后面就是秘密！
——密码漫谈（续）

罗懋康

上期回顾：本文的前一部分刊登在本刊 2010 年 4 月创刊号的第 47 至 57 页。第一节介绍了密码的基本原理，特别是密码操作的五个基本要素。接下来，作者回顾了从远古时代至近代，密码学发展的起源和历史，特别是密码如何被军事家们不断地更新、发展和应用。在第三节，作者介绍了古代的密码，描述了密码的本质，以及让一项信息保持绝密的五种方法。作者还介绍了古代加密和解密常用的几类典型方法。

4. 生死之间，不见刀光剑影——密码的攻防

由于密码所关系的，经常都是一些生死攸关的事，因而围绕密码，历史上也就有着许许多多惊心动魄的事件展开。

图 22. 玛丽女王

(1) 玛丽女王：1578 年，因国内危机而逃亡英格兰的苏格兰玛丽女王被伊丽莎白女王软禁。1586 年 1 月 6 日，玛丽收到一批秘密信件，里面是她过去的侍从、当时在欧洲大陆的 24 岁的安东尼 • 贝宾顿（Anthony Babington）和她另外一些忠实追随者准备营救她的计划。

这些信件都是用密码写成、由贝宾顿交给一个对玛丽女王表现非常忠诚的天主教神甫吉法德带进监狱交给玛丽的。然而，贝宾顿怎么也没想到，这个吉法德却是伊丽莎白女王的间谍，执行英格兰大臣沃尔辛汉姆（Walsingham）爵士的命令。其结果，自然是所有这些信件都首先出现在沃尔辛汉姆的办公桌上。

图 23. 沃尔辛汉姆勋爵

这还不算贝宾顿和玛丽们最倒霉的事，更倒霉的是，沃尔辛汉姆不仅是负责君王安全的间谍首脑，而且还一直重视密码学的研究，在伦敦建立了一所密码学校，培养了一批专门人才。当他得到这批信件时，便让当时全欧洲最优秀的密码破译专家和笔迹摹仿专家托马斯 • 菲利普斯（Thomas Phelippes）将其破译了出来，汇报给了伊丽莎白。

此时的伊丽莎白，出于种种互相矛盾的利害考虑，对是否就此除掉玛丽女王举棋不定，沃尔辛汉姆猜透了伊丽莎白心里为难的原因，决定推动她杀掉玛丽女王，方式是设法构造玛丽图谋杀害伊丽莎白的证据。

他让间谍吉法德去告诉已经来到伦敦准备营救玛丽的贝宾顿们，现在要想武力营救玛丽是不可行的，因为玛丽

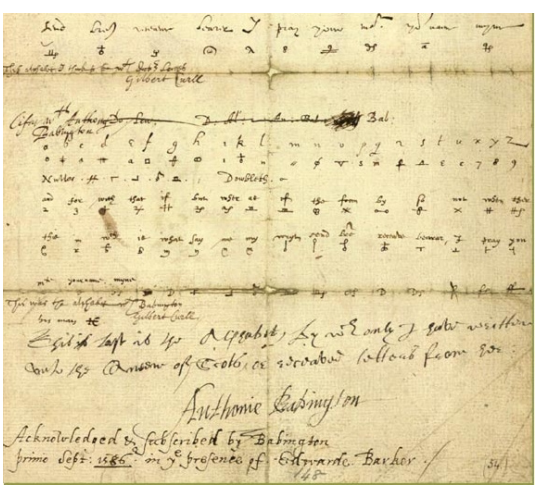

图 24. 玛丽女王解码密钥

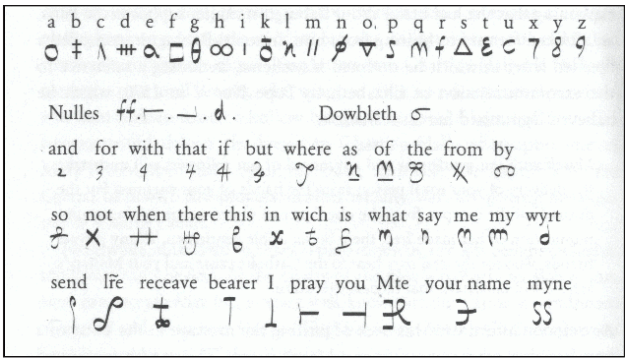

图 25. 贝宾顿与玛丽女王通信的密码表

被严密看守，并被指示稍有异动发生便立即处死。唯一可行的办法是暗杀伊丽莎白女王，然后便可利用玛丽是英格兰国王亨利八世的姐姐的孙女、伊丽莎白女王的表侄女这一王室血缘关系和名义让玛丽接掌英格兰王位，这样的话，所有问题就自然不复存在了。

贝宾顿们折服于吉法德的严密分析，立即重新拟定行动计划，并再次给玛丽女王写了一封信，说明他们将暗杀伊丽莎白女王，同时要求外国干涉、煽动英格兰天主教徒暴动（英国是新教的势力范围，天主教徒受压）。这封信还是由吉法德带给玛丽女王，并由玛丽女王签署了回信，表明她完全同意刺杀伊丽莎白女王的这一计划。

当然，这一切都在沃尔辛汉姆掌握之中；更可怕的是，他还让菲利普斯在玛丽女王的回信中，摹仿玛丽女王的口吻和笔迹加上附言，让贝宾顿列出重要成员的名字。于是，所有密谋者被一网打尽；最后，玛丽女王也在审判庭上，被自己那封由菲利普斯按沃尔辛汉姆的指示添加了私货，从而半真半假、自己也无从分辨的密谋信件推上了断头台。

(2) 裴炎宰相： 与玛丽女王死于密码差相仿佛，中国古代也有一个类似的例子，这就是由于密码被武则天识破而丢命的宰相裴炎的故事。

公元 684 年，柳州司马徐敬业在扬州起兵，讨伐武则天。这事在历史上固然有名，但被后世流传更广的，却是骆宾王为此所起草的“古今第一檄文”《为徐敬业讨武曌檄》。骆宾王这篇檄文，端的是文辞华丽，音韵铿锵，磅礴豪迈，雄奇激越：

“海陵红粟，仓储之积靡穷；江浦黄旗，匡复之功何远？班声动而北风起，剑气冲而南斗平。喑呜则山岳崩颓，叱咤则风云变色。以此制敌，何敌不摧！以此图功，何功不克！”“或膺重寄于话言，或受顾命于宣室。言犹在耳，忠岂忘心！一抔之土未干，六尺之孤何托？”“请看今日之域中，竟是谁家之天下？”

