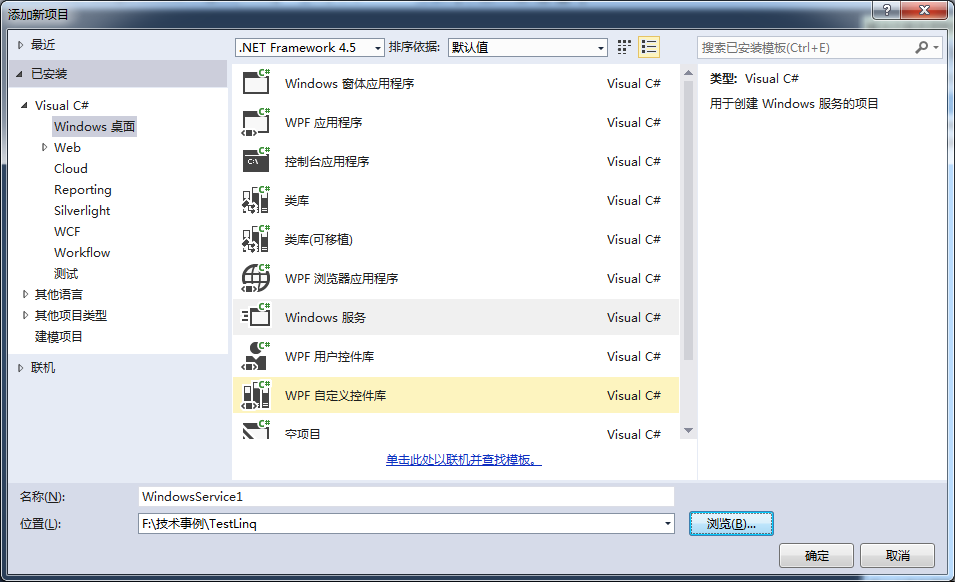
**[VS2013创建Windows服务 || VS2015+Windows服务简易教程](https://www.cnblogs.com/wyt007/p/8675957.html)**

转自：  
https://www.cnblogs.com/no27/p/4849123.html  
https://blog.csdn.net/ly416/article/details/78860522

**VS2013创建Windows服务**

一、创建服务

1、文件-》新建-》项目-》windows桌面-》windows服务，修改你要的项目名称。我这不改名，仍叫WindowsService1，确定。



2、其中的Program.cs文件是入口，Service1.cs是服务文件，所有的逻辑都在这。Service1.cs包含两部分，一部分是Designer，可以在这里面添加各种组件。一部分是后台文件，里面可以写一些逻辑，默认包含3个方法：构造函数、OnStart和OnStop，还可以添加OnPause和OnContinue方法。

3、修改Service1.cs文件如下（可以自行添加一些逻辑代码）

[复制代码](javascript:void(0);)

复制代码

namespace WindowsService1

{

public partial class Service1 : ServiceBase

{

public Service1()

{

InitializeComponent();

base.ServiceName = "KangryTest"; //设置服务名称，与后面的安装服务名称要一致

}

protected override void OnStart(string[] args)

{

//服务启动时相关代码。

}

protected override void OnStop()

