(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 103038750 A
(43)申请公布日 2013.04.10

(21)申请号 201180024883.4
(22)申请日 2011.03.31
(30)优先权数据
61/319,658 2010.03.31 US
61/320,242 2010.04.01 US
(85)PCT申请进入国家阶段日
2012.11.20
(86)PCT申请的申请数据
PCT/US2011/030801 2011.03.31
(87)PCT申请的公布数据
WO2011/123692 EN 2011.10.06
(71)申请人 安全第一公司
地址 美国加利福尼亚
(72)发明人 R·L·奥尔西尼 M·S·奥黑尔
(74)专利代理机构 中国国际贸易促进委员会专利商标事务所 11038
代理人 付建军
(51)Int.Cl.
G06F 11/10(2006.01)
G06F 21/60(2013.01)
G06F 21/62(2013.01)
H04L 9/08(2006.01)
H04L 29/08(2006.01)

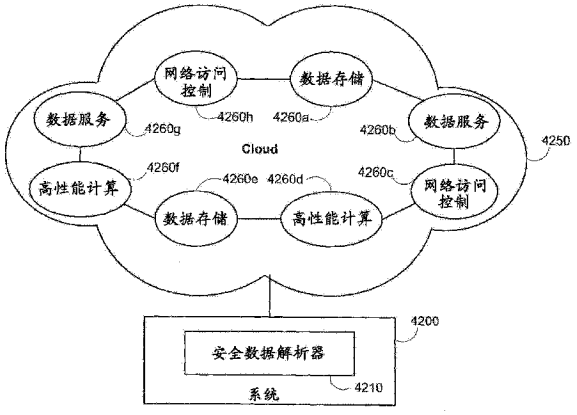
权利要求书 4 页 说明书 78 页 附图 55 页

(54)发明名称

对移动中数据进行保护的系统和方法

(57)摘要

本发明的系统和方法提供了一种使数据可证地安全和可访问的方案，致力于比特级的数据安全性，从而消除了对多个周边硬件和软件技术的需要。数据安全性直接包括或者交织在比特级的数据中。本发明的系统和方法使得企业兴趣团体能够利用通常的企业基础设施。因为安全性已被交织到数据中，可以在不损害数据安全性和访问控制的情况下使用该通常的基础设施。在一些应用中，在被发送到多个位置(例如私有或公有云)之前，数据被进行认证、加密和解析或者被分裂成多个份。数据在传送到存储位置的同时被隐藏，并且没有正确的访问证明的用户无法访问。



CN 103038750 A

1. 一种用于对从使用第一分裂密钥通过信息分散算法设置的加密数据产生的一组数据份进行重建的方法，该方法包括：

至少接收重建所述一组数据份所需的最小数目的数据份；以及

在不解密所述最小数目的数据份的情况下从所述最小数目的数据份重建所述一组数据份。

2. 根据权利要求 1 的方法，其中，响应于确定一个或多个所述数据份已被泄露而执行所述重建。

3. 根据权利要求 1 的方法，还包括把至少一个重建的数据份存储在存储网络上。

4. 根据权利要求 3 的方法，其中，存储网络包括私有云、公有云、混合云、可移动存储装置和海量存储装置中的一个。

5. 根据权利要求 1 的方法，其中，重建包括：

利用认证密钥对所述最小数目的数据份进行认证；

使用所述分裂密钥从认证后的最小数目的数据份重构所述加密数据；

通过使用所述分裂密钥把所述加密数据分裂而重新产生所述一组数据份。

6. 一种用于对从使用第一加密密钥通过信息分散算法设置的加密数据产生的一组数据份重新加密钥的方法，该方法包括：

至少接收重建所述一组数据份所需的最小数目的数据份；

把所述最小数目的数据份与第一认证密钥相关联；

在不解密所述最小数目的数据份的情况下从所述最小数目的数据份重建所述一组数据份；以及

通过将重建的一组数据份与第二加密密钥相关联而对重建的一组数据份重新加密钥。

7. 根据权利要求 6 的方法，还包括：

检索与所述最小数目的数据份关联的首标；

从检索到的首标提取密钥加密密钥；

利用密钥加密密钥对第二加密密钥进行加密；以及

把加密的第二认证密钥存储在重新加密钥后的数据份的首标内。

8. 根据权利要求 6 的方法，还包括把重新加密钥后的数据份中的至少一个存储在存储网络上。

9. 根据权利要求 8 的方法，其中，存储网络包括私有云、公有云、混合云、可移动存储装置和海量存储装置中的一个。

10. 一种用于对从使用第一分裂密钥通过信息分散算法设置的加密数据产生的一组数据份重新加密钥的方法，该方法包括：

至少接收重建所述一组数据份所需的最小数目的数据份；

在不解密所述最小数目的数据份的情况下从所述最小数目的数据份重建所述一组数据份；以及

通过将重建的一组数据份与第二分裂密钥相关联而对重建的一组数据份重新加密钥。

11. 根据权利要求 10 的方法，还包括：

检索与所述最小数目的数据份关联的首标；

从检索到的首标提取密钥加密密钥；

利用密钥加密密钥对第二分裂密钥进行加密；以及

把加密的第二分裂密钥存储在重新加密钥后的数据份的首标内。

12. 根据权利要求 10 的方法，还包括把重新加密钥后的数据份中的至少一个存储在存储网络上。

13. 根据权利要求 11 的方法，其中，存储网络包括私有云、公有云、混合云、可移动存储装置和海量存储装置中的一个。

14. 一种用于在存储网络的文件系统上把存根与一组数据份相关联的方法，该方法包括：

从通过信息分散算法设置的加密数据产生所述一组数据份；

产生与所产生的数据份关联的一组存根，其中每个存根对应于一个相应的数据份，并且其中每个存根包括与相应数据份关联的信息；以及

把该组存根存储在存储网络上的位置。

15. 根据权利要求 14 的方法，其中，所述信息包括相应数据份的名称、相应数据份的创建日期、相应数据份的最后修改时间、指向相应数据份在文件系统内的位置的指针中的一个。

16. 根据权利要求 14 的方法，其中，存储网络包括与私有云、公有云、混合云、可移动存储装置和海量存储装置中的一个关联的一个或多个存储装置。

17. 根据权利要求 14 的方法，还包括：

接收观看与产生的数据份关联的信息的命令；

从存储网络上的所述位置检索所述存根；

从所述存根提取所述信息以创建数据份的文件系统；以及

显示数据份的文件系统。

18. 根据权利要求 14 的方法，其中存根被存储在产生的数据份的首标内，并且其中检索包括检索产生的数据份的首标。

19. 根据权利要求 18 的方法，其中，少于所有首标的首标被检索。

20. 根据权利要求 14 的方法，其中存根被存储在存根目录中，并且其中检索包括从存根目录检索存根。

21. 根据权利要求 14 的方法，还包括：

接收存储网络中用于存储存根的虚拟目录或物理目录的指示。

22. 根据权利要求 21 的方法，其中，所述指示是从用户接收的。

23. 一种用于安全数据处理的加速的协处理器加速装置，包括：

存储器，用于存储数据；

耦接到存储器的主处理器；和

耦接到主处理器和存储器的协处理器，被配置为执行专门的安全解析功能，所述安全解析功能包括加密数据、分裂数据和解密数据中的至少一种。

24. 根据权利要求 23 的装置，其中，分裂数据包括信息分散算法（IDA）的使用。

25. 根据权利要求 23 的装置，还包括耦接到协处理器的现场可编程门阵列。

26. 根据权利要求 25 的装置，其中，FPGA 执行对解析的数据进行加密或者对加密的数据进行解密中的至少一个。

27. 根据权利要求 23 的装置，其中，协处理器经由 PCIe 总线耦接到主处理器。

28. 根据权利要求 23 的装置，其中，协处理器经由 HT 总线耦接到主处理器。

29. 根据权利要求 23 的装置，其中，存储器包括用于主处理器的专门存储器。

30. 根据权利要求 23 的装置，其中，存储器包括用于协处理器的专门存储器。

31. 根据权利要求 23 的装置，其中，协处理器是独立磁盘冗余阵列（RAID）处理单元，其执行一个或多个 RAID 功能。

32. 一种用于使用便携式装置保护数据的方法，该方法包括：

至少部分地基于密钥从一组数据产生至少两个数据部分，其中所述至少两个数据部分和所述密钥足以重构该组数据；以及

把所述密钥存储在便携式装置上。

33. 根据权利要求 32 的方法，其中，便携式装置是可移动存储装置。

34. 根据权利要求 33 的方法，其中，所述可移动存储装置经由通用串行总线（USB）接口耦接到端用户装置。

35. 根据权利要求 32 的方法，还包括把至少一个产生的数据部分存储在便携式装置上。

36. 根据权利要求 32 的方法，其中，所述密钥是加密密钥、分裂密钥和认证密钥中的一个。

37. 根据权利要求 32 的方法，其中，所述至少两个数据部分是使用信息分散算法（IDA）和与 IDA 关联的分裂密钥产生的。

38. 一种用于使用便携式装置保护数据的方法，该方法包括：

至少部分地基于密钥从一组数据产生至少两个数据部分，其中所述至少两个数据部分和所述密钥足以重构该组数据；以及

把至少一个产生的数据部分存储在便携式装置上。

39. 根据权利要求 38 的方法，其中，便携式装置是可移动存储装置。

40. 根据权利要求 39 的方法，其中，所述可移动存储装置经由通用串行总线（USB）接口耦接到端用户装置。

41. 根据权利要求 38 的方法，还包括把所述密钥存储在便携式存储装置上。

42. 根据权利要求 38 的方法，其中，所述密钥是加密密钥、分裂密钥和认证密钥中的一个。

43. 根据权利要求 38 的方法，其中，所述至少两个数据部分是使用信息分散算法（IDA）和与 IDA 关联的分裂密钥产生的。

44. 一种用于保护要分裂并存储在存储网络上的文件的文件名的方法，该方法包括：

使用认证算法处理所述文件的文件名以获得认证值；以及

通过在存储网络上的份位置搜索具有与所述文件的认证值匹配的认证值的数据份的文件名，检索与所述文件对应的数据份。

45. 根据权利要求 44 的方法，还包括：

使用信息分散算法产生与认证的文件名关联的一个或多个数据份；以及

把产生的数据份存储在存储网络中的一个或多个数据份位置。

46. 根据权利要求 44 的方法，其中，存储网络包括私有云、公有云、混合云、可移动存储

装置和海量存储装置中的一个。

47. 根据权利要求 44 的方法，其中，所述认证算法是 HMAC-SHA256 算法。

48. 根据权利要求 44 的方法，还包括在进行所述处理前把附加信息附于所述文件的文件名末尾。

49. 根据权利要求 48 的方法，其中，所述附加信息包括与数据份位置关联的编号。

50. 一种用于保护要分裂并存储在存储网络上的文件的文件名的方法，该方法包括：

使用加密算法对所述文件的文件名进行加密；

使用信息分散算法产生与加密的文件名关联的一个或多个数据份；

把产生的数据份存储在存储网络中的一个或多个数据份位置；以及

通过解密产生的数据份之一的文件名，重新产生所述文件的文件名。

51. 根据权利要求 50 的方法，其中，存储网络包括私有云、公有云、混合云、可移动存储装置和海量存储装置中的一个。

52. 根据权利要求 50 的方法，其中，所述加密算法是 AES 算法。

53. 根据权利要求 50 的方法，还包括在进行所述加密前把附加信息附于所述文件的文件名末尾。

54. 根据权利要求 53 的方法，其中，所述附加信息包括与数据份位置关联的编号。

对移动中数据进行保护的系统和方法

[0001] 相关申请的交叉引用

[0002] 本申请要求于2010年3月31日提交的美国临时专利申请No.61/319,658以及于2010年4月1日提交的美国临时专利申请No.61/320,242的优先权。这些临时专利申请中的每个的内容通过在此引用而全部并入本文。

技术领域

[0003] 本发明一般涉及用于对移动中数据进行保护的系统和方法。可以与在共同拥有的美国专利No.7391865和在2005年10月25日提交的共同拥有的美国专利申请No.11/258839、2006年11月20日提交的美国专利申请No.11/602667、2007年11月7日提交的美国专利申请No.11/983355、2007年12月5日提交的美国专利申请No.11/999575、2008年4月18日提交的美国专利申请No.12/148365、2008年9月12日提交的美国专利申请No.12/209703、2009年1月7日提交的美国专利申请No.12/349897、2009年2月23日提交的美国专利申请No.12/391028、2010年5月19日提交的美国专利申请No.12/783276、2010年11月24日提交的美国专利申请No.12/953877、以及2011年1月27日提交的美国临时专利申请No.61/436991、2009年11月25日提交的美国临时专利申请No.61/264464、2010年3月31日提交的美国临时专利申请No.61/319658、2010年4月1日提交的美国临时专利申请No.61/320242、2010年5月28日提交的美国临时专利申请No.61/349560、2010年8月12日提交的美国临时专利申请No.61/373187、2010年8月18日提交的美国临时专利申请No.61/374950和2010年9月20日提交的美国临时专利申请No.61/384583中描述的其它系统和方法结合地使用本文所述的系统和方法。上述的每个在先提交的申请的全部公开通过在此引用而并入本文。

发明内容

[0004] 协作的需要要求企业共享其数据。该共享要求因为遗留烟囱管架构而复杂化，该烟囱管架构的维护昂贵且不会调整。这些复杂的基础设施因风险转移和灾难恢复要求和策略而更加受限制。此外，这些限制直接导致低下的资源利用率、昂贵的点产品以及不一致的信息共享。遗留烟囱管环境的关键驱动物是需要保护数据的保密性、可用性和完整性。因为该环境随着时间而进化，信息共享和协作由于安全性关注和脆弱性的增长而受到限制。随着时间经过，这些遗留环境需要大量的对等(ad-hoc)安全性修补，这进一步限制了信息共享和协作。然而，这些修补并不是针对数据可用性和数据安全性之间的权衡的根由提出的。

[0005] 现有的信息保障(Information Assurance，IA)方案是复杂的，难以调整并且易受安全脆弱性的影响。通过设计，这些方案无法提供数据安全性和数据可用性两者。基于这种IA方案的灾难恢复计划通常较差，很少有效地执行，维护昂贵，并且在企业共享的数据的量增长时难以调整。

[0006] 某些数据安全性方案(例如VPN和基于令牌的基础设施)是昂贵的并且包括对部署和维护两者的严重挑战。此外，一些产品仅解决作为目标的安全问题，但它们低效、昂贵、麻

烦并且难以管理。此外，这些方案没有提供对下述基本问题的端到端解决方案：使用一个或多个移动装置越来越多地访问云中的数据的位于远方的端用户的安全连接性和数据传送。

[0007] 转而使用云存储(“云”)的远程用户的数目的攀升也产生了云中的以及向或从云传输数据时的增加的数据安全性问题。具体地，这种云存储可以是公有的、私有的、安全的或其任意组合。此外，云存储可由多于一个的存储提供商提供。当遇到新的老练的数据安全威胁时，这些威胁对于个人用户和企业同样严重。远程用户需要使用云作为存储介质与其它人进行协作的灵活性，但是需要在不使数据暴露于安全风险的情况下实现。

[0008] 因此，需要既保护企业数据同时又提供对其的访问。另外，需要在没有服务中断且不关乎用户位置的情况下提供这种安全访问。事实上，需要一种易于部署、无需用户干预、不需要额外的硬件、高度安全且不损害生产率的端到端解决方案。事实上，需要提供一种密码系统，其安全性与用户位置无关，同时在数据处于移动中或者从一个位置转移到另一个位置时仍支持数据的安全性。

[0009] 因此，本发明的一个方面提供一种基于服务器(例如，SecurityFirst Corp. 的Bitfiler)的安全数据方案，该方案使数据可证地安全和可访问，同时消除了多个周边硬件和软件技术的需要。该基于服务器的方案致力于比特级的安全性。换言之，数据安全性直接包括或者交织在比特级的数据中。在一些实施例中，该基于服务器的方案可以是在Windows或Linux平台上运行的软件应用。在一些实施例中，通过在内核级操作，实现了性能和易用性的大幅改进。在一些实施例中，该基于服务器的方案使得能够建立企业的兴趣团体(Communities of Interest,COI)，其能够在硬件和软件两方面支持共用企业基础设施。因为安全性已被交织到数据中，可以在不损害数据安全性和访问控制的情况下使用该共用基础设施。在同一基础设施内并且在一个安全存储系统内，多个COI可以共存。对于该基于服务器的方案，没有法庭可分辨的数据存储在任何装置或介质上。该基于服务器的方案可以与现有的企业访问控制系统整合，允许简化的部署而无需改动当前建立的访问方案。

[0010] 在另一个方面，本发明的基于服务器的方案与硬件和软件无关。该基于服务器的方案适用于现有企业网络、存储和安全性方案。该基于服务器的方案还适用于任何协作、CRM和ERP应用。由该基于服务器的方案提供的内建安全性使得能够使用刚出现的成本效益高的技术和服务，诸如用于基于云的存储、基于云的计算和基于云的应用的基础设施。

[0011] 本发明的基于服务器的方案可以支持Security First Corp. 的SecureParser Extended™(SPx)核心技术。在一些实施例中，SecureParser SPx利用多因素秘密共享算法来提供防卫级安全性。数据被认证、加密(FIPS 140-2认证、符合Suite B)、分裂、被添加冗余位、进行完整性检查并再次加密，然后被发送到多个位置(本地的和/或地理上分散的例如在私有或公有云中的位置)。可以使用任何合适的信息分散算法(IDA)来分裂数据。数据在转移到存储位置的同时被隐藏，并且没有正确的访问证明的用户无法访问。

[0012] 本发明的另一方面包括一种重建存储在存储网络中的要保护的数据的一组数据份(share)中的第一子集的方法。该方法包括从安全存储网络检索数据份的第二子集。数据份的第二子集足以重构所述数据。该方法还包括对数据份的第二子集进行认证，并使用数据份的第一子集重建与所述一组数据份对应的加密数据。该方法还包括通过把所述加密数据分裂重新产生所述一组数据份并对重新产生的数据份进行重新认证。该方法还包括在存储网络中至少存储重新产生的数据份的第一子集。

[0013] 在一些实施例中，分裂包括使用信息分散算法。在一些实施例中，认证包括使用认证密钥。在一些实施例中，重新产生所述一组数据份包括使用分裂密钥。在一些实施例中，存储网络包括私有云、公有云、混合云、可移动存储装置和海量存储装置中的一个。在一些实施例中，首标对应于少于各个份的全部首标。

[0014] 在一些实施例中，重新认证包括使用与用于所述认证的第一认证密钥不同的第二认证密钥。在一些实施例中，该过程包括从安全存储网络检索与数据份的第二子集关联的首标，从检索到的首标中提取密钥加密密钥，用该密钥加密密钥对第二认证密钥加密，并把加密的第二认证密钥在重新产生的数据份的首标内存储在存储网络中。

[0015] 在一些实施例中，重新产生包括使用与用于产生所述一组数据份的第一分裂密钥不同的第二分裂密钥。在一些实施例中，该过程包括从安全存储网络检索与数据份的第二子集关联的首标，从检索到的首标中提取密钥加密密钥，用该密钥加密密钥对第二分裂密钥加密，并把加密的第二分裂密钥在重新产生的数据份的首标内存储在存储网络中。

[0016] 在另一方面，本发明涉及一种用于保护数据的系统。该系统包括用于存储数据的存储器；主处理器，耦接到存储器，被配置为执行数据分裂和数据加密中的至少一个；和协处理器，耦接到主处理器和存储器。该协处理器被配置为执行专门的安全解析功能，包括对解析的数据进行加密或者对加密的数据进行解密中的至少一个。在一些实施例中，该系统包括耦接到协处理器的现场可编程门阵列。该 FPGA 执行对解析的数据进行加密或者对加密的数据进行解密中的至少一个。在一些实施例中，该协处理器经由 PCIe 总线耦接到主处理器。在一些实施例中，该协处理器经由 HT 总线耦接到主处理器。

[0017] 在另一方面，本发明涉及一种使用便携式装置来保护数据的方法。该方法包括以下步骤：至少部分地基于密码密钥，从一组数据产生至少两个数据部分，以及把该密钥存储在便携式装置上。所述两个数据部分和所述密钥足以重构所述一组数据。在一些实施例中，便携式装置是可移动存储装置。在一些实施例中，该可移动存储装置经由通用串行总线(USB)接口耦接到端用户装置。在一些实施例中，该方法还包括把所述至少两个数据部分存储在便携式装置上。

[0018] 在另一方面，本发明涉及一种使用便携式装置来保护数据的方法。该方法包括以下步骤：至少部分地基于密码密钥，从一组数据产生至少两个数据部分，以及把产生的数据部分中的至少一部分存储在便携式装置上。所述两个数据部分和所述密钥足以重构所述一组数据。在一些实施例中，便携式装置是可移动存储装置。在一些实施例中，该可移动存储装置经由通用串行总线(USB)接口耦接到端用户装置。在一些实施例中，该方法还包括把所述密钥存储在便携式装置上。

[0019] 在一些实施例中，一个或多个密码密钥可存储在诸如 USB 存储装置的用户装置上。这些密码密钥可用于对存储在端用户装置自身上或其它地方(例如公有或私有云存储中)的数据进行加密或解密。例如，用户可把密码密钥存储在 USB 存储装置上并使用该密钥对远程地存储在由 Dropbox 提供的公有云中的加密的数据份进行解密。

[0020] 在一些实施例中，为了使得能够在多个不同的端用户装置的每一个进行数据观看和 / 或重构，可将一个或多个密码密钥和 / 或一个或多个数据份存储在便携式用户装置(诸如 USB 存储装置)上。另外，一个或多个数据份也可被存储在云存储装置上。因此，拥有该便携式用户装置的用户可从不同的端用户装置访问该便携式用户装置和 / 或从分散在便

携式用户装置(如果需要的话还有云存储装置)上的份重建数据。

[0021] 在另一方面,本发明涉及一种对从使用第一分裂密钥通过信息分散算法设置的加密数据产生的一组数据份进行重建的方法。该方法包括至少接收重建所述一组数据份所需的最小数目的数据份,在不解密该最小数目的数据份的情况下从该最小数目的数据份重建所述一组数据份。在一些实施例中,响应于确定所述一组数据份已被损坏而执行重建。在一些实施例中,该最小数目的数据份与第一认证密钥关联,并且重建包括把重建的所述一组数据份与第二认证密钥关联。

[0022] 在一些实施例中,该方法还包括以下步骤:检索与该最小数目的数据份关联的首标,从检索到的首标中提取密钥加密密钥,用该密钥加密密钥对第二认证密钥进行加密,并在重建的数据份的首标内恢复加密的第二认证密钥。在一些实施例中,使用与第一分裂密钥不同的第二分裂密钥重建该最小数目的数据份。在一些实施例中,该方法还包括检索与该最小数目的数据份关联的首标,从检索到的首标中提取密钥加密密钥,用该密钥加密密钥对第二分裂密钥进行加密,并在重建的数据份的首标内恢复加密的第二分裂密钥。在一些实施例中,该方法还包括把重建的数据份中的至少一个存储在存储网络上的步骤。在一些实施例中,存储网络包括私有云、公有云、混合云、可移动存储装置和海量存储装置中的一个。

[0023] 在另一方面,本发明涉及一种在存储网络的文件系统上把存根(stubs)与一组数据份相关联的方法。该方法包括从通过信息分散算法设置的加密数据产生一组数据份,并且产生与产生的数据份关联的一组存根。每个存根分别对应于一个数据份,并且每个存根包括与相应的数据份关联的信息。所述一组存根存储在存储网络上的位置。该信息包括相应数据份的名称、相应数据份的创建日期、相应数据份的最后修改时间、指向相应数据份在文件系统内的位置的指针中的一个。存储网络包括与私有云、公有云、混合云、可移动存储装置和海量存储装置中的一个关联的一个或多个存储装置。在一些实施例中,该方法还包括以下步骤:接收观看与产生的数据份关联的信息的命令,从存储网络上的所述位置检索存根,从存根提取信息以创建数据份的文件系统,并且显示数据份的文件系统。在一些实施例中,存根存储在产生的数据份的首标内,并且检索包括检索产生的数据份的首标。在一些实施例中,检索少于全部首标的首标。在一些实施例中,存根存储在存根目录中,并且检索包括从存根目录检索存根。在一些实施例中,该方法还包括接收要存储存根的虚拟或物理目录的指示。在一些实施例中,从用户接收该指示。

附图说明

[0024] 在下文中结合附图更加详细描述本发明,这些附图旨在例示而非限制本发明,在附图中:

[0025] 图 1 示出了根据本发明的实施例的各方面的密码系统的框图;

[0026] 图 2 示出了根据本发明的实施例的各方面的图 1 的信任引擎的框图;

[0027] 图 3 示出了根据本发明的实施例的各方面的图 2 的事务引擎的框图;

[0028] 图 4 示出了根据本发明的实施例的各方面的图 2 的储存器(depository)的框图;

[0029] 图 5 示出了根据本发明的实施例的各方面的图 2 的认证引擎的框图;

[0030] 图 6 示出了根据本发明的实施例的各方面的图 2 的密码引擎的框图;

[0031] 图 7 示出了根据本发明的另一个实施例的各方面的储存器系统的框图；

[0032] 图 8 示出了根据本发明的实施例的各方面的数据分裂过程的流程图；

[0033] 图 9A 示出了根据本发明的实施例的各方面的登记过程的数据流；

[0034] 图 9B 示出了根据本发明的实施例的各方面的互用性过程的流程图；

[0035] 图 10 示出了根据本发明的实施例的各方面的认证过程的数据流；

[0036] 图 11 示出了根据本发明的实施例的各方面的签名过程的数据流；

[0037] 图 12 示出了根据本发明的另一个实施例的各方面的加密 / 解密过程的数据流；

[0038] 图 13 示出了根据本发明的另一个实施例的各方面的信任引擎系统的简化框图；

[0039] 图 14 示出了根据本发明的另一个实施例的各方面的信任引擎系统的简化框图；

[0040] 图 15 示出了根据本发明的实施例的各方面的图 14 的冗余模块的框图；

[0041] 图 16 示出了根据本发明的一个方面的评估认证的过程；

[0042] 图 17 示出了根据本发明的如在图 16 中所示的一个方面向认证分配值的过程；

[0043] 图 18 示出了在如图 17 所示的本发明的一个方面中执行信任仲裁的过程；以及

[0044] 图 19 示出了根据本发明的实施例的各方面的用户与卖方之间的样本事务，其中，初始基于 web 的合同导致由双方签名的销售合同。

[0045] 图 20 示出了具有向用户系统提供安全性功能的密码服务提供商模块的样本用户系统。

[0046] 图 21 示出了在加密以及加密主密钥与数据存储在一起的情况下解析、分裂和 / 或分离数据的过程。

[0047] 图 22 示出了在加密以及加密主密钥与数据分离地存储的情况下解析、分裂和 / 或分离数据的过程。

[0048] 图 23 示出了在加密以及加密主密钥与数据存储在一起的情况下解析、分裂和 / 或分离数据的中间密钥过程。

[0049] 图 24 示出了在加密以及加密主密钥与数据分离地存储的情况下解析、分裂和 / 或分离数据的中间密钥过程。

[0050] 图 25 示出了小工作组对本发明的密码方法和系统的利用。

[0051] 图 26 是采用根据本发明的一个实施例的安全数据解析器的例示性物理令牌安全性系统的框图。

[0052] 图 27 是根据本发明的一个实施例的将安全数据解析器集成到系统中的例示性布置的框图。

[0053] 图 28 是根据本发明的一个实施例的例示性移动中数据系统的框图。

[0054] 图 29 是根据本发明的一个实施例的另一个例示性移动中数据系统的框图。

[0055] 图 30-32 是根据本发明的一个实施例的集成了安全数据解析器的例示性系统的框图。

[0056] 图 33 是根据本发明的一个实施例的解析和分裂数据的例示性过程的处理流程图。

[0057] 图 34 是根据本发明的一个实施例的将数据部分恢复成原始数据的例示性过程的处理流程图。

[0058] 图 35 是根据本发明的一个实施例的以比特级分裂数据的例示性过程的处理流程

图。

[0059] 图 36 是根据本发明的一个实施例的可以按任何合适组合(具有任何合适添加、删除或修改)使用的例示性步骤和特征的处理流程图。

[0060] 图 37 是根据本发明的一个实施例的可以按任何合适组合(具有任何合适添加、删除或修改)使用的例示性步骤和特征的处理流程图。

[0061] 图 38 是根据本发明的一个实施例的可以按任何合适组合(具有任何合适添加、删除或修改)使用的在份内存储密钥和数据成分的简化框图。

[0062] 图 39 是根据本发明的一个实施例的可以按任何合适组合(具有任何合适添加、删除或修改)使用的使用工作组密钥在份内存储密钥和数据成分的简化框图。

[0063] 图 40A 和 40B 是根据本发明的一个实施例的可以按任何合适组合(具有任何合适添加、删除或修改)使用的针对移动中数据的首标产生和数据分裂的简化和例示性处理流程图。

[0064] 图 41 是根据本发明的一个实施例的可以按任何合适组合(具有任何合适添加、删除或修改)使用的例示性份格式的简化框图。

[0065] 图 42 是根据本发明的一个实施例的把安全数据解析器整合到与云计算资源连接的系统中的例示性结构的框图。

[0066] 图 43 是根据本发明的一个实施例的把安全数据解析器整合到通过云发送数据的系统中的例示性结构的框图。

[0067] 图 44 是根据本发明的一个实施例的使用安全数据解析器来保护云中的数据服务的例示性结构的框图。

[0068] 图 45 是根据本发明的一个实施例的使用安全数据解析器来保护云中的数据存储的例示性结构的框图。

[0069] 图 46 是根据本发明的一个实施例的使用安全数据解析器来保护网络访问控制的例示性结构的框图。

[0070] 图 47 是根据本发明的一个实施例的使用安全数据解析器来保护高性能计算资源的例示性结构的框图。

[0071] 图 48 是根据本发明的一个实施例的使用安全数据解析器来保护多个存储装置中的数据存储的例示性结构的示意图。

[0072] 图 49 是根据本发明的一个实施例的使用安全数据解析器来保护多个私有和公有云中的数据存储的例示性结构的示意图。

[0073] 图 50 是根据本发明的一个实施例的使用安全数据解析器经由公共互联网来保护多个私有和公有云中的数据存储的例示性结构的示意图。

[0074] 图 51 是根据本发明的一个实施例的使用安全数据解析器来保护用户的可移动存储装置中的数据存储的例示性结构的示意图。

[0075] 图 52 是根据本发明的一个实施例的使用安全数据解析器来保护多个用户存储装置中的数据存储的例示性结构的示意图。

[0076] 图 53 是根据本发明的一个实施例的使用安全数据解析器来保护多个公有和私有云以及至少一个用户存储装置中的数据存储的例示性结构的示意图。

[0077] 图 54 是根据本发明的一个实施例的用于安全数据解析器的协处理器加速装置的

示意图。

[0078] 图 55 是根据本发明的一个实施例的针对安全数据解析器使用图 54 的协处理器加速装置的例示性加速过程的第一处理流程图。

[0079] 图 56 是根据本发明的一个实施例的针对安全数据解析器使用图 54 的协处理器加速装置的例示性加速过程的第二处理流程图。

[0080] 图 57 示出了根据本发明的例示性实施例的把数据分裂成 N 份并存储的过程。

[0081] 图 58 示出了根据本发明的例示性实施例的对数据份进行重建和 / 或重新施加密钥(re-key)的过程。

具体实施方式

[0082] 本发明的一个方面是提供一种密码系统,在该密码系统中,一个或多个安全服务器或信任引擎存储密码密钥和用户认证数据。该系统可以跨云中的一个或多个存储装置存储数据。云可以包括私有存储装置(仅一组特定用户可访问)或公有存储装置(向存储提供商进行了订阅的任何用户组都可访问)。

[0083] 用户通过对信任引擎的网络访问,访问传统密码系统的功能,然而,信任引擎没有发布实际密钥和其它认证数据,因此这些密钥和数据仍是安全的。密钥和认证数据的这种服务器中心存储提供了用户无关的安全性、移植性、可用性和直率性。

[0084] 因为用户能够确信或信任密码系统来执行用户和文档认证以及其它密码功能,所以多种多样的功能可以包括在该系统内。例如,通过例如对协议参与者进行认证,代表或针对参与者对该协议进行数字签名,并且存储由每个参与者数字签名的协议的记录,信任引擎提供商可以确保不会出现协议抵赖。此外,该密码系统可以监视协议并且例如基于价格、用户、卖方、地理位置、使用地点等来确定应用不同程度的认证。

[0085] 为了便于完全理解本发明,具体实施方式的其余部分参照附图描述本发明,其中,相同元素始终由相同标号进行表示。

[0086] 图 1 示出了根据本发明的实施例的各方面的密码系统 100 的框图。如图 1 所示,密码系统 100 包括通过通信链路 125 进行通信的用户系统 105、信任引擎 110、认证机构 115 和卖方系统 120。

[0087] 根据本发明的一个实施例,用户系统 105 包括具有一个或多个微处理器(例如,基于 Intel 的处理器)的传统通用计算机。此外,用户系统 105 包括适当的操作系统,例如能够包括图形或窗口的操作系统(例如,Windows、Unix、Linux 等)。如图 1 所示,用户系统 105 可以包括生物测定装置 107。生物测定装置 107 可以有利地获取用户的生物测定并且将获取的生物测定传送给信任引擎 110。根据本发明的一个实施例,生物测定装置可有利地包括具有与在以下文献中公开的类似的属性和特征的装置 :1997 年 9 月 5 日提交的题目为“RELIEF OBJECTIMAGE GENERATOR”的美国专利申请 No.08/926277、2000 年 4 月 26 日提交的题目为“IMAGING DEVICE FOR A RELIEF OBJECT ANDSYSTEM AND METHOD OF USING THE IMAGE DEVICE”的美国专利申请 No.09/558634、1999 年 11 月 5 日提交的题目为“RELIEF OBJECTSENSOR ADAPTOR”的美国专利申请 No.09/435011 和 2000 年 1 月 5 日提交的题目为“PLANAR OPTICAL IMAGE SENSOR AND SYSTEMFOR GENERATING AN ELECTRONIC IMAGE OF A RELIEF OBJECTFOR FINGERPINT READING”的美国专利申请 No.09/477943,上述所有的美国

专利申请由当前受让人拥有并且通过引用并入本文。

[0088] 此外，用户系统 105 可以通过传统的服务提供商(例如，拨号、数字用户线(DSL)、线缆调制解调器、光纤连接等)连接到通信链路 125。根据另一个实施例，用户系统 105 通过网络连接性(例如，局域网或广域网)连接通信链路 125。根据一个实施例，操作系统包括 TCP/IP 栈，该 TCP/IP 栈处理在通信链路 125 上传递的所有的出入消息通信。

[0089] 尽管参照上述实施例公开了用户系统 105，但是本发明不限于此。相反，熟练技术人员从这里的公开可以识别用户系统 105 的大量的替代实施例，包括能够发送或从另一个计算机系统接收信息的几乎任何计算装置。例如，用户系统 105 可以包括但不限于能够与通信链路 125 进行交互的计算机工作站、交互式电视、交互亭、个人移动计算装置(例如，数字助理、移动电话、膝上型电脑等)、无线通信装置、智能卡、嵌入式计算装置等。在这些替代系统中，操作系统很可能不同并且针对特定装置进行改动。然而，根据一个实施例，操作系统有利地继续提供与通信链路 125 建立通信所需的适当的通信协议。

[0090] 图 1 示出了信任引擎 110。根据一个实施例，信任引擎 110 包括用于访问和存储敏感信息的一个或多个安全服务器，敏感信息可以是任何类型或形式的数据，例如是但不限于文本、音频、视频、用户认证数据以及公共和私有密码密钥。根据一个实施例，认证数据包括被设计为唯一识别密码系统 100 的用户的数据。例如，认证数据可以包括用户识别号、一个或多个生物测定、以及由信任引擎 110 或用户产生但是在登记时由用户初始回答的一系列提问和回答。上述提问可以包括人口数据(例如，出生地、地址、周年纪念等)、个人数据(例如，母亲未婚时的名字、喜欢的冰激凌等)、或被设计为唯一识别用户的其它数据。信任引擎 110 将与当前事务关联的用户的认证数据与先前(例如在登记时)设置的认证数据进行比较。信任引擎 110 可以有利地要求用户在每次事务时生成认证数据，或者信任引擎 110 可以有利地允许用户定期地(例如，在一串事务的开始时或者在登录到特定卖方网站时)生成认证数据。

[0091] 根据用户生成生物测定数据的实施例，用户向生物测定装置 107 提供身体特征，例如但不限于面部扫描、手扫描、耳扫描、虹膜扫描、视网膜扫描、血管模式、DNA、指纹、笔迹或语音。生物测定装置有利地生成身体特征的电子模式或生物测定。为了登记或认证，电子模式通过用户系统 105 传送至信任引擎 110。

[0092] 一旦用户生成了适当的认证数据并且信任引擎 110 确定该认证数据(当前认证数据)与在登记时设置的认证数据(登记认证数据)之间的明确匹配，信任引擎 110 向用户提供完整的密码功能。例如，正确认证的用户可以有利地采用信任引擎 110 执行哈希处理、数字签名、加密和解密(常总称为加密)、建立或分布数字证书等等。然而，密码功能中使用的私有密码密钥在信任引擎 110 之外将不可用，从而确保了密码密钥的完整性。

[0093] 根据一个实施例，信任引擎 110 产生并存储密码密钥。根据另一个实施例，至少一个密码密钥与每个用户关联。此外，当密码密钥包括公钥技术时，在信任引擎 110 内产生但不从其发放与用户关联的每个私钥。因此，只要用户可以访问信任引擎 110，用户就可以使用他或她的私钥或公钥执行密码功能。有利的是，这种远程访问使用户可以通过实际任何互联网连接(例如，蜂窝和卫星电话、信息亭、膝上型电脑、宾馆房间等)保持完全移动并访问密码功能。

[0094] 根据另一个实施例，信任引擎 110 使用针对信任引擎 110 产生的密钥对，执行密码

功能。根据这个实施例，信任引擎 110 首先对用户进行认证，并且在用户正确地生成与登记认证数据匹配的认证数据后，信任引擎 110 使用它自身的密码密钥对，代表认证的用户执行密码功能。

[0095] 熟练技术人员将从这里的公开认识到，密码密钥可以有利地包括对称密钥、公钥和私钥的全部或一些。此外，熟练技术人员将从这里的公开认识到，可以采用可从商业技术获得的大量的算法（例如，RSA、ELGAMAL 等）来实现上述密钥。

[0096] 图 1 还示出了认证机构 115。根据一个实施例，认证机构 115 可有利地包括发放数字证书的信任第三方组织或公司，例如 VeriSign、Baltimore、Entrust 等。信任引擎 110 可有利地通过一个或多个传统的数字证书协议（例如，PKCS10）向认证机构 115 发送对数字证书的请求。作为响应，认证机构 115 将发放多个不同协议中的一个或多个（例如，PKCS7）的数字证书。根据本发明的一个实施例，信任引擎 110 从几个或所有的主要认证机构 115 请求数字证书，从而使得信任引擎 110 可访问与任何请求方的证书标准对应的数字证书。

[0097] 根据另一个实施例，信任引擎 110 在内部执行证书发放。在这个实施例中，当证书被请求时（例如，在密钥产生时）或者在请求时请求的证书标准中，信任引擎 110 可以访问用于产生证书的证书系统和 / 或可以在内部产生证书。在下文中将更加详细地公开信任引擎 110。

[0098] 图 1 还示出了卖方系统 120。根据一个实施例，卖方系统 120 有利地包括 Web 服务器。典型的 Web 服务器可以使用几种互联网标记语言或文档格式标准（例如，超文本标记语言（HTML）或可扩展标记语言（XML））之一在互联网上提供内容。Web 服务器从如 Netscape 和 Internet Explorer 的浏览器接受请求，然后返回适当的电子文档。多个服务器或客户端技术能够用于提高 Web 服务器的能力以超越它的传递标准电子文档的能力。例如，这些技术包括公共网关接口（CGI）脚本、安全套接字层（SSL）安全性和动态服务器页面（ASP）。卖方系统 120 可有利地提供与商业、个人、教育或其它事务有关的电子内容。

[0099] 尽管参照上述实施例公开了卖方系统 120，但是本发明不限于此。相反，熟练技术人员将从这里的公开认识到，卖方系统 120 可有利地包括参照用户系统 105 描述的装置中的任何一个或者它们的组合。

[0100] 图 1 还示出了连接用户系统 105、信任引擎 110、认证机构 115 和卖方系统 120 的通信链路 125。根据一个实施例，通信链路 125 优选包括互联网。本文所用的互联网是计算机的全球网络。本领域普通技术人员公知的互联网的结构包括网络骨干和从骨干分支的网络。这些分支继而具有从它们分支的网络，如此类推。路由器在网络级之间移动信息包，然后从网络到网络移动信息包，直到包到达它的目的地附近。目的地网络的主机将信息包从该目的地导向适当的终端或节点。在一个有利的实施例中，互联网路由集线器包括使用现有技术公知的传输控制协议 / 互联网协议（TCP/IP）的域名系统（DNS）服务器。路由集线器经由高速通信链路连接到一个或多个其它路由集线器。

[0101] 互联网的一个流行部分是万维网。万维网包含不同的计算机，这些计算机存储能够显示图形和文本信息的文档。在万维网上提供信息的计算机通常称作“网站”。网站由具有关联的电子页面的互联网地址定义。电子页面能够通过统一资源定位符（URL）进行识别。通常，电子页面是组织文本、图形图像、音频、视频等的呈现的文档。

[0102] 尽管在针对其优选实施例公开了通信链路 125，但是本领域普通技术人员将从这

里的公开认识到，通信链路 125 可以包括各种交互通信链路。例如，通信链路 125 可以包括交互电视网络、电话网络、无线数据传输系统、双向线缆系统、定制的私有或公共计算机网络、交互亭网络、自动柜员机网络、直接链路、卫星或蜂窝网络等。

[0103] 图 2 示出了根据本发明的实施例的各方面的图 1 的信任引擎 110 的框图。如图 2 所示，信任引擎 110 包括事务引擎 205、储存器 210、认证引擎 215 和密码引擎 220。根据本发明的一个实施例，信任引擎 110 还包括海量存储器 225。进一步如图 2 所示，事务引擎 205 与储存器 210、认证引擎 215 和密码引擎 220 以及海量存储器 225 进行通信。此外，储存器 210 与认证引擎 215、密码引擎 220 和海量存储器 225 进行通信。此外，认证引擎 215 与密码引擎 220 进行通信。根据本发明的一个实施例，上述通信中的一些或全部可有利地包括将 XML 文档发送至与接收装置对应的 IP 地址。如上所述，有利的是，XML 文档使设计者可以建立他们自己定制的文档标记，从而实现应用之间以及组织之间的数据的定义、传输、验证和解释。此外，上述通信中的一些或全部可包括传统的 SSL 技术。

[0104] 根据一个实施例，事务引擎 205 包括例如可从 Netscape、Microsoft、Apache 等获得的传统 Web 服务器的数据路由装置。例如，Web 服务器可以有利地从通信链路 125 接收输入数据。根据本发明的一个实施例，输入数据被寻址至信任引擎 110 的前端安全系统。例如，前端安全系统可以有利地包括防火墙、搜索已知攻击概况的入侵检测系统、和 / 或病毒扫描器。在通过了前端安全系统后，数据由事务引擎 205 接收并且被路由至储存器 210、认证引擎 215、密码引擎 220 和海量存储器 225 之一。此外，事务引擎 205 监视来自认证引擎 215 和密码引擎 220 的输入数据，并且通过通信链路 125 将该数据路由至特定系统。例如，事务引擎 205 可有利地向用户系统 105、认证机构 115 或卖方系统 120 路由该数据。

[0105] 根据一个实施例，使用传统的 HTTP 路由技术，例如采用 URL 或统一资源指示符 (URI) 来路由数据。URI 与 URL 类似，然而，URI 通常指示文件或动作 (例如，可执行文件、脚本等) 的源。因此，根据一个实施例，用户系统 105、认证机构 115、卖方系统 120 和信任引擎 210 的部件有利地包括通信 URL 或 URI 内的充足数据，以供事务引擎 205 在整个密码系统内正确地路由数据。

[0106] 尽管参照其优选实施例公开了数据路由，但是熟练技术人员将认识到大量的可能的数据路由方案或策略。例如，XML 或其它数据包可以有利地进行拆包并且通过它们的格式、内容等进行识别，从而使得事务引擎 205 可以在整个信任引擎 110 内正确地路由数据。此外，熟练技术人员将认识到，有利的是，例如当通信链路 125 包括局域网时，数据路由可进行改动以适应符合特定网络系统的数据传输协议。

[0107] 根据本发明的另一个实施例，事务引擎 205 包括传统的 SSL 加密技术，从而在特定通信期间，通过事务引擎 205，上述系统可以对它们自身进行认证，反之亦然。本发明中使用的术语“1/2SSL”是指服务器 (但客户机不必) 进行 SSL 认证的通信，术语“全 SSL”是指客户机和服务器均进行 SSL 认证的通信。当本公开使用术语“SSL”时，通信可以包括 1/2SSL 或全 SSL。

[0108] 由于事务引擎 205 将数据路由至密码系统 100 的各个部件，所以事务引擎 205 可有利地建立审计索引 (audit trail)。根据一个实施例，审计索引包括在整个密码系统 100 内由事务引擎 205 进行路由的数据的至少类型和格式的记录。这种审计数据可有利地存储在海量存储器 225 中。

[0109] 图 2 还示出了储存器 210。根据一个实施例，储存器 210 包括一个或多个数据存储设施，例如目录服务器、数据库服务器等。如图 2 所示，储存器 210 存储密码密钥和登记认证数据。密码密钥可有利地对应于信任引擎 110 或者密码系统 100 的用户（例如，用户或卖方）。登记认证数据可有利地包括被设计用于唯一识别用户的数据，例如用户 ID、口令、提问的回答、生物测定数据等。有利的是，可以在用户登记时或者在另一个替代的以后时间获取这个登记认证数据。例如，信任引擎 110 可以包括登记认证数据的周期性或其它的更新或重新发放。

[0110] 根据一个实施例，从事务引擎 205 到认证引擎 215 和密码引擎 220 的通信以及从认证引擎 215 和密码引擎 220 到事务引擎 205 的通信包括安全通信（例如，传统的 SSL 技术）。此外，如上所述，可以使用 URL、URI、HTTP 或 XML 文档传送到达和来自储存器 210 的通信的数据，有利地在上述任何一个内部嵌入了数据请求和格式。

[0111] 如上所述，储存器 210 可有利地包括多个安全数据存储设施。在这种实施例中，安全数据存储设施可被构造为使得对一个个体数据存储设施的安全性的危害将不会危害存储在其中的密码密钥或认证数据。例如，根据这个实施例，密码密钥和认证数据被进行数学运算，从而在统计学上充分地对存储在每个数据存储设施中的数据进行随机化。根据一个实施例，个体数据存储设施的数据的随机化使得该数据无法被破译。因此，对个体数据存储设施的危害仅仅生成随机化的无法译解的数字，并且不会危害作为整体的任何密码密钥或认证数据的安全性。

[0112] 图 2 还示出了包括认证引擎 215 的信任引擎 110。根据一个实施例，认证引擎 215 包括数据比较器，该数据比较器被构造为将来自事务引擎 205 的数据与来自储存器 210 的数据进行比较。例如，在认证过程中，用户将当前认证数据提供给信任引擎 110 从而事务引擎 205 接收当前认证数据。如上所述，事务引擎 205 识别优选在 URL 或 URI 中的数据请求，并且将认证数据路由至认证引擎 215。此外，当被请求时，储存器 210 将与用户对应的登记认证数据转发至认证引擎 215。因此，认证引擎 215 具有当前认证数据和登记认证数据以进行比较。

[0113] 根据一个实施例，到达认证引擎的通信包括安全通信（例如，SSL 技术）。此外，安全性（例如，使用公钥技术的超级加密）可以设置在信任引擎 110 部件内。例如，根据一个实施例，用户用认证引擎 215 的公钥对当前认证数据进行加密。此外，储存器 210 还用认证引擎 215 的公钥对登记认证数据进行加密。这样，仅有认证引擎的私钥能够用于对传输进行解密。

[0114] 如图 2 所示，信任引擎 110 还包括密码引擎 220。根据一个实施例，密码引擎包括密码处理模块，该模块被构造为有利地提供传统的密码功能（例如，公钥基础设施（PKI）功能）。例如，密码引擎 220 可以有利地向密码系统 100 的用户发放公钥和私钥。以这种方式，在密码引擎 220 处产生密码密钥并且将其转发至储存器 210，从而在信任引擎 110 之外至少无法获得私有密码密钥。根据另一个实施例，密码引擎 220 至少对私有密码密钥数据进行随机化和分裂，从而仅仅存储随机化的分裂数据。与登记认证数据的分裂类似，分裂过程确保在密码引擎 220 之外无法获得存储的密钥。根据另一个实施例，密码引擎的功能可以与认证引擎 215 组合并且由认证引擎 215 执行。

[0115] 根据一个实施例，到达和来自密码引擎的通信包括安全通信，诸如 SSL 技术。此

外，可有利地采用 XML 文档以传送数据和 / 或进行密码功能请求。

[0116] 图 2 还示出了具有海量存储 225 的信任引擎 110。如上所述，事务引擎 205 保持与审计索引对应的数据并且将这个数据存储在海量存储器 225 中。类似地，根据本发明的一个实施例，储存器 210 保持与审计索引对应的数据并且将这个数据存储在海量存储装置 225 中。由于审计索引数据包括由储存器 210 接收到的请求的记录及其响应，所以储存器审计索引数据与事务引擎 205 的审计索引数据类似。此外，海量存储器 225 可用于存储在内部包含了用户的公钥的数字证书。

[0117] 尽管参照其优选实施例和替代实施例公开了信任引擎 110，但是本发明不限于此。相反，熟练技术人员将从这里的公开中认识到信任引擎 110 的大量替代。例如，信任引擎 110 可有利地仅仅执行认证，或者仅仅执行例如数据加密和解密的一些或所有的密码功能。根据这些实施例，可有利地去除认证引擎 215 和密码引擎 220 之一，从而为信任引擎 110 建立更加简单的设计。此外，密码引擎 220 还可以与认证机构进行通信从而在信任引擎 110 内实现认证机构。根据另一个实施例，信任引擎 110 可以有利地执行认证以及例如数字签名的一个或多个密码功能。

[0118] 图 3 示出了根据本发明的实施例的各方面的图 2 的事务引擎 205 的框图。根据这个实施例，事务引擎 205 包括具有处理线程和侦听线程的操作系统 305。有利的是，操作系统 305 可与在例如可从 Apache 得到的 Web 服务器的传统高容量服务器中发现的操作系统类似。侦听线程针对输入数据流，监视来自通信链路 125、认证引擎 215 和密码引擎 220 之一的输入通信。处理线程识别输入数据流的特定数据结构(例如，上述数据结构)，从而将输入数据路由至通信链路 125、储存器 210、认证引擎 215、密码引擎 220 或海量存储器 225 之一。如图 3 所示，有利的是，通过例如 SSL 技术可以保护输入和输出数据。

[0119] 图 4 示出了根据本发明的实施例的各方面的图 2 的储存器 210 的框图。根据这个实施例，储存器 210 包括一个或多个轻量目录访问协议(LDAP)服务器。可以从各种制造商(例如，Netscape、ISO 等)获得 LDAP 目录服务器。图 4 还示出了目录服务器优选存储与密码密钥对应的数据 405 和与登记认证数据对应的数据 410。根据一个实施例，储存器 210 包括单个逻辑存储结构，用于将认证数据和密码密钥数据索引至唯一用户 ID。该单个逻辑存储结构优选包括确保存储在其中的数据的高度信任或安全性的机制。例如，储存器 210 的物理位置可有利地包括大量的传统安全性措施，诸如受限雇员访问、现代监视系统等。除了物理安全性以外或者替代物理安全性，计算机系统或服务器可以有利地包括保护存储的数据的软件方案。例如，储存器 210 可有利地建立并存储与采取的动作的审计索引对应的数据 415。此外，有利的是，可以用与传统的 SSL 技术结合的公钥加密，对输入和输出通信进行加密。

[0120] 根据另一个实施例，储存器 210 可以包括不同的且物理分离的数据存储设施，如进一步参照图 7 所公开的。

[0121] 图 5 示出了根据本发明的实施例的各方面的图 2 的认证引擎 215 的框图。与图 3 的事务引擎 205 类似，认证引擎 215 包括操作系统 505，操作系统 505 至少具有传统的 Web 服务器(例如，可从 Apache 获得的 Web 服务器)的变型版本的侦听和处理线程。如图 5 所示，认证引擎 215 包括对至少一个私钥 510 的访问。有利的是，私钥 510 可用于例如对来自事务引擎 205 或储存器 210 的数据进行解密，该数据用认证引擎 215 的对应公钥进行了加

密。

[0122] 图5还示出了包括比较器515、数据分裂模块520和数据组装模块525的认证引擎215。根据本发明的优选实施例，比较器515包括能够比较与上述生物测定认证数据有关的潜在复杂模式的技术。该技术可以包括用于模式比较(例如，表示指纹模式或语音模式的模式比较)的硬件、软件或者组合方案。此外，根据一个实施例，认证引擎215的比较器515可有利地比较文档的传统哈希值以呈现比较结果。根据本发明的一个实施例，比较器515包括对该比较应用启发法(heuristics)530。有利的是，启发法530可以应对围绕认证尝试的境况(例如，时间、IP地址或子网掩码、购买概况、电子邮件地址、处理器序列号或ID等)。

[0123] 此外，生物测定数据比较的性质会导致从当前生物测定认证数据与登记数据的匹配产生变化的置信度。例如，与可仅仅返回肯定或否定匹配的传统口令不同，指纹可被确定为部分匹配，例如90%匹配、75%匹配或10%匹配，而非简单地为正确或不正确。诸如声纹(voice print)分析或面部识别的其它生物测定识别法会共有这个概率认证的属性，而非绝对认证。

[0124] 当利用这种概率认证进行工作时或者在认为认证低于绝对可靠的其它情况下，期望应用启发法530以确定设置的认证的置信水平是否高到足够对正在进行的事务进行认证。

[0125] 有时候存在如下情况：谈及的事务是相对低值事务，其中，它可以较低的置信水平被接受而得到认证。这可以包括具有与之关联的低货币值(例如，$10购买)的事务或者低风险的事务(例如，进入会员制网站)。

[0126] 相反，为了对其它事务进行认证，会期望在允许事务进行之前要求高的认证置信度。这些事务可以包括大货币值的事务(例如，签署几百万美元供货合同)或者在发生不正确认证的情况下具有高风险的事务(例如，远程登录政府计算机)。

[0127] 如下文所述，与置信水平和事务值组合的启发法530的使用可用于允许比较器提供动态的上下文敏感认证系统。

[0128] 根据本发明的另一个实施例，比较器515可有利地跟踪特定事务的认证尝试。例如，当事务失败时，信任引擎110可以请求用户重新输入他或她的当前认证数据。认证引擎215的比较器515可有利地利用尝试限制器535以限制认证尝试的次数，由此禁止假扮用户的认证数据的暴力尝试。根据一个实施例，尝试限制器535包括监视事务的重复认证尝试并且例如将给定事务的认证尝试限制为三次的软件模块。因此，尝试限制器535例如将用于假扮个人的认证数据的自动尝试限制为仅三次“猜测”。当三次失败时，尝试限制器535可有利地拒绝另外的认证尝试。有利的是，例如通过不管正在发送的当前认证数据如何比较器515都返回否定结果，而实现这种拒绝。另一方面，事务引擎205可有利地阻止属于三次尝试先前已经失败的事务的任何另外的认证尝试。

[0129] 认证引擎215还包括数据分裂模块520和数据组装模块525。数据分裂模块520有利地包括具有对各种数据进行数学运算从而充分地将数据随机化并分裂成多个部分的能力的软件、硬件、或组合模块。根据一个实施例，不可以从个体部分重建原始数据。数据组装模块525有利地包括被构造为对上述充分随机化的部分进行数学运算从而使它们的组合提供原始译解数据的软件、硬件或组合模块。根据一个实施例，认证引擎215利用数据分裂模块520将登记认证数据随机化并分裂成多个部分，并且利用数据组装模块525将这

些部分重新组装成可用的登记认证数据。

[0130] 图 6 示出了根据本发明的一个实施例的各方面的图 2 的信任引擎 200 的密码引擎 220 的框图。与图 3 的事务引擎 205 类似，密码引擎 220 包括操作系统 605，操作系统 605 至少具有传统的 Web 服务器(例如，可从 Apache 得到的 Web 服务器)的变型版本的侦听和处理线程。如图 6 所示，密码引擎 220 包括功能与图 5 中类似的数据分裂模块 610 和数据组装模块 620。然而，根据一个实施例，与上述的登记认证数据不同，数据分裂模块 610 和数据组装模块 620 处理密码密钥数据。但是，熟练技术人员将从这里的公开认识到，数据分裂模块 910 和数据分裂模块 620 可以与认证引擎 215 的相应模块进行组合。

[0131] 密码引擎 220 还包括密码处理模块 625，密码处理模块 625 被构造为执行大量的密码功能中的一个、一些或全部。根据一个实施例，密码处理模块 625 可以包括软件模块或程序、硬件或二者。根据另一个实施例，密码处理模块 625 可以执行数据比较、数据解析、数据分裂、数据分离、数据哈希法、数据加密或解密、数字签名验证或创建、数字证书产生、存储、或请求、密码密钥产生等等。此外，熟练技术人员将从这里的公开认识到，密码处理模块 825 可有利地包括公钥基础设施，诸如良好隐私(PGP)、基于 RSA 的公钥系统、或者大量的替代密钥管理系统。此外，密码处理模块 625 可以执行公钥加密、对称密钥加密或二者。除了上述以外，密码处理模块 625 可以包括用于执行无缝、透明的互用性功能的一个或多个计算机程序或模块、硬件或二者。

[0132] 熟练技术人员还将从这里的公开认识到，密码功能可以包括一般与密码密钥管理系统有关的大量或多样的功能。

[0133] 图 7 示出了根据本发明的实施例的各方面的储存器系统 700 的简化框图。如图 7 所示，储存器系统 700 有利地包括多个数据存储设施，例如，数据存储设施 D1、D2、D3 和 D4。然而，本领域普通技术人员容易理解，储存器系统可以仅仅具有一个数据存储设施。根据本发明的一个实施例，数据存储设施 D1 到 D4 中的每个可以有利地包括参照图 4 的储存器 210 公开的一些或全部部件。与储存器 210 类似，数据存储设施 D1 到 D4 优选通过传统 SSL 与事务引擎 205、认证引擎 215 和密码引擎 220 进行通信。通信链路例如传送 XML 文档。来自事务引擎 205 的通信可有利地包括对数据的请求，其中，该请求被有利地广播至每个数据存储设施 D1 到 D4 的 IP 地址。另一方面，事务引擎 205 可以基于大量的标准(例如，响应时间、服务器负载、维护时间表等)，向特定数据存储设施广播请求。

[0134] 响应于来自事务引擎 205 的对数据的请求，储存器系统 700 有利地将存储的数据转发至认证引擎 215 和密码引擎 220。相应的数据组装模块接收转发的数据并且将该数据组装成可用格式。另一方面，从认证引擎 215 和密码引擎 220 到数据存储设施 D1 到 D4 的通信可以包括要存储的敏感数据的传输。例如，根据一个实施例，认证引擎 215 和密码引擎 220 可有利地利用它们各自的数据分裂模块，将敏感数据划分成多个不可译解的部分，然后将敏感数据的一个或多个不可译解的部分发送至特定数据存储设施。

[0135] 根据一个实施例，每个数据存储设施 D1 到 D4 包括分立且独立的存储系统(例如，目录服务器)。根据本发明的另一个实施例，储存器系统 700 包括多个在地理上分离的独立数据存储系统。通过将敏感数据分布到不同且独立的存储设施 D1 到 D4(存储设施 D1 到 D4 中的一些或全部可有利地在地理上分离)，储存器系统 700 提供冗余连同附加的安全措施。例如，根据一个实施例，为了对敏感数据进行译解和重新组装，仅需要来自多个数据存储设

施 D1 到 D4 中的两个的数据。因此，四个数据存储设施 D1 到 D4 中的两个可以由于维护、系统故障、电源故障等而不工作，这不会影响信任引擎 110 的功能。此外，根据一个实施例，由于存储在每个数据存储设施中的数据被随机化并且不可译解，所以对任何个体数据存储设施的危害不会必然危害敏感数据。此外，在具有地理分离的数据存储设施的实施例中，对多个地理远离的设施的危害变得愈加困难。实际上，即使是无良雇员，想要破坏所需的多个独立的地理远离的数据存储设施也是具有极大挑战的。

[0136] 尽管参照其优选和替代实施例公开了储存器系统 700，但是本发明并不限于此。相反，熟练技术人员将从这里的公开认识到储存器系统 700 的大量的替代物。例如，储存器系统 700 可以包括一个、两个或更多个数据存储设施。此外，可对敏感数据进行数学运算，从而使得为了对敏感数据进行重新组装和译解，需要来自两个或更多的数据存储设施的部分。

