(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 102770869 B
(45)授权公告日 2015.11.25

(21)申请号 201080064671.4
(22)申请日 2010.12.21
(30)优先权数据
12/646,059 2009.12.23 US
(85)PCT国际申请进入国家阶段日
2012.08.23
(86)PCT国际申请的申请数据
PCT/US2010/061517 2010.12.21
(87)PCT国际申请的公布数据
WO2011/079112 EN 2011.06.30
(73)专利权人 起元技术有限责任公司
地址 美国马萨诸塞州
(72)发明人 B. 拉森 R.A. 沙皮罗
(74)专利代理机构 北京林达刘知识产权代理事务所(普通合伙) 11277
代理人 刘新宇
(51)Int.Cl.
G06F 21/54(2013.01)
G06F 21/51(2013.01)

(56)对比文件
EP 1526429 A2,2005.04.27,
US 2009/0094673 A1,2009.04.09,
US 2007/0186112 A1,2007.08.09,
CN 101110097 A,2008.01.23,

审查员 汤明达

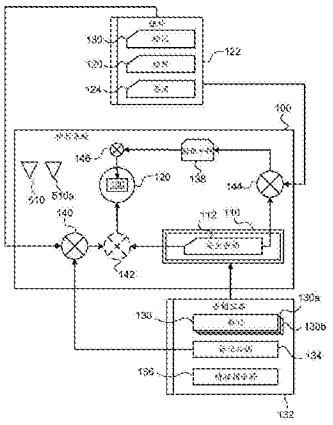
权利要求书5页 说明书11页 附图18页

(54)发明名称

计算资源的安全执行

(57)摘要

控制对计算特征的访问包括:准备用于由执行系统(100)执行的计算资源(120),该执行系统已经提供包含识别值(130)的主要描述符(132)并且已经将特征指示符(136)与该主要描述符关联;访问包含该识别值并且加密地分配给该计算资源(120)的次要描述符(138);以及基于该特征指示符(136)批准对该执行系统的计算特征(510,510a)的计算资源访问。



CN 102770869 B

1. 一种控制对计算特征的访问的方法，该方法包括：

准备用于由执行系统执行的第一计算资源，该执行系统已经被提供了主要描述符，每个主要描述符与相应的识别值相关联并且每个主要描述符与相应的特征指示符相关联；

访问包含第一识别值的次要描述符，该第一识别值与所述主要描述符中的第一主要描述符相关联，所述次要描述符被加密地分配给该第一计算资源；

接受来自第一计算资源以执行第二计算资源的指令；

将包含第一识别值的次要描述符的至少一个实例加密地分配给第二计算资源，包括将第二计算资源与所述次要描述符的实例组合，并且加密地标记所述组合；

基于与次要描述符的实例相关联的特征指示符批准对该执行系统的计算特征的第二计算资源访问；以及

在第二计算资源的执行期间，验证次要描述符的实例被加密地分配给该第二计算资源，以确定第二计算资源被授权仅访问根据与第一计算资源相关联的特征指示符使能的计算特征。

2. 如权利要求 1 所述的方法，其中该计算资源被加密。

3. 如权利要求 1 所述的方法，其中通过组合该第一计算资源和该次要描述符，并且加密地标记该组合，次要描述符被加密地分配给第一计算资源。

4. 如权利要求 1 所述的方法，其中加密地标记该组合包括：使用分配给该第一主要描述符的私有加密密钥加密从该组合得出的值。

5. 如权利要求 1 所述的方法，其中验证次要描述符的实例包括：使用分配给该第一主要描述符的公共加密密钥解密从该第二计算资源和该次要描述符的实例的组合得出的值。

6. 如权利要求 1 所述的方法，还包括：

在执行该第一计算资源之前生成次要描述符的第一实例；

在执行该第一计算资源之前生成次要描述符的第二实例；以及

比较次要描述符的两个实例以确定该第一计算资源是否被授权执行。

7. 如权利要求 6 所述的方法，还包括加密该次要描述符的第一实例。

8. 如权利要求 7 所述的方法，还包括加密该次要描述符的第二实例。

9. 如权利要求 6 所述的方法，其中该次要描述符的第一实例具有与该次要描述符的第二实例相同的第一数据值、以及与该次要描述符的第二实例不同的第二数据值。

10. 如权利要求 6 所述的方法，还包括：

将具有用于执行该第一计算资源的指令的第一容器分配给该第一计算资源和该次要描述符的第一实例；以及

将把用于执行该第一计算资源的指令表示为数据的第二容器分配给该第一容器和该次要描述符的第二实例。

11. 如权利要求 10 所述的方法，还包括加密地标记该第二容器。

12. 如权利要求 1 所述的方法，还包括：

接受来自该第一计算资源以访问第二计算资源的指令；

在系统上执行该第二计算资源；

将该次要描述符加密地分配给该第二计算资源；以及

基于该特征指示符对执行的第二计算资源提供对计算特征的访问。

13. 如权利要求 12 所述的方法，其中在第一计算资源已经完成执行后，该第二计算资源继续执行。

14. 如权利要求 12 所述的方法，其中该第一计算资源在第一系统上执行，并且该第二计算资源在第二系统上执行。

15. 如权利要求 12 所述的方法，其中将该次要描述符加密地分配给该第二计算资源包括：访问包含该次要描述符的实例的存储位置。

16. 如权利要求 1 所述的方法，还包括：

访问加密地分配给包含第二识别值的第二次要描述符的第二计算资源；

执行该第二计算资源；

访问加密地分配给包含第二识别值的第二主要描述符的第二特征指示符；以及

基于该第二特征指示符，对执行的第二计算资源提供对计算特征的访问。

17. 一种用于控制对计算特征的访问的系统，该系统包括：

第一计算机系统，配置为准备用于执行的第一计算资源，包括提供主要描述符，每个主要描述符与相应的识别值相关联；以及

第二计算机系统，配置为执行该计算资源，包括

将相应的特征指示符与相应的主要描述符关联；

访问包含第一识别值的次要描述符，该第一识别值与所述主要描述符的第一主要描述符相关联，该次要描述符被加密地分配给该第一计算资源；

接受来自第一计算资源以执行第二计算资源的指令；

将包含第一识别值的次要描述符的至少一个实例加密地分配给第二计算资源，包括将第二计算资源与所述次要描述符的实例组合，并且加密地标记所述组合；

基于与次要描述符的实例相关联的特征指示符批准对该执行系统的计算特征的第二计算资源访问；以及

在第二计算资源的执行期间，验证次要描述符的实例被加密地分配给该第二计算资源，以确定第二计算资源被授权仅访问根据与第一计算资源相关联的特征指示符使能的计算特征。

18. 一种用于控制对计算特征的访问的系统，该系统包括：

用于准备用于执行的第一计算资源的部件，包括提供主要描述符，每个主要描述符与相应的识别值相关联；以及

用于执行该计算资源的部件，包括

将相应的特征指示符与相应的主要描述符关联；

访问包含第一识别值的次要描述符，该第一识别值与所述主要描述符的第一主要描述符相关联，该次要描述符被加密地分配给该第一计算资源；

接受来自第一计算资源以执行第二计算资源的指令；

将包含第一识别值的次要描述符的至少一个实例加密地分配给第二计算资源，包括将第二计算资源与所述次要描述符的实例组合，并且加密地标记所述组合；

基于与次要描述符的实例相关联的特征指示符批准对计算特征的第二计算资源访问；以及

在第二计算资源的执行期间，验证次要描述符的实例被加密地分配给该第二计算资

源，以确定第二计算资源被授权仅访问根据与第一计算资源相关联的特征指示符使能的计算特征。

19. 如权利要求 17 所述的系统，其中该计算资源被加密。

20. 如权利要求 17 所述的系统，其中将该次要描述符加密地分配给该第一计算资源包括组合该第一计算资源与该次要描述符，并且加密地标记该组合。

21. 如权利要求 17 所述的系统，其中加密地标记该组合包括使用分配给该主要描述符的私有加密密钥加密从该组合得出的值。

22. 如权利要求 17 所述的系统，其中验证该次要描述符的实例包括：使用分配给该第一主要描述符的公共加密密钥解密从该第二计算资源和该次要描述符的实例的组合得出的值。

23. 如权利要求 17 所述的系统，其中执行计算机系统配置来：

在执行该第一计算资源之前生成次要描述符的第一实例；

在执行该第一计算资源之前生成次要描述符的第二实例；

比较描述符的两个实例以确定该第一计算资源是否被授权执行。

24. 如权利要求 23 所述的系统，其中准备用于执行的该第一计算资源包括加密该次要描述符的第一实例。

25. 如权利要求 24 所述的系统，其中准备用于执行的该第一计算资源包括加密该次要描述符的第二实例。

26. 如权利要求 23 所述的系统，其中该次要描述符的第一实例具有与该次要描述符的第二实例相同的第一数据值、以及与该次要描述符的第二实例不同的第二数据值。

27. 如权利要求 23 所述的系统，其中准备用于执行的该第一计算资源包括

将具有用于执行该第一计算资源的指令的第一容器分配给该第一计算资源和该次要描述符的第一实例，以及

将把用于执行该第一计算资源的指令表示为数据的第二容器分配给该第一容器和该次要描述符的第二实例。

28. 如权利要求 27 所述的系统，其中准备用于执行的该第一计算资源包括加密地标记该第二容器。

29. 如权利要求 17 所述的系统，其中准备用于执行的该第一计算资源包括

接受来自该第一计算资源以访问第二计算资源的指令；以及

执行该计算资源包括：

在系统上执行该第二计算资源，

将该次要描述符加密地分配给该第二计算资源，以及

基于该特征指示符，对执行的第二计算资源提供对计算特征的访问。

30. 如权利要求 29 所述的系统，其中在第一计算资源已经完成执行后，该第二计算资源继续执行。

31. 如权利要求 29 所述的系统，其中该计算资源在第一系统上执行，并且该第二计算资源在第二系统上执行。

32. 如权利要求 29 所述的系统，其中将该次要描述符加密地分配给该第二计算资源包括访问包含该次要描述符的实例的存储位置。

33. 如权利要求 18 所述的系统，其中该计算资源被加密。

34. 如权利要求 18 所述的系统，其中将该次要描述符加密地分配给该第一计算资源包括组合该第一计算资源与该次要描述符，并且加密地标记该组合。

35. 如权利要求 18 所述的系统，其中加密地标记该组合包括使用分配给该第一主要描述符的私有加密密钥加密从该组合得出的值。

36. 如权利要求 18 所述的系统，其中验证该次要描述符的实例包括：使用分配给该主要描述符的公共加密密钥解密从该第二计算资源和该次要描述符的实例的组合得出的值。

37. 如权利要求 18 所述的系统，还包括

用于在执行该第一计算资源之前生成次要描述符的第一实例的部件；

用于在执行该第一计算资源之前生成次要描述符的第二实例的部件；以及

用于比较描述符的两个实例以确定该第一计算资源是否被授权执行的部件。

38. 如权利要求 37 所述的系统，还包括用于加密该次要描述符的第一实例的部件。

39. 如权利要求 38 所述的系统，还包括用于加密该次要描述符的第二实例的部件。

40. 如权利要求 37 所述的系统，其中该次要描述符的第一实例具有与该次要描述符的第二实例相同的第一数据值、以及与该次要描述符的第二实例不同的第二数据值。

41. 如权利要求 37 所述的系统，还包括

用于将具有用于执行该第一计算资源的指令的第一容器分配给该第一计算资源和该次要描述符的第一实例的部件，以及

用于将把用于执行该第一计算资源的指令表示为数据的第二容器分配给该第一容器和该次要描述符的第二实例的部件。

42. 如权利要求 41 所述的系统，还包括用于加密地标记该第二容器的部件。

43. 如权利要求 18 所述的系统，还包括：

用于接受来自该第一计算资源以访问第二计算资源的指令的部件；

用于在系统上执行该第二计算资源的部件；

用于将该次要描述符加密地分配给该第二计算资源的部件；以及

用于基于该特征指示符，对执行的第二计算资源提供对计算特征的访问的部件。

44. 如权利要求 43 所述的系统，其中在第一计算资源已经完成执行后，该第二计算资源继续执行。

45. 如权利要求 43 所述的系统，其中该第一计算资源在第一系统上执行，并且该第二计算资源在第二系统上执行。

46. 如权利要求 43 所述的系统，其中将该次要描述符加密地分配给该第二计算资源包括访问包含该次要描述符的实例的存储位置。

47. 如权利要求 1 所述的方法，还包括：

将令牌附加到该第一计算资源，其中该令牌配置来在该第一计算资源被该计算特征执行的同时保持与该第一计算资源相关联的识别值。

48. 如权利要求 47 所述的方法，其中在执行该资源期间，该第一计算资源仅访问由与包括在该令牌中的识别值相关联的特征指定的该执行系统的特征。

49. 如权利要求 17 所述的系统，其中该第一计算机系统配置来将令牌附加到该第一计算资源，其中该令牌配置来在该第一计算资源被该计算特征执行的同时保持与该第一计算

资源相关联的识别值。

50. 如权利要求 49 所述的系统，其中在执行该第一计算资源期间，该资源仅访问由与包括在该令牌中的识别值相关联的特征指定的该执行系统的特征。

51. 如权利要求 18 所述的系统，其中用于准备所述计算资源的部件还包括用于将令牌附加到该第一计算资源的部件，其中该令牌配置来在该第一计算资源被该计算特征执行的同时保持与该第一计算资源相关联的识别值。

52. 如权利要求 51 所述的系统，其中在执行该资源期间，该第一计算资源仅访问由与包括在该令牌中的识别值相关联的特征指定的该执行系统的特征。

计算资源的安全执行

技术领域

[0001] 本描述涉及计算资源的安全执行。

背景技术

[0002] 诸如计算机软件的计算资源的供应商在提供这些资源给第三方时有风险。供应商可能希望限制对计算资源的特定元件的访问。在一些情况下，一些团体应当被批准访问特定元件，而其它的被拒绝访问。在其它情况下，资源的某些方面应当对所有团体屏蔽。恶意行为者可能试图规避对资源施加的限制。

