(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 118891628 A
(43)申请公布日 2024.11.01

(21)申请号 202380028428.4
(22)申请日 2023.01.25
(30)优先权数据
17/716,823 2022.04.08 US
(85)PCT国际申请进入国家阶段日
2024.09.18
(86)PCT国际申请的申请数据
PCT/US2023/011567 2023.01.25
(87)PCT国际申请的公布数据
WO2023/196045 EN 2023.10.12
(71)申请人 微软技术许可有限责任公司
地址 美国华盛顿州
(72)发明人 I•阿加瓦尔 B•D•凯利 V•索尼
(74)专利代理机构 北京市金杜律师事务所 11256
专利代理师 丁君军
(51)Int.Cl.
G06F 21/62(2006.01)
G06F 12/06(2006.01)
G06F 12/0868(2006.01)
G06F 21/64(2006.01)

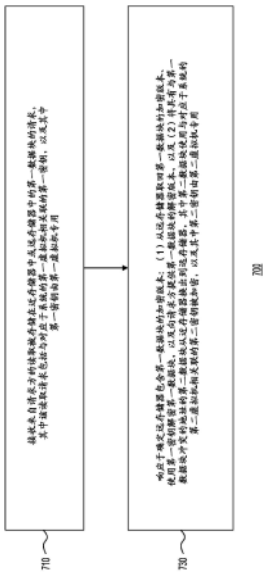
权利要求书3页 说明书15页 附图9页

(54)发明名称

与直接交换高速缓存集成的保密计算架构

(57)摘要

描述了用于与直接交换高速缓存集成的保密计算架构的系统和方法。用于管理近存储器和远存储器的示例方法包括:响应于确定远存储器包含第一数据块的加密版本,从远存储器取回第一数据块的加密版本,使用由与系统相关联的第一虚拟机专用的第一密钥解密第一数据块,以及向请求方提供第一数据块的解密版本。该方法还包括将具有与第一数据块冲突的地址的第二数据块从近存储器换出到远存储器,其中第二数据块使用由与系统相关联的第二虚拟机专用的第二密钥被加密。



CN 118891628 A

1.一种用于管理具有近存储器和远存储器的系统的方法,所述方法包括:

接收来自请求方的读取被存储在所述近存储器中或所述远存储器中的第一数据块的请求,其中所述读取请求包括与对应于所述系统的第一虚拟机相关联的第一密钥,其中所述第一密钥由所述第一虚拟机专用;以及

响应于确定所述远存储器包含所述第一数据块的加密版本:(1)从所述远存储器取回所述第一数据块的所述加密版本、使用所述第一密钥解密所述第一数据块,以及向所述请求方提供所述第一数据块的解密版本,以及(2)将具有与所述第一数据块冲突的地址的第二数据块从所述近存储器换出到所述远存储器,其中所述第二数据块使用与对应于所述系统的第二虚拟机相关联的第二密钥被加密,并且其中所述第二密钥由所述第二虚拟机专用。

2.根据权利要求1所述的方法,其中确定所述远存储器包含所述第一数据块的加密版本包括:分析与所述第一数据块相关联的元数据部分,所述元数据部分具有与所述近存储器是否包含所述第一数据块或者所述远存储器是否包含所述第一数据块相关的信息。

3.根据权利要求1所述的方法,其中所述远存储器与远存储器系统相关联,所述远存储器系统具有由至少一个物理链路分离的根端口和端点,并且其中对应于所述读取请求的通过所述至少一个物理链路的事务被加密,从而引起在通过所述至少一个物理链路的运输期间对所述第一数据块的双重加密。

4.根据权利要求1所述的方法,其中所述远存储器与远存储器系统相关联,所述远存储器系统具有由至少一个物理链路分离的根端口和端点,并且其中所述方法还包括:针对通过所述至少一个物理链路的在所述根端口与所述端点之间的事务的集合执行完整性检查。

5.根据权利要求1所述的方法,还包括分析与所述第一数据块相关联的元数据部分,所述元数据部分具有:(1)与所述近存储器是否包含所述第一数据块或者所述远存储器是否包含所述第一数据块相关的第一信息,(2)包括与被存储在所述近存储器中的所述第二数据块相关联的第一可信域标识符值的第二信息,以及(3)包括与被存储在所述远存储器中的所述第一数据块相关联的第二可信域标识符值的第三信息,并且其中所述第一可信域标识符值和所述第二可信域标识符值中的每个值由与所述近存储器相关联的近存储器控制器管理,并且所述第一可信域标识符值和所述第二可信域标识符值都不被传输给所述远存储器。

6.根据权利要求1所述的方法,还包括分析与所述第一数据块相关联的元数据部分,所述元数据部分具有:(1)与所述近存储器是否包含所述第一数据块或者所述远存储器是否包含所述第一数据块相关的第一信息,以及(2)无论所述第二数据块被存储在所述近存储器中还是所述远存储器中都包括与所述第二数据块相关联的相同可信域标识符值的第二信息。

7.根据权利要求1所述的方法,其中所述第一数据块和所述第二数据块中的每个数据块包括针对与所述系统相关联的中央处理单元(CPU)的高速缓存行。

8.一种具有近存储器和远存储器的系统,所述系统包括:

近存储器控制器,被配置为接收来自请求方的读取被存储在所述近存储器中或所述远存储器中的第一数据块的请求,其中读取请求包括与对应于所述系统的第一虚拟机相关联的第一密钥,其中所述第一密钥由所述第一虚拟机专用;以及

所述近存储器控制器还被配置为响应于确定所述远存储器包含所述第一数据块的加密版本:(1)从所述远存储器取回所述第一数据块的所述加密版本、使用所述第一密钥解密所述第一数据块,以及向所述请求方提供所述第一数据块的解密版本,以及(2)将具有与所述第一数据块冲突的地址的第二数据块从所述近存储器换出到所述远存储器,其中所述第二数据块使用与对应于所述系统的第二虚拟机相关联的第二密钥被加密,并且其中所述第二密钥由所述第二虚拟机专用。

9.根据权利要求8所述的系统,其中所述近存储器控制器还被配置为分析与所述第一数据块相关联的元数据部分,所述元数据部分具有与所述近存储器是否包含所述第一数据块或者所述远存储器是否包含所述第一数据块相关的信息。

10.根据权利要求8所述的系统,其中所述远存储器与远存储器系统相关联,所述远存储器系统具有由至少一个物理链路分离的根端口和端点,并且其中对应于所述读取请求的通过所述至少一个物理链路的事务由所述远存储器系统加密,从而引起在通过所述至少一个物理链路的运输期间对所述第一数据块的双重加密。

11.根据权利要求8所述的系统,其中所述远存储器与远存储器系统相关联,所述远存储器系统具有由至少一个物理链路分离的根端口和端点,并且其中,使用消息认证码,完整性检查针对通过所述至少一个物理链路的任何事务被执行。

12.根据权利要求8所述的系统,其中所述近存储器控制器还被配置为分析与所述第一数据块相关联的元数据部分,所述元数据部分具有:(1)与所述近存储器是否包含所述第一数据块或者所述远存储器是否包含所述第一数据块相关的第一信息,(2)包括与被存储在所述近存储器中的所述第二数据块相关联的第一可信域标识符值的第二信息,以及(3)包括与被存储在所述远存储器中的所述第一数据块相关联的第二可信域标识符值的第三信息,并且其中所述第一可信域标识符值和所述第二可信域标识符值中的每个值由所述近存储器控制器管理,并且所述第一可信域标识符值和所述第二可信域标识符值都不被传输给所述远存储器。

13.根据权利要求8所述的系统,其中所述近存储器控制器还被配置为分析与所述第一数据块相关联的元数据部分,所述元数据部分具有:(1)与所述近存储器是否包含所述第一数据块或者所述远存储器是否包含所述第一数据块相关的第一信息,以及(2)无论所述第二数据块被存储在所述近存储器中还是所述远存储器中都包括与所述第二数据块相关联的相同可信域标识符值的第二信息。

14.根据权利要求8所述的系统,其中所述系统还包括中央处理单元CPU,并且其中所述第一数据块和所述第二数据块中的每个数据块包括针对所述CPU的高速缓存行。

15.一种用于管理具有近存储器和远存储器的系统的方法,其中所述远存储器与远存储器系统相关联,所述远存储器系统具有由至少一个物理链路分离的根端口和端点的,所述方法包括:

针对通过所述至少一个物理链路的在所述根端口与所述端点之间的事务的集合执行完整性检查,其中在完成所述完整性检查之前,与所述事务的集合相关联的数据被释放以用于由所述系统进一步处理;

接收来自请求方的读取被存储在所述近存储器中或所述远存储器中的第一数据块的请求,其中所述读取请求包括与对应于所述系统的第一虚拟机相关联的第一密钥,并且其

中所述第一密钥由所述第一虚拟机专用;以及

响应于确定所述远存储器包含所述第一数据块的加密版本:(1)从所述远存储器取回所述第一数据块的所述加密版本、使用所述第一密钥解密所述第一数据块,以及向所述请求方提供所述第一数据块的解密版本,其中与所述解密相关联的时延足以允许完成所述完整性检查,以及(2)将具有与所述第一数据块冲突的地址的第二数据块从所述近存储器换出到所述远存储器,其中所述第二数据块使用与对应于所述系统的第二虚拟机相关联的第二密钥被加密,并且其中所述第二密钥由所述第二虚拟机专用。

与直接交换高速缓存集成的保密计算架构

背景技术

[0001] 多个用户或租户可以共享系统,包括计算系统和通信系统。计算系统可以包括公共云、私有云或具有公共和私有部分两者的混合云。公共云包括执行各种功能的全球服务器网络,包括存储和管理数据、运行应用以及递送内容或服务,诸如流媒体视频、供应电子邮件、提供办公生产力软件或处理社交媒体。服务器和其他组件可以位于世界各地的数据中心。虽然公共云通过互联网向公众提供服务,但企业可以使用私有云或混合云。私有云和混合云两者还包括容纳于数据中心中的服务器的网络。

[0002] 多个租户可以使用与云中的服务器相关联的计算、存储和网络化资源。可以使用安装在数据中心中的计算节点(例如服务器)上的主机操作系统(OS)来供应计算、存储和网络化资源。每个主机OS可以允许多个虚拟机访问与相应计算节点相关联的计算和存储器资源。由于通过由主机OS支持的虚拟机对存储器资源的不均匀使用,存储器资源的量可能无法被高效地分配。作为示例,大量存储器可能未被主机服务器利用。

[0003] 共享存储器的供应可以缓解这些问题中的一些问题。然而,共享存储器在由附加的物理链路(和控制器)与CPU分离时可能针对租户创建附加的安全挑战。

发明内容

[0004] 在一个方面中,本公开涉及用于管理具有近存储器和远存储器的系统的方法。该方法可以包括接收来自请求方的对读取存储在近存储器中或远存储器中的第一数据块的请求,其中读取请求包括与对应于系统的第一虚拟机相关联的第一密钥,其中第一密钥由第一虚拟机专用。该方法还可以包括响应于确定远存储器包含第一数据块的加密版本:(1)从远存储器取回第一数据块的加密版本、使用第一密钥解密第一数据块,以及向请求方提供第一数据块的解密版本,以及(2)将具有与第一数据块冲突的地址的第二数据块从近存储器换出到远存储器,其中第二数据块使用与对应于系统的第二虚拟机相关联的第二密钥被加密,并且其中第二密钥由第二虚拟机专用。

[0005] 在另一方面中,本公开涉及具有近存储器和远存储器的系统。该系统可以包括近存储器控制器,其被配置为接收来自请求方的对读取存储在近存储器中或远存储器中的第一数据块的请求,其中读取请求包括与对应于系统的第一虚拟机相关联的第一密钥,其中第一密钥由第一虚拟机专用。近存储器控制器还可以被配置为响应于确定远存储器包含第一数据块的加密版本:(1)从远存储器取回第一数据块的加密版本、使用第一密钥解密第一数据块,以及向请求方提供第一数据块的解密版本,以及(2)将具有与第一数据块冲突的地址的第二数据块从近存储器换出到远存储器,其中第二数据块使用与对应于系统的第二虚拟机相关联的第二密钥被加密,并且其中第二密钥由第二虚拟机专用。

[0006] 在又一方面中,本公开涉及用于管理具有近存储器和远存储器的系统的方法,其中远存储器与具有由至少一个物理链路分离的根端口和端点的远存储器系统相关联。该方法可以包括针对通过至少一个物理链路的在根端口和端点之间的事务的集合执行完整性检查,其中在完成完整性检查之前,释放与该事务的集合相关联的数据以用于由系统进一

