(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 111010266 B
(45)授权公告日 2023.04.07

(21)申请号 201911250793.1
(22)申请日 2019.12.09
(65)同一申请的已公布的文献号
申请公布号 CN 111010266 A
(43)申请公布日 2020.04.14
(73)专利权人 广州市百果园信息技术有限公司
地址 511400 广东省广州市番禺区市桥街兴泰路274号C栋西塔5-13层
(72)发明人 李岩 李东 陈耿书
(74)专利代理机构 北京品源专利代理有限公司 11332
专利代理师 孟金喆
(51)Int.Cl.
H04L 9/06(2006.01)
H04L 9/00(2022.01)

(56)对比文件
CN 106663387 A,2017.05.10
CN 109478995 A,2019.03.15
CN 107005404 A,2017.08.01
US 2011013767 A1,2011.01.20
CN 106059752 A,2016.10.26

审查员 安佳

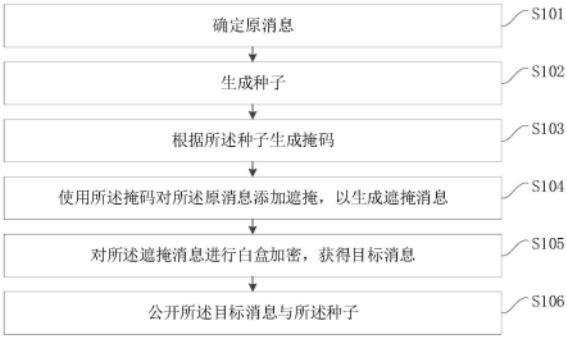
权利要求书4页 说明书18页 附图7页

(54)发明名称

消息的加解密、读写方法、装置、计算机设备和存储介质

(57)摘要

本发明实施例公开了一种消息的加解密、读写方法、装置、计算机设备和存储介质，该消息的加密方法包括：确定原消息；生成种子；根据所述种子生成掩码；使用所述掩码对所述原消息添加遮掩，以生成遮掩消息；对所述遮掩消息进行白盒加密，获得目标消息；公开所述目标消息与所述种子。本实施例可保证在CPA语义下的安全性，容纳其他现有的白盒加密算法，并且，占用资源较少，运算速度较快，可应用于移动环境，填补了学术设计和工业需求之间的空白，将学术设计改进为实用的工业解决方案。



CN 111010266 B

1.一种消息的加密方法,其特征在于,包括:

确定原消息;

生成种子;

根据所述种子生成掩码;

使用所述掩码对所述原消息添加遮掩,以生成遮掩消息;

对所述遮掩消息进行白盒加密,获得目标消息;

公开所述目标消息与所述种子;

所述生成种子为,白盒加密API收到输入的所述原消息后,生成种子,其中,所述种子具有唯一性,不重复使用,所述种子为数字、字符串、字符中的至少一种;

所述根据所述种子生成掩码包括,将所述种子输入到掩码生成函数中生成掩码,所述种子与所述掩码之间具有固定的变换关系,所述掩码具有唯一性。

2.根据权利要求1所述的方法,其特征在于,在所述确定原消息之后,还包括:

在所述原消息中填充指定的数据。

3.根据权利要求1所述的方法,其特征在于,所述生成种子,包括:

随机生成一数值,作为种子。

4.根据权利要求1所述的方法,其特征在于,所述根据所述种子生成掩码,包括:

确定所述原消息的长度;

确定哈希函数;

将所述种子输入至所述哈希函数中进行运算,以生成所述长度的掩码。

5.根据权利要求4所述的方法,其特征在于,所述哈希函数包括SHAKE函数或者SHA函数;

所述将所述种子输入至所述哈希 函数中进行运算,以生成所述长度的掩码,包括:

将所述种子作为输入至所述SHAKE函数中进行运算,以输出所述长度的掩码;

或者,

在所述种子的基础生成多个新的种子;

将所有种子分别输入至SHA函数中进行运算,以分别输出多个摘要信息;

组合多个所述摘要信息,获得第一数据集;

从所述第一数据集中提取所述长度的数据,作为掩码。

6.根据权利要求1所述的方法,其特征在于,所述根据所述种子生成掩码,包括:

确定所述原消息的长度;

对所述种子进行所述白盒加密,以生成所述长度的掩码。

7.根据权利要求6所述的方法,其特征在于,所述对所述种子进行所述白盒加密,以生成所述长度的掩码,包括:

在所述种子的基础生成多个新的种子;

将所有种子分别进行白盒加密,以分别生成多个密文;

组合多个所述密文,获得第二数据集;

从所述第二数据集中提取所述长度的数据,作为掩码。

8.根据权利要求1-7任一项所述的方法,其特征在于,所述使用所述掩码对所述原消息添加遮掩,以生成遮掩消息,包括:

对所述原消息与所述掩码进行异或运算，获得遮掩消息。

9.一种消息的解密方法，其特征在于，包括：

确定目标消息和种子；

对所述目标消息进行白盒解密，获得遮掩消息；

根据所述种子生成掩码；

使用所述掩码对所述遮掩消息去除遮掩，获得原消息；

所述目标消息为经过白盒加密API使用所述种子对原消息加密之后的密文，所述目标消息与所述种子一同公开；

所述根据所述种子生成掩码包括：掩码生成函数使用所述种子作为输入来产生所述掩码，所述种子与所述掩码之间具有固定的变换关系，所述种子具有唯一性，所述掩码也具有唯一性。

10.根据权利要求9所述的方法，其特征在于，所述使用所述掩码对所述遮掩消息去除遮掩，获得原消息，包括：

对所述目标消息与所述掩码进行异或运算，获得原消息。

11.根据权利要求8所述的方法，其特征在于，还包括：

从所述原消息中去除在先填充的数据。

12.一种消息的写方法，其特征在于，包括：

当接收到应用生成的原消息时，生成种子；

根据所述种子生成掩码；

使用所述掩码对所述原消息添加遮掩，以生成遮掩消息；

对所述遮掩消息进行白盒加密，获得目标消息；

对所述原消息存储所述目标消息和所述种子；

白盒加密API收到输入的所述原消息后，生成种子，其中，所述种子具有唯一性，不重复使用，所述种子为数字、字符串、字符中的至少一种；

掩码生成函数使用所述种子作为输入来产生所述掩码，所述种子与所述掩码之间具有固定的变换关系，所述种子具有唯一性，所述掩码也具有唯一性。

13.一种消息的读方法，其特征在于，包括：

当接收到应用对原消息的读操作时，确定所述原消息对应的目标消息和种子；

对所述目标消息进行白盒解密，获得遮掩消息；

根据所述种子生成掩码；

使用所述掩码对所述遮掩消息去除遮掩，以获得原消息；

将所述原消息发送至所述应用；

所述目标消息为经过白盒加密API使用所述种子对所述原消息加密之后的密文，所述目标消息与所述种子一同存储在存储空间，在存储空间中存储所述目标消息、所述种子与所述原消息之间的映射关系；应用在运行过程中，按照需求执行度操作，以读取所述原消息，在存储空间通过该映射关系查找所述目标消息与所述种子；

掩码生成函数使用所述种子作为输入来产生所述掩码，所述种子与所述掩码之间具有固定的变换关系，所述种子具有唯一性，所述掩码也具有唯一性。

14.一种消息的加密装置，其特征在于，包括：

原消息确定模块，用于确定原消息；

种子生成模块，用于生成种子；

掩码生成模块，用于根据所述种子生成掩码；

遮掩添加模块，用于使用所述掩码对所述原消息添加遮掩，以生成遮掩消息；

白盒加密模块，用于对所述遮掩消息进行白盒加密，获得目标消息；

密文公开模块，用于公开所述目标消息与所述种子；

所述生成种子为，白盒加密API收到输入的所述原消息后，生成种子，其中，所述种子具有唯一性，不重复使用，所述种子为数字、字符串、字符中的至少一种；

将所述种子输入到掩码生成函数中生成掩码，所述种子与所述掩码之间具有固定的变换关系，所述掩码具有唯一性。

15.一种消息的解密装置，其特征在于，包括：

密文确定模块，用于确定目标消息和种子；

白盒解密模块，用于对所述目标消息进行白盒解密，获得遮掩消息；

掩码生成模块，用于根据所述种子生成掩码；

遮掩去除模块，用于使用所述掩码对所述遮掩消息去除遮掩，获得原消息；

所述目标消息为经过白盒加密API使用所述种子对原消息加密之后的密文，所述目标消息与所述种子一同公开；

掩码生成函数使用所述种子作为输入来产生所述掩码，所述种子与所述掩码之间具有固定的变换关系，所述种子具有唯一性，所述掩码也具有唯一性。

16.一种消息的写装置，其特征在于，包括：

种子生成模块，用于当接收到应用生成的原消息时，生成种子；

掩码生成模块，用于根据所述种子生成掩码；

遮掩添加模块，用于使用所述掩码对所述原消息添加遮掩，以生成遮掩消息；

白盒加密模块，用于对所述遮掩消息进行白盒加密，获得目标消息；

密文存储模块，用于对所述原消息存储所述目标消息和所述种子；

白盒加密API收到输入的所述原消息后，生成种子，其中，所述种子具有唯一性，不重复使用，所述种子为数字、字符串、字符中的至少一种；

掩码生成函数使用所述种子作为输入来产生所述掩码，所述种子与所述掩码之间具有固定的变换关系，所述种子具有唯一性，所述掩码也具有唯一性。

17.一种消息的读装置，其特征在于，包括：

密文确定模块，用于当接收到应用对原消息的读操作时，确定所述原消息对应的目标消息和种子；

白盒解密模块，用于对所述目标消息进行白盒解密，获得遮掩消息；

掩码生成模块，用于根据所述种子生成掩码；

遮掩去除模块，用于使用所述掩码对所述遮掩消息去除遮掩，以获得原消息；

原消息发送模块，用于将所述原消息发送至所述应用；

所述目标消息为经过白盒加密API使用所述种子对所述原消息加密之后的密文，所述目标消息与所述种子一同存储在存储空间，在存储空间中存储所述目标消息、所述种子与所述原消息之间的映射关系；应用在运行过程中，按照需求执行度操作，以读取所述原消

息,在存储空间通过该映射关系查找所述目标消息与所述种子;

掩码生成函数使用所述种子作为输入来产生所述掩码,所述种子与所述掩码之间具有固定的变换关系,所述种子具有唯一性,所述掩码也具有唯一性。

18.一种计算机设备,其特征在于,所述计算机设备包括:

一个或多个处理器;

存储器,用于存储一个或多个程序;

当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现如权利要求1-8中任一所述的消息的加密方法、或者如权利要求9-11中任一所述的消息的解密方法、或者如权利要求12所述的消息的写方法、或者如权利要求13所述的消息的读方法。