[0137] 如上所述，认证引擎 215 和密码引擎 220 分别包括数据分裂模块 520 和 610，用于分裂任何类型或形式的敏感数据（例如，文本、音频、视频、认证数据和密码密钥数据）。图 8 示出了根据本发明的实施例的各方面的数据分裂模块执行的数据分裂过程 800 的流程图。如图 8 所示，当认证引擎 215 或密码引擎 220 的数据分裂模块接收到敏感数据“S”时，数据分裂过程 800 在步骤 805 开始。优选地，在步骤 810，数据分裂模块产生充分随机数、值、或者比特串或集“A”。例如，可以以本领域普通技术人员可获得的用于生成适用于密码应用的高质量随机数的大量的不同传统技术，产生随机数 A。此外，根据一个实施例，随机数 A 包括可为任何适当长度的比特长度，例如，短于、长于或等于敏感数据 S 的比特长度。

[0138] 此外，在步骤 820 中，数据分裂过程 800 产生另一个统计随机数“C”。根据优选实施例，有利的是，可以并行完成统计随机数 A 和 C 的产生。数据分裂模块然后将数 A 和 C 与敏感数据 S 进行组合以产生新的数“B”和“D”。例如，数 B 可以包括 A XOR S 的二进制组合，数 D 可以包括 C XOR S 的二进制组合。XOR 函数或“异或”函数对于本领域普通技术人员是公知的。优选地，分别在步骤 825 和 830 中分别发生上述组合，并且根据一个实施例，上述组合也可以并行发生。数据分裂过程 800 然后进行到步骤 835，在步骤 835 中，对随机数 A 和 C 以及数 B 和 D 进行配对以使得这些配对均不包含通过它们自身可以重新组织并译解原始敏感数据 S 的充分数据。例如，这些数可以如下配对 :AC、AD、BC 和 BD。根据一个实施例，上述配对中的每个被分布至图 7 的储存器 D1 到 D4 之一。根据另一个实施例，上述配对中的每个被随机分布至储存器 D1 到 D4 之一。例如，在第一数据分裂过程 800 期间，配对 AC 可以例如通过 D2 的 IP 地址的随机选择而发送至储存器 D2。然后，在第二数据分裂过程 800 期间，配对 AC 可以例如通过 D4 的 IP 地址的随机选择而发送至储存器 D4。此外，这些配对可以全部存储在一个储存器上，并且可以存储在所述储存器的分离的位置上。

[0139] 基于上面的描述，有利的是，数据分裂过程 800 将敏感数据的各部分安置在四个数据存储设施 D1 到 D4 的每个中，从而使得没有单个数据存储设施 D1 到 D4 包括用于重建原始敏感数据 S 的充足的加密数据。如上所述，通过将数据随机化成个体不可用的加密部分，提高了安全性并且即使数据存储设施 D1 到 D4 之一受到危害仍可以保持数据的信任。

[0140] 尽管参照其优选实施例公开了数据分裂过程 800，但是本发明不限于此。相反，熟练技术人员将从这里的公开认识到数据分裂过程 800 的大量替代。例如，数据分裂过程可有利地将数据分裂成两个数，例如，随机数 A 和数 B，并且通过两个数据存储设施随机分布 A

和 B。此外，通过产生附加的随机数，数据分裂过程 800 可有利地在大量的数据存储设施之间分裂数据。数据可被分裂成任何希望的、选择的、预定的或者随机指定的大小单元，包括但不限于一个比特、多个比特、字节、千字节、兆字节或更大、或者大小的任何组合或序列。此外，改变从分裂过程得到的数据单元的大小，可以使数据更加难于恢复成可用形式，由此提高了敏感数据的安全性。本领域普通技术人员易于理解，分裂数据单元大小可以是多种多样的数据单元大小或大小的模式或者大小的组合。例如，数据单元大小可被选择或预定为都具有相同大小、不同大小的固定集合、大小的组合，或者随机产生大小。类似地，根据固定或预定的数据单元大小、数据单元大小的模式或组合、或者随机产生的每份的数据单元大小，数据单元可被分布成一份或多份。

[0141] 如上所述，为了重建敏感数据 S，需要对数据部分进行去随机化和重新组织。该过程可以有利地分别在认证引擎 215 和密码引擎 220 的数据组装模块 525 和 620 中执行。数据组装模块（例如，数据组装模块 525）从数据存储设施 D1 到 D4 接收数据部分，并且将数据重新组装成可用形式。例如，根据数据分裂模块 520 采用图 8 的数据分裂过程 800 的一个实施例，数据组装模块 525 使用来自数据存储设施 D1 到 D4 中的至少两个的数据部分重建敏感数据 S。例如，对 AC、AD、BC 和 BD 的配对被分布为任何两个提供 A 和 B 或者 C 和 D 之一。要注意，S=AXOR B 或者 S=C XOR D 指示当数据组装模块接收到 A 和 B 或者 C 和 D 之一时，数据组装模块 525 可有利地重新组装敏感数据 S。因此，当例如响应于信任引擎 110 的组装请求，数据组装模块 525 从数据存储设施 D1 到 D4 中的至少前两个接收到数据部分时，数据组装部分 525 可以组装敏感数据 S。

[0142] 基于以上数据分裂和组装过程，可用格式的敏感数据 S 仅仅存在于信任引擎 110 的有限区域内。例如，当敏感数据 S 包括登记认证数据时，可用的非随机化的登记认证数据仅在认证引擎 215 中可获得。同样地，当敏感数据 S 包括私有密码密钥数据时，可用的非随机化的私有密码密钥数据仅在密码引擎 220 中可获得。

[0143] 尽管参照其优选实施例公开了数据分裂和组装过程，但是本发明不限于此。相反，熟练技术人员将从这里的公开认识到用于分裂和重新组装敏感数据 S 的大量替代。例如，公钥加密可用于进一步保护数据存储设施 D1 到 D4 中的数据。此外，本领域普通技术人员易于理解，本文所述的数据分裂模块也是可纳入、进行组合或以其它形式成为任何预先存在的计算机系统、软件套装、数据库或其组合的一部分的本发明的独立且不同的实施例，或者本发明的其它实施例，诸如这里公开和描述的信任引擎、认证引擎和事务引擎。

[0144] 图 9A 示出了根据本发明的实施例的各方面的登记过程 900 的数据流。如图 9A 所示，当用户希望向密码系统 100 的信任引擎 110 登记时，登记过程 900 在步骤 905 开始。根据这个实施例，用户系统 105 有利地包括例如基于 Java 的询问用户输入登记数据（例如，人口数据和登记认证数据）的客户机侧 Java 小程序（applet）。根据一个实施例，登记认证数据包括用户 ID、口令、生物测定等等。根据一个实施例，在询问过程中，客户机侧 Java 小程序优选与信任引擎 110 进行通信以确保选择的用户 ID 是唯一的。当用户 ID 不是唯一时，信任引擎 110 可有利地建议一个唯一用户 ID。客户机侧 Java 小程序收集登记数据并且例如通过 XML 文档将登记数据传输至信任引擎 110，具体地讲传输至事务引擎 205。根据一个实施例，用认证引擎 215 的公钥对该传输进行编码。

[0145] 根据一个实施例，用户在登记过程 900 的步骤 905 中执行单次登记。例如，用户将

他或她自己作为特定人员(例如，Joe User)登记。当 Joe User 希望作为 Joe User (Mega 公司的 CEO)进行登记时，根据这个实施例，Joe User 登记第二次，接收第二唯一用户 ID 并且信任引擎 110 不将这两个身份进行关联。根据本发明的另一个实施例，登记过程 900 为一个用户 ID 提供多个用户身份。因此，在以上例子中，信任引擎 110 将有利地将 Joe User 的两个身份进行关联。熟练技术人员从这里的公开应该明白，一个用户可以具有许多身份，例如 Joe User (户主)、Joe User (慈善基金成员)等。即使用户可以具有多个身份，根据这个实施例，信任引擎 110 优选仅仅存储一组登记数据。此外，有利的是，用户可以在需要时添加、编辑 / 更新或删除身份。

[0146] 尽管参照其优选实施例公开了登记过程 900，但是本发明不限于此。相反，熟练技术人员将从这里的公开认识到用于收集登记数据(具体地为登记认证数据)的大量替代。例如，Java 小程序可以是基于公共对象模型(COM) 的 Java 小程序等。

[0147] 另一方面，登记过程可包括分级登记。例如，在最低的登记等级，用户可以经由通信链路 125 进行登记而不生成关于他或她的身份的文档。根据提高的登记等级，用户使用信任第三方(例如，数字公证人)进行登记。例如，用户可以亲自到信任第三方，生成证明(例如，出生证明、驾照、军官证等)，并且信任第三方可有利地将例如它们的数字签名包括在登记提交中。信任第三方可以包括真实公证人、政府机构(例如，邮局或机动车部门)、大公司中招募雇员的人力资源人员等。熟练技术人员从这里的公开应该明白，在登记过程 900 期间可以进行大量的不同等级的登记。

[0148] 在接收到登记认证数据后，在步骤 915，事务引擎 205 使用传统的全 SSL 技术将登记认证数据转发至认证引擎 215。在步骤 920 中，认证引擎 215 使用认证引擎 215 的私钥对登记认证数据进行解密。此外，认证引擎 215 利用数据分裂模块对登记认证数据进行数学运算从而将该数据分裂成至少两个独立不可译解的随机数。如上所述，至少两个数可以包括统计随机数和二进制异或数。在步骤 925 中，认证引擎 215 将随机数的每个部分转发至数据存储设施 D1 到 D4 之一。如上所述，有利的是，认证引擎 215 还可以对哪些部分传送到哪个储存器进行随机化。

[0149] 在登记过程 900 中，用户常常还希望发放数字证书使得他或她可以从密码系统 100 之外的其它人接收加密的文档。如上所述，认证机构 115 通常根据几个传统标准中的一个或多个发放数字证书。通常，数字证书包括每人皆知的用户或系统的公钥。

[0150] 不管在登记时或者在另一个时间用户是否请求了数字证书，该请求都通过信任引擎 110 传送到认证引擎 215。根据一个实施例，该请求包括例如具有用户的正确名称的 XML 文档。根据步骤 935，认证引擎 215 将该请求传送给密码引擎 220 以指示密码引擎 220 产生密码密钥或密钥对。

[0151] 在被请求时，在步骤 935，密码引擎 220 产生至少一个密码密钥。根据一个实施例，密码处理模块 625 产生密钥对，其中，一个密钥用作私钥，一个密钥用作公钥。密码引擎 220 存储私钥，并且根据一个实施例存储公钥的副本。在步骤 945 中，密码引擎 220 向事务引擎 205 发送对数字证书的请求。根据一个实施例，该请求有利地包括例如嵌入在 XML 文档内的诸如 PKCS10 的标准化请求。对数字证书的请求可有利地对应于一个或多个认证机构以及这些认证机构要求的一个或多个标准格式。

[0152] 在步骤 950 中，事务引擎 205 将这个请求转发至认证机构 115，在步骤 955 中，认证

机构 115 返回数字证书。有利的是，返回数字证书可以是例如 PKCS7 的标准化格式或者是一个或多个认证机构 115 的专有格式。在步骤 960 中，数字证书由事务引擎 205 接收，副本被转发至用户并且副本由信任引擎 110 存储。信任引擎 110 存储证书的副本从而使得信任引擎 110 将不需要依赖于认证机构 115 的可用性。例如，当用户希望发送数字证书或者第三方请求用户的数字证书时，对数字证书的请求通常被发送至认证机构 115。然而，如果认证机构 115 正在进行维护或者成为故障或安全危害的牺牲品，则无法获得数字证书。

[0153] 在发放密码密钥后的任何时间，密码引擎 220 可有利地利用上述的数据分裂过程 800 从而使得密码密钥被分裂成独立不可译解的随机数。与认证数据类似，在步骤 965 中，密码引擎 220 将随机数传送给数据存储设施 D1 到 D4。

[0154] 熟练技术人员将从这里的公开认识到，用户可以在登记后的任何时间请求数字证书。此外，系统之间的通信可有利地包括全 SSL 或公钥加密技术。此外，登记过程可以发放来自多个认证机构(包括信任引擎 110 内部或外部的一个或多个专有认证机构)的多个数字证书。

[0155] 如在步骤 935 到步骤 960 中所公开的，本发明的一个实施例包括对最终存储在信任引擎 110 上的证书的请求。根据一个实施例，由于密码处理模块 625 发放由信任引擎 110 使用的密钥，所以每个证书对应于一个私钥。因此，信任引擎 110 通过监视用户拥有的或与用户关联的证书，可以有利地提供互用性。例如，当密码引擎 220 接收对密码功能的请求时，密码处理模块 625 可以调查请求用户拥有的证书以确定该用户是否拥有与该请求的属性匹配的私钥。当存在这种证书时，密码处理模块 625 可以使用该证书或者与之关联的公钥或私钥执行被请求的功能。当不存在这种证书时，密码处理模块 625 可以有利并透明地执行多个动作以尝试补救适当密钥的缺失。图 9B 示出了互用性过程 970 的流程图，根据本发明的实施例的各方面，它公开了上述步骤以确保密码处理模块 625 使用适当密钥执行密码功能。

[0156] 如图 9B 所示，互用性过程 970 在步骤 972 开始，在步骤 972 中，密码处理模块 925 确定希望的证书的类型。根据本发明的一个实施例，有利的是，在对密码功能的请求或者由请求者提供的其它数据中指定证书的类型。根据另一个实施例，证书类型可以通过该请求的数据格式进行确定。例如，密码处理模块 925 可以有利地识别与特定类型对应的请求。

[0157] 根据一个实施例，证书类型可以包括一个或多个算法标准，例如，RSA、ELGAMAL 等。此外，证书类型可以包括一个或多个密钥类型，例如，对称密钥、公钥、例如 256 比特密钥的强加密密钥、较不安全的密钥等。此外，证书类型可以包括一个或多个上述算法标准或密钥、一个或多个消息或数据格式、一个或多个数据封装或编码方案(例如，Base 32 或 Base 64)的升级或替换。证书类型还可以包括与一个或多个第三方密码应用或接口、一个或多个通信协议或者一个或多个证书标准或协议的兼容性。熟练技术人员将从这里的公开认识到，在证书类型中可存在其它差别，并且可以如本文公开地执行从或到这些差别的翻译。

[0158] 一旦密码处理模块 625 确定了证书类型，互用性过程 970 进行到步骤 974，并且在步骤 974 中确定用户是否拥有与确定的类型匹配的证书。当用户拥有匹配的证书时，例如，信任引擎 110 通过例如其先前存储可访问该匹配的证书，加密处理模块 825 知道匹配的私钥也存储在信任引擎 110 内。例如，匹配的私钥可存储在储存器 210 或储存器系统 700 内。有利的是，密码处理模块 625 可以请求从例如储存器 210 组装匹配的私钥，然后在步骤 976

中，使用该匹配的私钥执行密码操作或功能。例如，如上所述，密码处理模块 625 可以有利地执行哈希法、哈希比较、数据加密或解密、数字签名验证或建立等。

[0159] 当用户不拥有匹配的证书时，互用性过程 970 进行到步骤 978，在步骤 978，密码处理模块 625 确定用户是否拥有交叉认证的证书。根据一个实施例，当第一认证机构确定信任来自第二认证机构的证书时，出现认证机构之间的交叉认证。换言之，第一认证机构确定来自第二认证机构的证书满足特定质量标准，由此可被“认证”为与第一认证机构自身的证书等效。当认证机构例如发放具有信任等级的证书时，交叉认证变得更加复杂。例如，第一认证机构通常基于登记过程的可靠度可以向特定证书提供三个等级的信任，而第二认证机构可以提供七个等级的信任。有利的是，交叉认证可以跟踪来自第二认证机构的哪些等级和哪些证书可替代来自第一认证机构的哪些等级和哪些证书。当在两个认证机构之间正式公开地完成上述交叉认证时，证书和等级的彼此映射通常称作“链锁（chaining）”。

[0160] 根据本发明的另一个实施例，有利的是，密码处理模块 625 可以开发由认证机构达成一致的交叉认证之外的交叉认证。例如，密码处理模块 625 可以访问第一认证机构的证书操作声明（CPS）或者其它公布的政策声明，并且例如使用特定信任等级要求的认证令牌（token），将第一认证机构的证书与另一个认证机构的证书进行匹配。

[0161] 当在步骤 978 中密码处理模块 625 确定用户拥有交叉认证的证书时，互用性过程 970 进行到步骤 976，并且使用交叉认证的公钥、私钥或二者执行密码操作或功能。另选地，当密码处理模块 625 确定用户不拥有交叉认证的证书时，互用性过程 970 进行到步骤 980，在步骤 980，密码处理模块 625 选择向其发放被请求的证书类型或者交叉认证的证书的认证机构。在步骤 982 中，密码处理模块 625 确定上面讨论的用户登记认证数据是否满足选择的认证机构的认证要求。例如，如果用户例如通过回答人口和其它提问经由网络登入，则与提供生物测定数据并且出现在第三方（例如，公证人）前的用户相比，提供的认证数据可以建立较低等级的信任。根据一个实施例，有利的是，可以在选择的认证机构的 CPS 中提供上述认证要求。

[0162] 当用户已经向信任引擎 110 提供满足选择的认证机构的要求的登记认证数据时，互用性过程 970 进行到步骤 984，在步骤 984，密码处理模块 825 从选择的认证机构获取证书。根据一个实施例，密码处理模块 625 通过登记过程 900 的下面的步骤 945 到 960 获取证书。例如，密码处理模块 625 可有利地利用密码引擎 220 已经可获得的一个或多个密钥对中的一个或多个公钥，从认证机构请求证书。根据另一个实施例，密码处理模块 625 可有利地产生一个或多个新密钥对，并且使用与之对应的公钥从认证机构请求证书。

[0163] 根据另一个实施例，信任引擎 110 可有利地包括能够发放一个或多个证书类型的一个或多个证书发放模块。根据这个实施例，证书发放模块可以提供上述证书。当密码处理模块 625 获取证书时，互用性过程 970 进行到步骤 976，并且使用与获取的证书对应的公钥、私钥或二者执行密码操作或功能。

[0164] 当在步骤 982 中用户没有向信任引擎 110 提供满足选择的认证机构的要求的登记认证数据时，在步骤 986 中密码处理模块 625 确定是否存在具有不同的认证要求的其它认证机构。例如，密码处理模块 625 可以寻找具有更低的认证要求但是仍发放选择的证书或其交叉认证的认证机构。

[0165] 当上述具有更低要求的认证机构存在时，互用性过程 970 进行到步骤 980 并且选

择该认证机构。另选地，当不存在这种认证机构时，在步骤 988 中信任引擎 110 可以从用户请求另外的认证令牌。例如，信任引擎 110 可以请求包括例如生物测定数据的新的登记认证数据。另外，信任引擎 110 可以请求用户出现在信任第三方前并且提供适当的认证证明（例如，出现在针对驾照、社会保险卡、银行卡、出生证明、军官证等等的公证前）。当信任引擎 110 接收到更新的认证数据时，互用性过程 970 进行到步骤 984 并且获取上述选择的证书。

[0166] 通过上述互用性过程 970，密码处理模块 625 有利地在不同的密码系统之间提供无缝透明的翻译和转换。熟练技术人员将从这里的公开认识到上述互用系统的大量的优点和实施方式。例如，互用性过程 970 的上述步骤 986 可有利地包括信任仲裁的方面（下文更加详细讨论），其中，认证机构可以在特殊境况下接受较低等级的交叉认证。此外，互用性过程 970 可以包括确保标准证书撤销之间的互用性以及标准证书撤销的利用，例如，利用证书撤销列表（CRL）、在线证书状态协议（OCSP）等。

[0167] 图 10 示出了根据本发明的实施例的各方面的认证过程 1000 的数据流。根据一个实施例，认证过程 1000 包括从用户收集当前认证数据并且将它与用户的登记认证数据进行比较。例如，认证过程 1000 在步骤 1005 开始，在步骤 1005，用户希望与例如卖方执行事务。这种事务例如可包括选择购买选项、请求访问卖方系统 120 的限制区或装置、等等。在步骤 1010 中，卖方向用户提供事务 ID 和认证请求。事务 ID 可有利地包括 192 比特量（32 比特时间戳，串连 128 比特随机量或“现时（nonce）”，再串连 32 比特卖方特定常数）。这种事务 ID 唯一识别事务从而使信任引擎 110 能够拒绝假冒事务。

[0168] 认证请求可有利地包括针对特定事务需要什么等级的认证。例如，卖方可以指定谈及的事务要求的特定置信等级。如果不能够使得认证达到这个置信等级，则如下文所述，在用户没有进一步认证以提升置信等级或者卖方与服务器之间的认证条款没有变化的情况下，将不会发生事务。在下文中更加完整地讨论这些问题。

[0169] 根据一个实施例，有利的是，可由卖方侧 Java 小程序或其它软件程序产生事务 ID 和认证请求。此外，事务 ID 和认证数据的传输可以包括使用传统的 SSL 技术（例如，1/2SSL 或换言之卖方侧认证的 SSL）加密的一个或多个 XML 文档。

[0170] 在用户系统 105 接收到事务 ID 和认证请求后，用户系统 105 从用户收集潜在包括当前生物测定信息的当前认证数据。在步骤 1015，用户系统 105 用认证引擎 215 的公钥至少对当前认证数据“B”和事务 ID 进行加密，并且将该数据传送到信任引擎 110。所述传输优选包括至少用传统的 1/2SSL 技术进行加密的 XML 文档。在步骤 1020 中，事务引擎 205 接收该传输，优选识别 URL 或 URI 中的数据格式或请求，并且将该传输转发至认证引擎 215。

[0171] 在步骤 1015 和 1020 中，卖方系统 120 在步骤 1025 使用优选的全 SSL 技术向信任引擎 110 转发事务 ID 和认证请求。该通信还可以包括卖方 ID，尽管还可以通过事务 ID 的非随机部分传送卖方标识。在步骤 1030 和 1035，事务引擎 205 接收该通信，建立审计索引的记录，并且产生对要从数据存储设施 D1 到 D4 进行组装的用户的登记认证数据的请求。在步骤 1040，储存器系统 700 向认证引擎 215 传送与用户对应的登记认证数据的部分。在步骤 1045，认证引擎 215 使用它的私钥对该传输进行解密并且将该登记认证数据与由用户提供的当前认证数据进行比较。

[0172] 步骤 1045 的比较可有利地应用启发式上下文敏感认证（上文提及并且在下文更

加详细讨论)。例如,如果接收到的生物测定信息没有完美匹配,则得到较低置信匹配。在特定实施例中,认证的置信等级针对事务的性质以及用户和卖方二者的期望进行平衡。在下文对此进行更加详细的讨论。

[0173] 在步骤1050,认证引擎215在认证请求内填入步骤1045的比较的结果。根据本发明的一个实施例,认证请求填充有认证过程1000的是/否或者真/假结果。在步骤1055,被填充的认证请求返回到卖方以由卖方遵照行事,例如允许用户完成发起该认证请求的事务。根据一个实施例,确认消息被传递至用户。

[0174] 基于上面的描述,有利的是,认证过程1000使敏感数据保持安全并且生成被构造为维护敏感数据的完整性的结果。例如,仅仅在认证引擎215内组装敏感数据。例如,登记认证数据在通过数据组装模块在认证引擎215内进行组装前是不可译解的,并且当前认证数据在通过传统的SSL技术和认证引擎215的私钥进行展开(unwrap)前是不可译解的。此外,发送至卖方的认证结果不包括敏感数据,并且用户甚至无法知道是否他或她生成了有效的认证数据。

[0175] 尽管参照其优选和替代实施例公开了认证过程1000,但是本发明不限于此。相反,熟练技术人员将从这里的公开认识到认证过程1000的大量替代。例如,有利的是,卖方可由几乎任何请求应用(甚至是驻留在用户系统105内的应用)替代。例如,在对文档进行解锁之前,客户机应用(例如,Microsoft Word)可使用应用程序接口(API)或密码API(CAPI)请求认证。另选地,邮件服务器、网络、蜂窝电话、个人或移动计算装置、工作站等都可以形成能够通过认证过程1000进行填充的认证请求。实际上,在提供上述信任的认证过程1000后,请求应用或装置可以访问或使用大量的电子或计算机装置或系统。

[0176] 此外,在认证失败的情况下,认证过程1000可以采用大量的替代过程。认证失败可以保持用户重新输入他或她的当前认证数据的相同事务ID和请求。如上所述,使用相同的事务ID使认证引擎215的比较器可以监视并限制针对特定事务的认证尝试的数目,由此建立更加安全的密码系统100

[0177] 此外,有利的是,认证过程1000可用于开发一流的单登录方案(例如,对敏感数据仓库(vault)进行解锁)。例如,成功或肯定认证可以向认证的用户提供自动访问几乎无限数目的系统和应用的任何数目的口令的能力。例如,用户的认证可以向用户提供对与多个在线卖方关联的口令、登录、金融证明等、局域网、各种个人计算装置、互联网服务提供商、拍卖商、投资经纪商等的访问。通过采用敏感数据仓库,用户可以选择真正大的和随机的口令,这是因为他们不再需要通过关联来记住它们。而且,认证过程1000提供对它们的访问。例如,用户可以选择长度为20多位的随机字母数字串,而非与可记忆数据、名称等关联的字母数字串。

[0178] 根据一个实施例,有利的是,与给定用户关联的敏感数据仓库可以存储在储存器210的数据存储设施内,或者被分裂并存储在储存器系统700内。根据这个实施例,在肯定的用户认证后,信任引擎110将被请求的敏感数据(例如,适当的口令)提供给请求应用。根据另一个实施例,信任引擎110可以包括用于存储敏感数据仓库的单独系统。例如,信任引擎110可以包括孤立的软件引擎,用于执行数据仓库功能并且形象化地驻留在信任引擎110的上述前端安全系统的“后方”。根据这个实施例,在软件引擎从信任引擎110接收到指示肯定的用户认证的信号后,软件引擎提供被请求的敏感数据。

[0179] 在另一个实施例中，数据仓库可由第三方系统实现。与软件引擎实施例类似，有利的是，在第三方系统从信任引擎 110 接收到指示肯定的用户认证的信号后，第三方系统可以提供被请求的敏感数据。根据另一个实施例，数据仓库可以在用户系统 105 上实现。在从信任引擎 110 接收到指示肯定的用户认证的信号后，用户侧软件引擎可以有利地提供上述数据。

[0180] 尽管参照替代实施例公开了上述数据仓库，但是熟练技术人员将从这里的公开认识到它的大量的另外实施方式。例如，特定数据仓库可以包括一些或所有的上述实施例中的方面。此外，任何的上述数据仓库可以在不同的时刻采用一个或多个认证请求。例如，任一数据仓库可以对于每一个或多个事务定期地要求认证，对于每一个或多个会话以及对一个或多个网页或网站的每次访问要求认证，以一个或多个其它指定间隔等要求认证。

[0181] 图 11 示出了根据本发明的实施例的各方面的签名过程 1100 的数据流。如图 11 所示，签名过程 1100 包括与在上文参照图 10 描述的认证过程 1000 类似的步骤。根据本发明的一个实施例，签名过程 1100 首先对用户进行认证，然后执行几个数字签名功能中的一个或多个(在下文进行更加详细的描述)。根据另一个实施例，有利的是，签名过程 1100 可以存储与之相关的数据，例如，消息或文档的哈希值等。有利的是，这个数据可用于审计或任何其它事件，例如，当参与方尝试抵赖事务时。

[0182] 如图 11 所示，在认证步骤中，用户和卖方可以有利地对消息(例如，合同) 达成一致。在签名过程中，有利的是，签名过程 1100 确保由用户签名的合同与由卖方提供的合同相同。因此，根据一个实施例，在认证过程中，卖方和用户在发送至认证引擎 215 的数据中包括消息或合同的各自副本的哈希值。通过仅仅采用消息或合同的哈希值，信任引擎 110 可有利地存储明显减小的数据量，从而提供更高效且成本效益高的密码系统。此外，有利的是，存储的哈希值可与讨论的文档的哈希值进行比较，以确定讨论的文档是否与由任一方签名的文档匹配。确定文档是否和与事务相关的一个文档相同的能力提供了附加证据，其能够用于对抗由一方对事务抵赖的主张。

[0183] 在步骤 1103，认证引擎 215 组装登记认证数据，并且将它与由用户提供的当前认证数据进行比较。当认证引擎 215 的比较器指示登记认证数据与当前认证数据匹配时，认证引擎 215 的比较器还将由卖方提供的消息的哈希值与由用户提供的消息的哈希值进行比较。因此，有利的是，认证引擎 215 确保用户同意的消息与卖方同意的消息相同。

[0184] 在步骤 1105 中，认证引擎 215 向密码引擎 220 发送数字签名请求。根据本发明的一个实施例，该请求包括消息或合同的哈希值。然而，熟练技术人员将从这里的公开认识到，密码引擎 220 实际上可以对任何类型的数据(包括但不限于视频、音频、生物测定、图像或文本) 进行加密以形成期望的数字签名。返回到步骤 1105，数字签名请求优选包括通过传统的 SSL 技术传送的 XML 文档。

[0185] 在步骤 1110 中，认证引擎 215 向数据存储设施 D1 到 D4 的每个发送请求，从而使得数据存储设施 D1 到 D4 的每个发送它们各自的与签名方对应的一个或多个密码密钥的部分。根据另一个实施例，密码引擎 220 采用上文所述的互用性过程 970 的一些或所有的步骤，从而使得密码引擎 220 首先确定签名方要从储存器 210 或储存器系统 700 请求的一个或多个适当密钥，并且采取行动以提供适当的匹配密钥。根据另一个实施例，有利的是，认证引擎 215 或密码引擎 220 可以请求与签名方关联并存储在储存器 210 或储存器系统 700

中的一个或多个密钥。

[0186] 根据一个实施例，签名方包括用户和卖方之一或二者。在这种情况下，有利的是，认证引擎 215 请求与用户和 / 或卖方对应的密码密钥。根据另一个实施例，签名方包括信任引擎 110。在这个实施例中，信任引擎 110 在证明认证过程 1000 正确地认证了用户、卖方或二者。因此，认证引擎 215 请求信任引擎 110 的密码密钥(例如，属于密码引擎 220 的密钥)以执行数字签名。根据另一个实施例，信任引擎 110 执行数字公证类功能。在这个实施例中，签名方包括连同信任引擎 110 的用户、卖方或二者。因此，信任引擎 110 提供用户和 / 或卖方的数字签名，然后用它自身的数字签名指示用户和 / 或卖方得到正确认证。在这个实施例中，有利的是，认证引擎 215 可以请求与用户、卖方或二者对应的密码密钥的组装。根据另一个实施例，有利的是，认证引擎 215 可以请求与信任引擎 110 对应的密码密钥的组装。

[0187] 根据另一个实施例，信任引擎 110 执行授权书类功能。例如，信任引擎 110 可以代表第三方对消息进行数字签名。在这种情况下，认证引擎 215 请求与第三方关联的密码密钥。根据这个实施例，有利的是，签名过程 1100 可以包括在允许授权书类功能之前对第三方进行认证。此外，认证过程 1000 可以包括对第三方约束(例如，规定何时以及在什么境况下可使用特定的第三方的签名的商业逻辑等)的检查。

[0188] 基于上面的描述，在步骤 1110 中，认证引擎从与签名方对应的数据存储设施 D1 到 D4 请求密码密钥。在步骤 1115 中，数据存储设施 D1 到 D4 将它们各自的与签名方对应的密码密钥的部分传输至密码引擎 220。根据一个实施例，上述传输包括 SSL 技术。根据另一个实施例，有利的是，上述传输可用密码引擎 220 的公钥进行超级加密。

[0189] 在步骤 1120 中，密码引擎 220 组装签名方的上述密码密钥并且用其对消息进行加密，从而形成数字签名。在签名过程 1100 的步骤 1125 中，密码引擎 220 向认证引擎 215 发送该数字签名。在步骤 1130 中，认证引擎 215 将填充了的认证请求连同哈希处理的消息的副本以及数字签名发送至事务引擎 205。在步骤 1135 中，事务引擎 205 将包括事务 ID、认证是否成功的指示以及数字签名的收据传输至卖方。根据一个实施例，有利的是，上述传输可以包括信任引擎 110 的数字签名。例如，信任引擎 110 可以用它的私钥对该收据的哈希值进行加密，从而形成要附加到至卖方的传输的数字签名。

[0190] 根据一个实施例，事务引擎 205 还向用户发送确认消息。尽管参照其优选和替代实施例公开了签名过程 1100，但是本发明不限于此。相反，熟练技术人员将从这里的公开认识到签名过程 1100 的大量替代。例如，可以用诸如电子邮件应用的用户应用替换卖方。例如，用户可能希望利用他或她的数字签名对特定电子邮件进行数字签名。在这种实施例中，有利的是，整个签名过程 1100 中的传输可以仅仅包括消息的哈希值的一个副本。此外，熟练技术人员将从这里的公开认识到，大量的客户机应用可以请求数字签名。例如，客户机应用可以包括字处理器、电子数据表、电子邮件、语音邮件、对限制系统区域的访问等。

[0191] 此外，熟练技术人员将从这里的公开认识到，有利的是，签名过程 1100 的步骤 1105 到 1120 可以采用图 9B 的互用性过程 970 的一些或所有的步骤，从而提供例如需要处理不同签名类型下的数字签名的不同密码系统之间的互用性。

[0192] 图 12 示出了根据本发明的实施例的各方面的加密 / 解密过程 1200 的数据流。如图 12 所示，通过使用认证过程 1000 对用户进行认证，开始解密过程 1200。根据一个实施

例，认证过程 1000 在认证请求中包括同步会话密钥。例如，在传统的 PKI 技术中，熟练技术人员明白，使用公钥和私钥的加密或解密的计算强度大并且可能要求大量的系统资源。然而，在对称密钥密码系统或消息的发送方和接收方共享用于对消息进行加密和解密的一个公同密钥的系统中，数学运算明显更简单并更快速。因此，在传统的 PKI 技术中，消息的发送方将产生同步会话密钥并且使用更简单更快速的对称密钥系统对消息进行加密。然后，发送方用接收方的公钥对会话密钥进行加密。加密的会话密钥将附加到同步加密的消息并且这两个数据都被发送至接收方。接收方使用他或她的私钥对会话密钥进行解密，然后使用会话密钥对消息进行解密。基于上面的描述，更简单更快速的对称密钥系统用于大部分的加密 / 解密处理。因此，在解密过程 1200 中，解密有利地假定已利用用户的公钥对同步密钥进行了加密。因此，如上所述，加密的会话密钥包括在认证请求中。

[0193] 返回到解密过程 1200，在步骤 1205 中用户已经得到认证后，认证引擎 215 将加密的会话密钥转发至密码引擎 220。在步骤 1210 中，认证引擎 215 向数据存储设施 D1 到 D4 的每个转发请求用户的密码密钥数据的请求。在步骤 1215 中，每个数据存储设施 D1 到 D4 将密码密钥的它们各自的部分传输至密码引擎 220。根据一个实施例，用密码引擎 220 的公钥对上述传输进行加密。

[0194] 在解密过程 1200 的步骤 1220 中，密码引擎 220 组装密码密钥并且用其对会话密钥进行解密。在步骤 1225 中，密码引擎将会话密钥转发至认证引擎 215。在步骤 1227 中，认证引擎 215 填充包括解密的会话密钥的认证请求，并且将填充了的认证请求发送至事务引擎 205。在步骤 1230 中，事务引擎 205 将认证请求连同会话密钥转发至请求应用或卖方。然后，根据一个实施例，请求应用或卖方使用会话密钥对加密的消息进行解密。

[0195] 尽管参照其优选和替代实施例公开了解密过程 1200，但是熟练技术人员将从这里的公开认识到解密过程 1200 的大量替代。例如，解密过程 1200 可以在同步密钥加密之前并且依赖于全公钥技术。在这种实施例中，请求应用可以将整个消息发送至密码引擎 220，或者可以采用一些类型的压缩或可逆哈希法以将消息发送至密码引擎 220。熟练技术人员从本文公开还将认识到上述通信可有利地包括以 SSL 技术打包的 XML 文档。

[0196] 加密 / 解密过程 1200 还提供文档或其它数据的加密。因此，在步骤 1235 中，请求应用或卖方可有利地向信任引擎 110 的事务引擎 205 发送对用户的公钥的请求。请求应用或卖方产生这个请求，因为请求应用或卖方例如使用用户的公钥对将用于对文档或消息进行加密的会话密钥进行加密。如在登记过程 900 中所述，事务引擎 205 例如将用户的数字证书的副本存储在海量存储器 225 中。因此，在加密过程 1200 的步骤 1240 中，事务引擎 205 从海量存储器 225 请求用户的数字证书。在步骤 1245 中，海量存储器 225 将与用户对应的数字证书发送至事务引擎 205。在步骤 1250 中，事务引擎 205 将数字证书发送至请求应用或卖方。根据一个实施例，加密过程 1200 的加密部分不包括用户的认证。这是因为请求卖方仅需要用户的公钥，而不请求任何敏感数据。

[0197] 熟练技术人员将从这里的公开认识到，如果特定用户没有数字证书，信任引擎 110 可采用一些或全部的登记过程 900 以为该特定用户产生数字证书。然后，信任引擎 110 可启动加密 / 解密过程 1200，并由此提供适当的数字证书。此外，熟练技术人员将从这里的公开认识到，加密 / 解密过程 1200 的步骤 1220 和 1235 到 1250 可有利地采用图 9B 的互用性过程的一些或所有的步骤，从而提供例如可能需要处理加密的不同密码系统之间的互用

性。

[0198] 图 13 示出了根据本发明的另一个实施例的各方面的信任引擎系统 1300 的简化框图。如图 13 所示，信任引擎系统 1300 包括多个不同的信任引擎 1305、1310、1315 和 1320。为了便于更加全面理解本发明，图 13 示出了具有事务引擎、储存器和认证引擎的每个信任引擎 1305、1310、1315 和 1320。然而，熟练技术人员认识到，每个事务引擎可有利地包括参照图 1 到图 8 公开的部件和通信通道中的一些、组合或全部。例如，一个实施例可有利地包括具有一个或多个事务引擎、储存器以及密码服务器或它们的任何组合的信任引擎。

[0199] 根据本发明的一个实施例，信任引擎 1305、1310、1315 和 1320 的每个在地理上分离，从而例如信任引擎 1305 可以驻留在第一位置，信任引擎 1310 可以驻留在第二位置，信任引擎 1315 可以驻留在第三位置，信任引擎 1320 可以驻留在第四位置。有利的是，上述地理分离减小了系统响应时间同时提高整个信任引擎系统 1300 的安全性。

[0200] 例如，当用户登录到密码系统 100 时，用户可能离第一位置最近并且可能希望得到认证。如参照图 10 所述，为了得到认证，用户提供当前认证数据（例如，生物测定等等），并且当前认证数据被与该用户的登记认证数据进行比较。因此，根据一个例子，有利的是，用户向地理上最近的信任引擎 1305 提供当前认证数据。信任引擎 1305 的事务引擎 1321 然后将当前认证数据转发至也驻留在第一位置的认证引擎 1322。根据另一个实施例，事务引擎 1321 将当前认证数据转发至信任引擎 1310、1315 或 1320 的认证引擎中的一个或多个。

[0201] 事务引擎 1321 还向例如信任引擎 1305 到 1320 的每个的储存器请求登记认证数据的组装。根据这个实施例，每个储存器将它的登记认证数据部分提供给信任引擎 1305 的认证引擎 1322。认证引擎 1322 然后利用例如来自前两个储存器的加密的数据部分进行响应，并且将登记认证数据组装成译解的形式。认证引擎 1322 将登记认证数据与当前认证数据进行比较并且将认证结果返回到信任引擎 1305 的事务引擎 1321。

[0202] 基于以上内容，信任引擎系统 1300 采用多个地理分离的信任引擎 1305 到 1320 中的最接近的一个执行认证过程。根据本发明的一个实施例，有利的是，可以在在用户系统 105、卖方系统 120 或认证机构 115 中的一个或多个上执行的客户机侧 Java 小程序处执行到最接近的事务引擎的信息路由。根据一个替代实施例，可以采用更加精密的判断过程以从信任引擎 1305 到 1320 进行选择。例如，该判断可以基于给定信任引擎的可用性、操作性、连接速度、负载、性能、地理接近、或者它们的组合。

[0203] 这样，信任引擎系统 1300 降低它的响应时间，同时保持与地理上远离的数据存储设施（例如，参照图 7 描述的那些数据存储设施，其中，每个数据存储设施存储敏感数据的随机化部分）关联的安全性优点。例如，在例如信任引擎 1315 的储存器 1325 处的安全危害不一定危害信任引擎系统 1300 的敏感数据。这是因为储存器 1325 仅仅包含不可译解的随机化数据，在没有更多数据的情况下它们是完全无用的。

[0204] 根据另一个实施例，有利的是，信任引擎系统 1300 可以包括与认证引擎类似地布置的多个密码引擎。密码引擎可有利执行密码功能，例如，参照图 1 到图 8 公开的那些密码功能。根据另一个实施例，信任引擎系统 1300 可有利地用多个密码引擎替代多个认证引擎，从而执行诸如参照图 1 到图 8 公开的密码功能的密码功能。根据本发明的另一个实施例，信任引擎系统 1300 可以用具有在上文公开的认证引擎、密码引擎或它们二者的一些或所有的功能的引擎，来替代各多个认证引擎。

[0205] 尽管参照其优选和替代实施例公开了信任引擎系统 1300，但是熟练技术人员将认识到，信任引擎系统 1300 可以包括信任引擎 1305 到 1320 的部分。例如，信任引擎系统 1300 可以包括一个或多个事务引擎、一个或多个储存器、一个或多个认证引擎、一个或多个密码引擎、或者它们的组合。

[0206] 图 14 示出了根据本发明的另一个实施例的各方面的信任引擎系统 1400 的简化框图。如图 14 所示，信任引擎系统 1400 包括多个信任引擎 1405、1410、1415 和 1420。根据一个实施例，信任引擎 1405、1410、1415 和 1420 的每个包括参照图 1 到图 8 公开的信任引擎 110 的一些或所有的部件。根据这个实施例，当用户系统 105、卖方系统 120 或认证机构 115 的客户机侧 Java 小程序与信任引擎系统 1400 进行通信时，这些通信被发送至信任引擎 1405 到 1420 的每个的 IP 地址。另外，信任引擎 1405、1410、1415 和 1420 的每个的每个事务引擎的行为与参照图 13 公开的信任引擎 1305 的事务引擎 1321 的行为类似。例如，在认证过程中，信任引擎 1405、1410、1415 和 1420 的每个的每个事务引擎将当前认证数据发送至它们各自的认证引擎并且发送对存储在信任引擎 1405 到 1420 的每个的每个储存器中的随机化数据进行组装的请求。图 14 没有示出所有这些通信，因为这种图示将变得极度复杂。从认证过程继续，储存器的每个然后将它的随机化数据的部分发送至信任引擎 1405 到 1420 的每个的每个认证引擎。信任引擎的每个的每个认证引擎利用它的比较器确定当前认证数据是否与由信任引擎 1405 到 1420 的每个的储存器提供的登记认证数据匹配。根据这个实施例，由每个认证引擎进行的比较的结果然后被发送至其它三个信任引擎的冗余模块。例如，来自信任引擎 1405 的认证引擎的结果被发送至信任引擎 1410、1415 和 1420 的冗余模块。因此，信任引擎 1405 的冗余模块同样地从信任引擎 1410、1415 和 1420 接收认证引擎的结果。

[0207] 图 15 示出了图 14 的冗余模块的框图。该冗余模块包括比较器，该比较器被构造为从三个认证引擎接收认证结果并且将该结果发送至第四个信任引擎的事务引擎。该比较器对来自三个认证引擎的认证结果进行比较，并且如果这些结果中的两个一致，则比较器得出认证结果应该与两个一致认证引擎的认证结果匹配。这个结果然后被发送回与不与所述三个认证引擎关联的信任引擎对应的事务引擎。

[0208] 基于上面的描述，冗余模块从来自优选与该冗余模块的信任引擎在地理上远离的认证引擎接收的数据，确定认证结果。通过提供这种冗余功能，信任引擎系统 1400 确保对信任引擎 1405 到 1420 之一的认证引擎的危害不足以危害该特定信任引擎的冗余模块的认证结果。熟练技术人员将认识到，信任引擎系统 1400 的冗余模块功能还可以应用到信任引擎 1405 到 1420 的每个的密码引擎。然而，在图 14 中没有示出这种密码引擎通信以避免复杂性。此外，熟练技术人员将认识到，针对图 15 的比较器的大量的替代认证结果冲突解决算法适用于本发明。

[0209] 根据本发明的另一个实施例，有利的是，信任引擎系统 1400 在密码比较步骤内利用冗余模块。例如，有利的是，可以在特定事务期间由一方或多方提供的文档的哈希比较过程中实现参照图 14 和图 15 公开的一些或所有的上述冗余模块。

[0210] 尽管针对特定的优选和替代实施例描述了上述发明，但是基于这里的公开本领域普通技术人员将想到其它实施例。例如，信任引擎 110 可以发放短期证书，其中，私有密码密钥发布给用户达到预定时长。例如，当前证书标准包括可以设置为在预定时间量后过期

的有效性字段。因此，信任引擎 110 可以向用户发布私钥，其中，该私钥例如在 24 小时内有效。根据这种实施例，有利的是，信任引擎 110 可以发放与特定用户关联的新的密码密钥对，然后发布该新的密码密钥对的私钥。然后，一旦发布了私有密码密钥，信任引擎 110 使这种私钥的任何内部有效使用立即过期，这是因为它不再能由信任引擎 110 保护。

[0211] 此外，熟练技术人员将认识到，密码系统 100 或信任引擎 110 可以包括识别任何类型的装置（例如但不限于膝上型电脑、蜂窝电话、网络、生物测定装置等）的能力。根据一个实施例，这种识别可来自在对特定服务的请求（例如，对导致访问或使用的认证的请求、对密码功能的请求等）中提供的数据。根据一个实施例，上述请求可以包括唯一装置标识符，例如，处理器 ID。另选地，该请求可以包括特定可识别数据格式的数据。例如，移动和卫星电话常常不包括对全 X509.v3 重加密证书的处理能力，因此不请求它们。根据这个实施例，信任引擎 110 可以识别提供的数据格式的类型，并且仅仅以同样方式进行回应。

[0212] 在上述的系统的附加方面中，可以使用将在下文描述的各种技术提供上下文敏感认证。例如如图 16 所示的上下文敏感认证提供不仅评估当用户尝试认证自身时由用户发送的实际数据还评估围绕该数据的产生和传递的境况的可能性。这些技术还可以支持用户与信任引擎 110 之间或者卖方与信任引擎 110 之间的事务特定信任仲裁，这将在下面描述。

[0213] 如上所述，认证是证明用户是他说他是的那个人的过程。通常，认证要求向认证管理机构展示一些事实。本发明的信任引擎 110 代表用户必须向其认证自己的机构。用户必须通过以下方式向信任引擎 110 展示他就是他说他是的那个人：知道仅该用户应该知道的事物（基于知识的认证），具有仅该用户应该具有的事物（基于令牌的认证），或者通过成为仅该用户应该是的事物（基于生物测定的认证）。

[0214] 基于知识的认证的例子包括但不限于口令、PIN 码或者密码锁。基于令牌的认证的例子包括但不限于房屋钥匙、物理信用卡、驾照或者特定电话号码。基于生物测定的认证的例子包括但不限于指纹、笔迹分析、面部扫描、手扫描、耳扫描、虹膜扫描、血管模式、DNA、语音分析或者视网膜扫描。

[0215] 每种类型的认证具有特定的优点和缺点，并且均提供不同的安全等级。例如，与偷听某人的口令并且重复它相比，建立与他人的指纹匹配的假指纹一般更加困难。每种类型的认证还要求认证机构知道不同类型的数据从而使用该形式的认证来验证某人。

[0216] 本文所用的“认证”广义地是指验证某人的身份为他说他是的那个人的全部过程。“认证技术”是指基于特定知识、物理令牌、或者生物测定读数的特定类型的认证。“认证数据”是指发送至或以其它方式向认证机构进行展示以建立身份的信息。“登记数据”是指初始提交给认证机构以建立与认证数据进行比较的基线的数据。“认证实例”是指与根据认证技术进行认证的尝试关联的数据。

[0217] 参照以上图 10 描述了对用户进行认证的过程中涉及的内部协议和通信。在如图 10 的步骤 1045 所示的比较步骤内发生这个过程的进行上下文敏感认证的部分。这个步骤在认证引擎 215 内发生，并且涉及对从储存器 210 取回的登记数据 410 进行组装并且将由用户提供的认证数据与它进行比较。在图 16 中示出并且在下文描述这个过程的一个特定实施例。

[0218] 在图 16 的步骤 1600 中，认证引擎 215 接收由用户提供的当前认证数据和从储存器取回的登记数据。这两组数据可以包含与不同的认证技术有关的数据。在步骤 1605 中，

认证引擎 215 将与每个个体认证实例关联的认证数据进行分离。这是必要的，从而将认证数据与用户的登记数据的适当子集进行比较(例如，指纹认证数据应该与指纹登记数据而非口令登记数据进行比较)。

[0219] 通常，对用户进行认证涉及一个或多个个体认证实例，这取决于用户可使用哪些认证技术。这些方法受到在用户的登记过程内由用户提供的登记数据的限制(如果用户当登记时没有提供视网膜扫描，则他将不能够使用视网膜扫描认证自己)以及受到用户当前可用的装置的限制(例如，如果用户在他的当前位置没有指纹读取器，则指纹认证是不实际的)。在一些情况下，单个认证实例足以对用户进行认证，然而，在某些境况下，可以使用多个认证实例的组合从而对于特定事务对用户进行更确信的认证。

[0220] 每个认证实例包括与特定认证技术(例如，指纹、口令、智能卡等)有关的数据以及针对该特定技术的围绕数据的捕捉和传递的境况。例如，尝试经由口令进行认证的特定实例不仅将产生与口令自身有关的数据，还将产生与该口令尝试有关的称为“元数据”的境况数据。这个境况数据包括诸如以下的信息：特定认证实例发生的时间、从其传递认证信息的网络地址、以及本领域技术人员已知的关于认证数据的起源可以确定的任何其它信息(连接的类型、处理器序列号等)。

[0221] 在许多情况下，仅仅少量的境况元数据可用。例如，如果用户位于使用掩蔽起源计算机的地址的代理服务器或网络地址翻译或另一技术的网络上，则仅仅可以确定代理服务器或路由器的地址。类似地，在许多情况下，例如处理器序列号的信息由于使用的硬件或操作系统的限制(系统运营商禁用这些特征)或者用户的系统与信任引擎 110 之间的连接的其它限制而不可用。

[0222] 如图 16 所示，一旦在步骤 1605 中提取并分离了在认证数据内表示的各个认证实例，认证引擎 215 针对每个实例的指示用户是他宣称的那个人的可靠性进行评估。通常，基于几个因素确定单个认证实例的可靠性。可将它们分组为在步骤 1610 中评估的和与认证技术关联的可靠性有关的因素以及在步骤 1815 中评估的与提供的特定认证数据的可靠性有关的因素。第一组包括但不限于正使用的认证技术的固有可靠性以及与该方法一起使用的登记数据的可靠性。第二组包括但不限于登记数据与由认证实例提供的数据之间的匹配度以及与该认证实例关联的元数据。这些因素中的每一个可以独立于其它因素而变化。

[0223] 认证技术的固有可靠性基于冒名顶替者提供他人的正确数据的难度以及认证技术的整体误差率。对于基于口令和知识的认证方法，这种可靠性常常是相当低的，这是因为无法防止某人向另一个人泄露他们的口令以及该第二人使用该口令。即使是更加复杂的基于知识的系统仍可能仅具有中等可靠性，这是因为知识从一个人传递到另一人是相当容易的。基于令牌的认证(例如，具有正确智能卡或者使用特定终端执行认证)在独自使用时的可靠性同样低，这是因为不能够保证正当的人拥有正确令牌。

[0224] 然而，生物测定技术的固有可靠性更高，因为一般难以以方便的方式(甚至是有意地)向他人提供使用你的指纹的能力。由于破坏生物测定认证技术更加困难，所以生物测定方法的固有可靠性通常高于纯粹基于知识或令牌的认证技术。然而，即使是生物测定技术也会有一些产生误接受或误拒绝的情况。通过相同生物测定技术的不同实施方式的不同可靠性，可以反映出这些情况。例如，由一个公司提供的指纹匹配系统可以提供比由一个不同公司提供的指纹匹配系统更高的可靠性，因为一个公司使用了更高质量的光学部件或更好

的扫描分辨率或者减少误接受或误拒绝的发生的一些其它改进。

[0225] 要注意，该可靠性可以通过不同方式进行表示。期望以可由启发法 530 和认证引擎 215 的算法用来计算每个认证的置信等级的一些度量来表示可靠性。表示这些可靠性的一个优选方式是百分比或分数。例如，可能向指纹被分配 97% 的固有可靠性，而可能仅向口令分配 50% 的固有可靠性。本领域技术人员将认识到，这些特定值仅仅是示例性的并且可以在特定实施方式之间变化。

[0226] 必须评估可靠性的第二因素是登记的可靠性。这是上文提及的“分级登记”过程的一部分。该可靠性因素反映在初始登记过程内提供的标识的可靠性。例如，如果个人最初以他们用身体向公证人或其它公共官员生成他们的身份的证据的方式进行登记并且此时记录并公证登记数据，则该数据的可靠性要强于经由网络在登记过程中提供并且仅通过数字签名或不真实依赖于该个人的其它信息进行担保的数据。

[0227] 具有不同可靠性等级的其它登记技术包括但不限于：在信任引擎 110 运营商的物理办公室进行登记、在用户的雇佣地点进行登记、在邮局或护照办公室进行登记、通过附属或信任方向信任引擎 110 运营商进行登记、登记身份还没有被识别为真实个人的匿名或笔名登记、以及本领域知道的其它方式。

[0228] 这些因素反映信任引擎 110 与在登记过程期间提供的标识的源之间的信任。例如，如果在提供身份证明的初始过程中与雇主关联地执行登记，则可认为这个信息对于公司内的用途是非常可靠的，但是对于政府机构或竞争者而言信任程度较低。因此，由这些其它组织的每个操作的信任引擎可以为这个登记分配不同的可靠性等级。

[0229] 类似地，在网络上提交但是通过在先前向同一信任引擎 110 登记过程中提供的其它信任数据进行认证的附加数据可以被认为与原始登记数据一样可靠，即使后一数据是在开放网络上提交的。在这种境况下，随后的公证将有效地提高与原始登记数据关联的可靠性的等级。这样，例如，通过向某登记官员展示与登记的数据匹配的个人的身份，匿名或笔名登记然后可被提升至全登记。

[0230] 通常，上述的可靠性因素是在任何特定认证实例之前确定的值。这是因为它们基于登记和技术而非实际认证。在一个实施例中，基于这些因素产生可靠性的步骤包括查找针对这个特定认证技术的先前确定的值以及用户的登记数据。在本发明的有利实施例的另一个方面中，这些可靠性可以包括在登记数据自身内。这样，这些因素连同从储存器 210 发送的登记数据被自动传递给认证引擎 215。

[0231] 尽管通常可以在任何个体认证实例之前确定这些因素，但是它们仍然对针对该用户使用该特定认证技术的每个认证实例具有影响。另外，尽管这些值可以随时间而变化(例如，如果用户以更可靠的方式重新登记)，但是它们并不取决于认证数据自身。通过对比，与单个特定实例的数据关联的可靠性因素对于每个场合会变化。在步骤 1815 中，针对每个新的认证必须对下述的这些因素进行评估从而产生可靠性得分。

[0232] 认证数据的可靠性反映了在特定认证实例中由用户提供的数据与在认证登记过程内提供的数据之间的匹配。这是认证数据是否与用户所宣称的个人的登记数据匹配的基本提问。通常，当数据不匹配时，认为用户没有得到成功认证，并且认证失败。对此进行评估的方式可以根据使用的认证技术而改变。通过如图 5 所示的认证引擎 215 的比较器 515 功能执行这种数据的比较。

[0233] 例如，通常以二元方式对口令的匹配进行评估。换言之，口令要么是完全匹配，要么是失败匹配。如果口令不是完全正确的，即使是部分匹配(接近于正确口令的口令)，通常也是不可以接受的。因此，当对口令认证进行评估时，由比较器 515 返回的认证的可靠性通常要么是 100% (正确) 要么是 0% (错误)，而不存在中间值的可能性。

[0234] 通常，与口令的规则类似的规则应用于基于令牌的认证方法(例如，智能卡)。这是因为具有一具有相似标识符或者与正确智能卡类似的智能卡，仍然与具有任何其它不正确令牌一样是错误的。因此，令牌同样趋于二元认证 :用户要么具有正确令牌，要么没有。

[0235] 然而，某些类型的认证数据(例如，问卷和生物测定) 通常不是二元认证者。例如，指纹可以以不同程度与基准指纹进行匹配。在一定程度上，这可能是由于在初始登记过程或者在随后的认证中捕捉的数据的质量的变化。(指纹可能被弄脏，或者人在特定手指上具有治疗伤疤或烧伤)。在其它情况下，由于信息自身有些可变并且基于模式匹配，所以数据不会那么完美地匹配。(由于背景噪声、或者记录语音的环境的声学或者由于人感冒，语音分析看起来接近但并非相当正确)。最终，在大量数据进行比较的情况下，就会有如下情况 :很多数据匹配良好，但是一些数据不匹配。(十个提问的问卷会得到针对个人提问的八个正确回答以及两个不正确回答)。针对这些理由中的任何理由，期望由比较器 515 向登记数据与特定认证实例的数据之间的匹配分配一个部分匹配值。这样，例如，可以将指纹说成是 85% 匹配，将声纹说成是 65% 匹配，将问卷说成是 80% 匹配。

[0236] 由比较器 515 生成的这种测量(匹配度) 是表示认证是否正确的基本问题的因素。然而，如上所述，这仅仅是可用于确定给定的认证实例的可靠性的因素之一。还要注意，即使可以确定某部分程度的匹配，最终，希望基于部分匹配提供二元结果。在另一种操作模式下，还可以基于匹配度是否通过特定阈值的匹配水平，将部分匹配视为二元的，即，要么完美(100%)，要么失败(0%)。这种过程可用于向系统提供简单的通过 / 失败等级的匹配，该系统否则以其它方式生成部分匹配。

[0237] 在评估给定的认证实例的可靠性时要考虑的另一个因素涉及提供针对这个特定实例的认证数据的境况。如上所述，这些境况是指与特定认证实例关联的元数据。这可以包括但不限于如下信息 :能够进行确定的认证者的网络地址、认证的时间、认证数据的传输模式(电话线、蜂窝电话、网络等)、以及认证者的系统的序号。

[0238] 这些因素可用于生成由用户通常请求的认证的类型的概况。然后，这个信息能够用于以至少两种方式评估可靠性。一种方式是考虑用户是否正在以与该用户的认证的正常概况一致的方式请求认证。如果用户通常在工作日(当她工作时)向一个网络地址进行认证请求并且在晚间或周末(当她在家时) 向一个不同的网络地址进行认证请求，则在工作日从家庭地址发生的认证是较不可靠的，因为它在正常认证概况之外。类似地，如果用户通常使用指纹生物测定并且在夜间进行认证，则在白天仅仅使用口令发起的认证是较不可靠的。

[0239] 境况元数据可用于评估认证的实例的可靠性的另外方式是确定境况提供认证者是它宣称的个人有多么确实。例如，如果认证来自已知与用户关联的序号的系统，则这是用户是他们宣称的人的良好境况指示符。相反地，如果认证来自已知位于洛杉矶的网络地址而用户已知位于伦敦，则这是该认证基于其境况而较不可靠的指示。

[0240] 当用户与卖方系统或信任引擎 110 进行交互时，cookie 或其它电子数据可以安置在用户使用的系统上。这个数据写入到用户的系统的存储器并且可以包含可由 Web 浏览器

或用户系统上的其它软件读取的标识。如果在会话之间这个数据被允许驻留在用户系统上(“持久 cookie”),则在特定用户的认证过程中,该数据可以与认证数据一起发送,作为这个系统的过去使用的进一步证据。实际上,给定实例的元数据(尤其是持久 cookie)可以自身形成一种基于令牌的认证器。

[0241] 一旦如上所述在步骤 1610 和步骤 1615 中分别产生基于认证实例的技术和数据的适当的可靠性因素,在步骤 1620 中它们用于产生提供的认证实例的总可靠性。这样做的一种方式是以百分比简单表示每个可靠性,然后将它们一起相乘。

[0242] 例如,假定:根据用户的过去认证概况从对于用户的家庭计算机完全已知的网络地址送入认证数据(100%),使用的技术是指纹识别(97%),初始指纹数据由用户的雇主通过信任引擎 110 进行登记(90%),认证数据与登记数据中的原始指纹模板之间的匹配非常好(99%)。这个认证实例的总可靠性于是可以被计算为这些可靠性的乘积(100%×97%×90%×99%=86.4% 可靠性)。

[0243] 这个计算的可靠性表示一个认证的实例的可靠性。还可以使用不同地对待不同可靠性因素的技术,例如,通过使用向每个可靠性因素分配不同权重的公式,来计算一个认证实例的总可靠性。另外,本领域技术人员将认识到,使用的实际值可以表示与百分比不同的值并且可以使用非算术系统。一个实施例可以包括由认证请求者用来设置每个因素的权重和用于建立认证实例的总可靠性的算法的模块。