发明内容

[0003] 在一个方面，通常，一种控制对计算特征的访问的方法包括：准备用于由执行系统执行的计算资源，该执行系统已经提供包含识别值的主要描述符并且已经将特征指示符与该主要描述符关联；访问包含该识别值并且加密地分配给该计算资源的次要描述符；以及基于该特征指示符批准对该执行系统的计算特征的计算资源访问。

[0004] 各方面可以包括以下特征的一个或多个。

[0005] 该计算资源被加密。

[0006] 加密地分配次要描述符给计算资源包括：组合该资源和该次要描述符，并且加密地标记该组合。

[0007] 加密地标记该组合包括：使用分配给该主要描述符的私有加密密钥加密从该组合得出的值。

[0008] 该方法还包括在计算资源的执行期间验证加密地分配给该计算资源的次要描述符。

[0009] 验证次要描述符包括：使用分配给该主要描述符的公共加密密钥解密从该计算资源和该次要描述符的组合得出的值。

[0010] 该方法还包括：在执行该计算资源之前生成次要描述符的第一实例；在执行该计算资源之前生成次要描述符的第二实例；以及比较描述符的两个实例以确定该计算资源是否被授权执行。

[0011] 该方法还包括加密该次要描述符的第一实例。

[0012] 该方法还包括加密该次要描述符的第二实例。

[0013] 该次要描述符的第一实例具有与该次要描述符的第二实例相同的第一数据值、以及与该次要描述符的第二实例不同的第二数据值。

[0014] 该方法还包括：将具有用于执行该计算资源的指令的内部容器(container)分配给该计算资源和该次要描述符的第一实例；以及将具有用于执行该计算资源的指令的、表示为数据的外部容器分配给该内部容器和该次要描述符的第二实例。

[0015] 该方法还包括加密地标记该外部容器。

[0016] 该方法还包括：接受来自该计算资源的指令以访问第二计算资源；在系统上执行

该第二计算资源；将该次要描述符加密地分配给该第二计算资源；以及基于该特征指示符对执行的第二计算资源提供对计算特征的访问。

[0017] 在第一计算资源已经完成执行后，该第二计算资源继续执行。

[0018] 该计算资源在第一系统上执行，并且该第二计算资源在第二系统上执行。

[0019] 将该次要描述符加密地分配给该第二计算资源包括：访问包含该次要描述符的实例的存储位置。

[0020] 该方法还包括：访问加密地分配给包含第二识别值的第二次要描述符的第二计算资源；执行该第二计算资源；访问加密地分配给包含第二识别值的第二主要描述符的第二特征指示符；以及基于该第二特征指示符，对执行的第二计算资源提供对计算特征的访问。

[0021] 在另一方面，通常一种用于控制对计算特征的访问的系统包括：开发计算机系统，配置为准备用于执行的计算资源，包括提供包含识别值的主要描述符；以及执行计算机系统，配置为执行该计算资源。执行该计算资源包括：将特征指示符与该主要描述符关联；访问包含该识别值并且加密地分配给该计算资源的次要描述符；以及基于该特征指示符批准对该执行系统的计算特征的计算资源访问。

[0022] 在另一方面，通常一种用于控制对计算特征的访问的系统包括：用于准备用于执行的计算资源的部件，包括提供包含识别值的主要描述符；以及用于执行该计算资源的部件。执行该计算资源包括：将特征指示符与该主要描述符关联；访问包含该识别值并且加密地分配给该计算资源的次要描述符；以及基于该特征指示符批准对该执行系统的计算特征的计算资源访问。

[0023] 在另一方面，通常一种计算机可读介质存储用于控制对计算特征的访问的计算机程序。该计算机程序包括用于使得计算机执行以下的指令：准备用于由执行系统执行的计算资源，该执行系统已经提供包含识别值的主要描述符并且已经将特征指示符与该主要描述符关联；访问包含该识别值并且加密地分配给该计算资源的次要描述符；以及基于该特征指示符批准对该执行系统的计算特征的计算资源访问。

[0024] 各方面可以包括以下优点的一个或多个。

[0025] 通过加密地分配描述符给计算资源，该分配在没有授权的情况下不能改变。通过基于分配给描述符的特征指示符提供访问给计算特征，计算特征在没有授权的情况下不能访问。

[0026] 从下面的描述以及权利要求中，本发明的其它特征和优点将变得明显。

附图说明

[0027] 图 1A 是执行系统和相关组件的概况。

[0028] 图 1B 是执行系统和相关组件的概况。

[0029] 图 2A 是数据流图的方块图。

[0030] 图 2B 是执行系统的方块图。

[0031] 图 3A 是密钥容器的方块图。

[0032] 图 3B 是组件存储器的方块图。

[0033] 图 4A 是用于生成标记令牌（brand token）的执行系统的部分。

[0034] 图 4B 是用于生成执行脚本的执行系统的部分。

[0035] 图 4C 是用于生成脚本容器的执行系统的部分。

[0036] 图 5A 是用于访问脚本容器的执行系统的部分。

[0037] 图 5B 是用于访问执行脚本的执行系统的部分。

[0038] 图 5C 是用于使能和禁用特征的执行系统的部分。

[0039] 图 6A 是用于生成另一标记令牌的执行系统的部分。

[0040] 图 6B 是用于访问另一标记令牌的执行系统的部分。

[0041] 图 7 是用于分配标记的过程的流程图。

[0042] 图 8 是用于访问特征指示符的过程的流程图。

[0043] 图 9 是用于执行可执行组件的过程的流程图。

[0044] 图 10 是用于提供对特征的访问的过程的流程图。

具体实施方式

[0045] 概述

[0046] 当计算资源在计算系统上执行时，该资源可以以特定方式操作，并且访问计算系统的特定特征。执行该资源的恶意行为者可能使用安全缺陷来克服由该资源的提供者或授权方对执行施加的任何限制。例如，恶意行为者可能试图修改资源本身以便以使用原始没有被设计为使用的资源的系统特征的方式操作。因此，计算系统可能保护相关计算资源的完整性以确保没有行为者能够在没有授权的情况下修改或篡改资源。在一些情况下，资源的完整性还可能希望防止任何行为者访问资源的底层结构或设计，即使该行为者被允许执行该资源。

[0047] 此外，计算资源的授权可以是不同于在其上执行资源的计算系统的授权的实体。例如，为计算系统提供系统软件的系统授权可以允许资源授权开发软件模块，作为用于由系统软件使用的资源。在该情况下，系统授权可能希望允许资源授权访问系统的一些特征，而不允许访问其它特征。因此，系统可以具有这样的机构，用于提供对可用资源的特定集合的计算资源访问，并且还提供对可用特征的其它各自集合的其它计算资源访问，其中各集合可以相互重叠或不重叠。

[0048] 复杂计算系统可以与许多不同的计算资源相关联，所有这些计算资源可以以这些计算资源的多个实例的形式执行，并且可以相互作用。例如，系统可以执行一个计算资源，其又导致第二计算资源的执行。同时，第三计算资源可以导致第二计算资源的另一实例的执行。这种复杂计算系统可以具有这样的机构，用于保持跟踪计算资源的哪些实例与计算资源的其它实例相关联。此外，系统可以仅基于计算资源(如操作者或用户执行的第一资源)之一的访问特权，使用该信息来允许访问特定特征。在该情况下，与第一计算资源相关联的所有执行实例将只能访问第一计算资源可访问的特征。

[0049] 图 1A 是示出相关联的主要元件之间的相互关系的计算系统的一个示例性安排的方块图。执行系统 100 包括执行模块 110，其处理可执行组件 120。可执行组件 120 包括能够执行的任何分立计算组成，如计算机程序、软件代码、或其它种类的可执行元件。

[0050] 可执行元件 120 可以存储在组件容器 122 中。可选地，组件存储器 122 中的可执行组件 120 可以使用对执行系统 100 可用的对称遮蔽密钥 112 加密。组件容器 122 还具有与可执行组件 120 有关的其它信息，包括标记 130 和电子组件签名 124。标记 130 是用于

组件的标识，指示特征 510，510a 哪个可用于该组件。当其它组件用相同标记标识时，与标记 130 相关联的组件具有对于相同特征 510，510a 的通路。特征 510，510a 可以是执行系统 100 的不同组件。例如，特征 510，510a 可以对应于命令、指令、软件处理、硬件设备、数据文件、一些其它种类的元件、或这些的任何组合。

[0051] 执行系统 100 能够访问密钥容器 132，其还包含一个或多个标记 130，130a，130b。执行模块 110 接收密钥容器 132 作为输入。作为对系统的其它部分的输入的系统的一些部分的关系在图中用箭头表示，其是本描述中遍及附图使用的惯例。密钥容器 132 中的每个标记 130 与标记公钥 134 相关联。执行系统 100 使用标记公钥 134 来验证组件容器 122 中的标记 130 有效。每个标记 130 与特征指示符 136 相关联，该特征指示符描述与标记 130 相关联的执行系统 100 的特征 510，510a。与标记 130 相关联的组件 120 只能访问特征指示符 136 描述的特征 510，510a。

[0052] 执行系统 100 通过访问组件容器 122 执行可执行组件 120。执行系统 100 具有签名验证模块 140，其确定组件签名 124 是否有效。签名验证模块 140 使用标记公钥 134 执行该有效性检验。如果可执行组件 120 被加密，则签名验证模块 140 将该验证的组件容器 122 传递给解密模块 142，以使用遮蔽密钥 112 解密。然后使得可执行组件 120 可用于执行系统 100。

[0053] 执行系统 100 还具有标记令牌化模块 144。该标记令牌化模块 144 准备包含与组件 120 相关联的标记 130 的标记令牌 138。标记令牌 138 用于在组件执行时维持与组件 120 相关联的标记 130。

[0054] 当执行模块 110 执行组件 120 时，标记令牌化模块 144 安全地将标记令牌 138 附加到组件 120。在执行期间，组件 120 只能访问与标记令牌 138 中的令牌 130 相关联的特征指示符 136 描述的执行系统 100 的特征。

[0055] 图 1B 是计算系统的另一示例安排的方块图。在图 1B 中，标记令牌 138 还用于将标记 130 与其它可执行组件 120a 相关联。如果可执行组件 120 导致另一可执行组件的实例 120a 的执行，则执行环境模块 146 将允许标记令牌 138 附加到其它可执行组件 120a。因此，其它可执行组件 120a 将只能访问与标记令牌 138 中的令牌 130 相关联的特征指示符 136 描述的特征，即使其它可执行组件 120a 在其组件容器 122a 中没有令牌 130。标记令牌 138 还可以使用执行环境模块 146a 附加到不可执行组件，如数据组件 121。执行环境模块在图中显示为多个实例 146，146a，146b，但是取决于实现，可能只有一个实例是必要的。

[0056] 可执行组件 120，120a 之一还可以访问另一执行系统 100a（如通过网络 104 访问的次要或远程系统）上的可执行组件 120b。另一执行系统 100a 也可以具有能够访问遮蔽密钥 112 的执行模块 110a，并且能够访问包含标记 130 的密钥容器 132a。执行环境模块 146b 可以允许标记令牌 138 附加到另一执行系统 100a 处理的可执行组件 120b。因此，可执行组件 120b 将在标记 130 下操作，即使组件容器 122b 不包含标记 130。可执行系统 100 可以根据标准网络协议、标准远程程序调用协议、定制协议、其它种类的协议、或这些的任何组合，与另一执行系统 100a 交互。

[0057] 在一些实现方式中，可执行组件 120 使用定向图形表示的“数据流图形”实现，图形中的顶点表示组件（数据组件 121 或可执行组件 120），并且图形中的定向链路或“边缘”表示各组件之间的数据流。图形是模块化实体。每个图形可以由一个或多个其它图形构

成，并且特定图形可以是更大图形中的组件。图形开发环境（GDE）提供用于指定可执行图形并且定义用于图形组件的参数的用户接口。实现这样的基于图形的计算的系统在美国专利5,966,072、EXECUTING COMPUTATIONS EXPRESSED AS GRAPHS中描述，在此通过应用并入其全部内容。

[0058] 参考图2A，数据流图形201的示例包括输入数据集组件202，其提供要由数据流图形201的可执行组件204a-204j处理的数据的集合。例如，数据集组件202可以包括与数据库系统相关联的数据记录或与事务处理系统相关联的事务。每个可执行组件204a-204j与由整个数据流图形201定义的计算的部分相关联。工作元件（例如，来自数据集合的各个数据记录）进入组件的一个或多个输入端口，并且输出工作元件（其在一些情况下是输入工作元件或输入工作元件的处理后版本）典型地留下组件的一个或多个输出端口。在图形201中，来自组件204e，204g和204j的输出工作组件存储在输出数据集组件202a-202c中。

[0059] 图2B示出示例性执行系统100，其耦合到存储系统210，并且配置为使用一个或多个安全程序（如数据流图）处理从要处理的数据源212接收的数据。数据源212可以包括一个或多个数据源（如存储设备或到再现数据流的连接），其每个可以以多种存储格式（例如，数据库表、电子表格文件、平坦文本文件（flat text file）或大型机使用的本地格式）的任何存储数据。执行系统100包括安全层214和操作系统216，安全层214使用存储系统210中存储的在此描述的描述符（例如，密钥容器）提供安全。在一些实现方式中，安全层214是操作系统216的一部分。

[0060] 执行系统100从数据源212接收数据用于处理。提供数据源212的存储设备对于执行系统100可以是本地的，例如，存储在连接到运行操作系统216的计算机的存储介质（例如，硬盘驱动器218）上，或者可以远离执行系统100，例如，容纳在通过远程连接与运行操作系统216的计算机通信的远程系统（例如，大型机220）上。

[0061] 数据存储系统210可以接收已经由开发系统230中的开发者234开发的安全程序232。开发者234能够开发用于在执行系统100中执行的安全程序，其中开发系统230配置为如在此描述的加密地分配描述符。在一些实现方式中，开发系统230是用于开发作为数据流图形的应用的系统（例如，使用上面描述的GDE）。除了在此描述的特征外，例如在美国公开No.2007/0011668、名为“Managing Parameters for Graph-Based Applications”中更详细地描述了这样的系统的其它特征，在此通过应用并入。