{

//服务关闭时相关代码。

}

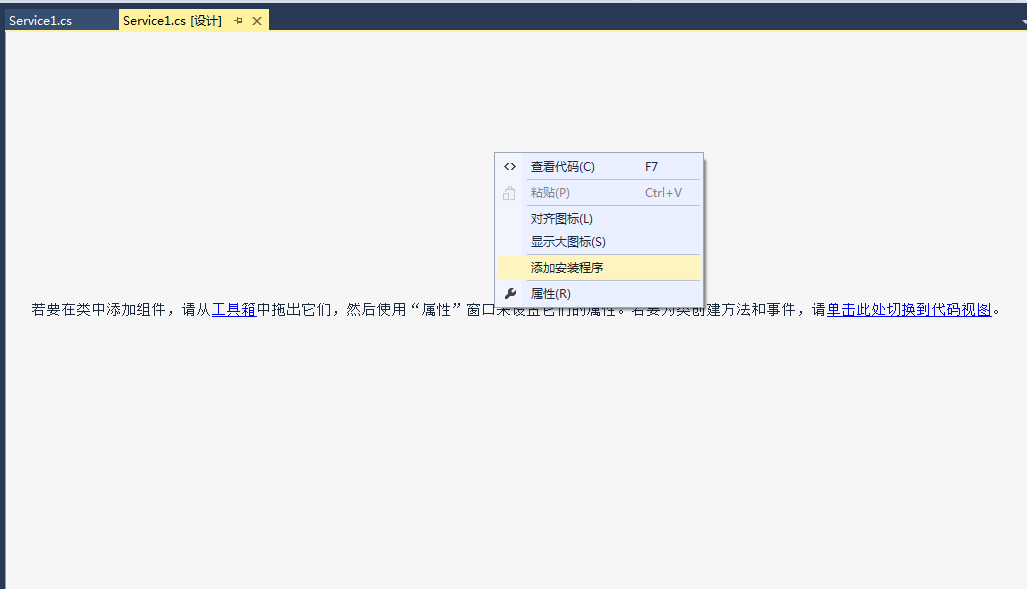
}

}

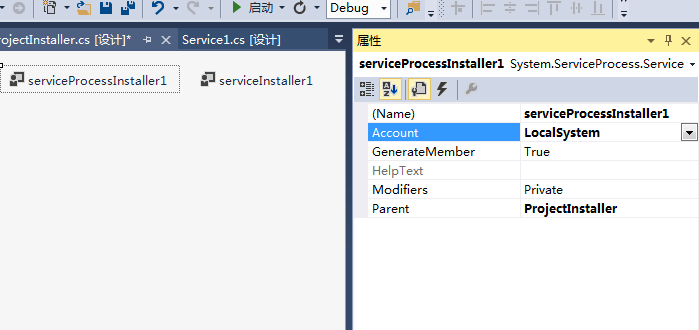
复制代码

[复制代码](javascript:void(0);)

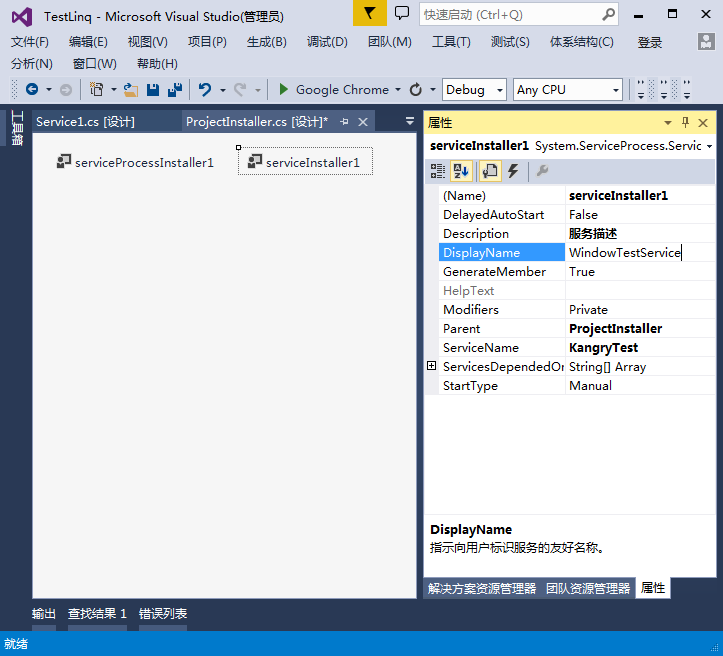
4、双击Service1.cs文件，进入设计页面，对着空白处右键-》添加安装器。



5、在安装器的设计界面，出现两个组件。点击serviceProcessInstaller1，在右下角的属性栏中，将Account修改为LocalSystem



6、选中ServiceInstaller1，右下角的属性框中，将ServiceName修改成第3步的ServiceName，其他自行选择。DelayedAutoStart表示开机后是否延迟启动。Description表示服务的描述，DisplayName表示服务显示名称。ServicesDependedOn表示依赖的服务项。StartType表示启动类型，分为自动启动，手动启动和禁用。