[0244] 如步骤 1620 所示,认证引擎 215 可以使用以上技术及其变型来确定一个认证实例的可靠性。然而,在许多认证情形下,它对于要同时提供的多个认证实例是有用的。例如,当使用本发明的系统尝试对自身进行认证时,用户可以提供用户标识、指纹认证数据、智能卡和口令。在这种情况下,三个独立的认证实例被提供给信任引擎 110 以进行评估。进行到步骤 1625,如果认证引擎 215 确定由用户提供的数据包括超过一个认证实例,则如步骤 1630 所示将依次选择每个实例并且如上所述在步骤 1610、1615 和 1620 中进行评估。

[0245] 要注意,许多所述的可靠性因素在这些实例中的一个与另一个之间不同。例如,这些技术的固有可靠性以及在认证数据与登记数据之间提供的匹配度很可能不同。另外,用户可能已经在不同时间在不同境况下针对这些技术的每一个提供了登记数据,针对这些实例的每一个也提供不同的登记可靠性。最终,即使这些实例中的每一个实例的数据被提交的境况相同,这些技术的使用也均可以不同地适应用户的概况。(例如,用户可以正常使用他们的口令和指纹,而不是他们的智能卡)。

[0246] 结果,这些认证实例的每一个的最终可靠性会彼此不同。然而,通过一起使用多个实例,认证的总置信等级将趋于增大。

[0247] 一旦认证引擎针对在认证数据中提供的所有的认证实例执行了步骤 1610 到 1620,在步骤 1635 中使用每个实例的可靠性,评估总认证置信等级。将各个认证实例可靠性组合成认证置信等级的这个过程可以通过与生成的各个可靠性相关的各种方法进行模拟,并且还可以解决这些认证技术中的一些之间的特定交互。(例如,与一个口令甚至是例如基本语音分析的相当弱的生物测定相比,诸如口令的多个基于知识的系统可以生成较低的置信度)。

[0248] 认证引擎 215 可以将多个并发认证实例的可靠性进行组合以产生最终置信等级的一个方式是将每个实例的不可靠性相乘以达到总不可靠性。通常,不可靠性是可靠性

的互补百分比。例如，84% 可靠的技术是 16% 不可靠的。生成 86%、75% 和 72% 的可靠性的上述的三个认证实例（指纹、智能卡、口令）分别将具有对应的 (100-86)%、(100-75)% 和 (100-72)%、或者 14%、25% 和 28% 的不可靠性。通过将这些不可靠性相乘，我们得到 14%×25%×28% 的累积不可靠性，即 0.98% 的不可靠性，这对应于 99.02% 的可靠性。

[0249] 在另一种操作模式中，可以在认证引擎 215 内应用附加因素和启发法 530 以解决各种认证技术的依存性。例如，如果某人未经授权地访问了特定家庭计算机，则他们很可能也可访问该地址的电话线。因此，基于发起电话号码以及认证系统的序号的认证不会使认证的总置信度增加多少。然而，基于知识的认证很大程度上独立于基于令牌的认证（即，如果某人盗取了你的蜂窝电话或钥匙，他们在没有你的 PIN 或口令的情况下不大可能知道你的 PIN 或口令）。

[0250] 另外，不同卖方或其它认证请求者可能希望对认证的不同方面进行不同的加权。这可以包括使用单独的加权因子或用于计算各个实例的可靠性的算法以及使用不同方式来评估具有多个实例的认证事件。

[0251] 例如，某些类型的事务的卖方（例如公司电子邮件系统）希望缺省地主要基于启发法和其它境况数据进行认证。因此，他们可以对与元数据和与围绕认证事件的境况关联的其它概况相关信息相关的因素施加高权重。通过与在工作时间内用户登录到正确机器相比不从用户要求更多，这种布置可用于减轻正常工作时间内用户的负担。然而，另一个卖方可能对来自特定技术（例如指纹匹配）的认证进行最重加权，这是由于这种技术最适于出于特定卖方的目的进行认证的策略判断。

[0252] 在一种操作模式下，在产生认证请求时由请求者定义这些不同的权重，并且这些不同的权重与认证请求一起发送至信任引擎 110。在另一种操作模式下，在初始登记过程内针对认证请求者还可将这些选项设置为偏好并且存储在认证引擎内。

[0253] 一旦认证引擎 215 针对提供的认证数据生成了认证置信等级，在步骤 1640 中这个置信等级用于完成认证请求，并且这个信息从认证引擎 215 转发至事务引擎 205，以包含在去往认证请求者的消息内。

[0254] 上述的过程仅仅是示例性的，本领域技术人员将认识到 :这些步骤不需要按照所示顺序执行，或者仅希望执行某些步骤，或者希望步骤的各种组合。另外，如果境况允许，某些步骤（例如，提供的每个认证实例的可靠性的评估）可以彼此并行地执行。

[0255] 在本发明的另一个方面中，提供了一种适应当由上述过程生成的认证置信等级无法满足要求认证的卖方或其它方的要求的信任等级时的情况的方法。在例如在提供的置信等级与希望的信任等级之间存在差距的境况下，信任引擎 110 的运营商能够向一方或双方提供提供替代数据或要求以弥合这个信任差距的机会。这里，这个过程将称作“信任仲裁”。

[0256] 信任仲裁可以发生在以上参照图 10 和图 11 描述的密码认证的框架内。如那里所示，卖方或其它方将请求与特定事务关联的特定用户的认证。在一种境况下，卖方简单地请求认证（要么肯定，要么否定），并且在从用户接收到适当数据后，信任引擎 110 将提供这种二元认证。在例如这些的境况下，基于在信任引擎 110 内设置的偏好，确定为了保护肯定认证所需的置信度。

[0257] 然而，卖方也可以请求特定的信任等级以完成特定事务。这个所需的等级可包括在认证请求内（例如，对这个用户认证达到 98% 置信度），或者可由信任引擎 110 基于与事务

关联的其它因素确定(即,针对这个事务适当地对这个用户进行认证)。一个这种因素可以是事务的经济值。针对具有较大经济值的事务,可能需要更高的信任度。相似地,针对具有高风险度的事务,可能需要高的信任度。相反,针对低风险或者低值的事务,卖方或其它认证请求者可能要求较低的信任等级。

[0258] 信任仲裁的过程发生于在图 10 的步骤 1050 中信任引擎 110 接收认证数据的步骤和在图 10 的步骤 1055 中将认证结果返回到卖方的步骤之间。在这些步骤之间,如图 17 所示发生导致信任等级的评估和潜在信任仲裁的过程。在执行简单的二元认证的境况下,图 17 所示的过程缩减为如上面参照图 10 所述使事务引擎 205 直接将提供的认证数据与识别的用户的登记数据进行比较,将任何差别标记为否定认证。

[0259] 如图 17 所示,在步骤 1050 中接收数据后的第一个步骤是在步骤 1710 中针对这个特定事务由事务引擎 205 确定肯定认证所需的信任等级。这个步骤可通过几种不同方法之一执行。当进行认证请求时,认证请求者可以对信任引擎 110 指定所需的信任等级。认证请求者还可以预先设置偏好,这个偏好存储在可由事务引擎 205 访问的储存器 210 或其它存储器内。然后,每当这个认证请求者进行认证请求时,可以读取和使用这个偏好。该偏好还可以与特定用户进行关联,作为使得总是需要特定的信任等级以对那个用户进行认证的安全性措施,该用户偏好存储在可由事务引擎 205 访问的储存器 210 或其它存储介质中。所需等级还可以由事务引擎 205 或认证引擎 215 基于在认证请求中提供的信息(例如,要认证的事务的值和风险等级)而导出。

[0260] 在一种操作模式下,策略管理模块或当产生认证请求时使用的其它软件用于为事务的认证指定所需的信任度。这可以用于提供当基于在策略管理模块内指定的策略分配所需的信任等级时要遵守的一系列规则。一种有利的操作模式是将这种模块包括在卖方的 Web 服务器内,以针对由卖方的 Web 服务器发起的事务适当地确定所需的信任等级。这样,可以根据卖方的策略向来自用户的事务请求分配所需的信任等级,并且这种信息可以与认证请求一起转发至信任引擎 110。

[0261] 这个所需的信任等级与卖方希望具有的、个人认证实际上是他将自己识别成的那个人的确定度相关。例如,如果事务是卖方希望合理的确定度的事务(由于货物是转手的),则卖方可以要求 85% 的信任等级。对于卖方仅对用户进行认证以允许他观看会员内容或在聊天室实践特权的情形,不利风险足够小从而卖方仅要求 60% 的信任等级。然而,为了参与上万美元的生产合同,卖方可以要求 99% 或更高的信任等级。

[0262] 这个所需的信任等级表示用户必须对自己进行认证以完成事务的度量。例如如果所需的信任等级是 85%,则用户必须向信任引擎 110 提供足以使信任引擎 110 以 85% 置信度说用户是他们说他们是的那个人的认证。这是所需的信任等级与生成肯定认证(达到卖方的满意)或可能的信任仲裁的认证置信等级之间的平衡。

[0263] 如图 17 所示,在事务引擎 205 接收到所需的信任等级后,在步骤 1720 中将所需的信任等级与认证引擎 215 针对当前认证计算的认证置信等级(参照图 16 所讨论的)进行比较。在步骤 1730 中如果认证置信等级高于事务的所需的信任等级,则过程移至步骤 1740,在步骤 1740 中,事务引擎 205 针对这个事务生成肯定认证。表示此意的消息然后将被插入到认证结果中并且由事务引擎 205 返回到卖方,如在步骤 1055 中所示(见图 10)。

[0264] 然而,如果在步骤 1730 中认证置信等级没有满足所需的信任等级,则对于当前认

证存在置信度差距，并且在步骤 1750 中进行信任仲裁。在下文参照图 18 更加完全地描述信任仲裁。下述的这个过程在信任引擎 110 的事务引擎 205 内发生。由于执行信任仲裁不需要认证或其它密码操作(除了事务引擎 205 与其它部件之间的 SSL 通信所需的那些以外)，所以可以在认证引擎 215 之外执行该过程。然而，如下所述，认证数据或其它密码或认证事件的任何重新评估将要求事务引擎 205 向认证引擎 215 重新提交适当的数据。本领域技术人员将认识到，信任仲裁过程可以代替地构造为部分或全部地在认证引擎 215 自身内发生。

[0265] 如上所述，信任仲裁是信任引擎 110 对卖方与用户之间的协商进行调停以尝试适当地保证肯定认证的过程。如在步骤 1805 所示，事务引擎 205 首先确定当前情况是否适于信任仲裁。这可以基于认证的境况(例如，这个认证是否已经经过多个循环的仲裁)以及卖方或者用户的偏好进行确定，这将在下文进一步进行讨论。

[0266] 在不可以仲裁的这些境况下，该过程进行到步骤 1810，在步骤 1810 中，事务引擎 205 产生否定认证，然后将它插入到认证结果，该认证结果在步骤 1055 中发送至卖方(见图 10)。可有利用于防止认证不确定地待决的一个限制是设置从初始认证请求开始的超时时段。这样，在时限内没有得到肯定认证的任何事务被拒绝进一步仲裁并且得到否定认证。本领域技术人员将认识到，这种时限可以根据事务的境况以及用户和卖方的希望而变化。还可以对在提供成功认证时可进行的尝试的数目进行限制。可以通过如图 5 所示的尝试限制器 535 对这些限制进行处理。

[0267] 如果在步骤 1805 中没有禁止仲裁，则事务引擎 205 然后将与事务方之一或二者进行协商。如在步骤 1820 中所示，事务引擎 205 可以向用户发送消息以请求某形式的附加认证从而提高生成的认证置信等级。在最简单的形式中，这可以简单指示认证是不充分的。还可以发送生成一个或多个附加认证实例以提高认证的总置信等级的请求。

[0268] 如果在步骤 1825 中用户提供一些附加认证实例，则事务引擎 205 向事务的认证数据添加这些认证实例并且将它发送至认证引擎 215，如在步骤 1015 所示(见图 10)，并且基于针对这个事务的预先存在的认证实例和新提供的认证实例重新评估该认证。

[0269] 一种附加类型的认证可以是来自信任引擎 110 的用于例如通过电话呼叫在信任引擎 110 运营商(或者信任的合作人)与用户之间进行某形式的人对人联系的请求。这个电话呼叫或其它非计算机认证能够用于提供与个人的私人联系以及执行某形式的基于问卷的认证。这还可以给出验证发起电话号码以及当呼入时潜在的用户语音解析的机会。即使不能够提供附加认证数据，与用户的电话号码关联的附加上下文仍可以提高认证上下文的可靠性。基于这个电话呼叫的任何修正的数据或境况被送入信任引擎 110 以用于对认证请求的考虑。

[0270] 此外，在步骤 1820 中，信任引擎 110 可以向用户提供购买保险的机会，从而有效地购买更加确信的认证。信任引擎 110 的运营商有时可能仅希望使得在认证的置信等级高于某阈值的情况下才开始可用这个选项。实际上，这个用户侧保险是当认证满足信任引擎 110 对于认证的正常所需信任等级但不满足这个事务的卖方的所需信任等级时信任引擎 110 对用户进行担保的方式。这样，即使用户仅仅具有生成针对信任引擎 110 足够的置信度的认证实例，他仍可以成功地认证为卖方要求的高等级。

[0271] 信任引擎 110 的这个功能使得信任引擎 110 可以对被认证为满足信任引擎 110 而

非满足卖方的某人进行担保。这与在向文档添加公证人的签名以向以后读取该文档的某人指示签名出现在文档上的人实际是对它进行签名的人时由公证人执行的功能类似。公证人的签名证实用户签名的动作。以相同的方式，信任引擎提供执行事务的人是他们说他们是的人的指示。

[0272] 然而，因为信任引擎 110 人工推升用户提供的置信等级，所以对于信任引擎 110 运营商存在更大的风险，因为用户实际上没有满足卖方的所需的信任等级。保险的花费被设计为抵消误肯定认证对于信任引擎 110（其可有效地对用户的认证进行公证）的风险。用户向信任引擎 110 运营商付费以承担认证为比实际提供的还要高的置信等级的风险。

[0273] 由于这种保险系统允许某人从信任引擎 110 有效地购买更高的置信等级，所以卖方和用户都会希望防止在某些事务中使用用户侧保险。卖方可能希望将肯定认证限制到它们知道实际认证数据支持它们要求的置信度的情况，因此可能向信任引擎 110 指示用户侧保险不被允许。类似地，为了保护他的在线身份，用户可能希望防止在他的帐户上使用用户侧保险或者可能希望将它的使用限制到没有保险的认证置信等级高于一定限制的情形。这可用作防止某人偷听口令或盗取智能卡并使用它们假冒认证到低置信等级并且然后购买保险以生成非常高的(假)置信等级的安全措施。在确定是否允许用户侧保险时可以对这些因素进行评估。

[0274] 如果在步骤 1840 中用户购买了保险，则在步骤 1845 中基于购买的保险调整认证置信等级，并且在步骤 1730 中把认证置信等级与所需的信任等级再次进行比较(见图 17)。该过程从步骤 1730 继续，并且可能导致步骤 1740 中的肯定认证(见图 17)或者返回步骤 1750 中的信任仲裁过程从而进一步进行仲裁(如果允许的话)，或者如果进一步仲裁被禁止则导致步骤 1810 中的否定认证。

[0275] 除了在步骤 1820 中向用户发送消息以外，在步骤 1830 中事务引擎 205 还可以向卖方发送指示未决认证当前低于所需信任等级的消息。该消息还可以提供关于如何前进到卖方的各种选项。这些选项之一是简单地向卖方通知当前认证置信等级是什么以及询问卖方是否希望维持他们的当前未满足的所需信任等级。这是有益的，因为在一些情况下，卖方可以具有用于对事务进行认证的独立手段或者可能已经使用缺省的一组要求，该组要求通常导致初始指定比手边的特定事务实际需要的更高的所需等级。

[0276] 例如，与卖方的所有输入购买订单事务预计满足 98% 信任等级是标准实践。然而，如果通过电话在卖方与长期顾客之间最近讨论了订单，并且紧接之后该事务得到认证，但是仅仅认证到 93% 置信等级，则卖方可能希望简单降低这个事务的接受阈值，因为电话呼叫有效地向卖方提供了附加认证。在某些境况下，卖方会愿意降低他们的所需信任等级，但是并不总是降低到当前认证置信等级。例如，以上例子中的卖方可能认为，在订单之前的电话呼叫可能值所需信任度下降 4%，然而，这仍大于用户生成的 93% 的置信度。

[0277] 如果在步骤 1835 中卖方确实调整了他们的所需的信任等级，则在步骤 1730 中对由认证生成的认证置信等级与该所需的信任等级进行比较(见图 17)。如果置信等级现在超过所需的信任等级，则在步骤 1740 中在事务引擎 205 中可产生肯定认证(见图 17)。如果没有超过，则在允许的情况下可以如上所述尝试进一步仲裁。

[0278] 除了请求对所需的信任等级进行调整外，事务引擎 205 还可以向请求认证的卖方提供卖方侧保险。这个保险用于与以上针对用户侧保险描述的用途相同的用途。这里，然

而，成本不对应于由信任引擎 110 在以上认证生成的实际认证置信等级时所承担的风险，保险的成本对应于卖方在认证中接受低信任等级时所承担的风险。

[0279] 不是仅仅降低他们的实际所需的信任等级，卖方可以选择购买保险以保护自己免受与用户认证中的低信任等级关联的附加风险。如上所述，在现有认证已经高于某阈值的条件下，卖方仅仅考虑购买这种保险来覆盖信任差距是有利的。

[0280] 得到这种卖方侧保险，允许卖方有如下选项：在对自己无附加成本的情况下直接降低他的信任要求；自己承受误认证的风险（基于所需的低信任等级）；或者针对认证置信等级与他的要求之间的信任差距购买保险，其中由信任引擎 110 运营商承担已经提供的低置信等级的风险。通过购买保险，卖方有效地保持他的高信任等级要求，因为误认证的风险被转移给信任引擎 110 运营商。

[0281] 如果在步骤 1840 中卖方购买了保险，则在步骤 1730 中对认证置信等级与所需的信任等级进行比较（见图 17），并且该过程如上所述继续。

[0282] 要注意，用户和卖方均响应来自信任引擎 110 的消息也是可以的。本领域技术人员将认识到，有多种方式可以处理这些情形。处理多个响应的可能性的一种有利模式是简单地以先来先服务方式处理响应。例如，如果卖方以降低的所需信任等级进行响应并且用户之后还立即购买保险以提升他的认证等级，则首先基于来自卖方的降低的信任要求对认证进行重新评估。如果认证现在是肯定的，则用户的保险购买被忽略。在另一种有利的操作模式下，可能仅针对满足新的降低的卖方的信任要求所需的保险等级向用户收费（如果即使在降低了卖方信任要求的情况下仍然存在信任差距的话）。

[0283] 如果在步骤 1850 中在信任仲裁过程中在针对认证设置的时限内没有从任何一方接收到响应，则在步骤 1805 中对仲裁进行重新评估。这有效地再次开始仲裁过程。如果在步骤 1805 中时限终止或者其它境况防止进一步仲裁，则在步骤 1810 中由事务引擎 205 产生否定认证并且在步骤 1055 中返回到卖方（见图 10）。否则，新消息可以发送至用户和卖方，并且该过程可以按期望被重复。

[0284] 要注意，对于某些类型的事务，例如，对不是事务的一部分的文档进行数字签名，并不一定有卖方或其它第三方，因此事务主要在用户与信任引擎 110 之间。在例如这些的境况下，信任引擎 110 将具有必须被满足以产生肯定认证的它自身的所需的信任等级。然而，在这些境况下，常常不希望信任引擎 110 向用户提供保险以使他可以提升他自己签名的置信度。

[0285] 可以使用在上文中参照信任引擎 110 描述的各种通信模式，执行上述以及在图 16 到图 18 中所示的过程。例如，这些消息可以是基于 web 的并且使用信任引擎 110 与实时下载到在用户或卖方系统上运行的浏览器的 Java 小程序之间的 SSL 连接进行发送。在另一种操作模式下，便于进行这种仲裁和保险事务的某些专用的应用可以由用户和卖方使用。在另一种操作模式下，安全电子邮件操作可用于对上述的仲裁进行调停，由此实现延迟评估和认证的批处理。本领域技术人员将认识到，不同的通信模式可适用于卖方的认证要求和境况。

[0286] 下面参照图 19 的说明描述了整合上述本发明的各个方面的样本事务。这个例子示出了由信任引擎 110 进行调停的用户与卖方之间的整个过程。尽管以上详细描述的各个步骤和部件可用于执行下面的事务，但是所示的过程集中于信任引擎 110、用户以及卖方之

间的交互。

[0287] 在步骤 1900 中，当用户在在线观看网页时在卖方的网站上填写订单表格时，事务开始。用户希望将签写有他的数字签名的这个订单表格提交给卖方。为了这样做，在步骤 1905 中，用户向信任引擎 110 提交订单表格和他对签名的请求。用户还将提供将如上所述用于对他的身份进行认证的认证数据。

[0288] 在步骤 1910 中，如上所述由信任引擎 110 将认证数据与登记数据进行比较，并且如果生成了肯定认证，则用用户的私钥签名的订单表格的哈希值与订单表格自身一起被转发送卖方。

[0289] 在步骤 1915 中，卖方接收到签名的表格，然后在步骤 1920 中，卖方将产生发票或与要进行的购买有关的其它合同。在步骤 1925 中，这个合同以及对签名的请求被发送回用户。在步骤 1930 中，卖方还向信任引擎 110 发送针对这个合同事务的认证请求（包括将由双方签名的合同的哈希值）。为了允许由双方对合同进行数字签名，卖方还包括针对自身的认证数据从而使得能够在以后对合同上的卖方的签名进行验证（如果需要的话）。

[0290] 如上所述，信任引擎 110 然后对由卖方提供的认证数据进行验证以确认卖方的身份，并且如果在步骤 1935 中该数据生成肯定认证，则当从用户接收到数据时从步骤 1955 继续。如果卖方的认证数据没有与卖方的登记数据匹配达到期望程度，则将一消息返回给卖方以请求进一步认证。如上所述，为了使得卖方向信任引擎 110 成功地认证自身，如果需要的话在这里可以执行信任仲裁。

[0291] 当在步骤 1940 中用户接收到合同时，他检查合同，在步骤 1945 中如果合同可以接受则产生认证数据以对它进行签名，然后在步骤 1950 中，向信任引擎 110 发送合同的哈希值以及他的认证数据。在步骤 1955 中，信任引擎 110 验证该认证数据，并且如果认证良好，则如下所述继续处理合同。如以上参照图 17 和图 18 所述，可以适当地执行信任仲裁以弥合认证置信等级与事务的所需的认证等级之间存在的任何信任差距。

[0292] 在步骤 1960 中，信任引擎 110 用用户的私钥对合同的哈希值进行签名，并且将该签名的哈希值发送至卖方，代表自己对完整消息进行签名，即，包括利用信任引擎 110 的私钥 510 加密的完整消息（包括用户的签名）的哈希值。在步骤 1965 中，这个消息由卖方接收。该消息代表签名的合同（使用用户的私钥加密的合同的哈希值）以及来自信任引擎 110 的收据（包括使用信任引擎 110 的私钥加密的签名的合同的消息的哈希值）。

[0293] 在步骤 1970 中，信任引擎 110 利用卖方的私钥类似地准备合同的哈希值，并且将由信任引擎 110 签名的这个合同转发至用户。这样，在步骤 1975 中，用户还接收由卖方签名的合同的副本以及由信任引擎 110 签名的传递签名的合同的收据。

[0294] 除了上述以外，本发明的附加方面提供了一种密码服务提供模块（SPM），该密码服务提供模块（SPM）可由客户机侧应用用作访问上述的由信任引擎 110 提供的功能的装置。密码 SPM 提供这种服务的一个有利方式是对第三方应用编程接口（API）与可经由网络或其它远程连接进行访问的信任引擎 110 之间的通信进行调停。在下文中参照图 20 描述样本密码 SPM。

[0295] 例如，在典型系统上，程序员可使用多个 API。每个 API 提供可由在系统上运行的应用 2000 进行的一组功能调用。提供适于密码功能、认证功能和其它安全功能的编程接口的 API 的例子包括由微软在它的 Windows 操作系统中提供的密码 API （CAPI）2010 和由

IBM、Intel 和开放组的其它成员发起的公共数据安全架构(CDSA)。在下文的讨论中，CAPI 将用作示例性安全 API。然而，还可以与 CDSA 或者本领域已知的其它安全 API 一起使用所述的密码 SPM。

[0296] 当调用密码功能时，用户系统 105 或卖方系统 120 使用这个 API。包括在这些功能中的可以是与执行各种密码操作(诸如用特定密钥对文档进行加密，对文档进行签名，请求数字证书，验证签名的文档上的签名)关联的请求以及本文所述或本领域技术人员知道的其它密码功能。

[0297] 通常，在 CAPI 2010 所处于的系统本地执行这些密码功能。这是因为一般调用的功能要求使用本地用户系统 105 的资源(例如，指纹读取器)或者使用在本地机器上执行的库进行编程的软件功能。通常由以上提及的提供执行密码功能使用的资源的一个或多个服务提供模块(SPM)2015、2020 执行对这些本地资源的访问。这些 SPM 可以包括执行加密或解密操作的软件库 2015 或者能够访问专用硬件 2025 (例如，生物测定扫描装置)的驱动程序和应用 2020。与 CAPI 2010 提供可由系统 105 的应用 2000 使用的功能的方式非常类似，SPM 2015、2020 向 CAPI 提供对与系统上的可用服务关联的低级功能和资源的访问。

[0298] 根据本发明，可以提供一种密码 SPM 2030，其能够访问由信任引擎 110 提供的密码功能并且使得应用 2000 通过 CAPI 2010 可获得这些功能。与 CAPI 2010 仅仅能够访问可通过 SPM 2015 和 2020 在本地获得的资源的实施例不同，这里所述的密码 SPM 2030 将能够向位于远处的可进行网络访问的信任引擎 110 提交对密码操作的请求以执行希望的操作。

[0299] 例如，如果应用 2000 需要密码操作，例如对文档进行签名，则应用 2000 对适当的 CAPI 2010 功能进行功能调用。CAPI 2010 继而将执行这个功能，利用通过 SPM 2015 和 2020 以及密码 SPM 2030 使其可获得的资源。在数字签名功能的情况下，密码 SPM 2030 将产生将通过通信链路 125 发送至信任引擎 110 的适当请求。

[0300] 在密码 SPM 2030 与信任引擎 110 之间发生的操作是可以在任何其它系统与信任引擎 110 之间进行的相同操作。然而，通过 CAPI 2010 可以使得用户系统 105 有效地获得这些功能，从而使这些功能看起来在用户系统 105 自己的本地可获得。然而，与普通的 SPM 2015 和 2020 不同，这些功能在远程信任引擎 110 上执行，并且响应于适当请求，结果经由通信链路 125 中继到密码 SPM 2030。

[0301] 这个密码 SPM 2030 使用户系统 105 或卖方系统 120 可以获得以其它方式可能无法获得的大量操作。这些功能包括但不限于：文档的加密和解密、数字证书的发放、文档的数字签名、数字签名的验证、以及对本领域技术人员显而易见的其它操作。

[0302] 在一个单独的实施例中，本发明包括对任何数据集执行本发明的数据保护方法的完整系统。这个实施例的计算机系统包括数据分裂模块，该数据分裂模块包括图 8 所示和本文所述的功能。在本发明的一个实施例中，在本文中有时称作安全数据解析器的数据分裂模块包括包含数据分裂、加密和解密、重构或重装功能的解析器程序或软件套装。这个实施例还可以包括一个数据存储设施或多个数据存储设施。数据分裂模块或安全数据解析器包括跨平台软件模块套装，该跨平台软件模块套装集成在电子基础设施内或者作为要求其数据元素的最大安全性的任何应用的附件。这个解析过程对任何类型的数据集，以及对任何和所有文件类型，或者在数据库中对该数据库中的任何数据行或列或单元进行运算。

[0303] 在一个实施例中，可以以模块分层方式设计本发明的解析过程，并且任何加密过

程适用于本发明的过程。本发明的解析和分裂过程的模块层可以包括但不限于：1）密码分裂，分散并安全存储在多个位置；2）加密，密码分裂，分散并安全存储在多个位置；3）加密，密码分裂，对每份加密，然后分散并安全存储在多个位置；以及 4）加密，密码分裂，用与在第一步骤中使用的加密不同类型的加密对每份进行加密，然后分散并安全存储在多个位置。

[0304] 在一个实施例中，这个过程包括根据产生的随机数或密钥的内容对数据进行分裂，并且对把要保护的数据分裂成两个或更多部分或份的解析或分裂数据（在一个实施例中，优选分裂成四个或更多部分的解析和分裂数据）的加密中使用的密钥执行相同的密码分裂，对所有的部分进行加密，然后根据请求者对隐私和安全的需要，将这些部分分散并存储回数据库中或者将它们重新定位到固定或可移动的任何指定装置。或者，在另一个实施例中，在由分裂模块或安全数据解析器对数据集进行分裂之前可以进行加密。如在这个实施例中所述进行处理的原始数据被加密并打乱并且得到保护。实际上，如果希望的话，加密的元素可以分散到任何地方，包括但不限于单个服务器或数据存储装置、或者多个独立的数据存储设施或装置之间。在一个实施例中，加密密钥管理可以包括在软件套装内，或者在另一个实施例中，可以集成到现有的基础设施或任何其它期望位置。

[0305] 密码分裂（密码术分裂）将数据划分成 N 份。该划分可以基于任何大小的数据单元，包括一个比特、多个比特、字节、千字节、兆字节或更大单元以及预定或随机产生的数据单元大小的任何模式或组合。基于随机或者预定的一组值，这些单元还可以具有不同的大小。这意味着数据能够被看作是这些单元的序列。按照这种方式，例如通过使用一个或多个预定或随机产生的数据单元大小的模式、序列或组合，数据单元自身的大小可以使数据更加安全。这些单元然后（随机地或者根据一组预定值）被分布到 N 份中。该分布还可以涉及在各份中搅乱（shuffle）单元的顺序。本领域普通技术人员易于明白，可以根据多种多样的可能选择（包括但不限于固定大小、预定大小、或者预定或随机产生的数据单元大小的一个或多个组合、模式或序列）执行将数据单元分布到多份中。

[0306] 这种密码分裂过程或密码术分裂的一个例子将考虑数据大小为 23 个字节，数据单元大小选择为 1 个字节，并且选择的份数为 4。每个字节将被分布到这 4 份之一中。假定随机分布，将获得密钥以建立 23 个随机数的序列（r1、r2、r3 到 r23），其中，每个随机数具有与这 4 份对应的 1 至 4 的值。数据的每个单元（在这个例子中，数据的 23 个独立字节）与对应于 4 份之一的 23 个随机数之一关联。通过将数据的第一字节安置到份号 r1、将第二字节安置到份 r2，将第三字节安置到份 r3 只至将数据的第 23 字节安置到份 r23，可以将数据的各字节分布到这 4 份中。本领域普通技术人员易于理解，多种多样的其它可行步骤或者步骤的组合或序列（包括数据单元的大小）可用于本发明的密码术分裂过程，并且以上例子是用于对数据进行密码术分裂的一个过程的非限制性描述。为了重建原始数据，将执行反向操作。

[0307] 在本发明的密码术分裂过程的另一个实施例中，用于密码术分裂过程的选项是在各份中提供充足冗余从而使得仅仅需要这些份的子集就能够将数据重装或恢复成它的原始或可用形式。作为一个非限制性例子，可以按照“4 取 3”密码术分裂进行密码术分裂，从而使得将数据重装或恢复成它的原始或可用形式仅仅需要这 4 份中的 3 份。这还称作“N 取 M 密码术分裂”，其中，N 是总份数，M 至少比 N 小 1。本领域普通技术人员易于理解，在本

发明的密码术分裂过程中建立这种冗余存在许多可能性。

[0308] 在本发明的密码术分裂过程的一个实施例中，数据的每个单元存储在两份（主要份和备份份）中。使用上述的“4 取 3”密码术分裂过程，可以丢失任何一份，并且由于仅仅需要全部 4 份中的 3 份，所以在没有丢失数据单元的情况下这仍足以重装或恢复原始数据。如本文所述，对应于这些份之一产生随机数。基于密钥，该随机数与数据单元关联并且存储在对应份中。在这个实施例中，一个密钥用于产生主要份和备份份随机数。如本文所述，对于本发明的密码术分裂过程，产生与数据单元的数目相等的从 0 到 3 的一组随机数（还称作主要份数字）。然后，产生与数据单元的数目相等的从 1 到 3 的另一组的随机数（还称作备份份数字）。然后，将数据的每个单元与主要份数字和备份份数字进行关联。或者，可以产生少于数据单元的数目的一组随机数并且重复该随机数组，但是这会降低敏感数据的安全性。主要份数字用于确定数据单元存储在哪个份中。备份份数字与主要份数字进行组合以建立 0 至 3 的第三份数字，并且这个数字用于确定数据单元存储在哪个份中。在这个例子中，用于确定第三份数字的式子是：

[0309] （主要份数字＋备份份数字）MOD 4= 第三份数字

[0310] 在上述的主要份数字为 0 至 3 且备份份数字为 1 至 3 的实施例中，确保了第三份数字与主要份数字不同。这导致数据单元存储在两个不同份中。本领域普通技术人员易于理解，除了本文公开的实施例以外，还有执行冗余密码术分裂和非冗余密码术分裂的许多方式。例如，可以使用不同的算法搅乱每份中的数据单元。例如，当原始数据分裂成多个数据单元时，或者在将数据单元安置到各份中以后，或者份已满以后，可以执行该数据单元搅乱。

[0311] 可以对任何大小的数据单元（包括但不限于一个比特、多个比特、字节、千字节、兆字节或更大）执行本文所述的各种密码术分裂过程和数据搅乱过程以及本发明的密码术分裂和数据搅乱方法的所有其它实施例。

[0312] 执行本文所述的密码术分裂过程的源代码的一个实施例的例子是：

[0313]

```
DATA [1:24]  - 具有要分裂的数据的字节数组
SHARES[0:3;1:24]  - 2 维数组，每行表示各份之一
RANDOM[1:24]  - 0..3 的范围内的数组随机数
S1=1;
S2=1;
S3=1;
S4=1;
```

[0314]

```
For J=1 to 24 do
    Begin
    IF RANDOM[J]==0 then
        Begin
        SHARES[1,S1]=DATA[J];
        S1=S1+1;
        End
    ELSE IF RANDOM[J]==1 then
        Begin
        SHARES[2,S2]=DATA[J];
        S2=S2+1;
        END
     ELSE IF RANDOM[J]==2 then
        Begin
        SHARES[3,S3]=data[J];
        S3=S3+1;
        End
     Else begin
        SHARES[4,S4]=data[J];
        S4=S4+1;
        End;
     END;
```

[0315] 执行本文所述的密码术分裂 RAID 过程的源代码的一个实施例的例子是：

[0316] 产生两组数字，PrimaryShare 是 0 到 3，BackupShare 是 1 到 3。然后，按照与上述的密码术分裂相同的过程，将每个数据单元放入 share[primaryshare[1]] 以及 share[(primaryshare[1]+backupshare[1])mod 4] 中。这个方法可调整至任何大小 N，其中，恢复数据仅仅需要 N-1 份。

[0317] 加密的数据元素的获取、重组、重装或重构可以利用任何数目的认证技术，包括但不限于生物测定，例如，指纹识别、面部扫描、手扫描、虹膜扫描、视网膜扫描、耳扫描、血管模式识别或 DNA 分析。根据需要，本发明的数据分裂和 / 或解析器模块可以集成到多种多样的基础产品或应用中。

[0318] 现有技术中已知的传统加密技术依赖于用于对数据进行加密的一个或多个密钥

并且使得在无密钥的情况下不可用。然而，数据仍然是一个整体并且完整，并且易受攻击。在一个实施例中，通过对加密的文件执行密码解析并分裂成两个或更多个部分或份(在另一个实施例中，优选为4个或更多份)，对每个数据份增加另一层加密，然后将这些份存储在不同物理和/或逻辑位置，本发明的安全数据解析器解决了这个问题。当通过使用可移动装置(例如，数据存储装置)或者通过在另一方的控制下对份进行安置从系统中物理地去除一个或多个数据份时，有效去除了对被保护数据的危害的任何可能性。

[0319] 在图21中示出并在下文中描述本发明的安全数据解析器的一个实施例的例子以及如何利用它的例子。然而，本领域普通技术人员易于理解，除了下面的非限制性例子以外，还可以以多种多样的方式利用本发明的安全数据解析器。作为一种部署选项，在一个实施例中，可以通过外部会话密钥管理或者会话密钥的安全内部存储，实现安全数据解析器。当实现时，将产生解析器主密钥，它将用于保护应用以及加密目的。还应该注意，通过将解析器主密钥加入到得到的保护数据中，可以实现由工作组、企业或扩展受众内的个人共享被保护数据的灵活性。

[0320] 如图21所示，本发明的这个实施例示出了由安全数据解析器对数据执行存储会话主密钥与解析的数据的过程的步骤：

[0321] 1. 产生会话主密钥并且使用RS1流密码对数据进行加密。

[0322] 2. 根据会话主密钥的模式将得到的加密的数据分离成四份或部分的解析数据。

[0323] 3. 在本发明的这个实施例中，会话主密钥将与保护数据份一起存储在数据储存器内。根据解析器主密钥的模式对会话主密钥进行分离并且将密钥数据附于加密的解析数据。

[0324] 4. 得到的四个数据份将包含加密的原始数据的部分以及会话主密钥的部分。针对这四个数据份的每一个产生流密码密钥。

[0325] 5. 对每份进行加密，然后将加密密钥存储在与加密的数据部分或份不同的位置：份1获得密钥4，份2获得密钥1，份3获得密钥2，份4获得密钥3。

[0326] 为了恢复原始数据格式时，颠倒这些步骤。

[0327] 本领域普通技术人员易于理解，根据需要，本文所述的方法的某些步骤可以以不同顺序执行或者重复多次。本领域技术人员还易于理解，可以以彼此不同的方式对这些数据部分进行处理。例如，可以仅仅对解析数据的一个部分执行多个解析步骤。可以通过任何希望的方式对解析数据的每个部分进行独特的保护，只要数据可以被重装置、重构、重形成、解密或恢复到它的原始或其它可用形式即可。

[0328] 如图22所示和本文所述，本发明的另一个实施例包括由安全数据解析器对数据执行的将会话主密钥数据存储在一个或多个单独的密钥管理表中的过程的步骤：

[0329] 1. 产生会话主密钥并且使用RS1流密码对数据进行加密。

[0330] 2. 根据会话主密钥的模式将得到的加密的数据分离成四份或部分的解析数据。

[0331] 3. 在本发明的该方法的这个实施例中，会话主密钥将存储在数据储存器中的单独的密钥管理表内。为这个事务产生唯一事务ID。将事务ID和会话主密钥存储在单独的密钥管理表中。根据解析器主密钥的模式将事务ID进行分离并且将该数据附于加密的解析数据或分离的数据。

[0332] 4. 得到的四个数据份将包含加密的原始数据的部分和事务ID的部分。

[0333] 5. 针对这四个数据份的每一个产生流密码密钥。

[0334] 6. 对每份进行加密，然后将加密密钥存储在与加密的数据部分或份不同的位置：份 1 获得密钥 4，份 2 获得密钥 1，份 3 获得密钥 2，份 4 获得密钥 3。

[0335] 为了恢复原始数据格式，需要颠倒这些步骤。

[0336] 本领域普通技术人员易于理解，根据需要，本文所述的方法的某些步骤可以以不同顺序执行或者重复多次。本领域技术人员还易于理解，可以以彼此不同的方式对这些数据部分进行处理。例如，可以仅仅对解析数据的一个部分执行多个分离或解析步骤。可以通过任何希望的方式对解析数据的各个部分进行独特保护，只要数据可以被重装、重构、重形成、解密或恢复到它的原始或其它可用形式即可。

[0337] 如图 23 所示，本发明的这个实施例示出了由安全数据解析器对数据执行的存储会话主密钥与解析数据的过程的步骤：

[0338] 1. 访问与认证的用户关联的解析器主密钥。

[0339] 2. 产生唯一会话主密钥。

[0340] 3. 从解析器主密钥与会话主密钥的异或函数导出中间密钥。

[0341] 4. 使用现有或新的加密算法以中间密钥为密钥对数据进行可选加密。

[0342] 5. 根据中间密钥的模式将得到的可选加密的数据分离成四份或部分的解析数据。

[0343] 6. 在该方法的这个实施例中，会话主密钥将与被保护数据份一起存储在数据储存器内。根据解析器主密钥的模式对会话主密钥进行分离并且将密钥数据附于可选加密的解析数据份。

[0344] 7. 得到的多个数据份将包含可选加密的原始数据的部分和会话主密钥的部分。

[0345] 8. 可选地，为四个数据份的每一个产生加密密钥。

[0346] 9. 可选地，用现有或新的加密算法对每个份进行加密，然后将加密密钥存储在与加密的数据部分或份不同的位置：例如，份 1 获得密钥 4，份 2 获得密钥 1，份 3 获得密钥 2，份 4 获得密钥 3。

[0347] 为了恢复原始数据格式时，需要颠倒这些步骤。

[0348] 本领域普通技术人员易于理解，根据需要，本文所述的方法的某些步骤可以以不同顺序执行或者重复多次。本领域技术人员还易于理解，可以以彼此不同方式对这些数据部分进行处理。例如，可以仅仅对解析数据的一个部分执行多个解析步骤。可以以任何希望的方式对解析数据的每个部分进行独特的保护，只要数据可以被重装、重构、重形成、解密或恢复到它的原始或其它可用形式即可。

[0349] 如图 24 所示和文本所述，本发明的另一个实施例包括由安全数据解析器对数据执行的将会话主密钥数据存储在一个或多个单独的密钥管理表中的过程的步骤：

[0350] 1. 访问与认证的用户关联的解析器主密钥。

[0351] 2. 产生唯一会话主密钥。

[0352] 3. 从解析器主密钥与会话主密钥的异或函数导出中间密钥。

[0353] 4. 使用现有或新的加密算法以中间密钥为密钥对数据进行可选加密。

[0354] 5. 根据中间密钥的模式将得到的可选加密的数据分离成四份或部分的解析数据。

[0355] 6. 在本发明的该方法的这个实施例中，会话主密钥将存储在数据储存器内的单独的密钥管理表中。为这个事务产生唯一事务 ID。将事务 ID 和会话主密钥存储在单独的密

钥管理表中或者将会话主密钥和事务 ID 传回调用程序以用于外部管理。根据解析器主密钥的模式对事务 ID 进行分离并且将数据附于可选加密的解析数据或分离的数据。

[0356] 7. 得到的四个数据份将包含可选加密的原始数据的部分和事务 ID 的部分。

[0357] 8. 可选地，为这四个数据份的每个产生加密密钥。

[0358] 9. 可选地，对每份进行加密，然后将加密密钥存储在与加密的数据部分或份不同的位置。例如，份 1 获得密钥 4，份 2 获得密钥 1，份 3 获得密钥 2，份 4 获得密钥 3。

[0359] 为了恢复原始数据格式，需要颠倒这些步骤。

[0360] 本领域普通技术人员易于理解，根据需要，本文所述的方法的某些步骤可以以不同顺序执行或者重复多次。本领域技术人员还易于理解，可以以彼此不同方式对这些数据部分进行处理。例如，可以仅仅对解析数据的一个部分执行多个分离或解析步骤。可以以任何希望的方式对解析数据的每个部分进行独特的保护，只要数据可以被重装、重构、重形成、解密或恢复到它的原始或其它可用形式即可。

[0361] 本领域技术人员易于理解，多种多样的加密方法适用于本发明的方法。一次一密乱码本（One Time Pad）算法常常被认为是最安全加密方法之一，并且适用于本发明的方法。使用一次一密乱码本算法要求产生与要保护的数据一样长的密钥。在诸如由于要保护的数据集的大小而导致产生和管理非常长的密钥的情况的某些境况下，不太希望使用这种方法。在一次一密乱码本（OTP）算法中，使用简单的异或函数 XOR。对于相同长度的两个二进制流 x 和 y，x XOR y 是指 x 和 y 的逐比特异或。

[0362] 在比特级，产生：

[0363] 0XOR 0=0

[0364] 0XOR 1=1

[0365] 1XOR 0=1

[0366] 1XOR 1=0

[0367] 在本文中针对要分裂的 n 字节秘密 s（或数据集），描述这个过程的例子。该过程将产生 n 字节随机值 a，然后设置：

[0368] b=a XOR s。

[0369] 要注意，可以通过下式导出“s”：

[0370] s=a XOR b。

[0371] 值 a 和 b 称作份或部分并且安置在分立的储存器内。一旦秘密 s 分被裂成两个或更多份，以安全方式将它丢弃。

[0372] 本发明的安全数据解析器可以利用这个函数，执行结合多个不同秘密密钥值 K1、K2、K3、Kn、K5 的多个 XOR 函数。在运算开始时，要保护的数据被传递给第一加密运算，安全数据 = 数据 XOR 秘密密钥 5：

[0373] S=D XOR K5

[0374] 为了将得到的加密数据安全地存储在例如四个份 S1、S2、S3、Sn 中，根据 K5 的值将数据解析并分裂成“n”个段或份。这个运算产生原始加密的数据的“n”个伪随机份。然后，可以用剩余的秘密密钥值对每个份执行接下来的 XOR 函数，例如：安全数据段 1= 加密的数据份 1XOR 秘密密钥 1：

[0375] SD1=S1XOR K1

[0376] SD2=S2XOR K2

[0377] SD3=S3XOR K3

[0378] SDn=Sn XOR Kn

[0379] 在一个实施例中，可能不希望任何一个储存器包含对其所持有的信息进行解密的足够信息，从而对份进行解密所需的密钥被存储在不同的数据储存器中：

[0380] 储存器 1：SD1，Kn

[0381] 储存器 2：SD2，K1

[0382] 储存器 3：SD3，K2

[0383] 储存器 n：SDn，K3

[0384] 此外，获取原始会话加密密钥 K5 所需的信息可以附于每个份。因此，在本文所述的密钥管理例子中，通过根据依赖于安装的解析器主密钥（TID1、TID2、TID3、TIDn）的内容而分裂成“n”份的事务 ID 来参考原始会话主密钥：

[0385] 储存器 1：SD1，Kn，TID1

[0386] 储存器 2：SD2，K1，TID2

[0387] 储存器 3：SD3，K2，TID3

[0388] 储存器 n：SDn，K3，TIDn

[0389] 在本文所述的结合会话密钥例子中，根据依赖于安装的解析器主密钥（SK1、SK2、SK3、SKn）的内容，会话主密钥分裂成“n”份：

[0390] 储存器 1：SD1，Kn，SK1

[0391] 储存器 2：SD2，K1，SK2

[0392] 储存器 3：SD3，K2，SK3

[0393] 储存器 n：SDn，K3，SKn

[0394] 根据这个例子，除非获得了所有四份，否则不能够对数据进行重装。即使捕获了所有四份，在无法访问会话主密钥和解析器主密钥的情况下也不可能重装或恢复原始信息。

[0395] 这个例子描述了本发明的该方法的一个实施例，并且在另一个实施例中还描述了用于将份安置到储存器中从而能够对所有储存器中的份进行组合以形成秘密认证材料的算法。所需的计算非常简单和快速。然而，对于一次一密乱码本（OTP）算法，由于密钥大小与要存储的数据的大小相同，所以可能会出现例如要保护大的数据集的、不太希望使用这种算法的境况。因此，将需要存储并发送原始数据量的大约两倍，这在某些境况下是不希望的。

[0396] 流密码 RS1

[0397] 流密码 RS1 分裂技术与本文所述的 OTP 分裂技术非常类似。替代 n 字节随机值，产生 n’=min(n,16) 字节随机值并且将其用作 RS1 流密码算法的密钥。RS1 流密码算法的优点在于，从小得多的种子数产生伪随机密钥。RS1 流密码加密的执行速度还被认为是本领域公知的三重 DES 加密的速度的近 10 倍，而不会危害安全性。RS1 流密码算法在本领域是公知的，并且可用于产生用于 XOR 函数中的密钥。RS1 流密码算法可与其它可买到的流密码算法（例如，RSA Security Inc 的 RC4TM 流密码算法）互操作，并且适用于本发明的方法。

[0398] 使用以上的密钥符号，K1 到 K5 现在是 n’ 字节随机值并且我们设置：

[0399] SD1=S1XOR E(K1)

[0400] SD2=S2XOR E(K2)

[0401] SD3=S3XOR E(K3)

[0402] SDn=Sn XOR E(Kn)

[0403] 其中,E(K1) 到 E(Kn) 是以 K1 到 Kn 为密钥的 RS1 流密码算法的输出的前 n' 个字节。如本文所述,将这些份现在安置到数据储存器内。

[0404] 在这个流密码 RS1 算法中,所需的必要计算几乎与 OTP 算法一样简单和快速。使用 RS1 流密码的这个例子的好处在于,对于每份,系统平均仅需要存储并发送比要保护的原始数据的大小多大约 16 字节。当原始数据的大小大于 16 字节时,由于 RS1 算法简单更短,所以 RS1 算法要比 OTP 算法更高效。本领域普通技术人员易于理解,多种多样的加密方法或算法适用于本发明,包括但不限于 RS1、OTP、RC4™、三重 DES 和 AES。

[0405] 与传统的加密方法相比,本发明的数据安全性方法和计算机系统提供了显著优点。一个优点是从将数据份移至可能位于不同的逻辑、物理或地理位置的一个或多个数据储存器或存储装置上的不同位置获得的安全性。例如,当对数据份进行物理分裂并且在不同人员的控制下时,危害数据的可能性大幅减小。

[0406] 由本发明的方法和系统提供的另一个优点是用于对数据进行保护以提供保持敏感数据的安全性的综合过程的本发明的方法的步骤的组合。该数据利用安全密钥进行加密并且根据该安全密钥分裂成一份或多份(在一个实施例中,4 份)。安全密钥利用参考指针被安全地存储,该参考指针根据安全密钥按四份保护。然后单独地对各数据份进行加密,并且将密钥与不同的加密的份安全地存储在一起。当进行组合时,根据本文公开的方法对数据进行保护的整个过程变成数据安全性的综合包。

[0407] 根据本发明的方法进行保护的数据易于取回和恢复、重构、重装、解密或以其它方式返回到它的原始或其它适合形式以供使用。为了恢复原始数据,可利用下面的项 :

[0408] 1. 数据集的所有份或部分。

[0409] 2. 用于保护数据的方法的处理流程的知识,以及对该处理流程进行再现的能力。

[0410] 3. 对会话主密钥的访问权。

[0411] 4. 对解析器主密钥的访问权。

[0412] 因此,希望设计一种安全安装,其中,以上元素中的至少一个可以与系统的其余组件物理分离(例如,在不同系统管理员的控制下)。

[0413] 通过使用解析器主密钥可以加强针对无良应用调用数据保护方法应用的保护。在本发明的这个实施例中,在采取任何动作之前,可以要求在安全数据解析器与应用之间进行互相认证握手。

[0414] 系统的安全性要求不存在用于重建原始数据的“后门”方法。针对可能出现数据恢复问题的安装,可以对安全数据解析器进行增强以提供四个份和会话主密钥储存器的镜像。例如 RAID (用于在几个盘上分散信息的廉价盘的冗余阵列)的硬件选项和例如复制的软件选项也能够帮助进行数据恢复计划。

[0415] 密钥管理

[0416] 在本发明的一个实施例中,数据保护方法使用三个密钥组用于进行加密操作。基于安装,每个密钥组可具有各自的密钥存储、取回、安全性和恢复选项。可使用的密钥包括但不限于 :

[0417] 解析器主密钥

[0418] 这个密钥是与安全数据解析器的安装关联的个体密钥。它安装在已经部署了安全数据解析器的服务器上。有多种适于保护这个密钥的选项，包括但不限于智能卡、单独的硬件密钥库、标准密钥库、定制密钥库或者例如位于受保护的数据库表内。

[0419] 会话主密钥

[0420] 每次当对数据进行保护时可以产生会话主密钥。会话主密钥用于在解析和分裂操作之前对数据进行加密。它还可以并入作为对加密的数据进行解析的手段（如果会话主密钥没有集成到解析数据内）。可以通过各种方式（包括但不限于标准密钥库、定制密钥库、独立数据库表）或者例如在加密的份内对会话主密钥进行保护。

[0421] 份加密密钥

[0422] 针对建立的数据集的每个份或部分，可产生单独的份加密密钥以进一步对份进行加密。份加密密钥可以存储在与已加密的份不同的份内。

[0423] 本领域普通技术人员易于理解，本发明的数据保护方法和计算机系统可广泛地应用于在任何设置或环境下的任何类型的数据。除了在互联网上或者在顾客与卖方之间执行的商业应用外，本发明的数据保护方法和计算机系统可高度应用于非商业或私有设置或环境。可以使用本文所述的方法和系统对希望针对任何未授权用户保持安全的任何数据集进行保护。例如，有利的是，通过利用本发明的方法和系统对数据进行保护，对公司或组织内的特定数据库的访问可以仅仅限于选择的用户。另一个例子是文档的产生、修改或访问，其中，希望限制访问或者防止未授权或意外访问或者在选择的个人、计算机或工作站之外的公开。本发明的数据保护的方法和系统可应用于任何非商业或商业环境或设置以进行任何设置的方式的这些和其它例子包括但不限于任何组织、政府机构或公司。

[0424] 在本发明的另一个实施例中，数据保护方法使用三个密钥组进行加密操作。基于安装，每个密钥组可以具有单独的密钥库、取回、安全性和恢复选项。可使用的密钥包括但不限于：

[0425] 1. 解析器主密钥

[0426] 这个密钥是与安全数据解析器的安装关联的独立密钥。它安装在已经部署了安全数据解析器的服务器上。存在各种适于保护这个密钥的选项，包括但不限于智能卡、单独硬件密钥库、标准密钥库、定制密钥库或者例如位于受保护的数据库表内。

[0427] 2. 会话主密钥

[0428] 每次当对数据进行保护时可以产生会话主密钥。会话主密钥与解析器主密钥结合使用以导出中间密钥。可以以多种方式（包括但不限于标准密钥库、定制密钥库、独立数据库表）或者例如在加密的份内对会话主密钥进行保护。

[0429] 3. 中间密钥

[0430] 每次当对数据进行保护时可以产生中间密钥。中间密钥用于在解析和分裂操作之前对数据进行加密。它还可以并入作为对加密的数据进行解析的手段。

[0431] 4. 份加密密钥

[0432] 针对建立的数据集的每个份或部分，可产生单独的份加密密钥以进一步对份进行加密。份加密密钥可以存储在与进行加密的份不同的份中。

[0433] 本领域普通技术人员易于理解，本发明的数据保护方法和计算机系统可广泛地应

用于在任何设置或环境下的任何类型的数据。除了在互联网上或者在顾客与卖方之间执行的商业应用外，本发明的数据保护方法和计算机系统可高度应用于非商业或私有设置或环境。可以使用本文所述的方法和系统对希望针对未授权用户保持安全的任何数据集进行保护。例如，有利的是，通过利用本发明的方法和系统对数据进行保护，对公司或组织内的特定数据库的访问可以仅仅限于选择的用户。另一个例子是文档的产生、修改或访问，其中，希望限制访问或者防止未授权或意外访问或者在选择的个人、计算机或工作站之外的公开。本发明的数据保护的方法和系统可应用于任何非商业或商业环境或设置以进行任何设置的方式的这些合其它例子包括但不限于任何组织、政府机构或公司。

[0434] 工作组、项目、个人 PC/ 膝上型电脑或跨平台数据安全性

[0435] 本发明的数据保护方法和计算机系统还用于由工作组、项目、个人 PC/ 膝上型电脑和例如用于企业、办公室、政府机构或建立、处理或存储敏感数据的任何设置的任何其它平台，对数据进行保护。本发明提供已知由诸如美国政府的组织所寻求的用于在整个政府组织上或者在州或联邦级政府之间实施的对数据进行保护的方法和计算机系统。

[0436] 本发明的数据保护方法和计算机系统提供不仅对普通文件进行解析和分裂还对任何类型的数据字段、集合和 / 或表进行解析和分裂的能力。此外，所有形式的数据（包括但不限于文本、视频、图像、生物测定和语音数据）能够在这个过程之下得到保护。本发明的保护数据的方法的可调整性、速度和数据吞吐量仅限于用户可以支配的硬件。

[0437] 在本发明的一个实施例中，如下所述在工作组环境中利用数据保护方法。在一个实施例中，如图 23 所示和如下文所述，本发明的工作组级数据保护方法使用信任引擎的私钥管理功能来存储一组用户共享安全数据所需的关联私钥（解析器组主密钥）和用户 / 组关系。本发明的方法具有根据如何部署解析器主密钥为企业、工作组或个体用户保护数据的能力。

[0438] 在一个实施例中，可以提供附加密钥管理和用户 / 组管理程序，从而用单点的支配和密钥管理实现宽范围工作组执行方式。密钥产生、管理和撤销由单个维护程序处理，随着用户数目的增加，这些都变得尤其重要。在另一个实施例中，还可以跨一个或几个不同的系统管理员设置密钥管理，这不允许任何一个人或组根据需要控制数据。这允许通过由组织定义的角色、责任、成员资格、权利等获得被保护数据的管理，并且对被保护数据的访问能够仅限于被许可或要求仅仅访问它们工作的部分的那些人，而诸如经理或执行官的其它人可以访问所有被保护数据。这个实施例使得能够在公司或组织内的不同组之间共享被保护数据，同时仅仅允许某些选择的个人（例如，具有授权和预定的角色和责任的那些人）观察整个数据。此外，本发明的方法和系统的这个实施例还允许例如在不同公司、或公司的不同部门或分支机构、或任何政府或组织等的任何不同组织部门、团体、机构、办公室等（其中要求一些共享但并非任一方被许可访问所有数据）之间共享数据。针对本发明的这种方法和系统的需要和利用的特别明显的例子是允许例如在政府区域、机构和办公室之间以及在大公司或任何其它组织的不同分支机构、部门或办公室之间进行共享但保持安全性。

[0439] 如下是本发明的方法在较小范围进行应用的例子。解析器主密钥用作安全数据解析器的对于组织的串行化或打标记。当解析器主密钥的使用范围从整个企业缩小至较小工作组时，本文所述的数据保护方法用于在用户组内共享文件。

[0440] 在图 25 所示和下文所述的例子中，定义了六个用户以及他们在组织内的头衔或

角色。边条表示用户根据他们的角色而所属于的五个可能的组。箭头表示用户在一个或多个组内的成员资格。

[0441] 当构造用于这个例子的安全数据解析器时，系统管理员通过维护程序从操作系统访问用户和组信息。这个维护程序基于用户在组中的成员资格产生并向用户分配解析器组主密钥。

[0442] 在这个例子中，在高级职员组中有三个成员。对于这个组，动作如下：

[0443] 1. 访问高级职员组的解析器组主密钥(在不可获得的情况下，产生密钥)；

[0444] 2. 产生将 CEO 与高级职员组进行关联的数字证书；

[0445] 3. 产生将 CFO 与高级职员组进行关联的数字证书；

[0446] 4. 产生将主管市场的副总裁与高级职员组进行关联的数字证书。

[0447] 针对每个组以及每个组内的每个成员将执行相同的一组动作。当维护程序完成时，解析器组主密钥变成组的每个成员的共享证明。当通过维护程序从组中去除用户时可以自动完成分配的数字证书的撤销而不会影响该组的剩余成员。

[0448] 一旦定义了共享证明，解析和分裂过程仍相同。当要对文件、文档或数据元素进行保护时，向用户提示当保护数据时要使用的目标组。得到的被保护数据仅仅可由目标组的其它成员访问。本发明的方法和系统的这个功能可以与任何其它计算机系统或软件平台一起使用，或者例如可以集成到现有的应用程序或者为文件安全性而独立使用。

[0449] 本领域普通技术人员易于理解，加密算法的任何一个或者组合适用于本发明的方法和系统。例如，在一个实施例中，加密步骤可以被重复以生成多层加密方案。此外，不同的加密算法或者加密算法的组合可用于重复加密步骤从而使得不同的加密算法被应用于多层加密方案的不同层。这样，加密方案自身可以变成用于保护敏感数据免遭未授权使用或访问的本发明的方法的组成部分。

[0450] 安全数据解析器可包括作为内部部件、外部部件或者这两者的错误检查部件。例如，在一个适当的方法中，当使用根据本发明的安全数据解析器建立数据的部分时，为了确保一个部分内数据的完整性，在这个部分内以预设间隔获取哈希值并且将它附于所述间隔的末端。该哈希值是数据的可预测且可再现的数值表示。如果数据内的任何比特变化，则哈希值将不同。扫描模块(作为安全数据解析器外部的独立部件或者作为内部部件)然后可以对由安全数据解析器产生的数据的部分进行扫描。将每个数据部分(或者根据某间隔或根据随机或伪随机采样而少于所有的数据部分)与所附的一个或多个哈希值进行比较并且可采取动作。这个动作可以包括：匹配和不匹配的值的报告、对不匹配的值的警告、或者调用某外部或内部程序以触发数据的恢复。例如，可以通过调用恢复模块基于根据本发明的不需要所有部分就可以产生原始数据的概念来执行数据的恢复。

