



北京大学网络攻防技术与实践课程

Rada分析作业讲解

北京大学计算机研究所信安中心



讲解内容

- 准备工作
- 行为分析
- 静态分析
- 问题解答



准备工作

□ 确认文件完整性

```
[root@icstMySQL scan32]# md5sum RaDa.zip  
a75de27ee59ab60e148efe7feee5dd3f RaDa.zip
```

□ 解压缩: **unzip**

□ 查看文件属性

- 右键点击文件->选择“属性”

□ 确认二进制文件格式

```
[root@icstMySQL scan32]# file RaDa.exe  
RaDa.exe: MS-DOS executable (EXE), OS/2 or MS Windows
```

准备工作

□ 取得文件中的可打印字符串

- 乱码
- 加壳了?

```
[root@icstMySQL scan32]# strings RaDa.exe
6B@>CEC
YMOM@./
RmR] .G^
^@n/
h^ry
N[5M
;yaO
W81b' #
ORaDa
=LVB5!
*#~t
;x@S
71;uM'
$;4=}
_ 'H
2Hw4'
331S
'4L;5\S
```



动态分析-行为监控

- ☐ 开启**Wireshark**
- ☐ 开启**Filemon**
- ☐ 开启**Regmon**
- ☐ 使用**Regshot**获取注册表快照
- ☐ 执行**RaDa.exe**
- ☐ 使用**RegShot**获取执行后的注册表快照，并生成报告
- ☐ 保存**Wireshark, Filemon, Regmon**的**Log**



观察行为

- 注册表项
 - **HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run**
 - **RaDa: C:\RaDa\bin\RaDa.exe.**
- 文件系统
 - **C:\RaDa, C:\RaDa\tmp ,C:\RaDa\bin.**
 - 将文件**RaDa.exe**复制到了**C:\RaDa\bin**目录下
- 网络行为
 - 通过**HTTP**协议请求**10.10.10.10\RaDa\RaDa_commands.html**
- 行为解读
 - 将自身复制至感染主机系统盘，并激活自启动
 - 尝试获取一个**HTML**页面“**commands**” → 猜测是否后门接收控制指令？
- 动态分析不够充分，是否有隐藏路径未触发条件（命令行、时间逻辑等因素）



静态分析-加壳识别

□ 探测是否加壳

■ 使用**FileAnalyze**工具的探测结果如下

```
C:\WINDOWS\system32\cmd.exe
Versions      : LINK - 6.0, OS - 4.0, User - 1.0, Sybssystem - 4.0
Subsystem     : Windows GUI
Image Flags(1) : ImageIsDLL (<.) DebugStrip (<.) IsExecutable(x)
Image Flags(2) : RelocatStrip(x) LineNumStrip(x) LocSymbStrip(x)
Image Flags(3) : 32bit image (x) 16bit image (<.) System file (<.)
Image Flags(bit) : 00000000100001111
HeapStack     : Reserve - 00100000/00100000, Commit - 00001000/00001000
Data Sizes    : InitDataSize : 00001000, UninitDataSize : 0000B000
Other Sizes(1) : CodeSize      : 00004000, ImageSize      : 00011000
Other Sizes(2) : HeaderSize    : 00001000, OpHeaderSize(NT) : 000000E0
Bases         : BaseOfCode     : 0000C000, BaseOfData      : 00010000
Alignments    : FileAlignment: 00000200, SectionAlignment: 00001000
Objects table  :
Object ( 1 ) : [ JDR0 | 0000B000 | 00001000 | 00000000 | 00000400 | E0000080 ]
Object ( 2 ) : [ JDR1 | 00004000 | 0000C000 | 00004000 | 00000400 | E0000040 ]
Object ( 3 ) : [ .rsrc | 00001000 | 00010000 | 00000E00 | 00004400 | C0000040 ]
Processed with(1): Packed with UPX v0.93 or v1.00 (PE)
Processed with(2): Windows executable
DOSEXE part sizes: Header 64 bytes, image 1104 bytes, overlay 19824 bytes
WINEXE part sizes: Header 49088 bytes, image -28096 bytes, overlay 0 bytes
Entrypoint      : DOS 64/00000040, RVA 64800/0000FD20, WIN 16672/00004120
Information(1)  : Overlay start from 00000000/0000E000
Extension ( 1 ) : DOS executal

C:\FILEAN~1>
```

Packed with UPX v0.93 or v1.00 (PE)