二、安装服务

选中项目右键选择“生成”，生成exe文件；

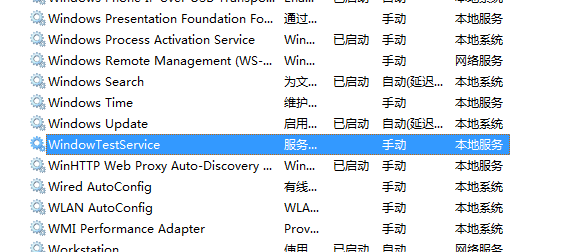
然后将从C:\Windows\Microsoft.NET\Framework\v4.0.30319中拷贝installutil.exe文件到生成目录（bin/Debug目的使installutil.exe和dp0WindowsService1.exe在同一级目录）下。在该目录新建“安装.bat”文件，使用记事本打开，输入如下命令：

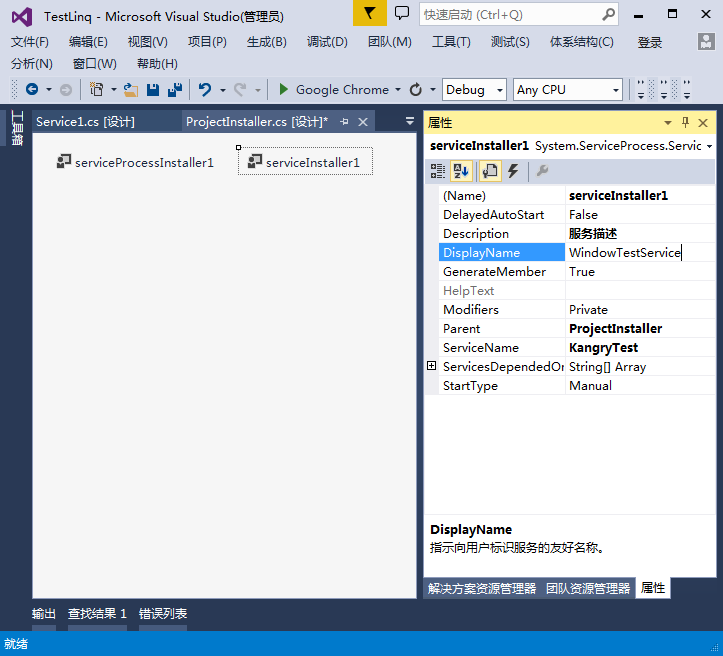
%~dp0InstallUtil.exe %~dp0WindowsService1.exe  
pause

