**重庆大学课程设计报告**

课程设计题目： 针对缓冲区溢出漏洞的远程攻击设计与实现

学 院： 计算机学院

专业班级： 信息安全1、2班

年 级： 2018

姓 名：王浩、李涵威、王勋、方香宜

学 号：

完成时间： 2021 年 1 月 7 日

成 绩：

指导教师： 杨吉云

重庆大学教务处制

课程设计指导教师评定成绩表

学号： 姓名： 班级：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 项目 | 分值 | 优秀  (100>x≥90) | 良好  (90>x≥80) | 中等  (80>x≥70) | 及格  (70>x≥60) | 不及格(x<60) | 评分 |
| 参考标准 | 参考标准 | 参考标准 | 参考标准 | 参考标准 |
| 学习态度 | 15 | 学习态度认真，科学作风严谨，严格保证设计时间并按任务书中规定的进度开展各项工作 | 学习态度比较认真，科学作风良好，能按期圆满完成任务书规定的任务 | 学习态度尚好，遵守组织纪律，基本保证设计时间，按期完成各项工作 | 学习态度尚可，能遵守组织纪律，能按期完成任务 | 学习马虎，纪律涣散，工作作风不严谨,不能保证设计时间和进度 |  |
| 技术水平与实际能力 | 25 | 设计合理、理论分析与计算正确，实验数据准确，有很强的实际动手能力、经济分析能力和计算机应用能力，文献查阅能力强、引用合理、调查调研非常合理、可信 | 设计合理、理论分析与计算正确，实验数据比较准确，有较强的实际动手能力、经济分析能力和计算机应用能力，文献引用、调查调研比较合理、可信 | 设计合理，理论分析与计算基本正确，实验数据比较准确，有一定的实际动手能力，主要文献引用、调查调研比较可信 | 设计基本合理，理论分析与计算无大错，实验数据无大错 | 设计不合理，理论分析与计算有原则错误，实验数据不可靠，实际动手能力差，文献引用、调查调研有较大的问题 |  |
| 创新 | 10 | 有重大改进或独特见解，有一定实用价值 | 有较大改进或新颖的见解，实用性尚可 | 有一定改进或新的见解 | 有一定见解 | 观念陈旧 |  |
| 论文(计算书、图纸)撰写质量 | 50 | 结构严谨，逻辑性强，层次清晰，语言准确，文字流畅，完全符合规范化要求，书写工整或用计算机打印成文；图纸非常工整、清晰 | 结构合理，符合逻辑，文章层次分明，语言准确，文字流畅，符合规范化要求，书写工整或用计算机打印成文；图纸工整、清晰 | 结构合理，层次较为分明，文理通顺，基本达到规范化要求，书写比较工整；图纸比较工整、清晰 | 结构基本合理，逻辑基本清楚，文字尚通顺，勉强达到规范化要求；图纸比较工整 | 内容空泛，结构混乱，文字表达不清，错别字较多，达不到规范化要求；图纸不工整或不清晰 |  |

指导教师评定成绩：

课程设计指导教师评定成绩表

学号： 姓名： 班级：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 项目 | 分值 | 优秀  (100>x≥90) | 良好  (90>x≥80) | 中等  (80>x≥70) | 及格  (70>x≥60) | 不及格(x<60) | 评分 |
| 参考标准 | 参考标准 | 参考标准 | 参考标准 | 参考标准 |
| 学习态度 | 15 | 学习态度认真，科学作风严谨，严格保证设计时间并按任务书中规定的进度开展各项工作 | 学习态度比较认真，科学作风良好，能按期圆满完成任务书规定的任务 | 学习态度尚好，遵守组织纪律，基本保证设计时间，按期完成各项工作 | 学习态度尚可，能遵守组织纪律，能按期完成任务 | 学习马虎，纪律涣散，工作作风不严谨,不能保证设计时间和进度 |  |
| 技术水平与实际能力 | 25 | 设计合理、理论分析与计算正确，实验数据准确，有很强的实际动手能力、经济分析能力和计算机应用能力，文献查阅能力强、引用合理、调查调研非常合理、可信 | 设计合理、理论分析与计算正确，实验数据比较准确，有较强的实际动手能力、经济分析能力和计算机应用能力，文献引用、调查调研比较合理、可信 | 设计合理，理论分析与计算基本正确，实验数据比较准确，有一定的实际动手能力，主要文献引用、调查调研比较可信 | 设计基本合理，理论分析与计算无大错，实验数据无大错 | 设计不合理，理论分析与计算有原则错误，实验数据不可靠，实际动手能力差，文献引用、调查调研有较大的问题 |  |
| 创新 | 10 | 有重大改进或独特见解，有一定实用价值 | 有较大改进或新颖的见解，实用性尚可 | 有一定改进或新的见解 | 有一定见解 | 观念陈旧 |  |
| 论文(计算书、图纸)撰写质量 | 50 | 结构严谨，逻辑性强，层次清晰，语言准确，文字流畅，完全符合规范化要求，书写工整或用计算机打印成文；图纸非常工整、清晰 | 结构合理，符合逻辑，文章层次分明，语言准确，文字流畅，符合规范化要求，书写工整或用计算机打印成文；图纸工整、清晰 | 结构合理，层次较为分明，文理通顺，基本达到规范化要求，书写比较工整；图纸比较工整、清晰 | 结构基本合理，逻辑基本清楚，文字尚通顺，勉强达到规范化要求；图纸比较工整 | 内容空泛，结构混乱，文字表达不清，错别字较多，达不到规范化要求；图纸不工整或不清晰 |  |

