(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 102656550 B
(45)授权公告日 2015.04.08

(21)申请号 201080056217.4
(22)申请日 2010.11.19
(30)优先权数据
12/634,470 2009.12.09 US
(85)PCT国际申请进入国家阶段日
2012.06.11
(86)PCT国际申请的申请数据
PCT/US2010/057438 2010.11.19
(87)PCT国际申请的公布数据
WO2011/071678 EN 2011.06.16
(73)专利权人 桑迪士克以色列有限公司
地址 以色列萨巴
(72)发明人 E. 科亨 E. 伊塔 L. 格林
U. 佩尔茨 I. 毛尔 Y. 哈勒维
A. 什缪尔
(74)专利代理机构 北京市柳沈律师事务所 11105
代理人 黄小临
(51)Int.Cl.
G06F 21/10(2013.01)
G06F 21/31(2013.01)
G06F 21/62(2013.01)
G06F 21/79(2013.01)
(56)对比文件
US 2008/0195797 A1,2008.08.14,
US 2008/0195797 A1,2008.08.14,
TW 200931254 A,2009.07.16,
US 2009/0271876 A1,2009.10.29,
US 6976165 B1,2005.12.13,
US 2005/0025316 A1,2005.02.03,
CN 101315674 A,2008.12.03,
审查员 杨华

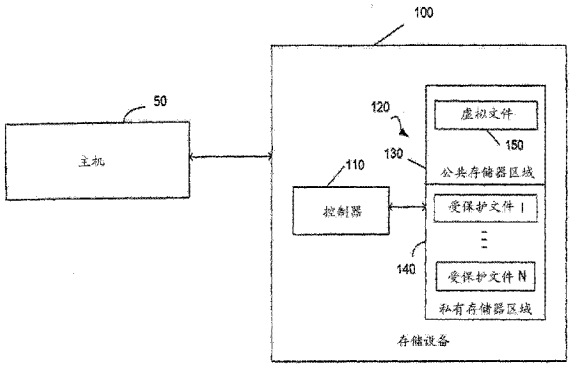
权利要求书2页 说明书8页 附图25页

(54)发明名称

使用公共存储器区域中的虚拟文件来访问私有存储器区域中的多个受保护文件的存储设备和方法

(57)摘要

公开了使用公共存储器区域中的虚拟文件来访问私有存储器区域中的多个受保护文件的存储设备和方法。在一个实施例中，存储设备接收来自主机的访问该公共存储器区域中的虚拟文件的请求，其中该虚拟文件与存储在该私有存储器区域中的多个受保护文件相关联。存储设备通过选择存储在该私有存储器区域中的该多个受保护文件中的一个并为主机提供对该受保护文件的访问来响应该请求。存储设备从主机接收访问虚拟文件的另外的请求，并通过选择存储在该私有存储器区域中的该多个受保护文件中的不同一个并为主机提供对该受保护文件的访问来响应该另外的请求。



CN 102656550 B

1. 一种使用公共存储器区域中的虚拟文件来访问私有存储器区域中的多个受保护文件的方法，该方法包括：

在具有公共存储器区域和私有存储器区域的存储设备的控制器中进行：

(a) 接收来自主机的访问该公共存储器区域中的虚拟文件的请求，其中该虚拟文件与存储在该私有存储器区域中的多个受保护文件相关联；

(b) 通过选择存储在该私有存储器区域中的该多个受保护文件中的一个并为主机提供对该受保护文件的访问来响应访问虚拟文件的该请求；以及

(c) 从主机接收访问虚拟文件的另外的请求，其中所述控制器通过选择存储在该私有存储器区域中的该多个受保护文件中的不同一个并为主机提供对该受保护文件的访问来响应该另外的请求。

2. 如权利要求 1 的方法，其中该控制器在执行存储规则集的应用之后进行 (a)-(c)，该规则集确定控制器如何管理该私有存储器区域中的该多个受保护文件。

3. 如权利要求 1 的方法，其中如果主机已经向存储设备验证，则进行 (b)-(c)，以及其中如果主机还未向存储设备验证，则该控制器通过提供专用广告来响应该请求。

4. 如权利要求 1 的方法，其中该虚拟文件的大小至少与最大的受保护文件的大小一样大。

5. 如权利要求 4 的方法，还包括为主机提供所选受保护文件的大小。

6. 如权利要求 4 的方法，其中所选受保护文件的大小小于虚拟文件的大小，以及其中该方法还包括用数据填充受保护文件以适应大小的差别。

7. 如权利要求 1 的方法，其中该虚拟文件被分配到该公共存储器区域中的物理区域。

8. 如权利要求 1 的方法，其中该虚拟文件未被分配到该公共存储器区域中的物理区域。

9. 如权利要求 1 的方法，其中使用专用文件系统访问该受保护文件。

10. 如权利要求 1 的方法，其中来自主机的请求包括两个或多个命令的复合命令。

11. 如权利要求 1 的方法，还包括进行以下的至少一个：

通过将修改的数据写到该虚拟文件来修改存储在该私有存储器区域中的受保护文件；以及

通过将新的受保护文件写到该虚拟文件来向该私有存储器区域添加新的受保护文件。

12. 如权利要求 1 的方法，其中受保护文件是加密的。

13. 如权利要求 1 的方法，还包括使用计数器来防止虚拟文件被复制。

14. 如权利要求 1 的方法，还包括使用数据速率控制机制来防止虚拟文件被复制。

15. 如权利要求 1 的方法，还包括使用滑动窗来防止虚拟文件被复制。

16. 一种存储设备，包括：

公共存储器区域；

私有存储器区域；以及

控制器，与该公共存储器区域以及私有存储器区域通信，其中该控制器被配置为：

(a) 接收来自主机的访问该公共存储器区域中的虚拟文件的请求，其中该虚拟文件与存储在该私有存储器区域中的多个受保护文件相关联；

(b) 通过选择存储在该私有存储器区域中的该多个受保护文件中的一个并为主机提供

对该受保护文件的访问来响应访问虚拟文件的该请求；以及

(c) 从主机接收访问虚拟文件的另外的请求，其中所述控制器被配置为通过选择存储在该私有存储器区域中的该多个受保护文件中的不同一个并为主机提供对该受保护文件的访问来响应该另外的请求。

17. 如权利要求 16 的存储设备，其中该控制器可操作以在执行存储规则集的应用之后进行 (a)-(c)，该规则集确定控制器如何管理该私有存储器区域中的该多个受保护文件。

18. 如权利要求 16 的存储设备，其中该控制器可操作以便如果主机已经向存储设备验证则进行 (b)-(c)，以及其中该控制器可操作以便如果主机还未向存储设备验证则通过提供专用广告来响应该请求。

19. 如权利要求 16 的存储设备，其中该虚拟文件的大小至少与最大的受保护文件的大小一样大。

20. 如权利要求 19 的存储设备，其中该控制器还可操作以为主机提供所选受保护文件的大小。

21. 如权利要求 19 的存储设备，其中所选受保护文件的大小小于虚拟文件的大小，以及其中该控制器还可操作以用数据填充受保护文件以适应大小的差别。

22. 如权利要求 16 的存储设备，其中该虚拟文件被分配到该公共存储器区域中的物理区域。

23. 如权利要求 16 的存储设备，其中该虚拟文件未被分配到该公共存储器区域中的物理区域。

24. 如权利要求 16 的存储设备，其中使用专用文件系统访问该受保护文件。

25. 如权利要求 16 的存储设备，其中来自主机的请求包括两个或多个命令的复合命令。

26. 如权利要求 16 的存储设备，其中该控制器还可操作以进行以下的至少一个：

通过将修改的数据写到该虚拟文件来修改该私有存储器区域中存储的受保护文件；以及

通过将新的受保护文件写到该虚拟文件来向该私有存储器区域添加新的受保护文件。

27. 如权利要求 16 的存储设备，其中受保护文件是加密的。

28. 如权利要求 16 的存储设备，其中该控制器还可操作以使用计数器来防止虚拟文件被复制。

29. 如权利要求 16 的存储设备，其中该控制器还可操作以使用数据速率控制机制来防止虚拟文件被复制。

30. 如权利要求 16 的存储设备，其中该控制器还可操作以使用滑动窗来防止虚拟文件被复制。

使用公共存储器区域中的虚拟文件来访问私有存储器区域中的多个受保护文件的存储设备和方法

背景技术

[0001] 诸如存储卡的存储设备通常被用于存储诸如数字音频(例如音乐)和/或视频(例如电影)文件的内容。为了防止内容被未授权访问,内容可以存储在存储设备中的私有存储器区域中,其仅可由验证的主机访问。通常,主机将其证明呈献给存储设备用于验证。如果主机被验证,则存储设备允许主机访问存储在私有存储器区域中的内容。尽管此安全系统防止未授权的主机访问存储在私有存储器区域中的内容,但是如果验证的主机上具有病毒或者其他恶意软件,则可能出现问题。在该情况下,一旦验证的主机被允许访问私有存储器区域,则主机上的恶意软件可以利用该访问来对私有存储器区域中存储的数据进行未授权动作。

发明内容

[0002] 本发明的实施例由权利要求限定,此部分中的任何内容不应被当作对那些权利要求的限制。

[0003] 作为例子,以下描述的实施例通常涉及使用公共存储器区域中的虚拟文件来访问私有存储器区域中的多个受保护文件的存储设备和方法。在一个实施例中,存储设备接收来自主机的访问该公共存储器区域中的虚拟文件的请求,其中该虚拟文件与存储在该私有存储器区域中的多个受保护文件相关联。存储设备通过选择存储在该私有存储器区域中的该多个受保护文件中的一个并为主机提供对该受保护文件的访问来响应该请求。存储设备从主机接收访问虚拟文件的另外的请求,并通过选择存储在该私有存储器区域中的该多个受保护文件中的不同一个并为主机提供对该受保护文件的访问来响应该另外的请求。

[0004] 提供了其他实施例,并且每个实施例可以单独使用或者组合在一起使用。现在将参考附图描述各个实施例。

附图说明

[0005] 图 1 是实施例的主机和存储设备的框图。

[0006] 图 2 是实施例的主机和存储设备的框图。

[0007] 图 3A 是实施例的主机的框图。

[0008] 图 3B 是实施例的存储设备的框图。

[0009] 图 4A 和 4B 是实施例的登录处理的流程图。

[0010] 图 5A 和 5B 是实施例的登录后读取处理的流程图。

[0011] 图 6A 和 6B 是实施例的未登录而读取处理的流程图。

[0012] 图 7 是实施例的访问常规文件的处理的流程图。

[0013] 图 8A 和 8B 是用于访问范围之外的数据并为文件填充另外的数据的实施例的处理的流程图。

[0014] 图 9A 和 9B 是实施例的确定文件的实际长度的处理的流程图。

[0015] 图 10A 和 10B 是使用加密的实施例的保护处理的流程图。

[0016] 图 11A 和 11B 是使用计数器的实施例的保护处理的流程图。

[0017] 图 12A 和 12B 是使用滑动窗的实施例的保护处理的流程图。

[0018] 图 13A 和 13B 是用于数据速率控制的实施例的处理的流程图。

[0019] 图 14A 和 14B 是用于添加内容的实施例的处理的流程图。

具体实施方式

[0020] 介绍

[0021] 以下实施例通常涉及使用公共存储器区域中的虚拟文件来访问私有存储器区域中的多个受保护文件的存储设备和方法。在这些实施例中，公共存储器区域包含担当对于私有存储器区域中的多个受保护文件的网关的虚拟文件。当存储设备中的控制器(例如根据访问虚拟文件的逻辑块地址的尝试)检测到主机正试图访问该虚拟文件时，控制器确定多个受保护文件中的哪个应该被提供给主机，如果存在该文件的话。

[0022] 存在与这些实施例相关联的几个优点。例如，Java 和其他预定义的应用程序接口(API)(比如在许多手持设备和移动电话中使用的)仅允许基于文件的命令并且不使用可以用于控制存储设备以提供对私有存储器区域中的具体受保护文件的访问的低级命令。这就是为什么有些存储设备可以准许主机访问整个私有存储器区域的原因，当主机上的恶意软件利用对于私有存储区域的此开放的访问时，这可能导致未授权的动作。尽管可以写入支持受保护文件流出私有存储区域的专有 API，但是非常少的移动设备将允许这样的 API。使用公共存储器区域中的虚拟文件作为对于私有存储器区域中的受保护文件的网关克服了此问题，因为 Java 和其他预定义的 API 使用的基于文件的命令可以用于访问虚拟文件，其中存储设备中的控制器负责为主机选择和提供对适当受保护文件的访问。以此方式，这些实施例提供了可以用在多个主机上以及不同操作系统上的通用解决方案，都同时使得存储设备能够以安全的方式控制数据流入以及流出存储设备。

[0023] 示例的虚拟文件实施例

[0024] 现在转向附图，图 1 是实施例的与存储设备 100 通信的主机 50 的框图。如在此使用的，短语“与……通信”意味着直接与之通信或者通过一个或多个组件间接与之通信，该一个或多个组件可能有或者可能没有在此示出或描述。主机 50 可以采取任何适当的形式，比如但不限于专用内容播放器、移动电话、个人计算机(PC)、游戏设备、个人数字助理(PDA)、信息站和 TV 系统，存储设备 100 也可以采取任何适当的形式，比如但不限于手持、可移除存储卡(例如闪存卡)、通用串行总线(USB)设备以及固态驱动器。优选地，存储设备 100 可移除地连接到主机 50，以便用户可以与各种主机一起使用存储设备 100。

[0025] 如图 1 所示，存储设备 100 包括控制器 110 和存储器 120。控制器 110 可以按任何适当的方式实现。例如，控制器 110 可以采取微处理器或者处理器以及例如存储可由(微)处理器执行的计算机可读程序代码(例如软件或固件)的计算机可读介质、逻辑门、开关、专用集成电路(ASIC)、可编程逻辑控制器以及嵌入的微控制器的形式。控制器的例子包括但不限于以下微控制器：ARC 625D、Atmel AT91SAM、Microchip PIC18F26K20 以及 Silicon Labs C8051F320。可以在控制器中使用的各个组件的例子在以下讨论的实施例中描述并示出在相关附图中。控制器 110 也可以作为存储器控制逻辑的部分而实现。

[0026] 还如图 1 所示，存储设备 100 包括存储器 120，其可以采取任何适当的形式。在一个实施例中，存储器 120 采取固态（例如，快闪）存储器的形式并且可以是一次可编程、数次可编程或者多次可编程的。但是，可以使用诸如光存储器和磁存储器的其他形式的存储器。尽管在图 1 中被示出为单个组件，但是控制器 110 和 / 或存储器 120 可以利用几个组件来实现。在图 1 中，存储器 120 包括公共存储器区域 130 和私有存储器区域 140。在此实施例中，公共和私有存储器区域 130、140 是单个存储器设备中的不同分区；但是，在其他实施例中，公共和私有存储器区域 130、140 是不同的存储器设备。公共存储器区域 130 通常可以没有限制地访问，而私有存储器区域仅可以由被授权的实体访问并且通常对主机不可见（例如隐藏的分区）。这样，私有存储器区域 140 可以用于存储仅应该由被授权的实体访问的多个内容文件（在此的文件 1-N）。“内容文件”可以采取任何适当的形式，比如但不限于数字视频（带有或者不带有随附的音频）（例如电影、一集电视剧、新闻节目等）、音频（例如歌曲、播客、一个或一系列声音、音频书等）、静止或运动图像（例如照片、计算机产生的显示等）、文本（带有或者不带有图形）（例如文章、文本文件等）、视频游戏或其他软件以及这些形式中的两个或多个的混合多媒体呈现。术语“内容”、“内容文件”以及“文件”在此将可互换地使用，并且存储在私有存储器区域 140 中的文件将被称为“受保护文件”。

[0027] 在此实施例中，公共存储器区域 130 包含虚拟文件 150。虚拟文件 150 在以下意义上是“虚拟的”：其作为公共存储器区域 130 的文件分配表（FAT）中的已分配的逻辑块地址而存在，但是不包含任何数据并且实际上不可访问。对于虚拟文件 150 甚至可能不存在在公共存储器区域 130 中分配的任何物理存储位置（尽管如下所述在某些实施例中在公共存储器区域 130 中可以分配相对小量的物理存储位置（例如 15MB）用于虚拟文件 150）。但是，因为虚拟文件 150 具有分配给其的在公共存储器区域 130 中的 FAT 表中的逻辑块地址，所以对主机 50 而言虚拟文件 150 看起来是常规文件。

[0028] 在此实施例中，虚拟文件 150 用作私有存储器区域 140 中的多个受保护文件的网关。即，当控制器 110（例如根据在读命令中指定的逻辑块地址）识别出正进行访问虚拟文件 150 的尝试时，控制器 110 将采取特殊动作而不是为主机提供对虚拟文件 150 本身的访问。例如，当控制器 110 接收到来自主机 50 的访问虚拟文件 150 的请求时，控制器 110 可以选择存储在私有存储器区域 140 中的多个受保护文件 1-N 中的一个并且通过为主机 50 提供对所选受保护文件的访问来对该请求做出响应。当相同的虚拟文件 150 稍后被访问时，控制器 110 可以选择多个受保护文件 1-N 中的不同一个并提供对其的访问。例如，如果虚拟文件 150 被命名为“audio.MP3”并且与存储在私有存储器区域 140 中的 20 首歌曲的播放列表相关联，则每次主机 50 请求访问“audio.MP3”时，控制器 110 可以选择播放列表上的另一歌曲并提供对其的访问。