步处理。该方法还可以包括:接收来自请求方的对读取存储在近存储器中或远存储器中的第一数据块的请求,其中读取请求包括与对应于系统的第一虚拟机相关联的第一密钥,并且其中第一密钥由第一虚拟机专用。

[0007] 该方法还可以包括响应于确定远存储器包含第一数据块的加密版本:(1)从远存储器取回第一数据块的加密版本、使用第一密钥解密第一数据块,以及向请求方提供第一数据块的解密版本,其中与解密相关联的时延足以允许完成完整性检查,以及(2)将具有与第一数据块冲突的地址的第二数据块从近存储器换出到远存储器,其中第二数据块使用与对应于系统的第二虚拟机相关联的第二密钥加密,并且其中第二密钥由第二虚拟机专用。

[0008] 本发明内容被提供来以简化形式介绍下文在具体实施方式中进一步描述的一系列概念。本发明内容不旨在于标识所要求保护的主题内容的关键特征或基本特征,其也不旨在于被用于限制所要求保护的主题内容的范围。

附图说明

[0009] 本公开以示例的方式示出,并且不受附图限制,其中相同的附图标记指示相似的元素。图中的元素是针对简单性和清楚性而示出的,并且不一定按比例绘制。

[0010] 图1是根据一个示例的具有近存储器和远存储器两者的系统的框图;

[0011] 图2显示了远存储器系统的框图;

[0012] 图3A和图3B显示了当存在未命中时与读取操作相关的示例事务流;

[0013] 图4是显示了当存在命中时与读取操作和写入操作相关的示例事务流的图,其中由读取操作和写入操作所针对的数据在近存储器中;

[0014] 图5显示了用于实现针对用于与直接交换高速缓存使用的保密计算架构的方法中的至少一些方法的示例系统的框图;

[0015] 图6显示了用于实现针对用于与直接交换高速缓存使用的保密计算架构的系统的数据中心;

[0016] 图7显示了用于管理具有近存储器和远存储器的存储器的示例方法的流程图;以及

[0017] 图8显示了用于管理具有近存储器和远存储器的存储器的另一示例方法的流程图。

具体实施方式

[0018] 本公开中所描述的示例涉及用于与直接交换高速缓存集成的保密计算架构的系统和方法。某些示例涉及在多租户计算系统中使用保密计算架构。多租户计算系统可以是公共云、私有云或混合云。公共云包括执行各种功能的全球服务器网络,该各种功能包括存储和管理数据、运行应用以及递送内容或服务,诸如流媒体视频、电子邮件、办公生产力软件或社交媒体。服务器和其他组件可以位于世界各地的数据中心。虽然公共云通过互联网向公众提供服务,但企业可以使用私有云或混合云。私有云和混合云两者还包括容纳于数据中心中的服务器网络。可以使用数据中心的计算和存储器资源来执行虚拟机。如本文所使用的,术语“虚拟机”涵盖但不限于任何可执行代码(以硬件、固件、软件的形式或以前述的任何组合),其实现针对无服务器计算的功能、应用、服务、微服务、容器或单内核。备选

地,虚拟机可以在与边缘计算设备、本地服务器或其他类型的系统(包括通信系统,诸如基站(例如,5G或6G基站))相关联的硬件上执行。

[0019] 与本公开的示例一致,虚拟机可以具有对近存储器(例如,本地DRAM)和远存储器(例如,池化存储器的被分配部分)的组合的访问。作为示例,数据中心中的计算节点可以被分配由池化存储器系统暴露的池化存储器,然后可以使其作为远存储器以在计算节点上运行的虚拟机是可访问的。池化存储器涉及包括由多个计算节点共享的任何物理存储器的存储器。此外,与虚拟机相关联的数据/指令可以从远存储器换入到近存储器中/从近存储器换出到远存储器。在该布置中,近存储器(例如,本地存储器)可以使用较昂贵的存储器来实现,并且远存储器(例如,池化存储器)可以使用较便宜的存储器来实现。作为示例,较昂贵的存储器可以对应于以较高数据速率操作的双倍数据速率(DDR)动态随机存取存储器(DRAM)(例如,DDR2 DRAM、DDR3 DRAM、DDR4 DRAM或DDR5 DRAM),并且较便宜的存储器可以对应于以较低数据速率操作的DRAM(例如,DRAM或DDR DRAM)。其他成本差异可以是与近存储器和远存储器相关联的可靠性或其他质量差异的功能。如本文所使用的,术语“近存储器”和“远存储器”将以相对方式被看待。因此,近存储器包括被用于存储从(多个)系统级高速缓存逐出的任何数据或指令的任何存储器,并且远存储器包括被用于存储从近存储器换出的任何数据或指令的任何存储器。近存储器和远存储器之间的另一区别涉及CPU和存储器之间的物理链路的相对数目。作为示例,假设近存储器经由近存储器控制器耦合,因此远离CPU至少一个物理链路,远存储器被耦合到远存储器控制器,其远离CPU至少一个物理链路。

[0020] 在某些环境中,云计算服务(包括虚拟机)的客户可能不完全信任由通过云计算提供方部署的管理程序提供的安全性,或者可能更喜欢具有附加的信任层。使用虚拟机,作为示例,在这种环境中,客户可能更喜欢使用他们自己的密钥来加密和解密数据以用于存储或从存储器中取回。在仅具有附接到执行虚拟机的处理器的本地存储器的传统计算系统中,可以维持跨存储器路径的加密和解密。然而,在包括近存储器(例如,本地存储器)和远存储器两者的系统中,这可能很困难,因为在处理器和存储器系统之间存在附加的物理链路。附加的物理链路可以创建附加的攻击面,从而使得恶意行为者能够窥探跨这种物理链路流动的数据(包括加密数据和明文数据两者)。解决这些问题的一些方面的一种可能的解决方法将依靠远存储器系统(例如,基于计算表达链路的规范)来加密和解密跨处理器与近存储器系统和远存储器系统之间的物理链路的数据。然而,这种加密和解密可能关于跨远存储器系统的物理链路的存储器事务而引入显著的时延。此外,考虑到针对边带信道攻击和其他入侵的可能性,当加密和解密密钥由远存储器系统处理时,很难确保它们的安全性。此外,虽然由远存储器系统的这种加密和解密可以保护在与远存储器系统相关联的物理链路上运输中的数据,但静态数据(存储在存储器中)不受保护。此外,可能也未供应加密数据关于静态数据的完整性检查。本公开中所描述的某些示例通过将安全性和完整性与直接交换高速缓存机制集成来解决保密计算问题。

[0021] 图1是根据一个示例包括与远存储器系统180耦合的计算节点110、140和170的系统100的框图。每个计算节点可以包括计算和存储器资源。作为示例,计算节点110可以包括中央处理单元(CPU)112;计算节点140可以包括CPU 142;以及计算节点170可以包括CPU 172。虽然图1中的每个计算节点都被显示为具有单个CPU,但每个计算节点可以包括附加的

CPU和其他设备,诸如图形处理器单元(GPU)、现场可编程门阵列(FPGA)、应用专用集成电路(ASIC)或其他设备。此外,每个计算节点可以包括近存储器,其可以被组织为存储器模块。作为示例,计算节点110可以包括以存储器模块122、124、126、128、130和132的形式的近存储器。计算节点140可以包括以存储器模块152、154、156、158、160和162的形式的近存储器。计算节点170可以包括以存储器模块182、184、186、188、190和192的形式的近存储器。这种存储器模块的示例包括但不限于双列直插式存储器模块(DIMM)或单列直插式存储器模块(SIMM)。被包括在这些模块中的存储器可以是动态随机存取存储器(DRAM)、闪存存储器、静态随机存取存储器(SRAM)、相变存储器、磁性随机存取存储器或任何其他类型的存储器技术,这些存储器技术可以允许存储器充当近存储器。

[0022] 继续参考图1,每个CPU还可以包括(多个)核心(例如,(多个)处理核心)。作为示例,CPU 112可以包括(多个)核心114,CPU 142可以包括(多个)核心144,以及CPU 172可以包括(多个)核心174。每个CPU还可以包括系统级高速缓存控制器(SLCC)和相关联的高速缓存存储器(例如,系统级高速缓存(未显示))。作为示例,CPU 112可以包括系统级高速缓存控制器(SLCC)116,CPU 142可以包括系统级高速缓存控制器(SLCC)146,以及CPU 172可以包括系统级高速缓存控制器(SLCC)176。此外,每个CPU还可以包括一个或多个集成存储器控制器。作为示例,CPU 112可以包括集成存储器控制器118,CPU 142可以包括集成存储器控制器148,以及CPU 172可以包括集成存储器控制器178。如果存储器模块包括DDR DRAM,则被包括在这种节点中的集成存储器控制器可以是双倍动态速率(DDR)DRAM控制器。

[0023] 每个计算节点可以被配置为执行若干虚拟机。在该示例中,计算节点110可以具有安装在其上的主机OS114,计算节点140可以具有安装在其上的主机OS144,以及计算节点170可以具有安装在其上的主机OS174。远存储器系统180可以包括逻辑池化存储器,其可以包括若干存储器模块。虽然图1中未显示,但远存储器系统180可以包括逻辑池化存储器控制器(稍后描述)。这种存储器模块的示例包括但不限于双列直插式存储器模块(DIMM)或单列直插式存储器模块(SIMM)。被包括在这些模块中的存储器可以是动态随机存取存储器(DRAM)、闪存存储器、静态随机存取存储器(SRAM)、相变存储器、磁性随机存取存储器或任何其他类型的存储器技术,这些存储器技术可以允许存储器充当池化存储器。

[0024] 由计算节点中的任何计算节点(例如,计算节点110、140或170)执行的主机OS中的任何主机OS(例如,主机OS114、144或174)都可以访问作为远存储器系统180的一部分而被包括的物理存储器的至少一部分。每个主机OS可以支持特定数目的虚拟机。作为示例,主机OS114可以支持虚拟机(VM)115、117和119,主机OS144可以支持虚拟机(VM)145、147和149,以及主机OS154可以支持虚拟机(VM)155、157和159。远存储器系统180可以在计算节点通电时或作为分配/解除分配操作的一部分将池化存储器的一部分分配给计算节点。所分配的部分可以包括存储器的一个或多个“切片”,其中切片是指由池化存储器控制器管理的存储器的部分的任何最小粒度(例如,与切片大小对齐的存储器页或任何其他存储器块)。可以使用任何合适的切片大小,包括1GB切片、2GB切片、8GB切片或任何其他合适的切片大小。池化存储器控制器可以基于与远存储器系统180相关联的分配/撤销策略来将切片分配给计算节点或撤销将切片分配给计算节点。如之前所解释的,与主机OS相关联的数据/指令可以从远存储器换入近存储器中/从近存储器换出到远存储器。在这种布置中,近存储器(例如,本地存储器)可以使用较昂贵的存储器来实现,以及远存储器(例如,池化存储器)可以使用

较便宜的存储器来实现。

[0025] 在一个示例中,计算节点110、140和170可以是数据中心的一部分。如在本公开中所使用的,术语数据中心可以包括但不限于由云服务提供方拥有的数据中心中的一些或全部、由云服务提供方拥有和操作的数据中心中的一些或全部、由通过服务提供方的客户操作的云服务提供方拥有的数据中心的一些或全部、数据中心的任何其他组合、单个数据中心,或甚至特定数据中心中的一些集群。在一个示例中,每个集群可以包括若干相同的计算节点。因此,集群可以包括计算节点,这些计算节点包括特定数目的CPU核心和特定量的存储器。代替计算节点,还可以使用其他类型的硬件,诸如边缘计算设备、本地服务器或其他类型的系统,包括通信系统,诸如基站(例如,5G或6G基站)。虽然图1将系统100显示为具有特定数目的组件,包括以特定方式布置的计算节点和存储器组件,但系统100可以包括不同地布置的附加的或更少的组件。作为示例,存储器控制器可以不与CPU集成,并且可以位于单独的基板上。作为另一示例,远存储器系统180可以被包括作为每个计算节点的一部分,而不是如图1中所的单独系统。

[0026] 图2显示了对应于图1中显示的远存储器系统180的示例远存储器系统200的框图。远存储器系统200可以包括用于将远存储器系统控制器耦合到计算节点(例如,图1的计算节点110、130和150)的一个或多个根端口202。远存储器系统200还可以包括若干池化存储器控制器和相关联的池化存储器模块。例如,远存储器系统200可以包括耦合到(多个)根端口202的远存储器控制器(FMC)210、FMC 220、FMC 230、FMC 240、FMC 250和FMC 260,如图2中所示。FMC 210、FMC 220、FMC 230、FMC 240、FMC 250和FMC 260中的每个FMC还可以被耦合到结构管理器280。FMC 210还可以被耦合到存储器模块212、214、216和218。FMC 220还可以被耦合到存储器模块222、224、226和228。FMC 230还可以被耦合到存储器模块232、234、236和238。FMC 240还可以被耦合到存储器模块242、244、246和248。FMC 250还可以被耦合到存储器模块252、254、256和258。FMC 260还可以被耦合到存储器模块262、264、266和268。每个存储器模块可以是双列直插式存储器模块(DIMM)或单列直插式存储器模块(SIMM)。