指导教师评定成绩：

课程设计指导教师评定成绩表

学号： 姓名： 班级：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 项目 | 分值 | 优秀  (100>x≥90) | 良好  (90>x≥80) | 中等  (80>x≥70) | 及格  (70>x≥60) | 不及格(x<60) | 评分 |
| 参考标准 | 参考标准 | 参考标准 | 参考标准 | 参考标准 |
| 学习态度 | 15 | 学习态度认真，科学作风严谨，严格保证设计时间并按任务书中规定的进度开展各项工作 | 学习态度比较认真，科学作风良好，能按期圆满完成任务书规定的任务 | 学习态度尚好，遵守组织纪律，基本保证设计时间，按期完成各项工作 | 学习态度尚可，能遵守组织纪律，能按期完成任务 | 学习马虎，纪律涣散，工作作风不严谨,不能保证设计时间和进度 |  |
| 技术水平与实际能力 | 25 | 设计合理、理论分析与计算正确，实验数据准确，有很强的实际动手能力、经济分析能力和计算机应用能力，文献查阅能力强、引用合理、调查调研非常合理、可信 | 设计合理、理论分析与计算正确，实验数据比较准确，有较强的实际动手能力、经济分析能力和计算机应用能力，文献引用、调查调研比较合理、可信 | 设计合理，理论分析与计算基本正确，实验数据比较准确，有一定的实际动手能力，主要文献引用、调查调研比较可信 | 设计基本合理，理论分析与计算无大错，实验数据无大错 | 设计不合理，理论分析与计算有原则错误，实验数据不可靠，实际动手能力差，文献引用、调查调研有较大的问题 |  |
| 创新 | 10 | 有重大改进或独特见解，有一定实用价值 | 有较大改进或新颖的见解，实用性尚可 | 有一定改进或新的见解 | 有一定见解 | 观念陈旧 |  |
| 论文(计算书、图纸)撰写质量 | 50 | 结构严谨，逻辑性强，层次清晰，语言准确，文字流畅，完全符合规范化要求，书写工整或用计算机打印成文；图纸非常工整、清晰 | 结构合理，符合逻辑，文章层次分明，语言准确，文字流畅，符合规范化要求，书写工整或用计算机打印成文；图纸工整、清晰 | 结构合理，层次较为分明，文理通顺，基本达到规范化要求，书写比较工整；图纸比较工整、清晰 | 结构基本合理，逻辑基本清楚，文字尚通顺，勉强达到规范化要求；图纸比较工整 | 内容空泛，结构混乱，文字表达不清，错别字较多，达不到规范化要求；图纸不工整或不清晰 |  |

