(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 113591041 B
(45)授权公告日 2021.12.31

(21)申请号 202111141640.0

(22)申请日 2021.09.28

(65)同一申请的已公布的文献号
申请公布号 CN 113591041 A

(43)申请公布日 2021.11.02

(73)专利权人 环球数科集团有限公司
地址 518063 广东省深圳市南山区粤海街道高新南九道10号深圳湾科技生态园10栋B座17层01-03号

(72)发明人 张卫平 丁烨 张浩宇

(74)专利代理机构 北京清控智云知识产权代理事务所(特殊普通合伙) 11919
代理人 马肃

(51)Int.Cl.
G06F 21/14(2013.01)
G06F 21/56(2013.01)

(56)对比文件
CN 112948827 A,2021.06.11
CN 111079097 A,2020.04.28
CN 106603198 A,2017.04.26
CN 111552931 A,2020.08.18
US 2019303623 A1,2019.10.03
US 2019332955 A1,2019.10.31
WO 2020251124 A1,2020.12.17
CN 109343937 A,2019.02.15
姜冲.基于深度学习的智能合约漏洞检测技术研究.《中国优秀博硕士学位论文全文数据库(硕士)信息科技辑 》.2021,第I138-582页.

审查员 赵玉华

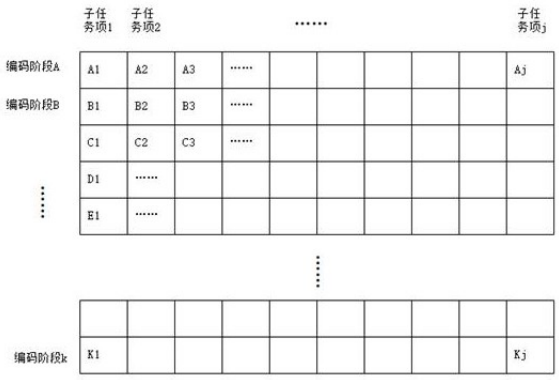
权利要求书2页 说明书8页 附图2页

(54)发明名称

一种防止代码注入或源码反编译的分布式编码系统

(57)摘要

本发明提供了一种防止代码注入或源码反编译的分布式编码系统;所述编码系统基于分布式编码系统制定;所述编码系统包括对需要完成编码任务的总任务进行编码阶段和子任务项的拆分并根据分布式编码系统内的运算节点运算能力,进行子任务项的分派后进行分布式编码;其后,通过分布式编码系统内的多个验证节点,对已经完成编码的代码进行多次的代码注入测试,寻找和分析其中可能存在的代码注入漏洞;进一步,通过在代码中写入密文段信息,以及由多个所述计算节点的共识下增加伪入口信息,使反编译的难度大幅提高,从而阻止外人对源码进行反编译操作。



CN 113591041 B

1.一种防止代码注入或源码反编译的分布式编码系统,其特征在于,所述编码系统包含调度模块、检查模块和处理模块;所述调度模块运行于分布式编码系统的至少一个节点上;所述调度模块对需要进行的编码任务生成任务编号和预留相应的分布式编码云存储空间,并根据拆分规则将编码任务拆分为k个编码阶段,并将每个所述编码阶段进一步拆分为j个子任务项;根据分布式编码系统的当前部署情况,所述调度模块分配所述子任务项到分布式编码系统内的指定节点进分编码处理;所述检查模块对完成编码的子任务项进行监察和校验,并反馈校验结果;所述处理模块位于分布式编码系统的所有运算节点上,用于对所述子任务项进行编码运算,加密编码结果,进行数字签名以及打包上传运算结果的操作;

其中,分布式编码系统内部建立一个联盟链组织并且在联盟链上维护一条代码主链;在接到编码任务时,由分布式编码系统上所有的n个节点通过共识机制推选第一节点;所述第一节点作为响应编码请求和分派编码任务操作的第一响应节点;通过调用所述调度模块,所述第一节点要求分布式编码系统中的其他节点担当编码节点或者验证节点的角色,并形成相应的分派节点记录以提供合法性和可回溯性的验证可能性;所述编码节点在完成编码任务后,请求验证节点中的其中至少一个对完成编码后的代码进行验证测试以找出其中可能存在的代码漏洞。

2.根据权利要求1所述一种防止代码注入或源码反编译的分布式编码系统,其特征在于,所述联盟链内的每一个节点都拥有一对属于所述节点的公钥Pkey和私钥Skey;所述公钥Pkey和私钥Skey通过非对称加密方式生成;所述节点的所述公钥Pkey广播到所述联盟链上,并由联盟链上所有节点记录;所述私钥Skey由节点自行保存和保密,并在进行所述子任务项的编码操作时,用于加密操作;通过所述公钥Pkey加密的信息只能由所述私钥Skey进行解密;通过所述私钥Skey加密的信息只能由所述公钥Pkey进行解密。

3.根据权利要求2所述一种防止代码注入或源码反编译的分布式编码系统,其特征在于,所述第一节点在被推选出来后,联盟链使用所述第一节点的第一公钥加密所述调度模块的登陆通证;所述第一节点使用第一私钥解密所述登陆通证,并获得所述调度模块的调度权限,执行任务调度操作。