19.一种计算机可读存储介质,其上存储有计算机程序,其特征在于,该计算机程序被处理器执行时实现如权利要求1-8中任一所述的消息的加密方法、或者如权利要求9-11中任一所述的消息的解密方法、或者如权利要求12所述的消息的写方法、或者如权利要求13所述的消息的读方法。

消息的加解密、读写方法、装置、计算机设备和存储介质

技术领域

[0001] 本发明实施例涉及安全领域的技术，尤其涉及一种消息的加解密、读写方法、装置、计算机设备和存储介质。

背景技术

[0002] 计算机设备中的数据保护技术和密码学有着紧密的联系，在加壳、反调试、数据存储等方面，往往都需要通过加密将敏感数据隐藏起来，而隐藏起来的敏感数据，将面临黑盒、灰盒、白盒等攻击。

[0003] 其中，白盒是指攻击者已经完全控制了整个操作过程且对此完全可见，攻击者可以自如地观察动态密码运行过程，并且内部算法的详细内容完全可见，可随意更改。

[0004] 举例而言，软件是在本地运行的，攻击者可以通过调试器运行程序，并观察软件运行的过程，所有涉及解密部分的代码也就一览无余了。

[0005] 为了保证数据的安全性，目前已发布了一些白盒加密算法，如白盒AES(Advanced Encryption Standard，高级加密标准)加密，将白盒加密算法部署到在开放设备上执行的应用中时，开发人员可能使用语义上不安全或效率低下的模式来实现白盒加密算法。例如：

[0006] ECB(电子密码簿)模式，ECB模式下的白盒加密算法将相同的块加密为相同的密文，容易暴露明文模式。

[0007] CBC(密码块链接)模式，对于未经身份验证的加密存在错误传播问题。重新使用IV(初始化向量)将泄漏明文的第一个块的信息。生成，隐藏和记住客户端白盒加密算法的每个动态IV都是不切实际的。

[0008] CTR(计数器)模式，在流密码中利用白盒加密算法。重用IV将完全破坏安全性，白盒加密提供的保护将被绕过。

[0009] 在实际应用环境中，白盒加密算法是对单个块进行加密，而消息一般含有多个块，整个消息在CPA(chosen-plaintext attack，选择明文攻击)下的语义是具有风险的。

发明内容

[0010] 本发明实施例提供一种消息的加解密、读写方法、装置、计算机设备和存储介质，以解决对消息进行白盒加密，在CPA模式下具有风险的问题。

[0011] 第一方面，本发明实施例提供了一种消息的加密方法，包括：

[0012] 确定原消息；

[0013] 生成种子；

[0014] 根据所述种子生成掩码；

[0015] 使用所述掩码对所述原消息添加遮掩，以生成遮掩消息；

[0016] 对所述遮掩消息进行白盒加密，获得目标消息；

[0017] 公开所述目标消息与所述种子。

[0018] 第二方面，本发明实施例还提供了一种消息的解密方法，包括：

[0019] 确定目标消息和种子；

[0020] 对所述目标消息进行白盒解密，获得遮掩消息；

[0021] 根据所述种子生成掩码；

[0022] 使用所述掩码对所述遮掩消息去除遮掩，获得原消息。

[0023] 第三方面，本发明实施例还提供了一种消息的写方法，包括：

[0024] 当接收到应用生成的原消息时，生成种子；

[0025] 根据所述种子生成掩码；

[0026] 使用所述掩码对所述原消息添加遮掩，以生成遮掩消息；

[0027] 对所述遮掩消息进行白盒加密，获得目标消息；

[0028] 对所述原消息存储所述目标消息和所述种子。

[0029] 第四方面，本发明实施例还提供了一种消息的读方法，包括：

[0030] 当接收到应用对原消息的读操作时，确定所述原消息对应的目标消息和种子；

[0031] 对所述目标消息进行白盒解密，获得遮掩消息；

[0032] 根据所述种子生成掩码；

[0033] 使用所述掩码对所述遮掩消息去除遮掩，以获得原消息；

[0034] 将所述原消息发送至所述应用。

[0035] 第五方面，本发明实施例还提供了一种消息的加密装置，包括：

[0036] 原消息确定模块，用于确定原消息；

[0037] 种子生成模块，用于生成种子；

[0038] 掩码生成模块，用于根据所述种子生成掩码；

[0039] 遮掩添加模块，用于使用所述掩码对所述原消息添加遮掩，以生成遮掩消息；

[0040] 白盒加密模块，用于对所述遮掩消息进行白盒加密，获得目标消息；

[0041] 密文公开模块，用于公开所述目标消息与所述种子。

[0042] 第六方面，本发明实施例还提供了一种消息的解密装置，包括：

[0043] 密文确定模块，用于确定目标消息和种子；

[0044] 白盒解密模块，用于对所述目标消息进行白盒解密，获得遮掩消息；

[0045] 掩码生成模块，用于根据所述种子生成掩码；

[0046] 遮掩去除模块，用于使用所述掩码对所述遮掩消息去除遮掩，获得原消息。

[0047] 第七方面，本发明实施例还提供了一种消息的写装置，包括：

[0048] 种子生成模块，用于当接收到应用生成的原消息时，生成种子；

[0049] 掩码生成模块，用于根据所述种子生成掩码；

[0050] 遮掩添加模块，用于使用所述掩码对所述原消息添加遮掩，以生成遮掩消息；

[0051] 白盒加密模块，用于对所述遮掩消息进行白盒加密，获得目标消息；

[0052] 密文存储模块，用于对所述原消息存储所述目标消息和所述种子。

[0053] 第八方面，本发明实施例还提供了一种消息的读装置，包括：

[0054] 密文确定模块，用于当接收到应用对原消息的读操作时，确定所述原消息对应的目标消息和种子；

[0055] 白盒解密模块，用于对所述目标消息进行白盒解密，获得遮掩消息；

[0056] 掩码生成模块，用于根据所述种子生成掩码；

[0057] 遮掩去除模块，用于使用所述掩码对所述遮掩消息去除遮掩，以获得原消息；

[0058] 原消息发送模块，用于将所述原消息发送至所述应用。

[0059] 第九方面，本发明实施例还提供了一种计算机设备，所述计算机设备包括：

[0060] 一个或多个处理器；

[0061] 存储器，用于存储一个或多个程序；

[0062] 当所述一个或多个程序被所述一个或多个处理器执行，使得所述一个或多个处理器实现如第一方面所述的消息的加密方法、或者第二方面所述的消息的解密方法、或者如第三方面所述的消息的写方法、或者如第四方面所述的消息的读方法。

[0063] 第十方面，本发明实施例还提供了一种计算机可读存储介质，其上存储有计算机程序，该计算机程序被处理器执行时实现如第一方面所述的消息的加密方法、或者第二方面所述的消息的解密方法、或者如第三方面所述的消息的写方法、或者如第四方面所述的消息的读方法。

[0064] 在本实施例中，确定原消息，生成种子，根据种子生成掩码，使用掩码对原消息添加遮掩，以生成遮掩消息，对遮掩消息进行白盒加密，获得目标消息，公开目标消息与种子，本实施例容纳其他现有的白盒加密算法，并且，占用资源较少，运算速度较快，可应用于移动环境，填补了学术设计和工业需求之间的空白，将学术设计改进为实用的工业解决方案。

[0065] 生成掩码的功能可插入其他现有的白盒加密算法，而无需更改其主逻辑，不同的种子可以生成不同的掩码，针对相同的块可以加密为不同的密文，因此，可以添加CPA下语义安全功能。有了掩码作为屏蔽层保护，以增加逆向工程的难度，加密任务可以安全地在ECB模式下运行，与其他非并行模式相比，ECB模式下的白盒加密的效率更高。

[0066] 种子可以被视为动态初始化向量，但它们不需要像IV一样保密。由于种子不同，使得掩码不同，可将相同的明文块加密为不同的密文块，保护了多块加密在CPA下的语义安全。因此，种子有助于白盒加密的传播，具有种子知识的攻击者将不会获得额外优势来获取有用的明文。

附图说明

[0067] 图1为本发明实施例一提供的一种消息的加密方法的流程图；

[0068] 图2为一种白盒加密API的示意图；

[0069] 图3A与图3B为一种生成掩码的示例图；

[0070] 图4是本发明实施例二提供的一种消息的解密方法的流程图；

[0071] 图5为一种白盒解密API的示意图；

[0072] 图6是本发明实施例三提供的一种消息的写方法的流程图；

[0073] 图7是本发明实施例四提供的一种消息的读方法的流程图；

[0074] 图8为本发明实施例五提供的一种消息的加密装置的结构示意图；

[0075] 图9为本发明实施例六提供的一种消息的解密装置的结构示意图；

[0076] 图10为本发明实施例七提供的一种消息的写装置的结构示意图；

[0077] 图11为本发明实施例八提供的一种消息的读装置的结构示意图；

[0078] 图12为本发明实施例九提供的一种计算机设备的结构示意图。

具体实施方式

[0079] 下面结合附图和实施例对本发明作进一步的详细说明。可以理解的是，此处所描述的具体实施例仅仅用于解释本发明，而非对本发明的限定。另外还需要说明的是，为了便于描述，附图中仅示出了与本发明相关的部分而非全部结构。

[0080] 实施例一

[0081] 图1为本发明实施例一提供的一种消息的加密方法的流程图，本实施例提供了一种新的可抗CPA的白盒加密方案，通过添加掩码生成函数(mask generation function，MGF)，对消息中不同的块生成不同的密文，MGF在白盒加密API(Application Programming Interface，应用编程接口)中与白盒加密算法合并，该方法可以由消息的加密装置来执行，该消息的加密装置可以由软件和/或硬件实现，可配置在计算机设备中，例如，移动终端(如手机、平板电脑、智能穿戴设备(如智能手表、智能眼镜等)等)、个人电脑、服务器等。

[0082] 对于移动终端、个人电脑等本地设备，可以以应用内置模块、插件等形式实现白盒加密。

[0083] 此外，对于服务器等非本地设备，可以以SDK(Software Development Kit，软件开发套件)的形式实现白盒加密，此时，用户首先选择一个密钥，然后使用它通过服务器初始化白盒加密。初始化完成后，密钥将以非明文格式生成并存储在云中，应用(客户端)应通过云API访问密钥。在服务器实现的白盒加密消除了在本地设备上进行逆向工程的风险，此时，应用(客户端)与服务器通信以完成加密操作(例如，检索密钥)。

[0084] 如图1所示，该方法具体包括如下步骤：

[0085] S101、确定原消息。

[0086] 如图2所示，原消息(message)提供给白盒加密API作为输入，等待加密。

[0087] 其中，该原消息为明文(plaintext)，可以包含一个或多个块。

[0088] 需要说明的是，针对不同的场景，原消息的形式有所不同，例如，在移动终端等本地设备，该原消息可以为密钥，即通过本实施例中白盒加密的方式进行密钥预埋，大大降低在本地设备上进行逆向工程的风险。

