



网站 新帖 搜索 帮助

快捷导航

请输入搜索内容

帖子

领取今日签到奖励

网站 【 软件安全 】 『 编程语言区 』

返回列表

1

2

3

1 / 3 页

[内核编程] 自己写的Xuetr工具驱动读写过滤驱动，并奉献上源码 [复制链接]

机智聪明的小迪 2018-8-15 08:18



先放代码：

[C] 纯文本查看 复制代码

```
001 | #include <ntddk.h>
002 |
003 | /*
004 |     First Driver
005 | */
006 |
007 | /*
008 |
009 |
010 | //未文档化的函数->通过名字获取设备对象
011 |
012 | NTKERNELAPI
013 | NTSTATUS
014 | ObReferenceObjectByName(
015 |     IN PUNICODE_STRING ObjectName,
016 |     IN ULONG Attributes,
017 |     IN PACCESS_STATE PassedAccessState OPTIONAL,
018 |     IN ACCESS_MASK DesiredAccess OPTIONAL,
019 |     IN POBJECT_TYPE ObjectType OPTIONAL,
020 |     IN KPROCESSOR_MODE AccessMode,
021 |     IN OUT PVOID ParseContext OPTIONAL,
022 |     OUT PVOID *Object
023 | );
024 |
025 | extern POBJECT_TYPE *IoDriverObjectType;
026 |
027 | PDRIVER_OBJECT g_FilterDriverObject;
028 |
029 | //保存以前的驱动请求例程
030 | PDRIVER_DISPATCH g_OrigReadCompleteRoutine;
031 |
032 | //驱动过滤例程
033 | NTSTATUS FilterReadCompleteRoutine(__in struct _DEVICE_OBJECT *DeviceObject, __inout
034 | {
035 |     KdPrint(("拦截到数据读取！"));
036 |
037 |     return g_OrigReadCompleteRoutine(DeviceObject, Irp);
038 | }
039 |
040 |
```

```

041 //卸载HOOK函数
042 NTSTATUS UnfilterDriverRoutine()
043 {
044     //检测地址是否有效（可读可写）
045
046     if (MmIsAddressValid(g_FilterDriverObject))
047     {
048         //写回原例程
049         g_FilterDriverObject->MajorFunction[IRP_MJ_DEVICE_CONTROL] = g_OrigReadCompleteRoutine;
050     }
051
052     return STATUS_SUCCESS;
053 }
054
055 //驱动过滤函数->配合 ObReferenceObjectByName
056 NTSTATUS FilterDriverQuery()
057 {
058     NTSTATUS Status ;
059     UNICODE_STRING usObjectName;
060
061     RtlInitUnicodeString(&usObjectName, L"\\Driver\\Xuetr");
062
063     //取出驱动对象
064     Status = ObReferenceObjectByName(
065         &usObjectName,
066         OBJ_CASE_INSENSITIVE,
067         NULL,
068         0,
069         *IoDriverObjectType,
070         KernelMode,
071         NULL,
072         (PVOID*)&g_FilterDriverObject
073     );
074
075     if (!NT_SUCCESS(Status)) {
076         //返回失败值
077
078         KdPrint(("取出驱动对象失败!!!"));
079
080         return Status;
081     }
082
083     //打印驱动指针对象地址
084     KdPrint(("驱动指针地址: 0x%x", g_FilterDriverObject));
085
086     //保存原有驱动实例
087     g_OrigReadCompleteRoutine = g_FilterDriverObject->MajorFunction[IRP_MJ_DEVICE_CONTROL];
088
089     //启动过滤
090     g_FilterDriverObject->MajorFunction[IRP_MJ_DEVICE_CONTROL] = FilterReadCompleteRoutine;
091
092     return STATUS_SUCCESS;
093 }
094
095 //卸载函数
096 NTSTATUS Unload(IN PDRIVER_OBJECT pDriverObject)
097 {
098     //清理符号连接和设备对象，避免内存泄漏
099
100
101
102
103
104
105
106
107
108
109
110

