# 数学与互联网安全

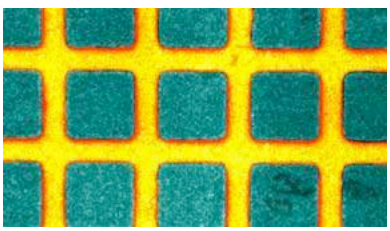
Joe Malkevitch ／文　袁晓明／译

2005 年 12 月离新年假期只有几天的时候，纽约的公交系统员工在纽约市区举行了大罢工。市民如果试图到商店去购物，可能会在路上堵上好几个小时、再走上很久的路、并且也很难拦到出租车。因此很多纽约市民选择了在网上进行购物。不过还是有很多人因为担心交易的安全性而对网上购物退避三舍。

大家心里的疑问包括：

- 网上金融交易是否安全？
- 通过互联网发送电子邮件是否受到第三方的监控从而导致邮件内容泄露？
- 网络病毒以及“流氓软件”正越来越猖狂，我们如何对付这些问题？

数学正在尽力解决上述这些问题，从而使得网络交易百分之百的安全，以及让电子邮件系统成为人们值得信赖的一种生活方式。实现这些目标的工具恰恰是一些以前被认为是毫无实用价值的数学知识！人们一直在发明新兴技术从而制造出更快更小的电脑芯片。一部分人正在致力于创造一个安全的互联网环境，而另一部分同时也在不停地试图通过互联网“提取”他人的财富（而不是在银行里排队取款！）。这是一场猫和老鼠的游戏，而这个游戏也受到电脑芯片技术的影响。



由美国国家标准与技术研究所（NIST, National Institute of Standards and Technology）开发的一项新的光刻技术，用于制造质量更好、速度更快的芯片

互联网技术涉及的数学知识十分广泛，从数据压缩和误差压缩技术，到信息传递的方法和安全设计都需要大量的数学知识。本文将着重讨论与互联网安全的几个数学问题。

## 密码学

一个战争指挥官总是希望他发给前线军官的命令不会落到敌方手中。因此，如果命令是通过书面来传递，那是十分不安全的。因为一旦命令书被敌方拦截，自己的计划就彻底暴露了。（当然要通讯员把命令背下来再进行传递也不现实，大家看看电视里的谍战片或反恐片就知道这种方式是多么不安全了。）尤利乌斯·凯撒（Julius Caesar）被认为是最早使用数学知识来对信息进行加密的人之一。据说他的加密方法是把一段文

字（也就是原始信息）的每一个英文字母都用字母表里其对应的后一个字母来代替，而字母表的最后一个字母就用其第一个字母来代替。按照这种方法，Caesar Cipher 这个短语就由 Dbftbs Djqifs 来代替。其实我们有很多种类似的加密方式来产生这样一段信息。因此，这样的加密方式足以迷惑“敌人”一段时间了。前面这个例子我们是把一段文字按照字母表往后移动 1 个位置。更一般的，我们可以按照字母表往后移动 $r$ 个位置。比如 $r = 5$ 的话，Caesar Cipher 这个短语经过加密以后就变成了 hfjxfw hnumjw。今天我们把这种加密方式都叫做凯撒密码。

但是，如果解密员碰巧想到一段文字的加密方式是把每一个英文字母按照字母表顺移同样的位置，那就算把每种可能的方式都试一遍来解密，其工作量都不大，因为毕竟只有 26 个英文字母。这个简单的例子已经说明了无论是互联网密码学还是军事密码学，“复杂度”和安全性两者之间都存在着有趣的联系。有时候，我们的目的仅仅是拖延对手的行动。比如，我们可以把一段毫无意义的信息加密，然后让对手花上整整一个小时去解密。不过，一旦一条信息被对方成功解密，下次我们再有一条类似的信息被对方拦截的话，解密时间可能就会从一小时急剧减少为 3 分钟了。所以我们必须想办法永远比“敌人”快一步。接下来的讨论里，我有时候混用“代码”和“密码”，以及“解码”和“解密”。不过，“密码”一般是指把一段文字的每一个符号都用同一字母表或其它字母表里的另一个符号来代替。相反，“代码”是指把一段文字的每一组符号用另一组符号来代替。