静态分析-自动脱壳

□ 自动脱壳失败

- 上一步已经探测出被加了**UPX**壳，使用**UPX**脱

```
C:\WINDOWS\system32\cmd.exe
12/16/2007 01:02 AM          5,448 LICENSE
12/16/2007 01:02 AM        18,795 NEWS
12/16/2007 01:02 AM          4,923 README
12/16/2007 01:02 AM           773 README.1ST
12/16/2007 01:02 AM          2,145 THANKS
12/16/2007 01:02 AM          2,315 TODO
12/16/2007 01:02 AM        43,436 upx.1
12/16/2007 01:02 AM        37,214 upx.doc
12/16/2007 01:02 AM       269,312 upx.exe
12/16/2007 01:02 AM        42,657 upx.html
          12 File(s)          446,862 bytes
           2 Dir(s) 6,618,353,664 bytes free

C:\upx302w>upx ..\RaDa.exe
          Ultimate Packer for eXecutables
Copyright (C) 1996,1997,1998,1999,2000,2001,2002,2003,2004,2005,2006,2007
UPX 3.02w Markus Oberhumer, Laszlo Molnar & John Reiser Dec 16th 2007

  File size      Ratio      Format      Name
-----
upx: ..\RaDa.exe: NotCompressibleException

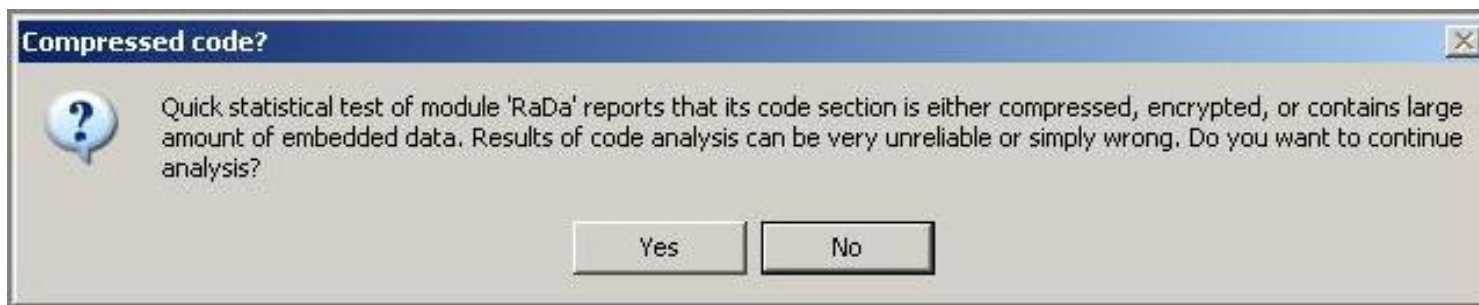
Packed 1 file: 0 ok, 1 error.

C:\upx302w>
```


静态分析-手工脱壳(0)

□ 步骤0: Ollydbg打开加壳代码

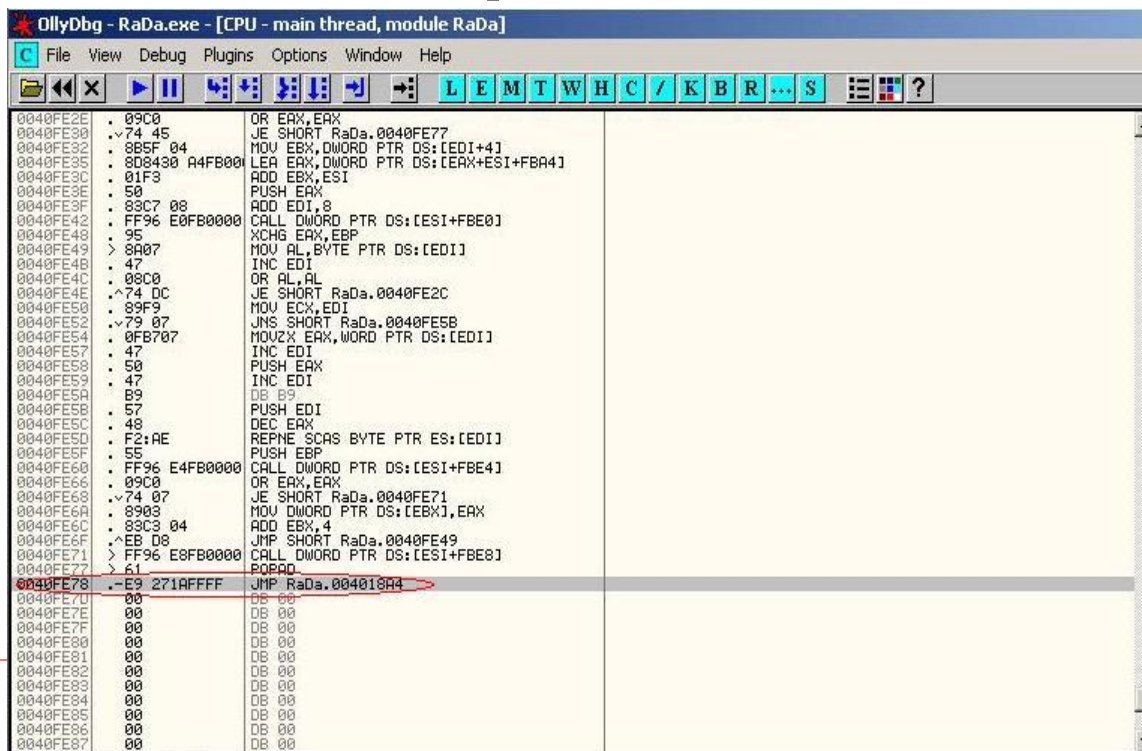
- 直接使用**UPX**脱壳行不通，考虑使用**Ollydbg**将运行时脱壳的代码**Dump**出来。用**Ollydbg**打开**RaDa.exe**，出现如下信息