指导教师评定成绩：

课程设计指导教师评定成绩表

学号： 姓名： 班级：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 项目 | 分值 | 优秀  (100>x≥90) | 良好  (90>x≥80) | 中等  (80>x≥70) | 及格  (70>x≥60) | 不及格(x<60) | 评分 |
| 参考标准 | 参考标准 | 参考标准 | 参考标准 | 参考标准 |
| 学习态度 | 15 | 学习态度认真，科学作风严谨，严格保证设计时间并按任务书中规定的进度开展各项工作 | 学习态度比较认真，科学作风良好，能按期圆满完成任务书规定的任务 | 学习态度尚好，遵守组织纪律，基本保证设计时间，按期完成各项工作 | 学习态度尚可，能遵守组织纪律，能按期完成任务 | 学习马虎，纪律涣散，工作作风不严谨,不能保证设计时间和进度 |  |
| 技术水平与实际能力 | 25 | 设计合理、理论分析与计算正确，实验数据准确，有很强的实际动手能力、经济分析能力和计算机应用能力，文献查阅能力强、引用合理、调查调研非常合理、可信 | 设计合理、理论分析与计算正确，实验数据比较准确，有较强的实际动手能力、经济分析能力和计算机应用能力，文献引用、调查调研比较合理、可信 | 设计合理，理论分析与计算基本正确，实验数据比较准确，有一定的实际动手能力，主要文献引用、调查调研比较可信 | 设计基本合理，理论分析与计算无大错，实验数据无大错 | 设计不合理，理论分析与计算有原则错误，实验数据不可靠，实际动手能力差，文献引用、调查调研有较大的问题 |  |
| 创新 | 10 | 有重大改进或独特见解，有一定实用价值 | 有较大改进或新颖的见解，实用性尚可 | 有一定改进或新的见解 | 有一定见解 | 观念陈旧 |  |
| 论文(计算书、图纸)撰写质量 | 50 | 结构严谨，逻辑性强，层次清晰，语言准确，文字流畅，完全符合规范化要求，书写工整或用计算机打印成文；图纸非常工整、清晰 | 结构合理，符合逻辑，文章层次分明，语言准确，文字流畅，符合规范化要求，书写工整或用计算机打印成文；图纸工整、清晰 | 结构合理，层次较为分明，文理通顺，基本达到规范化要求，书写比较工整；图纸比较工整、清晰 | 结构基本合理，逻辑基本清楚，文字尚通顺，勉强达到规范化要求；图纸比较工整 | 内容空泛，结构混乱，文字表达不清，错别字较多，达不到规范化要求；图纸不工整或不清晰 |  |

指导教师评定成绩：

**目 录**

[一、项目概述](#_Toc60835274)

[二、任务要求](#_Toc60835275)

[三、需求分析](#_Toc60835276)

[四、详细设计](#_Toc60835277)

[五、结果及分析](#_Toc60835278)

[六、分工情况说明](#_Toc60835286)

[七、参考资料](#_Toc60835295)

1. **项目概述**

构造一个基于socket通信的client-server类型软件，该软件具有缓存区溢出漏洞。根据漏洞特征通过远程链接实现对server端的攻击，做到绕过密码进行提权，并上传病毒破坏server端。

1. **任务要求**
2. 设计并实现一个客户端和服务端的远程控制软件，要求如下：
3. 它们之间采用Socket方式链接，服务端监听端口不对外公开；
4. 设计服务端和客户端之间的控制协议，包括：
5. 含有两种级别权限，高级别权限除了拥有低级别功能外还具有文件传输功能；
6. 两级权限登录验证，低级别权限验证通过后再输入高级别权限密码进行验证；
7. 低级别权限密码采用明文对比，高级别权限密码采用DES加密后对比；
8. 服务端可根据客户端传输命令产生相应的动作；
9. 服务端设计一个缓冲区溢出漏洞，通过该漏洞可跨过高级别权限密码验证；
10. 根据服务端的漏洞设计一个攻击方案并编程实现，要求如下：
11. 逆向分析服务端程序，解析出低级别验证密码；
12. 利用工具软件或者编程实现扫描服务端的监听端口，前者得分低于后者；
13. 针对服务端漏洞程实现缓冲区溢出攻击，通过弹对话框的形式提示攻击成功；
14. 在3）的基础上，传输一个文件到服务端并加密服务端上指定的文件；
15. 在4）的基础上，传输的文件还可以感染其它可执行文件；
16. **需求分析**
17. **服务端**
18. 普通用户认证：使用明文验证
19. 普通用户功能：
20. Echo功能：返回用户输入的文字
21. 高级别权限用户认证：使用DES加密进行验证
22. 高级别权限用户功能：
23. Echo功能：返回用户输入的文字
24. 上传文件功能：用户可根据路径上传文件
25. 执行文件功能：用户可根据文件名执行文件
26. **客户端**
27. 普通用户认证：客户端启动时执行的认证
28. 高级别权限用户认证：低级权限认证后的认证
29. Echo命令：打印客户端输入的文字
30. 上传文件命令：上传客户端指定路径文件
31. 执行文件命令：执行客户端输入的文件
32. **攻击脚本**
33. 突破普通权限认证：根据静态逆向服务端获取普通用户密码
34. 利用堆栈溢出突破高级权限认证：根据堆栈溢出原理进行提权
35. 上传病毒文件：根据静态逆向服务端获取上传文件的格式并上传病毒文件
36. 执行病毒文件：根据静态逆向服务端获取执行文件的格式并执行病毒文件
37. **病毒**
38. 加密指定文件：加密服务端指定文件
39. 感染其它可执行文件：感染其它文件，将病毒代码节写入可执行文件
40. **端口扫描软件**
41. 扫描网段端口：提供host、port信息扫描网段开放端口
42. **详细设计**
43. **服务端**

