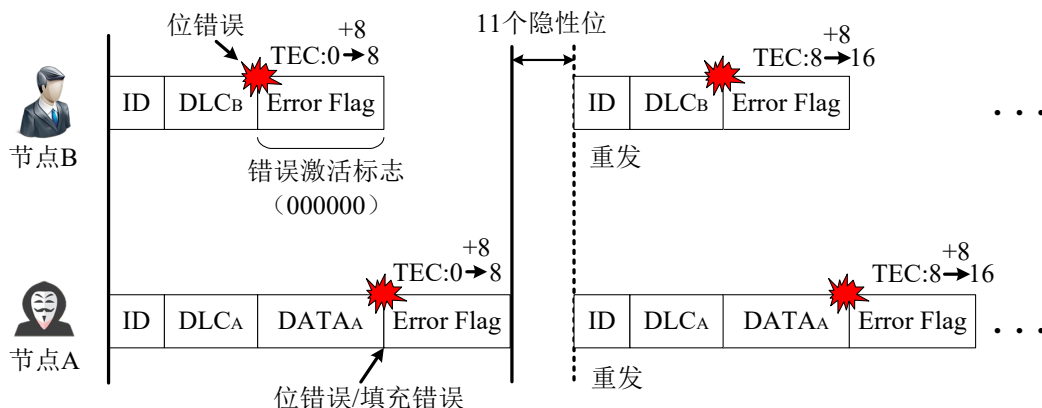


图 5-2 Bus-off 攻击第一阶段

第一阶段至第二阶段的过渡阶段：如图 5-3 所示，在进行 16 次重发操作后，当节点 B 和 A 的 TEC 均为 128 时，二者都进入错误认可状态。同样地，当再次发送报文时，节点 B 仍会发生位错误。然而，由于节点 B 处于错误认可

模式下, 因此 B 会发送一个由六位隐性位 (111111) 组成的错误认可信号。此时, 攻击节点 A 发送报文时不再发生位错误, 因此 A 成功发送报文, 同时节点 B 会不断发送错误信息一直持续到攻击节点 A 发送 EOF 为止。与第一阶段相比, 攻击节点 A 未发生错误, 因此成功传输其帧, 而节点 B 在稍后重新发送成功。综上所述, 由于一个位错误 (+8) 和一个成功的重传 (-1), 被攻击节点的 TEC 先加 8 再减 1 变为 135, 而攻击节点 A 的 TEC 由 128 变为 127。因此, 攻击节点 A 返回到错误激活状态, 而节点 B 仍保持在错误认可状态。

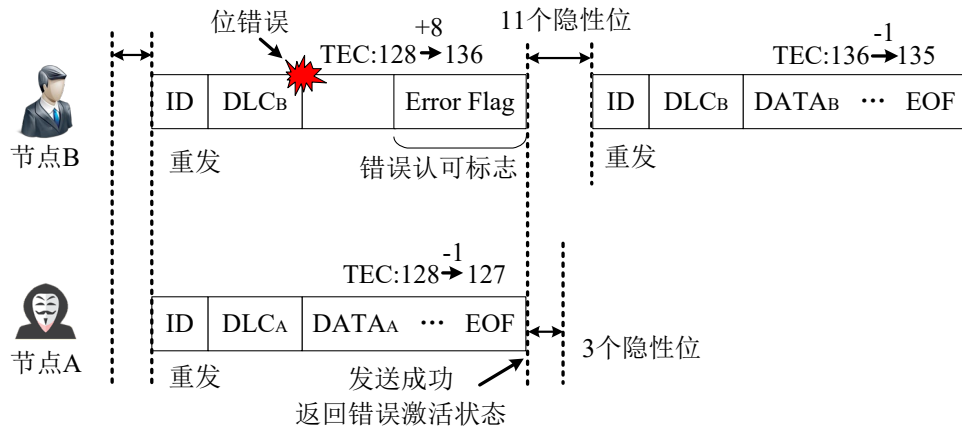


图 5-3 Bus-off 攻击第一与第二阶段之间的过渡阶段

2) Bus-off 攻击第二阶段

第二阶段的攻击过程如图 5-4 所示, 与过渡阶段一样, 节点 B 仍处于错误认可模式, 攻击节点仍可正常发送报文, 因此攻击节点可以进一步降低其 TEC。另一方面, 节点 B 的错误计数器增加了 7 (加 8 减 1), 节点 B 可成功发送其报文, 因此其仍处于错误认可模式。在第二阶段, 攻击节点 A 在节点 B 发送每条目标报文时都重复此过程, 直到节点 B 的 TEC 超过 255, 节点 B 被强制进入总线脱离模式。

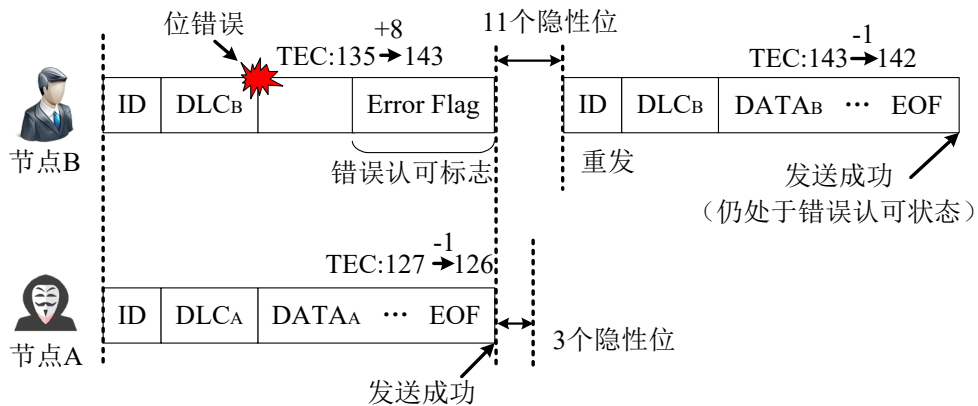


图 5-4 Bus-off 攻击第二阶段

(2) 非自动重发模式下 Bus-off 攻击过程

非自动重发模式下 Bus-off 攻击过程与自动重发模式下攻击过程相似,存在的不同之处包括以下两点:

1) 在攻击的第一阶段,攻击节点 A 需发送 16 次恶意报文才能使节点 B 进入错误认可状态;而在自动重发模式下,节点 A 只需发送一条恶意报文即可使节点 A 进入错误认可状态。

2) 在攻击的第二阶段,被攻击节点 B 进入错误认可状态后遇到错误后不再重发,因此每过一个报文周期 B 节点的 TEC 只加 8 而不减 1;与此同时,攻击节点 A 在发送恶意报文时均不受被攻击节点 B 发出的错误认可标志影响,因此 B 节点的 TEC 每次减 1。

CAN 控制器无论是处于自动重发模式还是非自动重发模式,黑客都能够成功进行 Bus-off 攻击。若黑客以发送高优先级(即低 ID 值)和周期较短报文的节点为攻击对象,那么黑客很可能会断开发送车辆加速或制动相关的重要报文的关键 ECU,从而造成严重的后果。

