不当行为检测仿真框架

车载网络

约瑟夫·卡梅尔、穆罕默德·拉希德·安萨里、乔纳森·珀蒂、阿尔诺·凯撒、

伊内斯·本·杰玛，帕斯卡·乌里恩

不当行为检测仿真框架
车载网络

约瑟夫·卡梅尔*†、穆罕默德·拉希德·安萨里‡ ，乔纳森·珀蒂‡ ，阿诺·凯撒* ，Ines Ben Jemaa* ，和帕斯卡
乌里安†

*IRT SystemX,法国帕莱索†Tel´ecom
ParisTech,法国巴黎‡OnBoard Security, Inc.,
美国威尔明顿

摘要 协同智能交通系统 (C-ITS) 是一项持续发展的技术,将在不久的将来改变我们的驾驶体验。在此类系统中,车辆和路边单元 (RSU) 通过在车辆网络上广播 V2X 消息进行合作。安全应用程序使用这些数据来及时检测和避免危险情况。C-ITS 中的错误行为检测 (MBD) 是一个活跃的研究课题,它包括监视交换的 Vehicle-to-X 通信 (V2X) 消息的数据语义,以检测和识别潜在的错误行为实体。检测过程包括对接收到的 V2X 消息执行合理性和一致性检查。

如果检测到异常,该实体可以通过向不当行为管理机构 (MA) 发送不当行为报告 (MBR) 来进行报告。
然后 MA 将调查该事件并决定是否撤销发送者。在本文中,我们提出了一个错误行为检测 (MBD) 模拟框架,使研究社区能够开发、测试和比较 MBD 算法。我们还通过运行示例场景并讨论其结果来展示其功能。不当行为检测框架(F2MD) 是开源的,可在我们的 github 上免费获得。

索引词 合作智能交通系统（C
ITS），错误行为检测,模拟

一、引言

智能交通系统 (ITS) 领域在过去几年中以更快的速度发展。

自动驾驶汽车现在已经成为现实,车辆可以获得高达 4 级的驾驶能力,即无需太多人为干预 [1]。已经应用和/或发明了大量技术来支持 ITS 的未来。雷达、激光雷达和摄像头等传感器大量用于高级驾驶辅助系统 (ADAS)、车道保持、前方碰撞警告等应用。然而,上述传感器仅在视线条件下工作。对于非视距应用,专用短程通信 (DSRC) 和蜂窝车联网 (C-V2X) 等技术一直在研发中。

DSRC 和 C-V2X 支持称为互联车辆 (CV) 的下一代车辆。互联车辆在称为车载自组织网络 (VANET) 的临时网络上广播它们的运动信息（例如,位置、速度、航向等）。这使简历能够“看到”超越视线,但它也带来了攻击面。
VANET 中积极参与的实体可能会对其运动学说谎,从而导致网络中断并直接导致道路安全问题。这种行为在 VANET 中被称为不当行为。 VANETS 中的不当行为包括无意中发送错误信息的故障实体和故意在 V2X 攻击中传输虚假信息的恶意实体。

VANET 中的错误行为检测 (MBD) 是一个活跃的研究领域,专注于开发检测与车辆移动、传输等相关的异常行为的机制。MBD 算法可以基于传感器或基于 V2X。在本文中,我们专注于基于 V2X 的 MBD。为了进行大量研究,这些技术必须在不同场景下大规模实施,具有各种车辆密度和许多其他变化。目前,只有少数广泛部署的 CV [2]、[3]。这些部署仍然存在各种问题,例如用例数量有限以及对生成数据的严格规定。

由于法规要求匿名和剥离可能有助于设计检测算法的信息,因此很难获得原始的、未触及的数据。此外,为分析找到合适的数据子集需要花费大量时间,而这些时间可用于实际设计 MBD 算法。因此,迫切需要模拟 VANET 并在这些模拟中评估 MBD 算法。

网络仿真中的 VEhicles 等仿真器
(VEINS) [4] 为 MBD 算法开发提供了一个平台。在本文中,我们提出了我们的不当行为检测框架(F2MD)。 F2MD 提供了一个单一的框架,可以：

·实施一组新的 V2X 攻击， ·实施多种 MBD 算法以
便于比较， ·评估攻击的有效性（我们的框架提供了 6 种攻击和 9 种错误行为），

·评估 MBD 算法的性能（我们的框架提供 15 种算法）,使用 8 个指标， ·实时可视化 MBD 算法性能，

·生成数据集以提供常见攻击数据集

2019年XXXX年稿件接收;修订 XXX,XX,2020。

VeReMi [5], [6],

·评估多种不当行为报告格式和全球不当行为检测算法。

本文结构如下。我们在第二节讨论相关工作。在第三节中,我们详细介绍了系统模型。
在第四节中,我们详细解释了我们提出的框架。
在第五节中,我们使用我们的框架运行多个示例来演示可以从中获得的结果范围。
第六节介绍了我们的结论和未来的工作。

二。相关工作

在本节中,我们概述了从通用物理层到基于机器学习的 MBD 技术的相关工作。我们还讨论了与这些技术的模拟相关的工作、用于评估其性能的指标以及每项工作中考虑的攻击。

范德海登等人。 [7] 对 C-ITS 中的 MBD 进行了调查。
他们提供了以节点为中心和以数据为中心的检测技术的分类,并根据它们的范围、安全性、隐私、泛化能力和所需资源对它们进行了比较。调查表明,目前还没有完美的技术。该调查以开放式挑战结束:阈值的定义、针对女巫攻击的保护以及向后端报告等等。这里提出的 F2MD框架旨在寻找解决这些开放挑战的方法。

卡梅尔等人。 [8] 调查了 MBD 算法在当前标准、法律合规性以及硬件/软件要求方面的可行性。作者表明,一些 MBD 算法不符合现行法规(法律或标准)。研究中讨论的大多数挑战都源于隐私保护法规。 F 2MD 框架可以通过集成各种假名更改策略 (PCP)(C-ITS 的当前隐私解决方案)来帮助找到保护隐私的 MBD 解决方案。因此,MBD 算法可以针对不同的 PCP 进行测试。

孙等。 [9] 探索了使用车对车通信 (V2V) 通信的应用程序和物理层可用的功能来验证攻击者的位置和移动性。在他们的攻击者模型中,攻击者在基本安全消息 (BSM) 中传输可疑位置。

这个攻击者可以是“独狼”,也可以与其他证实虚假数据的攻击者勾结。作者仅考虑在自我车辆的通信范围内至少有一辆诚实车辆的直线高速公路场景。

他们使用到达角估计 (AoA)、多普勒速度测量 (DS)、扩展卡尔曼滤波器 (EKF) 和来自邻近车辆的输入来验证攻击者的位置和移动信息。他们根据真阳性率、假阳性率、真阴性率和假阴性率来评估其机制。作者在 MATLAB 中实现了他们的框架,并在相同的环境中进行了评估。

在[10]中,作者提出了一种基于支持向量机(SVM)和Dempster-Shafer证据理论(DST)的MBD机制来检测虚假消息注入。基于 SVM 的分类器用于根据消息内容和车辆属性检测虚假消息。另一个基于 SVM 的分类器用于根据车辆在消息传播方面的行为来评估车辆是否可信。然后,DST 被可信的第三方使用,聚合关于同一车辆的多个信任评估报告,并得出综合信任值。作者根据真阳性率、假阳性率和准确性评估他们系统的性能。他们在 VEINS 中进行了模拟。

将他们的提案放入我们的框架中,DST 将成为 MA 的一部分。我们的F2MD包含对生成大量虚假报告的 MA 的压力攻击。正如作者所指出的,这种类型的攻击将有助于测试和改进这种 DST 算法。

所以等。 [11] 提出了一个框架,使用合理性检查作为特征向量来评估两个机器学习模型(SVM 和 KNN)。他们根据分类准确性和精确召回特性评估了这些模型的性能。他们对位置合理性、运动合理性和其他定量特征进行评分,以输入 SVM 和 KNN 模型。作者考虑了来自 VeReMi 数据集 [5] 的位置欺骗攻击。攻击和检测是在 LuST 场景 [12] 上进行的。他们的模拟数据是使用 VEINS 生成的,并在 MATLAB 中进行了评估。尽管他们的研究很全面,但攻击类型仅限于位置欺骗攻击。

可以安全地假设模拟是 MBD 算法评估的关键部分。对于 V2V 模拟,模拟器应包含模拟真实世界 V2V 场景的网络和移动模型。 NS3 和 OMNET++ 等模拟器为网络模拟提供了功能丰富的环境。然而,这些模拟器并未模拟 VANET 的一个关键方面,即车辆的移动模型。

