KINGSUN

软件狗加密技术透视

王全国　任晨光　编著

清华大学出版社

软件狗加密技术透视 王全国 任晨光 编著 清华 版社

软件狗加密技术透视

王全国　任晨光　编著

清华大学出版社

(京)新登字 158 号

内 容 提 要

本书剖析了软件狗的加密原理，详细介绍了软件狗的编程方法，总结了软件狗的发展历史，展望了软件狗的未来。同时以软件狗为锲，将读者带入了整个软件加密领域。本书不仅可以满足对软件加密感兴趣的读者以及想用软件狗进行加密的程序员，而且对于具有初级计算机知识想深入了解计算机的读者也是一本好的参考书。

好的工具可以让你事半功倍，好的工具书可以让你在知识的领域更上一层楼。

图书在版编目(CIP)数据

软件狗加密技术透视/王全国，任晨光编著．—北京：清华大学出版社，1996.5
ISBN 7-302-02044-2

Ⅰ.软… Ⅱ.①王… ②任… Ⅲ.电子计算机-软件-加密-密码-技术 Ⅳ.TP309

中国版本图书馆 CIP 数据核字(96)第 04887 号

出版者：清华大学出版社(北京清华大学校内，邮编 100084)
印刷者：北京市海淀区清华园印刷厂
发行者：新华书店总店北京科技发行所
开 本：787×1092 1/16 印张：15.5 字数：367 千字
版 次：1996 年 6 月第 1 版 1996 年 6 月第 1 次印刷
书 号：ISBN 7-302-02044-2/TP・1007
印 数：0001—4000
定 价：26.00 元

前　言

目前，大家对国内软件是否要加密莫衷一是，有的说不需要，有的说需要。说需要的一方，其理由是，如果软件不加密，将会有非法盗版出售，用户也不愿意购买这样的产品，最终的受害者将是软件开发商，所以软件必须加密后出售。说不需要的一方，其理由是，任何一种软件加密方法，最终都会被破译，而且软件加密后增加了维护的难度，更何况国外的大多数软件都是不加密的，与其费事费力研制加密技术，不如腾出人力、物力和财力来搞好售后工作，使得正版软件的购买者得到称心如意的服务，以此来吸引用户。笔者比较倾向于后者，但在目前，对软件进行加密还是很有必要的，可以保证投资的及时回收，不过决不可以因此而忽视售后服务，我们最终的目标是要提高全社会的法制观念，鼓励人们购买正版软件，使得商品软件不必加密后再出售。

既然这样，那我们写这本书的目的是什么呢？

当前市场上能见着的软件加密方法很多，大多是在DOS下开发的。软件加密技术主要由密钥技术、反跟踪技术和代码插入技术构成。其中较常见的密钥有磁盘加密法、加密卡法和软件狗法。这三种方法各有优劣之处，其特点如下：

	磁盘加密法	加密卡法	软件狗法
成本	低	高	较　低
易安装性	不需要安装	复　杂	容　易
使用方便性	不　方　便	方　便	方　便
可靠性	一　般	稳　定	稳　定
读写速度	慢	快	快
系统开销	判别时需插盘 （可安装的除外）	占扩展槽和地址空间	并行口
产品盘备份	不　能	可　以	可　以
兼容性	稍　差	最　好	稍　差

由此可见，软件狗是一种比较好的软件加密方法，在目前市场上获得了最为广泛的应用。读者知道，软件加密的特点在于不为他人知道，一旦把它公开了，大家都会用，也就不值钱了。所以对于软件狗这样一种优秀的软件加密方法，我们是否应该把它公布于众呢？尽管目前各种盗版、侵权现象仍很严重，但是，中国政府对知识产权已加强了保护，人们也逐渐认识到了购买正版软件的重要性，国内有的软件开发商也开始尝试不加密就上市自己的产品。另外，软件狗加密方法已经流行一段时间了，公布其中的一些细节，虽然增加了软件狗设计者的设计难度，但是反过来也促进了它的发展，使得软件狗的加密方法能更上一层

楼。怀着这样一种想法，同时也带着让广大读者都来分享这种加密技术的愿望，我们最后决定编写这本书。

如果读者仅仅想了解一下软件狗，并学会使用它的话，您只需阅读第一章即可。如果想深入了解软件狗，并学会它的制作方法的话，您就需要阅读完全书。

本书由王全国提供原始资料，任晨光完成编写工作，金山编辑室的傅占华小姐负责联系和整理工作。编写过程中参考了众多书籍，并且得到了许多朋友的帮助，在此对他们深表谢意。由于时间仓促，加上编著者知识有限，书中错误在所难免，希望读者不吝指教。