5.4 Bus-off 攻击实现方法分析

要实现 Bus-off 攻击,黑客注入的攻击报文必须满足之前提出的条件。以下分别对四个条件实现的可行性进行分析。报文 J 与 Z 分别为恶意报文和被攻击的正常报文。

条件 N4 要求:报文 J 和 Z 的控制场或数据场内至少有一位字符不同,其中 J 为显性位(0),而 Z 为隐性位(1),且报文 J 和 Z 在该位之前的所有对应位都应相同。若要满足条件 N4,则黑客需要在嗅探到该 ID 报文后修改控制场或数据场的数据。由于正常报文中的 DLC 一般不会设为全 0,同时,考虑到每个 CAN ID 报文的 DLC 不随时间变化,因此黑客可以在嗅探到该 ID 报文后,获取 DLC 值并进行相应修改 DLC 将其设为全 0,这样,黑客即可满足条件 N4。

条件 N2 要求:报文 J 和 Z 的 ID 相同。若要满足条件 N2,黑客需要预先知道受攻目标报文的 ID。对于 CAN 总线来说,其任一节点只能接受通过其报文过滤器的报文而无法获取其他报文。由于黑客只能读取接受到的报文内容,因此能否满足 N2 取决于攻击 ECU 节点上过滤器的配置状况。

若车内某节点能够接收到总线上绝大多数报文,如用于监测汽车运行状态的状态采集节点,那么黑客即可利用该节点轻易获取大量报文数据并展开攻击从而满足条件 N2。而从安全的角度而言,这种 ECU 的脆弱性非常高。

若车内某节点只接收少部分 ID 的报文,那么黑客仍可通过配置此节点的过滤器使该节点接收所有报文进而实施攻击,同时通过直接修改该节点的过滤器,

黑客也可接收其想要得到的报文，从而满足条件 N2。

条件 N3 要求报文 J 和 Z 同步发送。即总线上攻击报文和受攻击报文之间的差别小于一位。若攻击时机不合适，就不会产生两个仲裁赢家，那么攻击就会失败。由于 CAN 报文的发送周期固定，因此黑客可能利用这一点使攻击报文与受攻击报文达到同步。例如，一旦黑客知道目标报文每 $T\text{ms}$ 发送一次，那么在受攻击报文上次发送开始后经过 $T\text{ms}$ 后，攻击节点可尝试发送其攻击报文。考虑到实际总线环境中的发送周期存在抖动，这种抖动会使受攻击报文偏离攻击报文发送的预设周期 T ，因此利用这种方法无法满足 N3。

从另一方面来说，考虑到 CAN 通信具有的特点：当总线繁忙时，节点会将要发送的报文缓存，而在总线进入空闲状态时，该节点会通过缓存器试图再次发送其报文。黑客可利用该缓存机制实施攻击。

为便于叙述，首先定义：恰好在报文 M 开始传输前完成传输的报文，称为报文 M 的前置报文，其 ID 称为前置报文 ID，前置报文的示意图如图 5-5 所示。

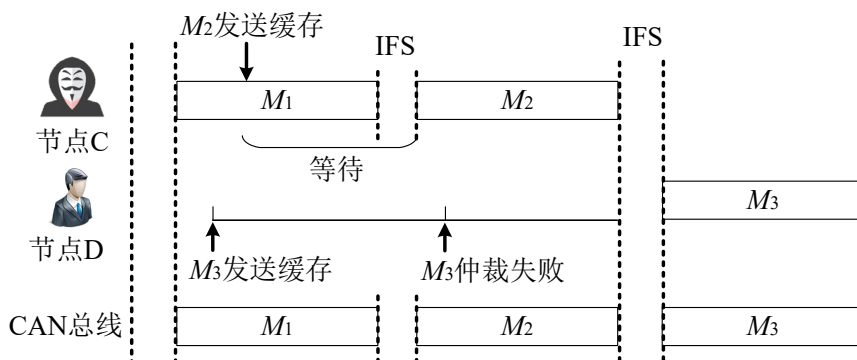


图 5-5 前置报文示意图

在图 5-5 中，节点 C 发送 M_1 、 M_2 报文，节点 D 发送 M_3 报文，其中 M_3 报文优先级最低。若这些报文按照图中时间进行发送并排队，那么 M_2 、 M_3 报文的前置 ID 分别为 M_1 与 M_2 的报文 ID。三条报文在总线上仅相隔 3 位 IFS。当两节点按照图中时刻发送报文时，由于仲裁与缓存机制，节点 C 和 D 的 CAN 控制器分别对 M_2 和 M_3 进行缓存，直到它们的前置报文在总线上传输结束为止。由于 CAN 报文的优先级和周期不变，因此这意味着一个特定 ID 的报文在发送后总是会跟随另一特定 ID 的报文，即对于指定的 ID，其拥有特定前置报文 ID。若 M_1 和 M_2 的发送周期为 M_3 的整数倍，例如， M_1 与 M_2 的发送周期为 5ms ， M_3 为 10ms ，那么 M_2 总是 M_3 的前置报文，因此，无论信号如何抖动，都可预测 M_3 的开始传输时间，即前置报文传输完成之后经过三个位后开始发送 M_3 。

在图中，黑客要攻击 M_3 报文，首先对 CAN 总线进行监听从而知晓 M_3 的前置报文为 M_2 ，也可知晓 M_2 的前置报文为 M_1 。黑客可在接收到 M_1 或 M_2 其中

任意一条时缓存 M_3 的攻击报文 M_3' 。而后，当 M_2 传输完成后，正常节点 D 发送报文 M_3 ，与此同时攻击节点的 CAN 控制器也会同时发送攻击报文 M_3' ，从而满足条件 N3。类似地，当攻击节点收到 M_1 时，也可通过缓冲其 M_2 对应的攻击报文 M_2' 来攻击正常报文 M_2 。若受攻击报文 M_3 的前置报文 ID 唯一，那么黑客可重复以上攻击，从而使攻击节点的 TEC 值持续增加。

即使受攻击报文无前置报文，那么黑客也可伪造一个前置报文以使时间同步，从而成功发起攻击。

如图 5-6 所示，受攻击节点 B 周期性地发送报文 Z，而报文 Z 无前置报文，因此，黑客可在 Z 传输之前，通过攻击节点依次向总线注入高于 Z 优先级的报文 P 和与 Z 优先级相同的攻击报文 G。在这种情况下，由于仲裁机制，正常报文 Z 会在 P 报文传输完成后进行发送从而与攻击报文 G 同时在总线上传输。综上所述，黑客通过伪造 P 作为 G 的前置报文，从而使攻击报文与受攻击报文同步传输。