采用c语言编写，服务端存有一个client结构体数组，包含有client的权限级别，0表示普通用户，1表示高级别权限用户。

1. 普通用户认证：固定用户密码，本次仿真仅提供一个普通用户，用户密码存放于服务端，可被逆向分析出来。
2. 普通用户功能：本次普通用户仅有一个echo功能，作用仅为回显用户输入的字符串。
3. 高级别权限认证：用户传输DES加密所需要的明文密码和密钥，在服务器端进行加密处理。本次方针仅提供一个高级用户，高级用户的DES加密密码作为全局变量存在。这样便不会存在用户端加密，而攻击者可以逆向出DES加密后的密码，进而只需要输入加密的密文就能突破提权的情况。
4. 高级别权限用户功能：
   * 1. Echo功能：将用户的命令进行解析，返回echo后的字符
     2. 上传文件功能：上传文件进行大小限制，限制在100kb，将文件存储在server可执行文件的当前目录，同时对于消息的格式为f+file\_name+”\r\n”+file\_content，服务端进行解析然后存储文件。
     3. 执行文件功能：使用windows的API为可执行文件创建进程来执行文件。
5. **客户端**

采用python编写

1. 普通用户认证：普通用户的消息采用0+消息内容，此处为0+密码传递给服务端，且一开始就会先执行普通用户登录
2. 普通用户功能：仅有echo功能，发送的信息以e开头，格式为e+message。
3. 高级别权限用户认证：以1开头，分别包含密码和密钥，且进行了密码密钥截断，防止客户端的缓冲区溢出攻击。
4. 高级别权限用户功能：
   * 1. Echo功能：和普通用户功能一样
     2. 文件上传功能：格式为f+file\_name+”\r\n”+file\_content。发送前检查文件是否存在。
     3. 文件执行功能：格式为r+文件名称，对命令语句进行限制，仅能执行一个文件。

整体的功能表单如下

|  |  |  |
| --- | --- | --- |
| **命令** | **命令格式** | **传输格式** |
| **普通用户登录** |  | 0+[password] |
| **高级别权限用户登录** |  | 1+[password]+[deskey] |
| **Echo** | echo [message] | e+message |
| **上传文件** | upload [file\_name] | f+file\_name+”\r\n”+file\_content |
| **运行文件** | run [file\_name] | r+file\_name |

1. **攻击脚本**

攻击脚本流程如下：

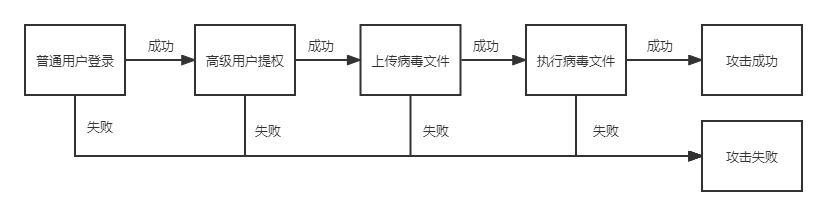


图 1脚本攻击流程

其中主要是缓冲区溢出漏洞，在server端的admin认证区域设立了缓冲区溢出漏洞，主要的代码如下：

|  |
| --- |
| **int admin = 0;**  **char keyAndPassword[16];**  **strcpy(keyAndPassword, command);**  **printf("%p %p\n", &admin, &keyAndPassword);** |

利用strcpy的缓冲区溢出漏洞，构造一个很长的字符串使得堆栈溢出，因为在进行用户登录成功判断是根据admin是否不等于0，因此只需要覆盖掉admin的值即可。而printf则是为了使得缓冲区溢出漏洞生效，不printf admin，则编译器会优化使得堆栈溢出失效。

1. **病毒**

整个病毒采用汇编书写，整体汇编流程如下：

1. 找到偏移量进行重定位
2. 搜索kernel32.dll找到找到pe导出表进而找到所有API地址，指针每次递减1000h即一个page大小，知道找到Pe头
3. 寻找user32.dll进而找到messagebox的地址
4. 根据需求寻找是系统目录还是当前目录
5. 进行感染，打开第一个文件，判断是否有足够空间加入新节，作为感染标志
6. 然后进行文件加密，这里因为只是仿真需要，代码内指定了文件路径，通过之前找到的api调用文件映射，将文件内容进行简单的异或加密。
7. 至此病毒文件工作结束。
8. **端口扫描**

