区块链与电子病历系统设计文档

目 录

[1. 概述 3](#_Toc520295650)

[1.1. 电子病历 3](#_Toc520295651)

[2. 区块链与医疗系统的矛盾 3](#_Toc520295652)

[3. 基于区块链的电子病历结构设计 4](#_Toc520295653)

[3.1. 可授权电子病历结构设计 4](#_Toc520295654)

[3.2. 区块结构设计 5](#_Toc520295655)

[4. 业务功能 5](#_Toc520295656)

[4.1. 用户病情查看 5](#_Toc520295657)

[4.2. 授权流程 6](#_Toc520295658)

[5. 系统设计 6](#_Toc520295659)

[5.1. 整体架构 6](#_Toc520295660)

[5.2. 签名链 6](#_Toc520295661)

[5.3. 电子病历数据 7](#_Toc520295662)

[5.4. 写入流程 7](#_Toc520295663)

[5.5. 冲突处理 7](#_Toc520295664)

[6. 问题与挑战 8](#_Toc520295665)

[6.1. 数据保密 8](#_Toc520295666)

[6.2. DDOS攻击 8](#_Toc520295667)

[6.3. 容量问题 8](#_Toc520295668)

# 概述

本文档主要探讨基于区块链的电子病历医疗系统的可行性及必要性。包括区块链的特点、适用场景以及电子病历系统初步设计。面向的读者为有一定计算机、数据库基础知识的技术人员。

## 电子病历

电子病历是由医生书写的病人的医疗记录，由多个科室和医生共同维护生成，电子病历所属权归病人所有。电子病历主要有以下几种使用场景：

1. 为后续的诊疗提供依据。
2. 为医学研究及保险用途提供数据。

为此提出了以下几个需求：

1. 不允许修改，如果更新需要同时保留新旧版本。
2. 可以授权给其他用户使用，由于电子数据的可复制性，此授权为赠与数据。

而区块链的难篡改、数据分布式存储等性质可以满足这些需求。

# 区块链与医疗系统的矛盾

以比特币为代表的区块链1.0，通过挖矿机制为系统提供了安全保障，但同时限制了系统吞吐率，比特币系统运行8年的数据总量是200G左右，很难满足其他应用的需求。医疗电子病历数据量大，总数据量可达一百多T，首先需要解决数据存储的问题。

区块链的开放性和隐私保护是一个矛盾的存在。开放性意味着所有人可以

获得数据，如何在开放性的前提下保护隐私也是一个需要考虑的问题。

加密和授权也是一个矛盾的存在，加密后的密文只能通过私钥解密，而私钥只能由本人保管。授权则是允许别人查看明文，如何协调加密和授权的关系也是一个难点。

# 本文工作

本文基于区块链的思想，结合医疗系统中碰到的问题，设计了一个基于区块链的医疗系统模型，

# 基于区块链的电子病历结构设计

## 可授权电子病历结构设计



本文提出的用于电子病历的结构如上图所示，该电子病历的结构可以同时用于诊疗数据的产生和病人对数据的授权。各个字段的解释如下：

|  |  |
| --- | --- |
| 字段名称 | 说明 |
| 写入者ID | 写入电子病历的用户ID，当时医生诊断病人时，写入者为医生；当病人授权医生（医院）时，写入者为病人 |
| 拥有者ID | 拥有数据的用户ID，当医生诊断病人是，拥有者为病人；当病人授权时，拥有者为医生（医院）。 |
| 安全等级 | 指示接下来的数据内容安全性，分为明文数据、脱敏数据和加密数据类型 |
| 数据类型 | 指示接下来的数据类型，如门诊数据，用药数据、体检数据、住院数据等 |
| 诊断内容 | 电子病历核心数据 |
| 签名 | 对[写入者、拥有者、安全等级、数据类型、诊断内容]的签名 |

## 区块结构设计



每个字段的解释如下：

|  |  |
| --- | --- |
| 字段名称 | 说明 |
| 区块地址 | 当前区块的哈希值（两次SHA256计算） |
| 上区块地址 | 上一个区块的地址 |
| Merkle根 | 用来教研所有的电子病历 |
| Nounce | PoW中的控制值 |
| 电子病历ID | 用来指向电子病历的列表 |

其中，Merkle根的计算过程可由下图简单计算：



# 系统设计

## 整体架构

系统整体包括一个区块链和电子病历数据。其中区块链为所有节点所共有，电子病历数据可以根据特定的应用场合决定是否在所有节点存储。区块链的结构大致如下，每个区块中含有电子病历的ID列表，每一个电子病历ID指向一个电子病历。



## 签名链

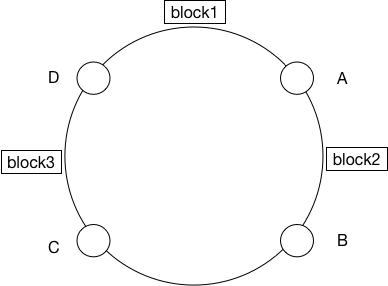
签名链上的每个区块只记录电子病历的hash值，称为签名区块，由于不

涉及原始数据，数据量较少，可以每个节点保存一份。其同步方式与比特币的方式相同。

## 电子病历数据

不同节点负责存储不同的电子病历数据。采用一致性hash的分区方式，每个区块的全部数据作为一个数据分区。系统初始化设置复制因子n，应满足预计数据总量\*n小于所有节点的磁盘容量之和，n越大数据越安全。n的选择需要在磁盘容量和安全性之间选择。