```

```
111
112     UNICODE_STRING usSysname;
113
114     RtlInitUnicodeString(&usSysname, L"\\?\\CEpassTP");
115
116     if (pDriverObject->DeviceObject != NULL)
117     {
118         IoDeleteSymbolicLink(&usSysname);
119
120         IoDeleteDevice(pDriverObject->DeviceObject);
121
122         KdPrint(("删除设备对象和符号连接完成!"));
123     }
124     UnfilterDriverRoutine();
125
126     return STATUS_SUCCESS;
127 }
128
129 //创建设备对象
130 NTSTATUS CreatDevice(PDRIVER_OBJECT pDriverObject)
131 {
132     //记录返回值
133     NTSTATUS Status;
134
135     //定义设备对象
136     PDEVICE_OBJECT pDevobj;
137
138     //设备对象名字字符串
139     UNICODE_STRING usDevName;
140
141     //设备符号连接字符串
142     UNICODE_STRING usSysName;
143
144     //初始化设备名
145     RtlInitUnicodeString(&usDevName, L"\\Device\\CEpassTP");
146
147     //创建设备->赋值创建状态
148     Status = IoCreateDevice(pDriverObject, \
149         0, \
150         &usDevName, \
151         FILE_DEVICE_UNKNOWN, \
152         FILE_DEVICE_SECURE_OPEN, \
153         TRUE, \
154         &pDevobj
155     );
156
157     //通过宏->判断创建状态
158     if (!(NT_SUCCESS(Status)))
159     {
160         return Status;
161     }
162
163     //或运算赋值
164     pDevobj->Flags |= DO_BUFFERED_IO;
165
166     //初始化符号连接名
167     RtlInitUnicodeString(&usSysName, L"\\?\\CEpassTP");
168
169     //创建符号连接
170     Status = IoCreateSymbolicLink(&usSysName, &usDevName);
171
172     if (!NT_SUCCESS(Status))
173     {
174         //不成功删除设备对象-返回错误值
175         IoDeleteDevice(pDevobj);
176
177         return Status;
178     }
179 }
```

```
180     }
181
182
183     return STATUS_SUCCESS;
184 }
185
186 //PIRP 传送数据包
187
188 NTSTATUS CreateCompleteRoutine(PDEVICE_OBJECT pDeviceobj, PIRP pIrp)
189 {
190     //注册标准参数实例例程，避免蓝屏
191     NTSTATUS Status;
192
193     Status = STATUS_SUCCESS;
194
195     KdPrint(("驱动例程创建完成!"));
196
197     pIrp->IoStatus.Status = Status;//设置传送状态
198     pIrp->IoStatus.Information = 0;//操作字节数返回
199
200     IoCompleteRequest(pIrp, IO_NO_INCREMENT);//终结IRP传递
201
202     return Status;
203 }
204
205 NTSTATUS CloseCompleteRoutine(PDEVICE_OBJECT pDeviceobj, PIRP pIrp)
206 {
207     //注册标准参数实例例程，避免蓝屏
208     NTSTATUS Status;
209
210     Status = STATUS_SUCCESS;
211
212     KdPrint(("驱动例程关闭成功!"));
213
214     pIrp->IoStatus.Status = Status;//设置传送状态
215     pIrp->IoStatus.Information = 0;//操作字节数返回
216
217     IoCompleteRequest(pIrp, IO_NO_INCREMENT);//终结IRP传递
218
219     return Status;
220 }
221
222 NTSTATUS ReadCompleteRoutine(PDEVICE_OBJECT pDeviceobj, PIRP pIrp)
223 {
224     //注册标准参数实例例程，避免蓝屏
225     NTSTATUS Status;
226
227     Status = STATUS_SUCCESS;
228
229     KdPrint(("例程内存读取!"));
230
231     pIrp->IoStatus.Status = Status;//设置传送状态
232     pIrp->IoStatus.Information = 0;//操作字节数返回
233
234     IoCompleteRequest(pIrp, IO_NO_INCREMENT);//终结IRP传递
235
236     return Status;
237 }
238
239 NTSTATUS WriteCompleteRoutine(PDEVICE_OBJECT pDeviceobj, PIRP pIrp)
240 {
241     //注册标准参数实例例程，避免蓝屏
242     NTSTATUS Status;
243
244     Status = STATUS_SUCCESS;
245
246     KdPrint(("例程写内存!"));
```