在 MATLAB 中执行的模拟需要一个随时可用的数据集,可以在该数据集上运行 MBD 算法以进行评估。

在 C-ITS 研究中,一种常用的网络和移动模拟器是 VEINS [4]。 VEINS 结合了 OMNET++ 和 SUMO(交通移动模拟器)来创建一个用于 V2X 模拟的层。 VEINS 提供 API 来创建在车辆本地运行的自定义应用程序。这些应用程序可以在接收到来自另一辆车的信标和/或在其他功能中改变自己的位置时做出反应。 VEINS 还提供了为不同道路网络生成自定义数据集的能力。但是,它不包括 MBD 算法或评估它们的能力。

据我们所知,只有一种仿真框架允许在 VANET 中评估 MBD 算法,即车辆参考不当行为 (VeReMi)

[5]. VeReMi 是 VEINS 的扩展,由两个主要部分组成:(i)一个数据集,包含来自行为不端和真实车辆的传输数据,用于离线评估 MBD 算法; (ii) 五种基于位置的攻击和四种基本的 MBD 算法。

在本文中,我们介绍了我们提出的模拟框架工作F2MD,它也基于 VEINS 并包含一个完整的功能集列表,从基础到高级,如第 IV 节所述。 F2MD 允许对在大型道路网络中模拟的大量车辆进行研究 [12]。

可以切换这些道路网络以对不同类型的道路网络进行评估。可以在F2MD 中将攻击编程到攻击车辆中,此功能可确保攻击中没有偏见。它包含基于位置和基于移动性的攻击的完整列表。它提供的 MBD 算法范围从基本的合理性检查到基于阈值、值聚合和行为分析的高级算法。 F2MD 还支持为 MBD 添加机器学习模型。所有 MBD 算法都可以扩展以进一步增强,或者可以轻松地将新算法添加到F2MD 框架中。 F2MD 能够生成要发送给 MA 的不当行为报告。这些报告的选择范围从基本报告到包含证据的报告,以帮助 MA 做出决定。最后， F2MD 提供了一种可视化功能,可以在线实时生成图形来分析 MBD 算法的性能。

总之,上述每项相关工作都使用各种方法来评估其 MBD 算法的性能,并且几乎没有可比性。 F2MD 提供了一个免费的开源统一平台来实施对 VANET 的攻击、实施 MBD 算法并使用通用指标评估它们的性能。

三、系统模型

A. C-ITS系统模型

协同智能交通系统 (C-ITS) 由移动实体（例如乘用车、卡车、售后市场设备、手持设备)和静态实体（例如路边单元 (RSU)、电动汽车充电站、交通管理中心)组成).每辆车都配备了一个车载单元 (OBU),使其能够将移动数据（例如远程信息处理、运动、操纵)广播到邻近实体或互联网以提供基于云的服务。主要目标是通过频繁广播（即高达 10Hz,如美国标准 [13] 所述)本地信标（在美国称为基本安全消息 (BSM),或在欧洲称为合作意识消息 (CAM))来提高道路安全。 BSM/CAM 包含位置、速度、加速度、制动状态和其他可选字段 [13]、[14]。接收车辆将使用 BSM/CAM 了解场景（通过在自动驾驶车辆的环境中融合其传感器数据)并据此采取行动（例如通知操作员、执行操作）。

使用 BSM/CAM 的安全应用示例之一是紧急电子刹车灯 (EEBL)。在 EEBL [15] 中,执行紧急制动（即以大于 0.4g 的水平减速)的车辆广播消息,其中包含指示其更大减速度的相应字段（例如 BSM 中的字段）。

在接收器方面,EEBL 应用程序会在前方同一车道或相邻车道的车辆发生紧急制动事件时向驾驶员发出警告。 EEBL 应用程序有望在直线和弯曲的道路几何形状中发挥作用。 C-ITS 安全应用程序已到位,以提高道路安全并防止致命事故。因此,导致此类系统故障的攻击可能会导致灾难性的后果。 C-ITS 系统可能会受到外部和内部攻击。但是,已经建立了一个安全架构来减轻这两种攻击的影响。

B. C-ITS 安全架构

在当前的 C-ITS 系统中,可能有两种类型的攻击者:外部攻击者（即没有有效凭证)和内部攻击者（即具有有效凭证）。外部攻击使用公钥基础设施 (PKI) 进行处理,而内部攻击则通过错误行为检测系统进行缓解。图 1 显示了 C-ITS 安全架构的各个阶段。

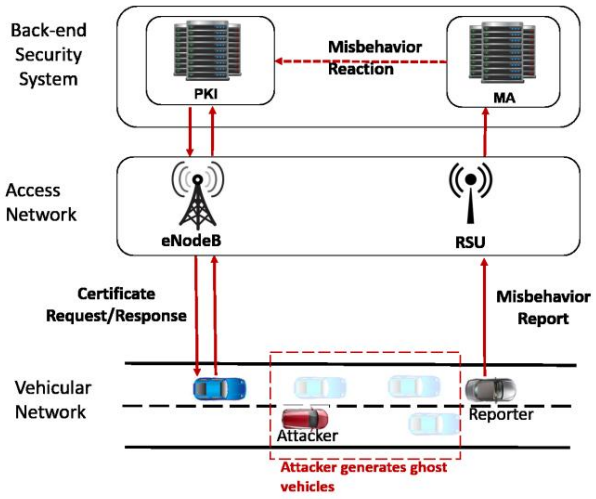


图 1 C-ITS 安全架构

外部 C-ITS 攻击者,即没有有效凭证的攻击者,可以通过使用 [16]、[17] 中指定的数字签名对媒体上发送的每条消息进行身份验证来阻止。实际上,设备身份验证依赖于管理加密材料（即生成、供应、更新、撤销)的 PKI。在美国, PKI 被称为安全凭证管理系统 (SCMS) [18],并使用非对称密码术,即 ECDSA,作为身份验证算法。此外,为了降低车辆的可追踪性,每辆车辆都使用短期加密凭证（称为假名)并定期更改它 [19]。因此,正如在下一节中讨论的那样,我们假设密码验证将处理外部攻击者并专注于内部攻击者。

C-ITS 中的内部攻击者由错误行为检测机制处理。正如我们在本文中所展示的,错误行为检测过程分为四个步骤：

1)本地不当行为检测:每个 C-ITS 实体都必须运行一个不当行为检测系统,以应对内部攻击者。

2)不当行为报告:检测后,实体将有机会通过向 MA 发送报告来表明不当行为。

3)全局不当行为检测： MA 将调查事件并可能触发不当行为实体的撤销。

4) Misbehavior reaction: MA 将发出适当的反应来保护系统（例如向 PKI 执行证书撤销请求）。

C. 攻击者模型

在本文中,我们考虑以下攻击者模型:

1) Insider:攻击者拥有所需的密码
在 C-ITS 中进行通信的凭据。

2)主动:攻击者主动参与C-ITS通信并发送虚假数据。

3)消息负载修改:我们假设攻击者可以修改其传出 BSM/CAM 中的任何字段。

如果攻击者可以完全访问其车辆的 CAN 总线,则这是可能的。攻击者可以发起中间人攻击并修改任何传感器数据。

4)传输速率修改:我们假设攻击者可以修改她车载单元的传输速率。我们假设攻击者修改了他车辆中的车载单元,根据攻击类型允许更快或更慢的传输速率。

5)假名证书访问:我们假设攻击者可以完全访问假名证书的使用。我们使用与上述相同的前提,即攻击者已经修改了她的车载单元。它应该允许攻击者随意使用假名证书。这将启用女巫攻击。

攻击者将执行第 IV-F 节中描述的攻击。
按照第 III-A 节中介绍的 EEBL 示例,攻击者会发送错误的减速值(可能连同相应的移动数据)以迫使受害者突然制动。没有有效凭据的攻击者将无法正确签署消息,并且接收方的验证将失败。然而,内部攻击者将发送经过身份验证的消息,不被他们愚弄的唯一方法是执行 MBD 算法。

四、框架

在本节中,我们将描述 F 2MD 的不同组件。下一节中描述的框架的所有部分也可以开源格式 [20] 下载。

A. 总体框架特征

该框架为 MBD 系统的实时仿真和评估提供了完整的解决方案。它通过大量 MBD、评估和其他通用 C-ITS 模块扩展了 VEINS。 F 2MD 的主要特征之一是其模块化。该体系结构分为几个功能级别:输入数据、局部检测、局部视觉输出、报告数据输出和全局检测。根据不当行为的评估级别,可以选择场景的复杂性、攻击方式和检测方法。此外,F2MD 是可扩展的。除了实现的 MBD 机制和攻击之外,它还提供了通过现有 API 使用附加模块扩展框架的可能性。我们框架的一个关键特征是它与非模拟模块集成,例如用于高级 MBD 和外部数据包报告日志记录的外部机器学习模块。图 2 和图 3 总结了这个想法,同时显示了体系结构的不同模块。这些模块在第 IV-C 至 IV-H 节中有详细说明。