注意前每个命令前要加一个%~dp0，表示将目录更改为当前目录。倘若不加，可能会出错。pause 一定要换行，否则报错。

最后双击安装.bat文件，就完成服务注册了。

在我的电脑上右键选择“管理”，打开“服务和应用程序”下的“服务”，就能看到我们注册的服务了。





三、卸载服务

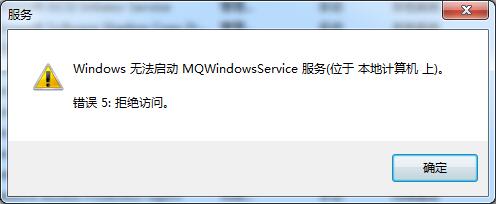
在该目录新建“卸载.bat”文件，使用记事本打开，输入如下命令：

%~dp0InstallUtil /u %~dp0WindowsService1.exe

pause

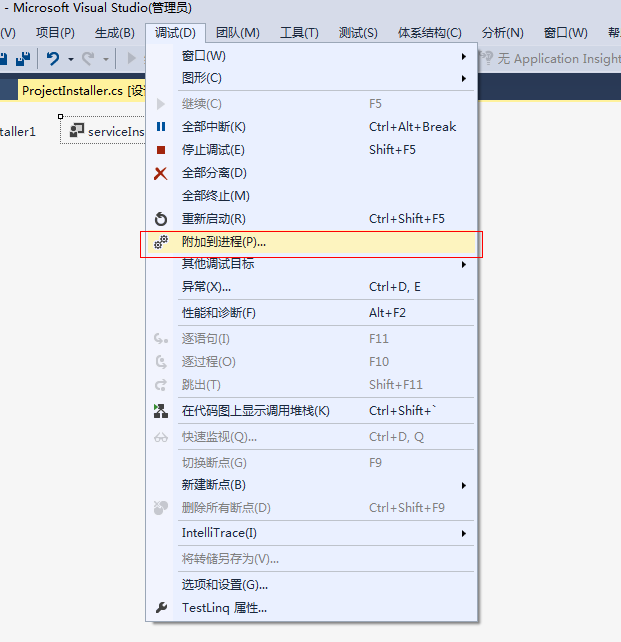
同样pause也要换行。

