**·栏目· 文章编号：1000-3428(20xx)0x-00xx-xx 文献标志码：A 中图分类号：**

基于区块链的物联网云服务质量可信认证模型

谈玉胜1，柳毅2

(1.广东工业大学，计算机学院，广州 510006；2. 广东省大数据安全与服务工程技术研究中心，广州 510000)

**摘 要：**在物联网云服务质量评价体系中，现行的物联网云服务模型缺少完善自动执行的机制来维护客户的权益，现利用区块链技术不可篡改的技术特性，用户难以找到物联网供应商在服务期间存在缺陷的证据。因此，将服务质量缺陷的事件记录到区块链中。扩展现有的物联网服务评价模型，利用无偏向随机选择方法引入证人模型，用于检测和报告服务质量是否达到标准。引入了证人池机制，提出了一种无偏向排序算法保证了证人池中选择证人的随机性。同时，引入证人审查机制检测潜在的恶意证人。最后，利用以太坊智能合约创建系统原型，最终实验结果表明该模型可行，性能符合设计要求。

**关键词：**区块链；物联网；服务质量；以太坊；智能合约

Internet of Things Cloud Service Quality Authentication Model based on block chain

TAN Yusheng1, LIU Yi2

(1.Guangdong University of Technology, Guangzhou 510006, China；2. Guangdong Big Data Security and Service Engineering Technology Research Center,Guangzhou 510000,China)

**【Abstract】**In the Internet of Things cloud service quality evaluation system, the current Internet of Things cloud service model lacks a perfect automatic execution mechanism to safeguard the rights and interests of customers. With the immutable technical characteristics of blockchain technology, it is difficult for users to find evidence of defects of Internet of Things suppliers during service.Therefore, quality-of-service defect events are logged to the blockchain.The existing iot service evaluation model is extended, and the unbiased random selection method is used to introduce witness model, which is used to detect and report whether the service quality meets the standard.The mechanism of witness pool is introduced and an unbiased sorting algorithm is proposed to ensure random selection of witnesses in witness pool.At the same time, witness audit mechanism is introduced to detect potential malicious witnesses.Finally, ethereum smart contract is used to create a prototype of the system, and the final experimental results show that the model is feasible and the performance meets the design requirements.

**【Key words】**Block chain; The Internet of things; Service quality; Ethereum; Intelligent contract

# 概述

当前大多数的物联网云服务业务模型是：普通用户通过使用物联网服务商提供的计算、存储和api服务，共享服务器资源。这种共享模式给用户带来极大的便利，使得小规模或初级用户也能以较低的价格获取便捷完善的物联网服务体验；同时也带来了一些挑战，如物联网服务供应商在服务期间出现宕机，用户从供应商获取相应的赔偿，存在一定的困难。业内普遍的处理方案是共同执行服务水平协议（SLA，Service Level Agreement）,这是一个商业概念，定义了客户与供应商之间的合同和财务协议。如果在服务期间内服务方出现了故障，客户可以向服务方索取赔偿或抵消部分费用；如果在服务期间内，服务方没有出现故障，则客户正常支付全部费用。以此保障双方的合法权益。

现阶段,区块链技术与物联网技术进行结合，是一种发展趋势[1]。区块链技术的不可篡改特性，为物联网设备接入后的数据共享和协同工作提供了良好的保障。以太坊[2]中的智能合约提供了一种可行的方式来自动化执行服务水平协议。使用预言机[3]（Oracle）作为可信第三方，为智能合约提供可信的链外信息和数据，以验证触发智能合约的条件。

物联网服务水平协议出现的问题主要有以下三种：

1. 对于用户的补偿方面，缺少一种自动的机制来执行物联网服务水平协议。
2. 在当前的模型中，服务提供者拥有更多权利，特别是在验证物联网服务质量缺陷和决定是否对用户进行赔偿方面。
3. 用户难以提供证据证明物联网服务商确实存在质量问题。

物联网服务质量可信认证模型需要解决的问题:如何保证物联网服务水平协议服务商与用户的权益；由哪一方检测和证明违反了服务水平协议；如何实现支付流程的自动化执行，包括补偿流程的自动化。