静态分析-手工脱壳(1)

□ 步骤1：寻找入口点

- 在打开后的代码中寻找**JMP**指令，这个可能是到达脱壳后代码的入口（看雪论坛-**UPX**脱壳经验）



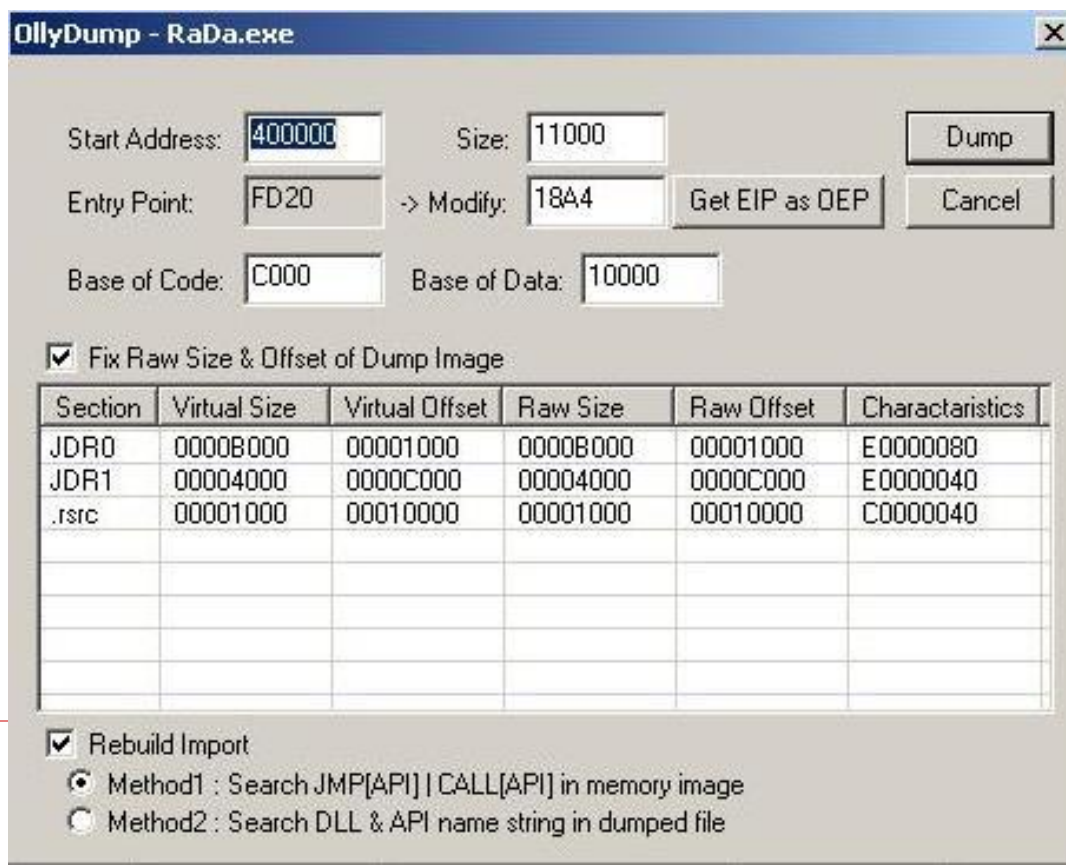
```

OllyDbg - RaDa.exe - [CPU - main thread, module RaDa]
File View Debug Plugins Options Window Help
L E M T W H C / K B R ... S
0040FE2E . 09C0 OR EAX,EAX
0040FE30 . 74 45 JE SHORT RaDa.0040FE77
0040FE32 . 8B5F 04 MOV EBX,DWORD PTR DS:[EDI+4]
0040FE35 . 0D8430 A4FB00 LEA EAX,DWORD PTR DS:[EAX+ESI+FBA4]
0040FE3C . 01F3 ADD EBX,ESI
0040FE3E . 50 PUSH EAX
0040FE3F . 83C7 08 ADD EDI,8
0040FE42 . FF96 E0FB0000 CALL DWORD PTR DS:[ESI+FB00]
0040FE48 . 95 XCHG EAX,EBP
0040FE49 > 8A07 MOV AL,BYTE PTR DS:[EDI]
0040FE4B . 47 INC EDI
0040FE4C . 09C0 OR AL,AL
0040FE4E . 74 DC JE SHORT RaDa.0040FE2C
0040FE50 . 8BF9 MOV ECX,EDI
0040FE52 . 79 07 JNS SHORT RaDa.0040FE5B
0040FE54 . 0FB707 MOVZX EAX,WORD PTR DS:[EDI]
0040FE57 . 47 INC EDI
0040FE58 . 50 PUSH EAX
0040FE59 . 47 INC EDI
0040FE5A . B9 DB B9
0040FE5B . 57 PUSH EDI
0040FE5C . 48 DEC EAX
0040FE5D . F2AE REPNE SCAS BYTE PTR ES:[EDI]
0040FE5F . 55 PUSH EBP
0040FE60 . FF96 E4FB0000 CALL DWORD PTR DS:[ESI+FB00]
0040FE66 . 09C0 OR EAX,EAX
0040FE68 . 74 07 JE SHORT RaDa.0040FE71
0040FE6A . 8903 MOV DWORD PTR DS:[EBX],EAX
0040FE6C . 83C3 04 ADD EBX,4
0040FE6F . EB D8 JMP SHORT RaDa.0040FE49
0040FE71 > FF96 E8FB0000 CALL DWORD PTR DS:[ESI+FB00]
0040FE77 > 61 POP EAX
0040FE78 > E9 271AFFFF JMP RaDa.004018A4
0040FE7D . 00 DB 00
0040FE7E . 00 DB 00
0040FE7F . 00 DB 00
0040FE80 . 00 DB 00
0040FE81 . 00 DB 00
0040FE82 . 00 DB 00
0040FE83 . 00 DB 00
0040FE84 . 00 DB 00
0040FE85 . 00 DB 00
0040FE86 . 00 DB 00
0040FE87 . 00 DB 00
  
