[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
H04L 1/00 (2006.01)
G06Q 99/00 (2006.01)

[12] 发明专利申请公布说明书

[21] 申请号 200680022176.0

[43] 公开日 2008 年 6 月 18 日

[11] 公开号 CN 101204035A

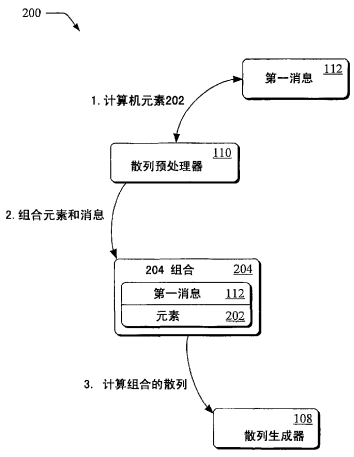
[22] 申请日 2006.6.13
[21] 申请号 200680022176.0
[30] 优先权
[32] 2005. 6.28 [33] US [31] 11/168,842
[86] 国际申请 PCT/US2006/022960 2006.6.13
[87] 国际公布 WO2007/001829 英 2007.1.4
[85] 进入国家阶段日期 2007.12.20
[71] 申请人 微软公司
地址 美国华盛顿州
[72] 发明人 G·A·余瓦尔 R·文卡特萨
[74] 专利代理机构 上海专利商标事务所有限公司
代理人 陈 斌
权利要求书 4 页 说明书 11 页 附图 6 页

[54] 发明名称

加强安全散列函数

[57] 摘要

描述了用于加强安全散列函数系统和/或方法。这些系统和/方法在一些实施例中基于消息并使用一过程创建随机出现的元素。 然后所述元素与该消息被组合。 可使用安全散列函数散列该组合。 使用相同的过程和安全散列函数，所述消息之后可被认证。



1．一个或多个具有计算机可读指令的计算机可读介质，当所述计算机可读指令被一计算机执行的时候，使得所述计算机执行下列的动作：

基于一消息并使用一过程创建随机出现的元素；以及

把所述元素和消息进行组合以提供一组合体，所述组合体能用一安全散列函数进行散列以提供一散列，所述散列具有与一由所述消息用相同的安全散列函数进行散列而得到的第二散列相等的长度，

其中所述过程能基于一第二消息创建一第二随机出现的元素，如果所述第二消息与所述第一消息相同，那么所述第二随机出现的元素与所述第一随机出现的元素相同。

2．如权利要求 1 所述的介质，其中所述创建随机出现元素的动作包括：使用块密码加密消息的一部分以提供一经加密的部分，以及对所述经加密的部分和所述消息的所述一部分计算异或（XOR）以提供所述随机出现的元素。

3．如权利要求 1 所述的介质，其中所述进行组合的动作包括串接所述随机出现的元素和所述消息。

4．如权利要求 1 所述的介质，其中所述过程包括使用块密码加密所述消息的一部分以提供一经加密的部分，以及对所述经加密的部分和所述消息的所述一部分计算一异或（XOR）以提供所述随机出现元素。

5．如权利要求 1 所述的介质，其中所述创建随机出现的元素的动作包括：

基于所述消息接收一中间随机出现的元素；

从所述中间随机出现的元素产生一随机出现的加密密钥；

用使用所述随机出现的加密密钥的加密函数对所述消息的一部分进行加密以提供一经加密的部分；以及

计算所述经加密的部分和所述一部分的异或（XOR）以提供所述随机出现

的元素。

6．如权利要求5所述的介质，其中所述加密函数包括128位高级加密标准块密码加密函数。

7．如权利要求1所述的介质，其中所述安全散列函数包括安全散列算法-1。

8．一个或多个具有计算机可读指令的计算机可读介质，当所述计算机可读指令被一计算机执行时，使得所述计算机执行下列的操作：

加密一数据块的第一子块和第二子块以提供经加密的第一子块和经加密的第二子块；

计算所述第一子块和所述经加密的第一子块的异或（XOR）以提供第一异或子块；

计算所述第二子块和所述经加密的第二子块的异或以提供第二异或子块；以及

组合所述第一异或子块和所述第二异或子块与所述第一子块和所述第二子块，以提供第一组合块和第二组合块。

9．如权利要求8所述的介质，其中所述加密的动作包括在所述第一子块和所述第二子块上执行块密码。

10．如权利要求8所述的介质，其中所述加密的动作包括：

使用从所述第二子块产生的第一加密密钥对所述第一子块加密；以及

使用从所述第一子块产生的第二加密密钥对所述第二子块加密。

11．如权利要求10所述的介质，其中所述加密的动作包括加密一第三子块和一第四子块，进一步包括使用从所述第一、第二或第四子块产生的第三加密密钥对所述第三子块加密，以及使用从所述第一、第二或第三子块产生的第

四加密密钥对所述第四子块加密。

12．如权利要求 11 所述的介质，其中所述第一组合块或所述第二组合块是所述第一异或块、第二异或块、第三子块和第四子块的串接。

13．如权利要求 8 所述的介质，其中所述加密的动作包括执行高级加密标准块密码加密函数。

14．如权利要求 8 所述的介质，其中所述块是消息的一部分，且还包括把所述块划分成所述第一子块和所述第二子块。

15．如权利要求 14 所述的介质，进一步包括把所述消息划分成数个块，且其中所述块是所述数个块之一。

16．如权利要求 8 所述的介质，其中所述块为 512 位长，而所述子块为 128 位长。

17．如权利要求 8 所述的介质，还包括在所述第一和第二组合块上计算散列。

18．一种方法，包括：

计算下式，其中 A、B、C 和 D 是消息的各部分，E 是加密函数，$_{cd}$ 是从部分 C 和 D 所产生的加密密钥，$_{ab}$ 是从部分 A 和 B 所产生的加密密钥：

