**深圳大学实验报告**

**课程名称： 计算机系统(2)**

**实验项目名称： 逆向工程实验**

**学院： 计算机与软件学院**

**专业： 计算机科学与技术**

**指导教师： 马晨琳**

**报告人： 学号： 班级：**

**实验时间： 2025年4月16日～ 2024年4月29日**

**实验报告提交时间： 2024年4月 17日**

**教务处制**

|  |
| --- |
| **一、实验目的：**   1. 理解程序（控制语句、函数、返回值、堆栈结构）是如何运行的 2. 掌握GDB调试工具和objdump反汇编工具 |
| **二、实验环境：**   1. 计算机（Intel CPU） 2. Linux64位操作系统 3. GDB调试工具 4. objdump反汇编工具 |
| **三、实验内容**  本实验设计为一个黑客拆解二进制炸弹的游戏。我们仅给黑客（同学）提供一个二进制可执行文件bomb和主函数所在的源程序bomb.c，不提供每个关卡的源代码。程序运行中有6个关卡（6个phase），每个关卡需要用户输入正确的字符串或数字才能通关，否则会引爆炸弹（打印出一条错误信息，并导致评分下降）！  要求同学运用**GDB调试工具和objdump反汇编工具**，通过分析汇编代码**，**找到在每个phase程序段中，引导程序跳转到“explode\_bomb”程序段的地方，并分析其成功跳转的条件，以此为突破口寻找应该在命令行输入何种字符串来通关。  本实验要求解决Phase\_1、Phase\_2、Phase\_3、Phase\_4、Phase\_5、Phase\_6。通过截图把结果写在实验报告上。 |
| **四、实验步骤和结果**  **第一关**   1. 输入反汇编命令查看汇编代码（保存在1.txt文件中）      1. 打开txt文件，首先找到main函数，发现它调用了从phase1到phase6这六个函数。     再找到phase1，代码如下：    代码大致意思如下：  所以我们需要找到位于0x401af8的字符串，其中的内容就是正确的过关字符串。我们用GDB调试工具查看位于0x401af8的字符串。   1. 利用gdb调试工具来破解Phase1中的神秘字符串     得到答案Science isn't about why, it's about why not?   1. 测试答案，正确，对下一关进行破解。     **第二关**   1. phase2代码如下      1. 函数对输入的六个数字进行累加和比较，判断[rbp]和[rbp+3]的值是否相等，不相等就会爆炸，并且r12d作为一个存贮累加结果的寄存器在这个过程中不能为0。符合这个条件的数字似乎很多，尝试“1 2 3 1 2 3”：      1. 答案通过，进行下一关   **第三关**   1. phase3代码如下      1. 这道题看上去设立了多个通关的方式，其中a的值可以用于选择通关方式，即选择一个目标值的地址，然后程序会比较b的值与选定地址的值，如果一致就能过关。这里我们直接看一下0x401b60处的值，然后输入1（即不跳转）应该就能过关了：      1. 如上图所示，如果输入的 a 导致跳转到 0x400f32，那么 %eax 被设置为 0x217，因此 b 必须为 0x217（535）。我们试一下这个组合：      1. 答案通过，进行下一关   **第四关**   1. phase4代码如下      1. phase\_4`函数主要是解析用户输入的值，并通过调用 func4 函数进行计算。最后，检查计算结果是否等于 55，如果是则通过，否则爆炸。那么我们现在来看下func4的作用。      1. func4其实就是一个简单的将edi里的分别减去1与2然后相加然后返回的功能，经过计算输入值n与输出正好符合斐波那契数列第n项的值。 2. 我们只需要找到斐波那契数列中数值为55的那一项的项数就能过关。答案为9。     **第五关**   1. phase5代码如下      1. 通过分析，我们知道这个函数获取了第一个输入的值后进行上面说明的一系列运算与比较，直到符合条件。之后，会从0x401ba0依次向后取值。也就是说，本题的关键是知道0x401ba0机器之后存了什么东西。用gdb查看      1. 得到这些数就简单了，第二个数只要是每次跳转的目标的累加和就行了。先输入7的话，我们手动在代码里走一遍，得到总和为93。     **第六关**   1. phase6代码如下      1. 查看fun6      1. 通过分析，我们发现func6是一个典型的二叉搜索树的搜索算法，其中：%rdi 寄存器起始包含指向树的根节点的指针，%esi 包含要在树中查找的整数值。函数通过递归地搜索树来查找值，如果找到该值，则返回包含该值的节点的指针；如果没有找到，返回 NULL。 2. 用户输入的数字首先会被转换为一个整数，这个整数是通过 strtol 函数从字符串解析而来。此整数用作 fun6 函数的输入参数，此函数将对这个有序链表执行一系列操作。给定一个整数（通过用户输入），函数遍历一个链表，比较每个节点的值，寻找这个整数。如果找到了，它返回一个指向该节点的指针；如果没找到，它将继续递归遍历。 3. 用户需要输入一个数字，该数字必须是链表中某个节点的值。成功的条件是：输入的数字经处理后返回的节点的下三个节点的值与链表中第一个节点的值相等。调试得到数字的集合，选择600进行输入：     **至此，六关任务均完成** |
| **三、实验总结与体会：**  通过本实验，我对GDB调试工具和objdump反汇编工具的使用能力有了显著提升。能够熟练地使用这些工具来分析程序的运行机制和内部结构，这在未来的程序开发和调试中将非常有帮助。  同时实验使我更深入地理解了程序在底层的运行原理，如函数调用、栈的操作、指令的执行等。特别是通过手动解析二进制文件，我能够更好地理解编译器是如何将高级语言转换为机器能执行的指令的。  每一个关卡的破解都需要我综合运用逻辑思维、分析能力和技术知识。这种挑战加深了我对逻辑结构的理解，并锻炼了面对复杂问题时的应对能力。  通过逆向工程和代码分析，我不仅学会了如何解析程序，也对软件的安全漏洞有了更为深刻的认识。这对未来从事相关工作，如软件安全分析、系统安全等领域，提供了宝贵的实践经验。 |

|  |
| --- |
| **指导教师批阅意见：**  **成绩评定：**  指导教师签字：马晨琳  2024年5月 日 |
| 备注： |