(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 112131304 B
(45)授权公告日 2021.05.04

(21)申请号 202010085010.5

(22)申请日 2020.02.10

(65)同一申请的已公布的文献号
申请公布号 CN 112131304 A

(43)申请公布日 2020.12.25

(73)专利权人 北京天德科技有限公司
地址 102488 北京市房山区阎富路69号院37号楼-1层至4层102三层10

(72)发明人 蔡维德

(51)Int.Cl.
G06F 16/27(2019.01)
G06F 21/60(2013.01)
G06F 21/62(2013.01)

(56)对比文件
CN 109981584 A,2019.07.05
CN 110674533 A,2020.01.10
CN 107807951 A,2018.03.16
CN 110334154 A,2019.10.15
US 2019377724 A1,2019.12.12
焦通等."区块链数据库:一种可查询且防篡改的数据库".《软件学报》.2019,第30卷(第9期),

审查员 姚晓斌

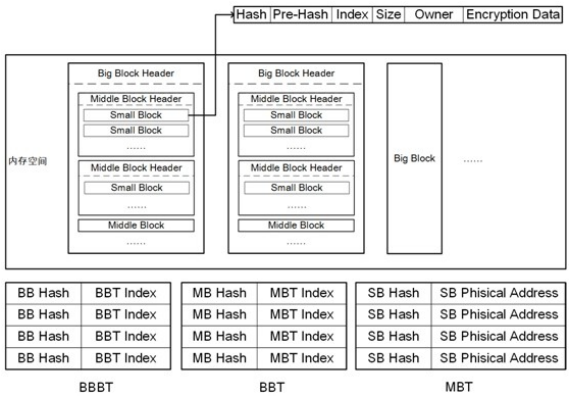
权利要求书2页 说明书4页 附图5页

(54)发明名称

一种基于区块链技术的新型计算及存储架构

(57)摘要

本发明提出一种基于区块链技术的新型计算及存储架构,以适应区块链操作系统的要求。基本数据结构包括BB(Big Block)、MB(Middle Block)以及SB(Small Block),通过BBBT(Big Big Block Table),BBT(Big Block Table),MBT(Middle Block Table)进行基于内容的哈希寻址。本发明既适用于内存管理也适用于分布式存储,支持服务器,手机,云等万物互联,使数据更加安全高效。



CN 112131304 B

1.一种基于区块链技术的新型计算及存储方法,其特征为:将基本数据结构分为三种区块:BB区块、MB区块以及SB区块,其中所述三种区块均由区块头和区块体两部分构成,所述BB区块的区块体部分存放所述MB 区块的数据;所述MB 区块的所述区块体存放所述SB区块的数据,所述SB 区块的所述区块体存放具体存储数据;

对于BB区块,其结构包括:

Hash_BB,Pre-Hash_BB,Index_BB,Size_BB,User List_BB属于所述区块头数据,Encryption Data_BB属于所述区块体的具体数据区域,其中Hash_BB表示当前区块的哈希值;Pre-Hash_BB表示上一个BB区块的哈希值,以此在形成连续的链式结构,内存或存储由多个BB区块组成;Index_BB表示BB区块的索引值,依据此索引值进行具体的寻址;Size_BB表示整个BB区块的大小;User List_BB表示对该BB区块具有访问权限的用户列表,多个用户可以对同一个BB具有访问权限;Encryption Data_BB表示BB区块的数据区域,所述数据区域由多个MB区块构成,并用非对称加密算法进行加密;

对于MB区块,其结构包括:

Hash_MB,Pre-Hash_MB,Index_MB,Size_MB,User List_MB属于所述区块头数据,Encryption Data_MB属于所述区块体的具体数据区域,其中Hash_MB表示当前MB区块的哈希值;Pre-Hash_MB表示上一个MB区块的哈希值,以此在BB区块中形成连续的链式结构,BB区块由多个MB区块组成;Index_MB 表示MB区块的索引值,依据此索引值进行具体的内存或存储寻址;Size_MB表示整个MB区块的大小;User List_MB表示对该MB区块具有访问权限的用户列表,多个用户可以对同一个MB具有访问权限;Encryption Data_MB表示MB区块的数据区域,所述数据区域由多个SB区块构成,并用非对称加密算法进行加密;