据说，当《为徐敬业讨武曌檄》传至京都，武则天初读时微露讥笑，但读到“一抔之土未干，六尺之孤何托”一句时，不觉耸然一惊，问侍臣：“此语谁为之？”有人答曰：“骆宾王之辞也。”武则天叹道：“此乃宰相之过，安失此人？”

据唐人张鷟《朝野佥载》和《新唐书·裴炎传》所载，徐敬业此次起兵，当朝宰相裴炎亦曾与谋。《朝野佥载》称：徐敬业约裴炎为内应，裴炎书“青鵞”二字作答。事泄，无人可解“青鵞”二字含意；武则天沉思片刻，曰此乃“十二月（青），我自与（鵞）”之意，也就是说答应将于十二月在朝中发动政变，以为徐敬业响应。

这里，“青鵞”相当于同时使用了替代法和移位法的密码，只可惜还是被破解了。

不过，此事不见于《旧唐书》，《通鉴考异》也认为这些记述“皆当时构陷炎者所言耳，非其实也”，这就是史家的事了。

(3) 生死攸关的六天，由密码决定： 1918 年，一战后期，同盟国中为首的德国，与协约国中的英、法、俄作战已近 3 年，双方伤亡已达 284 万 8 千人。此时的德国，虽然由于俄国在十月革命后宣布退出战争而似得转机，但此前 1917 年 4 月 2 日，由于德国“齐默尔曼电报”密码被秘密破译而导致的美国对德国的宣战（呵呵，另一个密码影响历史走向的事例，来龙去脉太长，还是暂付想象吧），却使德国的压力有增无减。不过，协约国方面的情况更为严重：德军当时停在距离索姆省的省会亚眠（Amiens）仅仅 16 公里的地方，距离巴黎也就百把公里。

双方都在紧张集聚力量，准备着决定双方各自命运的最后一战。

图 26. 武则天《升仙太子碑》拓片

1918 年 3 月 5 日，一战后期的德国，启用了由纳贝尔（Fritz Nebel）上校发明的全新的战地密码，也就是密码史上著名的 ADFGX 战地密码体制。这套密码仅用 ADFGX 这 5 个字母表达全部的密文。但直至 4 月 1 日，26 天中，协约国方面对这些德文密电一筹莫展。

4 月 1 日，是西方传统上的愚人节；就像上帝真要愚弄这帮法国佬一样，这一天，法国截收了 18 封这种用 ADFGX 战地密码加密的电报，却只能干瞪眼。

事实上，后来知道，这些 ADFGX 密码是通过“方表替代”和“密钥移位”两个过程的加密而得的。对比于当初破译这种密码时在黑暗中万千艰难的摸索，我们现在可以比较轻松地来看看它是怎么加密的了：

[1] 替代

首先构造一张由行、列都由 ADFGX 这 5 个字母作为标号、空格中随意填有 a 到 z 各个字母的用于替代的方表。

	A	D	F	G	X
A	q	w	e	r	t
D	u	i	o	p	a
F	s	d	f	g	h
G	j	k	l	z	x
X	c	v	b	n	m

图 27. ADFGX 密码的替代方表

由于这是一个 5 × 5 的方表，只有 25 个空格，又由于 y 在德语中使用较少，所以 y 在表中略去。

假定要加密的明文是“Let us go”。首先全部改为小写、删除空格，将明文变为“letusgo”。然后，对第 1 个字母 l，在上面的方表中找到其对应的行、列编号分别为 G、F，因此 l 就以 GF 替代。照此办理，直到完成全部 7 个字母的替代编码：

l	e	t	u	s	g	o
GF	AF	AX	DA	FA	GF	DF

[2] 移位

将这些编码连起来，变成 GFAFAXDAFAGFDF。

现在假设要求密钥的长度为 n（从安全的角度考虑，这个 n 当然越大越好；事实上，在 ADFGX 当年的使用中，这个密钥序列的长度一般要取到 20 左右），将 1 到 n 这 n 个自然数的顺序打乱，重新排列；比如，取密钥长度为 8，将 12345678 打乱成 63482517。

6	3	4	8	2	5	1	7
G	F	A	F	A	X	D	A
F	A	G	F	D	F		

图 28. ADFGX 的移位表

将重新排列后的长度为 8 的序列 63482517 分开写成一行，作为 8 个纵列的编号，然后将刚才连起来的编码中的字母顺序逐一填到这 8 个纵列中去，由左至右，到头再返回左边继续。

然后，将每个纵列的字母，不再管 63482517 的顺序，而是按 12345678 的自然顺序，逐一取出排序：

1	2	3	4	5	6	7	8
D	AD	FA	AG	XF	GF	A	FF

连起来，就得到了明文“letusgo”的最终加密结果：DADFAAGXFGFAFF。

因此，ADFGX 密码通过自己才掌握的方表替代和密钥移位，将每个字母加密成 ADFGX 这 5 个字母中的 2 个。

其实，发明 ADFGX 密码的纳贝尔上校是很谨慎的，他曾经提出：替换 - 换位之后形成的密文，应该再作一次移位，才能作为最后的密文。

但德国无线电和密码机关人员认为先前的替代和移位已经够结实了，除非上帝本人来，是没人破得了的，何况，作为战地密码，再往复杂里搞不仅容易出错，也白白增加加密和解密的时间；而在战场上，什么比时间更重要呢？于是，这个给敌军找麻烦的主意被否决了。

现在回到 1918 年 4 月 1 日这个 ADFGX 密码让法军郁闷的愚人节。

前面提到，这一天，法军一共截获了德军用 ADFGX 战地密码加密的 18 份密电；面对这些不知所云的密电，法军密码分析员乔治•潘万（Georges Painvin）似乎已经绞尽脑汁。可他却丝毫不敢懈怠：面对着正在疯狂攻击的德军，事实上他已身系正在苦苦支撑着的法军的生死存亡，早已完全是在超负荷工作，根本没有休息时间，玩儿命了！

好在，潘万的冥思苦索已经得出以下 3 个判断：

（i）德军所用的是复合加密，即先用替代方表加密，

再用密钥移位表加密；

（ii）经过频率分析得知，该方表每天一换，也就是说，图 27 中那种方表，虽然每天都还是 5 × 5 的，但是填写顺序每天就完全不同；

（iii）经过频率分析得知，该换位表的密钥每天一换，也就是说，图 28 中那种移位表，每列字母头顶上的数字排成的序列，不仅它们的长度每天要变，而且它们之间的排列顺序也每天要变。

