**LAB5 Part3**

**Task1: Get Familiar with the Shellcode**

**关闭地址空间随机化**

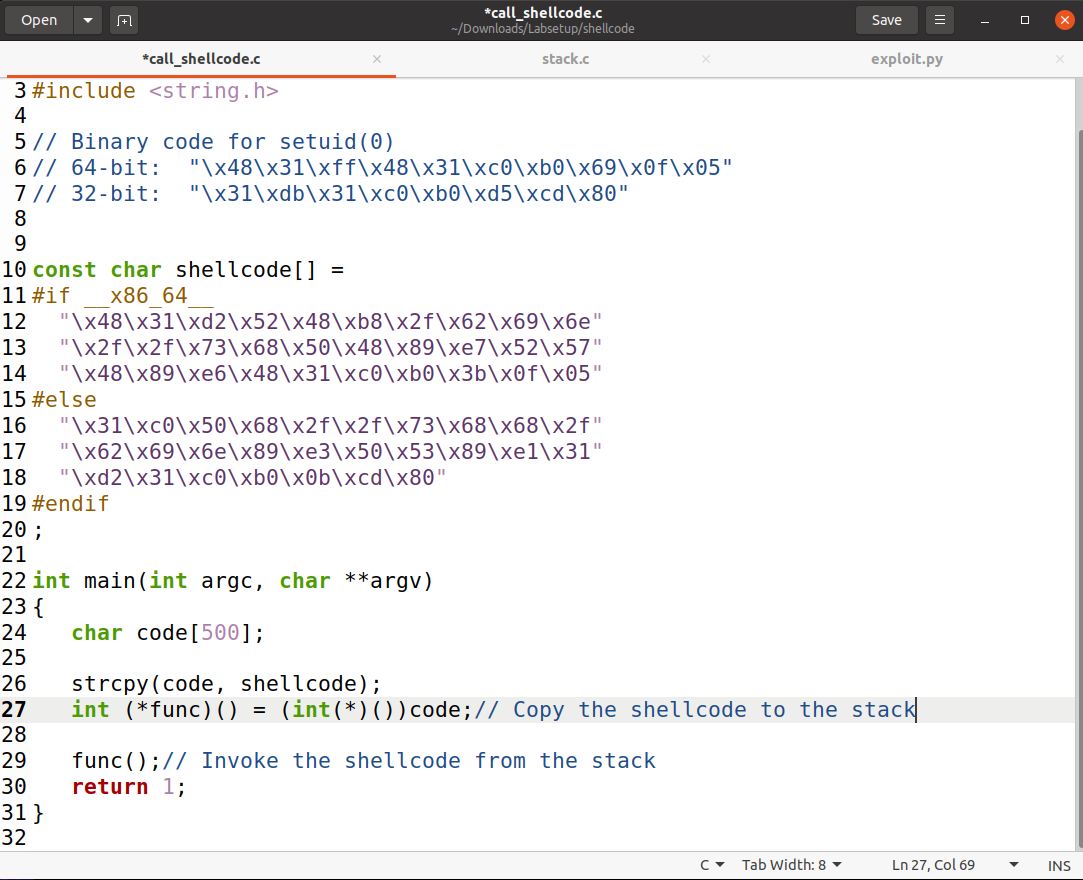
系统会随机化堆和堆栈的起始地址。猜测地址是缓冲溢出攻击的关键步骤之一。可以使用以下命令丢弃此功能：

sudo sysctl -w kernel.randomize\_va\_space

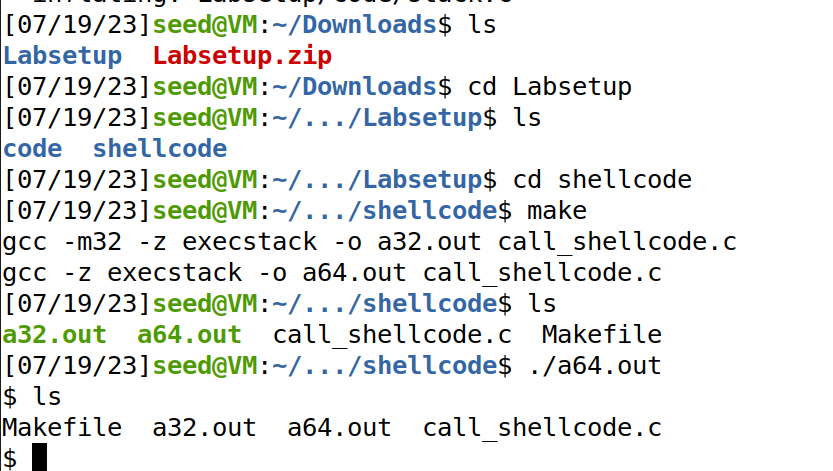
因为现在/bin/sh符号链接指向/bin/dash，而在第一次实验中，我们知道dash对set-UID程序有个安全策略，放弃特权。所以我们需要改变符号链接到/bin/zsh，使用如下命令：