采用python进行构建。调用argparse进行命令解析，在指定端口范围内使用通过socket进行连接测试，连接成功即端口开放。

1. **结果及分析**
2. **Client-server测试**：

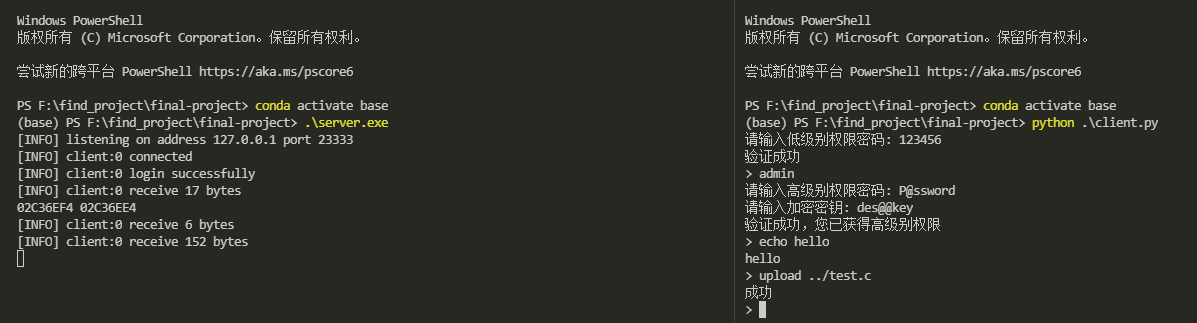


图 2测试结果1

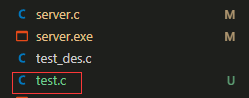


图 3上传文件成功

这里执行文件命令执行一个弹窗程序

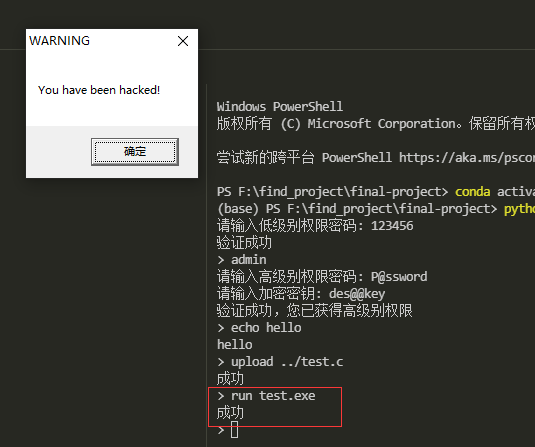


图 4测试结果2

程序正常工作。

1. **攻击方分析：**

首先尝试使用端口软件扫描打开的端口，因为本地端口较多，这里只尝试server的端口是否开放

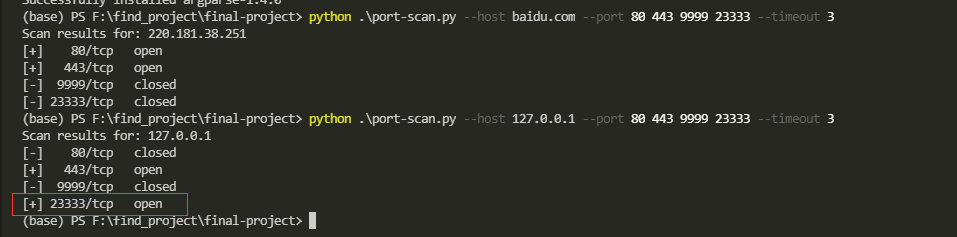


图 5端口扫描结果

可以看见server.exe的23333端口开放了。

然后开始静态分析该server，首先处理普通用户登录。查看string的时候看见123456，大概率是一个密码，查看对应的Graph使用

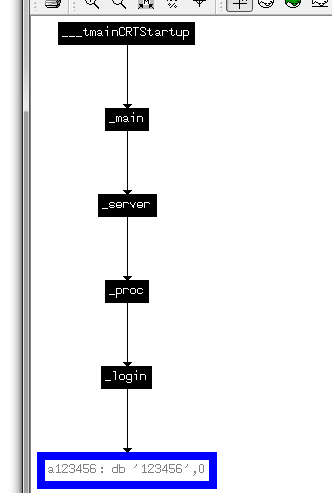


图 6“123456”引用情况

刚好是被login使用，同时可以跳到Login进行查看：

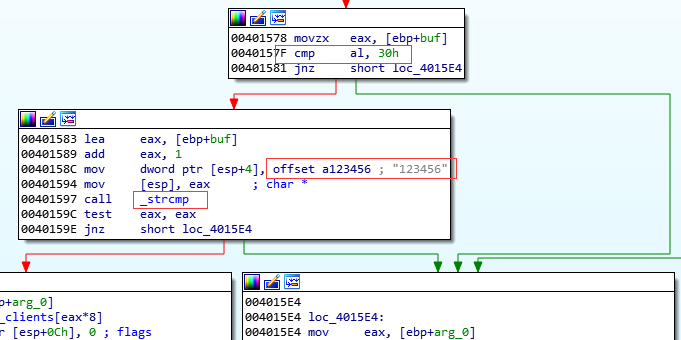


图 7login登录逻辑