对于SB区块,所述SB区块为内存或存储中最小的数据单元,每一个SB区块表示一个交易或一个具体存储数据, 其结构包括:

Hash_SB,Pre-Hash_SB,Index_SB,Size_SB,Owner属于所述区块头数据,Encryption Data_SB属于所述区块体的具体数据区域,其中Hash_SB表示当前区块的哈希值;Pre-Hash_SB表示上一个SB区块的哈希值,以此在MB区块中形成连续的链式结构,MB区块由多个SB区块组成;Index_SB 表示SB区块的索引值,依据此索引值进行具体的内存或存储寻址;Size_SB表示整个SB区块的大小;Owner表示拥有该SB区块的用户,每一个SB只能有一个拥有者;Encryption Data_SB表示SB区块的数据区域,所述数据区域表示一个交易信息或一个具体存储数据,并用非对称加密算法进行加密;

所有数据的寻址方式基于三张映射表:BBBT映射表,BBT映射表,和MBT映射表;其中所述BBBT映射表包含Hash_BB和对应BBT Index信息,BBT映射表包含Hash_MB和对应MBT Index信息,MBT映射表包含Hash_SB和对应SB区块的物理位置信息;所述BBBT映射表是整个内存的全局寻址表,用来寻找具体的BB区块的位置,根据BBBT中存储的BBT Index字段找到具体的BB区块对应的BBT映射表,在BBT映射表中根据MBT Index字段找到相应的MB区块对应的MBT映射表,在MBT映射表中根据SB区块的Hash_SB值找到对应的SB区块的物理位置;使用BBBT映射表、BBT映射表及MBT映射表基于内容进行哈希寻址;在MBT映射表中,SB的哈希值对应的是相应的机器列表,选择某一机器获得相应SB区块数据,三张映射表采用分布式存储方式,每个机器存储距离自己最近的区块的信息,并提供附近机器的区块信息,从而快速定位到具体SB区块。

2.根据权利要求1所述的一种基于区块链技术的新型计算及存储方法,其特征为:所述数据的结构和所述哈希寻址的方式针对内存和/或分布式存储。

3.根据权利要求1所述的一种基于区块链技术的新型计算及存储方法,其特征为:为硬件和软件资源提供统一资源标识符。

4.根据权利要求1所述的一种基于区块链技术的新型计算及存储的方法,其特征为:所述数据的安全性通过私钥加密,每个区块的区块体部分使用非对称加密进行密码学处理,不同用户拥有不同的公钥和私钥,由密钥管理模块统一存储管理,加解密在CPU和/或GPU上处理,保证加解密速度,密钥管理模块以及加解密模块都加入操作系统内核中,在内核态执行程序更快速,用户使用公钥加密,使用私钥解密,保证操作系统出问题时数据仍然安全。

一种基于区块链技术的新型计算及存储架构

技术领域

[0001] 本发明属于区块链技术领域及计算机操作系统领域,特别涉及到计算机内存,本地存储以及分布式存储的数据结构以及寻址方式,以及将区块链技术应用到操作系统底层的相关技术。

背景技术

[0002] 传统的计算机的内存管理以及寻址方式如图1所示,采用段页式结构,将进程按逻辑模块分段,每个段都有自己的段号,再将段分成若干大小固定的页。内存空间的管理仍然和分页存储管理一样,将内存分成若干个和页面大小相同的存储块,最后将进程的各个页分别装入各个内存块中。传统的内存结构和寻址方式对于区块链技术的支持有一定的局限性,能更加优化地支持区块链运行的操作系统内存结构同样也应该是区块形式的,传统的内存数据结构以及寻址方式已经不适用于新型的区块链操作系统。

[0003] 传统分布式系统的存储结构如图2所示,由客户端、元数据服务器和数据服务器构成。客户端负责发送读写请求,缓存文件元数据和文件数据。元数据服务器负责管理元数据和处理客户端的请求,是整个系统的核心组件。数据服务器负责存放文件数据,保证数据的可用性和完整性。在这样的架构下,一旦元数据服务器遭到攻击或者出现错误,整个数据存储系统将会瘫痪。

发明内容

[0004] 本发明提出一种基于区块链技术的新型计算及存储架构,设计了一套操作系统的内存数据结构以及寻址方式,能够适用于区块链操作系统的要求。本发明将元数据与存储数据放在一起构成新的数据结构,既适用于内存寻址也适用于分布式存储寻址,且基于内容寻址,支持服务器,手机,云等万物互联,使数据更加安全高效。同时,为了保证安全性,本发明在数据中加入了加解密技术手段,系统结构与传统操作系统相比发生了重大变化。

