基于区块链的医疗数据共享与隐私保护模型的设计与实现

摘要

本系统是针对协同诊治医疗系统下的一个子系统进行优化，并结合区块链技术对其重新设计，旨在改善原有医疗系统中电子医疗数据的存取、共享方式。系统引入去中心化分布式数据存储服务，并利用以太坊平台所提供的区块链服务，采用。。。。。技术，或创新性改进什么算法或模型等等，都可以写在这里，否则，摘要感觉没做什么东西。。。。从而实现电子医疗数据能在具有很高的安全性下进行存储、共享，从而实现对患者的隐私保护。经过实践，这一套子系统在实现医疗数据共享的同时能够保证极高的安全性，换言之，对患者隐私的保护程度是极高的。

关键词：医疗数据；区块链；隐私保护；共享

**Design and implement of medical data sharing and privacy protection model based on Blockchain**

**Abstract**

This system is an optimized and redesign system with Blockchain implement of Collaborative Diagnosis and Treatment Medical System aiming at optimizing the way original medical data access and sharing did in old medical system. This system uses decentralized database service and Ethereum blockchain service to access and share medical data in a very high security level in order to protect patient’s privacy. Practice shows that this subsystem has a very high security level when accessing medical data, which is greatly usefully for protecting patient’s privacy.（摘要定稿后再翻译）

Keywords: Medical data, Blockchain, Privacy protection, Sharing

目录(目录须与正文一致)

[1 绪论 4](#_Toc6996070)

[1.1 设计与实现的背景、目的和意义 4](#_Toc6996071)

[1.2 国内外研究现状 5](#_Toc6996072)

[1.3 论文内容与组织 6](#_Toc6996073)

[2 区块链相关理论 6](#_Toc6996074)

[2.1 区块链 6](#_Toc6996075)

[2.1.1 分布式账本 7](#_Toc6996076)

[2.1.2 非对称加密和授权技术 8](#_Toc6996077)

[2.1.3 共识机制 9](#_Toc6996078)

[2.1.4 智能合约 10](#_Toc6996079)

[2.2 默克尔树 12](#_Toc6996080)

[2.3 以太坊 13](#_Toc6996081)

[2.4 本章小节 13](#_Toc6996082)

[3 基于区块链分布式存储系统在医疗系统上的应用方案 14](#_Toc6996083)

[3.1 研究目的和研究内容 14](#_Toc6996084)

[3.2 拟解决的关键科学问题 14](#_Toc6996085)

[3.3 电子病历和电子医疗数据共享框架研究 14](#_Toc6996086)

[3.4 在电子医疗数据共享模型上的分布式存储系统 15](#_Toc6996087)

[3.5 电子医疗数据共享模型安全机制 16](#_Toc6996088)

[3.5.1 小节概览 16](#_Toc6996089)

[3.5.2 AES加密 16](#_Toc6996090)

[3.5.2.1 AES数据结构 16](#_Toc6996091)

[3.5.2.2 AES加密过程 17](#_Toc6996092)

[3.5.2.3 AES解密过程和AES使用示例 18](#_Toc6996093)

[3.5.2.4 小节总结 19](#_Toc6996094)

[3.6 本章总结 19](#_Toc6996095)

[4 医疗数据共享与隐私保护电子病历系统设计 19](#_Toc6996096)

[4.1 整体框架设计 19](#_Toc6996097)

[4.2 存储设计 21](#_Toc6996098)

[4.3 开发环境/运行环境 21](#_Toc6996099)

[4.4 开发技术 22](#_Toc6996100)

[4.4.1 Node.js和Express 22](#_Toc6996101)

[4.4.2 Solidity 22](#_Toc6996102)

[4.4.3 AES加密算法实现 24](#_Toc6996103)

[4.5 API设计、界面设计与演示 25](#_Toc6996104)

[4.6 系统正确性分析 26](#_Toc6996105)

[4.7 本章小结 27](#_Toc6996106)

[5 问题与解决方案 28](#_Toc6996107)

[5.1 遇到的技术问题和解决方法 28](#_Toc6996108)

[5.2 技巧总结 28](#_Toc6996109)

[5.3 本章总结 28](#_Toc6996110)

[6 总结与展望 29](#_Toc6996111)

[6.1 总结 29](#_Toc6996112)

[6.2 展望 29](#_Toc6996113)

[致谢 29](#_Toc6996114)

（下面正文的精炼表述还得加深，要深入把自己做的工作不含糊的展现清楚。我不具体改）

# 绪论

## 设计与实现的背景、目的和意义

数字化医疗管理信息是未来发展趋势，提出和研发新世代医疗数字服务的解决方案是我国电子健康研究计划的计划之一，新解决方案不仅在改善医疗水平，还将降低医疗成本，并可解决病患的隐私安全这一十分重要且颇具挑战性的问题。

医疗数据是病人不可侵犯的隐私之一。电子医疗数据泄露的问题常有发生。对于储存医疗数据的中心服务器，其责任相当重大，若它被不法分子攻破，大量医疗数据将外泄，这是对患者的隐私极大的侵犯。这也是传统数据库的缺点之一，传统数据库使用客户机-服务器的网络体系结构，中央服务器对访问的用户进行认证和授权，中心服务器的权力高度集中，借助其权限可以访问、修改甚至删除各个文件。在传统的解决方案中，医疗数据的权限、隐私、安全等各方面条件仍需要进一步改善，要在保持数据的完整性和无误性的条件下提高其安全性。可见这并不是一个可靠的解决方案。

在处理电子医疗数据这种涉及隐私的数据中，数据的安全性表现尤为重要。目前，整体趋势是利用区块链技术对患者医疗数据进行存储，这样相比传统型的解决方案表现更为安全。

区块链数据库由分散的节点组成，完整性和透明性是区块链的关键特征。完整性指数据的完整性能得到保证，透明性表示每个用户都可以验证区块链信息。如果借助区块链数据库，上述缺点将可被完美解决。区块链所提供的安全性、透明性、完整性正是医疗数据共享与隐私保护所需的。目前，区块链社区已经相对比较成熟，且区块链技术在医疗行业也有很多相关解决方案。

本模型是针对协同诊治医疗系统下的一个子系统的优化。每个医院都有其一套系统，都有其自己的数据库、前端机、服务器等。本系统旨在独立在各个医院外实现一个系统，让数据能正确安全地接入每个医院的前端机，即该系统是对医院和病人分别开放的，病人和医院均可使用该系统，病人使用该系统去授权医院资格以查看和修改病历。

## 国内外研究现状

目前，国内推崇医疗大数据，其旨在解决就诊便利性问题，各大医院都有着其自己管理的平台，并且拥有其提供给用户使用的客户端，以获取病人数据。在大数据时代，医疗数据保护手段主要还是采取传统数据的隐私保护手段，如密码学。在这方面，学者们提出了数种医疗保护方法，其中包括标识隐私匿名保护、医疗数据的分级保护制度、基于访问控制的隐私保护。在数据发布领域，学者提出了多种隐私保护模型，主要包括：t-closeness、I-多样性、k-匿名模型等。这些模型的原理都是通过隐藏公开的隐私数据和具体个人之间的关系，但是保证发布数据的信息公开可用[[[1]](#endnote-1)]。

鉴于对区块链的监管制度在全世界各地很多地方还没得到很好的完善，区块链在医疗数据或者是医疗方面都处于起步阶段，无论是国外还是国内，区块链技术的应用都未广为普及，在医疗领域中，无论是推广还是运行都还面临了很多未知。对于其去中心化的特性，传统的医疗管理机构也将受到冲击。[[[2]](#endnote-2)]所以区块链在医疗数据层面还是有很大的可探索空间。

## 论文内容与组织

本文是对医疗协同诊治系统下的病患就诊和医生看诊的流程进行了重新设计，文中包含对该流程下每个子流程的技术作介绍，以及如何结合以太坊提供的区块链服务和结合AES加密算法来实现共享医疗数据过程的隐私保护。

本文的主要分为六章。

第一章，绪论，分析当前数字化医疗系统以及医疗数据电子化及其设计的隐私问题在医疗领域的情况，以及就当前区块链发展形势对区块链在医疗方面的应用的现状和未来作分析。

第二章，相关理论介绍，介绍系统设计所涉及的相关理论和技术知识。

第三章，基于区块链分布式存储系统在医疗系统上的应用方案，介绍如何基于区块链去设计系统。

第四章，医疗数据共享与隐私保护电子病历系统设计，对设计进行实现，并且介绍电子病历系统的设计和如何对电子病历相关信息加密存储。

第五章，总结系统设计与开发过程所遇到的问题和解决方案。

第六章，总结与展望。

# 2 区块链相关理论

## 区块链

区块链一词产生于比特币白皮书的译文。区块链技术是一种结合了分布式数据存储、点对点传输、加密算法和共识机制等技术的应用模型，它本质上就是一个去中心化的数据库。在传统的数据库——或者说“中心化”的数据库，在安全上有很致命的缺陷，例如，它由于数据高度集中，数据可以在用户全然不知情的情况下被篡改甚至被删除，也可以在没被用户授权的情况下被他人所阅览。区块链能很好地解决这一问题