可以看见它是经过了和0比较和123456比较，所以猜想需要传输的message以0开头，并带上密码。

然后为了验证这个消息格式打开commandhandler，可以看见存在多个分支，每个分支都先与一个字符比较，所以消息格式基本上就是特定字符标识功能。

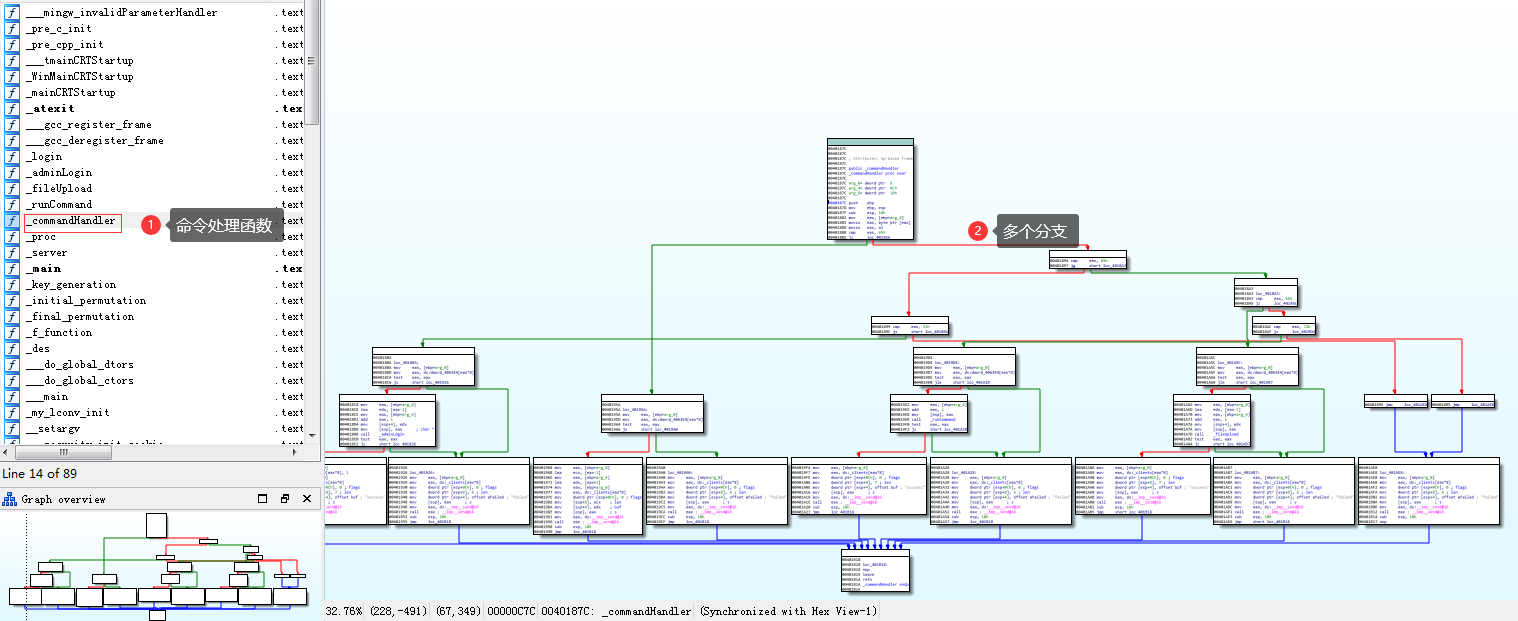


图 8处理函数

一般都是处理函数来处理用户命令，这里可以看见各个函数的传输格式。

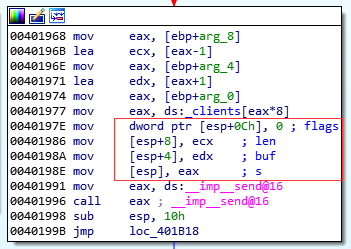


图 9echo函数的处理

根据整个分支，可以看见和e比较后正确的流程如下，该流程将buf重新send回client，故估计为echo命令。

对于后续的分析，可以尝试使用idapro的汇编翻译成c语言功能，方便更好分析，对于文件上传，可以很清楚地看见文件上传的代码反汇编如下：

|  |
| --- |
| **if ( v3 == 102 )**  **{**  **if ( dword\_4063E4[2 \* a1] > 0 && fileUpload(a2 + 1, a3 - 1) )**  **result = \_send\_16(\*(&clients + 2 \* a1), "success", 7, 0);**  **else**  **result = \_send\_16(\*(&clients + 2 \* a1), "failed", 6, 0);**  **}** |

