黑块记录器:不可变的黑匣子通过区块链为机器人记录日志

鲁芬·怀特, 詹卢卡·卡亚扎,奥古斯丁·科尔特西、Young Im Cho 和 Henrik I. Christensen

摘要:事件数据记录在机器人研究中至关重要,它能够提供对机器人情境理解、行为状态演变及其结果的长期洞察。此类记录在调试复杂的机器人应用程序或事后分析实验时非常宝贵。随着机器人技术发展成熟并投入生产,事件日志的作用和要求都将不断扩展,包括为审计人员和监管机构调查事故或欺诈行为提供证据。鉴于涉及自动驾驶汽车的重大公共事件日益增多,导致人员死亡,并影响到监管政策的制定,维护此类事件日志的完整性、真实性和不可否认性,以确保问责制至关重要。作为移动信息物理系统,机器人带来了超越传统IT的新威胁和漏洞:无人监管的物理系统访问或机器人与原始设备制造商(OEM)之间的事后串谋可能会导致先前记录被截断或篡改。在这封信中,我们通过完整性证明和分布式账本解决了日志记录的不变性问题,并特别考虑了移动和公共服务机器人的部署。

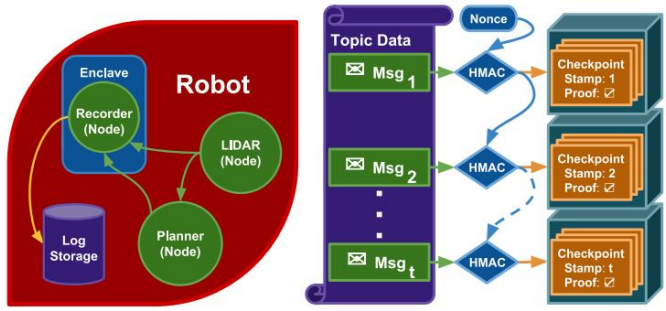


图 1. 不可变日志的高层概述。左图展示了一个示例部署,其中一个封闭进程通过直接从各个源捕获消息流量来生成日志。在将日志数据流式传输到任意存储设备的同时,通过向外部区块链提交跨度检查点(由链接的完整性证明组成,这些证明被索引为检查点交易,如右图所示),使数据不可变。

索引词 机器人安全、联网机器人、软件中间件、加密机器人、分布式账本。

一、引言

作为信息物理系统(CPS),正在日益机器人作为网络基础设施的一部分进行部署,并与物联网 (IoT) 紧密相连。

在互联生态系统中使用机器人绝非易事。
我们如何确保这些机器人的安全?如何验证操作是否正确?如何记录操作以便于追溯,或在发生故障时记录操作?

分布式机器人系统的设计、部署和验证的一个重要部分是监控和记录运行时事件数据的能力。

鉴于人们对自动驾驶汽车和自主无人机的广泛兴趣,部署现实世界“蜜罐”的可能性令人担忧[1]。考虑到近年来汽车漏洞利用的历史[2],毫无疑问,缺乏安全性构成了真正的威胁[3]–[6]。

稿件收到日期:2019年2月24日;接受日期:2019年6月21日。出版日期:2019年7月15日;当前版本日期:2019年8月1日。副主编A. Dietrich和主编P. Rocco在评估了审稿意见后,推荐发表此函。(通讯作者:Ruffin White。)

R.White 和 HI Christensen 就职于加利福尼亚大学圣地亚哥分校,美国加利福尼亚州圣地亚哥市 92093(电子邮件:rwhitema@ucsd.edu;hichristensen@ucsd.edu)。

G. Caiazza 和 A. Cortesi 来自威尼斯 Ca Foscari 大学,意大利威尼斯 30123(电子邮件:840009@stud.unive.it;cortesi@unive.it)。
YI Cho 就职于韩国首尔嘉泉大学,地址:首尔 461-701(电子邮件:yicho@gachon.ac.kr)。

数字对象标识符 10.1109/LRA.2019.2928780

记录由多个组件组成的集成机器人系统的运行情况,并生成全面的运行轨迹,对于质量控制、调试、系统验证等至关重要。对于调试信息流或机器人出现意外行为的情况,事件记录至关重要。然而,当机器人信息物理系统的绝对安全性无法得到保证时,此类事件日志的准确性就会变得脆弱。

数字取证调查 (DFI) [7] 使用数字日志作为事件后分析或电子设备入侵检测系统 (IDS) 的证据。通过持续广播系统状态的简化加密承诺,设备始终处于“达摩克利斯之剑”的威胁之下,这通过在不依赖特定硬件的情况下强制问责来激励诚实。考虑到移动机器人的大规模制造限制,包括材料制造、可维护性、海量数据速率,以及使用小容量、高成本的防篡改存储设备(例如一次写入多次读取 (WORM) 存储器),在经济上将无利可图。

此外,考虑到移动机器人平台的局限性,例如:计算能力、网络带宽和机载能源容量有限,大量传输加密日志不仅在技术上不切实际,而且违反了国际数据保留政策。正如 Veitas等人[8] 所详述的,虽然以共享服务器为中心的数据保留因其简单的架构而受到原始设备制造商的青睐,但它也与政府监管机构和隐私倡导组织存在直接冲突。

因此,我们的目标是在以下威胁下验证机器人事件数据的完整性、真实性和完整性:

恶意/错误的插入、遗漏或替换。为此,我们探索了基于加密链接完整性证明的事件数据记录器 (EDR) 的应用,并通过分布式账本进行传播,如图 1 所示。

在这封信中,我们介绍了黑块记录器 (BBR),这是一种结合使用数字签名算法 (DSA)、密钥哈希消息认证码 (HMAC) 和通过分布式账本技术 (DLT) 的智能合约 (SC) 来实现防篡改日志记录的方法,同时考虑到移动机器人部署可用的资源有限。

