计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2021,30(11):91−98 [doi: 10.15888/j.cnki.csa.008148]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于区块链的工业互联网安全平台①

于金刚[1,2], 赵培培[1,2], 仲启强[3], 王海汀[1,2], 李　姝[4]

[1](中国科学院大学, 北京 100049)
[2](中国科学院 沈阳计算技术研究所, 沈阳 110168)
[3](中国矿业大学 (北京), 北京 100083)
[4](沈阳理工大学 装备工程学院, 沈阳 110159)
通讯作者: 李　姝, E-mail: lishucx@163.com

摘　要: 在传统的工业互联网平台中, 终端设备产生数据的安全和隐私问题是阻碍工业互联网发展的主要瓶颈. 伴随着终端数据量几何式的增长, 保护数据的安全性和完整性已经成为工业互联网的核心研究领域. 本文首先设计了一种基于区块链的设备和数据管理的体系架构, 提供了一个可靠的防篡改的数据库. 然后, 利用数字证书对平台采用权限访问控制机制, 提高平台的准入的安全等级. 其次, 通过链码间接地对终端设备及其配置文件进行管理, 避免了终端设备随意地接入对数据造成污染的问题. 最后, 通过终端设备自身的公私钥来对终端设备产生的数据进行打包加密处理, 利用区块链的共识机制, 存储在区块链上. 通过实验表明, 所提出的方案具有良好的稳定性、安全性和可操作性.

关键词: 区块链; 工业互联网; 共识机制; 数据安全; 链码

引用格式: 于金刚,赵培培,仲启强,王海汀,李姝.基于区块链的工业互联网安全平台.计算机系统应用,2021,30(11):91–98. http://www.c-s-a.org.cn/1003-3254/8148.html

Industrial Internet Security Platform Based on Blockchain

YU Jin-Gang[1,2], ZHAO Pei-Pei[1,2], ZHONG Qi-Qiang[3], WANG Hai-Ting[1,2], LI Shu[4]

[1](University of Chinese Academy of Sciences, Beijing 100049, China)
[2](Shenyang Institute of Computing Technology, Chinese Academy of Sciences, Shenyang 110168, China)
[3](China University of Mining and Technology (Beijing), Beijing 100083, China)
[4](School of Equipment Engineering, Shenyang Ligong University, Shenyang 110159, China)

Abstract: In the traditional industrial Internet platform, the security and privacy issues of data generated by terminal equipment are the main bottlenecks hindering the development of the industrial Internet. With the geometric growth of terminal data, protecting the security and integrity of data has become the core research area of the industrial Internet. This study first designs blockchain-based equipment and data management architecture, providing a reliable, tamper-proof database. Then, the digital certificate is employed to provide the authority access control mechanism for the platform to improve the security level of platform access. Secondly, the terminal equipment and its configuration files are managed indirectly through the chain code, which avoids the data pollution caused by the random access of the terminal equipment. Finally, the data generated by the terminal equipment is packaged and encrypted through the public and private keys of the equipment itself and then stored on the blockchain with the consensus mechanism of the blockchain. Experiments show that the proposed scheme has good stability, safety, and operability.

Key words: Blockchain; industrial Internet; consensus mechanism; data security; chain code

① 基金项目: 辽宁省博士科研启动基金 (2019-BS-257)
Foundation item: Scientific Research Start-Up Fund for Doctorate of Liaoning Province (2019-BS-257)
收稿时间: 2021-01-21; 修改时间: 2021-02-23; 采用时间: 2021-03-03; csa 在线出版时间: 2021-10-22

1 引言

工业互联网通过开放的、全球化的通信网络平台, 把设备、生产线、员工、工厂、仓库、供应商、产品和客户紧密地连接在一起, 共享工业生产全流程的各种要素资源, 工业互联网平台面向数字化、自动化、智能化以及网络化, 实现了效率的提升和成本的降低. 通过对海量的数据采集、汇聚和分析, 来实现资源的优化配置. 数据是工业互联网平台的核心, 数据安全是工业互联网平台的命脉同时也是制约工业互联网发展的瓶颈, 目前如何保证平台数据的完整性、可用性、防篡改以及防止数据泄漏是困扰其发展的主要问题.

区块链也称分布式共享账本, 是由密码学保护的不可变的记录数据库. 它允许数字资产的交换与存储, 而不需要第三方的监督. 区块链技术有去中心化存储、全网验证、自动执行三大特性, 从根本上保证了工业互联网平台的数据完整性、可用性和可持续性, 避免了数据的泄漏和污染问题, 极大地提高了平台的安全性.