自从凯撒密码提出后，密码学得到了极大的发展。其中一个很简单的想法是加密的信息不一定要与原始信息有相同的字数。如果字母“*I*”或者“*a*”作为一个单词单独出现在一段加密字符串里，那这段加密字符串就很容易被破解了。通常的做法是，把一段信息的每 5 个字母分成一组（不记字间的空格和标点符号），然后每组都用另外 5 个字母代替。这样加密的字符显得更难破解。如果原始信息的字符数不是 5 的倍数，那我们可以额外地将一些符号定义为这个“转换”里的空集，然后用这些符号来补齐缺少的字符数。

另外一个简单的想法是用多字码密码，也就是说给原始信息加密的字母表按照某个密钥随字母逐字变化。由于原始信息的每一个字母都对应到密钥里的一个字母，用以加密的字母表也会不断变化。这个想法的代表人物是利昂 • 阿尔伯蒂（Leone Alberti），他也是射影几何的先驱。最初人们都相信这样一个“复杂”的加密系统是不可攻破的。不过，如果密钥含有的字符数不多，并且由同一加密系统得到的密文样本足够多的话，我们可以用一些统计的方法来破解这样的密码。如果一个钥匙是随机产生并只使用过一次，也即一次性密钥，那这样的密码是无法破解的。如今当我们试图对一些信息进行安全保护时，面临的一个很重要的问题是如何交换密钥，特别是随机产生的大量钥匙。

利昂 • 阿尔伯蒂（1404-1472）

在现代生活里，人们更是意识到数学在安全事物中的重要性，其中不得不提的人物是阿兰 • 图灵（Alan Turing）。

阿兰 • 图灵（1912-1954）

希特勒在攻占荷兰、法国等国家后，试图进一步攻占英国。当时，英国全力以赴试图通过使用通讯和信号智能技术以及解密信息技术来阻止德军侵占不列颠群岛。英军在布莱切利公园成立了一个由语言学家和数学家组成的特别行动组，尽量从拦截到的敌军信息里获得有用的信息。包括马里安•雷耶夫斯基（Marian Rejewski）在内的波兰数学家首先获得了成功，他们将破解的德军代码交给英军。

马里安•雷耶夫斯基（1905-1980）

图灵及团队根据波兰提供的信息，成功地破解了德军“恩尼格玛机”（Enigma machine）以及其它密码系统发出的密码。毫无疑问，图灵等人的成功改变了这场战争的结果，也改变了历史。

二战时德军的恩尼格玛机

图灵与数学家戈登•威尔士曼（Gordon Welchman）还发明了一种特殊的“计算机”来破解德军恩尼格玛机产生的密码。

图灵与威尔士曼共同发明的“炸弹机”（Bombe）

在美国，包括语言学家和数学家在内的男男女女们同样在战争中发挥了作用。其中最著名的可能是威廉•弗里德曼（William Friedman）（他是业余数学家）以及他的妻子、语言学家伊丽莎白。二战期间，美军破获了日本军队大量的密码，也因此成功取得了战争的胜利。

伊丽莎白•弗里德曼与威廉•弗里德曼

如今，密码学不再仅仅是为军事与外交服务了，它越来越多地用在资本财富的建设上，例如通讯与互联网。对于电子转账、电子邮件或者网上购物等系统的用户而言，他们总是担心自己的交易是否按照计划进行而没有被“劫持”。当然，这涉及到的问题也是十分地复杂多样。当我们给别人发送一条信息时，我们自然要考虑这些问题：第一，对方收到的信息是否就是我们发出的原始信息，而不是被人更改过的；第二，现在发出的信息是否会危害到将来发出的信息的安全；第三，别人会不会通过我们发送信息的方式而盗用我们的资料，然后以我们的名义再给别人发送邮件。