这封信的结构如下:第二部分(相关工作)讨论了关于不可变和防篡改日志、分布式账本技术的文献,以及现有方法在机器人中的局限性。第三部分(EDR 角色、要求和原语)讨论了我们想要强制执行的属性的细节,以及所提出的框架处理的信任设置。第四部分(方法)阐述了我们框架中实现的完整性证明、智能合约和许可区块链架构,包括设计机制和开发选择。第五部分(实施)讨论了实施的细节,以评估我们提出的框架是否能够针对移动机器人场景进行完整性验证和运行时优化。最后,第六部分(结论和未来工作)讨论了针对移动机器人领域较新的可用共识方法的工作和扩展。

现实世界部署中的实用性和可扩展性。

二、相关工作

首先,我们简要介绍基于代币的账本及其主要属性;然后,我们更详细地讨论分布式账本技术(DLT)、不可变日志、可信计算的概念,以及它们与自主系统事件数据记录的相关性。

基于代币的区块链是一种点对点 (p2p) 分布式账本,其安全性源自公钥加密。

网络中的每个参与者在 Merkle 树 [9] 中都有一个公共地址,例如,由其公钥的哈希值得出,该地址在所有其他参与者中唯一地标识用户。

用户之间的交易通过提供用户的区块链地址、余额转移和最后一个被接受的区块的输出哈希值作为输入来定义。候选交易被签名后在 p2p 网络中广播,并由验证器收集并将它们聚合成区块。当验证器通过解决共识算法的挑战来“挖掘”候选区块时,就会生成一个候选区块,然后它将被提议并添加到先前交易区块的链中。只有在确定提议的分叉是网络中所有交易都有效的最长链后,验证器才会采用该分叉。该方法的安全性由所使用的共识算法的拜占庭容错 (BFT) 来保证,并且依靠破坏共识算法的难度或固有成本来阻止恶意行为者。

不熟悉此 DLT 架构的读者可以参考开创性著作 [10],以获得更深入的介绍。

A. 分布式账本技术

在 DLT 出现之前,水平可扩展的分布式数据库(DDB)通常用于复制记录状态可信存储设备。然而,当依赖 CPS 基础设施进行数据保留时,在设备暂时可用或受损的情况下,审核传统 DDB 更新的完整性可能会演变为一个约束不足的问题。在使用分布式账本技术 (DLT) 传播数据完整性时,可以避免在剩余 DDB 副本之间重建时间顺序变更的事后共识,即使这些副本的凭证可能已被撤销。

例如,比特币[10]提供了一种替代使用可信第三方来处理和调解交易的方式;其主要重点在于即使在相互不信任的验证者的情况下也能引入分布式信任。由此产生的分布式账本包含按时间顺序排列的共识证据,每个参与者都可以轻松审计。

正如 BitFury 和 Garzik 的白皮书 [11][12] 所讨论的,基于区块链的账本在银行和其他金融机构中越来越受欢迎,一些利用区块链的不可篡改性和共识机制来验证交易的应用也在不断发展。然而,公共金融区块链由于交易吞吐量低/有限以及传统工作量证明 (PoW) [13] 共识机制所消耗的能源和机会成本而受限于可扩展性。为了克服这些限制并实施企业级安全机制,通过定义公共和私有分布式账本,出现了一些替代方案。

在公共账本中,提交交易没有任何限制。私有账本则将这些操作限制在预先定义的实体列表中。账本进一步分为许可账本和非许可账本。在许可账本和非许可账本中,充当验证者的对等节点的身份受到限制;例如,白名单公钥。公共非许可账本用于比特币等加密货币;公共许可账本用于控制“认证”验证者;私有许可账本的工作方式类似于企业分布式数据库;私有非许可账本是不可能的。Linux 基金会的 Hyperledger 项目 [14] 甚至出现了更多新颖的账本方法,该项目旨在通过创建开源企业标准库来提高分布式账本的性能。

B. 不可变日志

不可变日志需要强大的防篡改日志记录功能。使用加密函数,我们能够确保日志条目的完整性、真实性和不可否认性。文献中已提出了几种实现不可变日志的方案;通常的思路是结合使用DSA和消息认证码(MAC)来明确地验证日志条目。在异构分布式环境中,可以强制执行问责制[15],并减少可信设备的数量。然而,由于需要中心机构存储和验证日志,因此有必要构建额外的信任链并部署分布式日志存储系统(例如分布式数据库)。使用分布式版本控制实现(例如IPFS [16])也是一个有效的选择。

然而,Merkle DAG 的使用并不包含智能合约等验证机制,而这些机制对于将验证逻辑应用于系统至关重要。

继 II-A 中的讨论之后,考虑到与区块链的相似性及其内在的安全特性,利用比特币提出了一个有吸引力的解决方案 [17]。Snow等人[18] 介绍了 Factom1 如何使用 OP_RETURN 交易在比特币链上分发不可变日志以存储其客户端日志条目。同样,Cucurull等人[19] 讨论了如何在 Scytl2 上逐步保护比特币区块链上的电子投票机结果。然而,加密货币开发者认为这是目前较为可疑的新兴趋势之一,并且滥用 OP_RETURN 来搭载任意数据存储在比特币区块链上 [20]。正如 Matzutt 等人[21] 所讨论的,这种滥用在原始加密货币区块链上存储非金融内容的影响是不可持续的。

另一方面,Sutton等人[22] 沿袭了 Cucurull等人提出的检查点概念,提出了一个使用关联数据的模型,通过构建哈希树（而非不断地将日志哈希值转储到链中)来优化区块链的使用。这种做法非常有必要,因为 OP_RETURN 的滥用会带来诸多弊端,无论是从上述协议角度来看,还是因为会产生交易费用。所有需要在加密货币区块链上发布的交易都需要支付一笔费用,这笔费用将被“销毁”,并从账户中扣除有限的余额。

考虑到多年来比特币汇率的波动增长,很显然这种昂贵的操作对于大规模部署来说是不可行的。

使用区块链存储不可变日志的另一个障碍是区块链的新鲜度属性 [23]。区块链的设计初衷是保留事件的顺序（即弱新鲜度）,但无法保证事件的准确时间（即强新鲜度）。Szalachowski [24] 的工作提供了一种使用中心化第三方的变通方案,但这在某种程度上违背了我们自身对分布式信任和可扩展性的目标。移动机器人可能会自主漫游,超出中心化基站或任何特定邻居基站的网络范围,因此任何约定的时间参考都必须基于分布式共识。