[0451] 可以使用附于所有数据部分或这些数据部分的子集中的任何地方的任何合适的完整性信息，执行任何其它合适的完整性检查。完整性信息可以包括能够用于确定数据部分的完整性的任何合适信息。完整性信息的例子可以包括：基于任何合适参数(例如基于各个数据部分)计算的哈希值、数字签名信息、消息认证码(MAC)信息、任何其它合适信息、或者它们的任何组合。

[0452] 本发明的安全数据解析器可用于任何合适应用中。即，本文所述的安全数据解析器在不同领域的计算和技术中具有不同的应用。在下文中讨论几个这种领域。应该明白，本

质上这些仅仅是例示并且任何其它合适应用可以利用安全数据解析器。还应该明白，所述的例子仅仅是例示性实施例，可以以任何合适方式对其进行修改以满足任何合适期望。例如，解析和分裂可以基于任何合适单元(例如，比特、字节、千字节、兆字节、它们的任何组合或者任何其它合适单元)。

[0453] 本发明的安全数据解析器可用于实现安全物理令牌，由此，为了访问存储在另一个存储区内的附加数据，可能需要存储在物理令牌内的数据。在一个合适方案中，物理令牌(例如，紧凑 USB 闪存驱动器、软盘、光盘、智能卡、或者任何其它合适的物理令牌)可用于存储根据本发明的解析数据的至少两个部分中的一个。为了访问原始数据，需要对 USB 闪存驱动器进行访问。因此，保持解析数据的一个部分的个人计算机在能够访问原始数据之前将需要附接具有解析数据的另一部分的 USB 闪存驱动器。图 26 示出了这个应用。存储区 2500 包括解析数据的一部分 2502。为了访问原始数据，需要使用任何合适通信接口 2508(例如，USB、串口、并口、蓝牙、IR、IEEE 1394、以太网或者任何其它合适的通信接口)把具有解析数据的一部分 2506 的物理令牌 2504 连接到存储区 2500。这在例如计算机上的敏感数据被独自留下并且遭受未授权访问尝试的情形下是有用的。通过去除物理令牌(例如，USB 闪存驱动器)，无法对敏感数据进行访问。应该明白，可以使用用于利用物理令牌的任何其它合适方法。

[0454] 本发明的安全数据解析器可用于实现安全认证系统，由此，使用安全数据解析器对用户登记数据(例如，口令、私有加密密钥、指纹模板、生物测定数据或任何其它合适的用户登记数据)进行解析和分裂。可以对用户登记数据进行解析和分裂，由此，一个或多个部分存储在智能卡、政府公共访问卡、任何合适的物理存储装置(例如，磁盘或光盘、USB 密钥驱动器等)上或者任何其它合适装置上。解析的用户登记数据的一个或多个其它部分可以存储在执行认证的系统内。这对认证过程提供了附加等级的安全性(例如，除了从生物测定源获得的生物测定认证信息以外，还必须经由适当的解析和分裂的数据部分获得用户登记数据)。

[0455] 本发明的安全数据解析器可以集成到任何合适的现有系统中，从而可以在每个系统的各自环境内提供对它的功能的使用。图 27 示出了例示性系统 2600 的框图，系统 2600 可以包括用于实现任何合适应用的软件、硬件或二者。系统 2600 可以是安全数据解析器 2602 可以被作为集成部件而进行翻新的现有系统。或者，例如可以从任何合适系统 2600 的最早设计阶段，将安全数据解析器 2602 集成到该系统 2600 内。安全数据解析器 2602 可以集成在系统 2600 的任何合适等级中。例如，安全数据解析器 2602 可以在充分后端等级集成到系统 2600 中，从而使得安全数据解析器 2602 的存在对于系统 2600 的端用户可以是实质上透明的。根据本发明，安全数据解析器 2602 可用于在一个或多个存储装置 2604 之间对数据进行解析和分裂。在下文中讨论在内部集成了安全数据解析器的系统的一些例示性例子。

[0456] 本发明的安全数据解析器可以集成到操作系统内核(例如，Linux、Unix、或者任何其它合适的商业或专用操作系统)。该集成可用于在装置等级保护数据，由此，例如，通常将存储在一个或多个装置内的数据被集成到操作系统内的安全数据解析器分离成一定数目的部分并且存储在所述一个或多个装置之间。当尝试访问原始数据时，同样集成到操作系统内的适当软件可以按照对于端用户透明的方式将解析的数据部分重组成原始数据。

[0457] 本发明的安全数据解析器可以集成到存储系统的卷管理器或者任何其它合适部件，以跨任何或所有支持的平台保护本地和联网的数据存储。例如，通过集成安全数据解析器，存储系统可利用由安全数据解析器提供的冗余(即，用于实现不需要数据的所有的分离部分就可以重构原始数据的特征)以防止数据损失。不管是否使用冗余，安全数据解析器还使得写入存储装置的所有数据可以成为根据本发明的解析而产生的多个部分的形式。当尝试访问原始数据时，同样集成到卷管理器或存储系统内的其它合适部件的适当软件可以按照对于端用户透明的方式将解析的数据部分重组成原始数据。

[0458] 在一个合适方案中，本发明的安全数据解析器可以集成到 RAID 控制器(作为硬件或者作为软件)。这使得可以将数据安全存储到多个驱动器上，同时在驱动器故障的情况下保持容错性。

[0459] 例如为了保护敏感表信息，本发明的安全数据解析器可以集成到数据库。例如，在一个合适方案中，可以根据本发明对与数据库表的特定单元(例如，个体单元、一个或多个特定列、一个或多个特定行、它们的任何组合、或者整个数据库表)关联的数据进行解析和分离(例如，其中，不同的部分存储在位于一个或多个位置处的一个或多个存储装置上或者单个存储装置上)。可以通过传统的认证方法(例如，用户名和口令)来授权为了观看原始数据而重组这些部分的访问。

[0460] 本发明的安全解析器可以集成到包括移动中数据(data in motion)(即，数据从一个位置到另一个位置的转移)的任何合适系统内。这些系统例如包括电子邮件、流式数据广播和无线(例如，WiFi)通信。关于电子邮件，在一个合适方案中，安全解析器可用于对外发消息(即，包含文本、二进制数据或二者(例如，附于电子邮件消息的文件))进行解析并且沿着不同路径发送解析数据的不同部分，由此建立多个数据流。如果这些数据流中的任何一个受到危害，则原始消息仍然安全，因为根据本发明，为了产生原始数据，该系统要求组合超过一个的部分。在另一个合适方案中，数据的不同部分可以沿着一个路径顺序地传送从而使得如果获得了一个部分，则这不足以产生原始数据。根据本发明，这些不同部分到达期望收件人的位置并且可以被组合以产生原始数据。

[0461] 图 28 和图 29 是这种电子邮件系统的例示性框图。图 28 示出了发件人系统 2700，它可以包括任何合适硬件，例如，计算机终端、个人计算机、手持装置(例如，PDA、黑莓)、蜂窝电话、计算机网络、任何其它合适硬件、或者它们的任何组合。发件人系统 2700 用于产生和 / 或存储消息 2704，消息 2704 例如可以是电子邮件消息、二进制数据文件(例如，图形、语音、视频等)或二者。根据本发明由安全数据解析器 2702 对消息 2704 进行解析和分裂。得到的数据部分可以经由一个或多个分立的通信路径 2706 通过网络 2708 (例如，互联网、内联网、LAN、WiFi、蓝牙、任何其它合适的有线或无线通信手段、或者它们的任何组合)传送至收件人系统 2710。可以在时间上并行地或者另选地根据不同数据部分的通信之间的任何合适的时间延迟，传送这些数据部分。收件人系统 2710 可以是关于发件人系统 2700 如上所述的任何合适硬件。根据本发明，沿着通信路径 2706 运送的单独的数据部分在收件人系统 2710 处被重组以产生原始消息或数据。

[0462] 图 29 示出了发件人系统 2800，它可以包括任何合适硬件，例如，计算机终端、个人计算机、手持装置(例如，PDA)、蜂窝电话、计算机网络、任何其它合适硬件、或者它们的任何组合。发件人系统 2800 用于产生和 / 或存储消息 2804，消息 2804 例如可以是电子邮件消

息、二进制数据文件(例如,图形、语音、视频等等)或二者。根据本发明由安全数据解析器2802对消息2804进行解析和分裂。得到的数据部分可以经由单个通信路径2806通过网络2808(例如,互联网、内联网、LAN、WiFi、蓝牙、任何其它合适的有线或无线通信手段、或者它们的任何组合)传送至收件人系统2810。这些数据部分可以经由通信路径2806相对于彼此串行传送。收件人系统2810可以是在上文关于发件人系统2800描述的任何合适硬件。根据本发明,沿着通信路径2806运送的各个数据部分在收件人系统2810处被重组以产生原始消息或数据。

[0463] 应该明白,图28和图29的布置仅仅是例示性的。可以使用任何其它合适布置。例如,在另一个合适方案中,图28和29的系统的特征可以进行组合,从而使用图28的多路径方案,并且其中,与在图29中通信路径2806一样,一个或多个通信路径2706用于运送超过一个的数据部分。

[0464] 安全数据解析器可以集成在移动中数据系统的任何合适等级。例如,在电子邮件系统的情况下,安全解析器可以集成在用户接口等级(例如,集成到Microsoft® Outlook中),在这种情况下,当使用电子邮件时,用户可以对安全数据解析器特征的使用进行控制。或者,可以在后端部件(例如在交换服务器)中实现安全解析器,在这种情况下,根据本发明,不需要任何用户干涉,可以自动对消息进行解析、分裂并且沿着不同路径进行传送。

[0465] 类似地,在数据(例如,音频、视频)的流式广播的情况下,输出数据可以被解析和分离成多个流,每个流包含解析数据的一部分。根据本发明,这多个流可以沿着一个或多个路径发送并且在收件人的位置进行重组。这种方案的好处之一是,避免了与对数据进行传统加密然后通过单个通信通道发送加密的数据相关联的相对较大的开销。本发明的安全解析器允许在多个并行流中发送移动中数据,从而提高了速度和效率。

[0466] 应该明白,为了通过任何传输介质(例如包括有线、无线或物理)的任何类型的移动中数据的保护和容错,可以集成安全数据解析器。例如,IP语音(VoIP)应用可以利用本发明的安全数据解析器。可以使用本发明的安全数据解析器来保护往来于任何合适个人数字助理(PDA)装置(例如,黑莓和智能电话)的无线或有线数据传输。对等和基于集线器的无线网络的使用无线802.11协议的通信、卫星通信、点对点无线通信、互联网客户机/服务器通信或者任何其它合适通信可以包括根据本发明的安全数据解析器的移动中数据能力。计算机外设装置(例如,打印机、扫描仪、监视器、键盘、网络路由器、生物测定认证装置(例如,指纹扫描仪)、或者任何其它合适外设装置)之间、计算机与计算机外设装置之间、计算机外设装置与任何其它合适装置之间的数据通信或者它们的任何组合可以利用本发明的移动中数据特征。

[0467] 本发明的移动中数据特征还可以应用于例如使用独立路线、媒介物、方法的安全份的物理传输、任何其它合适的物理传输或者它们的任何组合。例如,数据的物理传输可以发生于数字/磁带、软盘、光盘、物理令牌、USB驱动器、可移动硬盘、具有闪存的消费电子装置(例如苹果IPOD或其它MP3播放器)、闪存、用于传输数据的任何其它合适介质、或者它们的任何组合。

[0468] 本发明的安全数据解析器向安全性提供了灾难恢复的能力。根据本发明,为了获取原始数据,不需要由安全数据解析器产生的所有部分的分离数据。也就是说,在存储的m个部分之中,n可以是这m个部分之中获取原始数据所需的最小数,其中,n<=m。例如,如果

四个部分的每个存储在与其它三个部分不同的物理位置中，那么如果在这个例子中 n=2，则这些位置中的两个可被危害从而数据被破坏或者不可访问，并且从其它两个位置中的部分仍可以获取原始数据。对于 n 或 m，可以使用任何合适值。

[0469] 此外，本发明的 m 取 n 特征可用于建立“双人规则”，由此为了避免委托单个个人或者任何其它实体对可能是敏感数据的事务进行全面访问，两个或更多不同实体（每个实体具有由本发明的安全解析器解析的分离数据的一部分）需要同意将它们的各部分放到一起以获取原始数据。

[0470] 本发明的安全数据解析器可用于向一组实体提供组范围密钥，该组范围密钥使组成员可以访问由该特定组授权访问的特定信息。组密钥可以是例如为了获取寻找的信息而需要与在中心存储的另一个部分进行组合的、由根据本发明的安全解析器产生的数据部分之一。例如，这个特征允许在组内实现安全协作。例如，它可以应用于专有网络、虚拟专用网、内联网、或者任何其它合适网络。

[0471] 安全解析器的这种用途的特定应用例如包括联合信息共享，在联合信息共享中，例如，对多国友好政府力量给予基于对每个对应国家授权的安全等级通过单个网络或双重网络（即，与涉及当前使用的相对较多人工处理的许多网络相比）传送在操作和其它方面敏感的数据的能力。这种能力还可应用于公司或其它组织，其中，需要由一个或多个特定个人（组织内或组织外）知道的信息可以经由单个网络传送，而不需要担心未授权个人观看该信息。

[0472] 另一个特定应用包括用于政府系统的多级安全性层级。也就是说，本发明的安全解析器可以提供使用单个网络以不同等级的机密信息（例如，不机密、机密、秘密、绝密）操作政府系统的能力。如果需要，可使用更多网络（例如，对于绝密使用单独网络），但是本发明允许比对每个等级的机密使用独立网络的当前布置实质更少的布置。

[0473] 应该明白，可以使用本发明的安全解析器的上述的应用的任何组合。例如，组密钥应用能够与移动中数据安全性应用一起使用（即，由此，通过网络传送的数据仅能够由对应组的成员访问，并且在这种情况下，当数据在移动时，根据本发明它在多个路径中进行分裂（或者以顺序得部分进行发送））。

[0474] 本发明的安全数据解析器可以集成到任何中间件应用中，从而使得应用能够将数据安全存储到不同数据库产品或不同装置，而不需要对应用或数据库进行改动。中间件是使得两个独立和已经存在的程序可以进行通信的任何产品的一般术语。例如，在一个合适方案中，集成了安全数据解析器的中间件可用于使得针对特定数据库编写的程序与其它数据库进行通信，而不用定制编码。

[0475] 本发明的安全数据解析器可以实现为具有例如本文所述的任何合适能力的任何组合。在本发明的一些实施例中，例如，安全数据解析器可以实现为仅仅具有某些能力，但是可以通过使用直接或间接与安全数据解析器对接的外部软件、硬件或二者而获得其它能力。

[0476] 图 30 例如示出了作为安全数据解析器 3000 的安全数据解析器的例示性实施方式。安全数据解析器 3000 可以实现为具有非常少的内置能力。如所示，根据本发明，安全数据解析器 3000 可以包括使用模块 3002 将数据解析和分裂成多个部分（这里也称作份）的内置能力。安全数据解析器 3000 还可以包括使用模块 3004 执行冗余从而能够实现例如上

述得 m 取 n 特征(即,不需要使用解析和分裂的所有的数据份就可以重建原始数据)的内置能力。根据本发明,安全数据解析器 3000 还可以包括使用模块 3006 将数据份安置到缓冲器内(这些数据份从这些缓冲器发送以传送至远程位置、进行存储等)的份分布能力。应该明白,任何其它合适能力可以内置到安全数据解析器 3000 中。

[0477] 组装数据缓冲器 3008 可以是用于存储将由安全数据解析器 3000 进行解析和分裂的原始数据(尽管不一定是它的原始形式)的任何合适存储器。在分裂操作中,组装数据缓冲器 3008 向安全数据解析器 3008 提供输入。在恢复操作中,组装数据缓冲器 3008 可用于存储安全数据解析器 3000 的输出。

[0478] 分裂份缓冲器 3010 可以是可用于存储从原始数据的解析和分裂获得的多个数据份的一个或多个存储器模块。在分裂操作中,分裂份缓冲器 3010 保持安全数据解析器的输出。在恢复操作中,分裂份缓冲器保持对安全数据解析器 3000 的输入。

[0479] 应该明白,对于安全数据分裂器 3000,可以内置能力的任何其它合适布置。可以内置任何附加特征,并且可以去除所例示的任何特征,使得所例示的任何特征更加健壮、较不健壮,或者以任何合适方式对所例示的任何特征进行修改。缓冲器 3008 和 3010 同样仅仅是例示性的并且可以以任何合适方法进行修改、去除或者添加。

[0480] 以软件、硬件或二者实现的任何合适模块可以由安全数据解析器 3000 调用或者对安全数据解析器 3000 进行调用。如果需要,即使是内置到安全数据解析器 3000 中的能力也可以由一个或多个外部模块进行替换。如所示,一些外部模块包括随机数产生器 3012、密码反馈密钥产生器 3014、哈希算法 3016、任何一个或多个类型的加密 3018 和密钥管理 3020。应该明白,这些仅仅是例示性外部模块。除了所示这些外部模块以外或者替代所示这些外部模块,还可以使用任何其它合适模块。

[0481] 密码反馈密钥产生器 3014 可以在安全数据解析器 3000 的外部,为每个安全数据解析器操作产生唯一密钥或者随机数(例如使用随机数产生器 3012),该唯一密钥或者随机数用作用于将原始会话密钥大小(例如,128、256、512 或 1024 比特的值)扩展至等于要解析和分裂的数据的长度的值的操作的种子值。任何合适算法可用于密码反馈密钥产生,例如包括 AES 密码反馈密钥产生算法。

[0482] 为了便于将安全数据解析器 3000 及其外部模块(即,安全数据解析器层 3026)集成到应用层 3024 (例如,电子邮件应用、数据库应用等)中,可以使用可利用例如 API 函数调用的包装层。可以使用便于将安全数据解析器层 3026 集成到应用层 3024 中的任何其它合适布置。

[0483] 图 31 例示性示出了当在应用层 3024 中发出写命令(例如,写入存储装置)、插入命令(例如,插入数据库字段中)或者发送命令(例如,经由网络)时可以如何使用图 30 的布置。在步骤 3100 中,识别要保护的数据并且调用安全数据解析器。该调用传递到包装器层 3022,在步骤 3102 中,包装器层 3022 将在步骤 3100 中识别的输入数据流式传输到组装数据缓冲器 3008。另外,在步骤 3102 中,任何合适的份信息、文件名、任何其它合适信息、或者它们的任何组合可以被存储(例如,作为包装器层 3022 中的信息 3106)。根据本发明,安全数据处理器 3000 然后对它从组装数据缓冲器 3008 作为输入获取的数据进行解析和分裂。它将数据份输出到分裂份缓冲器 3010。在步骤 3104 中,包装器层 3022 从存储的信息 3106 获得任何合适份信息(即,在步骤 3102 中由包装器层 3022 存储的份信息)和份位置(例如,

来自于一个或多个配置文件)。包装器层 3022 然后适当地对输出份(从分裂份缓冲器 3010 获得)执行写操作(例如,写入一个或多个存储装置,在网络上传送,等等)。

[0484] 图 32 例示性示出了当执行读(例如,从存储装置)、选择(例如,从数据库字段)、或者接收(例如,从网络)时可以如何使用图 30 的布置。在步骤 3200 中,识别要恢复的数据并且从应用层 3024 调用安全数据解析器 3000。在步骤 3202 中,从包装器层 3022 获得任何合适份信息并且确定份位置。包装器层 3022 将在步骤 3200 中识别的数据的部分加载到分裂份缓冲器 3010 中。然后,安全数据解析器 3000 根据本发明对这些份进行处理(例如,如果仅可获得四份中的三份,则可以使用安全数据解析器 3000 的冗余能力,从而仅使用这三份就能够恢复原始数据)。然后,恢复的数据存储在组装数据缓冲器 3008 中。在步骤 3204 中,应用层 3022 将存储在组装数据缓冲器 3008 中的数据转换成它的原始数据格式(如果需要)并且将原始格式的原始数据提供给应用层 3024。

[0485] 应该明白,图 31 所示的原始数据的解析和分裂以及图 32 所示的将数据部分恢复为原始数据仅仅是例示性的。除了所示的这些以外或者替代所示的这些,还可以使用任何其它合适的过程、部件或二者。

[0486] 图 33 是根据本发明的一个实施例的用于将原始数据解析和分裂成两个或更多数据部分的例示性处理流程的框图。如所示,期望进行解析和分裂的原始数据是明文 3306(即,单词“SUMMIT”用作例子)。应该明白,可以根据本发明对任何其它类型的数据进行解析和分裂。产生会话密钥 3300。如果会话密钥 3300 的长度与原始数据 3306 的长度不兼容,则可以产生密码反馈会话密钥 3304。

[0487] 在一个合适方案中,在进行解析、分裂或在二者之前,可以对原始数据 3306 进行加密。例如,如图 33 所示,可将原始数据 3306 与任何合适值(例如,与密码反馈会话密钥 3304 或者与任何其它合适值)进行异或。应该明白,除了所示的 XOR 技术以外或者替代所示的 XOR 技术,还可以使用任何其它合适的加密技术。还应该明白,尽管针对逐字节的操作示出图 33,但是该操作可以以比特级或任何其它合适等级进行。还应该明白,如果希望,不需要对原始数据 3306 进行任何加密。

[0488] 然后,对得到的加密的数据(或者原始数据(在没有进行加密的情况下))进行哈希处理以确定如何在输出桶(例如,在所示例子中是四个)之间将加密(或者原始)数据分裂。在所示例子中,基于字节进行哈希处理并且哈希处理是密码反馈会话密钥 3304 的函数。应该明白,这仅仅是例示性的。如果希望,可以在比特级执行哈希处理。哈希处理可以是除密码反馈会话密钥 3304 以外的任何其它合适值的函数。在另一个合适方案中,不需要使用哈希处理。而是可以采用用于对数据进行分裂的任何其它合适技术。

[0489] 图 34 是根据本发明的一个实施例的用于从原始数据 3306 的两个或更多的解析和分裂部分恢复原始数据 3306 的例示性处理流程的框图。该处理包括基于密码反馈会话密钥 3304 对这些部分进行反向哈希处理(即,与图 33 的过程反向)从而恢复加密的原始数据(或者原始数据,如果在解析和分裂之前没有进行加密的话)。然后可使用加密密钥来恢复原始数据(即,在所示的例子中,通过将密码反馈会话密钥 3304 与加密的数据进行 XOR 处理,使用密码反馈会话密钥 3304 对 XOR 加密进行解密)。这恢复了原始数据 3306。

[0490] 图 35 示出了在图 33 和图 34 的例子中如何实现比特分裂。哈希处理可用于确定分裂每个数据字节的比特值(例如,基于密码反馈会话密钥、基于任何其它合适值)。应该明

白，这仅仅是在比特级实施分裂的一个例示性方法。可使用其它合适技术。

[0491] 应该明白，可以针对任何合适哈希算法形成本文所述的哈希功能。这些算法例如包括 MD5 和 SHA-1。可以在不同时间由本发明的不同部件使用不同的哈希算法。

[0492] 在根据以上例示性过程或者通过任何其它过程或算法确定了分裂点后，确定向哪些数据部分附加左段和右段的每一个。任何合适算法可用于执行这个确定。例如，在一个合适方案中，可建立所有可能分布（例如，按照针对左段和针对右段的目的地的配对的形式）的表，由此，可以通过对可被产生并扩展至原始数据的大小的会话密钥、密码反馈会话密钥或任何其它合适的随机或伪随机值中的对应数据使用任何合适哈希函数，来确定针对左段和右段的每个的目的地份值。例如，可以产生随机或伪随机值中的对应字节的哈希函数。该哈希函数的输出用于确定从所有目的地组合的表中选择哪个目的地的配对（即，针对左段的一个目的地和针对右段的一个目的地）。基于这个结果，分裂数据单元的各段被附加到由作为哈希函数的结果而选择的表值所指示的对应的两份。

[0493] 根据本发明可以将冗余信息附加到数据部分从而不需要使用所有的数据部分就能够恢复原始数据。例如，如果希望四个部分中的两个部分就足以恢复数据，则份中的附加数据可以例如以轮循（round-robin）方式相应地附加到每份（例如，当原始数据的大小是 4MB 时，份 1 获得它自身的份以及份 2 和份 3 的份；份 2 获得它自身的份以及份 3 和份 4 的份；份 3 获得它自身的份以及份 4 和份 1 的份；份 4 获得它自身的份以及份 1 和份 2 的份）。根据本发明，可以使用任何这种合适的冗余。

[0494] 应该明白，根据本发明，任何其它合适的解析和分裂方案可用于从原始数据集产生多个数据部分。例如，可以逐比特地随机或伪随机处理解析和分裂。可以使用随机或伪随机值（例如，会话密钥、密码反馈会话密钥等），由此，针对原始数据中的每个比特，关于随机或伪随机值中的对应数据的哈希函数的结果可以指示向哪个份附加对应比特。在一个合适方案中，随机或伪随机值可以被产生为或者扩展至原始数据的大小的 8 倍，从而可以关于原始数据的每个比特，对随机或伪随机值的对应字节执行哈希函数。根据本发明，可以使用逐比特级解析并分裂数据的任何其它合适算法。还应该明白，根据本发明，冗余数据例如可以以上文刚描述的方式附加到数据份。

[0495] 在一个合适方案中，解析和分裂不需要是随机或伪随机的。而且，可以使用用于解析和分裂数据的任何合适确定性算法。例如，可以采用将原始数据分解成多个顺序份，作为一种解析和分裂算法。另一个例子是逐比特地解析并分裂原始数据，按照轮循方式顺序地向数据份附加每个对应比特。还应该明白，根据本发明，冗余数据可以例如以上述方式附加到数据份。

[0496] 在本发明的一个实施例中，在安全数据解析器产生原始数据的多个部分后，为了恢复原始数据，某一个或更多的产生的部分可以是强制性的。例如，如果这些部分之一用作认证份（例如，保存在物理令牌装置上）并且如果使用安全数据解析器的容错特征（即，恢复原始数据不需要所有的数据部分），则即使安全数据解析器可访问足够数目的原始数据的部分来恢复原始数据，在它恢复原始数据之前仍需要存储在物理令牌装置上的认证份。应该明白，例如基于应用、数据类型、用户、任何其它合适因素、或者它们的任何组合，可要求任何数目和类型的特定份。

[0497] 在一个合适方案中，安全数据解析器或者安全数据解析器的某外部部件可以对原

始数据的一个或多个部分进行加密。为了恢复原始数据，需要提供并解密加密的部分。可用不同的加密密钥对不同的加密部分进行加密。例如，这个特征可用于实现更安全的“双人规则”，由此，第一用户将需要具有使用第一加密进行加密的特定份，第二用户将需要具有使用第二加密密钥进行加密的特定份。为了访问原始数据，这两个用户将需要具有他们各自的加密密钥并且提供他们各自的原始数据的部分。在一个合适方案中，可用公钥对一个或多个数据部分(可能是恢复原始数据所需的强制性份)进行加密。然后可用私钥对该份进行解密从而用于恢复成原始数据。

[0498] 可使用利用强制性份的任何这种合适范例，其中，恢复原始数据不需要所有份。

[0499] 在本发明的一个合适实施例中，可以随机或伪随机地处理，将数据分布到有限数目的数据份中，从而基于统计观点，任何特定数据份接收特定数据单元的概率与剩余份中的任何一个将接收该数据单元的概率相等。结果，每个数据份将具有近似相等的数据比特量。

[0500] 根据本发明的另一个实施例，有限数目的数据份中的每个不需要具有从原始数据的解析和分裂接收数据单元的相等概率。而且，某一份或多份可具有比其余份要高或低的概率。结果，某些份与其它份相比，比特大小可以更大或更小。例如，在两份的情况下，一份可具有接收数据单元的 1% 概率，而第二份具有 99% 概率。由此，应该明白，一旦数据单元被安全数据解析器分布到两份之中，第一份应该具有约 1% 的数据，第二份应该具有 99% 的数据。根据本发明，可使用任何合适概率。

[0501] 应该明白，安全数据解析器可以被设计为根据精确(或者近似精确)的百分比向份分布数据。例如，安全数据解析器可以被设计为将数据的 80% 分布给第一份，将剩余 20% 的数据分布给第二份。

[0502] 根据本发明的另一个实施例，安全数据解析器可以产生多个数据份，这些数据份中的一个或多个具有预定大小。例如，安全数据解析器可以将原始数据分裂成多个数据部分，其中，这些部分之一是精确的 256 比特。在一个合适方案中，如果不可以产生具有需要大小的数据部分，则安全数据解析器可以填塞该部分以使其为正确大小。可使用任何合适大小。

[0503] 在一个合适方案中，数据部分的大小可以是加密密钥、分裂密钥、任何其它合适密钥、或者任何其它合适数据元素的大小。

[0504] 如上所述，在解析和分裂数据的过程中，安全数据解析器可以使用密钥。为了清楚和简洁，在本文中这些密钥将被称作“分裂密钥”。例如，先前介绍的会话主密钥是一种分裂密钥。另外，如上所述，可以在由安全数据解析器产生的数据份内对分裂密钥进行保护。用于保护分裂密钥的任何合适算法可用于在数据份之间对它们进行保护。例如，Shamir 算法可用于保护分裂密钥，由此，产生可用于重建分裂密钥的信息并且将其附加到数据份。根据本发明可以使用任何其它这样的合适算法。

[0505] 类似地，可以根据任何合适算法(例如，Shamir 算法)在一个或多个数据份内对任何合适的加密密钥进行保护。例如，可使用 Shamir 算法或者任何其它合适算法保护用于在解析和分裂之前对数据集进行加密的加密密钥、用于在解析和分裂后对数据部分进行加密的加密密钥或者这二者。

[0506] 根据本发明的一个实施例，可使用诸如全包变换的全都或全不变换(All or

Nothing Transform,AoNT),通过对分裂密钥、加密密钥、任何其它合适数据元素或者它们的任何组合进行变换,进一步保护数据。例如,用于在根据本发明的解析和分裂之前对数据集进行加密的加密密钥可通过 AoNT 算法进行变换。然后,变换的加密密钥可以根据例如 Shamir 算法或者任何其它合适算法分布到各数据份之中。为了重建加密密钥,必须恢复加密的数据集(例如,如果根据本发明使用冗余,则不需要使用所有数据份),从而访问关于根据本领域技术人员公知的 AoNT 的变换的必要信息。当获取了原始加密密钥时,它可用于对加密的数据集进行解密以获取原始数据集。应该明白,本发明的容错特征可以与 AoNT 特征进行结合使用。也就是说,冗余数据可以包括在数据部分中,从而恢复加密的数据集不需要所有的数据部分。

[0507] 应该明白,除了与在解析和分裂之前的数据集对应的各个加密密钥的加密和 AoNT 以外或者代替地,AoNT 可应用于用于在解析和分裂后对数据部分进行加密的加密密钥。同样地,AoNT 可应用于分裂密钥。

[0508] 在本发明的一个实施例中,例如可使用工作组密钥对根据本发明使用的加密密钥、分裂密钥或二者进行进一步加密,从而向保护的数据集提供额外等级的安全性。

[0509] 在本发明的一个实施例中,可以提供一个审计模块,每当调用安全数据解析器对数据进行分裂时,该审计模块进行跟踪。

[0510] 图 36 示出了使用根据本发明的安全数据解析器的部件的可能选项 3600。在图 36 中,在下面概要说明选项的每个组合并且用适当的步骤编号进行标记。安全数据解析器实质上可以是模块式的,从而可以在图 36 所示的每个功能块内使用任何已知算法。例如,其它密钥分裂(例如,秘密共享)算法(例如,Blakely 算法)可用于替代 Shamir,或者 AES 加密可由其它已知加密算法(例如,三重 DES)替代。图 36 的例子中所示的标签仅仅描绘用于本发明的一个实施例中的算法的一个可能组合。应该明白,可使用任何合适算法或者算法组合来替代所标记的算法。

[0511] 1)3610、3612、3614、3615、3616、3617、3618、3619

[0512] 在步骤 3610 中使用先前加密的数据,该数据最终可以分裂成预定数目的份。如果分裂算法要求密钥,则在步骤 3612 中可使用密码术安全伪随机数产生器来产生分裂加密密钥。可选地,在步骤 3614 中可以使用全都或全不变换(AoNT)将分裂加密密钥变换成变换分裂密钥,然后在步骤 3615 中分裂加密密钥成为分裂成具有容错性的预定数目的份的密钥。在步骤 3616 中,该数据然后可分裂成预定数目的份。可以在步骤 3617 中使用容错方案,从而不需要全部的份就可以重新产生数据。一旦建立了份,在步骤 3618 中认证 / 完整性信息可以嵌入到这些份中。可选地,在步骤 3619 中可对每份进行后置加密。

[0513] 2)3111、3612、3614、3615、3616、3617、3618、3619

[0514] 在一些实施例中,可使用由用户或外部系统提供的加密密钥对输入数据进行加密。在步骤 3611 中提供外部密钥。例如,该密钥可以从外部密钥库提供。如果分裂算法要求密钥,则在步骤 3612 中可使用密码术安全伪随机数产生器产生分裂加密密钥。可选地,在步骤 3614 中可以使用全都或全不变换(AoNT)将分裂密钥变换成变换分裂加密密钥,然后在步骤 3615 中分裂密钥成为分裂成具有容错性的预定数目的份的密钥。在步骤 3616 中,该数据然后分裂成预定数目的份。可以在步骤 3617 中使用容错方案,从而不需要全部的份就可以重新产生数据。一旦建立了份,在步骤 3618 中认证 / 完整性信息可以嵌入到这些份

中。可选地，在步骤 3619 中可对每份进行后置加密。

[0515] 3）3612、3613、3614、3615、3612、3614、3615、3616、3617、3618、3619

[0516] 在一些实施例中，在步骤 3612 中可以使用密码术安全伪随机数产生器产生加密密钥以对数据进行变换。在步骤 3613 中可以使用产生的加密密钥对数据进行加密。可选地，在步骤 3614 中可使用全都或全不变换（AoNT）将加密密钥变换成变换加密密钥。然后，在步骤 3615 中，变换加密密钥和 / 或产生的加密密钥可以被分裂到具有容错性的预定数目的份中。如果分裂算法要求密钥，则在步骤 3612 中使用密码术安全伪随机数产生器产生分裂加密密钥。可选地，在步骤 3614 中可以使用全都或全不变换（AoNT）将分裂密钥变换成变换分裂加密密钥，然后在步骤 3615 中分裂密钥成为分裂成具有容错性的预定数目的份的密钥。在步骤 3616 中，该数据然后可分裂成预定数目的份。可以在步骤 3617 中使用容错方案，从而不需要全部的份就可以重新产生数据。一旦建立了份，在步骤 3618 中认证 / 完整性信息可以嵌入到这些份中。可选地，在步骤 3619 中可对每份进行后置加密。

[0517] 4）3612、3614、3615、3616、3617、3618、3619

[0518] 在一些实施例中，数据可以被分裂成预定数目的份。如果分裂算法要求密钥，则在步骤 3612 中使用密码术安全伪随机数产生器产生分裂加密密钥。可选地，在步骤 3614 中可以使用全都或全不变换（AoNT）将分裂密钥变换成变换分裂密钥，然后在步骤 3615 中分裂密钥成为分裂成具有容错性的预定数目的份的密钥。在步骤 3616 中，该数据然后可分裂成预定数目的份。可以在步骤 3617 中使用容错方案，从而不需要全部的份就可以重新产生数据。一旦建立了份，在步骤 3618 中认证 / 完整性信息可以嵌入到这些份中。可选地，在步骤 3619 中可对每份进行后置加密。

[0519] 尽管以上四个选项组合优选用于本发明的一些实施例，但是在其它实施例中任何其它合适的特征、步骤或选项的组合可与安全数据解析器一起使用。

[0520] 安全数据解析器通过使物理分离容易，可以提供灵活的数据保护。可以首先对数据进行加密，然后基于“n 取 m”容错性将数据分裂成多个份。这使得当不能够获取全部数目的份时仍能够重新产生原始信息。例如，一些份在传输过程中可能丢失或损坏。如下文更加详细的描述，基于附于各份的完整性信息和容错性，可以重建丢失或损坏的份。

[0521] 为了建立这些份，可选地，安全数据解析器利用多个密钥。这些密钥可以包括下面密钥中的一个或多个：

[0522] 前置加密密钥：当选择各份的前置加密时，外部密钥可以传递至安全数据解析器。这个密钥可以被产生并且存储在外部的密钥库中（或者其它位置）并且可用于可选地在数据分裂之前对数据进行加密。

[0523] 分裂加密密钥：这个密钥可以在内部产生并且由安全数据解析器用于在分裂之前对数据进行加密。然后，可以使用密钥分裂算法将这个密钥安全地存储在各份内。

[0524] 分裂会话密钥：这个密钥不用于加密算法，而且当选择了随机分裂时，它可用作数据划分算法的密钥。当使用随机分裂时，可在内部产生分裂会话密钥并且由安全数据解析器用其将数据划分成多个份。可以使用密钥分裂算法将这个密钥安全地存储在各份内。

[0525] 后置加密密钥：当选择了各份的后置加密时，外部密钥可以传递至安全数据解析器并且用于对各个份进行后置加密。这个密钥可以产生并且存储在外部的密钥库中或者其它合适位置。

[0526] 在一些实施例中，当以这种方式使用安全数据解析器保护数据时，只有存在所有所需的份和外部加密密钥才可以重装信息。

[0527] 图 37 示出了在一些实施例中使用本发明的安全数据解析器的例示性总览过程 3700。如上所述，安全数据解析器 3706 的两个非常合适的功能可以包括加密 3702 和备份 3704。这样，在一些实施例中，安全数据解析器 3706 可以与 RAID 或备份系统或硬件或软件密码引擎集成在一起。

[0528] 与安全数据解析器 3706 关联的主密钥过程可以包括前置加密过程 3708、加密 / 变换过程 3710、密钥保护过程 3712、解析 / 分布过程 3714、容错过程 3716、份认证过程 3716 和后置加密过程 3720 中的一个或多个。如图 36 所详示，这些过程可以按照几个合适顺序或组合执行。使用的过程的组合和顺序可以取决于特定应用或使用、希望的安全等级、希望可选的前置加密、后置加密还是二者、希望的冗余、基础或集成系统的能力或性能、或者任何其它合适因素或因素的组合。

[0529] 例示性过程 3700 的输出可以是两个或更多的份。如上所述，在一些实施例中，数据可以随机地（或伪随机地）分布到这些份中的每一个。在其它实施例中，可使用确定性算法（或者随机、伪随机和确定性算法的某合适组合）。

[0530] 除了信息资产的个别保护外，有时候需要在不同用户组或者关注团体之间共享信息。于是需要在用户组内控制对各份的访问或者在仅仅允许组成员对份进行重装的那些用户之间共享证明。为此，在本发明的一些实施例中，可将工作组密钥部署给组成员。由于工作组密钥受到危害可潜在允许组外的人访问信息，所以工作组密钥应该被保护并保持保密。在下文讨论用于工作组密钥部署和保护的一些系统和方法。

[0531] 通过对存储在各份内的密钥信息进行加密，工作组密钥概念允许对信息资产的加强保护。一旦执行了这个操作，即使发现了所有所需的份和外部密钥，攻击方无法访问工作组密钥的话也没有希望重建信息。

[0532] 图 38 示出了在份内存储密钥和数据成分的例示性框图 3800。在图 3800 的例子中，省去了可选的前置加密和后置加密步骤，但是在其它实施例中可以包括这些步骤。

[0533] 分裂数据的简化过程包括在加密阶段 3802 使用加密密钥 3804 对数据进行加密。根据本发明，加密密钥 3804 的部分然后可以被分裂并存储在各份 3810 中。分裂加密密钥 3806 的部分也可存储在各份 3810 中。使用分裂加密密钥，数据 3808 于是被分裂并存储在各份 3810 中。

[0534] 为了恢复数据，根据本发明可以获取并恢复分裂加密密钥 3806。然后，可以反转分裂操作以恢复密文。还可以获得并且恢复加密密钥 3804，然后使用该加密密钥对密文进行解密。

[0535] 当利用工作组密钥时，可以稍微改变以上过程以用工作组密钥保护加密密钥。于是，在将加密密钥存储在份中之前，加密密钥可以通过工作组密钥进行加密。在图 39 的例示性框图 3900 中示出了修改的步骤。

[0536] 使用工作组密钥分裂数据的简化过程包括在阶段 3902 中首先使用加密密钥对数据进行加密。然后，在阶段 3904 中可以用工作组密钥对加密密钥进行加密。然后，用工作组密钥加密的加密密钥可以被分裂成多个部分并且存储在各份 3912 内。分裂密钥 3908 也可以被分裂并存储在各份 3912 中。最后，使用分裂密钥 3908 将数据 3910 的部分分裂并存

储在各份 3912 中。

[0537] 为了恢复数据，根据本发明可以获取并恢复分裂密钥。然后，根据本发明可以反转分裂操作以恢复密文。可以获取并恢复加密密钥(使用工作组密钥对其进行了加密)。然后，可以使用工作组密钥对加密密钥进行解密。最后，可使用加密密钥对密文进行解密。

[0538] 有多种用于部署并保护工作组密钥的安全方法。选择哪个方法用于特定应用取决于多个因素。这些因素可以包括所需的安全等级、成本、便利性、以及工作组中用户的数目。下面提供了在一些实施例中使用的一些常用技术。

[0539] 基于硬件的密钥存储

[0540] 基于硬件的方案通常为密码系统内的加密 / 解密密钥的安全性提供最强保证。基于硬件的存储方案的例子包括将密钥存储在便携式装置(例如，智能卡 / 加密狗) 中的防篡改密钥令牌装置或者非便携式密钥存储外设。这些装置被设计为防止由未授权方对密钥材料进行容易的复制。密钥可由信任的机构产生并且分布给用户，或者在硬件内产生。此外，许多密钥存储系统提供多因素认证，其中，密钥的使用要求访问物理对象(令牌) 以及密码短语(passphrase) 或生物测定二者。

[0541] 基于软件的密钥存储

[0542] 尽管对于高安全性部署或应用会期望专用的基于硬件的存储，但是可以选择其它部署将密钥直接存储在本地硬件(例如，盘、RAM 或诸如 USB 驱动器的非易失性 RAM 存储器) 上。针对内部攻击或者在攻击方能够直接访问加密机的情况下，这提供较低等级的保护。

[0543] 为了保护盘上的密钥，基于软件的密钥管理常常通过在从其它认证度量(包括口令和密码短语、其它密钥的存在(例如，来自基于硬件的方案)、生物测定、或者上述度量的任何合适组合) 导出的密钥之下以加密的形式存储密钥来对密钥进行保护。由这些技术提供的安全性的等级可以在从由一些操作系统(例如，MS Windows 和 Linux) 提供的相对较弱密钥保护机制到使用多因素认证实现的更健壮方案的范围内。

[0544] 有利的是，本发明的安全数据解析器可用于多个应用和技术中。例如，电子邮件系统、RAID 系统、视频广播系统、数据库系统、磁带备份系统、或者任何其它合适系统可以具有以任何合适等级集成的安全数据解析器。如上所述，应该明白，还可以集成安全数据解析器以用于通过任何传输介质(例如包括有线、无线或者物理传输介质) 的任何类型的移动中数据的保护和容错。作为一个例子，IP 语音(VoIP)应用可以利用本发明的安全数据解析器解决与在 VoIP 内通常存在的回声和延迟有关的问题。通过使用容错可以消除对漏掉包进行网络再试的需要，即使在丢失预定数目的份的情况下这仍可以保证包传送。数据包(例如，网络包) 还可以以最小延迟和缓冲“即时地(on-the-fly)”被高效地分裂和恢复，从而获得针对各种类型的移动中数据的综合方案。安全数据解析器可作用于网络数据包、网络语音包、文件系统数据块、或者任何其它合适的信息单元。除了与 VoIP 应用集成在一起以外，安全数据解析器可以与文件共享应用(例如，对等文件共享应用)、视频广播应用、电子投票或民意调查应用(可以实现例如 Sensus 协议的电子投票协议和盲签名)、电子邮件应用、或者要求或希望安全通信的任何其它网络应用集成在一起。

[0545] 在一些实施例中，本发明的安全数据解析器在两个不同阶段(即，首标产生阶段和数据划分阶段) 可以提供对移动中网络数据的支持。在图 40A 和 40B 中分别示出了简化的首标产生过程 4000 和简化的数据划分过程 4010。可以对网络包、文件系统块、或者任何其

它合适信息执行这些过程中的一个或二者。

[0546] 在一些实施例中，在网络包流开始时可以执行一次首标产生过程 4000。在步骤 4002 中，可以产生随机（或伪随机）分裂加密密钥 K。然后，在 AES 密钥包装步骤 4004 中，（例如，使用上述的工作组密钥）可选地对分裂加密密钥 K 进行加密。尽管在一些实施例中可以使用 AES 密钥包装，但是在其它实施例中可以使用任何合适的密钥加密或者密钥包装算法。AES 密钥包装步骤 4004 可以对整个分裂加密密钥 K 进行操作，或者该分裂加密密钥可以被解析成几个块（例如，64 比特块）。如果需要，AES 密钥包装步骤 4004 然后可以对分裂加密密钥的块进行操作。

[0547] 在步骤 4006 中，秘密共享算法（例如，Shamir）可用于将分裂加密密钥 K 分裂成多个密钥份。然后，每个密钥份可以嵌入到输出份之一中（例如，份首标中）。最后，份完整性块和（可选的）后置认证标记（例如，MAC）可以附于每份的首标块。每个首标块可以被设计为适合安置在单个数据包内。

[0548] 在首标产生完成后（例如，使用简化的首标产生过程 4000），安全数据解析器可使用简化的数据分裂过程 4010 进入数据划分阶段。在步骤 4012 中，使用分裂加密密钥 K 对流中的每个输入数据包或数据块进行加密。在步骤 4014 中，可以对从步骤 4012 获得的密文计算份完整性信息（例如，哈希值 H）。例如，可以计算出 SHA-256 哈希值。然后，在步骤 4016 中，可以使用根据本发明以上描述的数据分裂算法之一将数据包或数据块划分成两个或更多数据份。在一些实施例中，数据包或数据块可以被分裂为使得每个数据份包含加密的数据包或数据块的基本随机的分布。然后，完整性信息（例如，哈希值 H）可以附于每个数据份。在一些实施例中，还可以计算可选的后置认证标记（例如，MAC）并且将其附于每个数据份。

[0549] 每个数据份可以包括允许数据块或数据包的正确重建所需的元数据。这个信息可以包括在份首标内。元数据可以包括诸如密码密钥份、密钥身份、份现时、签名 /MAC 值和完整性块的信息。为了使带宽效率最大化，元数据可以以紧凑二进制格式存储。

[0550] 例如，在一些实施例中，份首标包括明文首标块，该明文首标块没有加密并且可以包括例如 Shamir 密钥份、每会话现时、每份现时、密钥标识符（例如，工作组密钥标识符和后置认证密钥标识符）的元素。份首标还可以包括通过分裂加密密钥进行加密的加密的首标块。完整性首标块也可以包括在该首标内，完整性首标块可以包括对任何数目的先前块（例如，先前两块）的完整性检查。任何其它合适值或信息也可以包括在份首标内。

[0551] 如图 41 的例示性份格式 4100 所示，首标块 4102 可与两个或更多的输出块 4104 关联。例如首标块 4102 的每个首标块可被设计为适合安置在单个网络数据包内。在一些实施例中，在首标块 4102 从第一位置发送至第二位置后，输出块然后可以被发送。或者，首标块 4102 和输出块 4104 可以同时并行地发送。可以经由一个或更多的类似或不类似的通信路径进行该发送。

[0552] 每个输出块可以包括数据部分 4106 和完整性 / 真实性部分 4108。如上所述，可以使用包括加密的预先划分的数据的份完整性信息（例如，SHA-256 哈希值）的份完整性部分，对每个数据份进行保护。为了在恢复时验证输出块的完整性，安全数据解析器可以比较每份的份完整性块并且然后颠倒分裂算法。然后，可以针对份哈希值验证恢复的数据的哈希值。

[0553] 如上所述，在本发明的一些实施例中，安全数据解析器可以与磁带备份系统结合使用。例如，根据本发明，各个带子可用作节点(即，部分 / 份)。可以使用任何其它合适的布置。例如，由两个或更多磁带组成的磁带库或子系统可被看作单个节点。

[0554] 根据本发明，冗余也可以与磁带一起使用。例如，如果数据集分配在四个磁带(即，部分 / 份)之中，则为了恢复原始数据需要这四个磁带中的两个。应该明白，根据本发明的冗余特征，恢复原始数据可能需要任何合适数目的节点(即，少于全部数目的节点)。这实质上增大了当一个或多个磁带过期时恢复的可能性。

[0555] 还可以用 SHA-256、HMAC 哈希值、任何其它合适值、或者它们的任何组合对每个磁带进行数字保护从而确保不会被篡改。如果磁带上的任何数据或者哈希值变化，则该磁带将不会是用于恢复的候选并且剩余磁带中的任何最小所需数目的磁带将用于恢复数据。

[0556] 在传统的磁带备份系统中，当用户请求将数据写入磁带或者从磁带读取数据时，磁带管理系统(TMS)呈现与物理磁带安装对应的数目。这个磁带安装指向将安装数据的物理驱动器。由人磁带操作员或者磁带机器人在磁带仓中装载磁带。

[0557] 在本发明之下，物理磁带安装可被认为是指向多个物理磁带的逻辑安装点。由于并行性，这不仅增大了数据容量还提高了性能。

[0558] 为了提高性能，磁带节点可以是或者可以包括用于存储磁带映像的盘的 RAID 阵列。由于总是可以在受保护的 RAID 中获得数据，所以这可以实现高速恢复。

[0559] 在任何上述实施例中，可以使用确定性的、概率性、或者既确定性又概率性的数据分布技术，将要保护的数据分布到多个份中。为了防止攻击方开始对任何密码块进行密码攻击，密码块中的比特可以被确定性地分布到份。例如，可以使用 BitSegment (比特段)例程执行分布，或者可以对 BlockSegment (块段)例程进行修改以允许将块的部分分布到多个份。这个策略可以防御累积了少于“M”份的攻击方。

[0560] 在一些实施例中，可以使用密钥式信息分散(例如，通过使用密钥式信息分散算法或“IDA”)，采用密钥式秘密共享例程。密钥式 IDA 的密钥还可以由一个或多个外部工作组密钥、一个或多个共享密钥、或者工作组密钥和共享密钥的任何组合进行保护。这样，可以采用多因素秘密共享方案。在一些实施例中，为了重建数据，至少需要“M”份外加工作组密钥(和 / 或共享密钥)。IDA (或者 IDA 的密钥)还可以被带入加密过程。例如，变换可以被带入明文(例如，在加密之前的前置处理层期间)并且还可以在对明文进行加密之前进一步对明文进行保护。

[0561] 例如，在一些实施例中，密钥式信息分散用于将数据集中的数据的唯一部分分布到两个或更多份中。密钥式信息分散可以使用会话密钥首先加密数据集，然后将数据集的加密的数据的唯一部分分布到两个或更多的加密数据集份，或者既对数据集加密又将数据集的加密的数据的唯一部分分布到两个或更多加密数据集份。例如，为了分布数据集或加密数据集的唯一部分，可以使用秘密共享(或者上述方法，例如 BitSegment 或 BlockSegment (块段))。会话密钥然后可以可选地进行变换(例如，使用全包变换或 AoNT)并且例如使用秘密共享(或者密钥式信息分散和会话密钥)进行共享。

[0562] 在一些实施例中，在密钥的唯一部分被分布或分配到两个或更多会话密钥份之前，可以使用共享密钥(例如，工作组密钥)对会话密钥进行加密。然后，通过将至少一个加密数据集份与至少一个会话密钥份进行组合可以形成两个或更多用户份。在形成用户份

时，在一些实施例中，所述至少一个会话密钥份可以交织到加密的数据集份中。在其它实施例中，所述至少一个会话密钥份可以在至少部分基于共享的工作组密钥的位置插入到加密数据集份中。例如，密钥式信息分散可用于将每个会话密钥份分布到唯一加密数据集份以形成用户份。在至少部分基于共享工作组的位置将会话密钥份交织或插入到加密数据集份可以提高面对密码攻击的安全性。在其它实施例中，一个或更多会话密钥份可以附于加密数据集份的开始或末端以形成用户份。然后，用户份的集合可以单独地存储在至少一个数据储存器上。该数据储存器或多个数据储存器可以位于同一物理位置(例如，位于同一磁或磁带存储装置上)或者在地理上分离(例如，位于不同地理位置处的物理分离的服务器上)。为了重建原始数据集，需要一组授权的用户份和共享的工作组密钥。

[0563] 即使在面对密钥获取启示器(oracle)时，密钥式信息分散仍是安全的。例如，取块密码 E 和针对 E 的密钥获取启示器，该针对 E 的密钥获取启示器取对块密码的输入 / 输出对的列表 (X_1, Y_1)、……、(X_C, T_C) 并且返回与输入 / 输出例子一致的密钥 K (例如，对于所有 i，$Y_i = E_K(X_i)$)。如果没有一致的密钥，则该启示器可以返回特异值 ⊥。这个启示器可以模拟可从输入 / 输出例子的列表恢复密钥的密码分析攻击。

[0564] 在存在密钥获取启示器的情况下，标准的基于块密码的方案可能会失败。例如，在存在密钥获取启示器的情况下，CBC 加密或者 CBCMAC 可能变得完全不安全。

[0565] 如果 Π^{IDA} 是 IDA 方案并且 Π^{Enc} 是由某块密码 E 的操作模式给出的加密方案，则如果在对手具有密钥获取启示器的模式下这两种方案与按照 HK1 或 HK2 的任意完美秘密共享方案(PSS)进行组合时达到健壮计算秘密共享(RCSS)目标，则 (Π^{IDA}, Π^{Enc}) 提供了面对密钥获取攻击的安全性。

[0566] 如果存在 IDA 方案 Π^{IDA} 和加密方案 Π^{Enc} 以使得这对方案提供面对密钥获取攻击的安全性，则实现这对方案的一个方法是具有“聪明(clever)”IDA 和“愚笨(dumb)”加密方案。实现这对方案的另一个方法是具有“愚笨”IDA 和“聪明”加密方案。

[0567] 为了示出聪明 IDA 和愚笨加密方案的使用，在一些实施例中，加密方案可以是 CBC 并且 IDA 可以具有“弱私密”性质。弱私密性质例如是指：如果对 IDA 的输入是块的随机序列 $M = M_1 \cdots M_l$ 并且对手从未授权的集合获得份，则存在某块索引 i 使得对手无法计算 M_i。通过首先向 M 应用信息理论 AoNT (例如，Stinson 的 AoNT)然后应用例如 BlockSegment 的简单 IDA 或者如 Rabin 方案(例如，Reed-Solomon 编码)的比特高效 IDA，可以建立弱私密 IDA。

[0568] 为了示出愚笨 IDA 和聪明加密方案的使用，在一些实施例中，可以使用以双重加密替代单加密的 CBC 模式。现在，即使在复制的情况下仍可以使用任何 IDA。由于对手将被拒绝任何单加密的输入 / 输出例子，所以对于对手来讲具有针对块密码的密钥获取启示器是无用的。

[0569] 尽管聪明 IDA 具有价值，但是在一些环境下它也是无关紧要的，这意味着提供面对密钥获取攻击的安全性所需的“聪明”本可被“推送”到别处。例如，在一些实施例中，不管 IDA 如何聪明，以及不管在 HK1/HK2 环境下用 IDA 尝试达到什么目标，聪明可被从 IDA 推出并且推进加密方案，从而留下固定且愚笨的 IDA。

[0570] 基于以上内容，在一些实施例中，可以使用“普遍健全”的聪明 IDA Π^{IDA}。例如，提供 IDA 从而使得对于所有的加密方案 Π^{Enc}，对 (Π^{IDA}, Π^{Enc}) 普遍地提供面对密钥获取攻击

的安全性。

[0571] 在一些实施例中，提供一种加密方案，该加密方案在面对密钥获取启示器时是RCSS安全的。该方案可以与HK1/HK2、与任何IDA集成以达到面对密钥获取的安全性。使用新方案可能特别有用，例如，使对称加密方案对于密钥获取攻击更加安全。

[0572] 如上所述，经典秘密共享概念通常不是密钥式的。因此，将秘密分解成多份，或者按照既不要求经销商也不要求重建秘密的一方持有任何类型的对称或非对称密钥的方式从这些份重建秘密。然而，可选地，本文所述的安全数据解析器可以是密钥式的。经销商可提供对称密钥，如果这个对称密钥用于数据共享，则进行数据恢复需要这个对称密钥。安全数据解析器可以使用该对称密钥把要保护的消息的唯一部分分散或分布到两个或更多份。

[0573] 共享的密钥可以实现多因素或两因素秘密共享(2FSS)。于是，对手需要排除两个基本不同类型的安全性以破坏安全性机制。例如，为了侵犯秘密共享目标，对手：(1)需要获得一组授权的玩家的份；(2)需要获得应该不能获得的秘密密钥(或者破坏通过该密钥进行密钥式控制的加密机制)。

[0574] 在一些实施例中，一组新的附加要求添加到RCSS目标。这些附加要求可以包括“第二因素”，即密钥拥有。可以加入这些附加要求而不减少原始的一组要求。一组要求可以涉及如果对手知道秘密密钥但没有获得足够份也不能够破坏方案(例如，经典或第一因素要求)，而另一组要求可以涉及如果对手具有秘密密钥但设法获得所有的份也不能够破坏方案(例如，新的或第二因素要求)。

[0575] 在一些实施例中，存在两个第二因素要求：私密要求和真实性要求。在私密要求中，涉及一个博弈，其中，由环境选择秘密密钥K和比特b。对手现在在秘密共享方案的域中提供一对等长消息M_1^0和M_1^1。环境计算M_1^b的份以获得份的矢量$S_1=(S_1[1]、\cdots、S_1[n])$，并且它将份$S_1$(所有它们)给予对手。使用相同密钥K和隐藏的比特b，对手现在可以选择另一对消息(M_2^0,M_2^1)并且任何事情如上进行。对手的工作是输出他相信为b的比特b'。对手私密优势是1小于b=b'的概率的两倍。该博弈获得如下概念：即使获知所有份，如果缺少秘密密钥，对手仍不能够获知关于共享秘密的任何事。

[0576] 在真实性要求中，可以涉及一个博弈，其中，环境选择私密密钥K并且在接下来对Share和Recover的调用中使用它。在一些实施例中，Share和Recover可以修改它们的语法以反映这个密钥的存在。然后，对手对它在秘密共享方案的域中选择的任何消息M_1、...、M_q进行Share请求。响应于每个Share请求，它获得份S_1、…、S_q的对应n矢量。对手的目的是伪造新的明文；如果它输出份S'的矢量使得当送给Recover算法时产生不在$\{M_1、\cdots、M_q\}$中的东西则它获胜。这是“明文完整性”概念。

[0577] 有两种实现多因素秘密共享的方案。第一种方案是通用方案，在以黑盒方式使用基本(R)CSS方案的意义上通用。认证的加密方案用于对要进行CSS共享的消息进行加密，然后例如可以使用秘密共享算法(例如，Blakely或Shamir)对得到的密文进行分配。

[0578] 潜在更加高效的方案是允许共享密钥是工作组密钥。也就是说，(1)可以使用共享密钥对(R)CSS方案的随机产生的会话密钥进行加密，(2)应用于消息(例如，文件)的加密方案可由认证的加密方案替代。这个方案会使性能仅蒙受最小的下降。

[0579] 尽管在上文描述了安全数据解析器的一些应用，但是应该清楚地明白，本发明可以与任何网络应用进行集成以提高安全性、容错性、匿名性、或者上述的任何合适组合。

[0580] 本发明的安全数据解析器可以用于实现云计算数据安全性方案。云计算是基于网络的计算、存储、或者这两者，其中可通过网络向计算机系统和其它装置提供计算和存储资源。云计算资源一般通过互联网进行访问，但是可以在任何合适的公有或私有网络上执行云计算。云计算可以提供计算资源和其下层的硬件部件(例如，服务器、存储装置、网络)之间的一个抽象级别，实现对计算资源池的远程访问。这些云计算资源可被统称为“云”。云计算可用于提供可动态调整且常常虚拟化的资源，作为互联网或任何其它合适的网络或网络组合上的服务。

[0581] 对于云计算，安全性是一个重要的顾虑，因为私有数据(例如来自企业的私有网络)可能在公有网络上传送并且可能在公众可访问或共享的系统(例如，Google (例如 Google Apps Storage)、Dropbox 或 Amazon (例如 Amazon 的 S3 存储设施))内进行处理和存储。这些公众可访问的系统不一定提供了加密的存储空间，然而它们确实向用户提供了在它们的服务器上存储一组文件的能力。安全数据解析器可用于保护云计算资源以及在云与端用户或装置之间传送的数据。例如，安全数据解析器可用于保护云中的数据存储、去往/来自云的移动中数据、云中的网络访问、云中的数据服务、对云中的高性能计算资源的访问、以及云中的任何其它操作。