```

静态分析-手工脱壳(2)

□ 步骤2: Dump出代码段

■ 使用OllyDump插件Dump代码



OllyDump - RaDa.exe

Start Address: 400000 Size: 11000 Dump

Entry Point: FD20 -> Modify: 18A4 Get EIP as OEP Cancel

Base of Code: C000 Base of Data: 10000

☒ Fix Raw Size & Offset of Dump Image

Section	Virtual Size	Virtual Offset	Raw Size	Raw Offset	Characteristics
JDR0	00008000	00001000	00008000	00001000	E0000080
JDR1	00004000	0000C000	00004000	0000C000	E0000040
.rsrc	00001000	00010000	00001000	00010000	C0000040

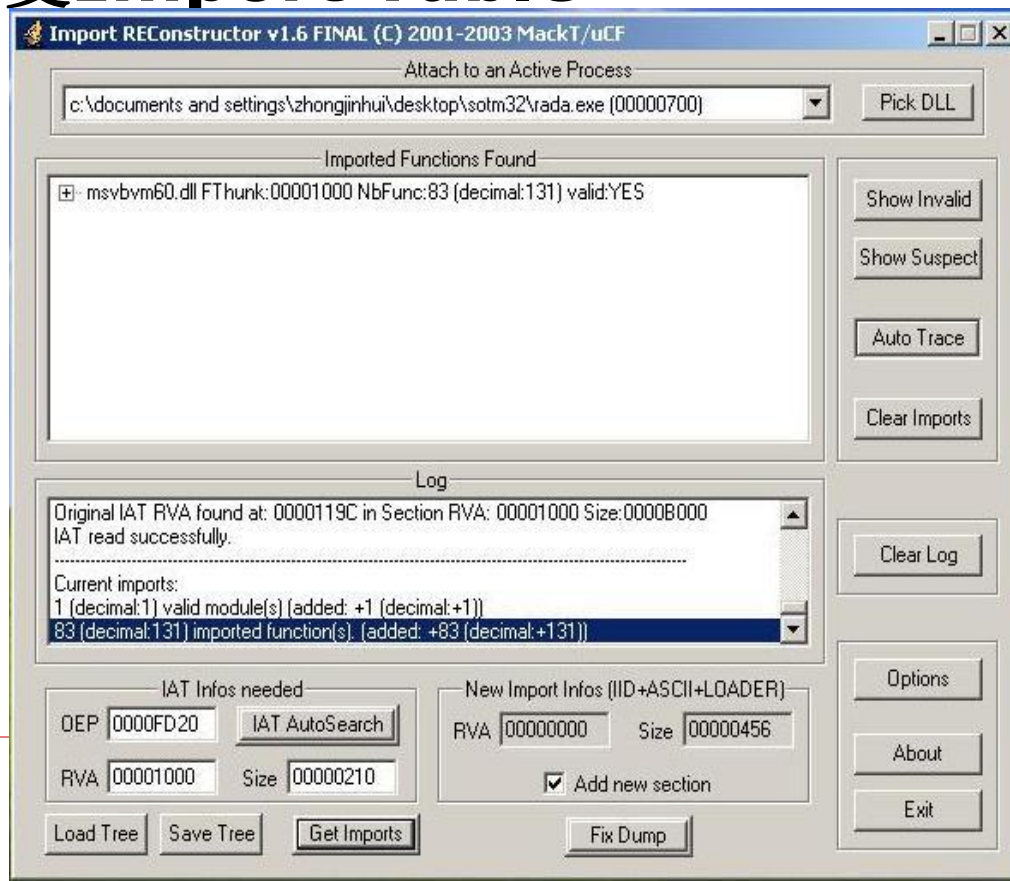
☒ Rebuild Import

- ☒ Method1 : Search JMP[API] | CALL[API] in memory image
- ☐ Method2 : Search DLL & API name string in dumped file

静态分析-手工脱壳(3)