现在，盯着已经越来越相信突破口就在它们身上的两份密电 CHI-110 和 CHI-104，潘万首先要解决的问题是：这么一连串全无间隔的字符，而且，CHI-104 电文中遗失了一个字母，以问号替代，怎么分组？换句话说，怎么断句？

CHI-110：

ADXDAXGFXGDAXXGXGDADFFGXDAGA
GFFFDXGDDGADFADGAAFFGXDDDXDDGXAX
ADXFFDDXFAGXGGAGAGFGFFAGXXDDAGGF
DAADXFXADFGXDAAXAG

CHI-104：

ADXDDXGFFDDAXAGDGDGXDGXDFGAG
AAXGGXG?DDFADGAAFFFDDDFFDGDGFDXX
XADXFDAXGGAGFGFGXXAGXXAAGGAAAAD
AFFADFFGAAFFA

由于潘万已经判断它们的最后一步是用一个移位表加密的，因此现在的问题具体来说就是，怎么把这两串字符按它们原来在图 28 那样的移位表中的纵向排列方式分割开来？要知道，对于潘万，这个移位表有多少列、多少行、有哪些列并没排满，这些可都是不知道的！

潘万注意到，这两份密电都是同一天截收的，因此它们用的方表、密钥和移位表都应该是相同的，他决定就从这一点插进去！

无穷无尽的思索、尝试、失败和从头再来，潘万终于走出了第一步，对这两份密电完成了分组：

CHI-110: ① ADXDA	② XGFXG	③ DAXXGX	④ GDADFF	⑤ GXDAG	⑥ AGFFFD	⑦ XGDDGA
CHI-110: ⑧ DFADG	⑨ AAFFGX	⑩ DDDXD	⑪ DGXAXA	⑫ DXFFD	⑬ DXFAG	⑭ XGGAGA
CHI-110: ⑮ GFGFF	⑯ AGXXDD	⑰ AGGFD	⑱ AADXFX	⑲ ADFGXD	⑳ AAXAG	
CHI-104: ① ADXDD	② XGFFD	③ DAXAGD	④ GDGXD	⑤ GXDFG	⑥ AGAAXG	⑦ GXG?D
CHI-104: ⑧ DFADG	⑨ AAFFF	⑩ DDDFF	⑪ DGDGF	⑫ DXXXA	⑬ DXFDA	⑭ XGGAGF
CHI-104: ⑮ GFGXX	⑯ AGXXA	⑰ AGGAA	⑱ AADAFF	⑲ ADFFG	⑳ AAFFA	

潘万大受鼓舞，继续不眠不休地进攻。两天两夜过去了，4 月 3 日，突然，仿佛就在一瞬间，ADFGX 的壁垒终于在潘万中尉顽强无比却又精妙无比的攻击下轰然倒塌，他终于成功地破译了 4 月 1 日这两份德军电文！接着，余下的 16 份电文的保护层，也就都在一鼓作气之下全部击碎了！

从这时开始，法军对于对面的德军，已经能够做到“知敌先机”了；但由于战场态势对于法军过于严峻，要对强大的德军做到“制敌先机”，法军还心有余而力不足，还得等待时机。

这个时机终于来了。1918 年 6 月 1 日，德军启用了 ADFGX 战地密码的升级版——ADFGVX 密码。

其实德军此时并不知道 ADFGX 密码已被法军破译，他们仍然认为这个密码牢固得足以抗御除了上帝本人外的天下一切攻击；他们之所以对这个密码升级，原因是 ADFGX 密码不能直接对阿拉伯数字编码、加密。

从图 27 的替代方表可以看出，25 格的表中，连 26 个拉丁字母都没法装完，更没有 0 ～ 9 这 10 个阿拉伯数字的空余位置。然而，战场信息显然又不可能离开大量的数字，这样一来，就必须将所有数字都以德文来表达；这种用某一种民族语言来表达数字的麻烦，在瞬息万变的战场上，特别是在战场上操作本来就非常复杂的加密、解密（脱密）过程中，有时足以令人疯掉。例如，365872，用中文表示是“三十六万五千八百七十二”，用英文表示就得是“three and sixty-five thousand and eight hundred and seventy-two”。

为此，发明 ADFGX 密码的纳贝尔上校在 ADFGX 中增加了一个字母 V，变成 ADFGVX，这样，图 27 的替代方表就变成了 6 × 6 = 36 个空格了，不仅可以将先前略去的 y 放入，而且还余下 10 个空格，刚好可以放置 0 ～ 9 这 10 个数字。

图 29. 法军密码分析员乔治·潘万中尉

而且，由于增加了方表格数，也就增加了方表中字符排列顺序的变化种类，同时也就增加了破译难度。

更而且，现在包含 0 ～ 9 这 10 个数字的方表将这些数字与字母一视同仁都编码为 ADFGVX 中的两个字母，再通过移位表移位，那么，有着诸如“the”、“any”、“back”之类固定搭配的语言单词，就和没有这类固定搭配的数字一起，被混合打乱、搅成一锅浆糊了，让敌人更加难以从词频、字频的角度发现蛛丝马迹。

至于为什么增加的字母是 V 而不是另外什么字母，原因是字母 V 的摩尔斯电码为“...-”，易于拍发也易于分辨和抄收。

在战场上，选用一些无论在拍发还是在抄收时都不容易出错的字母作为密码字符，这一点非常重要：枪林弹雨中，密码操作员精神高度紧张，如果事先设计密码时对此考虑不周，这时出错的概率必然大大增加。

很完美，是不是？可惜，他们遇上的是一个天才级的对手，乔治 • 潘万！

在法国这边，结合战场形势，已经基本可以肯定德国人即将发动一场对于双方都是决定性的强大攻势；再从德国人并不知道 ADFGX 密码已被破译的情况下，却“悍然”启用强度更高的 ADFGVX 密码来看，德国人对这一攻势的期望之高可见一斑！因此，这一攻势之于法国命运的重要性，可想而知。

而且，关键是德国人要的只是协约国这边在战役结束之前不能破译即可，而协约国特别是法国这边，却必须在德国发起攻势之前——还不能是已经临近敌人进攻开始的“之前”，还必须得让自己有起码的反应、调动、准备的时间——破译这个密码，否则在此之后，败局已定，无论多么完美的破译也都没用了。这一点，德国人很清楚，法国人很清楚，潘万中尉也很清楚。

