(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 101558598 B
(45)授权公告日 2013.03.27

(21)申请号 200780046451.7

(22)申请日 2007.12.14

(30)优先权数据
11/611,827 2006.12.15 US

(85)PCT申请进入国家阶段日
2009.06.15

(86)PCT申请的申请数据
PCT/US2007/087526 2007.12.14

(87)PCT申请的公布数据
WO2008/076861 EN 2008.08.21

(73)专利权人 高通股份有限公司
地址 美国加利福尼亚

(72)发明人 A·甘特曼 G·G·罗丝 J-H·崔 J·W·内伦贝格二世

(74)专利代理机构 永新专利商标代理有限公司 72002
代理人 刘瑜 王英

(51)Int.Cl.
*H04L 9/06*(2006.01)
*H04L 9/18*(2006.01)
*G09C 1/00*(2006.01)

(56)对比文件
US 4979832 A,1990.12.25,
US 4870683 A,1989.09.26,
WO 2004/045134 A1,2004.05.27,
US 6075859 A,2000.06.13,

审查员 李凯

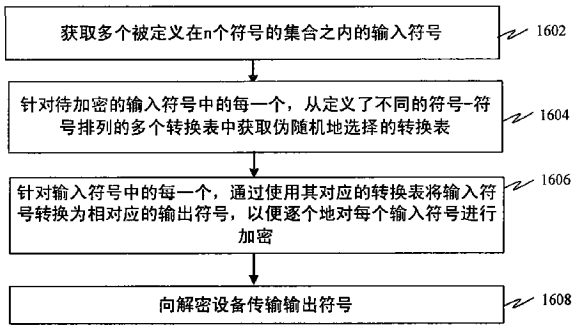
权利要求书 5 页 说明书 17 页 附图 14 页

(54)发明名称

组合合成器加密方法

(57)摘要

另一个特征提供了一种保护加密符号安全的有效加密方法。通过使用单独的、伪随机地选择的转换表来对每个明文符号进行加密。该转换表是基于伪随机数和符号混洗算法在运行中高效地生成的，而非将符号的每种可能排列都预先存储为转换表。类似地，接收设备可在运行中生成反转换表，以便对接收到的加密符号进行解密。




CN 101558598 B

1. 一种在加密设备上运行的、用于保护在电话和安全服务器之间的通信的方法，包括：

通过对从所述电话接收的 DTMF 音调进行转变，来获取多个输入符号；

针对待加密的所述输入符号中的每一个，从定义了不同的符号－符号排列的多个转换表中获取伪随机地选择的转换表；

针对所述输入符号中的每一个，通过使用其对应的转换表将所述输入符号转换为相对应的输出符号，以便逐个地对每个输入符号进行加密；以及

以对应于所述输出符号的 DTMF 音调向所述安全服务器传输所述输出符号。

2. 根据权利要求 1 所述的方法，其中，针对待加密的所述输入符号中的每一个获取所述伪随机地选择的转换表的操作包括：

针对第一输入符号获取伪随机地选择的第一转换表；以及

针对第二输入符号获取伪随机地选择的第二转换表；

并且其中，针对所述输入符号中的每一个通过使用其对应的转换表将所述输入符号转换为相对应的输出符号的操作包括：

通过使用所述第一转换表将所述第一输入符号转换为第一输出符号；以及

通过使用所述第二转换表将所述第二输入符号转换为第二输出符号。

3. 根据权利要求 1 所述的方法，其中，所述多个输入符号是由 N 个符号的集合定义的，其中，N 为正整数，并且转换表是所述 N 个符号的排列。

4. 根据权利要求 3 所述的方法，其中，针对待加密的所述输入符号中的每一个获取所述伪随机地选择的转换表的操作包括：

针对待加密的第一输入符号获取第一伪随机数；以及

混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同的排列并使用该排列作为所述第一输入符号的转换表。

5. 根据权利要求 4 所述的方法，其中，通过如下操作获取所述第一伪随机数：

针对所述第一输入符号生成伪随机数，其中，该伪随机数为 k 位长，并且 k 为正整数；

判断所述伪随机数是否在最大数 Pmax 之内，其中，Pmax 是小于最大阈值 $2^k$ 的 N 阶乘的最大倍数；

如果所述伪随机数大于所述最大数 Pmax，则将所述伪随机数丢弃；

针对所述第一输入符号获取不同的伪随机数，直到获取到可接受的伪随机数为止，其中，所述可接受的伪随机数小于或等于最大数 Pmax；以及

求所述可接受的伪随机数除以 N 阶乘后的余数，以获取所述第一伪随机数。

6. 根据权利要求 4 所述的方法，其中，混洗所述 N 个符号的集合的操作包括：

以所述 N 个符号的集合中的所有符号来初始化排列向量 P；以及

基于所述第一伪随机数来混洗所述排列向量中的符号。

7. 根据权利要求 1 所述的方法，还包括：

将所述输出符号传输给解密设备。

8. 根据权利要求 1 所述的方法，还包括：

针对待加密的所述输入符号中的每一个，从定义了不同的符号－符号排列的所述多个转换表中获取伪随机地选择的第二转换表；以及

针对所述输出符号中的每一个，通过使用其对应的第二转换表将所述输出符号转换为相对应的第二输出符号，以便逐个地对每个输入符号进行进一步加密。

9. 一种用于保护在电话和安全服务器之间的通信的加密设备，包括：

用于通过对从所述电话接收的 DTMF 音调进行转变，来获取多个输入符号的模块；

用于针对待加密的所述输入符号中的每一个，从定义了不同的符号 - 符号排列的多个转换表中获取伪随机地选择的转换表的模块；

用于针对所述输入符号中的每一个而通过使用其对应的转换表将所述输入符号转换为相对应的输出符号，以便逐个地对每个输入符号进行加密的模块；以及

用于以对应于所述输出符号的 DTMF 音调向所述安全服务器传输所述输出符号的模块。

10. 根据权利要求 9 所述的设备，还包括：

用于针对第一输入符号获取伪随机地选择的第一转换表的模块；

用于针对第二输入符号获取伪随机地选择的第二转换表的模块；

用于通过使用所述第一转换表将所述第一输入符号转换为第一输出符号的模块；以及

用于通过使用所述第二转换表将所述第二输入符号转换为第二输出符号的模块。

11. 根据权利要求 9 所述的设备，其中，所述多个输入符号是由 N 个符号的集合定义的，其中，N 为正整数，并且转换表是所述 N 个符号的排列，所述设备还包括：

用于针对待加密的第一输入符号获取第一伪随机数的模块；以及

用于混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的转换表的模块。

12. 根据权利要求 11 所述的设备，还包括：

用于针对所述第一输入符号生成伪随机数的模块，其中，该伪随机数为 k 位长，并且 k 为正整数；

用于判断所述伪随机数是否在最大数 Pmax 之内的模块，其中，Pmax 是小于最大阈值 $2^k$ 的 N 阶乘的最大倍数；

用于如果所述伪随机数大于所述最大数 Pmax 则将所述伪随机数丢弃的模块；

用于针对所述第一输入符号获取不同的伪随机数直到获取到可接受的伪随机数为止的模块，其中，所述可接受的伪随机数小于或等于最大数 Pmax；以及

用于求所述可接受的伪随机数除以 N 阶乘后的余数以获取所述第一伪随机数的模块。

13. 根据权利要求 9 所述的设备，还包括：

用于将所述输出符号传输给解密设备的模块。

14. 根据权利要求 9 所述的设备，还包括：

用于针对待加密的所述输入符号中的每一个从定义了不同的符号 - 符号排列的所述多个转换表中获取伪随机地选择的第二转换表的模块；以及

用于针对所述输出符号中的每一个而通过使用其对应的第二转换表将所述输出符号转换为相对应的第二输出符号以便逐个地对每个输入符号进行进一步加密的模块。

15. 一种用于保护在电话和安全服务器之间的通信的加密设备，包括：

用于从所述电话接收 DTMF 音调的输入接口；

耦合到所述输入接口的处理电路，该处理电路用于：

通过对所述 DTMF 音调进行转变来获取多个输入符号 ;

针对待加密的所述输入符号中的每一个,从定义了不同的符号 - 符号排列的多个转换表中获取伪随机地选择的转换表 ;以及

针对所述输入符号中的每一个,通过使用其对应的转换表将所述输入符号转换为相对应的输出符号,以便逐个地对每个输入符号进行加密 ;以及

用于以对应于所述输出符号的 DTMF 音调向所述安全服务器传输所述输出符号的输出接口。

16. 根据权利要求 15 所述的设备,还包括 :

耦合到所述处理电路的输出接口,该输出接口用于传输所述输出符号。

17. 根据权利要求 15 所述的设备,其中,所述多个输入符号是由 N 个符号的集合定义的,其中,N 为正整数,并且转换表是所述 N 个符号的排列。

18. 根据权利要求 17 所述的设备,还包括 :

耦合到所述处理电路的密钥流生成器,该密钥流生成器用于针对待加密的第一输入符号从所述密钥流生成器处获取第一伪随机数 ;以及

耦合到所述处理电路的转换表生成器,该转换表生成器用于混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的转换表。

19. 根据权利要求 18 所述的设备,其中,所述处理电路还用于 :

针对所述第一输入符号生成伪随机数,其中,该伪随机数为 k 位长,并且 k 为正整数 ;

判断所述伪随机数是否在最大数 Pmax 之内,其中,Pmax 是小于最大阈值 $2^k$ 的 N 阶乘的最大倍数 ;

如果所述伪随机数大于所述最大数 Pmax,则将所述伪随机数丢弃 ;

针对所述第一输入符号获取不同的伪随机数,直到获取到可接受的伪随机数为止,其中,所述可接受的伪随机数小于或等于最大数 Pmax ;以及

求所述可接受的伪随机数除以 N 阶乘后的余数,以获取所述第一伪随机数。

20. 根据权利要求 17 所述的设备,其中,所述处理电路还用于 :

针对待加密的所述输入符号中的每一个,从定义了不同的符号 - 符号排列的所述多个转换表中获取伪随机地选择的第二转换表 ;以及

针对所述输出符号中的每一个,通过使用其对应的第二转换表将所述输出符号转换为相对应的第二输出符号,以便逐个地对每个输入符号进行进一步加密。

21. 一种在解密设备上运行的、用于保护在电话和安全服务器之间的通信的方法,包括 :

接收来自于加密设备的对应于输入符号的 DTMF 音调 ;

根据所述 DTMF 音调获取多个所述输入符号 ;

针对待解密的所述输入符号中的每一个,从定义了不同的符号 - 符号排列的多个反转换表中获取伪随机地选择的反转换表 ;以及

针对所述输入符号中的每一个,通过使用其对应的反转换表将所述输入符号转换为相对应的输出符号,以便逐个地对每个输入符号进行解密。

22. 根据权利要求 21 所述的方法,其中,多个所述输入符号是由 N 个符号的集合定义的,其中,N 为正整数,并且反转换表是所述 N 个符号的排列,并且所述方法还包括 :

针对待解密的第一输入符号获取第一伪随机数；以及

混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的反转换表。

23. 根据权利要求 22 所述的方法，其中，通过如下操作来获取所述第一伪随机数：

针对所述第一输入符号生成伪随机数，其中，该伪随机数为 k 位长，并且 k 为正整数；

判断所述伪随机数是否在最大数 Pmax 之内，其中，Pmax 是小于最大阈值 $2^k$ 的 N 阶乘的最大倍数；

如果所述伪随机数大于所述最大数 Pmax，则将所述伪随机数丢弃；

针对所述第一输入符号获取不同的伪随机数，直到获取到可接受的伪随机数为止，其中，所述可接受的伪随机数小于或等于所述最大数 Pmax；以及

求所述可接受的伪随机数除以 N 阶乘后的余数，以获取所述第一伪随机数。

24. 根据权利要求 21 所述的方法，还包括：

针对待解密的所述输入符号中的每一个，从定义了不同的符号 - 符号排列的所述多个反转换表中获取伪随机地选择的第二反转换表；以及

针对所述输出符号中的每一个，通过使用其对应的第二反转换表将所述输出符号转换为相对应的第二输出符号，以便逐个地对每个输入符号进行进一步解密。

25. 一种用于保护在电话和安全服务器之间的通信的解密设备，包括：

用于接收来自于加密设备的对应于输入符号的 DTMF 音调的模块；

用于根据所述 DTMF 音调获取多个所述输入符号的模块；

用于针对待解密的所述输入符号中的每一个而从定义了不同的符号 - 符号排列的多个反转换表中获取伪随机地选择的反转换表的模块；以及

用于针对所述输入符号中的每一个而通过使用其对应的反转换表将所述输入符号转换为相对应的输出符号以便逐个地对每个输入符号进行解密的模块。

26. 根据权利要求 25 所述的设备，其中，多个所述输入符号是由 N 个符号的集合定义的，其中，N 为正整数，并且反转换表是所述 N 个符号的排列，并且，所述设备还包括：

用于针对待解密的第一输入符号获取第一伪随机数的模块；以及

用于混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的反转换表的模块。

27. 根据权利要求 26 所述的设备，其中，通过如下模块来获取所述第一伪随机数：

用于针对所述第一输入符号生成伪随机数的模块，其中，该伪随机数为 k 位长，并且 k 为正整数；

用于判断所述伪随机数是否在最大数 Pmax 之内的模块，其中，Pmax 是小于最大阈值 $2^k$ 的 N 阶乘的最大倍数；

用于如果所述伪随机数大于所述最大数 Pmax 则将所述伪随机数丢弃的模块；

用于针对所述第一输入符号获取不同的伪随机数直到获取到可接受的伪随机数为止的模块，其中，所述可接受的伪随机数小于或等于所述最大数 Pmax；以及

用于求所述可接受的伪随机数除以 N 阶乘后的余数以获取所述第一伪随机数的模块。

28. 根据权利要求 25 所述的设备，还包括：

用于针对待解密的所述输入符号中的每一个而从定义了不同的符号 - 符号排列的所

述多个反转换表中获取伪随机地选择的第二反转换表的模块；以及

用于针对所述输出符号中的每一个而通过使用其对应的第二反转换表将所述输出符号转换为相对应的第二输出符号以便逐个地对每个输入符号进行进一步解密的模块。

29. 一种用于保护在电话和安全服务器之间的通信的解密设备，包括：

用于接收来自于加密设备的对应于输入符号的 DTMF 音调的输入接口；以及

耦合到所述输入接口的处理电路，该处理电路用于：

根据所述 DTMF 音调获取多个所述输入符号；

针对待解密的所述输入符号中的每一个，从定义了不同的符号 - 符号排列的多个反转换表中获取伪随机地选择的反转换表；以及

针对所述输入符号中的每一个，通过使用其对应的反转换表将所述输入符号转换为相对应的输出符号，以便逐个地对每个输入符号进行解密。

30. 根据权利要求 29 所述的设备，其中，多个所述输入符号是由 N 个符号的集合定义的，其中，N 为正整数，并且反转换表是所述 N 个符号的排列，并且，所述设备还包括：

耦合到所述处理电路的密钥流生成器，该密钥流生成器用于针对待解密的第一输入符号从所述密钥流生成器处获取第一伪随机数；以及

耦合到所述处理电路的反转换表生成器，该反转换表生成器用于混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的反转换表。