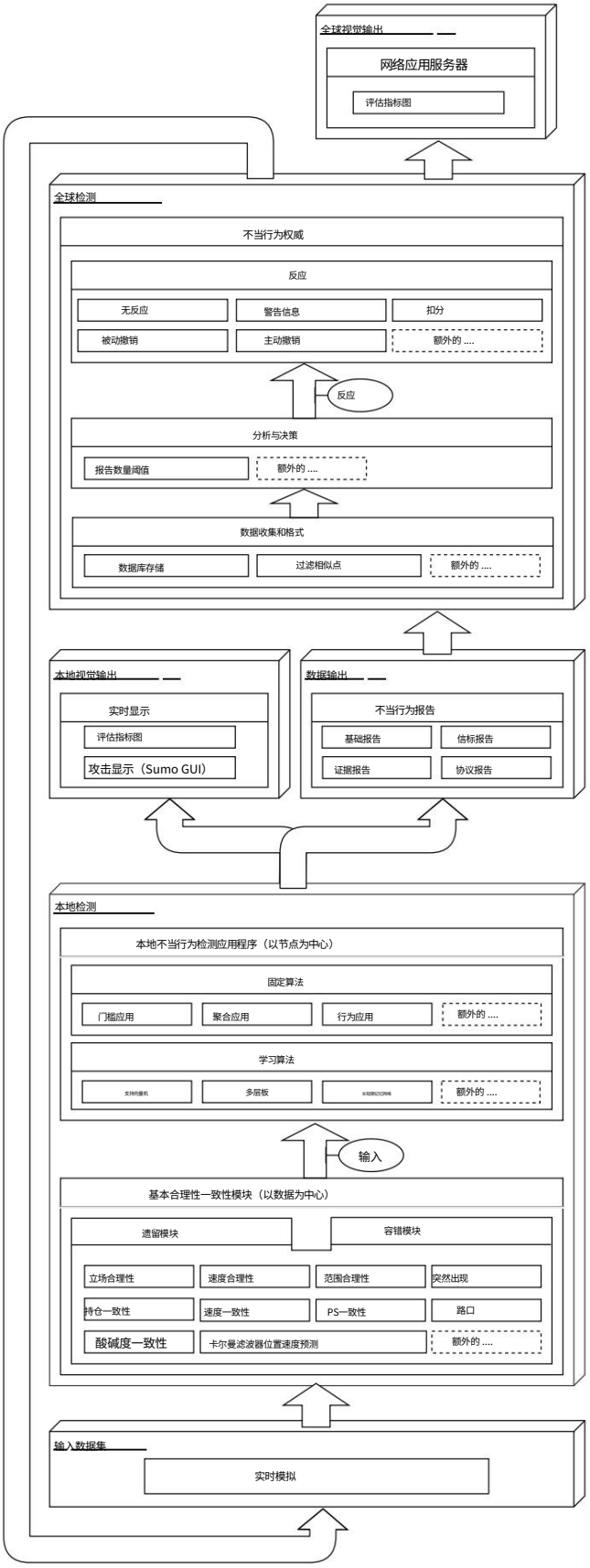


图 2. 主要模块示意图

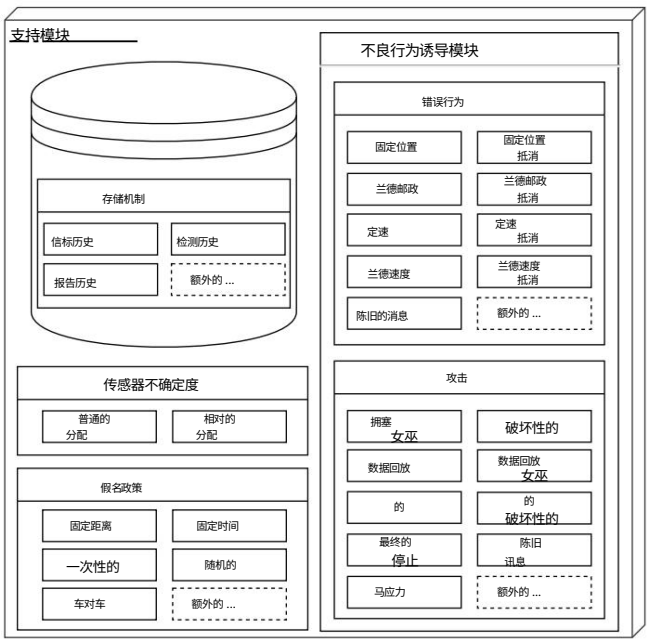


图 3. 次级模块的示意图

B. 框架输入数据

该框架所需的第一个输入是相扑场景。我们确实提供了两种开箱即用的场景。第一个场景来自巴黎-萨克雷地区。该网络结合了一些类似郊区的网格和一些有机网络特性。

这是一个相对较小的网络,我们为其生成可变的车辆密度。我们将其用作校准和微调的测试台,因为它可以提供快速且可预测的结果。

第二种情况是 LuST SUMO 网络 [12]。这个场景是一个基于人口普查数据和卢森堡真实交通信息的SUMO网络。这种情况运行起来要慢得多,但对最终模拟结果更有价值。除了 SUMO 场景外,VEINS 模拟还需要 OMNeT++ 配置。 OMNeT++ 配置包括信标参数,例如标头位长度和信标间隔。它还包括网络接口卡 (NIC) 设置,例如 txPower、比特率、Recall 和 thermalNoise。我们强烈建议将这些设置保留为 VEINS 默认值。这些值可以在我们的 github 上的 OMNeT++ 配置文件中找到。

最后,我们的框架需要特定的输入,如攻击类型、攻击者密度、报告格式和 PCP。

此列表并不详尽,但给出了预期输入类型的一般概念。一些输入甚至特定于攻击类型或 PCP。例如,如果我们选择周期性的假名变化,那么我们必须设置平均变化周期。我们的 github 中包含完整的输入列表。

C. 局部检测

在这个框架中,我们为本地 MBD 提供了丰富的模块。该模块提供了使用简单方法自定义和测试不同算法的简单方法。

本地检测逻辑如下。系统运行

对每条收到的消息进行基本的合理性和一致性检查。结果被传输到本地的不当行为应用程序,该应用程序决定是否向 MA 发送报告。因此,局部检测可以在两个位置进行定制:基本合理性(通常称为检测器)和更智能的检测应用(通常称为数据融合)。为此,我们实施了多个版本的基础检查和多个不良行为应用程序。我们还提供了一种基于实时机器学习的 MBD 应用程序的方法。

1) 合理性检查:受文献启发,我们提取了一组基本的 MBD 检查。以下检查或检测器在其旧版本和容错 (ET) 版本中实现。旧版本计算合理性检查的速度要快得多,并返回二进制输出以显示消息的某个方面是否合理。 ET 版本通常计算合理性检查的速度较慢,但会返回一个反映消息不合理性规模的不确定因素 [21]。

以下是已实施的所有本地合理性检查的列表: ·范围合理性:检查发送 ITS 站 (ITS-S) 的位置是否在自我 ITS-S 最大范围内(映射到自我 ITS 的预定义值-S 最大无线电覆盖范围)。 ·位置合理性:检查发送ITS-S 的位置是否在合理的地方(例如在路上,没有物理障碍物重叠等)。 ·速度合理性:检查发送 ITS-S 通告的速度是否低于预定义的阈值。 ·位置一致性:检查来自同一个ITS-S 的两个连续信标是否具有合理的分离距离。

·速度一致性:检查来自同一个ITS-S 的两个连续信标是否具有合理的加速或减速。

·位置速度一致性:检查来自同一个ITS-S 的两个连续信标是否具有一致的速度和间隔距离。 ·信标频率:检查发送ITS-S 的信标频率是否符合标准。 ·位置航向一致性:检查来自同一个ITS-S 的两个连续信标中的位置是否与该ITS-S 通告的航向相对应。 ·交点检查:检查来自两个不同ITS-S 的两个信标是否没有重叠位置(即两个ITS-S 相互重叠)。

·突然出现:检查是否没有ITS-S突然出现在一定范围内出现。
·卡尔曼滤波器跟踪:检查ITS-S 广告信息是否在卡尔曼滤波器预测值的合理范围内。 [22] 中提出了这种检查。