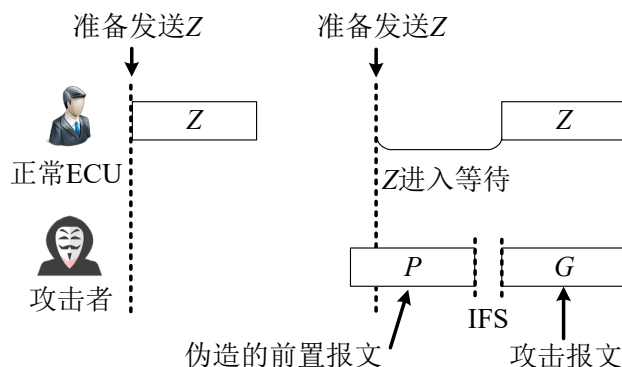


图 5-6 伪造前置报文以实现攻击报文与受攻击报文的同步传输示意图

因此，即使受攻击报文无前置 ID 报文，黑客也可通过伪造一个前置报文进而满足条件 N4。攻击实现的基本思路如下：黑客首先通过监听受攻击报文并获取其发送周期；通过设置攻击节点，使其在接收到受攻击报文后延时一段时间 t 后发送伪造的前置报文和攻击报文，使得攻击报文和受攻击报文同时发送到总线上继而触发受攻击节点错误。这种攻击方式成功的关键在于必须在受攻击报文发送之前刚好注入伪造报文，也就是得到 t 的值，下面对 t 的取值进行分析。

虽然大部分车内报文预设的发送周期固定，但由于发送节点时钟漂移、任务调度、执行时间等变化，使得实际的发送周期存在随机抖动，如图 5-7 所示。受攻击报文 Z 预设的发送周期为 T ，其理想的发送时刻为 t_{ori}^i ，由于存在抖动 J ，因此其实际发送时间可能是 t_{ori}^i 、 $t_{ori}^i + J$ 或 $t_{ori}^i - J$ 时发送。

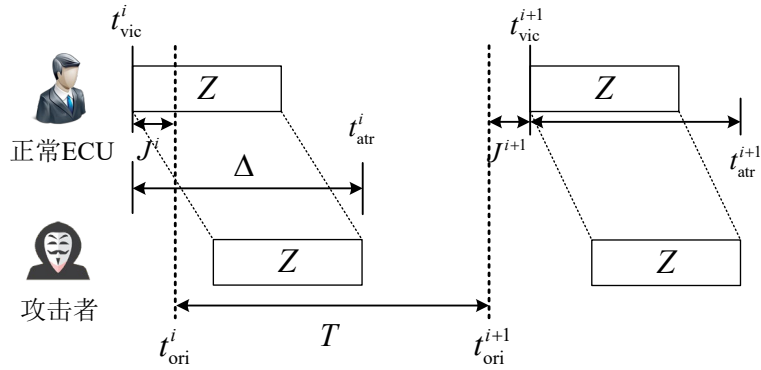


图 5-7 正常报文发送与恶意节点接收时间关系示意图

如图 5-7 所示， t_{vic}^i 与 t_{vic}^{i+1} 分别为恶意节点接收受攻击报文 Z 的时刻，其中 i 为发送序列索引，由于在发送和接收存在一定延迟，因此攻击节点在 Δ 后收到报文 Z ，由于在正常情况下受攻击报文中的数据场字节数为常数，因此 Δ 为常数。对于抖动 J ，由热噪声的随机性与中心极限定理可知，多个不相关噪声源的复合效应趋近于高斯分布。因此图 5-7 中的随机变量 J^i 和 J^{i+1} 符合如下分布： $J \sim N(0, \sigma_v^2)$ 。因此在图 5-7 中，攻击节点收到被攻击报文的时刻 t_{atr}^i 与 t_{atr}^{i+1} 分别为式 (5-1) 与式 (5-2) 所示：

$$t_{atr}^i = t_{vic}^i + \Delta = t_{ori}^i + J^i + \Delta \quad (5-1)$$

$$t_{atr}^{i+1} = t_{vic}^{i+1} + \Delta = t_{ori}^i + T + J^{i+1} + \Delta \quad (5-2)$$

由图 5-7 可知，式中 $J^i < 0$ 且 $J^{i+1} > 0$ ，由式 (5-1) 与 (5-2) 得式 (5-3)：

$$t_{vic}^{i+1} = t_{atr}^i - \Delta - J^i + T + J^{i+1} = t_{atr}^i - \Delta + T + J^* \quad (5-3)$$

式中 $J^* \sim N(0, 2\sigma_v^2)$ ，在式中 J^* 是唯一的随机变量，而式中的其他变量可被测得。

因此式 (5-3) 表明，黑客可获得受攻击报文的下次发送时间的近似估计。

黑客不仅要在受攻击报文发送之前开始发送其伪造的前置报文，而且还需保证在受攻击报文在准备发送时，总线上的伪造的前置报文还未发送完成。因此，黑客必须满足以下三个条件：

$$t_{asm} < \min(t_{vic}^{i+1}) = t_{atr}^i + T - \Delta + \min(J^*) \quad (5-4)$$

$$t_{asm} + N > \max(t_{vic}^{i+1}) = t_{atr}^i + T - \Delta + \max(J^*) \quad (5-5)$$

$$N = kC > \max(J^*) - \min(J^*) \quad (5-6)$$

式中 t_{asm} 为攻击节点开始向总线发送伪造前置报文时间， N 表示发送伪造前置报文占用总线的总时间， k 为伪造前置报文的数量， C 为每个前置报文占用总线时间。由于随机变量 J^* 有界，其边界可近似为 $|\max(J^*)| = |\min(J^*)| = \Pi\sqrt{2}\sigma_v$ ， σ_v 可测得， Π 为置信区间参数。若 Π 分别取 3 与 4，则随机变量 J^* 的置信度为 99.73%

与 99.99%。因此， $t = t_{\text{atr}}^i - \Delta + T - 2\sqrt{2}\Pi\sigma_v$ ，攻击节点在收到受攻击报文 Z 之后 t 时刻之前开始发送伪造前置报文，这样即可满足式 (5-4)。

攻击节点还需满足式 (5-6)，即伪造的前置报文占用总线时间大于 $\max(J^*) - \min(J^*) = 2\sqrt{2}\Pi\sigma_v$ ，可设置伪造前置报文的数量以满足该条件。为防止攻击被发现，黑客需注入最少的前置报文以实现该攻击，因此需最大化 C 从而使 k 最小化。为最大化 C ，黑客可利用 CAN 的位填充机制，使伪造前置报文的填充位尽可能多，因此黑客可将数据场数据设为全 0，在这种情况下即有：

$$C_{\max}^* = [8L + 44 + (8L/5)] / S_{\text{bus}} = 121.4 / S_{\text{bus}} \quad (5-7)$$

式中 C_{\max}^* 即为 C 的最大值， L 为数据场字节数，44 为除数据场外数据帧的位数， S_{bus} 为总线波特率。例如，若 CAN 总线波特率为 500kbps，黑客注入一个伪造前置报文可占用总线至少 0.2428ms。因此，受攻击节点发送报文的时刻即使存在抖动，那么攻击节点只要注入足够数量的伪造前置报文即可屏蔽抖动带来的影响，实现攻击所需要 k 的最小值 k_{\min} 如下式 (5-8)：

$$k_{\min} = \left\lceil \frac{\max(J^*) - \min(J^*)}{C_{\max}^*} \right\rceil = \left\lceil \frac{2\sqrt{2}\Pi\sigma_v S_{\text{bus}}}{121.4} \right\rceil \quad (5-8)$$

