软件漏洞分析实验教学探究

彭碧涛,王常吉,罗海蛟
(广东外语外贸大学网络空间安全学院,广东广州　510006)

[摘　要]针对软件漏洞分析课程的特点,提出了该课程实验建设面临的困难和挑战,阐述了该课程实验建设的思路,然后从常用软件安全工具、栈溢出和堆溢出、Shellcode 编写、其他内存攻击技术、其他类型软件漏洞、Windows 安全机制、漏洞挖掘以及 CVE 分析和利用等知识点设计了实验内容,最后对整个实验教学内容进行了总结,指出后续的改进和建设方向。

[关键词]软件漏洞;实验教学;漏洞分析;软件安全

[作者简介]彭碧涛(1978—),男,湖北天门人,广东外语外贸大学网络空间安全学院副教授,博士,主要从事软件漏洞分析与挖掘、恶意代码检测研究。

[原文出处]《湖北开放职业学院学报》2021(20)
[本刊网址]http://www.oacj.net

随着计算机软件的目标客户越来越广泛,需求越来越庞杂,使得软件的复杂性大幅增加,被披露的安全漏洞数量也呈现逐年增长的趋势。然而软件会因为自身安全漏洞而遭受外界利用攻击,导致隐私泄露、非法提权和勒索等严重不安全后果。软件安全事件不仅会对个人、企业造成影响,严重的软件漏洞甚至会对国家安全造成威胁。因此培养掌握软件漏洞分析和挖掘等相关技术的安全专业人才,提高相应的软件安全开发能力和软件漏洞处理能力,对于国民经济安全和国家安全都至关重要。

目前国内本科阶段开设软件漏洞分析相关课程的高校包括武汉大学、中国地质大学、南京邮电大学、北京邮电大学和暨南大学等,课程为“软件安全”“软件系统安全”和“软件漏洞分析与防范”等;还有一些高校如华中科技大学、中国科学院大学等在研究生阶段开设“漏洞分析”“软件安全与测试”和“软件安全漏洞分析与发现”等课程。但是整体而言,相比较密码学、Web 攻防等相等成熟的信息安全课程,软件安全类课程还属于新兴的课程,软件漏洞分析作为软件安全的核心部分,其教学内容、教学方式、实验体系等都还处于探索阶段。

软件漏洞分析涉及汇编语言、软件开发、软件逆向、Web 开发、网络通信协议、操作系统内核、程序静态分析和动态分析方法等多方面的知识,对计算机理论基础、程序开发能力和系统分析能力等都有一定的要求;这类课程对学生基础要求高,建设难度大,实践性很强。软件漏洞分析课程的体系内容,不仅要学习相关的理论知识,同时需要通过系统的实验来对培养学生的实践动手能力。如何设计软件漏洞分析实验教学环节,通过实验教学来高学生对于软件漏洞工作机理的理解、软件漏洞的利用、软件漏洞的挖掘能力,对于提升学生在现实工作中网络空间安全实际工程能力至关重要。

一、软件漏洞分析实验教学建设的挑战

软件漏洞分析需要从二进制和汇编语言的形式,对程序的运行机制和逻辑进行分析;需要通过对程序进行静态和动态执行分析,并且结合二者分析的结果来实现对漏洞机理的理解;在软件漏洞利用上,更是需要对操作系统下程序运行机理非常清楚;多维度、多层次和正逆思维结合导致软件漏洞分析相对复杂,对学生各方面能力水平要求较高,对设计合理的软件漏洞分析实验教学提出了很大的挑战:

(一)多种语言编程能力的挑战

软件漏洞分析需要掌握高级编程语言 C++、Java、Python、Ruby 和 JavaScript 等;C++和 Java 一般是软件的开发语言,使用 Python 来编写漏洞的挖掘、漏洞的利用代码,使用 Ruby 在 Metasploit Frame

中发布漏洞信息；同时需要掌握汇编语言，要对软件逆向的各种汇编结构非常熟悉，对指针、寄存器、堆栈，程序的各种结构精通，能够将高级语言代码与汇编代码灵活转换；在涉及堆喷射相关技术实验时，还需要对 JavaScript 语言有一定的掌握。