$C \oplus E_{ab}(C)$

$D \oplus E_{ab}(D)$

$A \oplus E_{cd}(A)$

$B \oplus E_{cd}(B)$

以及，

计算下式以提供散列 h，其中 H 是安全散列函数：

$$H(C \oplus P \| D \oplus Q \| A \| B \| A \oplus R \| B \oplus S \| C \| D)=h.$$ 。

19．如权利要求 18 所述的方法，其中 E 包括 128 位高级加密标准（AES）加密函数。

20．如权利要求 18 所述的方法，其中 H 包括安全散列算法-1（SHA-1）。

加强安全散列函数

背景

安全散列函数常常被用于认证消息。例如，数字签名依靠安全散列函数。人们能通过签署使用安全散列函数计算得到的文件的散列而实现对该文件的数字签署。以后，该数字签名能通过使用相同的散列函数计算声称是被该人签署过的文件的文件散列而得到认证。如果第一散列和第二散列是相同的，那么该文件被认为是相同的。如果两份文件相同，那么数字签名就得到了认证。

这两份文件被认为是相同的——而不是已知是相同的，因为两个不同的文件的散列也有可能是相同的。这种情况被称作为“冲突”。

冲突的一个例子可用数学术语来显示。假设一安全散列函数“H(M)”可在任意长度的消息“M”上运行并返回一固定长度的散列“h”。由此，“h = H(M) ”，其中“h”具有一固定长度。然而，这没有解决如下的可能性：如果消息“M1”大于固定长度散列“h”，那么两个不同的消息“M1”和“M2”可能具有相等的散列“h”，这样“H(M1) = H(M2) ”。如果“H(M1) = H(M2) ”，那就会产生冲突。

发生冲突的概率对于确定任何信息可靠的概率是很重要的。例如，对于产生 160 位的散列的安全散列函数，两个随机消息具有相同的散列的概率是 $1/2^{160}$。但对于其中任何两个消息可能具有相同散列的一组随机消息来说，，该组消息的数量不需要如人们所预计的那么大。对于 160 位的散列来说，该消息组只需要具有 2^{80} 个消息。

因此，试图引起造成冲突（即，使 2 个消息具有相同的散列）的人可以在 2^{80} 个或者更少数量的 160 位的散列中产生该冲突。例如，假设 Willy 想诈骗 George。Willy 可以写两个合同，一个对 George 有利而另一个只对 Willy 自己有利。Willy 可对每个文档做微小的改变（例如添加一个空格）并对每个文档运行散列值。他可以继续这样做直到有利于 Willy 的合同之一的散列值与有利于 George 的合同之一的散列值相互匹配。通过这样的做法，Willy 能在有利于 George 的合同“M_g”和表利于 Willy 的合同“M_w”之间制造出一冲突，使得

“$H(M_g)= H(M_w)$ ”。一旦 Willy 完成这个工作之后，他让 George 使用 George 用来签署该合同的散列值为“h”的协议来签署有利于 George 的合同。在将来的某个时间，Willy 用 George 没有签署过的有利于 Willy 的合同来替换之前 George 签署过的有利于 George 的合同。而 Willy 可以说服审判员（例如法院的法官）George 签署过有利于 Willy 的合同因为有利于 Willy 的合同将匹配 George 对于有利于 George 的合同的签名的散列值。

用大的散列例如 160 位的散列中造成冲突，直到最近都被认为是非常困难的。根据当前的处理速度改变和计算 2^{80} 个消息的散列要使用数百台计算机花费数百或者数千年的时间。然而最近，一些人已经证明通过对消息的位做小的并且可控制的改变，只需要用 2^{69} 个消息就可以制造出冲突。如果这是真的，那么几百台计算机在数月中就可能产生冲突；在 5 或 10 年内，可能一台计算机能在一年之内产生冲突。

这种可能性和其它攻击使制造潜在冲突变得更加容易，从而使得人们对一些安全散列函数的安全性和用途产生了怀疑。

发明概述

描述用于加强安全散列函数系统和/或方法（“工具”）。在一些实施例中，所述工具可以基于消息创建随机出现元素。所述工具之后将元素与消息组合。使用一安全散列函数将该组合散列，其中所得到的散列与用相同的安全散列函数对不具有该元素的消息计算所得的散列的长度相同。通过这种做法，所述工具可通过降低一些类型的企图制造冲突的攻击的有效性来加强安全散列函数。

在一些其它的实施例中，所述工具可加密信息的子块，计算子块的异或（XOR）并加密子块，接着把子块和异或子块组合。通过这种做法，可对不能通过操纵消息中的位而轻易控制的位的可复制组合进行散列。所得到的散列比通过对消息自身计算而得到的散列更安全。本概述用于以一种简化的方式引入一种概念的节选，其将在下面的详细描述中进一步描述。本概述不是为了识别所要求的主题的关键或基本特征，也不是为了确定所要求主题的范围。