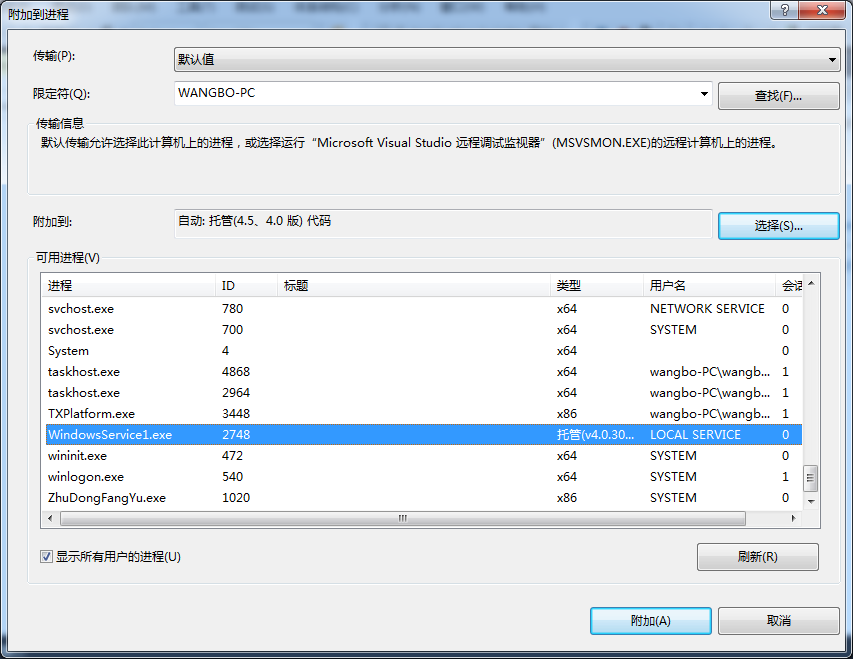
如果在启动过程中遇到如下问题，请将整个项目加上EVERYONE权限。



四、调试

在VS2013中选择“调试”-“附加到进程”，如下：

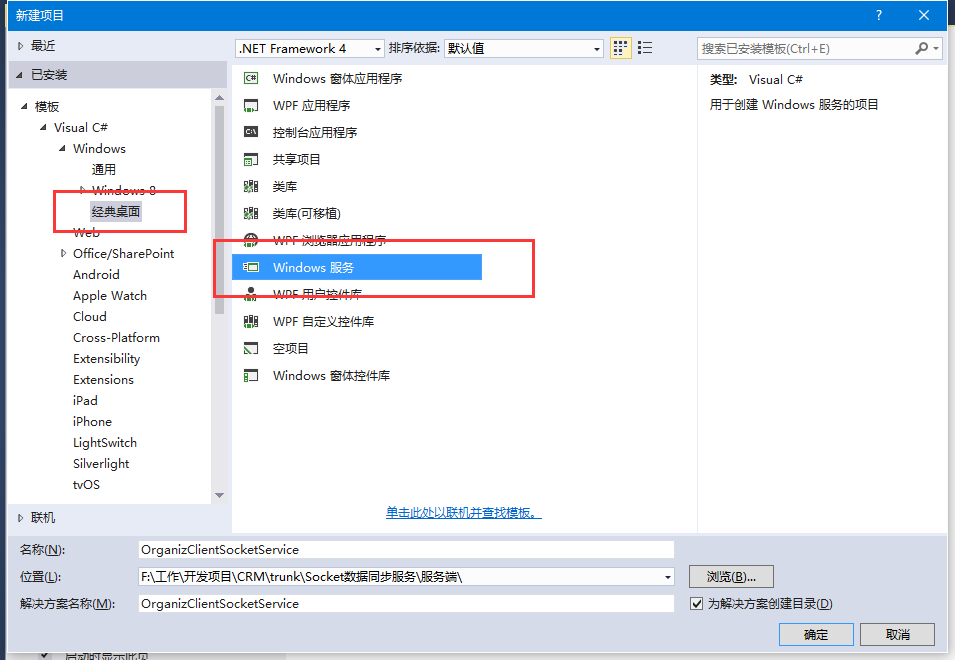




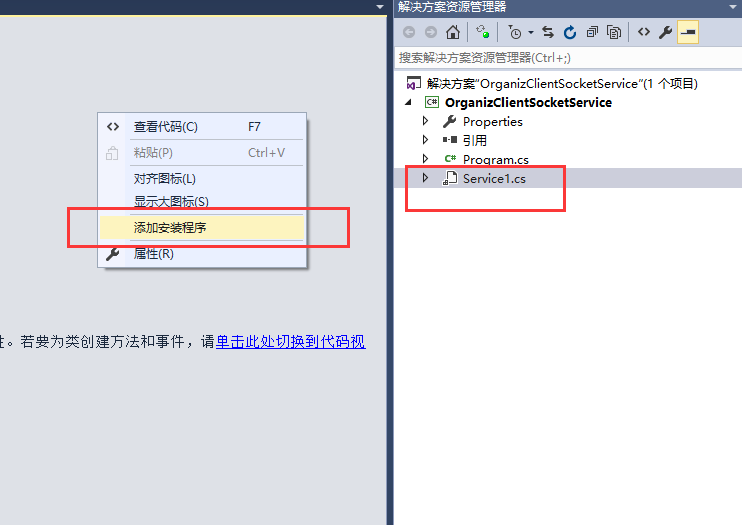
这样就可以调试了。

**VS2015+Windows服务简易教程**

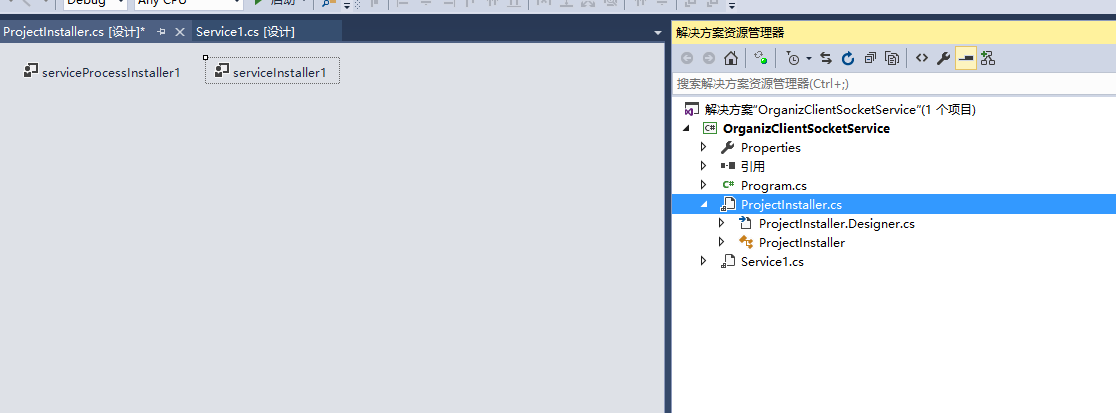
1、新建windows服务项目，我这里选择的是Framework4.0，没有选择高版本是为了防止在服务在一些低版本系统上无法正常运行。