每个数据分区通过区块ID计算hash值，hash值比其大的最近的n个节点负责存储此分区数据。数据分区信息为系统元数据，节点间可定期同步元信息。



以上图为例，A、B、C、D为四个节点，系统采用n=2副本，block1的hash处于A与D之间，block1的电子病历数据存放在A、B节点上。同理，block2的数据存放在B、C节点上，block3的数据存放在D、A节点上。

为了进一步实现负载均衡，可以采用虚拟节点的方式，将每个实际节点对应256个虚拟节点，虚拟节点同样均匀分布在环上。

## 写入流程

1. 医生通过客户端生成电子病历数据，由病人签名后发送给其他节点，此签名为病人现场输入公钥进行签名。
2. 节点接收到签名的电子病历后，进行检查，将合格的电子病历缓存到本地。
3. 对于符合要求的电子病历，打包写入签名链后，广播签名区块。
4. 节点收到签名区块后，检查其正确性，并判断签名区块的数据是否需要存储在本地，如果需要，则持久化缓存的电子病历，否则丢弃相关电子病历。

## 冲突处理

只有医院有写入电子病历的需求，因此不需要挖矿机制来限制写入速率。块大小限制在1M，当电子病历的hash达到一个块，就将电子病历写入区块链，每个节点以最长链为主链。

# 业务功能

## 医疗记录写入

在一次医疗诊断结束之后，医生需要对电子病历执行一个写入操作。在真实场景中，因为数据属于病人，所以数据的保护方式（明文，脱敏，密文）在写入时需要医生和病人共同产生。假设医生D需要给病人P产生一个医疗记录，则写入数据需要经历以下几个步骤：

1. 医生D产生真实数据的病历内容，交由P进行保护处理，病人可以选择是明文存储，脱敏处理还是加密处理。如果选择加密处理则病人需要将信息用自己的公钥进行加密。
2. 医生D将信息按照电子病历格式填入电子病历（写入者ID为医生ID，拥有者ID为病人ID），并交由病人P
3. 病人P对当前病历内容进行两次SHA256哈希计算，并对哈希值用自己私钥进行签名，签名之后即可将电子病历发送至节点处理。

上述的过程可以保证数据为病人所拥有，由于诊断结果的产生是医生，所以医生在数据写入之前是对数据有访问权限的。如果存在恶意的医生，在信息加密之前将数据备份一次并作其他处理，将会导致信息的泄露。由于数据的最初产生在医生，因此没有系统可以避免这种情况，即在医生是恶意的情况下无法避免医生知道病历的内容。

## 病情查看

对于病人来说，可以通过本系统查看到自己所有的医疗记录，检查自己的医疗记录过程如下：

1. 获取当前块，并遍历当前块中所有的医疗记录，如果医疗记录的拥有者ID为病人公钥并且该医疗记录不是授权单，则该电子病历为病人的看病记录。
2. 根据当前块找到上一块，重复执行（1）中操作。
3. 重复执行（2）直至遍历完所有区块。

我们注意到，医生找到自己的医疗记录是需要遍历所有的区块的。在区块数目较多的情况下，可能是一个比较耗时的工作。因此为了方便找到病人的医疗记录详情，也可以考虑在电子病历中添加一个字段为上一病历ID，在产生医疗记录是就记录上一次看病的医疗记录ID，这样就不需要遍历所有区块即可以根据指针找到某以病人的所有看病记录。



## 授权流程

在医疗系统中，电子病历的核心数据理应为病人所拥有，若未经病人允许，医生和医院不应当有权限读取数据内容。在我们的系统中，用户可以允许数据以三种方式保存：

1. 明文存储:

在明文存储模式下，电子病历中的核心数据以明文的方式存储。

2. 密文存储

在密文存储模式下，电子病历中的核心数据以密文的方式存储。

3. 脱敏处理

在脱敏存储模式下，和病人标志有关的信息被加密存储，其他一般医疗诊断信息采用明文的方式存储。

病人是数据的拥有者，只有病人才有权利将自己的数据授权给其他用户读取，而被授权的人也只能拥有对数据的读取能力，无法再将数据授权给其他用户。因此在授权流程中，需要检查用户是否具有授权能力。

医院或者医生如果想看到病人的数据，则需要病人先授权。在密文存储模式下，假设病人A想要授权电子病历X给医生B，则授权的过程如下：

1. A获得电子病历X，并用自己的私钥解密得到真实电子病历。
2. 检查A是否有授权能力（如果没有源病历ID则有授权能力），如果没有授权能力则授权失败。
3. A获得B的公钥。
4. A重新产生一个电子病历，写入者为A，拥有者为B，源病历ID为X的ID，同时将真实数据用B的公钥加密，并写入新的电子病历。
5. 将电子病历发送到处理节点。
6. 节点接收到电子病历后，检查源病历是否是初始病历，如果不是则抛弃。

# 问题与挑战

## 数据保密

数据以密文存储，黑客即使拿到数据也无法解析。

## DDOS攻击

暂无DDOS攻击的有效解决方式。

## 容量问题

电子病历区块链实际维护者为医院，因此节点数量与参与的医院有关，参与的医院越多，系统越安全。