本文在已有云服务水平协议的基础上，针对物联网云服务场景下用户权益的缺陷问题展开研究。主要贡献包括以下两个方面。

1. 引入了证人审计机制（WAM，Witness Audit Mechanism）
2. 结合区块链技术，制定了证人管理方案和无偏向随机选择方法（WMURS，Witness Management and Unbiased Random Selection）

# 1 相关理论与研究

## 1.1 SLA协议

SLA是云服务服务提供者与用户之间的服务质量协议[4]，可以适用于物联网云服务背景下，在协议被违反时保护用户的合法权益。SLA是连接用户与物联网服务提供者的桥梁，包含参数表示、参数映射、 违约惩罚三个方面。

SLA模型功能框架如图1所示。



图 1 SLA功能框架

一个典型的SLA生命周期由多个实施阶段组成，包含协商、建立、监控、违规报告和终止。研究工作主要集中在SLA参数的语法定义，便于系统的在线处理；优化资源分配算法，减少SLA冲突；根据云服务SLA的系统调查报告[5]，表明大部分的工作集中在监控和部署阶段，小部分工作集中在协商和建立阶段。目标是是的供应商和用户的利益最大化。在业界，Amazon推出了CloudWatch服务用于自动监控和通知SLA服务。但是，作为受信任的第三方执行监控和违规报告，用户没有别的选择，只能假定该服务是可信的。为了解决可信问题，还需要引入其它技术方案。

## 1.2 以太坊与智能合约

以太坊（Ethereum）是由比特币开发商Vitalik Buterin在2013年设计的[6]，用于促进区块链上分布式应用程序的开发。以太坊是一个开放的区块链平台可以建立和使用去中心化应用。以太坊是从2015年开始的第二代区块链，提出了以太坊虚拟机（Ethereum Virtual Machine）,这是一组表示虚拟机状态的字节值。在以太坊上运行的程序被称为智能合约（Smart Contract）。智能合约的工作流程如图2所示。



图 2 智能合约工作流程

智能合约是区块链上以计算机程序的方式定义的和自动执行的各种合约，用于开发分布式应用。 通过计算机协议促进、验证和执行合约。将智能合约与服务质量协议结合起来，重点注重语义的表达，实现协商阶段的不可篡改与自动化执行，具有一定的实用价值。区块链作为执行平台，链上的交互是不可变的，因此可以保证只能合约的可信度。

## 1.3 相关研究

国内外学者对物联网服务质量协议已有较多研究成果。这些方案都有其优势与不足。适用于不同的应用环境与网络模型。目前，物联网服务质量协议的可信模型还没有得到完全解决，还需要做进一步的深入研究。

文献[7]分别从硬件和传输协议角度保证了可信赖的执行环境。提出了一个认证的数据反馈系统：Town Crier，使用可信硬件后端，依靠智能合约提供源认证数据。该方案存在的问题是需要特殊的基础设施支持，实际应用环境中需要的是通用性与普适性的解决方案。文献[8]引入了一个服务器性能监视器的角色来检测违规并通知用户。提出了基于构建在区块链上的公共SLA契约方法，实现了SLA合同平台原型，并将其应用到实例中。然而提议的解决方案缺乏对已识别违规行为的可信性分析，并且仍然面临在区块链之外发生事件的共识挑战。

对于基于区块链的系统当事件发生在区块链之外时，如何可信地将随机事件上链。文献[3]的解决方案是使用预言机充当区块链的数据载体代理。Oraclize作为可信赖的第三方机构提供预言机服务。但是它可能会存在单点故障，这就偏离了区块链的去中心化思想。为了解决这个问题，文献[9]提出了在分布式预言机上工作，其工作方式是：只有在预言机之间达成协议时，事件的结果数据才会上链并触发事务。但是这个理念也存在一定的不足，对于预言机而言，如果没有足够的激励机制，预言机自身的可信问题并不能得到解决。