编著者

1996 年 2 月

目　录

第一章　软件狗快速入门及使用

软件狗加密是软件加密的一种方法，所以在讲述软件狗之前，我们先来看一下软件加密的基本原理。

软件加密一般有以下三种方法：

外壳式：通过把一段加密代码附加在需加密的软件上，在软件执行时，加密代码首先执行，检查是否有程序在跟踪，如果没有再检查"密钥"是否存在，在完全没有错误的情况下，再去执行原来那个软件。

这种方法的优点是不必对源代码进行修改，缺点是，一旦外壳被解密，软件就什么保护都没有了。

内含式：修改源程序，在软件中嵌进对加密代码的调用，最后与加密代码一起编译连接而成。

这种方法虽然比较可靠，但也容易被跟踪，主要在软件狗和加密卡上使用。

结合式：把外壳式和内含式结合起来，取长补短，互相检查。

我们这里讲的软件狗加密技术，主要就是指内含式的加密方法。如果跟黄玫瑰软件制作组出品的 BITSHELL 配合，加以外壳保护，就能达到结合式加密方法的效果。

一种好的软件加密保护技术，其关键在于不易让人破译复制和判读的可靠性，另外还要保证软件的兼容性。这个道理是显而易见的。如果密钥很容易被复制，加密也就没有什么用了。如果对密钥的判读，有时正确，有时不正确，或者在一台机器上可以工作，但在另一台机器上不能工作，这就达不到加密的目的了，对用户也有很大的损害。软件的兼容性是指，不能为了达到加密的目的，而使得软件本来在某种条件下可以运行，加密后却不能运行了。要达到这些目标，并不等于意味着需要做多复杂、多高深的工作，目前的软件加密保护技术都是在人们意想不到技术上作文章，一旦了解了其原理，便会发觉原来如此之简单。软件狗技术亦如此，早期的第一、第二代软件狗不必多说，即使目前常用的第三代软件狗，其关键也仅仅在于一片 EEPROM 芯片（也有用 PAL，GAL 等其它芯片的）。但是，应该说，第一个想到利用该芯片做软件加密的人的确非常有创意。

本章的目的在于让读者认识、了解软件狗，并且学会使用软件狗。

1.1　认识软件狗

软件狗是插在微机并行口上的一个软件保护装置，它包括主机检查程序和密钥（亦称加密盒）两部分。主机检查程序就是前面说的加密代码的一部分，加密盒是用来存放密码的。一般来说，软件狗插在并行口上，不会影响打印机的正常工作。常见的软件狗加密盒外形，如两个一公一母的 D 形 25 针连结器倒接在一起（如图 1.1 所示），公头（DB25/M）插在并行口上，母头（DB25/F）可接打印机，相当于原来的并行口。整个软件狗的硬件电路

板就在这约5厘米见方的加密盒子里。

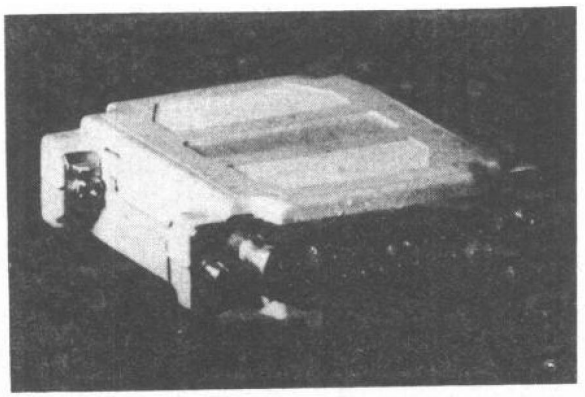

图 1.1　软件狗加密盒外形

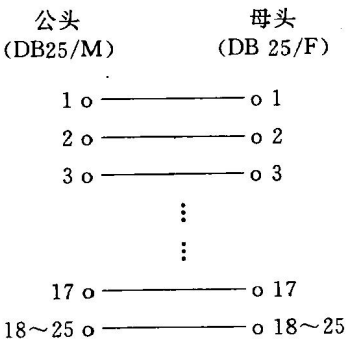

图 1.2　公头与母头的连接

电路板上的公头(DB25/M)和母头(DB25/F)之间的管脚是一一对应、直接相联结的(如图 1.2 所示),以保证并行口的作用不变。存储密码或起信号加密变换作用的器件及其它辅助元件就跨接在这 25 根线上面,应用程序以特定的方式跟它们沟通、核对。除了某些设计不良的情况之外,一般不会影响打印机的正常工作,打印机工作时也不会影响它们。