31. 根据权利要求 30 所述的设备，其中，所述处理电路还用于：

针对所述第一输入符号生成伪随机数，其中，该伪随机数为 k 位长，并且 k 为正整数；

判断所述伪随机数是否在最大数 Pmax 之内，其中，Pmax 是小于最大阈值 $2^k$ 的 N 阶乘的最大倍数；

如果所述伪随机数大于所述最大数 Pmax，则将所述伪随机数丢弃；

针对所述第一输入符号获取不同的伪随机数，直到获取到可接受的伪随机数为止，其中，所述可接受的伪随机数小于或等于最大数 Pmax；以及

求所述可接受的伪随机数除以 N 阶乘后的余数，以获取所述第一伪随机数。

32. 根据权利要求 29 所述的设备，其中，所述处理电路还用于：

针对待解密的所述输入符号中的每一个，从定义了不同的符号 - 符号排列的所述多个反转换表中获取伪随机地选择的第二反转换表；以及

针对所述输出符号中的每一个，通过使用其对应的第二反转换表将所述输出符号转换为相对应的第二输出符号，以便逐个地对每个输入符号进行进一步解密。

# 组合合成器加密方法

## 技术领域

[0001] 各种实例总体上涉及安全通信，并且更具体地涉及一种通过有效地生成转换表来保护符号安全的流加密的方法。

## 背景技术

[0002] 通常，通过生成伪随机数的密钥流并将其与明文符号相结合以生成加密输出或密文的方式，来进行流加密。通常，由于其自反性，因此二进制密钥流符号和二进制明文符号会基于逐位的原则通过使用异或 (XOR) 运算进行组合。然而，由于在某些时候更期望加密整个明文符号，而不是对明文符号按位进行加密。因此，不能使用 XOR 运算。典型地，将一个明文符号加入到密钥流符号模 n 中，以获取密文符号。但是，知晓特定符号的位置的主动攻击者能够通过将其从所传输的密文符号中减去，来改变该符号。例如，可通过将每个数字与来自密钥流模十 (10) 的伪随机地生成的数字 (0 到 9) 相加，来对一组明文数字 (0 到 9) 进行加密。然而，知晓特定的明文数字“1”但输出的密文数字为“7”的攻击者可能确定出该密钥流数字为“6”，并且随后能够正确地加密为该特定的符号位置选择的任意其它数字。由于攻击者可以判断该明文符号和密钥流符号是如何组合的，因此，这样的局部解密减弱了余下的加密信息的安全性。也就是说，一旦暴露了密文符号与明文符号间的关系，那么由于该信息可能导致其它密文符号被攻击者解密，因此余下的密文符号的安全会受到威胁。此外，尽管简单的数学运算（例如，加，减，乘等）能迅速并高效地组合明文符号和密钥流符号，但是使用更复杂的数学函数会在加密时引起明显的延时或要求更大的处理资源。

[0003] 因此，需要一种有效的加密方法，这种加密方法不会在一个加密符号被暴露或破坏的情况下减弱其它符号的安全性。

## 发明内容

[0004] 提供了一种在加密设备上运行的、用于保护在电话和安全服务器之间的通信的方法。该加密设备通过对从所述电话接收的 DTMF 音调进行转变，来获取多个输入符号。针对待加密的所述输入符号中的每一个，从定义了不同的符号 - 符号排列的多个转换表中获取伪随机地选择的转换表。针对所述输入符号中的每一个，通过使用其对应的转换表将所述输入符号转换为相对应的输出符号，以便逐个地对每个输入符号进行加密。以对应于所述输出符号的 DTMF 音调向所述安全服务器传输所述输出符号。

[0005] 此外，针对待加密的所述输入符号中的每一个，从定义了不同的符号 - 符号排列的所述多个转换表中获取伪随机地选择的第二转换表。针对所述输出符号中的每一个，通过使用其对应的第二转换表将所述输出符号转换为相对应的第二输出符号，以便逐个地对每个输入符号进行进一步加密。

[0006] 在一个实例中，针对待加密的所述输入符号中的每一个获取所述伪随机地选择的转换表的操作包括：(a) 针对第一输入符号获取伪随机地选择的第一转换表，以及 (b) 针对第二输入符号获取伪随机地选择的第二转换表。针对所述输入符号中的每一个通过使用其

对应的转换表将所述输入符号转换为相对应的输出符号的操作包括：(a) 通过使用所述第一转换表将所述第一输入符号转换为第一输出符号，以及 (b) 通过使用所述第二转换表将所述第二输入符号转换为第二输出符号。所述多个输入符号是由 N 个符号的集合定义的，其中，N 为正整数，并且转换表是所述 N 个符号的排列。

[0007] 在另一个实例中，针对待加密的所述输入符号中的每一个获取所述伪随机地选择的转换表的操作包括：(a) 针对待加密的第一输入符号获取第一伪随机数，以及 (b) 混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的转换表。

[0008] 在一个配置中，通过如下操作来获取所述第一伪随机数：(a) 针对所述第一输入符号生成伪随机数，其中，该伪随机数为 k 位长，并且 k 为正整数，(b) 判断所述伪随机数是否在最大数 Pmax 之内，其中，Pmax 是小于最大阈值 $2^k$ 的 N 阶乘的最大倍数，(c) 如果所述伪随机数大于所述最大数 Pmax，则将所述伪随机数丢弃，(d) 针对所述第一输入符号获取不同的伪随机数，直到获取到可接受的伪随机数为止，其中，所述可接受的伪随机数小于或等于最大数 Pmax，以及 / 或者 (e) 求所述可接受的伪随机数除以 N 阶乘后的余数，以获取所述第一伪随机数。混洗所述 N 个符号的集合的操作包括：(a) 以所述 N 个符号的集合中的所有符号来初始化排列向量 P，以及 (b) 基于所述第一伪随机数来混洗所述排列向量 P 中的符号。

[0009] 还提供了一种包括输入接口和处理电路的、用于保护在电话和安全服务器之间的通信的加密设备。该输入接口可以用于从所述电话接收 DTMF 音调。该处理电路用于：(a) 通过对所述 DTMF 音调进行转变来获取多个输入符号，(b) 针对待加密的所述输入符号中的每一个，从定义了不同的符号 - 符号排列的多个转换表中获取伪随机地选择的转换表，以及 / 或者 (c) 针对所述输入符号中的每一个，通过使用其对应的转换表将所述输入符号转换为相对应的输出符号，以便逐个地对每个输入符号进行加密。所述加密设备还可以包括用于以对应于所述输出符号的 DTMF 音调向所述安全服务器传输所述输出符号的输出接口。

[0010] 所述多个输入符号是由 N 个符号的集合定义的，其中，N 为正整数，并且转换表是所述 N 个符号的排列。所述加密设备还包括：(a) 耦合到所述处理电路的密钥流生成器，该密钥流生成器用于针对待加密的第一输入符号从所述密钥流生成器处获取第一伪随机数，以及 / 或者 (b) 耦合到所述处理电路的转换表生成器，该转换表生成器用于混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的转换表。

[0011] 在一个实例中，所述处理电路还用于：(a) 针对所述第一输入符号生成伪随机数，其中，该伪随机数为 k 位长，并且 k 为正整数，(b) 判断所述伪随机数是否在最大数 Pmax 之内，其中，Pmax 是小于最大阈值 $2^k$ 的 N 阶乘的最大倍数，(c) 如果所述伪随机数大于所述最大数 Pmax，则将所述伪随机数丢弃，(d) 针对所述第一输入符号获取不同的伪随机数，直到获取到可接受的伪随机数为止，其中，所述可接受的伪随机数小于或等于最大数 Pmax，以及 / 或者 (e) 求所述可接受的伪随机数除以 N 阶乘后的余数，以获取所述第一伪随机数。

[0012] 在一个配置中，所述加密设备的处理电路还用于：(a) 针对待加密的所述输入符号中的每一个，从定义了不同的符号 - 符号排列的所述多个转换表中获取伪随机地选择的第二转换表，以及 / 或者 (b) 针对所述输出符号中的每一个，通过使用其对应的第二转换表

将所述输出符号转换为相对应的第二输出符号，以便逐个地对每个输入符号进行进一步加密。

[0013] 因此，提供了一种加密设备，包括：(a) 用于通过对从所述电话接收的 DTMF 音调进行转变，来获取多个输入符号的模块，(b) 用于针对待加密的所述输入符号中的每一个而从定义了不同的符号 - 符号排列的多个转换表中获取伪随机地选择的转换表的模块，(c) 用于针对所述输入符号中的每一个而通过使用其对应的转换表将所述输入符号转换为相对应的输出符号以便逐个地对每个输入符号进行加密的模块，以及 / 或者 (d) 用于以对应于所述输出符号的 DTMF 音调向所述安全服务器传输所述输出符号的模块。所述加密设备还包括：(a) 用于针对待加密的所述输入符号中的每一个从定义了不同的符号 - 符号排列的所述多个转换表中获取伪随机地选择的第二转换表的模块，以及 / 或者 (b) 用于针对所述输出符号中的每一个而通过使用其对应的第二转换表将所述输出符号转换为相对应的第二输出符号以便逐个地对每个输入符号进行进一步加密的模块。

[0014] 在一个配置中，加密设备包括：(a) 用于针对第一输入符号获取伪随机地选择的第一转换表的模块，(b) 用于针对第二输入符号获取伪随机地选择的第二转换表的模块，(c) 用于通过使用所述第一转换表将所述第一输入符号转换为第一输出符号的模块，以及 / 或者 (d) 用于通过使用所述第二转换表将所述第二输入符号转换为第二输出符号的模块。

[0015] 在一个实例中，加密设备还包括 (a) 用于针对待加密的第一输入符号获取第一伪随机数的模块，以及 / 或者 (b) 用于混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的转换表的模块。

[0016] 提供了一种机器可读介质，其具有一个或多个可用于对符号进行加密的指令，当处理器执行所述指令时，所述指令使得所述处理器执行以下操作：(a) 获取多个输入符号，(b) 针对待加密的所述输入符号中的每一个，从定义了不同的符号 - 符号排列的多个转换表中获取伪随机地选择的转换表，以及 / 或者 (c) 针对所述输入符号中的每一个，通过使用其对应的转换表将所述输入符号转换为相对应的输出符号，以便逐个地对每个输入符号进行加密。所述多个输入符号是由 N 个符号的集合定义的，其中，N 为正整数，并且转换表是所述 N 个符号的排列。

[0017] 所述机器可读介质还包括一个或多个指令，当处理器执行所述指令时，所述指令使得所述处理器进一步执行以下操作：(a) 针对待加密的第一输入符号获取第一伪随机数，以及 / 或者 (b) 混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的转换表。

[0018] 在一个配置中，所述机器可读介质还包括一个或多个指令，当处理器执行所述指令时，所述指令使得所述处理器进一步执行以下操作：(a) 针对所述第一输入符号生成伪随机数，其中，该伪随机数为 k 位长，并且 k 为正整数，(b) 判断所述伪随机数是否在最大数 Pmax 之内，其中，Pmax 是小于最大阈值 $2^k$ 的 N 阶乘的最大倍数，(c) 如果所述伪随机数大于所述最大数 Pmax，则将所述伪随机数丢弃，(d) 针对所述第一输入符号获取不同的伪随机数，直到获取到可接受的伪随机数为止，其中，所述可接受的伪随机数小于或等于最大数 Pmax，以及 / 或者 (e) 求所述可接受的伪随机数除以 N 阶乘后的余数，以获取所述第一伪随机数。

[0019] 提供了一种在解密设备上运行的、用于保护在电话和安全服务器之间的通信的方法。接收来自于加密设备的对应于输入符号的 DTMF 音调。根据所述 DTMF 音调获取多个所述输入符号。针对待解密的所述输入符号中的每一个，从定义了不同的符号 - 符号排列的多个反转换表中获取伪随机地选择的反转换表。然后，针对所述输入符号中的每一个，通过使用其对应的反转换表将所述输入符号转换为相对应的输出符号，以便逐个地对每个输入符号进行解密。多个所述输入符号是由 N 个符号的集合定义的，其中，N 为正整数，并且反转换表是所述 N 个符号的排列。该方法还包括 :(a) 针对待解密的第一输入符号获取第一伪随机数，以及 / 或者 (b) 混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的反转换表。

[0020] 在一个实例中，通过如下操作来获取所述第一伪随机数 :(a) 针对所述第一输入符号生成伪随机数，其中，该伪随机数为 k 位长，并且 k 为正整数，(b) 判断所述伪随机数是否在最大数 Pmax 之内，其中，Pmax 是小于最大阈值 $2^k$ 的 N 阶乘的最大倍数，(c) 如果所述伪随机数大于所述最大数 Pmax，则将所述伪随机数丢弃，(d) 针对所述第一输入符号获取不同的伪随机数，直到获取到可接受的伪随机数为止，其中，所述可接受的伪随机数小于或等于最大数 Pmax，以及 / 或者 (e) 求所述可接受的伪随机数除以 N 阶乘后的余数，以获取所述第一伪随机数。

[0021] 在另一个配置中，所述方法还包括 :(a) 针对待解密的所述输入符号中的每一个，从定义了不同的符号 - 符号排列的所述多个反转换表中获取伪随机地选择的第二反转换表，以及 / 或者 (b) 针对所述输出符号中的每一个，通过使用其对应的第二反转换表将所述输出符号转换为相对应的第二输出符号，以便逐个地对每个输入符号进行进一步解密。

[0022] 提供了一种包括输入接口和处理电路的、用于保护在电话和安全服务器之间的通信的解密设备。该输入接口可以用于接收来自于加密设备的对应于输入符号的 DTMF 音调。该处理电路用于 :(a) 根据所述 DTMF 音调获取多个所述输入符号，(b) 针对待解密的所述输入符号中的每一个，从定义了不同的符号 - 符号排列的多个反转换表中获取伪随机地选择的反转换表，以及 / 或者 (c) 针对所述输入符号中的每一个，通过使用其对应的反转换表将所述输入符号转换为相对应的输出符号，以便逐个地对每个输入符号进行解密。

[0023] 多个所述输入符号是由 N 个符号的集合定义的，其中，N 为正整数，并且反转换表是所述 N 个符号的排列。该解密设备还包括 :(a) 耦合到所述处理电路的密钥流生成器，该密钥流生成器用于针对待解密的第一输入符号从所述密钥流生成器处获取第一伪随机数，以及 / 或者 (b) 耦合到所述处理电路的反转换表生成器，该反转换表生成器用于混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的反转换表。

[0024] 多个所述输入符号是由 N 个符号的集合定义的，其中，N 为正整数，并且反转换表是所述 N 个符号的排列。该处理电路还用于 :(a) 针对所述第一输入符号生成伪随机数，其中，该伪随机数为 k 位长，并且 k 为正整数，(b) 判断所述伪随机数是否在最大数 Pmax 之内，其中，Pmax 是小于最大阈值 $2^k$ 的 N 阶乘的最大倍数，(c) 如果所述伪随机数大于所述最大数 Pmax，则将所述伪随机数丢弃，(d) 针对所述第一输入符号获取不同的伪随机数，直到获取到可接受的伪随机数为止，其中，所述可接受的伪随机数小于或等于所述最大数 Pmax，以及 / 或者 (e) 求所述可接受的伪随机数除以 N 阶乘后的余数，以获取所述第一伪随机数。

[0025] 在另一个实例中，所述处理电路还用于：(a) 针对待解密的所述输入符号中的每一个，从定义了不同的符号－符号排列的所述多个反转换表中获取伪随机地选择的第二反转换表，以及／或者 (b) 针对所述输出符号中的每一个，通过使用其对应的第二反转换表将所述输出符号转换为相对应的第二输出符号，以便逐个地对每个输入符号进行进一步解密。

[0026] 因此，提供了一种用于保护在电话和安全服务器之间的通信的解密设备，包括：(a) 用于接收来自于加密设备的对应于输入符号的 DTMF 音调的模块；(b) 用于根据所述 DTMF 音调获取多个所述输入符号的模块，(c) 用于针对待解密的所述输入符号中的每一个而从定义了不同的符号－符号排列的多个反转换表中获取伪随机地选择的反转换表的模块，以及／或者 (d) 用于针对所述输入符号中的每个而通过使用其对应的反转换表将所述输入符号转换为相对应的输出符号以便逐个地对每个输入符号进行解密的模块。多个所述输入符号是由 N 个符号的集合定义的，其中，N 为正整数，并且反转换表是所述 N 个符号的排列。该解密设备还包括：(a) 用于针对待解密的第一输入符号获取第一伪随机数的模块，以及／或者 (b) 用于混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的反转换表的模块。通过如下操作来获取所述第一伪随机数：(a) 用于针对所述第一输入符号生成伪随机数的模块，其中，该伪随机数为 k 位长，并且 k 为正整数，(b) 用于判断所述伪随机数是否在最大数 Pmax 之内的模块，其中，Pmax 是小于最大阈值 $2^k$ 的 N 阶乘的最大倍数，(c) 用于如果所述伪随机数大于所述最大数 Pmax 则将所述伪随机数丢弃的模块，(d) 用于针对所述第一输入符号获取不同的伪随机数直到获取到可接受的伪随机数为止的模块，其中，所述可接受的伪随机数小于或等于最大数 Pmax，以及／或者 (e) 用于求所述可接受的伪随机数除以 N 阶乘后的余数以获取所述第一伪随机数的模块。

[0027] 提供了一种机器可读介质，其具有一个或多个用于对符号进行解密的指令，当处理器执行所述指令时，所述指令使得所述处理器执行以下操作：(a) 获取多个被定义在 n 个符号的集合之内的输入符号，(b) 针对待解密的所述输入符号中的每一个，从定义了不同的符号－符号排列的多个反转换表中获取伪随机地选择的反转换表，以及／或者 (c) 针对所述输入符号中的每一个，通过使用其对应的反转换表将所述输入符号转换为相对应的输出符号，以便逐个地对每个输入符号进行解密。所述多个输入符号是由 N 个符号的集合定义的，其中，N 为正整数，并且反转换表是所述 N 个符号的排列。