# 2 基于智能合约的证人模型

本节将介绍基于区块链的认证模型，具体介绍证人的角色和功能，如何检测和报告违规行为，以及在区块链上如何使用智能合约实施服务质量协议的系统架构。

## 2.1 引入证人池机制的SLA协议

证人是区块链的正常参与者，它们是自愿参与到系统中并且通过监控服务获取自身收入。证人角色默认是自私的，以获得自身收益最大化为目标[10]。

对于传统的物联网服务质量协议，主要参与的两个角色是物联网云服务提供商Provider和物联网客户Customer。使用一个基础示例来演示SLA协议的工作流程。假设一个物联网云服务提供商根据客户的请求，提供一个具备公网IP地址并定义了数据节点类型的服务器Virtual Machine。在有效服务时间Tservice内，客户Customer可以通过透传连接到服务器。服务质量协议SLA可以做出承诺：在有效服务时间Tservice内，Provider提供的服务器始终是可访问的，如果满足条件，在服务结束后Customer必须支付费用Fservice，否则在违反SLA的情况下，Customer必须获得一笔补偿费用Fcompensation。这里我们假设Fservice＞Fcompensation，也就是说Customer需要给Provider支付的费用为Fservice - Fcompensation。此外，对于Customer自身的网络问题引起的无法访问服务器不属于违反SLA协议[11]。

为了确认是否真的违反了SLA协议，需要在传统SLA协议角色的基础上，再引入一个新的证人角色Witness，利用以太坊交易平台创建一个可信的交易环境。这些Witness也是区块链的一般参与者，自愿参数SLA协议，通过提供监控服务获得收入。为了解决SLA协议中的信任问题，创建一个由n个Witness组建的证人委员会{w1,w2,w3,…,wn}来参与SLA协议，它们一起监控SLA协议中的违规事件。并从Provider和Customer双方获得证人费作为回报。这里对证人角色做一个假设，证人都是自私的，都以自身收益最大化为目标。对于不符合假设的证人则视为懒惰证人，会通过下文算法将其排除在证人委员会之外。

## 2.2 系统架构

系统由两类智能合约构成，第一类是证人池智能合约，是系统的基本智能合约；第二类是特定的物联网服务质量协议智能合约，用于SLA的自动执行。证人池智能合约主要有三个功能，分别是证人管理、具体SLA的生成和证人委员会的选举。区块链的任何用户都可以通过钱包地址在证人池中注册，成为证人。用户可以保持在线状态，等待被选中执行SLA协议。所有的证人参与这个系统的动机都是获得收益，系统中的证人越多，系统越可信[12]。系统整个架构如图3所示。



图 3 系统架构图

物联网服务供应商Provider利用证人池智能合约为自己生成SLA智能合约，Provider与Customer协商详细的SLA条款，服务时间Tservice、服务费用Fservice、补偿费用Fcompensation以及确定执行本SLA协议的证人Witness的数量。证人越多违规检测的结果越可信，同时Provider与Customer也要共同额外支出更多的费用。系统可以根据Customer表单提交的结果，Provider传入这些参数在证人池智能合约中生成新的SLA智能合约。同时利用公正和随机的无偏向排序算法，从证人池中选举证人委员会。证人委员会中的成员是独立公正的无偏袒的，监控SLA协议的执行，于此同时，Provider提供物联网服务器给Customer使用，并且在SLA中发布服务细节。接着，证人委员会开始监测服务，并将服务器公网IP通知给证人委员会成员。服务期间Customer可以正常使用服务器，证人委员会中的证人也可以不断ping服务器的ip地址，如果在Tservice内服务器出现故障，证人立即向智能合约报告违规事件[13]。

具体的SLA工作机制为：当系统收到证人委员会中的第一个举报时，开始计时，此后，SLA智能合约接收来自委员会中其它证人的举报。直到没有其它证人举报了为止[14]。如果SLA收到证人委员会中举报的人数X，证人委员会的人数为Y，当前仅当X/Y > 1/2时，Tservice期间内的违规行为将会被自动确认。例如，证人委员会中有三个成员参与监控SLA协议，只有其中至少两个证人报告该事件时，才能确认服务违规。同时，该SLA智能合约也定义了如果同一证人在Tservice时间内只有一次报告机会，预防恶意证人多次汇报扰乱机制。每一个证人都想要最大化自己的收益，他们都是诚实的参与者根据实际情况报告违规情况。最后SLA智能合约存在两种执行结果：一种是Tservice时间后，没有违反协议，Customer正常支付的费用为Fservice；另一种是Tservice时间后，报告违反了协议，Customer需要获得补偿，所需支付的费用为Fservice - Fcompensation；不论哪种情况，证人委员会都会从智能合约中获取报酬。