为了防止程序被非法复制,所做的加密保护措施一般都包括两部分。首先是要有保存密码数据的载体,即密钥;其次是夹杂在应用程序中的主机检查程序,即加密代码。密钥应该能保证不易被解密、复制;如一般用磁盘作加密时,加密部分无法用 COPY、DISKCOPY、PCTOOLS 等工具加以复制。另外,当检查程序用特殊方法去读密码时,密码应该能很容易地被读出,而不致影响应用程序的正常执行。当发现密码不对或密钥不存在时,就让主机挂起、重新起动或采用别的措施。

在早期,第一代软件狗起信号加密变换作用的器件,最多只简单采用一些电阻、二极管等,检查方法也比较简单,很容易被人解密;第二代软件狗采用了低功耗 TTL,CMOS 等逻辑元件,在电路上做了一些加密工作,检查时也要比第一代软件狗多一道手续,解密的难度自然也增加了;第三代软件狗采用了 PAL(Programmable Array Logic)、PEEL(Programmable Electrically Erasable Logic Device)、GAL(Generic Array Logic)等可编程器件,但目前最流行的器件大概要算串行读写的 EEPROM(Serial Electrically Erasable PROM)了。这些器件由于密码编制的灵活性和制成密钥后在程序中插入检查的方便性,极大地增加了解密的难度。从使用的角度看,PAL、PEEL、GAL 等逻辑器件只能读取数据,不能随时写入数据,密码的重新设定比较麻烦;而 EEPROM 芯片可随意读写,用在软件狗上灵活性相当大,譬如可以为每一个软件狗单独设一个密码,以增加解密的难度;另外,从 EEPROM 器件的电气性能上来说也非常适合做软件狗;因此,这种器件在软件狗的设计中获得了广泛的应用,是目前软件狗制作者的首选芯片。出于这个原因,本书的重点也就在于介绍利用 EEPROM 芯片制作的软件狗。由于解密技术紧跟加密技术的发展,目前人们已经开始研制更好的防解密的软件狗,新一代的软件狗采用单片机为核心

芯片，极大地提高了解密的难度。

出售的软件狗，一般来说，出于保密的目的，其加密盒都是用胶封起来的，人们看不到里面的电路，其实原理非常简单。拿目前最常用的软件狗来说，其关键的芯片就是一片EEPROM。我们在第二章将对EEPROM的知识作详细的介绍，读者现在只需要知道它像一般RAM存储器一样可读写(只不过读写是串行的)，即使断电后也能保存数据不变。常用的EEPROM型号是93C46，它是64×16bit的结构，也就是说一个93C46具有64个16位bit单元的容量，每次处理数据也都是16位。有的93C46，如Microchip、ATMEL、CSi等品牌的93C46可以通过切换，变为128×8bit或64×16bit两种模式，这对软件狗制作来说就更灵活了，其加密效果也更好。当然也有人采用容量更大的93C56，93C66或容量小一点的93C06、93C26等EEPROM芯片。

因为软件狗是插在微机的并行口上，所以检查程序是通过并行口的I/O地址去读写EEPROM。具体的读写方式跟硬件线路以及EEPROM的时序有关，因此，一般的检查程序是针对某一种硬件线路；但是，这些程序大同小异，大体上是差不多的。读者只要掌握本书后面几章所讲述的软件狗制作技术，对市场上一般软件狗的工作原理就了如指掌了。如果您想自己动手做一个软件狗，那也不难，根据您的硬件线路，修改本书给出的读写程序，编译连接后就可以用下一节的实用工具来编辑您的软件狗了。然后，您把相应修改过的检查程序嵌进您的应用程序中，用软件狗来保护您的成果就没有问题了。当然，如果您想使您的软件狗具备很强的防解密能力，可能需要花点功夫去钻研一番。