Hyperledger 是由 Linux 基金会[1] 托管的一组开源的工业区块链框架项目, 如 Fabric、Sawtooth、Burrow 和 Iroha 等. Hyperledger Fabric 是 IBM 提供的一个开源的区块链开发平台[2], 开发者可以根据自己的项目需求进行平台开发. Hyperledger Fabric 提供模块化架构、成员组件来创建灵活的许可区块链平台. 智能合约在 Fabric 中被称为链码 (chain code), 链码是用于实现应用逻辑和事务功能的编程代码. Fabric 使用执行-排序-确认的架构模式而不是排序-执行[3] 的架构模式, 克服了许可区块链所存在的并发事务执行的不确定性、所有节点执行、信任模型不灵活、硬编码一致性等问题.

Hyperledger Fabric 包含一个账本子系统, 这个子系统包括两个组件: 世界状态和交易日志. 世界状态组件描述了账本的当前状态, 它是账本的数据库. 而交易日志包括所有交易的历史, 它是世界状态的更新历史. 账本则是世界状态数据库和交易历史日志的集合. 如果一个事务改变了已经存储在账本上的任何值, 或者向账本添加了新的数据, 这被认为是区块链的新状态, 它将被永久保存, 并且不能撤销区块链的先前状态[4].

在 Fabric 中智能合约被称为链码, 链码是一组可编程的功能函数, 存储在区块链上并自动执行其条款, 而不需要可信的中介. 链码的功能几乎和传统合同一样, 传统合同包括各方都有义务遵守的条件, 因此链码被设计成通过自动验证条件并根据条件结果自动运行后续步骤来自动化相同的任务. 从而, 链码可以使交易自动化, 而不需要中央权威机构的干预. 大多数情况下, 链码只会访问账本的数据库组件和世界状态, 但不会查询交易记录. 通过使用链码, 我们可以在区块链上定义特定条件下更复杂的交易, 可以开发相应的应用程序, 而不仅是传输如供应链、业务流程管理和医疗保健等各式各样的数字货币.

2 相关工作

当前区块链、物联网、工业互联网和共享经济在各自领域都已经经过了长期的研究和发展, 形成了各自比较成熟的体系, 但是, 在工业互联网面临转型的关键期背景下, 工业互联网与区块链的融合研究和开发还处于初始阶段. 当前, 工业互联网设备每天产生大量的数据, 必须要以安全的方式对其进行整理、处理和存储.

Internet Engineering Task Force (IETF) ACE[5] 提出了一种在受限的环境下进行身份验证和授权的通用框架, 访问终端设备需要进行身份的验证和授权, 以此来保证系统的安全. Object SeCurity ARchitecture for the Internet of things (OSCAR) 通过保护应用层的付费负载, 解决了数据报传输层安全 (DTLS) 协议[6] 的主要限制. 这种方法允许高效的多路广播、异步通信和缓存, 资源服务器将受保护的资源存储在本地, 或者以加密和签名的格式存储在代理服务器上, 客户端向负责的密钥服务器请求解密密钥, 可以通过使用不同的密钥加密不同的资源来提供访问控制. Yu 等人也提出基于区块链的物联网数据共享模型[7], 利用智能网关将物联网设备产生的数据上传到区块链中实现共享, 解决了对传统中心化机构信任的问题.

终端设备的监视和管理对工业互联网平台的安全十分重要, 它保证了工业设备的持续、高效和无故障运行. 目前, 大多数流行的网络监管系统和工具[8] 主要是对网络的状态进行测量和跟踪. 他们通过插件和扩展提供不同级别的管理. 网络监视的一种方法是利用探针测量网络的指标进行主动的监视终端设备状态, 通过配置成对的网络设备, 向网络中注入额外的流量, 以监控关键指标, 保证设备运行的质量. 主动网络测量方法有 IETF 提出的单向主动测量协议 (OWAMP)[9] 和双向主动测量协议 (TWAMP)[10], 其缺点就是需要额外的流量以及对资源的影响.

本文提出了一种基于区块链技术的工业互联网安全平台架构, 网络管理员首先通过身份认证进行身份授权, 然后将更改的设备配置文件上传到区块链网络中来间接地控制网络终端设备, 网络终端设备通过获取区块链网络中与自己相关的配置文件来更新自身的配置. 另外, 终端设备所产生的数据对其进行签名打包, 在区块链系统中达成共识[11] 后, 存入分布式数据库中, 进一步增强了终端数据的安全性, 这与传统工业互联网平台架构模式形成了鲜明的对比.