□ 步骤3：修复PE文件

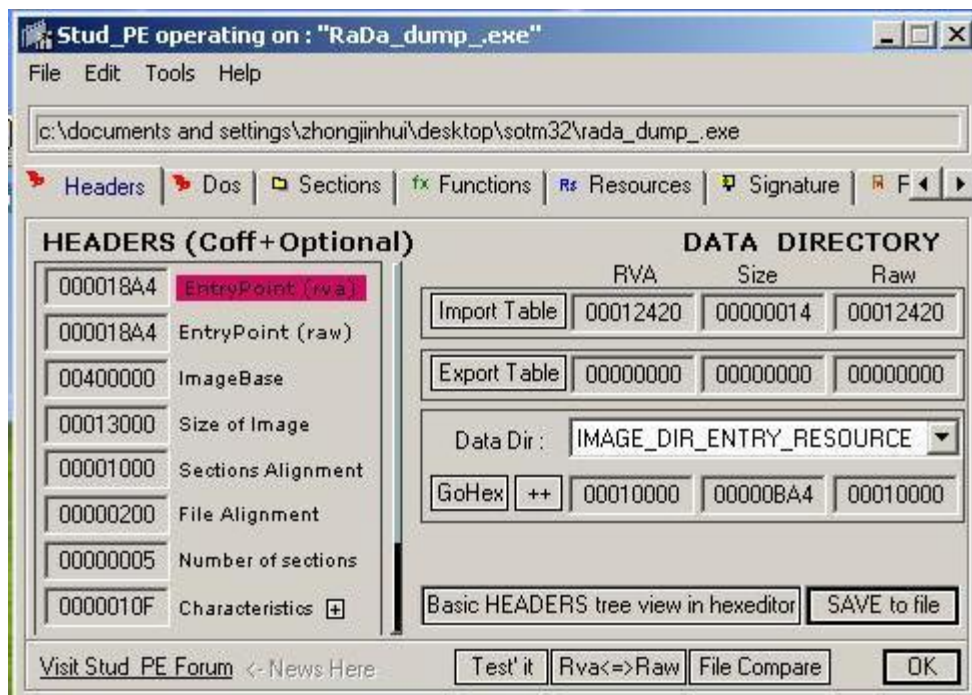
■ 修复Import Table



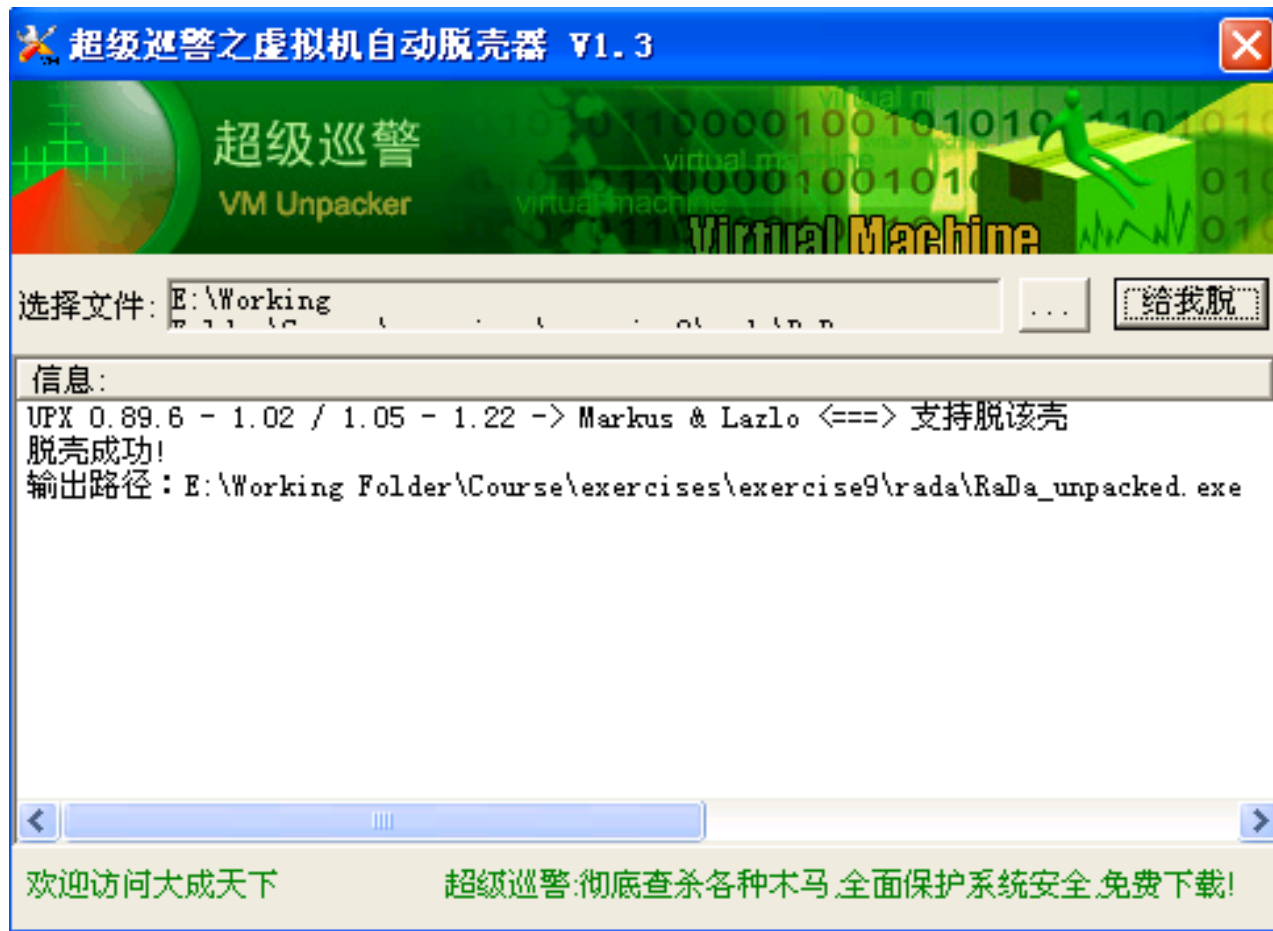
静态分析-手工脱壳(4)