[0005] 本发明提出的基于区块链技术的计算机内存或存储的数据结构为区块式结构,最小的数据为一个交易或一个具体存储数据,这样方便CPU以最快的速度找到具体交易或存储数据,进行数据的读写。

[0006] 本发明提出的基本数据结构如图3所示,由三种区块构成:BB(Big Block)、MB(Middle Block)以及SB(Small Block)。三种区块均由区块头和区块体两部分构成。BB(Big Block)区块体部分存放MB(Middle Block)数据;MB(Middle Block)区块体存放SB(Small Block)数据,SB(Small Block)区块体存放具体存储数据。

[0007] BB(Big Block), MB(Middle Block)区块的结构包含如图4所示。Hash,Pre-Hash,Index,Size,User List属于区块头数据,Encryption Data属于区块体的具体数据区域。

[0008] 进一步地,对于BB(Big Block)区块各个字段的含义如下:

[0009] Hash:表示当前区块的哈希值。

[0010] Pre-Hash:表示上一个BB区块的哈希值,以此在形成连续的链式结构,内存或存储

由多个BB组成。

[0011] Index:BB区块的索引值,依据此索引值进行具体的寻址。

[0012] Size:表示整个BB区块的大小。

[0013] User List:表示对该BB区块具有访问权限的用户列表,多个用户可以对同一个BB具有访问权限。

[0014] Encryption Data:表示BB区块的数据区域,该数据区域由多个MB(Middle Block)构成,并可用相应的非对称加密算法进行加密。

[0015] 进一步地,对于MB(Middle Block)区块各个字段的含义如下:

[0016] Hash:表示当前MB区块的哈希值。

[0017] Pre-Hash:表示上一个MB区块的哈希值,以此在BB区块中形成连续的链式结构,BB区块由多个MB区块组成。

[0018] Index:MB区块的索引值,依据此索引值进行具体的内存或存储寻址。

[0019] Size:表示整个MB区块的大小。

[0020] User List:表示对该MB区块具有访问权限的用户列表,多个用户可以对同一个MB具有访问权限。

[0021] Encryption Data:表示MB区块的数据区域,该数据区域由多个SB(Small Block)构成,并可用相应的非对称加密算法进行加密。

[0022] 进一步地,SB(Small Block)区块是内存或存储中最小的数据单元,每一个SB区块表示一个交易或一个具体存储数据,如图5所示结构中各字段说明如下:

[0023] Hash:表示当前区块的哈希值。

[0024] Pre-Hash:表示上一个SB区块的哈希值,以此在MB区块中形成连续的链式结构,MB区块由多个SB区块组成。

[0025] Index:SB区块的索引值,依据此索引值进行具体的内存或存储寻址。

[0026] Size:表示整个SB区块的大小。

[0027] Owner:表示拥有该SB区块的用户,每一个SB只能有一个拥有者。

[0028] Encryption Data:表示SB区块的数据区域,该数据区域表示一个交易信息或一个具体存储数据,并可用相应的非对称加密算法进行加密。

[0029] 在内存或存储中采用本发明提出的三层区块结构更加适应于区块链操作系统,方便交易或存储数据的读写以及寻址,且数据可进行层层加密,每个SB区块都有唯一的拥有者,仅有拥有者使用其私钥可以解密数据,保证每个数据块的安全性。

[0030] 本发明提出的数据在内存或存储中的寻址方式基于三张映射表,BBBT(Big Big Block Table),BBT (Big Block Table),MBT (Middle Block Table),具体结构如图6所示。

[0031] 进一步地,下面以内存寻址为例进行描述。BBBT(Big Big Block Table)是整个内存的全局寻址表,用来寻找具体的BB(Big Block)区块的位置,根据BBBT中存储的BBT Index字段找到具体的BB区块对应的BBT (Big Block Table)映射表,在BBT表中根据MBT Index字段找到相应的MB(Middle Block)区块对应的MBT (Middle Block Table)表,在MBT表中根据SB(Small Block)的Hash值找到对应的SB区块的物理位置。

