密码学的转型

ABRAHAM LEMPEL

以色列理工学院电气工程系，海法，以色列

本综述重点阐述公钥密码系统这一新概念、近期提出的实现方案，以及与密码复杂性相关的问题。同时，文章还简要概述了经典密码学、当代密码学的基本原理，并对当时已成为官方标准的数据加密标准（DES）做了简短描述。

关键词和短语：密码学，密码，加密，解密，密码系统，密码复杂性，公钥密码系统

CR分类：3.15, 3.71, 5.14, 5.30

前言

本文旨在向读者传达公钥密码学当前所处的转型阶段及其不断发展的技术现状。这种转型是多方面的：就其相关性范围而言，它已从一个处理军事和外交通信的政府垄断领域，演变为商业（尤其是银行业）的主要关切点，最近更成为广大公众的关注焦点。其技术已从纸笔操作和各种机械设备扩展到大型高速电子计算机。其安全性的侧重点已从统计不确定性转向计算复杂性。最后，同样重要的是，在概念上，它已从传统的私钥方案发展到能提供即时保密性和双向身份验证的公钥密码系统。

这种最新的转型也反映了密码学这门"艺术"的快速发展状态。自从学术圈最近将其接纳为一个合法的研究领域后，公钥密码学的支持者正努力将其从一门艺术转变为一门科学，即用可靠的度量和标准取代直觉，依靠证明而非认证。

本综述的主要重点是关于公钥密码系统的新概念及其近期提出的实现方案（第3节）。在探讨此主题之前，本文按时间顺序介绍了密码学的历史，简要概述了经典密码学及其主要方案（第1节），阐述了当代密码学的基本原理，并对当时已成为官方标准的数据加密标准做了简短描述（第2节）。第4节讨论了密码复杂性这一有问题的概念；通过一个易于破解的NP完全密码的例子，展示了此概念与复杂性理论中NP完全性概念的差异。文中还指出了脱离上下文应用复杂性度量的危险，以及需要进行一些基础研究而非仅仅提出更多实现方案的必要性。

本文的初步版本...

目录

前言 引言

1. 经典密码学
   1.1 凯撒密码
   1.2 简单替换
   1.3 多表替代密码
   1.4 换位
   1.5 乘积密码
2. 当代密码的基本原理
   2.1 流密码
   2.2 分组密码
   2.3 数据加密标准
3. 公钥密码系统
   3.1 Rivest-Shamir-Adleman (RSA) 方案
   3.2 Merkle-Hellman (MH) 方案
   3.3 McEliece 方案
   3.4 Graham-Shamir (GS) 方案
   3.5 仅签名方案
4. 论密码复杂性的复杂性
   示例：一个易于破解的NP完全密码

致谢
参考文献

...曾作为1978年9月的Sperry报告[LEMP78]出现，其中还包含一份带注释的密码学文献目录。这份由Diffie和Hellman编纂的文献目录后来在其他地方发表[DIFF79]。

我们想指出，文中所有对"已知"或"现有"方法和方案的引用，均指非机密且非专有的知识。在公钥系统领域，我们已尽力涵盖截至1978年8月所有已知的方案；鉴于该主题发展迅速，我们可能遗漏了一些。

引言

密码学是设计和破解保密系统的艺术，或者说是一门有志成为科学的技术。其设计或合成部分通常称为密码学，而破解或分析部分则称为密码分析学。作为一门艺术，密码学可以追溯到有文字记载历史的早期；作为一门有志成为科学的技术，它正处于寻找适当的安全标准和复杂性度量的早期阶段，目前仍主要依赖于通过大量人力/计算机年努力进行的"认证"，而非严格的证明。

关于密码学最完整的历史记载在D. Kahn于1967年出版的《破译者：秘密书写的故事》[KAHN67]一书中，涵盖了从古埃及法老时代一直到第二次世界大战的时期。虽然该书更关注密码学的影响而非其技术发展，但它为该主题提供了极佳的入门介绍。从1978年夏季的视角来看，Kahn著作所涵盖的许多世纪可以被归为密码学的经典时代。尽管从简单的凯撒

密码（归功于尤利乌斯·凯撒）到第二次世界大战中使用的复杂的转子机之间存在巨大的复杂性差距，但它们的共同点在于它们都出现在电子计算机问世之前。

随着电子计算机的引入而变得可用的计算能力开启了密码学的现代时代；自那时起，计算能力的急剧增长及其对社会的深远影响极大地拓宽了密码学的相关范围。这个主要涉及确保军事和外交通信安全与破解的领域，在不到十年前还被认为是受到严格政府控制、甚至完全是政府垄断的，如今已成为商业（尤其是银行业）的主要关切点，最近更成为广大公众的关注焦点。在计算机数据库和电子资金转账（EFT）不断增长的时代，人们难以高估那些能提供足够保护以防止对存储数据的未经授权（通常是远程）访问、在公开可访问的通信链路上使数据对未经授权的窃听者不可读、并包含可作为可靠双向身份验证的数字签名的加密方案的重要性。这些艰巨的目标是为了满足真实的市场需求，数据加密标准（DES）[NBS77]部分回应了这些需求。DES是IBM Lucifer[SMIT71, FEIS73]的缩减和修改版本，自1977年1月起已作为官方的美国国家标准局（NBS）方案。DES（以及目前已知的任何其他使用中的方案）的主要缺点是，它要求每对通信方预先建立私钥，因此无助于缓解日益复杂的密钥管理问题。

在这个方向上迈出的重要一步是在1976年由斯坦福大学的Diffie和Hellman [DIFF76]以及Merkle [MERK78a]完成的，他们引入了"公钥分发"和"陷门单向函数"的概念，用于"公钥密码系统"，这种系统不需要预先的密钥通道，并允许不可伪造的数字签名。这些起初看起来聪明但不切实际的想法迅速获得了发展势头，并很快被麻省理工学院的Rivest、Shamir和Adleman [RIVE78a]提出的一个易于实现的方案所跟进。其他没有数字签名功能的实现方案由Merkle和Hellman [MERK78b]提出，不久之后McEliece [McEL78]也提出了方案。

这些新加密方案的影响并未被任何相关方忽视：各种安全机构显然正在审查它们[KOLA78]；工业界正在探索实施方案；同样重要的是，学术圈——这一切的发源地——已将密码学接纳为一个合法的研究领域，并且是一个非常活跃的领域。

尽管应该给予应有的肯定，但需要指出的是，这些新的加密方案尚未被证明构成了一种"需要数百万年才能破解的新型密码"，正如1977年8月《科学美国人》[GARD77]一篇文章的标题所宣称的那样。同月，《科学》[KOLA77b]杂志一篇题为"密码学：处于革命的边缘？"的文章则使用了更恰当的标题；从这个标题中移除问号还有待实现。问题在于，似乎缺乏证据证明这些新方案确实如它们看起来那样难以破解。它们的强度在于解决某些数学问题的计算复杂性，而使用目前已知的方法处理这些问题非常困难。鉴于此类加密方案的深远影响，问题在于底层数学能否、或者能在多大程度上抵御高度积极的数学界的一致努力。另一方面，同样的动机可能会导致新方案的发明，或者建立更真正符合密码学需求的密码复杂性度量标准，从而为更严格地断言不可破译性铺平道路。

