一种基于 WMI 和 Hash 算法的软件激活方案①

吴建军 (浙江师范大学 行知学院 浙江 金华 321004)

摘 要： 软件激活是对计算机软件进行保护的一种手段。基于 WMI，可以获得用户计算机的身份识别信息；对该计算机身份识别信息进行 Hash 算法可获得“硬件指纹”，同时又能保护用户隐私。结合使用软件安装序列号，配合 SHA512 生成的激活码，可实现一个安全、实用的软件激活方案。

关键词： 软件激活；WMI；Hash 算法；SHA512

A Software Activation Scheme Based on WMI and Hash Algorithm

WU Jian-Jun

(Xingzhi College, Zhejiang Normal University, Jinhua 321004,China)

Abstract: Software activation is a means to protect the computer software. It could get the identifying information of personal computers' identities based on WMI and gain "Hardware fingerprint" through Hash algorithm of figure identifying information in the case of protecting the user's privacy. Using software installation serial number, together with activation code generated with SHA512 can achieve a safe and useful software activation scheme.

Keywords: software activation; WMI; hash algorithm; SHA512

1 引言

计算机软件是现代社会主要的技术基础之一，是信息时代的重要产物，对软件的保护问题已经成为当今世界保护知识产权的一项重要的内容，受到了各国政府的普遍重视。在采用软件著作权法等法律保护手段的同时，软件设计者设计“软件激活”是一种重要的技术保护手段。对计算机软件进行技术保护的手段很多，激活技术是一种软件知识产权保护技术，用来识别软件产品是否经过合法授权。

本文设计的软件激活方案，基于 Windows 操作系统，使用 Visual Studio 2008 软件设计平台实现，用户通过匿名的方式对软件进行合法激活，不会妨害用户的隐私和安全，同时也确保了软件作者的合法权益。

2 采用WMI技术获取用户计算机识别信息

2.1 WMI

在 Windows 操作系统下，基于“Windows 管理规范”(Windows Management Instrumentation)，简称 WMI，WMI 提供程序在 WMI 和操作系统、应用程序以及其它系统的组件之间充当中介。“系统属性”、“系统信息”和“服务”等 Windows 组件，是基于启用 WMI 的，借此可以收集和显示系统配置等信息。

当前，计算机主要硬件中都由硬件厂家设置了单独的序列号，该信息是惟一的，用于硬件的身份确认。基于 WMI 获取用户计算机硬件的身份确认信息，经过适当处理后，将该信息作为软件激活的身份识别。只有获取了用户计算机的硬件识别信息，才能为软件激活做好准备。

2.2 获取硬件信息的程序设计

在 Visual Studio 2008 软件设计平台，结合 WMI 技术，可按如下方式获取计算机硬件信息。

(1) 在 Visual Studio 2008 的编程项目中，添加引用“System.Management”是必须的，它确保本项目中，能正确应用 WMI 技术。

(2) 获取计算机硬件信息

例如，获取主板序列号，代码如下：

① 收稿时间:2009-07-20;收到修改稿时间:2009-09-09

```
' 声明 Wmiboard 为主板信息类(Win32_ BaseBoard)的实例，从中获取主板序列号信息
Dim Wmiboard As New System.Management.ManagementObjectSearcher("SELECT * FROM Win32_BaseBoard")
' 声明当前获取的计算机主板信息字符串变量
Dim myboardSN As String = ""
' 遍历 Wmiboardobj 对象的属性，捕获当前计算机的主板信息，赋值给主板信息字符串变量 myboardSN
Try
    For Each Wmiboardobj As ManagementObject In Wmiboard.Get
        myboardSN = Wmiboardobj ("SerialNumber")
    Next
Catch ex As Exception
End Try
```

上述代码中，从 WMI 的主板信息类“ Win32_BaseBoard ”中，获取用户计算机的主板序列号信息，保存到字符串变量“ myboardSN ”中。用类似的方法，还可以获取硬盘驱动器类“ Win32_DiskDrive ”、CPU 信息类“ Win32_Processor ”等其它硬件信息。在实际使用中，主板、硬盘、CPU 等硬件的序列号都是惟一的，将它们的信息进行处理，形成本激活方案中的用户计算机身份识别信息。

3　Hash算法简介

密码学中的 Hash 函数是一种将任意长度的消息压缩到某一固定长度的消息摘要的函数，利用它可以对数据产生一个短的“ 指纹 ”(即消息摘要)[1]。本文中，将计算机硬件信息进行 Hash 算法处理，获得了“ 硬件指纹 ”。Hash 函数是一种单向算法，一旦数据被转换，将无法再获得其原始值。当一段纯文本时进行 Hash 运算时，即使只更改段落中的一个字母，随后的散列计算都会产生不同的值；即使只更改了消息的一个位，强哈希函数就可能生成相差 50% 的输出。

在.Net Framework 开发环境中，可以使用多种 Hash 算法，常用的是 SHA1 和 MD5，本方案中采用 SHA512。2004 年，我国密码学者王小云教授成功破译了 MD5 算法，因此认为这种算法已不再安全。而后，SHA1 算法也暴露出不安全的方面，因此，本案例中采用更为安全的 SHA512 算法[2](它是当前 SHA 系列算法中的最高版本)。在当前.Net Framework 中的 SHA512 算法，可以获得哈希值是 512 位的，攻击者对其生日攻击[3]需要约 2256 的 Hash 计算，它较其它的 128 或 256 位(生日攻击次数对应为 264、2128)Hash 算法来说，安全性有了显著提高。