[0032] 进一步地,在分布式存储中,数据结构同样采用上述的区块式结构,将数据的元数

据信息与数据放在一起，避免了元数据服务器遭受攻击的情形，采用基于内容的寻址方式，数据结构如图7所示。

[0033] 在分布式存储中，对于每一个SB都有多个副本存储在不同机器所以对于MBT表有相应修改，将SB区块的物理位置改为对应的位置列表。分布式存储中MBT结构如图8所示，在MBT中SB的哈希值对应的是相应的机器列表，选择某一机器获得相应SB数据，三张表采用分布式存储方式，每个机器存储距离自己最近的区块的信息，并提供附近机器的区块信息，这样可以快速定位到具体SB区块。

[0034] 优选地，为了保护数据的安全性，每个区块的区块体部分使用非对称加密进行密码学处理。如图9所示，不同用户拥有不同的公钥和私钥，由密钥管理模块统一存储管理。加解密除可CPU处理外，还可在GPU上进行，保证加解密速度。密钥管理模块以及加解密模块都加入操作系统内核中，在内核态执行程序更快速，用户使用公钥加密，使用私钥解密，保证了数据的安全性。

[0035] 综上所述，本发明提出的基于区块链技术的新型计算机存储数据结构以及寻址方式有如下优点：

[0036] 1.本发明提出的数据结构和寻址方式不仅可应用于内存管理、分布式存储，还应用于网络资源管理等。基于内容的哈希寻址，统一(uniform)的数据结构，提供统一资源标识符，应用在操作系统、内存、存储、网络、云、手机、路由等设备上，这样的设计可以节省大量解释传输数据的时间，支持标准化的硬件和软件，使操作系统、内存、存储、数据库、手机、服务器、路由、应用都使用同样数据结构和同样安全机制。

[0037] 2.元数据与数据放在一起，避免处理元数据的系统遭受攻击以至于数据流泄露。

[0038] 3.数据可以加解密，每一个区块都有自己的加解密处理，数据安全性更高。能够快速分解BB,MB及SB，每个区块可以层层加密(像区块链系统一样)，还可以支持安全多方计算，保证数据的隐私与传统的方式不同。

[0039] 4.安全性不是通过手机、操作系统或是应用系统来保证，而是通过私钥加密来保证。操作系统出问题的时候，数据仍然安全，不像传统系统，操作系统出问题，数据安全性就可能无法保证。

[0040] 5.支持以“安全为第一”(对应“计算力第一”)原则的系统，数据可以不发送给其他主机，如果其他机器需要数据，可以将计算软件发送过来，计算后结果送回。这样原数据还是保存在原地，没有离开过，拥有人可以决定永远不送原数据给任何人、单位、或是机器。本发明提出的统一的数据结构，支持送软件和软件结果，而不送数据的功能。

[0041] 6.可兼容传统的存储方式。

附图说明

[0042] 图1为传统的计算机内存管理以及寻址方式示意图；

[0043] 图2为传统的分布式存储架构示意图；

[0044] 图3为本发明提出的基本数据结构示意图；

[0045] 图4为本发明提出的BB,MB区块的基本数据结构示意图；

[0046] 图5为本发明提出的SB区块的基本数据结构示意图；

[0047] 图6为本发明提出的三张映射表BBBT,BBT,MBT的基本结构示意图；

[0048] 图7为本发明提出的分布式存储中SB区块的结构示意图；

[0049] 图8为本发明提出的分布式存储中MBT的结构示意图；

[0050] 图9为本发明提出的实施例中一个操作系统的内存数据结构及寻址方式示意图；

[0051] 图10为本发明提出的实施例中一个操作系统的内存寻址流程示意图；

[0052] 图11位本发明提出的实施例中一个操作系统的内存数据加解密流程示意图。

[0053] 具体实施案例

[0054] 为了更加清晰地描述本发明的结构特点及技术方案，使得本发明描述的内容更加易于理解，下面将举例进行说明。

[0055] 一个操作系统的内存数据结构以及寻址方式如图9所示，可适用于区块链操作系统的要求。

[0056] 数据结构由BB,MB,SB构成，三个映射表由BBBT,BBT,MBT组成。

[0057] BB是一个大的区块，BB由区块头和若干中等区块组成，BB的区块头包含上一个BB的哈希值，时间戳，区块大小等参数，若干中等区块组成了BB的数据部分，数据部分会用公钥进行加密，只有拥有私钥的人才能进行解密。