在本节中, 我们介绍了目前区块链技术在物联网和工业互联网的研究现状, 并对当前工业互联网在安全管理方面的技术和缺点进行了分析. 本文其余部分的结构如下: 在第 3 节中, 叙述了基于区块链的工业互联网安全平台具体设计细节. 第 4 节通过实验数据对平台的性能进行了测试和分析. 结论和未来的工作将在第 5 节中叙述.

3 平台架构设计

3.1 架构设计

本文基于区块链的设备及数据管理架构设计如图 1 所示, 操作人员使用数字证书对自己进行身份验证, 然后拥有对特定设备或者设备组权限的操作人员可以修改记录在区块链上的设备配置文件. 对于操作人员的访问控制方法, 本文使用了基于角色的访问控制方法和基于规则的控制方法, 根据用户在系统中的角色来规范用户的访问行为. 此外, 系统还规定了谁有权通过提交事务来更改其他用户的访问权限, 并且所有的事务记录都将永久的保存在区块链上, 没有人可以删除或者更改. 对于设备配置文件的更改先通过语法验证来检查配置文件的正确性, 来尽量避免因人为错误导致配置文件错误从而影响终端设备的正常运转. 操作人员的数字证书还用于对新配置文件的签名, 以便对配置文件进行标志和归属.

事务是由时间戳、操作人员 ID、设备 ID 以及加密的设备配置文件的散列值组成. 一旦事务被写入新的区块中并添加在区块链上, 事务信息就会被分发到区块链网络中的各个对等节点. 配置文件被上传到区块链网络后, 新的区块将触发事件, 所有被操作人员管理的设备将检查新的配置文件是否影响自己, 对受影响的设备, 设备 ID 从区块链网络中下载新的配置文件, 然后设备使用其私有密钥解密从区块链中下载的加密配置文件, 并应用修改后的配置信息. 区块链网络中保存所有更改的历史记录, 供安全和审计人员审查.

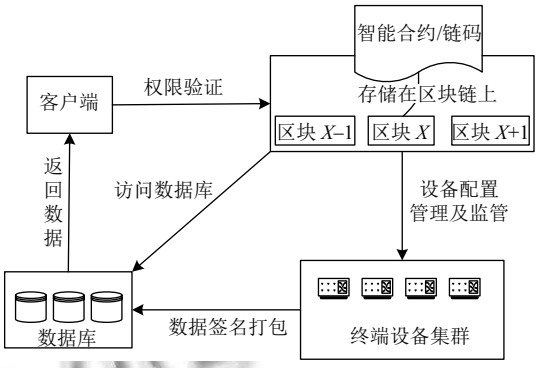

图 1 基于区块链的设备及数据管理架构

通过区块链的身份权限特点, 本文为终端每个终端设备生成不同的公钥和私钥, 每个终端设备都有自身 IP 地址与该地址在系统平台中所对应的身份证书, 从而形成一张设备终端 (IP) 与公私钥对应的列表[12]. 从而杜绝了因为终端设备的随意接入和恶意替换导致的设备数据污染问题. 除此之外, 目前的工业互联网终端设备生产数据的数据都是高频数据采集, 对网络传输、数据缓存等方面带来性能和成本上的巨大压力, 利用区块链技术的分布式存储, 可以有效地解决中心化采集的弊端, 减轻系统数据存储压力和设备边缘数据缓存的压力.

对于存储在系统中的数据, 操作人员通过身份验证后进行访问操作, 系统根据不同角色的权限来受限访问数据库中的数据信息, 从而达到根据角色权限等级达到数据隔离的目的. 本文中对于拥有终端设备配置文件操作的角色即拥有该终端设备所产生数据的访问权, 因为数据是用设备的公钥进行加密存储的, 只有拥有的设备私钥的操作人员才能解密数据. 因此, 只有拥有终端设备操作权限的角色, 才拥有终端设备的公私钥, 才能查看其产生的数据. 而且, 每次数据库操作的记录都会永久的存储在区块链上, 任何人无法更改访问记录, 供安全和审计人员审查.

3.2 权限管理

本文使用 Hyperledger Composer 框架[13] 来实现系统的访问权限控制, 它是 Fabric 的上层架构. 所有组件都可以由 Composer 模块化结构的定义, 然后将其打包为一个组件并部署到 Fabric 区块链上. Composer 模块的主要部分是 model 文件集、访问控制语言文件、查

询文件和 JavaScript 文件集.