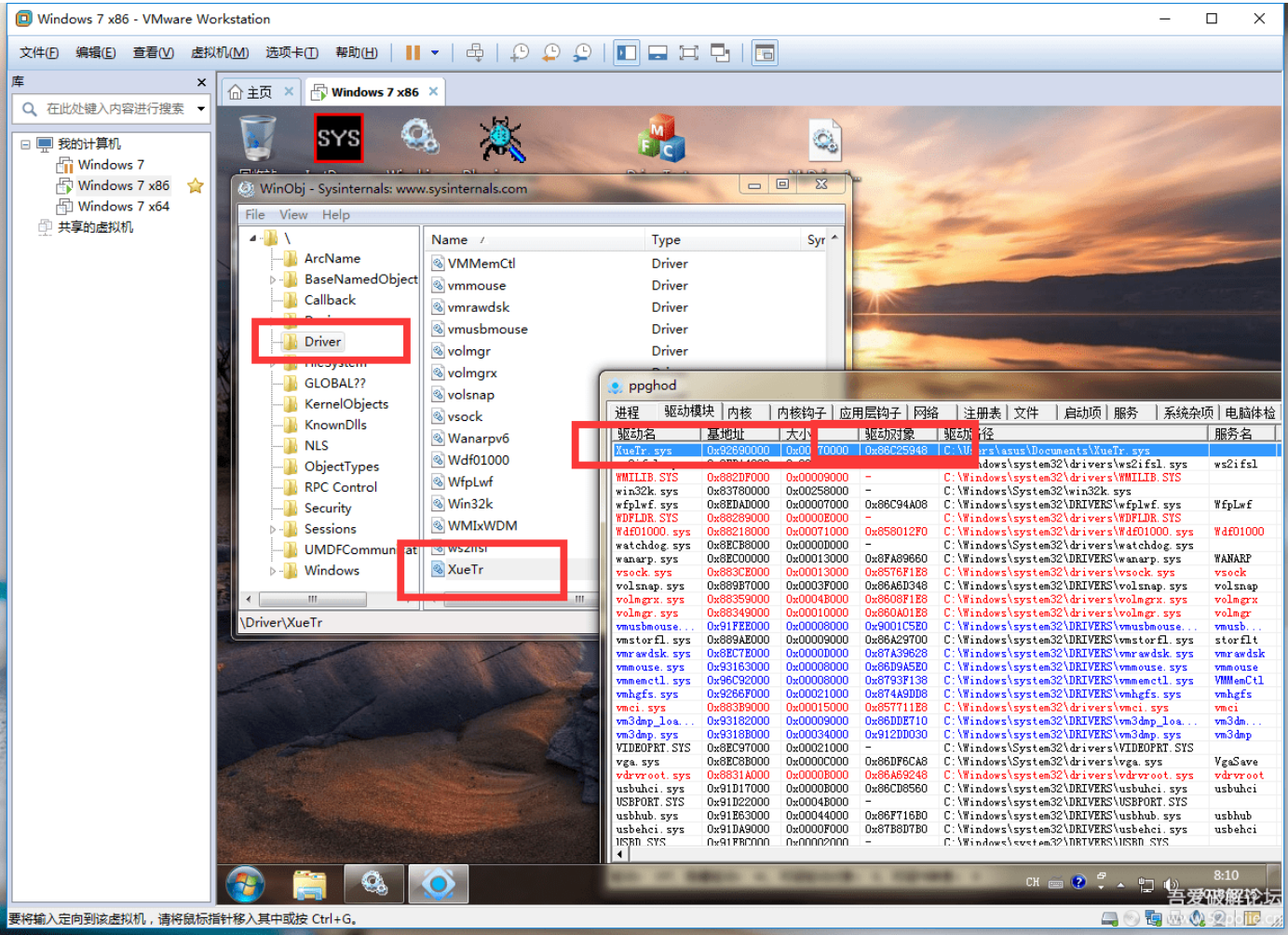
```

250
251     pIrp->IoStatus.Status = Status;//设置传送状态
252     pIrp->IoStatus.Information = 0;//操作字节数返回
253
254     IoCompleteRequest(pIrp, IO_NO_INCREMENT);//终结IRP传递
255
256     return Status;
257 }
258
259
260 NTSTATUS DriverEntry(IN PDRIVER_OBJECT pDriverObject, IN PUNICODE_STRING RegistryPath)
261 {
262     //KdBreakPoint();//断点
263
264     NTSTATUS Status;
265
266     Status = CreateDevice(pDriverObject);
267
268     if (!NT_SUCCESS(Status))
269     {
270         KdPrint(("创建设备对象失败!"));
271     }
272     else
273     {
274         KdPrint(("创建设备对象成功! \n"));