[0027] 继续参考图2,在一个示例中,远存储器控制器中的每个远存储器控制器可以被实现为符合计算表达链路(CXL)规范的池化存储器控制器。在该示例中,与远存储器系统200相关联的存储器模块中的每个存储器模块可以被配置为Type3 CXL设备。结构管理器280可以经由总线206与数据中心控制平面290通信。在一个示例中,结构管理器280可以被实现为符合CXL规范的结构管理器。从数据中心控制平面290接收到的控制信息可以包括指定在给定时间处将来自存储器池的存储器的哪些切片被分配给任何特定计算节点的控制信息。响应于该控制信息,结构管理器280可以以时分复用的方式将来自远存储器内的存储器的切片分配给特定计算节点。换言之,在某一时间处,特定存储器切片只能被分配给特定计算节点,而不能被分配给任何其他计算节点。作为该示例的一部分,与CXL.io协议(其是基于PCIe的非相干I/O协议)相关联的事务可以被用于配置存储器设备以及被包括在远存储器系统200中的CPU与存储器模块之间的链路。CXL.io协议还可以由与各种计算节点相关联的CPU用于设备发现、枚举、错误报告和管理中。备选地,也可以使用支持这种配置事务的任何其他I/O协议。可以经由与CXL.mem协议相关联的事务来处理对存储器模块的存储器访问,该CXL.mem协议是支持存储器事务的存储器访问协议。作为示例,可以经由CXL.mem协议处理与CPU中的任何CPU相关联的加载指令和存储指令。备选地,也可以使用允许将CPU加载/

存储指令转化为与被包括在远存储器系统200中的存储器模块相关联的读/写事务的任何其他协议。

[0028] 每个远存储器控制器(例如,FMC 210、FMC 220、FMC 230、FMC 240、FMC 250和FMC 260中的任何FMC)可以维护段表,该段表指示可以以关于部分大小的任何合适粒度分配/取消分配的远存储器(例如,被实现为池化存储器)的不同部分。更一般地,远存储器控制器可以维护任何合适的表,该表表示可用/被分配的存储器切片,指示与切片有关的任何相关信息(例如,被分配/未分配的状态、指示被分配的切片被分配到哪个计算节点的所有权状态、使用信息的新近度、分配信息的新近度、与被分配的切片被分配给的计算节点有关的主机类型或其他元数据)。例如,针对2TB存储器池,可以以1GB切片粒度分配/取消分配部分,例如,段表中可以有2K(例如,2048)个段,指示不同的1GB切片。作为示例,段表中的段可以包括32位段标识符,该32位段标识符包括指示部分被分配给哪个主机的8位、指示该部分是否曾被访问过的1位值、指示用于寻址该部分中的数据的目标地址解码方案的3位解码器映射,和/或指示对该部分的最近访问的计数值的16位漏桶计数器。例如,上述段表可以包括池化存储器控制器的SRAM的8KB区域。针对段表的上述模式是非限制性的,以及段表可以包括用于跟踪存储器的分配的任何合适数据。虽然图2将远存储器系统200显示为具有特定数目的组件,包括以特定方式布置的池化存储器控制器和存储器模块,但远存储器系统200可以包括不同地布置的附加的或更少的组件。此外,可以使用多个开关。此外,结构管理器280可以与附加的或更少的池化存储器控制器共享。

[0029] 为了在图1的系统100的上下文中使用直接交换高速缓存,近存储器必须具有与远存储器的固定比率。在该示例中,假设近存储器具有与远存储器相同的大小。这意味着对近存储器中的位置的任何访问都将以直接交换高速缓存方式操作。因此,这些访问将首先在被指定为近存储器的存储器内执行查找。与非优化的直接交换高速缓存布置一致,近存储器中的任何命中都将直接从近存储器(例如,本地存储器)中得到服务,而近存储器中的未命中将引起对应的远存储器和近存储器位置之间的交换操作。交换操作(例如,将数据从远存储器中的位置交换到近存储器中的位置或将数据从近存储器中的位置换出到远存储器中的位置)可以以高速缓存行的粒度级别执行。因此,在该示例中,数据块相当于高速缓存行。然而,在该示例中,每个位置在给定时间处只能具有两个高速缓存行中的一个。另一高速缓存行存在于远存储器中。

[0030] 每个高速缓存行可以包括数据部分(例如,512位)和元数据部分(例如,128位)的组合。数据部分可以包含表示用户数据或由计算节点执行的指令的数据。元数据部分可以包括表示数据部分中的数据的各种属性的数据。元数据部分还可以包括错误检查和校正位或其他合适类型的信息。此外,元数据部分可以包括具有适当数目的位的标签,以区分高速缓存行的位置。元数据信息的单个位可以被用于区分两个高速缓存行(例如,CL$A和CL$B)的位置。单个位的使用假设针对近存储器的可交换范围和针对远存储器的可交换范围之间的1:1的固定比率。然而,本公开不限于使用针对近存储器的可交换范围和针对远存储器的可交换范围之间的1:1的固定比率。作为示例,可以使用1:3的比率。在这种情况下,可能需要附加的标记位来编码有关就具有高速缓存行的存储器的区域而言的高速缓存行的位置的信息。

[0031] 每个VM(例如,图1的VM 115、117、...159中的任何VM)可以具有虚拟可信平台模块

(TPM)实例,该实例在与特定VM相关联的安全环境中运行。虚拟TPM实例可以用作专用安全保险库,其用于存储用于加密和解密针对特定VM的数据的任何密钥。换言之,每个VM可以有至少一个密钥,允许每个VM将该特定密钥提供给包括在正在执行VM的CPU内(或与之相关联)的内部存储器控制器。虚拟TPM还可以允许VM以可信方式启动。每个内部存储器控制器(例如,图1的内部存储器控制器118、148和178中的任何内部存储器控制器)可以包括高级加密标准(AES)引擎。AES引擎可以提供用于基于每个VM密钥基础来加密和解密数据的硬件加速。AES引擎可以支持使用各种密钥大小,包括128、192和256位的密钥大小。加密/解密也可以仅使用软件来执行。

[0032] 图3A和图3B显示了当存在未命中时与读取操作相关的示例事务流300,因为所请求的数据不在近存储器中。在读取操作期间,CPU(例如,图1的CPU 112、142或172中的任何CPU)可以发出命令,该命令由与内部存储器控制器(例如,图1的内部存储器控制器118、148和178中的任何内部存储器控制器)相关联的主代理处理以读取数据。在该示例中,如图3A中所示,存在来自主代理的读取高速缓存行CL$A的读取请求。读取请求不仅包括针对高速缓存行的地址,而且还包括与做出请求的VM相关联的密钥。主代理可以是最后一级高速缓存控制器(例如,图1的SLCC 116、SLCC 146和SLCC 176中的任何SLCC)或控制针对给定高速缓存行的相干性的任何其他控制器。主代理可以确保如果与CPU相关联的多个核心正在请求对高速缓存行的访问,则这些请求由相同的控制逻辑(例如,主代理)处理。内部存储器控制器(例如,前面关于图1所描述的内部存储器控制器中的任何内部存储器控制器)检查近存储器的内容。在操作上,响应于针对高速缓存行CL$A的读取请求,如果元标记部分指示近存储器包含高速缓存行CL$B,则其导致未命中。因此,内部存储器控制器向远存储器系统发送针对高速缓存行CL$A的请求。此外,在内部存储器控制器中针对映射到近存储器位置的其他高速缓存行条目设置阻止条目。

[0033] 在该示例中,远存储器系统被假设为符合CXL规范的系统,因此针对高速缓存行CL$A的请求去往CXL根端口。该示例还假设跨每个CXL链路(从CXL根端口到CXL端点)的任何事务都被加密。CXL根端口可以被视为类似于PCIe根端口,并且CXL端点可以被视为类似于PCIe端点。因此,具有根端口和端点的任何技术可以被用于枚举链路。链路级加密引起对跨链路传输的数据的双重加密。

[0034] 继续参考图3A,此外,CXL根端口可以生成消息认证码(MAC)以确保跨物理链路的数据的完整性。跨物理链路的数据可以以包括MAC报头的flit(例如,512位)的形式传输。在一个示例中,可以同时处理多个flit以生成针对多个flit的MAC,使得不针对每个flit生成MAC。作为示例,可以处理四个flit以生成MAC。可以使用加密哈希函数(例如,HMAC)或使用分组密码算法来生成MAC。

[0035] 仍然参考图3A,读取A请求从CXL根端口行进到CXL端点,后者继而从远存储器取回数据。CXL端点通过基于接收到的flit在本地上生成MAC并且将所生成的MAC与接收到的MAC进行比较来验证接收到的数据的完整性。针对高速缓存行CL$A的取回到的数据从CXL端点行进回到CXL根端口。CXL端点还为特定数量的flit生成MAC并且将其传输回到CXL根端口。CXL根端口通过基于接收到的flit在本地生成MAC并且将所生成的MAC与接收到的MAC进行比较来验证接收到的数据的完整性。数据由内部存储器控制器接收,该内部存储器控制器使用每个VM密钥(密钥A)来解密数据并且将数据提供给请求方(例如,主代理)。数据(A)还

被存储在近存储器中,作为近存储器和远存储器之间的数据的交换的一部分。

[0036] 此外,如图3B中所示,作为该交换操作的一部分,将对应于高速缓存行CL$B的数据写入远存储器。作为该过程的一部分,已经加密的数据(由内部存储器控制器使用每个VM密钥加密)被传输到CXL根端口。CXL根端口还加密跨CXL根端口和CXL端点之间的物理链路的数据。CXL根端口还生成消息认证码(MAC)以确保跨物理链路的数据的完整性。CXL端点将对应于高速缓存行CL$B的数据写入远存储器。完成(CMP)消息被传输回到内部存储器控制器。

[0037] 关于图3A和图3B中显示的完整性相关处理(使用MAC),系统可以以两种模式操作。一种模式可以被称为遏制模式,而另一种模式可以被称为滑行模式。在遏制模式下,CXL端点可能仅在完整性检查通过之后才释放数据。结果,可能需要缓冲若干flit(例如,四个flit),直到由CXL端点接收到MAC以及已经执行了完整性检查(例如,通过将本地生成的MAC与接收到的MAC进行比较)。这可能影响与远存储器相关联的操作的时延。然而,图1和图2中描述的系统可以通过确保近存储器的命中率高来减轻这种影响,使得更少的存储器访问需要在近存储器和远存储器之间交换数据。

[0038] 另一种缓解时延的方式可以是使用另一种模式——滑行模式。在一个示例中,在滑行模式下,接收到的数据无需等待接收MAC即可被释放。当(例如,由CXL端点)接收到MAC时,可以将本地生成的MAC与接收到的MAC进行比较。虽然该模式减少时延,但在已经完成完整性检查之前可能传输受损或否则损坏的数据。然而,如关于图3A和图3B所解释的,作为具有加密/解密的直接交换高速缓存的一部分,来自远存储器的数据不被直接发送给请求方(例如,本地代理)。相反,数据首先被发送到内部存储器控制器,该内部存储器控制器需要在将数据发送给请求方之前解密数据。存在与解密操作相关联的特定量的时延。在一些实例中,解密时延可以是足以在滑行模式下完成完整性检查的时间,以及从而允许在内部存储器控制器将数据提供给请求方之前通知内部存储器控制器任何完整性违规。换言之,在几乎所有情况下,都可以使内部存储器控制器意识到任何完整性检查相关的问题,因为内部存储器控制器将需要特定时间量来解密数据,该特定时间量可能几乎是与完整性检查所需的相同时间量。总之,完整性检查和解密操作可以被并行化。因此,代替使用遏制模式来确保完整性并且遭受更高的时延,可以使用具有较低时延的滑行模式,因为损坏的数据被提供给请求方的可能性非常小。