[0028] 所述机器可读介质还包括一个或多个指令，当处理器执行所述指令时，所述指令使得所述处理器进一步执行以下操作：(a) 针对待解密的第一输入符号获取第一伪随机数，以及／或者 (b) 混洗所述 N 个符号的集合以获取所述 N 个符号的集合的不同排列并使用该排列作为所述第一输入符号的反转换表。

[0029] 在另一个实例中，所述机器可读介质还包括一个或多个指令，当处理器执行所述指令时，所述指令使得所述处理器进一步执行以下操作：(a) 针对待解密的所述输入符号中的每一个，从定义了不同的符号－符号排列的所述多个反转换表中获取伪随机地选择的第二反转换表，以及／或者 (b) 针对所述输出符号中的每一个，通过使用其对应的第二反转换表将所述输出符号转换为相对应的第二输出符号，以便逐个地对每个输入符号进行进一步解密。

## 附图说明

[0030] 图 1 示出了可以将安全设备沿着通信线路耦合到电话以保护电话和安全服务器之间的特定通信的系统；

[0031] 图 2 示出了用于保护电话和安全服务器（属于图 1 的发放机构 (issuing institution)）之间的特定通信的方法的流程图；

[0032] 图 3 示出了能够在传输期间保护 DTMF 音调的电话服务安全服务器的一个实例的方框图；

[0033] 图 4 示出了在安全服务器上运行的用于保护来自电话设备的 DTMF 音调的方法；

[0034] 图 5 示出了可用于在传输期间保护 DTMF 音调的安全设备的一个实例的方框图；

[0035] 图 6 示出了在安全设备上运行的用于保护来自电话设备的 DTMF 音调的方法；

[0036] 图 7 是用安全服务器来对自身进行认证的移动通信设备的方框图；

[0037] 图 8 示出了用于对通过通信网络到电话服务站的移动通信设备进行认证的方法的流程图；

[0038] 图 9 示出了通过伪随机地选择用于每个待加密的符号的转换表来保护明文符号的组合合成器的方框图；

[0039] 图 10 示出了用于将明文符号转换成经加密的符号的符号 - 符号转换表的实例；

[0040] 图 11 示出了如何使用不同的转换表来对明文符号进行加密以获得经加密的符号的一个实例；

[0041] 图 12 示出了从用于 n 个符号的集合的多个可能排列中选择转换表的算法，其中 n 是正整数；

[0042] 图 13 示出了可以通过使用多个转换表对单个明文符号进行加密来实现符号认证的另一个加密方案的方框图；

[0043] 图 14 示出了如何使用多个转换表来对每个明文符号进行加密以获得对应的经加密的符号；

[0044] 图 15 示出了如何采用两个转换表来将明文符号转换或者加密成经加密的符号的实例；

[0045] 图 16 示出了根据一个实例来执行明文加密的方法；

[0046] 图 17 示出了如何通过使用一个或多个反转换表来对经加密的符号进行解密以获得单个明文符号的方框图；

[0047] 图 18 示出了根据一个实例来执行明文加密的方法；

[0048] 图 19 示出了根据一个实例的加密模块的方框图；

[0049] 图 20 示出了根据一个实例的解密模块的方框图。

## 具体实施方式

[0050] 在以下说明中给出了具体细节，以提供对实例的彻底理解。然而，本领域的普通技术人员应该理解，也可以不用这些具体细节来实现实例。例如，在方框图中可以不示出电路，以免不必要的细节模糊了实例。

[0051] 并且应当注意到，可以将实例描述为被描绘成流程图、流图、结构图或者方框图的

过程。尽管流程图可以将操作描述为顺序的过程，但是许多操作可以被并行地或同时地执行。另外，可以重新排列操作的顺序。当过程的操作完成时，该过程结束。过程可以对应于方法、函数、手续、子例程、子程序等等。当过程对应于函数时，它的结束对应于函数返回到调用函数或主函数。

[0052] 此外，存储介质可以表示用于存储数据的一个或多个设备，包括只读存储器(ROM)、随机访问存储器(RAM)、磁盘存储介质、光存储介质、闪速存储介质和/或用于存储信息的其它机器可读介质。术语“机器可读介质”包括但不限于：便携式的或固定的存储设备、光存储设备、无线信道和能够存储、包含或携带指令和/或数据的各种其它介质。

[0053] 此外，可以通过硬件、软件、固件、中间件、微码或它们的组合来实现各种配置。当用软件、固件、中间件或微码来实现时，可以把用于执行本文所描述的任务的程序代码或代码片段存储在诸如存储介质或其它存储模块之类的机器可读介质中。处理器可以执行已定义的任务。代码片段可以表示手续、函数、子程序、程序、例程、子例程、模块、软件包、类、指令的组合、数据结构或程序声明。可以通过传递和/或接收信息、数据、变元、参数或存储器内容来将代码片段耦合到另一个代码片段或硬件电路。可以经由包括了存储器共享、消息传递、令牌传递和网络传输等等的适当的模块来传递、转发或传输信息、数据、变元、参数、数据等。可以用硬件和/或软件来实现本文所公开的方法。

[0054] 可以用通用处理器、数字信号处理器(DSP)、专用集成电路(ASIC)、现场可编程门阵列(FPGA)或其它可编程逻辑组件、分立门或晶体管逻辑、分立硬件组件或可用来执行本文所描述的功能的其任意组合来实现或执行结合本文所公开的实例而描述的各种示例性的逻辑块、模块、电路、元件和/或组件。通用处理器可以是微处理器，替代地，该处理器也可以是任何常规的处理器、控制器、微控制器或状态机。处理器也可以实现为计算组件的组合，例如，DSP和微处理器、多个微处理器、结合DSP内核的一个或多个微处理器或者任何其它这样的配置的组合。

[0055] 结合本文所公开的实例而描述的方法或算法可直接通过硬件、由处理器所执行的软件模块或二者的组合来实现，其还可通过处理单元、编程指令或其它指令的形式来实现，并且，其还可被包含在单个设备或分布在多个设备中。软件模块可以位于RAM存储器、闪速存储器、ROM存储器、EPROM存储器、EEPROM存储器、寄存器、硬盘、移动磁盘、CD-ROM或者本领域公知的任何其它形式的存储介质中。存储介质可被耦合至处理器，从而使处理器能够从该存储介质中读取信息，且能够向该存储介质写入信息。替代地，存储介质可以是处理器的组成部分。

[0056] 一个特征提供了一种小型安全设备，可将该小型安全设备串联插入到客户的电话线中，该安全设备充当双因子认证方案中的第二因子并且对DTMF音调进行加密，从而防止了敏感信息的公开。该设备不会干扰电话的正常操作。该设备包括小形状因子的外壳，该外壳还可为银行以及与银行相关的支付服务提供标注品牌的机会。可以从该设备所耦合的电话线来为该设备供电。在一个配置中，多个这样的设备可以沿电话线连成串或级联，以提供与多个不同方（例如银行）的安全通信。

[0057] 另一个特征提供了一种对加密符号的安全进行保护的有效加密方法。通过使用单独的伪随机地选择的转换表来对每个明文符号进行加密。该转换表是基于伪随机数和符号混洗算法在运行中高效生成的，而非将符号的每种可能的排列都预先存储为转换表。类似

地，接收设备可在运行中生成反转换表，以便对接收到的经加密的符号进行解密。

[0058] 保护 DTMF 音调

[0059] 图 1 示出了系统，其中，安全设备 102 沿着通信线路耦合到电话 104，以保护在电话 104 和安全服务器 108 之间的特定通信。安全设备 102 可以是与电话 104 和通信网络 106 之间的电话线连成一线或串联的小型设备。安全设备 102 可以耦合到接近或邻近电话 104 的电话线。

[0060] 在一个实例中，安全设备 102 可以与账户和 / 或发放机构 108（例如，银行、信用卡公司等等）相关联。例如，银行可以向它的客户发放这样的安全设备 102，每个安全设备与客户或客户的账户唯一地相关联。发放机构 108 可以具有安全服务器 110，其有助于与客户的电话交易。

[0061] 图 2 是流程图，其示出了用于保护在电话 104 和属于图 1 的发放机构 108 的安全服务器 110 之间的特定通信的方法。安全设备 102 可以具有活动操作模式和不活动（被动）操作模式。当使用安全设备 102 呼叫除发放机构 108（例如安全服务器 110）之外的某个人时，安全设备 102 是不活动的，并且呼叫仅不经改变地通过安全设备 102，DTMF 音调也是如此。然而，当电话 104 向发放机构发起呼叫时 208，安全服务器 110（例如安全服务器 110）会发送用来唤醒安全设备的激活信号 210。激活信号可以足够长并且 / 或者是唯一的（例如具有足够多的数字或符号），从而能够合理地确保安全设备 102 不会被意外触发。在一个实例中，该激活信号实际上可以不携带任何信息，而仅仅去触发或激活安全设备 102 以便从不活动（被动）模式切换到活动模式。例如，激活信号可以是由安全设备 102 所识别的一小段音乐或音调。安全设备 102 侦听并识别来自安全服务器 110 的激活信号（例如，一组独特的 DTMF 音调），并且改变到活动模式 212。在一个实例中，一旦接收到激活信号，安全设备 102 可以开始对从电话到安全服务器 110 的全部 DTMF 音调进行加密。

[0062] 在一个配置中，可以在安全设备 102 和安全服务器 110 之间实施查询响应 (challenge-response) 方案。除了发送激活信号之外，安全服务器 110 还可以向安全设备发送随机查询 214。安全设备 102 接收查询，生成应答（例如，标识符和对查询的响应）216，并且向安全服务器 110 发送应答。该应答可以包括与安全设备 102 相关联的标识符以及对查询的响应。安全设备 102 还可以生成会话密钥，该会话密钥可用于对从电话 104 到安全服务器 110 的后续 DTMF 音调进行加密。

[0063] 该应答通知安全服务器 110 其正在与相关联的安全设备进行通信。安全服务器 110 可以使用标识符来查找特定客户的账户 220，从而省去了客户手动对其自身进行识别的麻烦（例如，免去了客户输入他们的账号）。安全服务器 110 还可基于随机查询以及在安全设备 102 和安全服务器 110 两者中都供应的认证密钥来验证该响应是正确的 222，以便对用户进行认证。应当注意到：由于安全设备 102 位于用户的电话的附近（例如，在用户的家中），所以攻击者必需窃取安全设备 102 才能够发起攻击。

[0064] 通过使用相同的查询和响应，安全服务器 110 对与安全设备 102 所计算的会话密钥相同的会话密钥 226 进行计算 224。如果安全服务器 110 不同意其从安全设备 102 接收到的应答（或者根本没有接收到响应），那么呼叫将被转接到用于进行更严格的识别和 / 或认证的替代路径。也就是说，安全设备 102 可以基于接收到的随机查询和认证密钥来计算它的响应。然后，安全服务器 110 可以通过（基于随机查询和认证密钥）计算本地响应来

验证接收到的响应，并且将其与从安全设备 102 接收到的响应进行比较。

[0065] 如果正确地认证了查询 - 响应，那么安全服务器 110 发送确认，其中，已使用最新获得的会话密钥对该确认进行了认证 228。该确认会通知安全设备 102 开始对来自电话 104 的 DTMF 音调进行加密。如果来自安全服务器的确认有问题（例如，它不是在特定最大时间内由安全设备 102 接收的，或者确认失败等等），那么安全设备 102 可以向用户生成警告信号。例如，如果查询响应认证失败，则灯闪烁（或者点亮）或者响起报警。另外，灯（例如，发光二极管——LED）可以发光以指示安全设备 102 是活动的和 / 或查询 - 响应被成功地认证。

[0066] 在查询 - 响应被成功地认证之后，在一个实例中，安全设备 102 可以使用会话密钥来对从电话 104 到安全服务器的传输进行加密。一旦加密开始，安全设备会截取来自电话的 DTMF 音调 232，并且改为传输经加密的 DTMF 音调 234。在 DTMF 音调的这种加密的一个实例中，可以把来自电话 104 的 DTMF 音调转换成将随后发送到安全服务器 110 的不同的 DTMF 音调。在另一个配置中，安全设备 102 可以将 DTMF 音调转变成数字符号，然后，数字符号被加密并且被发送到安全服务器 110。安全设备 102 还可不经修改或加密地在两个方向上传递其它对象（非 DTMF 音调或符号）。因为要求用户首先去做的一件事就是输入与其账户相关联的 PIN 号码，所以与该 PIN 号码相关联的 DTMF 音调会被加密并且形成用于认证的第二因子。类似地，安全服务器 110 可以使用会话密钥来对经由安全设备从电话接收到的 DTMF 音调进行解密 236。

[0067] 在替代的配置中，可以将安全设备 102 配置为识别与特定发放机构 108 相关联的特定电话号码。当安全设备 102 识别出电话已拨打了特定电话号码时，它可以自动地切换到活动模式和 / 或对从电话到安全服务器 110 的全部 DTMF 音调进行加密。

[0068] 安全设备 102 可以继续对来自电话 104 的 DTMF 音调进行加密，直到呼叫终止为止，当呼叫终止时，安全设备 102 将切换回不活动模式，在不活动模式中允许全部 DTMF 音调不经改变地通过。

[0069] 因为安全设备 102 可以具有小的形状因子并且易于插入到电话线中。用户可以具有与单个机构或账户相关联的多个安全设备，以使得用户能够从不同的地点（例如家、办公室等）安全地访问账户。用户还可以具有与各个不同的机构和 / 或账户相关联的多个安全设备。这些多个安全设备可以沿电话线串联地耦合。连成一串的不活动安全设备仅向该连成一串的不活动安全设备中的下一个安全设备传送信号。如果安全服务器激活了该串中的安全设备，那么它会对来自电话的 DTMF 音调进行加密。

[0070] 在另一个实例中，安全设备 102 可以为来自一个电话或地点的多个用户提供服务。在该情况下，安全服务器可以识别出该安全设备是与多个用户或账户相关联的。为了区分开每个用户，安全服务器可以发送语音提示以要求用户输入 PIN 或者输入可识别特定用户或账户的其它标识符。

[0071] 图 3 示出了允许在传输期间保护 DTMF 音调的电话服务安全服务器的一个实例的方框图。安全服务器 302 可以包括处理电路 304，例如小型和 / 或低功率微处理器。安全服务器 302 可以包括用于将安全服务器 302 耦合到通信网络的第一通信模块 306。认证模块 308 允许安全服务器 302 对与其通信的安全设备进行认证。DTMF 解密模块 308 允许安全服务器 302 对从安全设备接收到的经加密的 DTMF 音调进行解密。

[0072] 图 4 示出了可在安全服务器上的运行的用于保护来自电话设备的 DTMF 音调的方法。从启用 DTMF 的电话接收呼叫 402。向与启用 DTMF 的电话相关联的安全设备发送激活信号 404。安全设备位于启用 DTMF 的电话的附近。之后，由安全服务器来认证安全设备。例如，向安全设备发送查询信号 406。安全服务器确定是否从安全设备处接收到了响应 408。如果否，那么可以假设在电话线上不存在安全设备 410。否则，安全服务器确定是否可以成功地认证接收到的响应 412。如果不能成功地认证接收到的响应，那么认证失败 414。否则，生成会话密钥 416。向安全设备发送经会话密钥认证的确认消息 418。安全服务器可以从安全设备处接收经加密的 DTMF 音调 420。安全服务器然后可以对接收到的 DTMF 音调进行解密以获得由电话所发送的信息 422。这种 DTMF 音调可以表示在传输期间通过对原始 DTMF 音调进行加密来防范窃听器的机密信息（例如，账号、口令、PIN 等等）。

[0073] 图 5 示出了可用于在传输期间保护 DTMF 音调的安全设备的一个实例的方框图。安全设备 502 可以包括处理电路 504，例如小型和 / 或低功率微处理器。可以通过与安全设备 502 耦合的电话线来为安全设备 502 供电。第一通信接口 A506 可被用来将安全设备 502 耦合到电话。第二通信接口 B508 可被用来将安全设备 502 耦合到通信网络。在被动操作模式中，安全设备 502 使全部 DTMF 音调都未经改变地通过。处理电路 504 可被配置为侦听（例如，来自安全服务器的）激活信号。DTMF 检测器 510 可被配置为检测 DTMF 激活信号以将安全设备切换到活动操作模式。在活动操作模式中，安全设备 502 可被配置为对来自安全服务器的认证查询进行响应。