1. 经典密码学

经典密码学的两个主要工具是代码和密码。两者都旨在将明文消息转换为隐藏文本的密文。代码是一个固定的预定字典，为最可能的消息分配代码字，因此其主要用途是针对那些可以从预选消息组合而成的文本。代码的固定性质也损害了其提供的安全性，因此代码通常与密码结合使用以产生加密的代码。

密码是一种通用方案，采用一组变换，能够将任何明文转换为密文。在任何给定时间应用的具体变换由当时使用的密钥控制。假定使用的密钥为（合法）发送方和接收方所知，但至少在事先不为密码分析者所知，后者的目标是破解该系统。

遵循文献和实践的趋势，本文仅讨论基于密码的密码系统，其一般结构如图1所示。E、D和CA模块分别表示加密、解密和密码分析方案。给定消息 M 和密钥 K，加密方案产生密文 $C = \mathbf{E}_K(M)$，这种函数表示法（而不是 $\mathbf{E}(K,M)$）意在表明，通常 K 是一个加密/解密参数，会在相当数量的消息中保持固定。给定 C 和相同的 K，解密方案可以容易地产生原始 $M = D_K(C)$。

假定密码分析者完全了解 $\mathbf{E}$ 和 $\mathbf{D}$ 方案，并且能够访问 C 和各种辅助信息SI，例如语言统计、正在进行的通信的一般上下文以及一些明文；他的任务是在已知E、D、C 和 SI，但未知 K 的情况下，产生 M 的最佳估计 $\tilde{M}$。虽然在某些情况下密码分析者可能在没有先找到 K 的情况下产生正确的 M，但破解密码系统通常意味着找到一个能够在给定"足够"SI的情况下产生 K 的方案。

因此，给定加密方案 $\mathbf{E}$ 所提供的保护通常根据密码分析者在试图确定 $\mathbf{E}$ 下允许的潜在密钥中实际使用的是哪一个时所面临的不确定性来衡量。这种方案 $\mathbf{E}$ 的低效率则根据底层语言的统计冗余度来衡量，这种冗余度可以扩展到包括SI的其他方面。

信息论的出现使得对这些概念（如不确定性和冗余度）进行定量处理成为可能。信息论的数学基础及其在经典密码系统中的应用由Shannon在两个里程碑式的著作[SHAN48, SHAN49]中建立，其三十年前的结论和建议甚至在最近NBS批准方案的设计中也作为指导方针。Shannon工作的细节超出了本综述的范围。他的一些主要结论可以总结如下：

密码分析之所以可能，是因为存在冗余度；因此，加密前进行数据压缩可以增强安全性。对于每个密码系统，都可以关联一个正整数 N_0，使得在最坏情况下，一个长度为 N_0 的单一密文唯一地确定了正在使用的密钥。Shannon称这个参数为系统的唯一解距离，并且他已经证明它可以近似等于密钥的先验不确定性与语言每字符冗余度的比值。因此，一个唯一解距离为 N_0 的密码系统，如果用于加密 N_0 个或更多字符，则变得不安全。

尽管信息论模型的基本假设之一是密码分析者拥有无限的计算资源，但Shannon充分认识到所谓密码系统工作因子的实际重要性，即密码分析的复杂性与解密复杂性的比率。在某种意义上，正是这颗种子孕育了现代方法的成长，这种方法将安全性建立在密码分析的复杂性而非含糊性之上。Shannon通过他称之为"混淆与扩散"来增加工作因子的建议在业界得到了很好的遵循，并且它是许多当代系统（包括DES）的基石。其主要原理在第2节讨论。我们以对主要经典密码的简短回顾来结束本节。在接下来的所有示例中，我们使用26个字母的英文字母表，当涉及算术运算时，使用数值对应关系 $\mathrm{A} \leftrightarrow 0$，$\mathrm{B} \leftrightarrow 1$，$\mathrm{C} \leftrightarrow 2$，…，$\mathrm{Z} \leftrightarrow 25$。

1.1 凯撒密码

最初，凯撒密码只包含一个密钥，其对应的唯一变换是：

$$\mathbf{E} : M \to M + 3 \pmod{26},$$

$$M = 0, 1, \ldots, 25.$$

后来它被推广为指代具有26个密钥的密码，$0 \leq K \leq 25$，对应字母表的26个循环移位：

$$\mathbf{E}_K : M + K \pmod{26}.$$

由于密钥数量如此之少，穷举密码分析使得这个密码完全不安全。

1.2 简单替换

简单替换是一种允许使用英文字母表的任何排列作为逐字母替换密钥的密码，因此包括凯撒密码作为其一个特例。例如，

这里的密钥数量是 $26! > 4 \cdot 10^{26}$，这足以消除穷举密码分析的可行性。然而，对英语语言的统计分析显示出相当高的冗余度，大约每字母3.2比特，在此密码的均匀密钥分布下，导致唯一解距离 $N_0 = 28$ 个字母。尽管这个理论推导的数字很小，但它甚至比Friedman[FRIE67]提到的实际破解点数字25还要大。简单替换密码的密码分析很快，因为它保留了在典型英语文本中高度非均匀的字母频率分布。关于此密码及随后描述的密码的密码分析技术的更详细描述，读者可参考Sinkov [SINK68]。

1.3 多表替代密码

多表替代密码使用一个周期为 n 的替换字母表序列，通过平滑语言统计特性（同一密文字母可能在一处代表频繁的明文字母，在另一处代表不频繁的明文字母）显著增强了安全性。同时，可能的有效密钥数量从26!增加到 $(26!)^n$。

多表替代的一个非常流行的简化版本是维吉尼亚密码，它使用一个凯撒型替换的周期序列作为密钥，通常由某个易于记忆的有意义的关键短语定义。例如，关键短语BEST MAN定义了将整数序列1, 4, 18, 19, 12, 0, 13（周期重复）模26加到代表明文的整数序列上的变换。周期为 n 的维吉尼亚密码的密钥数量是 26^n，这是具有相同周期的一般多表替代密码可用密钥数量的一小部分。然而，当 $n \to \infty$，或者在实际情况下，当周期长度与消息长度相同时，人们就得到了所谓的"一次一密"，或称Vernam密码，它提供了可证明的完美安全性[SHAN49]。华盛顿和莫斯科之间的"热线"使用一次一密，并且据信[GARD77]这也是苏联特工被允许加密其消息的唯一方法。

在预先准备和分发一次一密垫以及其最低效地利用密钥符号方面涉及的巨大密钥管理问题，使得它们仅适用于高度敏感的通信。

1.4 换位

换位是一种作用于长度为 n 的单词（分组）的分组密码，使用 $n!$ 个位置排列作为密钥来转置单词的字母。加密时，首先将明文字符流分割成 n 个字母的单词，然后依次对每个单词应用特定密钥规定的换位。例如，如果 $n = 5$，使用的密钥是