# 3 核心算法

物联网服务质量可信认证模型所采用的关键技术是对SLA违规行为进行自动检测，得到使Customer和Provider的都信服的检测结果。使用无偏向排序算法，确保被选入证人委员会的证人都是随机且独立的。同时也使用了证人审计算法，分析了来自恶意证人的欺诈行为，确保参与者必须诚实才能利益最大化。

## 3.1 无偏向随机选择算法

从证人池中选举证人委员会的方法使用的是无偏向随机选择算法，针对特定的SLA合同证人的选择必须是公正的，引入以太坊作为模型的可信方，在证人池智能合约中实现该算法，对于Customer或者Provider都不能有偏向。算法伪代码如算法1所示。

**算法1** 无偏向随机选择算法

**输入**：

已注册的证人集合set\_registered；

在线证人数量online\_witness;

证人委员会中需要的人数N；

第b个区块链的哈希值

生成的新区块链长度

证人的在线状态state

证人的信誉度：reputation

证人的地址witness\_address;

**输出**：

1.seed = 0; // 初始化随机数种子

2.for():

3. seed += ; // 将个区块哈希值累加

4.end for // 结束循环获取随机种子

5. for(int j = 0; j < N; j++):// 选择证人

6. index = seed % len(set\_registered)+1;

7. check(set\_registered[index].state)

&& check(set\_registered[index].reputation)

//判断证人在线状态与信誉度

8. set\_registered.add(witness\_address)

//将证人添加到证人委员会集合中

9.end for //证人满足数量结束循环

10. return set\_registered;//返回证人委员会集合

在证人池智能合约中，开放了提供注册接口，区块链用户都可以注册到证人池中，同时区块链用户可以将自身状态切换为“在线”或“离线”，表明当前是否愿意加入证人池选举活动。证人池中存储的是地址列表记录了区块链用户注册的顺序。 为了从证人池中选择N个证人委员会成员，证人池智能合约定义了两个接口：请求和排序。利用区块的哈希值生成随机数种子，新的种子是根据前一个种子的哈希值生成的，这个过程重复进行，直到选定出N个证人委员会成员[15]。这样既保证了算法的输出，又保证了算法的随机性，无论是对于Customer还是Provider都不能随意操纵排序结果，从而控制证人委员会的选举结果。

## 3.2 证人审查选择算法

无偏向排序算法保证了选举的证人的公平独立，但是仍然需要一种审计机制来保证系统能够检测到恶意或者懒惰证人，并将其从证人池中开除[16]。由于智能合约中的交易是公开并永久存储在链上的，因此可以通过目击者的行为历史来审查证人的声誉，而不是根据其它人的反馈来评估证人的信誉度[17]。

**算法2** 证人审查算法

**输入**：

恶意证人集合set\_malicious；

默认声誉值reputation;

证人委员会set\_registered；

**输出**：

1. flag = event.isValid;//获取SLA结果

//获取证人委员会中所有投赞成票的集合

1. setYes = set\_registered.getYes();

//获取证人委员会中所有投反对票的集合

1. setNo = set\_registered.getNo();

//对证人委员会的每一个地址进行遍历

1. for(String address : set\_registered):

// 如果最终结果为违规

1. if(flag==false):

// 所有赞成票的地址扣除声誉

1. for(String address:setYes):

address.reputation--;

else:

// 否则所有反对票的地址扣除声誉

1. for(String address:setNo):

address.reputation--;

// 如果证人声誉小于0

8. if(address.reputaion < 0):

// 将证人地址加入恶意地址集

set\_malicious.add(address);