1.2　常用软件狗介绍及使用

这一节我们先浏览市场上常见的软件狗，再给您介绍由黄玫瑰软件制作组研制的软件狗的用法。

1.2.1　常见的软件狗

目前市场上大致有以下几种插在并行口上的加密狗出售，分别是软件狗、微狗和网络狗。软件狗就是本书要重点剖析的一种加密狗，它的核心就是一片EEPROM。微狗是软件狗的一个升级产品，加密和防破译性能要比软件狗好，它采用单片机技术制作，可对抗逻辑分析仪对信号的检测，本书在第四章将对它作介绍。网络狗，顾名思义就是在网络上用的加密狗，是在软件狗的基础上增加了一些网络的特点。

这些加密狗，一般都有以下特点：

1. 不占用并行口，因为它虽然插在并行口上，但是它又提供了一个跟原来一致的并行口。

2. 软件具有防解密功能，可对抗各种调试工具的跟踪，如Debug、TurboDebugger、Soft-ICE等等。

3. 一狗一密码或一种线路，软硬件不可互换，就像一把锁一把钥匙那样。

4. 提供各种语言的编程接口以及一套实用工具，方便用户在自己开发的程序中嵌入加密模块。

5. 提供对可执行文件的加密工具，以便用户对已有的产品进行加密。

6. 使用寿命长，除非不正确的操作，一般能用 10 年以上。

制作和出售软件狗的公司、单位很多，如黄玫瑰软件制作组、理德商用技术有限公司、北京金天地软件发展有限公司等等，您只要注意一下计算机杂志上的广告就能得知。各公司生产的软件狗除了上述特点外，一般都有一些为吸引用户而附加的功能，主要是一些工具软件，其核心的技术却是大同小异。

1.2.2　软件狗的使用

讲了那么多软件狗的知识，现在让我们来实际使用一下软件狗。一般出售的软件狗，除了提供给您一个加密盒和一套主机的存取程序外，还带有一个对软件狗进行编辑的实用工具，以方便用户自己设置密码。前面我们已经说了，加密盒（密钥）是软件狗的密码数据载体，而主机存取程序是嵌入到用户的应用程序中，对加密盒中保存的密码进行核对。密码的写入有几种方法，其一，您可以用专用烧入器写入，但这需要您购买一个昂贵的烧入器；其二，您可以自己编一个写入程序，这需要花点时间；其三，您可以用随软件狗出售的实用工具来写入。黄玫瑰软件制作组研制的 BOXMAKER 就是一个功能非常强的实用工具，可以对以图 1.3 所示方式连接的软件狗进行编辑。我们先讲如何在您的应用程序中调用软件狗的存取程序，再讲如何对软件狗进行编辑。

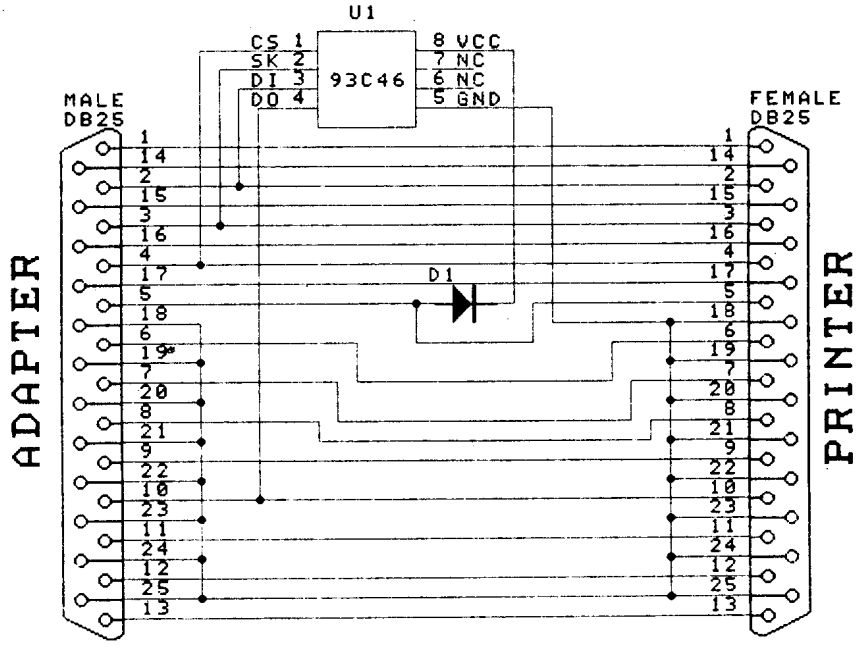

图 1.3　软件狗加密盒的电路连接

为了在您的程序中调用软件狗读写函数，您需要在您的程序中先声明，然后在编译时再连接在一起。以第三代软件狗的存取程序 BOX34.ASM 中的读函数为例，在用汇编语

言、PASCAL 或 C 语言调用时需声明如下：

在汇编语言中：

```
EXTRN AsmBBRead
```

在 PASCAL 语言中：

```
{$F+}
function PasBBRead(Lpt:word;Pos:byte;Buf:pointer;Len:byte):word;external;
{$F-}
```