附图简述

图 1 示出其中可操作各个实施例的示例性操作环境。

图 2 是处理消息的示例性流程图。

图 3 是使用块加密和异或操作的安全散列函数的示例性过程。

图 4 示出示例性的消息的示例块、所述块的子块和已组合的块。

图 5 是降低对加密密钥的控制的示例性过程。

图 6 示出示例性的消息的异或块、加密子块和已组合的块。

在本文和附图中始终是用相同的数字表示同样的组件和特征。

详细描述

纵览

下面的文档描述了加强安全散列函数的系统和/或方法（“工具”）。所述工具可通过降低一些类型的企图制造冲突的攻击的有效性来加强安全散列函数。例如，所述工具可增强安全散列函数，防止基于对消息的位进行小而可控制的改变而进行的攻击。

在一些实施例中，所述工具可以基于消息并使用一过程创建随机出现元素。所述工具之后把元素与消息组合。该组合可使用一安全散列函数进行散列，其中所得到的散列与用相同的安全散列函数对不具有该元素的消息计算所得的散列的长度相同。另一随机出现元素也可以基于另一消息并使用相同的过程创建，如果两个消息是相同的，那么所得到的另一随机出现元素与第一随机出现元素相同。通过这种做法，另一组合可在将来的某个时刻从另一消息中创建，如果两个消息相同，那么根据这另一个组合计算所得的散列将与通过第一组合计算所得的散列相同。这可实现消息的认证。

在另一实施例中，所述工具在消息的块上执行块加密，得到经加密的块。所述工具然后计算一操作，例如，所述已加密块和块的加（ADD）、减（SUB）或异或（XOR），得到加（ADD）、减（SUB）或异或（XOR）块。所述工具然后把加（ADD）、减（SUB）或异或（XOR）块与所述块组合，形成经组合的块。所述经组合的消息之后被散列。经组合的消息中的位不能通过改变消息中的位被轻易地控制，它能潜在地阻止通过仔细地改变消息的位来制造冲

突的企图。

在又一实施例中，所述工具把消息的块分成子块。然后所述工具基于一个或多个子块计算加密密钥。该工具用一个具有该密钥的加密函数对一子块进行加密从而提供经加密的子块，其中所述密钥从一个不同于所述被加密的子块的子块中计算得到。这能被重复直到经加密的子块的数量与子块的数量相同。经加密的子块和子块然后被串接。所得到的串接包含与所述块大小相同的两个串接块。每个串接块包含子块和经加密的子块。所述工具还能建立包括子块和经加密的子块的每个串接块，其中经加密的子块是使用从所述两个串接块中的另一个计算所得的密钥进行加密。所述串接块被一安全散列函数散列。

示例性操作环境

在详细描述所述工具之前，下面对于示例性操作环境的讨论被用于辅助读者理解所述工具在哪里以及如何被使用。下面所提供的描述仅仅构成了一个例子，并不旨在将所述工具的应用限制在任何一种特定操作环境。

图 1 示出一种此类操作环境 100，包括具有（诸）处理器 104 和计算机可读介质 106 的计算机 102。所述处理器能访问和/或执行计算机可读介质。计算机可读介质包括或已经接入到能计算一安全散列的散列生成器 108、能加强一安全散列函数和两个消息的散列预处理器 110，其中所述两个消息为第一消息 112 和第二消息 114。

预处理消息

下面的讨论描述了示例性的方法，根据这些方法，操作环境 100 的诸元件能通过在对消息计算散列之前处理该消息而加强一安全散列函数。该过程能有效降低对手对于被用于计算散列的位的控制。

参考图 2，示出用于处理第一消息 112 的示例性流程图 200。流程图 200 示出环境 100 中的元件执行的一组动作以及它们之间伴随的通信。所述动作和伴随的通信用箭头标记。该流程图可在任何合适的硬件、软件、固件或它们的组合中实施。在软件和固件的情形中，该图代表了被实现为的计算机可执行指令的操作组。

在箭头 1，散列预处理器 110 基于第一消息 112 计算可复制的、随机出现的元素 202。所述工具通过对消息一部分和消息的经加密的一部分执行逻辑操作而计算该元素，比如在下面的实施例中所描述的逻辑操作。元素 202 实际上并不是随机的，但显得足够随机，以使得不能容易地通过对第一消息的位的改变而控制该元素。所述元素是可复制的；相同的元素可基于与用于产生所述元素相同的消息和相同的协议而产生。因此，如果两个相同的消息被用同样的方法处理并且所得到的组合被使用相同的散列函数进行散列，那么所述的相同的消息将具有相同的散列。这就使得消息可通过散列被认证。

在箭头 2，散列预处理器 110 建立一组合 204，其包括第一消息 112 和元素 202。该组合包含不能被轻易控制的位。因此，尽管该组合包括第一消息的位和依赖于那些位(元素 202)的位，所述组合的位不会轻易地受到通过操纵第一消息来进行的恶意操纵。在这一实施例中，所述组合可以是所述元素和第一消息的任意组合，以使得该组合可以基于第一消息复制。

在箭头 3，散列生成器 108 计算该组合的散列。该散列能有效地允许使用和散列生成器相同的散列函数以及相同的元素来使得第一消息得到认证。通过以该散列预处理器用相同的方式处理相同的第一消息而容许该相同的元素。

在这个实施例中，散列生成器 108 没有改动。然而，用于计算散列的位是可修改的，但不是散列函数本身。这就使得能够继续使用标准安全散列函数和系统，例如安全散列算法-1（“SHA-1”）和安全散列系统（SHS）。