例如，若 $\sigma_v = 0.025\text{ms}$ ，CAN 总线波特率为 500kbps，当 Π 取 3 时，取 $k_{\min} = [0.874] = 1$ ，即注入一条伪造前置报文即可满足攻击条件。

若黑客只使用一条伪造前置报文进行攻击，则该伪造前置报文 ID 可随意选择，并且可在每次发送后都重新设置新的报文 ID 以增强攻击的迷惑性。若黑客要使用两条伪造的前置报文进行攻击，那么第一条伪造前置报文 ID 也可随意选择，但应使用比受攻击报文优先级更高的 ID 注入第二条伪造前置报文。

5.5 Bus-off 攻击试验

5.5.1 试验设计

试验采用三个 CAN 通信节点模拟车内 ECU 通信，通信节点主控芯片仍采用 STM32F407ZGT6。通信波特率为 500kHz，通信网络结构图如图 5-8 所示。STM32 中 CAN 总线发送出现错误后有两种处理方式，分别为自动重传与禁止自动重传模式。若设为禁止自动重传模式，则发送操作只会执行一次，如果发送操作失败了，硬件都不会再自动重新发送该报文；而在自动重传模式下，如果发送操作失败了，那么硬件会尝试自动重新发送该报文，直到发送成功。对于 STM32 来说，可将 CAN_MCR 寄存器中的 NART 位置 0 或 1，即可将 CAN

控制器配置为自动重传与非自动重传模式。下面分别在两种发送模式下进行 Bus-off 攻击。

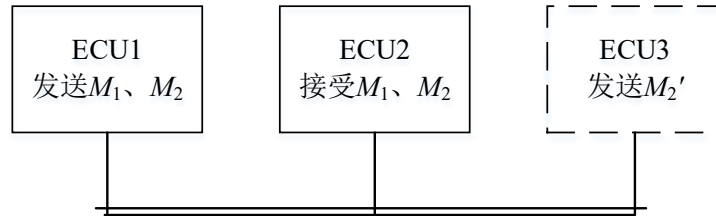


图 5-8 CAN 通信结构图

图 5-8 中，节点 ECU1 与 ECU2 为正常节点，正常情况下 ECU1 以 50ms 为周期连续发送三条报文 M_1 、 M_2 ，对于 M_2 而言， M_1 为其前置报文；节点 ECU2 接收报文 M_1 、 M_2 。ECU3 为恶意节点，其以节点 ECU1 作为 Bus-off 攻击对象，通过对其进行配置使其接收到 M_1 后发出恶意报文 M_2' ，使 M_2' 与 M_2 实现同步发送，以触发节点 ECU1 出现发送错误。为便于观察攻击过程中报文的发送、接收情况和错误计数器的变化情况，将报文与错误计数器寄存器的值在 LCD 上进行显示，同时利用示波器观测电压信号的变化情况，示波器型号为 Tektronix MSO5104，将示波器的信号采集端的高低电平触头分别与总线 CAN 高与 CAN 低相连。实验过程为首先由 ECU1 与 ECU2 进行正常通信，随后加入 ECU3 节点，Bus-off 攻击随即开始，可从 LCD 上观察到报文发送接收情况以及错误计数寄存器的值。

5.5.2 试验结果

（1）自动重传模式下实验结果与分析

在自动重传模式下，实验结果为节点 ECU1 出现发送错误且错误不断累积，经过一段时间后 ECU1 脱离总线不再发送报文，而恶意节点 ECU3 在攻击中也出现发送错误，但其最终未脱离总线，攻击成功。多次实验表明该攻击的成功率为 100%

图 5-9 为攻击第一阶段时总线电压信号图，由图可知攻击第一阶段出现的 16 个连续的错误帧。图 5-10 为图 5-9 的错误帧部分放大图，由图可知，受攻击节点与恶意节点同时发送报文并产生电平叠加，且受攻击节点首先进入错误激活状态，随后恶意节点也进入了错误激活状态，发生错误后二者发送的报文电压信号再次叠加。

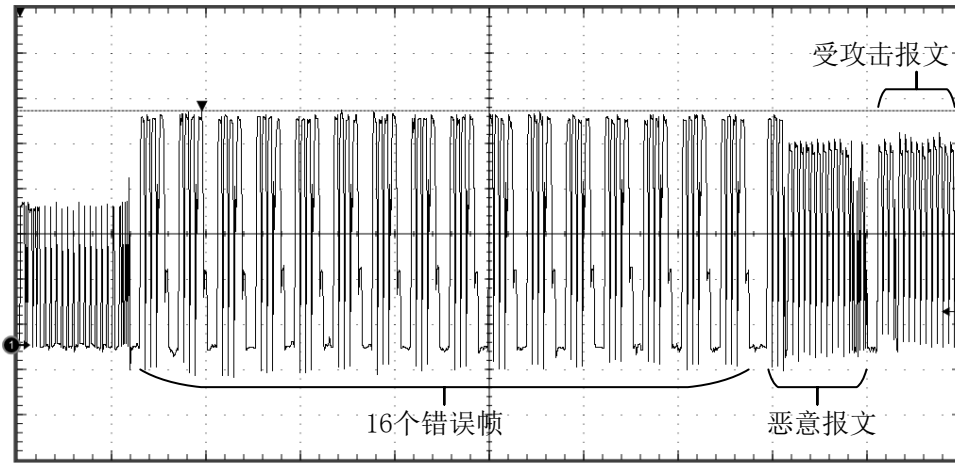


图 5-9 Bus-off 攻击发生第一阶段电压波形图

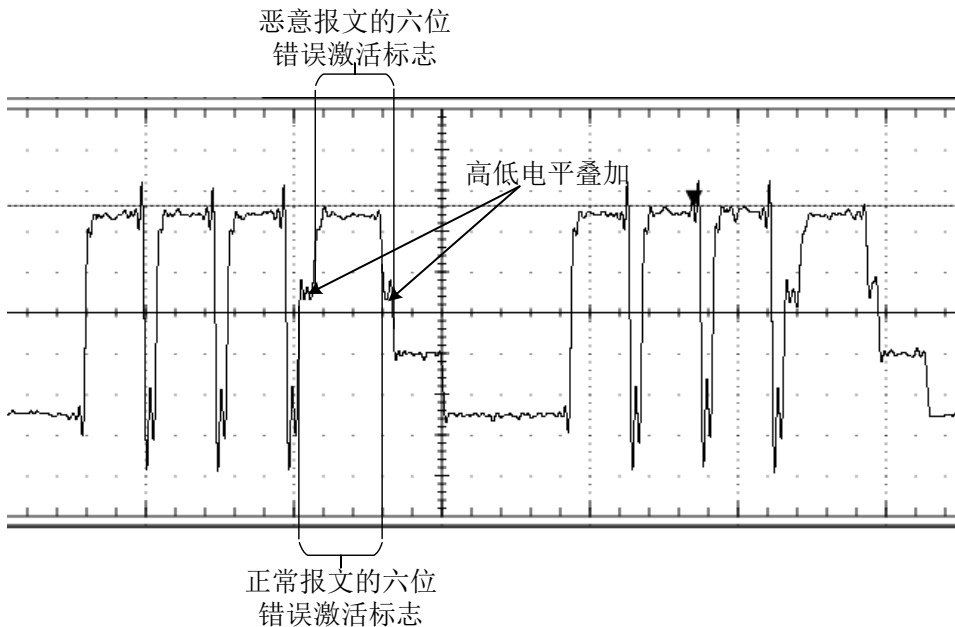
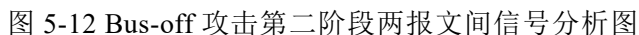
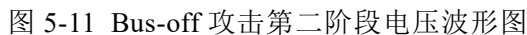