迄今为止,Crosby等人[25] 的一项重要工作在使用分布式账本技术 (DLT) 方面领先于其他许多研究,他们提出了一种基于历史树的高效数据结构,用于防篡改日志记录。虽然使用历史树进行验证效率较高,为 O(log2n),但添加检查点的运行时间不再是常数,而是 O(log2n),而不是哈希列表的 O(1)。因此,考虑到机器人和离线审计基础设施之间计算资源的不平衡,我们的方法选择了哈希列表,因为日志长度方面的开销是常数,同时引入索引以实现审计的并行化。

C. 事件数据记录器

事件数据记录器 (EDR) 在汽车行业已变得普遍,部分原因在于政府安全法规的合规性,以及原始设备制造商 (OEM) 对可保责任和风险管理的激励措施。EDR 类似于航空业的黑匣子记录器,用于在部署期间记录车辆内部和外部数据,例如发动机健康和状态、转向和制动操作以及事故报告（例如障碍物距离或撞击的惯性力）。

在准备全面部署EDR的交通基础设施中,自动驾驶汽车或许是首当其冲的。鉴于EDR所保存的数据普遍且关键,目前行业和监管机构都对其隐私和安全提出了质疑。

Veitas等人的工作包括与这些特定问题有关的两部分系列;第一部分提出了政策扫描[26],一种技术战略设计方法;即制定具体的行动和产品来指导技术采用。

政策扫描 (Policy Scan) 的开发旨在解决特定类型的定义不明确的问题,即观察、分析并将技术发展与政策要求、社会治理和社会期望相结合。第二篇论文 [8] 将政策扫描应用于自动驾驶和智能出行领域,提出了一项提案,旨在使未来自动驾驶汽车在协作式智能交通系统 (C-ITS) 中使用 EDR,使其更易于社会接受且符合法律规定。

在上述基础工作以及 Tau-rer等人[27]的研究基础上,我们设计了 BBR 作为 EDR 实现,这是一种受生物启发的机器人数据安全记录方法,它符合规定的车载数据记录、存储和访问管理要求,同时还可以使用开源机器人中间件和分布式账本软件扩展到通用自主 AI 应用程序。

III. EDR角色、要求和原语

本文中,我们根据前期工作 [8], [27] 中的角色、需求和原语,正式定义了 EDR 系统,以列举我们的设计/实现一致性。粗体术语将在后续演示合规性时引用。

A. 义务角色和观察员

审计员:要求观察方进行调查并验证记录档案。例如监管机构或政府。

保管人:日志内容的义务主体,负责日志保存。例如机器人或自动驾驶汽车 OEM。

所有者:负责确保日志完整性/真实性/机密性的中介方。例如,最终用户或运营商。

记者:一个独立机构,忠实负责记录事件。例如 Trusted Logger 或 Recorder Enclave。

B. 记录、存储和访问要求

R1数据提供条件:需要代表同意对所跟踪的日志资产进行过渡控制的所有者。

R2公平、不扭曲的竞争:信任应该被打破在所有验证者（又称托管人)之间分发和共享。

R3数据隐私和数据保护:日志的共置必须防止监护人以外的人参与。

R4防篡改访问和责任:完整性和身份验证日志的真实性必须来自独立的报告者。

R5数据可用性经济:健康和透明度日志取决于是否授予审计员适当的访问权限。

1https://www.factom.com
2https://www.scytl.com/en

C. 定义的原语和系统属性

P1物理数据源的安全识别:证明
在保管人和报告人信任的设备之间。
P2元数据丰富:日志事件上下文可能相关
给各自的所有人、保管人和报告人。
P3数据交换和消息传递:经过身份验证的加密
用于建立各方之间的安全连接。
P4数据记录和存储:报告保持灵活性
QoS 方面以及资源消耗方面合理。
P5访问管理:权利、义务和授权
各方必须明确定义并可执行。

四、方法

在我们的方法中,在 EDR 中使用 DLT 需要开发两个主要组件:完整性证明和智能合约规范。本节详细介绍了两者的设计及其合理性,以适应移动机器人和开源框架的限制。

A. 增量完整性证明

为了在不影响系统性能或公开披露私有日志内容的情况下保持日志的完整性,就像在 [19] 中一样,我们利用 HMAC [28] 的抗碰撞和抗原像特性,将日志检查点与密钥旋转链接在一起,以适应R3。借用 [19] 中建立的术语,我们定义了一个日志检查点(Chki) ,它通过使用前一个摘要(hi-1)作为密钥字节,从日志消息 (LogMsgi) 计算当前 HMAC 摘要(hi)时,与前一个检查点链接在一起:

$$Chk_i = (i, h_i)\ h_i = HMAC(h_{i-1}, LogMsg_i);$$
$$\text{其中} h_0 \leftarrow \${0, 1}^m \quad (1)$$

为了保护隐私,包含一个随机数作为创世摘要（h0）,以将初始熵注入链接的完整性证明中,确保具有相似起始内容的单独记录不会重复连续证明的相同告密签名。

这与之前的研究有所不同,之前的研究将日志完整性证明与基于代币的区块链和之前的金融交易输出进行卷积,以实现不可篡改。通过将之前的检查点摘要作为 HMAC 的密钥,我们将日志验证简化为简单的哈希链检查任务:即依次遍历日志文件中的 LogMsgi,确保最后链接的摘要与发布到账本中的最终证明相对应,从而满足R4要求。

通过将索引 (i) 纳入检查点,可以细粒度地处理丢失或损坏的日志事件时产生的部分验证或分类差异。如果索引也同样嵌入到日志内容中,大型日志文件的验证就可以轻松并行化,从而加速整个验证过程。