[0029] 任何适当的实现方式可以用于允许控制器 110 处理对于虚拟文件 150 的主机请求。例如，在图 2 所示的一个实施例中，应用（或者 caplet（胶囊式应用程序）160）存储确定控制器 110 管理私有存储器区域 140 中的受保护文件的方式（例如要访问哪些受保护文件以及何时）的规则集。在此实施例中，caplet 160 被存储在私有存储器区域 140 中以防止黑客篡改规则集。控制器 110 使用 caplet 160 中的规则集来确定如何对来自主机 50 的访问虚拟文件 150 的请求作出反应。私有存储器区域 140 还存储将多个受保护文件 1-N 与虚拟文件 150 相关联的数据库 170。以此方式，数据库 170 担当多个受保护文件 1-N 的播放

列表。在一个实施例中，数据库 170 与多个受保护文件 1-N 一起被预加载到存储设备 100 中。如果数据库 170 被加密，则 caplet 160 可以保持解密数据库 170 的密钥。

[0030] 在操作中，当存储设备 100 被上电时，caplet 160 规划 (program) 控制器 110 以根据预加载的数据库 170 控制主机 50 对虚拟文件 150 的访问尝试。例如，响应于访问虚拟文件 150 的请求，控制器 110 可以基于数据库 170 选择多个受保护文件 1-N 中的一个并提供对其的访问。作为另一例子，控制器 110 可以确定主机 50 是被授权还是未被授权，以及可以据此提供对不同类型的内容的访问，如果存储设备 100 是可以与可能被授权或者可能未被授权的各种主机一起使用的便携式存储卡，则这可能是希望的。因此，如果主机 50 是被授权的播放器，则控制器 110 可以通过提供对受保护内容读访问而对来自主机 50 的访问虚拟文件 150 的请求做出响应。否则，控制器 110 可以通过提供对替换的文件的访问来对该主机请求做出响应。以此方式，规则集可以指定如果主机 50 上的应用被授权则控制器 110 应该提供对受保护文件 (例如电影) 的访问 ；否则，控制器 110 可以提供对替换的文件 (例如该电影的预告片) 的访问，该替换文件可以存储在公共或者私有存储器区域 130、140 中。以此方式，访问虚拟文件 150 的类似主机请求可以由控制器 110 基于主机 50 是被授权还是未被授权的实体而不同地处理 (例如如果用户没有登录或者未被存储设备 100 验证，则存储设备 100 可以返回专门的广告 (commercial))。当然，这仅仅是一个例子，并且可以使用其他类型的规则和条件。例如，文件的选择可以基于时间、关于用户的信息、主机是否登录到存储设备 50 中以及如何登录或者任意其他适当的条件。从这些例子可以看出，在不同的情况下可以不同地对待虚拟文件 150，并且公共存储器区域 130 中的单个虚拟文件 150 可以用于提供对私有存储器区域 140 中的一个或多个受保护文件的访问。

[0031] 现在返回图 2，在此实施例中，主机 50 运行主机应用 60 和媒体播放器 (在此 Java 播放器 70)。主机应用 60 控制 Java 播放器 70 并且还向存储设备 100 的控制器 110 发送另外的指令，这将在以下描述。如上所述，在此实施例中，虚拟文件 150 (例如名为“audio.MP3”) 与存储在私有存储器区域 140 中的多个受保护文件 1-N 相关联。每次主机 50 请求访问“audio.MP3”时，控制器 110 可以选择多个受保护文件 1-N 中的不同一个并提供对其的访问。在此实施例中，虚拟文件 150 的大小至少和私有存储器区域 140 中的最大的受保护文件的大小一样大。如果所选的受保护内容小于虚拟文件 150 的大小，则控制器 110 可以用有效数据填充该内容，以便主机 50 中的 Java 播放器 70 不出故障。例如，如果受保护文件是音频文件，则控制器 110 可以实时地 (on the fly) 用诸如无声的 MP3 帧或者期望的头部信息的有效数据来填充该音频文件。这样做以能够支持不同的主机先行缓存 (cache-ahead) 机制。为了防止控制器 110 播放填充的数据，控制器 110 可以通知主机 50 文件的实际播放长度，并且主机 50 可以在其到达填充的数据之前停止文件的回放。

[0032] 考虑其中 Java 播放器 70 使用 JSR-135 协议向存储设备 110 发送请求来读取虚拟文件 150 的例子，该 Java 播放器 70 认为虚拟文件 150 是音频文件。存储设备 100 中的控制器 110 (例如根据在请求中指定的逻辑块地址) 检测到正进行访问虚拟文件 150 的尝试，并且基于 caplet160 中的规则，控制器 110 还向主机应用 60 通知歌曲 A 的实际大小，该实际大小可能小于虚拟文件 150 的大小，如上所述。主机应用 60 监视回放，并且当歌曲 A 完成时，主机应用 60 经由 JSR-75 协议向控制器 110 发送命令以停止流式传输数据并跳到下一歌曲。(JSR-135 协议不支持此类型的命令。) 响应于此命令，控制器 110 更新数据库 170

以指向下一歌曲(歌曲 B)。主机应用 60 指示 Java 播放器 70 再次请求访问虚拟文件 150。但是,这次,控制器 110 为 Java 播放器 70 选择并提供对歌曲 B (不是歌曲 A)的访问。然后在 Java 播放器 70 继续请求回放另外的歌曲时,上述处理重复。

[0033] 存在可以与这些实施例一起使用的许多替换。例如,在一个实施例中,存储设备 100 使用“专用文件系统”用于回放预加载的受保护文件以最小化存储设备 100 定位和访问该预加载的受保护文件所花费的时间以及最小化提取关于预加载到的受保护文件的细节(例如音轨名称、艺术家名字、专辑名称、音轨持续时间等)所花费的时间。以此方式,不考虑在向受保护文件提供网关功能性时所涉及的控制器 110 开销,访问虚拟文件 150 的时间应该与访问另一类型的文件的时间大约相同。为了实施此专用文件系统,受保护文件可以按连续的逻辑块地址(LBA)的顺序预加载到私有存储器区域 140 中。然后在产生期间建立表格,用于指定每个受保护文件的确切开始 LBA,连同任何其他相关数据比如文件持续时间。因为读取这样的表格比打开文件以收集所需信息更快,所以使用此专用文件系统提供了更快的响应时间。

[0034] 如从以上描述可以理解,“专用文件系统”可以用于降低定位和访问不同的预加载内容所需的时间量。这可以通过定义用于以具体方式保存仅回放所需的数据的专用结构(例如表格)来进行。根据产品特定要求(例如音轨名称、艺术家名字、专辑名称、音轨持续时间等),此数据可以在不同的产品之间改变。此数据结构可以按连续的 LBA 顺序预加载到存储设备 100 中的安全位置中,保持该数据的表格与之前指定的其他数据一起位于每个文件的确切开始 LBA 处。表格中的每个条目可以表示具体文件,该具体文件可以使用表格中的条目键 (key) 来标识。基于访问具体文件的请求,在表格中定位该文件,并检索相关数据。例如,使用虚拟文件 150 从开头播放歌曲可能需要知道文件的开始 LBA、其实际持续时间以及要播放的专辑名称。按这样的处理文件的方式,与处理 FAT 表和解析文件内容所需的时间相比,定位和回放文件所需的时间量急剧降低。

[0035] 作为对于提供更快的响应时间的另一替换,存储设备 100 可以支持使用“复合命令”。复合命令是并入了控制回放系统的不同方面的两个或多个命令——比如例如改变虚拟文件状态和正播放的歌曲两者——的单个命令。在其中希望快速响应时间的诸如音频流的环境下可能尤其希望使用复合命令。例如,复合命令可以指定“跳过频道”和“跳过前面的两首歌曲”或者“跳过歌曲并进入暂停”。在许多情况下,可能需要同时发生几个模式改变。例如,“播放歌曲、进入暂停模式、然后跳到下一歌曲”的序列可能需要在跳过发生后立即开始下一歌曲。在此情况下,可能需要在总线上发送两个命令以便进行此改变。在另一例子中,可能需要同时控制系统的不同方面,比如改变虚拟文件状态和正播放的歌曲两者。再次,想法是减少处理存储设备 100 上的状态改变所需的时间量以及将命令从主机 50 传输到存储设备 100 以及传输回所花费的时间量。

[0036] 在另一替换中,代替为虚拟文件 150 分配公共存储器区域 130 中的实际存储器,可以通过下述操作来减少存储器消耗 :使公共存储器区域 130 针对主机 50 将其本身模拟为扩展的存储区域,以便对于该扩展的(模拟的)存储区域的主机请求,控制器 110 就像该主机请求是对存储设备的私有存储器区域 140 的主机请求那样处理。结果,虚拟文件 150 不消耗公共存储器区域 130 中的实际存储器。这允许存储大的文件(例如电影)而不消耗珍贵的存储器。例如考虑其中虚拟文件 150 是 15MB 并且公共存储器区域 130 是 30MB 的情况。使用

上述技术，存储设备 100 可以将其本身标识为具有 45MB 公共存储器区域 130，其中 30MB 被映射到物理地址而 15MB 没有被映射到物理地址。

[0037] 尽管上述实施例涉及从存储设备中读出受保护文件，但是在替换实施例中，虚拟文件 150 用于在私有存储器区域 140 中添加新内容，如果主机 50 被授权这样做的话。例如，当用户购买歌曲时，主机应用 70 可以控制私有存储器区域 140 以在其空闲空间中添加新内容，然后使用虚拟文件 150 将新歌曲数据写到私有存储器区域 140。在另一例子中，存储的广告可以被更新为新广告。这可以通过将虚拟文件 150 链接到此私有文件然后通过向虚拟文件 150 写入而改变其内容来实现。存储设备 100 可以截获写命令，并将写操作改变到私有存储器区域 140 中的正确地点。这使能够更新私有存储器区域 140 以按安全的方式保持更新后的内容和购买的内容。

[0038] 另外的替换涉及复制保护机制。因为虚拟文件 150 在公共存储器区域 130 中是可访问的，所以可以容易地通过来自任何主机的简单复制命令而复制虚拟文件 150。以下替换可以用于防止虚拟文件 150 的复制，由此确保对存储在私有存储器区域 140 中的受保护内容的访问仅被给予被允许访问该内容的验证的实体。否则，主机 50 可以简单地将虚拟文件 150 复制到公共存储器区域 130 中的另一地址范围并自由地访问受保护内容。

[0039] 一个示例的复制保护机制使用加密方案来加密通过对虚拟文件 150 的读取操作从存储设备 100 发送出的受保护内容。这可以通过使用会话密钥来进行，每次在主机应用 60 和存储设备 100 之间应用验证处理时（例如在登录期间）更新会话密钥。此会话密钥对主机 50 和存储设备 100 两者是已知的，因此存储设备 100 可以使用该会话密钥来加密受保护内容，并且主机 50 可以使用该会话密钥来解密该受保护内容。以此方式使用会话密钥创建了在存储设备 100 和主机应用 60 之间的安全信道。

[0040] 另一示例复制保护机制在存储设备 100 中使用计数器以对已被主机 50 读取的扇区计数。当计数器达到零时，存储设备 100 可以开始发送无声音频的缓冲。主机 50 在适当的验证后以后可以指示存储设备 100 随着回放时间而增加计数器。设置计数器的此操作可以在文件读取期间（例如在歌曲的回放期间）进行。在进行尝试复制文件的情况下，存储设备 100 可以检测到该尝试并开始向主机 50 返回无效数据。这防止复制歌曲，因为在不增加计数器的情况下黑客仅可以获取一定量的歌曲（例如 5MB 歌曲中的 1MB 音频）。

[0041] 在另一示例复制保护机制中，在检查从主机 50 到虚拟文件 150 的读取操作的样式之后，由存储设备 100 实施数据速率控制（“DRC”）。如果这些读操作不像预期那样发生（例如由于快速和急速的文件访问，存储设备 100 检测到正发生虚拟文件 150 的复制），存储设备 100 可以返回无效数据或者使读取处理失败。以此方式，存储设备 100 对从存储设备 100 读取的受保护文件实施数据速率控制。主机 50 可能能够在向存储设备 100 进行适当的验证之后配置存储设备 DRC 机制以符合当前主机 50 的具体特征。

[0042] “滑动窗”也可以被用作示例的复制保护机制。利用此机制，存储设备 100 允许主机 50 读取仅在特定 LBA 范围内的实际数据。访问在此 LBA 范围之外的数据返回无声音频的缓冲。主机 50 在适当的验证之后可以重配置允许的范围。因为允许的范围移动，所以在此将其称为“滑动窗”。以上讨论的计数器的概念可以被用作滑动窗（例如对于目前复制从 1 到 500），或者可以允许仅从虚拟文件 150 内部的特定位置开始回放数据。

[0043] 在另一实施例中，虚拟文件 150 可以用于从本地的 (native) 媒体播放器回放。这

可以通过为每个内容播放列表使用相对大的虚拟文件来进行。此虚拟文件可以实时地 (on the fly) 一首接一首地连接起来。存储设备 100 也可以为要在主机本地的播放器中显示的每个信道的内容提供 ID3 标签。以此方式，可以在每个可用主机（例如手持机）或者操作系统上使用存储设备 100。可以通过允许存储设备感测尝试跳过歌曲的方式（例如通过检测用户何时试图快进 / 倒回 / 拖动进度条）来增强此方案。

[0044] 示例的存储设备和处理流程

[0045] 如上所述，这些实施例的存储设备可以按任何适当的方式实现。以下段落和参考的附图描述了一个示例实现方式。应该理解，此实现方式仅仅是例子，并且在此示出和描述的细节不应被解释到权利要求中，除非其中明确列出。

[0046] 返回到附图，图 3A 和 3B 是实施例的存储设备 300 和主机 350 的框图。首先以图 3B 开始，存储设备 300 包括控制器 310 和存储器 320。控制器 310 包括用于与存储器 320 相接口的存储器接口 311 和用于与主机 350 相接口的主机接口 312。控制器 310 还包括中央处理单元（CPU）313、可操作以提供加密和 / 或解密操作的密码引擎 314、读存取存储器（RAM）315、存储用于存储设备 300 的基本操作的固件（逻辑）的只读存储器（ROM）316 以及存储用于加密 / 解密操作的设备专用密钥的非易失性存储器（NVM）317。应该注意，存储设备专用密钥可以存储在存储设备内的其他存储器区域中。图 3B 中所示的组件可以按任何适当的方式实现。

[0047] 在此实施例中，存储器 320 包括由主机 350 上的文件系统管理的公共分区 325 以及由控制器 310 内部管理的隐藏受保护系统区域 335。隐藏受保护系统区域 335 存储内容加密密钥（CEK）340、内容、数据和 caplet （胶囊式应用程序）342，如上所述。隐藏受保护系统区域 335 是“隐藏的”，因为其由控制器 310 （而不是由主机控制器 360）内部管理，并且是“受保护的”，因为存储在该区域 335 中的对象被用存储在控制器 310 的非易失性存储器 317 中的特有密钥加密。（存储设备硬件特有密钥可以存储在控制器 310 的非易失性存储器 317 中或者存储设备 300 内的其他区域。）因而，为了访问存储在该区域 335 中的对象，控制器 310 将使用密码引擎 314 和存储在非易失性存储器 317 中的密钥来解密加密的对象。优选地，存储设备 300 采取来自在 SanDisk 公司的 TrustedFlash™平台上构建的系列产品的安全产品的形式。公共分区 325 包含虚拟文件 330。

[0048] 现在转向图 3A 的主机 350，主机 350 包括控制器 360，该控制器 360 具有用于与存储设备 300 相接口的存储设备接口 361。控制器 360 还包括中央处理单元（CPU）363、可操作以提供加密和 / 或解密操作的密码引擎 364、读存取存储器（RAM）365 和只读存储器（ROM）366。应该注意，框 360 中的每个组件可以实现为整个主机系统中的单独的芯片。主机 350 还包括应用 370，该应用 370 包括内容引擎 371、文件系统 API 372、内容解码器 API 373 和主机证书 374。

[0049] 存储设备 300 和主机 350 经由存储设备接口 361 和主机接口 362 彼此通信。对于涉及数据的安全传输的操作，优选存储设备 300 和主机 350 中的密码引擎 314、364 用于彼此相互验证并提供密钥交换。相互验证处理要求主机 350 和存储设备 300 交换特有证书 ID。在相互验证完成后之后，优选会话密钥被用于建立用于在存储设备 350 和主机 300 之间的通信的安全信道。