[0058] MB即中等区块，由区块头和若干小区块构成，区块头包含上一个MB的哈希值，时间戳，区块大小等参数，若干小区块组成了MB的数据部分，数据部分会用公钥进行加密，只有拥有该区块的用户才能进行解密。

[0059] SB是最小的数据块，由标签和数据组成，标签字段包含该SB的哈希值以及拥有者和块大小，分布式操作系统的元数据信息会存储在SB的标签中，数据字段是真正的数据，数据字段会用公钥进行加密，只有拥有该区块的用户才能进行解密。

[0060] BBBT,BBT,MBT是三张存储位置表，CPU要根据三张表去找寻具体的区块，寻址方式如图10所示。首先根据开始地址和BB Hash在BBBT中找到相应的BBT的索引位置，确定BBT位置后查找相应的BBT；在查找到的BBT中根据MB Hash找到相应的MBT的位索引置，确定MBT位置后查找相应的MBT；在查找到的MBT中根据SB Hash找到相应的SB的位置，至此，找到了相应的数据块。

[0061] 本实施例中，区块的数据使用RSA非对称算法，加解密流程如图11所示，使用GPU进行密码学计算。

[0062] 采用这样的数据结构以及寻址方式，基于内容的哈希寻址，更符合区块链操作系统的特点，寻址更高效，在此基础上，我们采用RSA非对称加密算法对每一层区块进行数据的加解密，只有拥有私钥的用户才能对数据解密，保证数据的安全性。

[0063] 虽然本发明已经参考特定的说明性实施例进行了描述，但是不会受到这些实施例的限定而仅仅受到附加权利要求的限定。本领域技术人员应当理解可以在不偏离本发明的保护范围和精神的情况下对本发明的实施例能够进行改动和修改。

