

基于 WCF 的软件在线注册方案^①

吴建军

(浙江师范大学 行知学院, 金华 321004)

摘 要: 计算机软件在线注册是实现软件版权保护的主要技术方案。WCF 技术实现了跨平台的通信, 为不同应用平台下的应用软件提供了统一的在线注册通信。同时, .Net 的加密技术为整个注册过程提供了安全保证。整个注册方案是安全、可靠和成本可控的。

关键词: Web 信息提取; 知识表示; 数据密集型 Web 页面; 基于本体的关键词库

Software Online Registration Solution Based on WCF

WU Jian-Jun

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

Abstract: Computer software online registration is to protect software copyright the main Technology. WCF technology has achieved a cross-platform communication for application software running on different platform provides a unified register online communications. At the same time, .Net encryption technology for the entire registration process to provide a security guarantee. The registration solution is safe, reliable and cost-controlled.

Key words: software online registration; WCF; .encryption technology

1 引言

计算机软件版权保护是知识产权保护的一项重要内容, 是软件行业发展的重要保障。从软件开发技术上实施软件版权保护是软件开发者的重要课题, 一套安全可靠、成本可控、使用简单的软件注册方案是进行软件保护的主要技术方法。笔者在本刊 2010 年 04 期的“一种基于 WMI 和 Hash 算法的软件激活方案”一文中, 对软件采用 WMI 和 Hash 算法而实现软件激活的方法进行了论证, 通过该方法获得了用户计算机的“硬件指纹”, 用户将该信息提供给软件供应商; 供应商通过预置的算法生成软件激活码, 返回给用户; 用户使用该激活码实现软件的正常使用。软件供应商提供给用户的激活码是与用户计算机硬件信息唯一匹配的, 该激活码无法在未经授权的计算机上激活相关软件, 即一个激活码只能为该软件在对应的一台计算机激活, 从而实现了软件版权保护。

但是, 由于整个注册激活的过程未通过网络实施

即时激活(需供应商收集用户注册信息进行处理后, 延时激活), 故对实际应用带来了一定的阻碍。通过对当前常见的软件注册方法比较分析, 笔者发现一些普通的应用软件在授权过程中也没能通过网络即时注册, 对用户的使用亦带来了较大不便; 或者, 一些软件未采用注册手段进行技术保护, 软件供应商的权益容易受到侵害, 这些情形都对软件产业的发展带来了较大的负面影响。因此, 采用一种安全可靠的方法通过网络在线即时注册激活是软件版权实施技术保护的有效手段, 能积极促进软件产业的健康发展。

同时, 当前的软件开发平台众多, 一个软件企业可能会有多个开发平台(如 Java, .Net), 一个软件也可能需多种运行模式(如 B/S、C/S、单机版等), 因此, 一个完整的软件注册方案, 需要能为多个软件, 甚至是为不同平台开发的软件提供统一的注册方法, 这无疑具有重要意义。基于“面向服务的构架”——SOA(Service Oriented Architecture)的理念, 实

① 收稿时间:2011-07-30;收到修改稿时间:2011-09-11

现了跨平台的解决方式^[1]，为本方案提供了软件设计的指导思想；Microsoft 的 WCF(Windows Communication Foundation)是完全基于 SOA 的通信框架^[2]，为本方案的设计和实现提供了全面和完善的技术支持。

2 软件在线注册方案概述

以常见的 Windows 操作系统安装为例，安装后需输入合法的序列号(本文的注册号)，通过网络或电话服务，激活软件，进入正常使用。本方案以此过程为基本思路，结合普通的计算机应用软件应用特点，实现了基于 Internet 注册、自动激活的过程，具有如下主要特点。

① 通用性强

整个方案通过 WCF 实现，不但可以跨开发平台应用，而且一个服务端程序可同时对多个应用软件提供注册功能。在一个 WCF 服务程序中，可以设计多个服务契约(Contracts)，一个契约可为一个软件提供注册，通过 WCF 在 IIS 中的寄宿服务，实现在线注册。在服务端，可设计一个数据库对所有注册信息进行数据管理。