先前的研究,例如 [19]、[22],区分了两种不同类型的检查点条目;第一种是链式证明中的增量链接,第二种是锚点,必须始终发布它才能提交新的密钥,同时公开过期的密钥,以便稍后验证完整性和真实性。我们的检查点方法没有这种区分,因此任何检查点或检查点序列可以立即发布。这确保了最新的检查点始终可以在短时间内提交,或者不必等待之前的交易在全球区块链中完成。

尤其对于机器人应用而言,由于移动计算需要自给自足的能源供应,因此可能会遭遇瞬间断电。需要状态加密 [19] 的完整性证明可能会导致记录器无法恢复,因为先前完成的交易包含对未来临时密钥的承诺,而该密钥必须在下一个检查点时披露。我们的方法允许记录器快速从上次已知的完整性证明中恢复,并从上次中断的地方继续对日志进行检查点操作(P4)。

B.智能合约

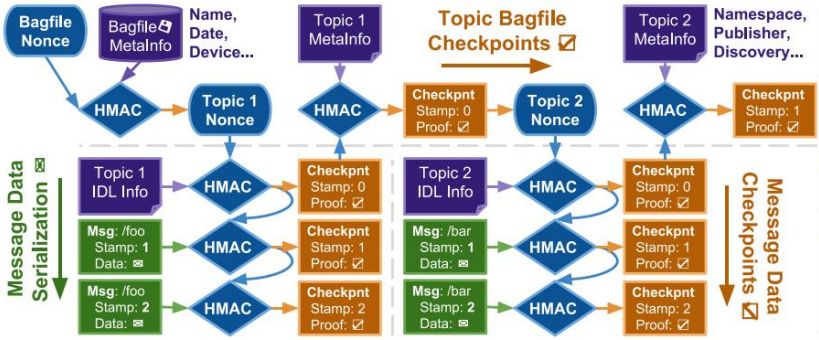
第四章A节形式化了增量完整性证明,以确保日志文件的不可篡改性;然而,这种高效的验证方法本身并不能提供所需的真实性和不可否认性。智能合约（SC)封装了DLT验证者在确定拟议检查点交易的有效性、处理R1并促进R4时必须遵守的访问控制逻辑。

与金融区块链中依赖彩色币或代币元数据来编码所有权不同,我们定义了一个专门的交易族来规范对账本状态的写入访问。然而,一个常见的标准是,候选交易的有效性必须是确定性可计算的;也就是说,在审议过程中不应使用账本当前状态和相关交易有效载荷之外的任何上下文。这确保了链中任何区块的有效性在未来都能被独立验证。

为了确保提交到区块链的检查点的真实性,交易通过椭圆曲线数字签名算法(ECDSA) 进行签名,从而有效地公证签名者的身份。为了实现我们的目的,我们还将身份注册到区块链中,方法是将其公钥注册到存储在分布式账本中的访问控制策略中,以供安全中心 (SC) 在验证候选检查点交易时使用。因此,我们限制记录者的权限,使其只能为预先授权的日志文件(P2) 附加检查点。

为确保交易的不可否认性,我们的安全链 (SC) 要求检查点索引保持单调递增。允许已发布的检查点索引的步幅,以便记录器能够降低完整性证明的传输速率（相对于本地生成的速率）,以此作为服务质量 (QoS),从而节省能源或无线网络带宽,并确保分布式账本状态的可持续规模。账本实际上是每个验证者必须本地维护才能参与的数据库,为了减少账本内存的增长,我们采用分页环形缓冲区来轮换给定日志文件的 n 个最新检查点。环形缓冲区的大小也可以根据R3 的数据保留窗口要求进行分配。然而,创世摘要始终会被保留,以确保整个日志在其第一条记录之前的无限期不变性。

先前工作中提出的一个特殊问题包括检查点终止的最终性问题,即在日志文件有意结束后,防止给定日志的进一步检查点附加到分类账。这样的



(a) Dependency diagram of linked integrity proof for single bag file database.

Topics				BBR	
id	name	type	format	nonce	digest
1	/foo	gps	rtps	$bits_A$	$bits_{A_0}$
2	/bar	imu	rtps	$bits_B$	$bits_{B_0}$
3	/baz	lidar	rtps	$bits_C$	$bits_{C_0}$

Messages				BBR
id	topic	time	data	digest
1	id 1	1.0s	bits	$bits_{A_1}$
2	id 2	3.1s	bits	$bits_{B_1}$
3	id 1	4.5s	bits	$bits_{A_2}$

(b) Database tables for topics and messages.

图 2. 通过二维数组哈希链实现数据提交及其数据库插入,其中主轴检查点是每个主题的创世块和元信息,而次轴则检查相应消息数据的插入。这种耦合提供了整个数据库的整体完整性证明。同时将主题保存为时间序列原子。

受到威胁或怀疑的记录者可能会采取行动
入侵,提供自我毁灭的威慑力,并减少
对手恢复的私钥的实用性。

之前使用基于代币的区块链的研究可以得出结论
通过输出事务的检查点记录
随机公共身份,其私钥未知。
这种极端的全有或全无的所有权丧失在概率上
最终的,但不提供任何其他出处地位,例如
为“停滞”、“危急”或逆转的方法,有助于在情况解决后有条件地恢复日志。

而是在这方面提供更细致的监管
检查点记录的生命周期。

五、实施

作为概念验证,我们实现了 Black Block Recorder
使用现有的开源机器人中间件和分布式
账本软件。之所以选择 ROS2,是因为它支持安全
多播网络(P1)和模块化 ROSBag2 插件设计,
实现内部/外部机器人的安全高效攻丝
网络(P3)。Hyperledger Sawtooth3被选为账本
其节能且 BFT 共识算法的框架,
多语言 SC 处理器、许可的 DLT 支持,以及
可并行的交易架构。

由于保管方和报告方都以物理 CPS 设备的形式出现,因此他们的身份特别容易被识别
攻击。在这里,两者都用于共同签署批量交易
用于验证器提交;因此附加了伪造检查点
必然导致保管人和报告人双方的腐败。