[0089] 在原消息的长度不足等情况下，可在原消息中填充指定的数据，如PKCS#7，将填充之后的原消息提供给白盒加密API作为输入。

[0090] 当然，在原消息的长度足够等情况下，原消息也可以直接提供给白盒加密API作为输入，本实施例对此不加以限制。

[0091] S102、生成种子。

[0092] 如图2所示，白盒加密API收到输入的原消息后，可以生成种子(seed)，其中，该种子具有唯一性，不重复使用，可以为数字、字符串、字符中的至少一种，可以作为白盒加密API的输出公开。

[0093] 进一步而言，种子的长度一般大于或等于128比特，优选为256比特。大于或等于128比特的种子，重复生成相同种子的概率极低，即完全相同的两段明文使用同一个seed的概率极低趋近于0，忽略不计，因此，可随机生成一数值，作为种子，该数值的长度大于或等于128比特，优选为256比特，如PRNG(Pseudo Random Number Generation，伪随机数生成算法)，C语言实现中的srand使用时间作为随机源，等等。

[0094] 种子生成的过程中无需借助外接硬件设备，也并不需要要求种子达到TRNG(True

Random Number Generation,真随机数生成)的级别,操作简单。

[0095] 在本实施例中,通过对种子进行定制化的工作,使得攻击者只能被动地观察种子的值,而不能对其进行修改。

[0096] 例如,种子生成功能被合并到白盒加密API中并与代码混淆一起应用,加大代码修改的难度。

[0097] S103、根据所述种子生成掩码。

[0098] 如图2所示,MGF使用种子作为输入来产生掩码(mask),种子与掩码之间具有固定的变换关系,由于种子具有唯一性,因此,掩码也具有唯一性。

[0099] 在一种情况中,可以使用外部的哈希函数生成掩码。

[0100] 在此情况中,可确定原消息的长度,确定哈希函数(hash functions),将种子输入至哈希函数中进行运算,以生成该长度的掩码。

[0101] 其中,哈希函数又称散列函数,可以把任意长度的输入(又叫做预映射,pre-image),通过散列算法,变换成固定长度的输出,该输出就是散列值。这种转换是一种压缩映射,使得散列值的空间通常远小于输入的空间,即哈希函数为把任意长度的消息压缩到某一固定长度的消息摘要的函数。

[0102] 在一个示例中,哈希函数包括SHAKE函数,如SHAKE-128、SHAKE-256等。SHAKE函数可产生任何所需的长度的输出。

[0103] 在本示例中,可将种子作为输入至SHAKE函数中进行运算,以输出该长度的掩码,该处理方式可表示为:

[0104] F(s,N)=D

[0105] 其中,f()为SHAKE函数,s为种子,N为输出(D)的长度,N与原消息的长度L相等,D为掩码。

[0106] 一般情况下,种L和N的值至少为128位,以防止暴力攻击和彩虹表攻击,如果还考虑到哈希碰撞,则L的值应至少为256位。

[0107] 在另一个示例中,哈希函数包括SHA(Secure Hash Algorithm,安全哈希算法)函数,如SHA-1、SHA-2、SHA-3。

[0108] SHA函数的思想是接收一段明文,以一种不可逆的方式将它转换成一段(通常更小)密文,也可以简单的理解为取一串输入码(称为预映射或信息),并把它们转化为长度较短、位数(160-512)固定的输出序列,即散列值(也称为信息摘要或信息认证代码)的过程。散列函数值可以说是对明文的一种“指纹”或是“摘要”,所以对散列值的数字签名就可以视为对此明文的数字签名。

[0109] 需要说明的是,SHAKE函数在是SHA-3系列中的可扩展输出哈希函数,在本示例中,SHA-3表示SHA3-224、SHA3-256、SHA3-384、SHA3-512,并不包含SHAKE函数。

[0110] 在本示例中,在种子的基础上生成多个新的种子,将所有种子分别输入至SHA函数中进行运算,以分别输出多个摘要信息,组合多个摘要信息,获得第一数据集,从第一数据集中提取该长度的数据,作为掩码。

[0111] 以SHA-1进行说明,如图3A所示,SHA-1产生160位摘要信息,如果原消息的长度为N位,则将生成总数(N/160+1)个摘要信息,若N=200,则生成(200/160+1)=2个摘要信息,若N=320,则生成(320/160+1)=3个摘要信息,等等。

[0112] 每个SHA-1计算的输入是种子加上增量n,即seed,seed+1,seed+2,…,seed+n(n＝0、1、2…),作为新的种子,分别使用seed,seed+1,seed+2,…,seed+n输入至SHA-1中,生成摘要信息,并按顺序将摘要信息串联在一起,得到第一数据集。

[0113] 最后,删除第一数据集右端多于N位的数据,剩余的N位数据即为掩码。

[0114] 在一种情况中,不使用外部的功能,而重复使用白盒加密功能生成掩码。

[0115] 在此情况中,可确定原消息的长度,对种子进行白盒加密,以生成长度的掩码。

[0116] 在具体实现中,可在种子的基础生成多个新的种子,将所有种子分别进行白盒加密,以分别生成多个密文,组合多个密文,获得第二数据集,从第二数据集中提取长度的数据,作为掩码。

[0117] 在一个示例中,如图3B所示,白盒加密(E_wb)产生128位密文,如果原消息的长度为N位,则将生成总数(N/128+1)个密文。

[0118] 每个SHA-1计算的输入是种子加上增量n,即seed,seed+1,seed+2,…,seed+n(n＝0、1、2…),作为新的种子,分别使用seed,seed+1,seed+2,…,seed+n输入至白盒加密(E_wb)函数中,生成密文,并按顺序将密文串联在一起,得到第二数据集。

[0119] 最后,删除第二数据集右端多于N位的数据,剩余的N位数据即为掩码。

[0120] 需要说明的是,掩码的长度与原消息的长度保持相同,可以方便后续对原消息与掩码进行异或运算等操作,当然,掩码的长度也可以大于明文,在原消息与掩码进行异或运算等操作时,截掉掩码中多余的部分数据(即多于原消息的部分数据)即可,本实施例对此不加以限制。

[0121] 当然,上述生成掩码的方式只是作为示例,在实施本实施例时,可以根据实际情况设置其他生成掩码的方式,哈希函数可以是任何标准的安全单向散列函数,包括但不限于SHAKE函数、SHA函数,本实施例对此不加以限制。另外,除了上述生成掩码的方式外,本领域技术人员还可以根据实际需要采用其它生成掩码的方式,本实施例对此也不加以限制。

[0122] S104、使用所述掩码对所述原消息添加遮掩,以生成遮掩消息。

[0123] 如图2所示,将掩码应用于明文(原消息)中,对原消息进行变换,遮掩其原本的内容,得到变换后的遮掩消息。

[0124] 在一个示例中,在原消息的长度与掩码的长度相等的情况下,可以对原消息与掩码进行异或运算XOR,获得遮掩消息,表示如下:

[0125] $m' = m \oplus D$

[0126] 其中,m为原消息,D为掩码,m'为遮掩消息。

[0127] 当然,上述使用异或运算生成遮掩消息的方式只是作为示例,在实施本实施例时,可以根据实际情况设置其他生成遮掩消息的方式,例如,在原消息的基础上使用掩码进行插值,获得遮掩消息,或者,将原消息与掩码相同位的数据代入固定的函数式中进行运算,获得遮掩消息,等等,而在安全级别基本相同的情况下,异或运算的速度较快,可优先使用异或运算,本实施例对此不加以限制。另外,除了上述生成遮掩消息的方式外,本领域技术人员还可以根据实际需要采用其它生成遮掩消息的方式,本实施例对此也不加以限制。

[0128] S105、对所述遮掩消息进行白盒加密,获得目标消息。

[0129] 如图2所示,在ECB等模式下,可使用白盒加密算法对遮掩消息进行白盒加密,以生成密文,作为目标消息。

[0130] 由于原消息(明文)具有一个或多个块,因此,目标消息(密文)也同样具有一个或多个块。

[0131] 白盒加密属于对称加密,是指能够在白盒环境下抵御攻击的一种特殊的加密方法。

[0132] 白盒加密的思想是混淆,混淆的作用是扰乱信息,是让信息以一种完全无法理解的形式存在,尽量让人无法理解中间的过程(也就是只能看到输入和输出,但无法理解结果是如何得到的),但不影响信息本身发挥作用(一个加了密的程序,在源码未解密前是无法执行的,但是经过混淆的程序,可以正确执行)。

[0133] 白盒加密将算法和密钥紧密捆绑在了一起,由算法和密钥生成一个加密表和一个解密表,然后可以独立用查找加密表来加密,用解密表解密,不再依赖于原来的加解密算法和密钥。

[0134] 正是由于算法和密钥的合并,所有可以有效隐藏密钥,与此同时也混淆了加密逻辑。具体而言白盒加密的一种实现思路就是将算法完全用查表来替代,因为算法已知,加密的密钥已知。所以将算法和密钥固化成查表表示,这就是白盒密钥的实现过程。

[0135] 在本实施例中,可以通过一些定制的工程工作在白盒加密中实现AES(Advanced Encryption Standard,高级加密标准),对称密钥信息存储在混淆的查找表中。

[0136] 现有的白盒加密算法查找表的生成遵循相同的方法和组件-组合的T-box和T_y表,以及XOR表。此外,现有的白盒加密算法没有给出强制性的表结构。

[0137] 因此,表格可使用本实施例定制的结构,并为每个表格添加额外的掩码,以增加逆向工程的难度。

[0138] 此外,内部/外部编码和混合双射也应用于所有查找表。

[0139] 在具体实现中,使用一个大小为数百KB的二进制文件存储密钥的信息。在加密/解密过程中,白盒加密API将读取二进制文件以加载密钥。如果密钥改变,二进制文件很容易被替换。

[0140] S106、公开所述目标消息与所述种子。

[0141] 如图2所示,在获得密文(即目标消息)与种子之后,则可以向公开该目标消息与种子。

[0142] 白盒解密属于对称解密,是指能够在白盒环境下抵御攻击的一种特殊的加密方法。

[0143] 在本实施例中,确定原消息,生成种子,根据种子生成掩码,使用掩码对原消息添加遮掩,以生成遮掩消息,对遮掩消息进行白盒加密,获得目标消息,公开目标消息与种子,本实施例容纳其他现有的白盒加密算法,并且,占用资源较少,运算速度较快,可应用于移动环境,填补了学术设计和工业需求之间的空白,将学术设计改进为实用的工业解决方案。

[0144] 生成掩码的功能可插入其他现有的白盒加密算法,而无需更改其主逻辑,不同的种子可以生成不同的掩码,针对相同的块可以加密为不同的密文,因此,可以添加CPA下语义安全功能。有了掩码作为屏蔽层保护,以增加逆向工程的难度,加密任务可以安全地在ECB模式下运行,与其他非并行模式相比,ECB模式下的白盒加密的效率更高。