在 C 语言中：

```
extern int _CBBRead(int LPT,int pos,int *buf,int length)
```

然后，用相应的编译工具把您的应用程序和 BOX34.OBJ 编译连接在一起就可以了。当然，您必须注意 BOX34.ASM 用的是大型存储模式，如果您用的是其它模式，请选用相应模式的存取程序。这里，BOX34S.ASM 是小模式，BOX34C.ASM 是紧凑模式，BOX34M.ASM 是中模式。

1.2.3　软件狗的编辑

黄玫瑰软件制作组在生产软件狗时，已对加密盒中 EEPROM 的存储单元进行了适当的划分，用户只能编辑部分区域。下面结合实用工具 BOXMAKER 的使用，我们来详细讲解对软件狗的编辑操作。

首先，用 XCOPY 命令把软盘上的程序拷到您的机器上，然后进入子目录 BOX\UTIL，运行 MBOX.EXE 文件，出现如下界面：

```
BoxMaker Version 2.0 (C)1993 Yellow Rose Software Co. [w]

========== KingSoft ==========
Port: 03BC   Box3: UnChecked   File: .INI   UserNo: 19930514 Mode: Auto
N: Make New Box      D: Dump Buffer      S: Set Port
0: Clear Box         C: Clear Buffer     X: Read Search
1: Fill Box          E: Fill Buffer      W: Write Search
F: Load File         L: Load Box         P: Write Box
O: Save File         M: Dump Box         V: Verify Box
I: File Format       Z: Set UserNo       A: Multiple W&V
R: Dir Files         +: Valid Soft       4: Valid Box Soft
T: Type File         *: Valid All Soft   8: Valid Box All
Q: Quit              /: InValid All Soft 6: InValid All Box
==========================================================
Choose:
```

这是BOXMAKER 实用工具的主菜单。版本说明下面的第一栏是信息状态，它表示 BOXMAKER 的当前状态。其中 Port：03BC 表示 BOXMAKER 默认软件狗插在其基地址为 3BCH 的并行口上。Box3：UnChecked 表示 BOXMAKER 在并行口上未找到能识别的软件狗，如果软件狗正确插在并行口上，则显示 Box3：Formated，或 Box3：Unformat。Formated 表示软件狗上的数据已格式化，Unformat 表示软件狗上的数据未格式化。File：.INI 表示默认的文件格式为.INI 型。UserNo：19930514 表示下一个用户序列号。Mode：Auto 表示在进行批量生产软件狗时，用户序列号自动递增。状态信息行以下部分是 27 个命令提示，代表了 BOXMAKER 的所有功能。最下面一行 Choose：是提示信息，请输入命令的意思。

BOXMAKER 对软件狗的操作如下图 1.4 所示：

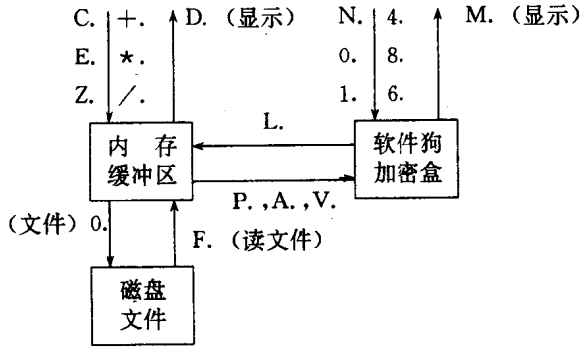

图 1.4　BOXMAKER 对软件狗的操作示意图

利用这个实用工具，既可以根据自己的需要对特定软件狗设置密码，又可以在研制和生产过程中对软件狗进行测试和批量写入。

下面对 BOXMAKER 的功能作一解释。假定软件狗已经插在 LPT1 上。

一、设置并行口

有三种方式可以设置软件狗插在哪个并行口上，这三种方式是 S、X、W 命令。

1. 输入命令 S，将出现如下提示：

```
Choose: S
Select Port: (1)-LPT1:03BC (2)-LPT2:0378
```

可以输入 1 或 2 直接选择一个并行口。

2. 输入命令 X，BOXMAKER 以读系统标识的方式搜寻软件狗插在哪个并行口上。

3. 输入命令 W，将出现如下提示：

```
Choose: W
Write Search will overwrite the Box, countinue (Yes/No)?
```