2) 高级不当行为检测:MBD 应用程序是检测逻辑的决策部分。

它们也被称为融合应用程序,因为决策通常基于多个因素的融合(合理性检查的结果、节点历史等)。我们实现了多个简单的例子。他们中的一些人使用

固定算法和其他算法基于人工智能。
固定算法直接在 VEINS 中实现,而学习应用程序在 python 中实现并通过特定的 API 访问。以下是基于固定算法的应用:

·阈值应用程序:如果某个消息未通过至少一项合理性检查,则会报告一个节点。如果检查结果低于特定阈值,则确定失败。

·聚合应用程序:此应用程序基于节点历史记录。某些消息的检查结果与最后 n 个结果聚合。如果聚合结果低于某个阈值,则报告一个节点。 ·行为应用程序:此应用程序基于不当行为事件的严重性。根据不当行为的严重性,节点会被置于超时状态,并将其发送的所有数据报告给 MA。重力是从基本合理性检查的结果中推导出来的。

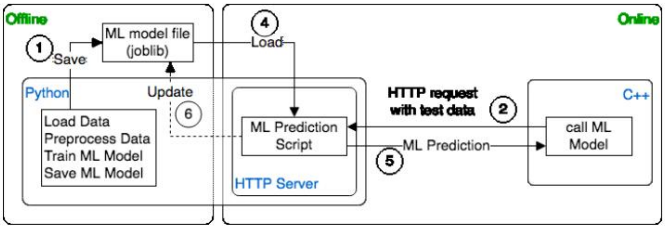


图 4. 用于机器学习建模的 Python-C++ 接口

3) 用于机器学习建模的 Python-C++ 接口:机器学习 (ML) 和深度学习 (DL) 最常用的编程生态系统之一是 Python。我们设计了框架的这一部分,以便任何人都可以使用自己的 ML/DL 模型扩展框架。我们已经开发了框架的 Python 和 C++ 部分之间的接口。该接口允许任何开发人员在 Python 中实现他们的 ML 模型,并让核心框架在模拟期间调用他们的模型。

图 4 表示 Python-C++ 接口设计。该界面是围绕典型的 ML 模型开发过程设计的。如图4所示,界面分为离线和在线两个阶段。所有文件在图中都表示为单独的模块。这意味着一个 ML 模型被分成两个文件。每个阶段一个文件。

在离线阶段,开发人员可以设计、训练和保存她的 ML 模型。我们支持用于 ML 建模的 scikit-learn [23] 和用于保存 ML 模型的 joblib 库 [24]。 joblib 在保存使用 NumPy [25] 的 ML 模型方面比 Python 的内置模块 pickle 效率高得多。

在在线阶段,ML 模型在侦听用户定义端口的 HTTP 服务器 (PyMLServer) 中运行。仿真核心(用 C++ 编写)调用 ML 模型(用 Python 编写)。如图 4 所示,模拟核心在 HTTP 请求中发送要针对 ML 模型进行测试的数据。

PyMLServer 调用 ML 预测脚本,该脚本执行如下操作: ·从 HTTP 请求加载需要进行预测处理的数据。

·加载离线阶段保存的机器学习模型。我们再次使用 joblib 来加载 ML 模型。

·使用来自的预测响应 HTTP 请求
机器学习模型。

·或者,预测脚本可以更新 ML 模型并将其保存回加载的模型。使用反向传播的模型可能需要这样的功能。

最后,模拟核心从 HTTP 响应中读取预测输出以执行进一步调查。基于之前的接口,我们实现了以下技术: ·支持向量机 (SVM) 分类器:一个二元

类 SVM 模型是根据从中提取的特征进行训练的
合理性检查与 [11] 类似,如前几节所述。该 SVM 将正版车辆与行为不端的车辆进行分类,其准确性在很大程度上取决于场景(网络、密度、攻击等)。

对于监督机器学习的扩展结果,我们建议读者参考 [11]。 ·多层感知器(MLP)分类器:基于MLP 的神经网络在与SVM 相同的数据上进行训练。我们发现 MLP 的准确性通常优于 SVM。但是,请记住,技术比较不是本文的目的。 ·长短期记忆(LSTM) 分类器: LSTM 也接受了与SVM 相同的数据训练。

LSTM 是 ML 算法的循环神经网络 (RNN) 系列的一部分,足以处理时间相关数据。 LSTM 的准确性通常是测试算法中最好的。但是,它也是计算速度最慢的算法。

D. 报告

本地 MBD 的一个重要目标是向中央 MA 发送报告以进行后处理。在此框架中,MBD 算法可以决定生成不当行为报告。

报告通过 HTTP 连接推送到全球 MA。
此外,可以在本地文件夹中以 json 或 xml 格式收集报告。

我们提出了一份受 [26] 中描述的协议启发的不当行为报告。报告由三个容器组成:Header Container、Source Container 和Evidence Container。

Header Container 包含每个报告中应该包含的基本信息:生成时间、发送者 ID、报告 ID 和报告类型。 Source Container 包含对报告的信标的合理性和一致性检查的结果,假设车辆仅在接收到的信标显示出一些不合理性后才被报告。证据容器应帮助 MA 进行调查并支持其结论。如果认为有帮助,证据容器可以由来自被报告者或报告者或任何邻近车辆的消息组成。它还可以包括一些其他数据,例如局部动态地图 (LDM),或来自报告者的直接传感器数据。 MA 要求的证据在 [26] 中有进一步的详细说明。

然而,报告格式尚未标准化,仍然是科学研究的主题。为此,我们认为

提供多种格式的不当行为报告有助于进一步调查和测试。该框架实现了以下版本：

· Base Report:这种格式只包括Header Container 和Source Container,没有证据。 ·信标报告:该版本包括一个基础报告和证据容器中报告的信标。 ·证据报告:此版本包含更完整的证据容器,具体取决于合理性检查失败的类型。例如,如果车辆未通过速度一致性测试,我们会将两个不一致的信标作为证据。

在框架的当前状态下,每条标记为行为不当的消息都会被报告。但是,并非每条消息都应单独报告。这将产生显着的网络开销,尤其是当车辆由于其系统中的故障组件而行为不当时。

因此,报告格式允许省略报告,这意味着不会经常报告车辆的相同行为。取而代之的是,当地车辆在收集证据的同时避免在一定时间后报告相同的行为。然后将证据以一份报告的形式发送给 MA。该协议假定智能 MA 优先考虑接收到的报告的内容而不是它们的数量。

E. 全局不当行为检测

不当行为管理局 (MA) 是接收车辆发送的报告的全球实体。然后 MA 应该决定做出合适的反应。我们定义了 MA 的三个主要组成部分： ·收集和格式:将收集的报告添加到数据库中。此操作将允许使用特定条件访问报告。例如,我们可以得到所有指控某个假名的报告或某个地区的所有报告。这些请求可能有助于分析阶段。我们还有一个过滤系统,如果启用,它可以聚合所有发出相同信号的报告（例如,速度不一致的两条消息的集合）。 ·分析和决策： MA 分析报告并输出正确的反应水平。我们实施了一种简单的非智能方法,该方法对每个反应级别的报告数量都有一个阈值。达到每个级别所需的报告数量是可以修改的。我们将输出设置为水平,以便它与我们的反应机制兼容。但是,可以开发其他产出。请注意,此组件将在框架的未来版本中演变为更复杂的元素。

·反应:不当行为反应仍然是一个广泛争论的话题。我们提出了一个基于级别的解决方案,具有 5 个级别的反应：

– 0 级:无反应– 1 级:向车辆发送警告信息– 2 级:车辆的警告点被扣除分数–级别 3:车辆无法请求更多证书的被动撤销–级别 4:当前证书的主动撤销该车辆被撤销。

目前,反应不会导致车辆行为发生变化,但是我们希望有一个更智能的系统,车辆会根据反应水平改变其行为（例如,传感器有故障的车辆会重新校准来自 MA 的警告）。

F. 不当行为机制

为了评估不同检测方法的质量,我们需要在系统中产生不当行为。出于这个原因,我们实施了两类不当行为:错误行为和攻击。错误行为作用于一个传感器数据,而攻击是更复杂的方案。该框架插入了预定义百分比的行为不端车辆。这些车辆可能都表现出一种类型的不当行为,也可能混合多种类型。下面介绍了所有已实施的不当行为机制的详细信息。

1) 错误行为:每辆车都应包括对数据的车载处理,以确保传输前的合理性。然而,这种预防性系统可能缺乏一些用例并且容易失败,尤其是在经济型车辆的情况下。在这里,我们考虑这种车载数据预处理系统出现故障的情况。我们从文献中提取了一组可能的错误行为 [27]。框架中实现了以下集合：