275         KdPrint(("注册表路径: %wZ ", RegistryPath));
276     }
277
278     //注册请求例程->发送到IO管理器进行处理
279
280     pDriverObject->MajorFunction[IRP_MJ_CREATE] = CreateCompleteRoutine;
281     pDriverObject->MajorFunction[IRP_MJ_CLOSE] = CloseCompleteRoutine;
282
283     pDriverObject->MajorFunction[IRP_MJ_READ] = ReadCompleteRoutine;
284     pDriverObject->MajorFunction[IRP_MJ_WRITE] = WriteCompleteRoutine;
285
286
287
288
289     pDriverObject->DriverUnload = Unload;
290
291
292     //开启驱动过滤
293
294     FilterDriverQuery();
295
296     return STATUS_SUCCESS;
297 }

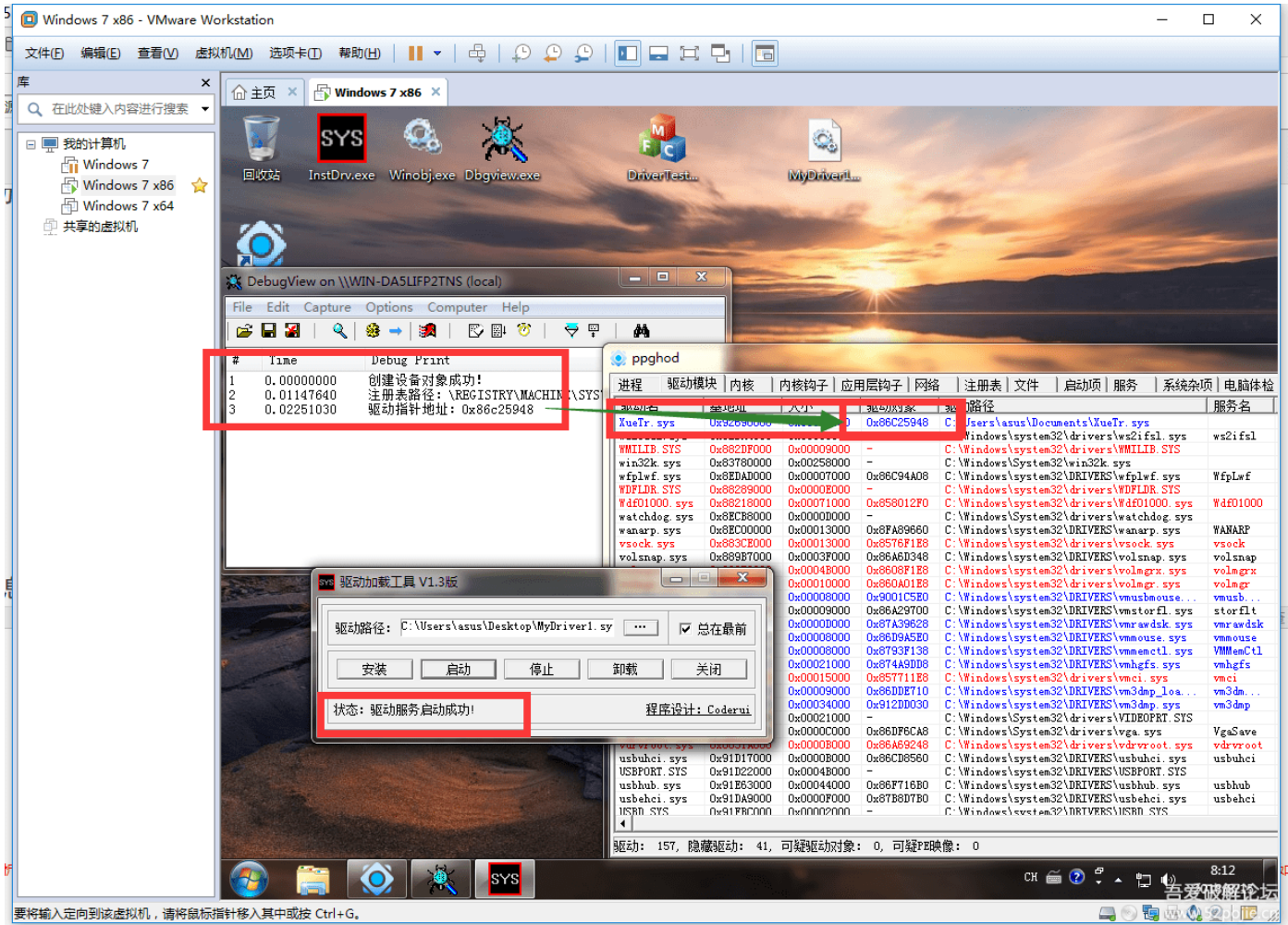
```

