

```
cd \windows\Microsoft.Net\Framework\v4.0.30319
installutil -u /servicename="Sample Service 1" "c:\program files\ RIT\SampleService\SampleService.exe"
installutil -i u /servicename="Sample Service 1" "c:\program files\ RIT\SampleService\SampleService.exe"
installutil -u /servicename="Sample Service 2" "c:\program files\RIT\SampleService\SampleService.exe"
installutil -i u /servicename="Sample Service 2" "c:\program files\ RIT\SampleService\SampleService.exe"
```

Code Sample

```
namespace SampleService
{
    [RunInstaller(true)]
    public class ServiceInstaller : System.Configuration.Install.Installer
    {
        /// <summary>
        /// c'tor
        /// </summary>
        public ServiceInstaller()
        {
            InitializeComponent();
        }

        public override void Install(System.Collections.IDictionary stateSaver)
        {
            RetrieveServiceName();
            base.Install(stateSaver);
        }

        public override void Uninstall(System.Collections.IDictionary savedState)
        {
            RetrieveServiceName();
            base.Uninstall(savedState);
        }

        #region Component Designer generated code
        private void InitializeComponent()
        {
            _serviceProcessInstaller1 = new System.ServiceProcess.ServiceProcessInstaller();
            _serviceInstaller1 = new System.ServiceProcess.ServiceInstaller();

            _serviceProcessInstaller1.Password = null;
            _serviceProcessInstaller1.Username = null;
            _serviceProcessInstaller1.Account = System.ServiceProcess.ServiceAccount.LocalSystem;

            _serviceInstaller1.ServiceName = _product;

            Installers.AddRange(new System.Configuration.Install.Installer[]
            {
                _serviceProcessInstaller1,
                _serviceInstaller1});
        }
        #endregion

        private void RetrieveServiceName()
        {
            var serviceName = Context.Parameters["servicename"];

            Console.WriteLine(string.Format("RetrieveServiceName: {0}", servicename));

            if (!string.IsNullOrEmpty(serviceName))
            {
                this._serviceInstaller1.ServiceName = serviceName;
                this._serviceInstaller1.DisplayName = serviceName;
            }
            else
            {
                _serviceInstaller1.ServiceName = _product;
            }
        }

        private readonly string _product = "Sample Service";
        private System.ServiceProcess.ServiceProcessInstaller _serviceProcessInstaller1;
        private System.ServiceProcess.ServiceInstaller _serviceInstaller1;
    }
}
```

Achtung um Verwechslungen zu vermeiden: der hier eingetragene Wert ist nur innerhalb es geöffneten GUI Dialogs beim manuellen Start des Service gültig. Beim Verlassen des Dialogs ist der hier eingetragene Wert verloren, Windows speichert diesen Wert nicht. Windows bietet dazu per GUI noch immer keine Möglichkeit, seinen Start Parameter zu verwalten. Das wäre aber eine sinnvolle Verbesserung, die Startparameter hier an dieser Stelle im Dialog verwalten zu können.

Ein registriertes Windows Service landet mit einem Settings zwangsläufig in der Registry. Also wird man bei HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services auch fündig:

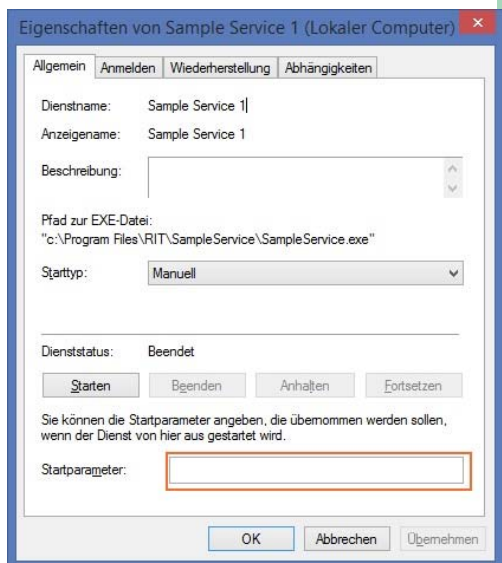


Abbildung 1: Service Startparameter



Abbildung 2: Registry Position der registrierten Services