[0582] 图 42 是云计算安全性方案的例示性框图。包括安全数据解析器 4210 的系统 4200 被耦接到包括云资源 4260 的云 4250。系统 4200 可包括任何合适的硬件，诸如计算机终端、个人计算机、手持式装置(例如，PDA、黑莓、智能电话、平板装置)、蜂窝电话、计算机网络、任何其它合适硬件、或者它们的任何组合。安全数据解析器 4210 可以集成在系统 4200 的任何合适级别。例如，安全数据解析器 4210 可以在非常后端的级别集成到系统 4200 的硬件和/或软件中，从而使安全数据解析器 4210 的存在对于系统 4200 的端用户实质上是透明的。上面例如参照图 27 和 28 更详细地描述了把安全数据解析器集成在合适系统内。云 4250 包括多个例示性云资源 4260，云资源 4260 包括数据存储资源 4260a 和 4260e、数据服务资源 4260b 和 4260g、网络访问控制资源 4260c 和 4260h、以及高性能计算资源 4260d 和 4260f。云资源可由多个云资源提供商(例如，Amazon、Google 或 Dropbox)提供。将参照图 43-56 更详细地描述这些云计算资源中的每个。这些云计算资源仅仅是例示性的。应该理解，可以从系统 4200 访问任何合适数目和类型的云计算资源。

[0583] 云计算的一个优点是系统 4200 的用户将能够访问多个云计算资源而不必投资于专用存储硬件。用户可具有动态地控制系统 4200 可访问的云计算资源的数目和类型的能力。例如，可向系统 4200 提供具有能基于当前需要而动态调整的能力的云中的按需存储资源。在一些实施例中，在系统 4200 上执行的一个或多个软件应用可将系统 4200 耦接到云资源 4260。例如，互联网 web 浏览器可用于将系统 4200 通过互联网耦接到一个或多个云资源 4260。在一些实施例中，与系统 4200 集成或者连接到系统 4200 的硬件可将系统 4200 耦接到云资源 4260。在这两个实施例中，安全数据解析器 4210 都可保护与云资源 4260 的通信和/或存储在云资源 4260 内的数据。云资源 4260 耦接到系统 4200 对于系统 4200 或系统 4200 的用户可以是透明的，从而云资源 4260 对于系统 4200 看起来就像是本地硬件资源。此外，共享的云资源 4260 对于系统 4200 看起来就像是专用硬件资源。

[0584] 在一些实施例中，安全数据解析器 4210 可以将数据加密和分裂从而没有法庭可分辨的数据会横越云或者会存储在云内。云的下层硬件部件(例如，服务器、存储装置、网

络）可以是地理上分散的，以在电网故障、天气事件、或者其它人为或自然事件的情况下确保云资源的连续性。结果，即使云内的一些硬件部件遭受灾难性故障，云资源仍将可访问。云资源 4260 可被设计为带有冗余，以便不管一个或多个硬件故障都能提供不中断的服务。

[0585] 在一些实施例中，本发明的安全解析器可首先将原始数据随机化，然后根据随机化或确定性技术把数据分裂。例如，如果以比特级进行随机化，则本发明的安全解析器可以根据随机化技术（例如，根据随机或伪随机会话密钥）打乱原始数据的比特以形成随机比特序列。然后安全解析器可以根据前面讨论的任何合适技术（例如，合适的信息分散算法（IDA））把这些比特分裂成预定数目的份。

[0586] 图 43 是用于保护通过云移动（即，数据从一个位置传送到另一个位置期间）的数据的云计算安全性方案的例示性框图。图 43 示出了发送方系统 4300，其可包括任何合适的硬件，诸如计算机终端、个人计算机、手持式装置（例如，PDA、黑莓）、蜂窝电话、计算机网络、任何其它合适硬件、或者它们的任何组合。发送方系统 4300 用于产生和 / 或存储数据，该数据例如可以是电子邮件消息、二进制数据文件（例如，图形、语音、视频等）、或者这两者。该数据被根据本发明的安全数据解析器 4310 解析并分裂。得到的数据部分可以通过云 4350 传送给接收方系统 4370。

[0587] 云 4350 可包括例示为云 4350a、4350b 和 4350c 的公有和私有云存储的任何合适组合。例如，云 4350a 和 4350c 可以是公众可访问的云存储资源，诸如 Amazon、Google 或 Dropbox 提供的那些。云 4350b 可以是例如一个企业或教育机构的特定组织之外的任何个人或团体无法访问的私有云。在其它实施例中，云可以是公有和私有云的混合。

[0588] 系统 4300 的接收方系统 4370 可以是上面针对发送方系统 4300 描述的任何合适硬件。根据本发明，分离的数据部分可以在接收方系统 4370 被重新组合以产生原始数据。当行进通过云 4310 时，数据部分可以在一个或多个通信路径上传送，所述通信路径包括互联网和 / 或一个或多个内联网、LAN、WiFi、蓝牙、任何其它合适的有线或无线通信网络、或者它们的任何组合。如上面参照图 28 和 29 所述，即使数据部分中的一些被泄露，原始数据也被安全数据解析器保护。

[0589] 图 44 是用于保护云中的数据服务的云计算安全性方案的例示性框图。在该实施例中，用户 4400 可通过云 4430 向端用户 4440 提供数据服务 4420。安全解析器 4410 可根据所公开的实施例保护数据服务。数据服务 4420 可以是通过云 4430 可访问的任何合适的应用或软件服务。例如，数据服务 4420 可以是作为面向服务的架构（SOA）系统的一部分而实现的基于 web 的应用。数据服务 4420 可以在云 4430 内的一个或多个系统上存储和执行。该云计算实现所提供的抽象使得数据服务 4420 对于端用户 4440 看起来就像虚拟化资源，而不管下层的硬件资源是怎样的。安全解析器 4410 可以保护数据服务 4420 和端用户 4440 之间的移动中数据。安全解析器 4410 还可保护与数据服务 4420 关联的存储数据。与数据服务 4420 关联的存储数据可以在实现数据服务 4420 的一个或多个系统内和 / 或在下面将要更详细地描述的分离的安全云数据存储装置内进行保护。虽然图 44 的数据服务 4420 和其它部分被显示在云 4430 之外，但是应该理解，这些元件中的任一个可以包括在云 4430 内。

[0590] 图 45 是用于保护云中的数据存储资源的云计算安全性方案的例示性框图。包括安全数据解析器 4510 的系统 4500 耦接到包括数据存储资源 4560 的云 4550。安全数据解

析器 4510 可以用于在一个或多个数据存储资源 4560 之间解析和分裂数据。每个数据存储资源 4560 可以表示一个或多个联网的存储装置。这些存储装置可被分配给单个用户 / 系统或者可被多个用户 / 系统共享。安全数据解析器 4510 提供的安全性允许来自多个用户 / 系统的数据在云存储提供商的相同存储装置或资源上安全地共存。该云计算实现提供的抽象使数据存储资源 4560 对于系统 4500 看起来就像单个虚拟化存储资源，不管下层的数据存储资源的数目和位置如何。当把数据写入数据存储资源 4560 或者从其读取数据时，安全数据解析器 4510 可以按照对于端用户透明的方式分裂并重组数据。这样，端用户将能够访问可按需动态调整的存储。

[0591] 使用安全数据解析器 4510 的云中的数据存储是安全的、能复原的、持久的以及私密的。安全数据解析器 4510 通过确保没有法庭可分辨的数据横越云或者被存储在单个存储装置中来保护数据。因为安全数据解析器提供的冗余（即，需要少于全部分离的数据部分的数据部分来重构原始数据），云存储系统是能复原的。在多个存储装置内和 / 或在多个数据存储资源 4560 内存储分离的部分确保了 ：即使存储装置中的一个或多个发生故障或不可访问也可以重构数据。云存储系统是持久的，因为数据存储资源 4560 内的存储装置的丢失对端用户没有影响。如果一个存储装置发生故障，则存储在该存储装置内的数据部分可以在另一存储装置处重建，而不必暴露该数据。此外，存储资源 4560（或者甚至是构成数据存储资源 4560 的多个联网的存储装置）可以在地理上被分散以限制多个故障的风险。最后，可以使用一个或多个密钥使存储在云中的数据保持私密。如上所述，可以根据唯一密钥把数据分配给用户或兴趣团体，从而使得仅有该用户或团体可访问该数据。

[0592] 使用安全数据解析器的云中的数据存储还可提供相对于传统的本地或联网存储的性能提升。通过并行地对多个存储装置进行分离的数据部分的读写，可以提高系统的吞吐量。吞吐量的提高允许使用更慢更便宜的存储装置，而不会实质影响存储系统的整体速度。

[0593] 图 46 是根据所公开的实施例的使用安全数据解析器来保护网络访问的例示性框图。安全数据解析器 4610 可以与网络访问控制块 4620 一起使用以控制对网络资源的访问。如图 46 所示，网络访问控制块 4620 可用于提供用户 4600 和端用户 4640 之间的安全网络通信。在一些实施例中，网络访问控制块 4620 可以提供对云（例如，云 4250，图 42）中的一个或多个网络资源的安全网络访问。可向授权的用户（例如用户 4600 和端用户 4640）提供组范围密钥，所述组范围密钥向用户提供在网络上安全通信和 / 或访问安全网络资源的能力。除非出示正确的证明（例如，组密钥），否则受保护的网络资源不会响应。这可以防止通常的联网攻击，例如，拒绝服务攻击、端口扫描攻击、中间人攻击和重放攻击。

[0594] 除了提供静止存储在通信网络内的数据的安全性以及通过通信网络的移动中数据的安全性之外，网络访问控制块 4620 可与安全数据解析器 4620 一起使用以在不同组的用户或兴趣团体之间共享信息。可以建立协作组以作为安全虚拟网络上的安全兴趣团体进行参与。可将工作组密钥部署给组成员以向组成员提供对网络和联网资源的访问。上面已经讨论了用于工作组密钥部署的系统和方法。

[0595] 图 47 是根据所公开的实施例的使用安全数据解析器来保护对高性能计算资源的访问的例示性框图。安全数据解析器 4710 可用于提供对高性能计算资源 4720 的安全访问。如图 47 所示，端用户 4740 可访问高性能计算资源 4720。在一些实施例中，安全数据解析器

4710 可提供对云(例如,云 4250,图 42)中的高性能资源的安全访问。高性能计算资源可以是大的计算机服务器或服务器群。这些高性能计算资源可向用户提供灵活的、可调整的和可配置的数据服务和数据存储服务。

[0596] 本发明的安全数据解析器可被配置为实现基于服务器的安全数据方案。本发明的安全解析器的基于服务器的方案指的是后端的基于服务器的静止数据(Data at Rest, DAR)方案。服务器可以是任何基于 Windows、基于 Linux、基于 Solaris 的操作系统、或者任何其它合适操作系统。该基于服务器的方案向用户提供了透明文件系统,即,用户观察不到数据的分裂的任何指示。当数据被提供给本发明的安全数据解析器的后端服务器时,数据被分裂成 N 份并被发送到安装 / 附接到该服务器的 N 的可访问(因此可用)的数据存储位置。然而,重建该数据仅需这些份中的某数目 M 份。在一些实施例中,本发明的安全解析器的基于服务器的方案可以首先把原始数据随机化,然后根据随机化或确定性技术把数据分裂。例如,如果以比特级进行随机化,则本发明的安全解析器可以根据随机化技术(例如,根据随机或伪随机会话密钥)打乱原始数据的比特以形成随机比特序列。然后,本发明的安全解析器的基于服务器的方案可以根据前面讨论的任何合适技术(例如,轮询)把这些比特分裂成预定数目的份。对于上面的图 42-47 的实施例以及下面的附图的实施例,将假定本发明的安全解析器可首先根据随机化或确定性技术来分裂数据。此外,在下述的实施例中,分裂数据可包括使用如上所述的任何合适的信息分散算法(IDA)(包括轮询或随机比特分裂)来分裂数据。

[0597] 上述方案使得能够恢复来自本地存储或诸如单个或多个云的远程存储的数据,因为即使当数据首先被随机化然后根据随机化或确定性技术被分裂时,也可从 N 个数据份中的任意 M 个数据份重建数据。下面特别参照图 48-56,提供本发明的安全解析器的基于服务器的方案的进一步说明。在一些实施例中,基于服务器的方案可以与上面参照图 42-47 描述的云计算实施例结合使用。

[0598] 在图 48-50 的实施例中,本发明的安全解析器的基于服务器的方案的实施例将与其结合公有云(例如 Dropbox)的实现以及其它私有、公有和混合云或云计算资源相关地进行描述。

[0599] 图 48 是根据本发明的一个实施例的使用安全数据解析器来保护私有和公有云中的多个存储装置中的数据存储的例示性结构的示意图。私有云 4804 包括处理器 4808,处理器 4808 被配置为实现本发明的安全解析器的基于服务器的方案并且产生加密的数据份 4816b、4818b、4814b、4812b、4820b 和 4822b。私有云 4804 可选地例如经由互联网连接可由端用户装置 4800 访问。远程用户可经由端用户装置 4800 访问他们的存储在私有云 4804 上的数据,并且还可把关于数据份产生和管理的命令从端用户装置 4800 发送到云 4804 的处理器 4804。这些加密的数据份的子集被存储在私有云 4804 内的存储装置上。具体地,数据份 4814b 存储在存储装置 4814a 上,而数据份 4812b 存储在存储装置 4812a 上。处理器 4808 还被配置为在其它的公有、私有或混合云 4802、4806 或 4810 中存储加密的数据份的其它子集。例如,云 4806 可包括 Amazon 提供的公有云资源,而云 4802 可包括由 Dropbox 提供的公有云资源。在该例示性实施例中,份 4818b 和 4816b 被分别存储在云 4802 中的存储装置 4818a 和 4816a 上,份 4822b 被存储在云 4806 中的存储装置 4822a 上,份 4820b 被存储在云 4810 中的存储装置 4820a 上。这样,私有云 4804 的提供商可以利用其它云存储提

供商的存储资源来存储数据份，从而减少了云 4804 的存储装置的存储负担。私有云 4804 的安全解析器在提供针对灾难的健壮数据存活性的同时保护数据，因为重建该数据将只需要 N 个解析的份中的 M 个，其中 M<N。例如，如果对公有或私有云 4806、4810 或 4802 之一的访问中断或丢失，使用加密的数据份的可用子集仍能访问和恢复数据。通常，重建该数据将只需要 N 个解析的份中的 M 个，其中 M<N。例如，如果对公有或私有云 4806、4810 或 4802 之一的访问中断或丢失，则使用加密的数据份的可用子集仍能访问和恢复数据。作为另一例示性例子，如果公有或私有云 4806、4810 或 4802 中的一个或多个内的存储资源停机或者因其它原因无法访问，则使用云内的加密的数据份的可访问子集仍能访问和恢复数据。

[0600] 图 49 是根据本发明的一个实施例的与图 48 的结构相似的使用安全数据解析器来保护多个私有和公有云中的数据存储的例示性结构的示意图。图 49 示出了例如经由互联网连接耦接到诸如膝上型计算机的端用户装置 4902 并且例如经由互联网连接耦接到公有云 4906 和 4908 的私有云 4904。公有云包括公众可访问的云存储资源，诸如 Dropbox 和 Amazon 提供的云存储资源(例如，Amazon 的 S3 存储设施)。上述互联网连接可以是安全或不安全的。在图 49 的例示性实施例中，公有云 4906 由 Dropbox 提供，而公有云 4908 由 Amazon 提供。来自端用户装置 4902 的数据可被发送到私有云 4904。私有云 4904 的处理器 4905 可被配置为实现本发明的安全解析器的基于服务器的方案并产生加密的数据份 4910a、4910b、4910c 和 4910d。份 4910a 和 4910b 存储在私有云 4904 内的存储装置上，而份 4910c 和 4910d 被分别发送到并存储在公有云 4906 和 4908 上。与图 48 的结构一样，私有云 4904 的提供商可以利用其它云存储提供商的存储资源来存储数据份，从而减少云 4904 的存储装置的存储负担。私有云 4904 的安全解析器在提供针对灾难的健壮数据存活性的同时保护数据，因为重建该数据将只需要 N 个解析的份中的 M 个，其中 M<N。例如，如果对公有或私有云 4906 或 4908 之一的访问中断或丢失，则使用加密的数据份的可用子集仍能访问和恢复数据。

[0601] 图 50 是根据本发明的一个实施例的使用安全数据解析器经由互联网 5006 来保护多个私有和公有云中的数据存储的另一例示性结构的示意图。在图 50 的结构中，与图 48 和 49 的结构相似，端用户装置 5002 经由公共可访问的互联网 5006 耦接到私有云 5008。私有云 5008 包括处理器 5001，处理器 5001 被配置为实现本发明的安全解析器的基于服务器的方案并且产生两组加密的数据份 :5014a-d 以及 5016a-d。这些加密的数据份中的一些存储在同一存储装置中，例如，份 5014b 和 5016a、份 5014c 和 5016b，而其它份存储在不同的存储装置中，例如，份 5016c、5016d。份 5014a 和 5014d 被分别发送到并存储在上文已经分别描述了的由公有云存储提供商 Google、Amazon 和 Dropbox 提供的公有云 5010 和 5012。与图 48 和 49 的结构一样，私有云 5008 的提供商可以利用其它云存储提供商的存储资源来存储数据份，从而减少私有云 5008 内的存储装置的存储负担。私有云 5008 的安全解析器因此在提供针对灾难的健壮数据存活性的同时保护数据，因为重建该数据将只需要 N 个解析的份中的 M 个，其中 M<N。因此，如果对公有或私有云 5010 或 5012 之一的访问中断或丢失，则使用加密的数据份的可用子集仍能访问和恢复数据。在一些实施例中，在端用户装置 5002 处可以要求诸如 USB 访问钥匙 5004 的可移动存储装置，用于对希望观看、加密或解密由私有云 5008 的处理器 5001 管理的数据的远程用户的身份进行认证。在一些实施例中，在端用户装置 5002 处可以要求诸如 USB 令牌 5004 的可移动存储装置，用以由私有云 5008

的处理器 5001 启动数据的加密、解密或分裂。在一些实施例中，使用任何合适的信息分散算法（IDA）来分裂数据。在一些实施例中，在分裂之前首先把数据随机化。在一些实施例中，用户可以管理其密码密钥自身。在这些实施例中，用户的密钥可以存储在诸如 USB 令牌 5004 的用户的端装置或者端用户装置 5002 上。在其它实施例中，可以使用任何合适的集中式或分散式密钥管理系统来管理用户或工作组的密码密钥。

[0602] 在一些实施例中，为了实现在多个不同的端用户装置的每个进行数据观看和 / 或重构，一个或多个密码密钥和 / 或一个或多个数据份可以存储在 USB 存储装置 5004 上。此外，一个或多个数据份也可存储在云 5010 和 / 或 5012 上。因此，拥有该便携式用户装置的用户可以从不同于装置 5002 的端用户装置访问 USB 存储装置 5004，以从在 USB 存储装置 5004（如果需要的话，还有云）上分散的份观看和 / 或重建数据。例如，两个数据份可以存储在 USB 存储装置 5004 上，两个数据份可以存储在云 5010 和 5012 的每个中。拥有 USB 存储装置 5004 的用户可以使用带有本发明的安全解析器的耦接到 USB 存储装置 5004 的任何计算装置来访问存储在装置 5004 上的这两个数据份。例如，用户可以使用第一膝上型计算机创建份并在 USB 存储装置 5004 和云上分散份，然后可以使用不同的第二膝上型计算机从 USB 存储装置 5004 和 / 或云 5010 和 5012 检索所述份，然后从检索到的份重构 / 重建数据。

[0603] 在一些实施例中，本发明的安全解析器通过确保丢失或被盗装置的数据保持安全并无法破译可以提供存储数据的保密性、可用性和完整性。在一些实施例中，本发明可以包括在启用任意 Windows 或 Linux 的 PC 或端用户装置（例如移动电话、膝上型计算机、个人计算机、平板计算机、智能电话、机顶盒等）的背景下在内核级运行的软件。在一些实施例中，诸如 Security First 公司的 FIPS 140-2 认证、符合 Suite B 的 SecureParser Extended(SPx) 的安全解析器可用于分裂要保护的数据。在一些实施例中，执行 FIPS 140-2AES 256 加密、随机比特数据分裂、完整性检查和重新加密分裂的份。在一些实施例中，使用任何合适的信息分散算法（IDA）来分裂数据。在一些实施例中，分裂是确定性的。在一些实施例中，也可以在分裂之前把数据随机化。在一些实施例中，没有正确的证明和访问权，存储到用户的端装置的安全位置（例如“C:”驱动器）的任何文件是不可见的。在一些实施例中，没有需要的密码密钥和认证过程，甚至无法看到或恢复文件名。

[0604] 在一些实施例中，创建一组 N 个份，本发明的安全解析器将这 N 个份存储在 N 个分离的（可能是地理上分散的）存储位置。例如，可以创建四（4）个加密的份，然后本发明的安全解析器把这四个加密的份存储在四（4）个分离的存储位置。图 51-53 示出了本发明的安全解析器的两个这种实施例，其中创建四个加密的份。

[0605] 图 51 是根据本发明的一个实施例的使用安全数据解析器来保护用户的可移动存储装置 5104 中和海量存储装置 5106 上的数据存储的例示性结构的示意图。图 51 示出了已经产生四个加密的份 5108a、5108b、5108c 和 5108d 的诸如膝上型计算机的端用户装置 5102。这些加密的份 5108a-d 中的每个存储在端用户装置 5102 的海量存储装置 5106 内的不同存储扇区中。端用户装置的安全解析器在提供针对灾难的健壮数据存活性的同时保护数据，因为重建该数据将只需要 N 个解析的份中的 M 个，其中 M<N。在图 51 的实施例中，存在 4 个份，重构数据将需要这些份中的 2 或 3 个。假定重构数据需要四个加密的份中的仅两个份或者四个加密的份中的三个份，如果一个或两个加密的份丢失，例如，如果海量存储 5106 的一个扇区损坏，则加速了灾难恢复过程。可以使用可移动存储装置 5104 存储为了观

看和 / 或解密和 / 或加密端用户装置 5102 的海量存储 5106 内的数据可能需要的一个或多个密码访问密钥。在一些实施例中，如果在可移动存储装置 5104 上没有密码密钥，则无法解密和 / 或重构加密的数据份 5108a-d。在一些实施例中，用户可以管理其密码密钥自身。在这些实施例中，用户的密钥可以存储在诸如可移动存储装置（例如 USB 存储器）5104 的用户的端装置或者端用户装置 5102 上。在其它实施例中，可以使用任何合适的集中式或分散式密钥管理系统来管理用户或工作组的密码密钥。

[0606] 在一些实施例中，为了实现在多个不同的端用户装置的每个进行数据观看和 / 或重构，一个或多个密码密钥和 / 或一个或多个数据份可以存储在 USB 存储装置 5104 上。此外，一个或多个数据份也可存储在云上。因此，拥有该便携式用户装置的用户可以从不同于装置 5102 的端用户装置访问 USB 存储装置 5104，以从在 USB 存储装置 5104（如果需要的话，还有云）上分散的份观看和 / 或重建数据。例如，两个数据份可以存储在 USB 存储装置 5104 上，两个数据份可以存储在端用户装置 5102 中。拥有 USB 存储装置 5104 的用户可以使用带有本发明的安全解析器的耦接到 USB 存储装置 5104 的任何计算装置来访问存储在 USB 存储装置 5104 上的这两个数据份。例如，用户可以使用第一膝上型计算机创建份并在 USB 存储装置 5104 和端用户装置 5102 上分散份，然后可以使用不同的第二膝上型计算机从 USB 存储装置 5104 检索所述份，并且假定这两个份足以重构数据，则从这两个份重构 / 重建数据。

[0607] 图 52 是根据本发明的一个实施例的使用安全数据解析器来保护多个用户存储装置中的数据存储的例示性结构的示意图。图 52 示出了已经产生四个加密的份 5208a、5208b、5208c 和 5208d 的诸如膝上型计算机的端用户装置 5202。这些加密的份 5208a-d 中的每个存储在地理上分散的存储位置和 / 或同一存储位置的不同部分。具体地，加密的份 5208c 和 5208d 存储在膝上型计算机 5202 的海量存储装置 5206 的两个不同扇区中，而加密的份 5308a 和 5308b 均存储在诸如 USB 存储装置的可移动存储装置 5204 上。端用户装置的安全解析器在提供针对灾难的健壮数据存活性的同时保护数据，因为重建该数据将只需要 N 个解析的份中的 M 个，其中 M<N。在图 52 的实施例中，存在 4 个份，重构数据将需要这些份中的 2 或 3 个。这样，加密的份在地理上和物理上被分散，假定重构数据需要四个加密的份中的仅两个份或者四个加密的份中的三个份，如果一个或两个加密的份丢失，则加速了灾难恢复过程。例如，如果海量存储 5202 的扇区之一损坏，或者如果诸如 USB 存储装置 5204 的可移动存储装置丢失，或者如果发生这两者的组合，则会出现这种丢失。

[0608] 在一些实施例中，替代于把加密的份存储在 USB 存储装置 5204 上或者除了把加密的份存储在 USB 存储装置 5204 上外，一个或多个密钥（例如加密密钥、分裂密钥或认证密钥）被存储在 USB 存储装置 5204 上。这些密钥可用于对存储在 USB 存储装置 5204 自身上或其它地方（例如端用户装置海量存储 5202 中或公有或私有云存储中）的数据份进行分裂、加密 / 解密、或认证。例如，用户可将一密钥存储在 USB 存储装置 5204 上并使用该密钥来解密存储在海量存储装置 5202 上的加密的数据份。作为另一例示性例子，两个数据份可以存储在 USB 存储装置 5204 上，两个数据份可以存储在端用户装置海量存储 5202 中。拥有 USB 存储装置 5204 的用户可以使用带有本发明的安全解析器的耦接到 USB 存储装置 5204 的任何计算装置来访问存储在 USB 存储装置 5204 上的密钥。例如，用户可以使用第一膝上型计算机把密钥存储在 USB 存储装置 5204 内，然后可以使用不同的第二膝上型计算机从 USB 存

储装置 5004 检索该密钥。该密钥然后可被用于对数据进行加密 / 解密、分裂、或认证。

[0609] 在一些实施例中，为了实现在多个不同的端用户装置的每个进行数据观看和 / 或重构，一个或多个密码密钥和 / 或一个或多个数据份可以存储在 USB 存储装置 5204 上。此外，一个或多个数据份也可存储在云上。因此，拥有该便携式用户装置的用户可以从不同于装置 5202 的端用户装置访问 USB 存储装置 5204，以从在 USB 存储装置 5204（如果需要的话，还有云）上分散的份观看和 / 或重建数据。例如，两个数据份可以存储在 USB 存储装置 5204 上，两个数据份可以存储在端用户装置 5202 中。拥有 USB 存储装置 5204 的用户可以使用带有本发明的安全解析器的耦接到 USB 存储装置 5204 的任何计算装置来访问存储在 USB 存储装置 5204 上的这两个数据份。例如，用户可以使用第一膝上型计算机创建份并在 USB 存储装置 5204 和端用户装置 5202 上分散份，然后可以使用不同的第二膝上型计算机从 USB 存储装置 5204 检索所述份，并且假定这两个份足以重构数据，则从这两个份重构 / 重建数据。

[0610] 图 53 是根据本发明的一个实施例的使用安全数据解析器来保护多个公有和私有云以及至少一个用户存储装置中的数据存储的例示性结构的示意图。图 53 示出了已经产生四个加密的份 5306a、5306b、5306c 和 5306d 的诸如膝上型计算机的端用户装置 5302。这些加密的份 5306a-d 中的每个存储在地理上分散的存储位置和 / 或同一存储位置的不同部分。具体地，加密的份 5306a 和 5306b 存储在膝上型计算机 5302 的海量存储装置 5308 的两个不同扇区中，而加密的份 5306c 通过在安全网络连接上的传输而存储在诸如 Amazon 的 S3 云存储的公众可访问云存储 5310 中，并且加密的份 5306d 通过在安全网络连接上的传输而存储在诸如 Dropbox 的云存储的公众可访问云存储 5312 中。这样，加密的份在地理上和物理上被分散，假定重构数据需要四个加密的份中的仅两个份或者四个加密的份中的三个份，如果一个或两个加密的份丢失，则加速了灾难恢复过程。例如，如果海量存储 5308 的扇区之一损坏，或者如果失去端用户装置 5302 与云 5310 和 5312 之间的互联网连接，则会出现这种丢失。

[0611] 在图 51-53 的每个实施例中，加密数据份的产生过程分裂过程对于用户是透明的。此外，本发明的安全解析器在提供针对灾难的健壮数据存活性的同时保护数据，因为重建数据将只需要 N 个解析的份中的 M 个，其中 M<N。例如，在上述的一些实施例中，重构或重建数据将只需要四（4）个解析的份中的两（2）个或三（3）个。如果硬盘驱动器的扇区损坏、或者可移动 USB 装置丢失、或者远程存储位置停机或无法访问，仍能够访问和恢复数据。此外，如果故障的驱动器的份被恢复，或者如果一个份被盗、离线或被破解（hack），数据仍可保持安全和受保护，因为任何单个的解析的份不包含法庭可分辨的信息。换言之，在没有首先具有对应的第二和 / 或第三个份、正确的用户认证、本发明的安全解析器以及在一些情况下的 USB 钥匙或 USB 存储装置的情况下，单个的解析的份无法被重构、解密、破解或恢复。

[0612] 在一些实施例中，本发明的安全解析器可用于移动装置，诸如苹果 iPad、RIM 黑莓、苹果 iPhone、摩托罗拉 Droid 电话、或者任何合适的移动装置。本领域技术人员将认识到这里公开的系统和方法适用于各种各样的端用户装置，包括但不限于移动装置、个人计算机、平板计算机、智能电话等。

[0613] 本发明的安全解析器可以使用一个或多个处理器实现，每个处理器执行诸如密钥产生、数据加密、份产生、数据解密等的安全解析器功能中的一个或多个。在一些实施例中，

分裂数据包括对数据进行密码分裂，例如随机比特分裂。在一些实施例中，使用任何合适的信息分散算法（IDA）来分裂数据。处理器可以是任何合适的处理器，例如 Intel 或 AMD，并且可以运行基于服务器的平台的后端。在一些实施例中，可以使用一个或多个专门的协处理器来加速本发明的安全解析器的操作。在下述的图 54-56 的实施例中，在一个或多个专门的协处理器上实现本发明的安全解析器的一个或多个功能，这实现了安全解析器功能的加速。在一些实施例中，协处理器可包括在安全解析器硬件平台的主板或子板中或者其任何合适组合之中。

[0614] 图 54 是根据本发明的一个实施例的用于安全数据解析器的协处理器加速装置 5400 的示意图。装置 5400 包括两个处理器：中央处理单元（CPU）或主处理器 5402、以及快速处理单元（RPU）或辅助处理器 5404。处理器 5402 和 5404 被彼此耦接，并且还耦接到存储器装置 5406 和海量存储装置 5408。这些装置的耦接可以包括使用互连总线。CPU 和 RPU 中的每个可包括单个微处理器或者将 CPU 和 / 或 RPU 配置为多处理器系统的多个微处理器。存储器 5406 可包括动态随机存取存储器（DRAM）和 / 或高速缓冲存储器。存储器 5406 可包括至少两个专门的存储装置，CPU 5402 和 RPU 5404 中的每个使用一个。海量存储装置 5408 可包括一个或多个磁盘或磁带驱动器或光盘驱动器，用于存储供 CPU 5402 和 / 或 RPU 5406 使用的数据和指令。海量存储装置 5408 还可包括用于各种便携式介质的一个或多个驱动器，诸如软盘、压缩盘只读存储器（CD-ROM）、DVD、闪存驱动器、或者集成电路非易失性存储器适配器（即 PCMCIA 适配器），用以向 CPU 5402 和 / 或 RPU 5406 输入数据和代码以及从其输出数据和代码。CPU 5402 和 / 或 RPU 5406 均可包括用于通信的一个或多个输入 / 输出接口，作为示例示出为通信总线 5410。通信总线还可包括用于经由网络 5412 的数据通信的接口。网络 5412 可包括一个或多个存储装置，例如云存储装置、NAS、SAN 等。经由通信总线 5410 至网络 5412 的接口可以是调制解调器、网卡、串行端口、总线适配器或者用于与一个或多个飞机上或地面上的系统通信的任何其它合适的数据通信机构。到网络 5412 的通信链路例如可以是光学的、有线的或无线的（例如经由卫星或蜂窝网络）。

[0615] 在一些实施例中，RPU 可包括独立磁盘冗余阵列（RAID）处理单元，其实现与协处理器加速装置 5400 关联的一个或多个存储装置的一个或多个 RAID 功能。在一些实施例中，RPU 5404 可包括执行阵列内建计算和 / 或 RAID 计算的通用或专用集成电路（IC）。在一些实施例中，RPU 5404 可经由诸如耦接到 RPU 的 PCIe 总线的 PCIe 连接而被耦接到 CPU 5402。如果 RPU 包括 RAID 处理单元，则 PCIe 连接可包括专门的 RAID 适配器。在一些实施例中，PCIe 卡可以以 10G 比特 / 秒（Gb/s）以上运行。在一些实施例中，RPU 5404 可经由 HT 连接耦接到 CPU 5402，诸如连接到 HT 总线的套接字 RPU。处理器 5402 和 5404 通常将访问相同的内存和海量存储装置，以使得同一数据对于这两个处理器都可访问。协处理器可执行专门的安全解析加速功能，包括但不限于数据分裂、加密和解密。这些功能彼此独立，并且可使用不同的算法执行。例如，加密可以使用任何上述技术执行，而分裂可以使用任何合适的信息分散算法（IDA）（诸如上述的那些算法）执行。在一些实施例中，RPU 可耦接到协处理器加速装置 5400 外部的现场可编程门阵列（FPGA）装置，该 FPGA 装置也能执行本发明的安全解析器的专门加速功能。

[0616] 图 55 是根据本明的一个实施例的针对安全数据解析器使用图 54 的协处理器加速装置 5400 的例示性加速过程的第一处理流程图。连续参照图 54 和 55，在该例示性实施例

中，RPU 5510 可以经由 HT 连接而耦接到 CPU 5520，诸如经由 HT 总线的套接字 RPU。图 55 的左侧示出了可由 CPU 执行的诸如数据分裂和份产生功能（图 39 中的 3910 和 3912）的安全解析器的某些功能，而诸如加密（例如 AES、IDA、SHA 算法）的其它功能（图 39 中的 3902、3904、3906）可由 RPU 执行。在图 55 的右侧示出了加密和加密份产生的这些功能，其中存在是 CPU 还是 RPU 执行特定安全解析器功能的指示。

[0617] 图 56 是根据本发明的一个实施例的针对安全数据解析器使用图 54 的协处理器加速装置 5400 的例示性加速过程的第二处理流程图。连续参照图 54 和 56，在该例示性实施例中，RPU 5610 可以经由 HT 连接而耦接到 CPU 5620，诸如经由 HT 总线的套接字 RPU。图 56 的左侧示出了可由 CPU 执行的诸如数据分裂和份产生功能（图 39 中的 3910 和 3912）的安全解析器的某些功能，而诸如加密（例如 AES、IDA、SHA 算法）的其它功能（图 39 中的 3902、3904、3906）可由 RPU 执行。在图 55 的右侧示出了加密和加密份产生的这些功能，其中存在是 CPU 还是 RPU 执行特定安全解析器功能的指示。

[0618] 参照描述本发明的安全解析器的基于服务器的方案的图 48-56 中的实施例，存在基于服务器的方案可启用或提供的本发明的安全解析器的几个另外的功能和特性。除了执行密码分裂和数据份重建，可以包括其它功能，诸如加密数据份的块级别更新和密码密钥管理。后面的描述将说明这些功能中的每一个。本领域技术人员将认识到，该功能可以容易地包括在参照图 48-56 描述的任何实施例中。

[0619] 在一些实施例中，本发明的安全解析器的基于服务器的方案允许对文件的块级别更新 / 改变，而不是更新 / 改变整个数据文件。在一些实施例中，一旦数据份被从安全解析器发送到云存储装置，为了更高效地操作，当用户或工作组更新下层数据时，代替于恢复整个数据文件，使用本发明的密码系统，可以仅将特定数据份的文件块级别的更新发送到云存储装置。因此，当仅对数据文件进行较小改变时，不执行也不需要恢复整个数据文件。

[0620] 在一些实施例中，本发明的安全解析器的基于服务器的方案针对每个数据份产生一个存根。在一些实施例中，存根可包括其关联数据份的属性列表并与该数据份存储在一起。在一些实施例中，存根可包括关于数据份的信息，例如包括数据份的名称、数据份的创建日期、数据份的最后修改时间、指向数据份在存储装置的文件系统内的位置的指针等。这些信息可用于向用户快速提供关于数据份的信息。在一些实施例中，用户可指定存储存根的存根目录。例如，用户可指定应在其上存储存根目录的存储装置上的特定虚拟或物理驱动器。例如，可以针对用户创建存根目录，其中目录中的每个存根把用户指引到在海量存储装置、可移动存储装置、公有云、私有云或它们的任何组合中由安全解析器存储的安全数据。这样，可以利用存根为用户产生数据份的虚拟文件系统。

[0621] 在一些实施例中，存根可存储在与数据份分离的位置、与数据份相同的位置、或者存储在这两者。在一些实施例中，当用户希望观看数据份的某信息时，他们可访问存根目录。在一些实施例中，代替于直接观看存根目录，存根被从存根目录取回，根据本发明的安全解析器的基于服务器的方案进行处理，随后用于向用户提供上述信息。这样，可以利用存根为用户产生数据份的虚拟文件系统。

[0622] 在一些实施例中，存根被存储在数据份的相应首标中。这样，如果用户希望观看存根中的信息，则存根被从首标中取回，根据本发明的安全解析器的基于服务器的方案进行处理，随后存根目录被产生并提供给用户。

[0623] 在一些实施例中，本发明的安全解析器的基于服务器的方案使用上述技术频繁检查存根和 / 或加密的数据份的数据完整性。即使当用户没有启动或提示时，本发明的安全解析器实质上主动地检索并检查数据份的数据完整性。如果数据份或存根丢失或损坏，本发明的安全解析器尝试重新创建或恢复该存根或数据份。

[0624] 本发明的安全解析器的基于服务器的方案可被配置为提供集中式密码密钥管理设施。具体地，用于加密 / 解密数据、数据份以及跨多个存储装置和系统的通信会话的密码密钥可以存储在企业的存储设施（例如，企业的私有云）内的中心位置。该集中式密钥管理设施还可以与基于硬件的基于密钥管理的方案（诸如 MD 州 Belcamp 的 SafeNet 公司提供的那些方案）对接，或者与基于软件的密钥管理系统对接。例如，现有的私有云可以经由认证 / 访问 / 授权系统控制对加密的数据份的访问，并且基于服务器的方案可使用认证信息以允许访问用于对那些份进行加密的密码密钥，从而允许用户对数据进行密码分裂或者恢复加密的数据份。换言之，本发明的安全解析器的基于服务器的方案可以与现有的认证 / 访问 / 授权系统结合地起作用。这样，不会迫使企业改变其管理用户和工作组对数据的访问的当前方式。

[0625] 在一些实施例中，本发明的安全解析器的基于服务器的方案可以在不解密任何加密的数据份的情况下执行份重建。在一些实施例中，本发明的安全解析器的基于服务器的方案可以在不解密任何加密的数据份的情况下使用一个或多个新密钥重新产生数据的分裂。图 57 示出了根据本发明的例示性实施例的把数据分裂成 N 份并存储的过程 5700。图 58 示出了根据本发明的例示性实施例的对数据份进行重建和 / 或重新施加密钥（re-key）的过程。在图 57 和 58 的每个中，过程的每个步骤可以是可选的。例如，不是必须在分裂数据之前对数据加密。

[0626] 参照图 57，安全解析器首先使用加密密钥对数据进行加密（5702）。该加密密钥可以在本发明的安全解析器的内部产生。该加密密钥可以至少部分地基于外部工作组密钥来产生。安全解析器然后使用分裂密钥把数据分裂成 N 个份（5704）。该分裂密钥可以在本发明的安全解析器的内部产生。该分裂密钥可以至少部分地基于外部工作组密钥来产生。安全解析器然后确保将仅需 N 个份中的 M 个份来重建所述数据（5706），并使用认证密钥对这 N 个份进行认证（5708）。该认证密钥可以在本发明的安全解析器的内部产生。该认证密钥可以至少部分地基于外部工作组密钥来产生。认证密钥、分裂密钥和加密密钥均使用密钥加密密钥进行包装（wrap）（5710）。KEK 然后被分裂并存储在 N 个份的首标内（5712）。这 N 个份然后被分散在 N 个存储位置。

[0627] 在一些实例中，用户或企业希望针对一组数据份使用新的分裂密钥和 / 或新的认证密钥。采用本发明的安全解析器的基于服务器的方案，可以在不解密任何数据份的情况下执行数据的这种重新施加密钥。在其它实例中，用户或企业希望重新产生一组新的数据份，因为一个或多个现有数据份已损坏、丢失或因其它原因无法访问。采用本发明的安全解析器的基于服务器的方案，可以在不解密任何剩余可用数据份的情况下执行丢失数据份的这种重建。参照图 58，假定 N-M 个数据份损坏或因其它原因无法访问，安全解析器从其存储位置检索 N 个份中的剩余 M 个份（5802）。使用认证密钥对这 M 个份进行认证（5804）。使用认证后的 M 个份，由安全解析器重构加密数据（5806）。然后使用分裂密钥重新产生 N 个份（5808），使用认证密钥对这 N 个份进行认证（5810）。如果对于步骤 5808 或 5810 使用了

不同的分裂密钥或认证密钥(5812),则检索 M 个份中的每个份的首标(5816),重构密钥加密密钥(5818),并且与步骤 5710 和 5712 (图 57)的处理相似,使用该密钥加密密钥来包装/加密新的分裂密钥和/或认证密钥(5820)。这 N 个份然后被存储在本发明的安全解析器的一个或多个存储装置中(5822)。如果在步骤 5808 或 5810 中没有使用不同的分裂密钥或认证密钥(5812),则丢失/无法访问的 N-M 个份被存储在本发明的安全解析器的一个或多个存储装置中(5814)。

[0628] 本发明的安全解析器的基于服务器的方案可以被配置为保护数据份(诸如上面与图 42-58 的实施例相关地描述的数据份)的文件名。在一些实施例中,当例如使用 IDA 把一个文件分裂成 N 个数据份时,产生的数据份被存储在存储网络中的一个或多个份位置。存储网络可以包括私有云、公有云、混合云、可移动存储装置、海量存储装置、或者它们的任何组合。在许多应用中,将有多于一个的文件被分裂并存储在存储网络中的份位置。换言之,可以有几个文件,每个文件可被分裂成 N 个数据份(例如使用 IDA),其中每个产生的数据份可以作为文件被存储在份位置。在这些应用中,具有把份位置的一个数据份与产生了该数据份的文件相关联的诸如文件名的唯一标识符是有利的。

[0629] 在一些实施例中,本发明的安全解析器可以被配置为使用原始文件(即,要分裂的文件)的文件名的一部分来命名数据份使之具有与原始文件相同的名字。作为例示性例子,如果原始文件“2010Budget.xls”被分裂成 4 个数据份,这些数据份可被命名为“2010Budget.xls.1”、“2010Budget.xls.2”、“2010Budget.xls.3”和“2010Budget.xls.4”,从而将每个产生的数据份与原始文件关联起来。通过该过程,本发明的安全解析器将能够高效地定位数据份并把它们与原始文件关联。然而,该过程的缺陷在于,它可能向第三方暴露了诸如以下事实的信息:预算信息是针对 2010 年的。在许多应用中,以这种方式暴露文件名是不可接受的,因此不能轻易地将数据份的文件名与原始文件的文件名关联。

[0630] 在一些实施例中,本发明的安全解析器可被配置为首先保护文件名,将使用诸如 HMAC-SHA256 的认证算法把原始文件的文件名哈希(hash)成无法反转的值。本发明的安全解析器将因此用 HMAC-SHA256 算法处理原始文件的文件名以获得“哈希的”文件名并接收安全的并且无法反转为原始文件的文件名的认证值。然后使用该哈希的文件名而不是原始文件的文件名来产生与原始文件关联的数据份的文件名。在这些实施例中,为了定位与原始文件的文件名关联的(存储网络上的)数据份,本发明的安全解析器将再次对原始文件名使用 HMAC-SHA256 算法并重新产生认证值。在一些实施例中,原始文件名和产生的份的文件名的认证值基本相等。本发明的安全解析器然后将在存储网络上的存储位置搜索与该认证值匹配的数据份文件名。存储网络可包括私有云、公有云、混合云、可移动存储装置、海量存储装置、或者它们的任何组合。在一些实施例中,使用原始文件名的全路径,从而针对具有全路径的文件(例如,“\Marketing\2010Budget.xls”)产生的认证值不同于针对具有全路径的文件(例如,“\Sales\2010Budget.xls”)产生的认证值。在一些实施例中,通过对文件的全路径(全路径包括份位置)进行哈希使得到的对应于每个数据份位置的数据份文件名不同。例如,通过把数据份的份编号附于原始文件的全路径末尾,例如\Sales\2010Budget.xls.1”,得到的数据份文件名对于每个数据份位置不同。

[0631] 在一些实施例中,如上所述,本发明的安全解析器通过使用诸如 AES 的加密算法对原始文件名的全路径加密来保护文件的文件名。这种加密确保了原始文件的文件名是安

全的，直到它被本发明的安全解析器基于对存储网络上的份位置的访问、检索到的数据份以及加密密钥而被解密。存储网络可包括私有云、公有云、混合云、可移动存储装置、海量存储装置、或者它们的任何组合。与上述的例子一样，通过首先把诸如数据份的份编号的附加信息附于原始文件的全路径末尾，可以创建每个份位置的唯一数据份文件名。

[0632] 尽管在上文描述了安全数据解析器的一些应用，但是应该清楚地明白，本发明可以与任何网络应用进行集成以提高安全性、容错性、匿名性、或者上述的任何合适组合。

[0633] 此外，鉴于本文的公开，熟练技术人员将容易想到其它的组合、添加、替代和修改。

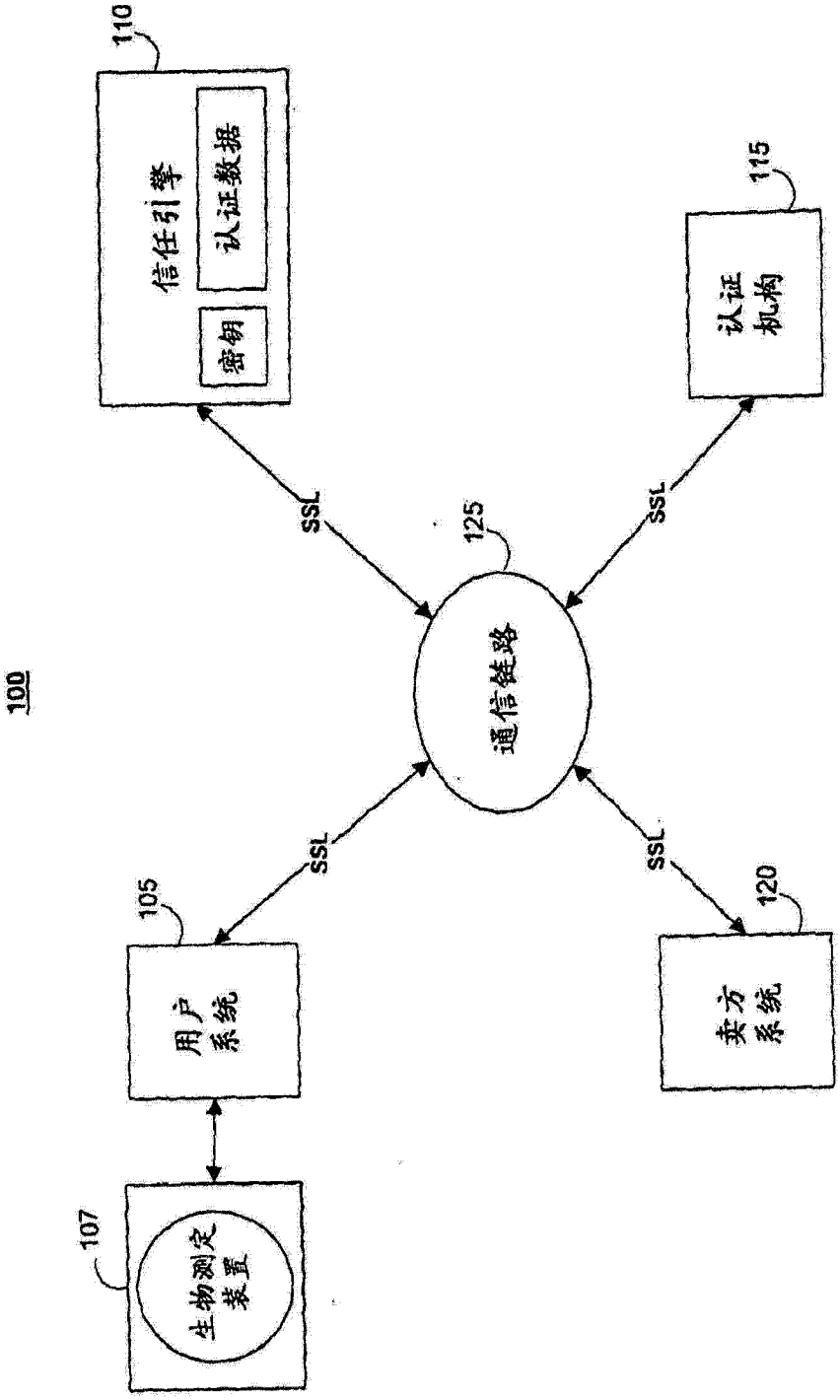

100
110
信任引擎
认证数据
密钥
115
认证机构
SSL
SSL
125
通信链路
SSL
SSL
105
用户系统
120
卖方系统
107
生物测定装置

图 1

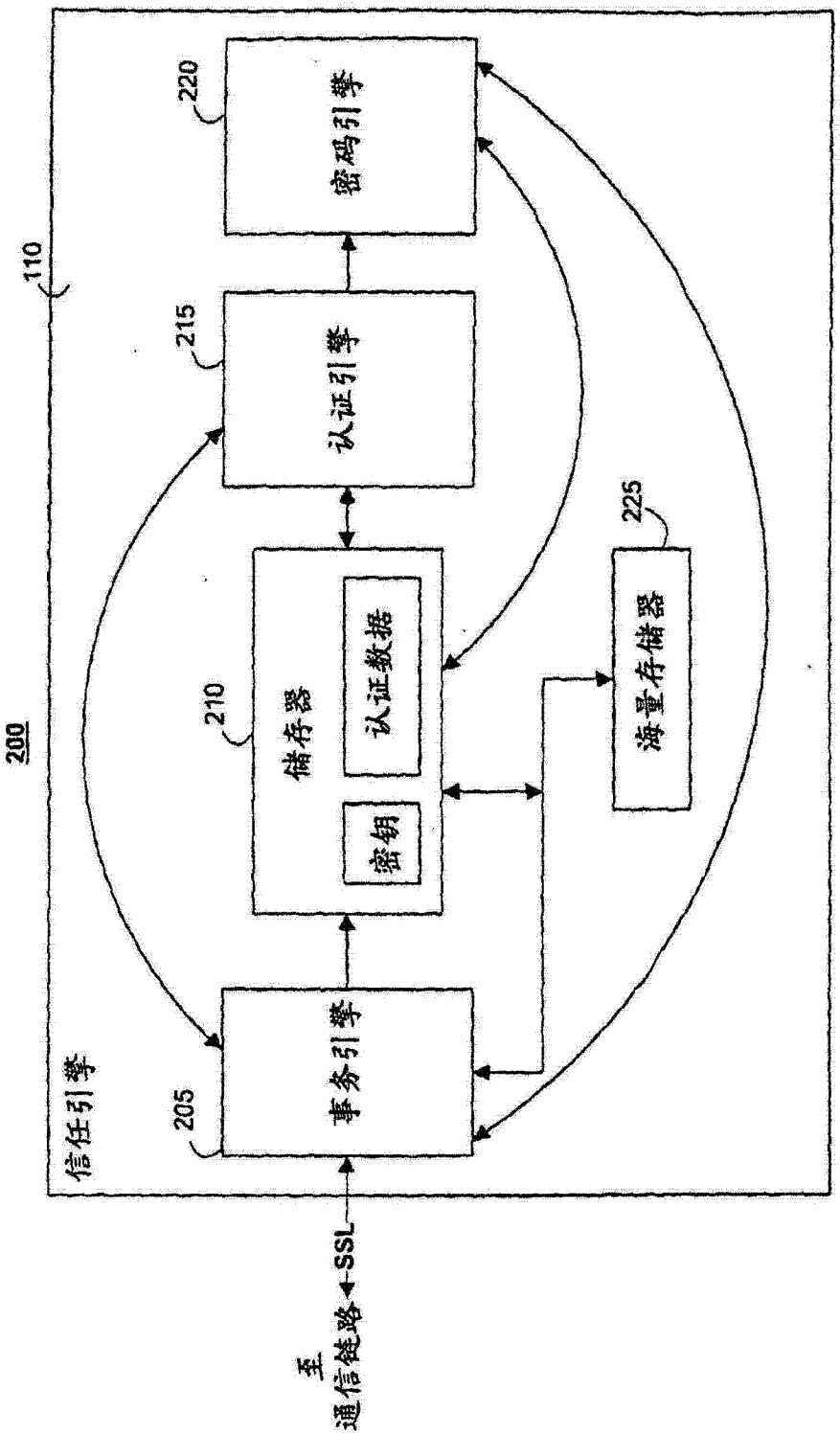

200
信任引擎
110
事务引擎
205
储存器
210
密钥
认证数据
认证引擎
215
密码引擎
220
海量存储器
225
SSL
至
通信链路

图 2

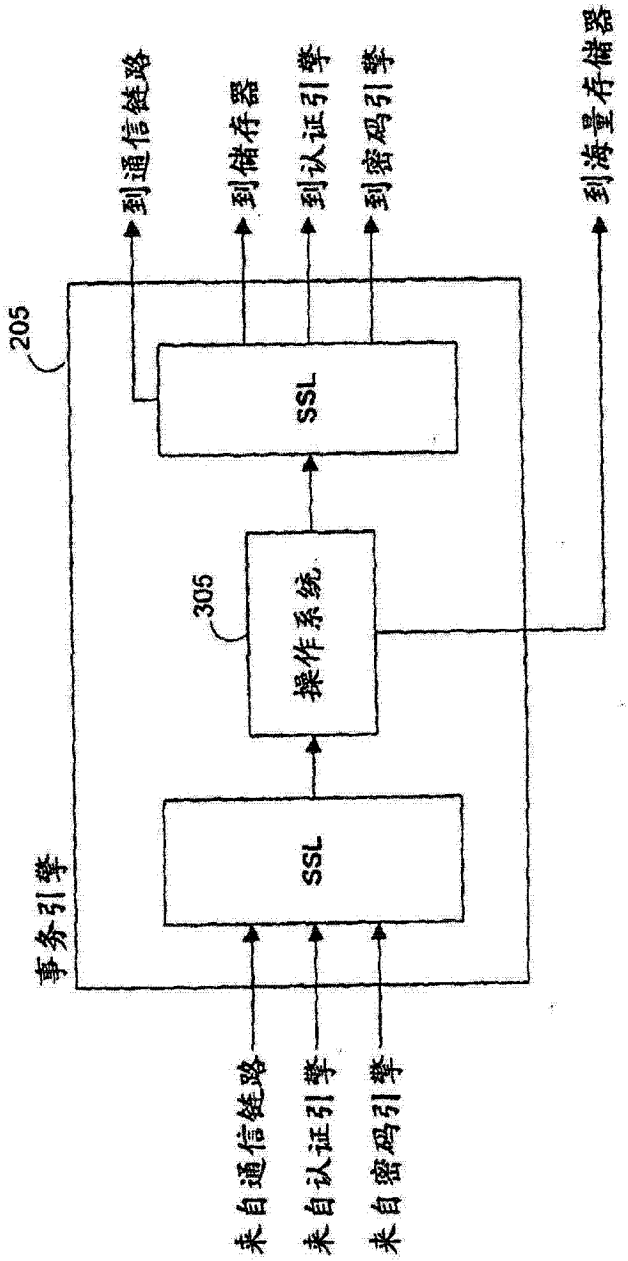

事务引擎
205
SSL
SSL
操作系统
305
来自通信链路
来自认证引擎
来自密码引擎
到通信链路
到储存器
到认证引擎
到密码引擎
到海量存储器

图 3

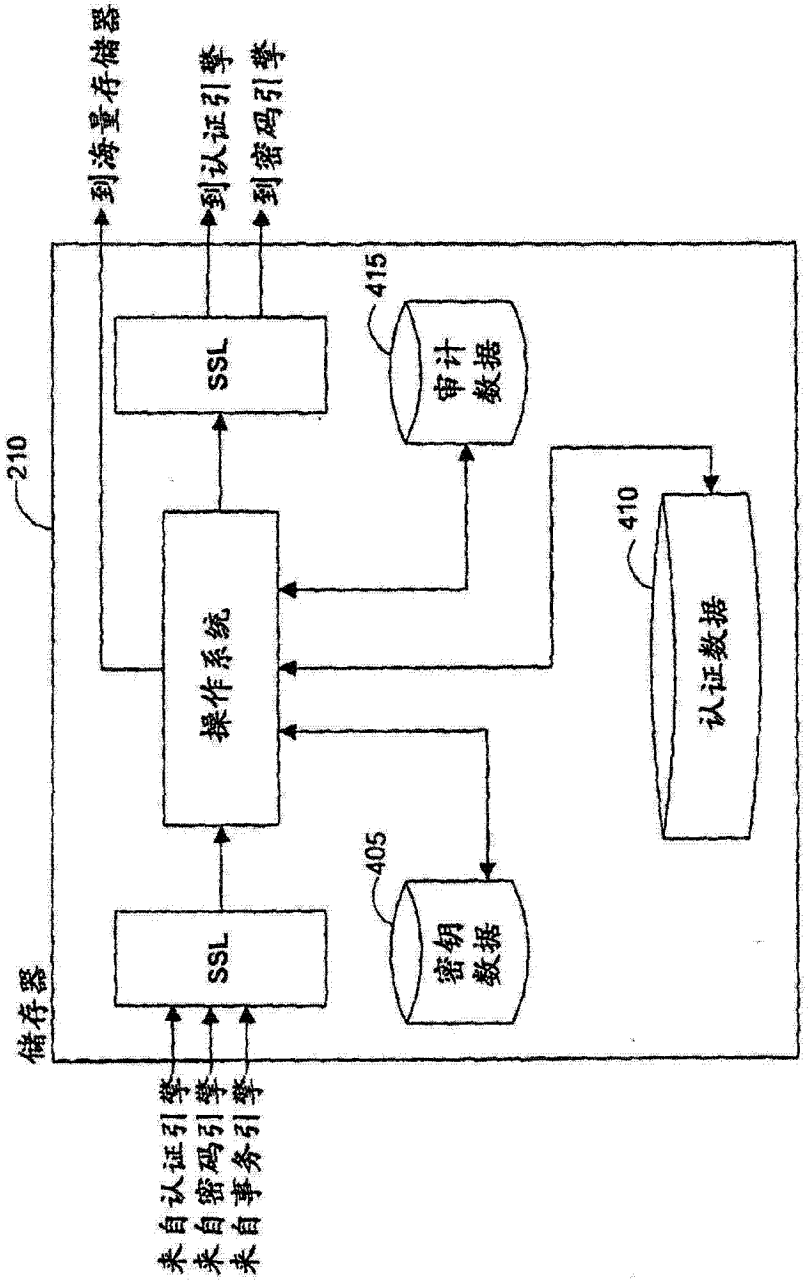

到海量存储器
到认证引擎
到密码引擎
415
SSL
审计数据
210
操作系统
410
认证数据
405
密钥数据
SSL
储存器
来自认证引擎
来自密码引擎
来自事务引擎

