

一种基于机器特征的软件版权保护方法

陈志贤^{1,2}

(1 南京工业大学 信息学院, 江苏 南京 210009; 2 南京大学 软件新技术国家重点实验室, 江苏 南京 210093)

摘 要: 设计了一种基于机器特征的软件版权保护方法, 给出了软件版权保护的模型, 着重阐述了基于部件特征值的序列号生成机制和实施步骤, 利用虚拟机保护来加密核心代码, 软件加壳等几个关键实现. 实验结果表明该方法可行的、有效的, 且具有良好的可扩展性.

关键词: 机器特征; 序列号; 版权保护; 虚拟机; 壳

中图分类号: TP301.1

文献标识码: A

文章编号: 1000-7180(2010)07-0007-03

A Method for Software Copyright Protection Based on Machine Feature

CHEN Zhi-xian^{1,2}

(1 College of Electronics & Information Engineering, Nanjing University of Technology, Nanjing 210009, China;
2 National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China)

Abstract: In this paper, a method for software copyright protection based on machine features is introduced. The model is given first, and then product key generating mechanism based on characteristic values of machine assembly and corresponding enforcement steps, encryption of core codes using virtual machine protection and software shell adding are presented in detail respectively. The experimental results show the feasibility and effectiveness of the system. Meanwhile it also has good scalability.

Key words: machine feature; product key; copyright protection; virtual machine; shell

1 引言

软件产业代表着科技发展的最新方向, 正逐渐成为信息产业的核心和灵魂, 引领和推动全球范围科学技术的快速发展. 但是, 软件产业的保护却面临着困难, 成为制约我国软件产业快速发展的一个重要因素. 因此, 探索符合时代发展潮流的保护模式已成当务之急^[1].

软件开发工作量大、投资多, 而软件复制容易且费用低廉. 因此, 软件保护工作是维护软件开发人员利益的必要手段, 是软件开发过程中的重要环节.

2 常见的软件版权保护技术

如何保护软件著作权, 防止非法用户盗版, 一直是广大软件开发者不断研究的课题. 目前, 软件的版权控制主要有以下几种较为常见的技术^[2-3]:

(1) 序列号保护方式

序列号又称为注册码, 随软件发布, 软件必须经过注册才能正常使用. 这种保护实现起来比较简单, 目前互联网上大约 80% 的软件都采用这种方式. 但早期的这种方式存在较大的缺陷: 易破解, 易盗版.

(2) Key File 保护方式

Key File(注册文件)是一种利用文件来注册软件的保护方式. 在软件每次启动时检查某个确定文

件的有效性,这个文件即 Key File,需要用户向软件作者付费购买。

(3) 磁盘保护

通过在软盘上格式化一些非标准磁道,在这些磁道上写入一些数据,如软件的解密密钥等,将软盘做成钥匙盘。然后软件运行的时候会去校验软盘中的数据是否正确,判断其是否为合法的钥匙盘,例如大家都很熟悉的杀毒软件 KV300 就是如此。

(4) 软件狗保护技术

软件狗是一种智能型加密工具,也称为加密锁。它是一个安装在并行口、串行口等接口上的硬件电路,同时有一套适用于各种语言的接口软件和工具软件。其缺点是成本较高,且不利于网络发布。

(5) 与硬件信息绑定的序列号保护技术

它将软件与机器硬件信息结合。用户得到共享软件后,安装时软件从用户的机器上取得该机器的一些硬件信息(如硬盘序列号、BIOS 序列号等)。具有成本低廉、方便易用等优点。

此外,还有 License 保护、时间限制、菜单功能限制、CD-Check、软件压缩等保护技术,这里就不再赘述。

3 基于机器特征的软件版权保护方案

设计了一种基于机器特征的序列号保护方案。该方案能克服传统序列号保护方式存在的缺陷,实现一机一码,安全性较高,且采用纯软件技术实现、对硬件无任何额外要求,具备良好的可扩展性。

3.1 方案描述

假定一台机器 m 是由 $x_1, x_2, \dots, x_n$ 等部件组成,这些部件可以是主板、CPU、硬盘、网卡等。选取物理序列号全球唯一且只读的部件,使该物理序列号可唯一标识该部件;那么多个部件的特征值通过自定义的逻辑运算综合处理函数处理后即可获得唯一表征该机器的总特征值。然后运用消息摘要算法 MD5 加密总特征值,可生成与总特征值唯一对应的密钥,该密钥就是被保护软件在这台机器上合法使用的序列号。