4.根据权利要求3所述一种防止代码注入或源码反编译的分布式编码系统,其特征在于,所述第一节点通过调用所述调度模块,发送其中一个所述子任务项到除所述第一节点外的(n-1)个节点进行预编码,通过测算所述(n-1)个节点运算能力以及负载比,计算每个节点的能力值;所述调度模块统计各节点的能力值,选择(n-1)个节点中的j个节点作为计算节点;j个所述计算节点组成计算节点组;所述第一节点将一个所述编码阶段中的j个所述子任务项分派到所述计算节点组,并编写形式为<任务编号-编码阶段-子任务项-计算节点编号>的计算节点分派记录;所述第一节点根据所述计算节点分派记录执行分派操作,将一个编码阶段中的j个所述子任务项分发到j个所述计算节点进行编码运算。

5.根据权利要求4所述一种防止代码注入或源码反编译的分布式编码系统,其特征在于,所述计算节点分派记录由所述第一节点写入所述代码主链的区块中;所述计算节点组通过联盟链内的共识机制由全体节点确认其合法性;所述联盟链向所述计算节点组内的j个所述计算节点发送由各自的公钥Pkey加密的启用所述处理模块权限的通证;每个所述计算节点通过各自的所述私钥Skey解密所述处理模块权限的通证,从而获得调用所述处理模块的权限,并利用所述处理模块进行编码处理。

6.根据权利要求5所述一种防止代码注入或源码反编译的分布式编码系统，其特征在于，在所述计算节点组被指定后，所述调度模块指定其余的(n-j-1)个节点为验证节点，并由所述调度模块生成验证节点分派记录<任务编号-编码阶段-子任务项-验证节点编号>。

7.根据权利要求6所述一种防止代码注入或源码反编译的分布式编码系统，其特征在于，所述验证节点分派记录由所述第一节点写入所述代码主链的区块中；所述验证节点组通过联盟链内的共识机制由全体节点确认其合法性；所述联盟链向所述验证节点组内的(n-j-1)个验证节点发送由各自的公钥Pkey加密的启用检查模块权限的通证；每个所述验证节点通过各自的所述私钥Skey解密所述检查模块权限的通证，从而获得调用所述检查模块的权限，并利用所述检查模块进行编码监察。

8.根据权利要求7所述一种防止代码注入或源码反编译的分布式编码系统，其特征在于，当一个所述计算节点完成一个所述子任务项并获得对应的代码段后，请求至少一个所述验证节点进行代码注入测试；至少一个所述验证节点通过利用已知的或者可能存在的代码注入方式，对所述代码段进行至少p次的尝试注入操作；p的数值由所述第一节点通过评估所述代码段的长度以及至少一个所述验证节点的能力值决定；至少一个所述验证节点同时调用所述检查模块的过滤器以及判断器记录在尝试代码注入操作后，所述代码段是否存在漏洞；所述验证节点在完成p次的注入操作后，获得检查记录，并将所述检查记录反馈到所述第一节点，并由第一节点调用所述调度模块，决定所述计算节点是否需要对所述子任务项进行重新编码以保证对代码注入的足够防御能力。

9.根据权利要求8所述一种防止代码注入或源码反编译的分布式编码系统，其特征在于，所述第一节点在获得一个所述编码阶段的所有代码段后，遍历所述编码阶段，并在其中随机位置加入由所述第一节点的所述私钥Skey加密后的一段哈希值密文段，并将所述密文段采用注释符号注释；所述第一节点将一个已添加所述密文段的所述编码阶段的完整代码进行字节数组化并将数组化后的完整代码通过所述第一节点的所述私钥Skey进行哈希运算加密，获得一段代表该段所述编码阶段的固定长度的哈希值字段。

10.根据权利要求9所述一种防止代码注入或源码反编译的分布式编码系统，其特征在于，所述第一节点通过将k个所述编码阶段加密后，获得k个所述哈希值字段；所述第一节点将k个所述编码阶段的代码整合后，再在代码开头加上强制验证程序，要求每次读取该段代码时，强制验证所述k个编码阶段的k个哈希值字段；所述第一节点将所述强制验证程序、完整代码以及所述k个哈希值字段打包后，上传到所述代码主链。

一种防止代码注入或源码反编译的分布式编码系统

技术领域

[0001] 本发明涉及分布式编码技术领域。具体而言，涉及一种防止代码注入或源码反编译的分布式编码系统。

背景技术

[0002] 随着人工智能的发展，针对越来越庞大的应用场景以及高速增长的万物互联要求，对计算系统的算力消耗不断高速增加，所要求的算力、算法甚至是网络通讯速度都对当前相应的技术领域提出前所未有的要求高度。尤其对于全自动人工智能应用场景，例如智慧景区、智慧医院等领域，涉及的节点、信息，需要处理和编码的数据量巨大，静态、动态的数据变化快，同时需要采用高度灵活的算法针对不同事件作出最优化响应。海量的编码要求，催生了分布式编码技术；利用分布式编码技术，对原有的一个编码任务进行任务切分，并通过分配机制发放到各运算节点上进行编码，最后汇总完成阶段性任务。分布式编码区别于传统中心式编码系统，大大利用了目前服务器节点分散布局，但同时无线网络速度和带宽增长速度快的系统特点，有效地提高了编码的效率。