图 5-10 Bus-off 攻击发生第一阶段错误帧电压波形叠加图

在连续发送 16 个错误帧后，两节点进入错误认可状态，即进入攻击的第二阶段。图 5-11 为攻击进行到第二阶段时的电压波形图，在此阶段，受攻击报文与恶意报文同时发送，信号产生叠加，但恶意节点不发生错误，恶意报文发送成功；而受攻击节点只发生一次错误，受攻击报文在经历一次发送失败后也发送成功。由图 5-11，恶意报文的应答间隙位与受攻击报文信号的第一位之间的时间间隔为 $50\mu\text{s}$ ，时钟周期为 $2\mu\text{s}$ ，即两报文信号间出现 25 位的低电平，下面结合图 5-12 分析说明出现 25 位低电平的成因。



自动重传模式下 Bus-off 攻击正常节点与恶意节点 TEC 变化如图 5-12 所示，a 与 b 分别为恶意节点与被攻击节点的错误计数随时间变化曲线，恶意节点在

0.3s 时进行攻击，在 0.3s 至 0.35s 期间为攻击进行的第一阶段，该阶段结束时恶意节点的错误计数为 127，被攻击节点的错误计数为 134。在 0.35s 至 1.3s 时为 Bus-off 攻击的第二阶段，该阶段内被攻击节点的错误计数持续增加直至到达 256，而恶意节点的错误计数持续减少，最终减少至 90，由于恶意节点是收到被攻击节点发出的报文 M_1 后才发送 M_2' ，因此当被攻击节点脱离总线后，恶意节点无法再收到 M_1 因此其也不发送 M_2' ，因此其错误计数至 90 后不再下降。

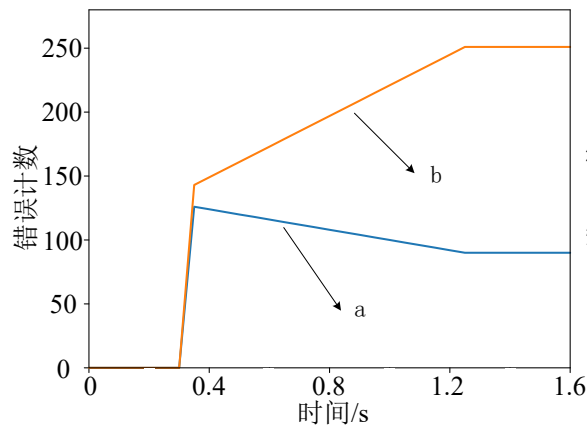


图 5-12 自动重传模式下 Bus-off 攻击正常节点与恶意节点 TEC 变化图

(2) 非自动重传模式下的实验结果与分析

在非自动重传模式下，实验结果为正常节点 ECU1 出现发送错误，但经过一段时间后，ECU1 返回正常状态；而恶意节点 ECU3 在最终反而脱离总线，攻击失败。多次实验表明该攻击的成功率为 0%，图 5-13 为正常节点与恶意节点 TEC 变化图，二者的 TEC 变化曲线分别为 a 与 b。

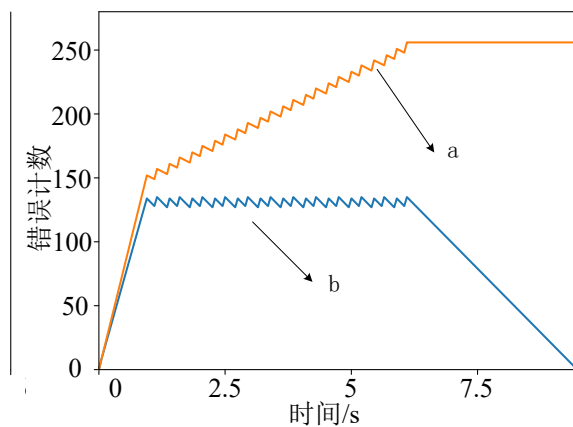


图 5-13 非自动重传模式下 Bus-off 攻击正常节点与恶意节点 TEC 变化图

由图 5-12，恶意节点在 0s 时进开始攻击，在 0s 至 0.8s 期间为攻击进行的第一阶段，该阶段结束时恶意节点与被攻击节点的错误计数分别为 128 和

113, 原因在于正常节点每一周期发送两条报文 M_1 、 M_2 , 每次发送时都是 M_1 发送成功而 M_2 发送失败, 每个周期其 TEC 加 7 (加 8 减 1), 且攻击刚开始的第一次 M_1 发送成功不会使 TEC 减 1, 因此理论上其错误计数 TEC_1 的结果为 113。

$$TEC_1 = 8 + 15 \times 7 = 113 \quad (5-9)$$

而恶意节点每发送一条报文都会遇到一次错误, 因此理论上其错误计数器 TEC_2 的计算结果为 128。

$$TEC_2 = 16 \times 8 = 128 \quad (5-10)$$

之后进入过渡阶段, 此时正常节点为错误激活状态, 恶意节点为错误认可状态, 由于攻击仍在进行, 因此二者的错误计数继续增加, 直到其同时进入错误认可状态, 此时二者的错误计数 TEC_1 与 TEC_2 分别为 134 与 152, 与实验结果相符, 而后两节点均进入错误认可状态。

$$TEC_1 = 113 + 7 \times 3 = 134 \quad (5-11)$$

$$TEC_2 = 128 + 8 \times 3 = 152 \quad (5-12)$$

当两节点都进入 Bus-off 攻击的第二阶段即错误认可状态, 由图 5-13 可知, 此时的错误计数变化曲线呈锯齿状, 以下分析其成因。

由于节点均为非自动重发模式且处于错误认可状态, 因此当正常节点发出 M_1 后, 在发送 M_2 之前会发出 8 位暂停发送场, 而恶意节点在收到 M_1 后立即发送 M_2' , 由于暂停发送场的存在, 使得 M_2' 与 M_2 不发生同步, 因此报文 M_1 、 M_2' 与 M_2 都可成功发送, 两节点的错误计数均下降, 此时攻击节点不起作用。如图 5-14 所示。