区块链上的是区块，区块是主要存储数据的单位，在区块链最出名的应用——比特币上，区块主要用来存储交易记录。每一个区块首尾链接，跟双向链表一样，前一个区块记录着后一个区块的唯一标识，后一个区块也记录着前一个区块的唯一标识。区块以某种按照时间顺序的排列规则，或者说产生规则，与前后的区块串联起来，形成区块链。广义来讲，区块链技术是利用块链式数据结构来验证与存储数据、利用分布式节点共识算法来生成和更新数据、利用密码学的方式保证数据传输和访问的安全、利用由自动化脚本代码组成的智能合约来编程和操作数据的一种全新的分布式基础架构与计算方式。[[[3]](#endnote-3)]上述是非常关于区块链清楚且准确介绍。或者通俗地讲，区块就是一个个收纳盒，收纳盒的表面上贴了两个标签，一个是记录前一个收纳盒的地址，一个记录的是后一个收纳盒的号码。收纳盒中放着很多东西，这些东西都是又一张表来决定，这张表决定了收纳盒中可以放什么内容，怎么放等规则。任何在收纳盒的内容发生一点点改变都会改变收纳盒的地址，比如说，小明想偷偷把某个收纳盒里的一封信改一个字，该收纳盒的地址为0x62b00Bd5dE47b22eA98922f4020fa0BaB85ad32B，小明把信拿出来，改了信里面的一个字，然后把信放回原来的收纳盒中。立刻地，信被拒收了，收纳盒中生成了原来一封一摸一样的信。如果把时间放很慢很慢地看这一过程，在小明把信放回收纳盒时，收纳盒的地址发生了改变，变成了0x42b00235dE47a22e365922f4020fa0Cad85ad84c，跟原来的完全不一样。然后该收纳盒前后的收纳盒发现它的地址与自己记载的不符合，马上拒绝了小明的行为。这一过程在区块链中实现是默克尔树。该一系列收纳盒由以太坊提供，以太坊提供了盒子给我们，我们可以在盒子里面按照一定规则去存取物品，这规则就是智能合约。以太坊对智能合约的操作我们也称其为交易，因为操作会消耗一定的以太币。图示如图1:

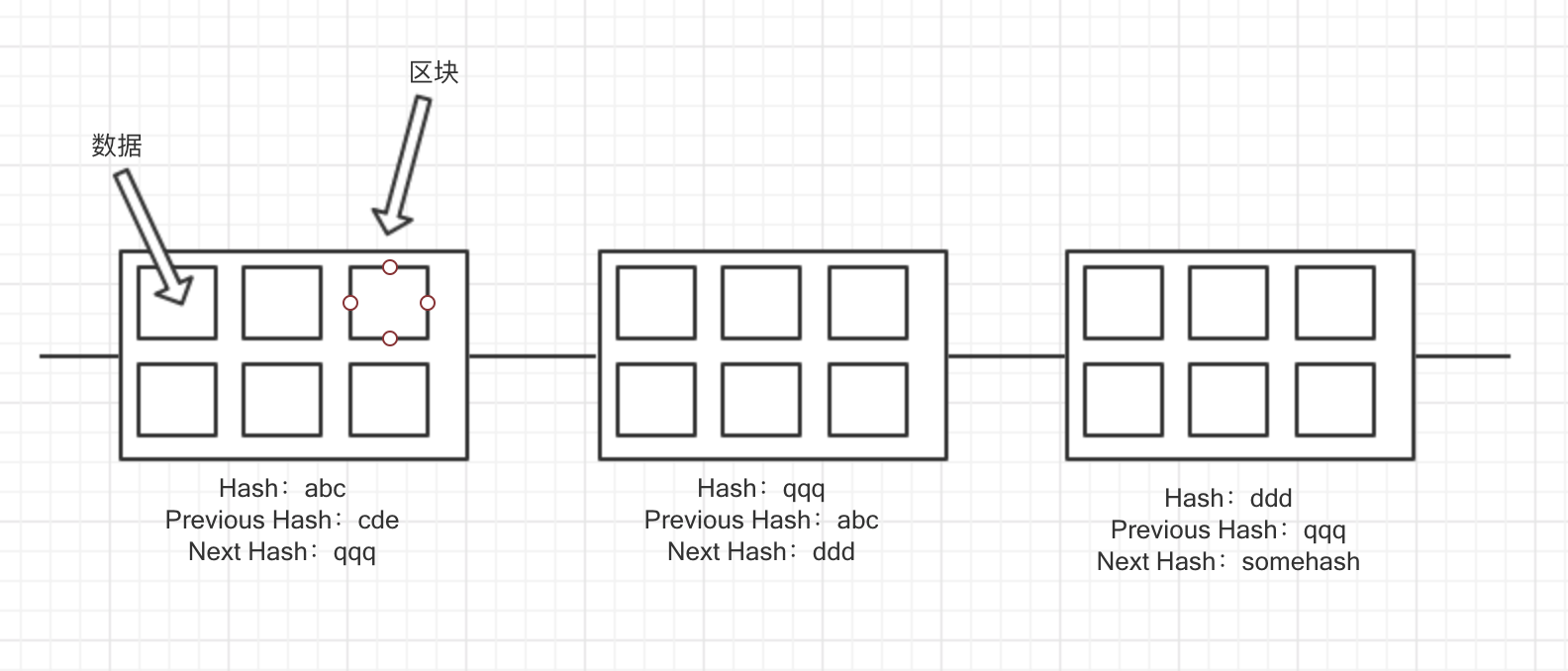


图1 图名？

区块链主要目的是解决交易的安全和信任问题，就这方面，区块链有如下四个方面的技术创新。

## 分布式账本

第一个技术创新是分布式账本（Distributed ledger），该技术是指交易记账由在不限地方的多个节点共同完成，每一个节点都记录了区块链上完整的账本，所以每个人都可以参加和监督交易的合法性。分布式账本，从实质上看，是一个可在不同地理位置或者在多个站点，又或者多个机构所组成的，在网络中进行数据分享的数据库。该网络中的所有参与到记账的节点，都将获得一个唯一且记载着准确数据的账本的副本。账本中一切都改动，都会使得网络中其他所有副本察觉到并对其作出反应，而且反应时间极短，故想对账本做不合法的举动，都会马上被拒绝。在账本里存储的信息的安全性和准确性是通过非对称加密算法去控制账本的访问权。由网络中所采取的共识规则，账本中所记录的内容可由一个或者多个，又或者是所有参与者共同更新。

## 非对称加密和授权技术

第二种技术是非对称加密和授权技术，该技术是指存储在区块链上的交易信息是完全公开的，但是账户的身份信息是完全高度加密的，只有数据的拥有者授权之后才能访问到。非对称加密的对立面就是对称加密，非对称加密也是就对称加密的弊端进行改进而产生的。在不安全的媒体环境下通讯双方互相交换信息，并且保证了信息数据安全地达成一致的加密算法，就是非对称加密算法。异于对称加密算法，非对称加密算法有两个密钥，分别是公开密钥（public key）和私有密钥（private key）。通信双方分别都拥有其自己的公开密钥和私有密钥，若用公开密钥来进行数据加密，那么只能用对应的私有密钥才能对数据解密；若用私有密钥进行对数据加密，那么只能用对应的公开密钥才能对其解密。因为加密和解密两个过程使用的是不同种类的密钥，所以这种算法叫非对称加密算法。

基于上述特性，区块链使用了非对称加密算法来进行信息数据转移。公开密钥，顾名思义，是对全网络公开可见的，网络中的任何人都可以用公开密钥来加密信息，保证其信息的真实性和公开透明性；私有密钥则只有信息接收者或者拥有者才知道，被加密过的信息数据只有通过对应私有密钥才能解密，这就保证了信息的安全性。以比特币网络为例，公开密钥就是我们的比特币接收地址，相当于我们的银行卡号，私有密钥则是我们的银行卡密码。若网络中A想发送1个比特币给B，首先B要将其公开密钥发送给A，然后A首先要用自己的私有密钥把比特币解密，再用B的公开密钥对信息进行加密，最后再发送给B。B接收以后，若要使用或转移比特币，用自己的私有密钥解密即可。通俗地讲，私有密钥是用来对信息进行签名的，私有密钥的持有人是否发送信息或者转移出价值等操作可以通过公开密钥签名验证信息来确认。

在区块链系统中，信息转移过程的信任机制主要是通过“非对称加密算法”来完成实现的。区块链上的交易（Transaction）信息是公开透明的，但账户的身份信息却是高度加密的，只有在数据拥有者授权的情况下才能访问到，身份验证授权保证了账户数据的安全和个人的隐私。

## 共识机制

第三种技术是共识机制，这个技术描述的是所有记账的节点之间是怎么样达成共识的，一个记录是否有效，取决于共识机制。这不仅仅是一种认定手段，也是防止数据被篡改的手段。区块链有几种不同的共识机制，适用不同的场景来应用。区块链的共识机制都具备“人人平等”和“小数服从多数”的特点，“人人平等”指当该节点满足所需条件时，所有节点都有权力去优先提出共识结果、直接被其他节点认同后，并切有可能成为最终共识结果。“小数服从多数”是指某个量所占地比例比分较大决定结果，该量不仅仅可以是节点个数，也可以是股权数等其他量。