图 4

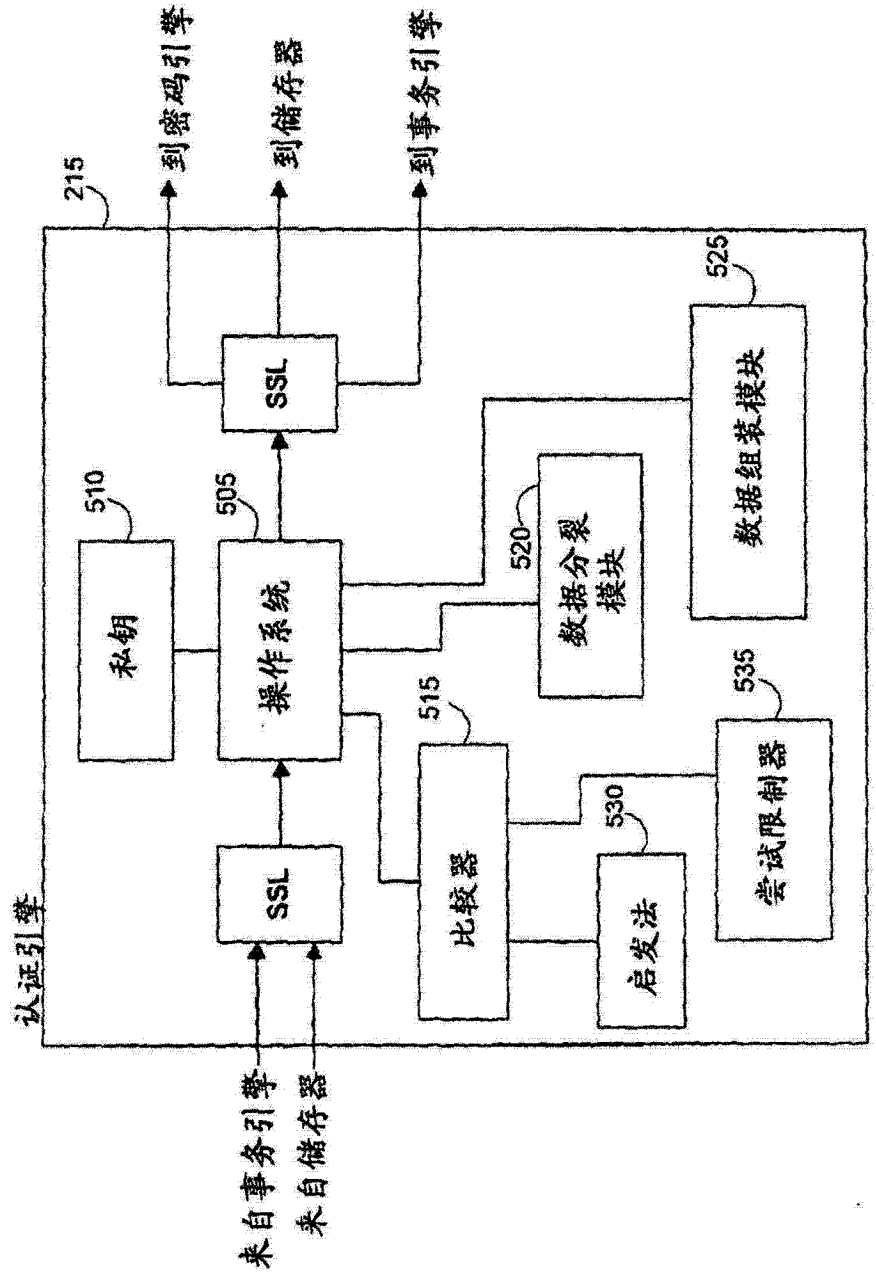

认证引擎
215
到密码引擎
到储存器
到事务引擎
来自事务引擎
来自储存器
SSL
SSL
私钥
操作系统
比较器
启发法
数据分裂模块
数据组装模块
尝试限制器
510
505
515
530
520
525
535

图 5

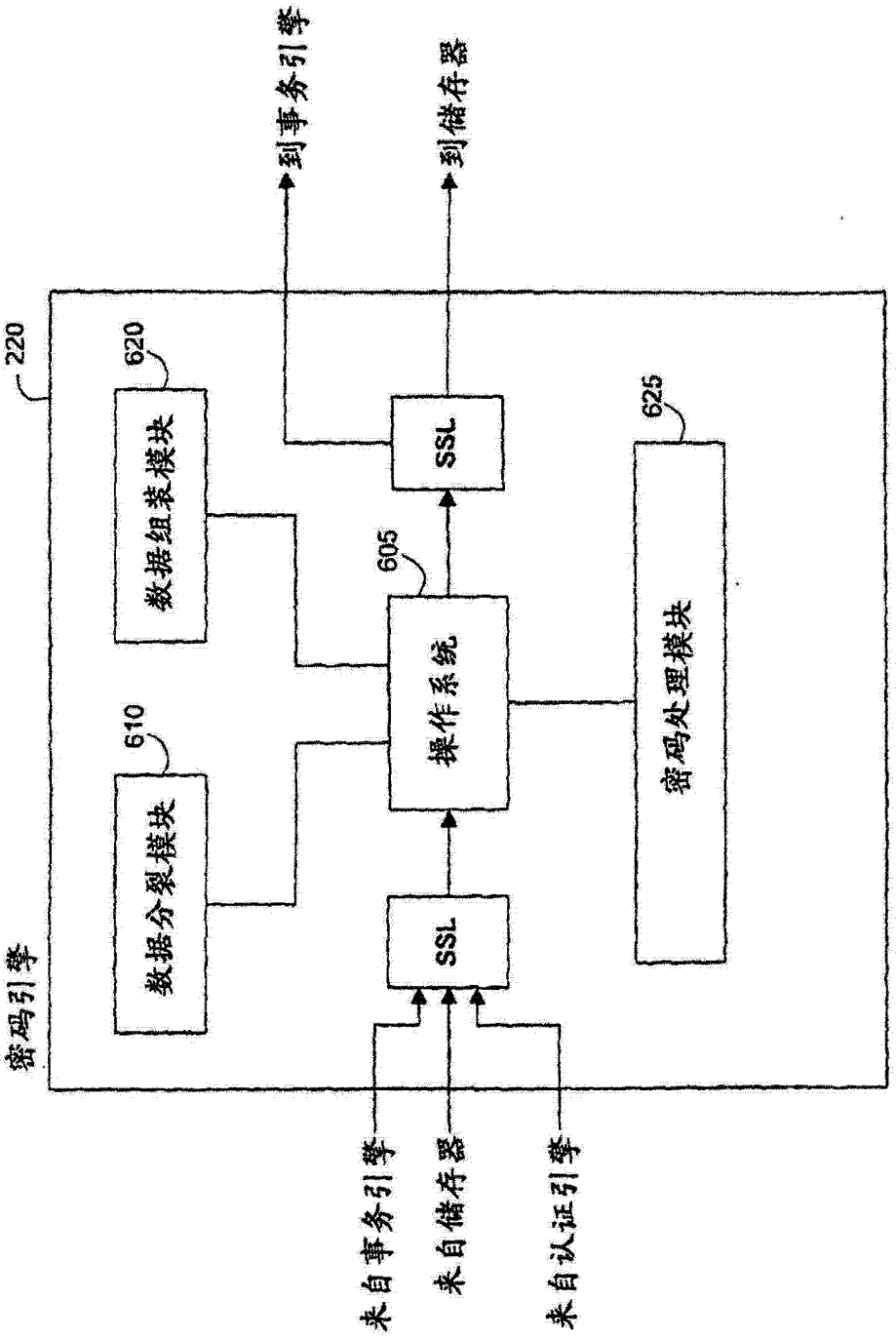

到事务引擎
到储存器
220
620
625
SSL
605
数据组装模块
操作系统
密码处理模块
610
数据分裂模块
SSL
密码引擎
来自事务引擎
来自储存器
来自认证引擎

图 6

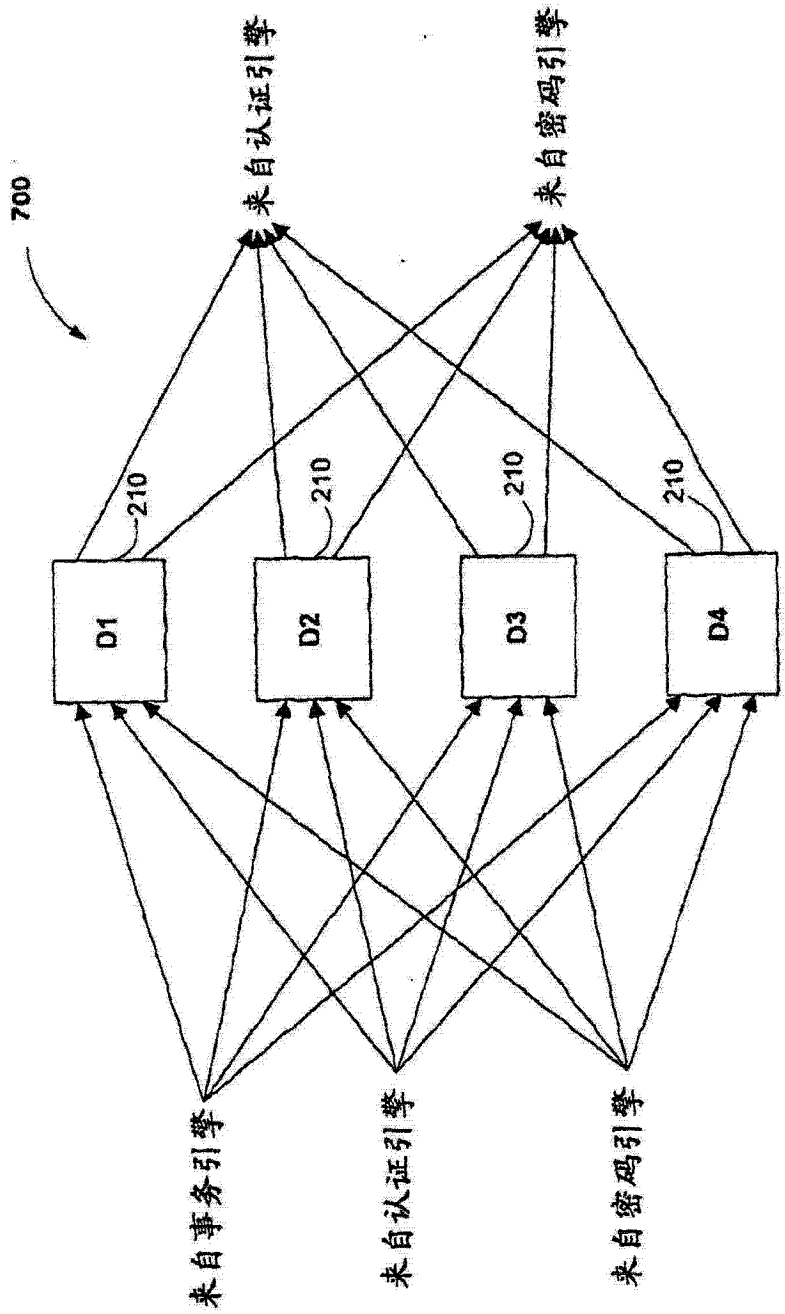

700
来自认证引擎
来自密码引擎
210
210
210
210
D1
D2
D3
D4
来自事务引擎
来自认证引擎
来自密码引擎

图 7

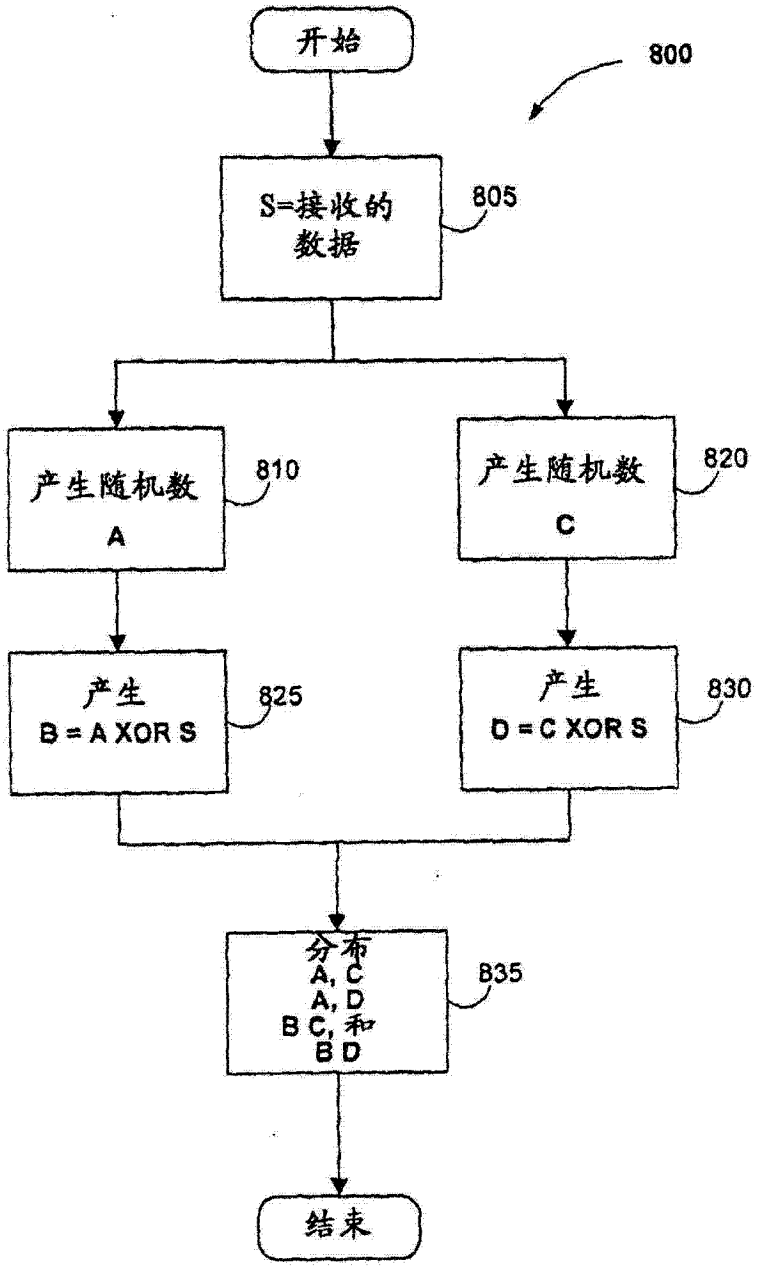

开始
800
S=接收的数据
805
产生随机数 A
810
产生随机数 C
820
产生 B = A XOR S
825
产生 D = C XOR S
830
分布 A, C A, D B C, 和 B D
835
结束

图 8

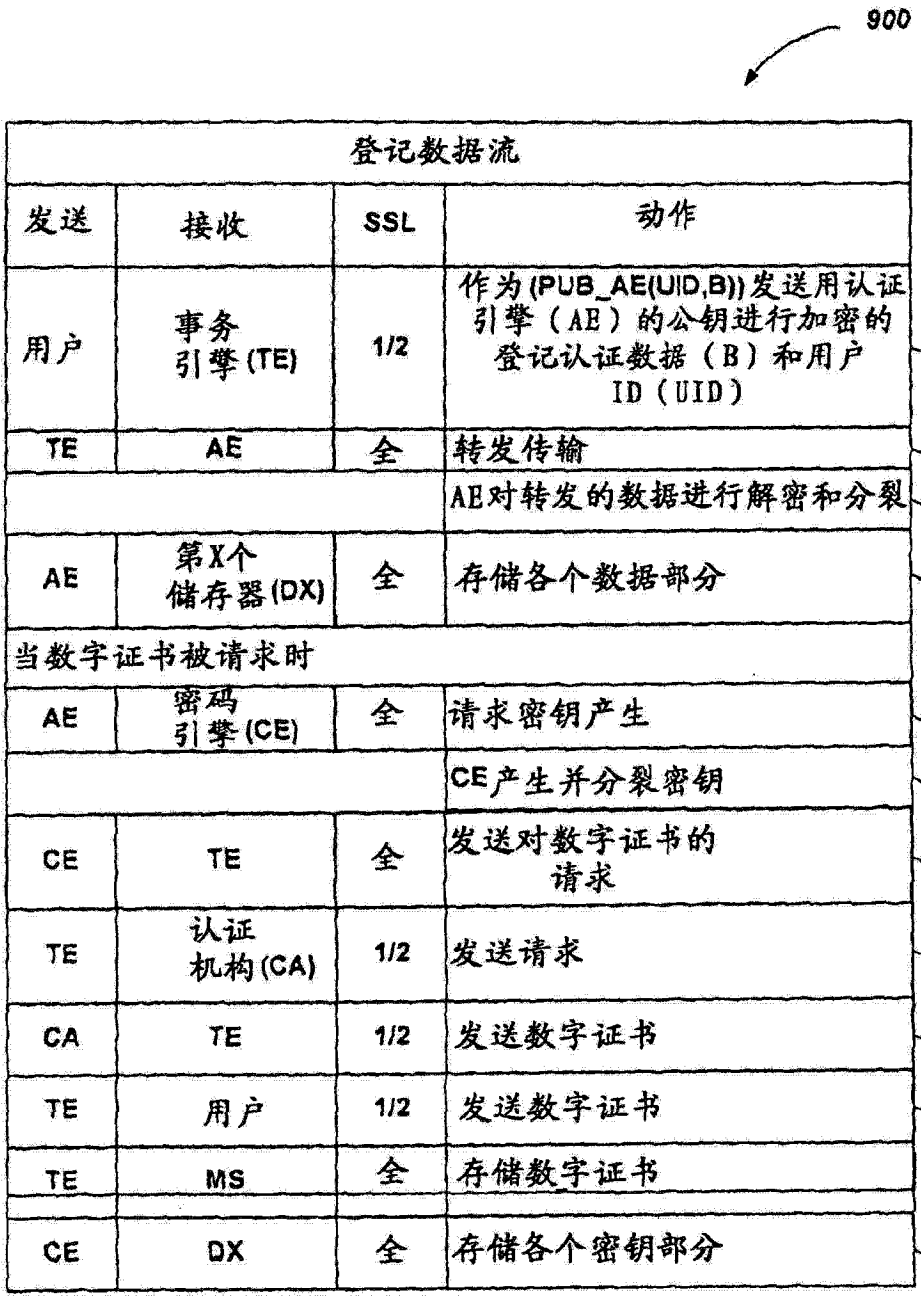

900
登记数据流
发送
接收
SSL
动作
用户
事务引擎(TE)
1/2
作为(PUB_AE(UID,B))发送用认证引擎（AE）的公钥进行加密的登记认证数据（B）和用户ID（UID）
905
TE
AE
全
转发传输
915
AE对转发的数据进行解密和分裂
920
AE
第X个储存器(DX)
全
存储各个数据部分
925
当数字证书被请求时
AE
密码引擎(CE)
全
请求密钥产生
930
CE产生并分裂密钥
935
CE
TE
全
发送对数字证书的请求
945
TE
认证机构(CA)
1/2
发送请求
950
CA
TE
1/2
发送数字证书
955
TE
用户
1/2
发送数字证书
960
TE
MS
全
存储数字证书
CE
DX
全
存储各个密钥部分
965

图 9A

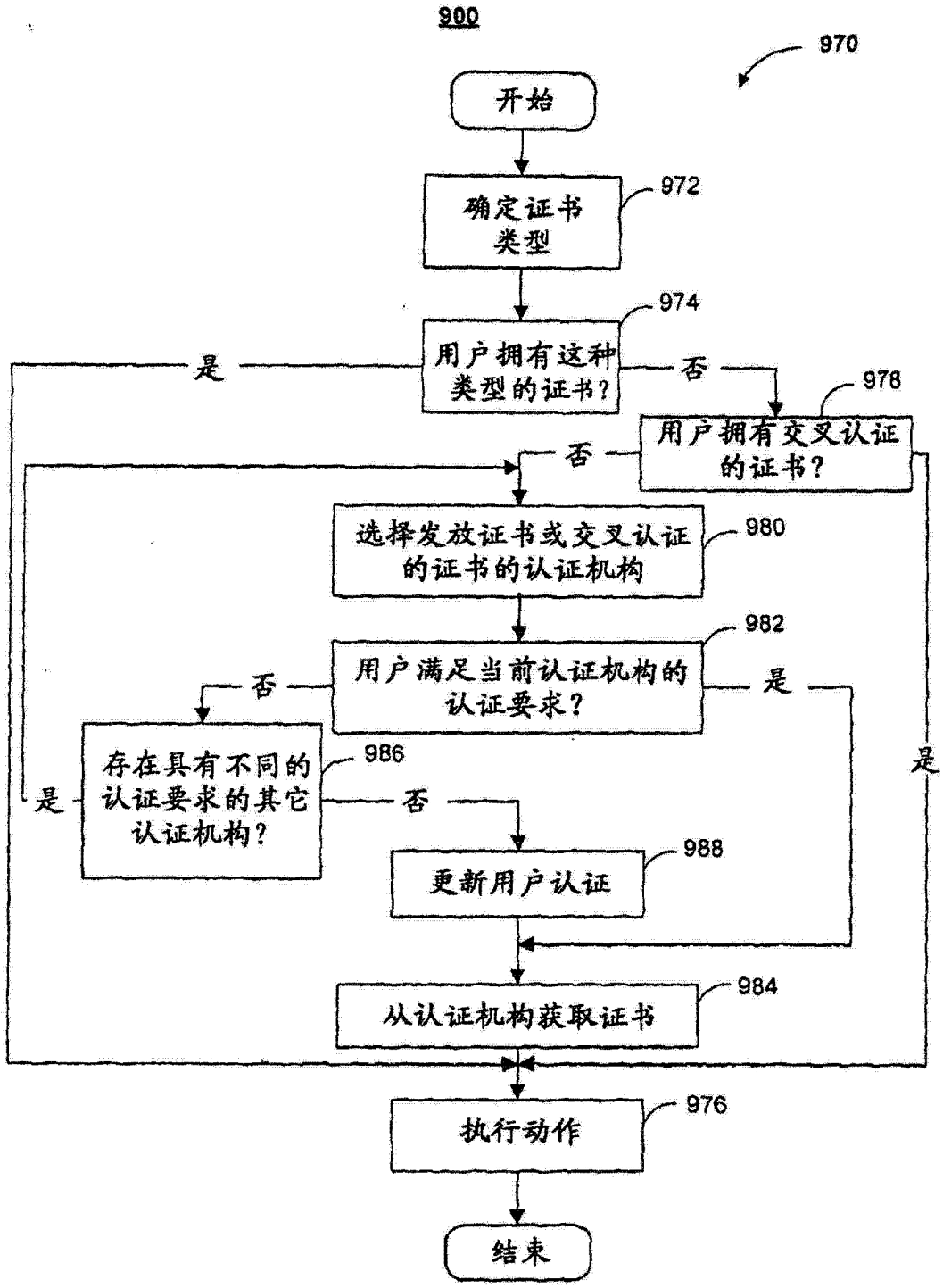

900
970
开始
确定证书类型
972
用户拥有这种类型的证书？
974
是
否
用户拥有交叉认证的证书？
978
否
选择发放证书或交叉认证的证书的认证机构
980
用户满足当前认证机构的认证要求？
982
否
是
存在具有不同的认证要求的其它认证机构？
986
是
否
更新用户认证
988
从认证机构获取证书
984
是
执行动作
976
结束

图 9B

1000

	认证数据流			
	发送	接收	SSL	动作
1005	用户	卖方	1/2	发生事务，例如选择购买
1010	卖方	用户	1/2	发送事务ID（TID）和认证请求（AR）
				从用户收集认证数据(B′)
1015	用户	TE	1/2	作为(PUB_AE(TID, B′))发送在认证引擎（AE）的公钥中包装的TID和B′
1020	TE	AE	全	转发传输
				请求并收集登记认证数据（B）
1025	卖方	事务引擎(TE)	全	发送 TID, AR
1030	TE	海量存储器(MS)	全	在数据库中建立记录
1035	TE	第X个储存器(DX)	全	UID, TID
1040	DX	AE	全	作为(PUB_AE(TID, BX))发送TID和登记时存储的认证数据的部分（BX）
1045				AE组装B并且与B′进行比较
1050	AE	TE	全	TID, 被填充的AR
1055	TE	卖方	全	TID, 是/否
	TE	用户	1/2	TID, 确认消息

图 10

1100

签名数据流			
发送	接收	SSL	动作
用户	卖方	1/2	发生事务，例如同意交易
卖方	用户	1/2	发送事务识别号(TID)、认证请求（AR）和同意消息（M）
			从用户收集当前认证数据(B′)和由用户接收的消息的哈希值(h(M′))
用户	TE	1/2	作为(PUB_AE(TID, B′, h(M′))，发送包装在认证引擎（AE）的公钥中的TID、B′,AR和h(M′))
TE	AE	全	转发传输
			收集登记认证数据
卖方	事务引擎(TE)	全	发送UID、TID、AR和消息的哈希值(h(M′))
TE	海量存储器(MS)	全	在数据库中建立记录
TE	第X个储存器(DX)	全	UID, TID
DX	AE	全	作为(PUB_AE(TID, BX))，发送TID和登记时存储的认证数据的部分(BX)
			原始卖方消息发送至AE
TE	AE	全	发送h(M)
			AE组装B，与B′进行比较并且比较h(M)和h(M′)
AE	密码引擎(CE)	全	请求数字签名和要签名的消息，例如哈希处理的消息
AE	DX	全	TID，签名UID
DX	CE	全	发送与签名方对应的密码密钥的部分
			CE组装密钥和签名
CE	AE	全	发送签名方的数字签名（S）
AE	TE	全	TID，被填充的AR、h(M)和S
TE	卖方	全	TID，receipt=(TID,是/否,S)，信任引擎的数字签名，例如用信任引擎的私钥加密的收据的哈希值({Priv_TE(h(receipt)))
TE	用户	1/2	TID，确认消息

1103 1105 1110 1115 1120 1125 1130 1135 1140

图 11

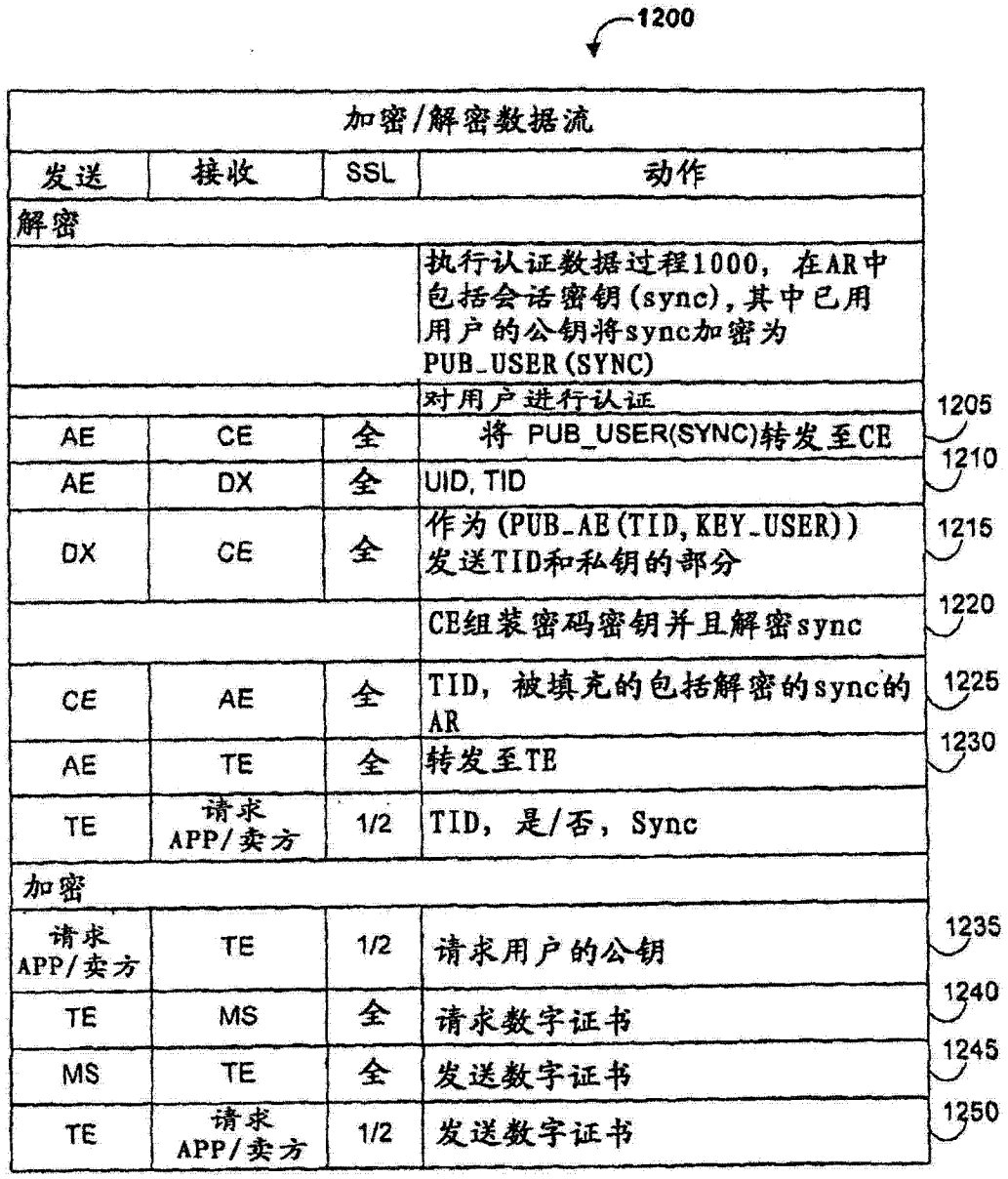

1200
加密/解密数据流
发送
接收
SSL
动作
解密
执行认证数据过程1000，在AR中包括会话密钥(sync)，其中已用用户的公钥将sync加密为PUB_USER(SYNC)
对用户进行认证
1205
AE
CE
全
将 PUB_USER(SYNC)转发至CE
1210
AE
DX
全
UID, TID
1215
DX
CE
全
作为(PUB_AE(TID, KEY_USER))发送TID和私钥的部分
1220
CE组装密码密钥并且解密sync
1225
CE
AE
全
TID，被填充的包括解密的sync的AR
1230
AE
TE
全
转发至TE
TE
请求APP/卖方
1/2
TID，是/否，Sync
加密
1235
请求APP/卖方
TE
1/2
请求用户的公钥
1240
TE
MS
全
请求数字证书
1245
MS
TE
全
发送数字证书
1250
TE
请求APP/卖方
1/2
发送数字证书

图 12

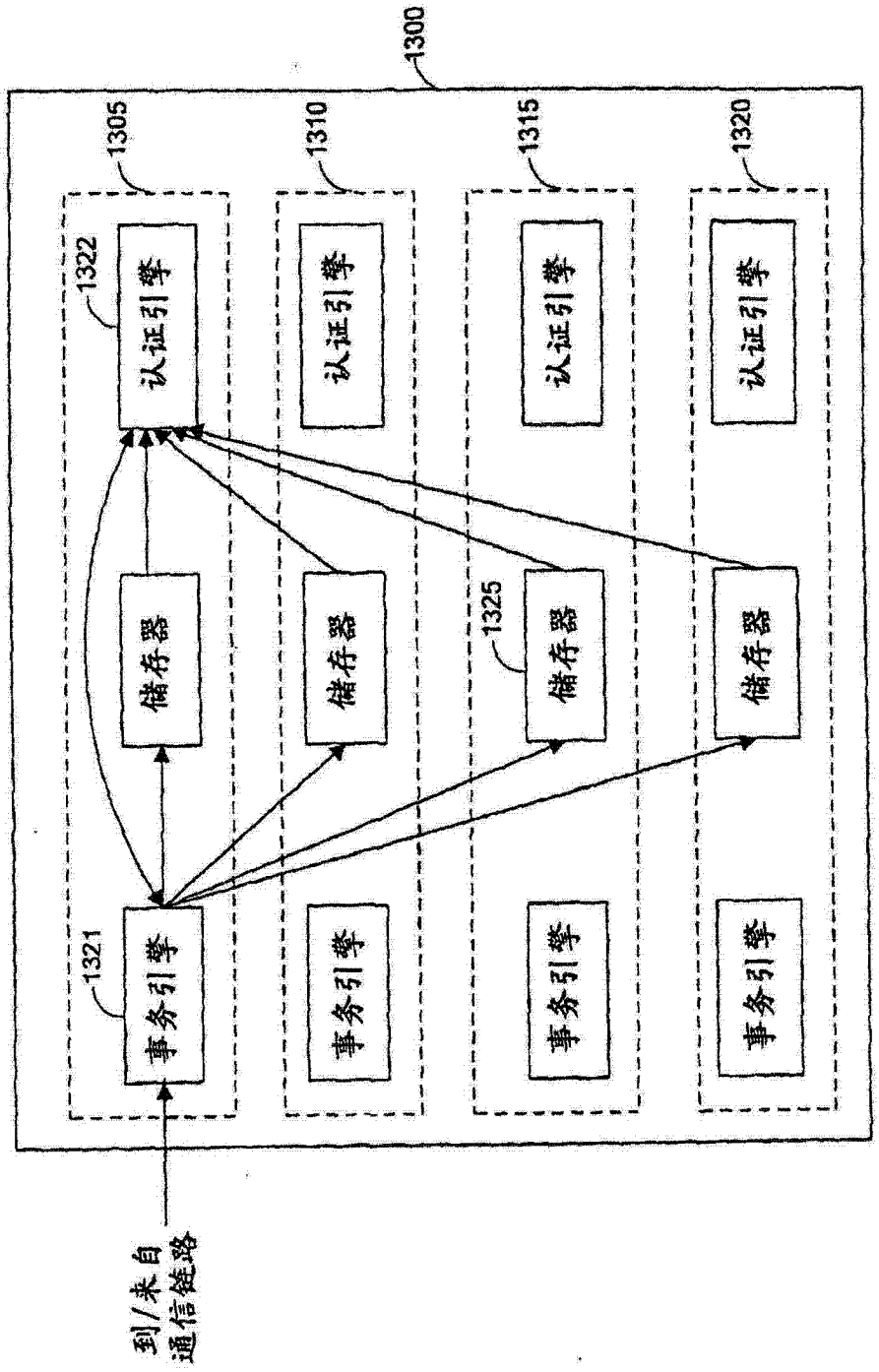

1300
1305
1310
1315
1320
1322
认证引擎
储存器
1325
事务引擎
1321
到/来自
通信链路

图 13

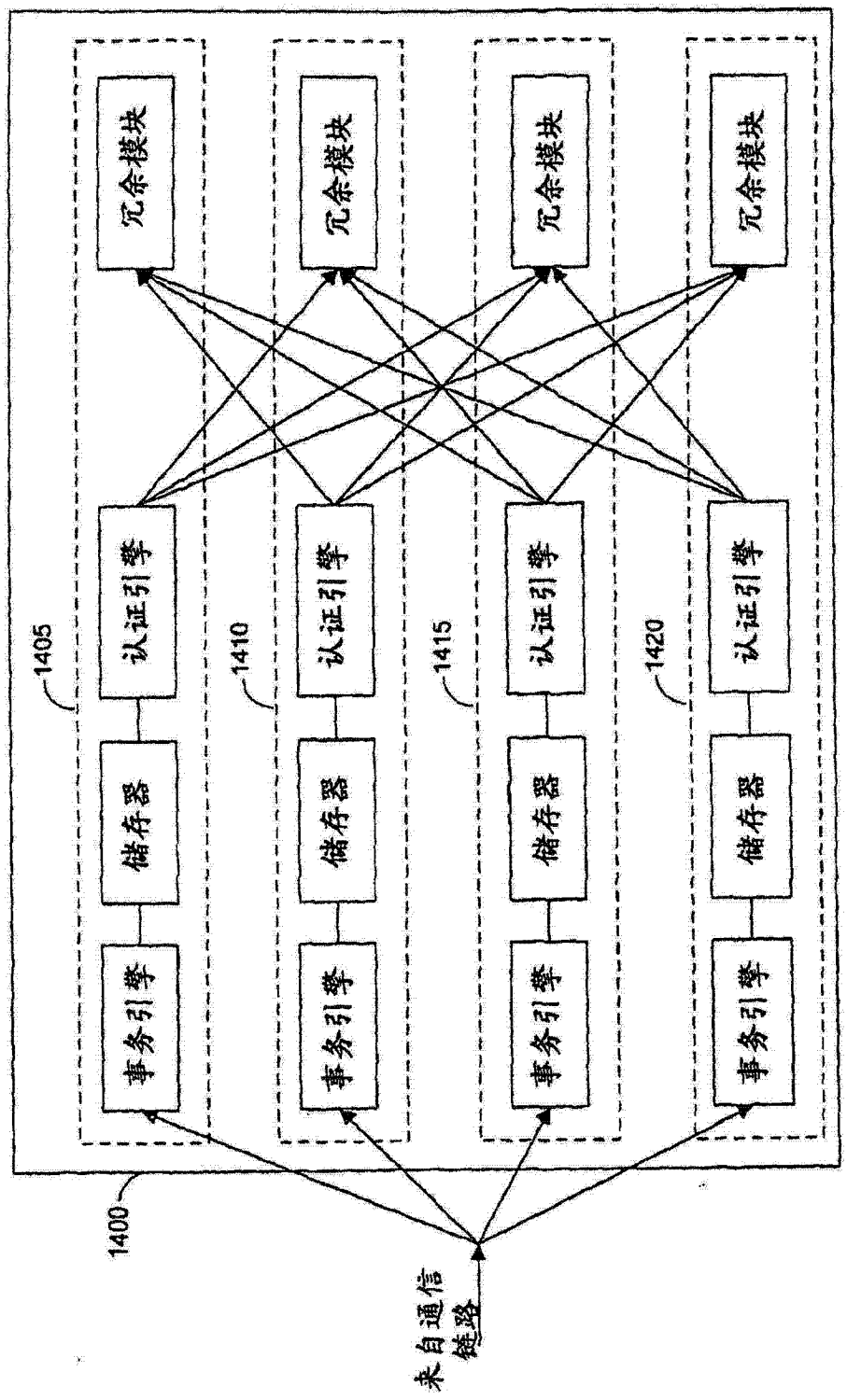

冗余模块
冗余模块
冗余模块
冗余模块
认证引擎
认证引擎
认证引擎
认证引擎
储存器
储存器
储存器
储存器
事务引擎
事务引擎
事务引擎
事务引擎
1405
1410
1415
1420
1400
来自通信链路

图 14

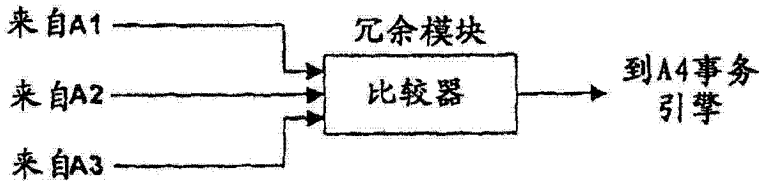


图 15

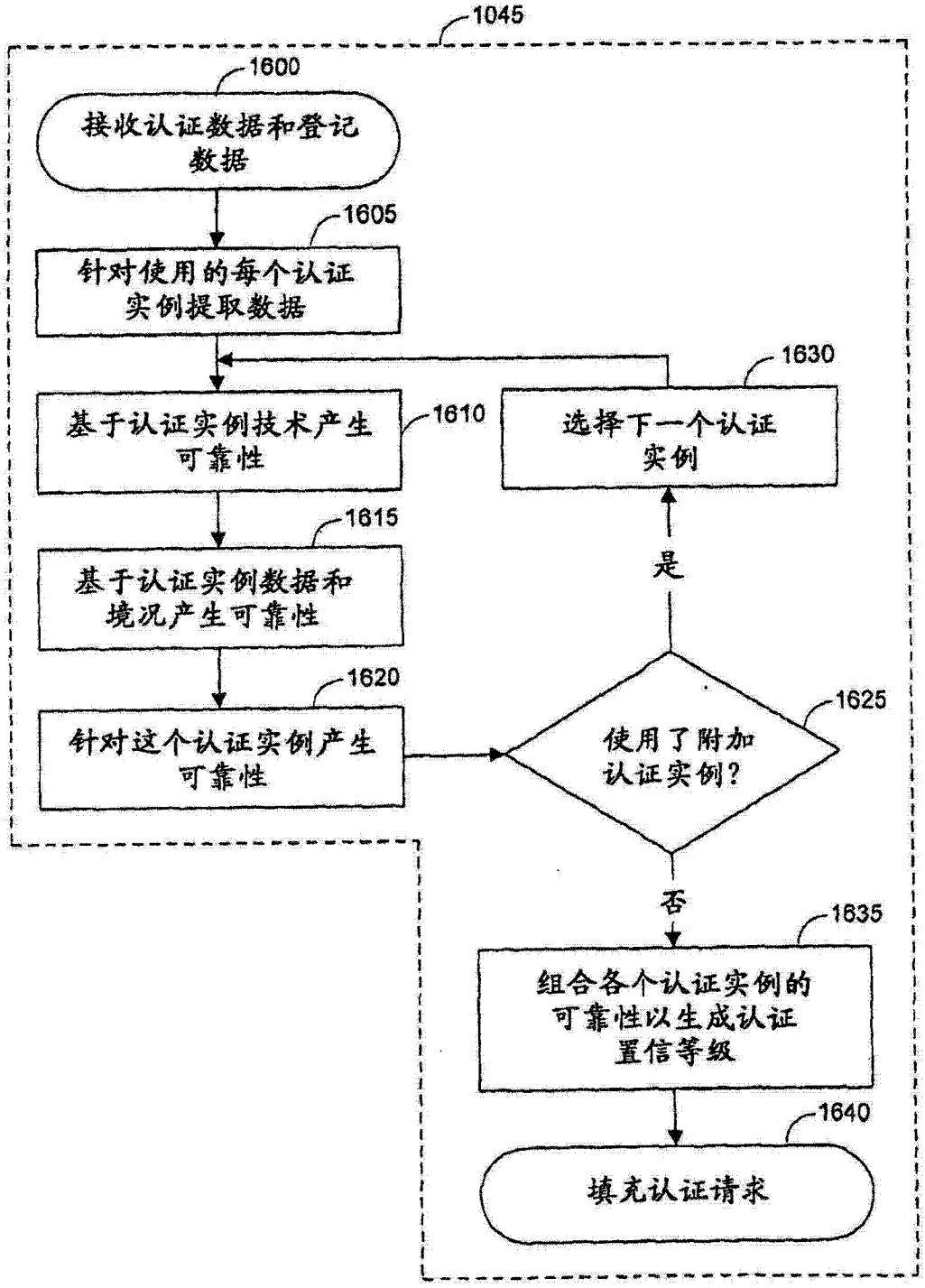

1045
1600
接收认证数据和登记数据
1605
针对使用的每个认证实例提取数据
1630
选择下一个认证实例
1610
基于认证实例技术产生可靠性
1615
基于认证实例数据和境况产生可靠性
1620
针对这个认证实例产生可靠性
是
1625
使用了附加认证实例?
否
1635
组合各个认证实例的可靠性以生成认证置信等级
1640
填充认证请求

图 16

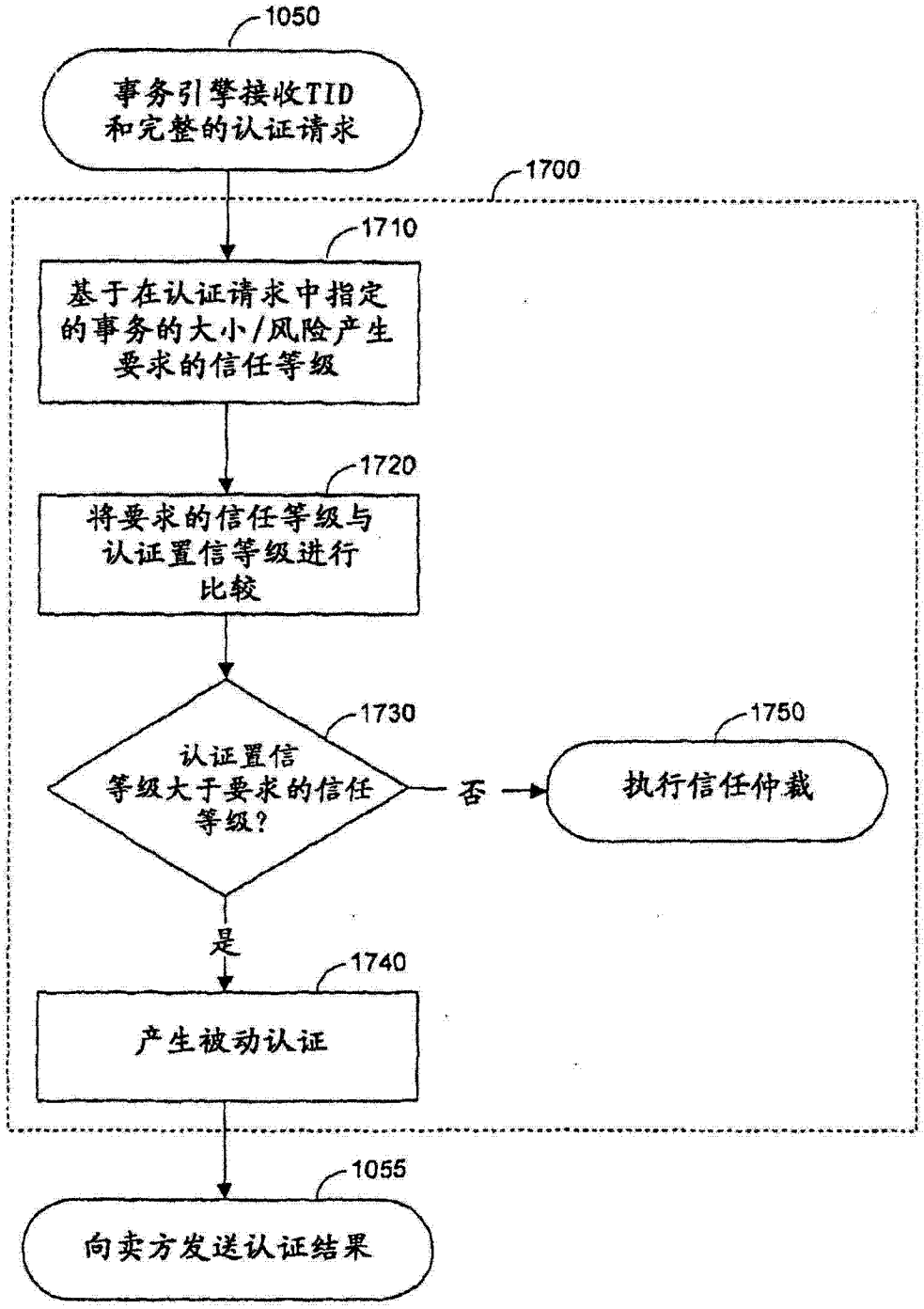

1050
事务引擎接收TID
和完整的认证请求
1700
1710
基于在认证请求中指定
的事务的大小/风险产生
要求的信任等级
1720
将要求的信任等级与
认证置信等级进行
比较
1730
认证置信
等级大于要求的信任
等级?
否
1750
执行信任仲裁
是
1740
产生被动认证
1055
向卖方发送认证结果

图 17

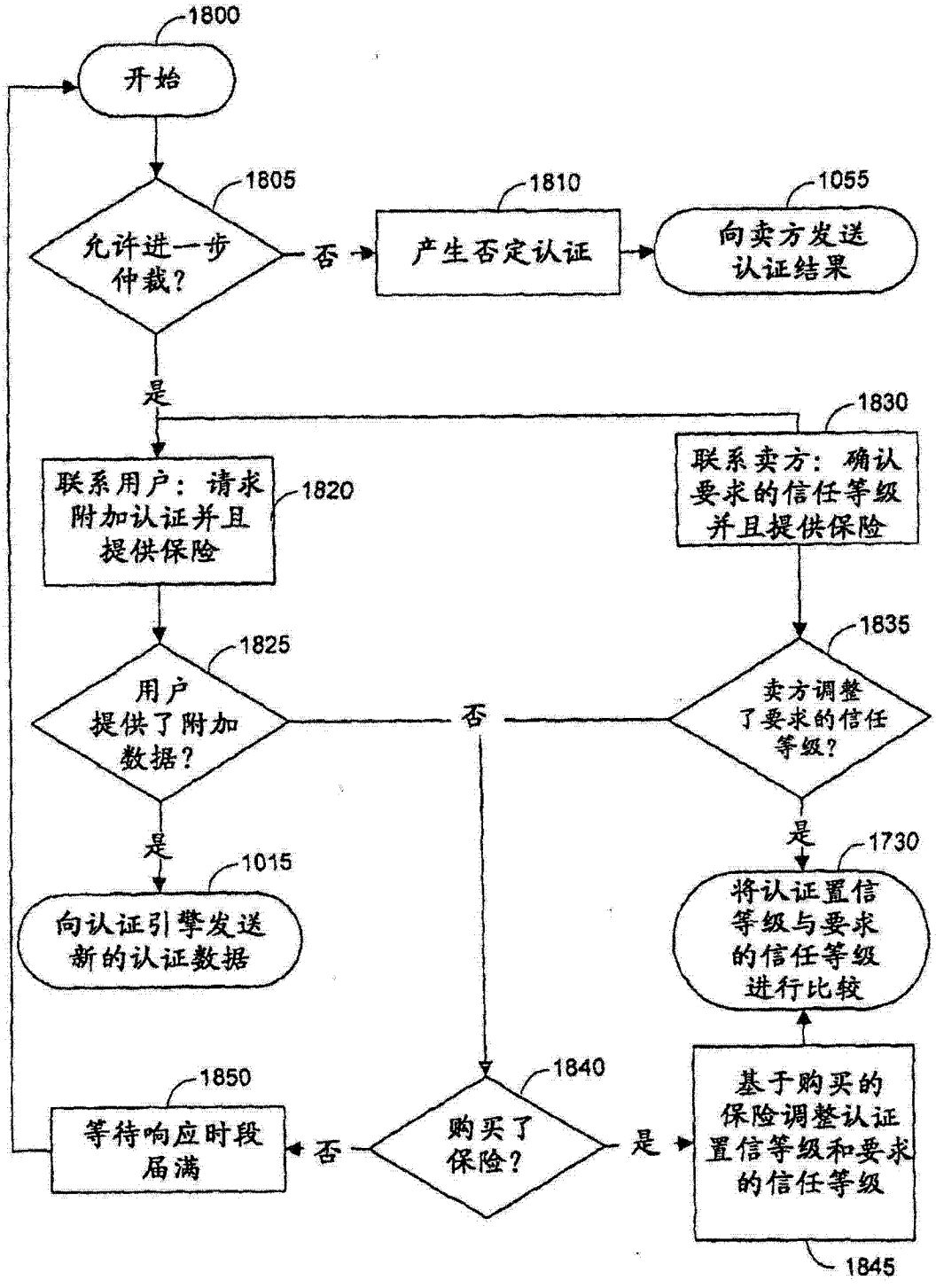

1800
开始
1805
允许进一步仲裁?
否
1810
产生否定认证
1055
向卖方发送认证结果
是
1830
联系卖方: 确认要求的信任等级并且提供保险
联系用户: 请求附加认证并且提供保险
1820
1835
卖方调整了要求的信任等级?
1825
用户提供了附加数据?
否
是
1730
将认证置信等级与要求的信任等级进行比较
1015
向认证引擎发送新的认证数据
1840
购买了保险?
否
是
基于购买的保险调整认证置信等级和要求的信任等级
1845
1850
等待响应时段届满

图 18

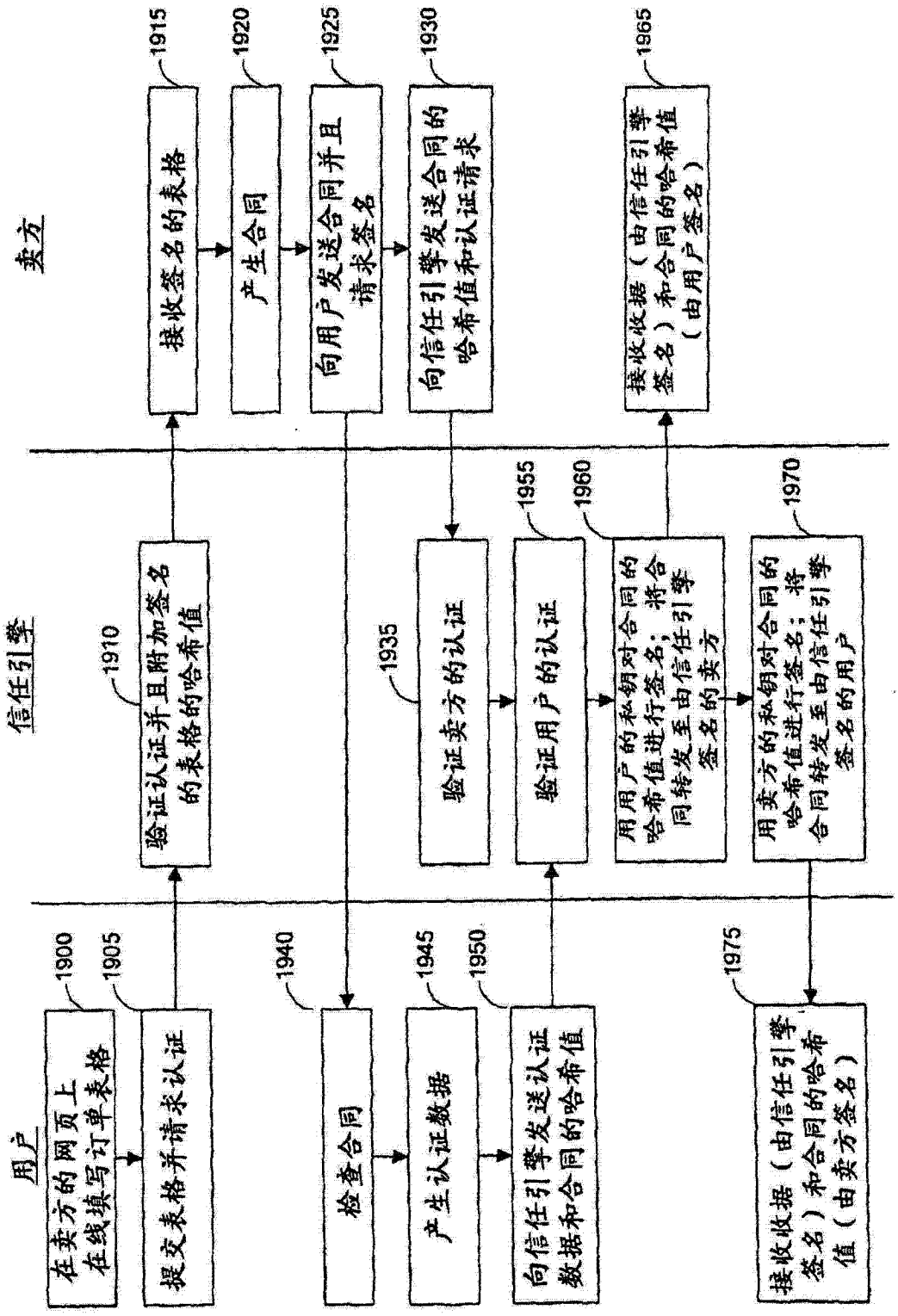

卖方
信任引擎
用户
1915
接收签名的表格
1920
产生合同
1925
向用户发送合同并且请求签名
1930
向信任引擎发送合同的哈希值和认证请求
1965
接收收据（由信任引擎签名）和合同的哈希值（由用户签名）
1910
验证认证并且附加签名的表格的哈希值
1935
验证卖方的认证
1955
验证用户的认证
1960
用用户的私钥对合同的哈希值进行签名；将合同转发至由信任引擎签名的卖方
1970
用卖方的私钥对合同的哈希值进行签名；将合同转发至由信任引擎签名的用户
1900
在卖方的网页上在线填写订单表格
1905
提交表格并请求认证
1940
检查合同
1945
产生认证数据
1950
向信任引擎发送认证数据和合同的哈希值
1975
接收收据（由信任引擎签名）和合同的哈希值（由卖方签名）

图 19

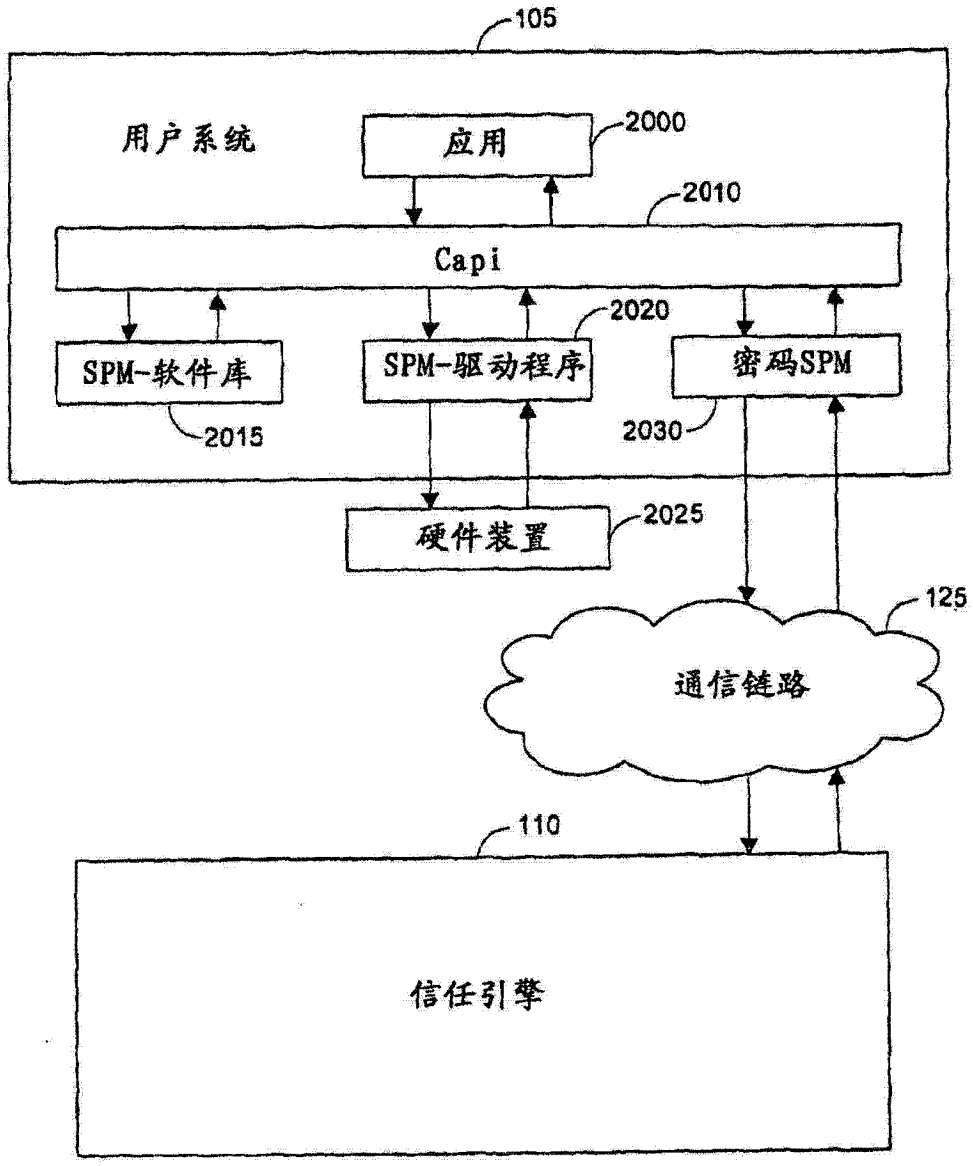

105
用户系统
应用
2000
2010
Capi
2020
SPM-软件库
SPM-驱动程序
密码SPM
2015
2030
硬件装置
2025
125
通信链路
110
信任引擎

图 20

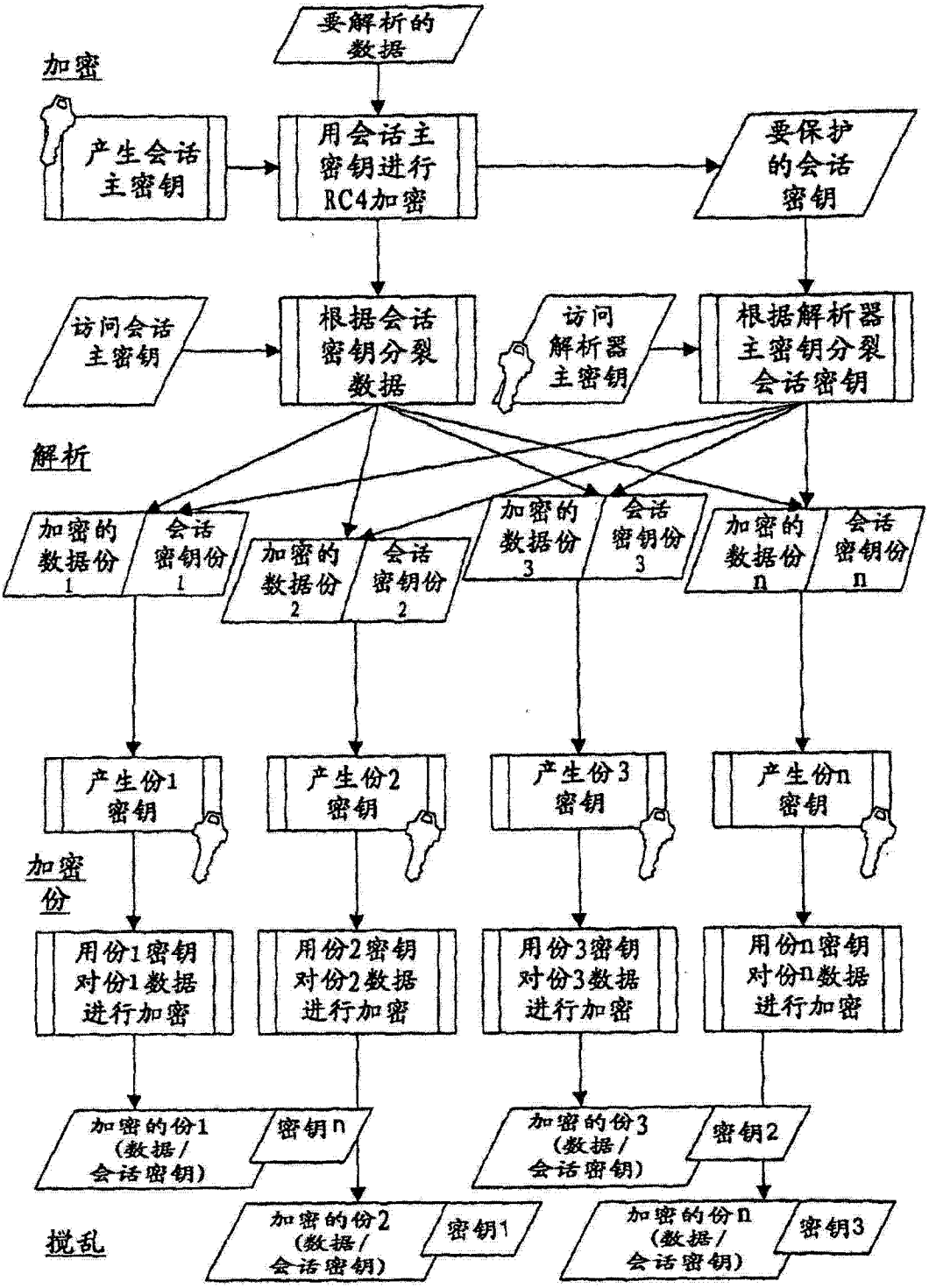

要解析的数据
加密
产生会话主密钥
用会话主密钥进行RC4加密
要保护的会话密钥
访问会话主密钥
根据会话密钥分裂数据
访问解析器主密钥
根据解析器主密钥分裂会话密钥
解析
加密的数据份1
会话密钥份1
加密的数据份2
会话密钥份2
加密的数据份3
会话密钥份3
加密的数据份n
会话密钥份n
产生份1密钥
产生份2密钥
产生份3密钥
产生份n密钥
加密份
用份1密钥对份1数据进行加密
用份2密钥对份2数据进行加密
用份3密钥对份3数据进行加密
用份n密钥对份n数据进行加密
加密的份1(数据/会话密钥)
密钥n
加密的份3(数据/会话密钥)
密钥2
加密的份2(数据/会话密钥)
密钥1
加密的份n(数据/会话密钥)
密钥3
搅乱