BOXMAKER 以写入的方式搜寻软件狗插在哪个并行口上，这将覆盖加密盒的所有数据，输入“N”则返回主菜单，输入“Y”则依次出现如下提示：

```
Search found at port: 03BC
press any key to countinue...
```

表示已在基地址为 03BCH 的并行口找到了软件狗。

二、初始化软件狗加密盒

初始化命令有 N、0、1 这三个，是直接对加密盒进行操作。操作的结果可以用 M 命令观看，M 命令是直接把加密盒中的数据按一定格式在屏幕上显示。

1. 输入命令 N，将出现如下提示：

```
Choose: N
Initialize will lost all Soft on the Box, Countinue (Yes/No)
```

初始化操作将丢失加密盒的所有数据，输入“N”则返回主菜单，输入“Y”则依次出现如下提示：

```
03BC Programming...
Make New Box3 successful
press any key to countinue...
```

初始化成功，按任一键返回主菜单。这时候，如果用 M 命令查看加密盒中的内容，将出现如下画面：

```
================ Dump Structed Box3 Buffer ================
System Area :
.....................................................................
  ID[Yellow Rose B3]  RndNo[1997]  UserNo[93051405]  Reserved[2A2A2A2A2A2A2A2A]
 ....................................................................
User Area :
..........
No 1—[5757.57.57]x   No 2—[5757.57.57]x   No 3—[5757.57.57]x   No 4—[5757.57.57]x
No 5—[5757.57.57]x   No 6—[5757.57.57]x   No 7—[5757.57.57]x   No 8—[5757.57.57]x
No 9—[5757.57.57]x   No10—[5757.57.57]x   No11—[5757.57.57]x   No12—[5757.57.57]x
No13—[5757.57.57]x   No14—[5757.57.57]x   No15—[5757.57.57]x   No16—[5757.57.57]x
No17—[5757.57.57]x   No18—[5757.57.57]x   No19—[5757.57.57]x   No20—[5757.57.57]x
GlobalCheckSum[273A]√      UserReserved[575757575757]      UserNo[93051405]
------------------------------ Hex Dump ------------------------------
```

```
00—  0:  59 65 6C 6C 6F 77 20 52 6F 73 65 20 42 33 97 19 39 33 30 35 31 34 30 35
18— 24:  2A 2A 2A 2A 2A 2A 2A 2A 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57
30— 48:  57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57
48— 72:  57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57
60— 96:  57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 57 3A 27 57 57 57 57 57 57
78—120:  39 33 30 35 31 34 30 35
———————————————————— ASCII Dump ————————————————————
Yellow Rose B3 93051405 * * * * * * * * WWWWWWWWWWWWWWWWWWWWWWWWW
WWWWWWWWWWWWWWWWWWW:'WWWWWW93051405
═══════════════════════════════════════════════════════════
press any key to countinue...
```

这是 BOXMAKER 对加密盒的初始化数据，它与版本号和具体的软件狗购买者有关。读者知道，93C46 的存储量为 64 个字，也就是 128 个字节，黄玫瑰软件制作组对此进行了适当的划分，分为系统区和用户区两大部分，也就是上面的 System Area 和 User Area。

系统区又分为系统标识(ID)、随机号码(RndNo)、系统用户号码(UserNo)、软件狗开发者保留区(Reserved)等 4 部分。

用户区分为 20 个软件标识(Software ID)、20 个软件标识的总校验和(GlobalCheck-Sum)、系统用户保留区(UserReserved)、一般用户序列号(UserNo)等 4 部分。其中，软件标识的前 2 个字节表示输出密码，中间的一个字节表示可安装次数，最后的一个字节前 4 位表示软件版本号，后 4 位表示软件标识的校验和。显示的软件标识后打×的表示无效，打√的表示有效。无效的软件标识可用主菜单上的命令 4 使它变成有效。