尽管对手可能已经完全控制消息自身中的位，通过如流程图 200 所示的对消息的预处理，对手不再能控制用于计算散列的位中的大部分。这是因为所述散列是根据预处理的消息计算出而并不仅仅是根据消息本身。希望制造冲突（两个不同消息具有相同的散列）的对手现在可企图通过对一将被散列的消息的子块的起始位做小而可控制的变化而制造冲突。然而，所述工具根据流程图 200 可降低对手提高在对手的信息和另外的消息之间制造冲突的可能性的能力。对手对信息的微小的变化将大大改变经预处理的消息从而潜在地减少对手制造冲突的能力。

块加密和异或操作

下面的章节描述了示例性的方法，根据这些方法，所述工具使用块加密和异或操作加强安全散列函数。需要明白和理解的是下面的描述并不旨在对所要求的主题内容的应用进行限制。

在图 3 中，示出一示例性过程 300，其例示代表单独操作或动作的一系列步骤。在下述的一个实施例中，诸如散列预处理器 110 的图 1 的操作环境 100 的元件实施这一过程。此处所揭示的这一过程和其它过程可在任何适当的硬件、软件、固件或它们的组合中实施；在软件和固件的情形中，这些过程代表了被实现为存储在计算机可读介质 106 中且可由处理器 104 执行的计算机可读指令的一组操作。

步骤 302 把消息划分成块。这些块可以是可由安全散列函数处理的大小。在图 4 所示的实施例中，散列预处理器 110 把第一消息 112 划分成一系列 512 位的块，其中 3 块在 402 中示出。

步骤 304 把消息划分成子块。每个子块从诸如块 402 的大块划分而成。在所示出的实施例中，4 个 128 位的子块 404 从每个块 402 划分而来。子块从这些块的其中之一划分而来并示出为标记有：“A”、“B”、“C”和“D”。

步骤 306 对消息的子块进行加密以提供经加密的子块。在一个实施例中，步骤 306 对消息的子块加密以提供经加密的子块，其中子块的数量是任意的。

实施被称为“K”、“L”和“T”的 3 个操作。K 函数产生一密钥。L 函数产生一填充（pad）使得即使密钥被得知，加密也不可逆。T 函数产生一标签（“t”），其确保子块的转换保持高概率的一对一。这可用以下方式在数学上表示，四个子块分别为 A、B、C 和 D：

$$K(A,B,C,D) = \alpha, \beta, \gamma, \delta$$

$$L(A, B, C, D) = A', B', C', D'$$

$$T_{key}(A, B, C, D) = t$$

更广义地说，可以在数学上表达为：

$$E_{\alpha}(A) = R, E_{\beta}(B) = S, E_{\gamma}(C) = P, E_{\delta}(D) = Q$$

$$U = R + A', V = S + B', W = P + C', X = Q + D'$$

$$T_{U,V,W,X}(A, B, C, D) = t$$

对于输出 t、U、V、W 和 X，一种情况如下所示：

$K(A, B, C, D) = CD, CD, AB, AB$

$L(A, B, C, D) = A, B, C, D$

$T_{key}(A, B, C, D) = A, B, C, D$

并且，为了避免允许对手选择 AB 作为密钥，函数 K 可为：

$K(A, B, C, D) = CD, CD, RS, RS$

为了进一步混合块，还可使用另一个双射函数。

下面示出本过程的一个例子。这里步骤 306 使用块密码加密原始子块 A、B、C 和 D。步骤 306 可由散列预处理器 110 来做这个步骤，由此使用具有加密密钥“$_n$”的加密函数“E_n”加密子块 A。这里“R”是加密子块 A 的结果，称为经加密的子块。所述加密密钥从另一子块的位产生，该另一子块由相同的块划分所得，在这种情况中，是子块 B、C、D 或 B、C 和/或 D 的组合。在该实施例中，密钥从子块 C 和 D 中产生并被标记为“$_{cd}$”。可在数学上表示为：

$E_{cd}(A)=R$

所使用的加密函数可为 128 位高级加密标准（“AES”）加密函数。密钥可从其它子块产生，以使得密钥是基于一个或多个其它子块并且也是可复制的。

同样的，步骤 306 能有效地加密其它子块 B、C 和 D 从而提供其它经加密的子块。可以如下在数学上表示为：

$E_{cd}(B)=S$

$E_{ab}(C)=P$

$E_{ab}(D)=Q$

因此，一些经加密的子块被建立，其中的每个都是可复制的并且是基于一个或多个子块。

步骤 308 计算一子块和一经加密的子块的异或从而提供一异或子块。

这里散列预处理器 110 计算一异或（数学符号中表示为 XOR 或“$\oplus$”），

结果如下：

$C \oplus P$

$D \oplus Q$

$A \oplus R$

$B \oplus S$

这些结果的每一个都表现为随机的并且不轻易地受到对手的控制。因此，即使对手例如用小而可控制的变化恶意修改 A、B、C 或 D 的位，然而不能用所允许修改 A、B、C 或 D 的位的相应控制等级来恶意地选择异或子块。

步骤 310 把子块和异或子块组合从以提供经组合的子块。每个经组合的块与所述块具有相等的位大小，尽管组合块的数量可能是其所基于的块数量的两倍之多。

如下所示，子块 A、B、C 或 D 与异或块 $C \oplus P$、$D \oplus Q$、$A \oplus R$ 和 $B \oplus S$ 相串接。它们组合成两个经组合的块：

$C \oplus P \| D \oplus Q \| A \| B$

和

$A \oplus R \| B \oplus S \| C \| D$

这些块在图 4 的 406 示出。这些经组合的块中的每个都包含对消息的位做小而可控制的变化后仍然无法轻易控制的元素。这些元素也使得使用对消息的位做小而可控制的变化的来制造冲突的企图变得更为困难。