[0039] 图4是显示当存在命中时与读取操作和写入操作相关的示例事务流400的图,其中读取操作和写入操作所针对的数据位于近存储器中。在读取操作期间,CPU(例如,图1的CPU 112、142或172中的任何)可以发出命令,该命令由与内部存储器控制器(例如,图1的内部存储器控制器118、148和178中的任何)相关联的主代理处理以读取数据。在该示例中,存在来自主代理的读取高速缓存行CL$A的读取请求。读取请求不仅包括高速缓存行的地址,而且还包括与做出请求的VM相关联的密钥。内部存储器控制器(例如,前面关于图1描述的任何内部存储器控制器)检查近存储器的内容。该示例中的元标记部分指示近存储器包含高速缓存行CL$A,从而导致命中。因此,内部存储器控制器从近存储器取回高速缓存行CL$A的数据。在该示例中,远存储器系统被假设为符合CXL规范的系统,因此如果已经存在未命中,则对高速缓存行CL$A的请求将去往CXL根端口。该示例还假设跨(从CXL根端口到CXL端点的)每个CXL链路的任何事务都被加密。内部存储器控制器使用每个VM密钥(密钥A)解密数据并且将数据提供给主代理。

[0040] 继续参考图4,在该示例中,关于写入A操作,当将高速缓存行写入存储器时,每次写入都需要前面有读取,以确保近存储器位置包含正在写入的地址。因此,内部存储器控制器首先发出读取命令以从近存储器读取数据,而该数据恰好是正在写入的数据,因此存在命中。接下来,内部存储器控制器利用每个VM密钥(例如,密钥A)加密数据并且将数据写入近存储器。最后,内部存储器控制器将完成(CMP)消息发送到主代理。

[0041] 此外,直接交换高速缓存还可以与可信执行环境访问控制(TEE-AC)架构中的可信域标识符(TDI)一起使用。在一个示例中,TDI位可以由在计算平台上运行的可信代理分配。可信代理可能受到VM的信任,因为每个VM都会审核和签署与可信代理相关联的代码。TDI可以用于解决威胁,诸如密文泄露、存储器损坏、别名和重新映射。作为示例,具有单独可信域的每个VM可以与其他VM和任何管理程序(或主机OS)软件隔离。TDI作为高速缓存行粒度元数据被保持在存储器中。然而,当存储器内容作为近存储器和远存储器之间的直接交换高速缓存的一部分可交换时,在执行交换操作时必须跨CXL物理链路运送元数据。为了避免跨CXL物理链路运送TDI,公开了两种技术。第一种技术将TDI添加到由内部存储器控制器保持的元数据中。第二种技术使用页映射到相同页,稍后将对此进行解释。

[0042] 作为第一种技术的一部分,针对每个高速缓存行的TDI位(无论其位于近存储器还是远存储器中)可以由内部存储器控制器(例如,图1中显示的任何内部控制器)管理,保持在近存储器中,并且永远不被传输到远存储器。下表1显示了当TDI为一位时用于跟踪TDI的附加位。下表1中显示的位可以在每个高速缓存行基础上被存储在近存储器中。

[0043]

位	描述
a	指示近存储器中的高速缓存行是高速缓存行$A 还是高速缓存行$B
b	近存储器中的高速缓存行的 TDI 值
c	远存储器中的高速缓存行的 TDI 值

[0044] 第二种技术利用了以下事实:在某些系统中,可信域分配是在页粒度基础上执行的。作为该技术的一部分,系统地址映射中的地址可以以给定近存储器/远存储器高速缓存行对始终映射到相同页的方式分配。这样,有利地,无需在CXL物理链路上传递TDI信息,因为TDI信息对于与特定VM相关联的近存储器和远存储器内容两者都具有相同的值。

[0045] 作为示例,可以设置系统地址映射以将可交换地址范围划分为更小粒度的页大小的区域。作为示例,假设2兆兆字节(TB)的存储器范围可用于与系统地址映射(页大小为1GB)一起使用,则1TB被配置为不可交换范围以及1TB被配置为可交换范围。可以使用低位地址位将该存储器范围(可交换范围)划分为半页大小的粒度区域,每个区域的大小为512MB。在该布置中,只要租户(例如,由该计算节点托管的任何虚拟机)被分配的地址范围等于1GB(较小粒度页大小区域的两倍),则与虚拟机相关联的高速缓存行对映射到相同页。分配给每个租户的地址范围可以被视为具有冲突集大小(例如,1GB),在该示例中,该冲突集大小被选择为与系统相关联的页大小相同的大小。主机OS(例如,管理程序)可以以1GB的增量向租户分配存储器。每个1GB增量不需要是连续的。每个冲突集(具有两个冲突的512MB可交换区域)对应于租户可访问的物理存储器中的单个512MB区域(例如,DRAM)。因此,单个

1GB页对应于物理存储器中的单个512MB区域。在该示例中,低位地址位(例如,地址位29)可以具有逻辑值“0”或“1”,以区分两个512MB冲突区域。当地址位29的逻辑值为“0”时,则高速缓存行的地址对应于512MB冲突区域之一,而当地址位29的逻辑值为“1”时,则高速缓存行的地址对应于另一个512MB冲突区域。其他类型的编码也可以用作寻址的一部分,以区分两个冲突区域。

[0046] 图5显示了用于实现用于集成存储器池和直接交换高速缓存的方法中的至少一些的示例系统500的框图。系统500可以包括(多个)处理器502、(多个)I/O组件504、存储器506、(多个)呈现组件508、传感器510、(多个)数据库512、网络化接口514和(多个)I/O端口516,它们可以经由总线520互连。(多个)处理器502可以执行存储在存储器506中的指令。(多个)I/O组件504可以包括诸如键盘、鼠标、语音识别处理器或触摸屏的组件。存储器506可以是非易失性存储器或易失性存储器(例如,闪存存储器、DRAM、SRAM或其他类型的存储器)的任何组合。呈现组件508可以包括显示器、全息设备或其他呈现设备。显示器可以是任何类型的显示器,诸如LCD、LED或其他类型的显示器。(多个)传感器510可以包括遥测或其他类型的传感器,其被配置为检测和/或接收信息(例如,收集到的数据)。(多个)传感器510可以包括遥测或其他类型的传感器,其被配置为检测和/或接收信息(例如,数据中心中的各个计算节点正在执行的各种虚拟机的存储器使用情况)。(多个)传感器510可以包括被配置为感测与CPU、存储器或其他存储组件、FPGA、主板、基板管理控制器等相关联的状况的传感器。(多个)传感器510还可以包括被配置为感测与机架、底盘、风扇、电源单元(PSU)等相关联的状况的传感器。(多个)传感器510还可以包括被配置为感测与网络接口控制器(NIC)、机架顶部(TOR)交换机、机架中部(MOR)交换机、路由器、配电单元(PDU)、机架级不间断电源(UPS)系统等相关联的状况的传感器。

[0047] 仍参考图5,(多个)数据库512可以用于存储任何收集或记录的数据,并且根据需要执行本文描述的方法。(多个)数据库512可以被实现为分布式数据库的汇集或单个数据库。(多个)网络化接口514可以包括通信接口,诸如以太网、蜂窝无线电、蓝牙无线电、UWB无线电或其他类型的无线或有线通信接口。(多个)I/O端口516可以包括以太网端口、光纤端口、无线端口、或其他通信或诊断端口。虽然图5将系统500显示为包括以某种方式布置和耦合的特定数量的组件,但其可以包括不同地布置和耦合的更少的或附加的组件。此外,与系统500相关联的功能可以根据需要进行分布。

[0048] 图6显示了根据一个示例的用于实现集成存储器池和直接交换高速缓存的系统的数据中心600。作为示例,数据中心600可以包括机架的若干集群,包括平台硬件,诸如计算资源、存储资源、网络化资源或其他类型的资源。计算资源可以经由计算节点提供,这些计算节点经由可以连接到交换机以形成网络的服务器配置。网络可以实现交换机的每个可能组合之间的连接。数据中心600可以包括服务器1 610和服务器N 630。数据中心600还可以包括数据中心相关的功能660,包括部署/监控670、目录/身份服务672、负载平衡674、数据中心控制器676(例如,软件定义网络(SDN)控制器和其他控制器)以及路由器/交换机678。服务器1 610可以包括(多个)CPU 611、主机管理程序612、近存储器613、(多个)存储接口控制器(SIC)614、远存储器615、(多个)网络接口控制器((多个)NIC)616以及存储磁盘617和618。远存储器615可以被实现为池化存储器,如前所述。服务器N 630可以包括CPU 631、主机管理程序632、近存储器633、(多个)存储接口控制器(SIC)634、远存储器635、(多个)网络

接口控制器((多个)NIC)636以及存储磁盘637和638。远存储器635可以被实现为池化存储器,如前所述。服务器1 610可以被配置为支持虚拟机,包括VM1 619、VM2 620和VMN 621。虚拟机还可以被配置为支持应用,诸如APP1 622、APP2623和APPN 624。服务器N 630可以被配置为支持虚拟机,包括VM1639、VM2 640和VMN 641。虚拟机还可以被配置为支持应用,诸如APP1 642、APP2 643和APPN 644。

[0049] 继续参考图6,在一个示例中,数据中心600可以使用虚拟可扩展局域网(VXLAN)框架为多个租户启用。每个虚拟机(VM)可以允许与相同VXLAN段中的VM通信。每个VXLAN段可以由VXLAN网络标识符(VNI)标识。虽然图6将数据中心600显示为包括以特定方式布置和耦合的特定数量的组件,但其可以包括不同地布置和耦合的更少的或附加的组件。此外,与数据中心600相关联的功能可以根据需要进行分布或组合。

[0050] 图7显示了用于管理具有近存储器和远存储器的存储器的示例方法的流程图700。在一个示例中,与该方法相关联的步骤可以由前面描述的系统(例如,图1的系统100和图2的系统200)的各个组件执行。步骤710可以包括接收来自请求方的读取存储在近存储器中或远存储器中的第一数据块的请求,其中读取请求包括与对应于系统的第一虚拟机相关联的第一密钥,其中第一密钥由第一虚拟机专用。作为示例,读取请求可以类似于前面关于图3A的事务流300描述的Rd A请求。如前所述,在读取操作期间,CPU(例如,图1的CPU 112、142或172中的任何)可以发出命令,该命令由与内部存储器控制器(例如,图1的内部存储器控制器118、148和178中的任何)相关联的主代理处理以读取数据。在该示例中,如图3A中所示,存在来自主代理的读取数据块(例如,高速缓存行CL$A)的读取请求。读取请求不仅包括高速缓存行的地址,而且还包括与做出请求的VM相关联的密钥。

[0051] 步骤720可以包括:响应于确定远存储器包含第一数据块的加密版本,(1)从远存储器取回第一数据块的加密版本、使用第一密钥解密第一数据块,以及向请求方提供第一数据块的解密版本,以及(2)将具有与第一数据块冲突的地址的第二数据块从近存储器换出到远存储器,其中第二数据块使用与对应于系统的第二虚拟机相关联的第二密钥加密,并且其中第二密钥由第二虚拟机专用。存储器控制器(例如,前面关于图1描述的任何近存储器控制器)可以分析与数据块相关联的元数据部分以确定远存储器包含数据块的加密版本。如关于图3A的事务流程300所描述的,当存在关于读取请求的未命中时可以将数据从近存储器换出到远存储器。此外,如前所述,近存储器控制器可以使用特定于代表其发起读取操作的虚拟机的密钥来解密数据。有利的是,使用每个VM密钥(第一VM的第一密钥和第二VM的第二密钥)可以允许虚拟机当中的保密计算和隔离。

[0052] 图8显示了用于管理具有近存储器和远存储器的存储器的示例方法的流程图800,其中远存储器与具有由至少一个物理链路分离的根端口和端点的远存储器系统相关联。在一个示例中,与该方法相关联的步骤可以由前面描述的系统(例如,图1的系统100和图2的系统200)的各种组件执行。步骤810可以包括对通过至少一个物理链路的在根端口和端点之间的事务的集合执行完整性检查,其中在完成完整性检查之前,释放与该事务的集合相关联的数据以供系统进一步处理。例如,如前面参考图3A所述,CXL根端口生成消息认证码(MAC)以确保跨物理链路的数据的完整性。跨物理链路的数据可以以包括MAC报头的flit(例如,512位)的形式传输。在一个示例中,可以同时处理多个flit以生成多个flit的MAC,使得不为每个flit生成MAC。例如,可以处理四个flit来生成MAC。可以使用加密哈希函数

(例如,HMAC)或使用分组密码算法来生成MAC。作为这些事务的对应方的CXL端点可以通过将本地生成的MAC与接收到的MAC进行比较来执行完整性检查。匹配可以指示完整性检查通过条件,而本地生成的MAC与接收到的MAC之间的匹配的缺乏可以指示完整性检查失败条件。如前所述,完整性检查可以在遏制模式或滑行模式下执行。