[0050] 图 4A 到 14B 是使用图 3A 和 3B 中的存储器设备 300 和主机 350 的实施例的几个

示例处理流程的流程图。具体地，图 4A 和 4B 是实施例的登录处理的流程图，图 5A 和 5B 是实施例的登录后读取处理的流程图，图 6A 和 6B 是实施例的未登录而读取处理的流程图，图 7 是实施例的访问常规文件的处理的流程图，图 8A 和 8B 是用于访问在范围之外的数据并用另外的数据填充文件的实施例的处理的流程图，图 9A 和 9B 是实施例的确定文件的实际长度的处理的流程图，图 10A 和 10B 是使用加密的实施例的保护处理的流程图，图 11A 和 11B 是使用计数器的实施例的保护处理的流程图，图 12A 和 12B 是使用滑动窗的实施例的保护处理的流程图，图 13A 和 13B 是数据速率控制的实施例的处理的流程图，以及图 14A 和 14B 是用于添加内容的实施例的处理的流程图。

[0051] 结论

[0052] 意图将以上详细描述理解为对本发明可以采取的所选形式的例示而不是对本发明的限定。意图仅以下权利要求、包括所有等效物来定义要求保护的本发明的范围。最后，应该注意，在此所述的任何优选实施例的任何方面可以单独使用或者彼此组合使用。

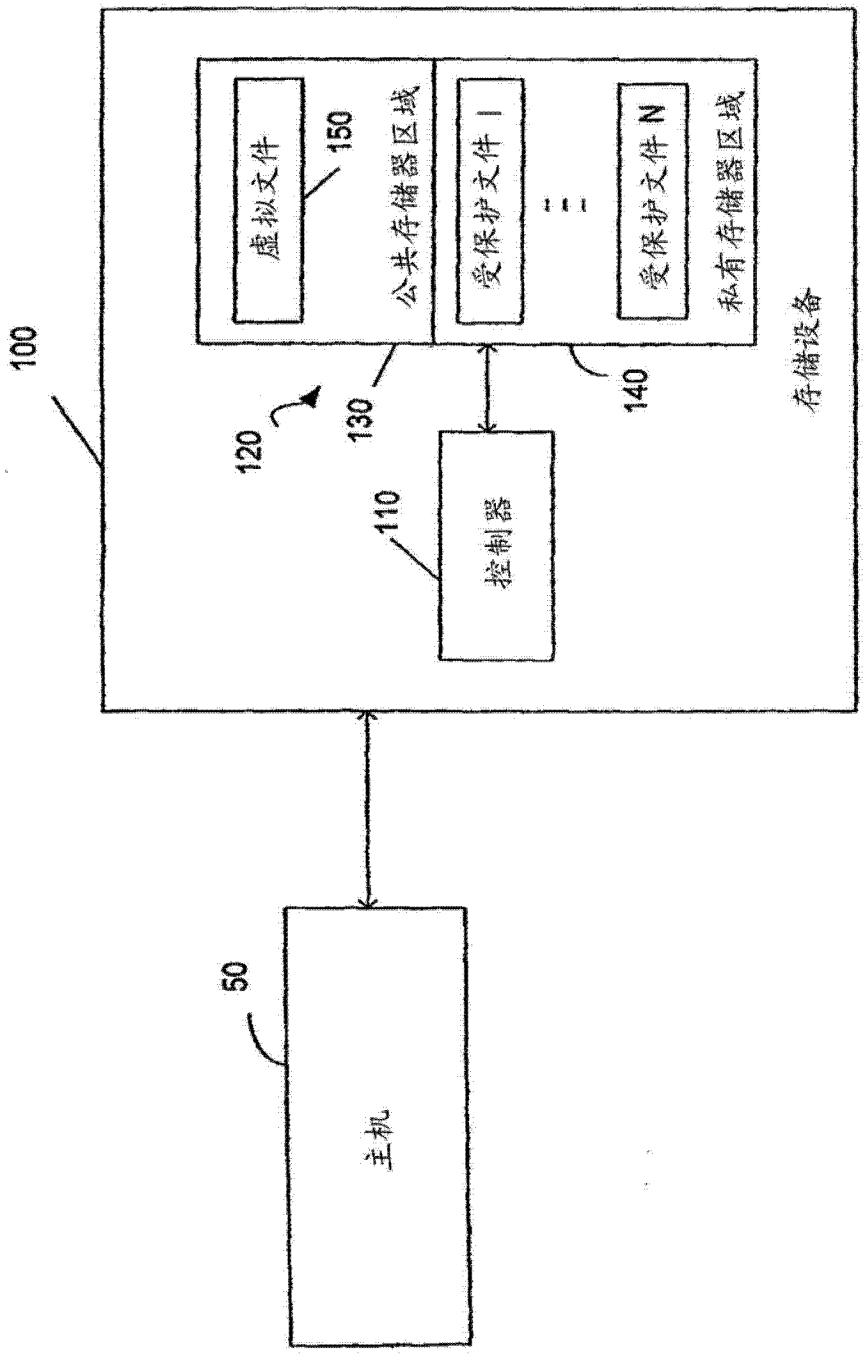

100
虚拟文件
150
公共存储器区域
受保护文件 1
受保护文件 N
私有存储器区域
存储设备
120
130
140
110
控制器
50
主机

图 1

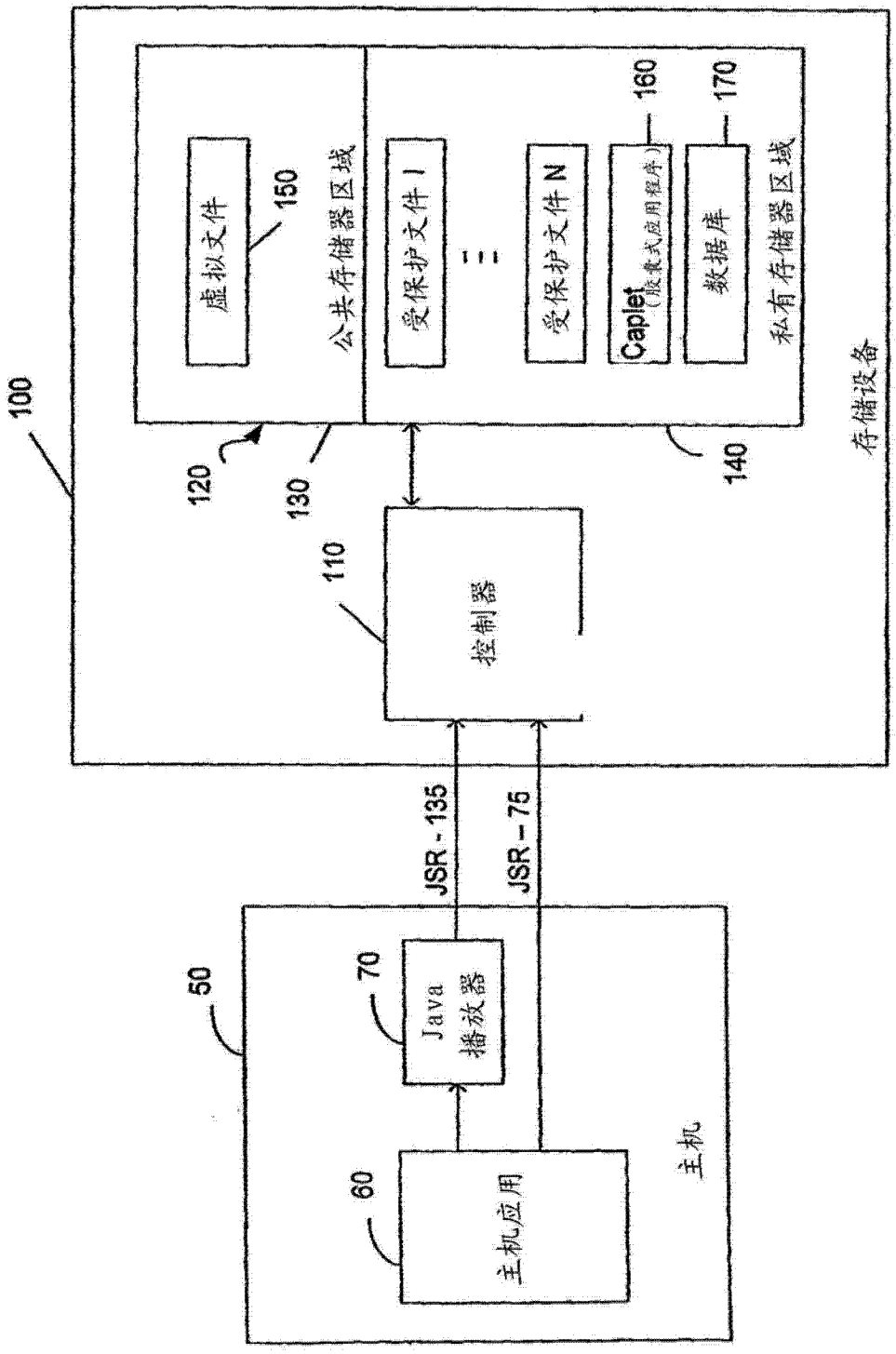

100
虚拟文件
150
公共存储器区域
受保护文件 1
受保护文件 N
Caplet
(嵌入式应用程序)
160
数据库
170
私有存储器区域
存储设备
120
130
140
110
控制器
JSR - 135
JSR – 75
50
70
Java
播放器
60
主机应用
主机

图 2

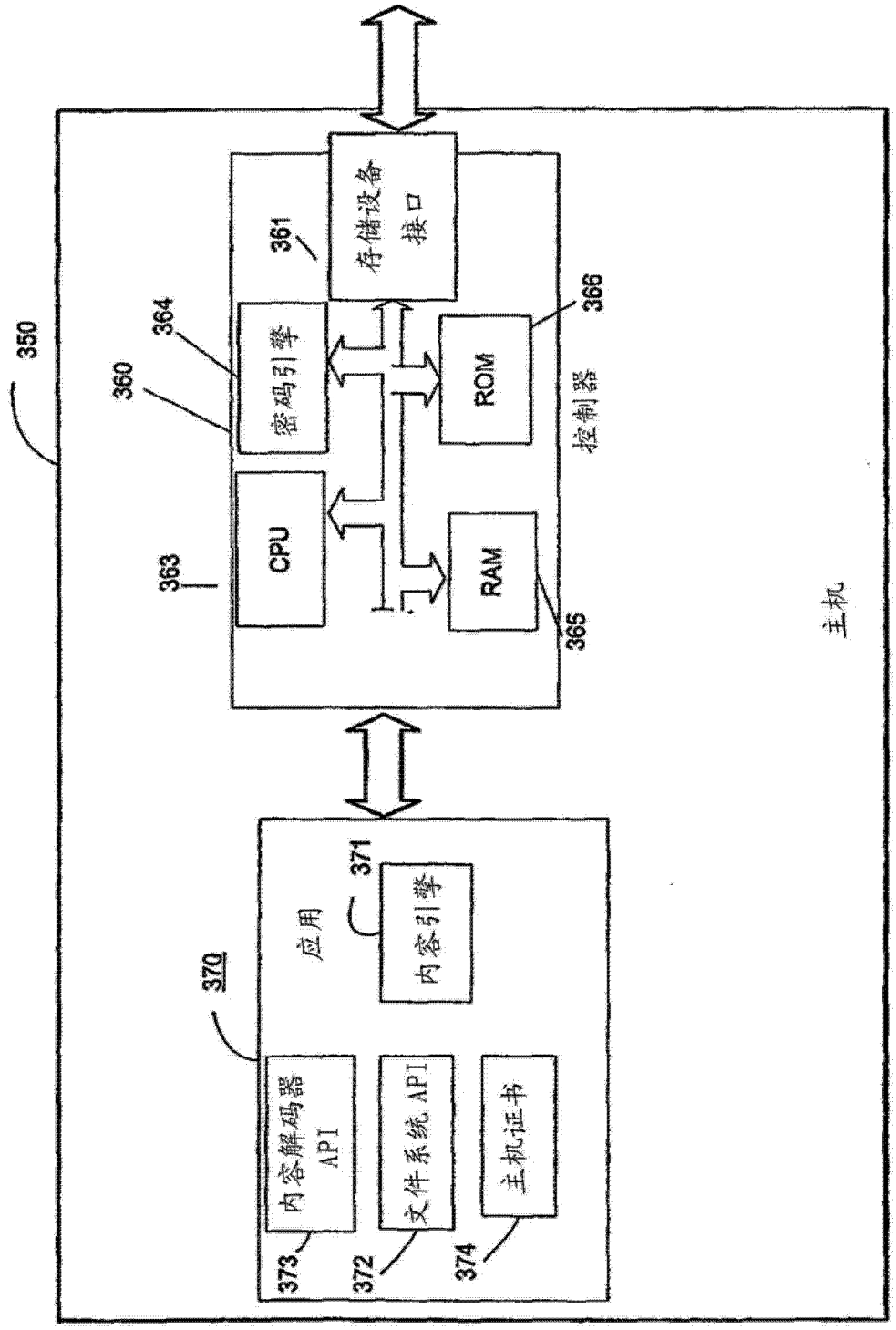

350
主机
361
存储设备接口
364
密码引擎
360
366
ROM
控制器
363
CPU
RAM
365
370
应用
371
内容引擎
内容解码器API
文件系统API
主机证书
373
372
374

图 3A

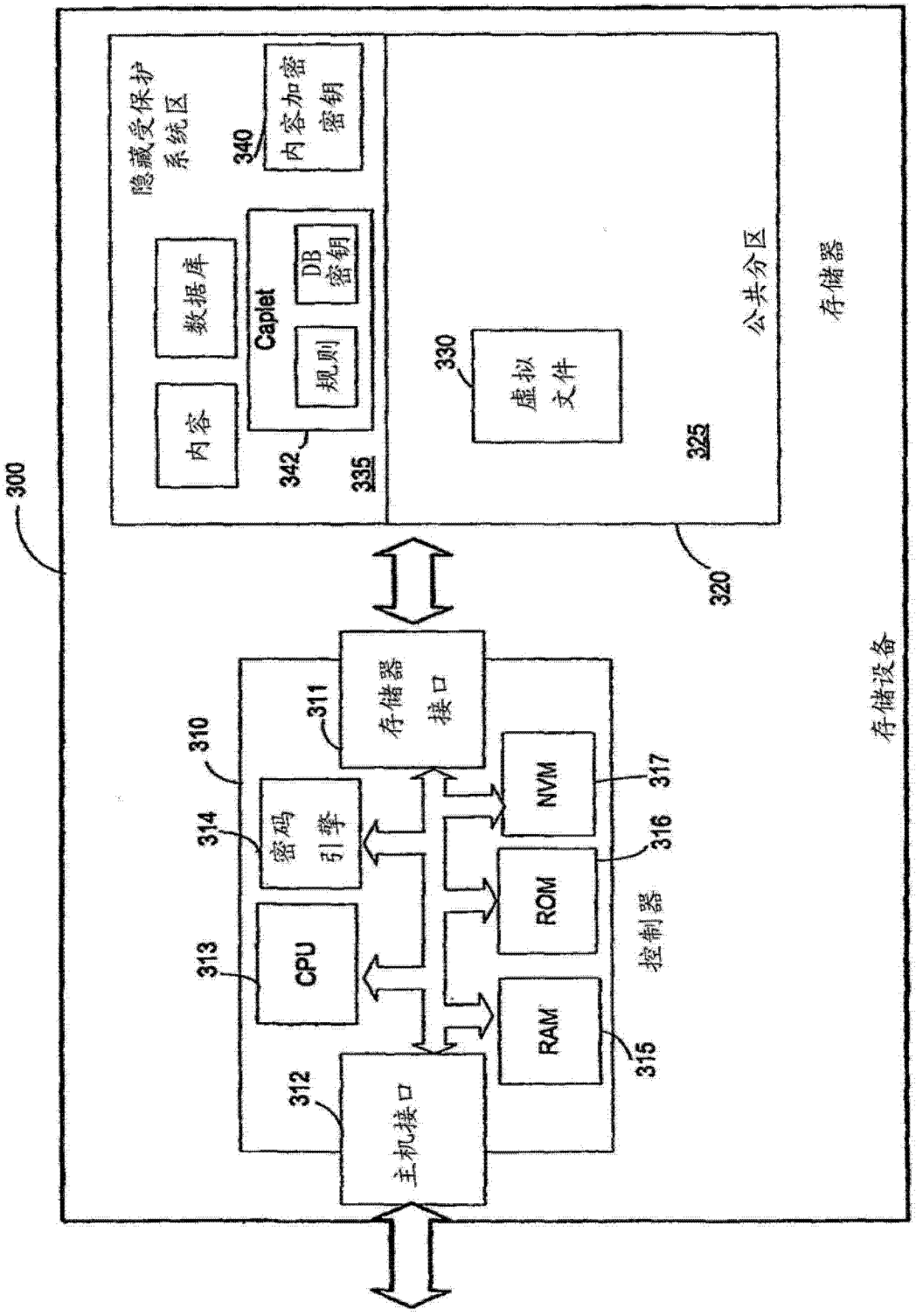

300
隐藏受保护系统区
340
内容加密密钥
数据库
Caplet
DB密钥
规则
内容
342
335
330
虚拟文件
公共分区
325
存储器
320
存储设备
存储器接口
311
310
密码引擎
314
NVM
317
316
ROM
控制器
313
CPU
RAM
315
312
主机接口