“阴谋”这个人们常常与间谍或者间谍行为联系在一起的词，一样会在互联网安全领域里大肆横行。

## 散列法

散列法即按照某种方法，用一串短很多的字符替代一串长字符。例如，我们可以仅仅用一首诗包含的字母个数来表示这首诗。乍看上去，散列法似乎是关乎数据压缩，而不是数据安全的。显然在散列系统里我们可以设法隐藏原始字符，使之很难破解。因此很容易理解为什么一串被打散的字符可以起到加密的作用。使用散列法需要注意的是避免两组不同的字符串最后被散列成同一字符串。这种情况叫做冲突。在这样定义的散列系统里，如果两首诗歌有相同的字符个数，那么就会产生冲突。因此，我们在设计散列系统时，必须要说明怎么处理冲突情况。散列法的另外一个好处就是，两列相似的字符串经过散列后，可能会大不一样。因此，即使有人试图对一个加密过的文档进行微小的修改，也会很容易地被发现。因为原始的和伪造过的两个文档的散列字符串大相径庭。同样我们可以理解为什么散列法与用户登录密码和数字签名密切相关。数字签名是一个电子确认系统，其功能与我们书信或支票中常用的手写签名类似。我们当然希望一个数字签名系统可以尽可能地降低伪造的风险。在 1990 年代中期，麻省理工学院的罗纳德 • 李维斯特（Ronald Rivest）设计的 MD-5（Message-Digest Algorithm 5）散列系统被广泛的使用。不过，数学家和计算机学家于 1996 年在这方面做出了一系列的研究，使得人们意识到 MD-5 有着安全性能方面的缺陷。于是 MD-5 被 SHA-1 取代。

散列法在用户密码管理系统里十分常用，因此它与互联网安全也早已密不可分。除了登录电子邮件账户，人们还常常需要使用用户密码来登录一些互联网提供的商业服务（例如报纸或金融网站的账户）。你可以很快根据一些简单的规则来改进自己的用户密码，从而使得自己的密码比大多数人的更安全。不幸地是，只要计算设备足够好，好几个有关散列系统的标准都被证明是有安全性能隐患的。最近，被认为是有关散列系统安全性能通用标准的 SHA-1 也被证实是不安全的。这是由中国数学家王小云和她的团队发现的。当然，人们对王小云的成果短期内是否会影响业界意见并不一致。尽管我们暂时还没有可破

王小云（1966-）

解 SHA-1 的计算设备，但王小云取得的成果足以让我们相信她的尖端成果有可能攻击现有的安全系统。比 SHA-1 更安全的散列系统确实已经出现了，不过这方面的进展并不快。

有这么多的加密办法，好像要从一段加密信息恢复出原始信息实在是太困难了。一般有两种破解办法。第一种方法是用统计技术。如果我们拿到大量的加密数据，那么我们可以试图寻找其中的一些特定模式（例如分析一些符号的出现频率的分布情况）来进行破解。另一种方法是注意到加密的人经常会有些特定的语言习惯，而我们可以利用这一点作为解密的切入点。比如有些人在一段信息的开始总是用同样的问候语。再比如有些信息都是以天气报告为开头，这也使得解密者可以试图猜测一些相同的字符是如何编码的。还有一点就是，一个被广泛使用的系统通常都有不少人参与管理与维护。这时候，很难确保该系统的信息不被流传出来。因此，用来解决替代密码的方法与基于母体技术的方法不同。从现实角度来看，很多人都接受以下观点，即只有在假设攻击安全系统的人已经完全知道该系统的设计方案的前提之下，才能真正解决安全问题。因此，“敌人”或许已经知道你正在使用 RSA（下面会解释）还是 Hill 密码（基于矩阵的系统）。