[0003] 进一步的，在当前复杂的网络空间环境下，对用户输入数据缺乏有效性验证及过滤的各种程序而言，黑客可通过构造各种畸形数据使得软件或者程序实体在具体执行过程中违背设计者初衷，从而改变程序控制流继而窃取程序控制权或者窃取程序背后的数据库等保密数据，是网络和系统当前所面临最为严重的安全威胁。针对注入型脆弱性与接收外部输入行为密切相关这一特征。

[0004] 进一步，代码在执行过程中，出于客户端的各种原因，可能会对源代码进行截取并进行反编译，从而获得源代码或者源代码所包含的逻辑、算法、执行流程等不应该被外界所知的知识部分，继而对源代码开发者的权益带来损失。

[0005] 查阅相关地已公开技术方案，公开号为US2021089645A1提出一种生成随机安全标签并写入指令块内，并在指令块执行时根据标签的指向两个异或域的随机互相验证的方式，阻止指令块出现异常时不会被发送到处理器执行；CN112231651(A)提出一种对源码核心进行MD5值进行非对称加密的方式，加强源码核心的解密难度，从而提高解密和反编译的时间和算力成本。然而以上技术方案都是针对过往中心化编码的运算方式，并未适合目前讨论的分布式编码方式。

发明内容

[0006] 本发明的目的在于，提供一种防止代码注入或源码反编译的分布式编码系统；所述编码系统充分利用了分布式系统具有的多节点、多并发特点，将程序的编码、执行和合法性验证的进程，通过分拆和指派后，由各节点分别进行编码和合法性验证，并在此过程中记录足够多的特征记录作为编码算法优化依据，有效提高了编码和执行时的效率和安全性；并且本编码系统利用区块链对完整代码进行加密性后的加密验证信息保存，有效防止加密信息在某一节点泄漏后对代码的一致性质疑 。

[0007] 本发明采用如下技术方案：

[0008] 一种防止代码注入或源码反编译的分布式编码系统，所述编码系统包含调度模块、检查模块和处理模块；所述调度模块运行于分布式系统上至少一个节点上；所述调度模块对需要进行的编码任务生成任务编号和预留相应的分布式云存储空间，并根据拆分规则将编码任务拆分为k个编码阶段，并将每个所述编码阶段进一步拆分为j个子任务项；根据分布式当前部署情况，所述调度模块分配所述子任务项到分布式系统内的指定节点进分编码处理；所述检查模块对完成编码的子任务项进行监察和校验，并反馈校验结果；所述处理模块位于分布式所有运算节点上，用于对所述子任务项进行编码运算，加密编码结果，进行数字签名以及打包上传运算结果的操作；

[0009] 其中，分布式系统内部建立一个联盟链组织并且在联盟链上维护一条代码主链；在接到编码任务时，由分布式系统上所有的n个节点通过共识机制推选第一节点；所述第一节点作为响应编码请求和分派编码任务操作的第一响应节点；通过调用所述调度模块，所述第一节点要求分布式系统中的其他节点担当编码节点或者验证节点的角色，并形成相应的分派节点记录以提供合法性和可回溯性的验证可能性；所述编码节点在完成编码任务后，请求验证节点中的其中至少一个对完成编码后的代码进行验证测试以找出其中可能存在的代码漏洞；

[0010] 所述联盟链内的每一个节点都拥有一对属于所述节点的公钥Pkey和私钥Skey；所述公钥Pkey和私钥Skey通过非对称加密方式生成；所述节点的所述公钥Pkey广播到所述联盟链上，并由链上所有节点记录；所述私钥Skey由节点自行保存和保密，并在进行所述子任务的编码操作时，用于加密操作；通过所述公钥Pkey加密的信息只能由所述私钥Skey进行解密；通过所述私钥Skey加密的信息只能由所述公钥Pkey进行解密；

[0011] 所述第一节点在被推选出来后，联盟链使用所述第一节点的所述第一公钥加密所述调度模块的登陆通证；所述第一节点通过第一私钥解密所述登陆通证，并获得所述调度模块的调度权限，执行任务调度操作；

[0012] 所述第一节点通过调用所述调度模块，发送其中一个所述子任务项到除所述第一节点外的(n-1)个节点进行预编码，通过测算所述(n-1)个节点运算能力以及负载比，计算每个节点的能力值；所述调度模块统计各节点的能力值，选择(n-1)个节点中的j个节点作为计算节点；j个所述计算节点组成计算节点组；所述第一节点将一个所述编码阶段中的j个所述子任务项分派到所述计算节点组，并编写形式为<任务编号-编码阶段-子任务项-计算节点编号>的计算节点分派记录；所述第一节点根据所述计算节点分派记录执行分派操作，将一个编码阶段中的j个所述子任务项分发到j个所述计算节点进行编码运算；

[0013] 所述计算节点分派记录由所述第一节点写入所述代码主链的区块中；所述计算节点组通过联盟链内的共识机制由全体节点确认其合法性；所述联盟链向所述计算节点组内的j个所述计算节点发送由各自的公钥Pkey加密的启用所述处理模块权限的通证；每个所述计算节点通过各自的所述私钥Skey解密所述处理模块权限的通证，从而获得调用所述处理模块的权限，并利用所述处理模块进行编码处理；