[0062] 开发系统230和执行系统100每个可以容纳在在适当操作系统（如UNIX操作系统）控制下的一个或多个通用计算机系统上，并且存储系统210可以包括与计算机系统集成或与其通信的存储介质。例如，计算机系统可以包括多节点并行计算环境，其包括使用多个中央处理单元（CPU）的计算机系统的配置，本地的（例如，诸如SMP计算机的多处理器系统）或本地分布的（例如，耦合为集束的多处理器或MPP），或者远程分布的（例如，经由局域网（LAN）和/或广域网（WAN）耦合的多处理器），或者它们的任何组合。

[0063] 2. 容器（Container）

[0064] 如图3A所示，密钥容器132可以包括多个标记（brand）130、130a和130b。一个可执行组件120可以根据一个标记130a操作，并且另一个可以根据另一标记130b操作。执行系统100处理可执行组件120，如果与该组件相关联的标记130在密钥容器132中可用。在一些示例中，密钥容器132将是数据文件、存储器位置或其它种类的容器。

[0065] 每个标记 130 具有对应的公钥 134，其允许签名验证模块 140 确定可执行组件 120 和标记 130 之间的关联是否有效。每个标记 130 还具有过期指示符 135，其允许标记管理者（authority）设置标记 130 有效的时间段。此外，每个标记 130 具有特征指示符 136，其描述可用于根据该标记 130 操作的可执行组件 120 的执行系统 100 的特征 510。

[0066] 如图 3B 所示，组件容器 122 具有与组件 300 相关的元件。组件 300 例如可以是可执行组件 120。在一些示例中，可以使用其它类型的组件 300，如具有可执行组件 120 使用的信息而自身不可执行的数据组件 121。在一些示例中，组件容器 122 可以是数据文件、存储器位置或其它类型的容器。

[0067] 组件 300 可选地被加密以屏蔽组件的结构或内容被未授权地观看或分析。加密模块 342 使用屏蔽密钥 112 来加密组件 300。加密模块 342 可以基于标准加密协议，如数据加密标准（DES）、先进加密标准（AES）、其它标准加密协议或这些的组合。加密模块 342 可以单独使用定制的加密协议或与标准加密协议结合使用。

[0068] 组件容器 122 包含与组件 300 相关联的标记 130。标记 130 用于分配一组特征给组件 300。

[0069] 组件容器 122 具有组件签名 124，用于检查组件容器 122 的内容的有效性。签名生成模块 340 通过根据电子签名协议处理组件 300 和标记 130，创建组件签名。该协议可以是用于电子签名的标准密码协议，如 RSA、数字签名算法（DSA）、椭圆曲线协议、或其它种类的密码协议。签名生成模块 340 可以单独使用定制的电子签名协议或与标准加密协议结合使用。

[0070] 组件签名 124 使用标记私钥 334 生成。标记私钥 334 可用于组件容器 122 的管理者，但在处理组件 300 的执行系统 100 处不可用。执行系统 100 只能验证标记 130 和组件 300 还没有改变，因为生成了组件签名 124。因此，组件签名 124 是将标记 130 与组件 300 相关联以及确保组件 300 的完整性的加密安全部件。

[0071] 此外，组件容器 122 具有容器号 302，其是用于该组件容器 122 的唯一标识符。其还具有容器格式 304，容器格式 304 指示构成该组件容器 122 的数据的特定安排。组件容器 122 还具有组件类型 306，组件类型 306 指示构成包含在内的组件 300 的数据的特定安排，包括该组件 300 是否被加密。

[0072] 3．标记令牌化

[0073] 如上所述，执行系统 100 将标记令牌 138 附加到可执行组件 120、120a 和 120b，以在执行前将标记 130 与每个组件相关联。执行系统 100 在每次之前采取这些令牌化步骤，以使得该关联安全，使得如果恶意行为者试图拦截执行处理以改变或替换标记 130，则可以检测到该恶意行为。

[0074] 如图 4A 所示，执行系统 100 使用标记令牌的多个实例 138a、138b。每个实例从组件的组将容器生成，在该示例中，该组件容器是可执行组件 120 的组件容器 122。

[0075] 执行系统 100 使用签名验证模块 140 验证组件签名 124。签名验证模块 140 使用标记公钥 134 来确定在组件容器 122 创建后的某个时间，可执行组件 120 或标记 130 是否已经被修改或者另外篡改。如果可执行组件 120 或标记 130 的任一已经被篡改，则签名验证模块 140 将检测差异，并且拒绝组件容器 122。

[0076] 标记令牌化模块 144、144a 使用组件容器 122 的内容来生成标记令牌实例 138a、

138b。示出了标记令牌化模块的两个实例，但是在一些实现方式中，可能只需要一个实例。标记令牌化模块144、144a将来自组件容器的标记130放入标记令牌的每个实例138a、138b中。标记令牌化模块144还使用组件类型306来生成指示可执行组件120是否加密的标志406。

[0077] 此外，随机数生成器410对于标记令牌的每个实例138a、138b创建不同的要素（salt）412。因此，标记令牌的每个实例138a、138b将包含与其它实例不同的唯一随机值。

[0078] 如图4B所示，标记令牌化模块144生成执行脚本420。执行脚本420包含用于执行可执行组件的实例120的指令，并且可以采取若干形式的任一。执行脚本420可以是一组解释或编译的命令，向执行模块110指示适当地处理可执行组件120所需的具体信息。

[0079] 可执行组件120与执行脚本420相关联。例如，执行脚本420然后可以包含构成可执行组件120自身的指令，或者可替代地它可以具有对可执行组件120的指针或引用。执行脚本420中的指令之一可以访问该指针或引用。

[0080] 如果可执行组件120在放入组件容器122之前被加密，则解密模块142使用安全密钥112处理解密可执行组件120。

[0081] 标记令牌的实例之一138a与执行脚本420一起使用。标记令牌化模块144通过生成执行号408并将其放入标记令牌实例138a中，完成标记令牌实例138a。相同执行号408应用于在用于可执行组件120的该执行的令牌化处理中使用的标记令牌的所有实例138。标记令牌化模块144保持跟踪它分配给标记令牌的该实例138a的执行号408，并且将使用该相同执行号408用于其它实例。

[0082] 一旦标记令牌实例138a完成，加密模块432使用安全密钥112加密标记令牌实例。标记令牌化模块144然后将加密的标记令牌438a放入执行脚本420中。如同可执行组件120，执行脚本420然后包含构成加密的标记令牌438a自身的数据，或者可替代地它可以具有对加密的标记令牌438a的指针或引用。

[0083] 如图4C所示，标记令牌化模块144生成脚本容器430并将执行脚本420放入其中。在一些实现方式中，脚本容器430是数据文件或数据结构。标记令牌化模块144还通过增加用于标记令牌的第一实例138a的相同执行号408，完成标记令牌的第二实例138b。标记令牌的第二实例138b利用加密模块342加密，并且加密的标记令牌438b放入脚本容器430中。

[0084] 令牌化签名生成模块450基于脚本容器430生成脚本容器签名424。令牌化签名生成模块450使用在执行系统100可用的令牌器（tokenizer）私钥452。当组件执行开始时，脚本容器签名424提供确保脚本容器430的有效性的机制。

[0085] 令牌化签名生成模块450以与其它地方描述的其它签名生成模块相似的方式操作，并且通过根据电子签名协议处理脚本容器430来创建脚本容器424。该协议可以是用于电子签名的标准密码协议，如RSA、数字签名算法（DSA）、椭圆曲线协议、或其它种类的密码协议。令牌化签名生成模块450可以单独使用定制的电子签名协议或与标准加密协议结合使用。在脚本容器430建立之后，可执行组件120准备用于安全执行。

[0086] 4．标记后执行

[0087] 如图5A所示，执行模块110接收包含可执行组件120的脚本容器430。执行模块110执行沿着脚本容器430的内容所需的步骤，该脚本容器已经构造为允许可执行容器120

的安全执行。

[0088] 执行模块 110 使用容器签名验证模块 550 沿着脚本容器签名 424。容器签名验证模块 550 使用容器公钥 454 来确定在脚本容器 430 创建后的某个时间，脚本容器 430 是否已经被修改或者另外篡改。容器签名验证模块 550 将检测任何差异，并且拒绝脚本容器 430。执行模块 110 还使用屏蔽密钥 112，从验证的脚本容器 430 提取加密的标记令牌 438b，并且应用解密模块 142。

[0089] 如图 5B 所示，执行模块 110 获取之前由解密模块 142 解密的标记令牌的解密实例 138b。执行模块 110 还执行之前从验证的脚本容器 430 获取的执行脚本 420 中包含的指令。执行脚本 420 包含可执行组件 120 以及其它的加密标记令牌 438a。执行模块 110 对该加密的标记令牌 438a 应用解密模块 142。

[0090] 如图 5C 所示，令牌验证模块 148 比较从执行脚本 420 提取的标记令牌的第一实例 138a 的内容和从脚本容器 430 提取的标记令牌的第二实例 138b。除了随机要素 412a、412b，由标记令牌化模块 144 创建的标记令牌的两个实例 138a、138b 将具有相等内容。如果要素 412a、412b 不同，但是包括标记 130 和执行号 408 的另外的内容相同，则令牌验证模块 148 将提供验证的标记 130 给执行模块 110。

[0091] 因为标记令牌的一个实例 138a 利用执行脚本 420 到达执行模块 110，并且标记令牌的另一实例 138b 利用脚本容器 430 到达，所以标记令牌的两个实例 138a、138b 可以用于沿着可执行组件 120 的真实性及其与标记 130 的关联。任何特定的执行脚本 420 不能通过将其放入新的脚本容器 430 内而重新使用，因为新的脚本容器 430 将没有具有匹配执行号 408 的标记令牌 138。来自执行脚本 420 的标记令牌实例 138a 不能用于脚本容器 430，因为来自执行脚本 420 和脚本容器 430 的标记令牌实例将没有不同的随机要素 412a、412b，如执行模块 110 期待的。此外，用于脚本容器 430 的新的标记令牌实例不能通过未授权行为者从执行脚本 420 内的标记令牌实例 138a 生成，因为标记令牌实例 138a 利用安全密钥 112 加密为加密标记令牌 438a。

[0092] 当通过令牌验证模块 148 将表示有效标记 130 的标记令牌实例 138a、138b 通知给执行模块 110 时，执行模块 110 访问密钥容器 132 以访问对应于标记令牌实例 138a、138b 中的标记 130 的、与密钥容器中的标记 130 相关联的过期指示符 135 和特征指示符 136。执行模块 110 使用特征指示符 136 来允许特征组 500 中的一些特征 510a、510b、510c、510d 可用于可执行组件，并且禁用特征组 500 中的其它特征 512a、512b。

[0093] 执行模块 110 访问来自执行脚本 420 的可执行组件 120，并且允许它仅使用根据标记 130 的特征指示符 120 允许的特征 510 执行。

[0094] 5．保持标记后执行环境

[0095] 与标记 130 相关联的可执行组件 120 可以启动其它可执行组件 120a 的执行。例如，这些其它可执行组件 120a 可能不与它们的容器文件 122a 中的标记 130 直接相关联，而是替代地在执行期间与标记 130 相关联。如果其它特征 512a、512b 的执行在一些其它环境中启动，则即使其它可执行组件 120a 已经访问其它特征 512a、512b，其它可执行组件 120a 也将只访问根据与初始可执行组件 120 相关联的标记 130 的特征指示符 136 使能的特征 510。因此，使用其它可执行组件 120a，初始可执行组件 120 被限制访问禁用特征 512a、512b。

[0096] 此外，即使在初始可执行组件 120 已经完成执行之后，其它可执行组件 120a、120b 也可以启动另外的可执行组件（如可执行组件 120b）。执行系统 100 保持跟踪与初始可执行组件 120 相关联的标记 130，使得将根据该标记 130 的特征指示符 136 确定可访问这些组件 120、120a、120b 的全部的特征 510。

[0097] 在一些实现方式中，执行系统 100 处理多个可执行组件 120，并且将标记 130 与这些可执行组件 120 相关联。例如，执行系统 100 可以通过增加用于保留属于一个可执行组件 120 的信息用于其它可执行组件 120a 的机构，在一个可执行组件 120 已经完成执行后保留该信息。此外，该机构不允许其它组件或恶意行为者访问或改变保留的信息，该机构也不具有增加不必要开销或用作不安全点的额外元件。

[0098] 如图 6A 所示，执行系统 100 具有执行环境模块 146，其接收来自令牌验证模块 148 的关于用于验证可执行组件 120 的标记 130 的标记令牌的实例 138a、138b 的内容的信息。执行环境模块 146 生成瞬态容器 622 和指向瞬态容器 622 的环境引用 610。环境引用 610 存在于可执行组件 130 的执行环境 605 的环境中。在一些示例中，执行环境 605 包含属于可执行组件 120 的执行的存储器位置、变量、数据文件或其它种类的信息，或者可以包含这些种类的信息的任何组合。

[0099] 瞬态容器 622 不是全部地对执行系统 100 可用。替代地，瞬态容器 622 只通过访问环境引用 610 可用。接着，环境引用 610 只在可执行组件 130 的执行环境 605 中可用。如果可执行组件 120 启动另一可执行组件 120a 的执行，则另一可执行组件 120a 被允许访问环境引用 610，因为它在与负责启动它执行的可执行组件 120 相同的执行环境 605 中操作。在一些示例中，瞬态容器 622 可以是在不使用环境引用 610 在情况下，使用执行系统 100 的普通文件系统不可访问的数据文件。瞬态容器 622 可用于另一执行系统 100a 处理的可执行组件 120b。

[0100] 执行环境模块 146 使用从令牌验证模块 148 获取的信息生成标记令牌的新的实例 138c。该标记令牌的实例 138c 具有与标记令牌的其它实例 138a、138b 相同的标记 130、标志 406 和执行号 408，并且还具有从随机数生成器 410 获取的新的随机要素 412c。执行环境模块 146 将标记令牌的新的实例 138c 放入瞬态容器 622 中。