□ 步骤3：修复PE文件

■ 修复Entry Point



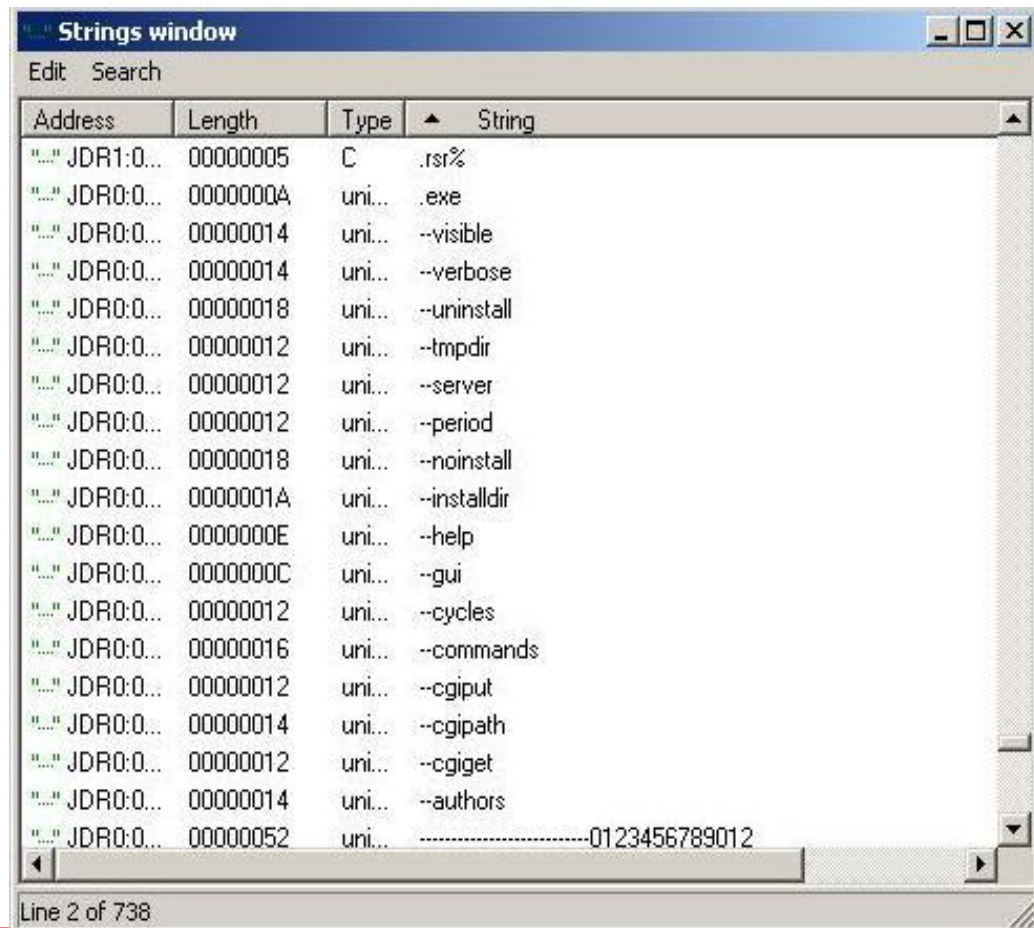
静态分析 – 超级巡警自动脱壳机



进一步分析脱壳文件-IDA Pro

□ Strings

- +Unicode
- 命令行参数
- Copyright (C)
2004 Raul Siles
& David Perez
- VMware Tools
- RaDa_commands
.html
- DDoS Smurf
- HKLM\\...\\Run
- C:\\RaDa\\bin
- __vba****: VB



行为分析

□ 尝试使用不同的参数，分析各个参数的含义

- **--gui**
- **--help**
- **--authors**
- **--server**
- ...





静态分析-二进制程序逻辑分析

- 命令行参数解析逻辑
- 以参数字符串为线索的分析
 - 搜索发现参数字符串都出现在**sub_405E40**中
 - 分析**sub_405E40**，发现这是处理参数配置的子程序，跟踪对不同参数的处理，理清参数的作用

--verbose	加了这个参数之后会显示"Starting DDoS Smurf remote attack..."，除此之外没有别的不同
--visible	决定在获取html文件时，IE窗口是否可见
--server	指定命令文件的位置，包括控制服务器的ip地址、访问协议及目录等，默认是 http://.10/RaDa 。
--commands	指定命令文件，默认是RaDa_commands.html
--cgipath	指定服务器上cgi文件的根目录，默认是cgi-bin
--cgiget	指定负责文件上传的cgi脚本，默认是upload.cgi
--cgiout	指定负责文件下载的cgi脚本，默认是download.cgi
--tmpdir	指定临时文件夹的位置，默认是C:\RaDa\tmp
--period	指定两次向服务器请求命令文件的时间间隔，默认是60秒
--cycles	指定多少次向服务器请求命令文件后退出，默认是0(没有限制)
--help	输出版权信息
--gui	使用该参数会使样本出现一个GUI窗口
--installdir	指定样本的安装路径，默认是C:\RaDa\bin
--noinstall	使用该参数，样本将不会安装、也不会添加注册表
--uninstall	卸载样本