[0014] 在所述计算节点组被指定后，所述调度模块指定其余的(n-j-1)个节点为验证节点，并由所述调度模块生成验证节点分派记录<任务编号-编码阶段-子任务项-验证节点编号>；

[0015] 所述验证节点分派记录由所述第一节点写入所述代码主链的区块中;所述验证节点组通过联盟链内的共识机制由全体节点确认其合法性;所述联盟链向所述验证节点组内的(n-j-1)个验证节点发送由各自的公钥Pkey加密的启用所述检查模块权限的通证;每个所述验证节点通过各自的所述私钥Skey解密所述检查模块权限的通证,从而获得调用所述检查模块的权限,并利用所述检查模块进行编码监察;

[0016] 当一个所述计算节点完成所述一个子任务项并获得对应的代码段后,请求至少一个所述验证节点进行代码注入测试;至少一个所述验证节点通过利用已知的或者可能存在的代码注入方式,对所述代码段进行至少p次的尝试注入操作;p的数值由所述第一节点通过评估所述代码段的长度以及至少一个所述验证节点的能力值决定;至少一个所述验证节点同时调用所述检查模块的过滤器以及判断器记录在尝试代码注入操作后,所述代码段是否存在漏洞;所述验证节点在完成p次的注入操作后,获得检查记录,并将所述检查记录反馈到所述第一节点,并由第一节点调用所述调度模块,决定所述计算节点是否需要对所述子任务项进行重新编码以保证对代码注入的足够防御能力;

[0017] 所述第一节点在获得一个所述编码阶段的所有代码段后,遍历所述编码阶段,并在其中随机位置加入由所述第一节点的所述第一私钥加密后的一段哈希值密文段,并将所述密文段采用注释符号注释;所述第一节点将一个已添加所述密文段的所述编码阶段的完整代码进行字节数组化并将数组化后的完整代码通过所述第一节点的所述第一私钥进行哈希运算加密,获得一段代表该段所述编码阶段的固定长度的哈希值字段;

[0018] 所述第一节点通过将k个所述编码阶段加密后,获得k个所述哈希值字段;所述第一节点将k个所述编码阶段的代码整合后,再在代码开头加上强制验证程序,要求每次读取该段代码时,强制验证所述k个编码阶段的k个哈希值字段;所述第一节点将所述强制验证程序、完整代码以及所述k个哈希值字段打包后,上传到所述代码主链。

[0019] 本发明所取得的有益效果是:

[0020] 1.本编码系统有别于以往中心化编码和防御的运行形式,充分调用分布式系统内多节点并发运算的特点,对完整的代码在可控环境下进行非法代码注入测试,以保证代码尽可能地排除注入漏洞;

[0021] 2.本编码系统利用机器编译代码的过程中会跳过代码注释的特点,在代码中加入加密后的密文并对密文进行注释,最后对代码段进行哈希值加密;则在遇到反编码时,由于无法完整反编译出注释部分,因此反编译的代码必然无法通过强制验证程序;

[0022] 3.本编码系统运用区块链技术保存编码后的各段代码,既为代码的一致性进行全节点的共识背书,亦通过将各段代码分段保存和加密,阻止反编码的算法操作。

[0023] 4.本编码系统适用于基于各类编程系统、语言或算法,具有良好的通用性效果。

附图说明

[0024] 从以下结合附图的描述可以进一步理解本发明。图中的部件不一定按比例绘制,而是将重点放在示出实施例的原理上。在不同的视图中,相同的附图标记指定对应的部分。

[0025] 图1为本发明将编码任务分解的示意图;

[0026] 图2 为本发明所述分布式编码系统的各部分组成示意图;

[0027] 图3 为所述检查模块提示找到怀疑非法代码注入操作的示意图;

[0028] 图4为所述编码系统内所述代码主链的区块示意图。

[0029] 附图标号说明:100-分布式编码系统;101-代码主链;102-第一节点;103-调度模块;104-处理模块;105-检查模块;106-计算节点组;107-验证节点组。

具体实施方式

[0030] 为了使得本发明的目的技术方案及优点更加清楚明白,以下结合其实施例,对本发明进行进一步详细说明;应当理解,此处所描述的具体实施例仅用于解释本发明 ,并不用于限定本发明。对于本领域技术人员而言,在查阅以下详细描述之后,本实施例的其它系统.方法和/或特征将变得显而易见。旨在所有此类附加的系统、方法、特征和优点都包括在本说明书内.包括在本发明的范围内,并且受所附权利要求书的保护。在以下详细描述描述了所公开的实施例的另外的特征,并且这些特征根据以下将详细描述将是显而易见的。

[0031] 本发明实施例的附图中相同或相似的标号对应相同或相似的部件;在本发明的描述中,需要理解的是,若有术语“上”、“下”、“左”、“右”等指示的方位或位置关系为基于附图所示的方位或位置关系,仅是为了便于描述本发明和简化描述,而不是指示或暗示所指的装置或组件必须具有特定的方位.以特定的方位构造和操作,因此附图中描述位置关系的用语仅用于示例性说明,不能理解为对本专利的限制,对于本领域的普通技术人员而言,可以根据具体情况理解上述术语的具体含义。

[0032] 实施例一:

[0033] 如附图1和附图2,一种防止代码注入或源码反编译的分布式编码系统,所述编码系统包含调度模块、检查模块和处理模块;所述调度模块运行于分布式系统上至少一个节点上;所述调度模块对需要进行的编码任务生成任务编号和预留相应的分布式云存储空间,并根据拆分规则将编码任务拆分为k个编码阶段,并将每个所述编码阶段进一步拆分为j个子任务项;根据分布式当前部署情况,所述调度模块分配所述子任务项到分布式系统内的指定节点进分编码处理;所述检查模块对完成编码的子任务项进行监察和校验,并反馈校验结果;所述处理模块位于分布式所有运算节点上,用于对所述子任务项进行编码运算,加密编码结果,进行数字签名以及打包上传运算结果的操作;

[0034] 其中,分布式系统内部建立一个联盟链组织并且在联盟链上维护一条代码主链;在接到编码任务时,由分布式系统上所有的n个节点通过共识机制推选第一节点;所述第一节点作为响应编码请求和分派编码任务操作的第一响应节点;通过调用所述调度模块,所述第一节点要求分布式系统中的其他节点担当编码节点或者验证节点的角色,并形成相应的分派节点记录以提供合法性和可回溯性的验证可能性;所述编码节点在完成编码任务后,请求验证节点中的其中至少一个对完成编码后的代码进行验证测试以找出其中可能存在的代码漏洞;

[0035] 所述联盟链内的每一个节点都拥有一对属于所述节点的公钥Pkey和私钥Skey;所述公钥Pkey和私钥Skey通过非对称加密方式生成;所述节点的所述公钥Pkey广播到所述联盟链上,并由链上所有节点记录;所述私钥Skey由节点自行保存和保密,并在进行所述子任务的编码操作时,用于加密操作;通过所述公钥Pkey加密的信息只能由所述私钥Skey进行解密;通过所述私钥Skey加密的信息只能由所述公钥Pkey进行解密;

[0036] 所述第一节点在被推选出来后,联盟链使用所述第一节点的所述第一公钥加密所

述调度模块的登陆通证;所述第一节点通过第一私钥解密所述登陆通证,并获得所述调度模块的调度权限,执行任务调度操作;

[0037] 所述第一节点通过调用所述调度模块,发送其中一个所述子任务项到除所述第一节点外的(n-1)个节点进行预编码,通过测算所述(n-1)个节点运算能力以及负载比,计算每个节点的能力值;所述调度模块统计各节点的能力值,选择(n-1)个节点中的j个节点作为计算节点;j个所述计算节点组成计算节点组;所述第一节点将一个所述编码阶段中的j个所述子任务项分派到所述计算节点组,并编写形式为<任务编号-编码阶段-子任务项-计算节点编号>的计算节点分派记录;所述第一节点根据所述计算节点分派记录执行分派操作,将一个编码阶段中的j个所述子任务项分发到j个所述计算节点进行编码运算;

[0038] 所述计算节点分派记录由所述第一节点写入所述代码主链的区块中;所述计算节点组通过联盟链内的共识机制由全体节点确认其合法性;所述联盟链向所述计算节点组内的j个所述计算节点发送由各自的公钥Pkey加密的启用所述处理模块权限的通证;每个所述计算节点通过各自的所述私钥解密所述处理模块权限的通证,从而获得调用所述处理模块的权限,并利用所述处理模块进行编码处理;

[0039] 在所述计算节点组被指定后,所述调度模块指定其余的(n-j-1)个节点为验证节点,并由所述调度模块生成验证节点分派记录<任务编号-编码阶段-子任务项-验证节点编号>;

[0040] 所述验证节点分派记录由所述第一节点写入所述代码主链的区块中;所述验证节点组通过联盟链内的共识机制由全体节点确认其合法性;所述联盟链向所述验证节点组内的(n-j-1)个验证节点发送由各自的公钥Pkey加密的启用所述检查模块权限的通证;每个所述验证节点通过各自的所述私钥解密所述检查模块权限的通证,从而获得调用所述检查模块的权限,并利用所述检查模块进行编码监察;

[0041] 当一个所述计算节点完成所述一个子任务项并获得对应的代码段后,请求至少一个所述验证节点进行代码注入测试;至少一个所述验证节点通过利用已知的或者可能存在的代码注入方式,对所述代码段进行至少p次的尝试注入操作;p的数值由所述第一节点通过评估所述代码段的长度以及至少一个所述验证节点的能力值决定;至少一个所述验证节点同时调用所述检查模块的过滤器以及判断器记录在尝试代码注入操作后,所述代码段是否存在漏洞;所述验证节点在完成p次的注入操作后,获得检查记录,并将所述检查记录反馈到所述第一节点,并由第一节点调用所述调度模块,决定所述计算节点是否需要重新对所述子任务项进行重新编码以保证对代码注入的足够防御能力;

[0042] 所述第一节点在获得一个所述编码阶段的所有代码段后,遍历所述编码阶段,并在其中随机位置加入由所述第一节点的所述第一私钥加密后的一段哈希值的密文段,并将所述密文段采用注释符号注释;所述第一节点将一个已添加所述密文段的所述编码阶段的完整代码进行字节数组化并将数组化后的完整代码通过所述第一节点的所述第一私钥进行哈希运算加密,获得一段代表该段所述编码阶段的固定长度的哈希值字段;