[0101] 图 6B 示出具有可执行组件 120a 但不具有标记 130 的组件容器 122a。执行模块 110 通过访问可执行组件的执行环境 605 准备可执行组件 120a。如果该可执行组件 120a 的该执行通过另一可执行组件 120 启动，则该可执行组件 120a 将在与另一可执行组件 120 相同的执行环境 605 内操作，并且因此对瞬态容器 622 的环境引用 610 将可用。执行模块 110 访问瞬态容器 622 内的标记令牌实例 138c，并且获取标记令牌中的标记 130。标记令牌实例 138c 还可以与相同的标记令牌的其它实例 138a、138b 比较以检查它的有效性。例如，如果恶意行为者已经将标记令牌的其它实例 138a、138b 之一复制到瞬态容器 622 中，则瞬态容器 622 中的标记令牌的实例 138c 将具有与其他实例中相同的随机要素 412a、412b，因此可以识别为有效。

[0102] 标记令牌化模块 144、144a 激活并从执行模块 110 获取标记 130。标记令牌化模块在图中显示为两个实例 144、144a，但是取决于实现，可能只有一个实例是必要的。标记令牌化模块 144、144a 然后构造标记令牌的新的实例 138d、138e，其包含从新的可执行组件 120a 的组件容器 122a 中的组件类型 306 确定的标记 130、标志 406a 以及从随机数生成器

410 获取的新的要素 412d、412e。这些标记令牌的新的实例 138d、138e 然后被标记令牌化模块 144、144a 用于准备如上面参考图 4A 到 4C 描述的可执行组件 120。

[0103] 6. 过程

[0104] 图 7 示出用于加密分配标记 130 给可执行组件 120 的示例性过程 700 的流程图。该过程基于可执行组件 120 和标记 130 生成 702 用于组件容器 122 的数字签名。该过程验证 704 该数字签名，并分配 706 标记 130 给标记令牌的多个实例 138a、138b。该过程加密 708 标记令牌的一个实例 138a，并且将其附加 710 到执行脚本 420。该过程加密 712 标记令牌的另一实例 138b，并且将加密的标记令牌与执行脚本 420 一起附加到脚本容器 430。该过程基于脚本容器 430 生成 716 用于脚本容器 430 的数字签名。

[0105] 图 8 示出用于访问分配给标记 130 的特征指示符 136 的示例性过程 800 的流程图。该过程验证 802 数字签名并访问脚本容器 430。该过程解密 804 标记令牌的实例 138b，并且访问 806 脚本容器 430 中的执行脚本 420。该过程解密 808 标记令牌的另一实例 138a，并且访问 810 执行脚本 420 中的可执行组件 120。该过程验证 812 标记令牌的两个实例 138a、138b。使用标记令牌实例 138a、138b 中的标记 130，该过程获取 814 分配给密钥容器 132 中的标记 130 的特征指示符 136。

[0106] 图 9 示出用于在执行系统 100 上执行可执行组件 120 的示例性过程 900 的流程图。该过程生成 902 瞬态容器 622，并且生成 904 指向瞬态容器 622 的环境引用。该过程生成 906 包含与可执行组件 120 相关联的标记 130 的标记令牌的实例 138c。该过程分配 908 标记令牌的实例 138c 给瞬态容器 622。该过程移除 910 除了环境引用 622 以外的任何其他对瞬态容器 622 的引用。I 过程执行 912 可执行组件 120，并且将环境引用 610 放入可执行组件 120 的执行环境 605 中。

[0107] 图 10 示出基于特征指示符 136 用于给可执行组件 120 提供对执行系统的特征 510 的访问的示例性过程 1000。该过程 1002 访问获取的与标记 130 相关联的特征指示符。该过程识别 1004 特征指示符 136 描述的特征 510 以及可用于执行系统 100 的特征组 500 中的对应特征。该过程允许 1006 特征 510 被可执行组件 120 访问。该过程关联 1008 标记 130 和具有被可执行组件 120 启动的执行的其他可执行组件 120a。

[0108] 7. 实现

[0109] 如上所述的用于控制对计算特征的访问的方法可以使用在计算机上执行的软件执行。例如，该软件形成在一个或多个编程的或可编程的计算机系统（其可以是各种架构，如分布式、客户端 / 服务器或网格式）上执行的一个或多个计算机程序中的过程，该计算机系统每个包括至少一个处理器、至少一个数据存储系统（包括易失性和非易失性存储器和 / 或存储元件）、至少一个输入设备或端口、以及至少一个输出设备或端口。该软件可以形成例如提供与数据流图的设计和配置有关的其它服务的更大程序的一个或多个模块。该图的节点和元素可以实现为计算机可读介质中存储的数据结构或符合数据库中存储的数据模型的其它有组织的数据。

[0110] 该软件可以提供在由通用或专用可编程计算机可读取的存储介质（如 CD-ROM）上，或者可以通过网络的通信介质传递（以传播信号编码）到该软件执行的计算机上。所有功能可以在专用计算机上执行，或者使用专用硬件（如协处理器）执行。该软件可以以分布式方式实现，其中由软件指定的不同计算部分由不同计算机执行。优选地，每个这样的计算机程

序存储或下载到通用或专用可编程计算机可读的存储介质或设备(例如,固态存储器或介质、或磁或光介质)上,用于在该存储介质或设备被计算机系统读取以执行在此描述的过程时配置并操作计算机。还可以考虑将本发明的系统实现为用计算机程序配置的计算机可读存储介质,其中如此配置存储介质,使得计算机系统以特定和预定方式操作以执行在此描述的功能。

[0111] 已经描述了本发明的大量实施例。无论如何,将理解的是可以进行各种修改而不偏离本发明的精神和范围。例如,上面描述的一些步骤可以与顺序无关,因此可以以不同于描述的顺序的顺序执行。

[0112] 要理解的是,前面描述意图图示而不是限制由所附权利要求的范围限定的本发明的范围。例如,上面描述的大量功能步骤可以以不同顺序执行,基本上不影响整体处理。其它实施例在下面的权利要求的范围内。

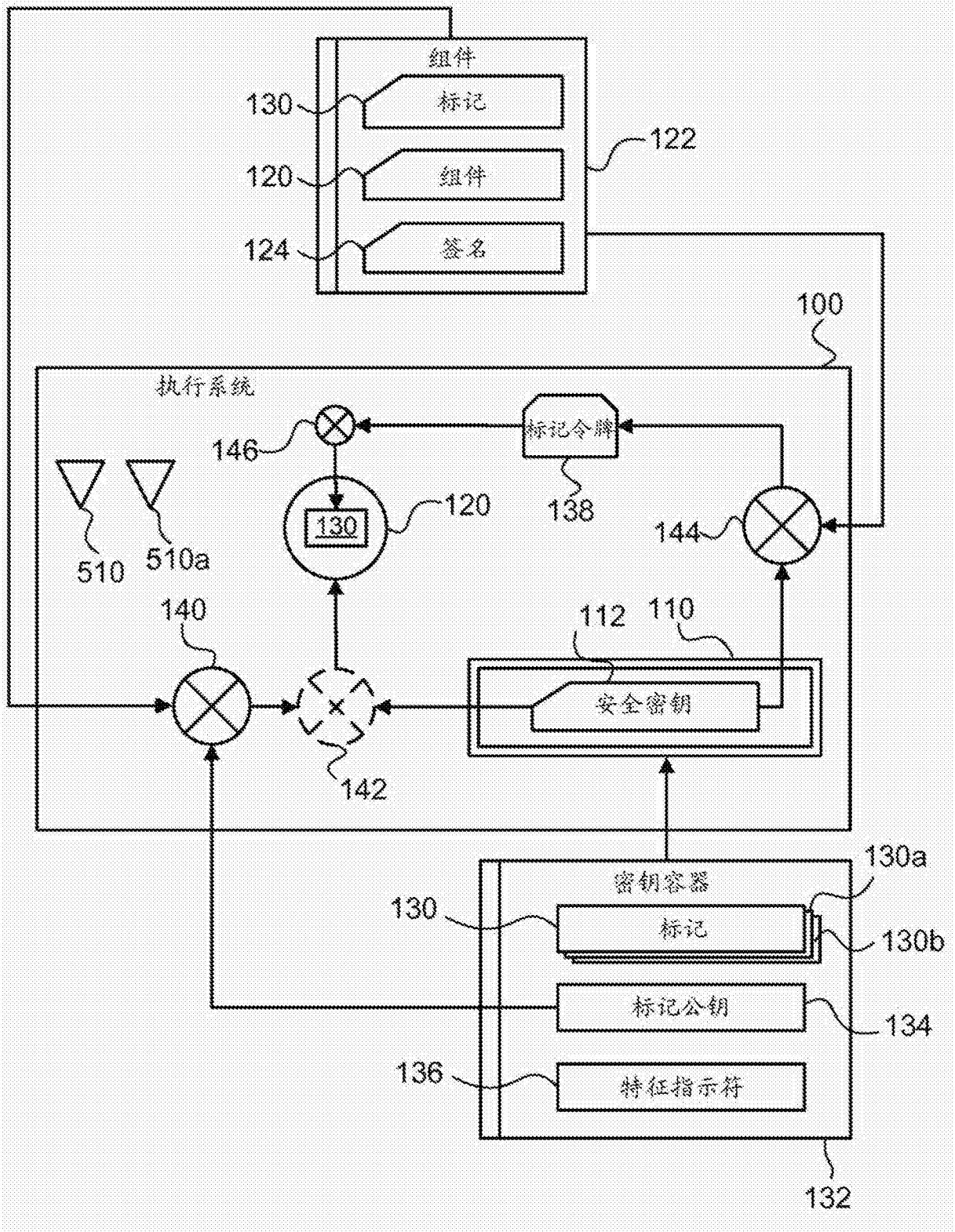

组件
130
标记
122
120
组件
124
签名
100
执行系统
146
标记令牌
138
130
120
144
510
510a
140
112
110
安全密钥
142
密钥容器
130a
130
标记
130b
标记公钥
134
136
特征指示符
132

图 1A

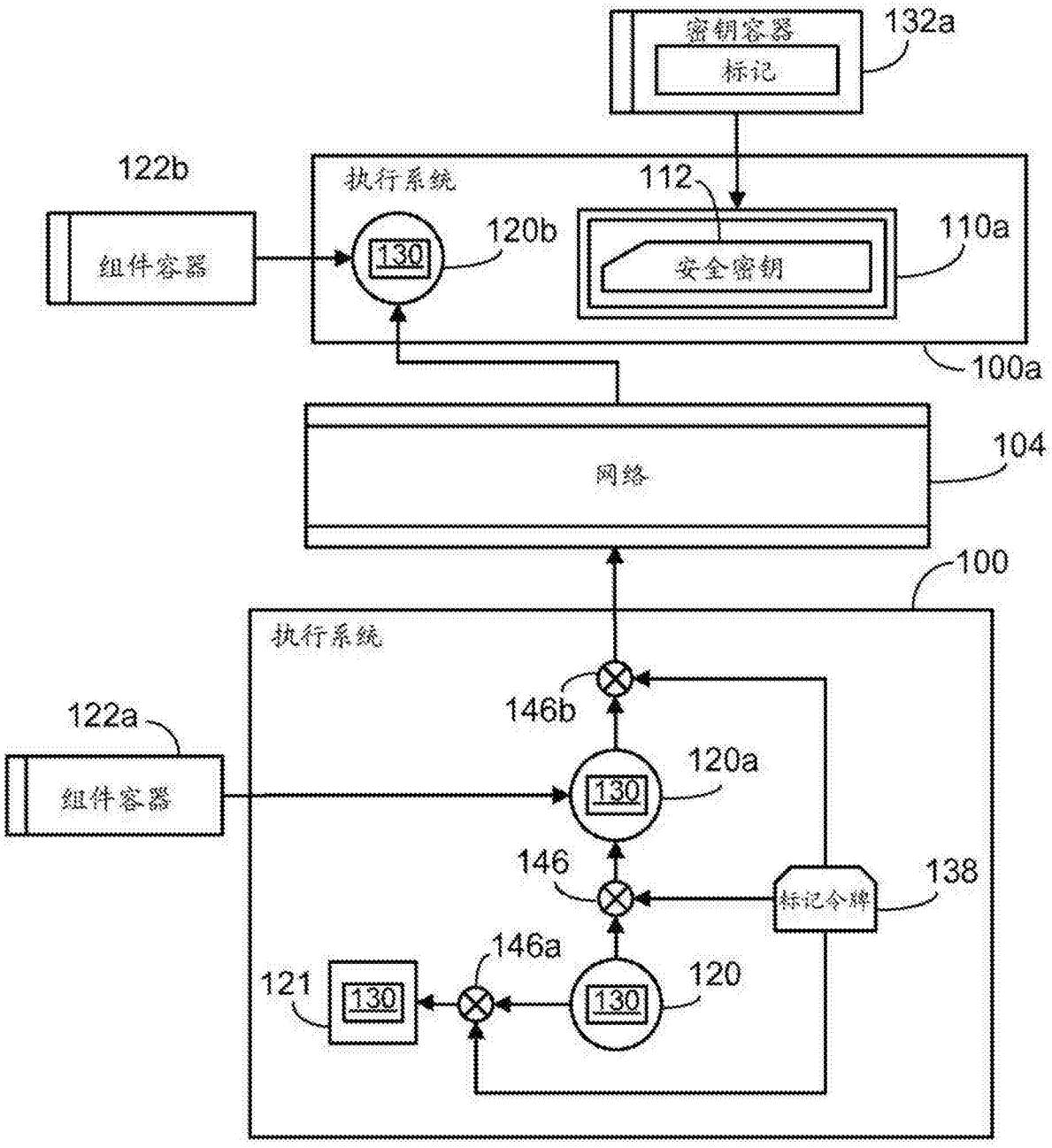

密钥容器
标记
132a
122b
执行系统
112
组件容器
130
120b
安全密钥
110a
100a
网络
104
100
执行系统
146b
122a
120a
组件容器
130
146
标记令牌
138
146a
121
130
130
120