图 3B

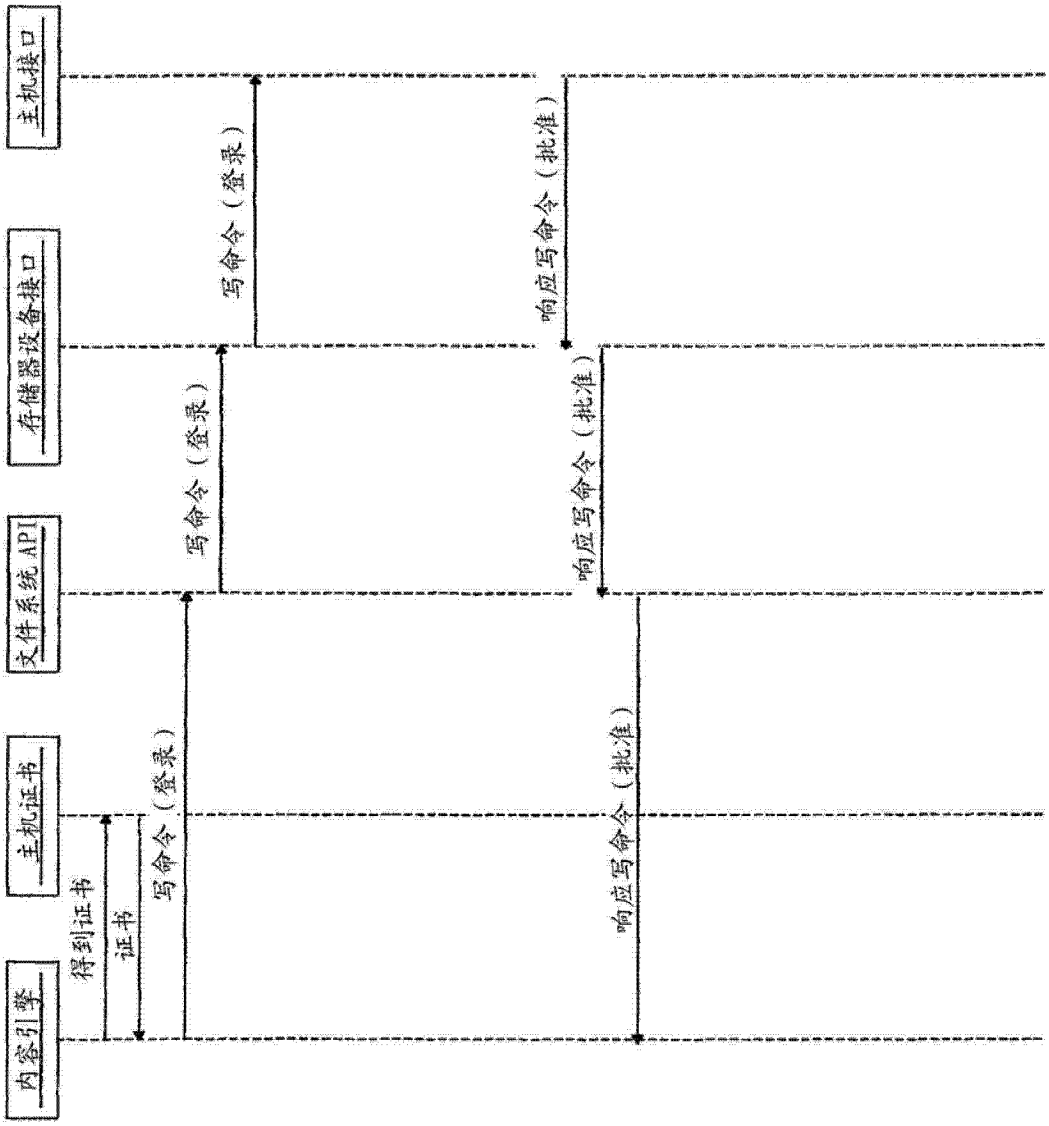

主机接口
存储器设备接口
文件系统 API
主机证书
内容引擎
写命令（登录）
响应写命令（批准）
写命令（登录）
响应写命令（批准）
写命令（登录）
响应写命令（批准）
得到证书
证书

图 4A

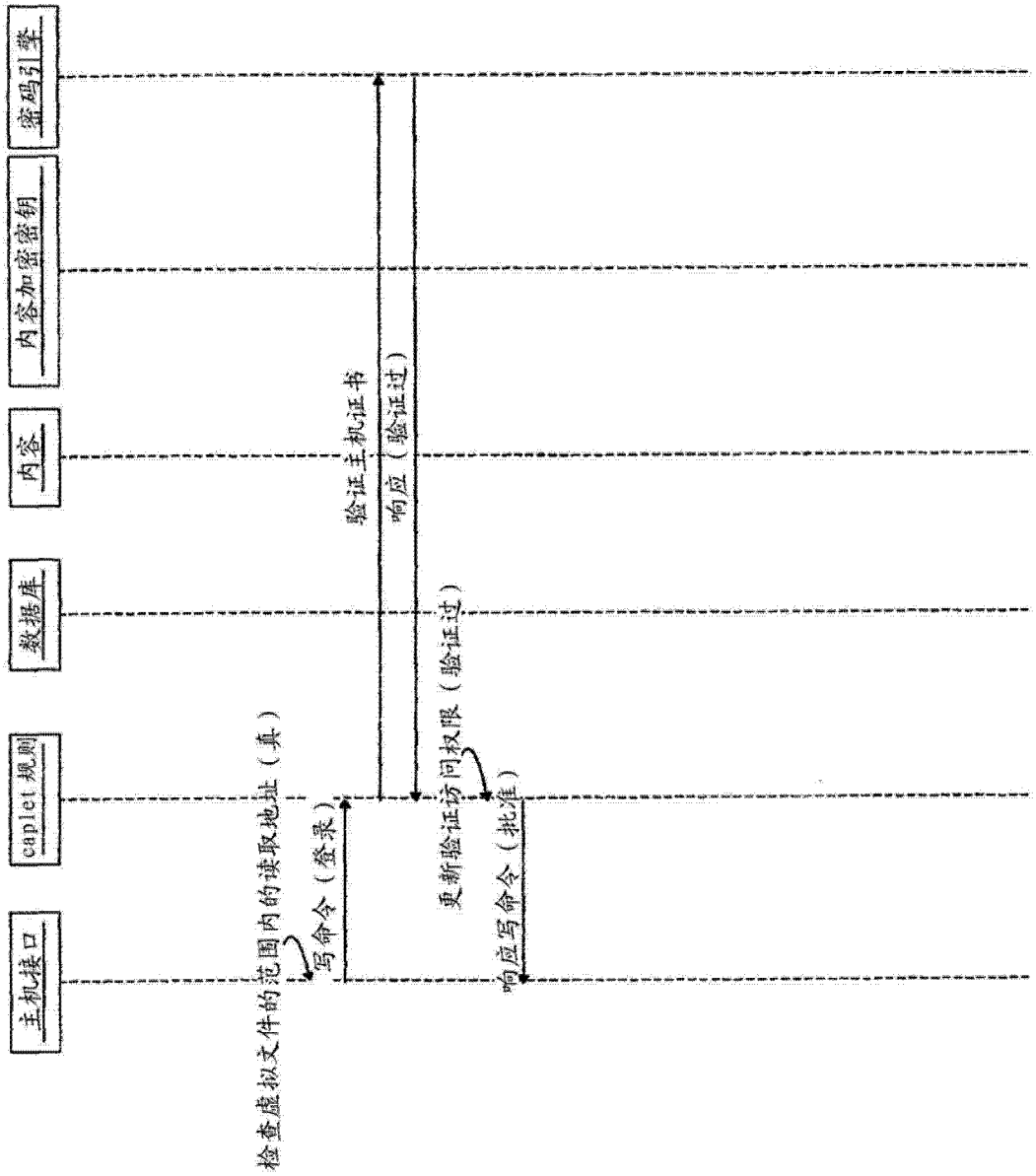

主机接口
caplet 规则
数据库
内容
内容加密密钥
密码引擎
检查虚拟文件的范围内的读取地址（真）
写命令（登录）
验证主机证书
响应（验证过）
更新验证访问权限（验证过）
响应写命令（批准）

图 4B

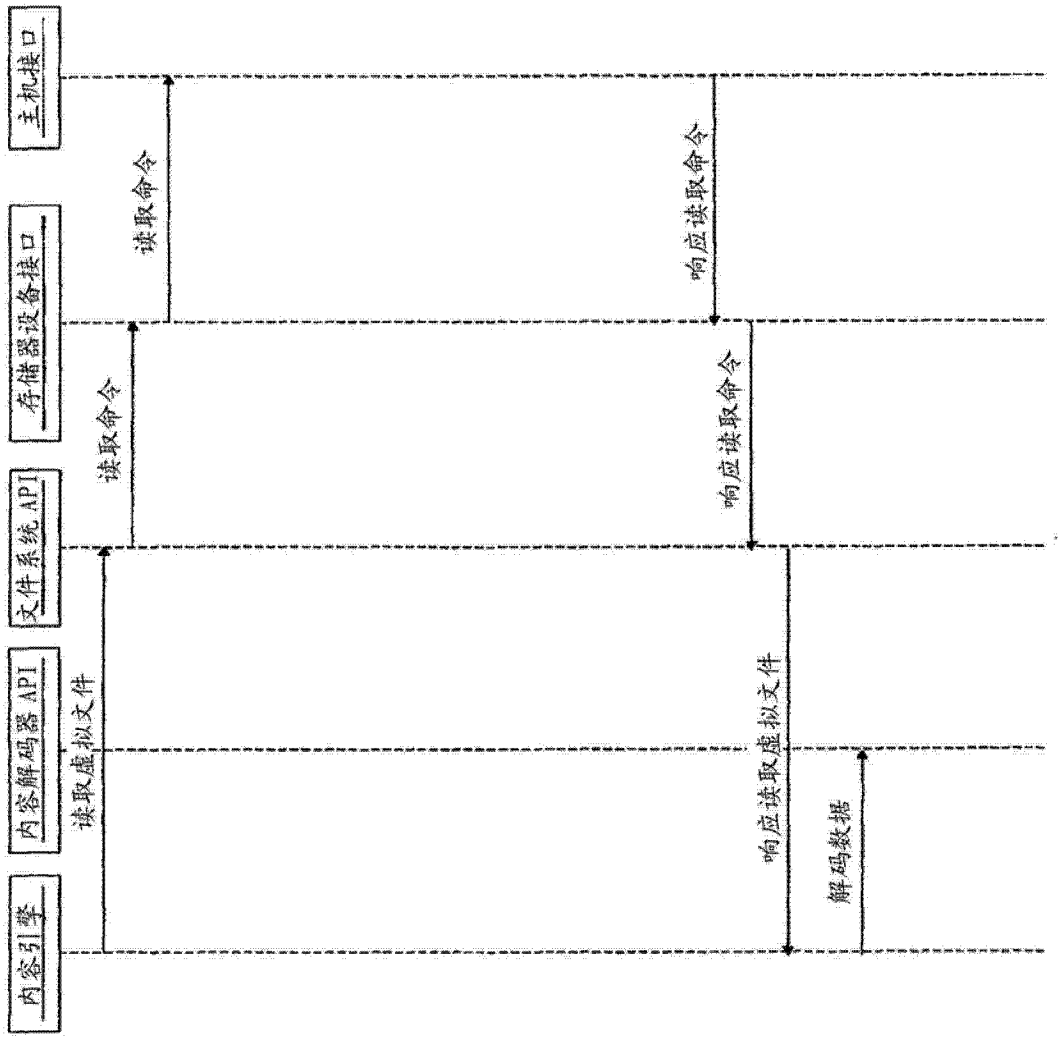

主机接口
存储器设备接口
文件系统 API
内容解码器 API
内容引擎
读取命令
响应读取命令
读取命令
响应读取命令
读取虚拟文件
响应读取虚拟文件
解码数据

图 5A

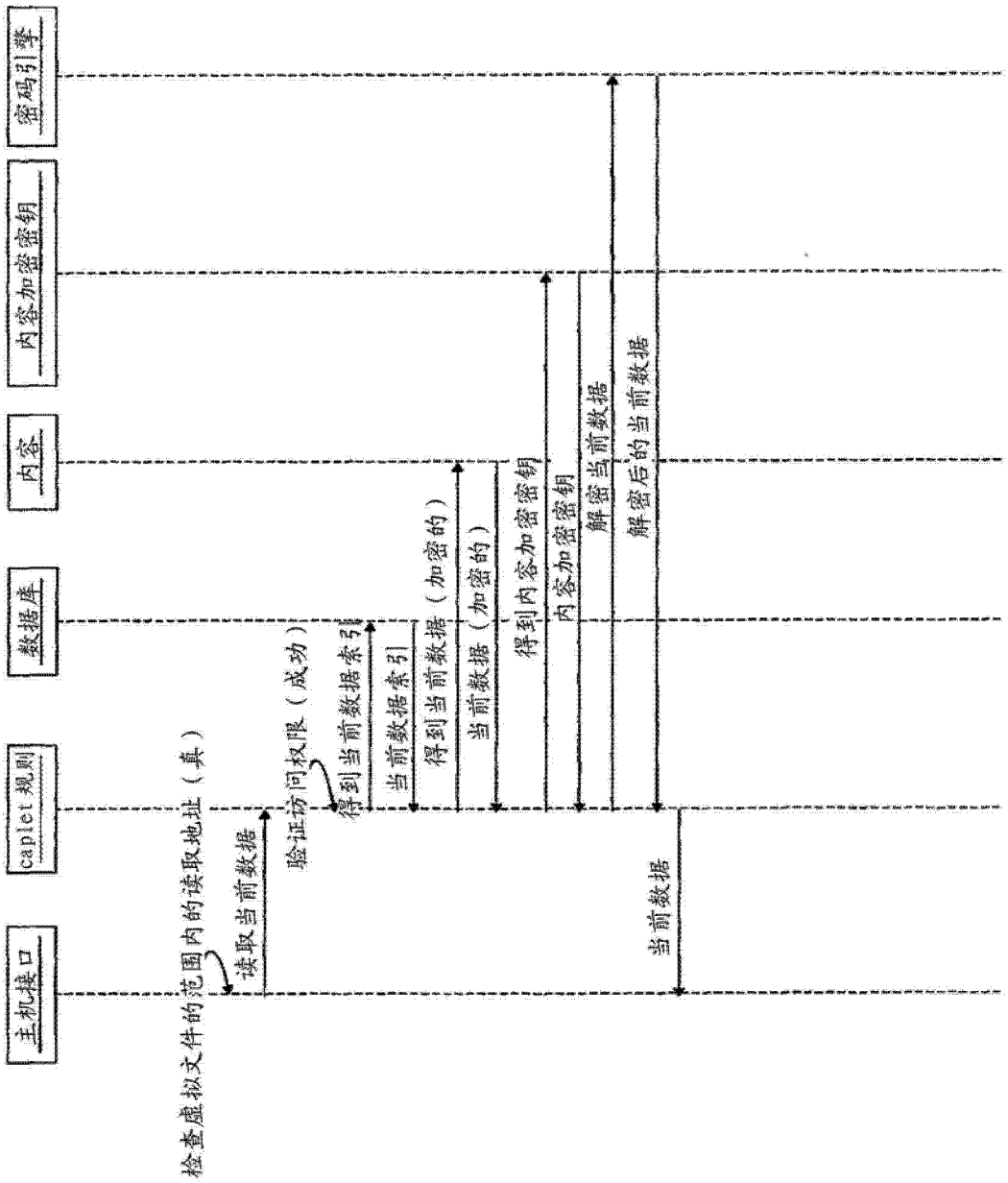

主机接口
caplet 规则
数据库
内容
内容加密密钥
密码引擎
检查虚拟文件的范围内的读取地址（真）
读取当前数据
验证访问权限（成功）
得到当前数据索引
当前数据索引
得到当前数据（加密的）
当前数据（加密的）
得到内容加密密钥
内容加密密钥
解密当前数据
解密后的当前数据
当前数据