[0043] 所述第一节点通过将k个所述编码阶段加密后,获得k个所述哈希值字段;所述第一节点将k个所述编码阶段的代码整合后,再在代码开头加上强制验证程序,要求每次读取该段代码时,强制验证所述k个编码阶段的k个哈希值字段;所述第一节点将所述强制验证程序、完整代码以及所述k个哈希值字段打包后,上传到所述代码主链;

[0044] 基于以下实施方式，所述分布式系统内的各个所述验证节点，可根据以往的运行和编码过程中获得的经验，对已完成编码的代码段进行反复多次的代码注入验证，这些验证包括对非法符号的读取漏洞、注入逻辑的有效辨别、长代码辨别、非法身份验证等各种可实现代码注入的手段；例如，如附图3，客户端在登陆过程中，使用了可疑的输入“admin’ or ‘1’=’1；所述检查模块根据可疑规则，判断其输入字符段与常用的电邮地址差异很大；其次，该输入中出现了多个“”符号，在一般的登陆中并不常见；其三，该语句一旦运行，将使数据库查询所有管理员账户下的资料，例如密码等，可以由所述检查模块识别为常见的非法操作；

[0045] 进一步的，所述验证节点组收集各个所述验证节点尝试后的代码注入手段作业验证库，由所述检查模块尝试分析和拓展各种代码注入手段的更多的出现可能，以提供更具体的辨别特征给予所述验证节点作为验证行为的修正和提高验证效能；

[0046] 进一步的，在代码中注释后的语句，都不会被机器执行读取，更不会被机器所识别后进行机器汇编，因此对代码的反编码过程中，亦不能识别其中被注释的语句；因此，原代码加密后的所述密文段，与反编码获得的代码再进行哈希加密后的所述密文段必然不相同；在此过程中，即使所述第一节点的所述第一私钥被意外泄漏亦不影响反编码后的代码无法通过强制验证的后果；

[0047] 进一步的，由于所有生成后的代码都分段地进行了加密，即使其中一段代码被完全反编码，由于完成所有代码段的反编码过程需要耗费极大的运算成本，其破解成本和收益相差太远，因此亦有效阻止了非法反编码的行为。

[0048] 实施例二：

[0049] 本实施例应当理解为至少包含前述任一一个实施例的全部特征，并在其基础上进一步改进；

[0050] 在正常执行代码的过程中，对系统的内存占用、处理器算力占用、网络带宽流量占用等其他计算机运算特征都具有一定的稳定性；尤其是调入内存中的各种函数库、语句句柄、输出字段等都具有相对固定的长度、时序；代码被加载时操作系统分配虚拟内存，把可执行文件中的代码段、数据段等映射到内存中，加载动态库，然后从入口点开始执行；所有的代码注入都会对内存中的数据进行读/写，从而实现注入代码的额外执行；

[0051] 然而一旦代码被修改，其中的执行逻辑则容易被扰动从而发生变化；因此，本实施例通过所述检查模块对执行代码验证过程中所述验证节点的运算过程中的特征量，从而针对各种可疑的代码执行过程进行标记和警示；

[0052] 在多个所述验证节点对代码进行注入验证测试时，进行代码运行标志事件的记录，包括：

[0053] 1．对动态库的调用情况，是否调用了源码中不涉及的动态库文件，进行额外的运算程序；

[0054] 2．运行同样代码段的总执行时间异常；

[0055] 3．尝试删除标志事件的记录或记录中的某一行；

[0056] 4．网络流量的异常高峰使用；

[0057] 5．对数据库的异常申请调用；

[0058] 以上仅举出其中几项常见的执行异常的特征，虽然以上特征并不能直接认为代码

存在被异常注入的可能，但所述检查模块仍然可能关注这些异常特征的出现，是否可以发展为一个具体的代码注入漏洞，并向所述第一节点提出修改建议，由所述第一节点根据编码任务的实际要求，决定是否要求所述计算节点对出现异常的所述子任务项甚至对所述编码阶段作出重新编码的要求。

[0059] 实施例三：

[0060] 本实施例应当理解为至少包含前述任一一个实施例的全部特征，并在其基础上进一步改进：

[0061] 在目前反编译的技术中，大部分需要先确定程序代码的入口特征，从而找源码的执行逻辑起点，才能在反编译后，确定源码的逻辑顺序

[0062] 常见的程序：

[0063] Delphy程序中，1. 入口程序是正常的压栈；2. 调用一个E8字节类型的call；E8字节Call中的内容调用了一个GetModuleHandleA()函数；最后进入GetModuleHandleA()函数获取的动态库调用情况，找到跳转的字节位置，刚可以找到代码入口；

[0064] Borland C++程序中，入口包括一个jmp语句，jmp语句中间是语段字符fb:C++ HOOK，之后的第一个调用是GetModuleHandleA()函数；

[0065] Visual C++程序中，Call指令指向的是地址为FF15的字节位，而且VC入口点特征第一个调用函数是GetVersion()；

[0066] 可以看出，常见程序在广泛的编码使用中，都具有较为明显的入口物征，为反编译的执行降低了难度；

[0067] 因此在本实施例中，通过多个所述计算节点组的共识后，对同一个所述编码阶段，设置t个伪入口点，并由j个所述计算节点中的t个所述计算节点负责完成所述t个伪入口点的编码；所述伪入口的代码要求具有高度的相似性；