·固定位置:车辆广播相同位置 (X,Y) 每个信标。
·固定位置偏移:车辆广播其真实具有固定偏移量 (ΔX,ΔY) 的位置。
·随机位置:车辆随机广播从操场的位置。
·随机位置偏移:车辆广播其真实位置,随机偏移限制为最大值(Δ(0 → Xmax),Δ(0 → Ymax)。

·固定速度:车辆向每个信标广播相同的速度(Vx) 。 ·固定速度偏移量:车辆以固定偏移量(ΔVX) 广播其实际速度。 ·随机速度:车辆广播随机速度有上限 (0 → Vmax) 。
· Random Speed Offset:车辆广播其真实速度,随机偏移限制为最大值（Δ（0 → Vmax））。 ·陈旧消息:车辆在添加预定义延迟 (Δt) 后广播其真实信息。

2)攻击:我们的攻击方案复杂程度各不相同。以下列表详细说明了我们框架中当前实现的内容：

· DoS:为了拒绝其他车辆访问网络,攻击车辆将信标频率增加一定系数。在我们的实现中,车辆还可以选择发送有效消息或随机数据。攻击者还可以选择更多

为了避免被发现,经常在他预先加载的假名之间更改和交替。

·破坏性:这种攻击旨在通过用旧信标数据淹没网络来破坏系统。攻击者从接收到的历史中选择一个随机信标并重播其数据。同时,攻击者能够增加信标频率以最大化效果。

值得注意的是,这些数据最初是由真实车辆生成的,因此在某些层面上是合理的。结果,这种攻击严重恶化了 C-ITS 的质量。与 DoS 攻击类似,攻击者也可以选择在他预先加载的假名之间进行切换。

·数据重放:攻击者选择一个目标并以一定的延迟重放其数据。因此,对于观察者来说似乎有两辆车紧随其后。攻击者可以在更换目标车辆时选择更换假名以避免被发现。 ·最终停止:经过一定的随机延迟后,攻击者停止更新信标位置并将速度设置为零,从而模拟突然停止。 ·拥塞女巫攻击:一种类型的女巫攻击包括生成幽灵车辆。为此,执行以下操作: –根据攻击车辆或选定目标车辆的数据计算幽灵车辆的位置、速度和航向。

–生成并维护一个假名列表,一个
每辆幽灵车的化名。
–攻击者的信标频率根据幽灵车辆的数量增加。

–幽灵车辆信标是多路复用的,因此每辆车每个周期发送一个信标。

这种攻击展示了使用该框架 (i) 操纵假名,(ii) 即时增加信标频率以及 (iii) 智能计算数据以服务于特定目标的能力。

· MA Stress:这种攻击不针对本地车辆。
相反,它通过发送大量虚假报告来针对全球实体(即 MA)。这些报告包含攻击者附近车辆的身份。攻击者可以选择为每个报告更改其身份。攻击者还可以增加向 MA 发送报告的频率。

G. 隐私

IEEE 和 ETSI 标准 [16]、[19]、[28] 中已包含假名的使用。然而,何时以及如何发生假名更改仍然是一个研究挑战。科学研究提出了多种方法来确定假名的位置和变化率 [29]。

以节点为中心的 MBD 机制依赖于处理过的车辆的一致身份。这种方法受到基于假名的隐私保护机制的极大影响。为此,我们在我们的框架中实施了以下假名变更政策 (PCP):

表一
阳性/阴性的定义

	行为不端	真的
检测到	真阳性 (TP)	误报 (FP)
未检测到的假阴性 (FN)	真阴性 (TN)	

·定期:车辆在预定义的时间段后更改其假名。 ·距离:车辆在预先确定后更改其假名

定义的公里数。

·一次性:假名用于固定数量的消息(包括信标和警告)。 ·随机:假名有预定义的机会

更改每条发送的消息。

H. 评估和可视化

在给定的场景中,传输消息的车辆可能是行为不当的或真实的,并且本地检测机制可以将消息分类为行为不当或真实的。因此,如表 I 所示,检测机制的评估首先要确定什么是真阳性/阴性,以及假阳性/阴性。

Recall (1) 衡量正确识别的错误信息在所有接收到的错误信息中所占的比例。 P recision (2) 测量在所有标记的消息中正确标记为行为不当的消息的比例。 F1score (3) 是 Recall 和 P recision 的调和平均值。如果我们对 Recall 和 P recision 赋予相同的重要性,它可以用作评估系统性能的单一指标。如果需要,我们可以通过计算 Fβscore 将更多权重分配给一个指标。

这个指标可能很有趣,因为在某些情况下,召回率比精确率更重要。

准确度 (ACC) (4) 是正一致率,在我们的例子中指的是系统中真实检测的比率。博彩公司知情度 (BM) (5) 描述了做出明智决策的概率。它显示了这个系统的决定比随机猜测好多少。

标记 (6) 是通过分类而不是偶然确定检测的概率。马修斯相关系数 (MCC) (5) 是知情度和标记度的几何平均值。当被测量的类的大小非常不同时,它特别有用,这通常是C-ITS 攻击者的情况。 Cohen 的 kappa (κ) (8) 是正一致性的度量,类似于准确性,但我们偶然减去一致性。

$$\text{回忆} = \frac{TP}{TP + FN} \quad (1)$$

$$\text{精度} = \frac{TP}{TP + FP} \quad (2)$$

$$\text{F1 分数} = 2 \times \frac{\text{召回率} \times \text{精度}}{\text{召回+精确}} \quad (3)$$

$$\text{累加器} = \frac{TP+TN}{TP + FP + TN + FN} \quad (4)$$

$$\text{体重} = -1 + \frac{TP}{TP+FN} \quad \frac{\text{田纳西州}}{TN+FP} \tag{5}$$

$$\text{马克} = \frac{TP}{TP+FP} + \frac{\text{田纳西州}}{\text{田纳西} + \text{联合国}} - 1 \tag{6}$$

$$\text{中冶} = \sqrt{\frac{TP \times TN+FP \times FN}{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{7}$$

$$k = \frac{\text{行政协调会} - \frac{(TP+FP)\times(TP+TN)+(TN+FP)\times(TN+FN)}{(TN+TP+FP+FN)^{2}}}{1 - \frac{(TP+FP)\times(TP+TN)+(TN+FP)\times(TN+FN)}{(TN+TP+FP+FN)^{2}}} \tag{8}$$

我们框架的目标之一是促进评估任何检测机制或任何可能影响检测率的变化。为了实现这一点,模拟器会在每个时间间隔写入运行机制当前状态的快照。然后,脚本实时解析数据、计算并绘制上述评估指标。为了进一步促进机制的评估和比较,模拟器和脚本支持在同一系统上同时运行两个机制。图 5 显示了在二进制基本合理性检查上运行的阈值应用程序与在容错合理性模块上运行的行为应用程序之间的实时比较。

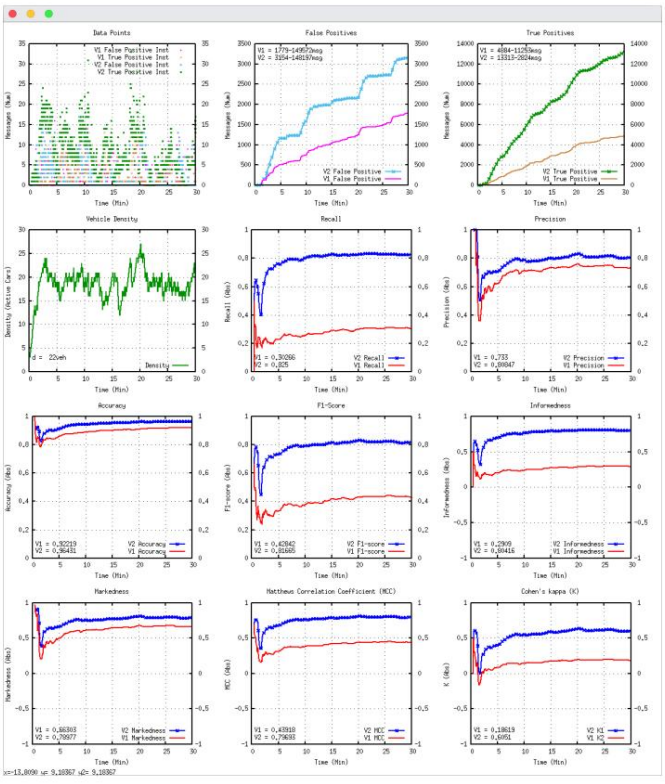


图 5. F2MD GUI:实时评估指标图（数据点、TP、FP、密度、召回率、精度、准确性、F1score、BM、MK、MCC、κ）

然而,可视化不限于检测结果。攻击和检测系统也在 SUMO 的 GUI 中可视化。模拟器使用SUMO的Traffic Control

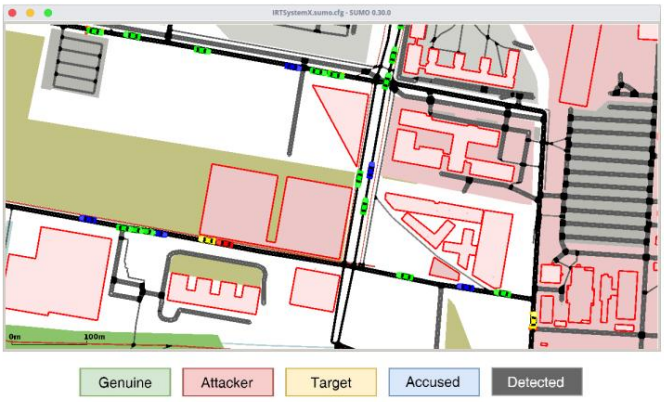


图 6. 带有F2MD 车辆颜色配置文件的Sumo GUI

接口 (TraCI) [30] 根据图 6 中指定的预期角色为车辆着色。

最后,由于 MA 是作为 HTTP 服务器实现的,我们提供了一个 Web 界面。 Web 界面实时运行,并作为 MA 当前状态的不同指标和评估的显示（图 7）。我们目前显示三个指标：