（二）多种软件漏洞分析工具的挑战

软件漏洞分析是一门实践性很强的课程，需要掌握大量的信息安全相关分析工具，以辅助完成软件漏洞的分析、挖掘和利用。常用的软件漏洞相关分析工具包括：PE 分析相关软件、二进制编辑软件、格式化文件解析软件、用户态和系统态调试软件、网络流量抓包和分析软件、网络安全渗透测试平台、补丁比较软件、和基本模糊测试漏洞挖掘软件等。

（三）编程思维方式的挑战

软件漏洞分析与利用的代码编写的思维方式与使用 Java 或 C++开发软件系统不相同，开发软件系统是一种正向思维，但是软件漏洞分析与利用的代码编写需要软件的逆向思维以及技巧，是一项极其细致、难度极高的工作，需要长期的训练和经验。使用 Java 和 C++开发系统可能一天可以写成百上千行代码，但是软件漏洞分析与利用的代码可能一周也写不出几百个字节。

（四）学习综合能力的挑战

软件漏洞分析与利用涉及多种安全工具的使用、多种程序开发语言的融合、多种软件系统层次的静态或动态调试、以及本地与网络多维度的分析，同时需要从二进制层面进行开发和调试，在其中会碰到很多问题，需要学生具有较好的学习综合能力。当碰到某一方面的问题时，需要学生能够迅速地查找资料、结合问题寻找到解决方案。

二、软件漏洞分析实验建设思路

针对软件漏洞分析课程的授课内容，以及其教与学面对的挑战，在建设软件漏洞分析课程的实验部分时，遵循以下思路：

（一）漏洞分析和漏洞利用相结合的思路

在软件漏洞分析学习每一个知识点的基础阶段，通过编写代码，设置典型的漏洞呈现环境，来快速的理解对应漏洞的原理和机制；在此基础上，构建漏洞利用代码，实现对该漏洞的有效利用，展示对系统的攻击。这样一方面可以加深对漏洞的理解，另一方面也可以提高学生的学习热情和兴趣。

（二）立足基础和与时俱进相结合的思路

在实验体系上，从最基础的栈溢出、堆溢出原理和利用入手，从基础的 Shellcode 编写实验开始，让学生打好扎实的软件漏洞基础；同时引导学生接触前沿，通过了解最新的 CVE、查阅相关的安全论坛信息来接触最新的漏洞安全技术。

（三）以点带面和层层递进相结合的思想

将实验教学体系分为一个一个相互承接递进的知识点，从最基础的堆栈溢出与利用入手，理解漏洞的原理；然后学习常用的 Shellcode 编写、MSF 框架和 Windows 安全框架；接着引入 Windows 下的内存攻击技巧、其他类型的常见漏洞和 windows 的安全机制；接下来学习简单的漏洞挖掘技术；最后通过真实的 CVE 来掌握整个知识体系。在每一个知识点设计一个典型的实验，同时引导学生自主拓展。

（四）教师指导和主动探索相结合的思路

软件漏洞分析实验部分的内容很多，而且每一类实验要完成和掌握也需要大量的时间，仅仅依靠课堂上的实验教学时间来完成和掌握实验所包含的知识是非常困难的，因此需要学生在教师的指导下，通过课前预习、课程练习、课后查找资料，主动探索来完成实验。

三、软件漏洞分析实验设计

（一）常用软件安全工具实验

PE 工具 PE View、Stud_PE 和 PE Explore，二进制编辑软件 UltraEdit，用户态调试软件 Ollydbg 和强大的软件分析工具 IDA Pro 等是软件漏洞分析的基础工具软件，会贯穿整个软件漏洞的课程，因此设计实验 1 来掌握这些软件的使用。实验 1 构造一个密码验证的程序，通过 PE 工具分析程序的结构，通过 Ollydbg 和 IDA pro 分析程序的执行流程，通过 Ollydbg 观察内存中密码验证的机制，然后设计多种修改程序的验证逻辑的方式，使用 UltraEdit 修改程序二进制代码来绕过密码验证。