A. 检查点集成

为了将我们的链接检查点方法集成到 ROSBag2 中,
我们扩展了现有的 SQLite 默认存储插件,以便额外计算和广播检查点。一个二维数组
创建哈希链是为了将 bagfile 数据库添加到附加
仅限数据结构。以下公式结合
图 2 中的表格和颜色编码流程图描述了
检查点主题插入的过程:

位袋←${0, 1}m位A (2)

←HMAC(bitsbag, Proto(namebag))位A0←HMAC(位 (3)

A, P inside(typeA, formatA)) (4)

位B ←HMAC(位A0, P inside(nameA)) (5)

bitsB0←HMAC(bitsB, P inside(typeB, formatB)) (6)

位C ←HMAC(位B0, P roto(nameB)) (7)

bitsC0←HMAC(bitsC, Proto(类型C ,格式C)) (8)

bagfile 的 nonce(蓝色)与 bagfile 结合
元数据(紫色),通过 protobuf 确定性序列化
为了避免散列项目列表的歧义,生成
第一个插入主题的 nonce。然后将其与
用于生成创世摘要的主题的 IDL 信息。
然后将前一个主题的创世摘要和元数据合并
为下一个主题播下随机数,报告如下
bagfile 本身的检查点(橙色)(P2)。此后,
对于每个附加主题,都会重复此循环。

对于消息(绿色),相应的先前摘要
主题与消息及其到达时间相结合
计算当前摘要,如下图和图 2 所示:

位A1←HMAC(位A0, P inside(timeA1 ,dataA1)) (9)

bitsB1←HMAC(bitsB0, P inside(timeB1 ,dataB1)) (10)

bitsA2←HMAC(bitsA1, P inside(timeA2 , dataA2)) (11)

通过这种方式,bagfile 和消息检查点足够松散地耦合,以审计数据来源,同时保持独立,以进行并发计算和原子记录保存。

甚至跨不同的主题流。
使用 BBR 存储和桥接插件的机器人部署可以
如图 3 所示。注意记录的可分离阶段
与在蓝色区域内签署检查点相比。这允许
用于交换数据库存储驱动程序的模块化集成
或替代分类账基础设施。

3Hyperledger Sawtooth: https://www.hyperledger.org/projects/sawtooth

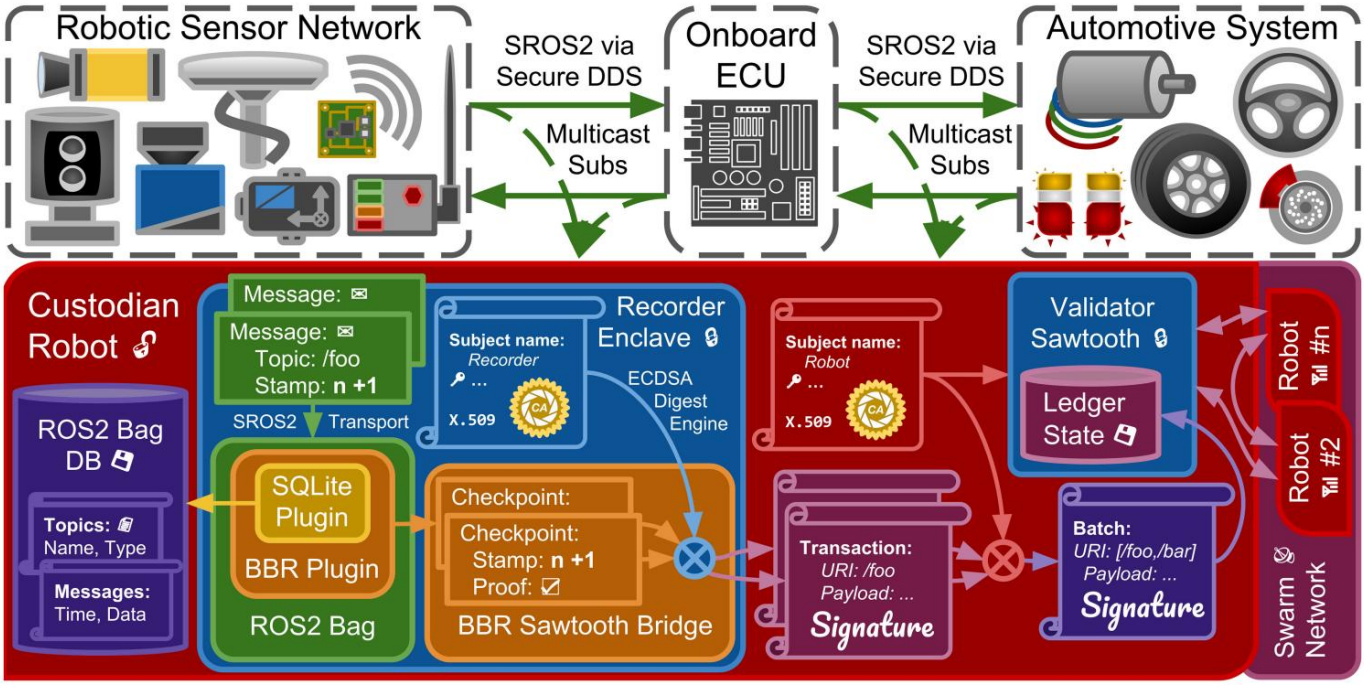


图 3. 不可变日志记录管道的流程图可视化。虽然每个机器人平台都存在嫌疑,但会为记录器进程保留一个安全区域（例如 TTE）。记录的输入会在区域内安全接收,并用于以加密方式导出特定于每个被跟踪输入资产的链接完整性证明。由于日志数据可能会流式传输到外部存储,因此相应的检查点事务会绑定到机器人的公共身份进行批处理,然后由密封在区域内的记录器私钥进行签名。因此,只有机器人的私钥才能用于签名和中继批处理事务以进行验证;附加伪造信息需要托管人及其指定的记录器共同串通。