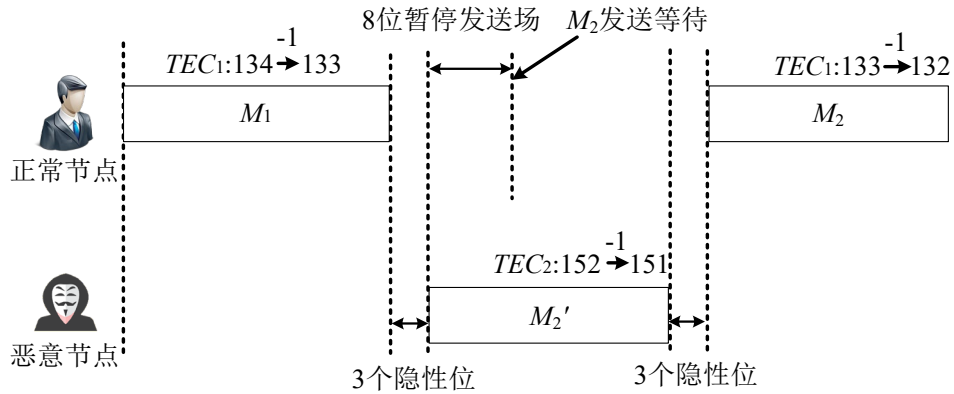


图 5-14 非自动重发模式下 Bus-off 攻击第二阶段报文发送情况

由图可知, 经过若干报文周期, 正常节点 TEC_1 会首先将到 128 以下, 此时该节点重新进入错误激活状态, 继而两节点发送报文 M_2' 与 M_2 时会再次同步从而两节点的错误计数都增加并又同时处于错误认可状态, 而后两节点的报文又

发送成功且两节点的错误计数再次下降，不断重复该过程，最终恶意节点的错误计数器首先到达 256，恶意节点脱离总线，攻击失败。

由实验可知，无论是恶意节点还是被攻击节点，错误计数增长较快的节点会脱离总线，而错误计数增长较慢的节点会恢复到正常状态。因此，若要使恶意节点攻击成功，可令恶意节点在发送恶意报文的同时发送一些合法报文从而使其错误计数始终小于被攻击节点的错误计数。

5.6 Bus-off 攻击特征分析与检测

Bus-off 攻击最显著的特征为：（1）被攻击节点错误计数会增加至 256；（2）总线上会出现大量错误帧。由于本文采取集中式的检测方式，即设置一个单独的检测节点专用于检测，而该节点无法获取总线上其他各节点的错误计数，因此不采用（1）作为检测特征。而检测节点可获取总线上的错误帧信息，进一步地，由于错误帧中的错误激活标志更易被检测到，且 Bus-off 攻击第一阶段的错误激活标志至少有 16 个，因此可将这 16 个错误激活标志作为检测特征。

然而，仅凭借该特征无法将攻击确认为 Bus-off 攻击，因此还需明确 Bus-off 攻击的其他特征。注意到 Bus-off 攻击第二阶段进行时，无论是自动重发模式还是非自动重发模式，总线上均会有两条紧挨着的相同 ID 的报文，这两条报文的时间间隔很短，表现为注入攻击。且经过一段时间后，由于被攻击节点脱离总线，因此其不再发送报文，此时表现为中断攻击。

综上所述，Bus-off 发生时的特征为：首先总线上短时间内出现 16 个及以上的错误激活标志；接着报文频率明显增加；最后报文频率变为 0。因此，可将错误帧中的错误激活标志与报文周期作为 Bus-off 攻击的检测特征。

检测方法流程如图 5-15 所示，检测开始首先进行初始化，并获取报文周期，判断是否出现错误激活标志，若检测到总线中出现大于或等于 16 个错误激活标志，则进入下一步，判断是否存在注入攻击与中断攻击，若均存在，则检测节点发出 Bus-off 攻击警报。实验表明，该方法对 Bus-off 攻击的检测率为 100%。

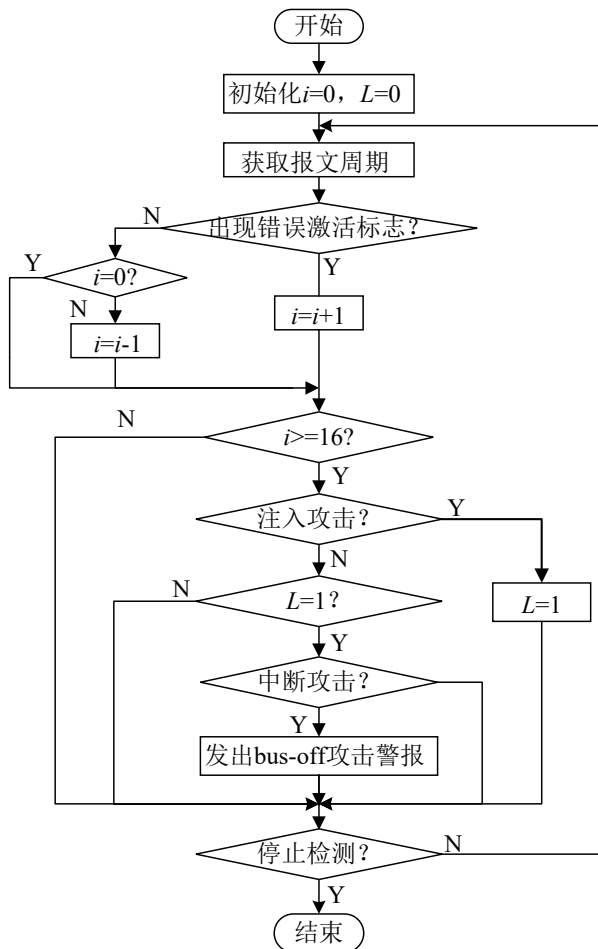


图 5-15 Bus-off 攻击检测方法流程图

5.7 本章小结

本章提出了一种新的针对车内 CAN 总线攻击方式——Bus-off 攻击。其原理是黑客利用 CAN 的错误处理机制,通过注入恶意报文并使其与被攻击报文进行同步发送以触发位错误,使被攻击节点误认为总线出现故障从而使被攻击节点脱离总线。该攻击的实现条件包括:被攻击报文为周期性报文;恶意报文和被攻击报文的 ID 相同且两报文同步发送;恶意报文与被攻击报文除仲裁场以外至少有一位不同。攻击过程中,被攻击的正常节点先后进入错误激活、错误认可及脱离总线状态。重点分析了攻击实现过程中恶意报文与被攻击报文的两种同步方法。利用模拟车内 ECU 通信网络实现了 Bus-off 攻击,实验结果表明 Bus-off 攻击成功率可达 100%。

分析了 Bus-off 攻击特征,在此基础上提出了针对 Bus-off 攻击的检测方法,该算法的检测特征为节点错误计数器与报文频率的变化。基于 STM32 进行了算法实现,结果表明所提检测方法可有效检测 Bus-off 攻击。

结 论

为保障车内 CAN 通信安全, 本文在分析车内 CAN 总线脆弱性及其面临的信息安全威胁的基础上, 提出了三种车内 CAN 总线入侵检测算法, 包括基于报文周期特性的自适应入侵检测算法、基于 DACHE 特征的入侵检测算法以及针对 Bus-off 攻击的入侵检测算法, 并对所提算法进行了实现。具体的研究成果如下:

(1) 分析了车内 CAN 通信网络架构及面临的主要安全威胁。

分析了车内通信网络架构与一种典型的车内 CAN 通信协议——SAE J1939 协议, 分析了该协议对数据帧仲裁场的规定以及数据场中的参数格式; 研究了车内 CAN 总线的脆弱性及针对其攻击的具体方式; 将报文周期与报文数据场内容作为入侵检测算法的检测特征, 并确定了检测精度、数据处理性能、自身安全性与对原系统的影响程度等入侵检测算法的评价指标。

(2) 提出并实现了一种基于报文周期特性的自适应入侵检测算法。

分析了实际车内正常 CAN 报文周期特性, 得出了在正常情况下周期性报文的周期波动大小与总线负载情况及报文优先级有关的结论; 给出了一种针对车内网络的基于报文周期的入侵检测算法, 并分析了周期波动对其检测精度的影响; 对检测阈值进行了优化, 提出了一种基于报文周期特性的自适应入侵检测算法, 针对周期变化幅度的不同给出了自适应检测阈值的确定方法; 试验结果表明该算法可检测到绝大多数注入攻击与所有中断攻击。

(3) 提出并实现了一种基于 DACHE 特征的入侵检测算法。

以 CAN 报文数据场内容为检测特征, 研究了基于汉明距离的入侵检测算法, 经分析得出了该算法在检测伪造与重放攻击时存在不足的结论; 对检测特征进行改进, 提出了一种基于数据场 DACHE 特征的入侵检测算法; 该算法以 DACHE 特征作为检测算法输入特征, 采用 BP 神经网络作为分类模型; 在数据预处理过程中, 针对数据类别不平衡的问题提出了一种欠采样的数据选择方法; 优化了神经网络反向传播算法, 即在迭代过程中加入动量项调整梯度方向, 加入自适应项调整迭代学习率, 使得收敛速度大大提高, 且不易陷入局部最优; 基于 Python 进行了算法实现并验证了算法有效性。

(4) 提出并实现了一种特殊的针对车内 CAN 总线攻击方式——Bus-off 攻击及其检测方法。

研究了 Bus-off 攻击的原理并给出了攻击的实现条件; 分析了 Bus-off 攻击的具体过程及实现方法, 重点分析了攻击实现过程中恶意报文与被攻击报文的

两种同步方法；利用模拟车内 ECU 通信网络实现了 Bus-off 攻击，实验结果表明 Bus-off 攻击成功率可达 100%；分析了 Bus-off 攻击特征，提出了针对 Bus-off 攻击的入侵检测方法；以错误帧及报文周期变化作为检测特征，基于车内 CAN 通信模拟平台实现了入侵检测方法，试验结果表明所提入侵检测方法可有效检测 Bus-off 攻击。

参考文献

- [1] 宋城. 国家发改委等联合印发《汽车产业中长期发展规划》[J]. 中国设备工程, 2017 (9): 1-1.
- [2] 李克强. 智能网联与未来汽车的技术变革[J]. 科学中国人, 2015(28): 20-21.
- [3] 张岩. 车联网时代防止黑客把汽车变成大杀器 一次震惊汽车界的黑客“挟持”[J]. 环境与生活, 2015 (10): 34-37.
- [4] Miller C, Valasek C. Remote exploitation of an unaltered passenger vehicle[J]. Black Hat USA, 2015.
- [5] 蒋苏晨. 智能汽车企业发展趋势研究[J]. 工程技术: 全文版, 2016 (11): 318.
- [6] 孙伟华, 何蔚, 李诗骋. 车联网时代的安全与隐私[J]. 标准科学, 2016 (1): 20-23.
- [7] 杨南, 康荣保. 车联网安全威胁分析及防护思路[J]. 通信技术, 2015, 48 (12): 1421-1426.
- [8] SAE Vehicle Electrical System Security Committee. SAE J3061-Cybersecurity Guidebook for Cyber-Physical Automotive Systems[J]. SAE-Society of Automotive Engineers, 2016.
- [9] Koscher K. et.al. Experimental security analysis of a modern automobile[C]. Proc. IEEE Symp. Secur. Privacy, 2010: 447-462.
- [10] Checkoway S., McCoy D., Kantor B. Comprehensive experimental analyses of automotive attack surfaces[C]. Proc. USENIX Secur. Symp., 2011: 1-16.
- [11] Greenberg A. Hackers remotely kill a jeep on the highway-with me in it[J]. Wired, 2015, 7 (2): 21-22.
- [12] Hoppe T., Kiltz S., Dittmann J. Security threats to automotive CAN networks-Practical examples and selected short-term countermeasures[C]. Proc. Int. Conf. Comput. Safety, 2008: 235-248.
- [13] Müter M., Groll A., Freiling F. C. A structured approach to anomaly detection for in-vehicle networks[C]. Proc. 6th Int. Conf. Inf. Assurance Security (IAS), 2010: 92-98.
- [14] Marchetti M., Stabili D., Guido A., Colajanni M. Evaluation of anomaly detection for in-vehicle networks through information-theoretic algorithms[C]. Proc. Res. Technol. Soc. Ind. Leveraging Better Tomorrow (RTSI), 2016: 1-6.
- [15] Müter M., Asaj N. Entropy-based anomaly detection for in-vehicle networks[C]. Proc. IEEE Intell. Vehicles Symp., 2011: 1110-1115.

-
- [16]Lee H., Jeong S. H., Kim H. K. OTIDS: A novel intrusion detection system for in-vehicle network by using remote frame[C]. Proc. PST (Privacy, Secur. Trust), 2017: 1-10.
- [17]Sagong S. U., Ying X., Clark A, L. Bushnell, R. Poovendran. Cloaking the clock: Emulating clock skew in controller area networks[C]. Proc. 9th ACM/IEEE Int. Conf. Cyber-Phys. Syst., 2018: 32-42.
- [18]Murvay P. S., Groza B. Source identification using signal characteristics in controller area networks[C]. IEEE Signal Process. Lett., 2014, 21(4): 395-399.
- [19]Cho K. T., Shin K. G. Fingerprinting electronic control units for vehicle intrusion detection[C]. Proc. 25th USENIX Secur. Symp., 2016: 911-927.
- [20]Tian D. et. al., An intrusion detection system based on machine learning for CAN-bus[C]. Proc. Int. Conf. Ind. Netw. Intell. Syst., 2017: 285-294.
- [21]Theissler A. Detecting known and unknown faults in automotive systems using ensemble-based anomaly detection[J]. Knowl.Based Syst., 2017, 123(1): 163-173.
- [22]Li H., Zhao L., Juliato M., Ahmed S., Sastry M. R., Yang L. L. POSTER: Intrusion detection system for in-vehicle networks using sensor correlation and integration[C]. ACM SIGSAC Conf. Comput. Commun. Secur., 2017: 2531-2533.
- [23]Kang M. J., Kang J. W. Intrusion detection system using deep neural network for in-vehicle network security[J]. PLoS ONE, 2016, 11(6): 35-41.
- [24]Kang M. J., Kang J. W. A novel intrusion detection method using deep neural network for in-vehicle network security[C]. IEEE 83rd Veh. Technol. Conf. (VTC Spring), 2016: 1-5.
- [25]Taylor A., Leblanc S., Japkowicz N. Anomaly detection in auto-mobile control network data with long short-term memory networks[C]. IEEE Int. Conf. Data Sci. Adv. Anal (DSAA), 2016: 130-139.
- [26]Narayanan S. N., Mittal S., Joshi A. OBD SecureAlert: An anomaly detection system for vehicles[C]. IEEE Int. Conf. Smart Comput (SMARTCOMP), 2016: 1-6.
- [27]Studnia I., Alata E., Nicomette V., Kaâniche M. A language-based intrusion detection approach for automotive embedded networks[J]. Int. J. Embedded Syst., 2018, 10(1): 1-12.
- [28]Bogdan Groza, Pal-Stefan Murvay. Efficient Intrusion Detection with Bloom Filtering in Controller Area Networks[J]. IEEE TRANSACTIONS ON

