学OD -- 消息断点 RUN跟踪

转载

ccx_john 于 2014-02-28 13:06:04 发布 1534 收藏
分类专栏： 加密解密逆向

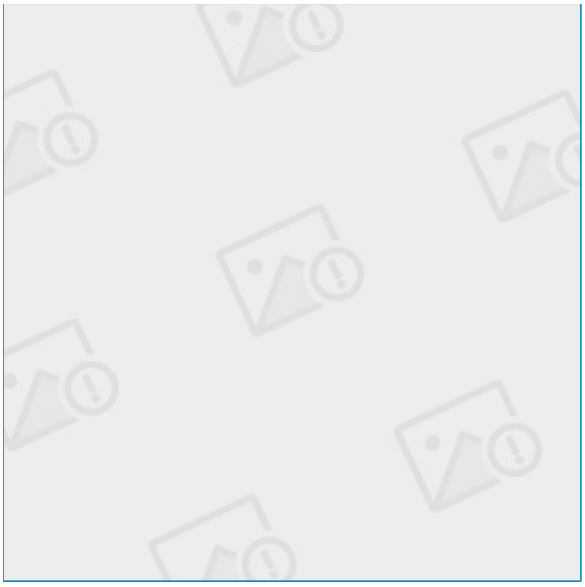

加密解密逆向 专栏收录该内容
4 篇文章 0 订阅
订阅专栏

学OD -- 消息断点 RUN跟踪

这一篇讲的是 消息断点和RUN跟踪的简单知识

这一篇没怎么看明白 大概使用知道了 怎么用不太清楚。

介绍本次软件特点输入后木有反应(纱布垃圾的。。哈哈)



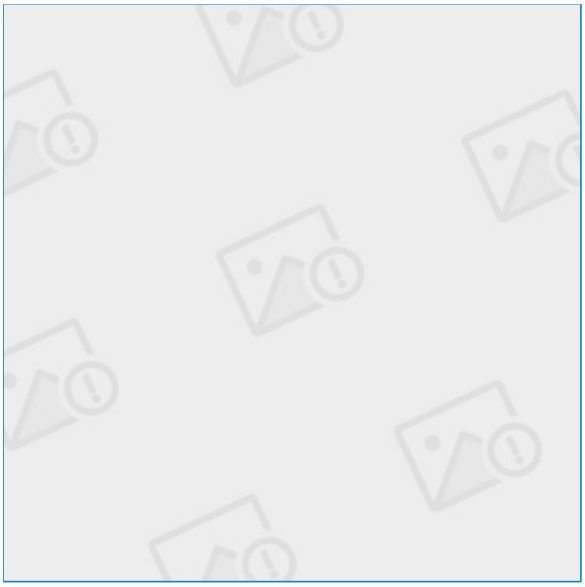
启动OllyDBG载入这个程序，F9让它运行。这个程序按我们前面讲的采用字串参考或函数参考的方法都很容易断下来。但我们今天主要学习的是消息断点及RUN跟踪，就先用消息断点来断这个程序吧。

首先我们要了解的是消息。(简单介绍机制)

Windows的中文翻译就是“窗口”，而Windows上面的应用程序也都是通过窗口来与用户交互的。现在就有一个问题，应用程序是如何知道用户作了什么样的操作的？这里就要用到消息了。Windows是个基于消息的系统，它在应用程序开始执行后，为该程序创建一个“消息队列”，用来存放该程序可能创建的各种不同窗口的信息。比如你创建窗口、点击按钮、移动鼠标等等，都是通过消息来完成的。通俗的说，Windows就像一个中间人，你要干什么事是先通知它，然后它才通过传递消息的方式通知应用程序作出相应的操作。说到这，又有个问题了，在Windows下有多个程序都在运行，那我点了某个按钮，或把某个窗口最大化，Windows知道我是点的哪个吗？这里就要说到另一个内容：句柄（handle）了。句柄一般是个32位的数，表示一个对象。Windows通过使用句柄来标识它代表的对象。比如你点击某个按钮，Windows就是通过句柄来判断你是点击了那一个按钮，然后发送相应的消息通知程序。

先载入程序填写用户名与注册码不要点check 回到OD查看消息

(点击菜单查看->窗口（或者点击工具栏上那个“W”的图标或者Alt + M组合键)。出现以下界面



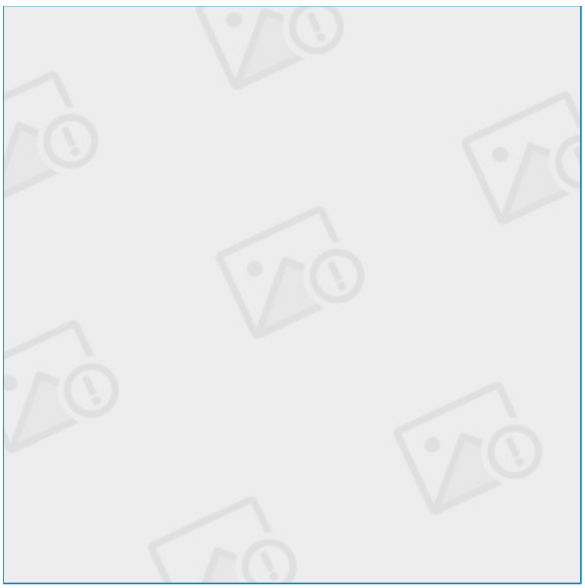
如果没有出现消息可以在框内右击刷新一下.