[0145] 种子可以被视为动态初始化向量,但它们不需要像IV一样保密。由于种子不同,使得掩码不同,可将相同的明文块加密为不同的密文块,保护了多块加密在CPA下的语义安

全。因此，种子有助于白盒加密的传播，具有种子知识的攻击者将不会获得额外优势来获取有用的明文。

[0146] 在一个示例中，下表显示了3块明文，它已经通过现有CPA下语义不安全的白盒加密(第2行)和本实施例中CPA下语义安全的白盒加密(第3行)使用相同的AES-128对称密钥进行了加密。

[0147]

明文	7ece650eb214 c97c97438621 bdaec791	145ad974369b 15021306a790 07892bb3	145ad974369b 15021306a790 07892bb3
CPA 下语义不安全的白盒加密	d4144ff0e4d7 180e8f0ce880 3597e9e8	23ab94bc7174 20f89b58d43a b6a71ec0	23ab94bc7174 20f89b58d43a b6a71ec0

[0148]

CPA 下语义安全的白盒加密	f03c572a9c21 9941fee784ae 954a98a4	2664e859b175 697c5f1b193c 74a6e2da	908d8156ec92 76aeda9d3796 3b2db336	6b89c75c95be55 9db25fc34b1a24 c9ad

[0149] 其中，第3行最后一个块为种子，128位种子附加在密文的末尾，用于恢复掩码值。

[0150] 从上表可以看出，明文中块2和块3中的明文相同，被现有CPA下语义不安全的白盒加密算法加密为相同的密文，没有考虑CPA下的语义安全性。

[0151] 本实施例中CPA下语义安全的白盒加密算法使用不同的种子生成不同的掩码对明文进行加密，加密为不同的密文，每个块中的密文具有独立值。

[0152] 此外，本实施例中CPA下语义安全的白盒加密(AES-128)在C++项目中实现并在移动环境上进行了测试。其中，OLLVM(Obfuscator-LLVM)也用于混淆源代码。测试的结果表明：

[0153] 查找表需要大约350KB的额外存储空间。

[0154] 单个加密操作平均花费2.9毫秒。

[0155] 因此，如果表与应用集成，则这样的大小是可以接受的，在实际的移动环境中部署该本实施例中CPA下语义安全的白盒加密算法是可以承受的，而负担得起的表格大小(数百KB)是遵循现有的白盒加密算法的一个重要原因。

[0156] 实施例二

[0157] 图4为本发明实施例二提供的一种消息的解密方法的流程图，本实施例提供了一种新的可抗CPA的白盒解密方案，通过添加掩码生成函数MGF，对消息中不同的块生成不同的密文，MGF在白盒解密API中与白盒解密算法合并，该方法可以由消息的解密装置来执行，该消息的解密装置可以由软件和/或硬件实现，可配置在计算机设备中，例如，移动终端(如手机、平板电脑、智能穿戴设备(如智能手表、智能眼镜等)等)、个人电脑、服务器等。

[0158] 对于移动终端、个人电脑等本地设备，可以以应用内置模块、插件等形式实现白盒解密。

[0159] 此外，对于服务器等非本地设备，可以以SDK的形式实现白盒解密，此时，用户首先

选择一个密钥，然后使用它通过服务器初始化白盒解密。初始化完成后，密钥将以非明文格式生成并存储在云中，应用(客户端)应通过云API访问密钥。在服务器实现的白盒解密消除了在本地设备上进行逆向工程的风险，此时，应用(客户端)与服务器通信以完成解密操作(例如，检索密钥)。

[0160] 如图4所示，该方法具体包括如下步骤：

[0161] S401、确定目标消息和种子。

[0162] 如图5所示，目标消息为经过白盒加密API使用种子(seed)对原消息加密之后的密文，目标消息与种子一同公开，通过公开的渠道可获取目标消息与种子，并将目标消息与种子提供给白盒解密API作为输入，等待解密。

[0163] 其中，该种子具有唯一性，不重复使用，可以为数字、字符串、字符中的至少一种。

[0164] 需要说明的是，针对不同的场景，原消息的形式有所不同，例如，在移动终端等本地设备，该原消息可以为密钥，即通过本实施例中白盒解密的方式进行密钥预埋，大大降低在本地设备上进行逆向工程的风险。

[0165] S402、对所述目标消息进行白盒解密，获得遮掩消息。

[0166] 如图5所示，在ECB等模式下，可使用白盒解密算法对目标消息进行白盒解密，还原遮掩消息，其中，对目标消息进行白盒解密算法的白盒解密算法与对目标消息进行白盒加密的白盒加密算法配对。

[0167] 在本实施例中，可以通过一些定制的工程工作在白盒解密中实现AES。

[0168] S403、根据所述种子生成掩码。

[0169] 如图5所示，MGF使用种子作为输入来产生掩码，种子与掩码之间具有固定的变换关系，由于种子具有唯一性，因此，掩码也具有唯一性。

[0170] 在一种情况中，可以使用外部的哈希函数生成掩码。

[0171] 在此情况中，可确定原消息的长度，确定哈希函数，将种子输入至哈希函数中进行运算，以生成该长度的掩码。

[0172] 在一个示例中，哈希函数包括SHAKE函数，如SHAKE-128、SHAKE-256等。SHAKE函数可产生任何所需的长度的输出。

[0173] 在本示例中，可将种子作为输入至SHAKE函数中进行运算，以输出该长度的掩码

[0174] 在另一个示例中，哈希函数包括SHA函数，如SHA-1、SHA-2、SHA-3。

[0175] 在本示例中，在种子的基础上生成多个新的种子，将所有种子分别输入至SHA函数中进行运算，以分别输出多个摘要信息，组合多个摘要信息，获得第一数据集，从第一数据集中提取该长度的数据，作为掩码。

[0176] 在另一种情况中，不使用外部的功能，而重复使用白盒加密功能生成掩码。

[0177] 在此情况中，可确定原消息的长度，对种子进行白盒加密，以生成该长度的掩码。

[0178] 在具体实现中，可在种子的基础生成多个新的种子，将所有种子分别进行白盒加密，以分别生成多个密文，组合多个密文，获得第二数据集，从第二数据集中提取长度的数据，作为掩码。

[0179] 当然，上述生成掩码的方式只是作为示例，在实施本实施例时，可以根据实际情况设置其他生成掩码的方式，哈希函数可以是任何标准的安全单向散列函数，包括但不限于SHAKE函数、SHA函数，本实施例对此不加以限制。另外，除了上述生成掩码的方式外，本领域

技术人员还可以根据实际需要采用其它生成掩码的方式，本实施例对此也不加以限制。

[0180] 需要说明的是，在本实施例中，由于生成掩码的方式与实施例一生成掩码的方式应用基本相似，所以描述的比较简单，相关之处参见实施例一的部分说明即可，本实施例在此不加以详述。

[0181] S404、使用所述掩码对所述遮掩消息去除遮掩，获得原消息。

[0182] 如图5所示，将掩码应用于遮掩消息中，对遮掩消息进行变换，还原其原本的内容，得到变换后的明文(原消息)。

[0183] 其中，白盒解密API中使用掩码对所述原消息添加遮掩的方式与白盒解密API中使用掩码对遮掩消息去除遮掩的方式配对。

[0184] 在一个示例中，若在先对原消息与掩码进行异或运算、生成遮掩消息，则可以对目标消息与掩码进行异或运算，获得原消息，表示如下：

[0185] $m = m' \oplus D$

[0186] 其中，m为原消息，D为掩码，m'为遮掩消息

[0187] 当然，上述使用异或运算还原原消息的方式只是作为示例，在实施本实施例时，可以根据实际情况设置其他还原原消息的方式，例如，若在先在原消息的基础上使用掩码进行插值、生成遮掩消息，则可以在遮掩消息中去除插值(掩码)、生成原消息，或者，若在先将原消息与掩码相同位的数据代入固定的函数式中进行运算、生成遮掩消息，则可以将遮掩消息与掩码相同位的数据代入该函数式中进行逆运算、生成原消息，等等，而在安全级别基本相同的情况下，异或运算的速度较快，可优先使用异或运算，本实施例对此不加以限制。另外，除了上述还原原消息的方式外，本领域技术人员还可以根据实际需要采用其它还原原消息的方式，本实施例对此也不加以限制。

[0188] 在原消息的长度不足等情况下，在提供给白盒加密API之前，可能在原消息中填充指定的数据，如PKCS#7，此时，可从原消息中去除在先填充的数据。

[0189] 当然，在原消息的长度足够等情况下，原消息直接提供给白盒加密API作为输入，此时，无需从原消息中去除在先填充的数据，本实施例对此不加以限制。

[0190] 在本实施例中，确定目标消息和种子，对目标消息进行白盒解密，获得遮掩消息，根据种子生成掩码，使用掩码对所述遮掩消息去除遮掩，获得原消息，本实施例容纳其他现有的白盒解密算法，并且，占用资源较少，运算速度较快，可应用于移动环境，填补了学术设计和工业需求之间的空白，将学术设计改进为实用的工业解决方案。

[0191] 生成掩码的功能可插入其他现有的白盒解密算法，而无需更改其主逻辑，不同的种子可以生成不同的掩码，针对相同的块可以加密为不同的密文，因此，可以添加CPA下语义安全功能。有了掩码作为屏蔽层保护，以增加逆向工程的难度，解密任务可以安全地在ECB模式下运行，与其他非并行模式相比，ECB模式下的白盒解密的效率更高。

[0192] 种子可以被视为动态初始化向量，但它们不需要像IV一样保密。由于种子不同，使得掩码不同，可将相同的明文块加密为不同的密文块，保护了多块加密在CPA下的语义安全。因此，种子有助于白盒解密的传播，具有种子知识的攻击者将不会获得额外优势来获取有用的明文。

[0193] 实施例三

[0194] 图6为本发明实施例三提供的一种消息的写方法的流程图，本实施例提供了一种

新的可抗CPA的白盒加密方案，通过添加掩码生成函数MGF，对消息中不同的块生成不同的密文进行存储，MGF在白盒加密API(Application Programming Interface，应用编程接口)中与白盒加密算法合并，该方法可以由消息的写装置来执行，该消息的写装置可以由软件和/或硬件实现，可配置在计算机设备中，例如，移动终端(如手机、平板电脑、智能穿戴设备(如智能手表、智能眼镜等)等)、个人电脑、服务器等。

[0195] 对于移动终端、个人电脑等本地设备，可以以应用内置模块、插件等形式实现白盒加密。

[0196] 此外，对于服务器等非本地设备，可以以SDK(Software Development Kit，软件开发套件)的形式实现白盒加密，此时，用户首先选择一个密钥，然后使用它通过服务器初始化白盒加密。初始化完成后，密钥将以非明文格式生成并存储在云中，应用(客户端)应通过云API访问密钥。在服务器实现的白盒加密消除了在本地设备上进行逆向工程的风险，此时，应用(客户端)与服务器通信以完成加密操作(例如，检索密钥)。