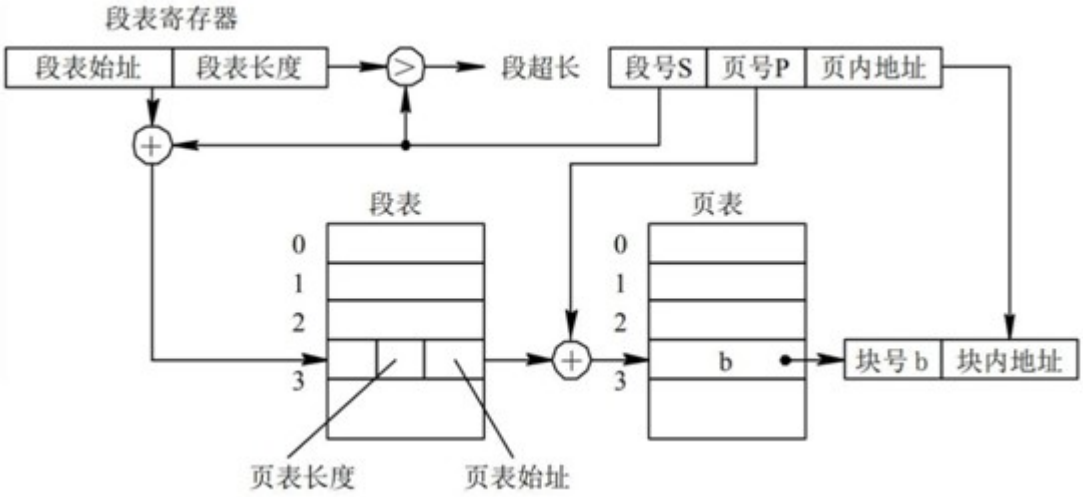

段表寄存器
段表始址
段表长度
>
段超长
段号S
页号P
页内地址
段表
0
1
2
3
页表
0
1
2
3
b
块号 b
块内地址
页表长度
页表始址

图1

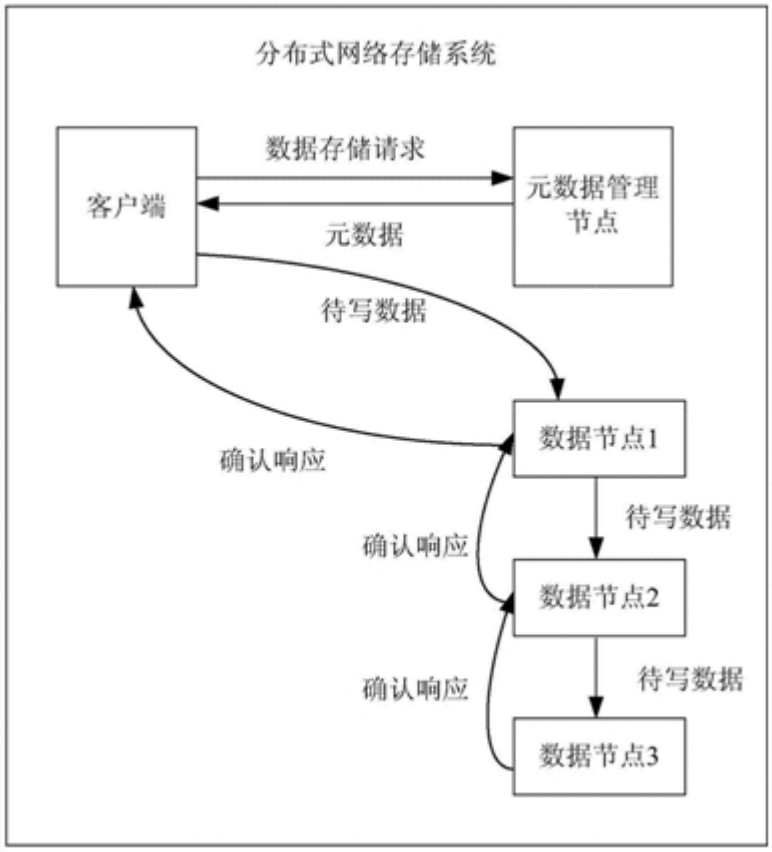

分布式网络存储系统
客户端
数据存储请求
元数据
元数据管理节点
待写数据
确认响应
数据节点1
待写数据
确认响应
数据节点2
待写数据
确认响应
数据节点3

图2

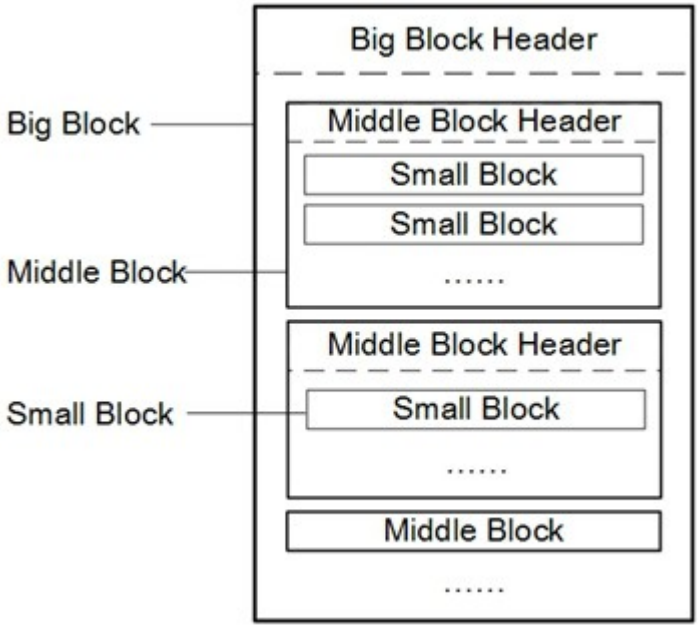


图3

Hash	Pre-Hash	Index	Size	User List	Encryption Data

图4

Hash	Pre-Hash	Index	Size	Owner	Encryption Data

图5

BB Hash	BBT Index
BB Hash	BBT Index
BB Hash	BBT Index
BB Hash	BBT Index

BBBT

MB Hash	MBT Index
MB Hash	MBT Index
MB Hash	MBT Index
MB Hash	MBT Index

BBT

SB Hash	SB Phisical Address
SB Hash	SB Phisical Address
SB Hash	SB Phisical Address
SB Hash	SB Phisical Address