现在下消息断点因为我们要点击Check按钮在消息窗口(上面的截图)中我们可以看到标题这一列好多是软件的控件上的文字。。

这里我们选择Check，**右击选择在ClassProc上设置消息断点**ClassProc这一列显示的是地址而且特别大估计是系统领域的地址。

先不管它 继续说消息断点

设置后弹出一个框如下



消息编辑框点击下拉菜单选择要设置断点的消息，这里有很多很多。。

因为Check是个按钮所以我们可找相关的消息即当按下按钮再抬起时，即

WM_LBUTTONUP这个消息点击确定即可之后可以看到所有消息的这个窗口上有三个地方显示了粉色可见只要是软件上的按钮都设置了这个消息断点。到此断点完成。

现在我们还要做一件事，**就是把RUN跟踪打开**。

这个RUN跟踪是干什么的？

简单的说，RUN跟踪就是把被调试程序执行过的指令保存下来，让你可以查看被调试程序运行期间干了哪些事。RUN跟踪会把地址、寄存器的内容、消息以及已知的操作数记录到RUN跟踪缓冲区中，你可以通过查看RUN跟踪的记录来了解程序执行了那些指令。在这还要注意一个缓冲区大小的问题，如果执行的指令太多，缓冲区满了的话，就会自动丢弃前面老的记录。

打开RUN跟踪：点击菜单上的调试按钮 选择打开/关闭RUN跟踪

点击后可以看见它下边的关闭RUN跟踪选项由禁用变可用此现象判断RUN跟踪是打开还是关闭。

设置RUN跟踪：保证当前在我们调试的程序领空，在反汇编窗口中点击右键，在弹出菜单中选择 RUN 跟踪->添加所有函数过程的入口

我们可以看到OllyDBG把识别出的函数过程都在前面加了灰色条，当这些指令前的灰色条被执行过后灰色就变为红色了。

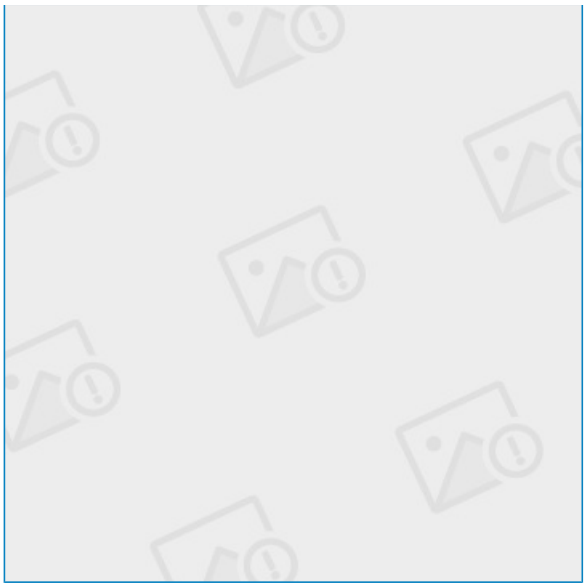
好了 点击Check运行程序。OD断下来如图:粉红色是条件断点

可以看到这里的地址很大是在系统领空。

我们以前都是断下后就在程序自己的领空我们再分析程序，就算不是也要想别的办法让它停在离断点最近的程序部分。。

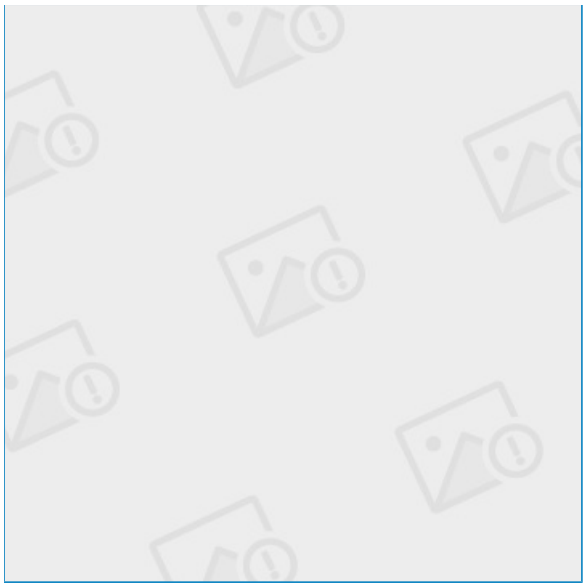
(按Alt+F9好像就可以，这里这么做的。。)

