缓冲区溢出实验

严文励

实验原理：

CCProxy6.2存在缓冲区溢出漏洞，超出缓冲区大小的内容将被写入到寄存器中。而程序要执行的下一条语句的跳转地址是储存在eip寄存器中的，之后执行的命令地址在esp寄存器中，也会被覆盖。因此，可以构造出一条输入来操纵内存，使系统执行我们想要它执行的内容。

环境配置：

安装windows XP虚拟机，我安装的是sp3的虚拟机，安装完成后打开C:\boot.ini文件，修改其中的” /noexecute=optin”为”/execute”，安装实验材料中提供给CCProxy6.2和dbg\_x86\_6.11.1.404，重启系统。

实验步骤：

(1)首先打开CCProxy，点击“设置”，去掉下方的“自动检测”查看本机局域网IP，本机为192.168.118.129，再勾选回“自动检测”。

(2)在不关闭CCProxy的前提下，打开cmd，输入命令telnet 192.168.118.129 23，成功连接到CCProxy Telnet

(3)在telnet中输入”ping “ + 2000个’A’，CCProxy报错且telnet失去和主机的连接。

(4)重复(2)的步骤，然后再打开一个cmd，输入命令"C:\Program Files\Debugging Tools for Windows (x86)\cdb.exe" -pn ccproxy.exe，其中C:\Program Files\是dbg\_x86\_6.11.1.404安装的目录。然后再telnet中输入”ping “ + 2000个’A’，然后在cdb中输入命令g，发现eip的值为41414141，即”AAAA”。

(5)写一个程序构造一个可以用于定位缓冲区长度的文本pattern.txt，我使用C语言实现，相关代码见附件中严文励\_202128015029011\_EX2A\_src.c。

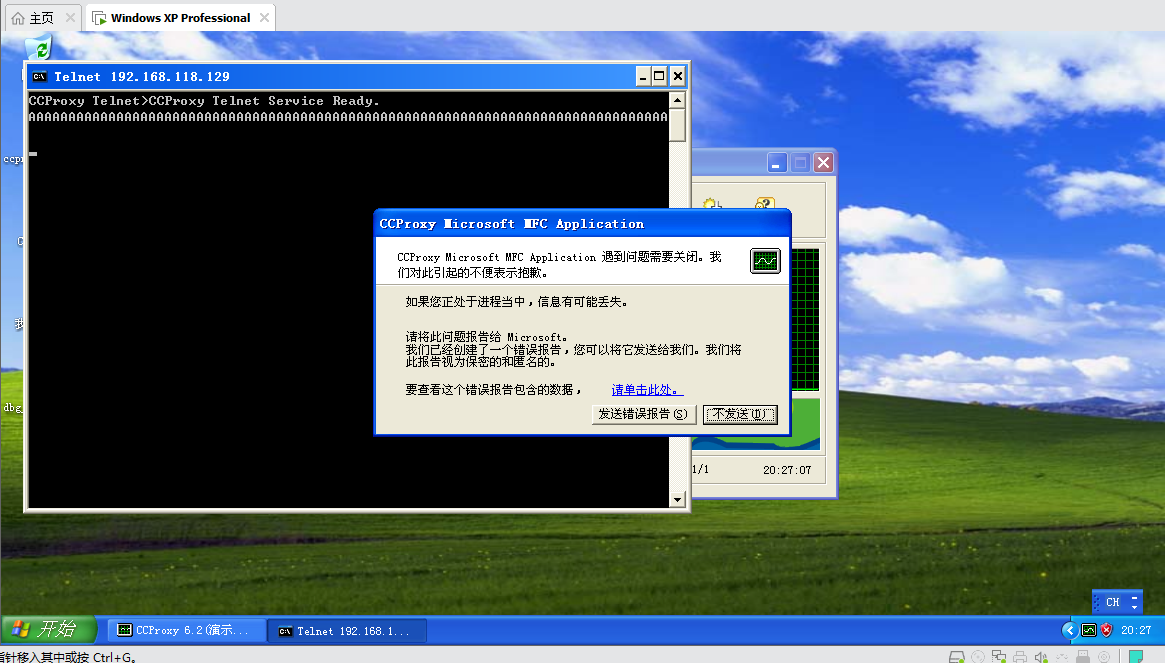
(6)重复(4)中的步骤，并将ping的内容改成pattern.txt中的内容，发现eip的值是68423768，即”hB7h”，考虑大小端的问题实际为”h7Bh”，使用文本搜索和字数统计功能得到”h7Bh”在pattern中的偏移为1012字节。