图 21

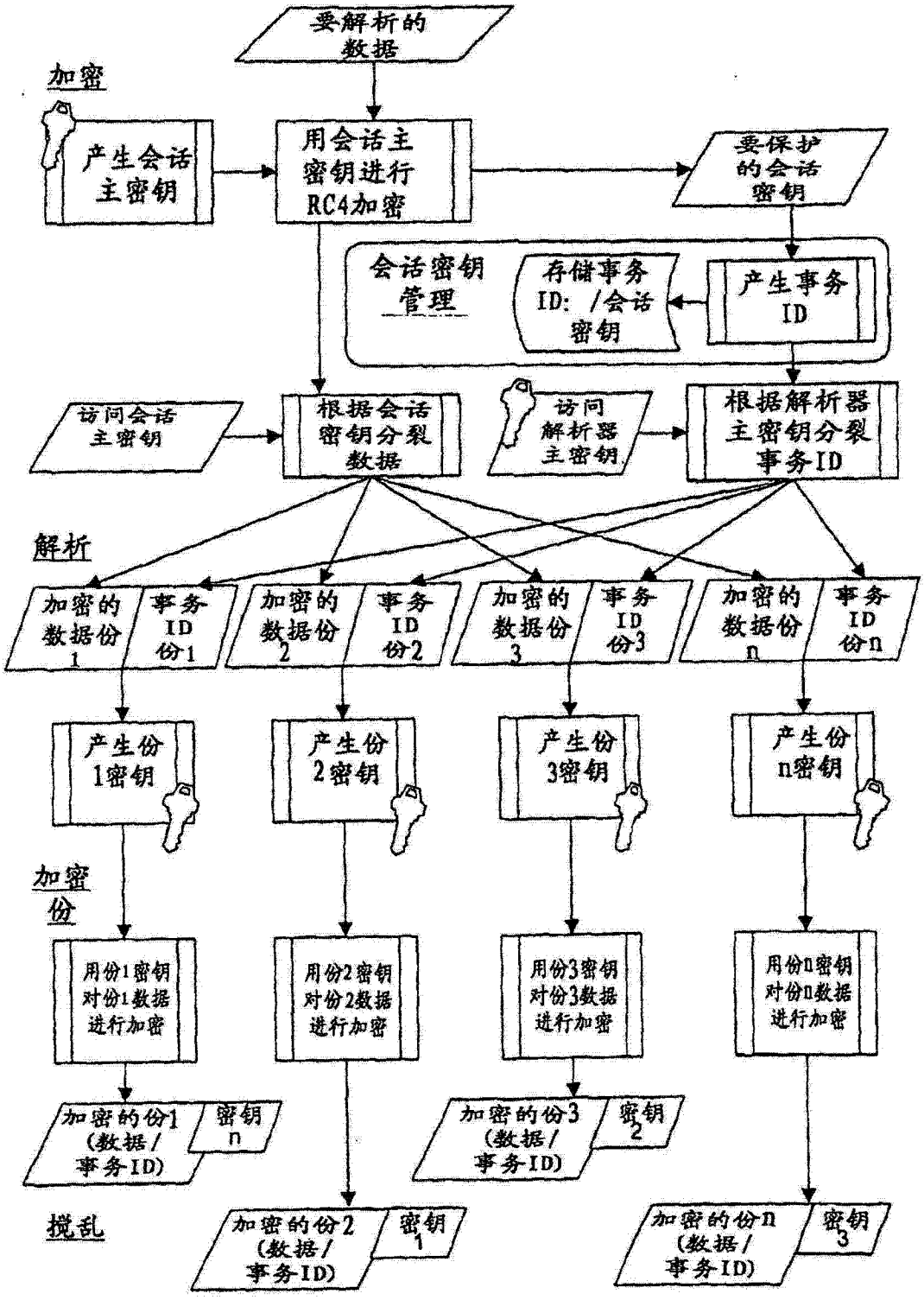

要解析的数据
加密
产生会话主密钥
用会话主密钥进行RC4加密
要保护的会话密钥
会话密钥管理
存储事务ID: /会话密钥
产生事务ID
访问会话主密钥
根据会话密钥分裂数据
访问解析器主密钥
根据解析器主密钥分裂事务ID
解析
加密的数据份1
事务ID份1
加密的数据份2
事务ID份2
加密的数据份3
事务ID份3
加密的数据份n
事务ID份n
产生份1密钥
产生份2密钥
产生份3密钥
产生份n密钥
加密份
用份1密钥对份1数据进行加密
用份2密钥对份2数据进行加密
用份3密钥对份3数据进行加密
用份n密钥对份n数据进行加密
加密的份1(数据/事务ID)
密钥n
加密的份3(数据/事务ID)
密钥2
搅乱
加密的份2(数据/事务ID)
密钥1
加密的份n(数据/事务ID)
密钥3

图 22

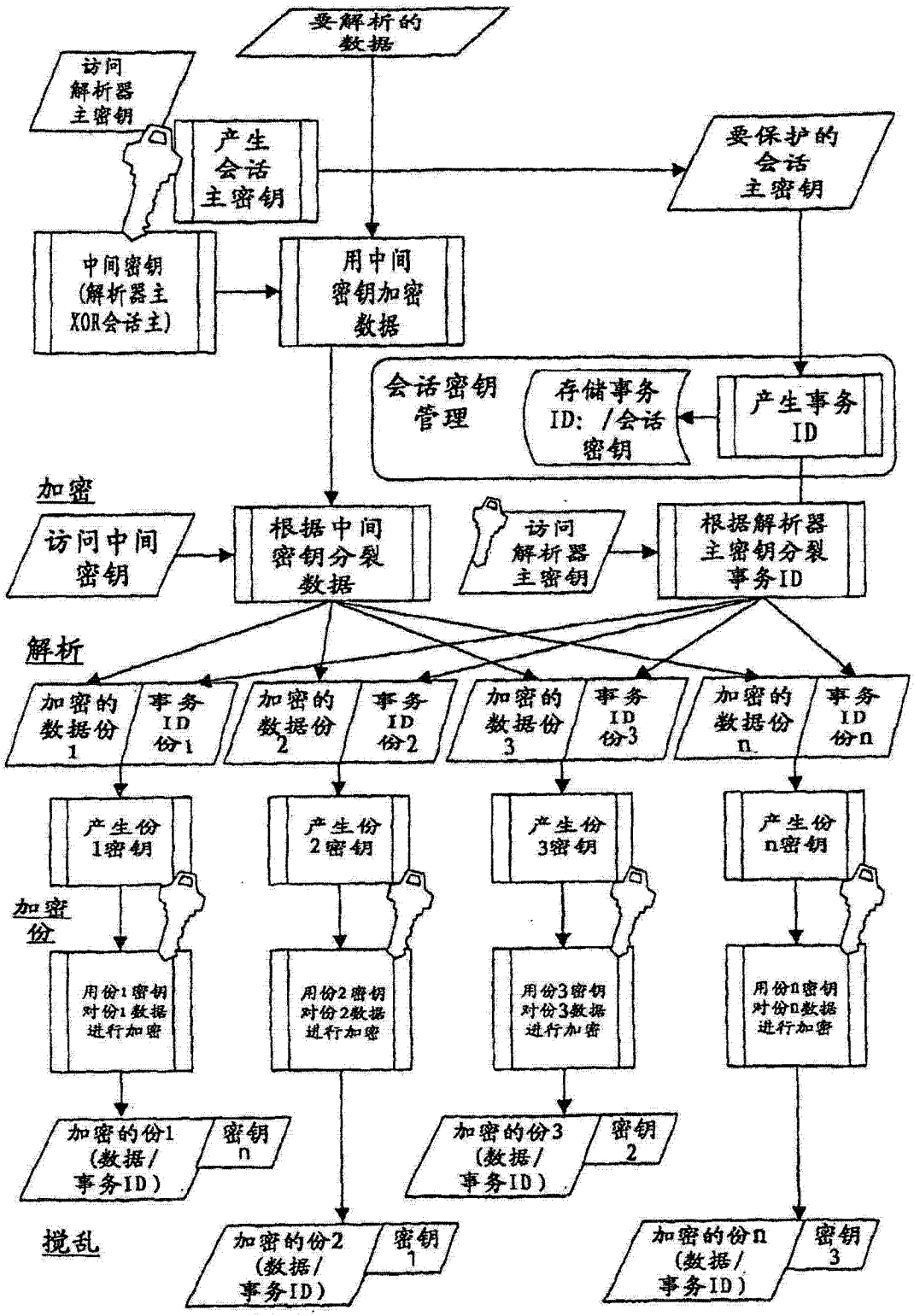

要解析的数据
访问解析器主密钥
产生会话主密钥
要保护的会话主密钥
中间密钥(解析器主XOR会话主)
用中间密钥加密数据
会话密钥管理
存储事务ID: /会话密钥
产生事务ID
加密
访问中间密钥
根据中间密钥分裂数据
访问解析器主密钥
根据解析器主密钥分裂事务ID
解析
加密的数据份1
事务ID份1
加密的数据份2
事务ID份2
加密的数据份3
事务ID份3
加密的数据份n
事务ID份n
产生份1密钥
产生份2密钥
产生份3密钥
产生份n密钥
加密份
用份1密钥对份1数据进行加密
用份2密钥对份2数据进行加密
用份3密钥对份3数据进行加密
用份n密钥对份n数据进行加密
加密的份1(数据/事务ID)
密钥n
加密的份3(数据/事务ID)
密钥2
搅乱
加密的份2(数据/事务ID)
密钥1
加密的份n(数据/事务ID)
密钥3

图 23

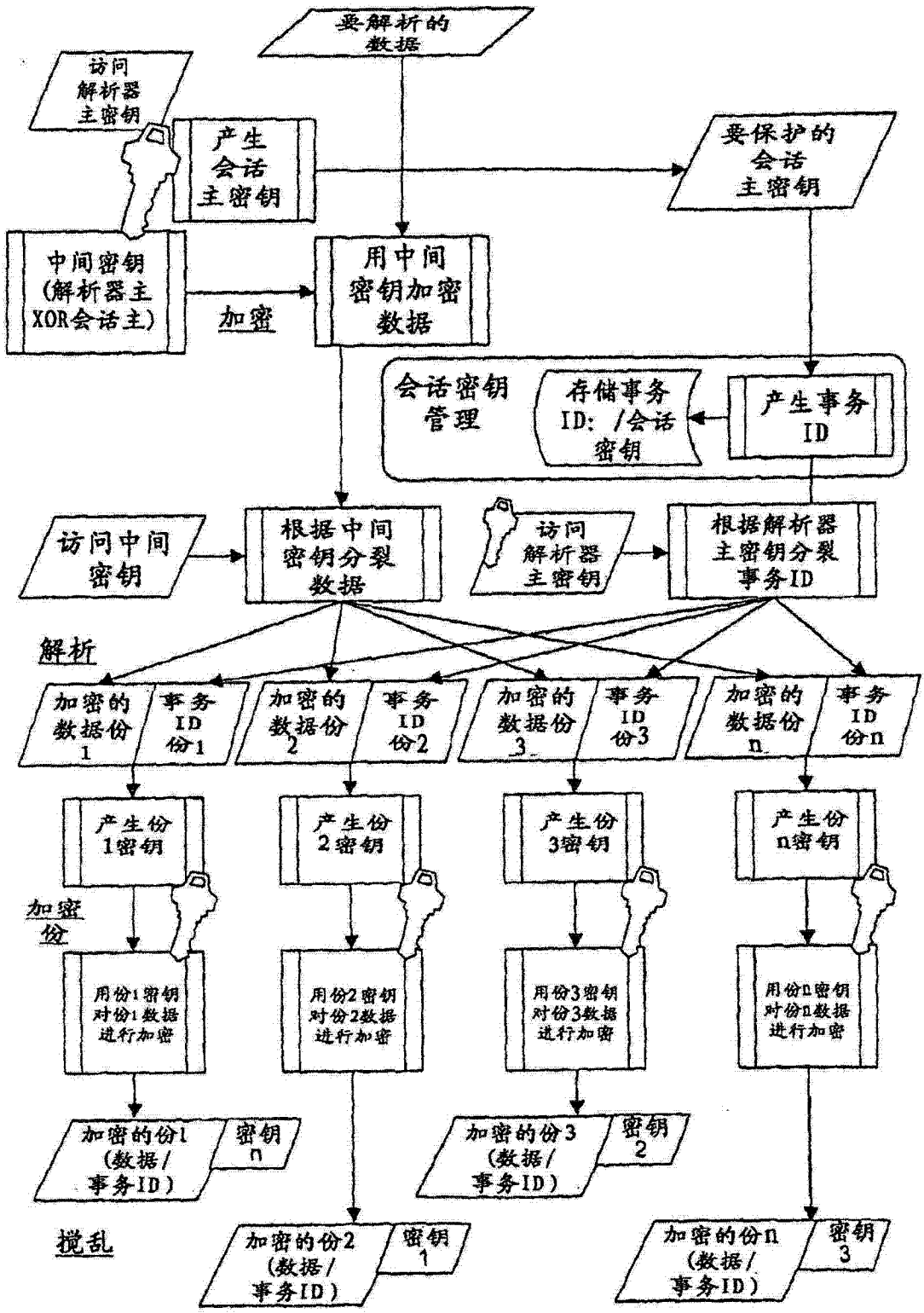

要解析的数据
访问解析器主密钥
产生会话主密钥
要保护的会话主密钥
中间密钥(解析器主XOR会话主)
加密
用中间密钥加密数据
会话密钥管理
存储事务ID: /会话密钥
产生事务ID
访问中间密钥
根据中间密钥分裂数据
访问解析器主密钥
根据解析器主密钥分裂事务ID
解析
加密的数据份1
事务ID份1
加密的数据份2
事务ID份2
加密的数据份3
事务ID份3
加密的数据份n
事务ID份n
产生份1密钥
产生份2密钥
产生份3密钥
产生份n密钥
加密份
用份1密钥对份1数据进行加密
用份2密钥对份2数据进行加密
用份3密钥对份3数据进行加密
用份n密钥对份n数据进行加密
加密的份1(数据/事务ID)
密钥n
加密的份3(数据/事务ID)
密钥2
搅乱
加密的份2(数据/事务ID)
密钥1
加密的份n(数据/事务ID)
密钥3

图 24

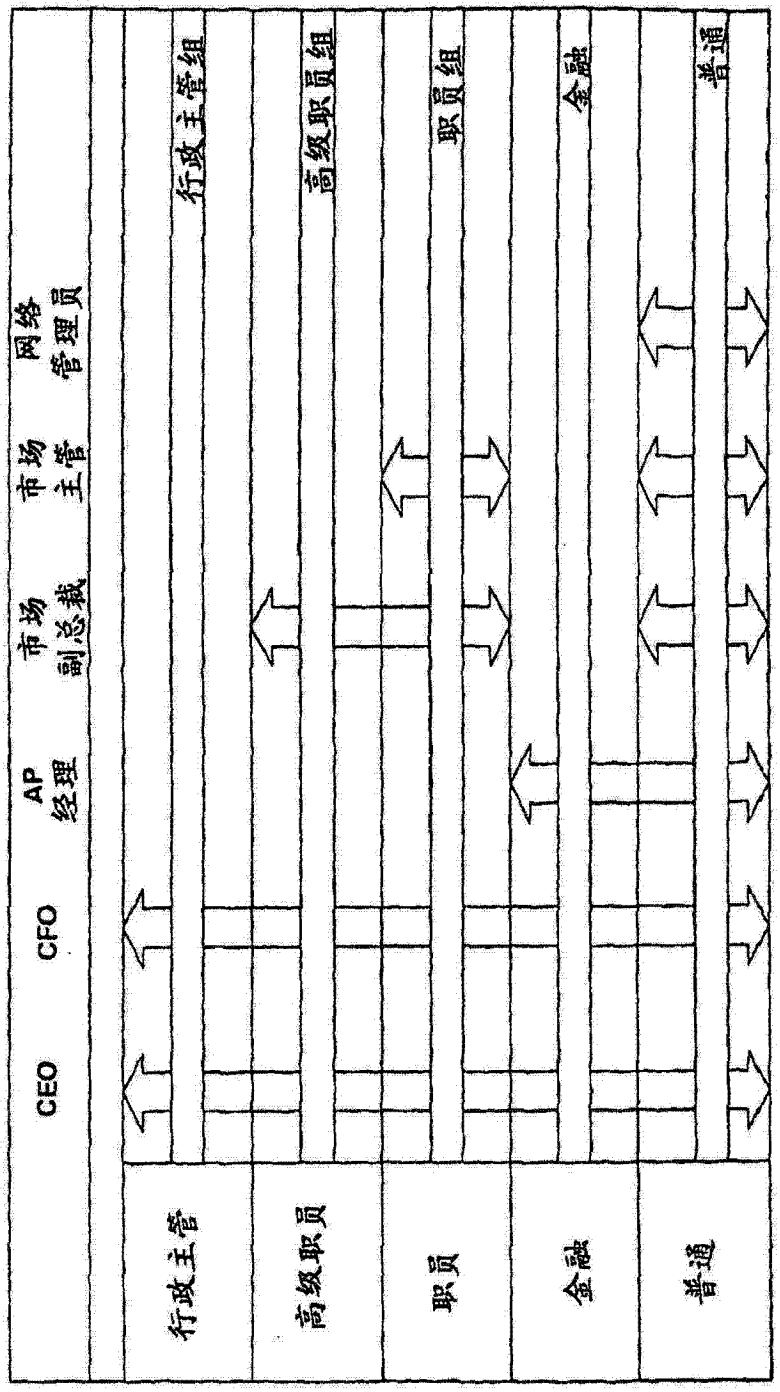

网络管理员
市场主管
市场副总裁
AP经理
CFO
CEO
行政主管组
高级职员组
职员组
金融
普通
行政主管
高级职员
职员
金融
普通

图 25

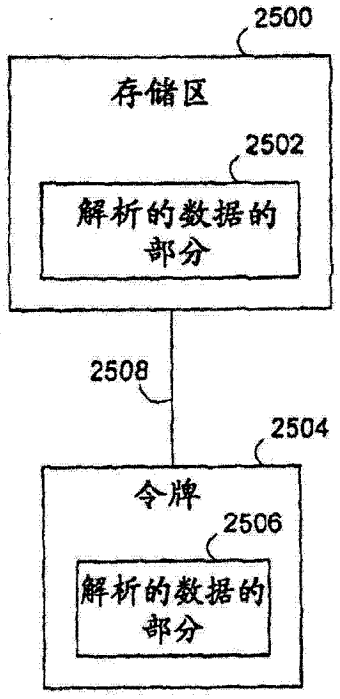

2500
存储区
2502
解析的数据的部分
2508
2504
令牌
2506
解析的数据的部分

图 26

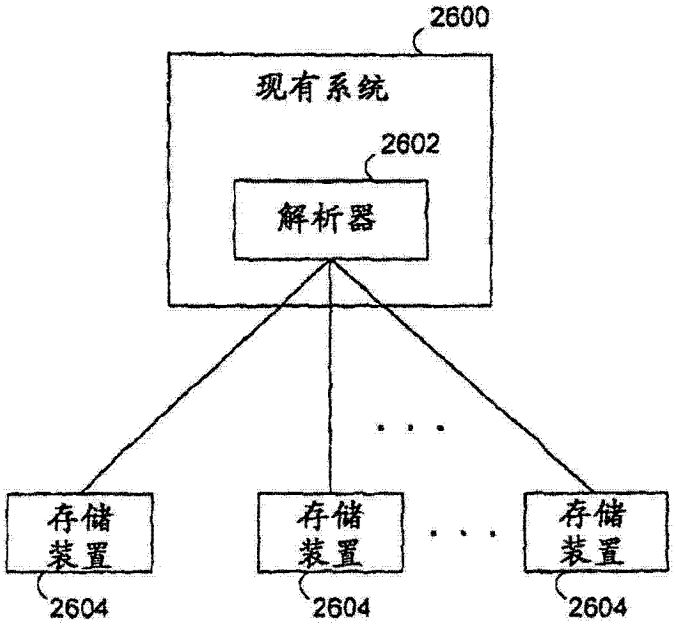


图 27

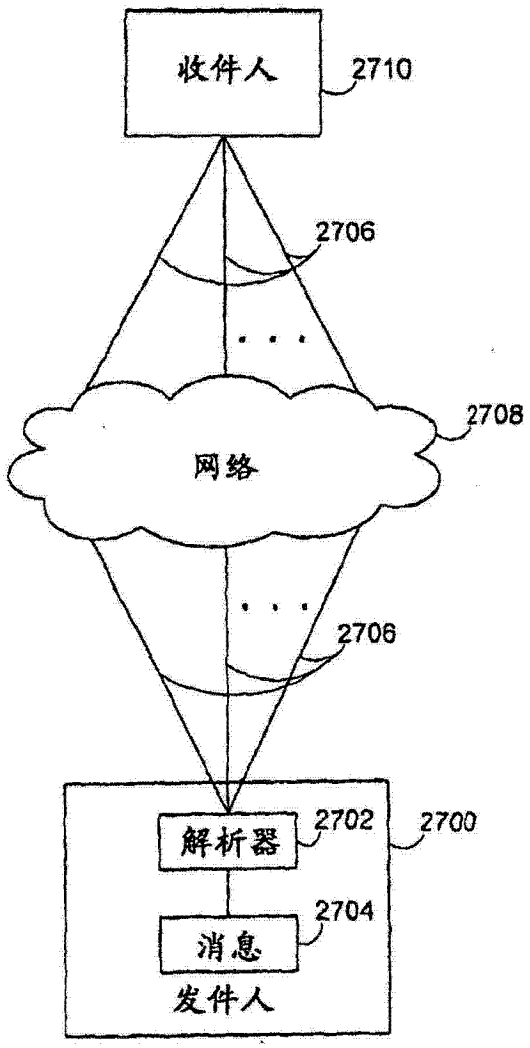

收件人
2710
2706
网络
2708
2706
解析器
2702
2700
消息
2704
发件人

图 28

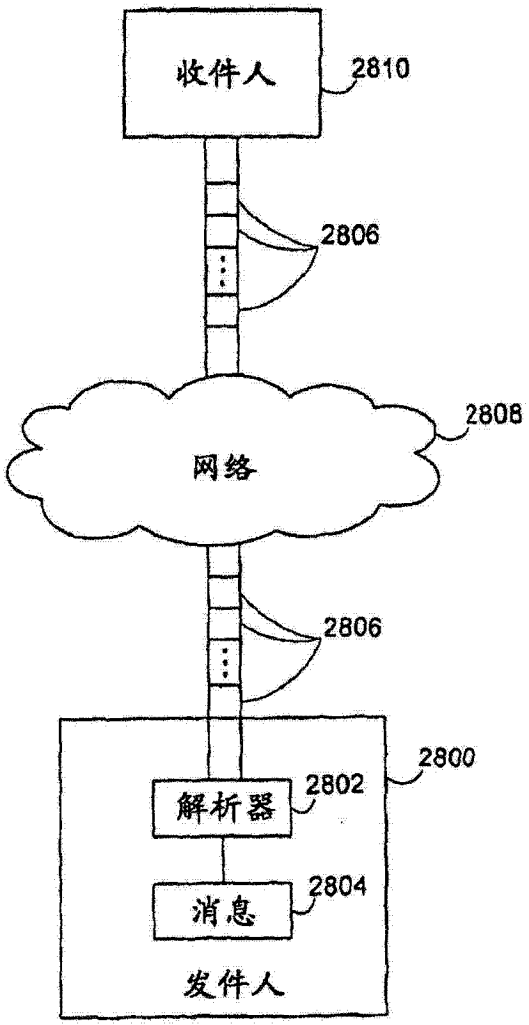

收件人
2810
2806
网络
2808
2806
2800
解析器
2802
消息
2804
发件人

图 29

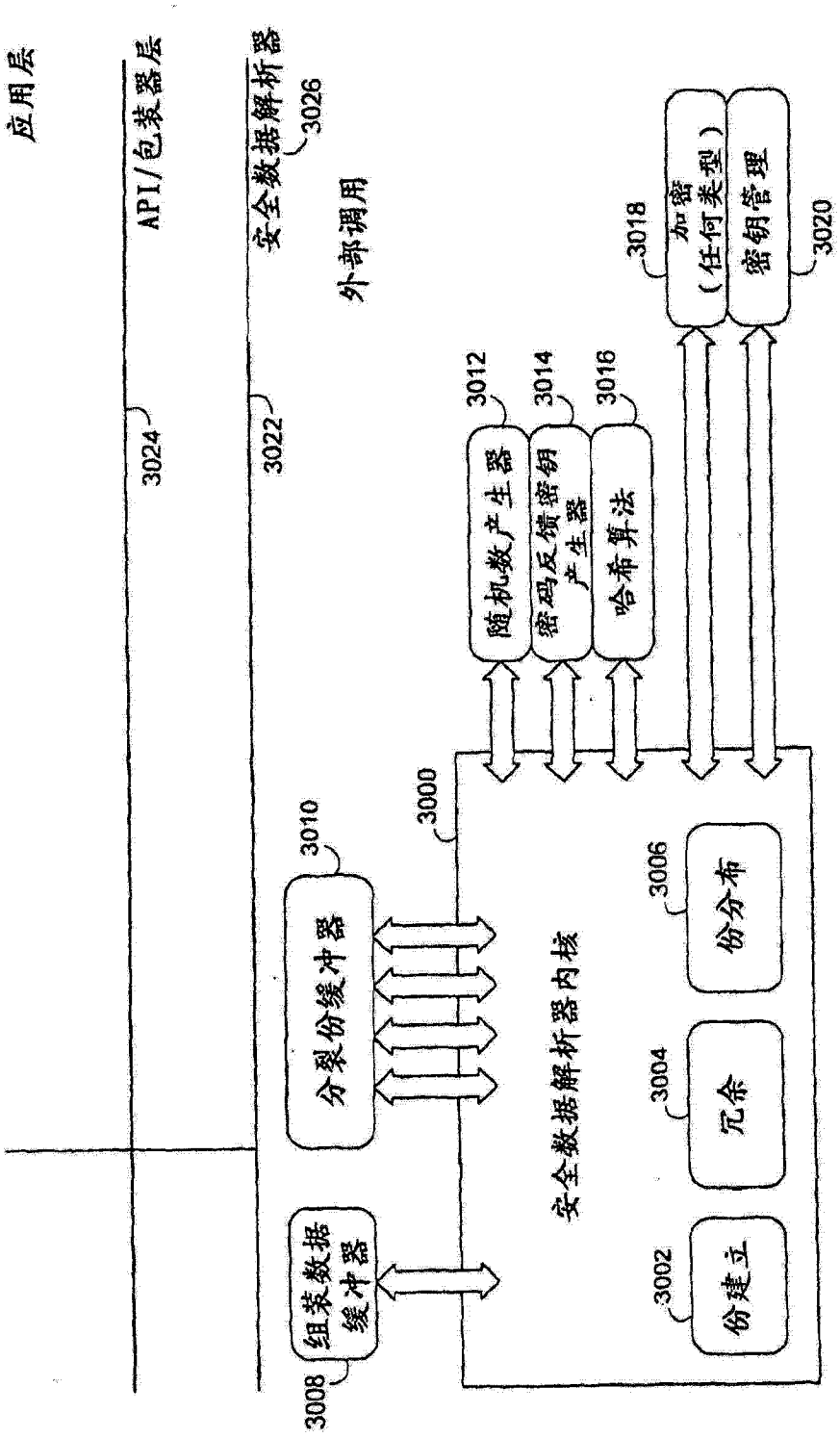

应用层
API/包装器层
安全数据解析器
3026
外部调用
3024
3022
3012
3014
3016
3018
加密（任何类型）
3020
随机数产生器
密码反馈密钥产生器
哈希算法
3000
3010
分裂份缓冲器
3006
份分布
安全数据解析器内核
3004
冗余
3002
份建立
组装数据缓冲器
3008

图 30

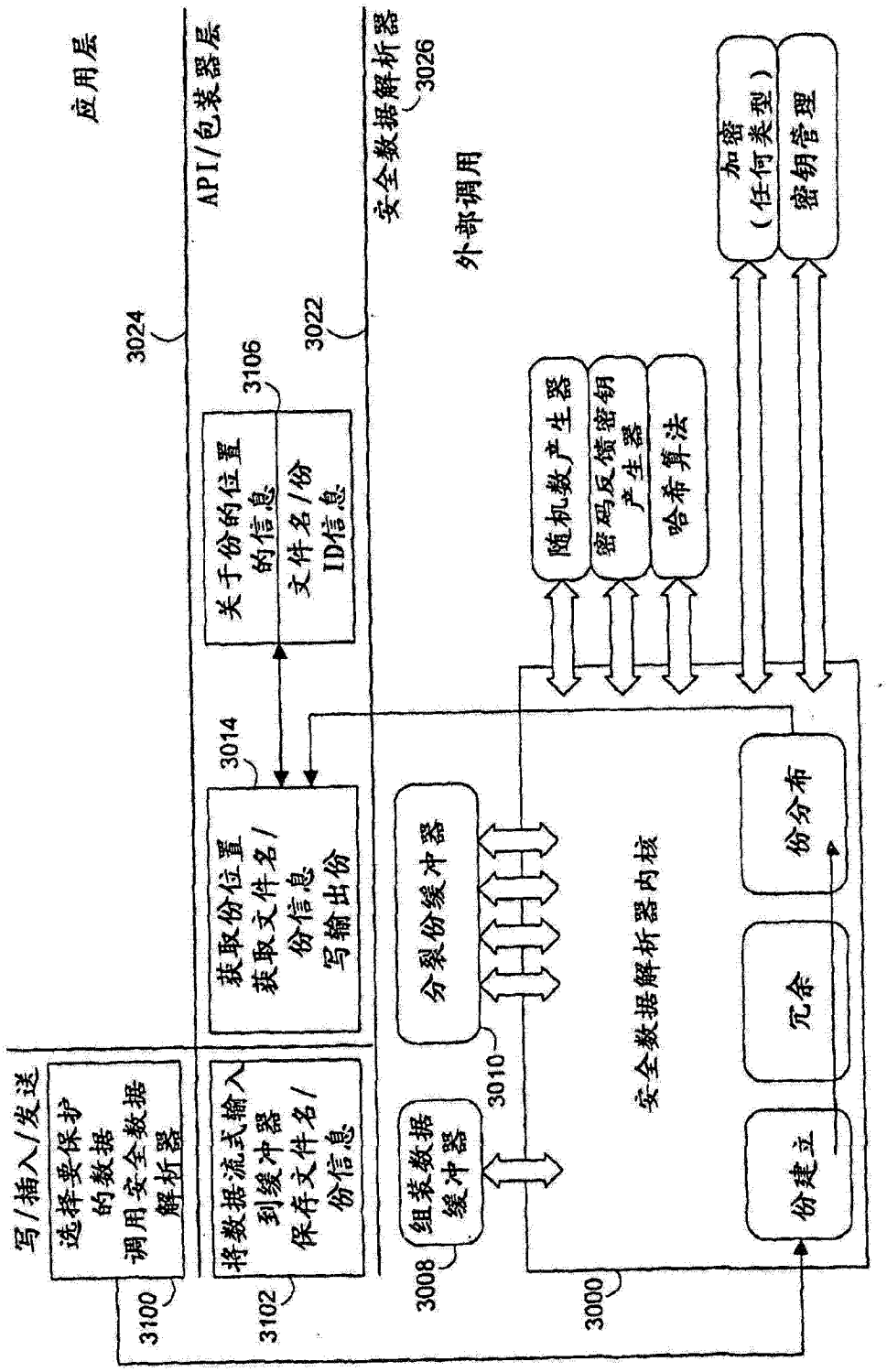

应用层
API/包装器层
安全数据解析器
3026
外部调用
3024
3022
3106
关于份的位置的信息
文件名/份ID信息
3014
获取份位置
获取文件名/份信息
写输出份
随机数产生器
密码反馈密钥产生器
哈希算法
加密（任何类型）
密钥管理
写/插入/发送
选择要保护的数据调用安全数据解析器
3100
将数据流式输入到缓冲器保存文件名/份信息
3102
分裂份缓冲器
3010
组装数据缓冲器
3008
安全数据解析器内核
3000
份分布
冗余
份建立

图 31

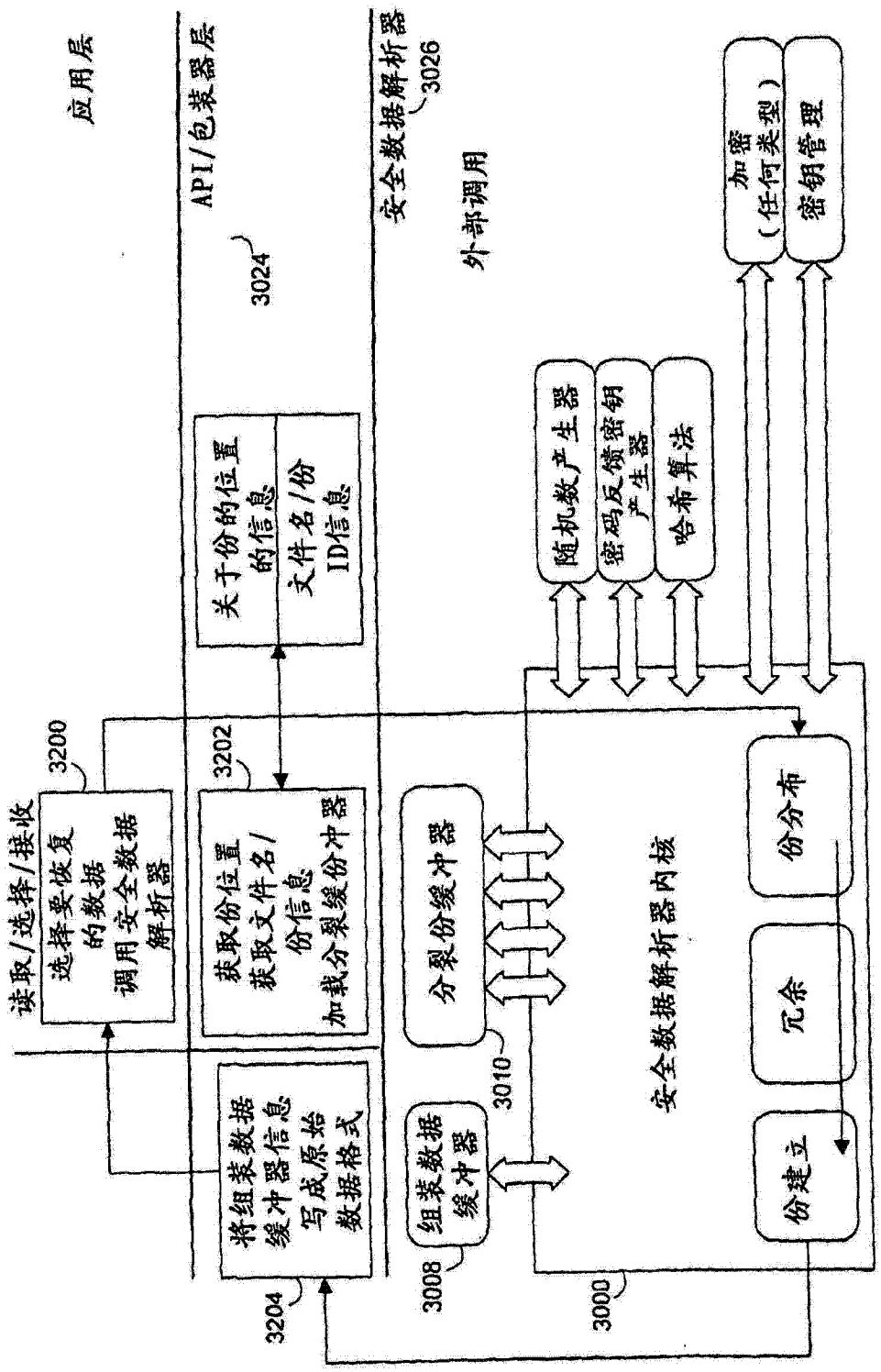

应用层
API/包装器层
3024
安全数据解析器
3026
外部调用
读取/选择/接收
3200
选择要恢复的数据调用安全数据解析器
3202
获取份位置获取文件名/份信息加载分裂缓份冲器
关于份的位置的信息
文件名/份ID信息
将组装数据缓冲器信息写成原始数据格式
3204
分裂份缓冲器
3010
组装数据缓冲器
3008
安全数据解析器内核
3000
随机数产生器
密码反馈密钥产生器
哈希算法
加密（任何类型）
密钥管理
份分布
冗余
份建立

图 32

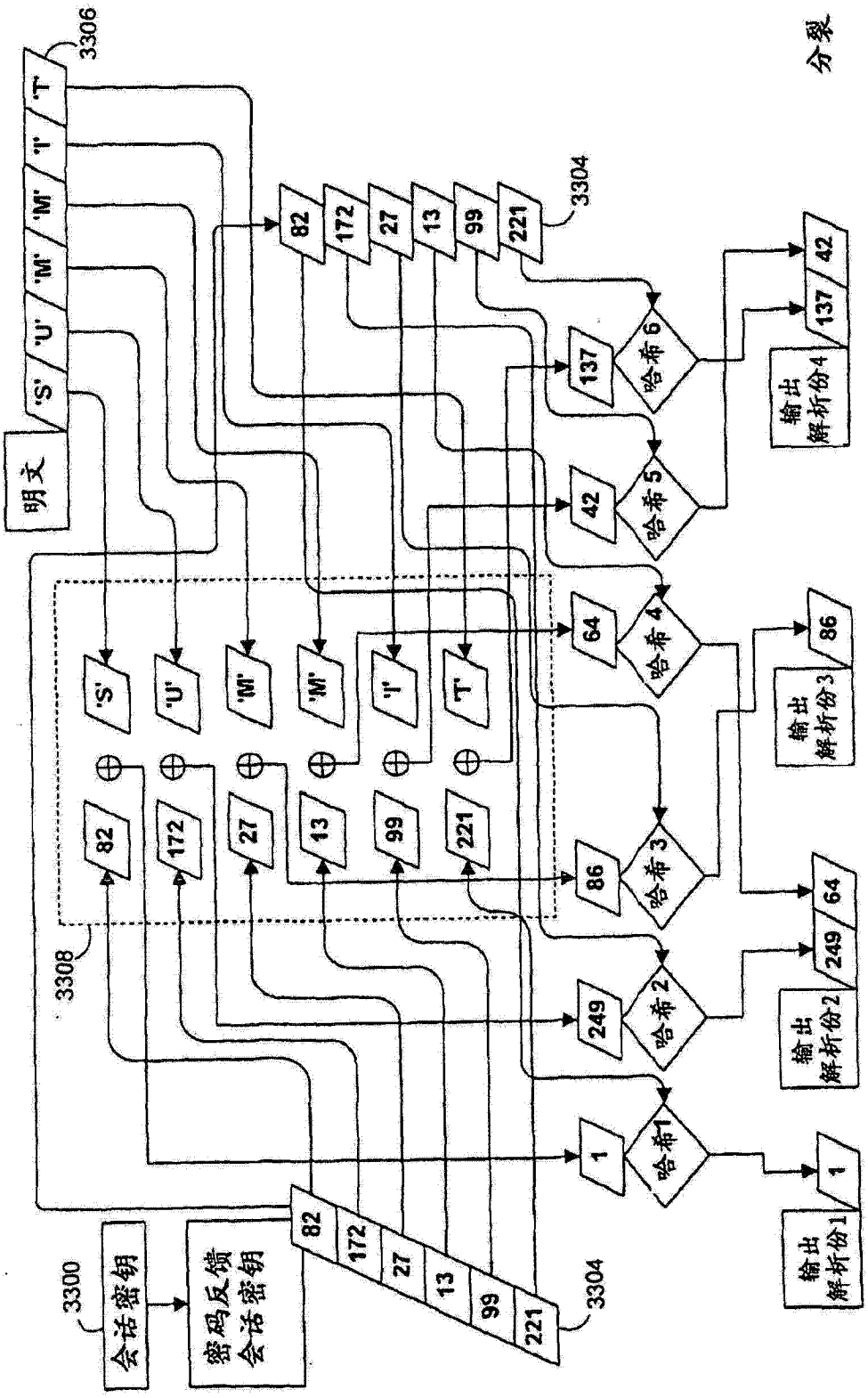

分裂
明文
'S' 'U' 'M' 'M' 'I' 'T'
3306
82 172 27 13 99 221
3304
哈希1
哈希2
哈希3
哈希4
哈希5
哈希6
1
249
86
64
42
137
输出解析份1
输出解析份2
输出解析份3
输出解析份4
3308
3300
会话密钥
密码反馈会话密钥

图 33

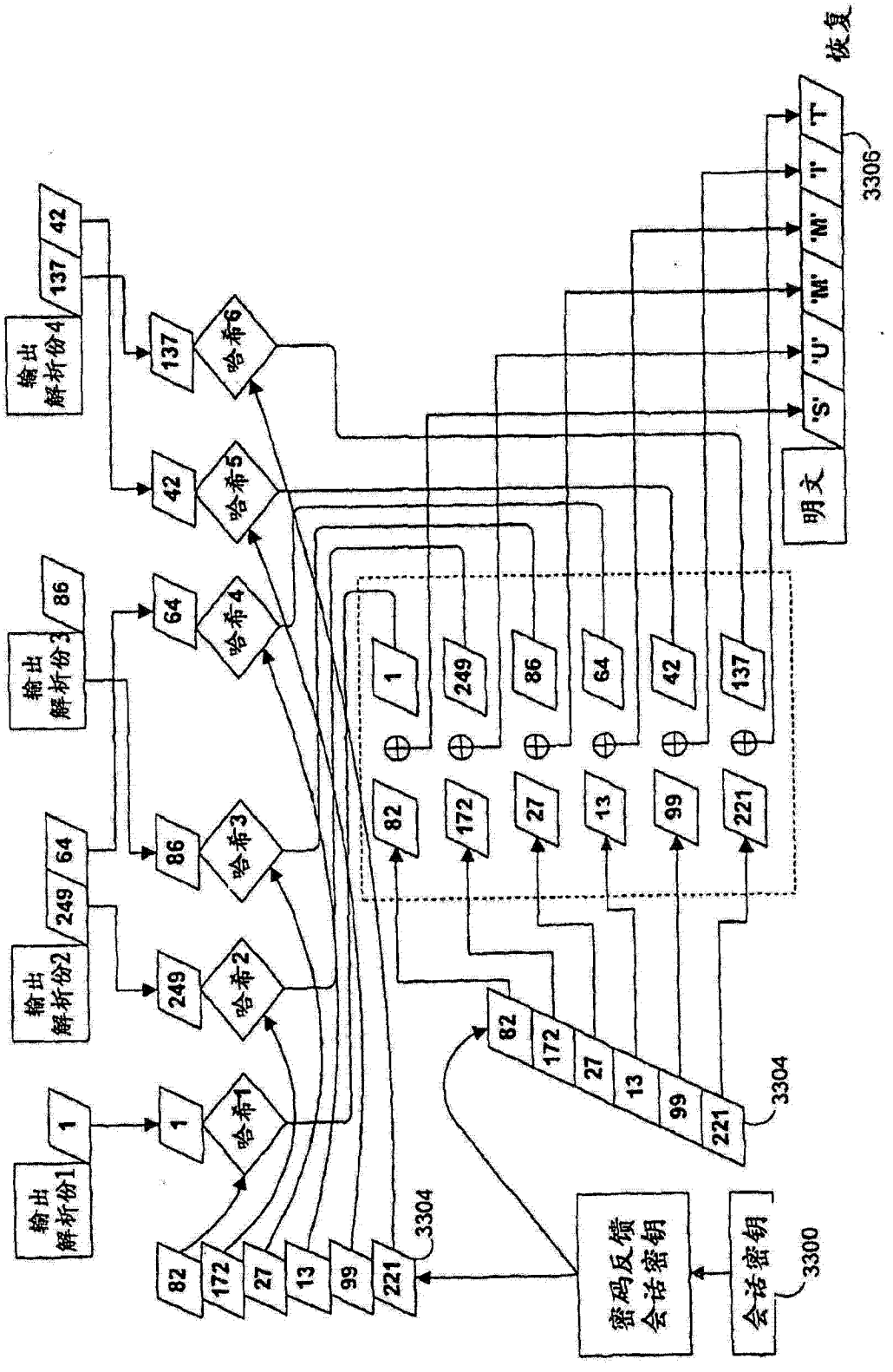

输出解析份1
输出解析份2
输出解析份3
输出解析份4
哈希1
哈希2
哈希3
哈希4
哈希5
哈希6
密码反馈会话密钥
会话密钥
3300
3304
3306
明文
恢复
'S'
'U'
'M'
'M'
'I'
'T'

图 34

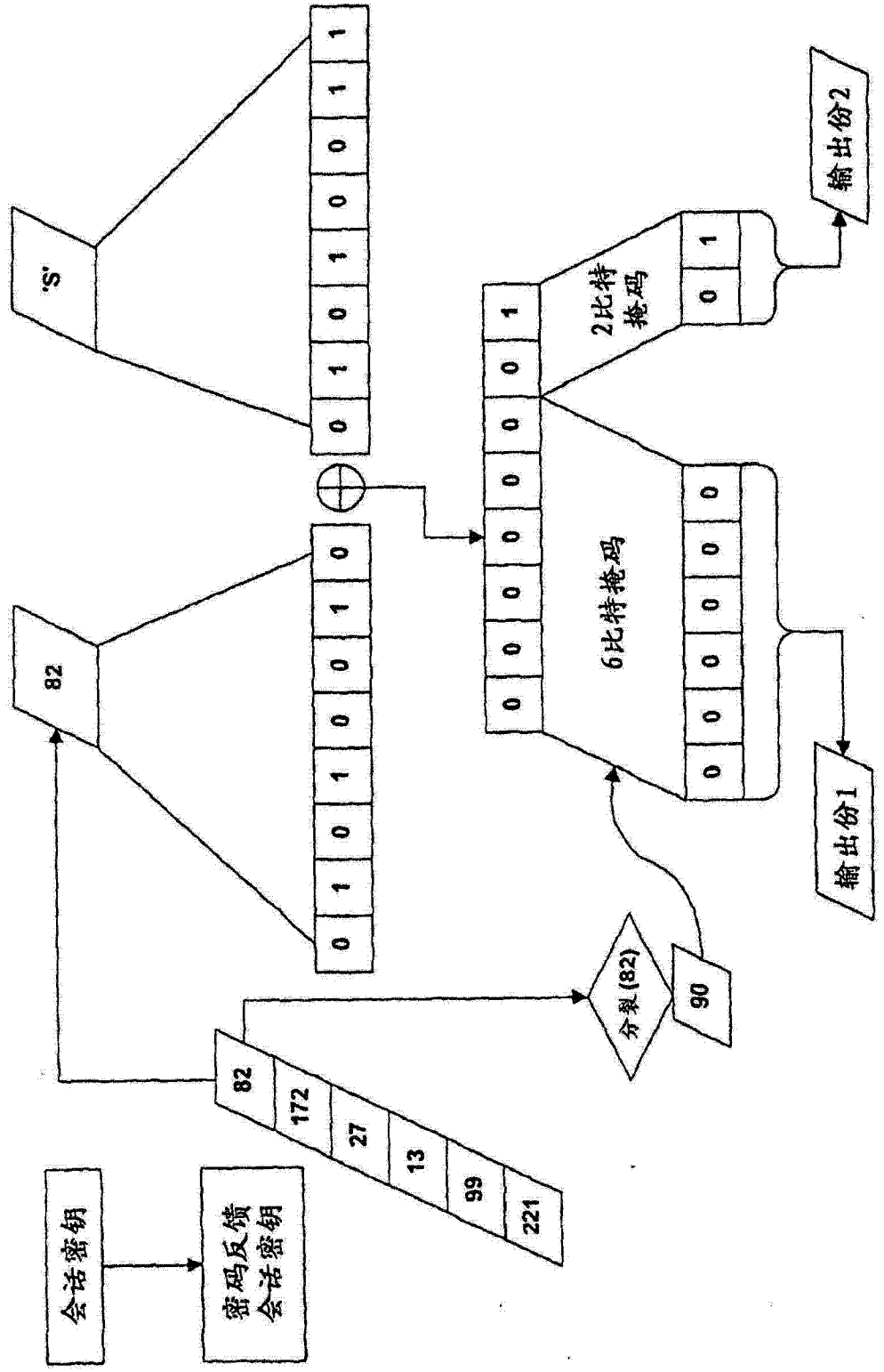

'S'
82
82
172
27
13
99
221
会话密钥
密码反馈
会话密钥
分裂(82)
90
6比特掩码
2比特
掩码
输出份1
输出份2

图 35

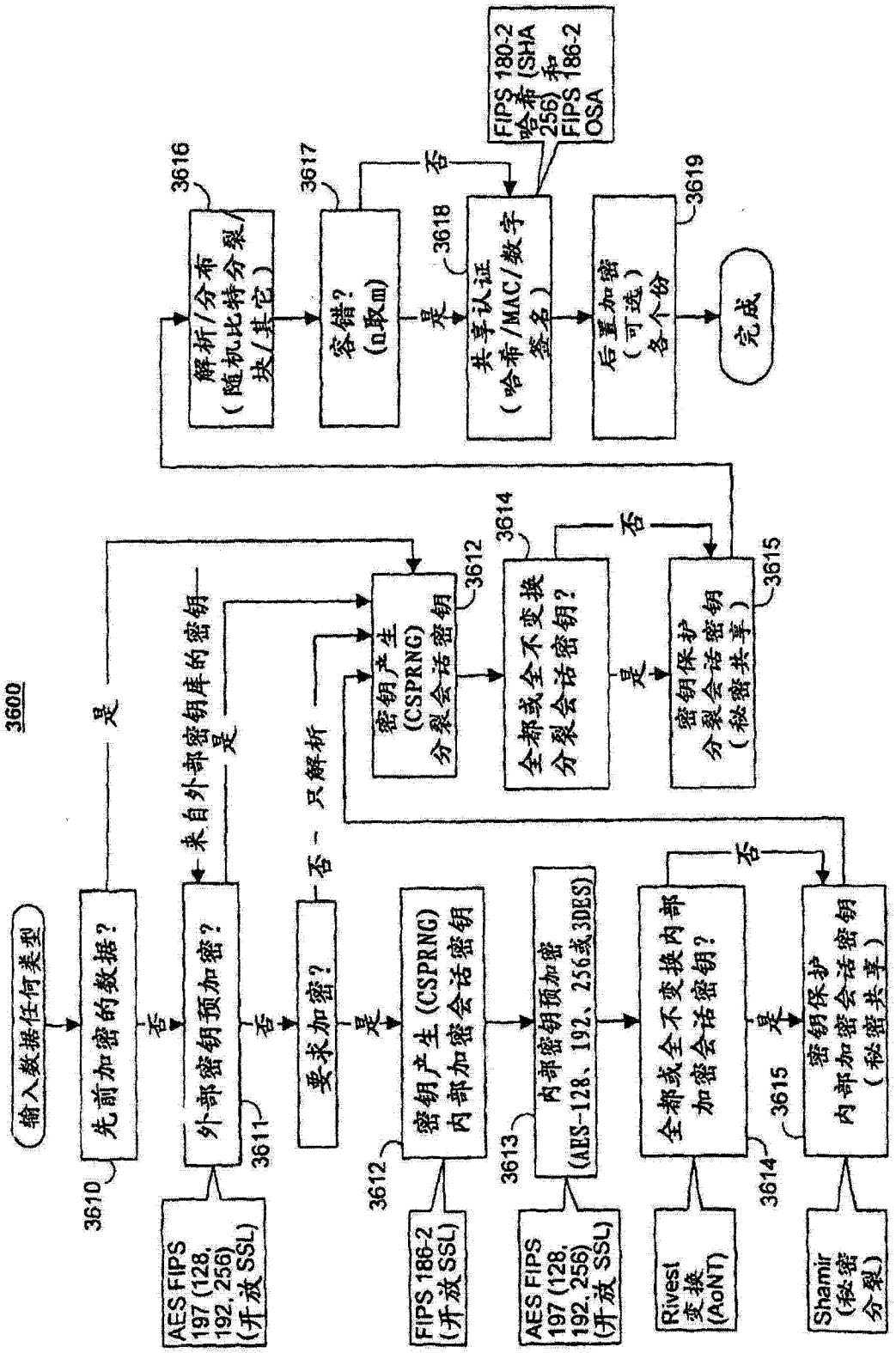

3600
输入数据任何类型
先前加密的数据?
3610
是
否
外部密钥预加密?
3611
来自外部密钥库的密钥
是
否
AES FIPS 197 (128, 192, 256) (开放 SSL)
要求加密?
否 - 只解析
是
密钥产生 (CSPRNG) 内部加密会话密钥
3612
FIPS 186-2 (开放 SSL)
内部密钥预加密 (AES-128、192、256或3DES)
3613
AES FIPS 197 (128, 192, 256) (开放 SSL)
全都或全不变换内部加密会话密钥?
3614
是
否
Rivest 变换 (AoNT)
密钥保护 内部加密会话密钥 (秘密共享)
3615
Shamir (秘密分裂)
密钥产生 (CSPRNG) 分裂会话密钥
3612
全都或全不变换分裂会话密钥?
3614
是
否
密钥保护 分裂会话密钥 (秘密共享)
3615
解析/分布 (随机比特分裂块/其它)
3616
容错? (n取m)
3617
是
否
共享认证 (哈希/MAC/数字签名)
3618
FIPS 180-2 (SHA 哈希 256) 和 FIPS 186-2 OSA
后置加密 (可选) 各个份
3619
完成

图 36

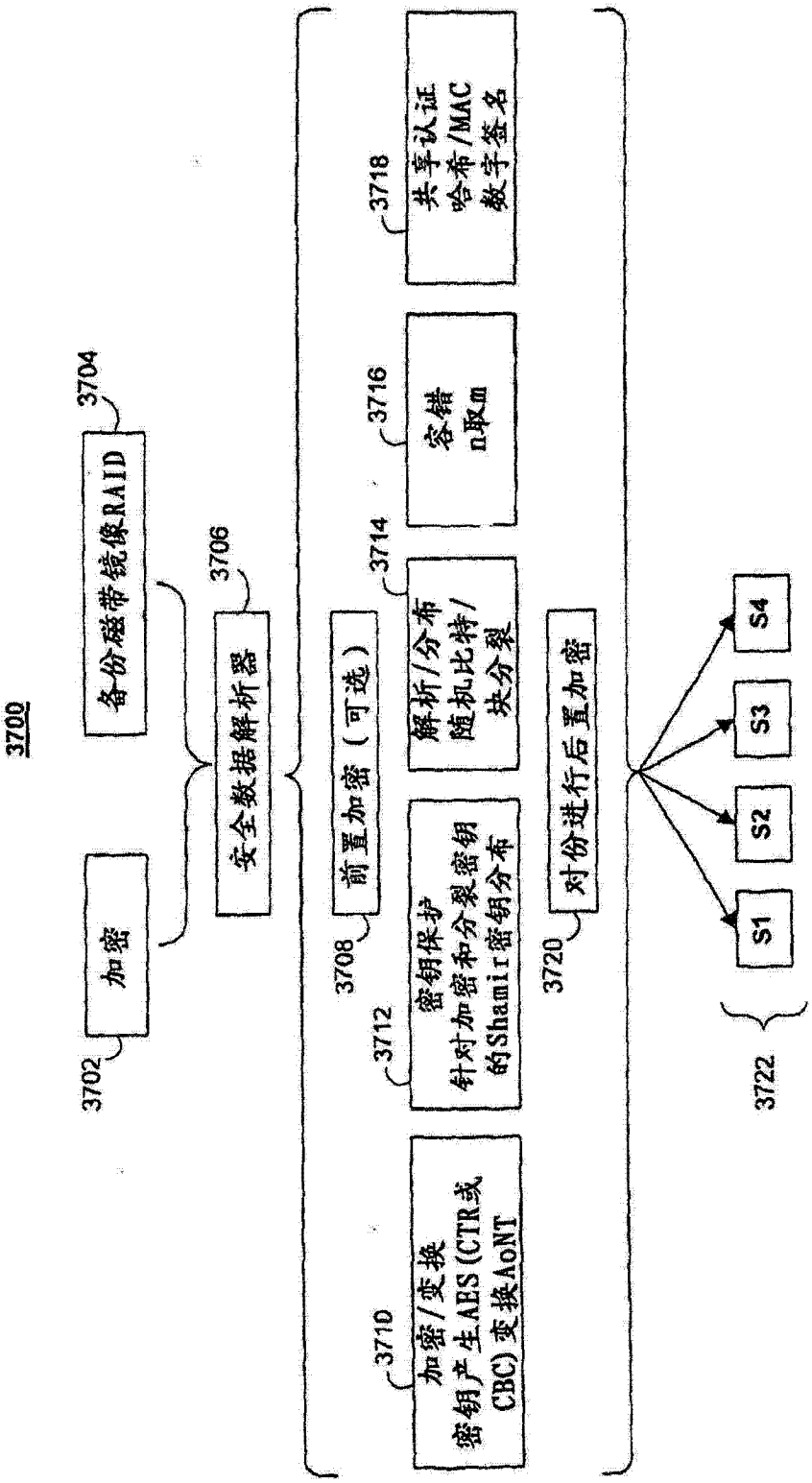

3700
3702
加密
3704
备份磁带镜像RAID
3706
安全数据解析器
3708
前置加密（可选）
3710
加密/变换
密钥产生AES(CTR或
CBC)变换AoNT
3712
密钥保护
针对加密和分裂密钥
的Shamir密钥分布
3714
解析/分布
随机比特/
块分裂
3716
容错
n取m
3718
共享认证
哈希/MAC
数字签名
3720
对份进行后置加密
S1
S2
S3
S4
3722