MBT

图6

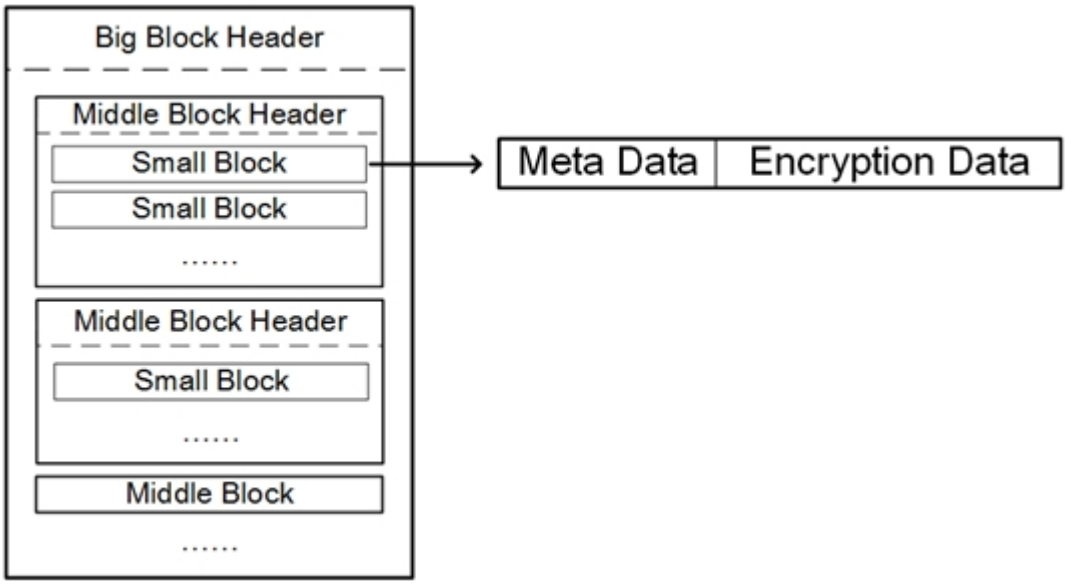


图7

SB Hash	SB Location List
SB Hash	SB Location List
SB Hash	SB Location List
SB Hash	SB Location List