在 model 模块中, 本文定义了参与者、资产、事务和事件. 参与者是系统中的所有用户以及需要经过身份验证的其他用户, 参与者有唯一的字符 ID 作为标识; 资产是系统的物理场所 (比如: 终端设备、数据以及历史记录), 通过智能合约和访问控制文件来管理这些资产; 事务的提交表示区块链的预先定义的条件状态更改, 比如用事务来授权用户的访问权限或者用事务来撤销用户的访问权限. 事件是系统中参与者所触发的行为, 参与者对历史记录的查看、对终端设备配置文件的更改和对数据的访问都是一个事件.

访问控制语言 (ACL) 中定义了访问控制的策略, 具体的访问策略包括如下几部分:

1) 系统的访问权限, 系统访问权限规定了谁可以进入系统, 因为 Fabric 是联盟链, 参与者实行准入原则, 没有许可的参与者无法进入系统;

2) 系统管理员的访问权限, 系统管理员拥有系统的最高权限, 它可以给普通的参与者分配角色, 也是维护系统运行的主要负责人;

3) 不同角色在系统默认情况下可以访问的系统资源不同, 有的角色能查看终端设备配置文件能够访问数据库数据, 还有的角色无权访问特定终端设备配置文件和无权访问数据库中的数据;

4) 不同角色可以发送的事务, 系统中不同的角色可以发送的事务不同, 有的可以更改终端设备的配置, 有的角色则无权更改只能查看;

5) 系统中所有参与者都可以访问的历史记录, 这些记录包括参与者对终端设备修改的日志, 以及参与者访问系统的日志, 以供安全审计.

ACL 模块由 5 个不同的集合定义: 参与者集合、资源集合、行为集合、条件集合和操作集合. 在本系统中, 用户可以根据系统中定义的角色来访问资源, 因此本文的访问控制策略是基于角色的访问控制策略[14]. 本文在 model 文件中定义了不同类型的参与者, 它们分别对应着系统中不同的角色, 特定的角色访问特定的资源并对资源进行特定的操作并做出特定的行为: 如创建、读取、更新、删除 (CRUD). 因此 ACL 模块就是基于角色访问控制的核心, 通过 ACL 模块对特定角色进行特定的访问控制操作, 从而保护系统中资产的安全.

事务处理函数是 JavaScript 文件的一部分, 可以将它转化为 Fabric 上的链码, 因此其作用可以视为智能合约. 事务处理函数预先定义好每个事务的处理逻辑以及需要满足的条件. 当提交相应的事务时, 将自动调用事务处理函数. 图 2 展示了基于 ACL 模块的访问权限过程.

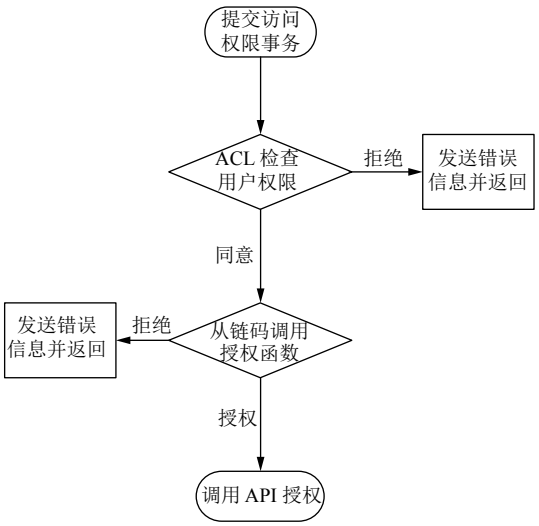

图 2 基于 ACL 模块的访问权限流程

查询文件是基于防篡改的查询策略, 根据区块链的数据结构特点可以保证查询的数据没有被篡改或者删除. Hyperledger Composer 提供了历史记录, 历史记录详细地记录了参与者的信息以及其行为. 历史记录模板如图 3 所示.

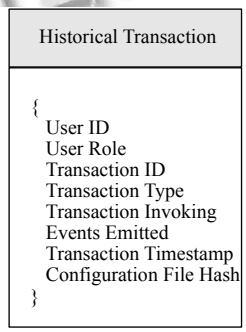

图 3 历史事务记录

3.3 终端设备的管理

对于工业互联网终端设备的管理, 被授权的管理员使用自己的数字证书[15] 进行身份验证, 然后管理员可以修改对被授权的特定设备或设备组在区块链网络中的设备配置文件, 最后管理员使用自己的数字证书对修改的配置文件进行签名, 以便标识和安全审计. 修

改后的配置文件经过语法验证来验证新配置文件的正确性, 以此尽量减少因为人为因素导致终端设备宕机故障的发生. 通过语义验证后, 新的设备配置文件信息经过加密会被分发到区块链网络中各个 peer 节点[16], 并通知所有被管理的终端设备, 终端设备检查新的配置文件更改是否影响其配置, 对产生影响的设备, 通过设备自身私钥下载区块链网络中新的设备配置文件进行更新, 并应用修改后的配置文件. 具体执行流程如图 4.