比特币采用的共识机制是工作量证明（Proof of work），简单理解就是一个证明，证明确认了你做过的工作量。如比特币圈常常有说到的“挖矿”就是工作过程。工作量证明系统主要的特征是该计算机需要做一个有一定难度的工作才能得出一个结果。验证方则很容易就可以通过其结果来检查该计算机是不是做了一定量的工作。这方案核心的特征是不对称性——工作对请求方是适中的，对于验证方来说是易于验证的。图示为工作量证明的过程。流程如图2

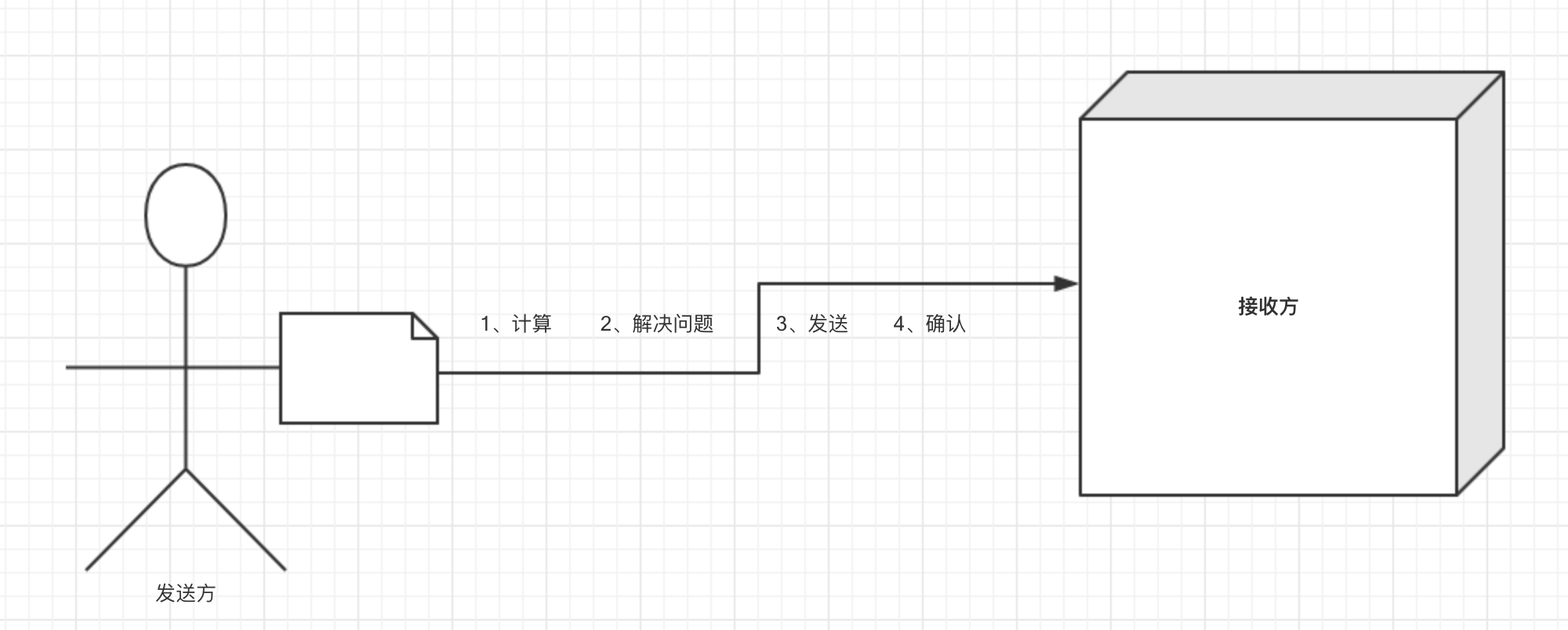


图2 图名

在比特币网络里任何一个节点，要生成一个新的区块，并将其写入区块链，此前必须解出比特币网络出的工作量证明的难题。这难题关键的三个要素是工作量证明函数、区块及难度值。工作量证明函数是这道题的计算方法，区块决定了这道题的输入数据，难度值决定了这道题的所需要的计算量。

以太坊本来采用的也是Proof of Work工作量证明共识算法，但是Proof of Work工作量证明共识算法最大的缺点是消耗资源十分庞大，“矿工”们往往拥有多台“矿机”，它们一起工作，矿工就是验证交易的人，要验证一笔交易的正确性要花时间去计算某难题，矿机就是带有高性能显卡的计算机。往往验证一笔交易需要大量计算，矿机们计算不仅需要花一定的时间，而且耗电量十分巨大，要对其散热降温处理是一笔十分大金钱输出。这从很多方面来看都是十分不友好的，尤其是对环保方面来说，这样是十分的不环保的。针对这一致命缺点，以太坊提出了Proof of Stake（POS/权益证明）共识算法。权益证明（PoS）依赖于验证者在该网络中的经济利益。在基于Proof of Stake 权益证明的公共区块链中，一组验证者轮流在下一个区块提出投票并投票，每个验证人投票的权重取决于其存款的大小，或者说是股权。Proof of Stake 通俗的理解就是，验证者们本身的财产就是其资本，越多财产其说话的分量就越足，其投票的权重就越大。 Proof of Stake 权益证明的显着优势，其中包括安全性，集中化风险降低和提高能源效率 。

## 智能合约

第四个技术是智能合约，智能合约是在这些保护数据的上面，构建可自动化按特定规矩执行的规则和条款。拿遗嘱来比喻，智能合约像一个脚本，只需通过运行该脚本，立遗嘱的人死亡后立刻执行合约的规则，财产无需第三方信托机构，直接划入到计划接受的人手中。一个智能合约是一套以数字形式定义的承诺（promises），包括合约参与方可以在上面执行这些承诺的协议。[[[4]](#endnote-4)]

图4是智能合约在区块链上：

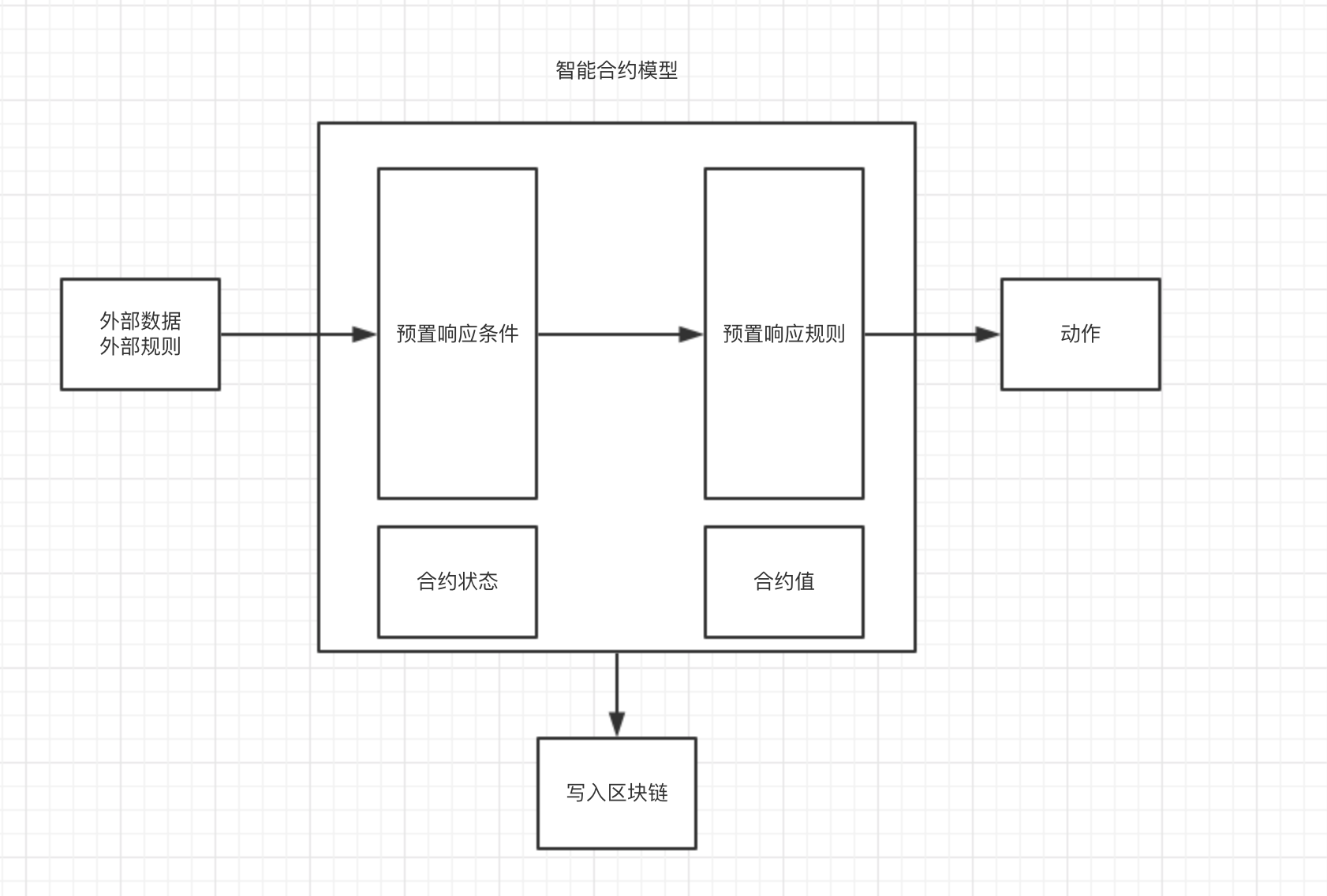


图4 图名

智能合约是“执行合约条款的计算机交易协议”，网络上区块链上的所有的用户都能看得到基于区块链的智能合约。但是这也会导致所有漏洞对所有人都可见包括安全漏洞等，而且可能无法及时地修复。智能合约一旦部署到网络里去，就没办法再修改，智能重新部署新的智能合约。所以一旦智能合约中出现程序错误、BUG等问题，在部署前没找出来，部署后将会引起极大的危险，可能会导致一些操作异常，没能达到想要的结果，另外，一些别有用心的攻击者可以利用该缺陷去对你的财产造成威胁。以太坊智能合约中的问题包括合约编程Solidity、编译器错误、以太坊虚拟机错误、对区块链网络的攻击、程序错误的不变性以及其他尚无文档记录的攻击。[[[5]](#endnote-5)]