[0197] 如图6所示，该方法具体包括如下步骤：

[0198] S601、当接收到应用生成的原消息时，生成种子。

[0199] 在具体实现中，应用可以包括浏览器、电子邮箱、记事本、通讯录等，该应用在运行的过程中会产生不同安全级别的数据。

[0200] 如图2所示，部分安全级别较高的数据可以作为原消息(message)，提供给白盒加密API作为输入，等待加密。

[0201] 其中，该原消息为明文(plaintext)，可以包含一个或多个块。

[0202] 需要说明的是，针对不同的场景，原消息的形式有所不同，例如，在移动终端等本地设备，该原消息可以为密钥，即通过本实施例中白盒加密的方式进行密钥预埋，大大降低在本地设备上进行逆向工程的风险。

[0203] 又例如，对于在移动终端中的通讯录，其可产生联系人信息等数据，如名称、手机号码、头像等，一般情况下，联系人信息中等，用户解锁移动终端之后可浏览该联系人信息，其他应用在授权的情况下可读取联系人信息，对于部分较为隐私的联系人信息，用户可以请求通讯录对该联系人信息进行加密，提高其安全性，用户解锁移动终端之后不可直接浏览该联系人信息，其他应用在不可读取该联系人信息，在解密联系人信息之后可浏览该联系人信息。

[0204] 如图2所示，白盒加密API收到输入的原消息后，可以生成种子(seed)，其中，该种子具有唯一性，不重复使用，可以为数字、字符串、字符中的至少一种，可以作为白盒加密API的输出公开。

[0205] S602、根据所述种子生成掩码。

[0206] 如图2所示，MGF使用种子作为输入来产生掩码(mask)，种子与掩码之间具有固定的变换关系，由于种子具有唯一性，因此，掩码也具有唯一性。

[0207] 在一种情况中，可以使用外部的哈希函数生成掩码。

[0208] 在此情况中，可确定原消息的长度，确定哈希函数(hash functions)，将种子输入至哈希函数中进行运算，以生成该长度的掩码。

[0209] 在一个示例中，哈希函数包括SHAKE函数，如SHAKE-128、SHAKE-256等。SHAKE函数可产生任何所需的长度的输出。

[0210] 在另一个示例中，哈希函数包括SHA函数，如SHA-1、SHA-2、SHA-3。

[0211] 在本示例中，在种子的基础上生成多个新的种子，将所有种子分别输入至SHA函数中进行运算，以分别输出多个摘要信息，组合多个摘要信息，获得第一数据集，从第一数据集中提取该长度的数据，作为掩码。

[0212] 在另一种情况中，不使用外部的功能，而重复使用白盒加密功能生成掩码。

[0213] 在此情况中，可确定原消息的长度，对种子进行白盒加密，以生成该长度的掩码。

[0214] 在具体实现中，可在种子的基础生成多个新的种子，将所有种子分别进行白盒加密，以分别生成多个密文，组合多个密文，获得第二数据集，从第二数据集中提取长度的数据，作为掩码。

[0215] 当然，上述生成掩码的方式只是作为示例，在实施本实施例时，可以根据实际情况设置其他生成掩码的方式，哈希函数可以是任何标准的安全单向散列函数，包括但不限于SHAKE函数、SHA函数，本实施例对此不加以限制。另外，除了上述生成掩码的方式外，本领域技术人员还可以根据实际需要采用其它生成掩码的方式，本实施例对此也不加以限制。S603、使用所述掩码对所述原消息添加遮掩，以生成遮掩消息。

[0216] 如图2所示，将掩码应用于明文(原消息)中，对原消息进行变换，遮掩其原本的内容，得到变换后的遮掩消息。

[0217] 在一个示例中，在原消息的长度与掩码的长度相等的情况下，可以对原消息与掩码进行异或运算XOR，获得遮掩消息。

[0218] 当然，上述使用异或运算生成遮掩消息的方式只是作为示例，在实施本实施例时，可以根据实际情况设置其他生成遮掩消息的方式，例如，在原消息的基础上使用掩码进行插值，获得遮掩消息，或者，将原消息与掩码相同位的数据代入固定的函数式中进行运算，获得遮掩消息，等等，而在安全级别基本相同的情况下，异或运算的速度较快，可优先使用异或运算，本实施例对此不加以限制。另外，除了上述生成遮掩消息的方式外，本领域技术人员还可以根据实际需要采用其它生成遮掩消息的方式，本实施例对此也不加以限制。

[0219] S604、对所述遮掩消息进行白盒加密，获得目标消息。

[0220] 如图2所示，在ECB等模式下，可使用白盒加密算法对遮掩消息进行白盒加密，以生成密文，作为目标消息。

[0221] 由于原消息(明文)具有一个或多个块，因此，目标消息(密文)也同样具有一个或多个块。

[0222] S605、对所述原消息存储所述目标消息和所述种子。

[0223] 如图2所示，在获得密文(即目标消息)与种子之后，则可以向将目标消息与种子一同存储在数据库等存储空间，并在数据库等存储空间中建立目标消息、种子与原消息之间的映射关系。

[0224] 需要说明的是，在本实施例中，由于加密的方式与实施例一加密的方式应用基本相似，所以描述的比较简单，相关之处参见实施例一的部分说明即可，本实施例在此不加以详述。

[0225] 在本实施例中，当接收到应用生成的原消息时，生成种子，根据种子生成掩码，使用掩码对原消息添加遮掩，以生成遮掩消息，对遮掩消息进行白盒加密，获得目标消息，对原消息存储目标消息和种子，本实施例容纳其他现有的白盒加密算法，并且，占用资源较

少，运算速度较快，可应用于移动环境，填补了学术设计和工业需求之间的空白，将学术设计改进为实用的工业解决方案。

[0226] 生成掩码的功能可插入其他现有的白盒加密算法，而无需更改其主逻辑，不同的种子可以生成不同的掩码，针对相同的块可以加密为不同的密文，因此，可以添加CPA安全功能。有了掩码作为屏蔽层保护，以增加逆向工程的难度，加密任务可以安全地在ECB模式下运行，与其他非并行模式相比，ECB模式下的白盒加密的效率更高。

[0227] 种子可以被视为动态初始化向量，但它们不需要像IV一样保密。由于种子不同，使得掩码不同，可将相同的明文块加密为不同的密文块，保护了多块加密在CPA下的语义安全。因此，种子有助于白盒加密的传播，具有种子知识的攻击者将不会获得额外优势来获取有用的明文。

[0228] 实施例四

[0229] 图7为本发明实施例四提供的一种消息的读方法的流程图，本实施例提供了一种新的可抗CPA的白盒解密方案，通过添加掩码生成函数MGF，对消息中不同的块生成不同的密文，MGF在白盒解密API中与白盒解密算法合并，该方法可以由消息的读装置来执行，该消息的读装置可以由软件和/或硬件实现，可配置在计算机设备中，例如，移动终端(如手机、平板电脑、智能穿戴设备(如智能手表、智能眼镜等)等)、个人电脑、服务器等。

[0230] 对于移动终端、个人电脑等本地设备，可以以应用内置模块、插件等形式实现白盒解密。

[0231] 此外，对于服务器等非本地设备，可以以SDK的形式实现白盒解密，此时，用户首先选择一个密钥，然后使用它通过服务器初始化白盒解密。初始化完成后，密钥将以非明文格式生成并存储在云中，应用(客户端)应通过云API访问密钥。在服务器实现的白盒解密消除了在本地设备上进行逆向工程的风险，此时，应用(客户端)与服务器通信以完成解密操作(例如，检索密钥)。

[0232] 如图7所示，该方法具体包括如下步骤：

[0233] S701、当接收到应用对原消息的读操作时，确定所述原消息对应的目标消息和种子。

[0234] 如图5所示，目标消息为经过白盒加密API使用种子(seed)对原消息加密之后的密文，目标消息与种子一同存储在数据库等存储空间，在存储空间中存储目标消息、种子与原消息之间的映射关系。

[0235] 应用在运行过程中，按照需求执行度操作，以读取原消息，此时，可在存储空间通过该映射关系查找目标消息与种子，并将目标消息与种提供给白盒解密API作为输入，等待解密。

[0236] 其中，该种子具有唯一性，不重复使用，可以为数字、字符串、字符中的至少一种。

[0237] S702、对所述目标消息进行白盒解密，获得遮掩消息。

[0238] 如图5所示，在ECB等模式下，可使用白盒解密算法对目标消息进行白盒解密，还原遮掩消息，其中，对目标消息进行白盒解密算法的白盒解密算法与对目标消息进行白盒加密的白盒加密算法配对。

[0239] S703、根据所述种子生成掩码。

[0240] 如图5所示，MGF使用种子作为输入来产生掩码，种子与掩码之间具有固定的变换

关系，由于种子具有唯一性，因此，掩码也具有唯一性。

[0241] 在一种情况中，可以使用外部的哈希函数生成掩码。

[0242] 在此情况中，可确定原消息的长度，确定哈希函数，将种子输入至哈希函数中进行运算，以生成该长度的掩码。

[0243] 在一个示例中，哈希函数包括SHAKE函数，如SHAKE-128、SHAKE-256等。SHAKE函数可产生任何所需的长度的输出。

[0244] 在另一个示例中，哈希函数包括SHA函数，如SHA-1、SHA-2、SHA-3。

[0245] 在本示例中，在种子的基础上生成多个新的种子，将所有种子分别输入至SHA函数中进行运算，以分别输出多个摘要信息，组合多个摘要信息，获得第一数据集，从第一数据集中提取该长度的数据，作为掩码。

[0246] 在另一种情况中，不使用外部的功能，而重复使用白盒加密功能生成掩码。

[0247] 在此情况中，可确定原消息的长度，对种子进行白盒加密，以生成该长度的掩码。

[0248] 在具体实现中，可在种子的基础生成多个新的种子，将所有种子分别进行白盒加密，以分别生成多个密文，组合多个密文，获得第二数据集，从第二数据集中提取长度的数据，作为掩码。

[0249] 当然，上述生成掩码的方式只是作为示例，在实施本实施例时，可以根据实际情况设置其他生成掩码的方式，哈希函数可以是任何标准的安全单向散列函数，包括但不限于SHAKE函数、SHA函数，本实施例对此不加以限制。另外，除了上述生成掩码的方式外，本领域技术人员还可以根据实际需要采用其它生成掩码的方式，本实施例对此也不加以限制。