MBT

图8

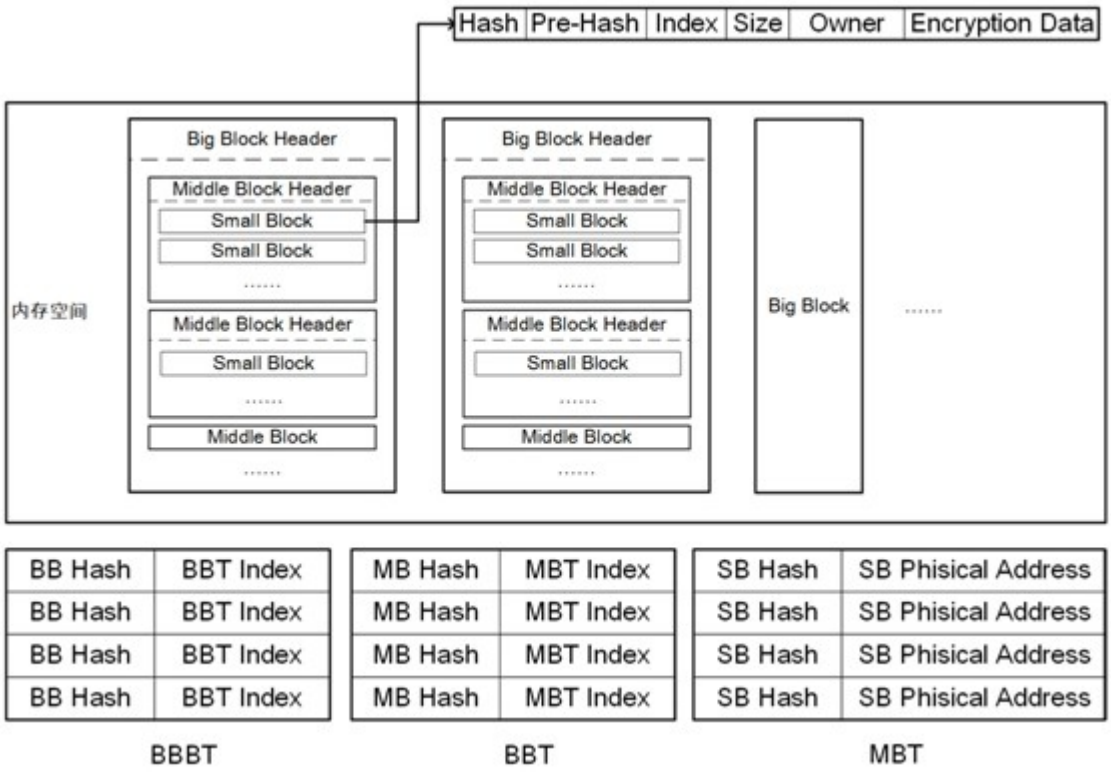

Hash
Pre-Hash
Index
Size
Owner
Encryption Data
内存空间
Big Block Header
Middle Block Header
Small Block
Small Block
.....
Middle Block Header
Small Block
.....
Middle Block
.....
Big Block Header
Middle Block Header
Small Block
Small Block
.....
Middle Block Header
Small Block
.....
Middle Block
.....
Big Block
.....
BB Hash
BBT Index
MB Hash
MBT Index
SB Hash
SB Phisical Address
BBBT
BBT
MBT

图9

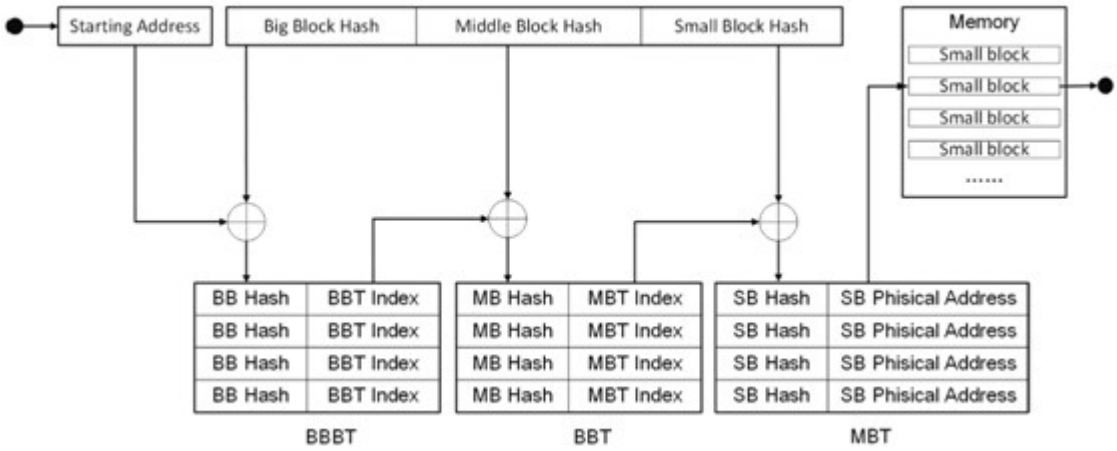

Starting Address
Big Block Hash
Middle Block Hash
Small Block Hash
Memory
Small block
Small block
Small block
Small block
.....
BB Hash
BBT Index
MB Hash
MBT Index
SB Hash
SB Phisical Address
BBBT
BBT
MBT

图10

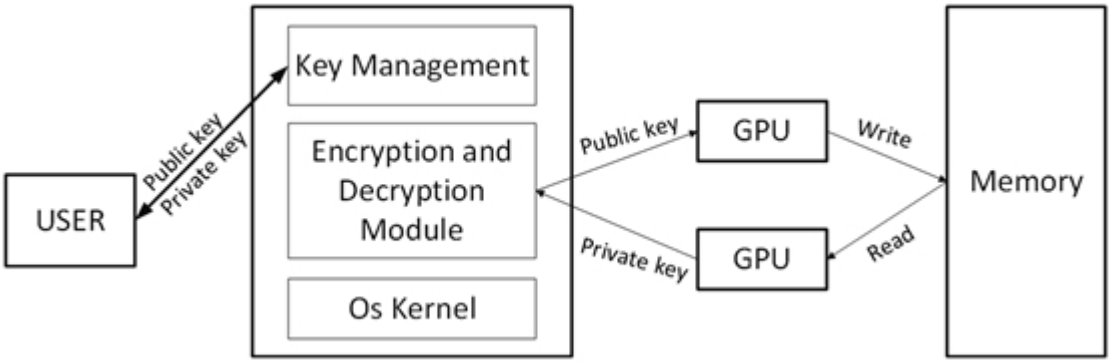

Key Management
Encryption and Decryption Module
Os Kernel
USER
Public key
Private key
Public key
Private key
GPU
GPU
Write
Read
Memory

图11