·累积和瞬时预测准确度， ·对于一些最相关的标识符,每个化名收到的报告数量， ·发布的累积百分比的雷达图

反应。

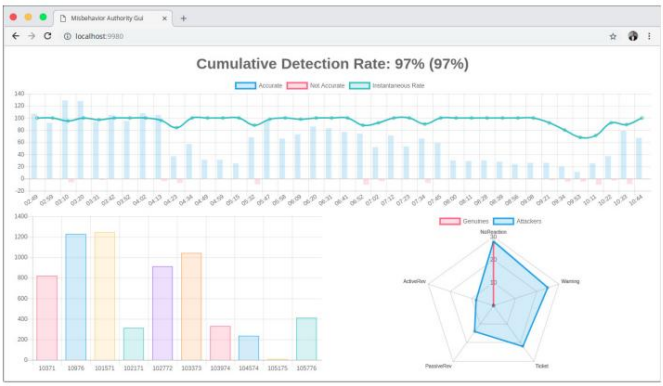
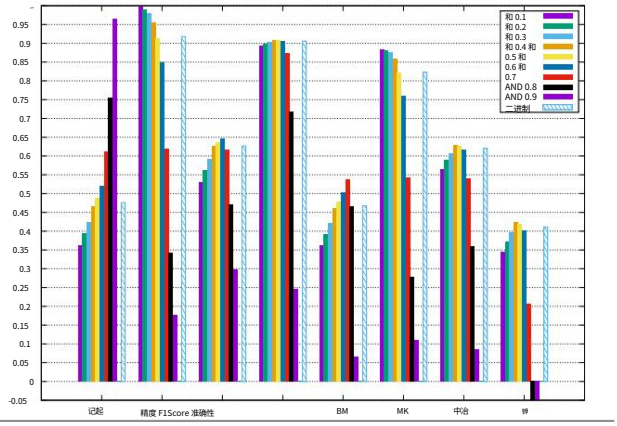


图 7. 不当行为权威 Web 界面

五、例子

为了演示框架的功能,我们运行了多个示例场景。每个示例都基于展示框架不同模块的功能。在以下场景中,我们使用第 IV-B 节中描述的基准测试网络。我们引入密度为 0.1 的行为不端的实体。引入系统的每辆行为不端的车辆都会选择其攻击类型,详见第 IV-F2 节。使用我们的 Github [31] 上提供的实现和场景可以重现以下结果。



行为不端类型	评估指标							
	Rec Pre F1s Acc BM MK MCC K 0.36 1.00 0.53 0.89 0.36 0.88 0.56							
AND 0.1	0.39 0.99 0.56 0.90 0.39 0.88 0.59 0.42 0.98 0.59 0.90 0.42 0.88 0.35							
AND 0.2	0.61 0.47 0.96 0.63 0.91 0.46 0.86 0.63 0.49 0.91 0.64 0.91 0.48 0.37							
AND 0.3	0.82 0.63 0.52 0.85 0.64 0.90 0.50 0.76 0.62 0.61 0.62 0.62 0.87 0.40							
AND 0.4	0.54 0.54 0.54 0.76 0.34 0.47 0.72 0.47 0.28 0.36 0.96 0.18 0.30 0.42							
AND 0.5	0.24 0.07 0.11 0.08 0.48 0.92 0.63. 0.701 0.70							0.42
AND 0.6								0.40
AND 0.7								0.21
AND 0.8								-0.35
AND 0.9								-0.08
二进制								0.41

图 8. 可变阈值评估图

A. 合理性检测器示例

如前所述,我们实现了两个版本的检测器。遗留版本输出一个二进制值,而 ET 版本在不确定的情况下分配一个因子。合理性因素是分配给对消息进行的每个合理性检查的分数。该分数是使用为特定消息中的每个字段公布的值和错误范围计算的。分数在完全不可信 (0) 和绝对可信 (1) 之间变化。

为了更好地理解这个值的影响,我们同时运行两个版本的检测器以及基于阈值的检测应用程序。我们在 0.1 和 0.9 之间改变阈值。正如我们在图 8 中看到的那样,我们可以看到 ET 版本检测器的精度和召回率之间存在明显的权衡。这验证了我们的假设,即不确定性因素量化了某个消息的合理性。

因此,合理性因素为不合理的场景提供了更具信息性的观点。这对于试图检测攻击的智能应用程序可能很有用。因此,对于相同的先前场景,我们训练了一个多层感知器 (MLP) 来对单个消息的合理性检查结果进行分类。表 II 显示 ET 模型比二进制模型有更好的结果。

然而,这种改进是以更长的处理时间为代价的。事实上,二进制检测器比表 III 中所示的 ET 对应物快 8 倍。

B. 本地应用示例

接下来,我们演示 MBD 应用程序模块的功能。我们测试了我们提供的所有应用程序

表二

使用MLP比较ET和双星探测器

行为不端类型	评估指标							
	记录	对于F1		Acc BM MK MCC κ				
二进制	0.48	0.92 0.63 0.93		0.91	0.48	0.83	0.63	0.42
和Δ	0.64	0.76		0.93	0.63	0.86	0.73	0.56
	+31% +1% +19% +3% +32% +4% +17% +32%							

表三

使用二进制或ET的MLP的处理时间

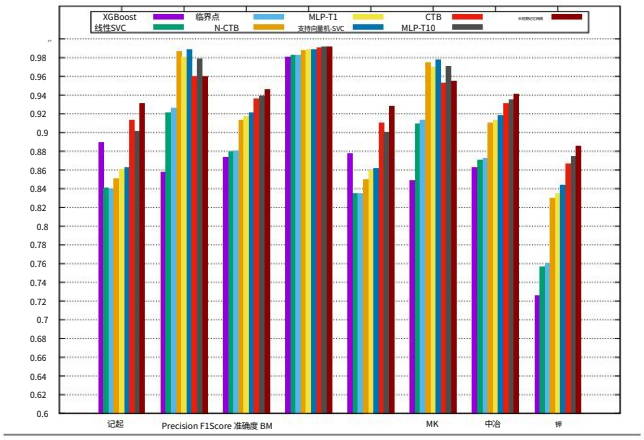
检测器版本二进制时间(µs) 17.117	和
140.581	

框架。我们的固定算法应用程序:阈值、聚合和行为分析。我们的机器学习模型:SVM 和 MLP。合理性检查的结果用作学习模型的输入特征。我们还创建了具有多个合理性检查结果作为特征的模型。

我们将用于创建特征的检查次数称为:模型的深度。我们在这里展示了用深度为 5 和 20 条消息训练的模型,相应地用 D5 和 D20 标记。

图 9 显示了不同检测场景的评估指标。这些指标是从第 IV-H 节中描述的 GUI 图中收集的。

尽管检测率很重要,但本地应用程序也必须占用资源。可以说,本地检测应用程序的主要目标是提醒全球实体。这意味着可以考虑以较低的检测率换取更快的处理。为此,我们还提供了每个申请的平均处理时间。前面场景的申请处理时间如表四所示。查看评估指标和处理