[0250] S704、使用所述掩码对所述遮掩消息去除遮掩，获得原消息。

[0251] 如图5所示，将掩码应用于遮掩消息中，对遮掩消息进行变换，还原其原本的内容，得到变换后的明文(原消息)。

[0252] 其中，白盒解密API中使用掩码对所述原消息添加遮掩的方式与白盒解密API中使用掩码对遮掩消息去除遮掩的方式配对。

[0253] 在一个示例中，若在先对原消息与掩码进行异或运算、生成遮掩消息，则可以对目标消息与掩码进行异或运算，获得原消息。

[0254] 当然，上述使用异或运算还原原消息的方式只是作为示例，在实施本实施例时，可以根据实际情况设置其他还原原消息的方式，例如，若在先在原消息的基础上使用掩码进行插值、生成遮掩消息，则可以在遮掩消息中去除插值(掩码)、生成原消息，或者，若在先将原消息与掩码相同位的数据代入固定的函数式中进行运算、生成遮掩消息，则可以将遮掩消息与掩码相同位的数据代入该函数式中进行逆运算、生成原消息，等等，而在安全级别基本相同的情况下，异或运算的速度较快，可优先使用异或运算，本实施例对此不加以限制。另外，除了上述还原原消息的方式外，本领域技术人员还可以根据实际需要采用其它还原原消息的方式，本实施例对此也不加以限制。

[0255] 在原消息的长度不足等情况下，在提供给白盒加密API之前，可能在原消息中填充指定的数据，如PKCS#7，此时，可从原消息中去除在先填充的数据。

[0256] 当然，在原消息的长度足够等情况下，原消息直接提供给白盒加密API作为输入，此时，无需从原消息中去除在先填充的数据，本实施例对此不加以限制。

[0257] S705、将所述原消息发送至所述应用。

[0258] 在解析得到原消息之后，则可以将该原消息返回给应用，实现原消息的读操作。

[0259] 需要说明的是，在本实施例中，由于解密的方式与实施例二解密的方式应用基本相似，所以描述的比较简单，相关之处参见实施例二的部分说明即可，本实施例在此不加以详述。

[0260] 在本实施例中，当接收到应用对原消息的读操作时，确定原消息对应的目标消息和种子，对目标消息进行白盒解密，获得遮掩消息，根据种子生成掩码，使用掩码对遮掩消息去除遮掩，以获得原消息，将原消息发送至应用，本实施例容纳其他现有的白盒解密算法，并且，占用资源较少，运算速度较快，可应用于移动环境，填补了学术设计和工业需求之间的空白，将学术设计改进为实用的工业解决方案。

[0261] 生成掩码的功能可插入其他现有的白盒解密算法，而无需更改其主逻辑，不同的种子可以生成不同的掩码，针对相同的块可以加密为不同的密文，因此，可以添加CPA下语义安全功能。有了掩码作为屏蔽层保护，以增加逆向工程的难度，加密任务可以安全地在ECB模式下运行，与其他非并行模式相比，ECB模式下的白盒加密的效率更高。

[0262] 种子可以被视为动态初始化向量，但它们不需要像IV一样保密。由于种子不同，使得掩码不同，可将相同的明文块加密为不同的密文块，保护了多块加密在CPA下的语义安全。因此，种子有助于白盒加密的传播，具有种子知识的攻击者将不会获得额外优势来获取有用的明文。

[0263] 实施例五

[0264] 图8为本发明实施例五提供的一种消息的加密装置的结构示意图，该装置具体可以包括如下模块：

[0265] 原消息确定模块801，用于确定原消息；

[0266] 种子生成模块802，用于生成种子；

[0267] 掩码生成模块803，用于根据所述种子生成掩码；

[0268] 遮掩添加模块804，用于使用所述掩码对所述原消息添加遮掩，以生成遮掩消息；

[0269] 白盒加密模块805，用于对所述遮掩消息进行白盒加密，获得目标消息；

[0270] 密文公开模块806，用于公开所述目标消息与所述种子。

[0271] 在本发明的一个实施例中，还包括：

[0272] 数据填充模块，用于在所述原消息中填充指定的数据。

[0273] 在本发明的一个实施例中，所述种子生成模块802包括：

[0274] 随机生成子模块，用于随机生成一数值，作为种子。

[0275] 在本发明的一个实施例中，所述掩码生成模块803包括：

[0276] 第一长度确定子模块，用于确定所述原消息的长度；

[0277] 哈希函数确定子模块，用于确定哈希函数；

[0278] 哈希函数运算子模块，用于将所述种子输入至所述哈希函数中进行运算，以生成所述长度的掩码。

[0279] 在本发明实施例的一个示例中，所述哈希函数包括SHAKE函数或者SHA函数；

[0280] 所述哈希函数运算子模块包括：

[0281] 第一散列运算单元，用于将所述种子作为输入至所述SHAKE函数中进行运算，以输出所述长度的掩码；

[0282] 或者，

[0283] 第一新种子生成单元，用于在所述种子的基础生成多个新的种子；

[0284] 第二散列运算单元，用于将所有种子分别输入至SHA函数中进行运算，以分别输出多个摘要信息；

[0285] 摘要信息组合单元，用于组合多个所述摘要信息，获得第一数据集；

[0286] 第一提取单元，用于从所述第一数据集中提取所述长度的数据，作为掩码。

[0287] 在本发明的一个实施例中，所述掩码生成模块803包括：

[0288] 第一长度确定子模块，用于确定所述原消息的长度；

[0289] 内部加密子模块，用于对所述种子进行所述白盒加密，以生成所述长度的掩码。

[0290] 在本发明实施例的一个示例中，所述内部加密子模块包括：

[0291] 第二新种子生成单元，用于在所述种子的基础生成多个新的种子；

[0292] 密文生成单元，用于将所有种子分别进行白盒加密，以分别生成多个密文；

[0293] 密文组合单元，用于组合多个所述密文，获得第二数据集；

[0294] 第二提取单元，用于从所述第二数据集中提取所述长度的数据，作为掩码。

[0295] 在本发明的一个实施例中，所述遮掩添加模块804包括：

[0296] 异或运算子模块，用于对所述原消息与所述掩码进行异或运算，获得遮掩消息。

[0297] 本发明实施例所提供的消息的加密装置可执行本发明任意实施例所提供的消息的加密方法，具备执行方法相应的功能模块和有益效果。

[0298] 实施例六

[0299] 图9为本发明实施例六提供的一种消息的解密装置的结构示意图，该装置具体可以包括如下模块：

[0300] 密文确定模块901，用于确定目标消息和种子；

[0301] 白盒解密模块902，用于对所述目标消息进行白盒解密，获得遮掩消息；

[0302] 掩码生成模块903，用于根据所述种子生成掩码；

[0303] 遮掩去除模块904，用于使用所述掩码对所述遮掩消息去除遮掩，获得原消息。

[0304] 在本发明的一个实施例中，所述掩码生成模块903包括：

[0305] 第一长度确定子模块，用于确定所述原消息的长度；

[0306] 哈希函数确定子模块，用于确定哈希函数；

[0307] 哈希函数运算子模块，用于将所述种子输入至所述哈希函数中进行运算，以生成所述长度的掩码。

[0308] 在本发明实施例的一个示例中，所述哈希函数包括SHAKE函数或者SHA函数；

[0309] 所述哈希函数运算子模块包括：

[0310] 第一散列运算单元，用于将所述种子作为输入至所述SHAKE函数中进行运算，以输出所述长度的掩码；

[0311] 或者，

[0312] 第一新种子生成单元，用于在所述种子的基础生成多个新的种子；

[0313] 第二散列运算单元，用于将所有种子分别输入至SHA函数中进行运算，以分别输出多个摘要信息；

[0314] 摘要信息组合单元，用于组合多个所述摘要信息，获得第一数据集；

[0315] 第一提取单元，用于从所述第一数据集中提取所述长度的数据，作为掩码。

[0316] 在本发明的一个实施例中，所述掩码生成模块903包括：

[0317] 第一长度确定子模块，用于确定所述原消息的长度；

[0318] 内部加密子模块，用于对所述种子进行所述白盒加密，以生成所述长度的掩码。

[0319] 在本发明实施例的一个示例中，所述内部加密子模块包括：

[0320] 第二新种子生成单元，用于在所述种子的基础生成多个新的种子；

[0321] 密文生成单元，用于将所有种子分别进行白盒加密，以分别生成多个密文；

[0322] 密文组合单元，用于组合多个所述密文，获得第二数据集；

[0323] 第二提取单元，用于从所述第二数据集中提取所述长度的数据，作为掩码。

[0324] 在本发明的一个实施例中，所述遮掩去除模块904包括：

[0325] 异或运算子模块，用于对所述目标消息与所述掩码进行异或运算，获得原消息。

[0326] 在本发明的一个实施例中，还包括：

[0327] 填充去除模块，用于从所述原消息中去除在先填充的数据。

[0328] 本发明实施例所提供的消息的解密装置可执行本发明任意实施例所提供的消息的解密方法，具备执行方法相应的功能模块和有益效果。

[0329] 实施例七

[0330] 图10为本发明实施例七提供的一种消息的写装置的结构示意图，该装置具体可以包括如下模块：

[0331] 种子生成模块1001，用于当接收到应用生成的原消息时，生成种子；

[0332] 掩码生成模块1002，用于根据所述种子生成掩码；

[0333] 遮掩添加模块1003，用于使用所述掩码对所述原消息添加遮掩，以生成遮掩消息；

[0334] 白盒加密模块1004，用于对所述遮掩消息进行白盒加密，获得目标消息；

[0335] 密文存储模块1005，用于对所述原消息存储所述目标消息和所述种子。

[0336] 实施例八

[0337] 图11为本发明实施例八提供的一种消息的读装置的结构示意图，该装置具体可以包括如下模块：

[0338] 密文确定模块1101，用于当接收到应用对原消息的读操作时，确定所述原消息对应的目标消息和种子；

[0339] 白盒解密模块1102，用于对所述目标消息进行白盒解密，获得遮掩消息；

[0340] 掩码生成模块1103，用于根据所述种子生成掩码；

[0341] 遮掩去除模块1104，用于使用所述掩码对所述遮掩消息去除遮掩，以获得原消息；

[0342] 原消息发送模块1005，用于将所述原消息发送至所述应用。

[0343] 实施例九

[0344] 图12为本发明实施例九提供的一种计算机设备的结构示意图。如图12所示，该计算机设备包括处理器1200、存储器1201、通信模块1202、输入装置1203和输出装置1204。

[0345] 存储器1201作为一种计算机可读存储介质，可用于存储软件程序、计算机可执行程序以及模块，如本实施例中的消息的加密方法对应的模块(例如，如图8所示的消息的加密装置中的原消息确定模块801、种子生成模块802、掩码生成模块803、遮掩添加模块804、白盒加密模块805和密文公开模块806)、或者本实施例中的消息的解密方法对应的模块(例