图 37

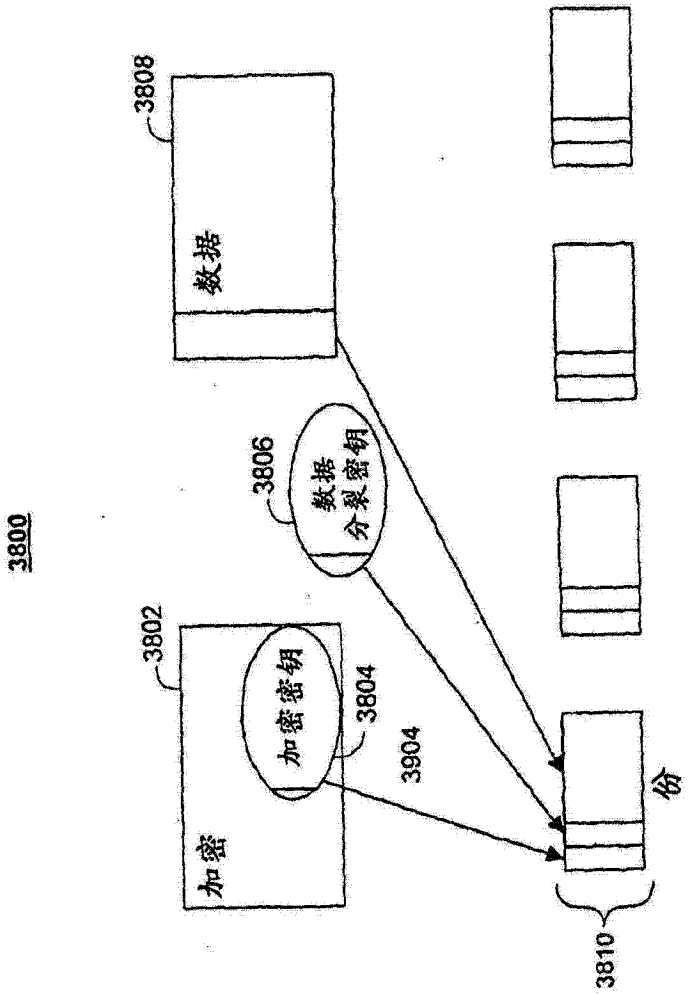


图 38

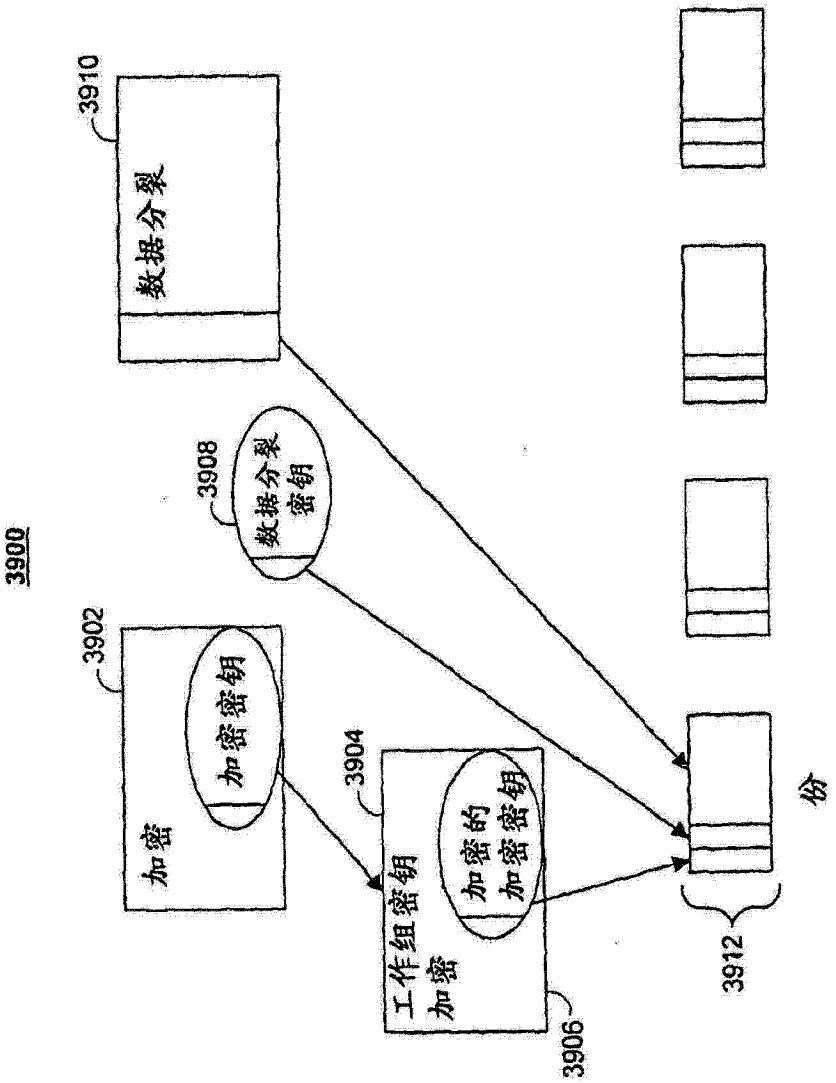

3900
3910
数据分裂
3908
数据分裂密钥
3902
加密
加密密钥
3904
工作组密钥加密
加密的加密密钥
3906
3912
份

图 39

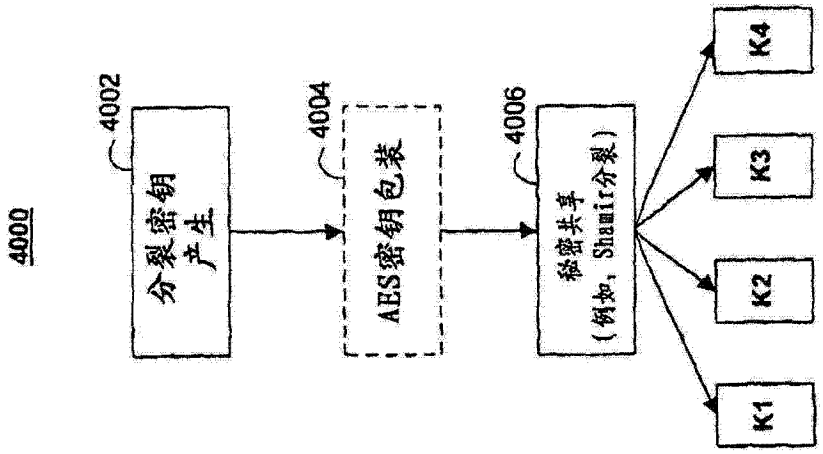

4000
4002
分裂密钥产生
4004
AES密钥包装
4006
秘密共享（例如，Shamir分裂）
K4
K3
K2
K1

图 40A

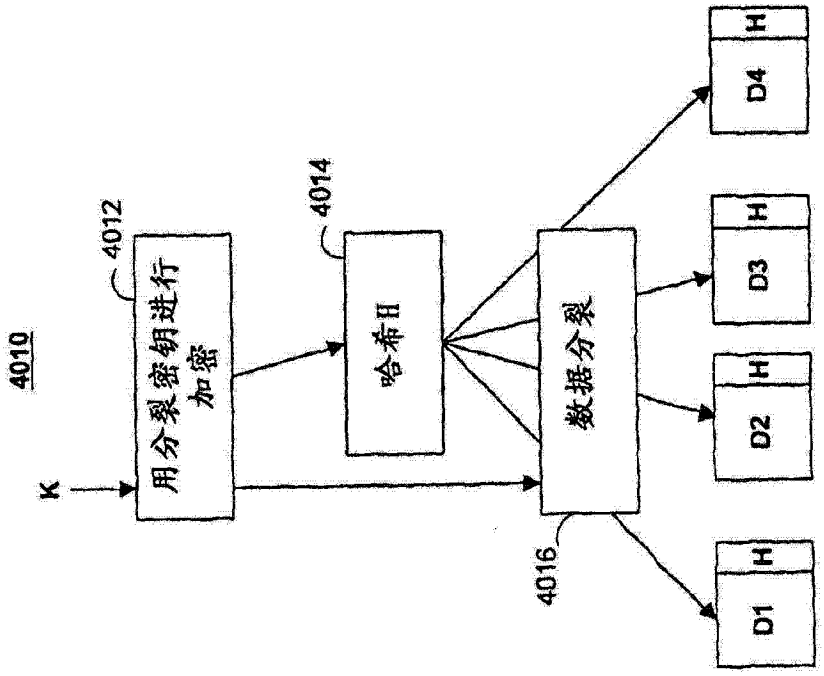

4010
K
4012
用分裂密钥进行加密
4014
哈希H
数据分裂
4016
H
D4
H
D3
H
D2
H
D1

图 40B

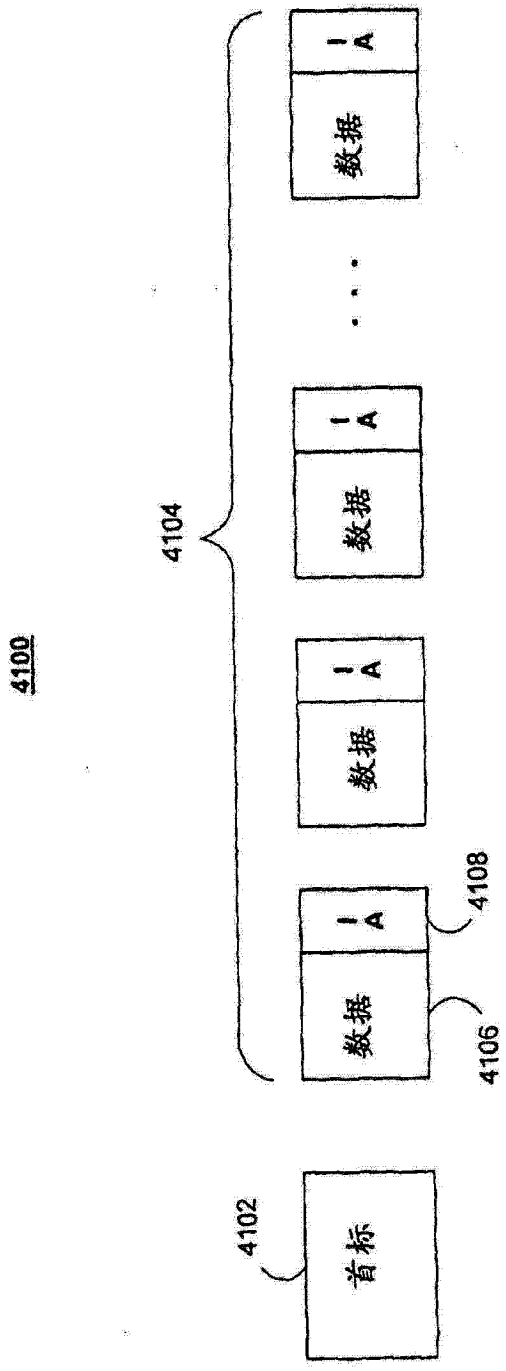


图 41

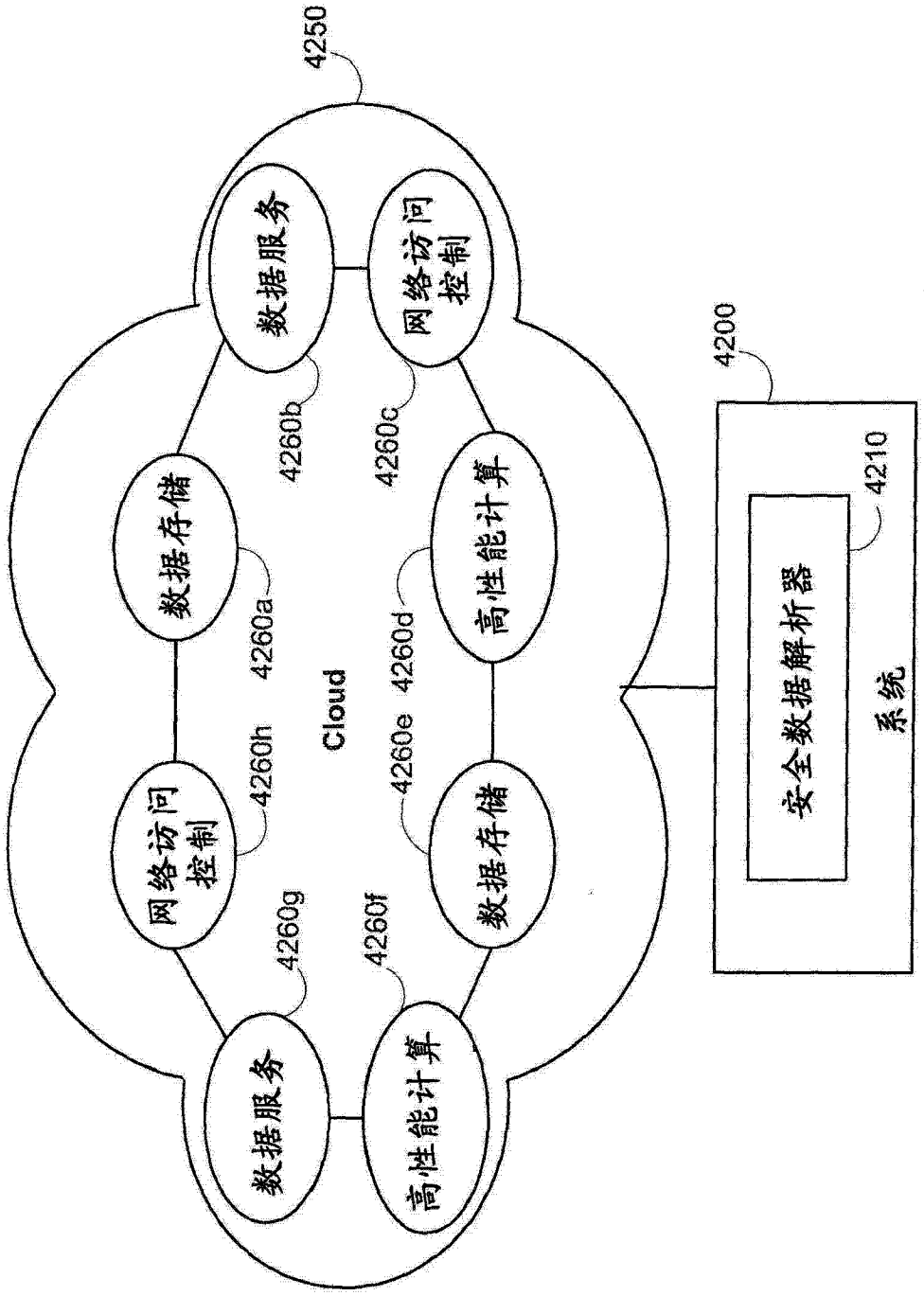

4250
数据服务
网络访问控制
4260b
4260c
数据存储
高性能计算
4260a
4260d
Cloud
4260h
4260e
网络访问控制
数据存储
4260g
4260f
数据服务
高性能计算
4200
4210
安全数据解析器
系统

图 42

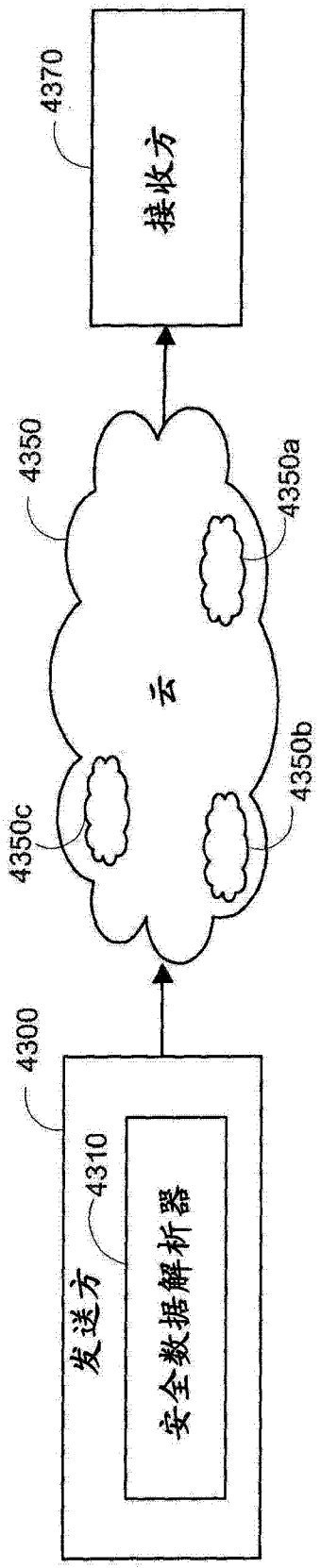

4370
接收方
4350
4350a
云
4350b
4350c
4300
发送方
4310
安全数据解析器

图 43

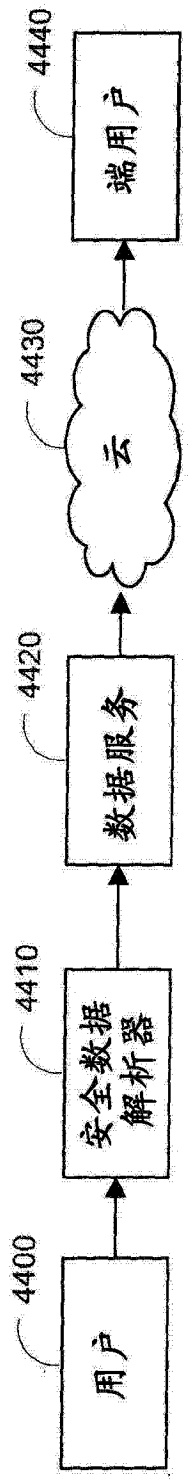

4440
端用户
4430
云
4420
数据服务
4410
安全数据解析器
4400
用户

图 44

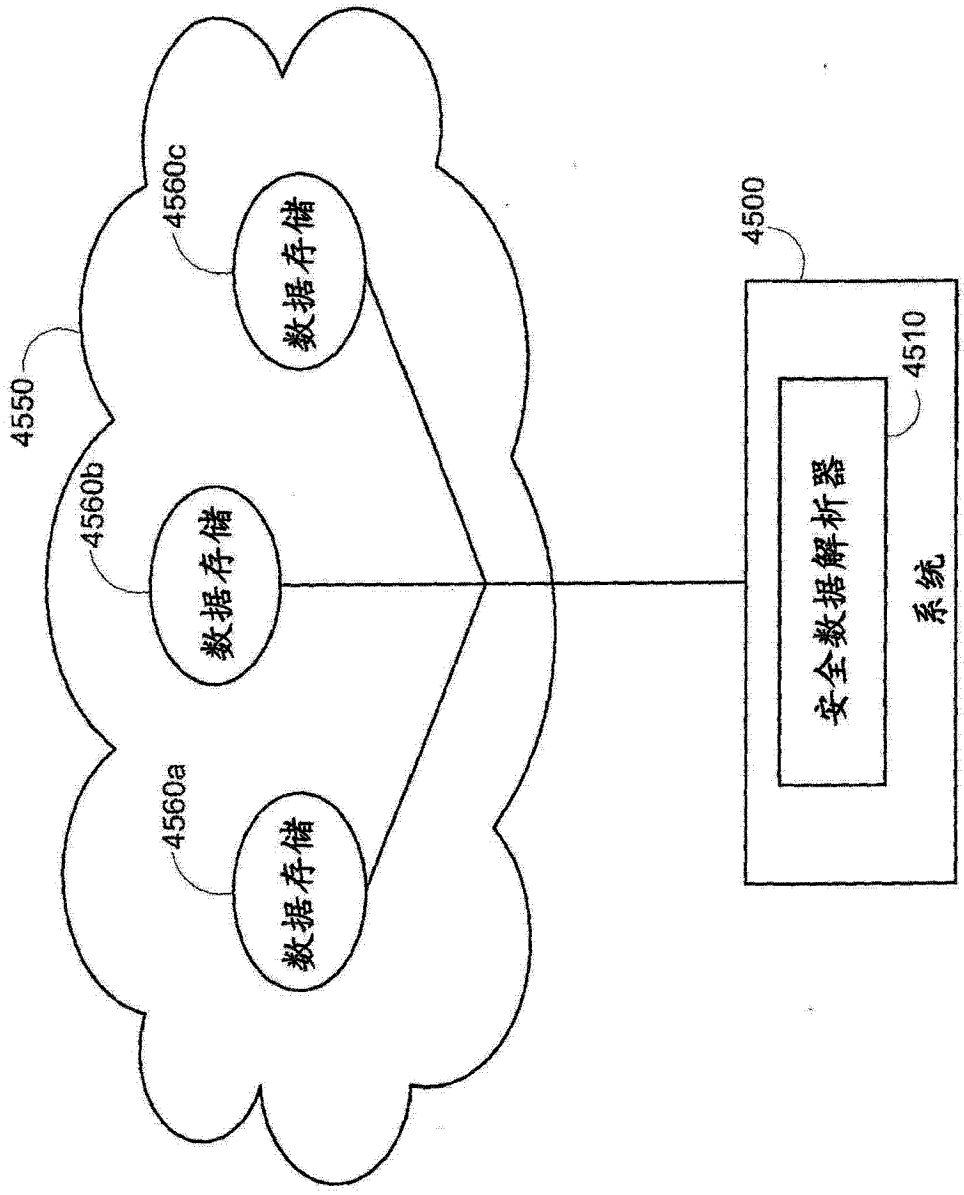

4550
4560c
数据存储
4560b
数据存储
4560a
数据存储
4500
4510
安全数据解析器
系统

图 45

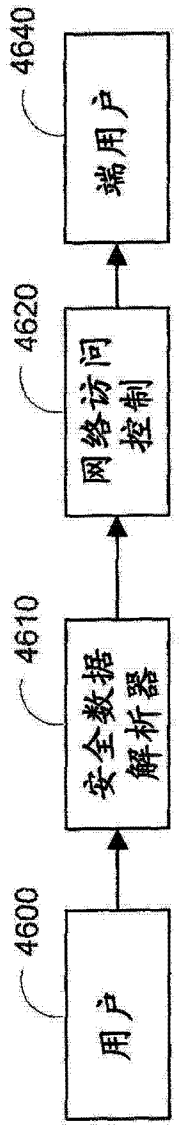

4640
端用户
4620
网络访问控制
4610
安全数据解析器
4600
用户

图 46

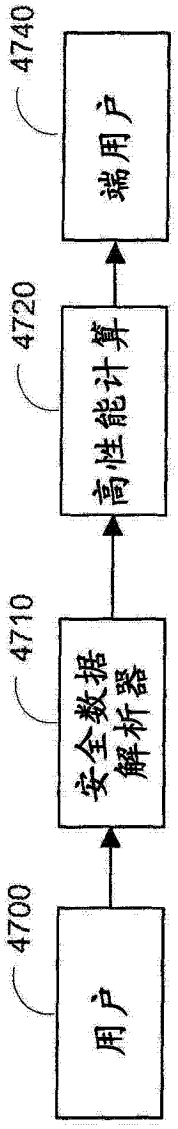


图 47

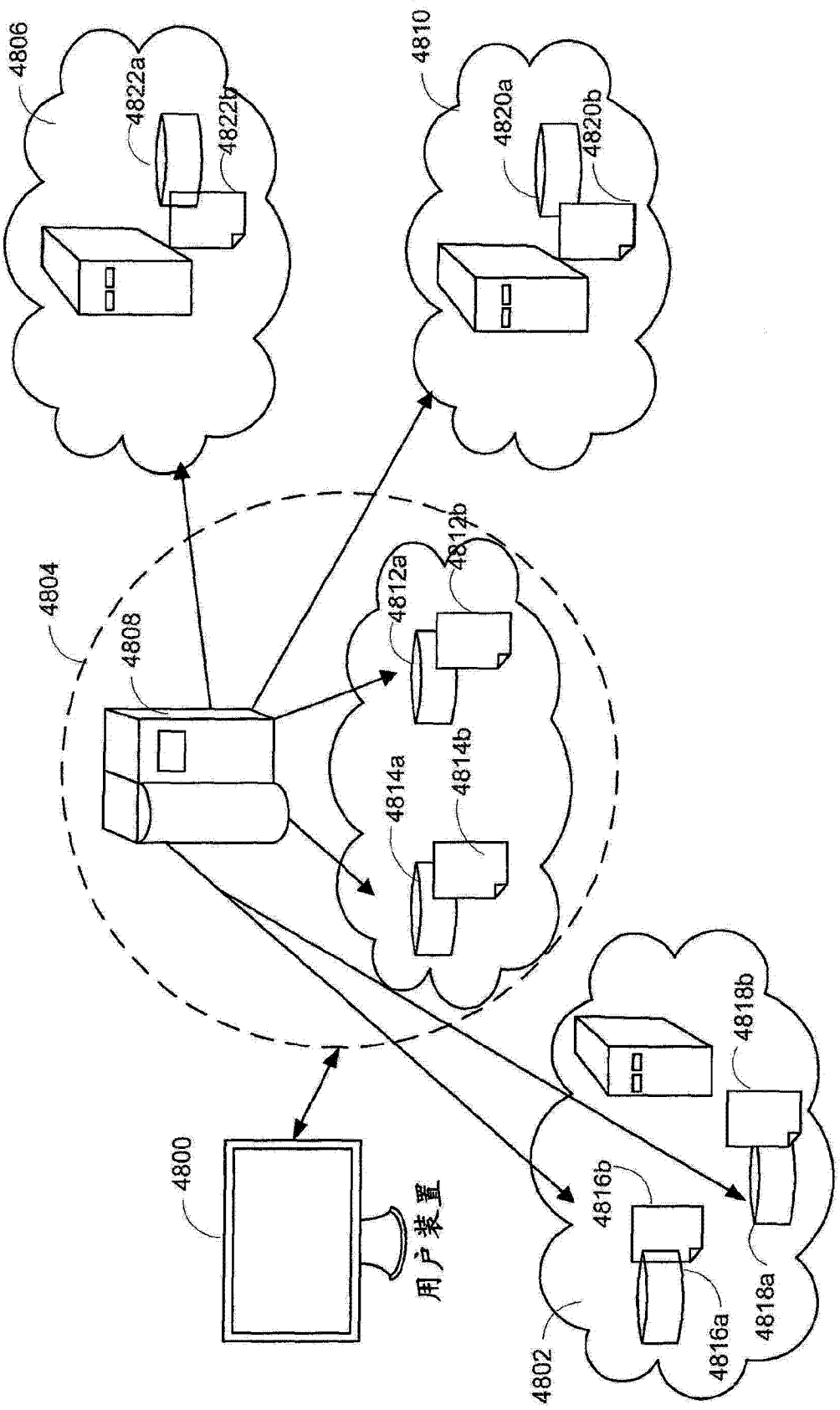

4806
4822a
4822b
4810
4820a
4820b
4804
4808
4812a
4812b
4814a
4814b
4800
用户装置
4802
4816a
4816b
4818a
4818b

图 48

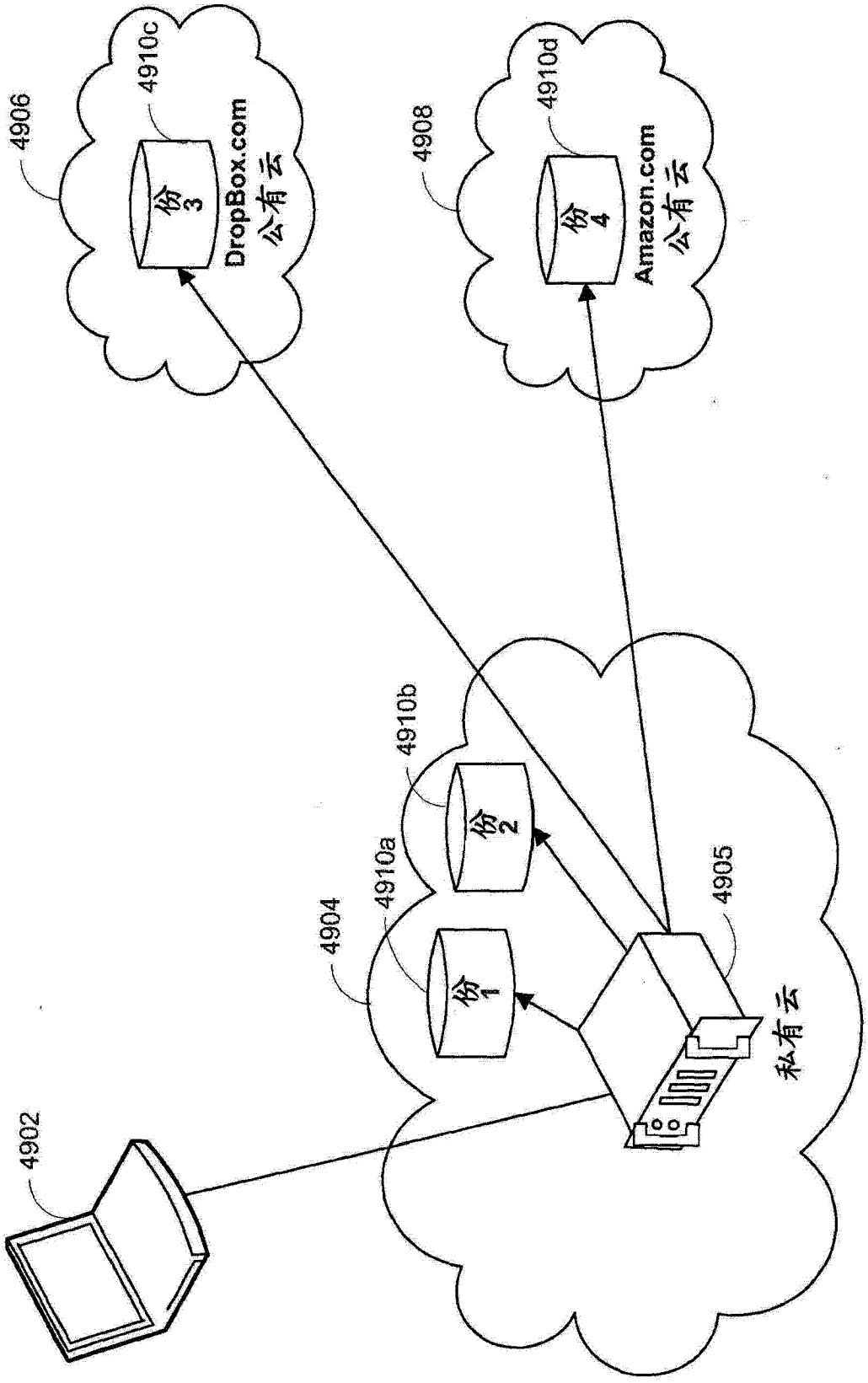

4906
4910c
份3
DropBox.com
公有云
4908
4910d
份4
Amazon.com
公有云
4910b
份2
4910a
4904
份1
4905
私有云
4902

图 49

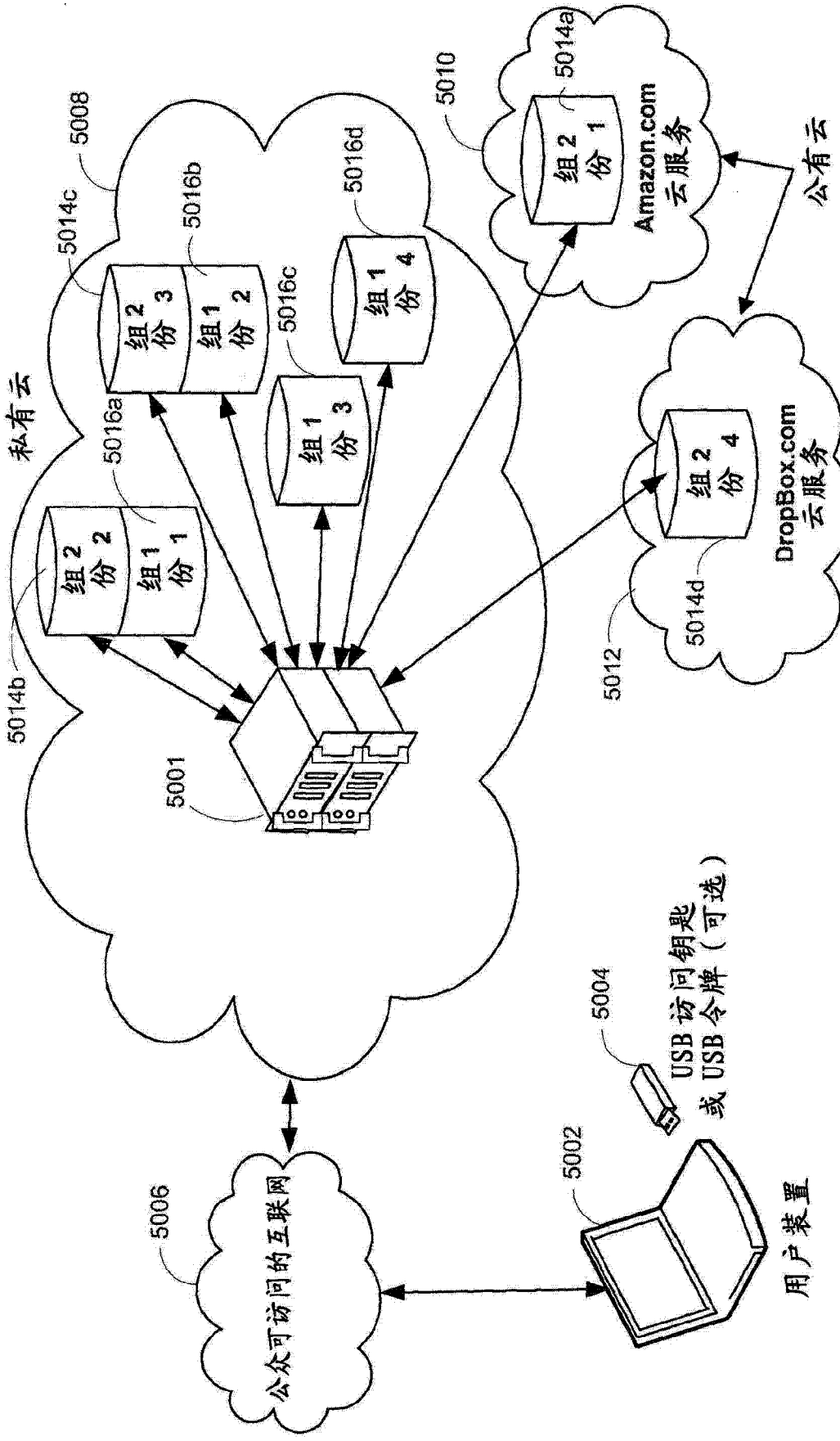

私有云
5008
5014c
5016b
5016d
5010
5014a
组 2 份 3
组 1 份 2
组 1 份 4
组 2 份 1
Amazon.com 云服务
公有云
5016c
5016a
组 1 份 3
组 2 份 2
组 1 份 1
5014b
5001
5012
组 2 份 4
5014d
DropBox.com 云服务
5004
USB 访问钥匙 或 USB 令牌（可选）
5002
5006
公众可访问的互联网
用户装置

图 50

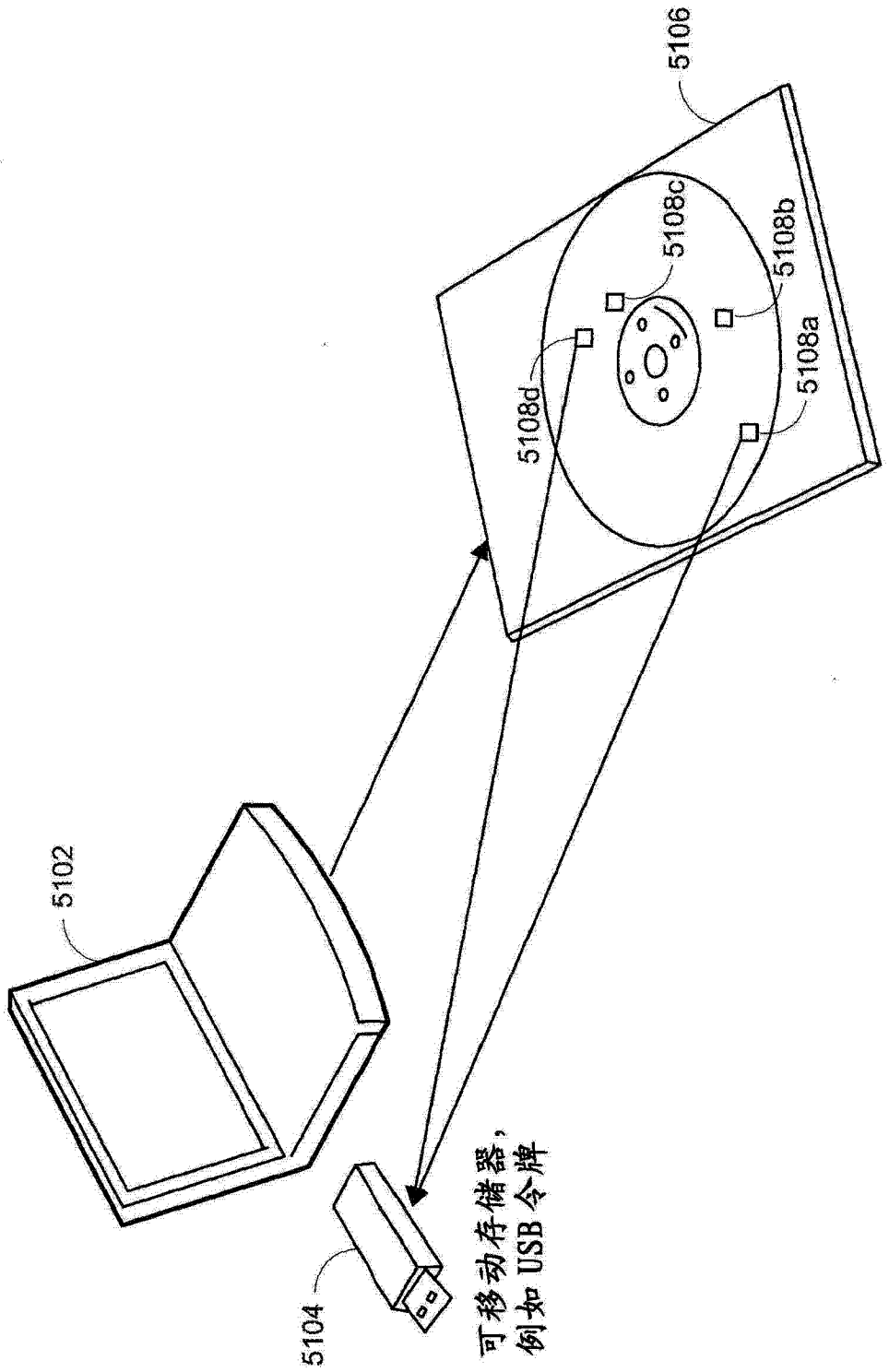


图 51

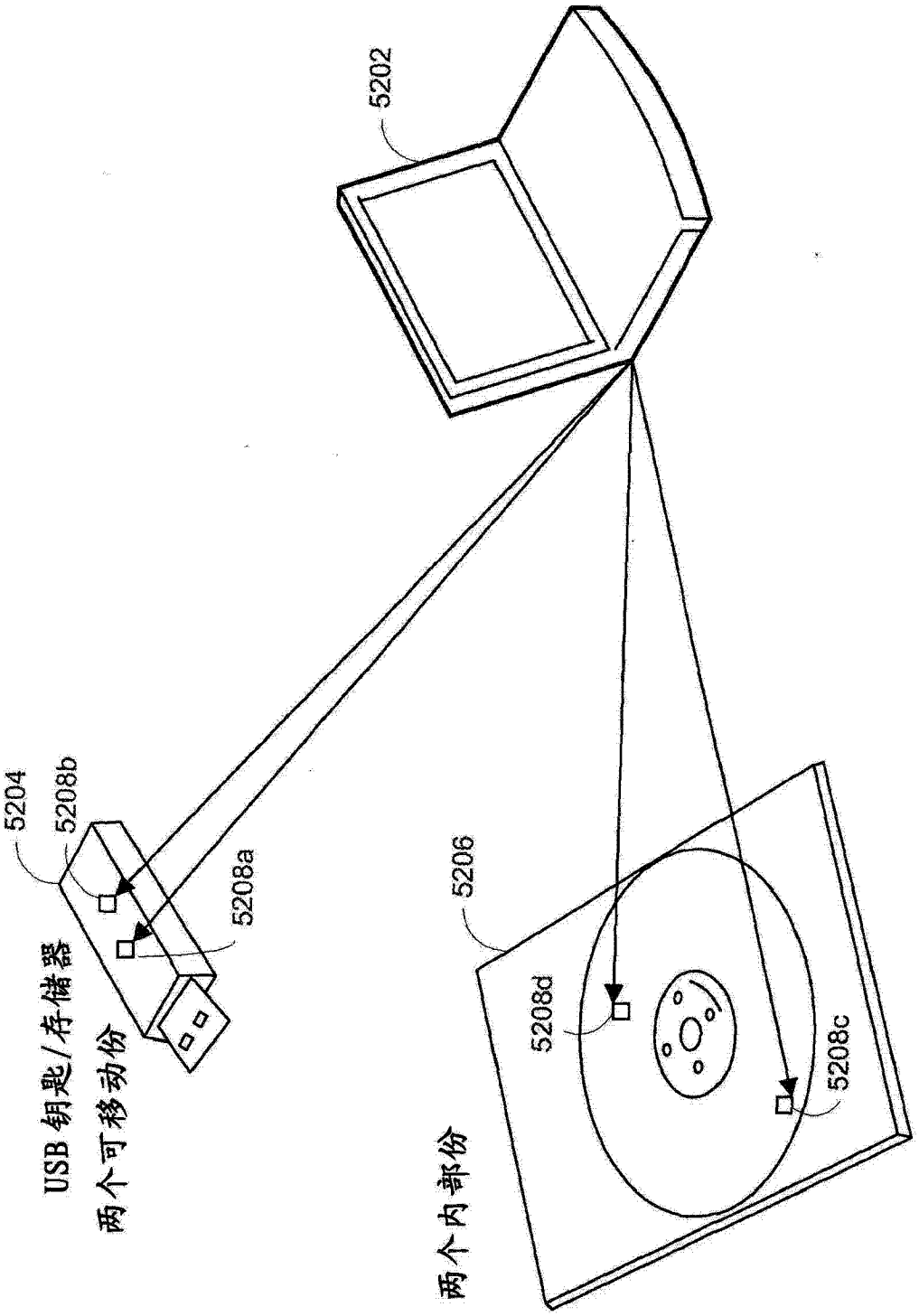

5202
5204
5208b
5208a
USB 钥匙/存储器
两个可移动份
5206
5208d
5208c
两个内部份

图 52

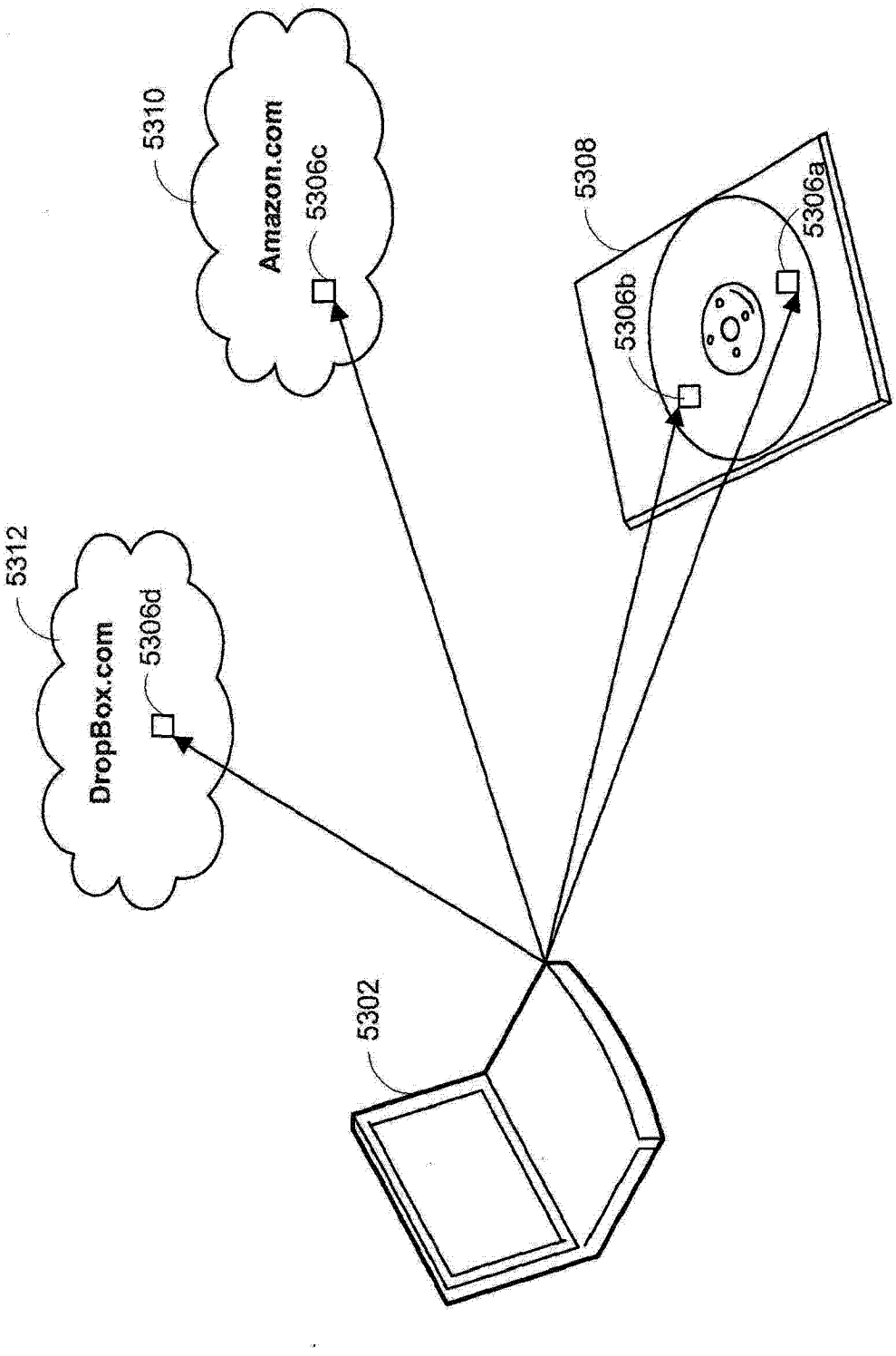

5310
Amazon.com
5306c
5308
5306a
5306b
5312
DropBox.com
5306d
5302

图 53

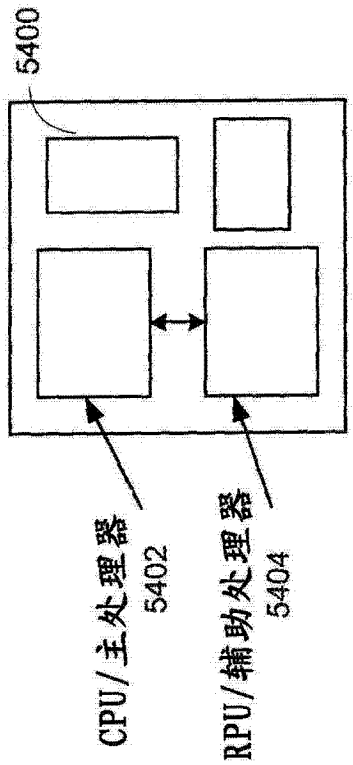


图 54

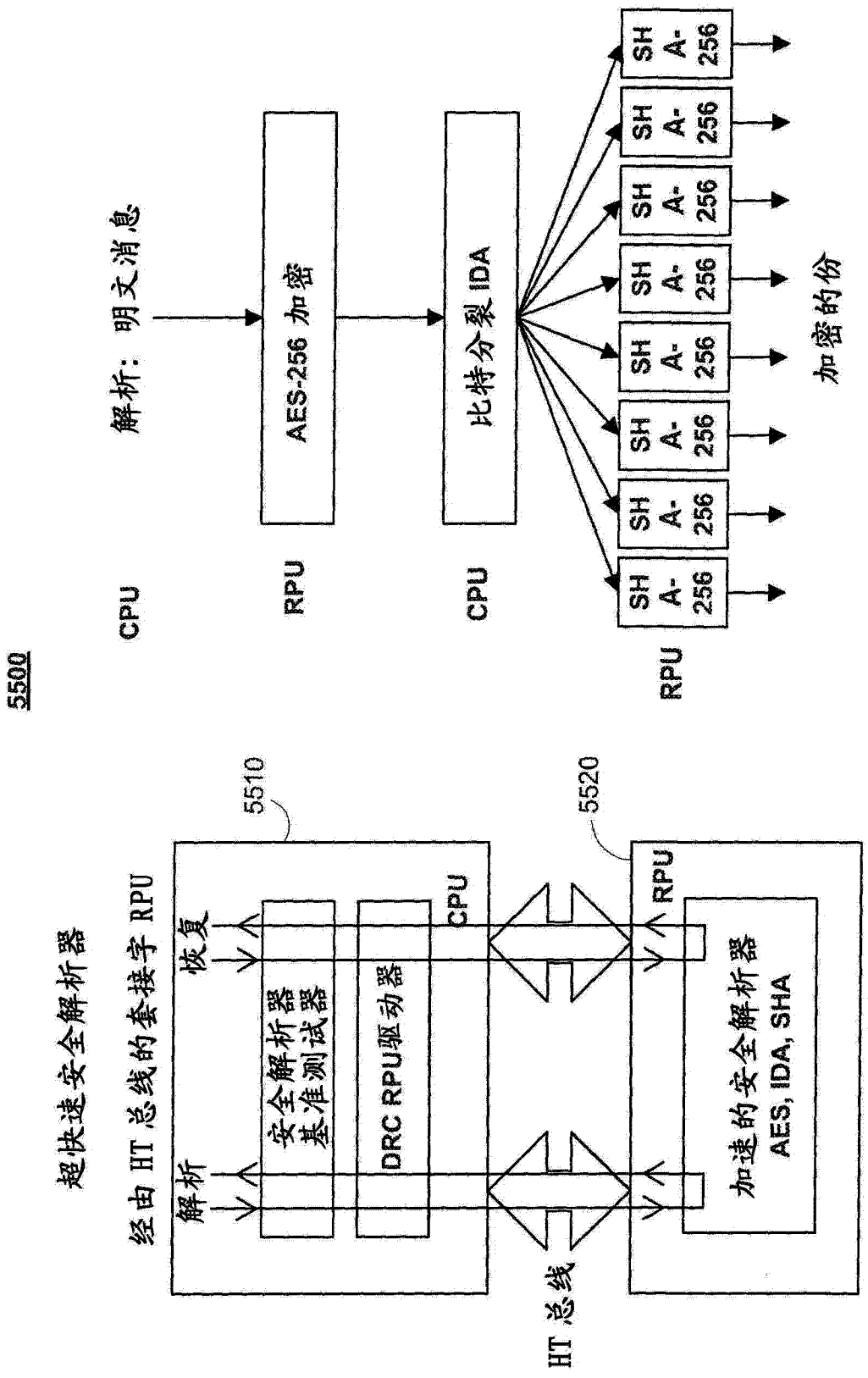

5500
CPU
解析：明文消息
RPU
AES-256 加密
CPU
比特分裂 IDA
SHA-256
SHA-256
SHA-256
SHA-256
SHA-256
SHA-256
SHA-256
加密的份
RPU
超快速安全解析器
经由 HT 总线的套接字 RPU
5510
恢复
解析
安全解析器基准测试器
DRC RPU驱动器
CPU
5520
RPU
加速的安全解析器 AES, IDA, SHA
HT 总线

图 55

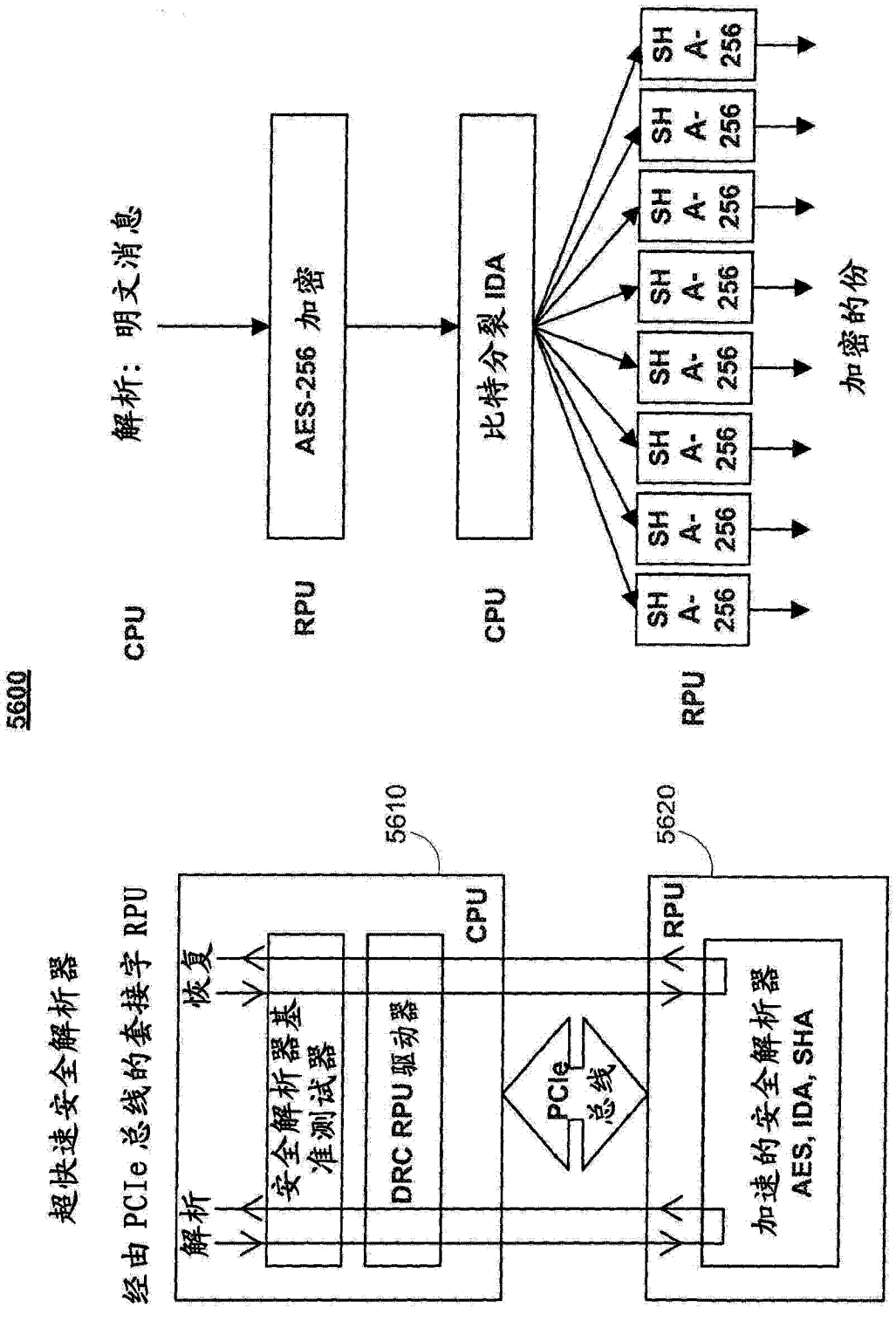

5600
CPU
解析：明文消息
RPU
AES-256 加密
CPU
比特分裂 IDA
RPU
SHA-256
加密的份
超快速安全解析器
经由 PCIe 总线的套接字 RPU
5610
CPU
恢复
解析
安全解析器基准测试器
DRC RPU 驱动器
PCIe 总线
5620
RPU
加速的安全解析器 AES, IDA, SHA

图 56

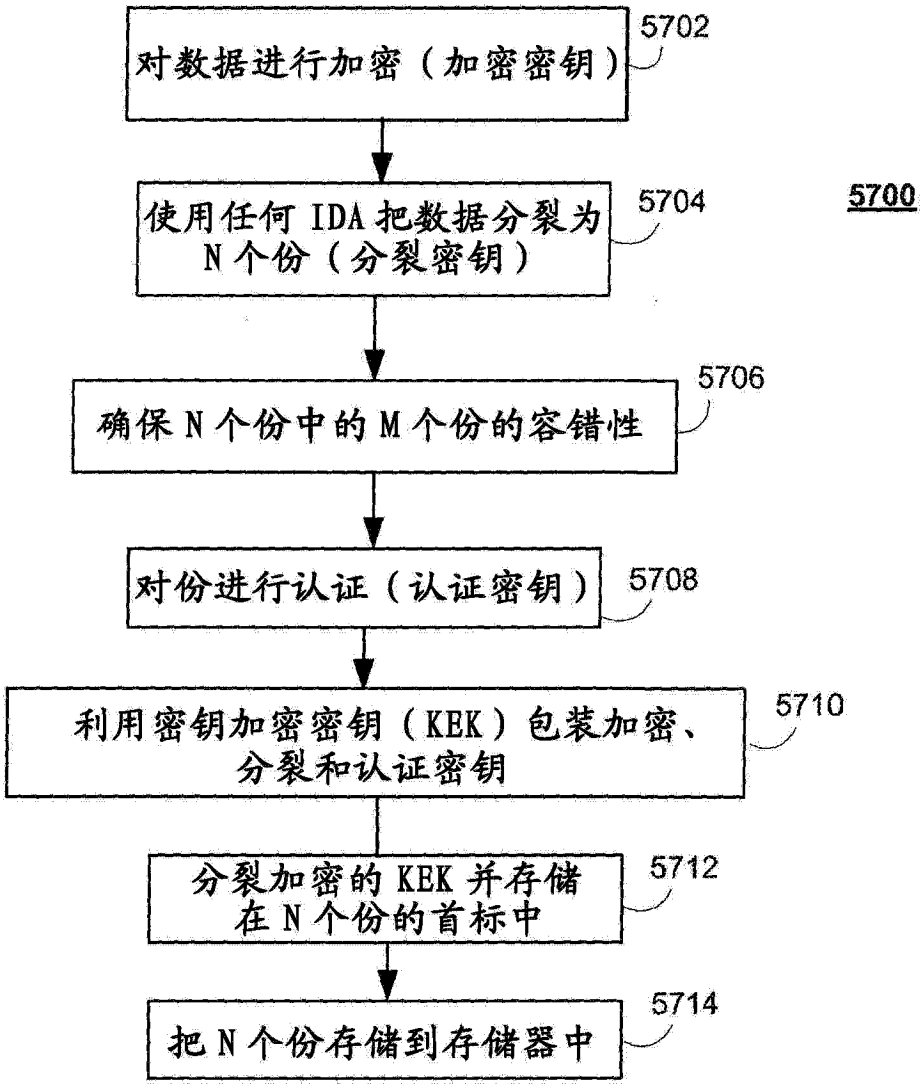

对数据进行加密（加密密钥）
5702
使用任何 IDA 把数据分裂为 N 个份（分裂密钥）
5704
5700
确保 N 个份中的 M 个份的容错性
5706
对份进行认证（认证密钥）
5708
利用密钥加密密钥（KEK）包装加密、分裂和认证密钥
5710
分裂加密的 KEK 并存储在 N 个份的首标中
5712
把 N 个份存储到存储器中
5714

图 57

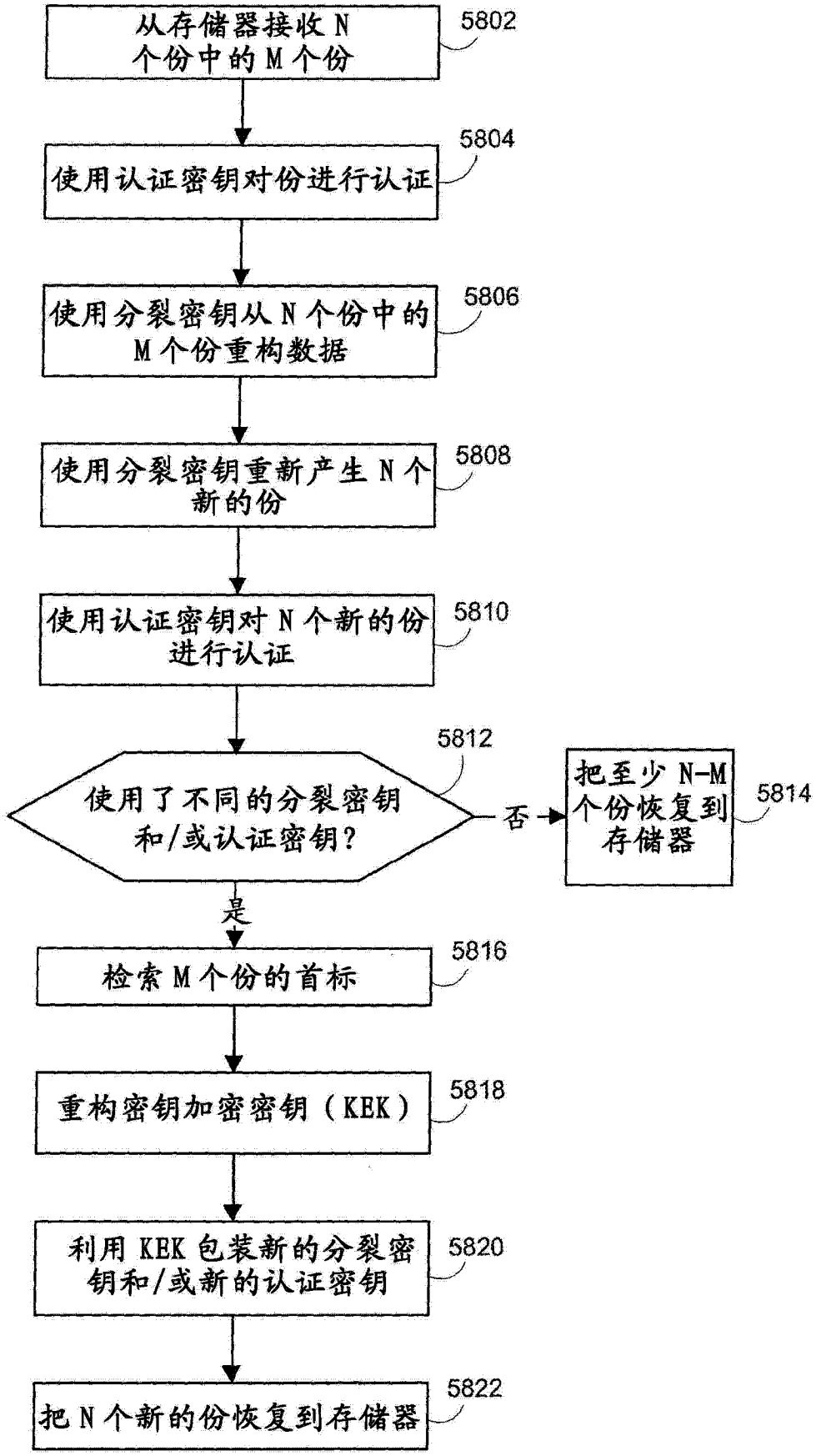

从存储器接收 N 个份中的 M 个份
5802
使用认证密钥对份进行认证
5804
使用分裂密钥从 N 个份中的 M 个份重构数据
5806
使用分裂密钥重新产生 N 个新的份
5808
使用认证密钥对 N 个新的份进行认证
5810
使用了不同的分裂密钥和/或认证密钥?
5812
否
把至少 N-M 个份恢复到存储器
5814
是
检索 M 个份的首标
5816
重构密钥加密密钥（KEK）
5818
利用 KEK 包装新的分裂密钥和/或新的认证密钥
5820
把 N 个新的份恢复到存储器
5822

图 58