[0053] 步骤820可以包括接收来自请求方的读取存储在近存储器中或远存储器中的第一数据块的请求,其中读取请求包括与对应于系统的第一虚拟机相关联的第一密钥,并且其中第一密钥由第一虚拟机专用。作为示例,读取请求可以类似于前面关于图3A的事务流300描述的Rd A请求。如前所述,在读取操作期间,CPU(例如,图1的CPU 112、142或172中的任何)可以发出命令,该命令由与内部存储器控制器(例如,图1的内部存储器控制器118、148和178中的任何)相关联的主代理处理以读取数据。在该示例中,如图3A中所示,存在来自主代理的读取数据块(例如,高速缓存行CL$A)的读取请求。读取请求不仅包括高速缓存行的地址,而且还包括与做出请求的虚拟机相关联的密钥。

[0054] 步骤830可以包括响应于确定远存储器包含第一数据块的加密版本:(1)从远存储器取回第一数据块的加密版本、使用第一密钥解密第一数据块,以及向请求方提供第一数据块的解密版本,其中与解密相关联的时延足以允许完成完整性检查,以及(2)将具有与第一数据块冲突的地址的第二数据块从近存储器换出到远存储器,其中第二数据块使用与对应于系统的第二虚拟机相关联的第二密钥加密,并且其中第二密钥由第二虚拟机专用。存储器控制器(例如,前面关于图1描述的任何近存储器控制器)可以分析与数据块相关联的元数据部分以确定远存储器包含数据块的加密版本。如关于图3A的事务流程300所描述的,当存在关于读取请求的未命中时可以将数据从近存储器换出到远存储器。此外,如前所述,近存储器控制器可以使用特定于代表其发起读取操作的虚拟机的密钥来解密数据。有利的是,使用每个VM密钥(第一VM的第一密钥和第二VM的第二密钥)可以允许虚拟机当中的保密计算和隔离。虽然图8在流程图800中将与该方法相关的步骤显示为按特定顺序执行,但这些步骤不需要按所示的顺序执行。作为示例,与完整性检查相关联的步骤可以以这种方式执行,使得可以一起处理对应于与不同读取或写入操作相关联的多个事务的数据。

[0055] 总之,本公开涉及一种用于管理具有近存储器和远存储器的系统的方法。该方法可以包括接收来自请求方的读取存储在近存储器中或远存储器中的第一数据块的请求,其中读取请求包括与对应于系统的第一虚拟机相关联的第一密钥,其中第一密钥由第一虚拟机专用。该方法还可以包括:响应于确定远存储器包含第一数据块的加密版本,(1)从远存储器取回第一数据块的加密版本、使用第一密钥解密第一数据块,以及向请求方提供第一数据块的解密版本,以及(2)将具有与第一数据块冲突的地址的第二数据块从近存储器换出到远存储器,其中第二数据块使用与对应于系统的第二虚拟机相关联的第二密钥加密,并且其中第二密钥由第二虚拟机专用。

[0056] 作为该方法的一部分,确定远存储器包含第一数据块的加密版本的步骤可以包括分析与第一数据块相关联的元数据部分,该元数据部分具有与近存储器是否包含第一数据块或者远存储器是否包含第一数据块相关的信息。远存储器可以与具有由至少一个物理链路分离的根端口和端点的远存储器系统相关联,并且其中对应于读取请求的通过至少一个物理链路的事务可以被加密,从而导致在通过至少一个物理链路的运输期间对第一数据块的双重加密。

[0057] 远存储器可以与具有由至少一个物理链路分离的根端口和端点的远存储器系统相关联,以及该方法还可以包括对通过至少一个物理链路的在根端口和端点之间的事务的集合执行完整性检查。

[0058] 该方法还可以包括分析与第一数据块相关联的元数据部分,该元数据部分具有:(1)与近存储器是否包含第一数据块或者远存储器是否包含第一数据块相关的第一信息,(2)包括与存储在近存储器中的第二数据块相关联的第一可信域标识符值的第二信息,以及(3)包括与存储在远存储器中的第一数据块相关联的第二可信域标识符值的第三信息,其中第一可信域标识符值和第二可信域标识符值中的每个由与近存储器相关联的近存储器控制器管理,以及第一可信域标识符值和第二可信域标识符值都不被传输到远存储器。

[0059] 该方法还可以包括分析与第一数据块相关联的元数据部分,该元数据部分具有:(1)与近存储器是否包含第一数据块或者远存储器是否包含第一数据块相关的第一信息,以及(2)无论第二数据块被存储在近存储器中还是远存储器中都包括与第二数据块相关联的相同可信域标识符值的第二信息。第一数据块和第二数据块中的每个可以包括用于与系统相关联的中央处理单元(CPU)的高速缓存行。

[0060] 在另一方面中,本公开涉及一种具有近存储器和远存储器的系统。该系统可以包括近存储器控制器,该近存储器控制器被配置为接收来自请求方的读取存储在近存储器中或远存储器中的第一数据块的请求,其中读取请求包括与对应于系统的第一虚拟机相关联的第一密钥,其中第一密钥由第一虚拟机专用。近存储器控制器还可以被配置为响应于确定远存储器包含第一数据块的加密版本:(1)从远存储器取回第一数据块的加密版本、使用第一密钥解密第一数据块,以及向请求方提供第一数据块的解密版本,以及(2)将具有与第一数据块冲突的地址的第二数据块从近存储器换出到远存储器,其中第二数据块使用与对应于系统的第二虚拟机相关联的第二密钥加密,并且其中第二密钥由第二虚拟机专用。

[0061] 近存储器控制器还可以被配置为分析与第一数据块相关联的元数据部分,该元数据部分具有与近存储器是否包含第一数据块或者远存储器是否包含第一数据块相关的信息。远存储器可以与具有由至少一个物理链路分离的根端口和端点的远存储器系统相关联,以及其中对应于读取请求的通过至少一个物理链路的事务可以由远存储器系统加密,从而导致在通过至少一个物理链路的运输期间对第一数据块的双重加密。

[0062] 远存储器可以与具有由至少一个物理链路分离的根端口和端点的远存储器系统相关联,并且其中,使用消息认证码,可以对通过至少一个物理链路的任何事务执行完整性检查。近存储器控制器还可以被配置为分析与第一数据块相关联的元数据部分,该元数据部分具有:(1)与近存储器是否包含第一数据块或者远存储器是否包含第一数据块相关的第一信息,(2)包括与存储在近存储器中的第二数据块相关联的第一可信域标识符值的第二信息,以及(3)包括与存储在远存储器中的第一数据块相关联的第二可信域标识符值的第三信息,并且其中第一可信域标识符值和第二可信域标识符值中的每个由近存储器控制器管理,以及第一可信域标识符值和第二可信域标识符值都不被传输到远存储器。

[0063] 近存储器控制器还可以被配置为分析与第一数据块相关联的元数据部分,该元数据部分具有:(1)与近存储器是否包含第一数据块或者远存储器是否包含第一数据块相关的第一信息,以及(2)无论第二数据块被存储在近存储器中还是远存储器中都包括与第二数据块相关联的相同可信域标识符值的第二信息。该系统还可以包括中央处理单元(CPU),

其中第一数据块和第二数据块中的每个可以包括用于CPU的高速缓存行。

[0064] 在另一方面,本公开涉及一种用于管理具有近存储器和远存储器的系统的方法,其中远存储器与具有由至少一个物理链路分离的根端口和端点的远存储器系统相关联。该方法可以包括对通过至少一个物理链路的在根端口和端点之间的事务的集合执行完整性检查,其中在完成完整性检查之前,释放与该事务的集合相关联的数据以供系统进一步处理。该方法还可以包括接收来自请求方的读取存储在近存储器中或远存储器中的第一数据块的请求,其中读取请求包括与对应于系统的第一虚拟机相关联的第一密钥,并且其中第一密钥由第一虚拟机专用。

[0065] 该方法还可以包括:响应于确定远存储器包含第一数据块的加密版本,(1)从远存储器取回第一数据块的加密版本、使用第一密钥解密第一数据块,以及向请求方提供第一数据块的解密版本,其中与解密相关联的时延足以允许完成完整性检查,以及(2)将具有与第一数据块冲突的地址的第二数据块从近存储器换出到远存储器,其中第二数据块使用与对应于系统的第二虚拟机相关联的第二密钥加密,并且其中第二密钥由第二虚拟机专用。

[0066] 作为该方法的一部分,确定远存储器包含第一数据块的加密版本的步骤可以包括分析与第一数据块相关联的元数据部分,该元数据部分具有与近存储器是否包含第一数据块或者远存储器是否包含第一数据块相关的信息。远存储器可以与具有由至少一个物理链路分离的根端口和端点的远存储器系统相关联,以及对应于读取请求的通过至少一个物理链路的事务可以被加密,从而导致在通过至少一个物理链路的运输期间对第一数据块的双重加密。

[0067] 该方法还可以包括分析与第一数据块相关联的元数据部分,该元数据部分具有:(1)与近存储器是否包含第一数据块或者远存储器是否包含第一数据块相关的第一信息,(2)包括与存储在近存储器中的第二数据块相关联的第一可信域标识符值的第二信息,以及(3)包括与存储在远存储器中的第一数据块相关联的第二可信域标识符值的第三信息,并且其中第一可信域标识符值和第二可信域标识符值中的每个由与近存储器相关联的近存储器控制器管理,以及第一可信域标识符值和第二可信域标识符值都不被传输到远存储器。

[0068] 该方法还可以包括分析与第一数据块相关联的元数据部分,该元数据部分具有:(1)与近存储器是否包含第一数据块或者远存储器是否包含第一数据块相关的第一信息,以及(2)无论第二数据块被存储在近存储器中还是远存储器中都包括与第二数据块相关联的相同可信域标识符值的第二信息。第一数据块和第二数据块中的每个可以包括用于与系统相关联的中央处理单元(CPU)的高速缓存行。

[0069] 应当理解,本文中描绘的方法、模块和组件仅仅是示例性的。备选地,或者此外,本文中描述的功能可以至少部分地由一个或多个硬件逻辑组件执行。例如,但不限于,可以使用的说明性类型的硬件逻辑组件包括现场可编程门阵列(FPGA)、专用集成电路(ASIC)、专用标准产品(ASSP)、片上系统(SOC)、复杂可编程逻辑器件(CPLD)等。在抽象但仍然明确的意义上,实现相同功能的组件的任何布置都被有效地“关联”使得实现期望的功能。因此,本文中组合起来实现特定功能的任何两个组件都可以被视为彼此“相关联”使得实现期望的功能,而与架构或中间组件无关。同样,任何两个如此关联的组件也可以被视为彼此“可操作地连接”或“耦合”以实现期望的功能。仅仅因为组件(其可以是装置、结构、系统或功能的

任何其他实现)在本文中被描述为与另一个组件耦合并不意味着这些组件必定是单独的组件。作为示例,描述为耦合到另一个组件B的组件A可以是组件B的子组件,组件B可以是组件A的子组件,或者组件A和B可以是另一个组件C的组合的子组件。

[0070] 与本公开中描述的一些示例相关联的功能还可以包括存储在非暂时性介质中的指令。如本文所使用的术语“非暂时性介质”是指存储数据和/或指令的任何介质,这些数据和/或指令使得机器以特定方式操作。示例性非暂时性介质包括非易失性介质和/或易失性介质。非易失性介质包括例如硬盘、固态驱动器、磁盘或带、光盘或带、闪存存储器、EPROM、NVRAM、PRAM或其他这种介质、或这种介质的网络化版本。易失性介质包括例如动态存储器,诸如DRAM、SRAM、高速缓存或其他这种介质。非暂时性介质不同于传输介质,但可以与传输介质结合使用。传输介质用于将数据和/或指令传递到机器或从机器传递数据和/或指令。示例性传输介质包括同轴线缆、光纤线缆、铜线和无线介质,诸如无线电波。

[0071] 此外,本领域技术人员将意识到,上述操作的功能之间的界限仅仅是说明性的。多个操作的功能可以被组合成单个操作,和/或单个操作的功能可以被分布在附加的操作中。此外,备选实施例可以包括特定操作的多个实例,以及可以在各种其他实施例中更改操作的顺序。

[0072] 虽然本公开提供了特定示例,但可以在不脱离如所附权利要求中阐述的本公开的范围的情况下进行各种修改和改变。因此,说明书和附图将被视为说明性的而非限制性的,并且所有这种修改都旨在于被包含在本公开的范围内。本文关于特定示例描述的任何益处、优点或问题的技术方案都不旨于被解释为任何或所有权利要求的关键、必需或基本特征或元素。