[0074] 在激活模式中，DTMF 检测器 510 还可被配置为检测经由通信接口 A506 接收到的（例如，来自电话的）DTMF 音调。如果检测到一个或多个 DTMF 音调，那么对 DTMF 音调进行加密或者通过 DTMF 加密模块 512 对 DTMF 音调进行修改。然后经过通信接口 B508 将加密的 DTMF 音调传输到安全服务器。

[0075] 图 6 示出了在安全设备上运行的用于保护来自电话设备的 DTMF 音调的方法。一旦在电话和安全服务器之间发起了呼叫，就给安全设备进行供电 602。也就是说，因为当进行呼叫时会对通信线路施加电压，所以安全设备可以从通信线路获得电力。在被动操作模式中，安全设备允许 DTMF 音调在第一通信接口和第二通信接口之间未经改变地通过 604。例如，第一通信接口可以耦合到电话，并且第二通信接口可以耦合到第二通信接口。安全设备监视传输以确定是否从安全服务器接收到 (DTMF) 激活信号 606。除非接收到激活信号，否则安全设备继续运行在被动模式。如果接收到 DTMF 激活信号，那么安全设备改变成活动操作模式 608。安全设备还可以侦听来自安全服务器的其它信号 610。

[0076] 安全设备可以从安全服务器接收查询 612。安全设备利用对查询的响应来进行应答 614。如果该响应有效，那么安全设备可以接收到表明安全服务器已成功地认证了该安全设备的确认 616。

[0077] 一旦激活并且正确地认证了安全设备，安全设备将侦听来自电话的 DTMF 音调。如果 DTMF 音调是通过第一通信接口从（与安全设备耦合的）电话接收的 618，那么将接收到的 DTMF 音调加密成不同的 DTMF 音调 620。在一个实例中，将来自电话的 DTMF 音调转换成随后被发送到安全服务器的不同 DTMF 音调。在另一个配置中，安全设备 102 将 DTMF 音调转变成数字符号，然后，将数字符号加密并且发送到安全服务器。接下来，将经加密的 DTMF 音调通过第二通信接口发送给安全服务器 622。安全设备继续对来自电话的 DTMF 音调进行

加密，直到呼叫结束为止，在呼叫结束时，安全设备返回到被动模式624。安全设备102防止来自电话的未加密的DTMF音调被传送到安全服务器。在一个实例中，安全设备102可以断开从电话到网络的全部输入（例如传输），同时保持活动。在该情况下，例如，如果客户需要与代表谈话，那么可以存在一些使客户或安全服务器能够重新连接到输入（例如，允许来自安全设备102的传输）的准备措施。

[0078] 蜂窝电话安全方案

[0079] 图7是移动通信设备的方框图，该移动通信设备被配置为利用安全服务器对其自身进行认证。移动通信设备702包括耦合到通信模块706和用户输入接口708的处理电路704。通信模块706使移动通信设备702能够通过无线通信网络710进行通信。处理电路704可以被配置为在呼叫期间利用一个或多个安全服务器来对其自身进行认证。例如，移动通信设备可以配置为具有可允许银行或金融机构对移动通信设备702的用户进行认证的认证密钥和/或用户标识符。银行或金融机构可以预先（例如，在建立或配置期间）提供认证密钥和/或用户标识符。另外，处理电路704还可以向用户请求PIN、口令和/或其它输入以完成认证程序。

[0080] 图8是流程图，其示出了用于对通过通信网络到电话服务站804的移动通信设备802进行认证的方法。移动通信设备802可以是移动电话，并且电话服务站804可以包括与银行或金融机构相关联的安全服务器。移动通信设备802和电话服务站804每个具有相同的认证密钥。

[0081] 移动通信设备可以向与电话服务站相关联的发放机构806发起呼叫。例如，发放机构可以是银行或金融机构。电话服务站向移动通信设备发送随机认证查询808。然后，移动通信设备基于随机查询和认证密钥生成响应809，并且将响应以及（可能的）用户标识符发送到电话服务站810。然后，电话服务站验证来自移动通信设备的响应是否正确812。可以由电话服务站基于其认证密钥和随机认证查询来对验证值进行计算并且将其与从移动通信设备接收到的响应进行比较来完成这一操作。如果成功地认证了响应，那么可以向移动通信设备发送认证确认814。移动通信设备可以向电话服务站请求敏感信息（例如，银行账户记录等等）816。如果成功地认证了移动通信设备，那么之后电话服务站向移动通信设备提供所请求的敏感信息818。这样，可以由电话服务站来认证移动通信设备（例如，移动电话），以在呼叫期间保护敏感信息的传输。

[0082] 威胁模型

[0083] 本文所描述的安全设备和/或方法所能解决的一类威胁是窃听攻击。在这种攻击中，攻击者可以将录音机连接到电话线上以侦听与用户在电话上输入的号码相关联的DTMF音调。这些DTMF音调可以标识正被呼叫的银行、用户的客户号码和/或账户号码、个人识别号码(PIN)、社会保险号以及其它个人和/或机密信息。然后，攻击者可以使用该信息来根据用户的账户执行诈骗交易。本文所描述的安全设备通过对DTMF音调进行加密并且提供进一步的认证来挫败这种攻击。因为大多数机构（例如，银行等等）可以使用两个因子来进行认证（例如，对安全设备的拥有和对PIN的知晓），所以其极少去要求其它敏感信息。仅仅截取经加密的DTMF音调将不会暴露有关于账号、PIN等等的任何信息。

[0084] 为了取得成功，攻击者必须干扰呼叫的过程，例如可以通过阻止呼叫到达期望的接收方（例如，期望的银行）的方式，伪装成期望的接收方，要求主叫方输入全部敏感信息。

为了挫败这种攻击，安全设备可以在从接收机构接收到“开始加密”信号（即，认证过的确认）之后开启安全指示器（例如灯）。主叫方（例如客户）仅需检查安全指示器，以确保在输入任何敏感或机密信息前安全设备已经在对它的音调进行加密。

[0085] 另一种类型的攻击是会话劫持攻击，在这种会话劫持攻击中，攻击者一直等待，直到用户已经建立了与达期望的接收方（例如，银行）的通信，从而激活了安全指示器之后，然后攻击者接管呼叫。然后，攻击者假装呼叫出错并且要求用户口头提供敏感信息。可替代地，攻击者可以要求用户输入（攻击者已知的）具体响应以便尝试建立逐个音调的加密模式，并且然后，使用该逐个音调的变换来对他们自己对银行的响应进行加密。为了解决这种类型的攻击，可以伪随机地、循环地和 / 或以抑制发现号码与音调的关系的方式来更改或修改这种逐个音调的加密。

[0086] 消息和会话认证

[0087] 可以将安全设备配置为通过使用例如消息认证码 (MAC) 函数来执行消息认证和会话密钥导出。例如，安全服务器可以通过对 $MAC_K$（查询）的单次调用的输出进行划分，来认证主叫方的安全设备。例如，典型的 MAC 函数可以返回 128 比特的输出，可以将其表示为 32 个 DTMF 音调。在安全服务器和安全设备已经计算出了 MAC 之后，安全服务器可以向安全设备发送前 16 个 DTMF 音调（其表示 MAC 的一部分），并且作为响应，安全设备发送回其它 16 个 DTMF 音调（其表示 MAC 的另一部分）。这样，安全服务器和安全设备两者可以彼此证明它们是认证过的或合法的。

[0088] 类似地，每一方都可以计算会话密钥以使得会话密钥＝ $MAC_K$（认证密钥 ‖ 查询），其中，认证密钥是预先加载到安全设备中的。为了防止当安全设备向安全服务器发送它的响应时泄漏会话密钥，响应可以包括附加信息。例如，响应可以是 :响应＝ $MAC_K$（“额外信息字符串”‖ 认证密钥 ‖ 查询）。

[0089] 流加密

[0090] 另一个特征提供了一种用于防护经加密的符号的安全的有效加密方法。通过使用单独的伪随机地选择的转换表来对每个明文符号进行加密。该转换表是基于伪随机数和符号混洗算法在运行中高效地生成的，而非将符号的每种可能的排列都预先存储为转换表。类似地，接收设备可在运行中生成反转换表，以便对接收到的经加密的符号进行解密。

[0091] 可以利用各种配置来实现该加密方法。例如，电话安全设备可以将 DTMF 音调转变成数字值，通过对每个数字值使用伪随机地选择的转换表来对数字值进行加密。然后，可以以数字形式或者作为与经加密的数字值相关的 DTMF 音调向安全服务器（例如电话服务站）传输经加密的数字值。

[0092] 因为 DTMF 音调由数字符号表示（或者与其相关），所以可以通过例如流加密来保护 DTMF 音调。在各个实例中，流加密可以使用诸如计数器模式的高级加密标准 (AES)、输出反馈 (OFB) 或者密本反馈 (CFB) 模式之类的分组密码所生成的密钥流。例如，可以使用 CBC-MAC 模式的分组密码来实现 MAC 函数。这可以是有利的，如果例如安全设备具有在硬件中实现的 AES 的话。

[0093] 如果利用软件实现这些函数，那么可以优选地使用专用流密码（例如非线性 SOBER(NLS)）。通过将待加密的数据用作密钥或随机数输入并且然后生成输出密钥流，从而可以将流密码用作 MAC 函数，虽然这样效率很低。由于所生成的密钥流的长度可以如所期

望的那样长，因此，可以在单次呼叫中生成响应和会话密钥两者。

[0094] 常规的流加密通常如下进行（不管使用真实流密码还是流化模式的分组密码）：生成伪随机数的密钥流并且将它们与明文（即，DTMF 音调的数字表示）进行组合以形成经加密的输出或密本。一般，使用异或 (XOR) 运算来组合密钥流和明文，这是因为它是自反的。然而，常规的启用 DTMF 的电话具有 10 个按键或更多个按键，每个按键具有唯一的音调。因此，不能使用 XOR 运算来利用密钥流对 DTMF 音调进行加密。可以改为将与电话按键相关联的 DTMF 音调转变成（关联到）不同的数字符号，可以将所述数字符号添加到从密钥流获得的伪随机数 / 符号中，以便生成加密的符号和密本。但是知道特定数字的位置的活动攻击者能够通过从所传输的密本减去数来改变该数。例如，攻击者知道对于特定 DTMF 音调而言，输入为“1”但是输出为“7”，则攻击者可以确定为该音调所生成的伪随机数是“6”，并且然后可以正确地对其为该特定数字位置所选择的任何字符进行加密。

[0095] 组合合成器

[0096] 一个特征提供了使用密钥流来获得或生成用于每个待加密的明文符号的伪随机选择或生成的转换表。不是从密钥流中得到伪随机数并且以相同的方式（例如，通过加上 mod(n)）改变明文，而是一个特征提供通过伪随机地选择多个转换表中的一个来转换输入流中的每个明文符号。转换表可以提供一组数或符号的不同的可能排列。这里将其称为组合合成器。

[0097] 图 9 示出了通过伪随机地选择用于每个待加密的符号的转换表来保护明文符号的组合合成器的方框图。使用密码生成器 902 来生成伪随机数 / 符号的密钥流 Si 904。对于输入流中的每个明文输入符号 Pi 908，使用伪随机数的密钥流 904 来根据多个可能的转换表生成或者获得不同的转换表 906。通过将明文输入符号 908 转换成伪随机输出，生成经加密的输出符号 Ci 910。

[0098] 这种转换操作定义在密钥流 904 的控制下的明文输入符号 908 的排列。可以将转换表 906 表示为 n 个元素的向量，并且可以通过查找转换表 906 的第 p 个元素来完成明文输入符号 908 的转换。给定经加密的输出符号 Ci，可以通过创建反排列表或者通过针对包含符号 Ci 的条目搜索表并且返回其索引作为 p 来完成反转换。

[0099] 通常来说，对于 n 个明文符号的集合而言，存在 n ！（阶乘）个可能的排列。可以从全部这种排列的集合中随机选择排列，并且将该排列作为转换表 906，以便将明文输入符号 Pi 908 转换成经加密的输出符号 Ci 910（还称为密本）。对于输入流中的每个明文符号而言，选择伪随机地选择的转换表。然后，见过加密输出符号 Ci 910 并且知道其对应于特定明文符号的攻击者依然不知道其它明文符号与对应的加密符号之间的任何对应关系。即，攻击者所能确定的全部信息是，改变加密的符号将产生与他们所知道的明文符号不同的明文符号，而不是它们应该是的其它明文符号。因此伪随机地选择的转换表不会泄漏明文输入符号和经加密的输出符号（密本）之间的关系，并且攻击者不能利用任意单个明文符号的知识来进行密文符号转换。

[0100] 在保护电话银行的一个实例中，将安全设备从电话接收的每个 DTMF 音调转变成（或关联到）数字明文符号。然后通过（基于来自密钥流的一个或多个伪随机数获得的）转换表来转换该明文符号，以获得加密符号。然后（以数字形式或者作为对应于经加密的符号的 DTMF 音调）向安全服务器传输经加密的符号，在安全服务器中通过反转换表对经加

密的符号进行解密。可以通过对在安全设备和用于生成相同密钥流的安全服务器两者上的密码生成器进行同步来生成或获得反转换表。在一个实例中，可以通过使用相同的种子（例如，会话密钥等等）来对密码生成器进行同步。

[0101] 在一个实例中，可以预先生成多个转换表，并且/或者安全设备和/或安全服务器存储该多个转换表。转换表可以被预先生成并且被存储，而非在运行中生成新的转换表（即输入符号的排列）。密钥流 904 的伪随机值/符号可被用来为每个将被加密的明文符号选择一个预先生成的转换表。预先生成的转换表可以为 n 个明文符号的集合定义每个排列或排列的子集。

[0102] 在另一个实例中，使用密钥流和伪随机混洗符号来形成转换表，从而实时生成转换表。注意：这些技术方案是等效的，因为存在 n ！ 个表，并且选择这些表中的一个所需要的密钥流的数量与通过混洗创建这种表所需要的数量相等。

[0103] 图 10 示出了用于将明文符号转换成加密符号的符号 - 符号转换表 1002 的实例。在该实例中，十六 (16) 个明文符号转换成不同的加密符号。在该实例中，示出了二进制的表示，这仅仅是举例说明可以使用 4 比特的加密符号来对 16 个明文符号进行加密。在另一个实例中，如果要对更大量或更少量的明文符号进行加密，则可以对每个符号使用不同数量的比特。例如，对于至多二百五十六 (256) 个明文符号，可以从密钥流中提取八 (8) 比特来生成每个加密符号。

[0104] 另一个特征提供了特定转换表内的明文符号与加密符号之间的一一对应关系。即，在特定转换表中没有两个明文符号转换成相同的加密符号。这允许解密设备将加密符号精确地解密为它的原始明文符号。

[0105] 在解密设备，可以生成符号 - 符号反转换表，以逆转加密设备的符号 - 符号转换表并且从而对接收的经加密的符号进行解密。

[0106] 图 11 示出了如何使用不同的转换表 1104 来对明文符号 1102 进行加密以获得经加密的符号 1106 的一个实例。对于每个明文符号 P0、P1、P2、P3、.....、Pi，使用不同的转换表 1104 来获得经加密的符号 C0、C1、C2、C3、.....、Ci，每个转换表 1104 具有不同的符号排列。

[0107] 对于集合中少量的符号而言，可以罗列出（例如，预先生成）这样的符号的全部排列，并且使用索引从排列中选择转换表。例如，对应具有十二 (12) 个可能符号的集合，所生成的可能的排列的数量是 12 ！ 或者 479001600。为了充分地选择排列，三十二 (32) 比特的密钥流足以选择一个排列作为转换表，而没有偏差。然而，随着集合中符号数量的增加，该方法变得不够用。例如，对于具有 256 个可能符号的集合而言，所生成的可能排列的数量是 256 ！ 或者 $8.5\times10^{506}$，其将占用伪随机密钥流的超过 1684 个比特以选择其中一个排列作为转换表。

[0108] 图 12 示出了从针对 n 个符号的集合的多个可能排列中选择转换表的算法，其中 n 是正整数。在该实例中，可以使用密码生成器，该密码生成器提供在 0 到 $2^k$-1 范围内均匀分布的 k 比特长（例如，8 比特、32 比特等等）的伪随机密钥流值。使用该密钥流来获得伪随机数 w 1202。因为 n ！ 可能不能被 $2^k$ 等分，所以不能无偏差地直接使用伪随机数 w。因为这个原因，将最大阈值 Pmax 定义为小于 $2^k$ 的 n ！ 的最大倍数。如果伪随机数 w 小于该最大阈值 Pmax，那么可以无偏差地使用伪随机数 w。否则，如果伪随机数 w 大于或者等于该