- INFORMATION FORENSICS AND SECURITY, 2019, 14 (4): 1037-1051.
- [29]车联网网络安全委员会. 车联网网络安全白皮书[Z]. 成都: 电子科技大学, 2016.
- [30]车联网网络安全委员会. 车联网网络安全防护指南细则(讨论稿)[Z]. 成都: 电子科技大学, 2016.
- [31]李均. CANSsee: An Automobile Intrusion Detection System[Z]. 北京: 奇虎 360, 2016.
- [32]Apollo official website[B/OL]. http://apollo.auto/platform/security_cn.html.
- [33]于赫. 网联汽车信息安全问题及 CAN 总线异常检测技术研究[D]. 长春: 吉林大学, 2016: 37-53.
- [34]闫鑫. 基于 Renyi 信息熵的 CAN 总线异常检测方法[D]. 吉林大学, 2017.
- [35]戚琦. 基于非参数 CUSUM 算法的车载 CAN 总线拒绝服务攻击检测[D]. 吉林大学, 2018.
- [36]杨宏. 基于智能网联汽车的 CAN 总线攻击与防御检测技术研究[D]. 天津: 天津理工大学, 2017: 40-41.
- [37]曾凡. 网联汽车入侵检测系统的研究与实现[D]. 电子科技大学, 2018.
- [38]曾润. 车载 CAN 总线网络异常数据检测技术与实现[D]. 北京邮电大学, 2018.
- [39]吴贻淮. 基于神经网络的车载 CAN 网络入侵检测系统的研究[D]. 成都信息工程大学, 2018.
- [40]肖剑, 李文江, 耿洪杨, 翟英博. 车联网中可抵抗 DoS 攻击的 RFID 安全认证协议[J/OL]. 北京邮电大学学报, 2019 (02): 1-6.
- [41]张亚丰, 洪征, 吴礼发, 周振吉, 孙贺. 基于状态的工控协议 Fuzzing 测试技术[J]. 计算机科学, 2017, 44 (05): 132-140.
- [42]钱铁云, 王毅, 张明明, 刘俊恺. 基于深度神经网络的入侵检测方法[J]. 华中科技大学学报(自然科学版), 2018, 46 (01): 6-10.
- [43]Seo Eunbi, Song, et.al. GIDS: GAN based Intrusion Detection System for In-Vehicle Network[J]. Annual Conference on Privacy Security and Trust-PST, 2018, 8, pp. 286-291.
- [44]陈万志, 徐东升, 张静. 工业控制网络入侵检测的 BP 神经网络优化方法[J]. 辽宁工程技术大学学报(自然科学版), 2019, 38 (01): 82-87.

攻读硕士学位期间发表的论文及其它成果

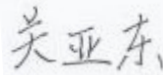
- [1] 关亚东, 许磊, 李中伟, 朱识天, 谭凯. 继电保护定值在线发放管理系统及其安全防护[J]. 电测与仪表, 2018, 55(23): 7-14+21.
- [2] 李中伟, 张啸, 武东升, 梁建权, 关亚东. 家庭能量管理系统多目标能量调度优化策略[J]. 自动化仪表, 2019, 40(04): 46-51.

哈尔滨工业大学学位论文原创性声明和使用权限

学位论文原创性声明

本人郑重声明：此处所提交的学位论文《车内 CAN 总线入侵检测算法研究》，是本人在导师指导下，在哈尔滨工业大学攻读学位期间独立进行研究工作所取得的成果，且学位论文中除已标注引用文献的部分外不包含他人完成或已发表的研究成果。对本学位论文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明。

作者签名：



日期： 2019 年 6 月 19 日

学位论文使用权限

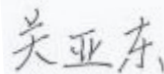
学位论文是研究生在哈尔滨工业大学攻读学位期间完成的成果，知识产权归属哈尔滨工业大学。学位论文的使用权限如下：

(1) 学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文，并向国家图书馆报送学位论文；(2) 学校可以将学位论文部分或全部内容编入有关数据库进行检索和提供相应阅览服务；(3) 研究生毕业后发表与此学位论文研究成果相关的学术论文和其他成果时，应征得导师同意，且第一署名单位为哈尔滨工业大学。

保密论文在保密期内遵守有关保密规定，解密后适用于此使用权限规定。

本人知悉学位论文的使用权限，并将遵守有关规定。

作者签名：



日期： 2019 年 6 月 19 日

导师签名：



日期： 2019 年 6 月 27 日

致 谢

研究生两年的时光如白驹过隙，转瞬即逝，这个阶段是我人生一个的非常重要的阶段。研究生阶段中的学习、工作和生活中都离不开各位老师的指导、实验室同学的帮助、父母的关怀以及女友的关心与支持，在此向大家表示衷心的感谢。

感谢我的导师金显吉老师，金老师在工程应用方面有非常丰富的知识和经验，对我的硕士学位论文提出了很多非常有价值的意见。感谢我的副导师李中伟老师对我的指导，李老师不仅在学习工作上督促我，还在生活中给予了我很多的帮助，帮助我养成了良好习惯，李老师的“德高为师，学高为范”的精神、处事泰然自若波澜不惊、待人温文尔雅且多为他人着想，这些闪光点是我终生学习的榜样。感谢网络与电气智能化研究室的佟为明老师、刘勇老师、赵志衡老师对我课题以及硕士学位论文的指导，正是在各位老师严谨与细致入微的指导下，我才能够成长为一名合格的硕士毕业生。

在读研期间认识了很多有趣而优秀的朋友们，大家不仅在学习、生活上互帮互助，也为实验室带来了欢声笑语，一起度过了忙碌、充实而欢乐的两年时光。再次特别感谢谭凯师弟在工作中的帮助，刘延龙、高吉星师兄在迷途中的指导。感谢我的室友们在生活中对我的帮助，特别感谢我的下铺黄同学在我忘记带饭卡的时候、木有卫生纸的时候慷慨相助。祝大家前程似锦、友谊地久天长。

感谢我的父母亲朋对我的关心与支持，正因为他们作为我最坚实的后盾与支持者，我才能走到今天。感谢我的女友孟钰同学，因为有你我感受到内心的笃定与爱情的甜蜜，以后还请多多关照。

最后，衷心感谢评审我的论文的各位老师，谢谢！