图 1B

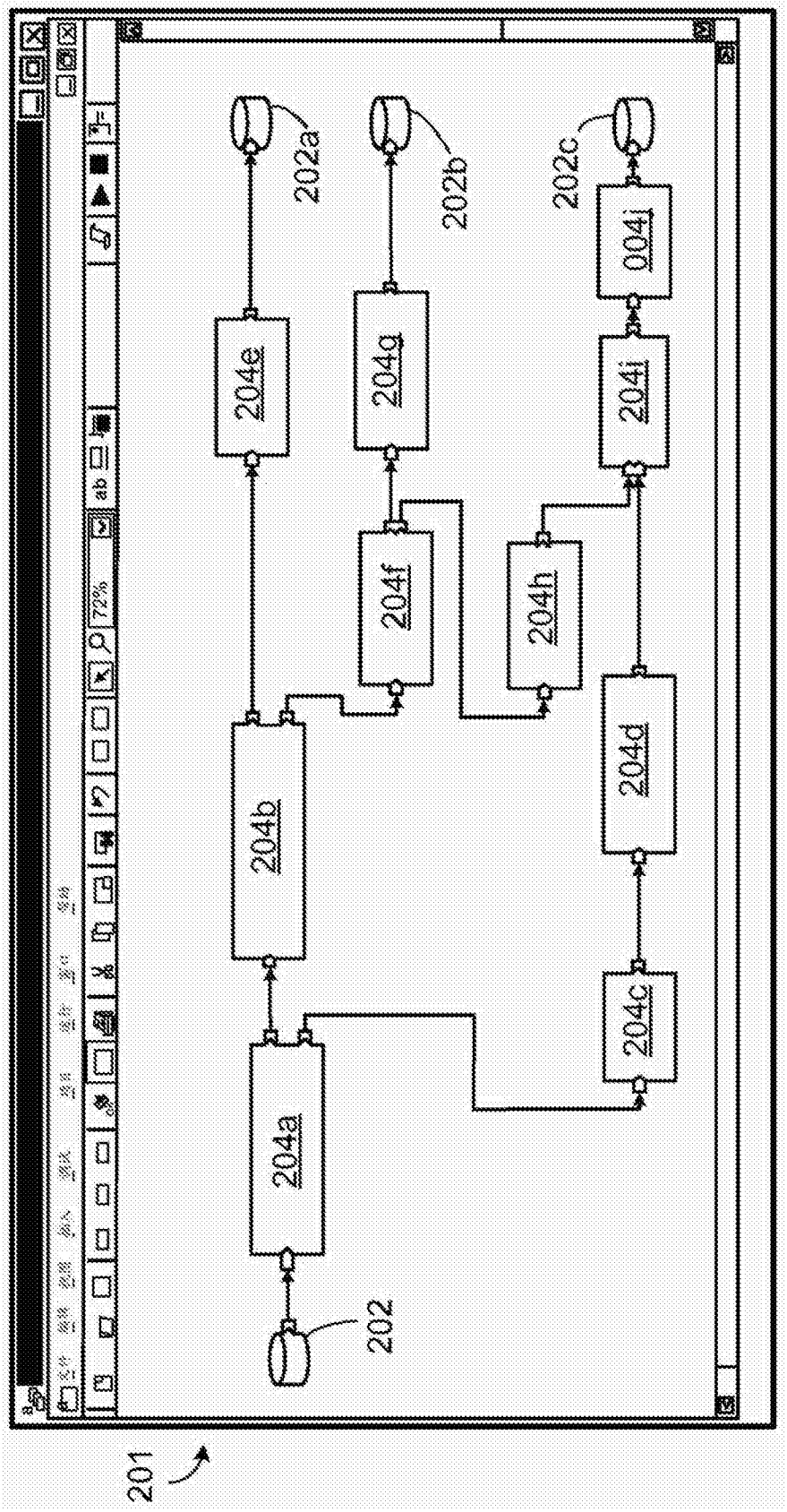

201
202
202a
202b
202c
204a
204b
204c
204d
204e
204f
204g
204h
204i
004j

图 2A

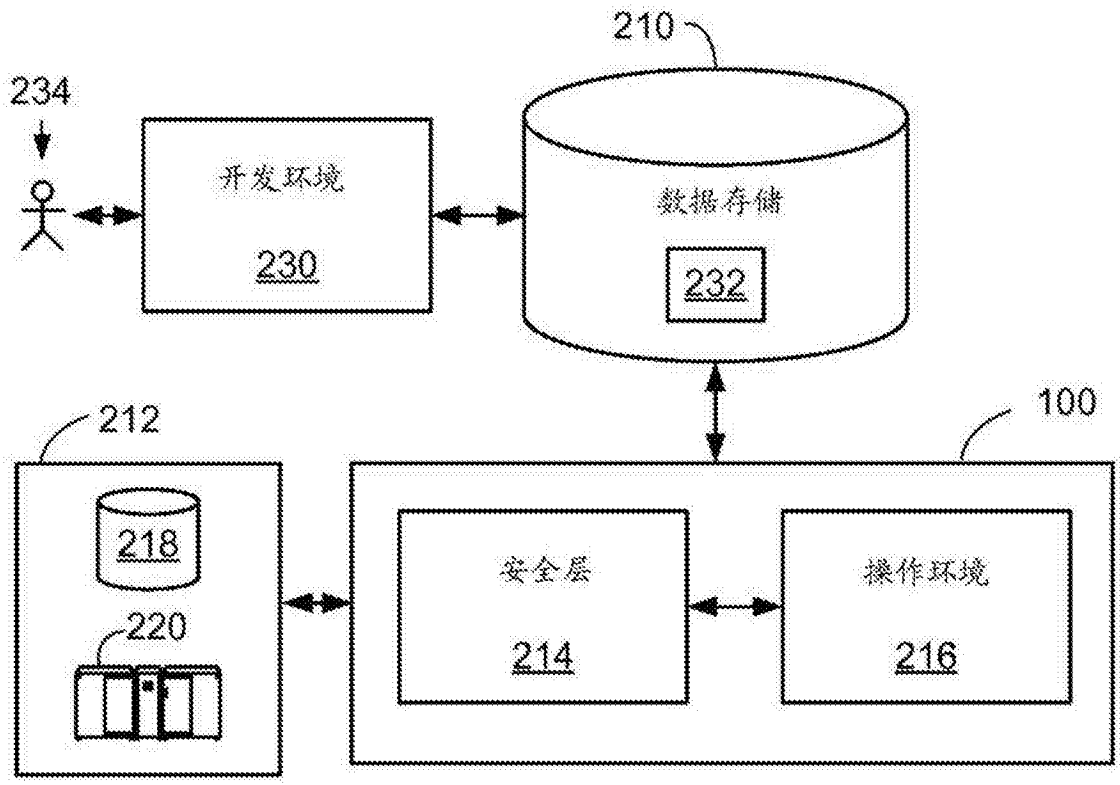


图 2B

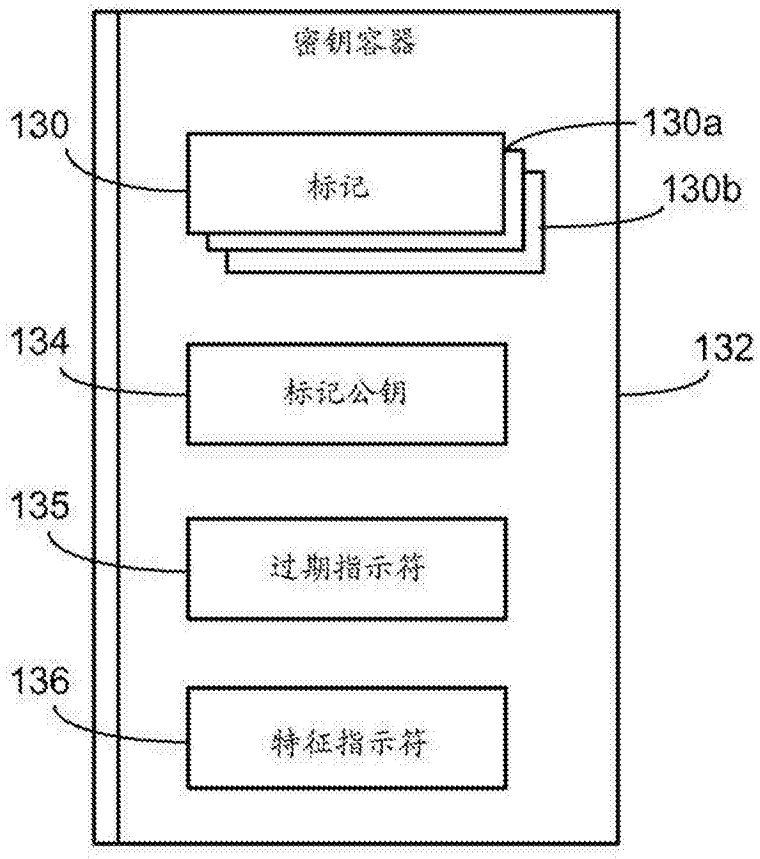


图 3A

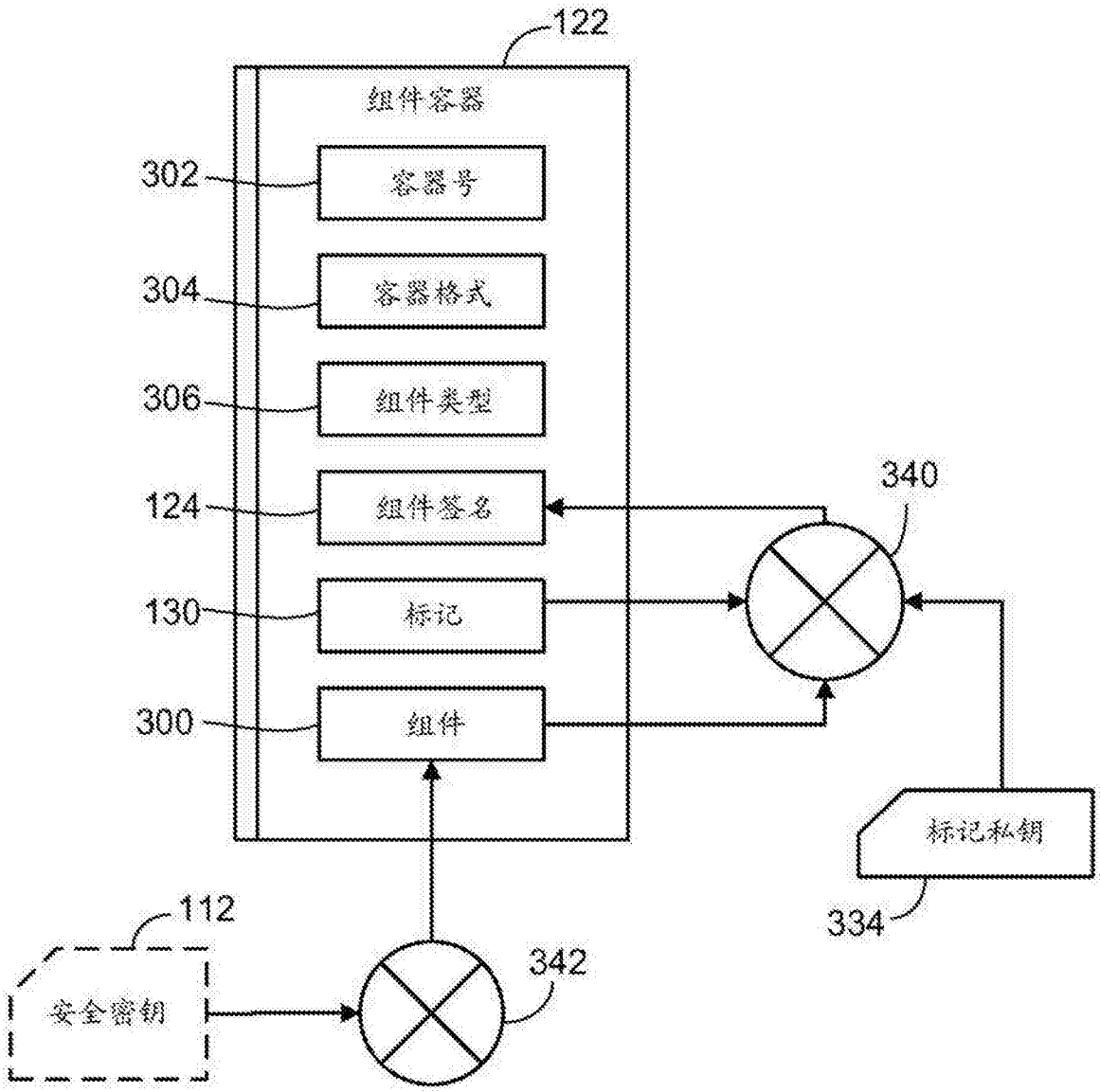

122
组件容器
302
容器号
304
容器格式
306
组件类型
124
组件签名
130
标记
300
组件
340
标记私钥
334
112
安全密钥
342