[0073] 此外,如本文所使用的术语“一”或“一个”被定义为一个或多于一个。而且,权利要求中对诸如“至少一个”和“一个或多个”的引导性短语的使用不应当被解释为暗示通过词语“一”或“一个”引入另一权利要求元素将包含这种引入的任何权利要求元素的特定权利要求限制为仅包含一个这种元素的发明,即使在相同权利要求包括引导性短语“一个或多个”或“至少一个”和诸如“一”或“一个”的词语时。对于定冠词的使用也是如此。

[0074] 除非另有说明,否则诸如“第一”和“第二”的术语被用于任意地区分这种术语描述的元素。因此,这些术语不一定旨在于指示这种元素的时间或其他优先级。

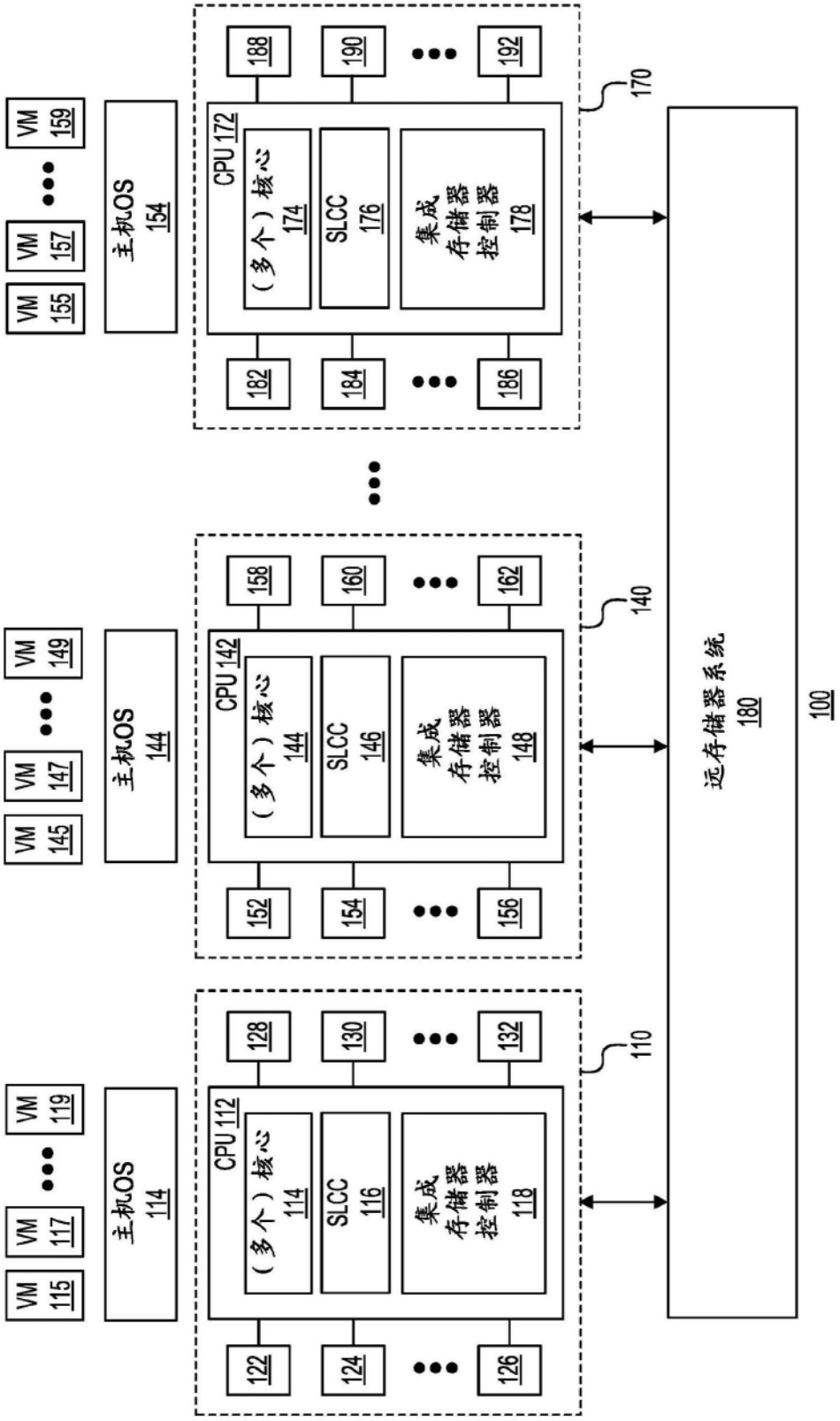

VM 115
VM 117
VM 119
主机OS 114
CPU 112
（多个）核心 114
SLCC 116
集成存储器控制器 118
122
124
126
128
130
132
110
VM 145
VM 147
VM 149
主机OS 144
CPU 142
（多个）核心 144
SLCC 146
集成存储器控制器 148
152
154
156
158
160
162
140
VM 155
VM 157
VM 159
主机OS 154
CPU 172
（多个）核心 174
SLCC 176
集成存储器控制器 178
182
184
186
188
190
192
170
远存储器系统 180
100

图1

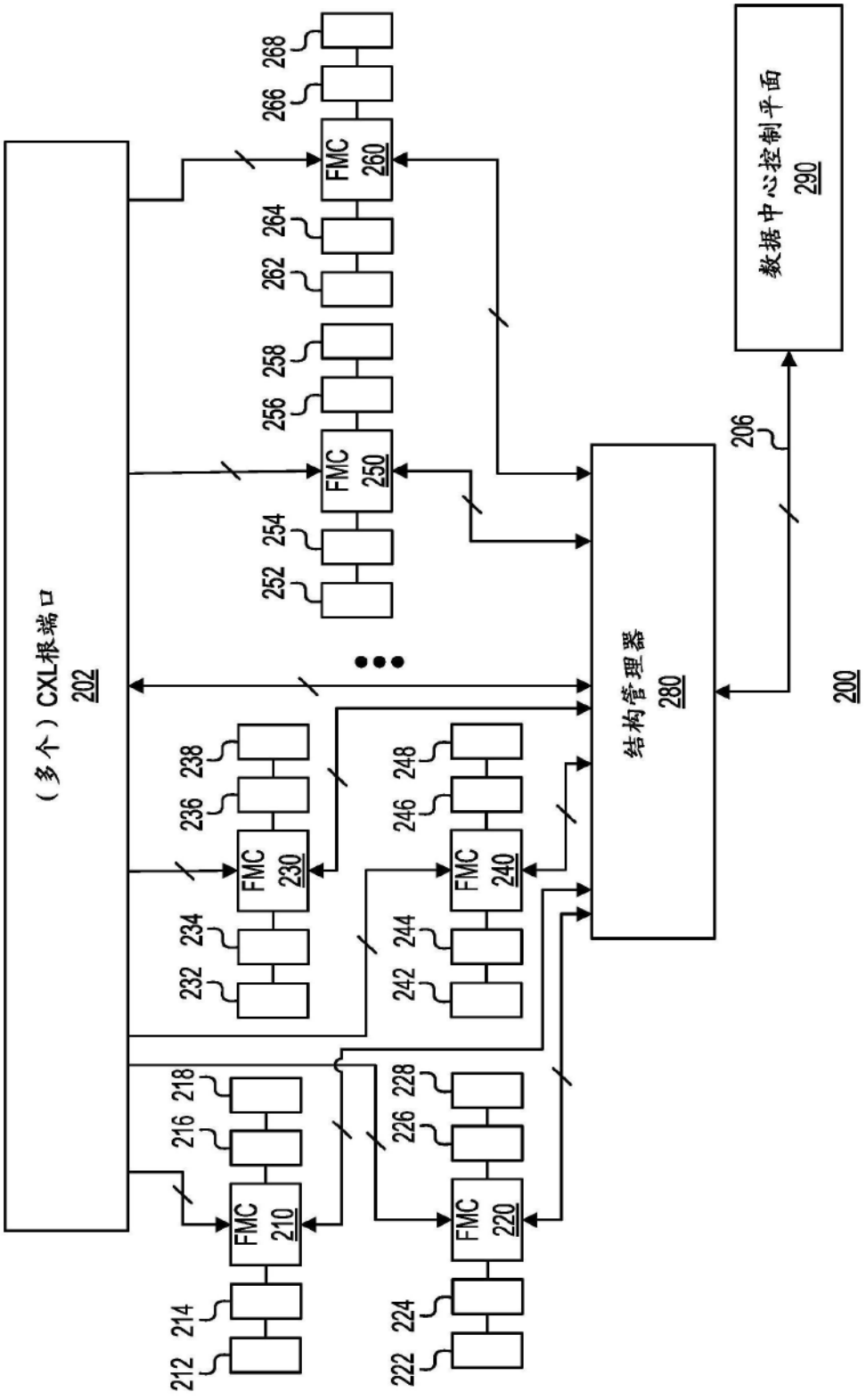

（多个）CXL根端口
202
结构管理器
280
数据中心控制平面
290
200
206
FMC
210
FMC
220
FMC
230
FMC
240
FMC
250
FMC
260
212
214
216
218
222
224
226
228
232
234
236
238
242
244
246
248
252
254
256
258
262
264
266
268

图2

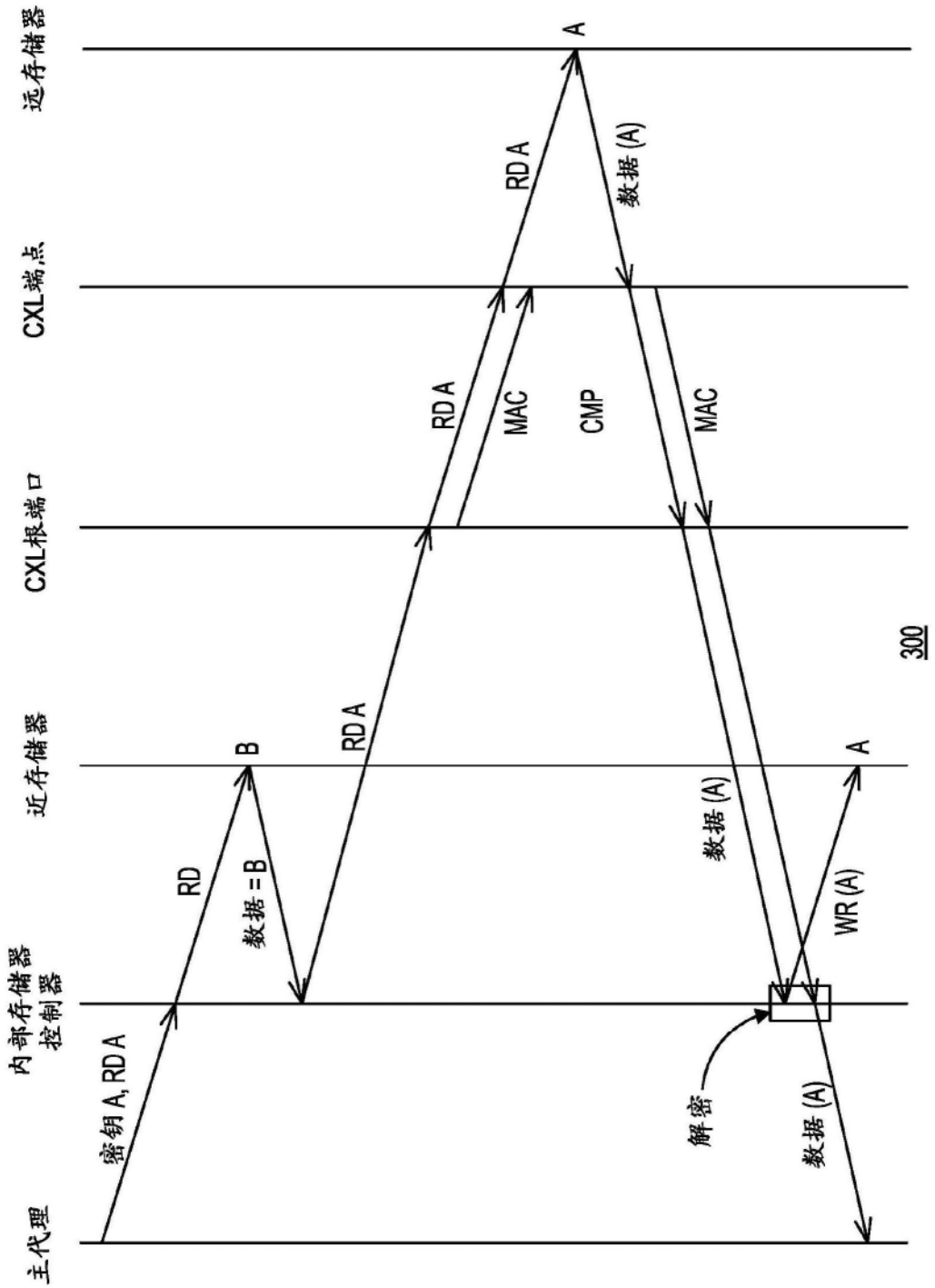

远存储器
CXL端点
CXL根端口
近存储器
内部存储器控制器
主代理
A
RD A
数据 (A)
RD A
MAC
CMP
MAC
300
B
RD A
A
RD
数据 = B
数据 (A)
WR (A)
密钥 A, RD A
解密
数据 (A)