如，如图9所示的消息的解密装置中的密文确定模块901、白盒解密模块902、掩码生成模块903和遮掩去除模块904)、或者本实施例中的消息的写方法对应的模块(例如，如图10所示的消息的写装置中的种子生成模块1001掩码生成模块1002、遮掩添加模块1003、白盒加密模块1004和密文存储模块1005)、或者本实施例中的消息的写方法对应的模块(例如，如图11所示的消息的写装置中的密文确定模块1101、白盒解密模块1102、掩码生成模块1103、遮掩去除模块1104和原消息发送模块1105)。处理器1200通过运行存储在存储器1201中的软件程序、指令以及模块，从而执行计算机设备的各种功能应用以及数据处理，即实现上述的消息的加密方法、或者消息的解密方法、或者消息的写方法、或者消息的读方法。

[0346] 本实施例提供的计算机设备，可执行本发明任一实施例提供的消息的加密方法、或者消息的解密方法、或者消息的写方法、或者消息的读方法，具体相应的功能和有益效果。

[0347] 实施例十

[0348] 本发明实施例十还提供一种计算机可读存储介质，其上存储有计算机程序，该计算机程序被处理器质性时实现上述消息的加密方法、消息的解密方法、消息的写方法和消息的读方法中的至少一者。

[0349] 注意，上述仅为本发明的较佳实施例及所运用技术原理。本领域技术人员会理解，本发明不限于这里所述的特定实施例，对本领域技术人员来说能够进行各种明显的变化、重新调整和替代而不会脱离本发明的保护范围。因此，虽然通过以上实施例对本发明进行了较为详细的说明，但是本发明不仅仅限于以上实施例，在不脱离本发明构思的情况下，还可以包括更多其他等效实施例，而本发明的范围由所附的权利要求范围决定。

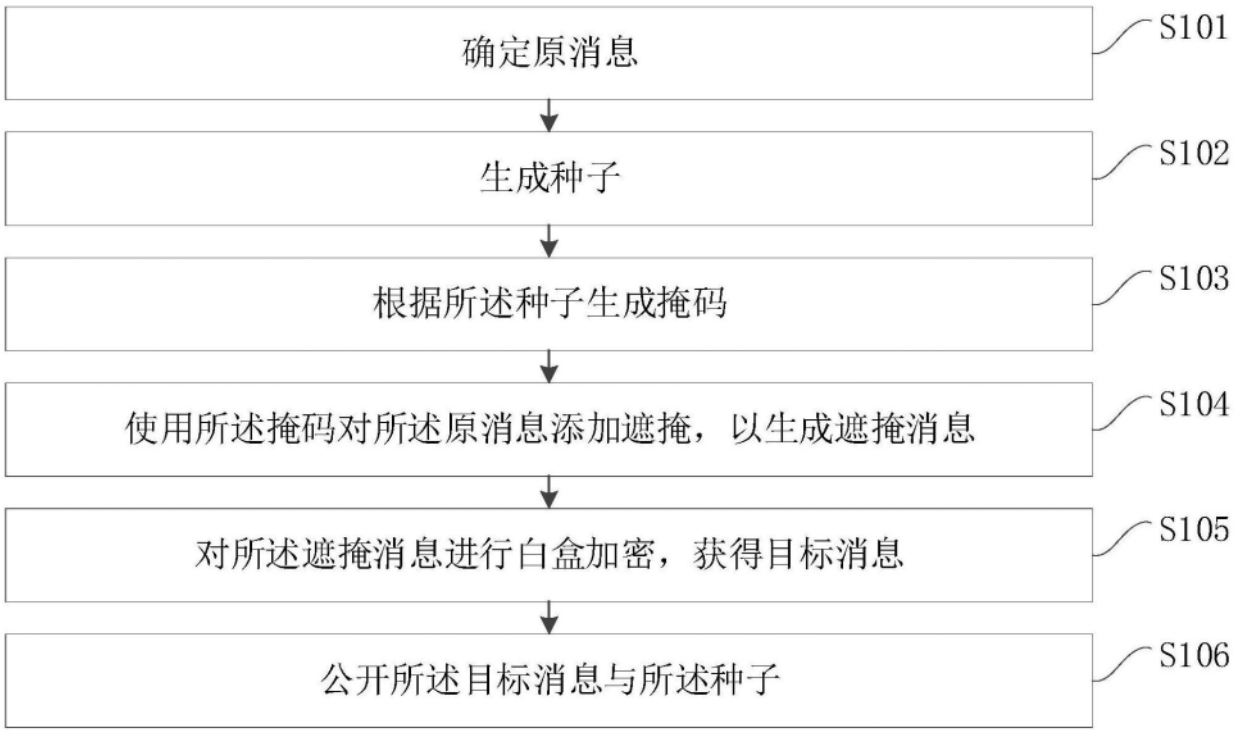

确定原消息
S101
生成种子
S102
根据所述种子生成掩码
S103
使用所述掩码对所述原消息添加遮掩，以生成遮掩消息
S104
对所述遮掩消息进行白盒加密，获得目标消息
S105
公开所述目标消息与所述种子
S106

图1

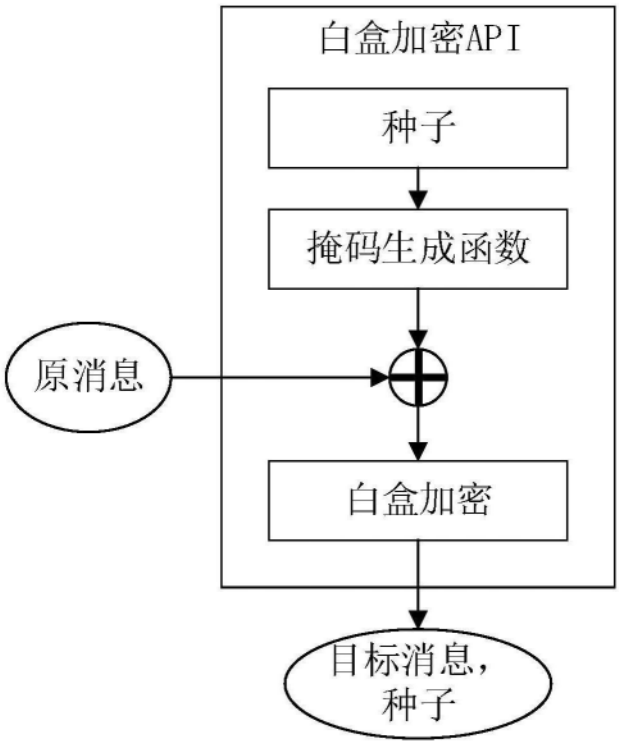

白盒加密API
种子
掩码生成函数
原消息
白盒加密
目标消息，
种子

图2

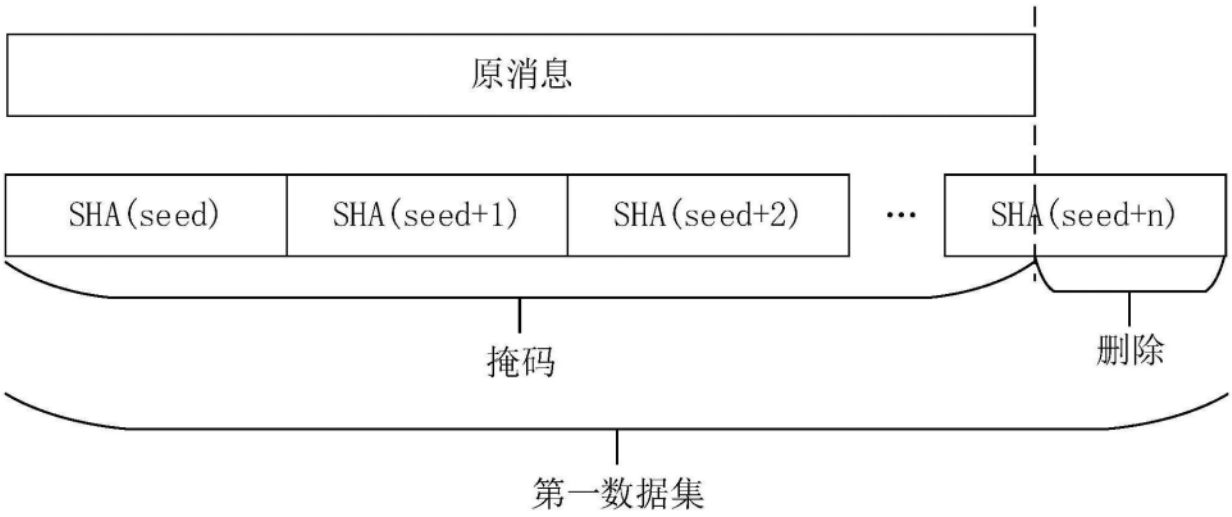

原消息
SHA(seed)
SHA(seed+1)
SHA(seed+2)
…
SHA(seed+n)
掩码
删除
第一数据集

图3A

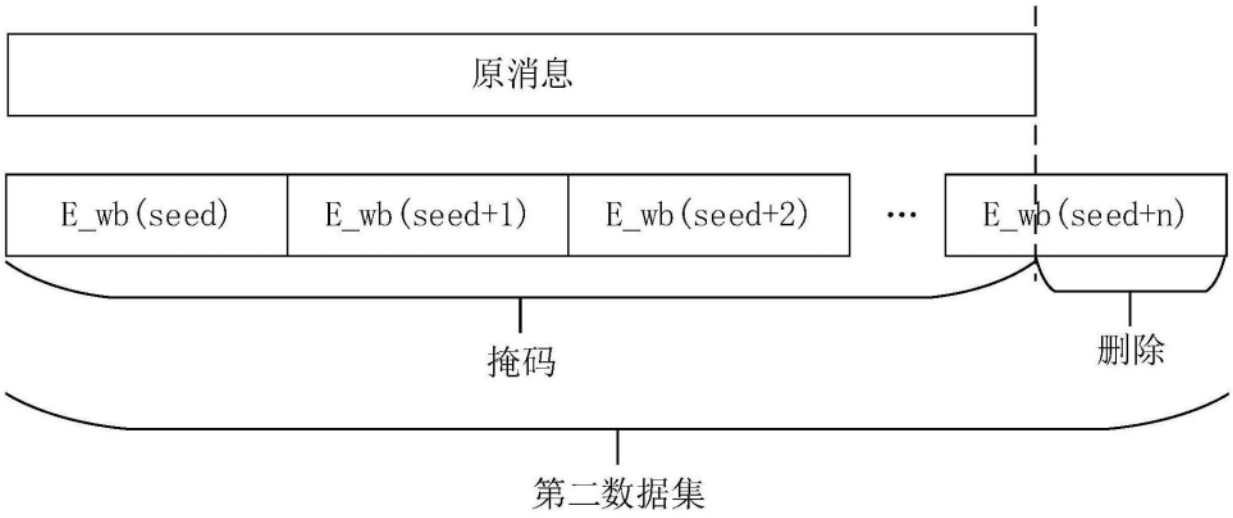

原消息
E_wb(seed)
E_wb(seed+1)
E_wb(seed+2)
…
E_wb(seed+n)
掩码
删除
第二数据集

图3B


确定目标消息和种子
S401
对所述目标消息进行白盒解密，获得遮掩消息
S402
根据所述种子生成掩码
S403
使用所述掩码对所述遮掩消息去除遮掩，获得原消息
S404