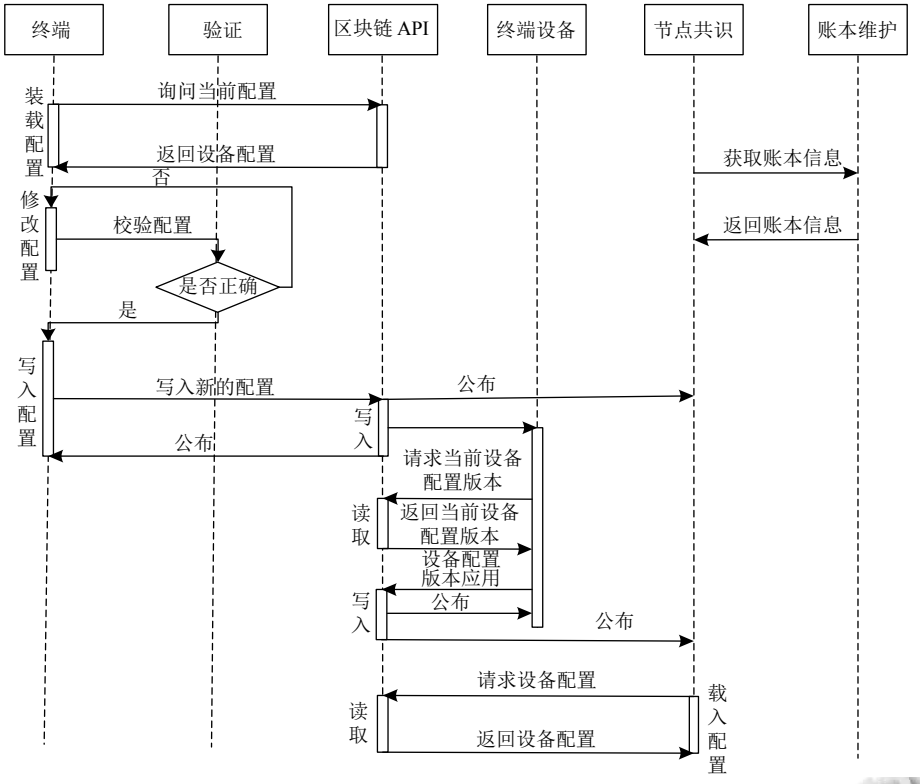

图 4 设备配置文件更改时序图

该流程为设备配置文件更改的过程, 具体时序如下:

1) 终端 $A_x(a_1, a_n)$ 从区块链上加载设备 $D_y(d_1, d_n)$ 或设备组 $G_z(g_1, g_n)$ 的配置文件, 并使用终端的自身的密钥 S_x 对配置文件进行解密.

2) 终端 A_x 修改配置文件.

3) 发送新的配置文件进行语义验证, 将人为的错误最小化.

4) 经过语义验证的配置文件被加密并写入到一个新的区块中, 同时终端 ID、设备 ID 和时间戳等一并写入, 区块被添加到区块链中.

5) 终端会收到一个关于新区块的通知.

6) 受影响的设备 D_y 将会下载新区块中的配置文件, 解密配置文件并应用更改.

7) 设备 D_y 将添加一个新区块到区块链中, 新区块信息包括新的配置文件是否已应用成功、配置文件的散列值以及下载和应用的时间戳.

安全是本文对工业互联网平台设计的核心, 链码是支撑系统安全最重要的组成部分, 它处理身份验证、授权、加密、语法验证、访问控制和安全审计. 为了增强平台的安全性, 链码设计的重要原则是链码不能包含任何的密钥, 所有必须的密钥必须作为参数传递[17], 用户的身份信息是由用户名和密码组成一组键值对, 必须向链码提交正确的身份密钥, 才能对区块链网络中终端设备配置文件进行 CRUD 操作. 如果用户只能提供正确的登录密码而不能提供正确的密钥, 那么用户可以进入系统但是 CRUD 任何一个操作将会失败. 由于链码将尝试解密配置文件作为一种安全措施, 如果解密失败, 整个请求失败, 这就是本文前面提到的密钥验证程序.