## 公钥系统

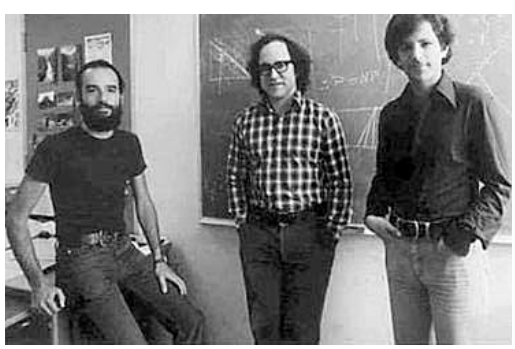
1970 年代中期，人们发现了一种全新的编码方式（也有人认为其实早就有人发现了），这是密码学的一个里程碑式的发展。传统的密码学里，需要共享一段加密信息的双方会设置一个系统，并需要完成一把钥匙的交换。直观上我们可以理解为这把钥匙将一段秘密发送的信息锁起来。当接收信息的人有一把一模一样的钥匙时，

自左至右依次为惠特菲尔德·迪菲（1944-）、马丁·赫尔曼（1945-）、拉尔夫·梅克尔（1952-）

他就可以解开这段被锁起来的信息。因此，在单钥系统里，加密和解密的双方有一把同样的钥匙。

公钥系统的出现改变了这一切。有关公钥系统的发展三言两语无法说清楚，不过人们普遍认为拉尔夫·梅克尔（Ralph Merkle）、惠特菲尔德·迪菲（Whitfield Diffie）和马丁·赫尔曼（Martin Hellman）三人在这个领域做出了非常重要的贡献。

公钥系统有两把钥匙。一把用于给某人 X 发送一段密码信息。正如电话簿上的电话号码，这把钥匙是公开的。另一把则不公开，它与公钥一起由 X 保留。公钥加密学与现代私钥系统还允许陌生人之间随意交换钥匙。这样的系统是由迪菲和赫尔曼在梅克尔的想法基础之上发明的，并用于不加密的通讯系统。这个系统已经申请了专利，不过现在专利保护的时间已经到期了。下面我们略微讨论一下最为人们所熟知的公钥系统，即以罗纳德·李维斯特、阿迪·萨莫尔（Adi Shamir）和伦纳德·阿德曼（Leonard Adleman）这三位发明者命名的 RSA 系统。



自左至右依次为阿迪·萨莫尔（1952-）、罗纳德·李维斯特（1947-）、伦纳德·阿德曼（1945-）

还有一个很常用的公钥系统是由塔西尔·埃尔格莫尔（Taher Elgamal）在 1984 年提出来的。这套系统主要用群论和复杂性的知识来保证安全。

塔西尔·埃尔格莫尔（1955-）

对 RSA 和 ElGamal（也包括其它系统）而言，一个很有用的概念就是同余。如果两个整数 $a$ 和 $b$ 被一个大于或等于 2 的正整数 $m$ 除以后，得到相同的余数，那么我们就说 $a$ 和 $b$ 同余，记为

$$a \equiv b \text{ modulo } m.$$

此时，$b-a$ 被 $m$ 除的余数是 0。以下是一些例子

$$12 \equiv 2 \text{ modulo } 5.$$

$$-3 \equiv 2 \text{ modulo } 5.$$

$$-3 \equiv 23 \text{ modulo } 13.$$

对于模 $m$，我们总共可以将一对同余数右边的那个数替换成介于 0 与 $m-1$ 之间的一个整数。比如，上述最后一对同余数，我们可以将 23 替换成 10。在以下表达式里，我们可以很容易确定“？”的值：

$$5^{72} \equiv ? \text{ modulo } 19,$$

办法是计算 $5, 5^2, 5^4, 5^8$ 等数取 19 的模，然后利用指数的二元表示法表示出 72 这个指数，并把答案找出来。不过，要找到整数 $k$ 使得同余表达式：

$$5^k \equiv 13 \text{ modulo } 19$$

成立就不那么简单了。类似于这样的寻找 $k$ 的问题被称为离散对数问题。当模非常大的时候，人们还不知道有什么方法比穷举法快很多。而对于解决离散对数问题和一些在设计公钥系统中用到的算法的复杂度，人们还知之甚少。一些基于 NP-complete 问题而建的系统崩溃了，而一些基于复杂度并没有完全搞清楚的问题而建的系统反而运行良好。下面我们将简单讨论一下被广泛用以衡量互联网安全性能的 RSA 系统。