1 2 3 4 5

4 3 1 5 2

那么单词的第一个字母变成第四个，第二个变成第三个，依此类推；明文TOP SECRET CIPHER，在解析为TOPSE CRETC IPHER后，被转换为密文

PEOTSECRCTHRPIE。

此密码也保留了单个字母的频率分布，但与简单替换不同，它破坏了语言的二连字母、三连字母和更高阶的统计特性，这从某种意义上说[HEll 77]使其成为比简单替换更好的自然语言密码。

1.5 乘积密码

虽然对于实际可行的 n 值，上述每个密码本身都相当弱，但是当这些密码被适当组合并迭代足够多次时，它们可能会提供非常高的安全性。当用密码 T 加密的明文被密码 S 再次加密时，最终的密文可以看作是乘积密码 $R = ST$ 加密的结果。乘积密码在工作因子（以及由此带来的安全性）增益方面的巨大潜力已被Shannon[SHAN49]充分认识到，并自此成为当代密码系统设计的主要原则。该原则的最终效用归功于这样一个事实：作为低成本构建块的简单替换和换位这两种非常基本的方案可以被高效地组合起来，形成像DES这样非常复杂的密码。

2. 当代密码的基本原理

几乎所有当代密码的安全性都基于破解它们所需的工作量，而非统计不确定性。假设是密码分析者拥有足够的辅助信息，如果能负担得起穷举搜索，就能唯一地确定密钥。例如，在Shannon模型中，这意味着密码分析者拥有的密文数量超过了唯一解距离。如今[DIFF76]认为，拥有几乎无限的密文是密码分析者可能掌握的最低级别的辅助信息。

系统在已知密文情况下受到的威胁称为唯密文攻击，任何在此攻击下成功的方案都完全无用。更现实的方法是假设密码分析者拥有相当数量的对应明文和密文。在这种情况下，系统面临已知明文攻击的威胁，这是一种强大得多的威胁。除非"物理获取"密钥，否则最终的威胁是选择明文攻击，即假定密码分析者能够获得他选择的对应明文和密文。在所有攻击中，都假定密码分析者完全了解被攻击系统的结构。

在当下，根据密码抵抗选择明文攻击的能力来评估密码是最合适的。根据Hellman等人[HELL76]的说法，NBS在测试DES时应用了威力较弱的已知明文攻击。由于密码认证仍然是启发式而非严格的程序，因此应该依赖更保守的密码强度估计。

在本节余下部分，我们回顾当前使用的私钥密码系统为应对上述威胁而采用的主要原理和实现技术。最新的公钥概念和建议的实现方案在第3节描述。由于当前大多数数据以二进制形式存储、传输和处理，我们在不失一般性的情况下，将注意力限制在二进制 $\{0,1\}$ 字母表上的密码。因此，明文和密文都以比特串的形式呈现。当明文字符串的加密逐位进行时，相应的密码称为流密码；否则，明文字符串首先被分解为单词，然后逐词加密。最常见的是，由于同步和缓冲考虑，单词具有固定长度 n，在这种情况下，相应的方案称为分组密码。

2.1 流密码

当代流密码是维吉尼亚密码的精炼现代版本。不使用相对较短、易于记忆的关键短语，而是使用密钥流生成器，通常是反馈移位寄存器，其可控的初始状态和反馈接线作为紧凑的加密密钥。假定密码分析者知道密钥流生成器的固定特性，因此一旦他得知紧凑密钥，就能产生整个密钥流。在许多情况下，足够长的密钥流子串唯一地确定了密钥流的其余部分，这使得流密码容易受到已知明文攻击，除非特别小心地使密钥流比特成为其前驱比特的非常复杂的函数。这排除了任何线性生成的密钥流，包括由低成本 n 级移位寄存器生成的极具吸引力的最大长度 $(2^{n} - 1)$ 伪随机序列。一段长度为 $2n$ 的已知明文足以破解此类密码[GEFF73]。

设计安全流密码（以及分组密码）的主要指导原则是Shannon的扩散和混淆原理。扩散要求尽可能长地散布或"消散"密钥流变量之间的相关性和依赖关系，以最大化密码分析所需的明文长度。混淆要求使相关变量之间的函数依赖关系尽可能复杂，以最大化密码分析所需的时间。最近已发表了几种根据此原理生成密钥流的方案[GEFF73, PERL76, PLES77]。其中最新的是Vera Pless[PLES77]的方案，该方案有趣地应用了J-K触发器网络来实现此目的。

2.2 分组密码

典型的当代分组密码的基本结构是一种迭代的乘积密码，以换位和简单替换为主要组成部分[Feis73]。典型的分组长度范围从32到128比特，通常是8的倍数。DES的分组长度是64比特。密钥大小通常等于分组长度；当密钥较短时，如DES的56比特密钥，会人为地将其扩展以适应整个分组长度。

系统的换位组件很容易为整个分组长度以所谓的P盒（P代表置换）形式实现，如图2所示，$n = 16$。然而，在整个分组长度字符上实现简单替换的全部功能是不可行的。长度为 n 的分组可以代表 2^{n} 个不同字符中的任何一个，允许 $2^{n}!$ 种不同的替换线路。因此，加密方案的替换阶段是在分组的小段（通常是4比特）上并行执行的。

这种4比特替换盒或"S"盒的内部结构如图3所示。它由三个阶段组成：第一阶段是二进制到十进制的转换器；第二阶段是对十进制数进行置换的P盒；第三阶段是十进制到二进制的转换器。

一个典型的 $n = 16$ 的乘积密码，由交替的P盒和S盒层组成，如图4所示。P盒通常是固定的或不带密钥的，其唯一功能是提供扩散。S盒则是带密钥的，用于执行非线性替换从而实现混淆。图4示例的16比特密钥将使得每个S盒由两个比特控制，从而允许每个S盒执行四种不同的依赖于密钥的替换。IBM Lucifer是一个按照这些通用指南设计的128比特分组密码[SMIT71, FEIs73]。它在所有层都使用一个固定的 $\mathbf{P}$ 盒，并且为了保证高度的混淆，每个S盒由单个密钥比特控制，该比特在两个精心预选的非线性替换之间进行选择。（一个完全通用的S盒可能会意外地以某种损害系统安全的方式被设定密钥。）

2.3 数据加密标准

DES[NBS77]是官方（1977年1月采纳）的国家标准局（NBS）方案，供联邦部门、机构及其他方用于计算机数据的密码保护。它是IBM开发的、经过缩减修改的Lucifer版本，具有64比特数据分组和56比特密钥。为了将密钥提升到完整分组大小，人为地将其扩展以获得64比特的"KEY"，即通过选择每个8比特字节的最后一位来确保该KEY字节具有奇校验，从而提供一定程度的错误检测。

加密时，明文分组首先经过初始置换IP，然后置换后的分组经过一个复杂的依赖密钥的计算，该计算由16个功能相同的迭代组成，最后通过将第16次迭代的输出进行初始置换的逆 $\mathbf{IP}^{-1}$ 操作来获得密文分组。