（二）栈溢出和堆溢出实验

实验 2 构造一个存在栈溢出漏洞的程序，在子函数的形参使用字符指针，同时对字符串的长度不

加以验证以形成典型的栈溢出漏洞;通过输入不同的特定字符串,导致栈溢出,分别实现修改函数的返回值和栈顶的返回指令指针,实现程序流程的劫持;在此基础上,植入 Shellcode 代码,程序被劫持后转向执行植入的 Shellcode 代码。

实验 3 构造一个堆溢出漏洞的程序,首先创建一个堆,在堆上分配缓冲节点,然后复制特定的 Shellcode 实现堆的溢出,当再次分配缓存节点时,导致 DWORD SHOOT,实现程序流程的劫持,从而执行 Shellcode 代码。

(三)Shellcode 编写实验

实验 4 首先学习使用 JMP ESP 技巧来克服栈帧移位导致的 Shellcode 动态变化的问题。在内存中使用任意一个 JMP ESP 的指令地址覆盖函数的返回地址,当函数返回后,被重定位去执行 JMP ESP 指令;在构造栈溢出的字符串时,将 Shellcode 存放在返回地址之后,这样,程序流程被劫持后,就恰好跳入 Shellcode 执行。在此基础上,实现 Shellcode 加壳和解壳:首先对原始的 Shellcode 进行简单的异或加壳编码,同时开发解码器,将解码器与加壳后的 Shellcode 结合在一起作为最终的 Shellcode 来完成实验。

(四)其他内存攻击技术实验

实验 5 第一部分设计栈缓冲区溢出被攻击的例子,但是漏洞利用的是 Windows 的异常处理机制。首先构造缓冲区溢出的字符串,该溢出会覆盖程序的异常处理机制;然后在程序中构造除数为 0 的异常,导致会进入异常处理;当程序异常发生时,会使用被覆盖的异常处理机制,从而实现了程序流程的被劫持,转而执行 Shellcode。实验 5 的第二部分攻击 C++的虚函数,通过构造溢出的字符串,对虚表指针和虚函数指针实现修改,当程序调用其虚函数时,实现对程序流程的劫持,最终跳转执行 Shellcode。

(五)其他类型软件漏洞实验

实验 6 首先通过简单实验,理解格式化字符串漏洞产生的原理;然后设计嵌入 Shellcode 的带格式化的字符串参数,当格式化字符串执行时,通过“%n”的格式化控制符实现用 Shellcode 地址覆盖返回地址,从而实现程序的劫持;在格式化字符串中,通过填充不同个数的“%x”控制符来偏移被写地址到字符串的结尾,通过“%[n]x”控制打印字符的个数,实现让被写数据指向 Shellcode。

实验 7 首先设计实验演示整数溢出的原理,通过将用户输入的 int 型变量赋值给无符号短整型的变量实现整数的溢出;接下来具体设计了 3 类可能导致安全漏洞的整数溢出例子:无符号数的下溢和上溢,符号问题和高位数复制到低位数时的截断问题;最后设计一个由于整数溢出引发栈溢出,从而导致程序流程被劫持,跳转执行 Shellcode 的实验。

(六)Windows 安全机制实验

微软在 Windows 系统中增加了很多提高内存保护的安全技术,包括 GS 编译技术、异常处理结构化的安全机制、数据执行保护、加载地址随机化等,这些安全机制在教学课堂上都会有讲授,但是由于教学时间限制,在实验设计中主要针对数据执行保护和加载地址随机化保护设计了相应的实验 8 和实验 9。

实验 8 设计实验利用 ROP 机制来实现对数据执行保护的破解和攻击,实验通过查找 ROP 指令,将其嵌入到 Shellcode 中,通过这写指令调用 ZwSetInformationProcess 函数来实现对系统数据执行保护的关闭;关闭数据执行保护后,再跳转进入 Shellcode 执行攻击代码。当然数据执行保护还有其他的绕过方法,在课堂上提供了思路,让学生课后去探索。