B. EDR 智能合约的交易家族

为了开发 BBR 的供应链,我们扩展了 Sawtooth 的参考供应链交易族 (TF),用于追踪资产的来源和其他上下文时间序列信息。图 4 中,我们使用数字资产建模语言 (DAML) 4 来规范化我们的供应链,DAML 是一种开源领域专用语言,用于直接表达合约、参与方、权利、义务和授权(P5)。

图 4a 中的第 1-14 行捕获了 EDR 协议的主状态控制链 (SC),其中主要相关方作为签署方输入,同时为一组外部相关方提供了对状态控制链状态的观察访问权限。创建关联记录的控制权完全委托给记录器。第 23-40 行捕获了特定记录的辅助状态控制链;即单个主题的日志检查点。同样,记录器可以选择追加或最终确定记录,同时断言已提交的检查点保持单调性。所有者也可以选择最终确定记录,此时状态控制链将被归档,并在分布式账本技术 (DLT) 中完全不可更改。

最后,第 16-22 行和第 42-44 行指定了记录器在选择上述 SC 的操作时必须提交的结构数据。完整的 DAML 模型,包括用于建立多方协议的待处理 SC,以及图 4b 中所示的测试场景,均已开源并公开发布。5随着 DAML 与 Sawtooth 仍处于早期开发阶段,BBR 的 TF 仍然使用 Rust 编程语言实现。DAML 模型忠实地体现了 SC 逻辑。

C.性能分析和QoS调整

为了初步验证BBR在机器人系统中的可处理性,我们通过评估一系列常见传感器消息大小和频率下的丢包率性能和CPU负载,对使用BBR存储插件和桥接接口所带来的开销进行了定量的基准测试比较。测试结果表明,BBR的当前性能与默认驱动插件的性能非常接近,同时单线程工作负载仍保持不饱和状态。参见图5。

中端工作负载期间的边际性能提升可能是由于 CPU 缓存未命中次数减少造成的,这是因为在持续开销的情况下进程空闲时间减少。

关于所描述的吞吐量下降,尽管BBR试图以写入数据库的速率对事件进行检查点操作,但实际上,出于之前所述的目的,这些检查点通过桥接接口的签名和传输应该受到速率限制,例如本地QoS限制或减轻外部验证器的工作负载（R2）;鉴于ECDSA交易签名仍然是管道中的主要加密瓶颈。回想一下,只要每个事件都被纳入哈希链,下采样检查点发布就不会抑制未发布检查点的日志段的防篡改属性,而只会抑制在日志中精确定位更改的分辨率。

4DAML 规范： https://daml.com 5EDR DAML 模型：
https://github.com/dledr/edr_daml

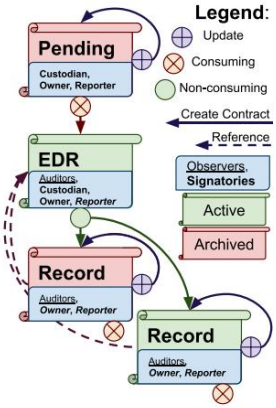
```
template Edr -- Smart Contract for EDRs
 with
   auditors: [Party] -- Regulatory Agencies
   custodian: Party  -- Robot/Vendor Identity
   owner: Party      -- User/Operator Identity
   reporter: Party   -- Logger/TEE Identity
 where
   signatory custodian, owner, reporter -- obligated
   observer auditors -- non-obligated parties
   ensure unique (custodian :: owner :: reporter)
   controller reporter can -- create many Records
     nonconsuming Edr_Record : ContractId EdrRecord
       with record: Record
       do create EdrRecord with edr = this; record

data Record = Record with  -- data type struct
   r_name: Text   -- "/sensors/exteroceptive/gps"
   r_type: Text   -- "gps"
   r_format: Text -- "rtps"
   r_nonce: Text  -- "<bits_A>"
   r_digest: Text -- "<bits_A_0>"
   r_checkpoints: [Checkpoint] -- monotonic list
```

```
template EdrRecord -- Smart Contract for Records in EDR
 with
   edr: Edr       -- Reference EDR of origin
   record: Record -- Initial Record state
 where
   signatory edr.owner, edr.reporter
   observer edr.auditors -- custodian can be excluded
   choice EdrRecord_Append : ContractId EdrRecord
       with checkpoints: [Checkpoint] -- [] for batching
       controller edr.reporter -- Only Reporter appends
       do let -- Update Record with added checkpoints
           is_valid = checkMonotonic record checkpoints
           _record = appendCheckpoints record checkpoints
         assert (is_valid == True) -- Error on invalid
         create EdrRecord with edr; record = _record
   choice EdrRecord_Finalize : () -- Archives Contract
     controller edr.owner, edr.reporter
       do return () -- Finalized Record is un-appendable

data Checkpoint = Checkpoint with -- data type struct
   c_proof: Text -- "<bits_A_1>"
   c_stamp: Int  -- 1
```

(a) Formal specification of Smart Contract model for Event Data Recording captured in DAML



(b) Smart Contract Diagram

图 4. 所示为使用分布式账本技术 (DLT) 的 EDR 合约模型简化版,该模型通过特定于 SC 的领域建模语言进行定义。EDR SC 首先通过待定合约进行初步提议,用于收集必要的多方签名。待定合约最终确定后,将用于创建已达成一致的 EDR SC,允许记录器创建多个引用记录并附加检查点,这些记录可由其自身或所有者最终确定。相应的 SC 图描绘了一个基本的示例场景。