依赖密钥计算的第 i 次迭代的输入是 L_{i-1}、R_{i-1} 和 K_i，$1 \leq i \leq 16$，其中 L_0 和 R_0 分别是IP置换后的明文分组的左右32比特半部分；K_{i} 是 i 和 KEY 的一个48比特函数，$K_{i} = \mathrm{KS}(i, \mathrm{KEY})$；并且

$$L_t = R_{t-1}, \quad R_t = L_t \oplus f\left(R_{t-1}, K_t\right),$$

其中 $\oplus$ 表示模2加法。

函数 f 是该方案的核心，它是一个非线性的、多对一的替换。f 的32比特输出获取方式如下。首先，将32比特的 R_{t-1} 扩展为48比特的 $\mathbf{E}(R_{t-1})$；接着，K_t 和 $\mathbf{E}(R_{t-1})$ 逐比特模2相加，其和 Q 被划分为 $Q = Q_1Q_2\cdots Q_8$，其中每个 Q_j，$1 \leq j < 8$，长度为6比特；然后，每个 Q_j 被送入一个固定的非线性、6输入4输出的替换盒 S_j；最后，八个 S_j 盒组合而成的32比特输出被送入一个P盒，其输出即为 f。有关IP、IP^{-1}、KS、E、S_j 盒和P盒的完整细节，读者可参阅NBS77。

解密时，密文分组经过完全相同的程序，但 K_{i} 以相反顺序应用，即从 K_{16} 开始到 K_{1} 结束。这样，第一个解密步骤IP将撤销最后一个加密步骤 $\mathbf{IP}^{-1}$；L_{i}, R_{i} 对将按照以下公式以相反顺序重新生成：

$$R_{t-1} = L_t, \quad L_{t-1} = R_t \oplus f(L_t, K_t);$$

最后一个解密步骤 $\mathbf{IP}^{-1}$ 将撤销第一个加密步骤 $\mathbf{IP}$，并得到原始明文分组。

如引言中所述，DES的采纳引发了一场尚未解决的争议。批评者提出的主要观察和建议如下[DIFF77, MORR77]：

1) 到20世纪80年代，一天内完成穷举搜索的机器将是可行的。（1976年8月的NBS研讨会估计，此类机器的最早交付日期是1990年。）
2) 密钥大小应至少增加到64比特，最好128比特。
3) 迭代次数应加倍。
4) 应使用公开设计的S盒。
5) 不应采纳提议的DES，但如果采纳，应使用多重加密。

自这些激烈争论提出以来，大约三年过去了。DES现在已是官方标准，而解决争议的将是时间而非理由。无论它是否服务于其他目的，这场争议无疑是刺激密码学兴趣复兴以及过去两年取得新成果的一个因素。

3. 公钥密码系统

公钥密码系统的概念由Diffie和Hellman于1976年提出[Diff76]，他们设想了一种用于私人通信的系统，该系统采用一个公共目录，每位订阅者在该目录中放置一个程序 $\mathbf{E}$，供其他订阅者用于加密发送给他的消息，同时保密其相应的解密程序 $\mathbf{D}$。要使这样的系统可行，必须有一种简单的方法让每位订阅者生成自己的 $\mathbf{E}$ 和 $\mathbf{D}$ 程序。此外，从作为消息集 $\{M\}$ 和密文集 $\{C\}$ 上的运算符来看，这些程序必须具备以下特性：

i) 如果 $C = \mathbf{E}(M)$，则 $M = \mathbf{D}(C)$ 或 $\mathbf{D}(\mathbf{E}(M)) = M$ 对每个 M 成立。
ii) $\mathbf{E}$ 和 $\mathbf{D}$ 都快速且易于应用。
iii) 公开披露 $\mathbf{E}$ 不会危及 $\mathbf{D}$。也就是说，从 $\mathbf{E}$ 推导出 $\mathbf{D}$ 在计算上是不可行的。

这种系统的存在将使从未谋面或从未通信过的订阅者之间能够进行即时安全通信。例如，如果订阅者A想向订阅者B发送一条私人消息 M，他会在目录中B名下查找 $\mathbf{E}_{\mathbf{B}}$，并在公开场合传输 $C = \mathbf{E}_{\mathbf{B}}(M)$。根据iii)，只有B能通过对其秘密的 $\mathbf{D}_{\mathbf{B}}$ 应用于 C 来解密 C。

为了获得数字签名功能，Diffie和Hellman提议使用具有以下附加性质的E-D对：

iv) $\mathbf{D}$ 可以应用于每个 M，并且如果 $S = \mathbf{D}(M)$，则 $M = \mathbf{E}(S)$ 或 $\mathbf{E}(\mathbf{D}(M)) = M$ 对每个 M 成立。

当iv)成立时，订阅者A可以"签署"他发送给订阅者B的消息，首先计算其消息相关签名 $S = \mathbf{D}_{\mathrm{A}}(M)$，然后计算密文 $C = \mathbf{E}_{\mathrm{B}}(S)$。现在，只有B能通过计算 $\mathbf{D}_{\mathrm{B}}(C) = \mathbf{D}_{\mathrm{B}}(\mathbf{E}_{\mathrm{B}}(S)) = S$ 从 C 恢复 S，并且当他计算

$\mathbf{E}_{\mathrm{A}}(S) = \mathbf{E}_{\mathrm{A}}(\mathbf{D}_{\mathrm{A}}(M)) = M$ 时，B可以确信 M 来自A，因为其他人无法应用A的秘密 $\mathbf{D}_{\mathrm{A}}$ 来计算 $S = \mathbf{D}_{\mathrm{A}}(M)$。

如上所述的数字签名起到了双向身份验证的作用。除了与消息相关外，S 还与签名者相关。虽然 B 可以确信接收到的消息确实来自 A，但通过签署他的消息，A 可以确信没有人能够将他未发送的消息归因于他。S 的双重依赖性也保护了签名不被附加到虚假消息上。

虽然Diffie和Hellman在他们出色的文章[DIFF76]中完整地描述了这些优雅的概念，但他们没有给出一个实用的公钥密码系统实现。然而他们指出，任何此类实现的方法都涉及计算困难的问题，例如所谓单向函数的求逆。自那以后发表的各种实现[RIVE78a, MERK78b, McEl78]都基于此类问题。如果函数 f 是可逆且易于计算的，但对于 f 定义域中几乎所有的 x，从 $y = f(x)$ 求解 x 在计算上是不可行的，则称 f 是单向的。换句话说，从 f 的完整描述计算 f^{-1} 在计算上是不可行的。

如果一旦知道某些私有的"陷门"信息，f^{-1} 就容易计算，而不知道这些知识时 f 是单向的，则称 f 为陷门单向函数。

显然，任何陷门单向函数 f 及其逆 f^{-1} 都可以作为公钥密码系统的E-D对。当且仅当 f 也是消息集 $\{M\}$ 上的置换时，数字签名功能才会成为系统的一部分。

Diffie和Hellman[DIFF76]以及独立地Merkle[MERK78a]也引入了公钥分发系统的概念。这种系统的目的是使每对订阅者能够在不安全的信道上安全地交换私钥，以用于像DES这样的传统密码系统。为此，作者提出了以下实现方案。