102的ASCII为f，对应于fileupload作为文件上传功能。

转到文件上传功能：

|  |
| --- |
| **for ( i = 0; i < a2 - 1 && (\*(\_BYTE \*)(i + a1) != 13 || \*(\_BYTE \*)(i + 1 + a1) != 10); ++i )**  **v3[i] = \*(\_BYTE \*)(i + a1);**  **if ( a2 - (i + 2) <= 0 )**  **return 0;**  **v4 = fopen(v3, "wb");**  **fwrite((const void \*)(a1 + i + 2), 1u, a2 - (i + 2), v4);**  **fclose(v4);** |

可以看见它还对文件键入了\r\n，因为和13和10相对应。

文件运行同理分析。

然后尝试动态分析堆栈，进行提权处理：

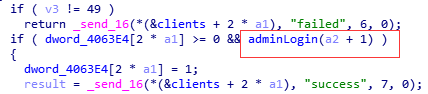


图 10adminLogin作为登录验证

首先可以看见此处对于adminLogin的返回值不是使用==判断，也就是只要堆栈溢出将其覆盖为非0值即可。

然后使用ollydbg进行动态调试分析：

首先在普通用户登录处打断点，可以发现，0+123456确实是普通用户登录凭证：

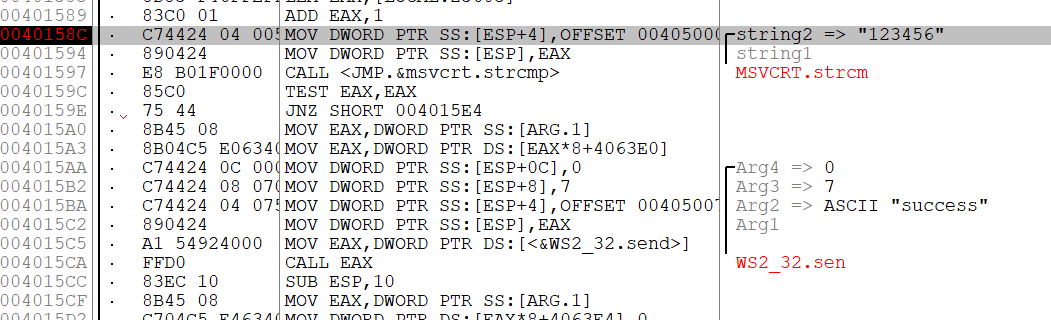


图 11动态调试普通用户登录

然后尝试高级别权限提权：

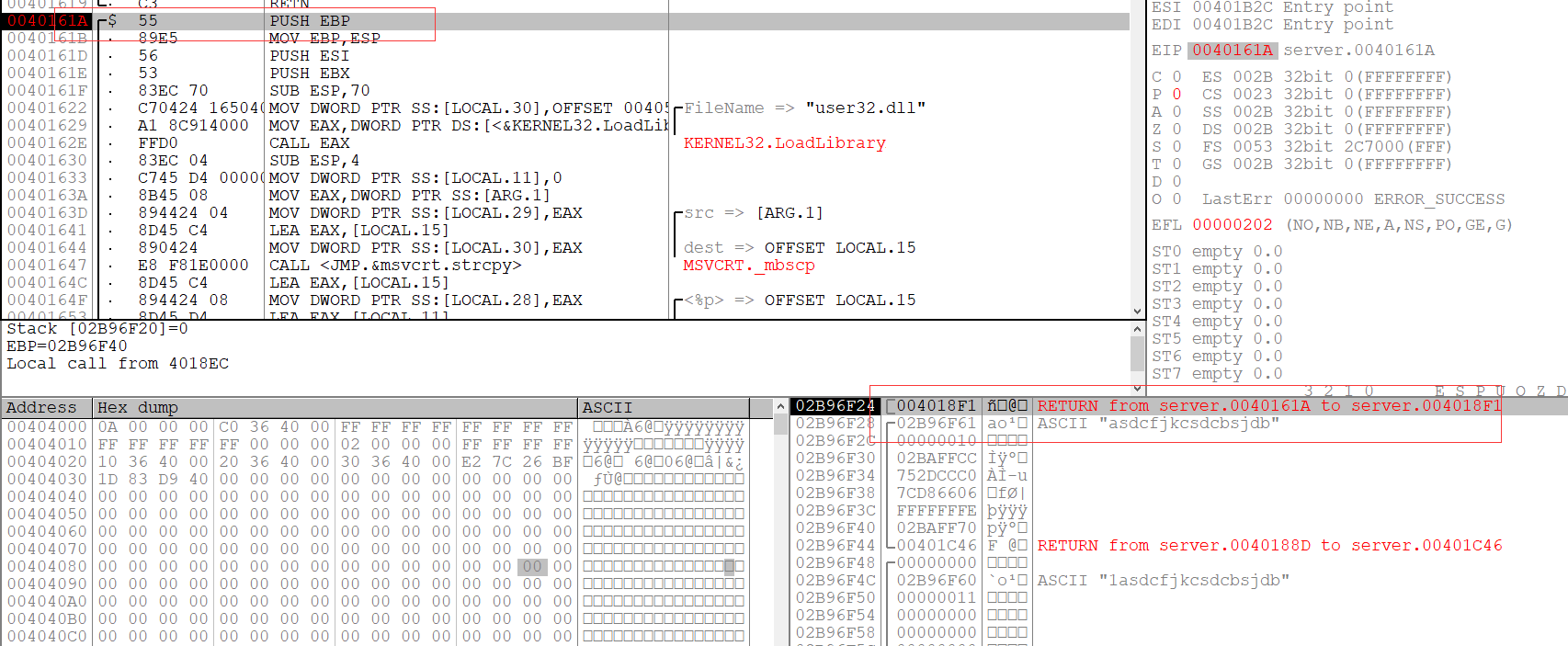


图 12admin函数