图 5B

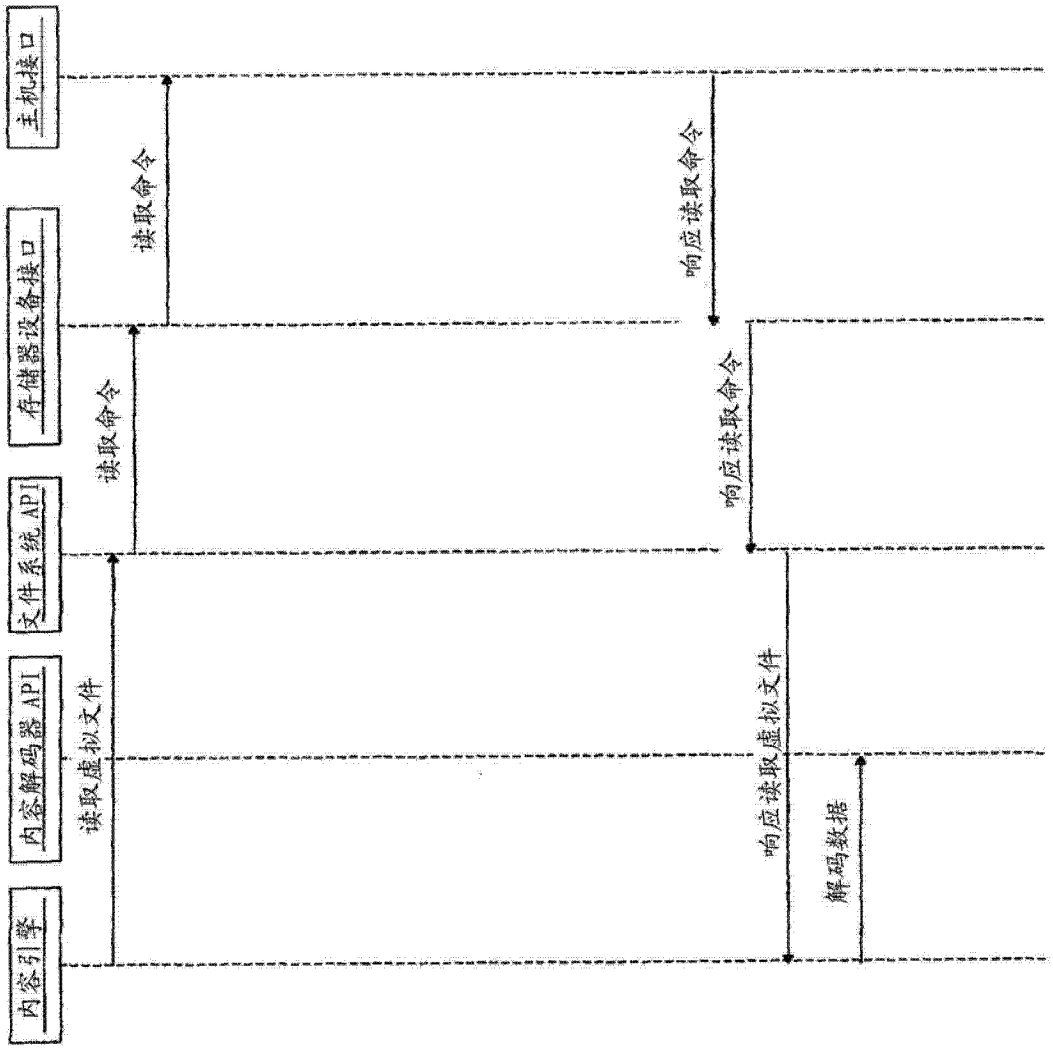

主机接口
存储器设备接口
文件系统API
内容解码器API
内容引擎
读取命令
读取命令
读取虚拟文件
响应读取命令
响应读取命令
响应读取虚拟文件
解码数据

图 6A

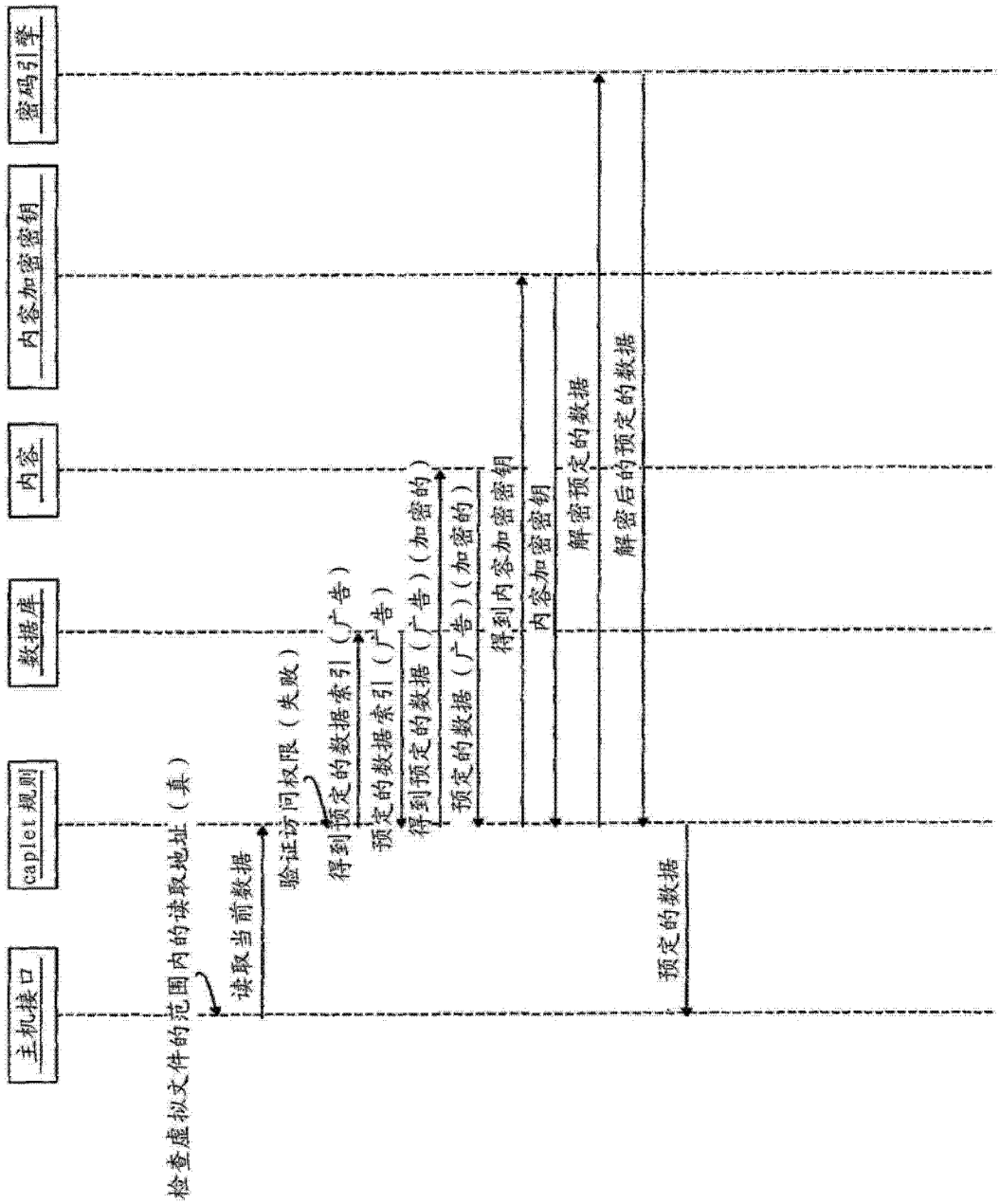

主机接口
caplet 规则
数据库
内容
内容加密密钥
密码引擎
检查虚拟文件的范围内的读取地址（真）
读取当前数据
验证访问权限（失败）
得到预定的数据索引（广告）
预定的数据索引（广告）
得到预定的数据（广告）（加密的）
预定的数据（广告）（加密的）
得到内容加密密钥
内容加密密钥
解密预定的数据
解密后的预定的数据
预定的数据

图 6B

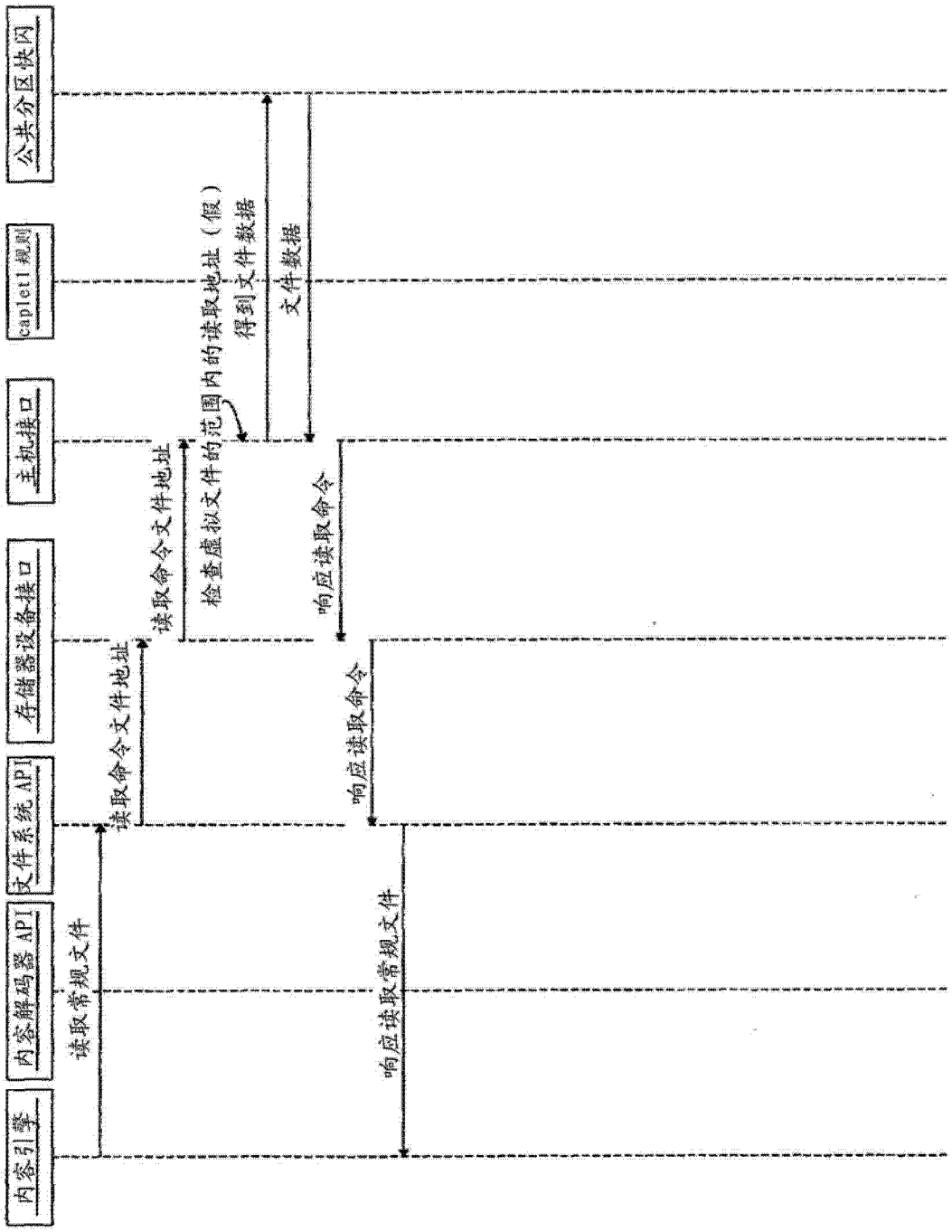

内容引擎
内容解码器 API
文件系统 API
存储器设备接口
主机接口
caplet1 规则
公共分区快闪
读取常规文件
读取命令文件地址
检查虚拟文件的范围内的读取地址（假）
得到文件数据
文件数据
响应读取命令
响应读取命令
响应读取常规文件

图 7

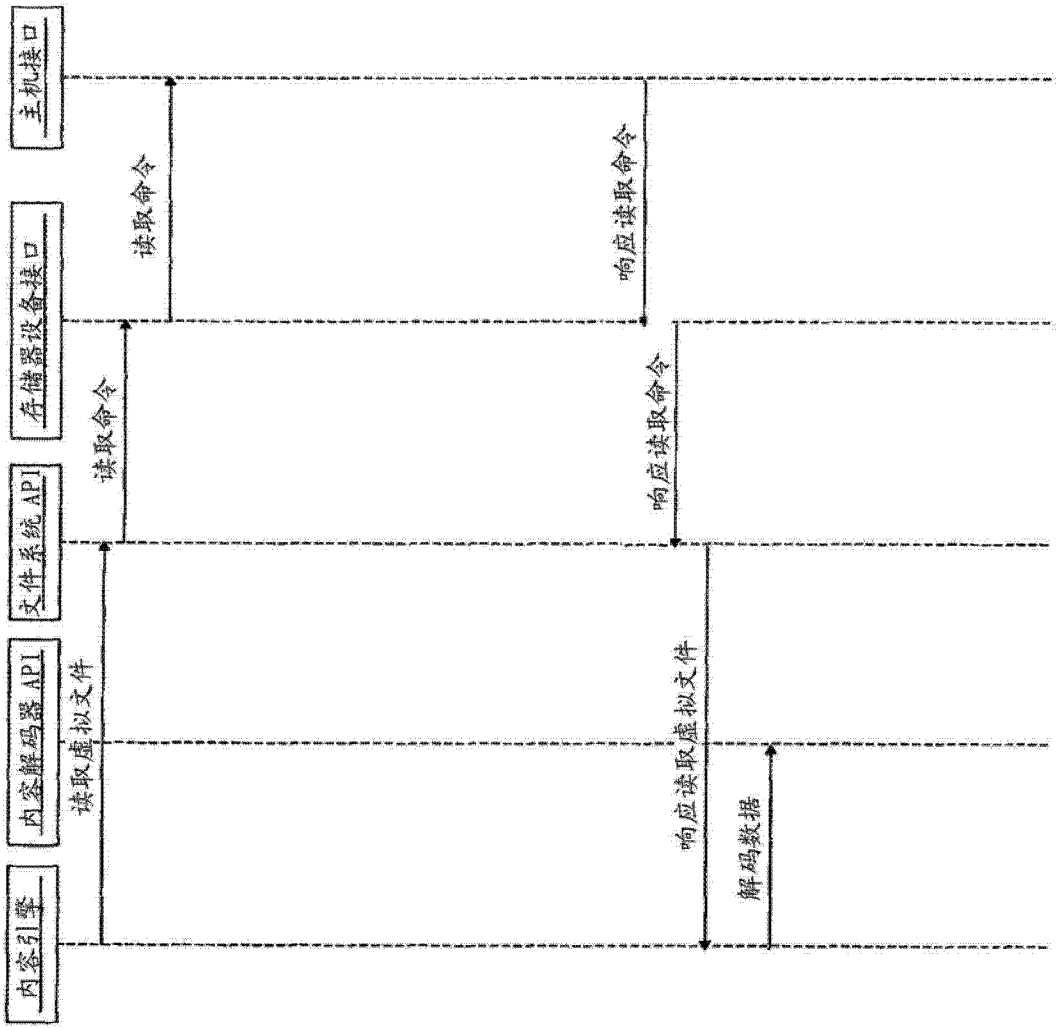

主机接口
存储器设备接口
文件系统 API
内容解码器 API
内容引擎
读取命令
读取命令
读取虚拟文件
响应读取命令
响应读取命令
响应读取虚拟文件
解码数据

