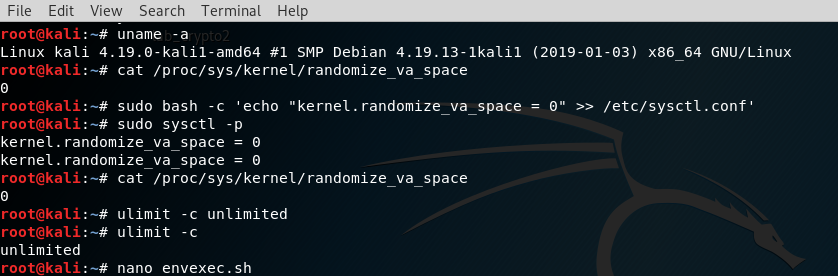
When more data is put into a fixed-sized buffer than the buffer is capable for is called buffer overflow. The extra information or data which overflowed has to go somewhere, example - into adjacent memory space, corrupting or overwriting the data held in that space. Those overflowed information or data is vulnerable which can be attacked that abuses a type of bug called a “buffer overflow”, in which a program overwrites memory adjacent to a buffer that should not have been modified intentionally or unintentionally. It causes a system can be crashed which lead to cyber-attack.

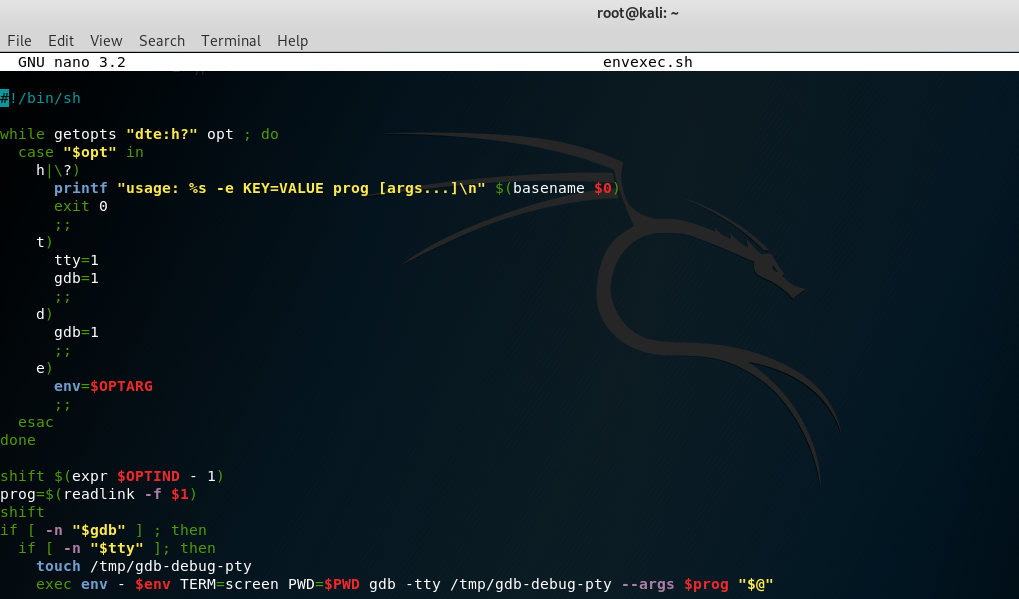
It is important to understand what buffer overflows are and how they pose to attack applications, and what techniques are used to exploit these vulnerabilities effective way.

I exploited on kali Linux operating system and how I did my lab is below.

I disabled memory randomization and enabled core dumps and verified that it is “0” and is unlimited.



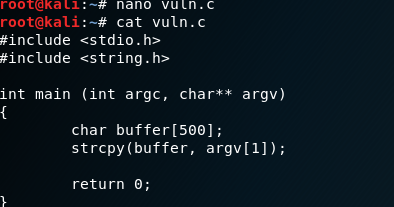
I wrote a envexec.sh script in bash typing command “nano” I copied a code from <https://gist.github.com/apolloclark/6cffb33f179cc9162d0a>







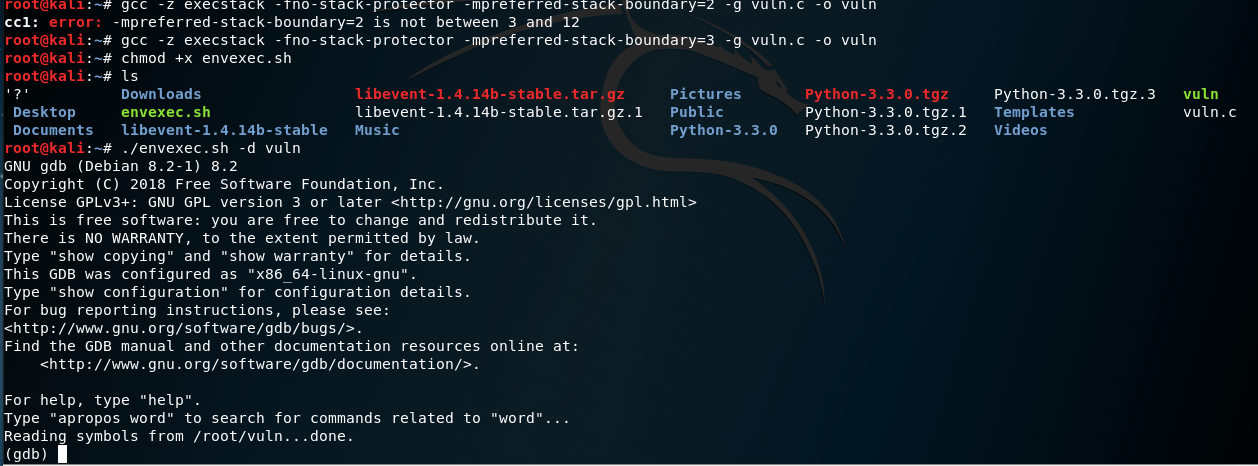
sameway I created another vulnerable code file named vuln.c