[0068] 进一步的，负责建立伪入口点的t个所述计算节点作出共识，在运行所述编码阶段的代码时将正确入口信息记入下一个所述编码阶段中；正确的入口信息，由于通过了t个所述计算节点的分析，具有足够的隐蔽性和可执行性，可以避免在正常执行代码时，耗费了额外的计算机资源用于反复的验证和确认；

[0069] 在遇到非法反编码时，如需要获取一个所述编码阶段的正确入口，则需要对下一个所述编码阶段进行完全的反编码，并且由此形成一个递进的反编译循环，大大提升了反编译的难度；同样其反编译的成本将明显大幅提高。

[0070] 在上述实施例中，对各个实施例的描述都各有侧重，某个实施例中没有详述或记载的部分，可以参见其它实施例的相关描述。

[0071] 虽然上面已经参考各种实施例描述了本发明，但是应当理解，在不脱离本发明的范围的情况下，可以进行许多改变和修改。也就是说上面讨论的方法，系统和设备是示例。各种配置可以适当地省略，替换或添加各种过程或组件。例如，在替代配置中，可以以与所描述的顺序不同的顺序执行方法，和/或可以添加，省略和/或组合各种部件。而且，关于某些配置描述的特征可以以各种其他配置组合，如可以以类似的方式组合配置的不同方面和元素。此外，随着技术发展其中的元素可以更新，即许多元素是示例，并不限制本公开或权利要求的范围。

[0072] 在说明书中给出了具体细节以提供对包括实现的示例性配置的透彻理解。然而，

可以在没有这些具体细节的情况下实践配置例如，已经示出了众所周知的电路，过程，算法，结构和技术而没有不必要的细节，以避免模糊配置。该描述仅提供示例配置，并且不限制权利要求的范围，适用性或配置。相反，前面对配置的描述将为本领域技术人员提供用于实现所描述的技术的使能描述。在不脱离本公开的精神或范围的情况下，可以对元件的功能和布置进行各种改变。

[0073] 综上，其旨在上述详细描述被认为是例示性的而非限制性的，并且应当理解，以上这些实施例应理解为仅用于说明本发明而不用于限制本发明的保护范围。在阅读了本发明的记载的内容之后，技术人员可以对本发明作各种改动或修改，这些等效变化和修饰同样落入本发明权利要求所限定的范围。