图 3B

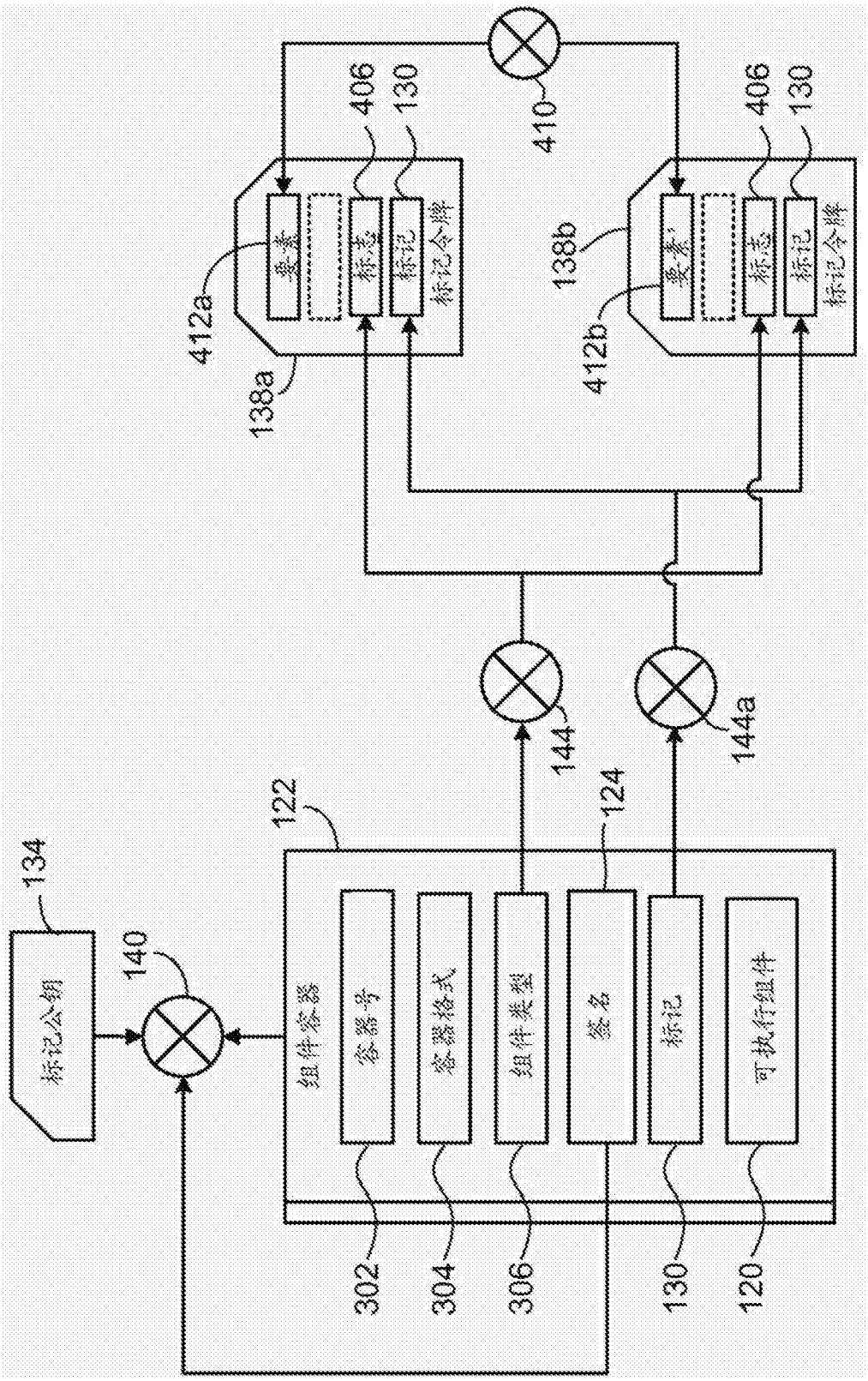

410
406
130
406
130
412a
要素
标志
标记
标记令牌
138a
138b
要素'
412b
144
144a
122
124
134
标记公钥
140
组件容器
容器号
容器格式
组件类型
签名
标记
可执行组件
302
304
306
130
120

图 4A

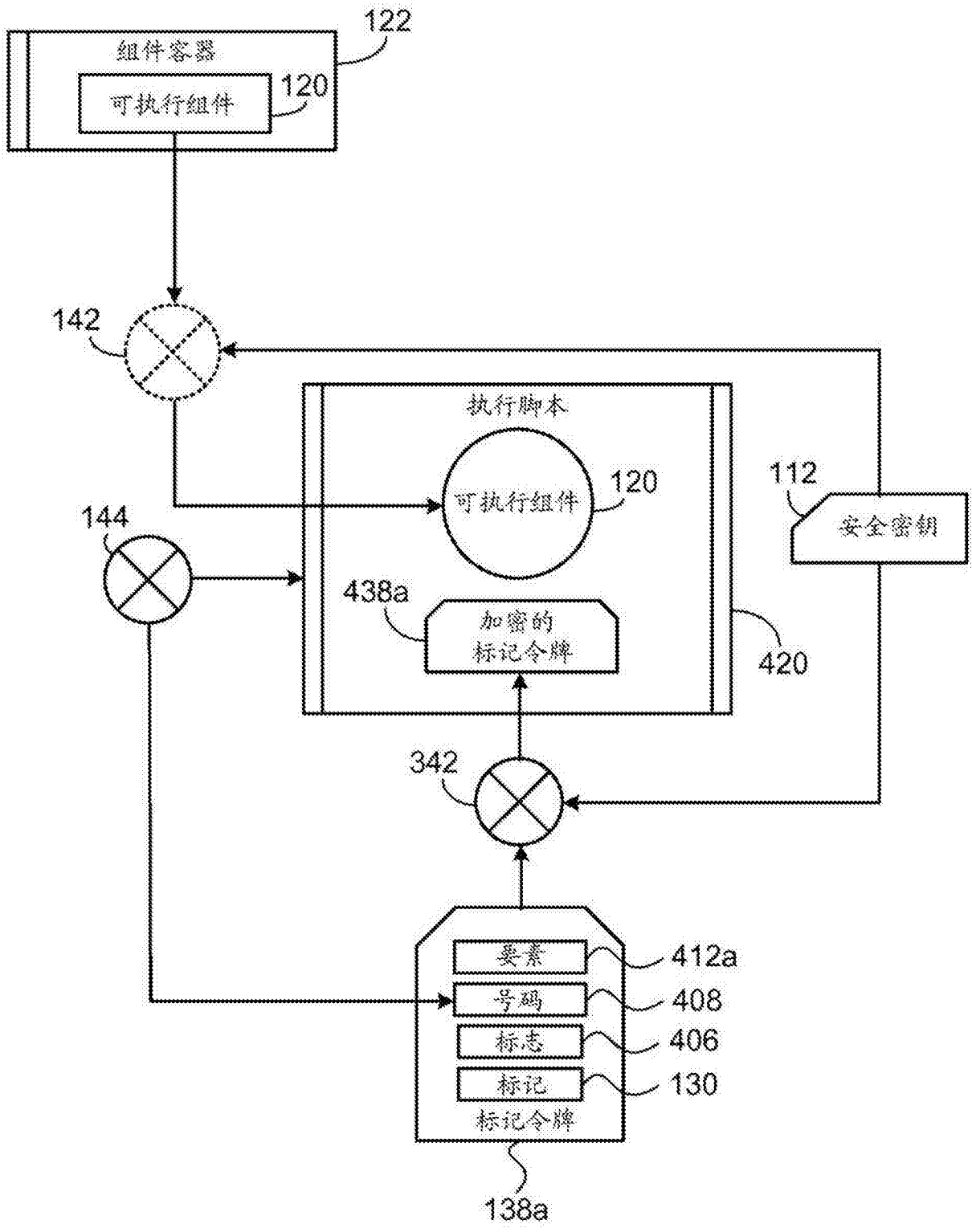

组件容器
122
可执行组件
120
142
执行脚本
可执行组件
120
112
安全密钥
144
438a
加密的
标记令牌
420
342
要素
412a
号码
408
标志
406
标记
130
标记令牌
138a

图 4B

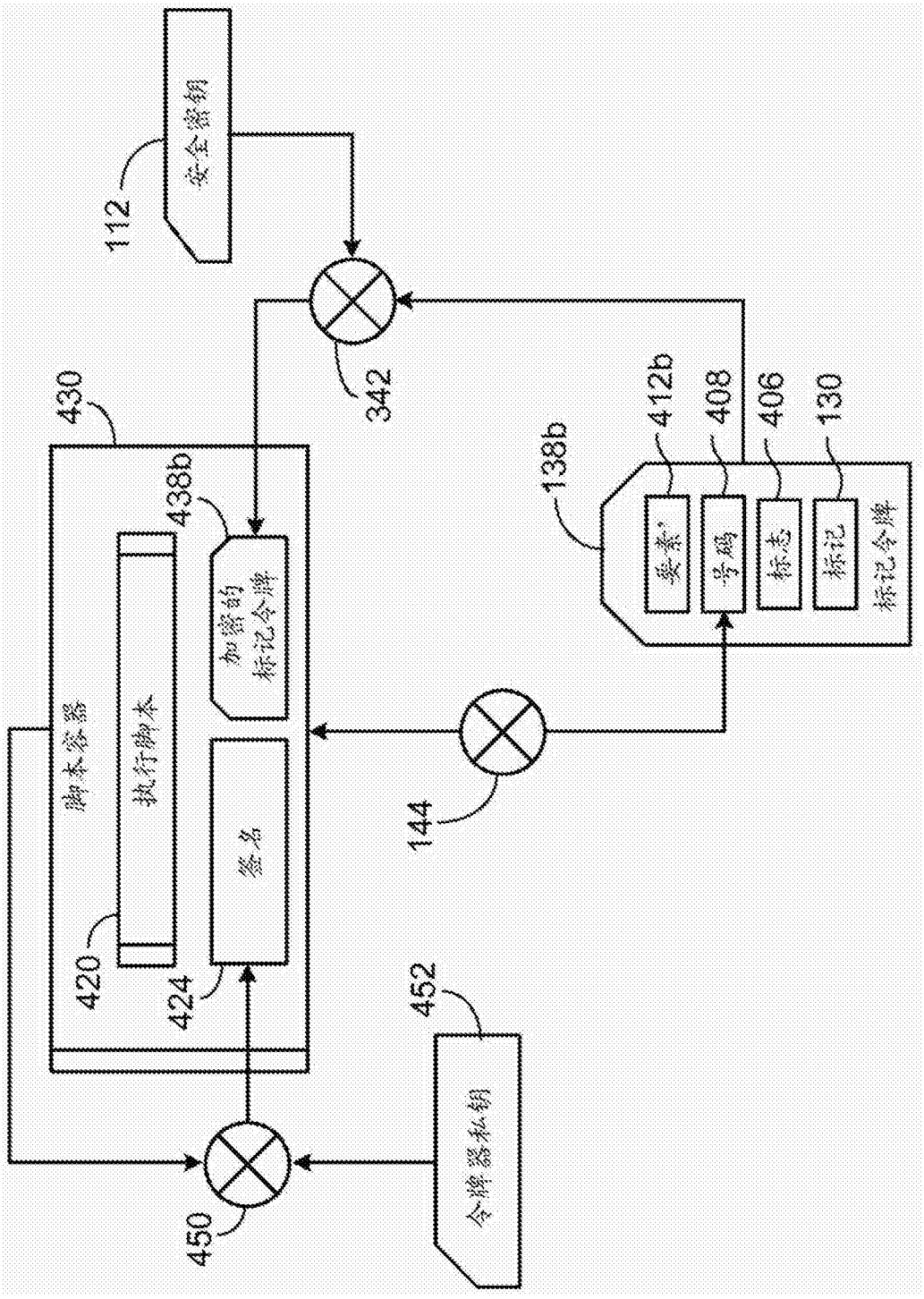

安全密钥
112
342
430
438b
加密的标记令牌
执行脚本
签名
脚本容器
420
424
144
138b
412b
408
406
130
要素'
字码
标志
标记
标记令牌
452
令牌器私钥
450

图 4C

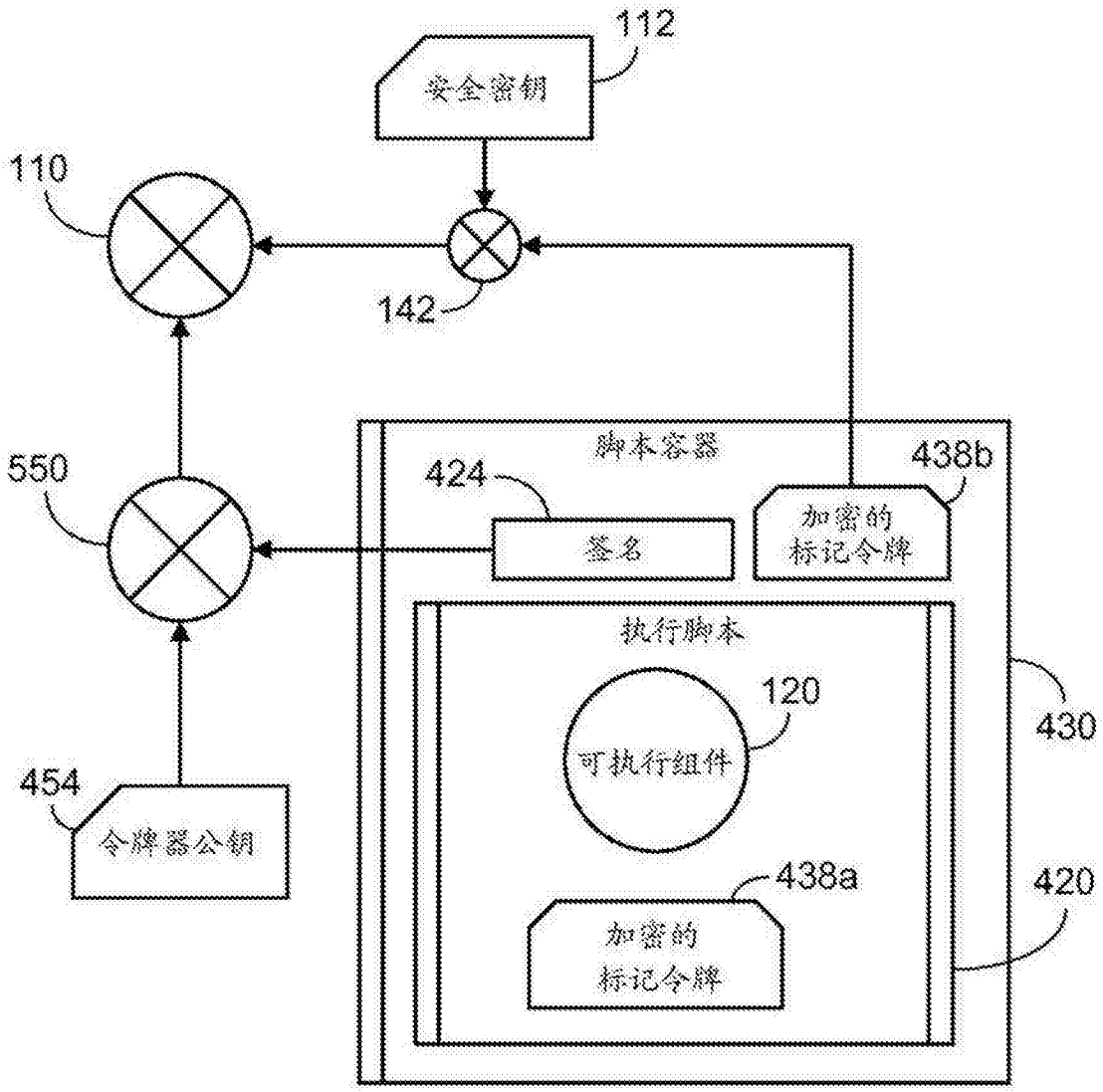

112
安全密钥
110
142
脚本容器
424
438b
550
签名
加密的
标记令牌
执行脚本
120
可执行组件
430
454
令牌器公钥
438a
420
加密的
标记令牌