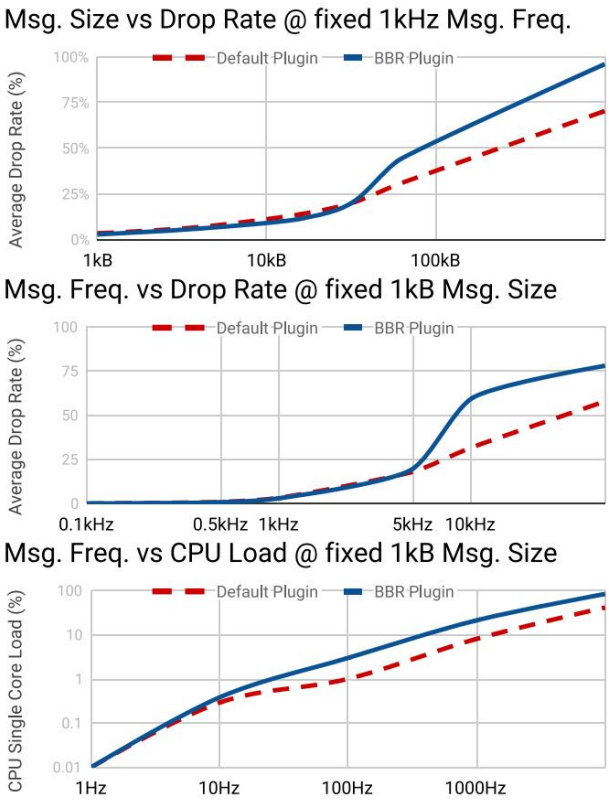


图 5. 我们的 BBR 存储/桥接插件与 ROSBag2 默认 SQLite 插件的 Droprate/Load 性能比较。基准测试基于 ROS2 Crystal、2.6 GHz Intel i7-6700HQ 处理器,并在环回接口上使用 RTI Connext RMW。

关于上行网络使用情况,验证器流量取决于共识算法、Gossip 协议、参与者数量以及特定于所使用的 DLT 实现/框架的提交交易的频率/大小。但是,为了分析我们 BBR 桥的特定网络带宽使用情况,表一列出了使用当前实现的序列化方案计算出的最低有效负载要求。

表一
BBR有效载荷分配

Payload	Size (bytes)	Requirements
Chk_i	36	Ledger Disk Storage
Signed Transaction	≥629	Network Bandwidth
Signed Batch	≥965	Network Bandwidth

这描述了给定一个签名批量传输的下限,该传输包含一个签名交易列表,其中包含一个检查点数组。虽然交易通常可以包含任意主题元数据,但检查点只是一个由 256 位哈希值和 32 位无符号整数组成的元组。作为参考,在运行 ROS1 导航堆栈的典型 TurtleBot3 上,rosbag 记录所有 20 个唯一主题,写入磁盘的速度约为 1.4 MB/s,每秒 1k 条消息,以 1 Hz 的频率批量发布每个检查点会导致 BBR 上行链路开销约为 400 Kbps。实际上,更合理的步幅是每秒每个主题一个检查点,可以将传输速率降低到 110 Kbps 左右,账本状态增长率可持续低于 1 KB/s （P4）。

六、结论

在这项工作中,我们使用分布式账本和链接完整性证明为机器人建立了一个安全日志记录框架,以确保连续事件数据记录的不变性。

真实性和不可否认性是通过传播检查点证明和智能合约来实现的,这些证明和智能合约尊重相互不信任的各方的性质,同时加强监管者、机器人和用户之间的合约共生关系。

事实上,虽然 BTF 适用于 DLT 网络,但损害 EDR SC 中涉及的任何单一方本身并不会导致无法追踪的附加伪造,而利用所有 EDR SC 方也不会导致过去的伪造。

关于通过 BBR 记录日志与使用传统 rosbag 相比产生的边际开销是否超过

确保事件记录保持防篡改的好处,我们得出结论,给定用于报告重要主题的适当QoS,在安全敏感的机器人领域应用BBR的实用性和实用性仍然具有优势。

我们预计,机器人 EDR 的这一应用领域将成为加密机器人和分布式账本交叉领域中众多令人兴奋的探索前沿之一,因为所提出的方法可以推广到未来的 DLT 架构。

分布式账本的安全性取决于管理全局状态的验证者的数量和健康状况,这会影响R2 和 R5。如果验证者池太小或由单一方主导,社区对账本的信任就会动摇。为了减轻验证的垄断,可以征召更多雾级物联网设备(从固定的 C-ITS 基础设施到移动现场机器人)作为额外的验证者,以帮助增强设备多样性。为了激励参与并避免公地悲剧,可以采用替代共识协议。Tangle [29] 是一种有向无环图 (DAG) 共识算法,用于物联网领域的分布式账本技术 (DLT) IOTA,它利用了一种吉祥的“预付现金”策略:要发出交易,身份必须首先验证比其希望提交的更多的交易,从而持续为网络安全做出贡献。

近年来,DAG 因其更高的异步交易吞吐量和更宽松的连接要求,相较于区块链数据结构,其应用越来越广泛。对于通过时间上不相交或半分区网络连接的大规模机器人部署,DAG 可以缓解使用传统同步分布式数据结构所带来的持续全局连接需求。

最后,鉴于目前已提出的支持并行检查点修正的链接完整性证明设计,缺乏能够充分利用这种并发性的 ROSBag2 存储插件仍然是一个缺陷。因此,开发一个兼容 SQLite 替代方案并支持并行写入的存储插件,将有利于高性能、高带宽的消息捕获以及不可变事件数据的记录。

致谢

作者要感谢 Hyperledger 项目的开源努力和文档;这两者都对开发这项工作和探索其应用起到了重要作用。

参考

[1] J. McClean、C. Stull、C. Farrar 和 D. Mascareñas,“机器人操作系统 (ROS) 的初步信息物理安全评估”,SPIE Proc. vol. 8741, 2013, Art. no. 874110. [在线]。可访问网址:http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/ 12.2016189

[2] C. Miller 和 C. Valasek,“远程利用未改装的乘用车”,载于 Proc. Defcon 23,2015年,第 2015 卷,第 1-91 页。[在线]
可访问网址: http://illmatics.com/RemoteCarHacking.pdf [3] K. Koscher
等人, “现代汽车的实验安全分析”,载于Proc. IEEE Symp. Secur. Privacy, 2010 年,第 447-462 页。

[4] S. Checkoway等人, “汽车攻击面综合实验分析”,载于 2016 年第 20 届 USENIX Conf. Secur.会议纪要,第 6 号。[在线]。可访问网址:http://www.usenix.org/events/security/tech/full_papers/Checkoway.pdf