行为不端类型	评估指标							
	Rec Pre F1s Acc BM MK MCC K 0.35 0.99 0.52 0.89 0.35 0.88 0.56							
聚集	0.34 0.49 0.91 0.64 0.91 0.48 0.82 0.63 0.42 0.51 0.95 0.66 0.91 0.50							
苏黎世	0.86 0.66 0.46 0.64 0.93 0.76 0.93 0.63 0.86 0.73 0.56 0.70 0.94 0.89							0.94
支持向量机	0.69 0.88 0.78 0.63 0.74 0.97 0.84 0.95 0.74 0.92 0.82 0.69 0.78 0.98							0.85
MLP D1	0.95 0.77 0.90 0.83 0.70							
MLP D5								
MLP D20								
行为								

图 9. 本地应用程序的检测评估

表IV
平均处理时间

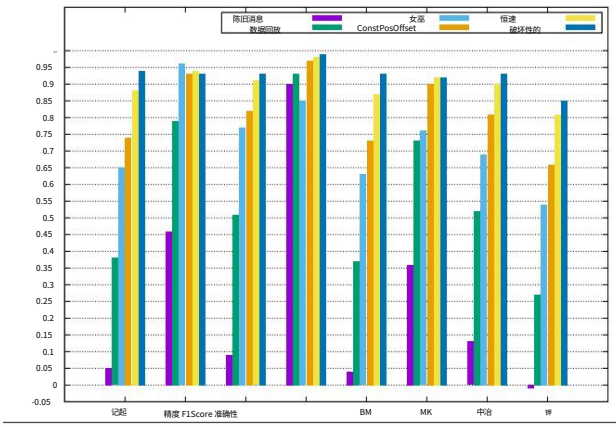
App Version	Aggre	Thre	SVM	MLP	行为
时间（微秒）	4.86		1.13	373.1	233.7 1.99

时间,可以得出结论,行为分析应用程序是总体上最好的选择。此应用程序为每个假名分配一个信任值并随时间更新此值。

从理论上讲,这种方法会受到更改假名的强烈影响。但是,此处测试的场景禁用了假名更改。

C. 攻击示例

到目前为止,我们的评估是基于不当行为类型的均匀组合。然而,检测率在很大程度上取决于不当行为的类型。为了证明这一事实,我们测试了使用 20 条消息深度训练的 MLP 检测单个攻击的能力。图 10 显示,我们的检测器很难检测到陈旧消息的不当行为类型。这个结果符合我们的预期,因为这里传输的消息是合理的,只是在几秒钟后发送。因此,陈旧的消息没有很多难以置信的特征。应该注意的是,SAE J2945/1 [15] 指定 BSM 在过去（和将来）生成 30 秒通过时间相关性检查。另一方面,我们看到破坏性攻击是最容易检测到的。这是因为这种类型的攻击会重播大量没有数据一致性的消息。因此,有大量难以置信的特征很容易被中性网络识别出来。



行为不端类型	评估指标							
	Rec	Pre	F1s	Acc	BM	MK	MCC	κ
StaleMessages	0.05	0.46	0.09	0.90	0.04	0.36	0.13	-0.01
DataReplay	0.38	0.79	0.51	0.93	0.37	0.73	0.52	0.27
Sybil	0.93	0.82	0.87	0.96	0.86	0.77	0.78	0.54
DataReplaySybil	0.93	0.85	0.89	0.97	0.87	0.79	0.80	0.66
ConstPos	0.93	0.88	0.90	0.98	0.87	0.81	0.81	0.81
ConstPosOffset	0.93	0.91	0.92	0.98	0.87	0.83	0.83	0.83
破坏性的	0.94	0.89	0.91	0.98	0.88	0.82	0.83	0.83
中性	0.98	0.87	0.92	0.90	Disruptive	0.94	0.93	0.93	0.99	0.93	0.92	0.93	0.85

图 10. 攻击检测评估

D. 报告和全局检测示例

最后,我们演示了框架的报告和全局检测。我们首先检查不同的

表五
平均报告大小比较

报告类型	报告大小（字节）			
	未压缩	lzma	压缩512.45	313.27 1090.00 479.73
根据灯塔证据	1979.84	545.89		

类型的报告。我们不同的报告格式有不同的容器,大小应该不同。使用与 VC 节中相同的场景,我们测量以 JSON 格式传输到 MA 的报告的大小。此测量是针对每个

第 IV-D 节中描述的格式。表 V 显示了

不同报告的平均大小。可以看到大小和报表格式是一致的,信息压缩后差异更小。请注意,尺寸仅用于比较目的,但是,这些尺寸并不与真实的最终报告成比例。欧洲电信标准协会 (ETSI) 和电气和电子工程师协会 (IEEE) 的所有 C-ITS 消息都有一个安全报头、一个签名和多个其他层,这些层未包含在此处 [13]、[17]。对于全局检测,我们目前的简单 MA 仍然仅基于某个假名收到的报告数量。因此,PCP 会显着影响结果。为了显示这种效果,我们启用了一个周期性的 PCP。

然后,我们模拟具有多个假名更改期的相同场景。图 11 显示了 MA 收到的每个假名的真实节点和行为不端节点的平均报告数。我们可以看到报告的数量受到更频繁的假名更改的显着影响。

事实上,MA 没有收到足够的具有相同报告 ID 的报告,这肯定会影响检测质量。

为了减轻这种影响,MA 可以分析报告的内容。带有 Evidence Container 的报告有利于此过程。此外,目前提议的报告协议不会为每条消息发送报告 [26]。

相反,为了减少开销,车辆会收集证据,然后发送更完整的报告。此过程将使基于报告数量的全局检测变得过时。

因此,对智能 MA 的需求至关重要。

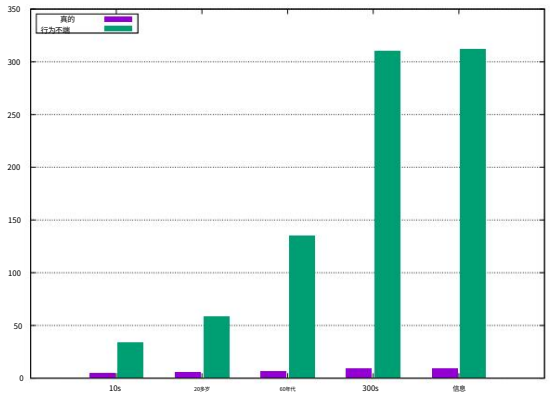


图 11. 不同假名变更期间 MA 收到的假名报告的平均数量

锯。结论

协作式智能交通系统容易受到虚假数据注入攻击,可能危及道路使用者的安全。在本文中,我们提出了一个名为F2MD 的仿真框架和源代码,使研究社区能够开发、测试和比较 MBD 算法。

我们在框架中实现了 (i) 全面的攻击列表,(ii) 一组广泛的基本和高级检测算法,(iii) 允许导入人工智能算法的 Python/C++ 桥,(iv) 基本的假名更改策略, (v) 用于分析 MBD 系统实时性能的可视化工具,以及 (vi) 不当行为权限和不当行为报告格式。我们通过运行示例场景展示了它的全部功能并讨论了它们的结果。

作为未来的工作,我们计划扩展假名更改策略（例如添加静默期 [32]）、攻击和检测器。该框架还将通过智能不当行为权威进行扩展,以评估全球不当行为调查（以及不当行为报告政策及其报告格式的影响)和 MA 的检测性能。我们还将通过考虑勾结攻击者和错误的不当行为报告注入来扩展攻击者模型。还考虑添加公钥基础设施模块。

七。致谢

这项研究工作是在以下框架下进行的
Techno logical Research Institute SystemX,因此在法国 Program Investissements d avenir 的范围内获得了公共资金。

参考

[1] SAE International，“SAE J3016 201806:道路机动车辆驾驶自动化系统相关术语的分类和定义”,标准,2018 年 6 月。

[2] J. O Hara，“怀俄明州 CV 飞行员”,https://goo.gl/FqvJHn。

[3] UTMRI，“UMTRI 联网车辆数据集”,https://github.com/caoc scar/ConnectedVehicleDocs。

[4] C. Sommer、R. German 和 F. Dressler，“用于改进 ivc 分析的双向耦合网络和道路交通模拟”,IEEE 移动计算交易,卷。 10,没有。 1,第 3-15 页,2011 年 1 月。

[5] RW Van der Heijden、T. Lukaseder 和 F. Kargl，“Veremi:用于对 vanet 中的不当行为检测进行比较评估的数据集”,第 14 届 EAI 国际通信网络安全和隐私会议论文集,2018 年 8 月.