基于机器特征的序列号生成的具体步骤如下:

(1) $x_i \xrightarrow{f} f(x_i), f(x)$ 为部件 x 的特征提取函数。即为标识部件 x_i 的特征值字符串。显然, x_i 与 $f(x_i)$ 之间是一一映射关系;

(2) $f(x_1), f(x_2), \dots, f(x_n) \xrightarrow{\varphi} \varphi(m) = \varphi(f(x_1), f(x_2), \dots, f(x_n)), \varphi(m)$ 是可唯一表征

该机器的总特征值,通过设计中不予公开的逻辑运算综合处理函数获得。

易证得 $m_1 \neq m_2 \Rightarrow \varphi(m_1) \neq \varphi(m_2)$ 。

(3) $\varphi(f(x_1), f(x_2), \dots, f(x_n)) \xrightarrow{\text{MD5}} \text{MD5}(\varphi(f(x_1), f(x_2), \dots, f(x_n)))$, 即利用 MD5 算法产生该机器总特征值的消息摘要。

(4) 该消息摘要为 128 位二进制码,为了方便用户输入,可进一步将其转化为 32 位十六进制码,作为最终的用户授权序列号。

序列号的合法性验证的具体实施步骤:

(1) 首先提取出某机器中各个部件的特征值 $f(x_i), 1 \leq i \leq n, n$ 为参与特征提取的机器部件数量;

(2) 运用逻辑运算综合处理函数对各部件的特征值进行处理,以获得唯一表征该机器的总特征值字符串 $\varphi(f(x_1), f(x_2), \dots, f(x_n))$;

(3) 利用 MD5 算法产生该机器总特征值的消息摘要;

(4) 读取机器中预先存放在系统注册表或密钥盘中的序列号,将之与步骤(3)中得到的消息摘要进行比较,如果相等,则说明此软件已注册,为合法授权用户;否则就是未经授权的非法盗版用户,应立即停止程序运行。

3.2 利用 VMProtect 保护关键代码

目前很多软件解密者都采用逆向分析技术,通过分析反汇编可执行程序的代码,来理解其代码功能,找到序列号判断的关键位置,如图 1 中的虚线方框所标处,通过修改可执行程序将之旁路,令软件执行流程直接从步骤(3)跳转到步骤(5),从而达到绕过用户的序列号合法性校验代码,使上述的软件保护机制失效。

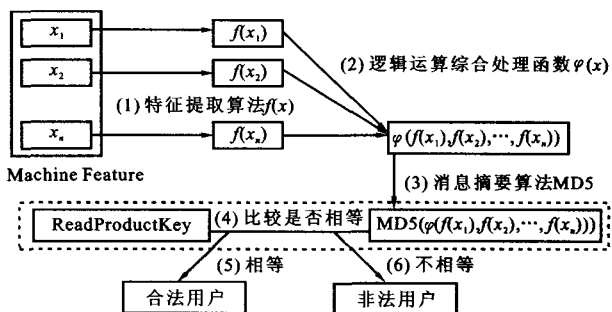


图 1 基于机器特征的软件版权保护模型

为此,在本设计中采用目前业界较为先进的 VMProtect 虚拟机保护软件对程序的关键代码进行保护。VMProtect 是新一代的软件保护系统,可以修

改软件产品的源代码,转换部分代码为在虚拟机上运行的字节码(bytecode),即把保护后的代码放到虚拟机中运行,这将使分析反编译后的代码和破解变得极为困难.不妨将虚拟机想象成为带有不同于 Intel 8086 处理器系统指令的虚拟处理器.例如,虚拟机没有比较两个操作数的指令,也没有条件跳转和无条件跳转指令等等.这样一来,破解者就需要开发一整套的解析引擎来分析和反编译字节码,以现有的解密理论,破解者想要还原出源代码几乎是不可能的^[4].