图 5A

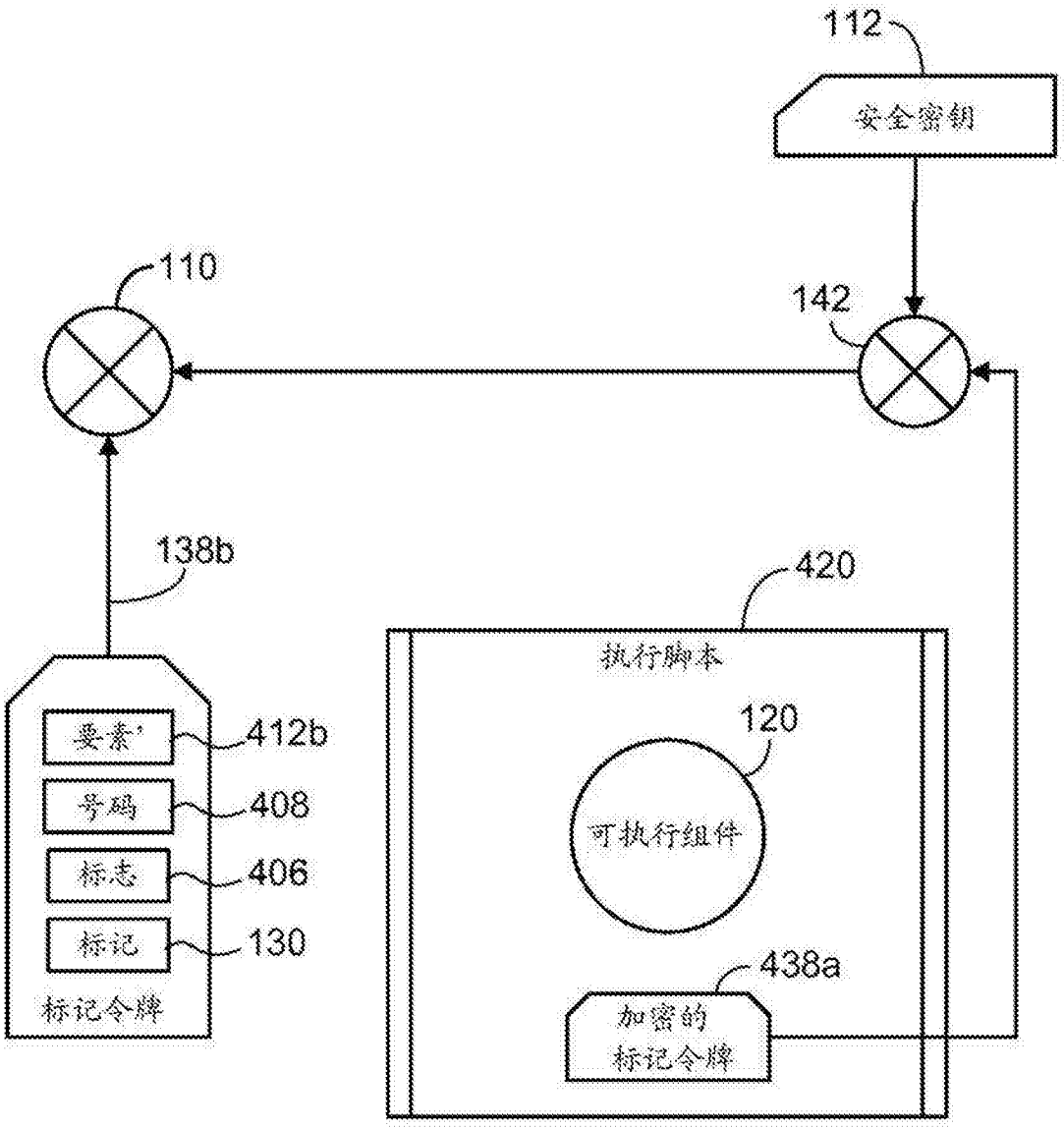

112
安全密钥
110
142
138b
420
执行脚本
120
可执行组件
要素'
412b
号码
408
标志
406
标记
130
标记令牌
438a
加密的
标记令牌

图 5B

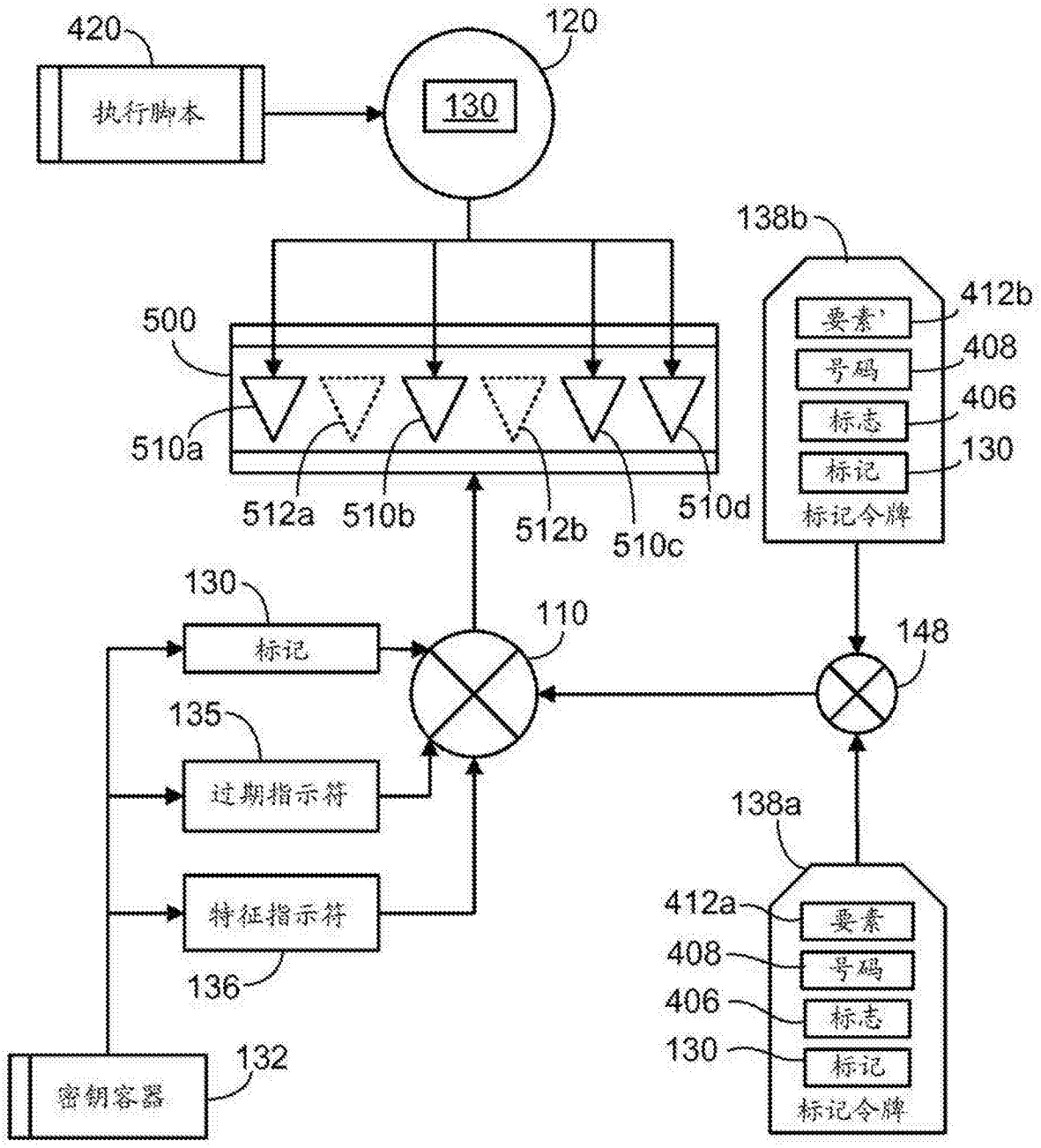

420
执行脚本
120
130
138b
412b
要素'
408
号码
406
标志
130
标记
标记令牌
500
510a
512a
510b
512b
510c
510d
130
标记
110
148
135
过期指示符
特征指示符
136
132
密钥容器
138a
412a
要素
408
号码
406
标志
130
标记
标记令牌

图 5C

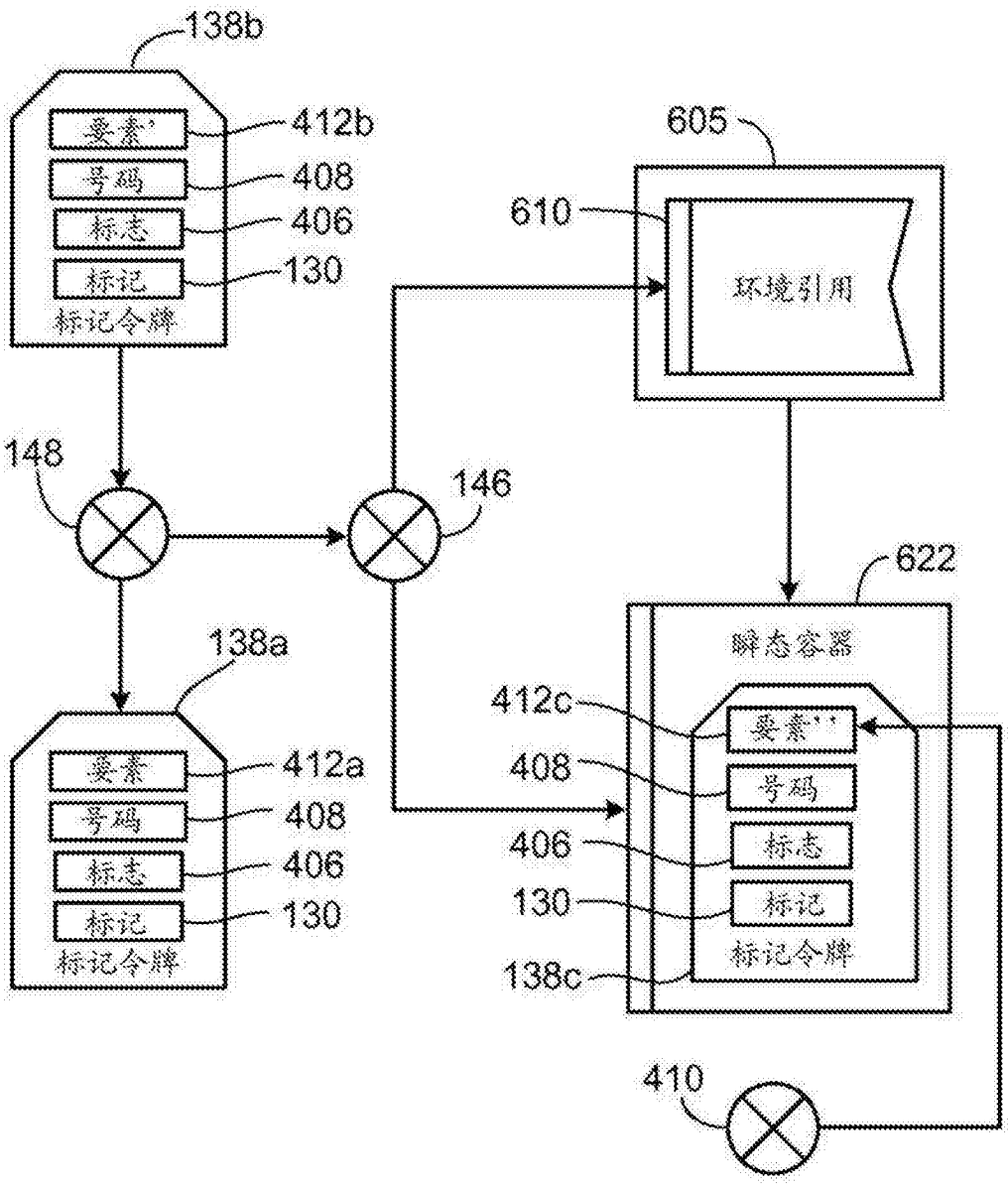

138b
要素'
412b
号码
408
标志
406
标记
130
标记令牌
605
610
环境引用
148
146
622
瞬态容器
138a
412c
要素''
要素
412a
408
号码
406
标志
130
标记
标记令牌
138c
410