② 安全可靠

当前，网络中的通信存在许多不安全因素。笔者通过技术手段发现，一些大型网站的电子邮件帐户登录过程中，用户的名称和密码是通过明文发送的，如果该信息被侦听即导致帐户泄密。作为保护软件版权的主要技术手段——软件注册，同样面临此类风险，明文传递的注册码和激活码容易导致注册算法的被破解。作为提供软件版权保护的方案，如果不能确保运行的安全，无疑会加大风险。为此，本方案采用了 .Net 开发环境中的 Hash 算法和对称加密技术，将通信数据等相关信息进行了加密处理，为注册方案提供了较好的保护。

同时，注册码采用了较长的字符串(如 16 个字符)，结合程序功能，对猜测注册码非法注册的方式是很容易抵制的。

③ 注册过程简单、快捷

用户安装软件后，初次启动会运行注册激活程序。用户只需在注册界面上输入购买软件时获得的注册码，确保当前计算机正常连接 Internet，就可实现即时注册激活。之后，软件的使用和更新都不受限制；在同一台计算机，在同一个操作系统下，对合法注册过的软件删除后的再安装，可直接激活(不需再输入注

册码)使用。

④ 可控性强

用户的注册码的有效注册次数是有限的，如 2 次。当用户实现成功注册后，服务端的数据库记录该注册码的使用情况；之后，超限次使用注册码不会被服务端接受。而已经合法注册的用户，在重新安装软件时，即使不输入注册码，服务端注册程序也会提供激活码。整个注册方案的注册信息统一保存在数据库中，可以有效实施管理。

本注册方案的 UML 序列图如图 1 所示。

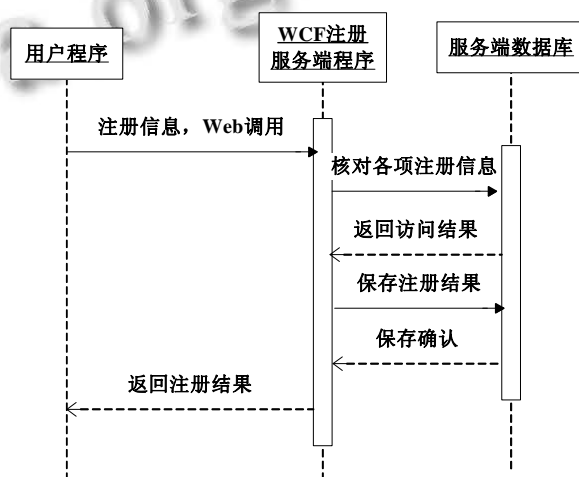


图 1 注册序列图

WCF 注册服务端程序与数据库被布置在 ASP.NET 网站，可通过 IIS 提供服务。用户程序与服务端程序之间的消息通信是加密的，确保通信安全；注册码是否合法，在服务端访问数据库确认；注册结果在数据库中保存。

3 本方案的主要技术

3.1 WMI

在 Windows 操作系统中，通过 WMI(Windows Management Instrumentation)可全面获得计算机硬件信息。在 Visual Studio 开发平台，可通过 System.Management.ManagementObjectSearcher()方法，访问主板信息类“Win32_BaseBoard”、驱动器类“Win32_DiskDrive”、CPU 信息类“Win32_Processor”等各类硬件信息，将这些信息进行综合，就形成了计算机唯一的身份信息。

3.2 加密与解密

本方案对信息的加密与解密技术主要用于两个方面：一是将用于软件注册的信息和激活码以 Hash 算法的单向加密技术进行处理，确保注册结果的唯一性；二是将软件在注册过程中的客户机与服务器之间的注册、激活信息进行对称加密，以较为安全的手段实现注册过程。

3.2.1 Hash 算法

采用 Hash 函数可将任意长度的消息压缩到某一固定长度的消息摘要，本方案对计算机身份信息处理成为一个指定长度的字符串。本方案采用 Visual Studio 中的当前最高版本 SHA512 算法^[3]，将用户计算机的身份信息转换成指定格式的字符串，文中称为“注册信息”。Hash 是一种单向算法，该注册信息是将用户计算机硬件信息通过 Hash 算法获得的单向信息，它既实现了注册信息的格式规范化，又不会直接收集用户的硬件信息，可保护用户计算机的隐私。