Diffie和Hellman方法的安全性基于计算模素数的对数的难度。给定一个素数 q 和一个整数 X，$1 \leq X \leq q - 1$，可以计算

$$Y = \alpha^X \pmod{q},$$

其中 α 是 $\mathrm{GF}(q)$ 的一个固定本原元，最多使用 $2\log_2 q$ 次乘法（见Knut69, pp. 398-422）。另一方面，对于精心选择的素数 q，目前已知最好的取对数方法（即从 Y 计算 X）大约需要 $q^{1/2}$ 次运算[PoHL78]。现在，每个用户A从集合 $\{1,2,\ldots,q-1\}$ 中随机选择一个数 X_{A}，并计算 $Y_{A} = \alpha^{X_{A}} \pmod{q}$。在保密 X_{A} 的同时，他将 Y_{A} 放入公共目录。当用户A和B想要通信时，他们使用 $K_{AB} = Y_{A}^{X_{B}} = Y_{B}^{X_{A}} = \alpha^{X_{A}X_{B}} \pmod{q}$ 作为他们的私钥。虽然A和B各自都能使用自己的秘密 X 和对方的公开 Y 轻易计算 K_{AB}，但拦截者C必须从 Y_{A} 和 Y_{B} 计算 K_{AB}。由于 $K_{AB} = Y_{A}^{\log_a Y_B} \pmod{q}$，该系统破解的难度至多与计算模 q 对数的难度相同。这两个问题是否确实等价仍未解决。

Merkle的方法基于"谜题"概念。谜题是来自一个可以在可行步骤数 $O(n)$ 内破解的密码系统的密文。然而，数字 n 应该足够大，以使需要 $O(n^2)$ 步的计算变得不可行。每个用户发布（或根据请求传输）n 个谜题，每个谜题"封装"一条由两部分组成的消息：一个随机分配的、唯一标识谜题的ID，以及一个从预定密钥空间中随机选择的密钥。当用户A想与用户B通信时，他随机选择B的一个谜题，以 $O(n)$ 步破解它，在公开场合向B传输该谜题的ID，并使用对应的密钥加密他发送给B的消息。如果第三方C希望找出那个密钥是什么，他唯一的方法是随机顺序逐个解决B的谜题，直到拦截到的ID被解出。C的预期计算步骤数显然是 $O(n^{2})$。

更传统的用于DES等密码系统的密钥生成和分发方法最近发表在IBM系统杂志的一个密码学特辑中[EHRS78, MATY78]。我们现在继续描述现有的公钥密码系统实现方案。

3.1 Rivest-Shamir-Adleman (RSA) 方案

RSA方案[RIVE78a]中的消息和密文以0到 $n - 1$ 之间的整数的形式出现。（任何将数据块表示为整数的标准方法都可以。）每个用户选择自己的 n 和另一对正整数 e 和 d，选择方式如下所述。用户放在公共文件中的加密密钥由数对 (n, e) 组成。相应的解密密钥由数对 (n, d) 组成，其中 d 部分保密。对于消息 M 的加密算法 $\mathbf{E}$ 和对于密文 C 的解密算法 $\mathbf{D}$ 是：

$$\mathbf{E}(M) = M^e \pmod{n},$$
$$\mathbf{D}(C) = C^d \pmod{n}.$$

（注意，两种操作都易于执行，并且加密和解密的结果都是0到 $n - 1$ 之间的整数。）

选择整数 e 和 d 以满足：

$$(X^e)^d = (X^d)^e = X^{ed} = X \pmod{n}$$

对于0到 $n - 1$ 之间的每个整数 X 都成立。因此，

$$\mathbf{E}(\mathbf{D}(X)) = \mathbf{D}(\mathbf{E}(X)) = X \pmod{n}$$

对于所提及范围内的所有 X 成立。

RSA方案的安全性取决于分解大数的难度。这种难度引入的方式如下。每个用户（私下）使用某种随机选择过程和最近发表的高效素性测试[MILL75, RABI76, SOLO77]选择两个大素数 p 和 q。然后整数 n 取为（秘密的）p 和 q 的乘积。接下来，从小于且与 $\phi(n) = (p - 1)(q - 1)$ 互素的整数集合中随机选取一个数 $d > \max\{p, q\}$。最后，使用欧几里得算法的一个变体（见KNUr69练习4.5.2.15），从 $\phi(n)$ 和 d 计算整数 e，$0 < e < \phi(n)$，使其成为 d 模 $\phi(n)$ 的乘法逆元。即，

$$e \cdot d = 1 \pmod{\phi(n)}.$$

Rivest, Shamir和Adleman[RIVE78a]用一个例子说明了他们的方案：$p = 47$，$q = 59$，$d = 157$。这些素数产生 $n = 47 \cdot 59 = 2773$，$\phi(n) = 46 \cdot 58 = 2668$，和 $e = 17$。由于 $n > 2626$，我们可以使用替换：空格 $= 00$，$A = 01$，$B = 02$, \ldots, Z = 26$，每块编码两个字母。使用这种数字表示，每个块为四位数字，消息IT IS ALL GREEK TO ME变成：

$$0920 \quad 1900 \quad 0112 \quad 1200 \quad 0718$$

$$0505 \quad 1100 \quad 2015 \quad 0013 \quad 0500.$$

由于二进制 $(e) = 10001$，每个块 M 的加密可以通过五次乘法完成：$M^{17} = ((M^2)^2)^2 \cdot M$。对于第一个块，我们得到 $920^{17} = 948$ (mod 2773)，整个消息被加密为：

$$0948 \quad 2342 \quad 1084 \quad 1444 \quad 2663$$

$$2390 \quad 0778 \quad 0774 \quad 0219 \quad 1655.$$

通过将每个密文块提升到 $d = 157$ 次幂模2773，可以再现明文。

现在，给定公开的 n 和 e，尝试破解密码的一种方法是将 n 分解为 p 和 q，计算 $\phi(n) = (p - 1)(q - 1)$，并使用欧几里得算法从 $\phi(n)$ 和 e 计算 d。除了分解部分，所有这些都相当简单。使用已知最快的方法（R. Schroeppel，未发表）分解一个200位的 n（Rivest等人[RIVE78a]推荐的尺寸）所需的操作次数超过 10^{23}，对于一台每秒 $1 - \mu s$/操作的计算机，这将需要超过 $3 \cdot 10^9$ 年才能完成。

人们可能会考虑其他不依赖于分解的方法来破解该方案。其中一种方法曾在Cryptologia[SIMM77]中被提出，但该方案的发明者已做好充分准备予以反驳[RIVE78b]。最近，有人（M. Rabin，未发表）证明，使用任何方法破解一个稍微更通用的方案实际上等价于分解。这一最新结果使RSA方案处于相当安全的地位，只要分解问题仍然困难。

应该指出，与Diffie和Hellman公钥分发方案类似，破解RSA方案也可以被视为计算模对数的问题，但增加了额外的复杂性，即模数是合数而非素数。关于选择和生成素数 p 和 q 以及整数 d 和 e 的方法的细节，读者可参阅RIVE78a。

3.2 Merkle-Hellman (MH) 方案

MH方案[MERK78b]中的消息是二进制 n 维向量，$M = (b_{1}b_{2}\dots b_{n})$，$b_{j} \in \{0,1\}$，公开加密密钥是一个所谓的陷门背包 n 维向量 $A = (a_{1}a_{2}\dots a_{n})$，其中 a_{j} 是按照下述方式选择的正整数。对于消息 M 的加密算法是：