在 Hex Dump 区显示的是上述加密盒中数据的 16 进制表示。

在 ASCII Dump 区显示的是上述加密盒中数据的 ASCII 码表示。

2. 输入命令 0，将出现如下提示：

```
Choose: 0
Fill out the Box with [00] (Yes/No)?
```

用 00H 填写加密盒的存储单元，输入“N”则返回主菜单，输入“Y”则依次出现如下提示：

```
03BC Programming...
Box Fill (00) OK!
press any key to countinue...
```

填写成功，按任一键返回主菜单。这时侯，如果用 M 命令查看加密盒中的内容，将出

现如下画面：

```
═══════════════ Dump Structed Box3 Buffer ═══════════════
System Area :
.....................................................................
ID[              ] RndNo[0000] UserNo[          ] Reserved[0000000000000000]
.....................................................................
User Area :
..........
No 1--[0000.00.00]x  No 2--[0000.00.00]x  No 3--[0000.00.00]x  No 4--[0000.00.00]x
No 5--[0000.00.00]x  No 6--[0000.00.00]x  No 7--[0000.00.00]x  No 8--[0000.00.00]x
No 9--[0000.00.00]x  No10--[0000.00.00]x  No11--[0000.00.00]x  No12--[0000.00.00]x
No13--[0000.00.00]x  No14--[0000.00.00]x  No15--[0000.00.00]x  No16--[0000.00.00]x
No17--[0000.00.00]x  No18--[0000.00.00]x  No19--[0000.00.00]x  No20--[0000.00.00]x
GlobalCheckSum[0000]√      UserReserved[000000000000]      UserNo[          ]
──────────────────────── Hex Dump ────────────────────────
00--  0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
18-- 24: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
30-- 48: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
48-- 72: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
60-- 96: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
78--120: 00 00 00 00 00 00 00 00
──────────────────────── ASCII Dump ─────────────────────────


═══════════════════════════════════════════════════════════
press any key to countinue...
```

加密盒中所有单元已用 00H 填写。

3. 输入命令 1，将出现如下提示：

```
Choose: 1
Fill out the Box with [FF] (Yes/No)?
```

用 FFH 填写加密盒的存储单元，输入"N"则返回主菜单，输入"Y"则依次出现如下提示：

```
03BC Programming...
Box Fill (FF) OK!
press any key to countinue...
```

填写成功，按任一键返回主菜单。这时侯，如果用 M 命令查看加密盒中的内容，将出现如下画面：

```
================ Dump Structed Box3 Buffer ================
System Area :
.....................................................................
ID[              ] RndNo[FFFF]  UserNo[          ] Reserved[FFFFFFFFFFFFFFFF]
.....................................................................
User Area :
..........
No 1—[FFFF.FF.FF]x No 2—[FFFF.FF.FF]x No 3—[FFFF.FF.FF]x No 4—[FFFF.FF.FF]x
No 5—[FFFF.FF.FF]x No 6—[FFFF.FF.FF]x No 7—[FFFF.FF.FF]x No 8—[FFFF.FF.FF]x
No 9—[FFFF.FF.FF]x No10—[FFFF.FF.FF]x No11—[FFFF.FF.FF]x No12—[FFFF.FF.FF]x
No13—[FFFF.FF.FF]x No14—[FFFF.FF.FF]x No15—[FFFF.FF.FF]x No16—[FFFF.FF.FF]x
No17—[FFFF.FF.FF]x No18—[FFFF.FF.FF]x No19—[FFFF.FF.FF]x No20—[FFFF.FF.FF]x
GlobalCheckSum[FFFF]x      UserReserved[FFFFFFFFFFFF]          UserNo[        ]
———————————————————— Hex Dump ————————————————————
 00— 0:  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
 18— 24:  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
 30— 48:  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
 48— 72:  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
 60— 96:  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
 78—120:  FF FF FF FF FF FF FF FF
———————————————————— ASCII Dump ————————————————————


==========================================================
press any key to countinue...
```

加密盒中所有单元已用 FFH 填写。

三、初始化编辑缓冲区

初始化缓冲区的命令有两个，C 和 E，结果可以用 D 命令查看，D 命令是把缓冲区中的数据按一定格式在屏幕上显示出来。

1. 输入命令 C，将出现如下提示：

```
Choose: C
Fill out the Buffer with [00] (Yes/No)?
```

用 00H 填写缓冲区的存储单元，输入“N”则返回主菜单，输入“Y”则依次出现如下提示：