[5] C. Miller 和 C. Valasek,“远程汽车攻击面调查”,载于 Proc. Defcon 22,2014 ,第 1-90 页。[在线]。可访问网址:http://illmatics.com/remoteattacksurfaces.pdf [6] S. Morante,JG Victores 和 C. Balaguer,“加密机器人学:机器人为何需要网络安全”,载于 Frontiers Robot. AI,第 2 卷,
第 23-26 页,2015 年 9 月。[在线]。可访问网址:http://journal.frontiersin.org/Article/10.3389/frobt.2015.00023/abstract [7] R. Rowlingson 和 Q. Ltd,“取证准备的十步流程”,载于 Int.
J. Digit. Evidence Winter,第 2 卷,第 3 期,2004 年。
[8] VK Veitas 和 S. Delaere,“自动驾驶汽车的车载数据记录、存储和访问管理”,2018 年, arXiv:1806.03243。

[9] G. Becker,“Merkle 签名方案、Merkle 树及其密码分析”,德国波鸿鲁尔大学,波鸿,技术代表, 2008 年。

[10] S. Nakamoto,“比特币:一种点对点的电子现金系统”,2008 年。
[在线]。网址:www.bitcoin.org
[11] B. Group 和 J. Garzik,“公有区块链与私有区块链。第二部分:无需许可的区块链”,2015 年。[在线]。可访问网址:http://bitfury.com/content/5-white-papers-research/public-vs-private-pt2-1.pdf [12] B. Group 和 J. Garzik,“公有区块
链与私有区块链。第一部分:许可型区块链”,2015 年。[在线]。可访问网址:http://bitfury.com/content/5-white-papers-research/public-vs-private-pt1-1.pdf [13] KJ O Dwyer 和 D. Malone,“比特币挖矿及其能源足迹”,载于Proc. IET Conf.,
2014 年 1 月,第 280-285 页。[在线]可访问网址: http://digital-library.theiet.org/content/conferences/10.1049/cp.2014.0699 [14] V. Dhillon、D. Metcalf 和 M. Hooper,《超级账本项目》。
伯克利,加州,美国:Apress,2017 年,第 139-149 页。[在线]。获取方式:https://doi.org/10.1007/978-1-4842-3081-7_10 [15] D. Butin、M.
Chicote 和 DL Métayer,“问责制的日志设计”,载于IEEE Secur. Privacy Workshops 会议录, 2013 年 5 月,第 1-7 页。
[16] J. Benet,“IPFS 内容寻址、版本化、P2P 文件系统”,2014 年, arXiv:1407.3561。[在线]。可访问网址:http://arxiv.org/abs/1407.3561 [17] N. Anderson,“区块链技术:会计领域的游戏规则改变者?”
2016 年。[在线]。可访问网址:https://www2.deloitte.com/content/dam/Delo-itte/de/Documents/Innovation/Blockchain_A%20game-
changer%20in%20accounting.pdf [18] P. Snow、B. Deery、J. Lu、D. Johnston 和 P. Kirby,“Factom 业务流程通过区块链上的不可变审计线索得到保护”,白皮书,Factom,美国德克萨斯州奥斯汀,2014 年 11 月。[在线]。可访问网址:https://github.com/FactomProject/FactomDocs/raw/master/Factom_Whitepaper.pdf [19] J. Cucurull 和 J. Puiggalí, 《安全日志的分布式不可变化》(计算机科学系列讲义),G. Barthe、E. Markatos 和 P.
Samarati 编。柏林,德国:Springer,2016 年,第 9871 卷,第 2 期,第 122-137 页。
[20] M. Bartoletti 和 L. Pompianu,“比特币 op_return 元数据分析”,载于《金融密码数据安全国际会议论文集》, 2017 年,第 218-230 页。
[21] R. Matzutt等人, “任意区块链内容对比特币影响的定量分析”,载于2018 年第 22 届国际金融加密数据安全会议论文集。
[22] A. Sutton 和 R. Samavi,“区块链支持的隐私审计日志”,
国际语义网会议, 2017 年,第 645-660 页。
[23] A. Gervais、H. Ritzdorf、GO Karame 和 S. Capkun,“篡改比特币区块和交易的交付”,载于第 22 届 ACM SIGSAC 计算通信安全会议论文集, 2015 年,第 692-705 页。[在线]
可访问网址:http://dl.acm.org/citation.cfm?doid=2810103.2813655 [24] P. Szalachowski,“迈向更可靠的比特币时间戳”,
Crypto Valley Conf.区块链技术, 2018 年 6 月,第 101-104 页,doi: 10.1109/CVCBT.2018.00018。
[25] SA Crosby 和 DS Wallach,“用于防篡改日志记录的高效数据结构”,载于2009 年第 18 届 USENIX Secur. Symp. 会议论文集,第 317-334 页。[在线]。可访问网址:http://dl.acm.org/citation.cfm?id=1855768.1855788
[26] VK Veitas 和 S. Delaere,“政策扫描和技术战略设计
方法论”,2018 年, arXiv:1806.03235。
[27] S. Taurer、B. Dieber 和 P. Schartner,“智能机器人的安全数据记录和仿生功能完整性”,载于Proc. IEEE/RSJ Int.
会议。英特尔。机器人系统, 2018 年 10 月,第 8723–8728 页。
[28] M. Bellare,“NMAC 和 HMAC 的新证明:无抗碰撞性的安全性”, J. Cryptol.,第 28 卷,第 4 期,第 844-878 页,2015 年 10 月。[在线]。
可访问网址:https://doi.org/10.1007/s00145-014-9185-x [29]
S. Popov,“The tangle”,IOTA白皮书, 2018年2月。[在线]可访问网址:https://assets.ctfassets.net/r1dr6vzfxhev/4i3OM9JTleiE8M6Y04li28/d58bc5bb71cebe4adc18fadea1a79037/Tangle_White_Paper_v1.4.2.pdf