图 8A

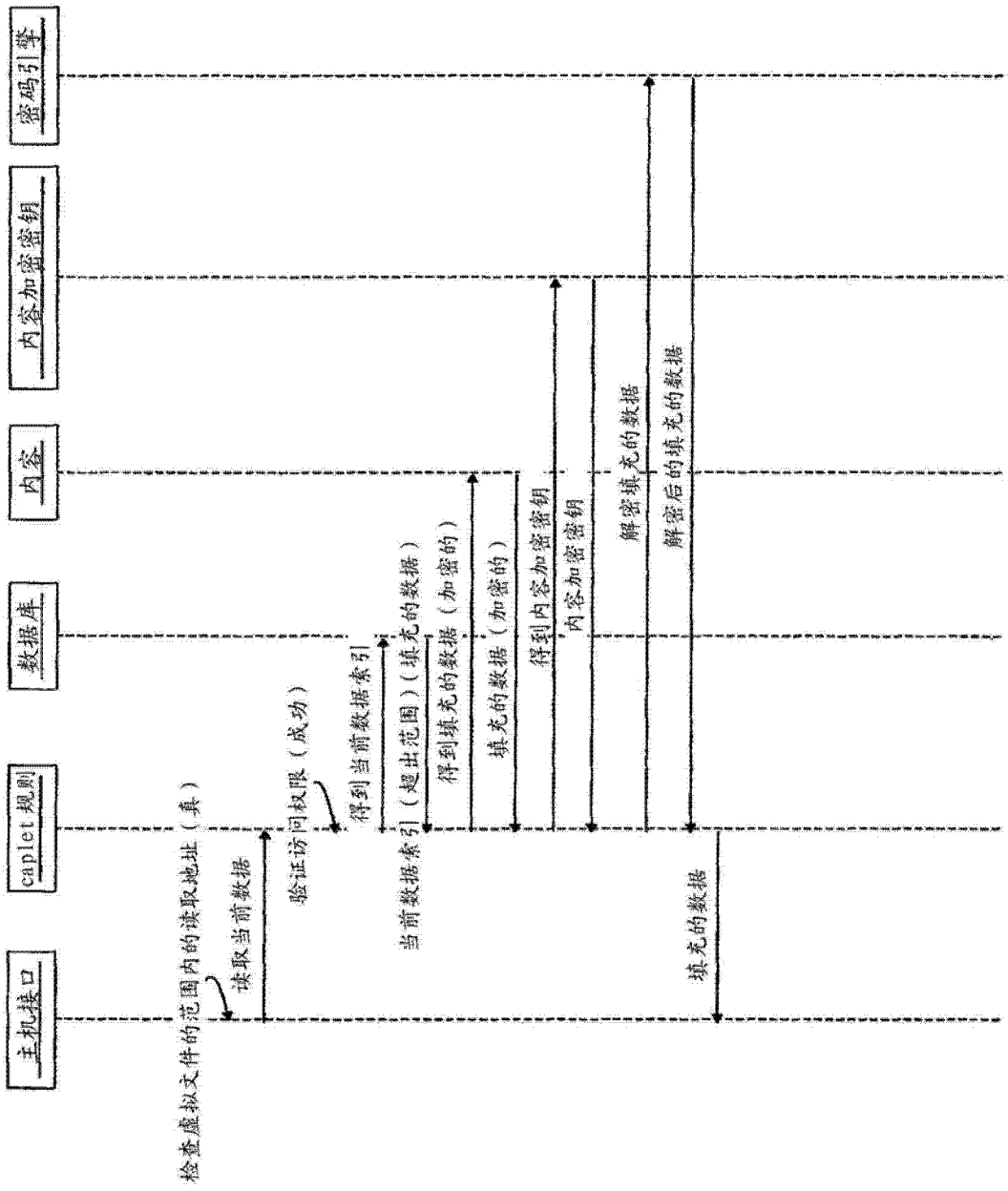

主机接口
caplet规则
数据库
内容
内容加密密钥
密码引擎
检查虚拟文件的范围内的读取地址（真）
读取当前数据
验证访问权限（成功）
得到当前数据索引
当前数据索引（超出范围）（填充的数据）
得到填充的数据（加密的）
填充的数据（加密的）
得到内容加密密钥
内容加密密钥
解密填充的数据
解密后的填充的数据
填充的数据

图 8B

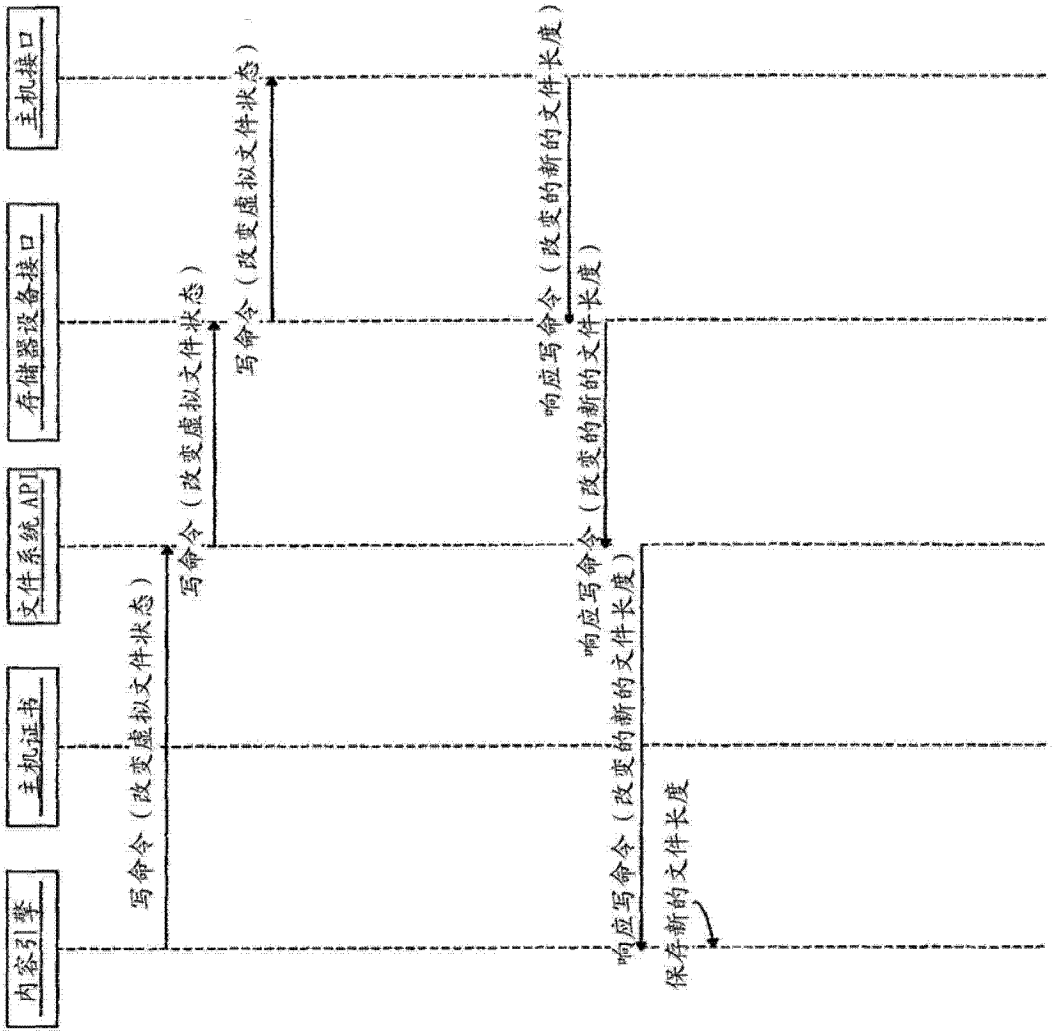

内容引擎
主机证书
文件系统 API
存储器设备接口
主机接口
写命令（改变虚拟文件状态）
写命令（改变虚拟文件状态）
写命令（改变虚拟文件状态）
响应写命令（改变的新的文件长度）
响应写命令（改变的新的文件长度）
响应写命令（改变的新的文件长度）
保存新的文件长度

图 9A

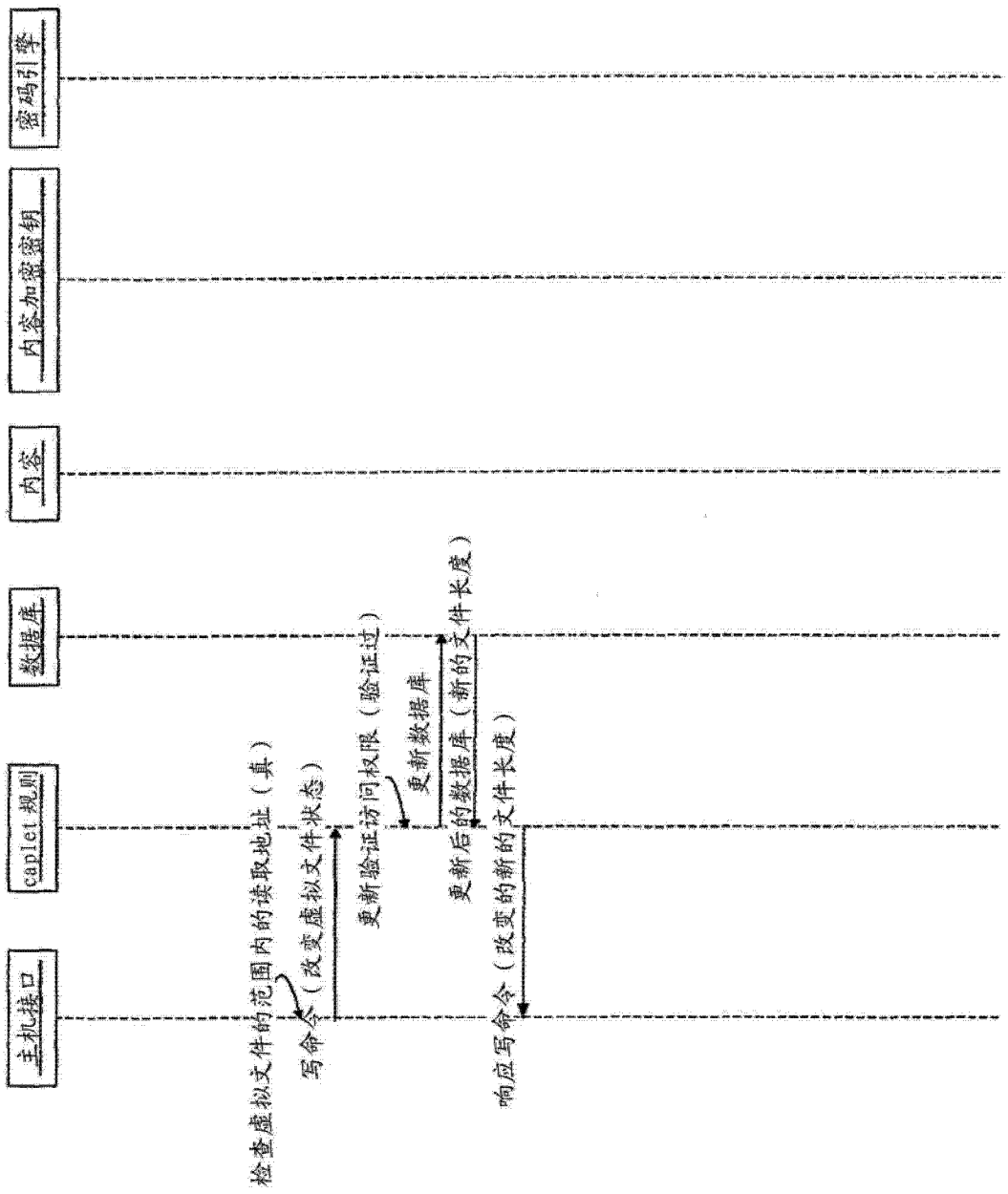

主机接口
caplet 规则
数据库
内容
内容加密密钥
密码引擎
检查虚拟文件的范围内的读取地址（真）
写命令（改变虚拟文件状态）
更新验证访问权限（验证过）
更新数据库
更新后的数据库（新的文件长度）
响应写命令（改变的新的文件长度）

图 9B

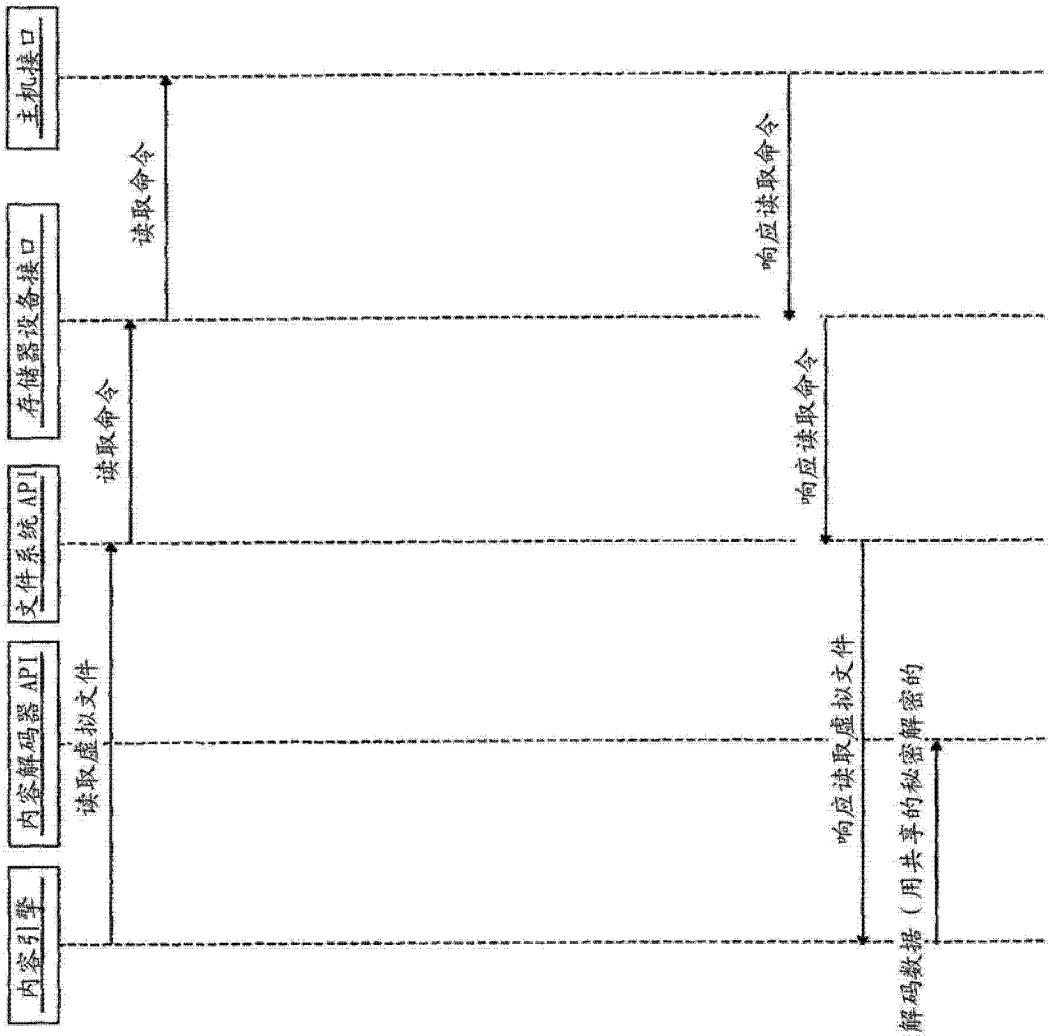

主机接口
存储器设备接口
文件系统API
内容解码器API
内容引擎
读取命令
响应读取命令
读取命令
响应读取命令
读取虚拟文件
响应读取虚拟文件
解码数据（用共享的秘密解密的

图 10A

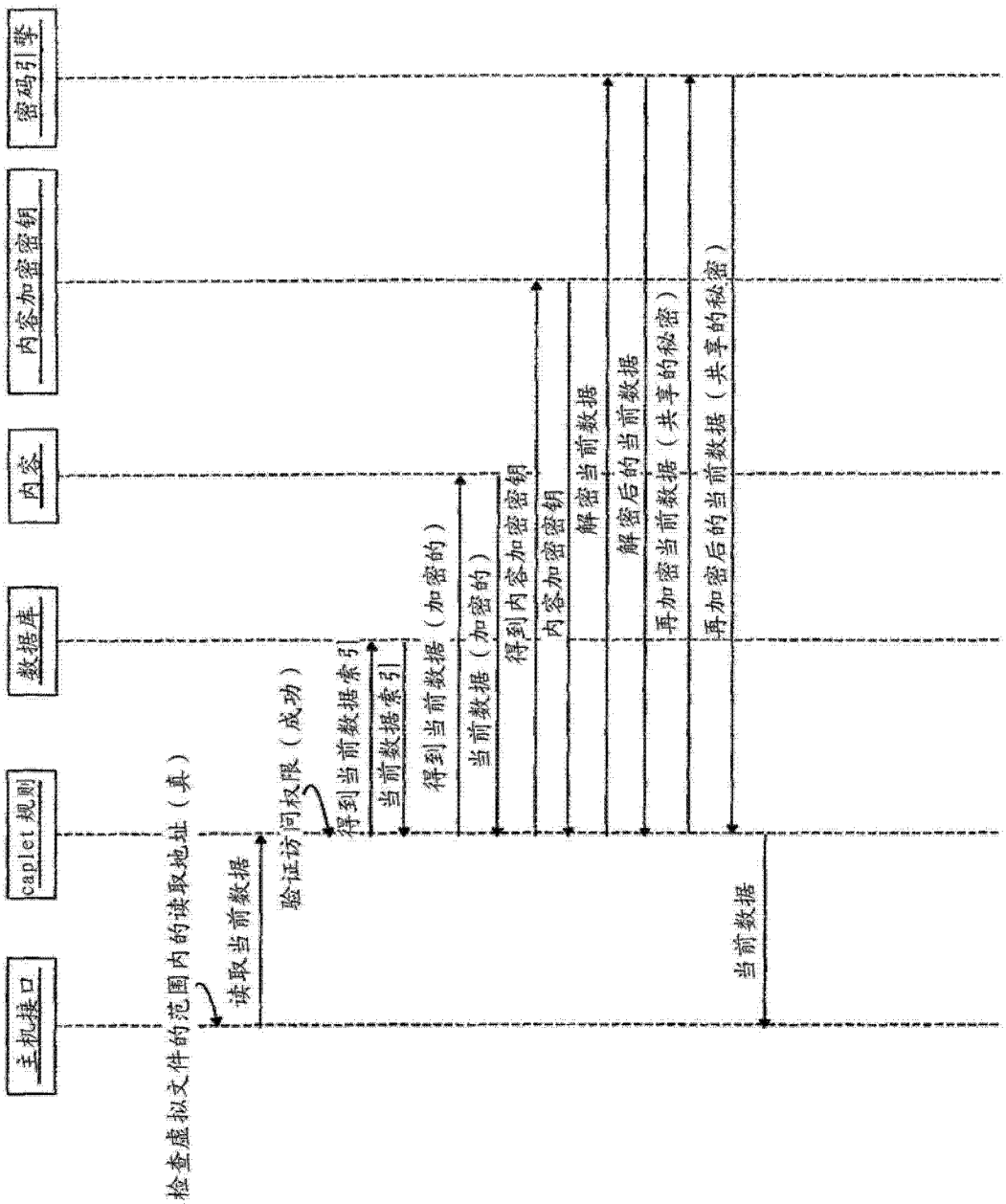

主机接口
caplet 规则
数据库
内容
内容加密密钥
密码引擎
检查虚拟文件的范围内的读取地址（真）
读取当前数据
验证访问权限（成功）
得到当前数据索引
当前数据索引
得到当前数据（加密的）
当前数据（加密的）
得到内容加密密钥
内容加密密钥
解密当前数据
解密后的当前数据
再加密当前数据（共享的秘密）
再加密后的当前数据（共享的秘密）
当前数据