最大阈值 Pmax，那么将它丢弃并且选择新的伪随机数 w，直到获得小于最大阈值 Pmax 的伪随机数 w 为止 1204。

[0109] 求伪随机数 w 除以 n ！ 的余数，使得 w ＝ w％ (n ！ )1206。因此在范围 0 到 n ！ 中获得无偏差的伪随机数 w，其可用于获得排列 ( 即，转换表 )。

[0110] 不存储预先产生的排列并且使用伪随机数 w 来选择一个这种排列，而是一个特征提供一种通过混洗基本排列的符号来生成排列以生成转换表。利用符号集的全部值来初始化基本排列向量 P，使得 P ＝ [0,1,2, ..... , n-1]1208。然后使用符号混洗算法 1210，以便使用伪随机数 w 来混洗基本排列向量 P 中的符号。

[0111] 符号混洗算法 1210 的一个实例将计数器 i 初始化为 n-1，其中，n 是集合中的符号的数量。当计数器 i ＞＝ 0 时，伪随机数 w ＝ w/(i+1)，变量 j ＝ w％ (i+1) 并且对排列向量 P 的值进行混洗，使得 $P_t[i] = P_{t-1}[j]$ 并且 $P_t[j] = P_{t-1}[i]$。注意 :在不脱离本发明的特征的情况下还可以使用其它符号混洗算法。

[0112] 在混洗了排列向量 P 之后，可以将它提供 1212 给使用它的任何应用，例如，作为转换表来对输入符号流进行加密。

[0113] 图 13 示出了另一种加密方案的方框图，在该加密方案中，通过使用多个转换表对单个明文符号进行加密来实现符号认证。即，通过转换表 A11304 对明文输入符号 Pi 1302 进行加密以获得第一加密输出符号 Ci’1310，可以基于从第一密码生成器 1308 获得的第一密钥流 Si’ 1306 来生成和选择转换表 A11304。第一加密输出符号 Ci’ 1310 然后作为第二转换表 A21312 的输入，可以基于从第二密码生成器 1316 获得的第二密钥流 Si 1314 来生成和选择第二转换表 A21312，第二转换表 A21312 用于获得第二加密输出符号 Ci 1318。这样，可以使用冗余来认证第一加密输出符号 Ci’1310。即，通过一起使用符号 Ci’1310 和 Ci 1318，符号 Ci 1318 对符号 Ci’ 1310 进行认证。因此，例如，如果攻击者成功地改变了符号 Ci’ 1310，那么符号 Ci 1318 不会正确认证符号 Ci’ 1310。

[0114] 图 14 示出了如何使用多个转换表 1404 和 1406 对每个明文符号 1402 进行加密以获得对应的加密符号对 1408。应该注意，可以伪随机地选择并且 / 或者生成转换表 1404 和 1406，以将每个明文符号 Pi 加密成符号对 Ci’ /Ci。

[0115] 图 15 示出了如何使用两个转换表来将明文符号 Pn 转换或者加密成一对符号 Cn’ 和 Cn 的实例。例如，对于第一明文符号 Pn ＝ ‘5’，第一转换表 A11502 提供第一输出符号 Cn’ ＝ 8( 即，‘5’ 转换成 ‘8’ )。第一输出符号 ‘8’ 然后可以作为第二转换表 A21504 的输入以获得第二输出符号 Cn ＝ 7( 即，‘8’ 转换成 ‘7’ )。因为第二输出符号 Cn 是基于第一输出符号 Cn’ 生成的，所以冗余符号 Cn 和 Cn’ 可以用于认证。如果在传输期间攻击者改变了任意一个符号或者两个符号，那么认证失败。例如，如果攻击者将 Cn’ 从 ‘8’ 修改为 ‘4’，那么接收符号 Cn’ 和 Cn ＝ ‘47’ 的接收方会发现 Cn ＝ ‘7’ 意味着 Cn’ ＝ ‘8’ 而不是 ‘4’。

[0116] 第二明文符号 P(n+1) 可以完成不同的转换表，即使第一明文符号和第二明文符号相同时也是如此。例如，对于第二明文符号 P(n+1) ＝ ‘5’，第一转换表 B1 1506 提供第三输出符号 C(n+1)’ ＝ ‘*’ ( 即，‘5’ 转换成 ‘*’ )。第三输出符号 C(n+1)’ ＝ ‘*’ 然后可以作为第二转换表 B2 1508 的输入以获得第四输出符号 C(n+1) ＝ ‘1’ ( 即，‘*’ 转换成 ‘1’ )。如前所述，符号对 C(n+1)’ 和 C(n+1) 的冗余使用可以作为一种认证形式。

[0117] 图 16 示出了根据一个实例来执行明文加密的方法。获得在 n 个符号的集合中定

义的多个输入符号 1602。从多个定义不同符号 - 符号排列的转换表中获得用于每个待加密的输入符号的、伪随机地选择的转换表 1604。使用用于每个输入符号的对应转换表，将输入符号转换成对应的输出符号，以便逐个地对每个输入符号加密 1606。然后可以向解密设备传输输出符号 1608。

[0118] 在该方法的一个实例中，获得第一明文符号，其中第一明文符号可以是集合中的 n 个符号中的一个。获得第一转换表，该第一转换表将 n 个符号转换成 n 个符号的不同的排列。可以通过使用伪随机数混洗 n 个符号来伪随机地生成第一转换表。然后使用第一转换表将第一明文符号转换成第一输出符号。

[0119] 可以获得第二转换表，该第二转换表将 n 个符号转换成 n 个符号的与第一转换表不同的排列。使用第二转换表将第一输出符号转换成第二输出符号。然后基于第一和 / 或第二输出符号传输经加密的符号。

[0120] 图 17 示出了如何通过使用一个或多个反转换表来对经加密的符号 Ci 进行解密以获得单个明文符号的方框图。即，可以通过第一反转换表 A11704 对加密输入符号 Ci 1702 进行解密，以获得第一解密输出符号 Ci’ 1710，可以基于从第一密码生成器 1708 获得的第一密钥流 Si’ 1706 来生成或选择第一反转换表 A1 1704。第一解密输出符号 Ci’ 1710 然后作为第二反转换表 A2 1712 的输入，可以基于从第二密码生成器 1716 获得的第二密钥流 Si 1714 来生成或选择第二反转换表 A2 1712，第二反转换表 A2 1712 用于获得明文输出符号 Pi 18。

[0121] 在可替代的配置中，其中，例如 Ci = (x,y)，可以按照与对符号 x 和 y 的加密顺序相逆的顺序对加密符号 x 和 y 进行解密以获得明文输出符号 Pi。

[0122] 图 18 示出了根据一个实例来执行明文解密的方法。获得 n 个符号的集合中定义的多个（加密的）输入符号 1802。从定义不同的符号 - 符号排列的多个反转换表中获得用于每个待解密的输入符号的、伪随机地选择的反转换表 1804。使用用于每个输入符号的对应反转换表来将输入符号转换成对应的输出符号，以便逐个地对每个输入符号进行解密 1806。

[0123] 在该方法的一个实例中，获得第一加密符号（输入符号），其中第一加密符号是集合中的 n 个符号中的一个。还获得第一反转换表，该第一反转换表将 n 个符号转换成 n 个符号的不同的排列。可以通过使用伪随机数混洗 n 个符号来伪随机地生成第一反转换表。使用第一反转换表将第一加密符号转换成第一输出符号。获得第二反转换表，该第二反转换表将 n 个符号转换成 n 个符号的与第一转换表不同的排列。使用第二反转换表将第一输出符号转换成第二输出符号。然后基于第一和 / 或第二输出符号来获得明文符号。

[0124] 图 19 示出了根据一个实例的加密模块的方框图。加密模块 1902 可以包括处理电路 1904，其被配置为向密钥流生成器 1906 提供种子。密钥流生成器 1906 生成被发送到处理电路 1904 的伪随机数或符号的密钥流。耦合到处理电路 1904 的输入接口 1908 可以接收明文符号流。为了对明文符号流进行加密，处理电路 1904 可以被配置为使用从密钥流获得的伪随机数来从转换表生成器 1910 获得转换表。转换表生成器 1910 可以被配置为使用伪随机数来（例如）以伪随机的、无偏差的方式混洗并且 / 或者组合基本表的符号，以将第一明文符号转换成加密的符号流中的第一加密符号。然后可以通过耦合到处理电路 1904 的输出接口 1912 传输加密的符号流。对于明文符号流中的每个明文符号，可以生成不同的

转换表，并且可以使用不同的转换表来转换该符号。

[0125] 图 20 示出了根据一个实例的解密模块的方框图。解密模块 2002 可以包括处理电路 2004，其被配置为向密钥流生成器 2006 提供种子。密钥流生成器 2006 生成被发送到处理电路 2004 的伪随机数或符号的密钥流。耦合到处理电路 2004 的输入接口 2008 可以接收加密的符号流。为了对加密的符号流进行解密，处理电路 2004 可以被配置为使用从密钥流获得的伪随机数来从反转换表生成器 2010 获得转换表。反转换表生成器 2010 可以被配置为使用伪随机数来（例如）以伪随机的、无偏差的方式混洗并且 / 或者组合基本表的符号，以提供转换表。处理电路 2004 然后使用反转换表，来将第一加密符号转换成明文符号流中的第一明文符号。可以通过耦合到处理电路 2004 的输出接口 2012 来传输明文符号流。

[0126] 为了使加密模块 1902 和解密模块 2002 分别正确地对符号进行加密和解密，它们可以具有相同的密钥流生成器并且具有互补的转换表生成器。为了对密钥流生成器 1906 和 2006 进行同步，可以为加密模块与解密模块之间的特定通信会话建立共用种子（例如，通过安全认证方案）。例如，可以将会话密钥作为用于密钥流生成器 1906 和 2006 的种子。

[0127] 虽然本文描述的一些实例涉及 DTMF 音调的加密，但是本文所述的加密方法可以用于许多其它类型的通信系统以保护所传输的信息。

[0128] 可以将图 1-18 所示的一个或多个组件、步骤和 / 或函数重新排列或组合成单个组件、步骤或函数或分散到多个组件、步骤或函数中而不会脱离本发明的范围。在不脱离本发明的情况下还可以添加附加的元件、组件、步骤和 / 或函数。图 1、2、3、5、7、9、13、17、19 和 / 或 20 中所示的装置、设备和 / 或组件可以被配置为执行图 2、4、6、8、10、11、12、14、15、16 和 / 或 18 中所述的一个或多个方法、特征或步骤。

[0129] 本领域技术人员将进一步理解，可以将本文结合公开的实例描述的各种示例性逻辑块、模块、电路和算法步骤实现为电子硬件、计算机软件或者两者的组合。为了清楚地使出硬件和软件的这种可互换性，上文通过功能描述了各种示例性的组件、块、模块、电路和方法。将该功能实现为硬件还是软件取决于特定的应用和强加在整个系统上的设计限制。应该注意，前述的配置仅仅是实例，而不要将前述的配置理解成是要限制本发明。这些实例的描述意图是说明性的，而不是要限制权利要求的范围。同样，本发明的教导可以容易地应用于其它类型的装置，并且许多替换、修改和变形对于本领域技术人员而言是显而易见的。

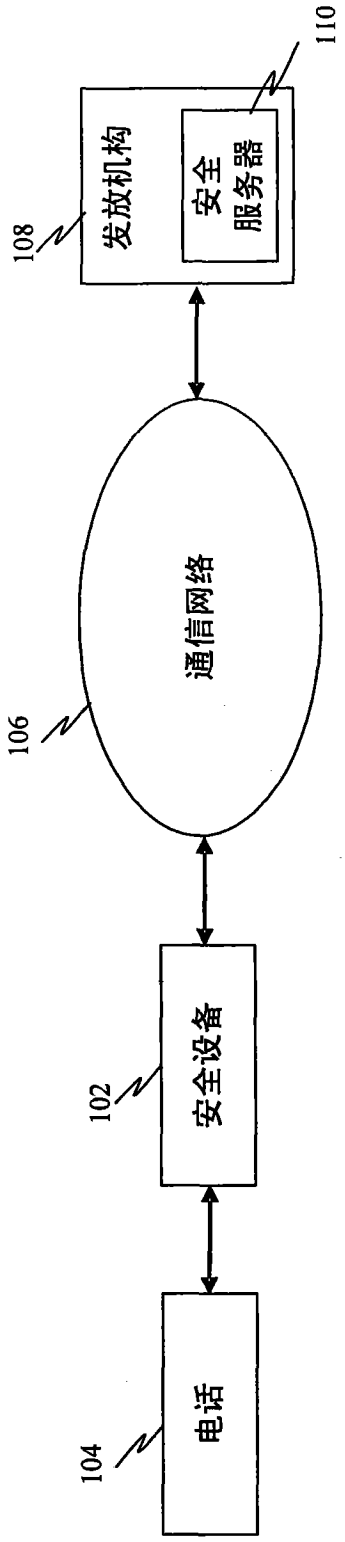



图 1

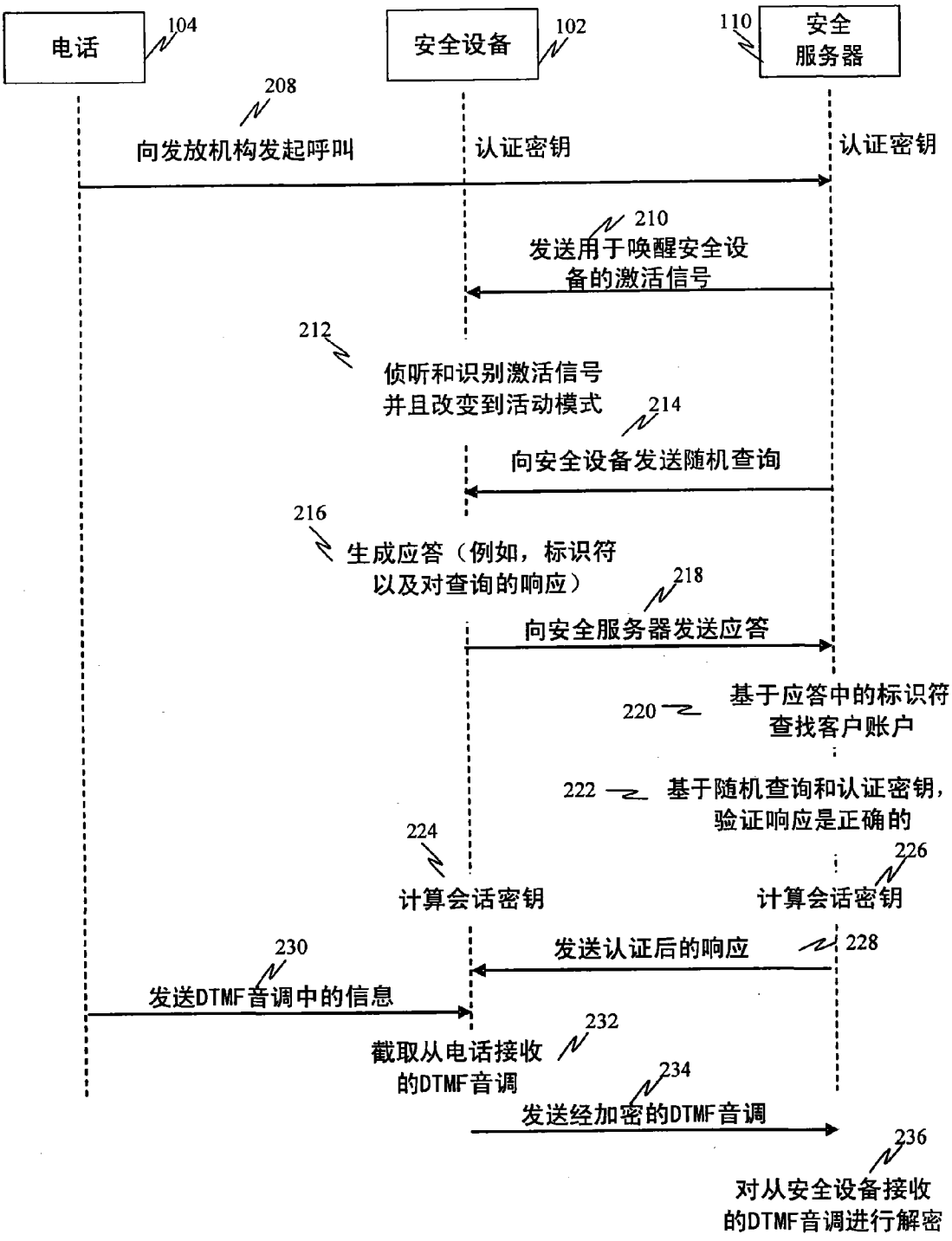

电话
104
安全设备
102
110
安全服务器
208
向发放机构发起呼叫
认证密钥
认证密钥
210
发送用于唤醒安全设备的激活信号
212
侦听和识别激活信号并且改变到活动模式
214
向安全设备发送随机查询
216
生成应答（例如，标识符以及对查询的响应）
218
向安全服务器发送应答
220
基于应答中的标识符查找客户账户
222
基于随机查询和认证密钥，验证响应是正确的
224
计算会话密钥
226
计算会话密钥
228
发送认证后的响应
230
发送DTMF音调中的信息
232
截取从电话接收的DTMF音调
234
发送经加密的DTMF音调
236
对从安全设备接收的DTMF音调进行解密