图 30. 联军统帅福煦元帅

在对截获的密电进行仔细端详以后，潘万的注意力很快集中到其中三份电文上。这三份电文有个共同特点：都是 GCI 电台发出的，电文的时间组都是 00 : 05。

基于此前他对 ADFGX 密码的成功破译，终于，他在第二天下午七时前，也就是 6 月 2 日 19 时前，完全还原了德军 6 月 1 日使用的 ADFGVX 的移位表和方表！

剩下的事情就没什么可说的了，他很快得出了这两份密电的明文：

“第 14 步兵师：司令部要求电告前线（情况）。第 7（军）司令部。”

“第 216 步兵师：司令部要求电告前线（情况）。第 7（军）司令部。”

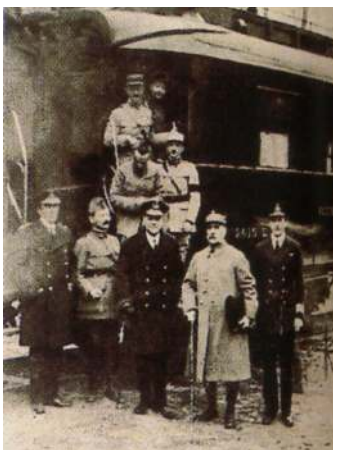

图 31. 贡比涅森林：福煦与德国签订停战协定后在福煦车厢前留影

但这对于法国来说，还没解决问题的全部，他们还必须尽早知道，德国将在何时、何地发起这场对于法国生死攸关的战役？

要知道，此时不仅德军前锋距巴黎已不足 70 公里，德军还占据了巴黎以北亚眠和蒂耶里堡两大突出部，对巴黎已形成了钳形进攻的态势！

这样的情况下，作为协约国联军统帅的法国福煦元帅，怎能不为猜测对面的德军统帅鲁登道夫元帅的想法而犯愁呢：他手里没有那么多预备队兵力，能让他布置到所有可能的德军进攻方向上，他必须知道鲁登道夫到底想在哪里动手。

好在，上帝此时对法国的心情似乎不错，让法国人的好运气再一次延续：6 月 2 日了，德军居然还在使用 6 月 1 日的替代方表和移位表！这已经够出奇的了，可到了 6 月 3 日，这种情况居然还在延续！真能让人晕倒！

这可犯的是密码学的大忌：“一次一密”做不到也就算了，但若连“一天一密”都不做到，这个战地密码最最起码的底线也就丢掉了！

6 月 3 日清晨，潘万的下级吉塔尔，面对着新截获的德军密电，不知德军今天的密钥又会把密文的分组搞成什么样；抱着死马当作活马医的态度，先用前天的分组方式试试，居然成功了！再用前天的替代表和移位表一试，让他都不敢相信自己的眼睛 : 居然都对了！这不是见鬼了么？

看看电文：“赶运弹药，不被发现（的话）白天也运。

就这么简单的十二个字，成为了协约国军队战场态势的一道分水岭！ 由此，赶紧辅以其他来源的情报和分析，法国

图 32. 贡比涅森林：德国凯特尔元帅与法国亨其格尔将军在福煦车厢中签署停战协定

人终于笑了：德国这次的进攻主力是第 18 集团军，进攻方向就是距离它在雷马奇的司令部 26 公里的贡比涅！

密码破译和无线电侦察，给协约国军队在这次致命的进攻之前，争取到了整整六天！

6 天之后的 6 月 9 日 04 时 20 分，德军准时发动了西线的第四次战役，一切都如法军判断的一样：

主攻部队：第 18 集团军；

战役目的：消除驻守在苏瓦松一带第 7 集团军右翼的威胁，并拉直亚眠、蒂耶里堡两个突出部之间的战线。

攻击方向：贡比涅。

战役的最后结果，是 1918 年 11 月 11 日，福煦代表协约国，与德国代表在贡比涅森林雷道车站的一节火车车厢里，接受了德国的投降，签订了停战协定；11 时，各战胜国鸣放礼炮 101 响，宣告第一次世界大战结束。此后，这节从此以福煦命名的车厢被放入了博物馆。

也正因如此，始终对德国在一战中的失败耿耿于怀的希特勒，在二战中击溃法国的抵抗、占领巴黎后，特别指定，谈判地点不设在巴黎，而是设在这片一战时令他心摧欲裂的贡比涅森林，而且，按照希特勒为了羞辱法国人而作出的指示，6 月 22 日下午 3 时 30 分，法国代表进场时才发现，要签订停战协定的场所，居然就是 1918 年德国在这里签订停战协定的那节福煦车厢，而且，在这节从博物馆中拉出来的车厢里，所有的摆设还都刻意恢复成了当年的模样。

这……太伤自尊了，法国人弱弱地表示难以接受。可在这种场合下，哪里还有他们表示不满的余地？希特勒、戈林、里宾特洛甫等人起身离去，在凯特尔元帅以典型容克贵族风格的冷漠有礼宣布完对法国的要求后，身为法国代表团团长的查尔斯·亨其格尔蒋军，代表法国在停战协定上签字。

此后，福煦车厢作为战利品，被德军运到柏林。后来，为了免于德国再次战败时的再次羞辱，希特勒下令炸毁了福煦车厢。

（4）恩尼格玛密码： 一战结束后，人们开始感觉“一张纸一支笔”的密码编写、拍发、抄收的方式效率实在不能再满足要求，开始研究各种各样的机械式和机电式密码机。这些密码机大都是将带有特别设计的导电触点的机械转轮以导线进行可变电气连接，来完成密码替代。

这些转轮机通常有一个键盘和一系列转轮，每个转轮是字母的任意组合，有 26 个位置，并且完成一种简单代替。例如：一个转轮可能被用线连起来使得可以用 F 代替 A，用 U 代替 B，等等，一个转轮的电气信号输出往往作为另一个转轮的输入。而且，设计者还往往给转轮装上各种各样的进位传动齿轮。这样，动态改变多个转轮之间的连接关系和传动关系，便可以对替代关系产生复杂的动态改变，使得一个明文字母在不同时候被替代成不同的密文字母，用以对抗字频攻击。

由德国发明家亚瑟·谢尔比乌斯（Arthur Scherbius）发明的电气编码机械“恩尼格玛”（ENIGMA，意为哑谜），就是这些密码机中最出色、最著名的代表。恩尼格玛在二战期间由德国人使用，而且，为了战时的需要，还大大地加强了它的基本设计。