[6] RW van der Heijden，“VeReMi 数据集”,https://github.com/VeReMi dataset/VeReMi。

[7] RW van der Heijden、S. Dietzel、T. Leinmuller 和 F. Kargl，“协作智能交通系统中的不当行为检测调查”,
IEEE 通信调查和教程,2018 年。

[8] J. Kamel、A. Kaiser、I. Jemaa、P. Cincilla 和 P. Urien，“合作智能交通系统 (c-its) 中不当行为检测机制的可行性研究”,2018 年 IEEE 第 87 届车辆技术会议: VTC2018-2018 年春季。

[9] M. Sun、M. Li 和 R. Gerdes，“用于实现虚假数据检测和安全车辆跟踪的 vanets 数据信任框架”,IEEE 通信和网络安全会议 (CNS),2017 年,第 1 页–9。

[10] C. Zhang、K. Chen、X. Zeng 和 X. Xue，“基于支持向量机和 vanets 证据的 dempster-shafer 理论的不当行为检测”,IEEE Access,卷。 6,第 59 860–59 870 页,2018 年。

[11] S. So、P. Sharma 和 J. Petit，“集成合理性检查和机器学习以检测 VANET 中的不当行为”,IEEE 第 17 届机器学习与应用国际会议 (ICMLA),2018 年。

[12] L. Codeca、R. Frank 和 T. Engel，“卢森堡相扑交通（欲望)情景:24 小时的车辆网络研究机动性”,2015 年 IEEE 车辆网络会议 (VNC),2015 年 12 月,第1-8。

[13] SAE International，“SAE J2735 201603,专用短程通信 (DSRC) 消息集字典”,标准,2016 年。

[14] ETSI，“ETSI EN 302 637-2 V1.4.0 (2018-08):智能交通系统（ITS）;车载通讯;基本应用程序集;第 2 部分:合作意识基本服务规范”,标准,2018 年 8 月。

[15] SAE International，“SAE J2945/1,V2V 安全通信的车载系统要求”,标准,2016 年。

[16] IEEE，“车辆环境中无线访问的 IEEE 标准 - 应用程序和管理消息的安全服务 - 修正案 1”,
IEEE Std 1609.2a-2017（IEEE Std 1609.2-2016 修正案）,第 1-123 页, 2017 年 10 月。

[17] ETSI TC ITS，“ETSI TS 103 097 V1.3.1 - 智能交通系统（ITS）;安全;安全标头和证书格式”,标准,TC ITS,2017 年 10 月。

[18] B. Brecht, D. Therriault, A. Weimerskirch, W. Whyte, V. Kumar, T.
Hehn 和 R. Goudy，“用于 v2x 通信的安全凭证管理系统”,IEEE 智能交通系统交易,第 1 期。 99,第 1-22 页,2018 年。

[19] ETSI，“ETSI TR 103 415 V1.1.1:智能交通系统（ITS）;安全;假名变更管理预标准化研究，”
2018 年 4 月。

[20] “不当行为检测框架 (F2MD)”,2019 年,[在线;于 2019 年 2 月 28 日访问]。 [在线的]。可用:https://www.irt systemx.fr/F2MD/ [21] J. Kamel、A. Kaiser、I. Ben Jemaa、P. Cincilla 和 P. Urien，“CaTch:一种可容忍不当行为的置信范围检测方法”,2019 年 IEEE 无线通信和网络会议 (WCNC) (IEEE WCNC 2019),摩洛哥马拉喀什,2019 年 4 月。

[22] A. Jaeger、N. Bißmeyer、H. Stubing 和 SA Huss，“车辆自组织网络中高效移动数据验证的新框架”,
国际智能交通系统研究杂志,第一卷。 10,没有。 1,第 11-21 页,2012 年 1 月。

[23] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O.
Grisel、M. Blondel、P. Prettenhofer、R. Weiss、V. Dubourg、J. Vanderplas、A. Passos、D. Cournapeau、M. Brucher、M. Perrot 和 E. Duchesnay，“Scikit-learn:机器学习在 Python 中，”机器学习研究杂志,卷。 12,第 2825–2830 页,2011 年。

[24] G. Varoquaux，“joblib 文档”,https://media.readthedocs.org/pdf/joblib/latest/joblib.pdf,2018 年。

[25] TE Oliphant，“numpy 指南”,http://www.numpy.org/,2006 年。

[26] J. Kamel、IB Jemaa、A. Kaiser 和 P. Urien，“C-ITS 的不当行为报告协议”,2018 年 IEEE 车载网络会议 (VNC),台湾台北,2018 年 12 月。

[27] J. Petit 和 R. Ansari，“V2X 验证工具”,https://bitbucket.org/onboardsecurity/dsrcvt,BlackHat 2018。

[28] ETSI，“ETSI TS 102 940 v1.3.1:智能交通系统（ITS）;安全; ITS 通信安全架构和安全管理”,2018 年 4 月。

[29] J. Petit、F. Schaub、M. Feiri 和 F. Kargl，“车辆网络中的假名方案:调查”,IEEE 通信调查和教程,卷。 17,没有。 1,第 228–255 页,2015 年。

[30] A. Wegener、M. Piorkowski、M. Raya、H. Hellbr ´ uck、S. Fischer 和 J.-P。 Hubaux，“Traci:耦合道路交通和网络模拟器的接口”,第 11 届通信和网络模拟研讨会论文集,ser。中枢神经系统 08。美国纽约州纽约市:ACM,2008 年,第 155–163 页。

[31] J. Kamel，“F2MD Github 存储库”,https://github.com/josephkamel/F2MD,2019 年。

[32] W. Xin、HM Moonam、J. Petit 和 W. Whyte，“实现隐私与安全之间的平衡:用于评估基于沉默的假名更改方案的微观模拟框架”,Transportation Research Record,卷。 2673,没有。 2,第 71-84 页。

Joseph Kamel是 Tel´ ecom ParisTech（信息处理和通信实验室)和技术研究 SystemX 研究所的博士候选人和研究员。目前，他正在研究智能交通系统（ITS）

安全合作自治 (SCA) 项目中的安全性。他的研究主要集中在联网车辆的网络安全,特别是合作 ITS 中的不当行为检测。

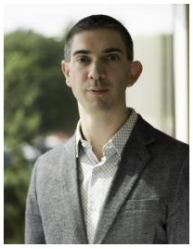
Joseph 拥有法国 Ecole des Mines de Saint-Etienne 的网络物理社会系统硕士学位和科学与执行工程硕士学位。

Ines Ben Jemaa 博士自 2016 年 9 月起担任 SystemX 的研究工程师。她于 2014 年 12 月获得 Mines ParisTech 和 INRIA 的车辆通信博士学位。在加入 SystemX 之前,她曾在 IFSTTAR 研究所担任博士后,并在凡尔赛圣康坦伊夫林大学担任讲师。 Ines Ben Jemaa 是多个国际会议和研讨会的技术程序委员会成员。

她的主要兴趣领域包括 VANET 的路由协议、多播通信、安全性以及隐私和物联网协议。

Mohammad “Raashid” Ansari是 Qualcomm Technologies Inc. 的高级研究工程师,从事系统安全方面的研究。他目前的研究重点是开发联网车辆的不当行为保护系统。他建立了概念验证来分析联网车辆的网络安全机制。他在 2018 年的 BlackHat 和 2017 年的 DARPA SDR Hackfest 上展示了用于联网车辆的黑客工具。Raashid 拥有美国新罕布什尔大学的电气工程理学硕士学位。他的论文

专注于车载网络的安全。



Jonathan Petit 博士是 Qual comm Technologies, Inc. 的首席工程师,负责系统安全方面的研究。特别是,他的研究重点是联网和自动驾驶车辆,以设计一个全面的不当行为保护系统。2013 年,Petit 博士率先发表了关于自动驾驶汽车网络安全的文章,然后在 2015 年对传感器进行了攻击。 Petit 博士拥有法国图卢兹第三大学 Paul Sabatier 计算机科学博士学位。

Pascal Urien是巴黎电信的全职教授。
他毕业于 Ecole Centrale de Lyon（1980 年)并获得计算机科学博士学位。他的主要研究兴趣包括网络安全和安全元素,尤其是分布式计算架构。他在这些领域拥有十五项专利和一百多篇出版物。

Arnaud Kaiser 博士于 2011 年在巴黎第 13 大学获得计算机科学博士学位。随后,他加入了法国原子能委员会的通信系统实验室,担任研究工程师直至 2014 年。他从事 IPv6 车辆网络和能源-IPv6 家庭网络的效率。随后,他加入了法国 SYSTEMX 技术研究所,担任合作智能交通系统 (C-ITS) 网络安全研究工程师。

自 2018 年 9 月起,他担任安全合作自治系统 (SCA) 研究项目的项目经理,该项目专注于 C-ITS 的不当行为检测。 Arnaud 指导从事 C-ITS 网络安全和隐私研究的博士生。他在主要的 IEEE 会议和期刊上发表了他的作品,并作为许多会议和期刊的审稿人积极参与社区活动。他还致力于与 C-ITS 相关的 ETSI 标准化活动。