--authors	如果确认不是在VMware的虚拟机中运行，则显示样本的作者；否则显示参数不存在



静态分析-二进制程序逻辑分析(2)

- 深入分析程序接受控制的逻辑
- 以**RaDa_commands.html**为线索的分析
 - **sub_404FB0**中，这个字符串被复制到了变量**dword_40C030**
 - 搜索**dword_40C030(Xref)**找到**sub_4052C0**
 - **sub_4052C0** 确保服务器在内网网段，到服务器中获取命令文件，解析并执行其中的指令

exe	在宿主主机中执行指定的命令
put	将宿主主机中的指定文件上传到服务器
get	将服务器中的指定文件下载到宿主主机中
screenshot	截取宿主主机的屏幕并保存到tmp文件夹
sleep	停止活动一段时间



静态分析-二进制程序逻辑分析(3)

- **VMware识别?**
- **以HKLM\\Software\\VMware, Inc.\\VMware Tools\\InstallPath为线索的分析**
 - 在**sub_404FB0(authors命令处理函数)**中, 被复制到变量**dword_40C070**
 - 搜索**dword_40C070** , 找到**sub_40AAA0**
 - **sub_40AAA0** 获取了网卡的配置信息, 并检查了**MAC**地址, 然后确认**HKLM\\Software\\VMware, Inc.\\VMware Tools\\InstallPath**是否存在
 - 逻辑理解

--authors	如果确认不是在VMware的虚拟机中运行, 则显示样本的作者; 否则显示参数不存在
-----------	---



问题1解答

□ 样本摘要及基本信息

- 大小: **20,992**字节

- **MD5: md5sum**计算

caaa6985a43225a0b3add54f44a0d4c7

- **PE**文件格式

- 运行在**Windows 2000, XP and 2003**及以上版的操作系统中

- **UPX**加壳并进行了壳伪装处理

- **UPX**段改名为**JDR**

- 版本号从**1.25**改为**0.99**



问题2解答

- 找出并解释这个二进制文件的目的
 - 后门工具 -> **HTTP Bot?**
 - 能够使远程的攻击者完全地控制系统
 - 它采用的通信方式使得只要系统能够通过浏览器上网，就能够获得来自攻击者的指令。