结果是一个介于0和 $a = \sum_{j=1}^{n} a_{j}$ 之间的整数 c。相应的密文 C 是 c 的标准二进制表示，$C = \text{binary}(c)$，使用 $\lceil \log_2(1 + a) \rceil$ 比特，其中 $\lceil x \rceil$ 表示大于等于 x 的最小整数。

虽然从 M 计算 C 很简单，但反过来对于某些精心选择的背包向量 A 来说，除非知道内置的陷门，否则显然非常困难。在我们进一步阐述密码分析的困难之前，我们先描述生成陷门背包的方法以及隐含的解密算法。

陷门信息或秘密解密密钥是一对大的正整数 w 和 m，它们满足：

i) w 小于 m 且与 m 互素；也就是说，w 有一个模 m 的乘法逆元 w^{-1}，$0 < w^{-1} < m$。
ii) 如果 $\hat{a}_j = w^{-1}a_j$ (mod m)，那么

并且

很容易验证i)和ii)隐含了以下简单的解密过程。给定 $C = \mathrm{binary}(c)$，

1) 从 C 计算 c；
2) 计算 $\hat{c} = w^{-1}c$ (mod m)；
3) 根据以下规则计算 $M = (b_{1}b_{2}\dots b_{n})$，$b_{j} = 0, 1$：
  3.1) 当且仅当 $\hat{c} \geq \hat{a}_n$ 时，设 $\hat{b}_n = 1$；
  3.2) 对于 $j = n - 1, n - 2, \ldots, 1$，当且仅当

时，设 $b_{j} = 1$。

Merkle和Hellman用一个例子说明了他们的方案：$n = 5$，$m = 8443$，$w = 2550$，以及 $\hat{A} = (171, 196, 457, 1191, 2410)$。那么陷门背包由 $A = w\hat{A} = (5457, 1663, 216, 6013, 7439)$ (mod 8443)给出，并且 $w^{-1} = 3950$ (mod 8443)。给定和

解密过程首先计算：

(mod 8443)。

由于 $\hat{c} >\hat{a}_5$，我们设 $b_{5} = 1$。然后按照3.2)的规则，我们确定 $b_{4} = 1$，$b_{3} = 0$，$b_{2} = 1$，以及 $b_{1} = 0$。

Merkle和Hellman建议使用长度 $n \geq 100$ 的背包向量。对于 $n = 100$，他们建议 m 均匀地从 $2^{201} + 1$ 到 $2^{202} - 1$ 之间的整数中选择；a_j 均匀地从 $(2^{j-1} - 1)2^{100} + 1$ 到 $2^{100+j-1}$ 之间的整数中选择，$j = 1, 2, \ldots, n$；以及 $\hat{w}$ 均匀地从2到 $m - 2$ 之间的整数中选择，然后反复除以 $\gcd(\hat{w}, m)$ 以获得满足 $\gcd(w, m) = 1$ 的 w（gcd代表最大公约数）。然后根据以下公式计算 a_j：

并将其放入公共文件。这些选择确保i)和ii)得到满足，并且 a_{j} 具有随机选择的整数集的外观。

为了增强安全性，Merkle和Hellman建议将陷门隐藏变换 $a_{j} = w\hat{a}_{j}$ (mod m) 迭代几次，每次使用不同的 w 和 m 对。他们还提出了他们方案的"乘法背包"版本，并建议在生成迭代陷门背包时结合这两个版本。然而，每次迭代都会略微增加分组长度，这导致该方案不具有"满射"性质；也就是说，并非密文范围内的每个整数 c 都可以作为 a_{j} 的某个子集的和获得。因此，MH方案不具备简单的数字签名功能。在他们引用的文章中，作者提出了一种修改的签名程序，对于足够密集的"进入"映射，在修改的签名可以执行之前需要进行有限次数的尝试。一种生成具有修改签名功能的密集背包的改进方法正在准备中[MERK78c]。

MH方案的安全性基于所谓背包问题的难度，在当前背景下，可以表述为确定给定的一组 n 个正整数的某个子集，使得子集成员的具有规定和值的问题。这个问题属于一类称为NP完全（参见第5节或KARP72, AHO74）的难题。然而，有三重阻碍使得人们不能仅仅基于背包问题是NP完全的而接受MH方案是安全的。首先，尚未证明带有MH陷门的问题仍然是NP完全的；其次，成为NP完全类成员的资格是在最坏情况下确定的；第三，目前尚不完全清楚NP完全问题的最坏情况样本到底有多难。下一节的讨论和例子将进一步阐明这些观点。

3.3 McEliece 方案

McEliece方案[McEL78]基于线性纠错码的一般解码问题的难度，这个问题虽然长期以来被认为很难，但直到最近才被证明[BERL78]是NP完全的。公开加密密钥是一个秩为 k 的 $k \times n$ 二进制矩阵 G，作为一个表面上任意的 (n, k) 线性码的生成矩阵。实际上，$G = S\hat{G}P$，其中 $\hat{G}$ 是一个易于解码的Goppa码[McEl77, Ch. 8]的生成矩阵，S 是一个随机的 $k \times k$ 二进制非奇异加扰矩阵，P 是一个随机置换矩阵，用于掩盖 G 的代数结构。码长为 $n = 2^m$，其中 m 足够大，码的维数为 $k \geq n - mt$，其中 t 是该码的纠错能力。密文消息是二进制 k 维向量，对于消息 M 的加密算法是：

其中 Z 是一个本地生成的、长度为 n、重量为 t 的随机向量。（注意，生成的密文长度为 n。）

秘密陷门信息包括矩阵 P 和 S 的逆 P^{-1} 和 S^{-1}。对于密文 C 的相应解密过程很容易执行如下：

1) 计算 $\hat{C} = CP^{-1}$。（注意，$\hat{C}$ 是由 $\hat{G}$ 生成的Goppa码的一个码字。）
2) 应用Patterson算法[PATT75]将 $\hat{C}$ 解码为 $\hat{M} = MS$。（操作次数为 $O(nt)$。）
3) 计算 $M = MS^{-1}$。

McEliece加密方案只覆盖了密文空间的一小部分，因此缺乏签名功能。

在讨论该方案的安全性时，McEliece认为最有希望的攻击方法是随机选择 n 个密文比特中的 k 个，希望这些比特中没有错误（因为 Z 有 t 个非零比特），并在可能的情况下，通过求 G 的对应 $k \times k$ 子矩阵的逆来计算 M。对于 $n = 1024$，$k = 524$，$t = 50$，McEliece计算出找到 M 所需的预期操作次数约为 10^{19}。对于相同的参数，他计算出能够"签署"一个随机选择的长度为1024的向量的概率仅为约 2^{-215}，这是一个非常小的比例。

3.4 Graham-Shamir (GS) 方案[2]

GS方案是MH方案的一个变体，本质区别在于隐藏的、易于求解的背包 $\hat{A} = (\hat{a}_1\hat{a}_2\dots \hat{a}_n)$。在十进制表示中，每个背包数 $\hat{a}_{j}$ 的形式为：