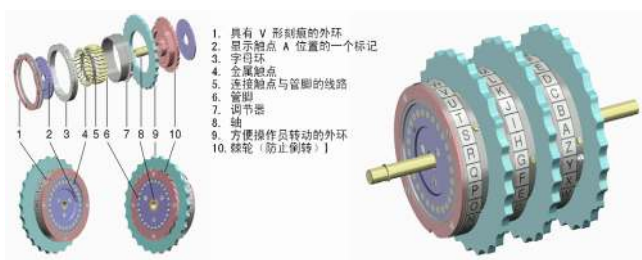

图 35. 恩尼格玛机的转轮结构和排列方式

恩尼格玛有 5 个转轮，每个转轮都有按不同位置和连接关系排列的 26 个字母，每次使用从这 5 个转轮中选择 3 个。机器中还有一块能将 6 对字母两两交换的连接板。恩尼格玛的密钥，就是由“3 个转轮相互之间的不同排列位置、3 个转轮各自的不同初始位置和连接板对 6 对字母的不同交换方式”构成；再考虑到从 5 个不同转轮中选择 3 个使用的可能性，这样的密钥数目有多大呢？这是一个令人头晕目眩的数字：

10×17576×6×100391791500

=105,869,167,644,240,000,

十亿亿零五千八百六十九万一千六百七十六亿

四千四百二十四万！

这个数字有多大呢？如果一秒钟尝试一个密钥，那么尝试完所有这些密钥需要 335,708,928 年！如果从过去算到现在，三亿多年前，那可还是石炭纪，连恐龙都还要再等一亿多年后才会在三叠纪出现。

如此复杂而强固的密码，可是却由于使用它的德国加密员在传送决定密钥使用方式的 3 个字母时为了避免错漏，而将这 3 个字母作了两次加密发送，导致两组（每组 3 个字母）不同的密文对应了相同的一组明文，给波兰总参二局密码处的密码专家马里安 • 雷杰夫斯基（Marian Rejewski）、杰尔兹 • 罗佐基（Jerzy Rozycki）和亨里克 • 佐加尔斯基（Henryk Zygalski）造成了后来终于撕裂恩尼格玛密码坚固外壳的细如发丝的隐蔽裂纹，最后在一系列卷入波、英、法、美多国情报人员和数学家、密码学家、工程师的充满阴谋陷阱、利益收买、暗夺明抢和卓绝思维、令人惊心动魄的行动后，被盟军破译和掌握。

在这个过程中，不仅有原为波兰波兹南大学数学教师的雷杰夫斯基最初的杰出贡献，而且，还有英国剑桥后来以天才计算机科学家、天才逻辑学家闻名于世的图灵（Alan M. Turing，1912—1954）和同事们设计的恩尼格玛专用解密机的强大推动。

所有德军恩尼格玛密码中，唯有海军的密码由于始终不

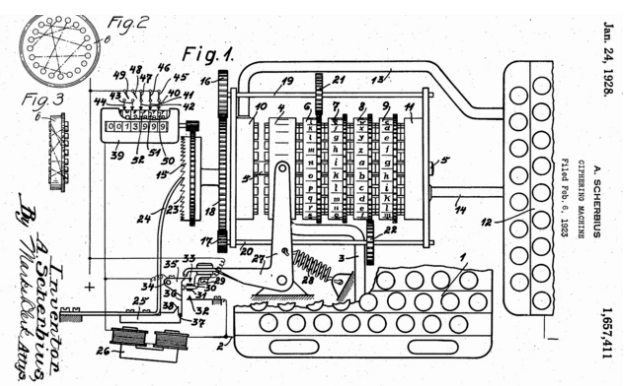

图 33. 谢尔比乌斯设计的密码机 1928 年取得的美国专利 1,657,411

惮其烦地严格遵守“尽可能保持最高强度”的使用原则，在盟军的密码攻击下基本上得以保全。

盟军的情报部门将破译出来的恩尼格玛密码称为“超级密码”（ULTRA）。虽然“超级密码”对二战到底有多大贡献还在争论中，但是人们都普遍认为，盟军在西欧的胜利能够提前两年，完全是因为恩尼格玛密码机被成功破译。

图 34. 二战德军恩尼格玛机

5. 当人力终于不能承受——现代密码

二战后，密码的需求日益增长，越来越难以像以往一样靠人力来完成；另一方面，电子学与计算机逐渐以越来越高的速度发展，加上信息的 2 进制表示在计算机中已成为基本信息形式，也促成了日益复杂的密码理论和技术。结果是，手工的加、解密运算越来越被计算机所取代，而且，计算机还可以加密任何 2 进制形式的资料，密码理论和技术的应用对象不再仅仅限于书写的文字。

这样一来，以语言学为基础的破译方法基本失效。不过，另一方面，计算机的强大计算能力却也同时促进了破译方法（密码分析）的发展，使得很多情况下的攻击尝试变得简单。

加密法的设计应该使得信息的安全性由密钥的安全性充分保证，而不应依赖加密法本身的安全性。这一由荷兰语言学家奥古斯特 • 柯克霍夫（Auguste Kerckhoffs）于 1883 年在《军事密码学》一书中提出并被称为柯克霍夫原则的加密法设计准则，已经得到普遍的采用。事实上，二战中德军的恩尼格玛密码机的设计，已经遵循了这一准则，而由美国国家安全局（NSA）和 IBM 为了抗御密码分析中新发展起来的差分分析法而制定、由美国国家标准局于 1977 年 1 月 15 日颁布为国家标准的数据加密标准（DES，Data Encryption Standard），更是典型地体现了这一准则。在计算机的计算能力以摩尔定律高速发展的二十年中，公开了算法的 DES 一直都在保护着银行、公司甚至核武器密钥的安全。直到 20 世纪 90 年代后期，DES 的加强版本 AES、3-DES 等加密法的应用才开始被提上日程。

不过，美国国家标准局公布的 DES 算法中，可没包括

$$S_1=\begin{pmatrix}14&4&13&1&2&15&11&8&3&10&6&12&5&9&0&7\\0&15&7&4&14&2&13&1&10&6&12&11&9&5&3&8\\4&1&14&8&13&6&2&11&15&12&9&7&3&10&5&0\\15&12&8&2&4&9&1&7&5&11&3&14&10&0&6&13\end{pmatrix}$$

$$\vdots \qquad\qquad \vdots$$

$$S_8=\begin{pmatrix}13&2&8&4&6&15&11&1&10&9&3&14&5&0&12&7\\1&15&13&8&10&3&7&4&12&5&6&11&0&14&9&2\\7&11&4&1&9&12&14&2&0&6&10&13&15&3&5&8\\2&1&14&7&4&10&8&13&15&12&9&0&3&5&6&11\end{pmatrix}$$