图3A

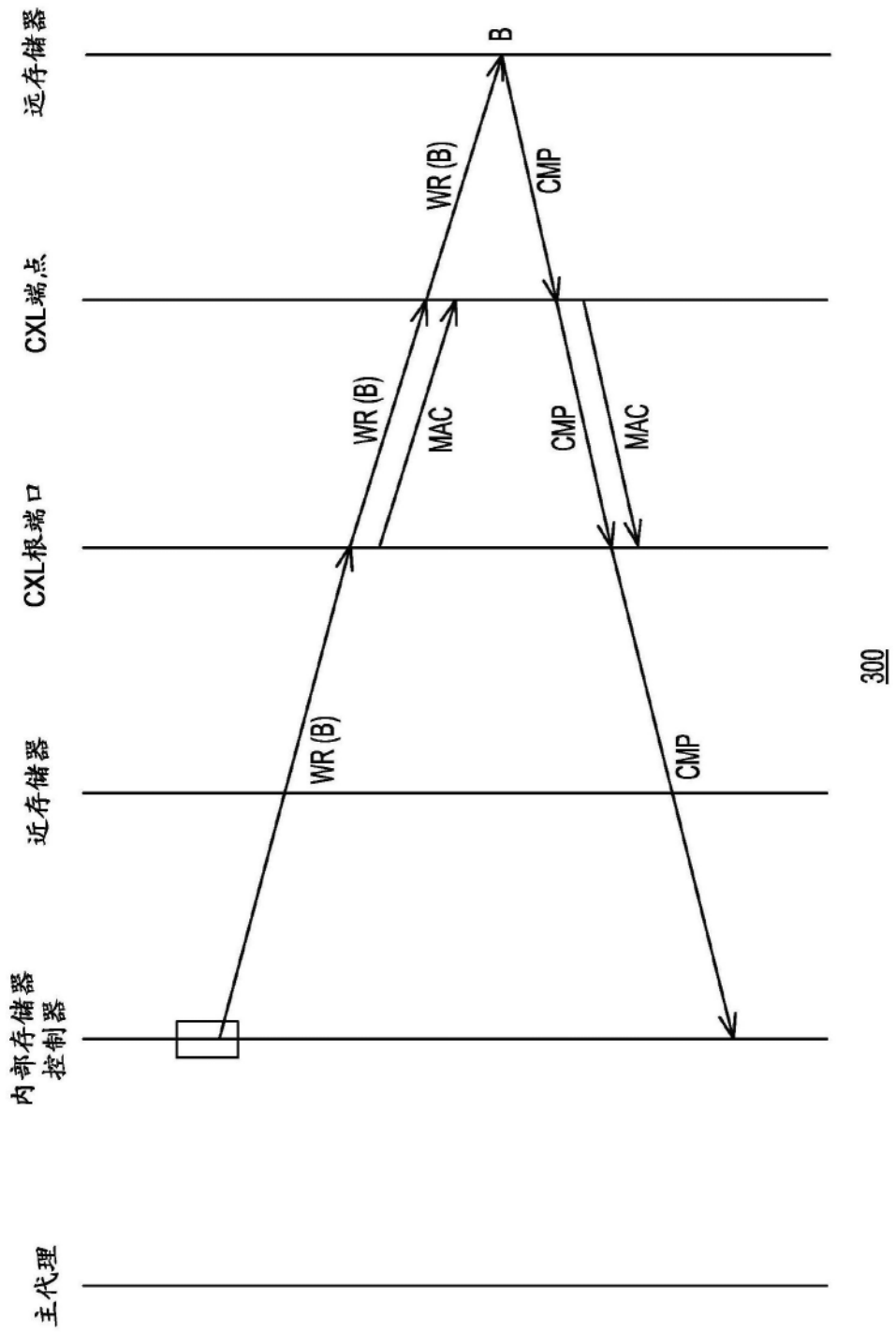

远存储器
CXL端点
CXL根端口
近存储器
内部存储器控制器
主代理
B
WR (B)
CMP
WR (B)
MAC
CMP
MAC
WR (B)
CMP
300

图3B

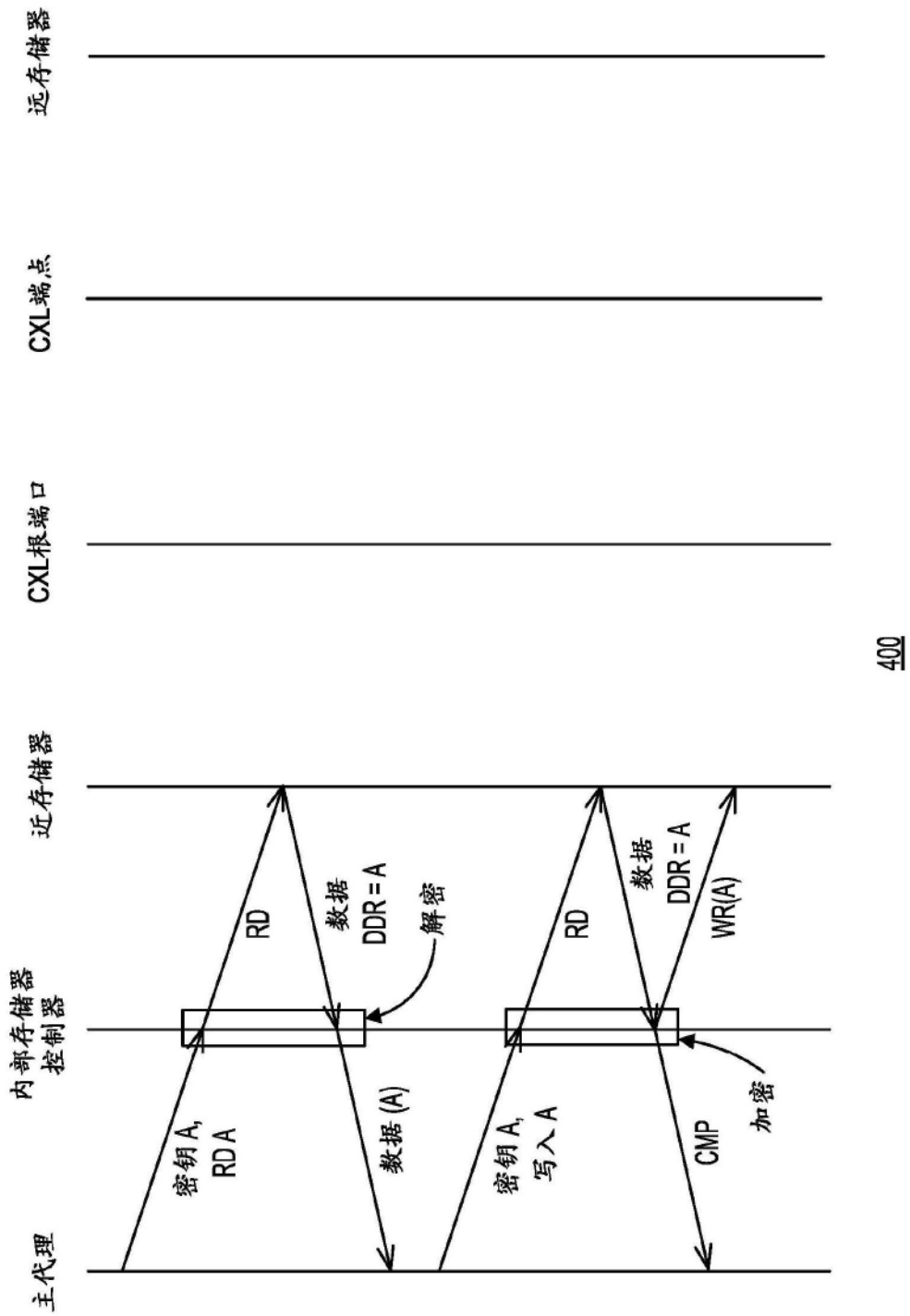

远存储器
CXL端点
CXL根端口
近存储器
内部存储器控制器
主代理
400
RD
数据
DDR = A
解密
密钥 A,
RD A
数据 (A)
RD
数据
DDR = A
WR(A)
密钥 A,
写入 A
CMP
加密

图4

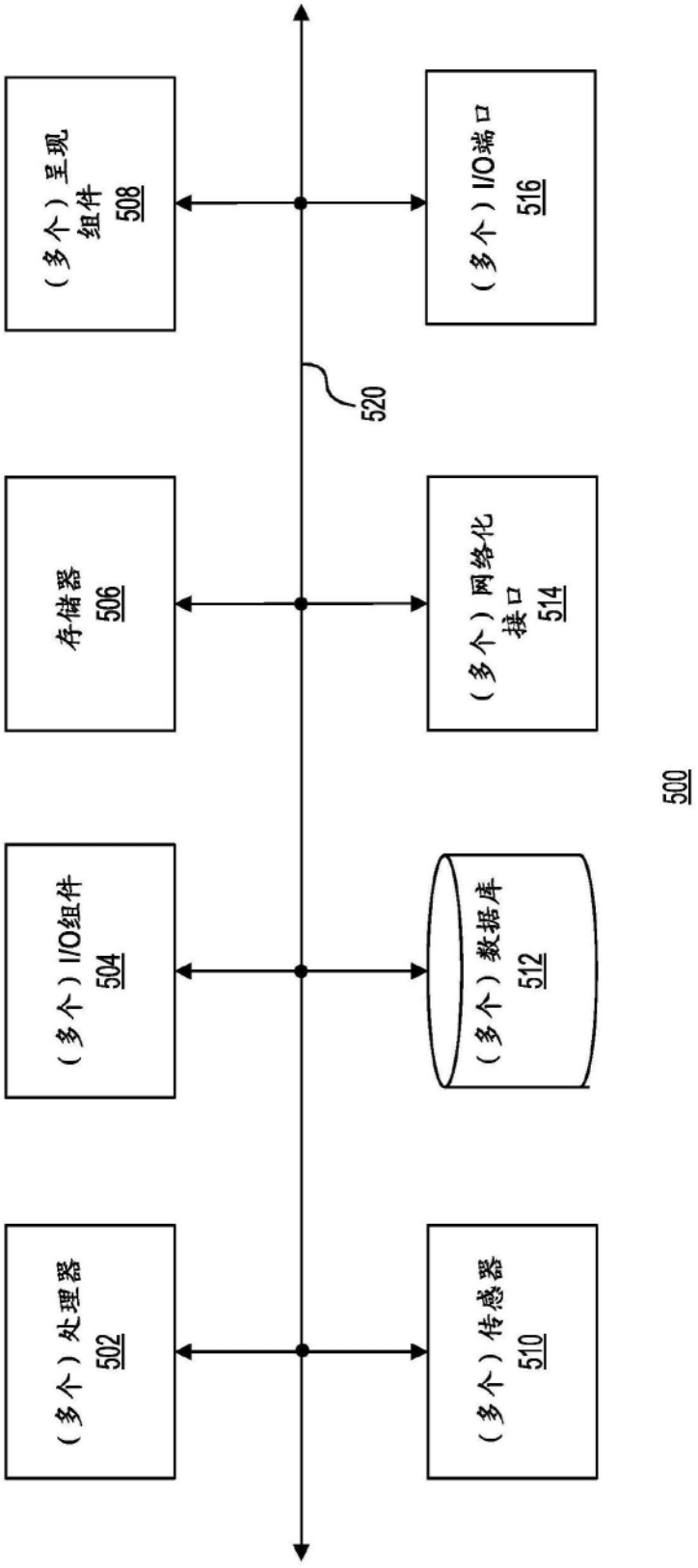

（多个）呈现组件 508
（多个）I/O端口 516
520
存储器 506
（多个）网络化接口 514
500
（多个）I/O组件 504
（多个）数据库 512
（多个）处理器 502
（多个）传感器 510