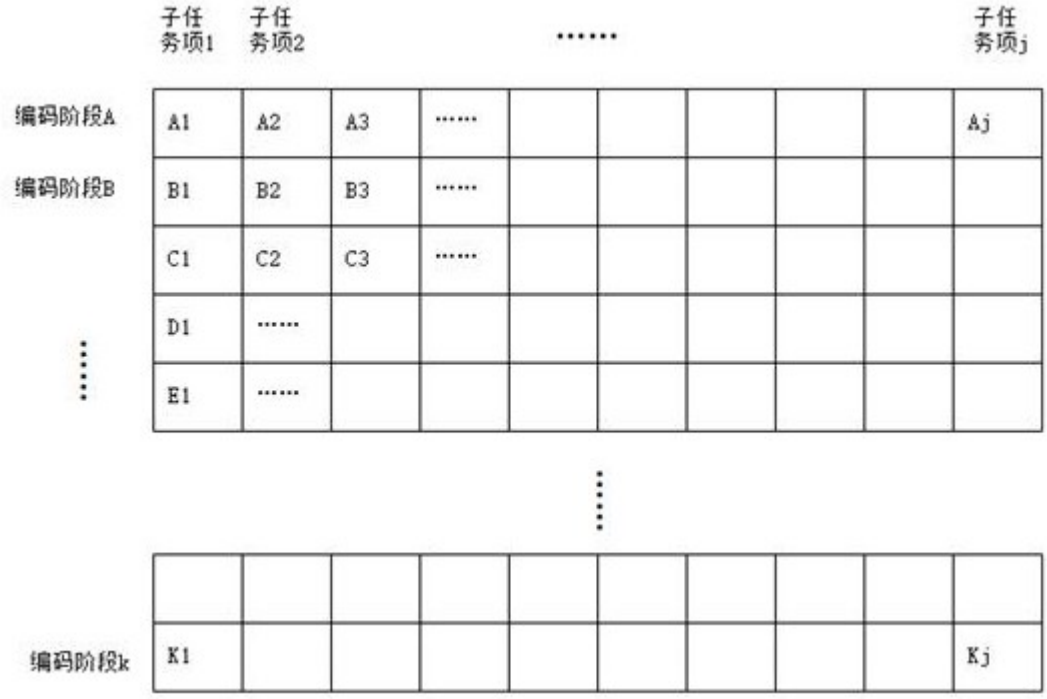

子任务项1
子任务项2
……
子任务项j
编码阶段A
编码阶段B
编码阶段k
A1
A2
A3
Aj
B1
B2
B3
C1
C2
C3
D1
E1
K1
Kj

图1

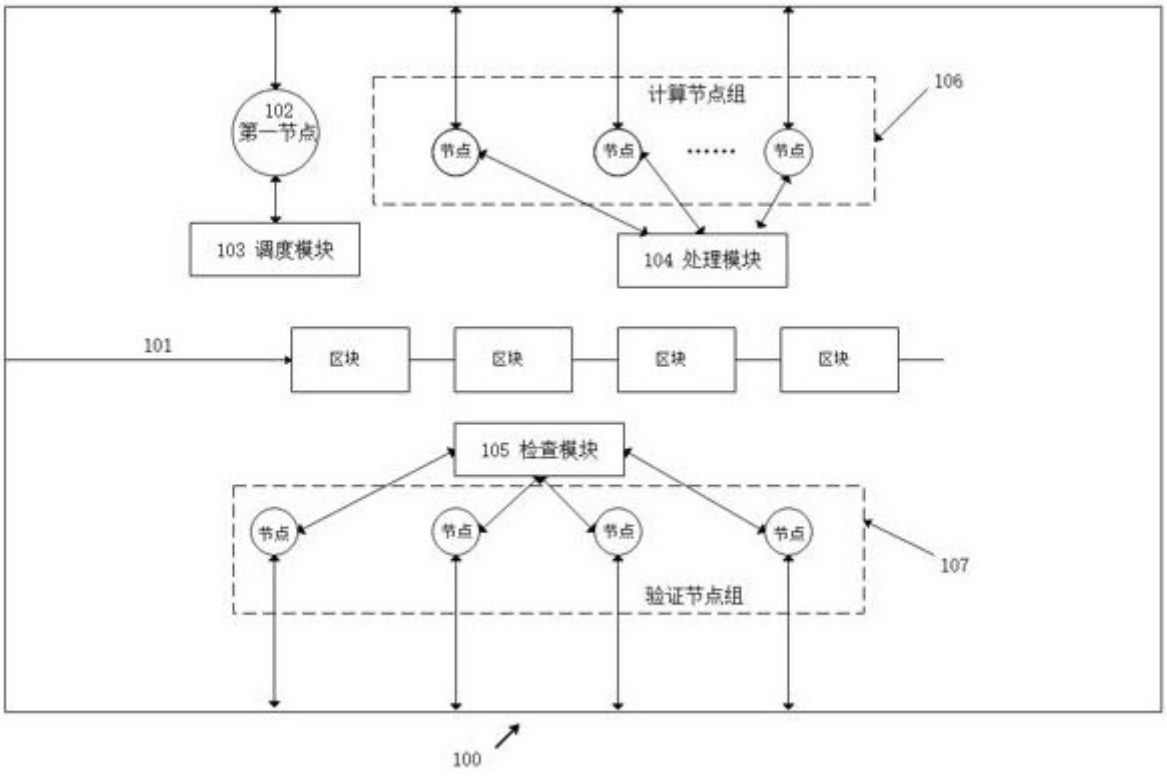

102
第一节点
计算节点组
106
节点
节点
节点
103 调度模块
104 处理模块
101
区块
区块
区块
区块
105 检查模块
节点
节点
节点
节点
107
验证节点组
100

图2


Email*
admin' or '1'='1
Invalid Email Format.
Password*
••••••••••••••••

图3

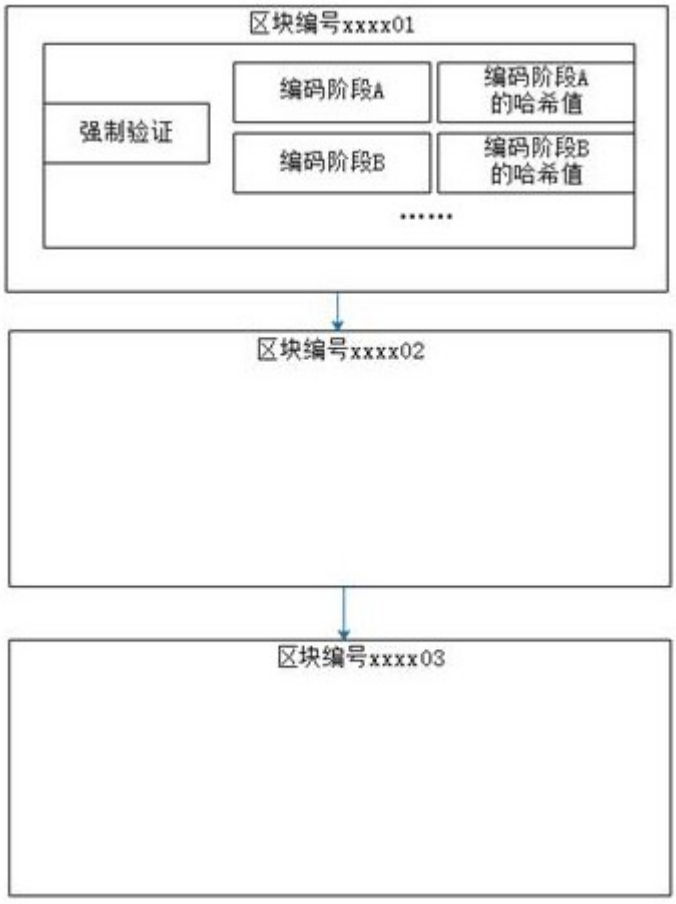

区块编号xxxx01
强制验证
编码阶段A
编码阶段A的哈希值
编码阶段B
编码阶段B的哈希值
.....
区块编号xxxx02
区块编号xxxx03

图4