算法中要用到的 8 个被称为 S 盒（S-box）的 4×16 数字矩阵的设计由来，这一点长期受到质疑和非议，被认为是政府秘密保留的窥探私人秘密的暗门。

由于加密方法越来越脱离语言学的具体理论，而更多地向二进制数字信息处理的方向发展，包括信息论、计算复杂性理论、统计学、组合学、抽象代数以及数论等等数学分支的理论越来越多、越来越深地被用于密码学。

自古以来，直到 1976 年以前，密码的加密和解密使用的都是同一个密钥，正如锁上房门和打开房门的都是同一把钥匙一样；这已经成为天经地义的认识。这种用同一个密钥加密和解密的体制称为“对称密钥体制”，也因其必需的密钥必须全部严格保密的私密性而被称作“私钥体制”。但是，对称密钥体制或私钥体制从来就隐含严重的问题:

[1] 由于加密和解密使用的是同样的密钥，因此在分发加密密钥时，事实上也就是在分发解密密钥，也就必须严格保证分发途径和分发过程不能出错，即使分发顺利完成了，此后密钥的安全性也还只能寄希望于掌握加密密钥的人的可靠性；

[2] 由于加密和解密使用的是同样的密钥，因此在分发加密密钥时，如果需要分发的处所不止一个，就都得分别分发；这在分发数量增多时，将会使密钥分发或传送的过程变得严重困难；

[3] 如果需要分发密钥的人不止一个，而又须得保证他们之间不能以自己的密钥互相解密，便须为不同的人配置不同密钥，而作为对应措施，自己这边也就必须分别配置多个不同密钥；这在配发数量增多时，将会使密钥管理变得严重困难。

这些问题一直就存在着，但很少有人去尝试改变，因为难以想像一个公开了的密钥还怎么能发挥它原有的作用。

革命性的突破发生在 1976 年，这一年，美国斯坦福大学教授马丁 • 赫尔曼（Martin Hellman）和他的研究助理怀特菲尔德 • 迪菲（Whitfield Diffie）以及博士生默克勒（R.C. Merkle），提出了被称作“公钥密码体制”的概念：

加密、解密用两个不同的密钥，加密用公钥（public key），即可以公开，不必保密，任何人都可以用；解密只能用私钥（private key），此钥必须严加管理，不能泄漏。

而且，他们还发明了防止篡改和抵赖的数字签名（digital signatures）技术，即用私钥签名，再用公钥验证。

概括地说，公开密钥由一个密钥对组成，只适用于下列两个串连的行为：

[I] 一个人对某些人的“密钥发布”

[II] 这些人对这个人的“密文集中”

我们可以用一把专门设计的“公钥机械锁”来平行说明这个体制的原理。

如下是一把有两个锁孔和两把钥匙 A、B 的“公钥机械锁”的立体图和俯视透视图。为显示清楚起见，略掉了锁的外壳和一些辅助的作动机构。其中，

[i] 钥匙 B 是可以公开派发、随意复制的公钥，用它插入右边的锁孔向右扭转，便可使得钥匙 B 的锁闭定位销在弹簧拉动下滑入钥匙 B 的锁闭滑槽；此时，锁舌 B 插入锁杆上的锁槽 B，使得锁杆不能再被拉起，达到了用公钥进行锁闭的效果。

[ii] 一旦用钥匙 B 锁闭，由于钥匙 B 的锁闭滑槽的阻挡，钥匙 B 已无法再左右扭动，这就达到了再用公钥无法开锁的效果。

[iii] 钥匙 A 是由自己私密保管、不能公开的私钥。当自己要开锁时，将钥匙 A 插入左边的锁孔向左扭动，则钥匙 A 的开锁滑槽带动钥匙 B 的开锁定位销向左边移动，最后将锁舌 B 从锁槽 B 中拉出，解脱锁舌 B 对锁杆的卡阻，使锁杆恢复可以向上拉起的状态，达到了用私钥开锁的效果。

[iv] 用钥匙 A 作锁闭用的公钥的情况与此对称。

这把“公钥机械锁”的锁闭、开启的机制，便是公钥密码体制的一个具体演示。

不过，后来有人提出真正最早发明公钥体制的并非迪菲和贺尔曼二人，认为 1997 年的公开文件中表明早在 1970 年，英国的情报部门“政府通信总部”GCHQ 的数学家 James H. Ellis 便已发明非对称钥匙密码，只是由于保密的原因，把这 “密码革命”的巨大荣誉让迪菲和赫尔曼得了。但这一

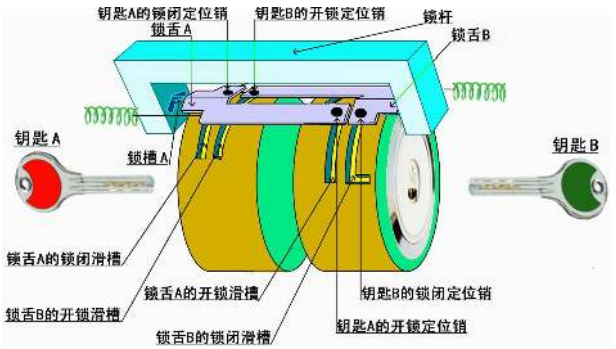

图 37. “公钥机械锁”的内部构造立体图

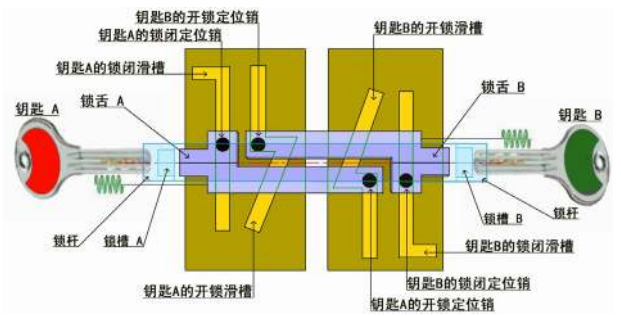

图 38. “公钥机械锁”的内部构造俯视图

点有争议。美国国家安全局也有过类似的宣称，不过就没多少人附和了。

公钥体制的概念提出后，人们立即开始投入其实现研究。比较典型的首先是美国斯坦福大学的默克勒（Ralph Merkle）和赫尔曼（Martin Hellman）在 1978 年提出的陷门背包公钥密码方法，以二人姓名首字母简称为 MH 法。