功能是过滤到Xuetr驱动的数据读取功能：

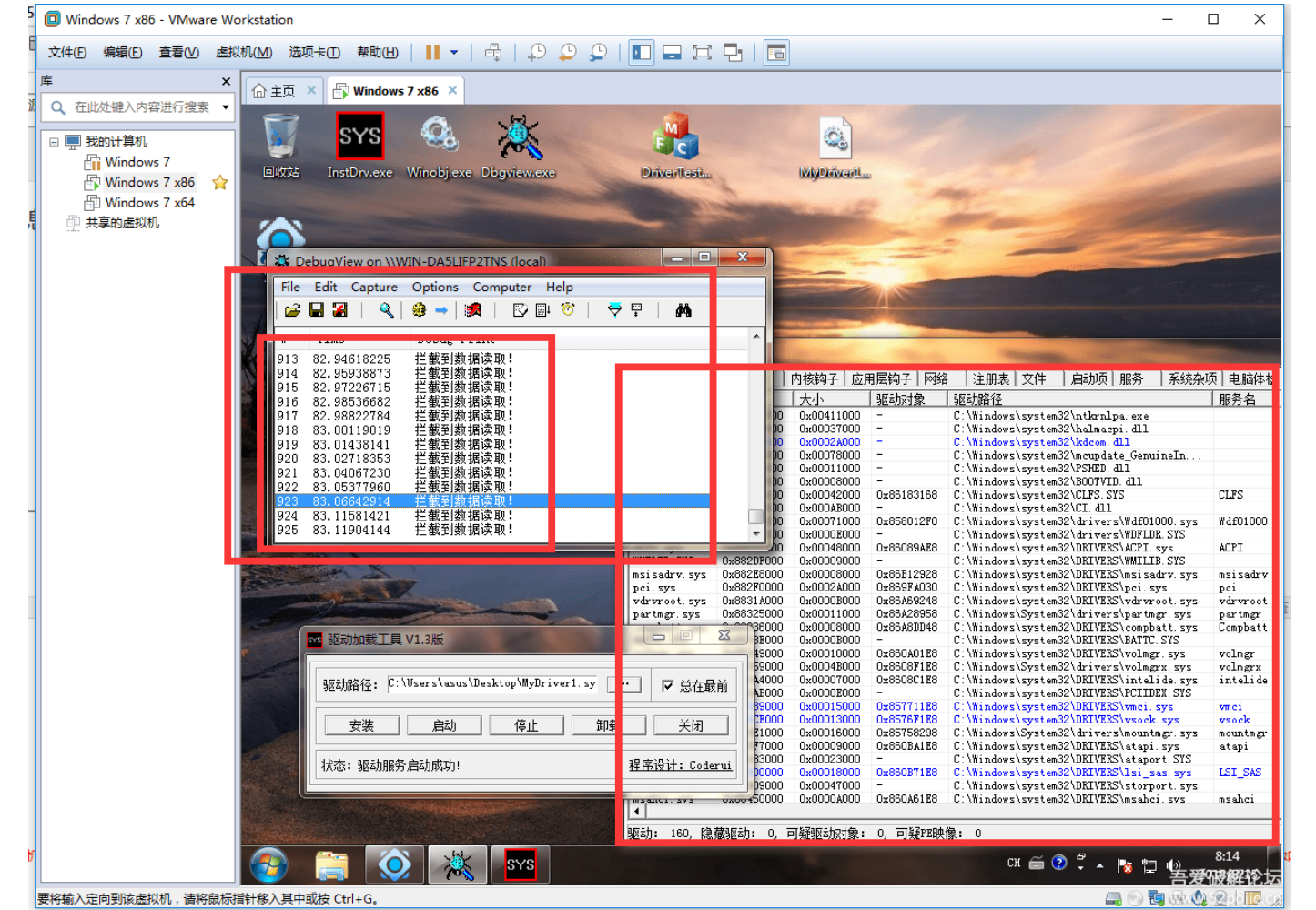
1. 首先我们看一下Xuetr的驱动



2. 然后我们加载自己的驱动，过滤消息



可以获取到指针了，然后我们刷新一下Xuetr工具驱动模块



成功了，看到没，检测到刷新工具的数据读取了！

我的开发平台是：VS2017 + WDK10 win7x86 x86驱动，如果代码不做任何修改，然后编译x64驱动，在64位系统直接执行也是没问题的

源码我做了很详细的注释，如果还有不懂的可以私信我哦，一起学习交流进步！祝愿我们早日过掉某P！过掉某E！

免费评分


	威望	吾爱币	热心值	理由	收起
纳新德白蛇	+ 1	+ 1		谢谢@Thanks!	
vihp	+ 1	+ 1		热心回复!	


 ban_op	+ 1	+ 1	热心回复！	
 Monitor	+ 1	+ 1	支持开源，鼓励开源	
 快乐王子	+ 1	+ 1	热心回复！	
 苏紫方璇	+ 1	+ 5	+ 1	感谢发布原创作品，吾爱破解论坛因你更精彩！
 hunteraa	+ 1	+ 1	用心讨论，共获提升！	
 www.52pojie.cn	+ 3	+ 1	谢谢@Thanks！	


[查看全部评分](#)

本帖被以下淘专辑推荐:

· [学习及教程](#) | 主题: 1305, 订阅: 985

 收藏 16

 免费评分

 淘帖 1

发帖前要善用【论坛搜索】功能，那里可能会有你要找的答案或者已经有人发布过相同内容了，请勿重复发帖。

回复

举报

 hui99995 2018-8-20 09:16

推荐

寒雪冰熊 发表于 2018-8-15 08:54
XP对于懂内核的人来说就是小菜一碟

您好 有一个过tp的项目有意参加嘛?

【吾爱破解论坛总版规】 - [让你充分了解吾爱破解论坛行为规则]

回复 支持

免费评分 举报

 寒雪冰熊 2018-8-20 10:23

推荐

吾爱破解论坛没有任何官方QQ群，禁止留联系方式，禁止任何商业交易。

hui99995 发表于 2018-8-20 09:16
您好 有一个过tp的项目有意参加嘛?

TP在WIN7X64以后必须得用VT突破 了