图 10B

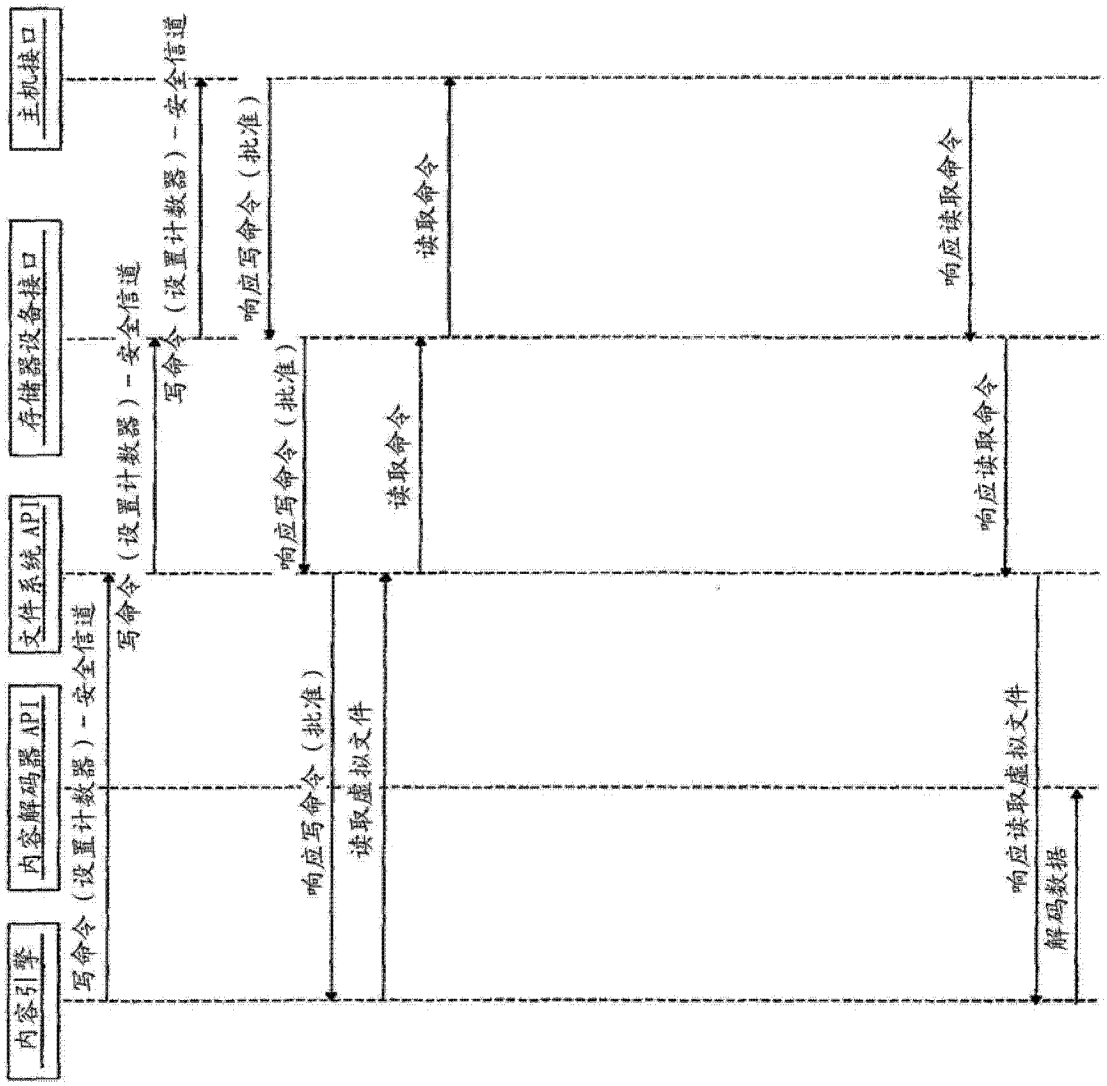

内容引擎
内容解码器 API
文件系统 API
存储器设备接口
主机接口
写命令（设置计数器）- 安全信道
写命令（设置计数器）- 安全信道
写命令（设置计数器）- 安全信道
响应写命令（批准）
响应写命令（批准）
响应写命令（批准）
读取虚拟文件
读取命令
读取命令
读取命令
响应读取命令
响应读取命令
响应读取命令
响应读取虚拟文件
解码数据

图 11A

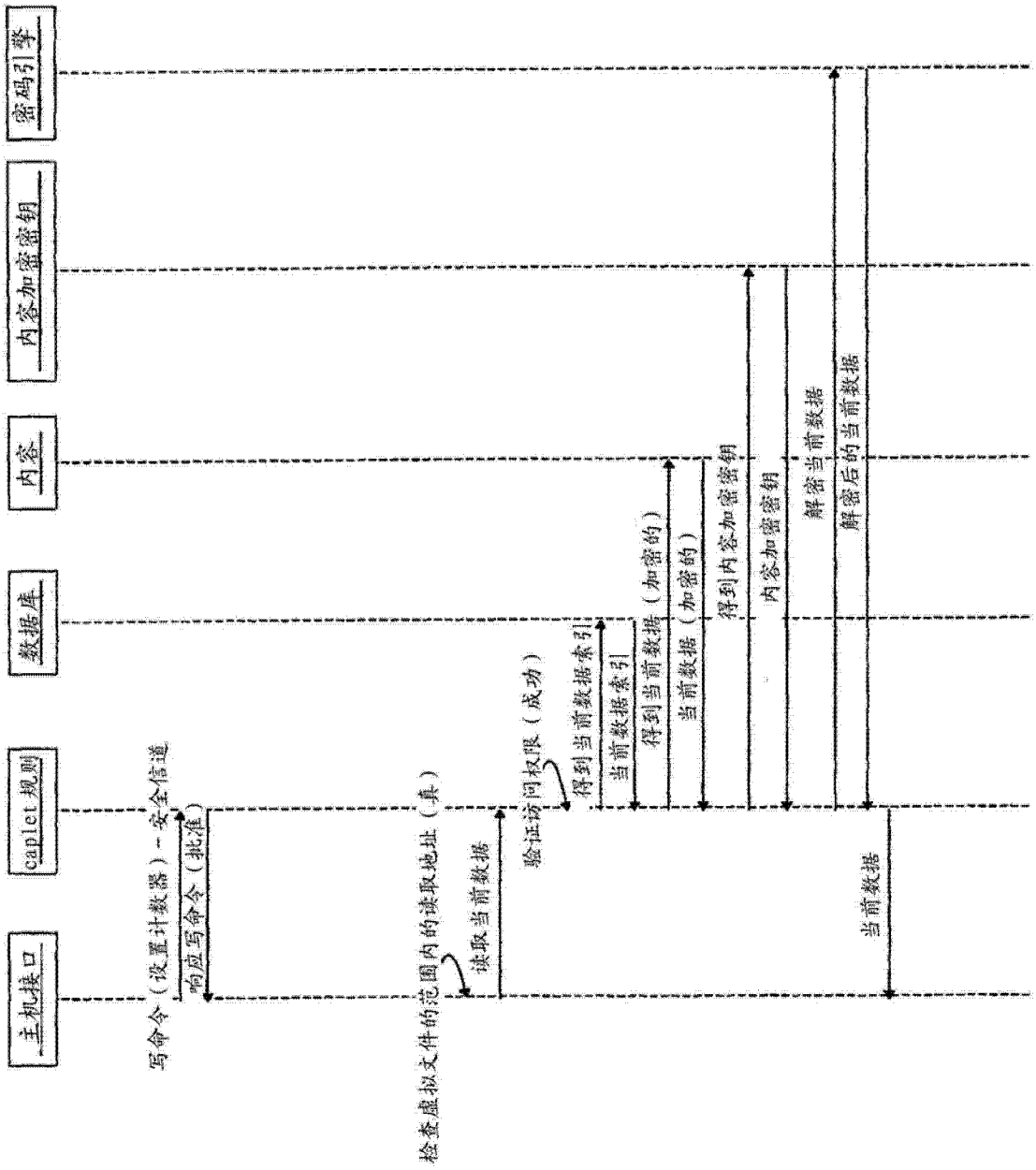

主机接口
caplet 规则
数据库
内容
内容加密密钥
密码引擎
写命令（设置计数器）- 安全信道
响应写命令（批准）
检查虚拟文件的范围内的读取地址（真）
读取当前数据
验证访问权限（成功）
得到当前数据索引
当前数据索引
得到当前数据（加密的）
当前数据（加密的）
得到内容加密密钥
内容加密密钥
解密当前数据
解密后的当前数据
当前数据

图 11B

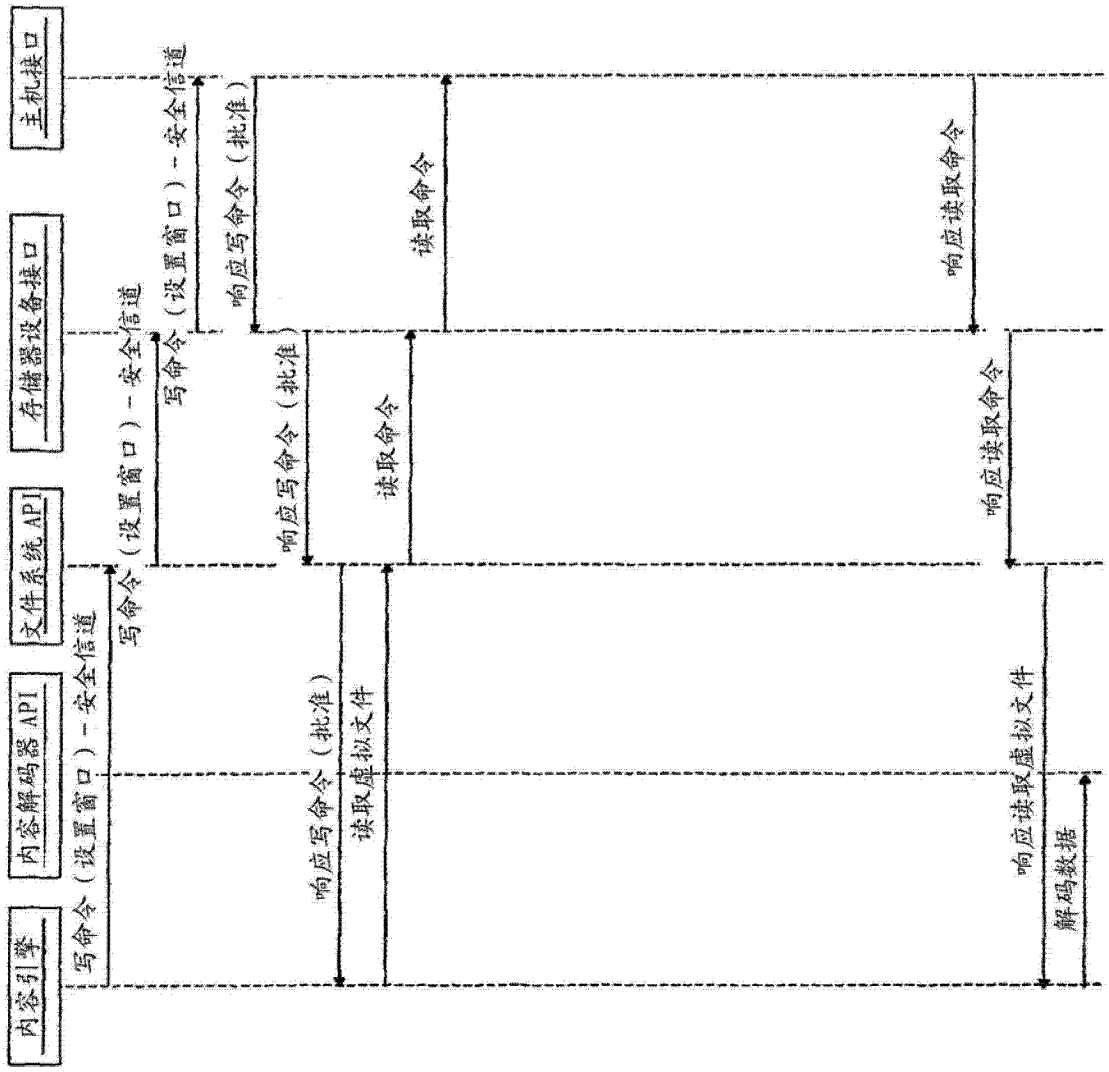

内容引擎
内容解码器API
文件系统API
存储器设备接口
主机接口
写命令（设置窗口）- 安全信道
写命令（设置窗口）- 安全信道
写命令（设置窗口）- 安全信道
响应写命令（批准）
读取虚拟文件
读取命令
响应写命令（批准）
读取命令
响应写命令（批准）
响应读取命令
响应读取命令
响应读取虚拟文件
解码数据

图 12A

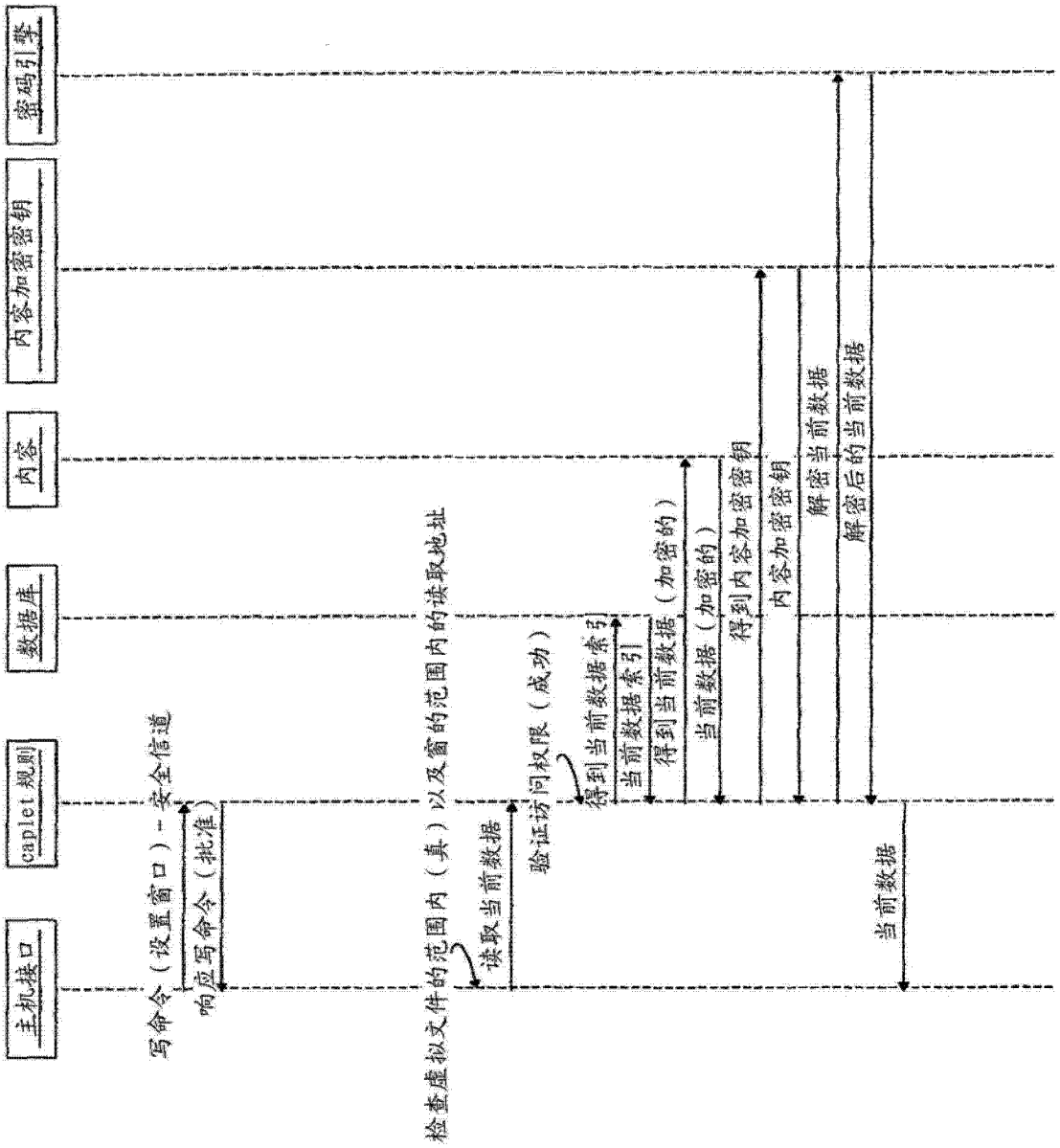

主机接口
caplet规则
数据库
内容
内容加密密钥
密码引擎
写命令（设置窗口）- 安全信道
响应写命令（批准）
检查虚拟文件的范围内（真）以及窗的范围内的读取地址
读取当前数据
验证访问权限（成功）
得到当前数据索引
当前数据索引
得到当前数据（加密的）
当前数据（加密的）
得到内容加密密钥
内容加密密钥
解密当前数据
解密后的当前数据
当前数据