3.2.2 Net Framework 环境下的对称加密技术

公共网络（如 Internet）不提供实体间安全通信的方法。如前文所述，这种网络上的通信容易被未经授权的第三方读取，甚至破解注册程序。而加密方法较好地防止他人查看数据，使用对称加密算法对数据进行加密，在加密状态下传输数据，然后由预定的接收方对数据进行解密。如果第三方截获了加密的数据，解密数据是很困难的^[3]。

本方案采用 .Net 的 Rijndael 对称加密算法实施对称加密，此算法支持 128、192 或 256 位的密钥长度，具有足够的安全强度。

3.3 WCF

WCF 是实现软件在线注册的核心技术。WCF 是 Windows 平台下统一的通信基础框架，它整合了 .Net 平台下所有与分布式系统有关的技术^[4]，.Net 为其提供了原生的支持；作为技术应用，WCF 在本方案中的应用由服务器端和客户端两部分组成，客户端程序集成在用户的应用程序中。WCF 服务端程序寄宿于 ASP.NET 的 IIS 服务器，共用 ASP.NET 网站的服务端口；客户端程序通过 Web 服务引用，通过 SOAP 方法实现与服务端的通信，经过 WCF 服务端程序的响应，完成软件注册过程。

在程序设计中，WCF 服务端程序设计主要是声明和设计服务契约(Service Contracts)。

① 在服务端设计服务接口，名为 IService1：

```
<ServiceContract(> Public Interface IService1
```

② 声明服务契约：

```
<OperationContract(> Function GetDataUsing DataContract(ByVal composite As CompositeType) As CompositeType
```

可以根据需要声明多个服务契约。

③ 服务契约编程：

```
Public Class Service1
```

```
Implements IService1
```

```
Public Function GetDataUsingDataContract(ByVal composite As CompositeType) As CompositeType Implements IService1.GetDataUsingDataContract
```

根据服务功能的需要，在各个契约（如 GetDataUsingDataContract()）中进行程序设计，对客户端的通信信息进行处理，将处理结果发回到客户端，并在服务端数据库记录数据。

4 软件在线注册的实现

在线软件注册的通信信息主要由三个方面：一是软件供应方提供的软件序列号（本文称为注册码），一般由数字和字母组成字符串；二是用户计算机的硬件注册信息，本方案将收集用户计算机的唯一信息按指定的 Hash 算法获得散列值，并转换成字符串（本文称为注册信息）；三是软件供应方根据前两项信息进行专用算法生成既定长度（如 16 位）、由数字和字母组成的激活码，使用该激活码激活用户端软件。

4.1 WCF 注册程序的服务端设计

WCF 服务端主要由接口(Interface IService1)和类(Class Service1)组成。Interface IService1 指明了服务操作名称；而所有设置的服务操作，都必须在 Class Service1 中实现。如图 2 所示，本方案在 Interface IService1 中声明了方法名称，如“aoaszcz”；在 Class Service1 中设计了实现相关方法的程序代码。

4.1.1 在 Interface IService1 中声明自定义方法

例如，声明自定义方法“aoaszcz”，代码如下：

```
<OperationContract(> Function aoaszcz(ByVal sfStream As Stream) As Stream
```

这是一个 WCF 的操作契约声明。将通信的信息均设置为文件流 Stream 格式，可更好地利用加密技术，实现安全通信。

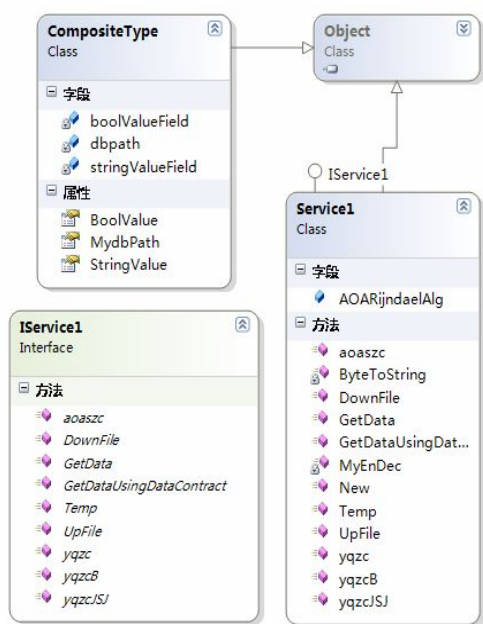


图 2 WCF 服务端对象设计

4.1.2 在 Class Service1 中实现操作契约的程序设计

① 设计对称加密对象

引入对称加密命名空间:

Imports System.Security

Imports System.Security.Cryptography

创建对称加密对象:

Public AOARijndaelAlg As Rijndael = Rijndael.