其中 R_{j1} 和 R_{j2} 是50到100位的随机数，0^{t} 是 t 个零的字符串，I_{j} 是一个 n 元组，在第 j 个位置为1，其余位置为0。整数 t 被选为将每个 R_{j2} 乘以九（消息是十进制的 n 元组）并将 n 个乘积 $9 \cdot R_{j2}$ 相加所产生的"进位溢出"的位数。很容易验证，在这些条件下，$\hat{c} = \underline{\boldsymbol{A}} \overline{\boldsymbol{M}}^{\mathrm{T}}$ 的十进制表示在 $\hat{a}_{j}$ 表示中单位向量部分 (I_{j}) 占据的 n 个位置上包含了消息 M。

与MH方案类似，GS方案的公开背包 $A = (a_{1}a_{2}\dots a_{n})$ 是通过一次或多次 $a_{j} = w\hat{a}_{j}$ (mod m)，$j = 1, 2, \ldots, n$ 形式的迭代获得的，其中 $m > 9 \cdot \sum_{j=1}^{n} \hat{a}$，且 $\gcd(w, m) = 1$。Graham和Shamir认为，简单背包 $\hat{A}$ 分量的随机侧翼 (R_{j1} 和 R_{j2}) 使得他们的公开背包 A 能够获得比MH方案可能更高的伪装程度。（注意，从每个 a_{j} 中删除 R_{j1} 部分会得到一个其分量满足简单MH背包的强递增性质ii)的背包。）

3.5 仅签名方案

公共仅签名（PSO）方案采用公共签名目录来提供不可伪造的双向身份验证，但不提供隐私。也就是说，PSO方案保护消息发送者免受预期接收者或冒名顶替者的伪造；它保护每个消息接收者免受发送者否认或冒名顶替者植入消息，但它需要额外的手段来防止拦截者窥探已签名消息的内容。

尽管本文主要关注提供隐私（无论是否有身份验证）的系统，但我们简要概述一下最近发表（1978年7月）的Adi Shamir的PSO方案[SHAM78]，因为它突显了隐私和身份验证之间一种似乎固有的权衡。MH和GS方案通过使用可逆（但非满射）的陷门背包实现隐私，而Shamir则使用满射（但非可逆）的陷门背包来实现身份验证。他的PSO方案工作方式如下。

每个订阅者将一个长且看似随机的背包向量 $A = (a_{1}a_{2}\dots a_{2k})$ 放入一个公共文件，供其他订阅者用于验证其签名消息。消息 m 是介于0和 $n - 1$ 之间的整数，其中 n 是一个 k 比特的数。秘密陷门信息是一个 $k\times 2k$ 的随机二进制矩阵 R，该矩阵与 A 的选择相协调以满足（细节见SHAM78）：

其中 $B = (2^0 2^1 \dots 2^{k-1})$。

为了签署消息 m，首先将其随机化以获得：

其中 Q 是一个临时生成的 $2k$ 比特随机二进制向量。接下来，使用 $\hat{m}$ 的二进制表示 $\hat{M}$，我们可以写：

其中 $\hat{C} = \hat{MR}$。最后，我们有：

其中 $C = \hat{C} +Q$ 是 m 的签名。

(m, C) 对的真实性可以通过对照公开向量 A 检查等式 $m = CA^{\mathrm{T}}$ 来轻松验证。如果给定 m 和 A 求解此等式得到 C 确实是不可行的，那么该签名是万无一失的。当然，该PSO方案提供的实际安全性尚未确定。关于此问题的详细讨论以及增强该方案安全性的方法，读者可参阅SHAM78。

该方案的一个微妙之处在于，它在完全牺牲隐私的情况下提供了不可伪造签名的安全性。当可逆线性变换（如一对一背包方案或McEliece方案所提供的）的密度增加时，其求逆难度降低。在极限情况下，当变换是满射时，问题就简化为求非奇异矩阵的逆。为了使问题保持困难，变换必须变成奇异的或非唯一可逆的，从而变得对隐私目的无用，因为它不能保证无歧义解密。

因此，值得怀疑的是，MH、GS和McEliece方案是否应该被勉强用于达到一种使签名在统计上可行，但可能会使隐私妥协到无法容忍的程度的密度。相比之下，RSA方案基于非线性变换，尽管变换既是可逆的又是满射的，但破解它显然是一个困难的问题。

4. 论密码复杂性的复杂性

当前实践的密码学（无论是DES还是新的公钥方案）的主要缺点之一是，缺乏证据证明这些方案确实如它们被认为（或声称）的那样难以破解。DES的发明者和赞助者将其强度证据归结为花费在试图破解它上的许多人/计算机年，而新的公钥密码的发明者则引用了密码分析者利用目前公开已知的最佳算法进行因式分解、整数背包填充、线性代数码解码等计算任务的难解性。（我们强调"公开"一词，因为如果某个有保密动机的组织拥有一个可行的算法，比如，分解200位整数，它几乎没有公开此事实的意愿，更不用说发表算法了。）

虽然参与DES争议的各方在具体日期上存在分歧，但他们都同意，对于资金充足且有动机的团体来说，当前版本的DES变得容易受到攻击只是几年（五到十年，取决于哪一方）的时间问题。鉴于公钥密码系统的重要性和深远影响，问题在于近期提出的实现方案是否会屈服于高度积极的数学界（学术或其他）的一致努力。

真正的任务和挑战是提出适当的密码安全标准以及相应的密码复杂性度量，以取代当前对密码"认证"模糊概念的依赖，转而采用更严格的不可破译性断言。

该领域的许多工作者都强烈认为，研究密码复杂性的适当框架与已有十年历史的组合复杂性理论密切相关。该理论的一个主要成就是认识到，许多吸引研究者兴趣和努力多年的、表面上无关的困难工程问题，从计算复杂性角度来看，都可以归为一类彼此可相互归约的等价问题。

如果存在一个确定性算法，可以在以"问题长度"（例如，以二进制表示问题参数所需的比特数）的某个多项式为界的计算时间内解决该问题的每个样本，则称该问题属于类 P。

在许多有趣的问题中，迄今为止尚未找到这样的算法（并非缺乏动机），例如旅行商问题、图着色问题、背包填充问题以及许多其他问题[KARP72, AHO74]。所有这些都属于一个称为NP（非确定性多项式）的类，其定义特性是，对于NP中的每个问题，都存在一个非确定性算法（即具有无限并行性）可以在多项式时间内解决该问题的每个样本。另一种说法是，对于NP中的每个问题，都存在一个确定性算法，对于该问题的每个样本，可以在多项式时间内检查猜测解的正确性。

显然，NP类包含P类。组合复杂性中一个未解决的大问题是NP是否严格大于P。Karp[KARP72]确定了NP的一个子类NPC（C代表完全），其特性是，如果NPC中的任何一个问题被发现属于P，那么所有NP都属于P，意味着 $\mathbf{P} = \mathbf{NP}$。所有上述问题已知都属于NPC。

鉴于为寻找NP问题的多项式时间算法所付出的所有努力以及NPC子类的显著特性，相信 $\mathbf{P} = \mathbf{NP}$ 的人很少。难怪许多密码学家认为NPC问题是构建密码系统的良好潜在候选者。我们在下面的例子中的目标是提醒读者注意可能的陷阱，以及在检查此类模型时需要保持的谨慎。