图5

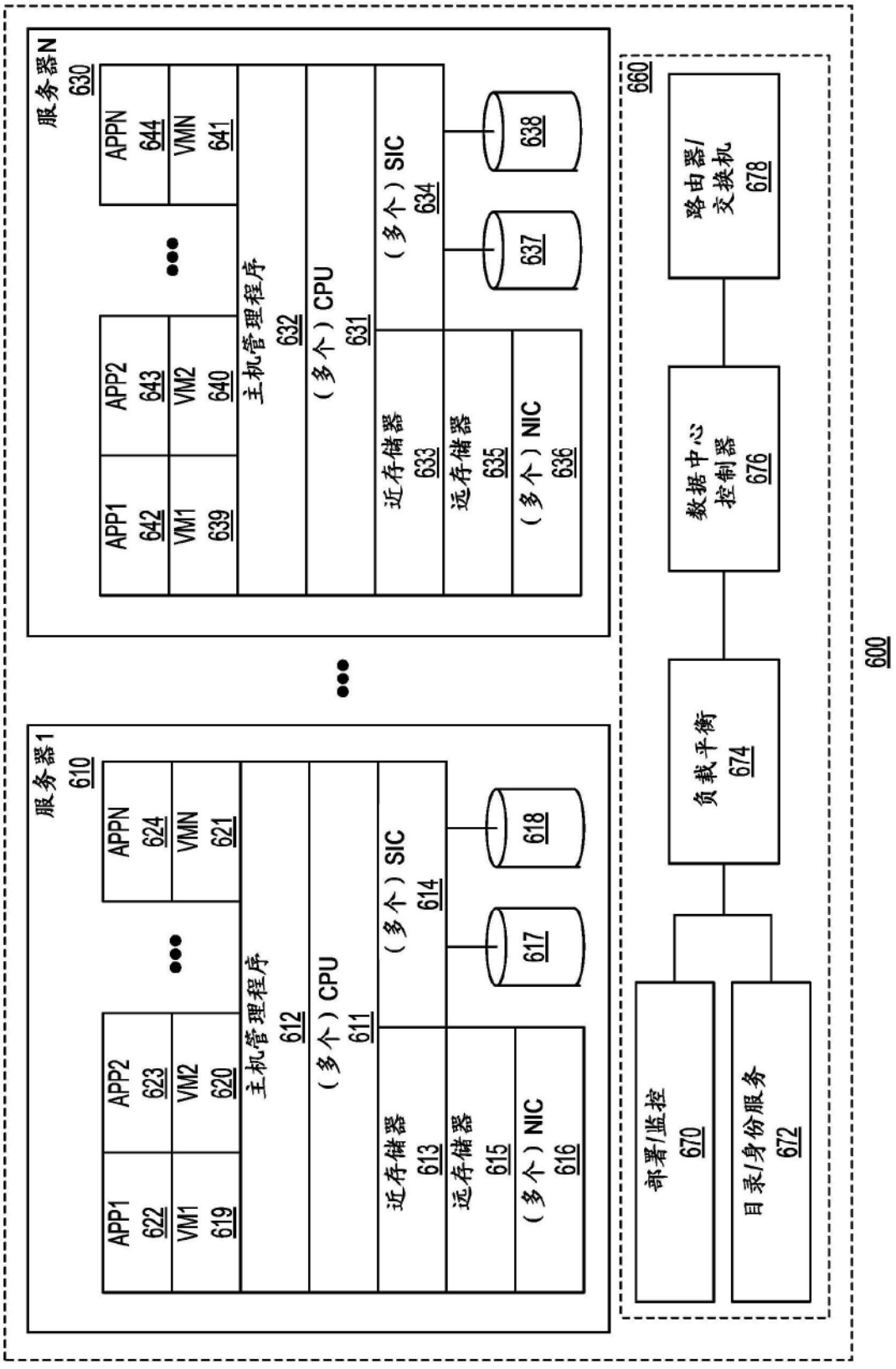

600
服务器1
610
APP1
622
VM1
619
APP2
623
VM2
620
APPN
624
VMN
621
主机管理程序
612
（多个）CPU
611
近存储器
613
远存储器
615
（多个）NIC
616
（多个）SIC
614
617
618
服务器N
630
APP1
642
VM1
639
APP2
643
VM2
640
APPN
644
VMN
641
主机管理程序
632
（多个）CPU
631
近存储器
633
远存储器
635
（多个）NIC
636
（多个）SIC
634
637
638
660
部署/监控
670
目录/身份服务
672
负载平衡
674
数据中心
控制器
676
路由器/
交换机
678

图6

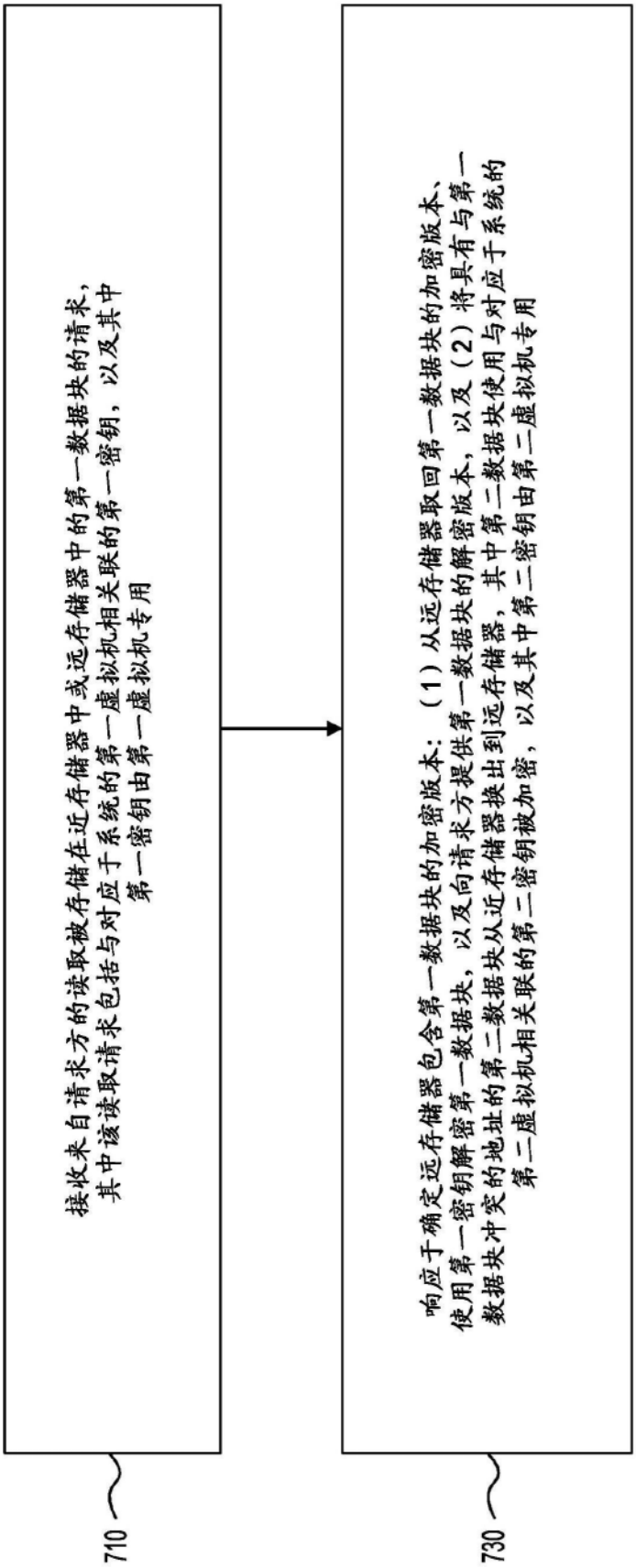

接收来自请求方的读取被存储在近存储器中或远存储器中的第一数据块的请求，其中该读取请求包括与对应于系统的第一虚拟机相关联的第一密钥，以及其中第一密钥由第一虚拟机专用
710
响应于确定远存储器包含第一数据块的加密版本：（1）从远存储器取回第一数据块的加密版本、使用第一密钥解密第一数据块，以及向请求方提供第一数据块的解密版本，以及（2）将具有与第一数据块冲突的地址的第二数据块从近存储器换出到远存储器，其中第二数据块使用与对应于系统的第二虚拟机相关联的第二密钥被加密，以及其中第二密钥由第二虚拟机专用
730
700

图7

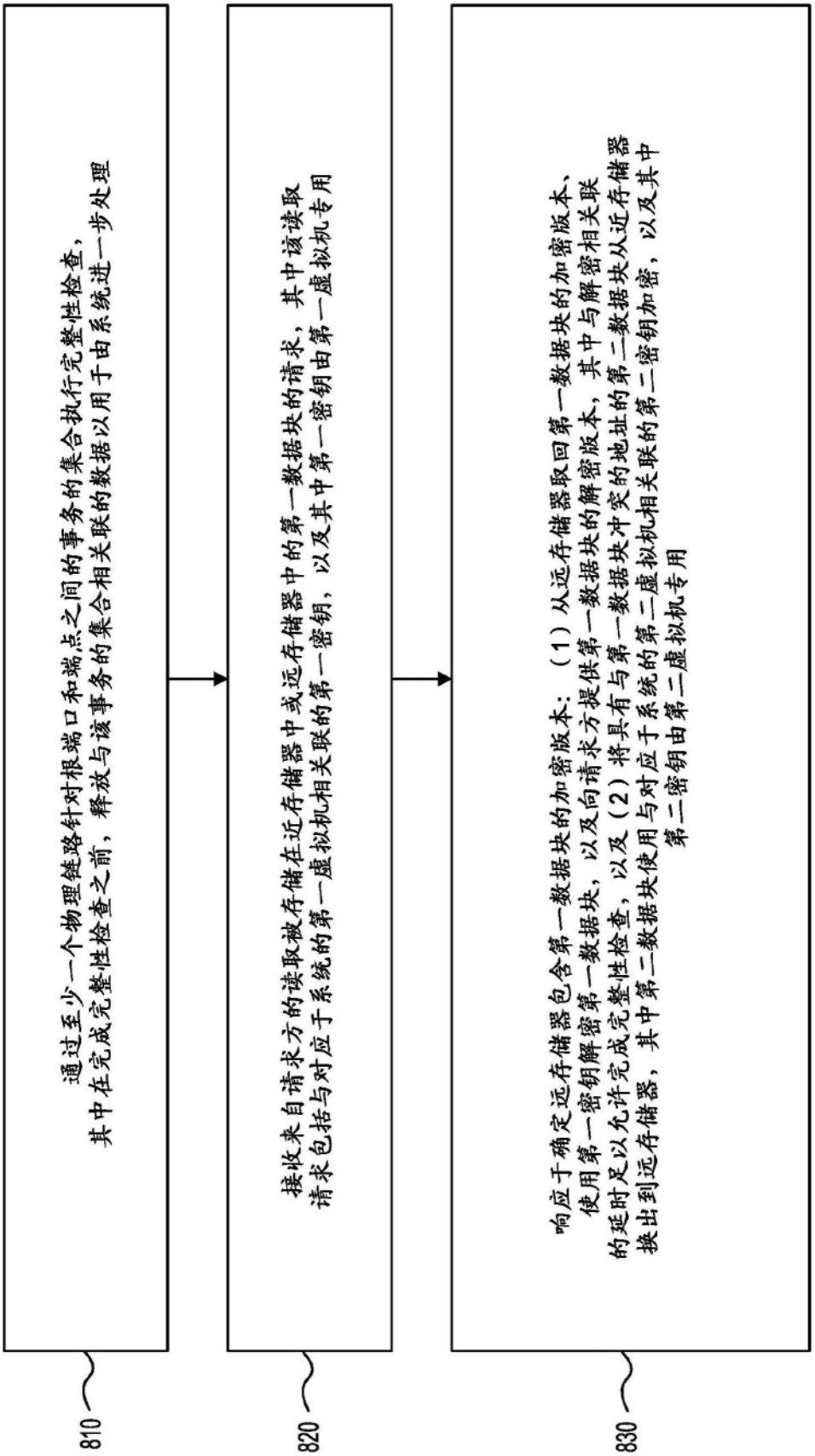

通过至少一个物理链路针对根端口和端点之间的事务的集合执行完整性检查，
其中在完成完整性检查之前，释放与该事务的集合相关联的数据以用于由系统进一步处理
810
接收来自请求方的读取被存储在近存储器中或远存储器中的第一数据块的请求，其中该读取
请求包括与对应于系统的第一虚拟机相关联的第一密钥，以及其中第一密钥由第一虚拟机专用
820
响应于确定远存储器包含第一数据块的加密版本：（1）从远存储器取回第一数据块的加密版本、
使用第一密钥解密第一数据块，以及向请求方提供第一数据块的解密版本，其中与解密相关联
的延时足以允许完成完整性检查，以及（2）将具有与第一数据块冲突的地址的第二数据块从近存储器
换出到远存储器，其中第二数据块使用与对应于系统的第二虚拟机相关联的第二密钥加密，以及其中
第二密钥由第二虚拟机专用
830
800

图8