Create()

初始化对称加密对象 AOARijndaelAlg:

AOARijndaelAlg.Key = System.Text.ASCIIEncoding.ASCII.GetBytes("*****")

AOARijndaelAlg.IV = System.Text.ASCIIEncoding.ASCII.GetBytes("*****")

在 .Net 中, 由 Rijndael 创建的对称加密对象的初始化向量 IV 和密钥 Key 都采用 16 位字符串 (相当于 128 位的密钥长度), 上述代码中的“*”号可以由数字、字母和特殊符号等表示, 具有可靠的保密性能。

② 实现操作契约

以“aoaszc”为例, 在 Class Service1 中实现操作契约。

首先, 创建该契约的方法结构如下:

Public Function aoaszc(ByVal sfStream As Stream)

As Stream Implements IService1.aosaszc

End Function

在上述方法中, 主要是设计三个内容: 一是根据

客户端发送给服务端的全部消息, 即一个加密后的文件流 sfStream, 把 sfStream 解密、解析, 识别用户的注册信息; 二是区分不同的用户请求, 对合法的注册信息生成对应的激活码, 并保存在服务端的数据库中; 三是根据服务端对客户端信息进行的处理, 将处理的结果组合、并加密成文件流 aoaszc 发送回客户端。部分重要代码如下。

其次, 解密用户发送的文件流 sfStream:

```
Dim dsfStream As New CryptoStream(sfStream,
AOARijndaelAlg.CreateDecryptor(AOARijndaelAlg.Key,
AOARijndaelAlg.IV), CryptoStreamMode.Read)
```

```
Dim sfReader As New StreamReader(dsfStream,
System.Text.Encoding.Default)
```

```
Dim sfString As String = sfReader.ReadToEnd()
```

通过上述代码处理, 加密的用户信息文件流 sfStream 转换成了字符串 sfString, 可以对字符串变量进行预设方法的解析处理, 就可以完整地获得用户提交的各项信息。

最后, 加密服务端发送给用户的信息文件流:

```
Dim CaoaszcStream As New CryptoStream
(aoaszcStream,AOARijndaelAlg.CreateEncryptor(AOARijndaelAlg.Key,
AOARijndaelAlg.IV), CryptoStream
Mode.Read)
```

```
Return CaoaszcStream
```

服务端发送回客户端的信息转换成文件流 aoaszcStream, 上述代码中, 使用同样的对称加密方法, 将其加密成 CaoaszcStream; 通过“aoaszc”契约, 将 CaoaszcStream 加密文件流传递给客户端。

4.1.3 服务端应用程序发布

本方案的 WCF 服务端应用程序采用 IIS 寄宿方式。在 Visual Studio 开发环境中, 寄宿 IIS 的方式比较简单, 前提条件是拥有提供 ASP.NET 的网站, 并运行 .Net 3.5 (或更高版本) 环境; 通过 Visual Studio 的 WCF 发布向导, 是服务端程序寄宿于该网站。客户端程序引用该 Web 服务即可调用 WCF 服务端程序。

4.2 客户端程序设计

客户端通过 WCF 通信技术, 将注册信息发送给服务端, 根据服务端发回的信息进行处理, 获得软件的注册激活。

4.2.1 客户端的服务引用

当服务端 WCF 服务程序在 ASP.NET 网站发布

后,客户端的程序开发项目按服务端程序的发布 URL 进行服务引用,例如将该服务引用命名为 WebWCF。WCF 服务的网络引用采用了统一的 SOAP 协议,不同的程序开发平台是可以通用的。

继续,在客户端项目的应用中,创建该服务实例 WebWCFobj:

```
Private WebWCFobj As WebWCF.ServiceIClient
```