图 2

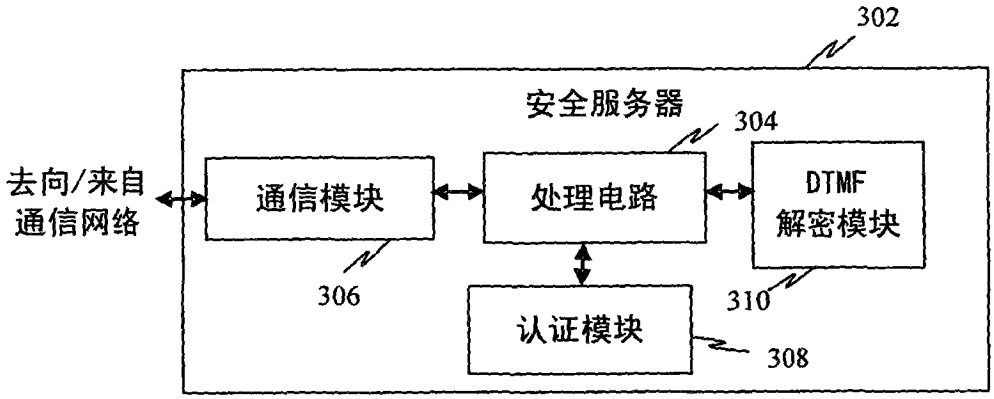

302
安全服务器
304
去向/来自
通信网络
通信模块
处理电路
DTMF
解密模块
306
310
认证模块
308

图 3

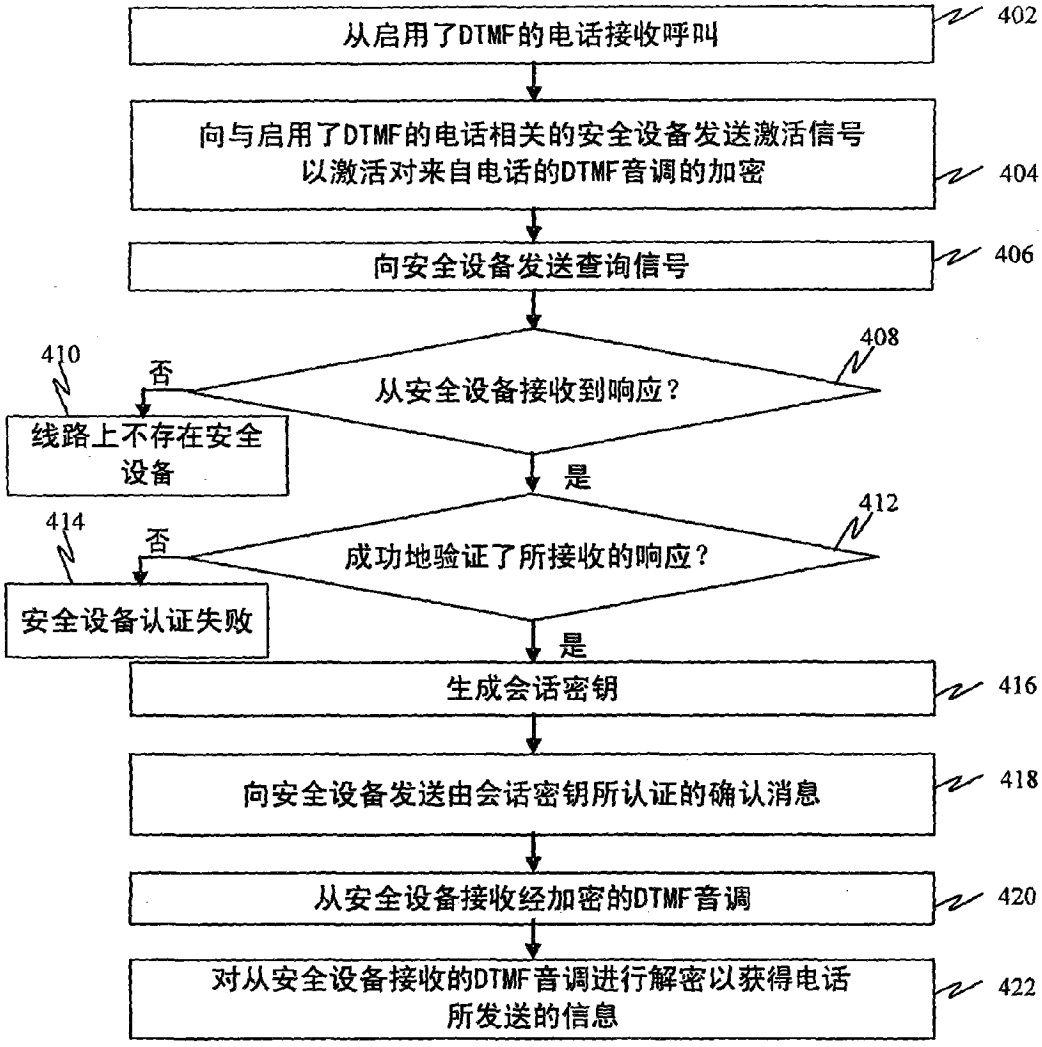

从启用了DTMF的电话接收呼叫
402
向与启用了DTMF的电话相关的安全设备发送激活信号
以激活对来自电话的DTMF音调的加密
404
向安全设备发送查询信号
406
408
从安全设备接收到响应？
否
410
线路上不存在安全
设备
是
412
成功地验证了所接收的响应？
否
414
安全设备认证失败
是
生成会话密钥
416
向安全设备发送由会话密钥所认证的确认消息
418
从安全设备接收经加密的DTMF音调
420
对从安全设备接收的DTMF音调进行解密以获得电话
所发送的信息
422

图 4

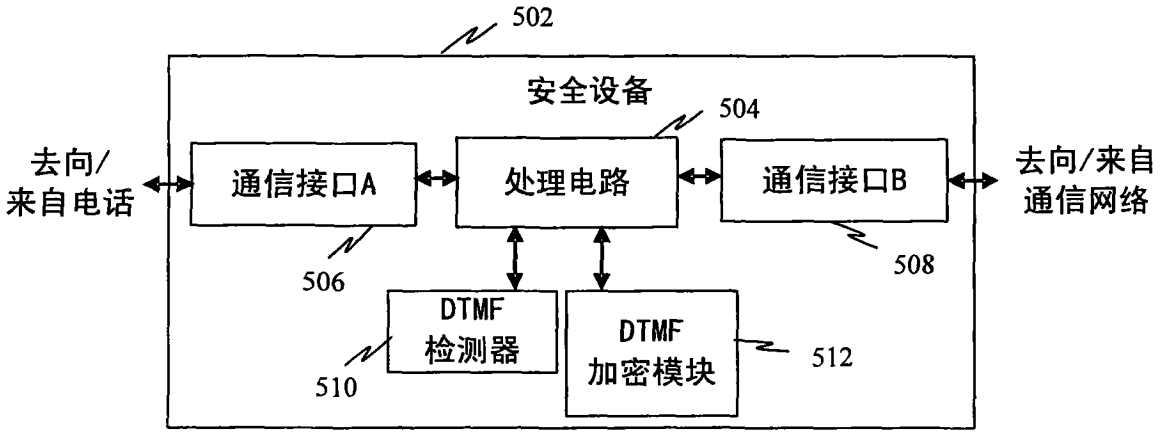

502
安全设备
504
去向/
来自电话
通信接口A
处理电路
通信接口B
去向/来自
通信网络
506
508
DTMF
检测器
510
DTMF
加密模块
512

图 5

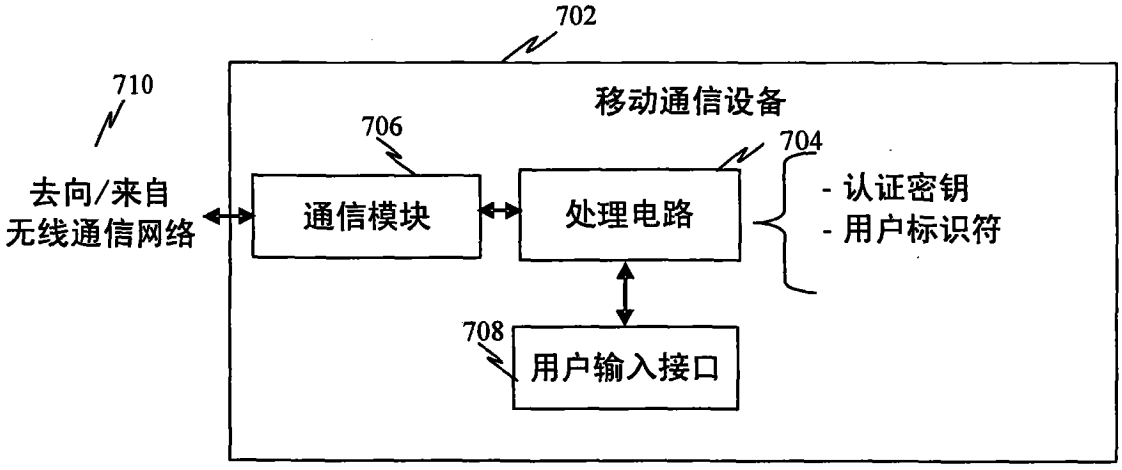

702
710
移动通信设备
706
704
去向/来自
无线通信网络
通信模块
处理电路
- 认证密钥
- 用户标识符
708
用户输入接口

图 7

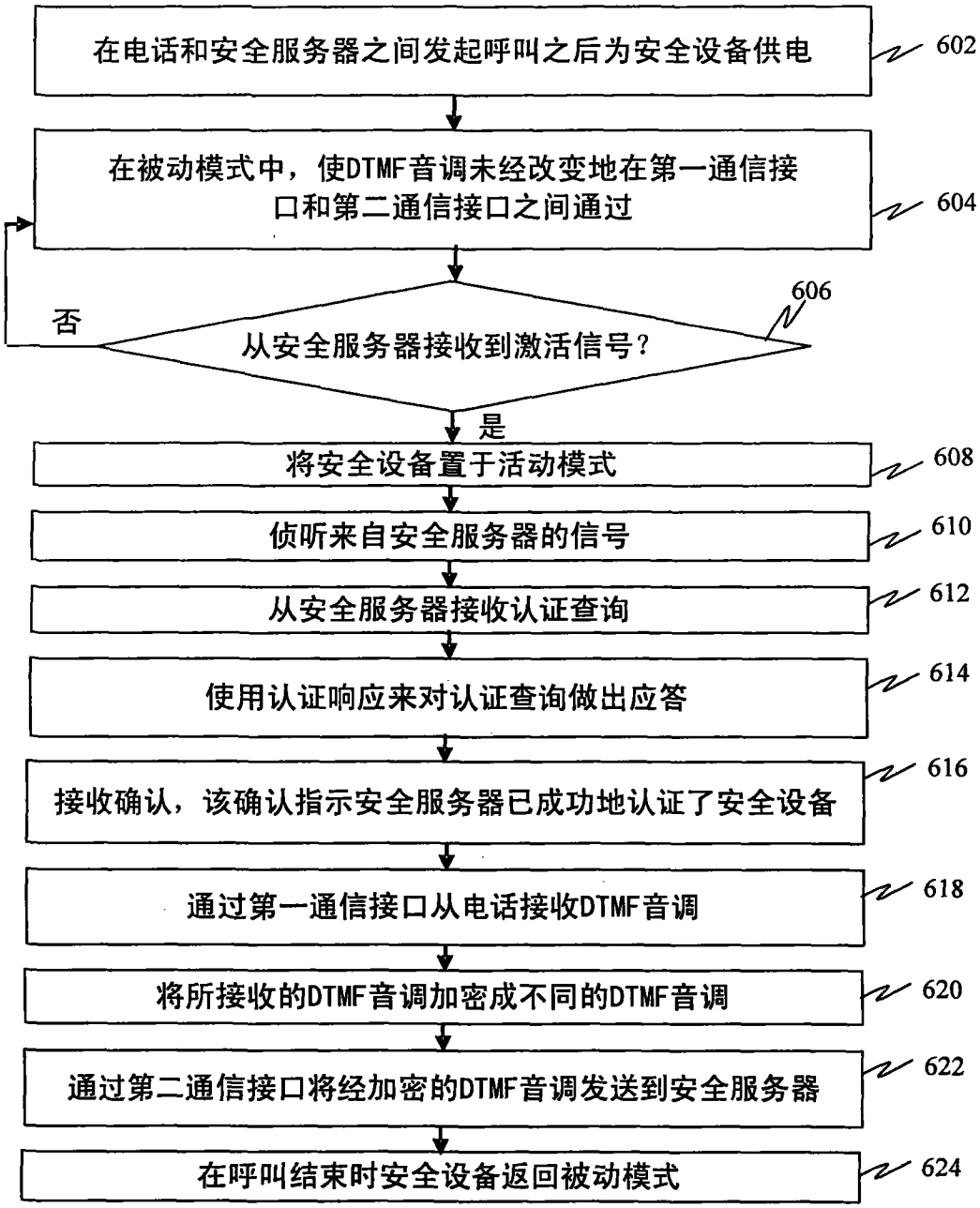

在电话和安全服务器之间发起呼叫之后为安全设备供电
602
在被动模式中，使DTMF音调未经改变地在第一通信接口和第二通信接口之间通过
604
606
否
从安全服务器接收到激活信号？
是
将安全设备置于活动模式
608
侦听来自安全服务器的信号
610
从安全服务器接收认证查询
612
使用认证响应来对认证查询做出应答
614
接收确认，该确认指示安全服务器已成功地认证了安全设备
616
通过第一通信接口从电话接收DTMF音调
618
将所接收的DTMF音调加密成不同的DTMF音调
620
通过第二通信接口将经加密的DTMF音调发送到安全服务器
622
在呼叫结束时安全设备返回被动模式
624