如何升级？如何获得积分？积分对应解释说明！

回复 支持

免费评分 举报



eoven8 2018-8-15 08:25

4#

《站点帮助文档》有什么问题来这里看看吧，这里有你想知道的内容！
看起来很不错呢

呼吁大家发布原创作品添加吾爱破解论坛标示！

回复 支持

免费评分 举报



xuehaiyouya 2018-8-15 08:27

5#

路过看看学习学习

如何快速判断一个文件是否为病毒！

回复 支持

免费评分 举报



jht168888 2018-8-15 08:33

6#

这个真难

回复 支持

免费评分 举报



a449096866 2018-8-15 08:40

7#

路过进来看看

回复 支持

免费评分 举报



不苦小和尚 2018-8-15 08:51

8#

看不懂先mark一下

回复 支持

免费评分 举报



寒雪冰熊 2018-8-15 08:54

9#

eoven8 发表于 2018-8-15 08:25
看起来很不错呢

XP对于懂内核的人来说就是小菜一碟

回复 支持

免费评分 举报



LZ520 2018-8-15 09:16

10#

过来学习学习

回复支持

免费评分 举报

 **lsrteam70** 2018-8-15 09:31 11#

过来学习学习

回复支持

免费评分 举报

 **小小学生** 2018-8-15 09:38 12#

好棒 谢谢 辛苦了 楼主啊

回复支持

免费评分 举报

下一 页 »

	高级模式

验证问答

换一个

发表回复

警告：本版块禁止灌水或回复与主题无关内容，违者重罚！

☐ 回帖并转播

☐ 回帖后跳转到最后一页

免责声明：
吾爱破解所发布的一切破解补丁、注册机和注册信息及软件的解密分析文章仅限用于学习和研究目的；不得将上述内容用于商业或者非法用途，否则，一切后果请用户自负。本站信息来自网络，版权争议与本站无关。您必须在下载后的24个小时之内，从您的电脑中彻底删除上述内容。如果您喜欢该程序，请支持正版软件，购买注册，得到更好的正版服务。如有侵权请邮件与我们联系处理。
Mail To:Service@52PoJie.Cn