VMPProtect v1.2 以上支持 SDK 了,可以编程时插入一个标记,然后在加密时,VMPProtect 会认出这些标记,并在有标记的地方进行保护.在程序源码中,用这对标签将一些核心代码包含,编译成 EXE 文件.本设计中调用 VMPProtect SDK 保护,关键代码的伪代码如下:

```
asm
db $ EB, $ 10, 'VMPProtect begin', 0 //标记开始处
end;
//程序的关键代码
if ReadProductKey = md5 ( Encry ( GetHDSerialNum,
GetMacAddress))
//合法授权用户
else
//非法盗版用户
asm
db $ EB, $ 0E, 'VMPProtect end', 0//标记结束处
end;
```

VMPProtect 是当前最强的虚拟机保护软件,经过 VMPProtect 处理的软件基本是没法分析原程序思路的,关键是一定要程序关键代码进行处理.另外,经虚拟机处理代码效率会降低,因此一些对效率要求比较高的代码就不适合用 VMPProtect 进行处理.

3.3 软件压缩加壳

在虚拟机保护的基础上,对生成的软件进行压缩加壳,即利用特殊的算法,对 EXE、DLL 文件里的资源进行压缩、加密.如上述 VMPProtect 处理过的软件,可以继续用 ASPProtect, Themida 等加壳软件进一步保护.

4 实验及结果分析

为了证明上述方案的可行性,对其进行详细的仿真和测试,随后对该软件版权保护方案的性能进行简要分析.

实验环境:操作系统为 Windows XP SP3,开发平台为 Delphi 7.0,虚拟机保护软件为 VMPProtect 1.22.3 版^[4],加壳工具为 ASPProtect 1.33 版^[5].选取的机器部件为硬盘、网卡.其中主机 A、B 的配置一致,CPU: Intel Core2 Duo T7250 2.00GHz;硬盘: FUJITSU MHW2120BH; 无线网卡: Intel PRO/ Wireless 3945ABG Network Connection.在主机 A 上运行的注册机软件界面如图 2 所示.

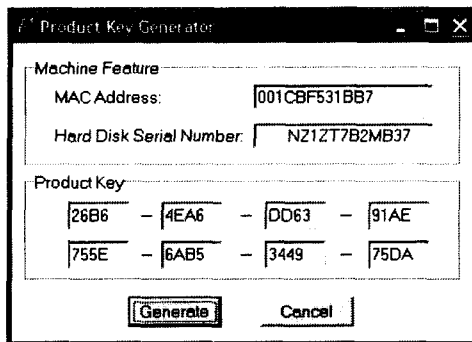


图2 注册机软件界面

两台实验机器的硬件配置完全一致,但在主机 A 上生成的序列号在主机 B 上无效,充分说明了该保护方法是跟机器的具体部件特征值相绑定的,同时机器中的任意一个部件更换都将导致原序列号失效,因此是可行的、有效的.选取的机器部件可以动态变化,说明该方法具有良好的可扩展性.

运用虚拟机保护软件和加壳工具对示例程序加密,文件大小的前后变化情况如表 1 所示.

表1 文件大小的变化情况比较

原始文件大小	VMPProtect 加密后	ASPProtect 加壳后
544kB	582kB	400kB

从表 1 可以看出,由于 ASPProtect 对示例程序的压缩及加密,最终生成的程序在文件大小上比原始程序小了约 26%,同时在反编译、防盗版性能上有了显著的提高.当然,利用 VMPProtect 加密也存在一些缺点,如影响程序的运行速度,因此不适合一些对速度要求很高的场合.

5 结束语

本文所设计的基于机器硬件特征的软件版权保护方法,在满足安全性的基础上,采用机器部件的特征信息绑定序列号,实现一机一码,克服了传统序列号保护方式的缺陷,同时具有良好的反汇编和反跟踪能力,具有一定的实用价值. (下转第 14 页)

25mW.

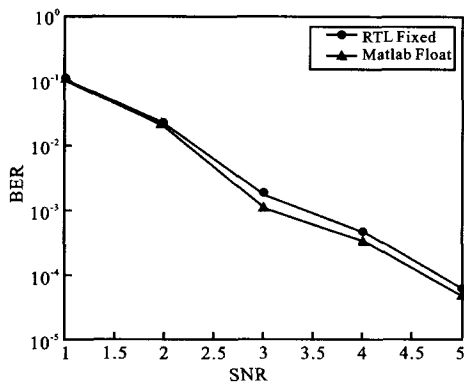


图8 RTL定点和Matlab浮点误码率比较

表4是本设计与其他 Viterbi 译码器的比较. 对比可知, 本设计的面积是这些设计中最小的, 设计的功耗也是最小的, 适合于未来低功耗 SOC 的集成使用.