|  |  |
| --- | --- |
| 1 | sudo ln -sf /bin/zsh /bin/sh |

Task: Invoking the Shellcode源码如下：

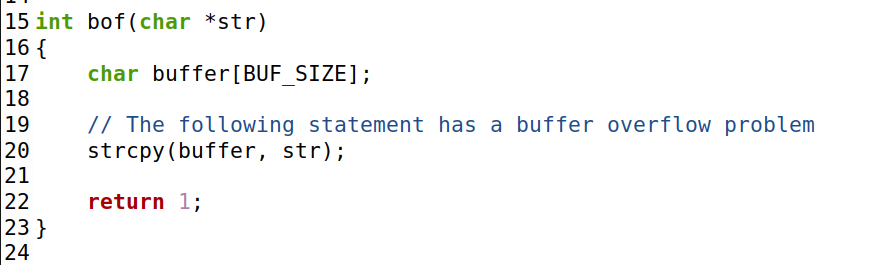
****

因为已经提供了Makefile文件，所以可以很简单的实现

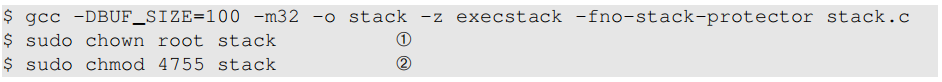
****

**Task 2: Understanding the Vulnerable Program**

从源码审计知道：漏洞出在bof函数中的strcpy(buffer, str)函数。

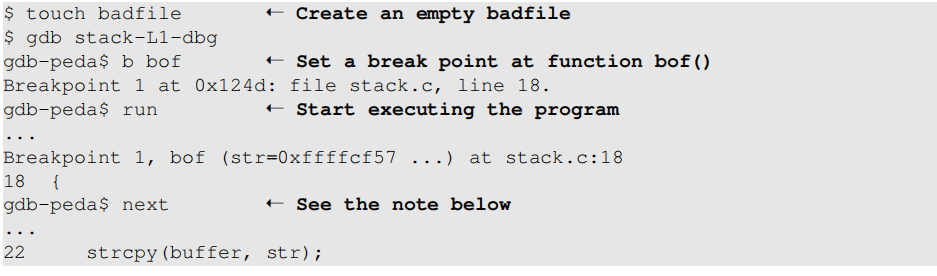


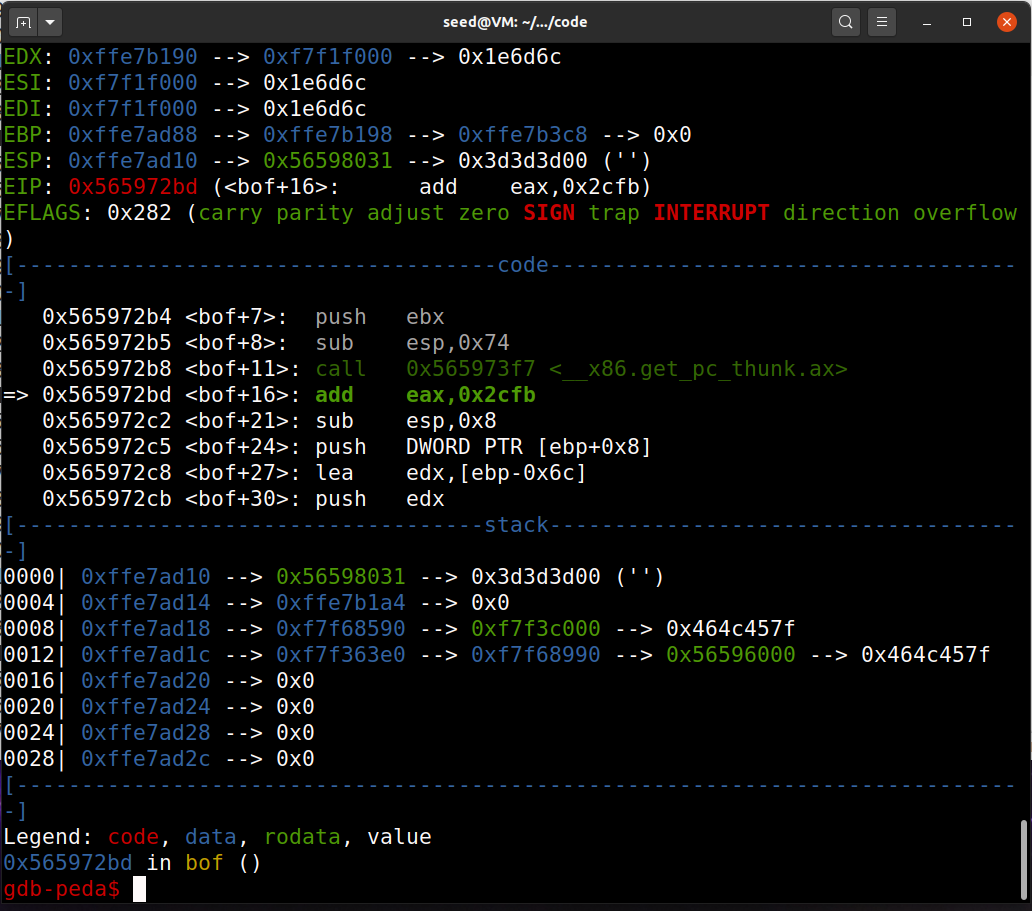
我们需要在badfile文件中填入payload造成缓冲区溢出漏洞，具体命令行已在官方实验pdf文件中给出