CRUD (创建、读取、更新、删除) 是对终端配置文件主要操作, 所以其操作设计首先要具备安全性.

创建操作通过链码请求来创建设备新的配置文件,

通过 ACL 模块对用户的用户名和密码进行身份验证, 如果身份验证成功, 调用链码的授权函数对用户进行授权, 将用户创建的配置文件进行加密并存储到区块链中. 配置文件的数据通过哈希运算, 把其哈希值作为事务的参数一并存储到区块链中. 初始化创建操作后, 管理员将被通知结果并获得配置文件哈希值以及要创建的终端设备 ID, 创建操作的记录将被存储到区块链中.

读取操作包含登录名、密码、解密密钥和配置文件 ID 等字段. 读取操作的验证和授权的执行过程与创建操作相同. 如果成功, 将执行一个密钥验证过程. 密钥验证过程尝试使用提供的密钥解密配置文件. 一旦解密, 原先存储的配置文件哈希值将与新计算的配置文件哈希值进行比较. 如果哈希值相等, 密钥验证成功并解密, 根据发起者的请求将配置文件发送给管理员或设备. 如果请求来自互联网设备, 则有关应用程序状态的消息被发送到区块链中, 读取操作全过程被记录在区块链上.

更新操作与读取操作流程很类似, 更新操作请求包含诸如登录用户名、密码、解密密钥、原始配置文件 ID 和一个新的配置文件 ID 等参数. 身份验证和授权成功后, 将执行密钥验证过程. 如果成功, 则使用新的配置文件更新存储在区块链中的原始配置文件, 计算新配置文件的哈希值并存储到区块链网络中. 操作记录会存储在区块链上, 并将操作结果通知管理员.

删除操作与其他操作一样, 需要执行身份验证、授权和密钥验证等过程. 如果删除操作执行成功, 终端设备 ID 的配置文件被标记为已删除, 不允许进一步操作, 后续的读取、更新或删除操作结果与给定设备 ID 下没有存储配置文件的操作结果相同, 删除操作的记录也会永久的存储在区块链上, 并通知管理员删除操作的结果.

3.4 数据的签名打包存储

保证终端数据的完整性、可用性和安全性一直是工业互联网平台研究的重点之一, 区块链技术的出现为解决这一问题提供了理论和技术支持[18]. 区块链技术是结合密码学的分布式架构, 因此可以利用密码学的非对称加密算法, 为每一个终端设备生成各自的公私钥, 终端设备唯一 MAC 地址与该设备在区块链系统所对应的公钥形成一张设备终端与公钥对应的列表, 作为终端设备的身份证明, 从而保证了终端设备的唯一性及不可篡改性.

工业互联网中数据的采集通常是高频次, 对网络传输造成了巨大的压力[19], 利用区块链分布式的特点, 可以将原本中心化的信息采集转换为分布式的信息采集, 从而减轻工业互联网平台数据传出、存储和边缘层数据缓存的压力. 终端设备产生的数据, 利用其自身的公钥对数据进行加密处理即数据的签名打包, 加密过的数据只有设备自身的私钥可以对数据进行解密, 即使终端设备被恶意替换或者被破坏, 因为其密钥的唯一性, 可以进行及时的识别, 对没有身份证明的设备所产生的数据在边缘层数据处理过程中进行删除, 防止对平台数据的污染, 杜绝了因设备的外部破坏对平台造成的损失, 极大地提高了终端设备接入平台的安全性.

通过上述对终端数据进行签名打包后, 经过区块链网络的共识机制, 把打包好的数据信息存储在区块链上, 一经上链的数据信息都将永久的保存在区块链网络中, 而且是分布式进行存储, 避免了中心式服务器遭受攻击, 导致整个平台瘫痪的风险. 而且经过加密后的数据只有拥有终端设备私钥的管理员才能查看加密数据的详细内容[20], 保护了终端数据隐私安全. 没有权限的节点, 可以起到对终端数据审计的作用, 数据一旦上链就会生成唯一的哈希值, 并永久的存储在区块链中, 如果管理员自身作弊, 就会改变数据自身的哈希值, 从而改变整个区块链网络的数据结构, 此数据也就失去了可信性, 因此使用区块链技术可以保障工业互联网平台数据的可用性以及完整性, 以此来防止平台数据被恶意篡改. 数据处理的整个过程如图 5 所示.