图 12B

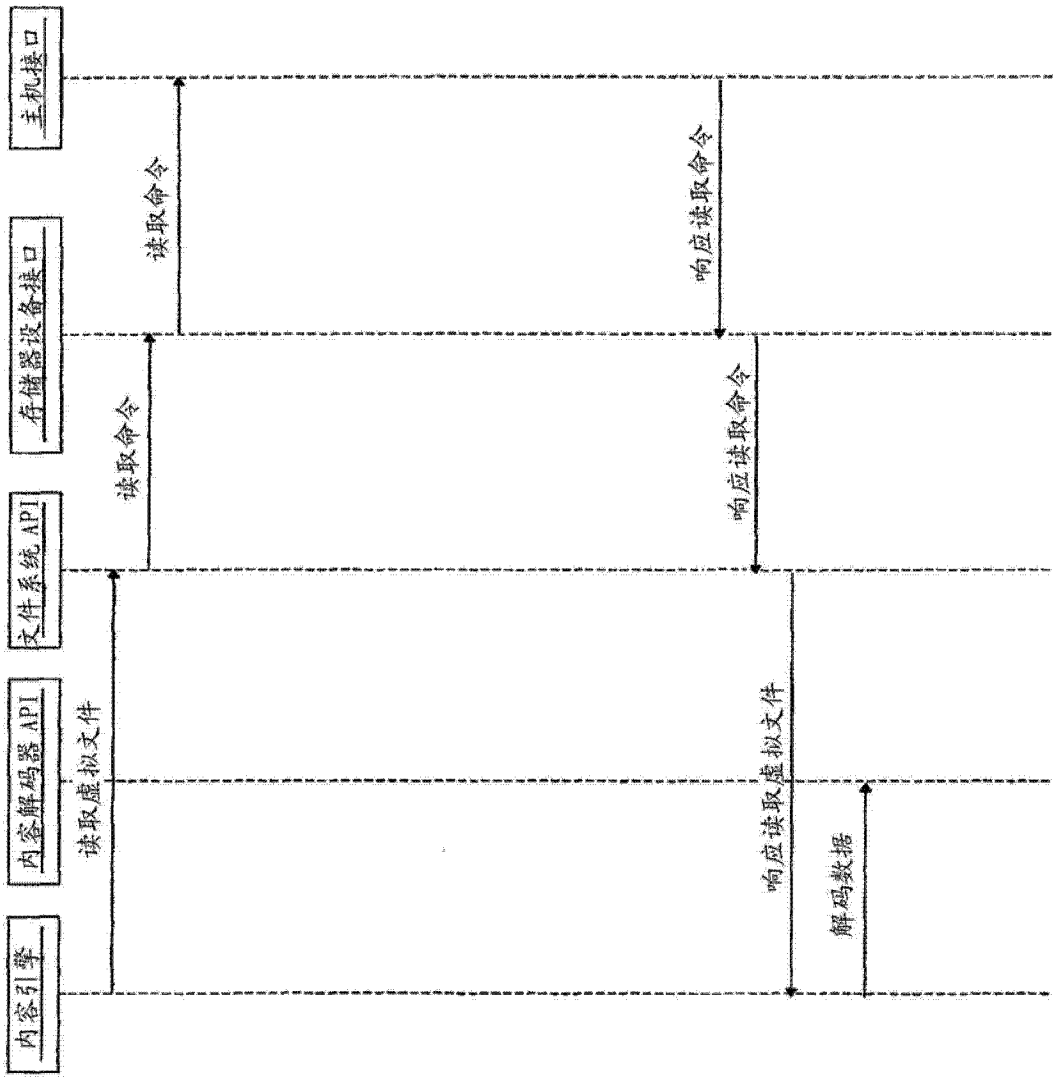

主机接口
存储器设备接口
文件系统 API
内容解码器 API
内容引擎
读取命令
读取命令
读取虚拟文件
响应读取命令
响应读取命令
响应读取虚拟文件
解码数据

图 13A

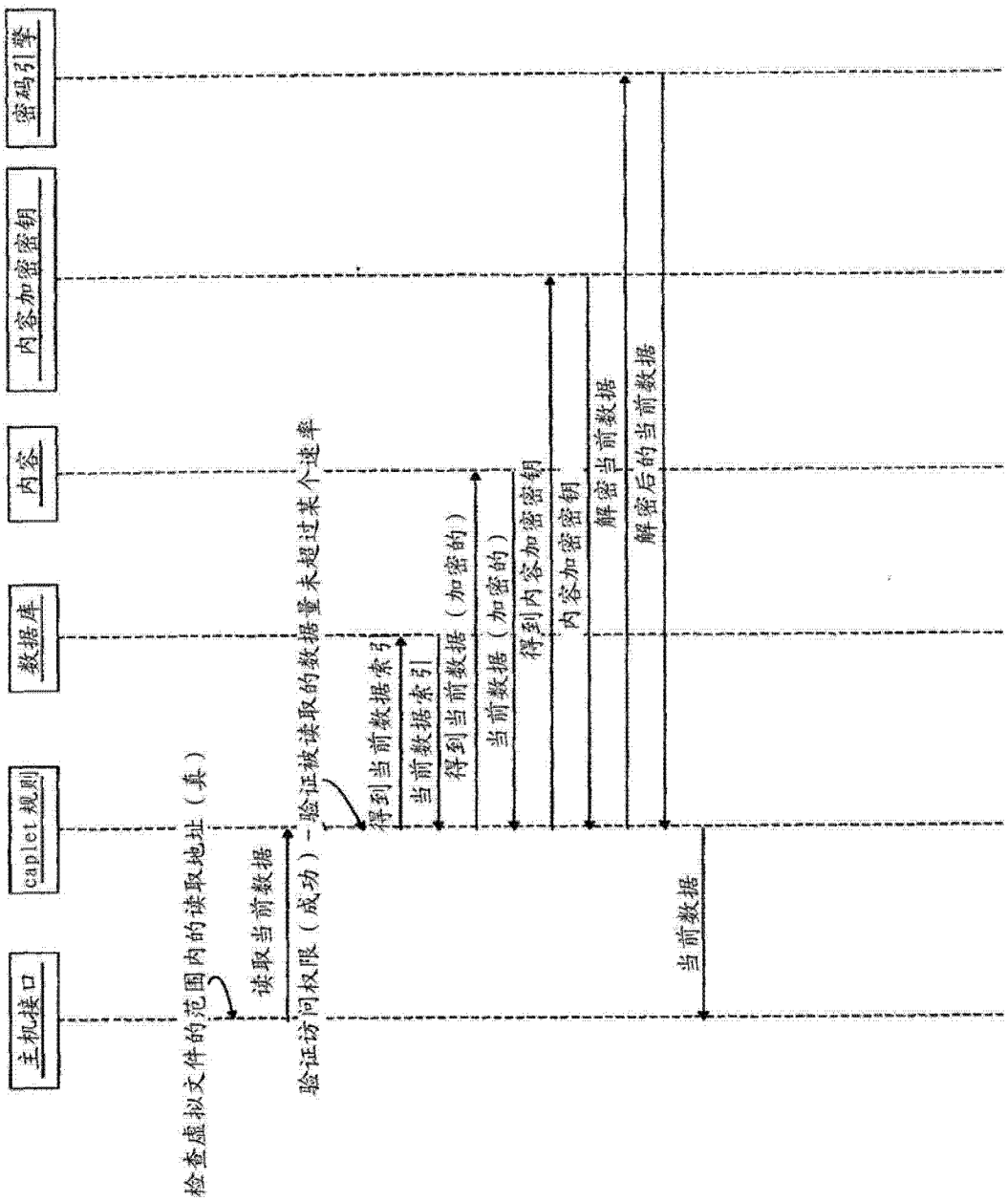

主机接口
caplet 规则
数据库
内容
内容加密密钥
密码引擎
检查虚拟文件的范围内的读取地址（真）
读取当前数据
验证访问权限（成功）
验证被读取的数据量未超过某个速率
得到当前数据索引
当前数据索引
得到当前数据（加密的）
当前数据（加密的）
得到内容加密密钥
内容加密密钥
解密当前数据
解密后的当前数据
当前数据

图 13B

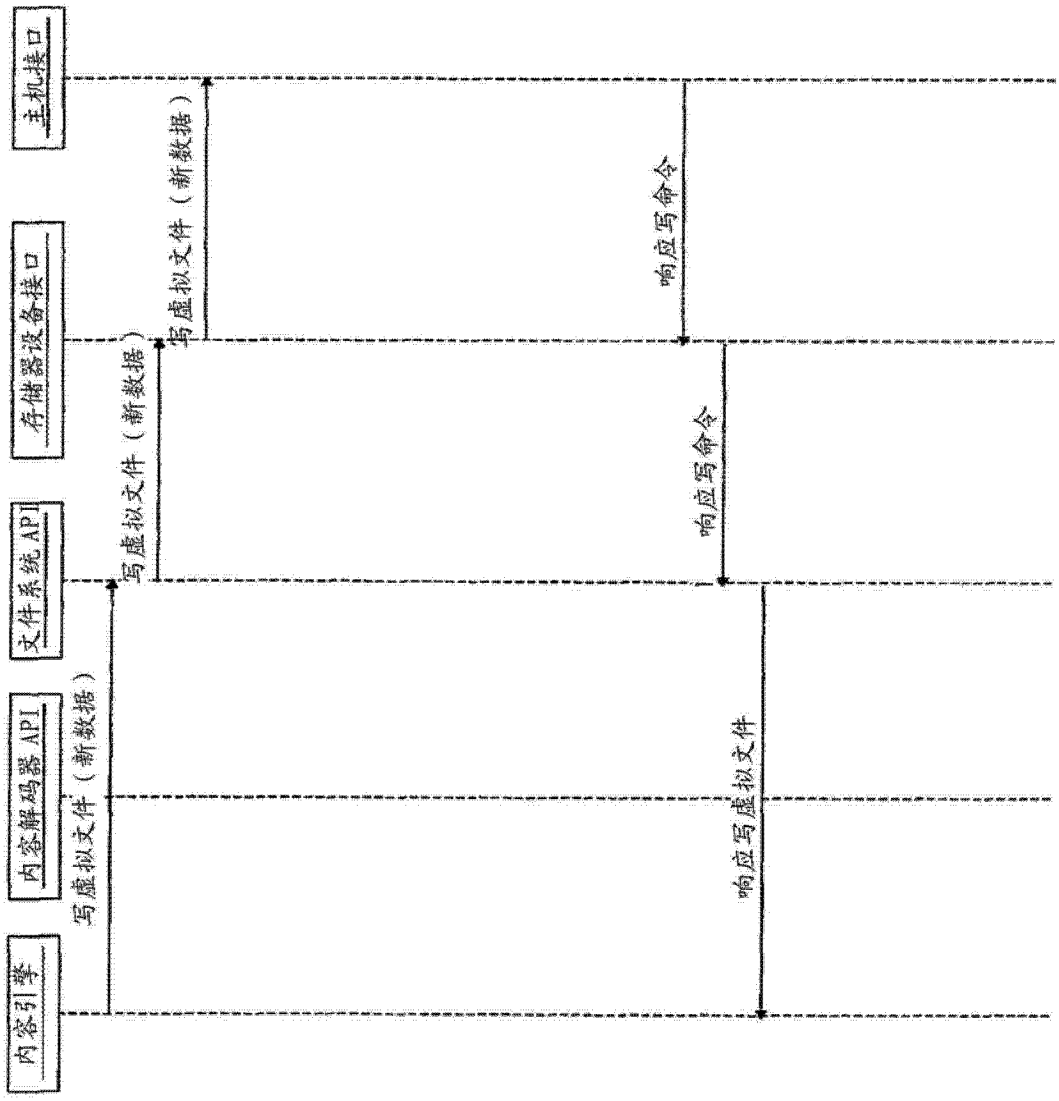

主机接口
存储器设备接口
文件系统 API
内容解码器 API
内容引擎
写虚拟文件（新数据）
写虚拟文件（新数据）
写虚拟文件（新数据）
响应写命令
响应写命令
响应写虚拟文件

图 14A

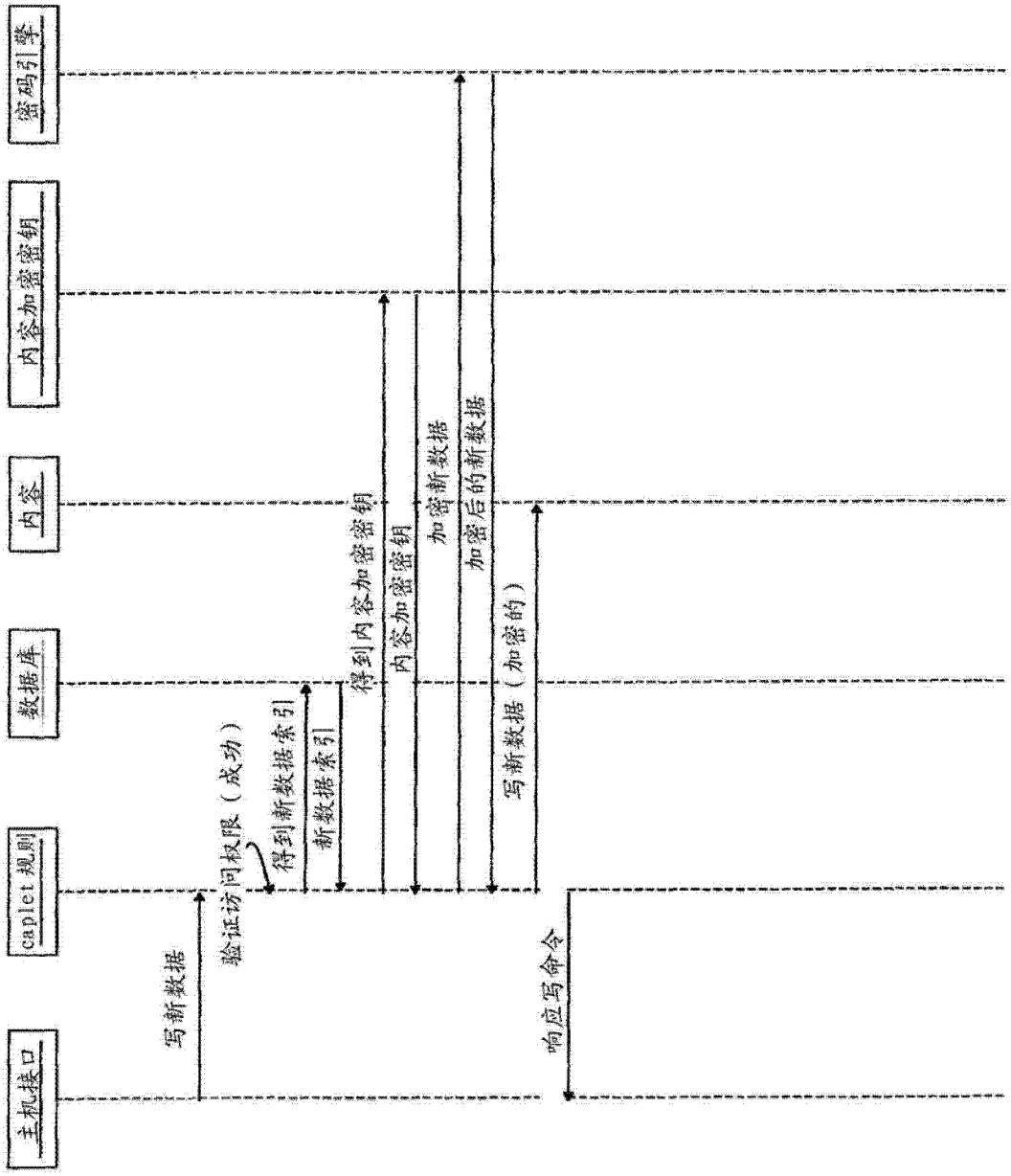

主机接口
caplet 规则
数据库
内容
内容加密密钥
密码引擎
写新数据
验证访问权限（成功）
得到新数据索引
新数据索引
得到内容加密密钥
内容加密密钥
加密新数据
加密后的新数据
写新数据（加密的）
响应写命令

图 14B