“背包问题”是组合数学中的一个经典问题，说的是：假如有一堆物品，所有重量都不同，问能否从中选择几件放入一个背包中使得总重量等于一个预先给定的值？用形式化一些的语言来描述：设有 n 个正整数 x_1，x_2，…，x_n（称为“重量序列”）和另一个正整数 S，问能否找到一个长度为 n 的由 0、1 构成的序列 $\{a_1, \cdots, a_n\}$，使得

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n = S \ ?$$

例如，假设这些物品的重量序列为 1，5，6，11，14，20，则可以用 5、6 和 11 组成一个重为 22 的背包，但却不可能组成一个重为 24 的背包。

背包问题的求解复杂度随着物品个数（即重量序列的长度）的增长而呈指数增长；也就是说，当物品个数比较大时，要对一个指定的背包重量用穷举的办法找出物品搭配的方式，将会变得非常困难。

例如，实际上使用的背包算法中的重量序列长度至少为 250 个数字，每个数字一般在 200 到 400 位之间。现在世界上速度最快的超级计算机是每秒千万亿次，中国 2009 年 10 月公布的巨型机“天河一号”的速度是 1206 万亿次，也是这个最高级别。就算用这样的计算机来对这样的背包问题进行穷举搜索求解，要试完所有可能的值也要 10^{36} 年，在太阳毁灭之前也没法算完。

默克勒和赫尔曼的想法是：选定一个由互不相同的正整数排成的序列作为重量序列；将明文用二进制编码，再将其按重量序列长度分段；然后将这些二进制的明文段分别与重量序列作乘积和（也就是按照一个二进制明文段中的 0、1 排列对那一堆物品作取舍，将取出来的物品重量求和）；所得的各个乘积和（背包重量）就作为密文：

明文:	1	1	1	0	0	1		0	1	0	1	1	0		0	0	0	0	0	0	1	1	0	0	0
重量序列:	1	5	6	11	14	20		1	5	6	11	14	20		1	5	6	11	14		1	5	6	11	14	20
密文:	1	+ 5	+ 6	+ 0	+ 0	+ 20	=	0	+ 5	+ 0	+ 11	+ 14	+ 0	=	0	+ 0	+ 0	+ 0	+ 0	=	0	+ 5	+ 6	+ 0	+ 0	+ 0	=
	32							30							0						11						

如上面这个例子，明文的二进制编码结果是 111001010111000000011000，重量序列取为 1，5，6，11，14，20；将明文的二进制序列按密钥长度 6 分为 4 段：111001，010110，000000，011000；将各段明文分别与重量序列作乘积和，分别得到 32，30，0，11，这就是明文的二进制序列 111001010111000000011000 加密后的结果—密文。

注意这里加密时是将先行取定的二进制明文序列作为背包问题的解，由此求出对应的背包重量，将其作为密文，这与背包问题的顺序刚好是倒着的：背包问题是先给出背包重量，然后问包中物品的重量是哪些。因此，解密时的运算才和背包问题同步：都是根据背包重量来求对重量序列中各项的选取方式。

数学上证明背包问题是著名的 NP 完全类困难问题，至今没有统一的好解法；而要通过硬搜索来解这个问题，正如前面所述，一般而言在实际上是不可能的。

既然如此，这种方法可就还不能作为一种加密方法；道理很简单：别人固然没法从密文找出你的明文了，可你自己也没法重新找出来，知道重量序列也没用！

好在默克勒和赫尔曼在这个基础上进一步研究出了一种自己作解密运算很简单、但让别人解密却继续令他崩溃的方法；而且，他们这个方法还对两年前由赫尔曼、迪菲和默克勒提出的“公钥体制”思想，首先给出了一个具体的实现。

我们不妨来看看这是如何实现的：

背包问题一般而言非常难解，但是，对于一种特殊的重量序列，所谓“超递增序列”（superincreasing sequence），它却又有非常简单方便的解法。

超递增序列是这样一种序列，其中每一项都大于它前面所有项之和。例如 1，3，6，13，27，52 是超递增序列，而 1，3，4，9，15，25 则不是。

也就是说，当一个背包问题的重量序列是超递增序列时，对于任意给定的背包重量，要从重量序列中找出相应的重量组合非常容易；而当重量序列不是超递增序列时，要做到这一点就变得非常困难，难度之大，正如上述。

默克勒和赫尔曼找到一种方法，可以将一个超递增序列按照给定的参数，以统一的方式转换为一个非超递增序列，而且，使得从非超递增序列按某一种重量选取方式产生出来的背包重量，用这个超递增序列去求出这种重量选取方式是轻而易举的事。也就是说，一个超递增序列与由它转换而得的非超递增序列，面对同样的背包重量时，在各自的序列中决定了同样的重量选取方式，比如，都是第 2 项、第 5 项、第 11 项、……，等等。

因此，如果以一个超递增序列作为私钥，将其转换而得的非超递增序列作为公钥，那么，他人用公钥按上面的方式加密后，再用这公钥却几乎完全没法重新求出明文来；而对于掌握私钥的人，则可从超递增序列轻易求出。就这样，默克勒和赫尔曼用背包算法成功地实现了“公钥密码体制”的思想。

不幸的是，幸福实在太过短暂，好景也实在太不长，就在他们提出这个称为 MH 方法的密码体制后的两年，1980 年，MH 方法便被以色列的密码学家、图灵奖获得者 Adi Shamir 和美国密码学家 Richard Zippel 破译（Shamir， A. and Zipple, R. E. (1980), On the security of the Merkle-Hellman cryptographic scheme. IEEE Transaction on Information Theory, 26, 339-340）！他们找出了一种从 MH 密码系统所给出的公钥（非超递增序列）重构出其对应的私钥（原来的超递增序列）的算法，宣告了这种加密算法的夭折。

后来至今，还有一些人想方设法改进 MH 方法，希望重新恢复其安全性，但都已经没有多大意义了。特别地，当另一种基于完全不同的原理（素数分解复杂性）的公钥密码——RSA 密码的地位越来越巩固时，事实上基于背包算法的加密方法已经基本走到了尽头。

1978 年，美国麻省理工学院（MIT）的 Ron Rivest、Adi Shamir 和 Len Adleman 公布了另一个公开钥匙系统 RSA。RSA 公钥密码算法是目前网络上进行保密通信和数字签名的最有效的安全算法之一，其安全性完全基于数论中大素数分解的困难性，所以，RSA 需采用足够大的整数。因子分解越困难，密码就越难以破译，加密强度就越高。