## 默克尔树

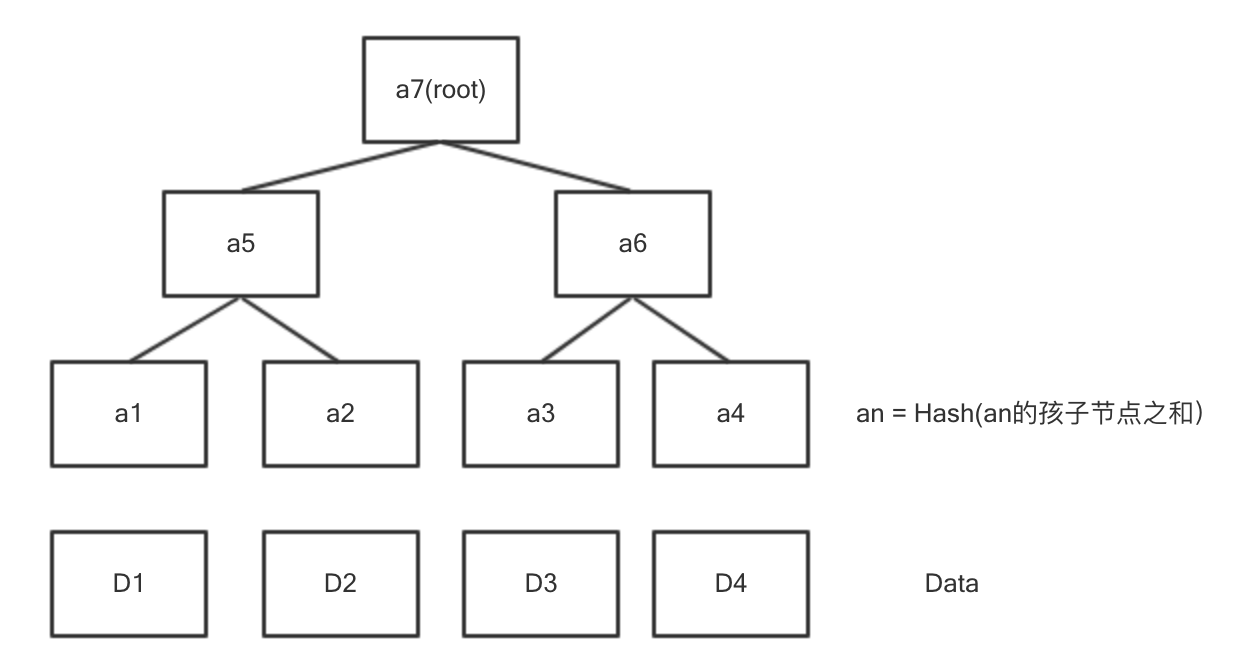
默克尔树（Merkle Tree）是一类二叉树，由一组叶节点、一组中间节点和一个根节点构成。如图5所示：

图5 图名

观察上图中，从底部开始往上看，Dn为叶子节点所携带的数据，即是叶子结点的值。往上看，a1、a2、a3和a4是叶子结点，它们是把数据（即D1、D2、D3、D4）进行哈希（hash）运算后得到的哈希值作为自己的值；再往上看，节点的值就是其孩子结点之和的哈希值，如a5 的值就是其孩子节点a1和a2的值的和经过哈希函数处理后的结果。一直往上，直至根节点的值。从这里可以看出，上一层的每个节点都与下一层的节点的值息息相关，只要稍微对其中某个节点作细微修改，小到甚至一个位的修改，最终会导致整棵树的变化。即任何叶节点内容被修改，叶节点的哈希值最终会改变根节点的散列值。因此，所确定的根节点的哈希值可以准确地用作一组事务的唯一摘要。默克尔树有很多经典的应用，例如Git、Dynamo、比特币等都应用了这个数据结构。这跟上面提到的区块里面的内容会改变区块本身的摘要是十分接近的，因为其就是默克尔树的一个应用。一个是默克尔树是从下往上逐层计算的，所以只要知道相邻的另一个节点的哈希值就可以一直往上计算直到根节点，另一个是根节点的哈希值可以准确的作为一组交易的唯一摘要。在电子虚拟货币上，如比特币，就可以依据这两点就可以来验证一笔交易是否存在，在以太坊基础上的分布式应用上，可以依据这两点就可以来验证一次就智能合约的一次操作是否存在。

## 以太坊

以太坊是一个平台，该平台是开源的，有智能合约功能的公共区块链的一个平台。以太坊有自己的专用的加密货币——以太币（Ether），结合该货币和其提供去中心化服务的以太虚拟机（Ethereum Virtual Machine）来处理点对点智能合约。以太坊是一个提供服务的平台，其提供多种模块让用户来搭建自己的应用，如果把搭建应用比喻成盖屋子，那么以太坊就是屋子材料的提供商，它提供较为完整的模块，在这个比喻中，它提供墙面、屋顶、地板等模块，搭建应用的用户只需要像叠积木一样将屋子用这些建材拼接起来，所以在以太坊上搭建应用是十分方便的，用户一般在以太坊上搭建的应用为分布式应用（Decentralized application）。详细地说，以太坊通过一套图灵完备的脚本语言（Ethereum Virtual Machinecode，简称EVM语言）来建立应用，这个语言和汇编语言相似，但是直接用汇编语言编程很不方便的，所以以太坊里的编程不需要直接地使用到EVM语言，而是使用类似C语言、Python等高级语言，然后其再通过其内置编译器转换到成EVM语言。

## 本章小节

本章节主要讲述了医疗数据共享和隐私保护模型设计所涉及的相关理论知识和技术理论，大致说明了以太坊提供的服务背后所利用到的技术，以及区块链的工作原理和主要关键技术。（上面技术技术介绍与叙述最好能在其中加入以下诸如。。。。本系统采用该技术。。。或本系统在该技术上做了。。改进应用。。。等等，总之，让审阅老师觉得这些内容有一定的相关性而不是毫无关系的堆写。）

# 基于区块链分布式存储系统在医疗系统上的应用方案

## （内容都是平淡的文字叙述，给人感觉这些内容似乎是你的又不是你的，似乎在叙述别人的理论，要把你自己做的工作或提出的东西明确一些地或自信地表述出来，表述清楚）

## 研究目的和研究内容

区块链技术可以保护患者的隐私，防止患者的病历被泄露，把患者的隐私信息记录在区块链上，结合利用加密技术保护患者的隐私信息。这也可以帮助医生更好地观察患者的病情，区块链可以实现资产全生命周期的完整记录，当资产流经整个供应链时，无论是患者的健康记录、患病史，又或是一瓶药，所有记录都清晰可见。

本论文主要针对基于区块链分布式存储系统在医疗系统上的应用模型进行研究，主要对现有医疗系统的整体和诊治流程的各个子系统进行必要的了解，对医疗系统中诊治流程的子系统进行重新设计，系统应该包含患者授权病历予指定医院，病历查看，医生编辑病历，病历数据，病历的存储等基本功能。研究如何去实现一个可以保护患者隐私信息的前提下能共享医疗数据的基于区块链分布式存储共享系统。

## 拟解决的关键科学问题

实现去中心化存储，并使敏感医疗数据独立于各个医院自己本身的系统，不依赖于某个特定的医疗机构，使其在能够确保数据能够在准确的前提下，安全地保护敏感数据的校验、传输、保持和同步的过程。在这个前提下设计访问控制权，涉及大量的权限检查和数据校验的过程。

## 电子病历和电子医疗数据共享框架研究

电子病历就是和现阶段普遍流行的纸质病历，或者是医院系统本身的电子病历无异，记录的内容都是病人的个人信息，患病记录，诊治方案等信息。但是此系统的电子病历是基于智能合约来交互的，也就是对电子病历的任何操作都被指定智能合约所限定，而且最后存储在区块链上。这样确保了共享前提下的不可篡改性和对隐私保护的安全性。