4　软件激活方案设计

4.1 激活方案概述

基于对 WMI 和 Hash 算法的应用，形成本文的软件激活方案。首先，用户的软件安装包是相同的，但合法用户还拥有一个由软件商提供的惟一、无序的软件序列号；一般地，该序列号可以在购买软件时获得。然后，如图 1 所示进行软件安装激活，软件安装时生成一个身份识别信息(相当于用户计算机硬件信息指纹)，并由用户输入软件序列号，一起发送给软件供应者；在软件供应者端，核对序列号后，生成激活码发送给用户。最后，用户在确认激活时，软件将从供应商获得的激活码与自身预设算法的结果进行比较，完成软件激活。

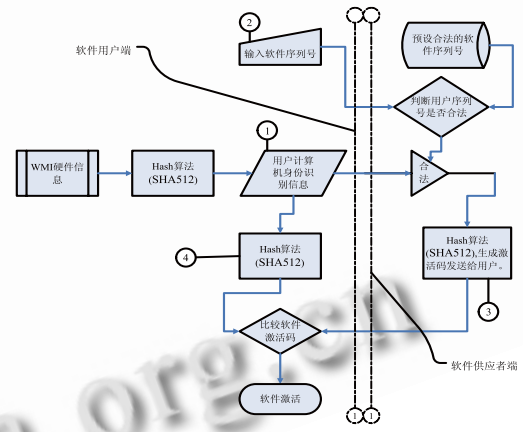

图 1　软件激活方案图

为了保证软件安全，软件序列号仅能有限次使用，必须避免同一序列号被大量安装使用的问题；同时，用户保存好合法的序列号和激活号，就可以在计算机重新安装操作系统后，仍能顺利地使用本软件。

4.2 程序设计

4.2.1 用户计算机身份识别信息

如前文所述，通过 WMI 获取的字符串变量“ myboardSN ”即是识别信息之一，可以根据需要选择一种或几种计算机硬件信息进行组合；再将该信息进行哈希算法 SHA512，获得计算机身份识别信息。

4.2.2 生成激活码

在 Visual Studio 2008 软件设计平台，使用哈希算法 SHA512 生成激活码。

(1) 导入命名空间

```
' 导入.NET 的命名空间，便于相关对象的引用
Imports System
Imports System.Security
Imports System.Security.Cryptography
Imports System.Text
```

(2) 编写代码，生成激活码

```
' 声明关键硬件信息特征的字符串变量和激活码字符串变量
Dim pwstring As String, myactiveCode As String
' 声明用于 Hash 算法的字节型变量
Dim pwbyte() As Byte, pwHash() As Byte
' 将第 2 节代码中获取的主板信息赋值到当前的硬件信息特征字符串变量中
pwstring = myboardSN
' 将硬件信息特征字符串转换为字节流，为 Hash 算法做准备
pwbyte = ASCIIEncoding.ASCII.GetBytes (pwstring)
' 使用 SHA512 算法，获得关键硬件信息的 Hash 值
pwHash = New SHA512Managed().ComputeHash(pwbyte)
' 将已获取的硬件信息 Hash 值(字节流)转化为字符串，作为激活码字符串
myactiveCode = ByteToString(pwHash)
' 设计以下函数，将字节流转换为字符串
Private Function ByteToString(ByVal pwdbyte() As Byte) As String
Dim i As Integer
Dim outstring As New StringBuilder (pwdbyte.Length)
For i = 0 To pwdbyte.Length – 1
  outstring.Append(pwdbyte(i).ToString("X2"))
Next
Return outstring.ToString()
End Function
```

上述程序中，将用户计算机注册信息“ myboardSN ”通过 SHA512 算法得出该组合信息的散列值，并将其转化为激活码字符串“ myactiveCode ”，该激活码信息作为用户软件的激活信息。

(3) 用户计算机对激活码的校验

在用户端的对应软件中，也设置了同样算法的程序，它在后台将当前计算机算出待核定的激活码，当用户获得软件设计者的激活码时，将两者进行核对，如果完全相同，则激活软件。

(4) 其它技术细节

为了确保用户在软件更新或重新安装后，不需要重新输入激活码，可以针对该软件设置注册表项，将该软件的激活码存放在注册表中，软件自动读取该激活码，避免用户重新输入激活码带来的麻烦。写入注册表代码如：

```
My.Computer.Registry.SetValue("HKEY_CURRENT_USER\Software\CompanyName\ProductName\KeyName", "Name", "value")
```

另外，用户将计算机身份识别信息给软件供应者时，除了使用 Web 网站注册，也可参考如下代码，将用户信息("c:\File.txt")发送到指定位置：

```
My.Computer.Network.UploadFile("c:\File.txt", "ftp://course.zjnu.cn/jh/", "username", "password")
```

从指定位置下载信息：

```
My.Computer.Network.DownloadFile("ftp://course.zjnu.cn/jh/File.txt",
Environment.CurrentDirectory + "\" + "File.txt", " username ", " password ", False, 20000, True)
```

配合使用上述手段，可以使软件对激活信息的收发是后台实现的，用户仅手工输入软件序列号，其它操作通过点击按钮即可完成。

5 总结

综上所述，采用 WMI 和 Hash 算法(SHA512)，从实用的角度设计了一种软件激活方案。将用户计算机 WMI 硬件信息进行 SHA512 获得注册身份信息，既保护了用户隐私，又确保了计算机身份识别。同时，在 .Net Framework 环境中，可以方便地使用 SHA512 处理信息，确保激活方案的安全。虽然，破解激活方案的手段是可能实现的，但作为实际应用中，破解成本与合法使用软件成本不匹配时，软件的保护就达到保护目的了。

参考文献

1 张福泰,李继国,王晓明,等.密码学教程.武汉:武汉大学出版社, 2006. 112－121.
2 National Institute of Standards and Technology. Secure Hash Standard. VA22161, Springfield National Technical Information Service, 200.
3 Yuval G. How to Swindle Rabin. Crypto logia, 1979, 3(3):187－190.