围绕这个 RSA 算法，在 1990 年代还发生了几起挑战美国出口规范的事件。其中一件是 Phillip Zimmermann 著名的 PGP（Pretty Good Privacy，良好隐私）加密程序事件。

1991 年 6 月，基于 RSA 算法的网络加密程序 PGP 的作者 Zimmermann 在美国将 PGP 连同其源代码一并公开在互联网上。在以 RSA 为其根本的 RSA Security 公司提出抗议后，商务部和联邦调查局对 Zimmermann 进行了长达数年的调查、询问。

事实上，PGP 的出现是美国国务院和国家安全局所最不愿意看到的事情。美国政府的 Clipper 计划和托管加密标准，就是企图保证当局能够随时窃听任何私人的电话和拆看任何私人的文件。Zimmermann 之所以把 PGP 在网上公开，就是为了在美国国会开始考虑限制计算机自由并制订严酷的法律之前，让大家有一个保护电子通信隐私的程序。PGP 也因此和 Linux 一样，被并列为最伟大的自由软件。

由于美国政府按照“美国国际贸易和武器管制条例”，把加密系统列入军火范畴，把加密软件的出口视同于武器走私，同时美国政府又规定源代码只能以书面形式而不能以电子形式出口，因此 Zimmermann 在他的 1995 年出版的《PGP Source Code and Internals》书中尽数列出 PGP 的源代码，让美国政府十分恼火。

图 40. PGP 作者 Zimmermann

当然，政府也不是白吃饭的，立即就此向法院提出对 Zimmermann 的诉讼。但是，紧接着就被柏克莱加州大学的研究生 Daniel Bernstein 以言论自由的理由发起对美国政府的反诉讼。最终，法院在 1999 年判决，印出密码算法的源代码属美国宪法言论自由保障范围。Zimmermann 的自由最终得以保全。

不过，虽然已经确定 RSA 的安全性完全依赖于大数分解的困难性，但却至今没能证明大数分解就必定困难，只证明了它的难度与数

学中一类著名的困难问题——NP 完全类问题相当；但 NP 问题到底有多难，也至今没能有一个确切的答案。

正因如此，加上此后大素数分解的新纪录不时被打破，例如，2007 年 3 月 6 日波恩大学等 3 个机构的计算机集群在 11 个月的计算后的结果便颇有演示意义：这个结果把一个 307 位的数字分解成了两个素数因子：

$2^{1039}-1=$
1159420574 0725730643698071 48876894640753899791 70201772498686835353882248385 9966756608 0006095408005179 47205399326123020487 44028604353028619141014409345 7970787973 8061870084377771 86828680898447128 22 00293520180607475545154137071 1023817

因子：

5585366661 9936291260 7492046583 1594496864 6527018488 6376480100 5234631985 3288374753
×
2075818194 6442382764 5704813703 5946951629 3970800739 5209881208 3870379272 9090324679 3823431438 8414483488 2534053344 7691122230 2815832769 6525376091 4101891052 4199389933 4109711624 3589620659 7216748116 1749004803 6597355734 0925320542 5523689

现在，人们对 RSA 安全性的信心已经开始有所动摇，目前建议其密钥（由数字组成）长度最好取到 1024 位或以上。

目前国际上公认比较安全实用的公钥密码体制是所谓的椭圆曲线密码体制，但也只是“公认”，并没能证明这一点。椭圆曲线密码的原理对于非专业人士来说已经过于专业，这里就不介绍了。

此外，还有正在热烈研究之中的量子密码，由物理学基本原理决定了被不露痕迹地中途截收是不可能的，这里也不介绍了。

6. 吁一口气，回到现实——几点建议

经过了如此漫长（但仍然挂一漏万，密码学的内容和历史实在太丰富了）走马观花式的浏览，作为非密码专业人士却又没法离开密码的我们，可以得出几个有用的结论：当我们不得不使用密码（实际上是前面叙述中所说的“密钥”）时，

(1) 尽可能不要采用别人站在你的角度容易设想到的密码，例如自己的或家人的姓名或姓名缩写、生日、电话号码、身份证号码、门牌号码之类的字符；要让自己想到：如果这样设置密码，固然带来很大的便利，但由此同时带来的巨大风险是否值得？

(2) 以无线电波或红外光波之类无线传输用作通过口令类型的密码（如汽车门钥、车库门钥等），只要密码不是可变的，其安全性都值得怀疑：试想，当你发送密码时别人也可以不露痕迹地接收，而且还不必非得破译其中的密码，直接记录下来便相当于复制了你的钥匙，在你离开后便可大摇大摆地打开车门、房门了。

(3) 在网上，没什么信息是绝对安全的，切记，凡是要跟网络连接的计算机，上面任何没有足够强度加密的信息，在有心人面前都如自己囊中之物一般。

(4) 绝大多数公开发售或网上下载的商业性加密方法，在专业人士特别是以国家力量支撑、配备了巨大计算能力的专业人士面前，都如篱笆墙一样，对破译只具有象征性的防御意义。例如，现在广泛使用的 RAR 压缩软件自带的加密功能，从算法设计看貌似够强固了，采用了数据加密标准的加强版 AES，但其密码验证机制的设计却有重大缺陷，难不住专业人员。因此，如果认为自己的秘密事关重大，建议使用在专业密码界有很好口碑的加密算法，特别是其中那些公开了源代码的加密算法，例如，前面介绍的 PGP 就是一个不错的选择。虽然这样做也仍然不能绝对保证安全，但毕竟是相对最为稳妥的选择了。

(5) 又要密码安全可靠，又要密码使用方便，这不是难为人吗？但还是有办法的：用一句自己熟悉却并不普遍、长度足够的句子（起码 16 个字以上），以其单字首字母的排列甚至就是单字本身的排列来作密码，尽量把其中的标点符号也包括进去，这就能比较好地兼顾这两个互相冲突的要求。

最后，祝长久好运！

全文完

注：本文中，除图 18、图 37、图 38 和各个表格为自绘外，其余图片取自网上公开资源，这里不及一一列明，谨一并致谢。另外，本文属于科普文章，参考了大量公开文献，但均只按本文体例需要取其含义、另予表述。

征文启事

本刊的数学烟云栏目主要用于介绍数学学科的发展和研究内容等，欢迎广大读者投稿。来稿请寄：math.cult@gmail.com。