示例：一个易于破解的NP完全密码

此示例由作者、Shimon Even和Yacov Yacobi共同推导。它展示了一个密码，即使在选择明文攻击下，破解其密钥的问题也是NP完全的。然而，给定足够的已知明文，破解密钥的概率趋近于1，并简化为求解 n 个未知数的 n 个独立线性方程组的简单问题，其中 n 是密钥比特数。

该方案是一种传统的私钥分组密码，其一般结构如图1所示。密钥长度为 n 比特，$K = (x_{1}x_{2}\dots x_{n})$，消息是长度为 m 的二进制分组，其中 $m = \lceil \log_2(1 + \sum_{j=1}^{n} a_{j}) \rceil$，且 $A = (a_{1}a_{2}\dots a_{n})$ 是一个具有正整数分量 a_{j} 的任意背包，假定密码分析者已知。为了获得消息 M 的密文 C，按如下步骤进行：

1) 在本地生成一个临时的随机二进制向量 R；
2) 计算数值 $s = A(K \oplus R)^{\mathrm{T}}$；
3) 设 $C = (M \oplus S, R)$，其中 $S = \mathrm{binary}(s)$。

注意，密文长度为 $m + n$，即 $M \oplus S$ 的 m 比特后跟 R 的 n 比特，并且对于每个 M 都会生成一个新的 R。解密也很简单：由于合法接收者知道 K，他可以将其与接收到的 R 相加并计算 s，从而得到 S 进而得到 M。

从密码分析者的角度来看，最坏的情况是所有已知明文的临时向量 R 保持不变。那么，在已知明文攻击和选择明文攻击下，破解密钥需要求解方程 $s = A(K \oplus R)^{\mathrm{T}}$ 以得到 K，已知 s, A 和 R。这当然等同于求解背包问题，该问题是NP完全的。

可能性大得多的事件是，给定足够的已知明文，密码分析者将拥有 n 对 (M_i, C_i)，其中 $C_i = (M_i \oplus S_i, R_i)$，$i = 1, 2, \ldots, n$，使得 n 个向量 $U_i = 1^n - 2R_i$ 在实数域上线性无关（1^n 是一个 n 维全1向量）。对于每个 $i = 1, 2, \ldots, m$，我们有：

其中 $\ast$ 表示分量乘法。因此，

且

令 $t_i = s_i - AR_i^{\mathrm{T}}$，并将 n 个方程写成矩阵形式，我们得到：

由于 U_{i} 线性无关且所有 j 满足 $a_{j} > 0$，我们可以很容易地解出密钥分量 x_{j}。可以证明（见LEMP78，附录1），$N \geq n$ 对 (M, C) 产生 n 个线性无关的 U_{i} 的概率下界在 $N = n$ 时约为 $\frac{1}{3}$；对于 $N = n + 1$，该下界翻倍，并且随着 $N - n$ 的增加迅速趋近于1。

此示例并非旨在以任何方式贬低难题作为密码学方案基础的潜在有用性。我们的意图仅仅是提醒读者，NP完全性的复杂性度量，以及那些未解决是否属于NPC类的难题的公认难度，可能与密码复杂性无关。通过比较Merkle-Hellman方案与我们示例中的方案，进一步强调了所涉及的复杂性。两个方案都基于背包问题，虽然目前尚不清楚破解MH方案是否属于NPC，但对于我们示例中的方案来说，这无疑是肯定的。然而，破解后者非常简单，而破解MH方案的可行方法尚未找到。

还应该指出，示例方案的大部分（如果不是全部）弱点是由于 C 对 M 的线性依赖。密码中的线性长期以来被认为是密码学家的诅咒和密码分析者的福音。在讨论具有某种线性度的方案中隐私和身份验证之间的权衡时，我们已经在第3节末尾提到了这一点。

很容易修改我们示例中的方案，使其密码分析与任何已知的难题一样困难。例如，可以用 M^s (mod pq)（如在RSA方案中）或 $\mathrm{DES}(M, S)$ 替换 $M \oplus S$，即使用 S 作为DES KEY对 M 应用DES方案的结果。我们提到这一点不是为了提出另一个其难解性仅是猜想的新方案。正如前面所建议的，我们相信需要努力建立一些真正符合密码复杂性背景的标准，然后才尝试发明符合这些标准的方案。正如RABI77中指出的，我们距离实现这个目标还很远。

致谢

作者很高兴地感谢在准备此综述过程中获得的帮助。特别感谢Martin Cohn在内容和风格上提供的宝贵帮助，感谢Martin Hellman帮助修订原始的Sperry报告，感谢Ron Graham和Adi Shamir允许描述他们未发表的方案。作者还要感谢Len Adleman、Shimon Even、Leland Gardner、Martin Hellman、Ralph Merkle、Nick Pippenger、Michael Rabin、Ron Rivest、Adi Shamir、Neil Sloane、Shmuel Winograd和Yacob Yacobi的有益讨论。

参考文献

（参考文献列表保持原样，未翻译）

专业术语翻译对照表

英文术语	中文翻译
Cryptography	密码学
Cipher	密码
Encryption	加密
Decryption	解密
Cryptosystem	密码系统
Cryptocomplexity	密码复杂性
Public-key cryptosystems	公钥密码系统
Data Encryption Standard (DES)	数据加密标准
Cryptanalysis	密码分析学
Key	密钥
Plaintext	明文
Ciphertext	密文
Classical cryptography	经典密码学
Caesar cipher	凯撒密码
Simple substitution	简单替换
Polyalphabetic ciphers	多表替代密码
Transposition	换位
Product ciphers	乘积密码
Stream ciphers	流密码
Block ciphers	分组密码
P box (Permutation box)	P盒（置换盒）
S box (Substitution box)	S盒（替换盒）
Confusion and diffusion	混淆与扩散
One-time pad	一次一密
Vigenère cipher	维吉尼亚密码
Unicity distance	唯一解距离
Ciphertext-only attack	唯密文攻击
Known-plaintext attack	已知明文攻击
Chosen-plaintext attack	选择明文攻击
Private-key cryptosystems	私钥密码系统
Trapdoor one-way functions	陷门单向函数
Public-key distribution	公钥分发
Digital signature	数字签名

英文术语	中文翻译
Rivest-Shamir-Adleman (RSA)	Rivest-Shamir-Adleman (RSA)
Merkle-Hellman (MH)	Merkle-Hellman (MH)
Knapsack problem	背包问题
McEliece scheme	McEliece 方案
Goppa code	Goppa 码
Graham-Shamir (GS) scheme	Graham-Shamir (GS) 方案
Signature-Only Schemes	仅签名方案
NP-complete	NP完全
P class	P类
NP class	NP类
NPC class	NPC类（NP完全类）
Work factor	工作因子
Redundancy	冗余度
Side information (SI)	辅助信息 (SI)
Key management	密钥管理
Authentication	身份验证，认证
Feistel structure	费斯妥结构
Linear complexity	线性复杂度
Probabilistic algorithm	概率算法
Primality test	素性测试