## 小测试

许多公钥加密系统人员都受到“单向函数”的启发。有些任务很容易完成，但他们的反向任务则很难，除非有特殊的附加信息。

如果你想感受一下有关复杂度的数学知识是什么以及它是如何与安全性能联系在一起的话，试试下面的例子，用一个带秒钟的表测试一下你需要花多少时间求解这些例子。

问题 1：123457 乘以 372451 等于多少？

问题 2：374251 的因子？（我确定除了 1 和它本身，还有别的因子）

你也许会发现，问题 1 虽然很繁琐，但是得到它的答案并不难。然而，即使你能解决问题 2，你花的时间可能要比问题 1 多的多。大多数人甚至都没有耐心去完成最终的解答。

问题 1 比问题 2 更简单，这意味着什么？问题 1 要求两个数的乘积，更具体地说是两个 6 位数的素数的乘积（素数是指它的因子仅有 1 和它本身），而问题 2 是要求将一个可以表示成两个素数乘积的数进行分解而得到这两个素数。（即已知 s 是某两个素数 $p$ 与 $q$ 的乘积，求解 $p$ 与 $q$.）

大部分人都认为还没有一个真正的快速算法可以在一部普通的电脑上快速地分解整数。但同时，人们也没有任何结果表明整数的分解是一个 NP-complete 的难题。NP-complete 问题是指一类难度一样的问题，也即如果这些问题的其中一个可以被多项式级别的时间求解的话，那么所有这一类问题都是多项式时间可解的。另一方面，如果其中一个问题需要指数级别的时间求解的话，那么所有这一类问题都是指数级别的时间可解的。不过，虽然很多人都认为一些问题需要指数级别的时间求解，但事实上这一点也没有人可以完全肯定。大家认为没有简单算法可以对一个大整数进行因子分解，而且直到不久前，人家也不知道究竟是否有办法用多项式级别的时间来验证一个数是否是素数。所以当马尼卓 • 阿格拉瓦尔（Manindra Agrawal）、尼瑞艾 • 卡亚尔（Neeraj Kayal）和尼廷 • 塞克希纳（Nitin Saxena）三人提出一个验证素数的多项式算法时，人们感到十分意外。现在有很多快速算法可以验证一个数是否是素数。有意思的是，其中最快的一些算法的确可以在绝大多数情况下检验素数，但偶尔也会把一个复合数误认为是素数。验证素数的多项式算法方面的进展也意味着我们有可能找到实现整数的因子分解的多项式算法。（值得一提的是，不是每个多项式算法在求解实际问题的时候都很快。而有些指数算法，例如求解线性规划的单纯型法，在求解实际问题时却十分有效。原因是实际碰到的很多问题并不会令单纯型法无效。）

下面简单介绍一下广为人知并被广泛使用的 RSA 公钥密码系统。我们假设已经将一段要发送的原始信息的文字（例如英文）分成一块一块的结构，而原始信息的每一块也用一个数字 $M$ 来表示。我们需要考虑怎么安全地发送 $M$。接下来，我们产生两个大素数 $p$ 和 $q$。相对来说，这比较容易（也有人认为如何选这样的素数直接影响到最后的加密安全）。现在我们选一个大于 1 的整数 $e$，它要与乘积 $(p-1)(q-1)$ 互素。两个整数称为互素，如果能被这两个整数都整除的最大整数是 1。例如，12 和 35 都不是素数，但它们互素。然后，我们找一个整数 $s$ 使得