最初首先由患者就自身个人信息创建一张空病历，相当于把自己的个人信息写到了区块链上，但是还没有记载任何的病历。个人信息应包括基础个人信息，如姓名、性别、年龄、联系方式、医保卡等，个人信息创建完毕后，系统返回一个病历账号，该账号相当于网络上该病历的唯一ID，通过该ID来检索到该病历。病历的账号地址应该存在用户客户端本地，此处利用到个人钱包实现，个人钱包本地存储用户的以太网网络地址和私有地址和该病历账号地址。本地个人钱包实现敏感数据不在数据库上存储可确保敏感数据不易因数据库遭到攻击而被泄露。然后患者要就医的时候，到系统提供的用户接口处预约医院。患者指定医院之后，通过智能合约的方法授权其病历于指定医院，指定医院可以查看、编辑患者的信息。医院完成编辑操作之后，保存后的电子病历会被上传到区块链存储。患者结束就诊后，医院结束被授权状态，无法再对该患者的病历操作。

区块链对基于智能合约的所有操作都有详细的记录，从操作者到病历拥有者，操作时间到操作内容，所有详细信息都会记录到区块链上，被授权后的医院的所有操作，都会被区块链所记录，未被授权的医院无法对病历进行任何操作。这样确保了共享医疗数据的实现，而且也确保了共享数据过程的隐私保护得到实现。

## 在电子医疗数据共享模型上的分布式存储系统

区块链其实其本质就是一个分布式数据库，数据存放于所处于的网络中上区块链的区块上，整个数据库的副本则完整地存储在网络上的所有节点中。节点间通过一定的共识对该分布式数据库进行维护，通过一定的方式维持所有副本一致性。该项目所利用的是以太坊所提供的区块链服务，利用其服务在以太坊上层搭建分布式应用。该系统所需求的是一个具有极高安全性和隐私性保护的软件服务系统，以太坊所提供的服务能满足该特点。该系统需要利用以太坊平台所提供的服务结构完成共享模型，实现患者和医院对病历的操作，其中利用的服务包括智能合约发布、交易发布和检索、就账户的操作等。结合以太坊所提供的服务接口完成相应功能后，再搭建一个数据库进行非重要信息的检索以提高用户端操作功能时的效率，最后结合用户端搭建的电子钱包，实现公开密钥、私有密钥和合约地址等信息的保存。

## 电子医疗数据共享模型安全机制

## 小节概览

基于区块链的医疗数据共享与隐私保护模型与普通的医疗数据共享系统最大的区别是敏感医疗数据不再存储于某个第三方服务提供者或者是某家医院的数据库，敏感医疗数据存储在以太坊所提供的区块链上，这样首先确保了病历不可篡改性，还有严格的授权操作和区块链所具备的非对称加密算法进一步确保了安全性。另外，从用户角度出发，用户的公开密钥和私有密钥、还有所部署的智能合约地址，均加密存储于本地，相当于替用户记住了账号密码。

本地对私有密钥的加密采用AES加密算法。采用AES加密的原因在于其是可逆加密且能将一寸十分长的私有密钥转换成一个用户自定的助记字符串，用户只需要记住他自己设定密码，然后就可以对加密后的私有密钥解密。AES全称为高级加密标准Advanced Encryption Standard，是美国联邦政府采用的一种区块加密标准。高级加密标准（AES）的设计思想有三则，分别是抵抗所有已知道的攻击、可运行于多个平台且其速度快、编码紧凑，还有设计简单。

## AES加密

## AES数据结构

AES加密算法的输入位是一个128位分组。这堆分组被描述成4×4的字节方阵，这个分组被复制到state数组中，分组会在加密或者解密的整个过程每个阶段都会被修改。在字节方阵中，方阵中每一位均为一个字，一个字大小为4字节。矩阵中，字是按列排序的。数据结构如图6:

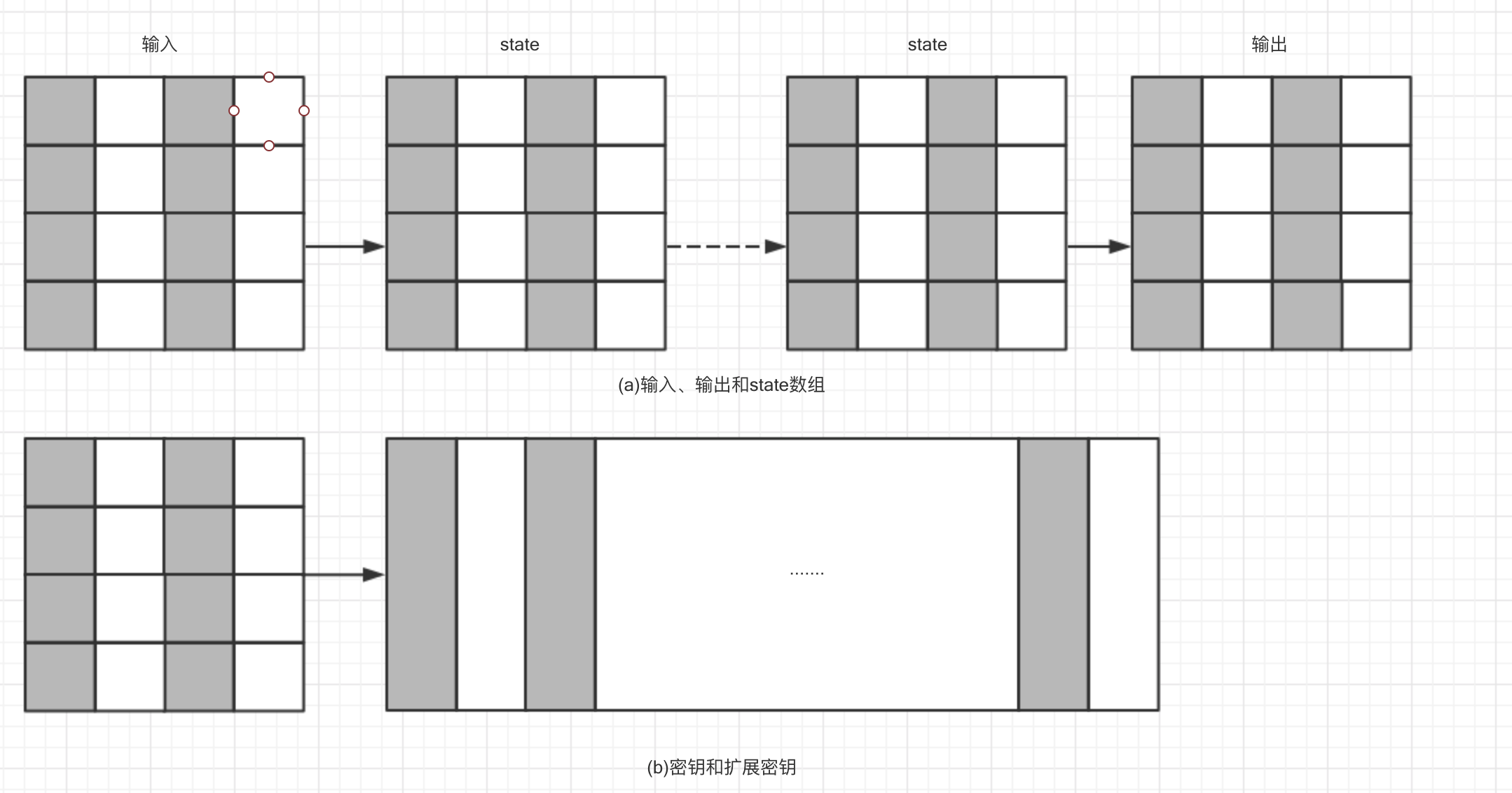


图6。。。。。

## AES加密过程

加密由多轮构成，且其轮数依赖于密钥长度。AES 的基本要求是采用对称分组的密码体制，最少支持128、192、256的密钥长度分组长度为128位。AES 加密数据块的大小最大值是256bit，密钥的长度理论上没有上限。[[[6]](#endnote-6)]设加密共N轮，前N-1轮则有四个不同的变换，分别是字节代替、行移位、列混淆和轮密钥加。以下为四种不同变换的解释：

1、字节代替（Sub-Bytes）：用一个S盒完成分组的字节到字节的代替，是一个简单的查表操作。该操作也是可逆的。

2、行移位（Shift-Rows）：行移位变换完成基于行的循环移位操作，其变换方法分四步，首先第0行不变，然后第1行循环左移1个字节，其次第2行循环左移两个字节，最后第3行循环左移3个字节。显然地，从上述操作可以得知，该操作也是可逆的。

3、列混淆（Mix-Columns）：利用域GF(28)上的算术特性的一个代替，实际上就是利用了矩阵乘法。由矩阵乘法的规则可以证明其可逆性。

4、轮密钥加（Add-RoundKey）：当前分组和扩展密钥的一部分进行按位异或XOR。

最后一轮仅包含三个变换。需注意的是，第0轮是在第一轮前，其包含一个起始的单变换（轮密钥加）。过程如图7:

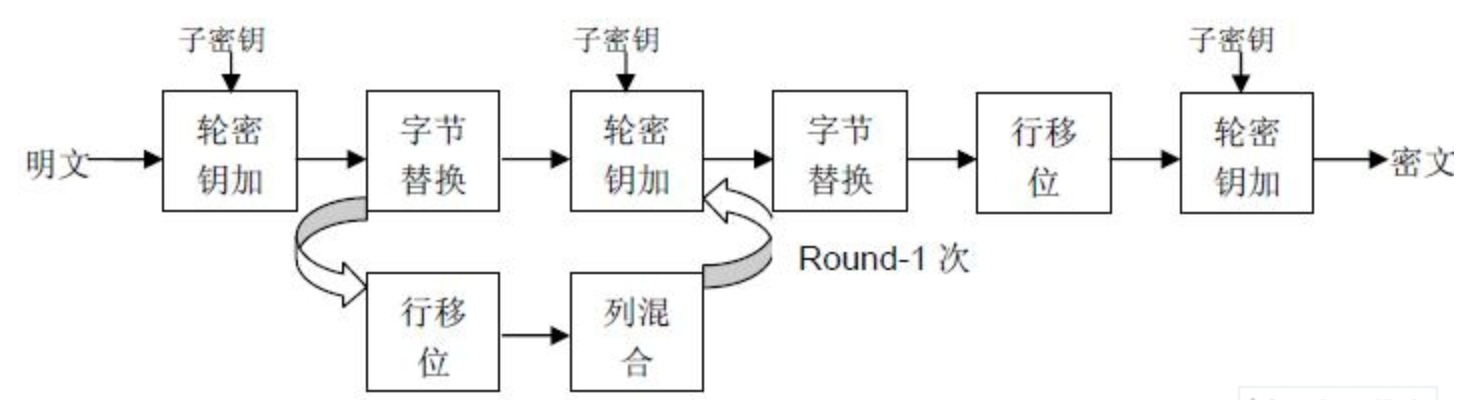


图7。。。。

首位轮密钥加的作用在于其可以增加算法的安全性，因为首尾需求密钥可以避免对密钥不需求的操作处于首尾而导致不需密钥即可取逆的行为。

轮密钥加容易被破解，但是结合另外三个阶段，使得其被提供混淆、扩散和非线性的功能。这三个阶段都不需要密钥，结合其特点可知其对算法的安全性没有起作用。然而该算法经历一个分组的XOR加密即密钥轮加，然后对该分组混淆扩散，再进行XOR加密，如此交替进行，这种方式非常有效且安全性十分高。

采用该加密算法最大的一个原因是因为其可逆性，该算法中每一个阶段均可逆。字节代替、行移位和列混淆都有它们相对应的逆函数。轮密钥加的逆就是用同样的轮密钥和分组相异或，其原理就是A⊕B⊕B = A。AES的解密算法是按逆序来利用扩展密钥，这是由AES的特定结构决定的。

## AES解密过程和AES使用示例

AES密文的解密过程与加密过程的顺序相反，加密过程四个阶段都具有可逆性，故解密过程每轮由四个相反顺序的阶段组成。

AES加密（待加密密钥，密码）= 加密后的密钥

AES加密（“好好学习，天天向上”，“19届汕头大学”）= “FlWANcI4cywTYWfPK3JAPa8rJQByne9LnD3a0TawJc8=”

AES解密（加密后的密钥，密码）= 待加密密钥

AES解密（“FlWANcI4cywTYWfPK3JAPa8rJQByne9LnD3a0TawJc8=”，“19届汕头大学”）= “好好学习，天天向上”。。。。。。。。

## 小节总结

用户从授权病历到管理密钥的一整个流程都得到了相应的保护。从授权病历开始，用户创建空白病历信息，把个人信息写到了区块链上，以太坊所提供的服务能确保查看这是用户本人且不可被篡改；再到授权病历于某个医院，医院成为授权者这一操作是经过智能合约执行的，医院查看和修改病历都会产生记录且记录到区块链上的。就智能合约中约定的行为才能对区块链交互，而智能合约要求用户提供其公开密钥、私有密钥。若这些密钥存在数据库中是十分不明智的，若数据库泄露，那么所有用户的信息也一起集中泄露。本地化加密存储是最好的解决方案，先把密钥用AES加密之后存到用户端本地去，若不法分子想获取用户的资料，首先要攻破用户本地系统获取到被加密的密钥，再解密后进行利用。但是这样是不科学的，因为攻击者若是希望能获取多个患者的信息，那么要逐个账户击破，因为密钥不是存在数据库上的而是用户本地；如果攻击者是希望获取某个患者的信息，他可能有患者某个信息，但是他没办法就这个信息去获取患者的公开密钥，从而去攻破其本地系统来获取私有密钥，因为所有公开密钥是人人可见的，但是公开密钥只有在获取私有密钥或者被授权的情况下才能知道属于谁，所以这是不可能实现的。

经过网络上和本地的安全确保，这样的系统的安全性十分强悍的。

## 本章总结

本章主要介绍了该医疗系统的安全性形式和如何实现提供有很高安全性的系统。

# 医疗数据共享与隐私保护电子病历系统的验证（验证设计太简单，似乎与方案是两张皮）

## 标题与内容不一致或换个更贴切的

系统分为用户端和医院接入端。用户端是提供给需就诊患者的，患者可在注册登陆之后在区块链上创建个人信息，然后预约某家医院，其个人信息也授权于该指定医院，若已有病历的病人，病历也一同授权于指定医院。医院方接收到通知后就可以进行对患者的信息进行查看、和新增病历等操作。整体框架设计如图8：

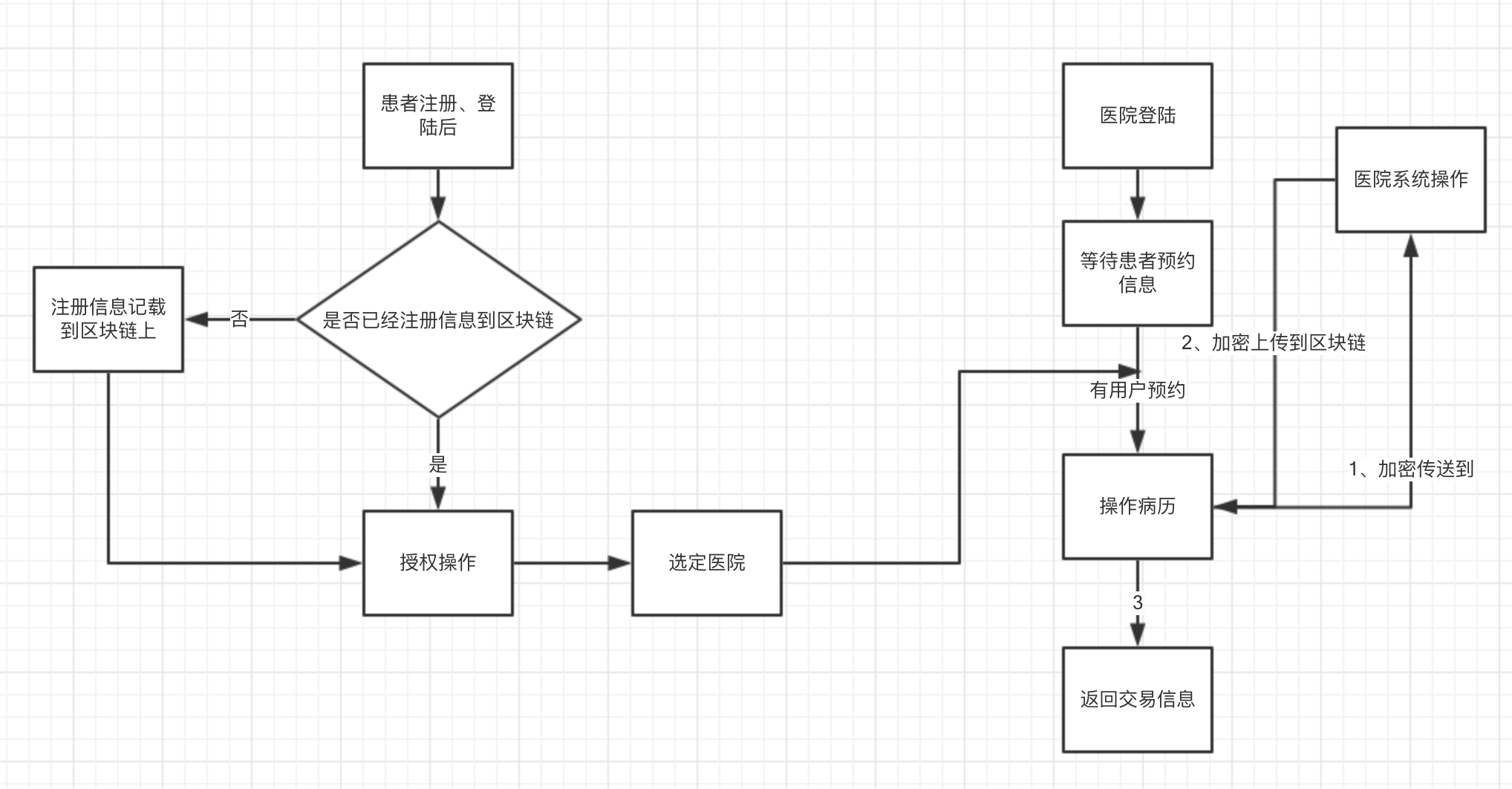


图8。。。。。

患者注册、登陆：患者登陆到系统，通过登陆去获取本地加密存储的私有密钥、智能合约地址（若已经注册信息到区块链）。

注册信息记载到区块链上：相当于创建一张个人信息表且附带一张没有记录的病历，记录到区块链上。

授权操作、选定医院：根据系统所接入、所提供的医院进行选择、授权，授权后医院将有权限去对病人的病历进行查看、新增病历等操作。

医院登陆、等待患者预约信息：医院登陆到系统，等待有患者预约自家医院，若有，会接收到通知且能对病历操作。

医院系统操作：医院一般有自己的系统，其系统一般包括写病历、处方等操作，故应该提供接口接入医院系统。

操作病历：系统将信息加密传输到其系统，医生对病人病情做诊断后，在其医院系统上撰写病历后，加密上传到本系统，然后由本系统上传到区块链，最后产生交易信息，按需求返回给用户端、医院。