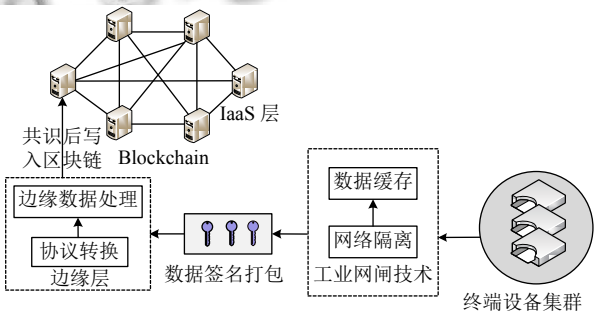

图 5 数据存储上链过程

4 实验分析

在本文的系统模型中, 我们通过定义不同的参与者来设定系统中的不同角色, 每个角色都有不同的等级的访问权限, 比如管理者和操作工. 本文使用 Hyperleager Caliper 测试框架来测试系统模型的运行性能, 根据测

试框架的架构, 测试模块有 3 个主要的函数分别是 init()、run()、end(), 其中 init() 函数处理测试的初始化阶段, run() 函数以异步方式重复生成和提交多个事务, end() 函数完成测试的最后部分. 本文将终端设备视为资产, 并把资产分配给不同的部门角色去管理. 使用 init() 函数来定义参与者的实例和资产, 随后测试事务通过 run() 函数被触发. 测试结果表明, 本文的系统模型在测试阶段能 100% 的正常运行. 图 6 和表 1 给出了由 Hyperledger Caliper 报告的性能指标和资源消耗的情况.

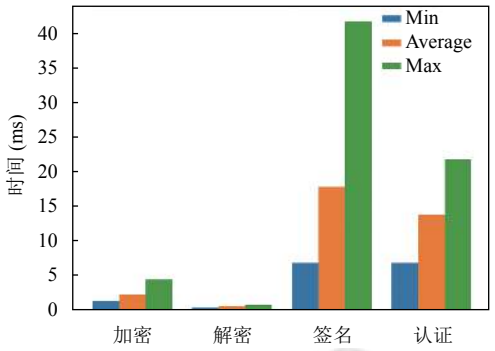

图 6 终端设备数据加密、解密、签名、认证的性能

表 1 资源消耗

类型	名字	内存 (max) (MB)	内存 (avg) (MB)	CPU (max) (%)	CPU (avg) (%)	流量进入	流量流出
Process	node bench-client	—	—	NaN	NaN	—	—
Docker	peer0.org1.example.com	356.5	323.4	19.32	12.12	17.8 MB	30.5 MB
Docker	peer0.org2.example.com	348.6	317.8	19.64	12.82	17.3 MB	29.6 MB
Docker	ca.org1.example.com	7.3	7.3	5.37	0.83	6.6 KB	4.5 KB
Docker	ca.org2.example.com	6.3	6.3	0.21	0.01	2.1 KB	0.05 KB
Docker	order.example.com	21.3	17.6	4.01	3.28	4.3 MB	6.7 MB
Docker	couchdb.org1.example.com	106	98	58.67	40.36	4.5 MB	8.3 MB
Docker	couchdb.org2.example.com	102	93	53.35	40.16	4.5 MB	8.3 MB
Docker	dev-peer0.org1.example.c···0.0.1	131.4	128.7	86.57	37.82	5.3 MB	3.8 MB
Docker	dev-peer0.org2.example.c···0.0.1	153.3	142.5	92.31	35.96	5.3 MB	3.4 MB

表 2 给出了设备配置文件在身份验证、下载和上传所需时间的 7 次测量. 为此场景设置了下载脚本的手动调用, TCL 脚本的大小为 1.15 KB, 新配置文件的大小为 1.97 KB, 设备之间的链接速度为 100 Mb/s. 对于 7 次不同时间测量的结果如表 2 所示.

表 2 身份验证、配置文件下载和上传时间的测量 (ms)

序号	验证	下载	上传	总时间
1	9	8	381	398
2	12	10	395	417
3	11	8	387	406
4	18	9	392	419
5	16	9	394	419
6	14	11	383	408
7	11	10	379	391

5 总结

在集中式系统中, 对资源的访问由第三方控制, 例如系统管理员, 他们完全控制系统的数据和操作, 因此他们总是遭受安全和信任问题的困扰. 本文是一个基于区块链技术的真实应用程序的实现, 用于探索区块链技术在工业互联网安全领域的应用. 通过利用 Hyperledger Fabric 和 Hyperledger Composer 的潜力, 我们实现了一个基于授权区块链的防篡改访问控制应用程序, 用于管理物理位置设备的访问和操作权限.