可以看见进入admin函数后，参数入栈，参数为输入的文字，加上它的大小。

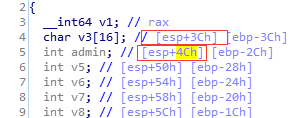


图 13idapro的adminLogin函数

因为返回的v4，故将v4重命名为admin函数，可以看见注释中，admin在栈中位置在16个char数组之后，即在栈中admin的位置更高。此时经过strcpy后栈中情况：

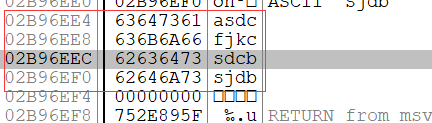


图 14栈中情况

16char即16byte，即2个dword，所以02B96EEC即admin，此时已经覆盖，可以达到提权效果。

由此攻击脚本内容便比较明确。

1. **病毒运行结果**

运行之后doc加密情况如下：

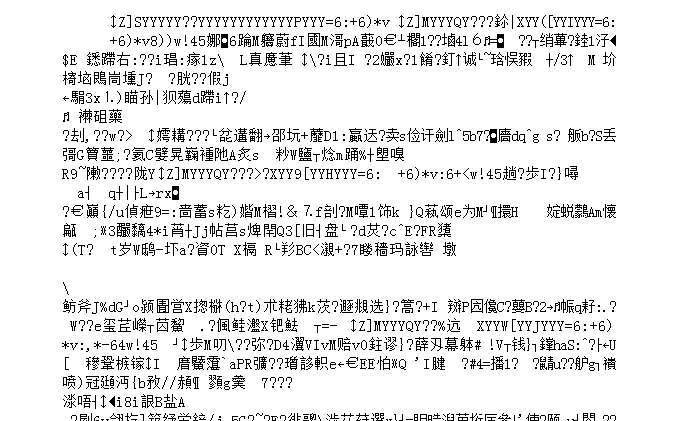


图 15doc文件加密

同时对同文件夹的server.exe加密情况如下：

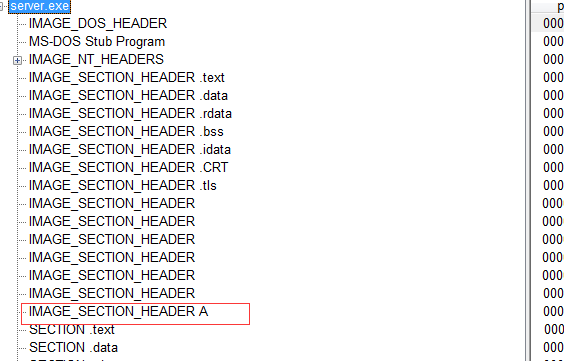


图 16文件感染情况

文件中多了一个节，感染成功。

1. **分工情况说明**

王 浩：整体架构，服务端，客户端，攻击脚本，病毒

李涵威：服务端，客户端，病毒，项目文档

王 勋：客户端，病毒

方香宜：客户端，攻击脚本

1. **参考资料**
2. 《计算机病毒与反病毒技术》，张仁斌等编写，清华大学出版社，2006年6月
3. 《Windows程序设计教程》，杨祥金等编写，清华大学出版社，2008年11月
4. 《0day安全：软件漏洞分析技术》，王清编写，电子工业出版社，第2版，2011.6