图4

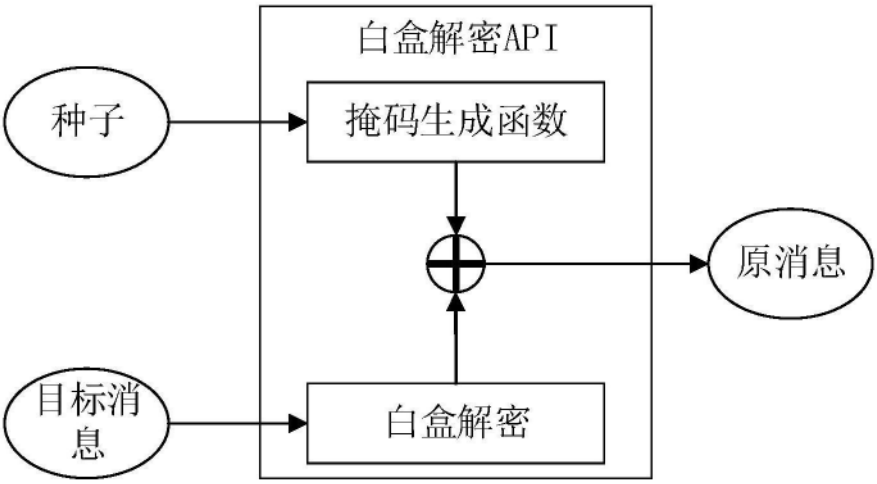

白盒解密API
种子
掩码生成函数
原消息
目标消息
白盒解密

图5

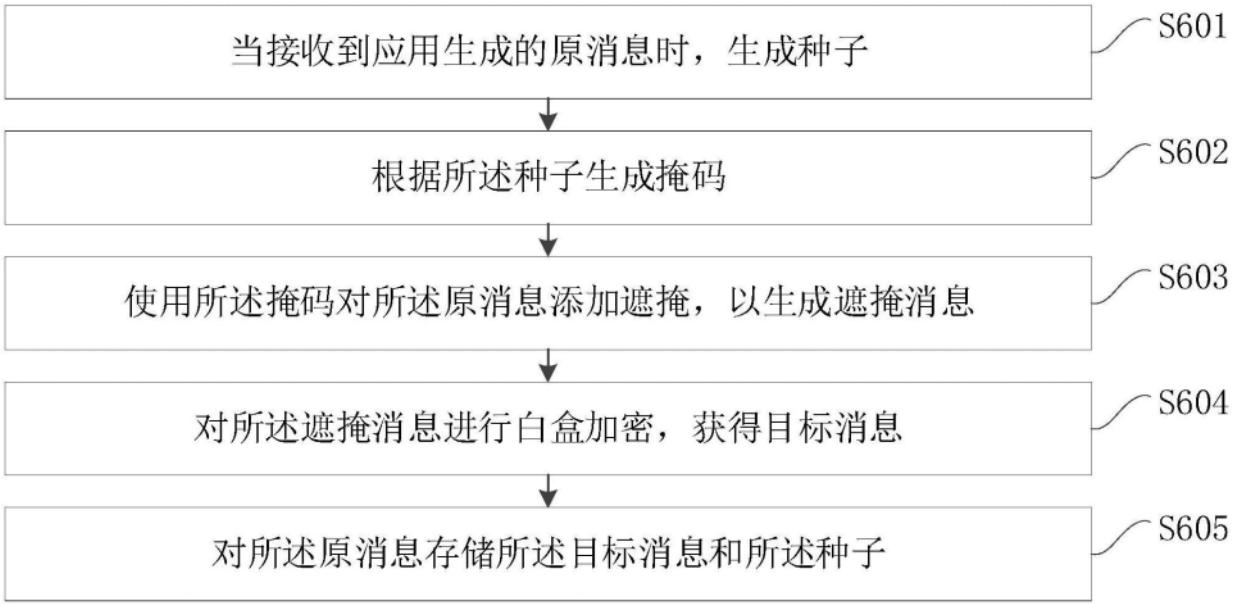

当接收到应用生成的原消息时，生成种子
S601
根据所述种子生成掩码
S602
使用所述掩码对所述原消息添加遮掩，以生成遮掩消息
S603
对所述遮掩消息进行白盒加密，获得目标消息
S604
对所述原消息存储所述目标消息和所述种子
S605

图6

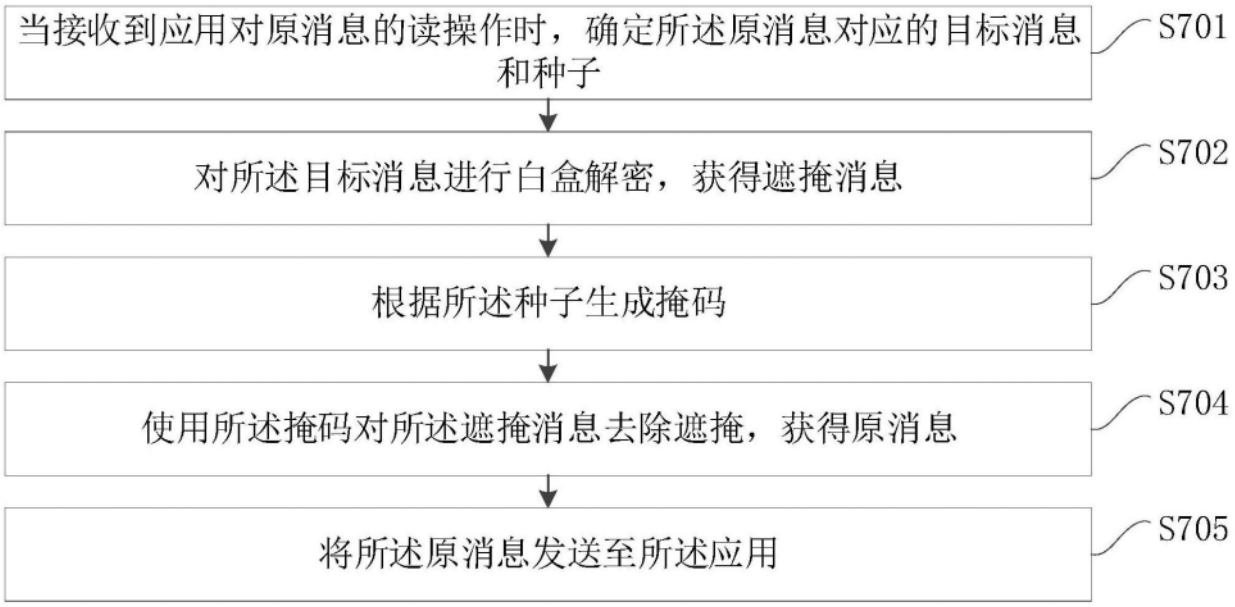

当接收到应用对原消息的读操作时，确定所述原消息对应的目标消息和种子
S701
对所述目标消息进行白盒解密，获得遮掩消息
S702
根据所述种子生成掩码
S703
使用所述掩码对所述遮掩消息去除遮掩，获得原消息
S704
将所述原消息发送至所述应用
S705

图7

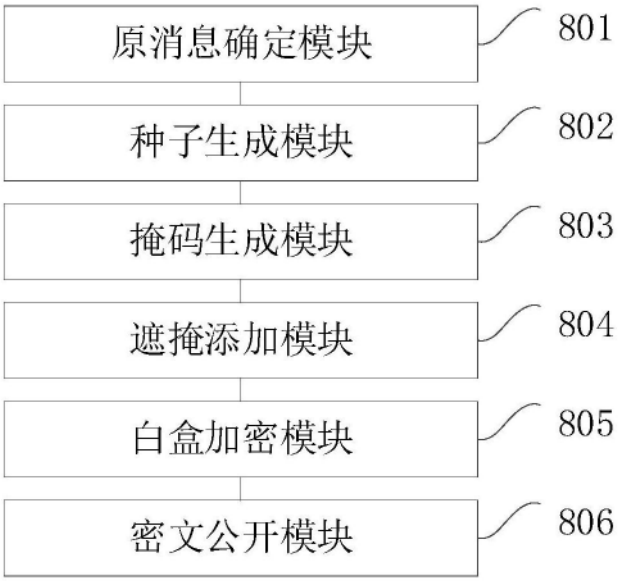

原消息确定模块
801
种子生成模块
802
掩码生成模块
803
遮掩添加模块
804
白盒加密模块
805
密文公开模块
806

图8


密文确定模块
901
白盒解密模块
902
掩码生成模块
903
遮掩去除模块
904

图9


种子生成模块
1001
掩码生成模块
1002
遮掩添加模块
1003
白盒加密模块
1004
密文存储模块
1005

图10


种子生成模块
1101
掩码生成模块
1102
遮掩添加模块
1103
白盒加密模块
1104
密文存储模块
1105

图11

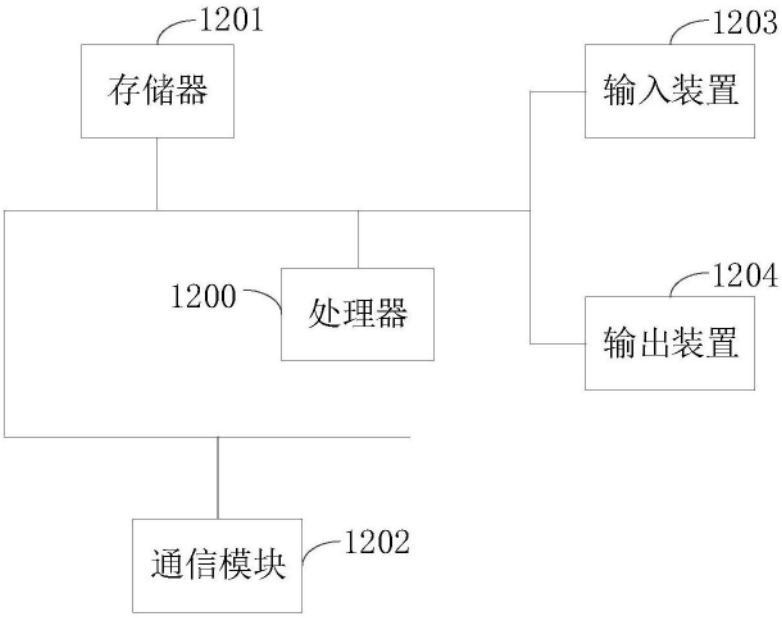


图12