一般情况下，恶意证人主要分为两类：懒惰证人[18]和牺牲证人。懒惰证人是指不愿意举报违规行为的证人，或者举报违规行为的收益不足以激励证人工作，他们选择不真正监视的策略总是保持沉默，从不报告违规行为，使用这种策略后。即使SLA最终状态被违反，懒惰证人也不会付出代价。同时，整个系统中由于总体报告违规的的次数应小于正常的次数导致懒惰证人仍然能够通过参与多轮服务总体获得收益。牺牲证人是指总是在特定时间点报告的证人，例如证人总是在SLA启动1分钟内报告违规，虽然一开始可能由于自己的恶意行为受到惩罚，但是可以在以后的历史中向他人展示自己的行为模式，从而获得最大利益。为了审查这些恶意证人，系统需要为我设计声誉值参数，当违规行为被证实时，没有做出举报行为的证人声誉扣除1点，举报违规行为，最后系统并没有违规的证人同样扣除1点声誉值。当证人的声誉值降至为0时，证人的选举流程也被终止，提出证人池，系统恶意地址表中记录恶意证人的地址，恶意证人无法再次加入证人池。这种算法的设计能有效减少懒惰证人与牺牲证人的参与，提高了SLA系统的稳定性与安全性。

# 4 实验与分析

为了测试当前物联网服务质量模型的可用性，在以太坊区块链测试网[19]“Rinkeby”上部署实现SLA智能合约。以太币是以太坊的加密货币，将用户的MetaMask账户地址发布到社交媒体上，并将该链接提交至Rinkeby便可以领取以太币，用于调试智能合约。在Rinkeby上生成若站账户来模拟不同的SLA模型中的角色，Provider、Customer、Witness。在每个模拟账户上使用以太币来执行接口。根据模型预付不同类型的费用。在实验前，首先部署证人池智能合约，让所有证人账户注册到证人池。Provider生成物联网SLA智能合约以启动SLA生命周期。测试多种场景，以验证不同接口的功能。结果表明，当前系统模型满足设计要求，无偏向排序算法和证人审查算法保证了系统的可信度。

实验主要分析该模型的性能信息，测试智能合约中接口的复杂性。由于需要执行接口中定义的程序，需要消耗电能。接口定义的工作逻辑越复杂，调用时所需要的事务费用就越多。在以太坊中，矿工在工作时，衡量工作量的单位是gas。因此我们可以通过实验记录每一个接口中的gas消耗量[20]。各接口gas消耗量如表1所示。

表1 SLA智能合约中各接口能耗分析比较

|  |  |  |  |
| --- | --- | --- | --- |
| 接口类型 | 用户 | 服务商 | 证人 |
| 注册 | 94567 | 128778 | 78456 |
| 打开 | 46387 | 56787 | 34567 |
| 关闭 | 16878 | 19452 | 12765 |
| 拒绝 | 23236 | 20222 | 18345 |
| 排序 | 45654 | 56787 | 35673 |

Provider与Customer相比，在整个生命周期中，Customer和Witness接口的消耗量更少。这符合我们模型的设计需求和现实需要，因为在大多数情况下，Provider在提供服务时赚取最多的收益，他们具有在SLA生命周期中提供更多资源的动机。图4显示了实验的研究结果。

图 4 SLA各角色接口工作量

Customer和Witness的消耗量是在基础的实验结果上得出的，还有进一步的优化空间，总体实验结果满足设计需求。

# 5 结束语

提出了一种物联网SLA实施的证人模型，确定了证人管理和无偏随机选择的机制；利用算法分析得出证人必须提供诚实的监督服务，以实现自身收益的最大化，同时提出了证人审查算法。最后利用以太坊的智能合约实现了一个原型系统。

实验研究证明了模型的可行性，将信任问题转化为经济问题，证人机制的合理性使得证人由于经济原则不得不诚实。对于物联网用户的与服务供应商之间的交易的公平与安全有着显著的意义。对于未来的工作，文中提出的可信SLA服务质量模型可以结合车联网、移动计算等领域提供可信的服务质量协议框架，具有广阔的应用前景。