$$(e)(s) \equiv 1 \text{ modulo } (p-1)(q-1),$$

之后，再计算 $p$ 和 $q$ 这两个秘密的素数的乘积 $n = pq$。为了发送 $M$ 这个信息，我们计算 $C$ 来产生一段密码电文：

$$C = M^{e} \text{ modulo } n,$$

其中 $e$ 和 $n$ 的值即是公钥。注意 $n$ 是两个大素数的乘积。如果 $n$ 不能分解的话，它的值对密码电文 $C$ 而言毫无价值。

解密者通过以下计算来解密：

$$C^{s} \text{ modulo } n,$$

其中解密者的私钥可以由 $s$ 和 $n$ 生成。这些值用于恢复原始信息 $M$。

为什么这样计算就可以得到 $M$ 呢？其中原因（我们省去细节）涉及到了数论中的两个经典定理，一个是费马（1601-1665）提出的“费马小定理”：

$$a^{p-1} \equiv 1 \text{ modulo } p,$$

其中 $p$ 是素数，而 $a$ 与 $p$ 互素。另外一个定理是欧拉（1707-1783）提出的，它涉及到一个表示与 $x$ 互素的整数个数的函数。我们记这个函数为 $\varphi(x)$，也称为欧拉函数或者熵函数。对于任意的素数 $p$,

$$\phi(p) = p - 1.$$

而且，如果 $x$ 与 $y$ 是互素的整数，我们有以下关系：

$$\phi(xy) = \phi(x)\,\phi(y).$$

于是，对于不同的素数 $p$ 和 $q$, 我们有

$$\phi(pq) = (p-1)(q-1).$$

欧拉还证明了一个涉及到 $\phi$ 函数 $\varphi(x)$ 的有关费马小定理的非常漂亮的推广：

$$a^{\phi(x)} \equiv 1 \text{ modulo } x,$$

其中 $x$ 是一个与整数 $a$ 互素的正整数。

由于 RSA 系统的安全性依赖于大整数分解的复杂性，使用 RSA 系统的公司对数学家和计算机学家提出了很多有关整数分解的难题。最近，有人分解了一个 193 位的数，而这在之前是被认为十分困难的。现在也有不少人悬赏可观的奖金来求解一些整数分解问题，不妨看看这个网站 (http://www.emc.com/domains/rsa/index.htm?id=2093)！

最近，一个有关 RSA 和其它公钥系统安全性问题的担忧出现了。这一担忧关乎复杂度。随着传统计算机的速度变得越来越快，人们不禁要问是否可以用穷举的方法来破解加密系统。之前使用很多年的商用传统编码基本都是 IBM 公司制定的 DES(Data Encryption Standard) 标准。由于 DES 不再安全，一种新的标准也应运而生。这个新的标准被称之为“高级加密标准”，并以其发明者 Joan Daemen 和 Vincent Rijmen 的名字组合而命名为 Rijndael。在取代 DES 而成为新标准之前的试用阶段，Rijndael 经过了极其严格的检查和考验并通过了各种测试，最终才投入使用。不过，大家认为 Rijndael 在数学上是极其“完美”的，以至于有人认为它会因为数学上的太过完美而最终被弃用。也许最后我们还是可以根据 Rijndael 系统漂亮的数学结构来找到还未被发现的漏洞去破解它吧。

与计算机以及计算机技术支持的电信业紧密相关的数字革新发展，都是基于使用功能强大的单片机或者是使用多芯片技术的并行机。不过，科学家们正在努力发展一种基于物理学中的量子力学概念的全新计算方式。这种新的计算方式被称为量子计算。数学家彼得·绍尔（Peter Shor）等人认为，一旦量子计算机成为现实，那么有些需要通过常规计算机花费很长时间求解的问题，其计算时间将会大大缩短。其中，Shor 指出整数的因子分解这个问题如果用量子计算机来处理的话，就会比用传统电子计算机快很多。因此，如果量子计算机出现了，那么 RSA 系统就无法再保证密码系统的安全了。这个涉及到物理、数学以及计算机的新兴学科究竟会发展到什么情况，人们正拭目以待。

互联网已经改变了美国人以及世界各地所有人的交流方式，也改变了人们的生活。如果互联网还像人们所期望的那样继续给人们的生活带来正面影响，那么数学家也将继续参与互联网的发展。

**后记：**有关互联网安全的文献的更新速度是非常快的。一些对电脑运算速度要求不高的安全保障方法以及在一定时间里最好的算法经常没过多久就失效了。读者可以在网上查阅互联网安全的最新发展趋势。

**本文原文链接：**
http://www.ams.org/samplings/feature-column/fcarc-internet

**注：**译者袁晓明博士是香港浸会大学数学系副教授。