表4 实现结果比较

	文献[8]	文献[9]	本文
描述	(2,1,7) 多码率	(2,1,7) 多码率	(2,1,7) 多码率
工艺/ μm	0.18	0.25	0.13
并行形式	全并行	-	全并行
频率/MHz	72	80	240
吞吐量/(Mb/s)	54	54	480
Core size/ mm^2	3.06	5.91	0.36
功耗/mW	68	53	25

6 结束语

针对 IEEE 802.11n 收发机对信道编译码器的多码率、高吞吐率的要求, 设计了一个多码率、高吞吐率、低功耗的 Viterbi 译码器, 并采用一种新的寄存器交换结构来降低幸存路径存储单元的功耗. 基于 SMIC 0.13 μm 工艺实现, 译码器具备面积小 (0.36 mm^2)、低功耗 (25 mW)、高吞吐率 (480 Mb/s),

可用于高数据吞吐率、低功耗的无线 WLAN 系统.

参考文献:

- [1] IEEE - SA Standards, IEEE 802.11n TM/D2.00 - 2007, Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications[S]. New York, USA, IEEE Stand, 2007.
- [2] Viterbi A J. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm [J]. IEEE Trans. Inf. Theory, 1967(13): 260 - 269.
- [3] Forney J G D. The viterbi algorithm[J]. IEEE, 1973, 61 (3): 268 - 278.
- [4] 黄君凯, 王鑫. 一种高速 Viterbi 译码器的优化设计及 Verilog 实现[J]. 微电子学与计算机, 2005, 22(2): 178 - 182.
- [5] 韩腊萍, 陈立潮. 改进 Viterbi 算法在机动目标跟踪中的应用[J]. 微电子学与计算机, 2004, 21(8): 111 - 114.
- [6] Lou H L. Linear distances as branch metrics for viterbi decoding of trellis codes[J]. ICASSP, 2000 (6): 3267 - 3270.
- [7] Kamuf M, Owall V, Anderson J B. Optimization and implementation of a viterbi decoder under flexibility constraints[J]. IEEE Transactions on Circuits and Systems - I: REGULAR PAPERS, 2008, 55(8): 2411 - 2422.
- [8] Chien Ching Lin, YenHsu Shih. Design of a power-reducing viterbi decoder for WLAN applications [J]. IEEE transactions on circuits and systems - I: Regular Paper, 2005, 52(6): 1148 - 1156.
- [9] Alfonso Troya, Koushik Maharatna. Low-power VLSI implementation of the inner receiver for OFDM-based WLAN systems [J]. IEEE Transactions on Circuits and Systems - I: Regular Papers, 2008, 55(2): 672 - 686.

作者简介:

朱勇旭 男, (1983-), 博士研究生. 研究方向为超大规模集成电路设计.

吴斌 男, (1976-), 博士, 副研究员. 研究方向为宽带通信 SOC 研发.

(上接第9页)

参考文献:

- [1] 中华人民共和国知识产权局. 软件保护如何才能“硬”起来[EB/OL]. (2006-06-26). <http://www3.cntnm.e.com/sipo2008/mjji/2006/200804/t20080401-362177.html>.
- [2] 段钢. 加密与解密[M]. 3版. 北京: 电子工业出版社, 2008.
- [3] 鲁保玉, 韩彦锋. 用机器指纹技术实现 Delphi 应用程序的版权控制[J]. 计算机应用与软件, 2005, 22(8): 140 - 142.

[4] VMProtect Software. VMProtect[EB/OL]. [2009-06-11]. <http://www.vmprotect.ru/vmprotect.php>.

[5] ASPACK Software. ASProtect[EB/OL]. [2009-06-11]. <http://www.aspack.com/asprotect.aspx>.

作者简介:

陈志贤 男, (1979-), 在职博士后. 研究方向为网络安全、数据挖掘、可信计算及计算机在通信中的应用等.