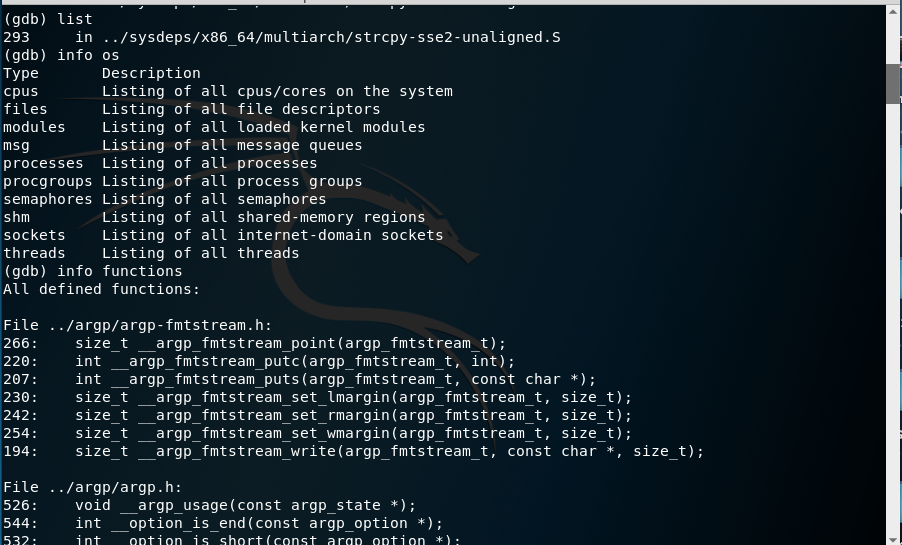
then I compiled the code and debugged

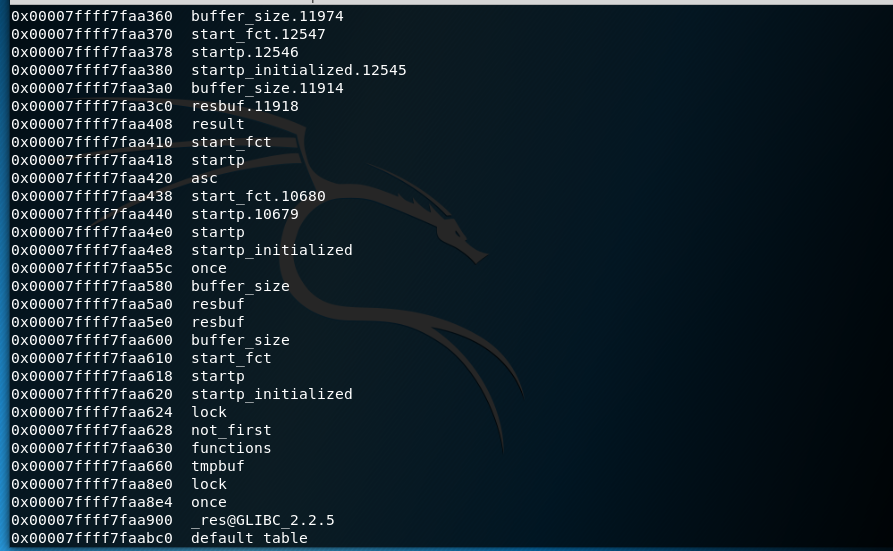
and run gbd to laod the program to analyze it.

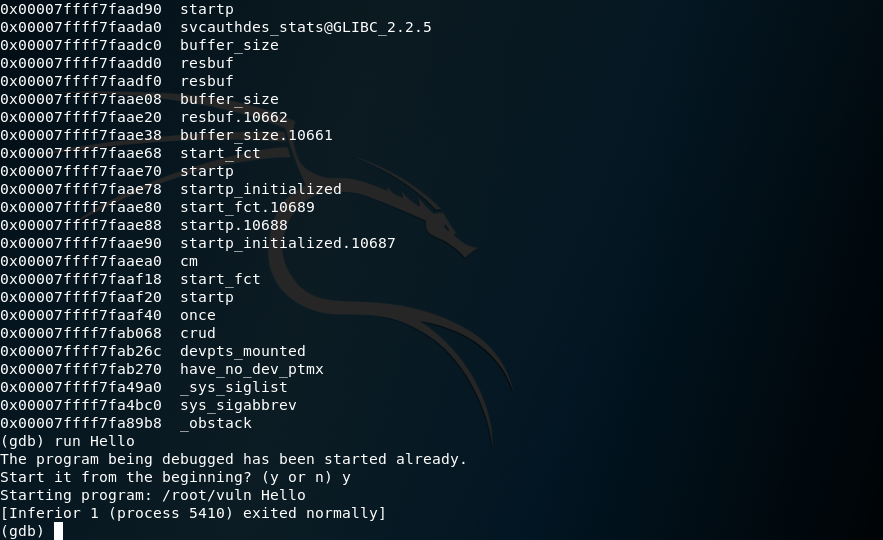


gdb commands

To get information I typed “list” , “disas main”, ” info os” ,“info functions” ,”info variables”







I Run the program with input hello.



I was able to write the return address and code executed. I have checked the rip register.