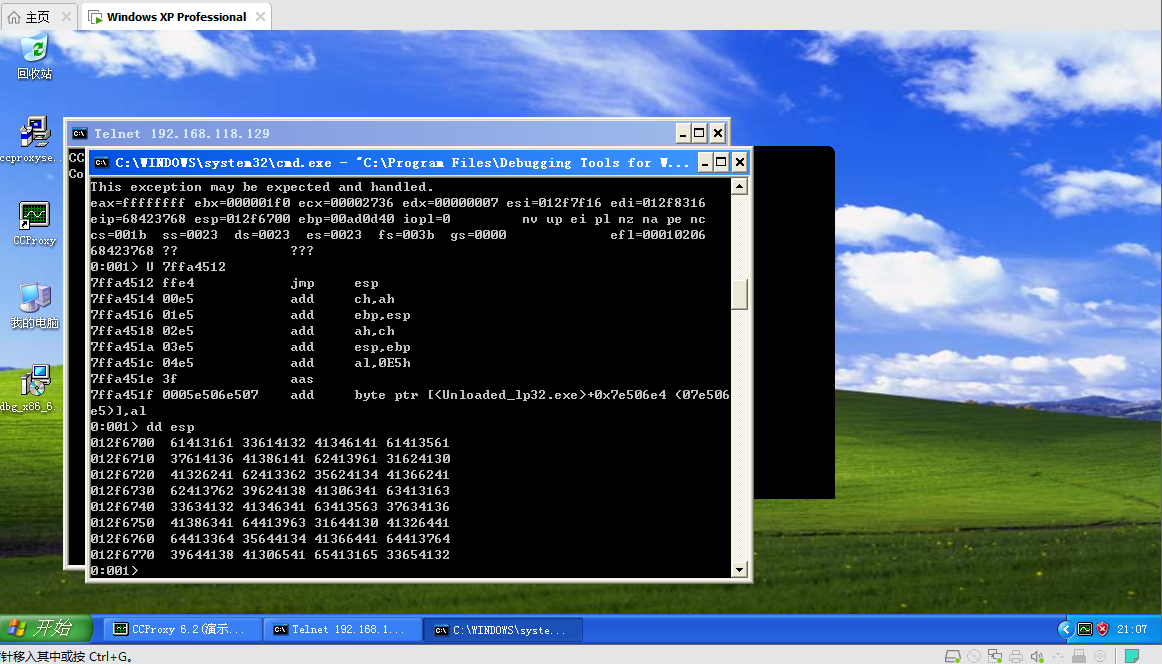
(7)在cdb中输入命令U 7ffa4512，确认此处指令是jmp esp，然后输入命定dd esp，得到esp的内容是61413161，即”aA1a”，用文本搜索华人字数统计功能得到”a1Aa”在pattern中的偏移为4字节。

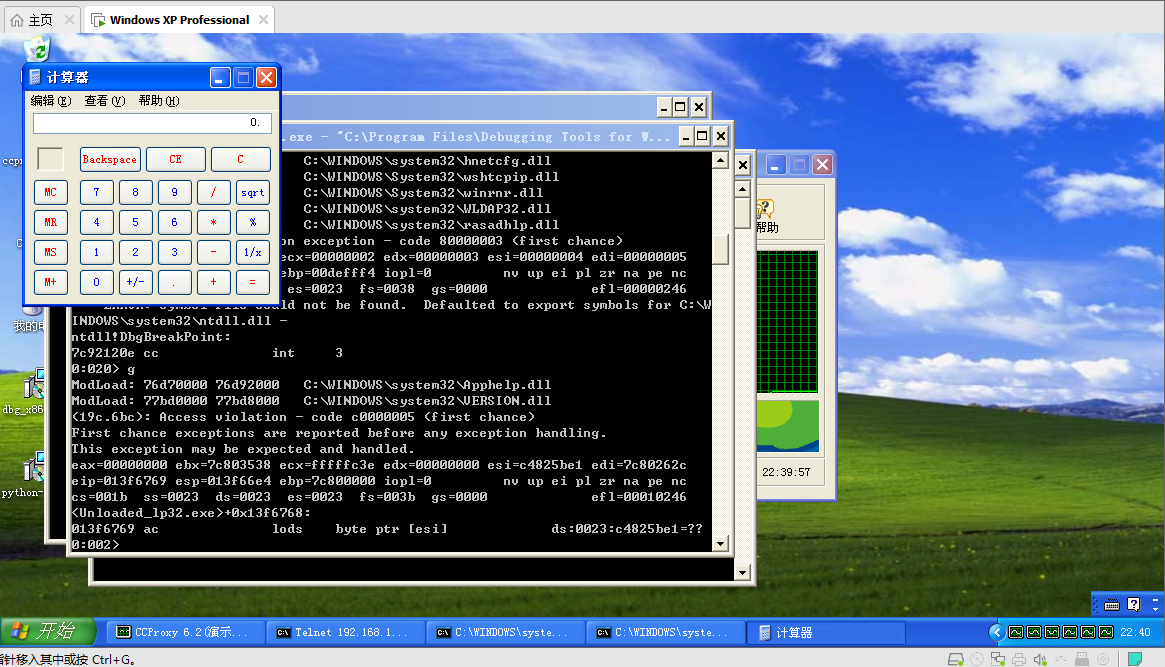
(8)由此确定了exploit代码的结构为：4个Nop，然后是shellcode，然后填充Nop至第1008字节，然后是ebp寄存器的值，然后是esp的地址。选用打开计算器的shellcode代码构造exploit代码。

(9)在虚拟机上用python socket连接进行攻击。相关代码见附件中严文励\_202128015029011\_EX2A\_src.py

实验结果：成功打开了计算器。见截图和视频。







难点与总结：

由于有较详细的讲解，发现漏洞的过程并不难。但在如何发送ping的内容时还是走了不少弯路。开始试过从主机连接虚拟机进行攻击，但socket连接不上。后来在虚拟机上用C进行socket编程，也比较麻烦。最后在虚拟机上安装了python完成了攻击。

如何防范此类漏洞：在设计软件时要注意缓冲区的边界。发布软件时经过针对此类问题的大量测试，或者进行过缓冲区安全的形式化验证。