## 存储设计

验证程序所用到的数据库表具体描述如表1至表2：

表1 账号信息表（patientInfo）

|  |  |  |
| --- | --- | --- |
| 列名（字段） | 解释 | 数据类型 |
| patient\_id | 患者ID | String |
| patient\_username | 患者登陆名 | String |
| patient \_name | 患者名字 | String |
| patient \_gender | 患者性别 | String |

表2 病历表（medicalRecord）

|  |  |  |
| --- | --- | --- |
| 列名（字段） | 解释 | 数据类型 |
| record\_id | 病历ID | String |
| patient\_id | 患者ID | String |
| patient\_name | 患者名字 | String |
| patient\_gender | 患者性别 | String |
| patient\_record | 病历内容 | String |

## 开发环境/运行环境

开发环境：MacOS

运行环境：IOS、MacOS

## 开发技术

本系统分前端和后端。前端实现IOS客户端，利用Swift语言作为主要开发语言；后端实现包括数据库、服务器和智能合约开发，Javascript、solidity 是主要开发语言。后端主要是基于Nodejs开发。

## Node.js和Express

Node.JS是一个让 JavaScript 运行在服务端的开发平台，它让 JavaScript 成为与PHP、Python、Perl、Ruby 等服务端语言平起平坐的脚本语言。[[[7]](#endnote-7)] Node对一些特殊用例进行优化，提供替代的API，使得V8在非浏览器环境下运行得更好。V8引擎执行Javascript的速度非常快，性能非常好。 Node是一个基于Chrome JavaScript运行时建立的平台， 用于方便地搭建响应速度快、易于扩展的网络应用。Node 使用事件驱动， 非阻塞I/O 模型而得以轻量和高效，非常适合在分布式设备上运行数据密集型的实时应用。

Express是Node.Js的一个框架，利用Express能进行Nojde.js 快速开发，快速搭建一个可运行的HTTP服务器。

## Solidity

Solidity是一种合约导向式语言，可被应用于各种不同的区块链平台上，可使程序开发人员能在区块链上（例如以太坊）编写智能合约。[[[8]](#endnote-8)] Solidity是以太坊所推荐使用的智能合约开发语言。利用Solidity，实现病历的共享、隐私保护、信息保存到区块链上等功能。

实现授权的主要代码段如下代码段1

modifier check() {

// 主要利用把授权医院的公开地址获取，通过该授权modifier，附加到函数中去即可。

require(

msg.sender == myaddress || msg.sender == checker,

"Only ath-person can do this ops."

);

\_;

}

modifier check\_doc() {

// 防止非医生对病历操作

require(

msg.sender == checker,

"Only doc can do this ops."

);

\_;

}

代码段1

实现对病历修改代码一览，如代码段2:

struct Patient {

uint weight;

bool checked;

address p;

uint age;

string gender;

string name;

string record;

}

function setRec(string memory \_rec) public check{

modify(Patient);

emit makeChange(msg.sender);// who changed it.

}

代码段2

Solidity就是用户和医生跟区块链交流的一个桥梁，以上代码段为整体Solidity代码段的冰山一角，也是重要的一个部分。

## AES加密算法实现

AES在上一章介绍过，主要用于本地加密保存私有密钥。主要实现是用JavaScript实现。由于网上关于AES基于JavaScript实现的库相当多，最常用的是crypto-js或者aes-js。本系统的AES加密算法严格按照AES定义去实现，以下是调用编写好的AES接口的代码段和输出结果。如图代码段3:

import aes from ‘./aes.js’;

let password = “0x62b00Bd5dE47b22eA98922f4020fa0BaB85ad32B”;

console.log(aes.run(password, “xiaoli”);

//输出 QIRAgwg5BTVIfUAORPs8GmziY37vxsUlT/iKLa7q/sDFol1FKRQr3vFJzbxF2WyL

代码段3

## API设计、界面设计与演示

本系统分为看诊与就诊两个流程，且分别由医生与患者操作。患者在客户端上操作，医生应该通过接口接入到其医院的系统，或者也用客户端。

就诊流程就是预约和共享医疗数据的流程，需求患者先登陆到系统，若尚未注册，则需先注册，部分个人信息的数据会被写到区块链上去，并且是受保护数据，即只有授权者或者本人才能查看。同时，看诊医生的医院也要登陆到系统，以接收预约信息和接收敏感信息和新增病历。如图——登陆成功后会返回登陆的用户的名字和用户属性（UserType），用户属性是规定用户是医院还是患者。

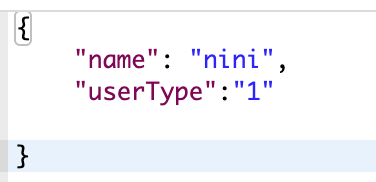


图9

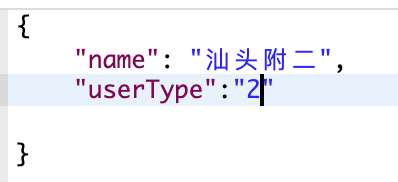


图10。。。。。

下图11为实际操作界面，分别为登陆注册页面和登陆成功页面

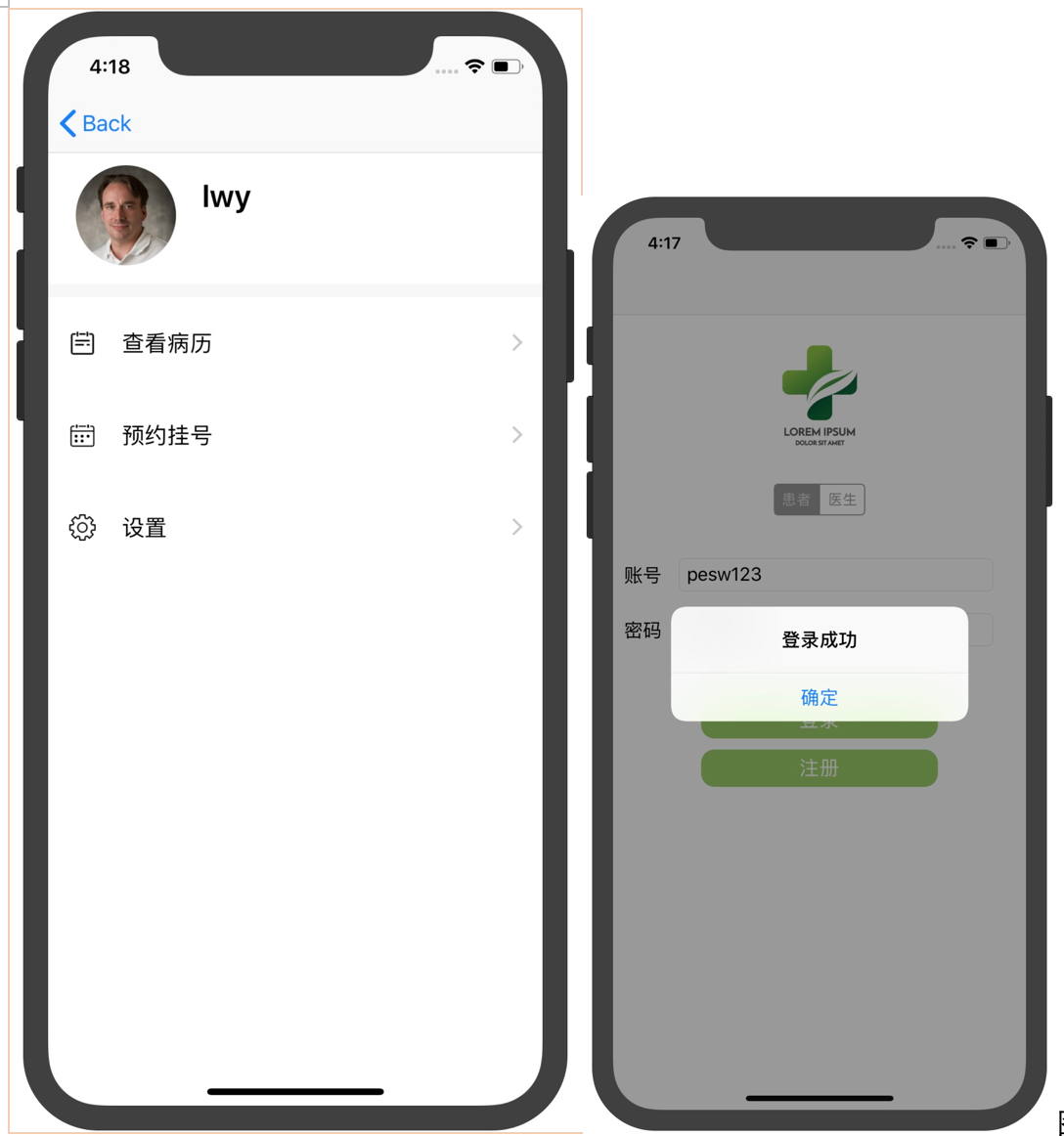


图11

预约所需用到的参数是医院的名字，然后就医院的名字找到其公开地址。然后在发送该请求后，若请求成功，即返回交易信息，因为这是一次通过智能合约去授权的操作，所以会产生交易信息，若交易失败，会返回相关原因，如“没被授权”等错误信息。由交易信息中可以看到操作的是谁（from），就哪张智能合约进行了操作（to），还有该交易的唯一标识（transactionHash），以及其他一些信息如本次交易所花费以太（因为服务建立在以太坊上），交易信息如图12，预约操作完成后如图13:

图12



图13

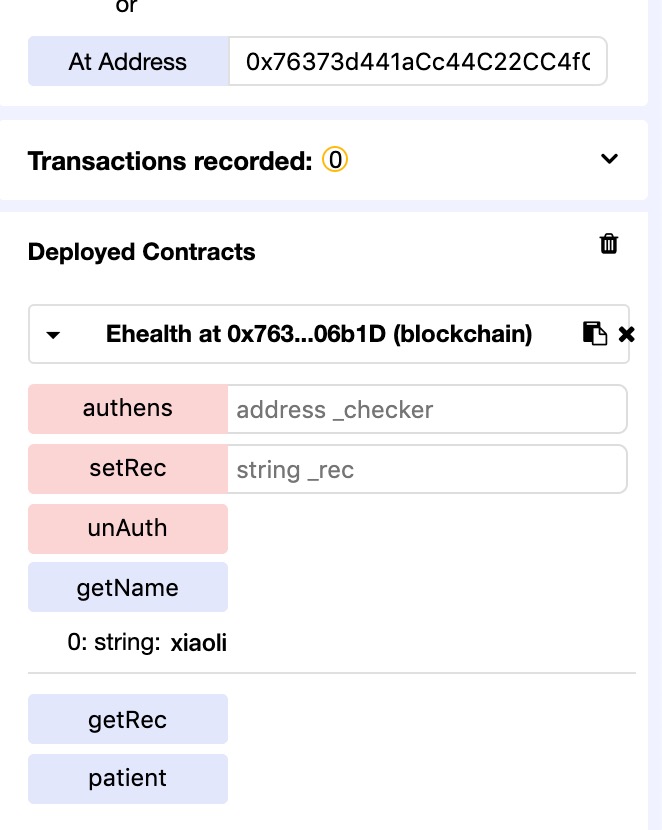
医生就诊完毕后应填写相应病历信息、诊治方案等，其中还应包括医生本人签名、医院科室和日期。新增病历成功后返回一个交易信息。

医生诊治完，提交新增病历后，用户的新一条病历将被写到区块链上，当患者在本医院结束诊治后，患者对医院的授权自动取消，医院丧失对该患者的医疗数据查看、新增等权力。

## 系统正确性分析

本系统主要设计主要利用以太坊服务提供的区块链去开发，在智能合约能正确编译、部署的情况下，并且调用智能合约所生成的所有方法且调用成功后返回交易信息，即可证明成功。

进一步测试和分析正确性。借助Solidity所提供的Remix编译器，将编译好的合约地址部署到Remix上去，然后再尝试调用其中函数，若调用成功，即测试正确性成功。如图，将在系统中产出的合约地址输入到At Address中，在下面出现合约内容，且能调用其函数。

****

经测试，系统达到预期所有功能，结果是正确的。

## 本章小结

本章主要讲述了系统整体框架的设计以及开发时所用到的一些相关技术，并且对系统整体正确性做了分析，以及系统每一个界面、接口的介绍。

# 问题与解决方案

## 遇到的技术问题和解决方法

刚开始接触区块链的时候，一大堆的基础知识要学习让我应接不暇。经过大量的阅读文献、博客文章，我才慢慢地开始入门区块链技术。对于区块链开发的平台选择，有Python和JavaScript以及其他语言选择，我选择了JavaScript和其Web3库，因为对JavaScript语言的熟悉，使得我对区块链开发的入门相对要轻松了些。然后Web3和Solidity的官方文档也是比较难读懂，因为设计很多以太坊的相关知识，所以在阅读官方文档的时候不得不一边了解接口的用途和用法，另一边又要理解每个参数的作用以及其含义。区块链中文社区还不算十分成熟，在用搜索引擎进行中文检索时，很多问题没办法找到解决方案，只能通过英文检索，找国外的资源和问题答复。

还有客户端和后端对接的时候，遇到的问题十分的多，因为项目还是部署在本地，客户端在连接后端的时候出现了跨域的问题。后来经过网上搜索和学习，才找到有相关库可以解决该问题。

在编写加密算法的时候也有很多很复杂的逻辑用JavaScript很难去实现，幸运的是JavaScript的中文社区已经很强大，问题能很快地被解决。

## 技巧总结

后端开发结合Postman去对所开发出来的API测试比用浏览器测试要快很多。用Robo T对数据库检索操作比用命令行要快。

## 本章总结

本章主要总结了在设计和开发过程中所出现的问题和解决方案，并且对过程中所总结出的一些开发技巧也作了总结。

# 总结与展望

## 总结

本文主要介绍了基于区块链等医疗数据共享与隐私保护模型的设计与实现，提出了一个可接入协同诊治平台下的一个子系统的电子医疗数据共享方案，且介绍了该方案所涉及的相关知识。电子病历是当下医疗发展的必然趋势，电子病历与普通病历实质上无差别，但是因为电子病历所处的环境致使一套更为安全的病历共享方案需要被提出。区块链就是电子病历很好的载体，不仅保证了其不可篡改性，而且保证了其隐私性。从系统整体出发，区块链能确保大量病历同时遭到泄露的情况不会发生，这是因为区块链特性；从每个患者使用系统出发，本地AES加密存储敏感信息更进一步为私密信息上锁，确保个人信息遭到泄露的情况不会发生。本文最后对系统的正确性进行验证，验证了方案的可行性。

## 展望

本系统的设计方案仅局限于一定数量医院的区域网络中，随着推广和发展，这样的区域网络范围会显得愈发局限，因为好的专科医院散步全国各地，患者不可能只在指定区域就诊。其次，系统对数据加密算法的是可逆加密算法，因为考虑到便利性，故采用可逆的加密算法，这样仍然还是存在被破解风险。希望在日后能就加密算法上提出一种更适合本系统的新解决方案。

# 致谢

在完成本篇论文的过程里，在整个项目进行过程中经历了许多问题和困难。因这篇论文涉及的知识比较广，需要学习的新知识比较多，从设计到实现的每一步都是十分困难的，比预期要难得多。在此，要感谢我的指导老师朱诗生老师，他从开题到设计，在从设计到实践整一个漫长的过程，给予了我明确的方向，每个阶段应该处理什么问题，每个问题主要的内容是什么，主要解决什么样的问题，并且给出怎样解决问题的一些建议。而且他每周在百忙中都抽出时间对我们进度进行督促并就遇到的问题给予解决方案建议，故在我的毕业设计中，朱老师功不可没。另外，还要感谢黄仁俊师兄在理论知识、技术知识方面给予我很大的支持，并且为我的设计提出了很多宝贵的意见。

在此，还要衷心感谢计算机系所有老师，他们给予了我在计算机科学与技术方面所有基础知识和专业知识的专业的指导，使得我具备了一名合格的计算机科学与技术毕业生应有的基本素养。

最后，感谢汕头大学以及汕头大学计算机系对我的培养。

下面参考文献格式不合格，

1. [1] 曹敏姿，张琳琳，毕雪华，赵楷. 个性化(α ，l) - 多 样 性 k - 匿 名 隐 私 保 护 模 型[J]. 计算机科学，2018，45(11). [↑](#endnote-ref-1)
2. [2] 赵延红, 原宝华, 梁军. 区块链技术在医疗领域中的应用探讨[J]. 中国医学教育技术, 2018,32(01):1-7. [↑](#endnote-ref-2)
3. [3] 袁勇,王飞跃. 区块链技术发展现状与展望[J]. 自动化学报,2016,42(04):481-494. [↑](#endnote-ref-3)
4. [4] Thomas Bocek. Digital Marketplaces Unleashed. Springer-Verlag GmbH. 2017-09-15: 169-184. ISBN 978-3-662-49274-1. [↑](#endnote-ref-4)
5. [5] Thomas Bocek. Digital Marketplaces Unleashed. Springer-Verlag GmbH. 2017-09-15: 169-184. ISBN 978-3-662-49274-1. [↑](#endnote-ref-5)
6. [6] 邹蕾. 基于AES的Flash加密实现[J]. 电脑编程技巧与维护, 2014(22):115-116. [↑](#endnote-ref-6)
7. [7] 黄丹华.Node.js开发实战详解[M].清华大学出版社:北京,2016:2. [↑](#endnote-ref-7)
8. [8] Allison, Ian. PwC blockchain expert pinpoints sources of ambiguity in smart contracts. IBTimes (News). 12 August 2016 [14 December 2016]. [↑](#endnote-ref-8)