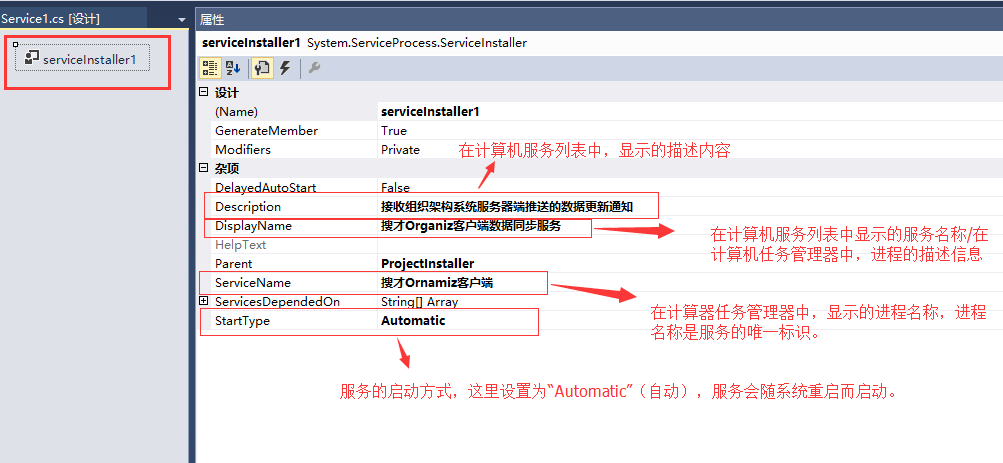
2、添加Windows服务的安装程序。

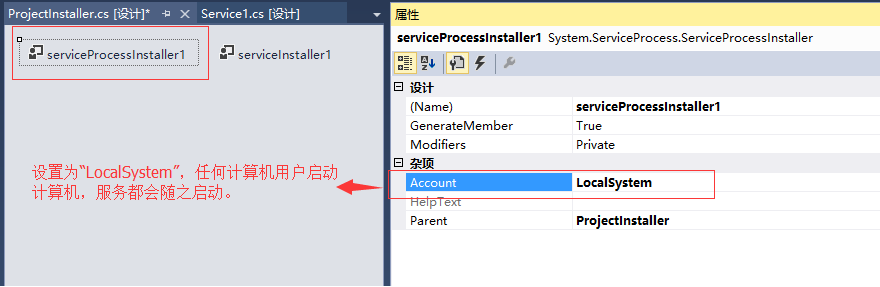


在默认Service1设计器界面空白处点击右键->添加安装程序，系统会自动新建一个带有默认配置的安装程序类，如下图：



新建完安装程序后，需要给默认的serviceInstaller1和serviceProcessInstaller1做一些基本的属性设置。如下图：

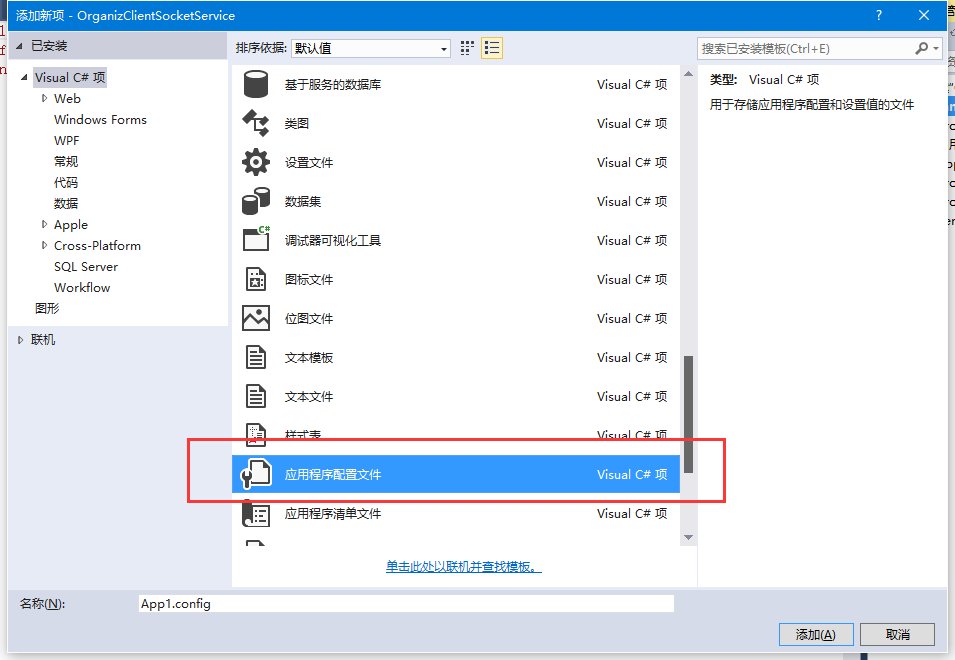




以上工作完成，安装程序配置完毕。

注意：如果不给服务添加安装程序，后面是没法把服务安装至windows系统里的。

 3、添加应用程序配置文件（如果有需要的话）。



如果项目有需要，一些应用程序的配置参数可以设置在此文件里（例如：数据库连接字符串）。

 4、编写windows服务主代码

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76 | using System;  using System.Collections.Generic;  using System.ComponentModel;  using System.Data;  using System.Diagnostics;  using System.Linq;  using System.ServiceProcess;  using System.Text;    using System.IO;    namespace OrganizClientSocketService  {      public partial class Service1 : ServiceBase      {          public Service1()          {              InitializeComponent();                System.Timers.Timer timer = new System.Timers.Timer();              timer.Elapsed += new System.Timers.ElapsedEventHandler(TimedEvent);              timer.Interval = 5000;//每5秒执行一次              timer.Enabled = true;          }            //定时执行事件          private void TimedEvent(object sender, System.Timers.ElapsedEventArgs e)          {              //业务逻辑代码          }            protected override void OnStart(string[] args)          {              this.WriteLog("搜才Organiz客户端数据同步服务：【服务启动】");          }            protected override void OnStop()          {              this.WriteLog("搜才Organiz客户端数据同步服务：【服务停止】");          }          protected override void OnShutdown()          {              this.WriteLog("搜才Organiz客户端数据同步服务：【计算机关闭】");          }            #region 记录日志          /// <summary>          /// 记录日志          /// </summary>          /// <param name="msg"></param>          private void WriteLog(string msg)          {                //string path = @"C:\log.txt";                //该日志文件会存在windows服务程序目录下              string path = AppDomain.CurrentDomain.BaseDirectory + "\\log.txt";              FileInfo file = new FileInfo(path);              if (!file.Exists)              {                  FileStream fs;                  fs = File.Create(path);                  fs.Close();              }                using (FileStream fs = new FileStream(path, FileMode.Append, FileAccess.Write))              {                  using (StreamWriter sw = new StreamWriter(fs))                  {                      sw.WriteLine(DateTime.Now.ToString() + "   " + msg);                  }              }          }          #endregion      }  } |

5、编译生成，安装windows服务至Windows系统。

　　　　完成开发后，对整各项目进行编译生成。在windows服务开发文件夹“\bin\Debug”下，就是我们需要安装的服务，建议把里面的所有文件拷贝至系统里的某个目录进行安装。

　　我是把整个个文件夹里的文件拷贝到c:\WindowService文件夹下。然后打开目录C:\Windows\Microsoft.NET\Framework64\v4.0.30319，拷贝里面的InstallUtil.exe文件至c:\WindowService文件夹下）。

　　注意：我的系统是windows10，64位系统，我的服务也将安装至64位系统下，所以我是进入C:\Windows\Microsoft.NET\Framework64\v4.0.30319目录拷贝InstallUtil.exe文件。各位安装的时候，根据你安装的目标系统，来觉得是拷贝哪个framework哪个版本，具体是64位的还是32位的也由你系统决定。

　　做好以上工作后就可以安装了，打开cdm就可执行安装了（一定要以管理员身份运行哟，要不然安装时会报“Windows服务安装异常：System.Security.SecurityException: 未找到源，但未能搜索某些或全部事件”）。

　　以下是安装命令、启动服务命令、停止服务命令、卸载服务命令：

　　　　安装命令：C:\WindowService\InstallUtil.exe C:\WindowService\OrganizClientSocketService.exe

　　　　启动服务命令：net start 搜才Organiz客户端数据同步服务

　　　　关闭服务命令：net stop 搜才Organiz客户端数据同步服务

　　　　卸载服务命令：C:\WindowService\InstallUtil.exe -u C:\WindowService\OrganizClientSocketService.exe