图 6

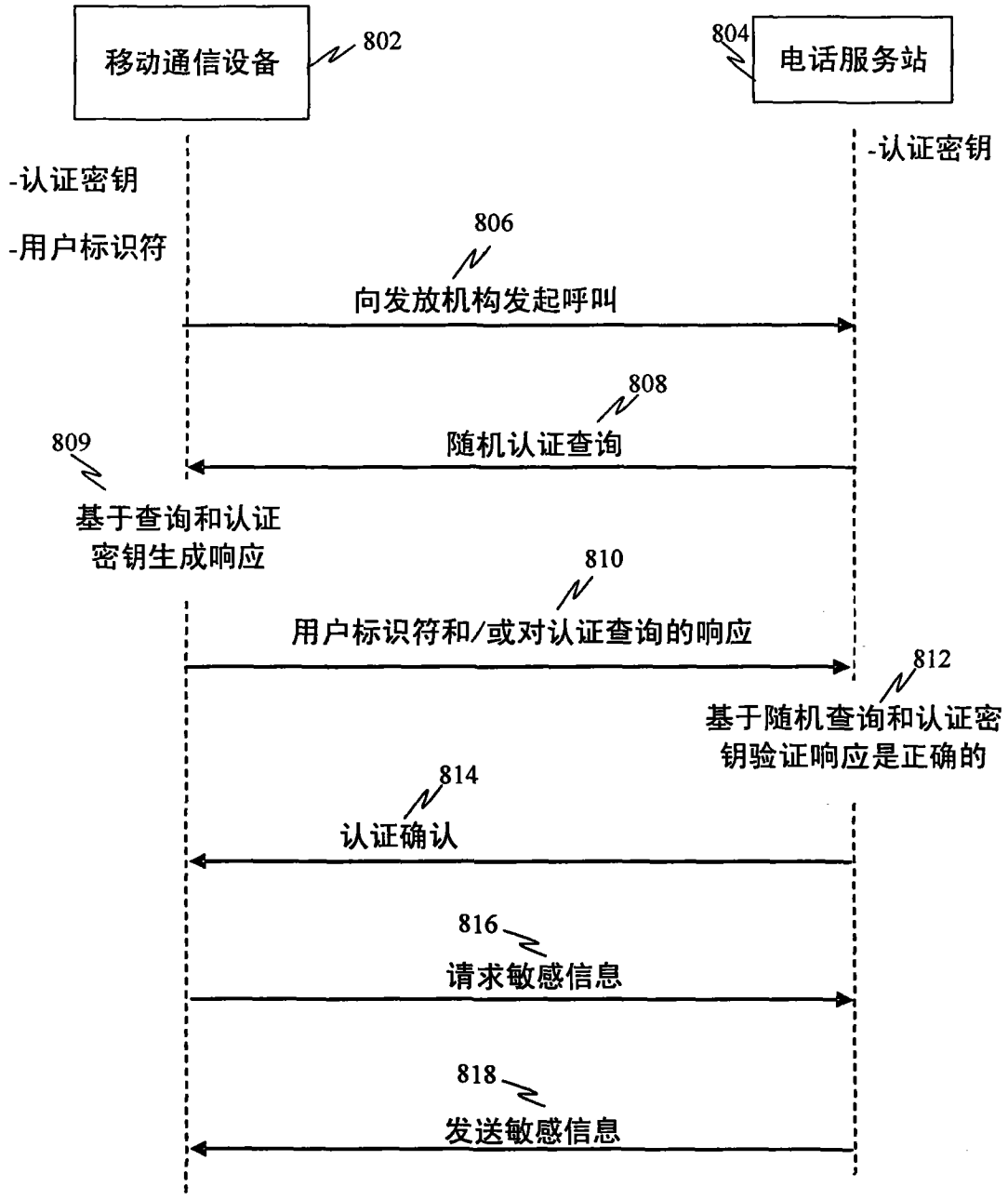

移动通信设备
802
804
电话服务站
-认证密钥
-认证密钥
-用户标识符
806
向发放机构发起呼叫
808
随机认证查询
809
基于查询和认证密钥生成响应
810
用户标识符和/或对认证查询的响应
812
基于随机查询和认证密钥验证响应是正确的
814
认证确认
816
请求敏感信息
818
发送敏感信息

图 8

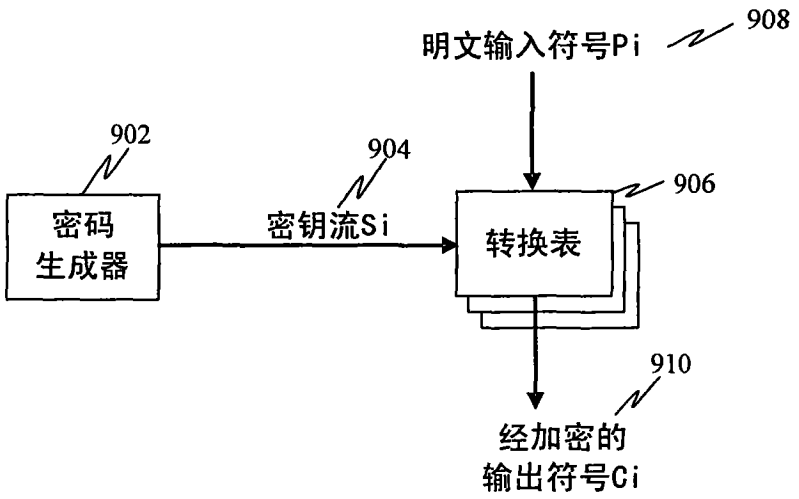



图 9

| 明文符号 | 明文符号的二进制形式 | 经加密的符号 | 经加密的符号的二进制形式 |
|---|---|---|---|
| 0 | 0000 | 5 | 0101 |
| 1 | 0001 | 9 | 1001 |
| 2 | 0010 | A | 1100 |
| 3 | 0011 | 0 | 0000 |
| 4 | 0100 | # | 1011 |
| 5 | 0101 | 1 | 0001 |
| 6 | 0110 | C | 1110 |
| 7 | 0111 | * | 1010 |
| 8 | 1000 | 7 | 0111 |
| 9 | 1001 | B | 1101 |
| * | 1010 | 4 | 0100 |
| # | 1011 | 6 | 0110 |
| A | 1100 | D | 1111 |
| B | 1101 | 2 | 0001 |
| C | 1110 | 8 | 1000 |
| D | 1111 | 3 | 0011 |

1002

图 10

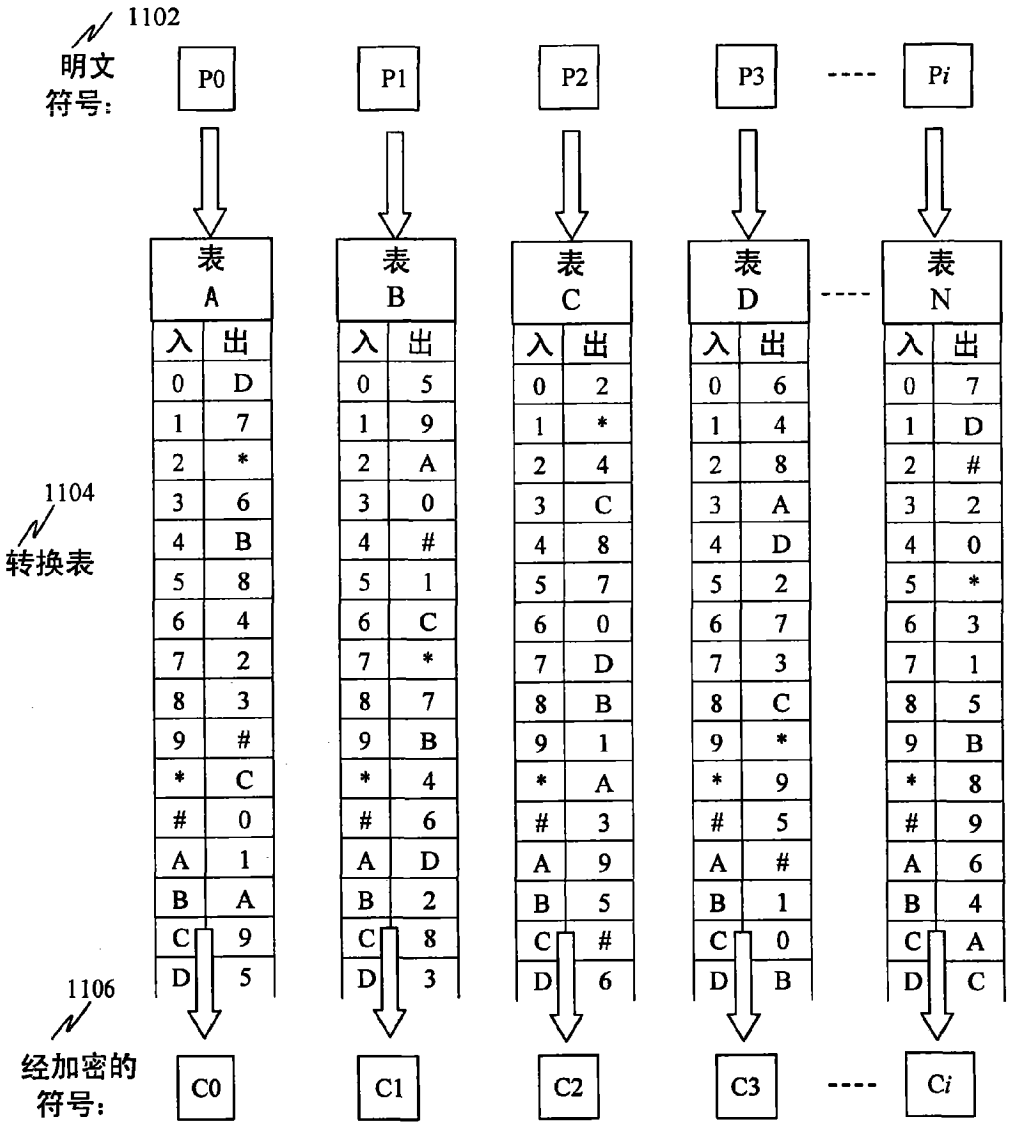

1102
明文符号:
P0
P1
P2
P3
----
Pi
表A
表B
表C
表D
表N
入 出
0 D
1 7
2 *
3 6
4 B
5 8
6 4
7 2
8 3
9 #
* C
# 0
A 1
B A
C 9
D 5
入 出
0 5
1 9
2 A
3 0
4 #
5 1
6 C
7 *
8 7
9 B
* 4
# 6
A D
B 2
C 8
D 3
入 出
0 2
1 *
2 4
3 C
4 8
5 7
6 0
7 D
8 B
9 1
* A
# 3
A 9
B 5
C #
D 6
入 出
0 6
1 4
2 8
3 A
4 D
5 2
6 7
7 3
8 C
9 *
* 9
# 5
A #
B 1
C 0
D B
入 出
0 7
1 D
2 #
3 2
4 0
5 *
6 3
7 1
8 5
9 B
* 8
# 9
A 6
B 4
C A
D C
1104
转换表
1106
经加密的符号:
C0
C1
C2
C3
----
Ci

图 11

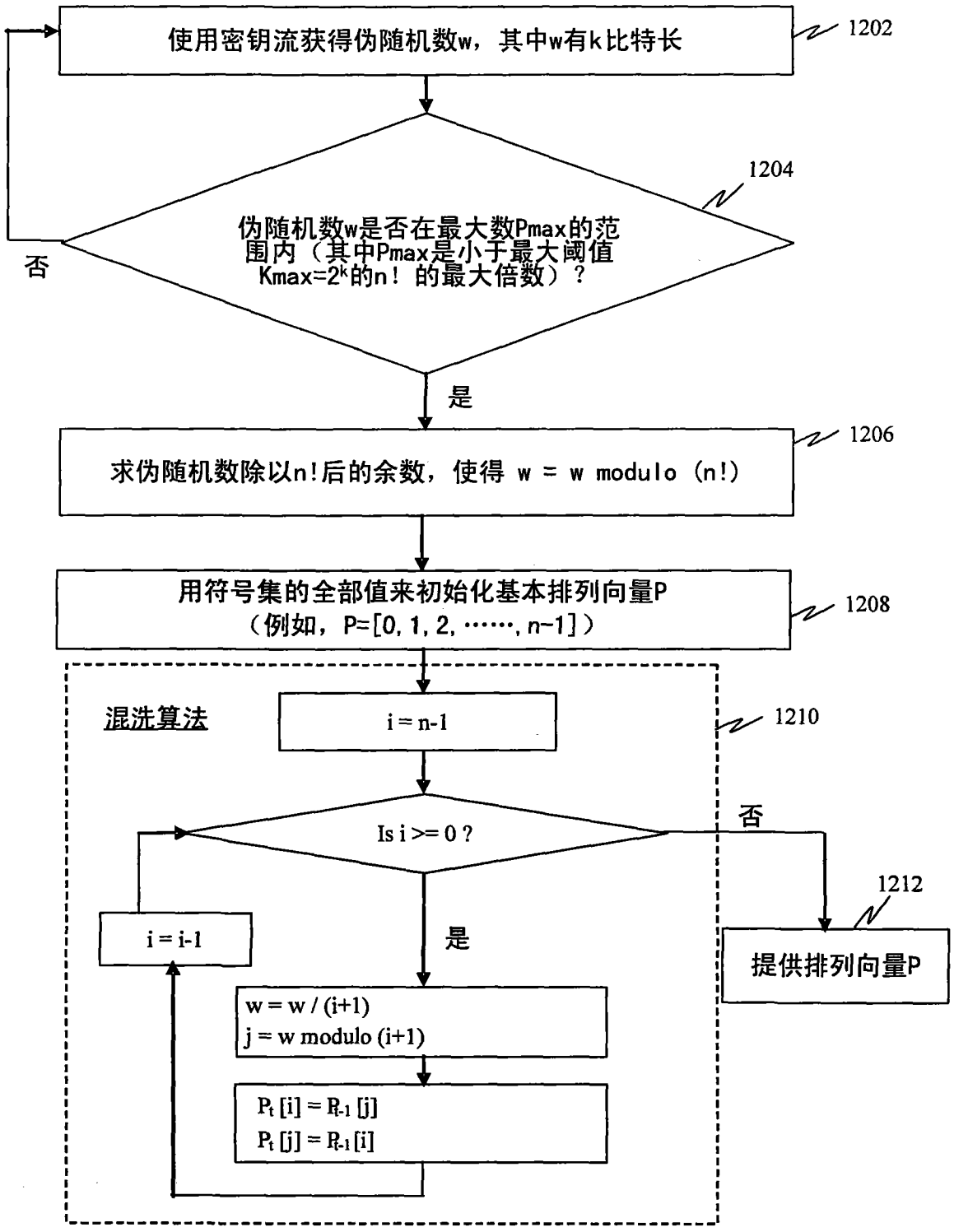

使用密钥流获得伪随机数w，其中w有k比特长
1202
伪随机数w是否在最大数Pmax的范围内（其中Pmax是小于最大阈值Kmax=2^k的n！的最大倍数）？
1204
否
是
求伪随机数除以n!后的余数，使得 w = w modulo (n!)
1206
用符号集的全部值来初始化基本排列向量P
（例如，P=[0, 1, 2, ……, n-1]）
1208
混洗算法
i = n-1
1210
Is i >= 0 ?
否
是
1212
i = i-1
w = w / (i+1)
j = w modulo (i+1)
Pt [i] = Pt-1 [j]
Pt [j] = Pt-1 [i]
提供排列向量P

图 12

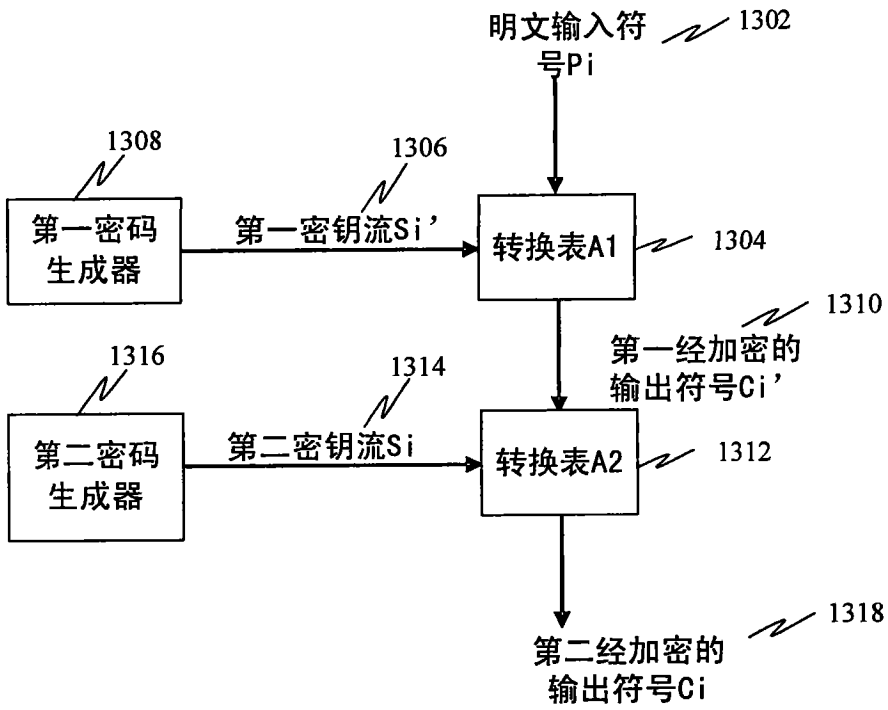

明文输入符号Pi
1302
1308
1306
第一密码生成器
第一密钥流Si'
转换表A1
1304
1310
第一经加密的输出符号Ci'
1316
1314
第二密码生成器
第二密钥流Si
转换表A2
1312
1318
第二经加密的输出符号Ci