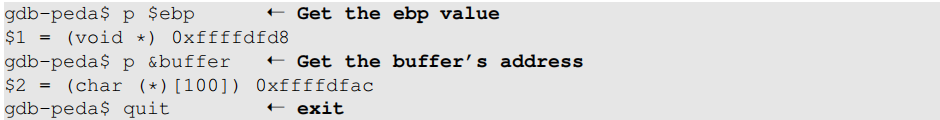


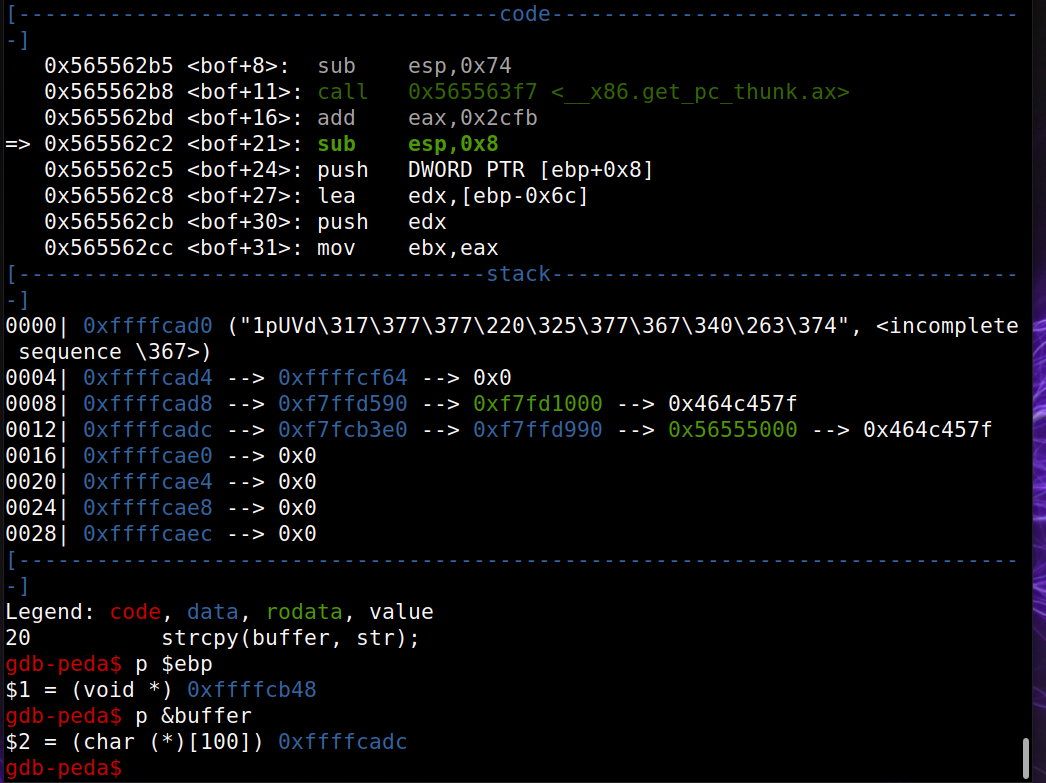
**Task 3: Launching Attack on 32-bit Program**