这些经组合的块中的每一个都可以是使得其能以与它们所基于的消息的块被散列的同样方式被散列的大小。尽管需要更多计算时间，许多安全散列函数能轻易地处理相同大小的额外的块。

步骤 304、306、308 和/或 310 可对每个块或该消息的一个或多个其他块重复。通过重复这些操作，大多数或所有的消息可被预处理从而包含可复制的、随机出现的元素，例如示出的实施例中的异或块。

步骤 312 在子块和异或子块的组合上计算一安全散列函数。如果消息包含

多于一个的块（这是经常出现的情况），那么安全散列函数在子块和异或子块的多个组合上计算。在示出的实施例中，散列生成器 108 使用 SHA-1 来计算子块和异或子块的组合的 160 位散列。通过这种做法，一散列可以基于消息并且可从该消息被复制地被计算。

例如，如果第二消息 114 如图 3 所述地被处理，如果第二消息 114 和第一消息 112 是相同的，那么第二消息所得到的散列将与第一消息 112 所得到的散列相同。如果散列是相同的，那么第一消息或第二消息就能得到认证。如果它们不相同，那就说明第一消息和第二消息不相同。

加密密钥

下面的讨论描述示例性方法，根据这些方法，所述工具通过降低对于在预处理消息的时候使用的加密密钥的潜在控制来加强安全散列函数。如图 2 或图 3 所述，所述工具部分通过创建元素来加强安全散列功能，所述元素不能被对手通过操纵从中导出元素的消息的位而被轻易地控制。

图 5 阐述了用于降低对于加密密钥的控制的示例性过程 500。过程 500 被示出为被例示为表示由诸如散列预处理器 110 的图 1 的操作环境 100 的元素实现的独立操作或动作的一系列步骤。

步骤 502 从消息产生可复制的、随机出现的元素。在如图 6 所示的实施例中，提供从步骤 308 中产生的异或子块 602。在这个实施例中，这些异或子块和组成异或子块的经加密的字块 P、Q、R 和 S 是中间的，因为它们用于建立将被组合成块用于散列的元素，而它们自身无法被散列。

因此，下面的内容可被接收：

$C \oplus P$

$D \oplus Q$

$A \oplus R$

$B \oplus S$

步骤 504 从来自该消息的可复制、随机出现的元素中产生随机出现的加密密钥。这里，散列预处理器 110 基于图 6 所示的异或子块 602 建立随机出现的

加密密钥“$_{RAE-n}$”，其中“$_n$”表示从中产生密钥的（诸）元素。这可表示为：

$$C \oplus P \| D \oplus Q \rightarrow {}_{RAE\text{-}cpdq}$$

$$A \oplus R \| B \oplus S \rightarrow {}_{RAE\text{-}arbs}$$

步骤 506 使用随机出现的加密密钥（ “RAE 密钥”）加密部分消息。这里从图 3 的步骤 304 中形成的子块 404 A、B、C 和 D 由在步骤 504 中从异或子块 602 所产生的 RAE 密钥加密。这就提供了经 RAE 密钥加密的子块 604，标记为 V、W、T 和 U。因此，使用具有 RAE 密钥 $_{RAE-n}$ 的加密函数 E_n 对子块 404 加密从而提供经 RAE-密钥加密的子块。

这可在数学上表示为：

$$E_{RAE\text{-}cpdq}(A) = V$$

$$E_{RAE\text{-}cpdq}(B) = W$$

$$E_{RAE\text{-}arbs}(C) = T$$

$$E_{RAE\text{-}arbs}(D) = U$$

这些经 RAE-密钥加密的子块可被当作根据图 3 中的过程 300 的经加密的块。这里 P=T、Q=U、R=V 且 S=W。例如，根据图 3 中的步骤 308 和 310，可得到下面的经组合的块 606：

$$C \oplus T \| D \oplus U \| A \| B$$

和

$$A \oplus V \| B \oplus W \| C \| D$$

结论

上述系统和方法加强了安全散列函数。这些系统和方法可极大地降低了企图在消息之间制造冲突的一些攻击的有效性。通过这种做法，可使用安全散列函数对消息进行更高确定性的认证。尽管该系统和方法是以针对结构特征和/或方法动作的语言进行描述的，需要理解的是在权利要求中所定义的系统和方法并不一定限于所描述的指定的特征或动作。相反，这些特定的特征和动作被