问题3解答

- 识别并说明这个二进制文件所具有的不同特性
- **RaDa.exe**被执行时，它会将自身安装到系统中，并通过修改注册表的方式使得每次系统启动，它都能够被启动，启动后循环执行一下操作：
 - (1)从指定的**web**服务器请求指定的**web**页面；
 - (2)解析获得的**web**页面，获取其中的指令
 - (3)执行解析出来的指令
 - (4)等待一段时间
 - (5)返回第(1)步
- 启动后，**RaDa**一直在后台运行，不会弹出任何窗口。它支持以下指令：

exe	在宿主主机中执行指定的命令
put	将宿主主机中的指定文件上传到服务器
get	将服务器中的指定文件下载到宿主主机中
screenshot	截取宿主主机的屏幕并保存到tmp文件夹
sleep	停止活动一段时间



问题4解答

- 识别并说明这个二进制文件所采用通讯方法
 - 通过**HTTP**协议进行通信，**RaDa**通过调用隐藏的**IE**实例向**web**服务器发送请求，获取命令。

□ Snort规则

- alert tcp any any -> any \$HTTP_PORTS (msg:"RaDa Activity Detected - Commands Request"; flow:to_server,established; content:"GET /RaDa/RaDa_commands.html"; depth:30; classtype:trojan-activity; sid:1000001; rev:1;)
- alert tcp any \$HTTP_PORTS -> any any (msg:"RaDa Activity Detected - Commands Page"; flow:from_server,established; content:"NAME=exe"; nocase; depth:1024; classtype:trojan-activity; sid:1000003; rev:1;)
- alert tcp any any -> any \$HTTP_PORTS (msg:"RaDa Activity Detected - Multipart Message"; flow:to_server,established; content:"boundary=-----0123456789012"; depth:1024; classtype:trojan-activity; sid:1000004; rev:1;)



问题5解答

- 识别并解释这个二进制文件中所采用的防止被分析或逆向工程的技术
 - (1)这个二进制文件使用**UPX**加壳之后又做了手工修改，使得难以脱壳，而不脱壳又会影响反汇编。
 - (2)脱壳之后，可以在文件找到字符串“**Starting DDoS Smurf remote attack.**”这会使分析者误以为样本可以发动**DDos**攻击，而实际上它没有提供任何用以发动**DDos**攻击的子程序。
 - (3)提供了一**help**参数，但是使用这个参数之后，除了输出版权信息外并没有提供其他有用的信息。**--verbose**参数也没有什么用处，**GUI**窗口中的“**Show config**”及“**Show usage**”也显示与一**help**参数相同的信息。
 - (4)通过查看网卡的**MAC**地址以及查看**VMware Tools**的注册表项来判断操作系统是否运行在**VMware**虚拟机上，如果是，则使用一**authors**参数时将不会输出作者信息。



问题6、7解答

- 对这个恶意代码样本进行分类（病毒、蠕虫等），并给出你的理由
 - 这是一个后门程序，运行并按照攻击者命令进行工作；如果是多对1控制，则可认为是**HTTP Bot**；
 - 这个样本不具有传播和感染的性质，所以它不属于病毒和蠕虫；
 - 它也没有将自己伪装成有用的程序以欺骗用户运行，所以他也不属于木马。
- 给出过去已有的具有相似功能的其他工具
 - **Bobax** – 2004年发现的木马，也是使用**HTTP**协议从指定的服务器下载命令文件，然后解析并执行其中的指令。
 - **Setiri**及其前辈**GatSlag**



问题8解答

- 提出你用于对抗由这个二进制文件带来的安全威胁的检测和防御方法
 - 提高用户的安全意识
 - 在系统上运行反病毒软件并实时更新特征库
 - 特别关注**ASEP**点挂接的自启动程序(**360, SReng, ...**)
 - 在服务器上限制**Web**访问
 - 研发并使用基于行为的、或基于异常的检测方法和方案



问题9解答-bonus

- 可能调查出这个二进制文件的开发作者吗？
如果可以，在什么样的环境和什么样的限定条件下？
 - 可以。
 - 在非**VMware**虚拟机上使用**—authors**参数运行就可以看到这个二进制文件的作者。



问题10解答-bonus

- 你可以预见在和这个二进制文件具有相同目的的工具上有什么技术进展吗？
 - 1对多控制关系：**HTTP**僵尸网络
 - 更好的管理终端
 - 更佳的控制隐蔽性：匿名、加密通讯、其他通讯协议(**P2P**僵尸网络)
 - 控制命令的认证和多样化：控制者身份认证、更多的控制命令支持
 - 后门程序的隐藏性：使用多态或变形技术、**Rootkit**进行主机隐藏

Thanks

诸葛建伟

zhugejianwei@icst.pku.edu.cn