通过对 WebWCFobj 的对象调用,可以像本地对象那样对服务端应用程序实现调用。

4.2.2 客户端的软件注册

客户端创建与服务端相同的对称加密对象 AOARijndaelAlg,将客户端需要提交的软件注册码、计算机硬件信息和身份识别等信息按预设方法组合成文件流 sfStream。将文件流 sfStream 加密,并调用远程的服务端应用程序对象 WebWCFobj,获得服务端传回的注册信息文件流 aoaCzcStream,主要代码:

```
Dim sfStreamEn As New CryptoStream(sfStream,  
AOARijndaelAlg.CreateEncryptor(AOARijndaelAlg.Key  
,AOARijndaelAlg.IV),CryptoStreamMode.Read)
```

```
aoaCzcStream = WebWCFobj.aoasz(sfStreamEn)
```

之后,将通过远程调用获得的激活信息文件流 aoaCzcStream 进行解密。客户端与服务端采用相同的对称加密、解密技术,并转换、解析成客户端所需的激活码等信息。

最后,客户端软件将获得的激活码进行比对,激活软件;在确认软件注册激活成功后,将当前软件的激活码采用 My.Computer.Registry.SetValue()方法保存到计算机注册表文件。

在正常应用中,客户端软件在启动过程中自动读取注册表的激活信息进行核对,实现软件身份的确认。

5 软件注册的实际应用与结果分析

上述激活机制在软件注册的实际应用中取得了满意的结果,主要体现在可靠性和安全性这两个方面,并且用户操作简单。

软件注册的信息是通过客户端自动调用 WCF 服务端程序,基于 SOAP 协议实现通信的。SOAP 通信协议保证了不同开发平台的应用程序对 WCF 服务端程序都能获得统一、规范的消息通信,注册过程通用、可靠。由于软件采用了对称加密文件流的方法,使注册信息获得了较好的保护。例如,客户端请求注册的一组信息:

身份标识、用户计算机硬件注册信息、软件商提供的注册码,由字符串(本例为“nyhwjjabc,F0DEF10 A4B4B4 45006991253938935735195158,w20202020202”)组成,其加密后的内容转换为: GALs KCJNuC3A/M/gZKt NPp7bRFxEkkmeDP00sUMmIM7j/ZfkQQBqKBwmpXjne9wBfoiY2HL7hkc727vPtjupKQ==。

上述信息由服务端处理后,得到回复的激活信息: Nu6cqxiBe0bzAlYw7a1ETm3FpLJH5WnvZE/tPWeEc/U=。

而此激活信息通过解密后,其原文内容是指定 16 位长的字符串“65BBFA9120908C2F”。

可见,通信过程中的信息被得到较好的保护。同时,上述的设计要素对用户而言是透明的,用户只要输入购买软件时获得的注册码、通过互联网即可即时获得软件激活。某一台计算机的激活码在其它计算机上无效。而对于软件供应商而言,提供给用户的注册码是有限次(如 2 次)使用的,注册码使用完后无法在其它计算机上继续注册,从而保护了软件版权。

6 结语

本项目的软件注册采用 WMI 硬件注册信息(采用 Hash 算法以指定格式转换成字符串),确认了用户计算机身份的唯一性;用户输入的注册码(16 位字符串)是由软件供应商提供的唯一号码,合法的注册码保存在提供 WCF 服务的数据库中;WCF 服务端程序按用户硬件信息和注册码在数据库进行遍历,在确认上述信息合法、且未超过使用次数后,按预置的 Hash 算法生成唯一的激活码,激活码保存在数据库并发回客户端程序;同时,整个通信信息是经对称加密的文件流,较好地确保了安全应用。本方案可设计成独立的注册服务程序,通用性强,安全可靠,具有良好的应用价值。

参考文献

- 1 梁爱虎.SOA 思想、技术与系统集成.北京:电子工业出版社,2007.
- 2 Klein S. Professional WCF Programming: NET Development with the Windows Communication Foundation. Wrox Press, 2007.
- 3 Microsoft. MSDN Library for Visual Studio 2010.
- 4 韩旭,王海波,柳克俊.基于.NET Framework WCF 的面向服务 SOA 中间件设计.小型微型计算机系统,2010,12: 2359-2364.