揭示为实现所要求的系统和方法的示例形式。

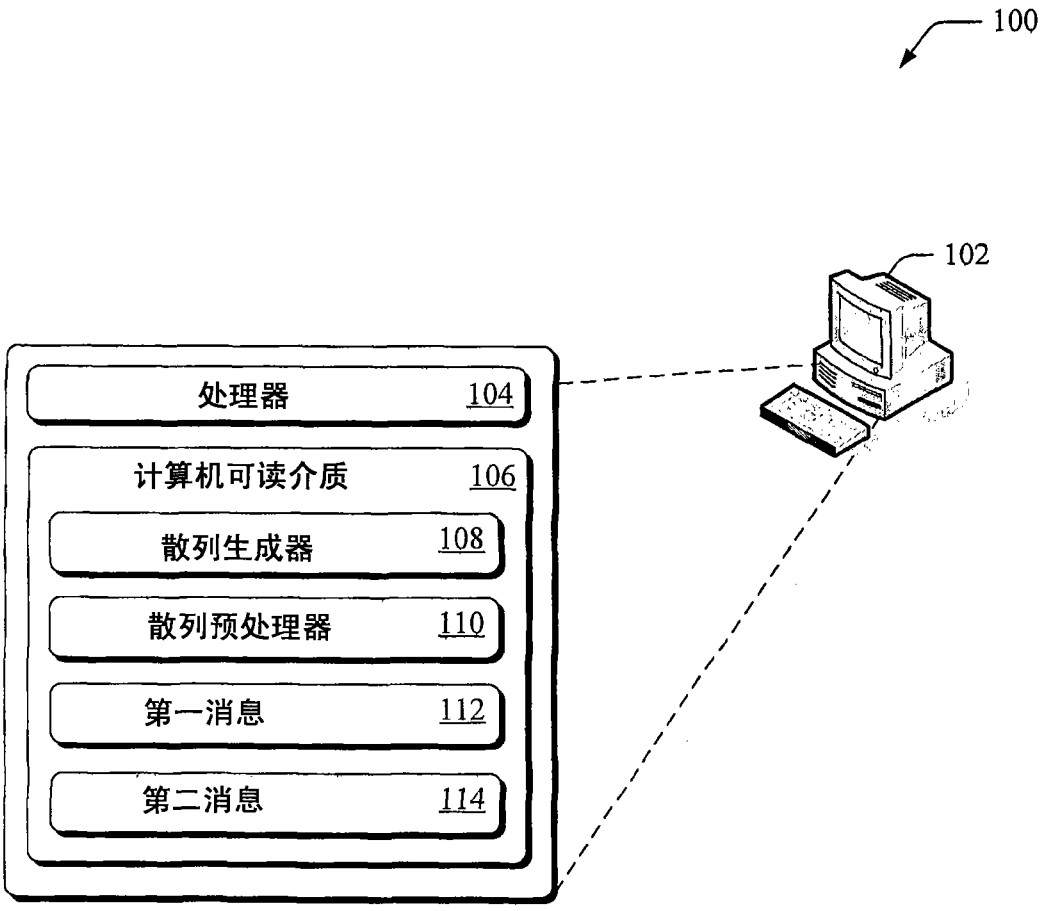

100
102
处理器 104
计算机可读介质 106
散列生成器 108
散列预处理器 110
第一消息 112
第二消息 114

图 1

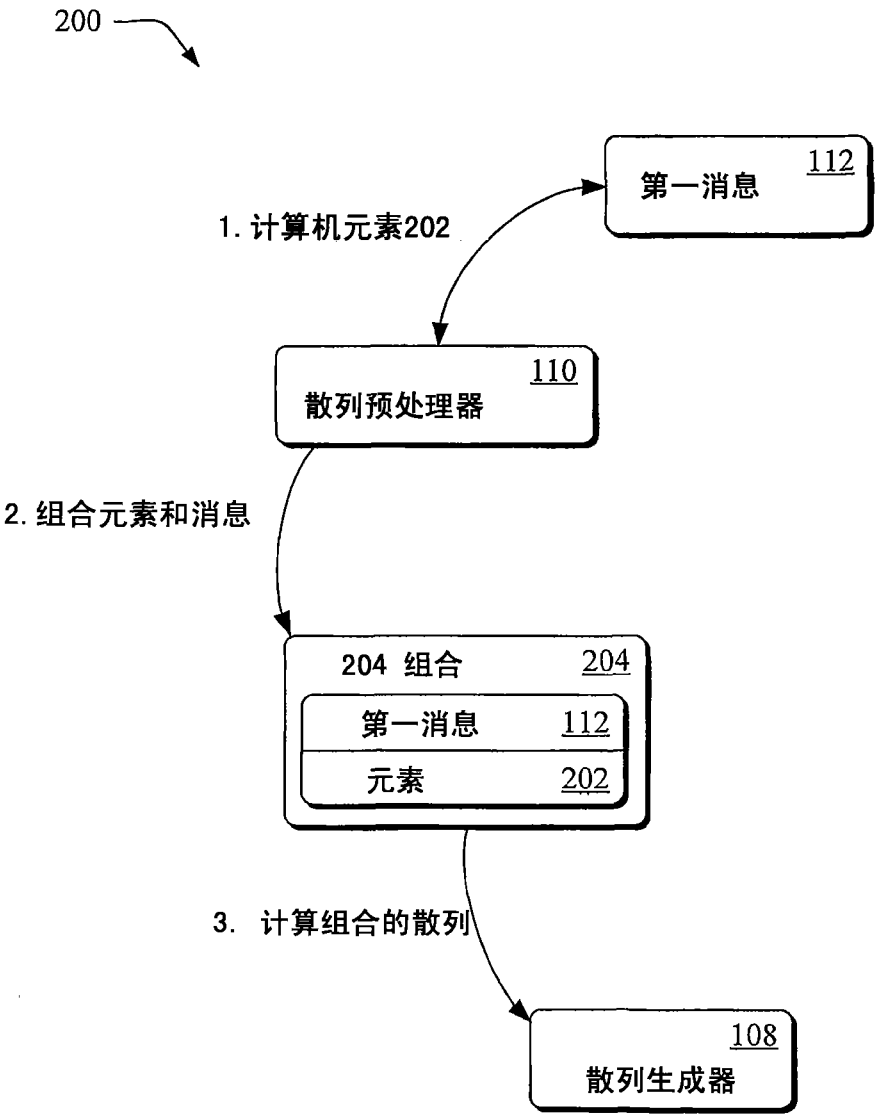

200
第一消息
112
1. 计算机元素202
散列预处理器
110
2. 组合元素和消息
204 组合
204
第一消息
112
元素
202
3. 计算组合的散列
散列生成器
108