图 13

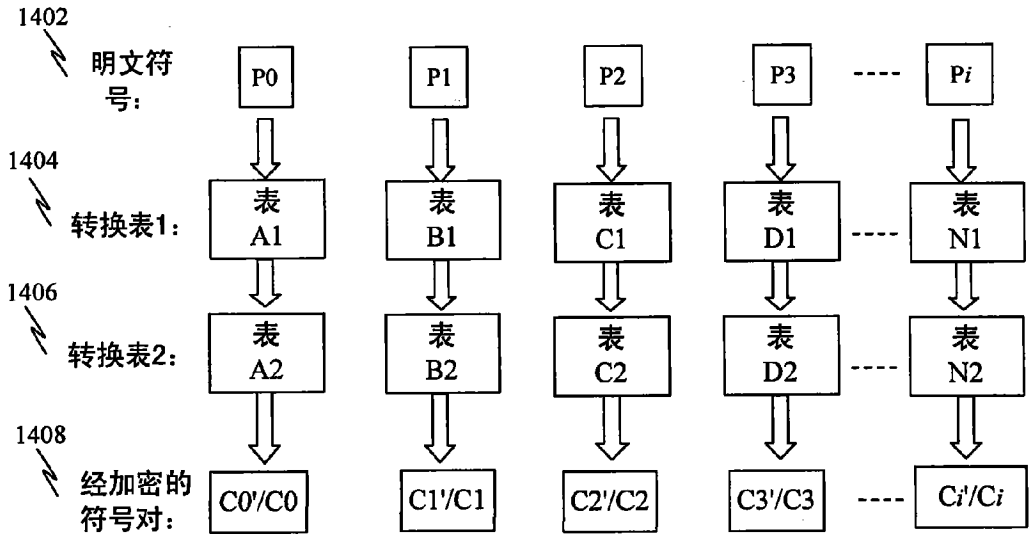

1402
明文符号:
P0
P1
P2
P3
----
Pi
1404
转换表1:
表A1
表B1
表C1
表D1
表N1
1406
转换表2:
表A2
表B2
表C2
表D2
表N2
1408
经加密的符号对:
C0'/C0
C1'/C1
C2'/C2
C3'/C3
Ci'/Ci

图 14

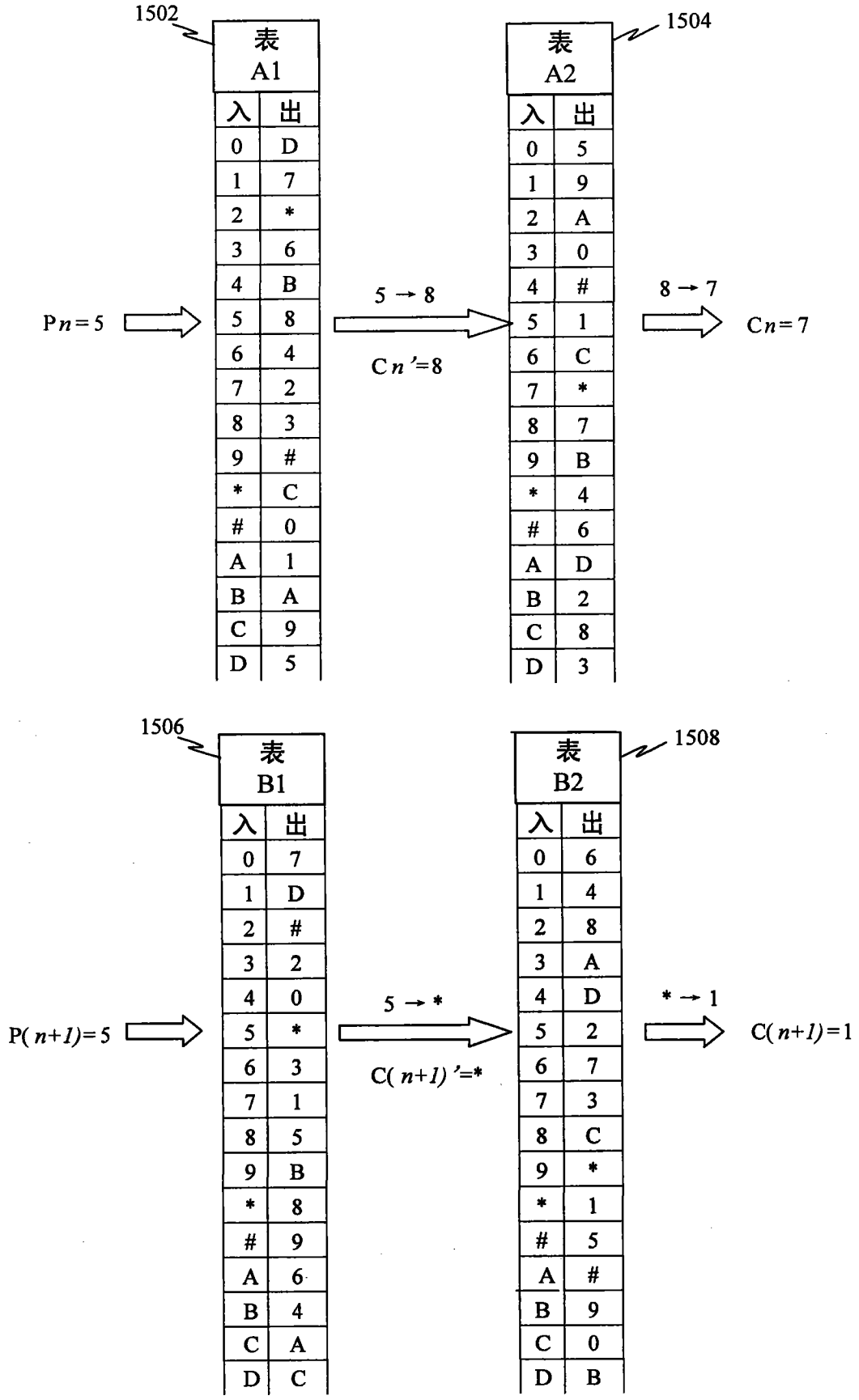

1502
表 A1
入 出
0 D
1 7
2 *
3 6
4 B
5 8
6 4
7 2
8 3
9 #
* C
# 0
A 1
B A
C 9
D 5
Pn=5
5 → 8
Cn'=8
1504
表 A2
入 出
0 5
1 9
2 A
3 0
4 #
5 1
6 C
7 *
8 7
9 B
* 4
# 6
A D
B 2
C 8
D 3
8 → 7
Cn=7
1506
表 B1
入 出
0 7
1 D
2 #
3 2
4 0
5 *
6 3
7 1
8 5
9 B
* 8
# 9
A 6
B 4
C A
D C
P(n+1)=5
5 → *
C(n+1)'=*
1508
表 B2
入 出
0 6
1 4
2 8
3 A
4 D
5 2
6 7
7 3
8 C
9 *
* 1
# 5
A #
B 9
C 0
D B
* → 1
C(n+1)=1

图 15

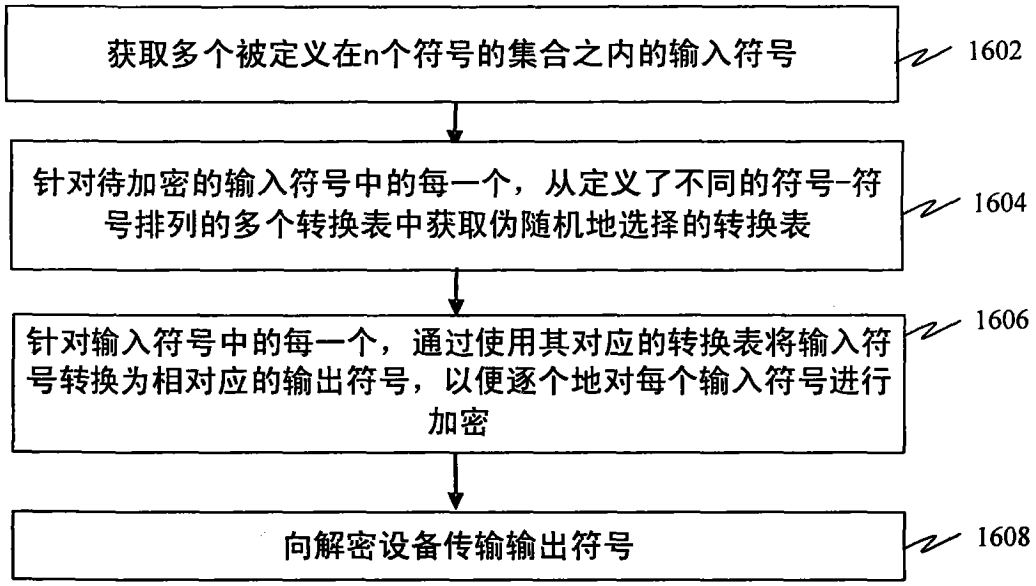

获取多个被定义在n个符号的集合之内的输入符号
1602
针对待加密的输入符号中的每一个，从定义了不同的符号-符号排列的多个转换表中获取伪随机地选择的转换表
1604
针对输入符号中的每一个，通过使用其对应的转换表将输入符号转换为相对应的输出符号，以便逐个地对每个输入符号进行加密
1606
向解密设备传输输出符号
1608

图 16

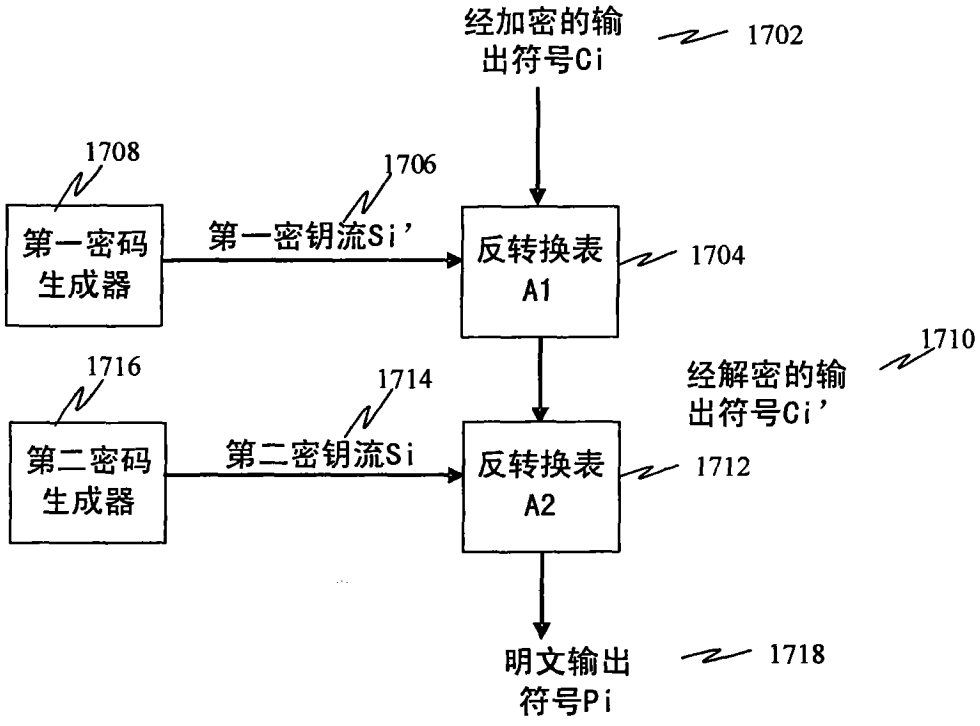

经加密的输出符号Ci
1702
1708
1706
第一密码生成器
第一密钥流Si'
反转换表A1
1704
1710
经解密的输出符号Ci'
1716
1714
第二密码生成器
第二密钥流Si
反转换表A2
1712
明文输出符号Pi
1718

图 17

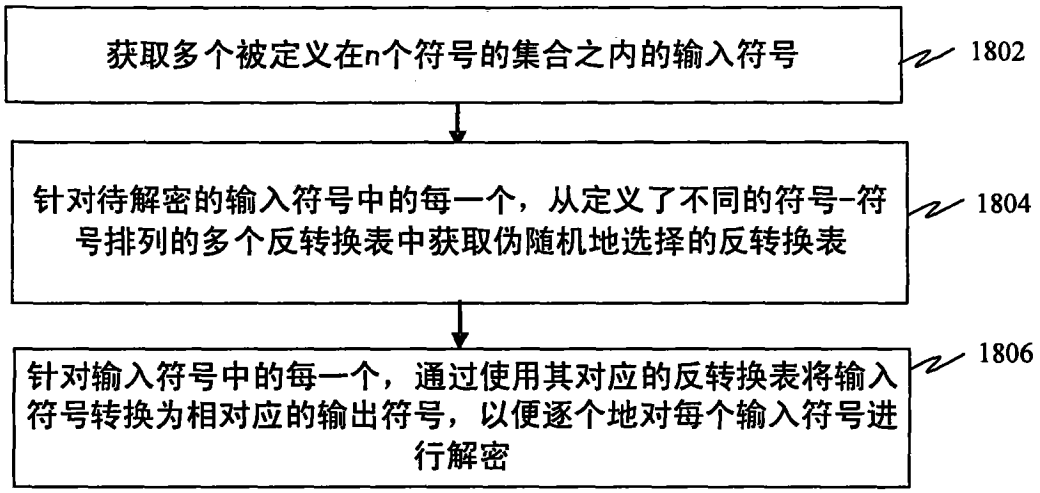

获取多个被定义在n个符号的集合之内的输入符号
1802
针对待解密的输入符号中的每一个，从定义了不同的符号-符号排列的多个反转换表中获取伪随机地选择的反转换表
1804
针对输入符号中的每一个，通过使用其对应的反转换表将输入符号转换为相对应的输出符号，以便逐个地对每个输入符号进行解密
1806

图 18

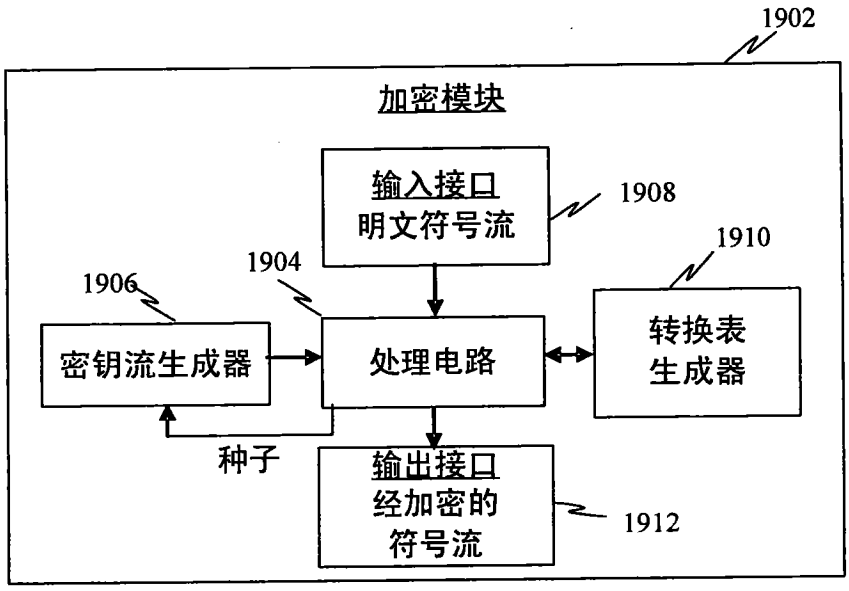

1902
加密模块
输入接口
明文符号流
1908
1910
1904
1906
密钥流生成器
处理电路
转换表
生成器
种子
输出接口
经加密的
符号流
1912

图 19

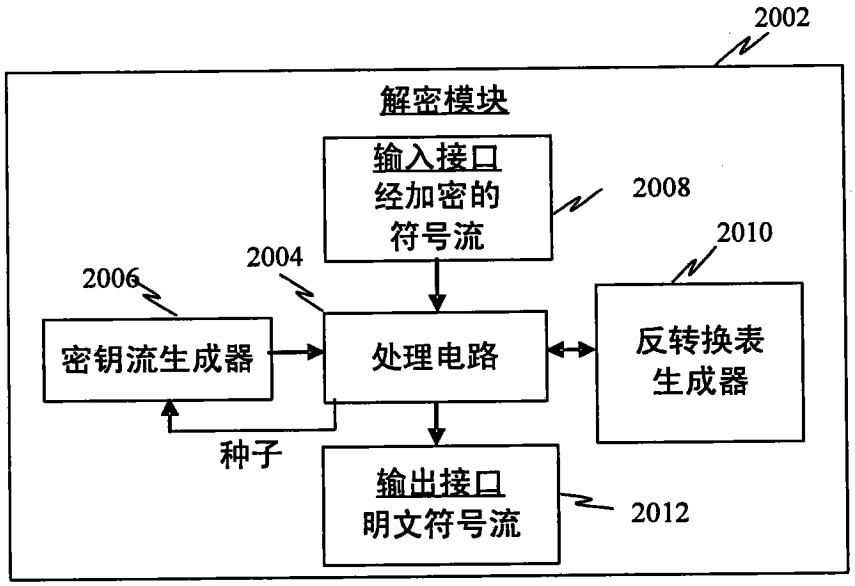

2002
解密模块
输入接口
经加密的
符号流
2008
2010
2004
2006
密钥流生成器
处理电路
反转换表
生成器
种子
输出接口
明文符号流
2012

图 20