2021/04/22_Windows32位内核_第2课_服务程序的编写InstDrv工具的实现

笔记本: Windows32位内核
创建时间: 2021/4/22 星期四 15:37
作者: ileemi

服务程序的编写

手动编译内核驱动需要编写相应的配置文件（SOURCES），高本版的WDK不在需要配置文件。
编译日志以及编译错误、警告都会生成对应的文件供查看。警告应该当作错误进行处理（内核中要求代码严谨，一旦出现问题就会影响到操作系统）。

> 低版本WDK，在手动编译内核驱动时，工程路径中不能有中文以及空格。

文件名使用.cpp时，入口函数名称在编译时会被粉碎，防止粉碎需要将入口函数使用 "**extern "C" {}**" 进行包裹。防止使用内核函数再次被粉碎，可添加一个头文件，将头文件以及函数声明使用 "**extern "C" {}**" 进行包裹。

驱动使用工具加载一般在测试的时候，发布的时候一般需要驱动自己安装加载并运行。

如何通过代码安装、加载驱动。对整个系统来说，可以理解为驱动就是操作系统的服务，驱动可以供 "任何人" 使用。

服务程序没有UI，Ring3环中也有服务，不需要界面。

驱动可以理解为内核级的服务。驱动的安装、运行和普通三环的普通服务程序本质上没有区别。程序操作系统把驱动当作服务进行处理。

ServiseMain:
服务程序不能双击运行，需要调用 StartService 来启动服务。启动服务前，需要创建服务（安装服务）。通过 "ControlService" 可以控件服务的停止，暂停等。通过 "DeleteService" 来卸载服务程序。
驱动启动工具的实现就是调用服务的API来实现的。

操作系统有服务对应的API，通过API去和对应的服务进行通讯。操作系统将驱动也定义为服务，但是驱动和三环的服务是有一定的差距的。驱动是内核级的服务。

安装高版本的WDK

VS 会多一个驱动模板选项。这样方便再VS中编写代码。

服务的安装会在注册表中记录服务的信息，卸载服务也就是删除注册表中相关的服务信息。

什么时候用服务程序?

1. 无UI操作且需要长期在后台运行的程序
2. 为系统所有程序提供服务

驱动也需要长期在后台运行，驱动的安装需要调用创建服务的API，启动驱动需要调用启动服务的API。

装载、卸载驱动三环程序可以进行控制。驱动安装成功后可以在注册表中查看到，在任务管理器中的服务窗口以及进程窗口都查看不到（驱动本身不是一个可执行文件，是操作系统的一个模块）。

在0环无法调用三环的API。 3环成的程序可以通过 "DeviceIoControl" 和0环的程序做交互（通过内核驱动定义好的通讯码）。内核驱动可以为任何一个三环程序进行服务，如果想为单一的程序进行服务需要自定义通讯码。

遍历内核驱动工具。