图 2

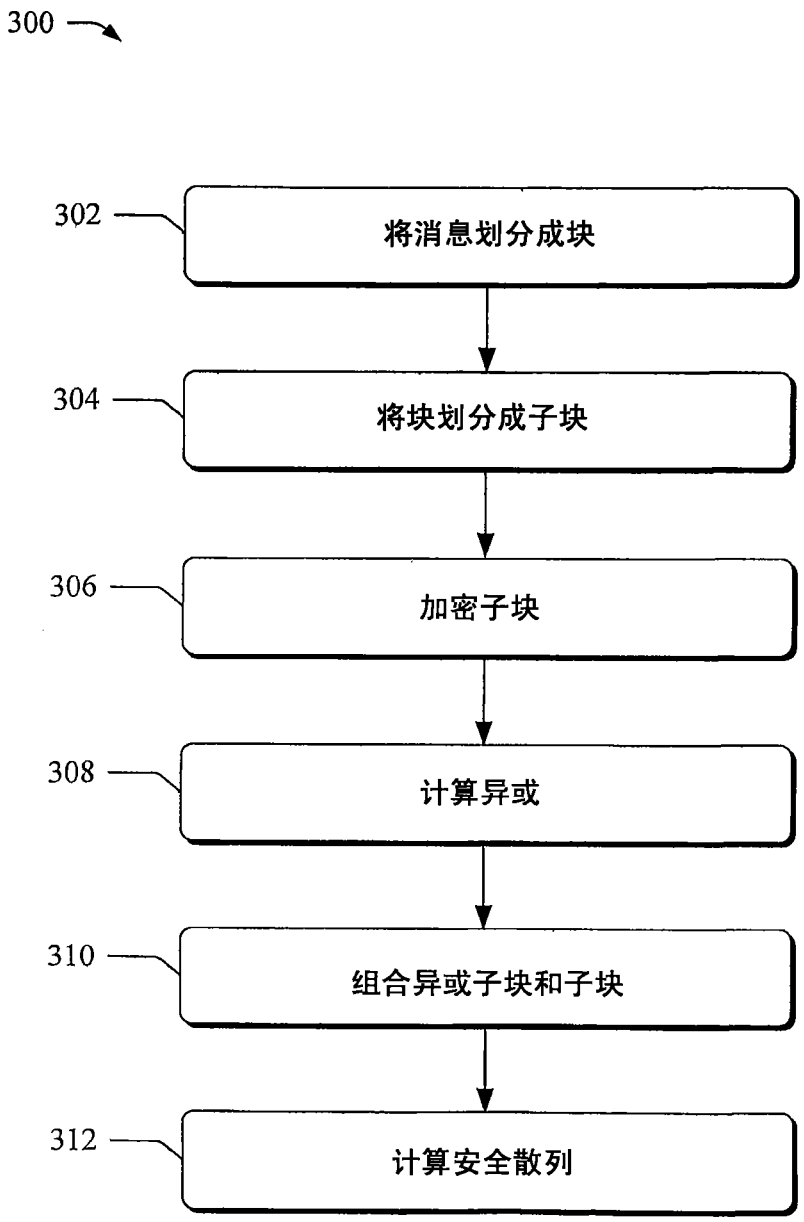

300
302
将消息划分成块
304
将块划分成子块
306
加密子块
308
计算异或
310
组合异或子块和子块
312
计算安全散列

图 3