参考文献

1. Ali M S, Vecchio M, Pincheira M, et al. Applications of blockchains in the Internet of Things: A comprehensive survey[J]. IEEE Communications Surveys & Tutorials, 2018, 21(2): 1676-1717.
2. Buterin V. A next-generation smart contract and decentralized application platform[J]. white paper, 2014, 3(37).
3. (2014) Ethereum and Oracles. [Online]. Available: https://blog.ethereum.org/2014/07/22/ethereum-and-oracles/
4. Alhamad M, Dillon T, Chang E. Conceptual SLA framework for cloud computing[C]//4th IEEE International Conference on Digital Ecosystems and Technologies. IEEE, 2010: 606-610.
5. Faniyi F, Bahsoon R. A systematic review of service level management in the cloud[J]. ACM Computing Surveys (CSUR), 2015, 48(3): 1-27.
6. Wood G. Ethereum: A secure decentralised generalised transaction ledger[J]. Ethereum project yellow paper, 2014, 151(2014): 1-32.
7. Zhang, Fan, et al. "Town crier: An authenticated data feed for smart contracts." Proceedings of the 2016 aCM sIGSAC conference on computer and communications security. 2016.
8. Nakashima H, Aoyama M. An automation method of sla contract of web apis and its platform based on blockchain concept[C]//2017 IEEE International Conference on Cognitive Computing (ICCC). IEEE, 2017: 32-39.
9. Ellis S, Juels A, Nazarov S. Chainlink: A decentralized oracle network[J]. Retrieved March, 2017, 11: 2018.
10. Scheid E J, Rodrigues B B, Granville L Z, et al. Enabling dynamic sla compensation using blockchain-based smart contracts[C]//2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM). IEEE, 2019: 53-61.
11. Wonjiga A T, Peisert S, Rilling L, et al. Blockchain as a trusted component in cloud SLA verification[C]//Proceedings of the 12th IEEE/ACM International Conference on Utility and Cloud Computing Companion. 2019: 93-100.
12. Maarouf A, Mifrah Y, Marzouk A, et al. An autonomic SLA monitoring framework managed by trusted third party in the cloud computing[J]. International Journal of Cloud Applications and Computing (IJCAC), 2018, 8(2): 66-95.
13. Ghosh N, Ghosh S K. An approach to identify and monitor SLA parameters for storage-as-a-service cloud delivery model[C]//2012 IEEE Globecom Workshops. IEEE, 2012: 724-729.
14. De Carvalho C A B, de Castro Andrade R M, de Castro M F, et al. State of the art and challenges of security SLA for cloud computing[J]. Computers & Electrical Engineering, 2017, 59: 141-152.
15. Venticinque S, Aversa R, Di Martino B, et al. A cloud agency for SLA negotiation and management[C]//European Conference on Parallel Processing. Springer, Berlin, Heidelberg, 2010: 587-594.
16. Zhou H, Ouyang X, Ren Z, et al. A blockchain based witness model for trustworthy cloud service level agreement enforcement[C]//IEEE INFOCOM 2019-IEEE Conference on Computer Communications. IEEE, 2019: 1567-1575.
17. Wang X, Su J, Hu X, et al. Trust model for cloud systems with self variance evaluation[M]//Security, Privacy and Trust in Cloud Systems. Springer, Berlin, Heidelberg, 2014: 283-309.
18. Arafat N A, Basu D, Bressan S. Topological Data Analysis with epsilon net Induced Lazy Witness Complex[C]//International Conference on Database and Expert Systems Applications. Springer, Cham, 2019: 376-392.
19. Pierro G A, Rocha H. The influence factors on ethereum transaction fees[C]//2019 IEEE/ACM 2nd International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB). IEEE, 2019: 24-31.
20. Ranganthan V P, Dantu R, Paul A, et al. A decentralized marketplace application on the ethereum blockchain[C]//2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC). IEEE, 2018: 90-97.

请在文章最后给出前3位作者的联系方式(长期有效的手机、电话及E-mail。

谈玉胜 15622331365 tanyusheng@outlook.com

柳毅 13434352431 yiliu@gdut.edu.cn