这时我们打开内存窗口M按钮找到CODE即代码段在CODE那一行F2下访问断点。

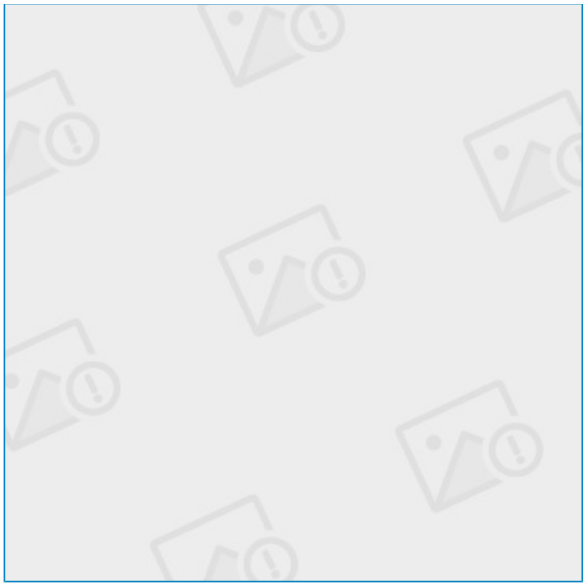


我们这里的意思就是在消息断点断下后，只要按F9键运行时执行到程序代码段的指令我们就中断，这样就可以回到程序领空了（当然在401000处所在的段不是绝对的，我们主要是要看程序的代码段在什么位置，其实在上面图中 OllyDBG 内存窗口的“包含”栏中我们就可以看得很清楚了）。设好访问断点后我们按F9键断下:

按F9键（或者按CTR+F12组合键跟踪步过 CTR+F11 跟踪步入）让程序运行，再点击菜单查看->RUN跟踪，或者点击工具栏上的那个“…”符号，打开RUN跟踪的记录窗口看看：

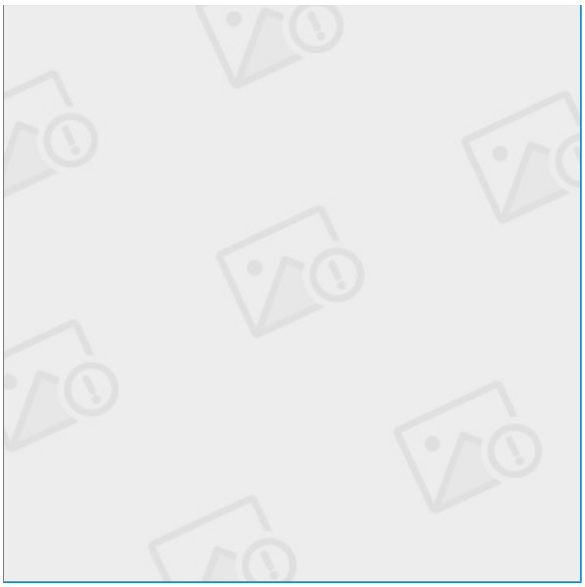


我们现在再来看看统计的情况：



我们只关心那些执行过一次的指令。**随便选择一条双击可以跳到相应的反汇编窗口中。**

在这我们选 00401082地址处的这一条。来到这:



在这里我们看到了我们熟悉的函数GetDlgItemTextA函数看到了我们输入的用户名的转换“2222222….”我输入的是2。

这里的指令都是执行过的。这也叫逆向分析嘛RUN跟踪的作用。。

我们反方向找到了提取用户名的地方了 所以可以下断了。。就在地址4010A6处的那条指令上按F2 即可。

现在删除所有其它的断点，点菜单调试->关闭RUN跟踪，（最终就为找这个断点。。。嘎嘎）

现在我们就可以开始分析了：作者说了

对于新手来说，可能这个crackme的难度大了一点。没关系，我们主要是学习OllyDBG的使用，方法掌握就可以了(以后破解的时候再回头看)

最后说明一下：

1、这个程序在设置了消息断点后可以省略在代码段上设访问断点那一步，直接打开RUN跟踪，消息断点断下后按CTR+F12组合键让程序执行，RUN跟踪记录中就可以找到关键地方。

2、对于这个程序，你可以不设消息断点，在输入用户名和注册码后先不按那个“Check”按钮，直接打开RUN跟踪，添加“所有函数过程的入口”后再回到程序中点“Check”按钮，这时在OllyDBG中打开RUN跟踪记录同样可以找到关键位置。

RUN跟踪还木有整明白感觉很强大消息断点需要脱壳经验啊。。继续。。。