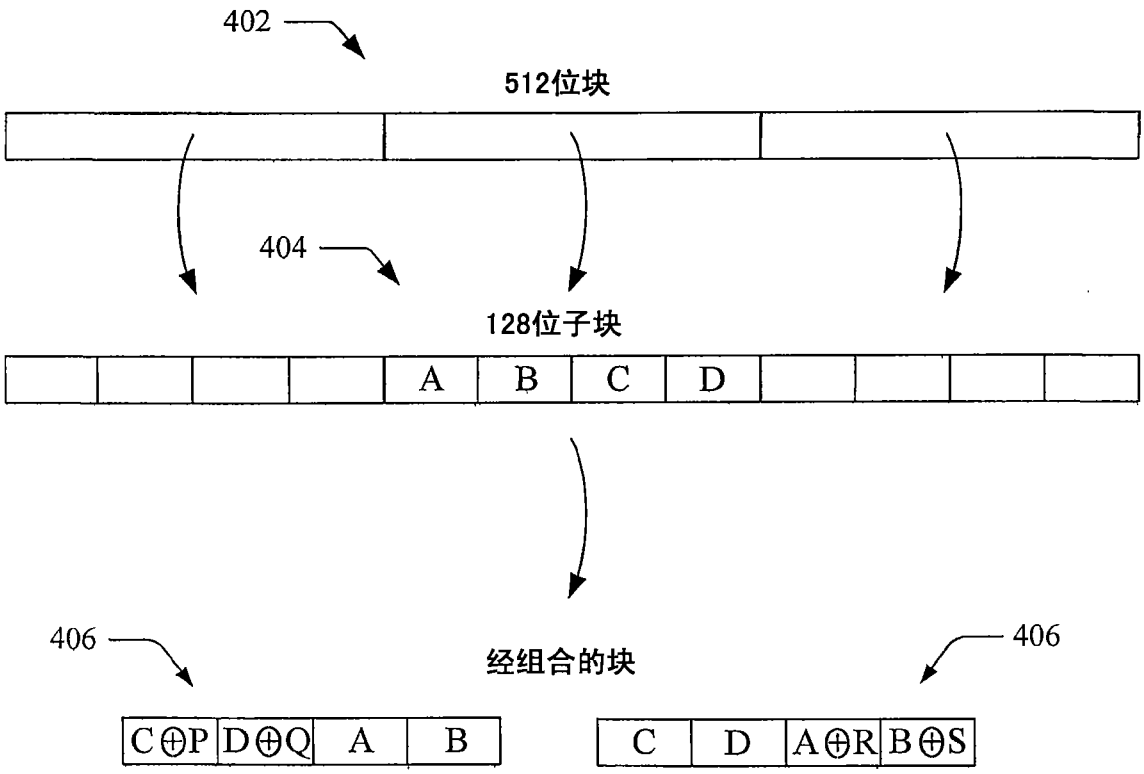


图 4

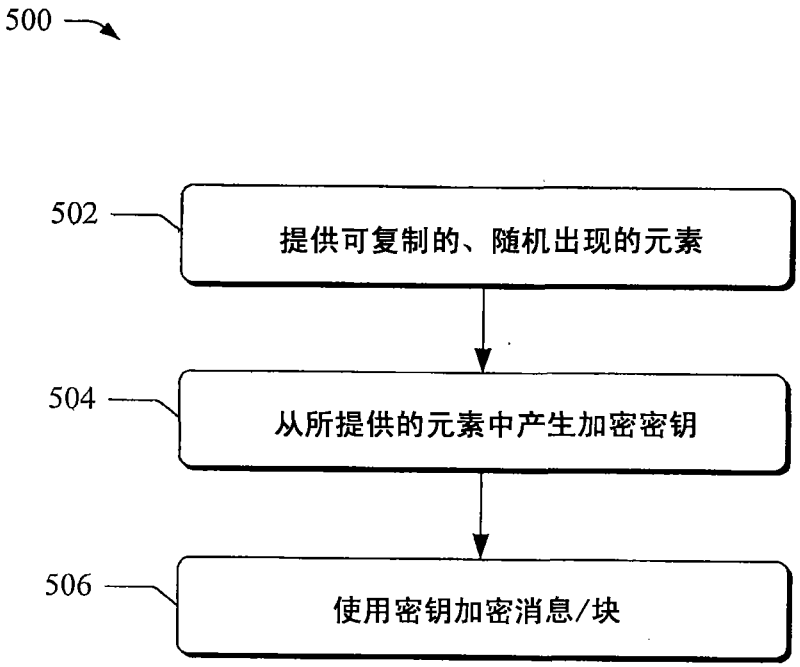


图 5

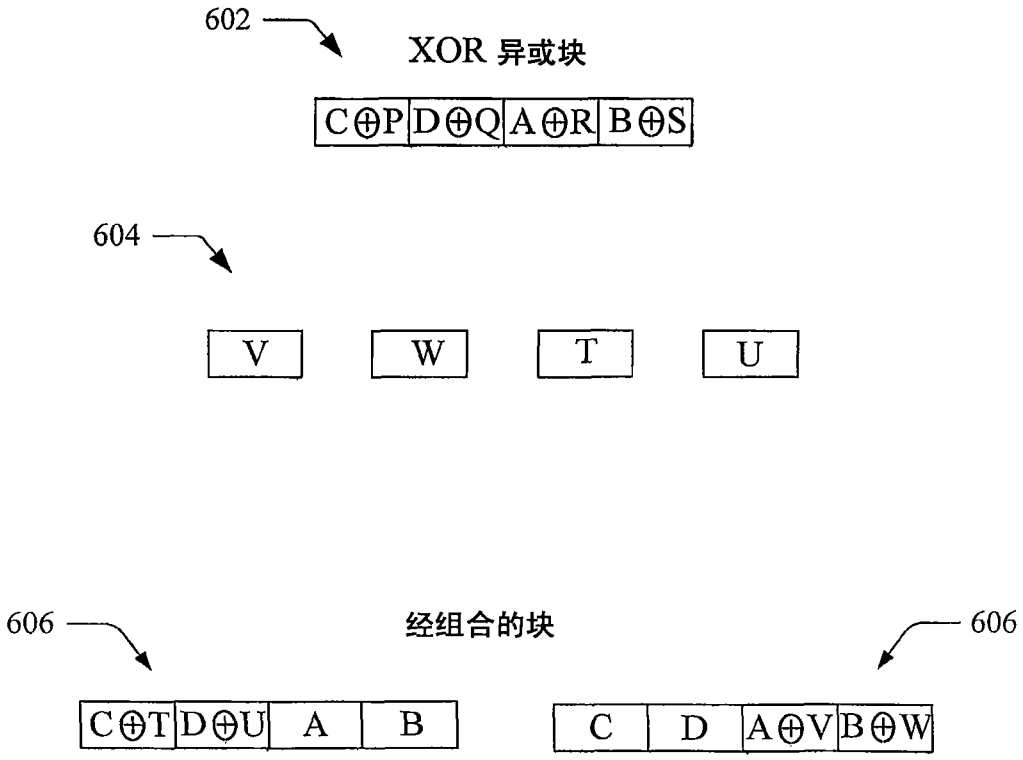


图 6