实验 9 设计首先设计简单的实验理解加载地址随机化;然后结合堆喷射技术:首先在内存中申请大量的空间,然后在每个空间都包含 0x90 和 Shellcode 代码,最后设计一个典型的溢出漏洞,将函数的返回地址设计为 0x0C0C0C0C,在函数执行返回时就会转入实现申请的内存空间,从而绕过了加载地址随机化执行攻击代码。

(七)漏洞挖掘实验

实验 10 漏洞挖掘实验主要利用模糊测试技术来实现漏洞挖掘。通过使用 Python 构造模糊测试的字符串,网络发送到存在漏洞的 FTP 服务器,同时使用 Wireshark 和科来网络分析系统对网络流量数据进行监控分析,当出现服务器崩溃时,分析此时的测试字符串,然后定位漏洞,最后使用 MSF 生成攻击代码,对 FTP 服务器进行攻击。

(八)CVE 分析和利用实验

实验 11 是一个具体的 CVE-2011-0104 进行分析和利用的实验。CVE-2011-0104 时在解析 XLB 文件中的 TOOLBARDEF Record 时存在的栈溢出漏洞,可导致任意代码执行。如果构造恶意的 XLB 文件,当打开该文件时,主机就会被完全控制。实验首先对漏洞进行了重现,然后对漏洞原理进行分析,最后使用 msf 生成漏洞攻击脚本,成功弹出计算器,实现了漏洞的利用。

最后课程大作业要求完成一个近两年的 CVE 漏洞分析、重现和成功利用的实验报告。学生通过上网检索文献、查找网络安全论坛和安全博客等,来搭建实验环境,完成实验内容。在实验过程中,学生会综合使用之前学到的漏洞分析和利用知识、会碰到很多问题;最后课程大作业的完成,既是对课程知识的汇总,更是培养了学生自我钻研、自我探索的精神。

四、结语

在软件漏洞实验中,每一个实验的成功完成都需要大量的时间,仅仅依靠实验课堂的时间来完成是远远不够的,这就要求学生需要课前预习,实验课堂上认真操练,课后继续花时间去调试去完成实验,对学生的自主钻研有比较高的要求;同时在实验完成过程中,会碰到很多问题,需要学生相互交流,去查询相关的专业安全网站,自己去寻找问题的解决方案。

软件漏洞实验课程从简单到复杂,逐步深入探寻 Windows 平台下的软件分析和漏洞技术,通过完成实验内容,学生会对漏洞的分析与利用技术有一定的了解,打下比较好的基础。但是由于软件漏洞实验课程还处于探索中,还存在很多内容有待建设,如详细的实验演示课件、有效的实验完成评估手段和合适的实验内容选取等。接下来我们将继续加强该门课程的建设,实现更有效的软件漏洞分析教学。

参考文献:

[1]李韵,黄辰林,王中锋. 基于机器学习的软件漏洞挖掘方法综述[J]. 软件学报,2020,31(7).

[2]王林章,陈恺,王戟. 软件安全漏洞检测专题前言[J]. 软件学报,2019,29(5).

[3]曹晓梅,朱枫,沙乐天. 软件漏洞分析与防范课程建设[J]. 软件导刊,2018,17(3).

[4]彭国军,张沪寅,傅建明. 软件安全精品课程的攻防实践探索[J]. 计算机教育,2020(8).

Research on Experiment Teaching of Software Vulnerability Analysis

PENG Bi-tao, WANG Chang-ji, LUO Hai-jiao

(School of Cyber Security, Guangdong University of Foreign Studies, Guangzhou Guangdong 510006, China)

Abstract: According to the characteristics of the software vulnerability analysis, the construction difficulties and challenges of the course experiment are analyzed, construction guiding idea is proposed, and the experiment content is designed which includes the commonly used software security tool, stack overflow, heap overflow, Shellcode writing, other memory exploitation, other types of software vulnerability, Windows security mechanism, and CVE analysis and exploitation. At last, it summarizes the whole experimental teaching content, and points out that the improvement and direction of future construction.

Key words: software vulnerability; experiment teaching; vulnerability analysis; software security