按如下步骤发动缓冲区溢出攻击：

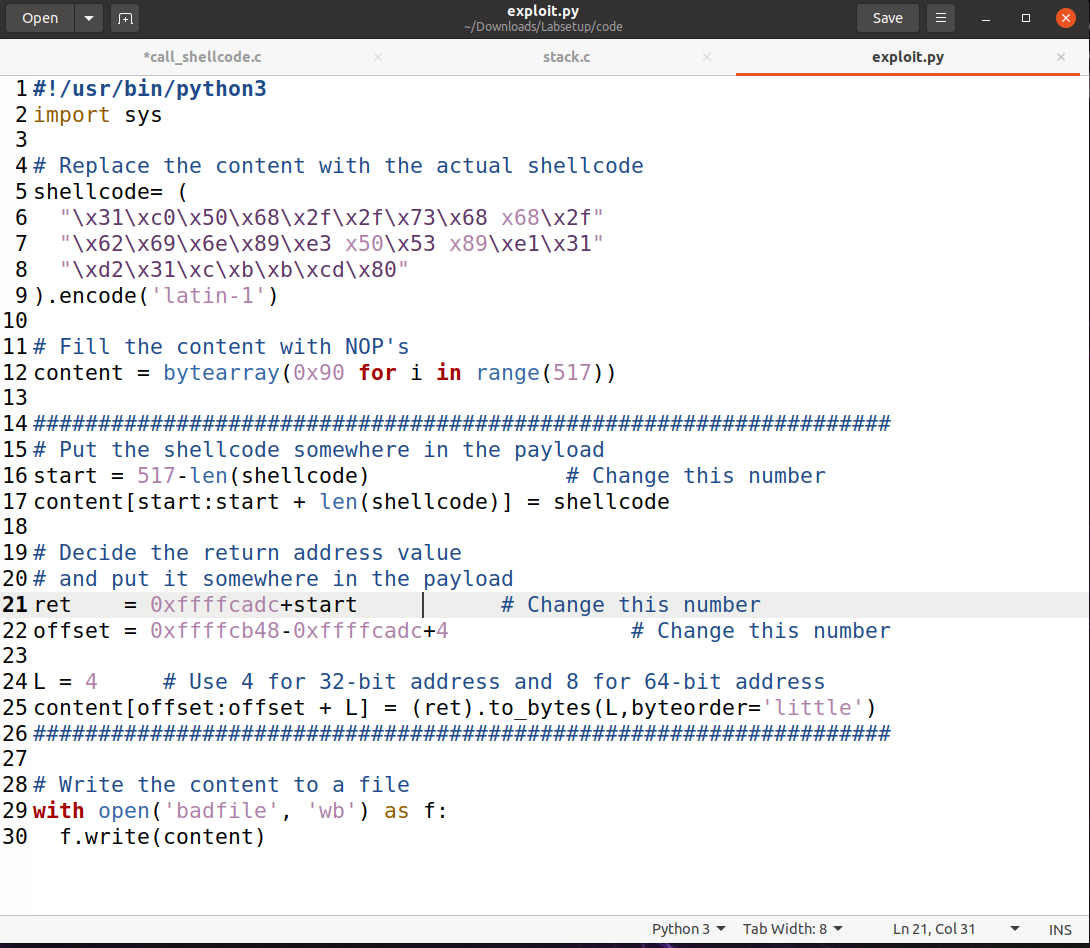


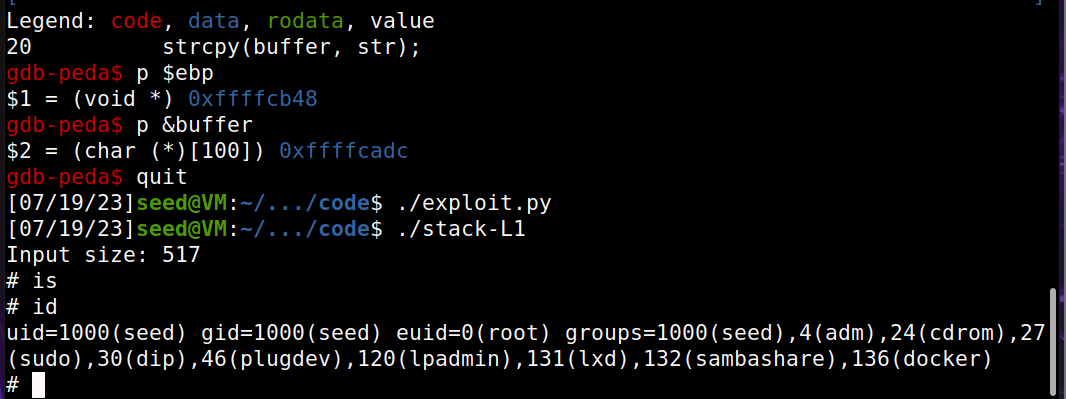






修改exploit.py文件



生成payload并攻击  


可以看到我们成功拿到root权限