系统提供了全面的事务日志查询, 只对通过认证和授权的用户进行访问. 此外, 系统的交易历史是可信赖的, 因为它是受到保护, 不接受非法的篡改. 对终端设备配置文件的操作都会经过严格的身份授权并通过对操作人员的密钥验证来一步保障对终端设备的操作安全. 由 Hyperledger Caliper 报告的分析结果说明了系统的稳定性和可伸缩性非常理想, 包括 100% 的操作成功率和主要的性能指标.

在未来的工作中, 我们将继续通过区块链技术进行访问控制管理的研究来集成逻辑和物理的访问控制, 并采用可靠的方法分析安全方面的问题以及使用新技术来消除可能会被利用的漏洞.

参考文献

1 The Linux Foundation. Hyperledger fabric. http://hyperledger-fabric.readthedocs.io/en/release, 2020. [2020-12-22].

2 Androulaki E, Barger A, Bortnikov V, *et al*. Hyperledger

fabric: A distributed operating system for permissioned blockchains. Proceedings of the 13th EuroSys Conference. Porto: ACM, 2018. 30.

3 Vukolić M. Rethinking permissioned blockchains. Proceedings of 2017 ACM Workshop on Blockchain, Cryptocurrencies and Contracts. Abu Dhabi: ACM, 2017. 3–7.

4 Hyperledger fabric documentation. http://hyperledger-fabric.readthedocs.io/en/release/. [2020-12-22].

5 Seitz L, Selander G, Wahlstroem E, *et al*. Authentication and Authorization for Constrained Environments (ACE): draft-ietf-ace-oauth-authz-07. https://datatracker.ietf.org/doc/html/draft-ietf-ace-oauth-authz-07. (2017-08-08) [2020-12-22].

6 Rescorla E, Modadugu N. Datagram transport layer security version 1.2. http://www.rfc-editor.org/rfc/rfc6347.txt. [2020-12-22].

7 Yu JG, Zhang H, Li S, *et al*. Data sharing model for Internet of Things based on blockchain. Journal of Chinese Computer Systems, 2019, 40(11): 2324–2329.

8 Hernantes J, Gallardo G, Serrano N. IT infrastructure-monitoring tools. IEEE Software, 2015, 32(4): 88–93. [doi: 10.1109/MS.2015.96]

9 Shalunov S, Teitelbaum B, Karp A, *et al*. RFC 4656: A One-Way Active Measurement Protocol (OWAMP). Fremont: IETF, 2006.

10 Hedayat K, Krzanowski R, Morton A, *et al*. RFC 5357: A Two-Way Active Measurement Protocol (TWAMP). Fremont: IETF, 2008.

11 Yuan Y, Ni XC, Zeng S, *et al*. Blockchain consensus algorithms: The state of the art and future trends. Acta Automatica Sinica, 2018, 44(11): 2011–2022.

12 Huang ZY. The exploration and application of blockchain in industrial internet platform. Cyberspace Security, 2018, 9(10): 22–25, 33.

13 Hyperledger composer. https://hyperledger.github.io/composer/. [2020-12-22].

14 Rouhani S, Pourheidari V, Deters R. Physical access control management system based on permissioned blockchain. arXiv: 1901.09873, 2018.

15 Yakubov A, Shbair WM, Wallbom A, *et al*. A blockchain-based PKI management framework. Proceedings of 2018 IEEE/IFIP Network Operations and Management Symposium. Taipei: IEEE, 2018. 1–6.

16 Du MX, Ma XF, Zhang Z, *et al*. A review on consensus algorithm of blockchain. Proceedings of 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC). Banff: IEEE, 2017. 2567–2572.

17 Conte de Leon D, Stalick AQ, Jillepalli AA, *et al*. Blockchain: Properties and misconceptions. Asia Pacific Journal of Innovation and Entrepreneurship, 2017, 11(3): 286–300. [doi: 10.1108/APJIE-12-2017-034]

18 Košt'ál K, Krupa T, Gembec M, *et al*. On transition between PoW and PoS. Proceedings of 2018 International Symposium ELMAR. Zadar: IEEE, 2018. 207–210.

19 Zheng ZB, Xie SA, Dai HN, *et al*. An overview of blockchain technology: Architecture, consensus, and future trends. Proceedings of 2017 IEEE International Congress on Big Data (BigData Congress). Honolulu: IEEE, 2017. 557–564.

20 Azaria A, Ekblaw A, Vieira T, *et al*. MedRec: Using blockchain for medical data access and permission management. Proceedings of the 2nd International Conference on Open and Big Data (OBD). Vienna: IEEE, 2016. 25–30.