图 6A

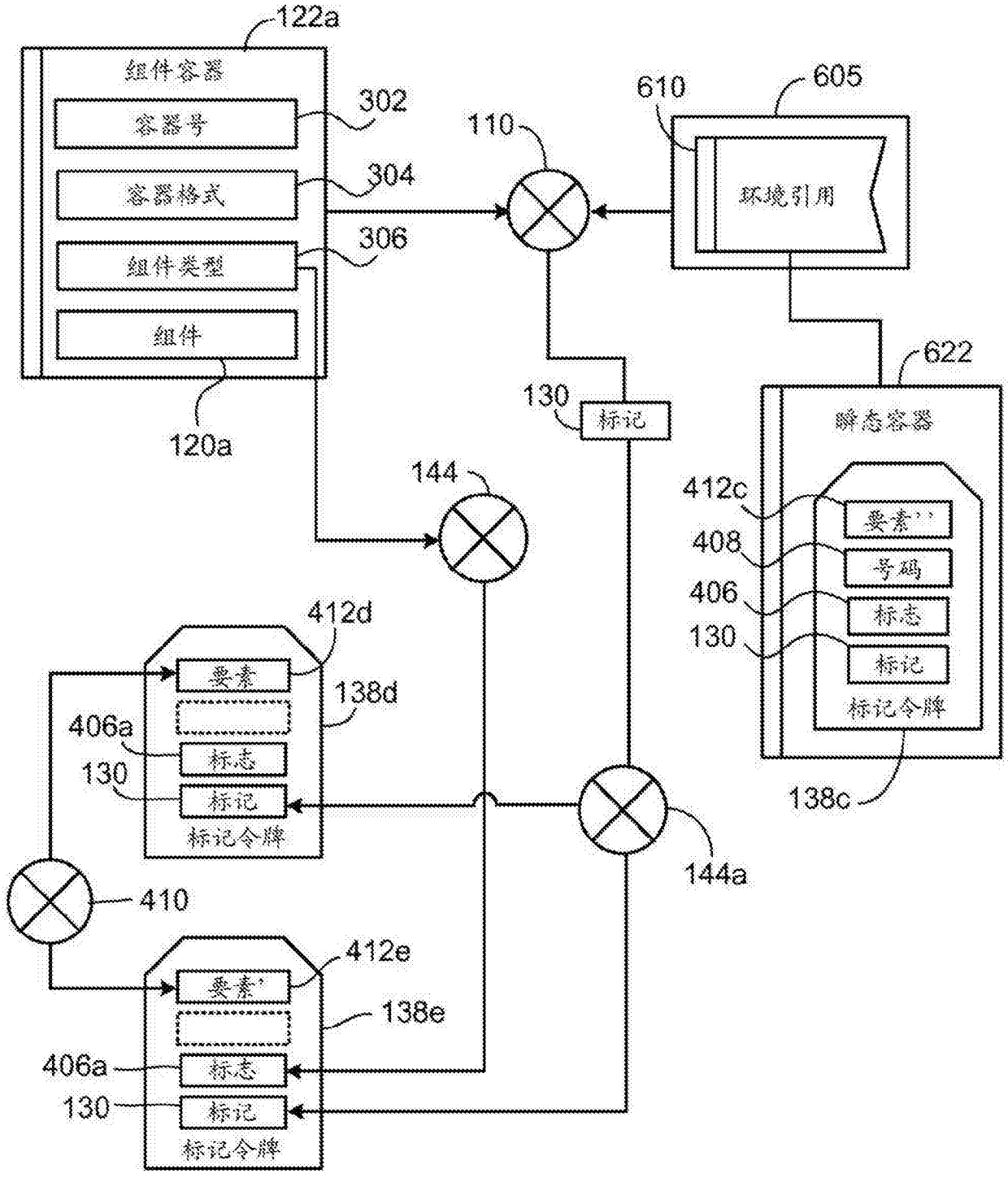

122a
组件容器
容器号
302
容器格式
304
组件类型
306
组件
120a
110
610
605
环境引用
622
瞬态容器
130
标记
144
412c
要素''
408
号码
406
标志
130
标记
标记令牌
138c
412d
要素
138d
406a
标志
130
标记
标记令牌
410
144a
412e
要素'
138e
406a
标志
130
标记
标记令牌

图 6B

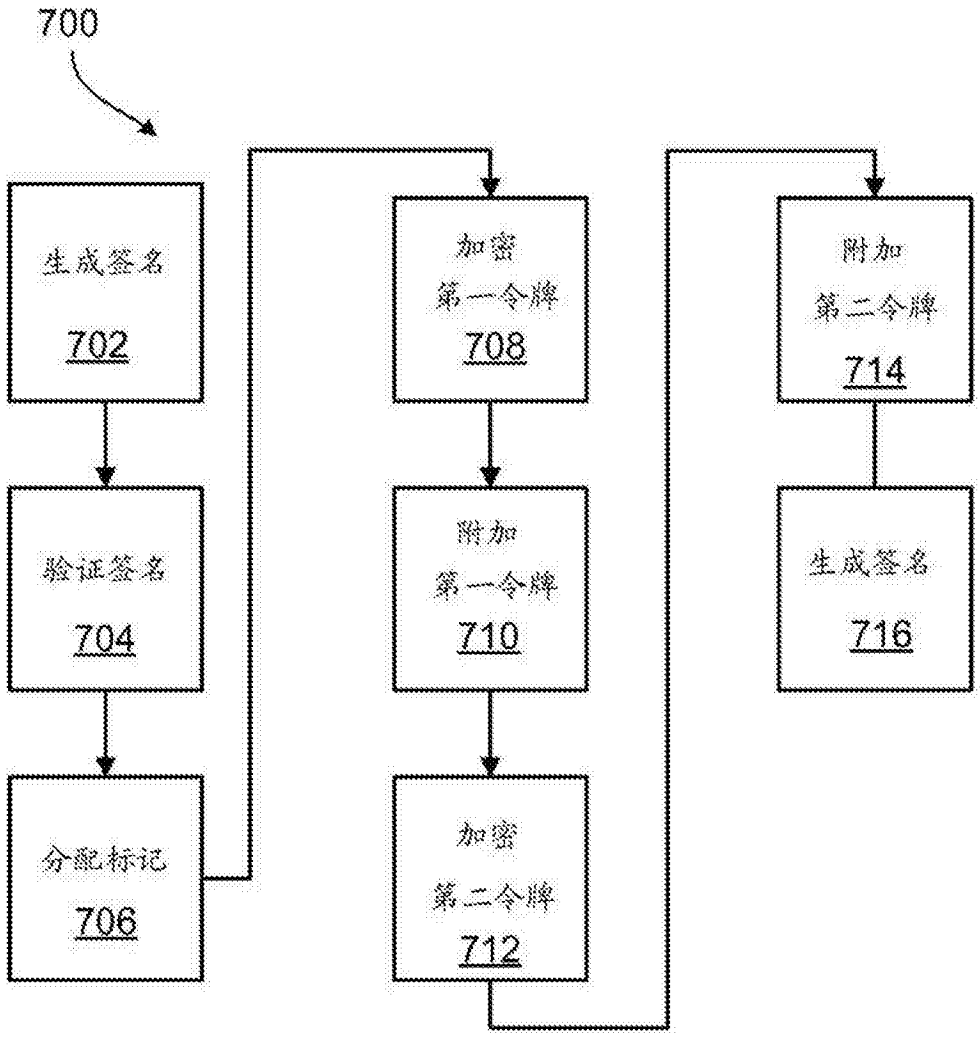

700
生成签名
702
验证签名
704
分配标记
706
加密
第一令牌
708
附加
第一令牌
710
加密
第二令牌
712
附加
第二令牌
714
生成签名
716

图 7

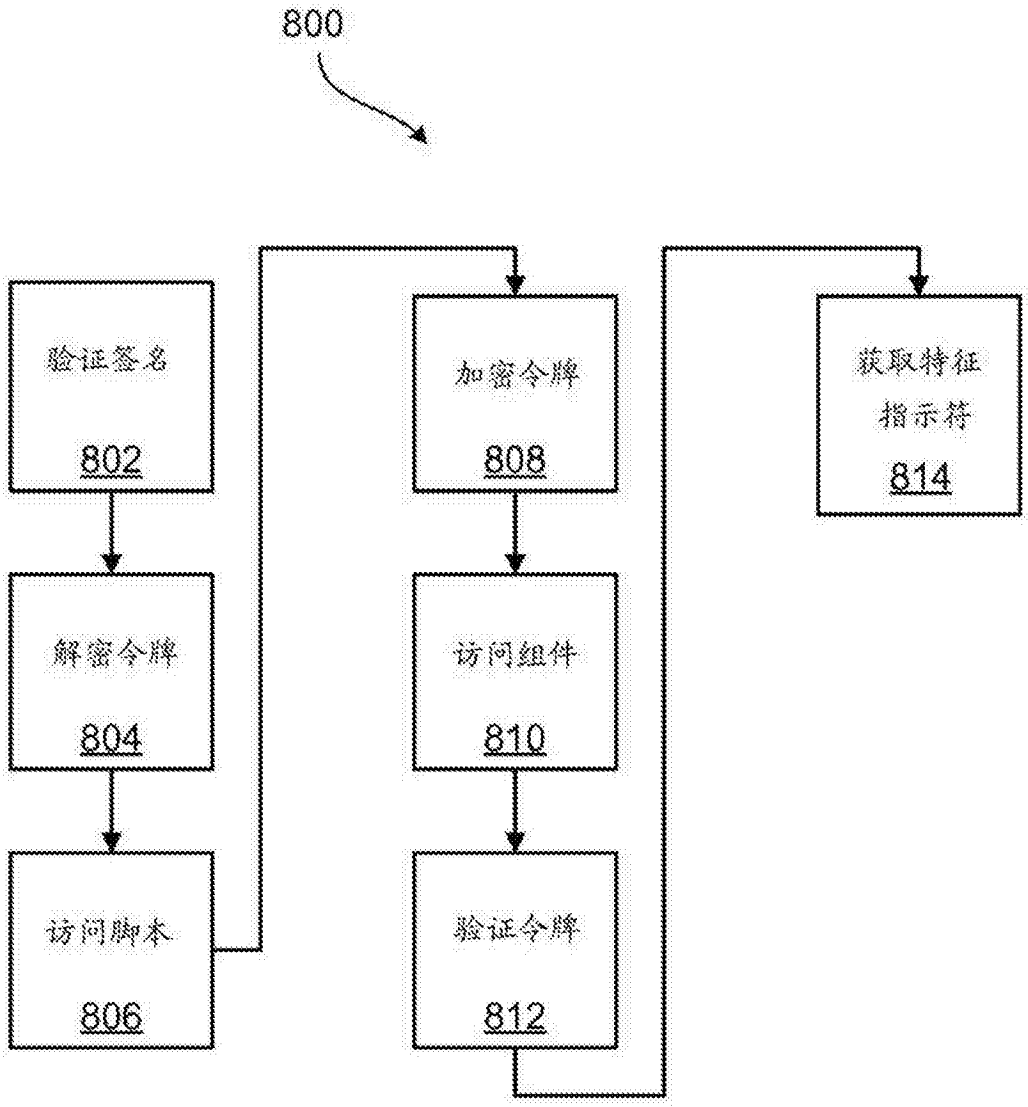

800
验证签名
802
解密令牌
804
访问脚本
806
加密令牌
808
访问组件
810
验证令牌
812
获取特征指示符
814

图 8

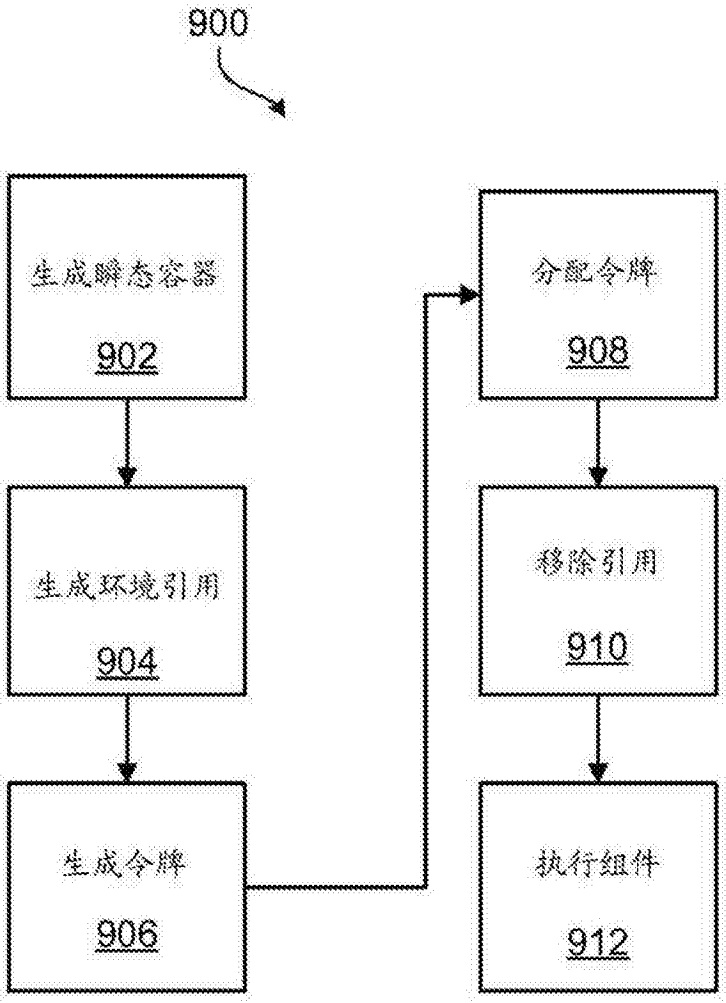


图 9

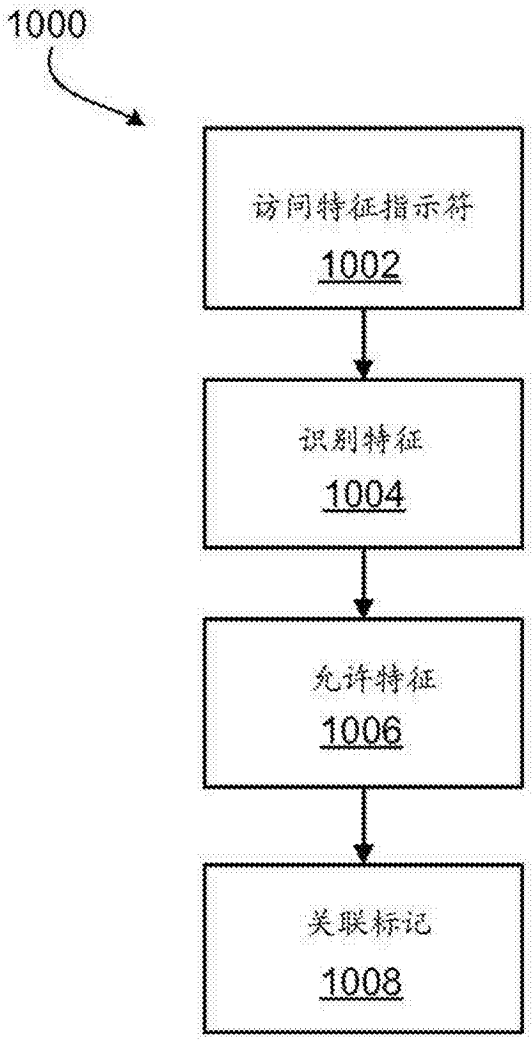


图 10