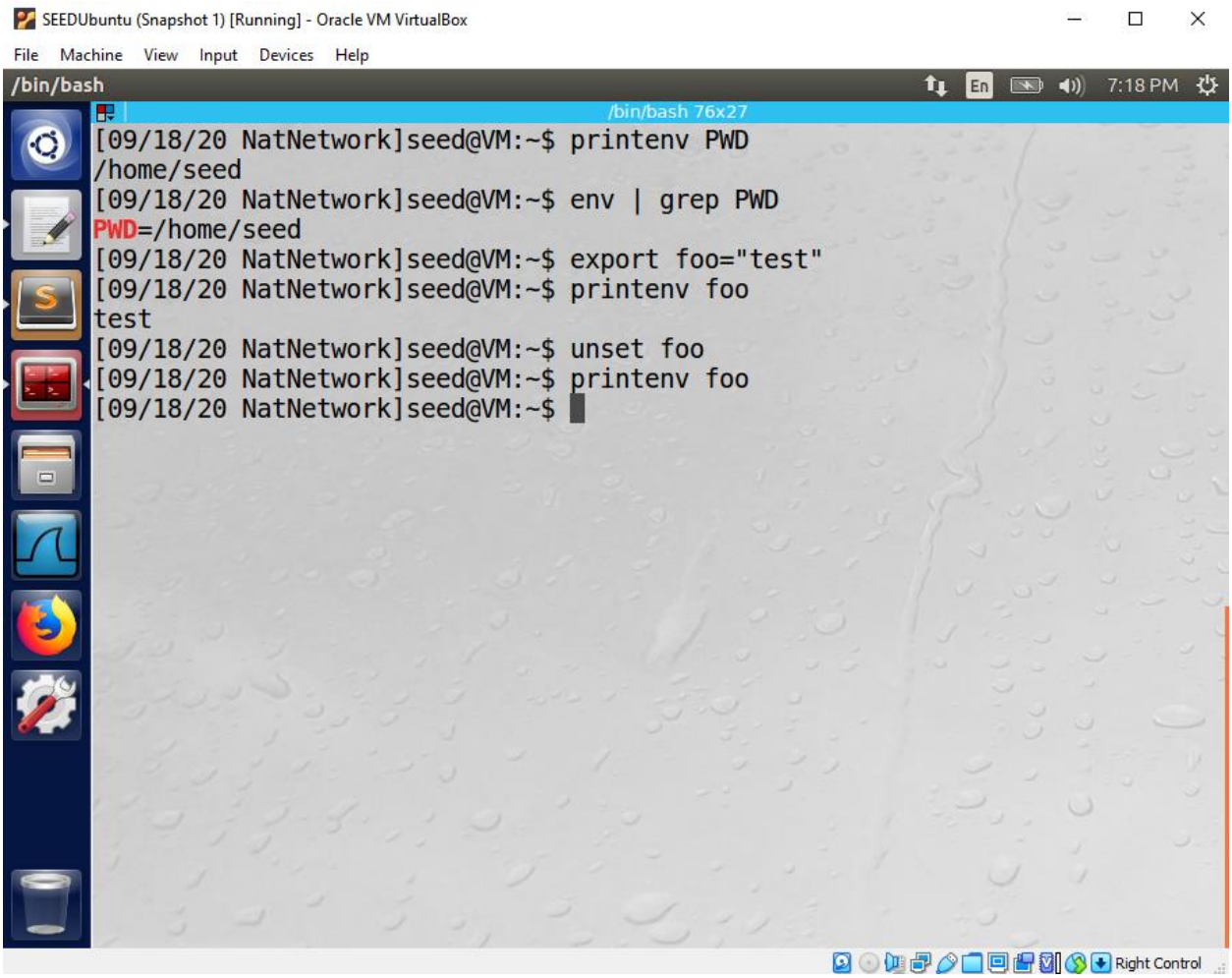
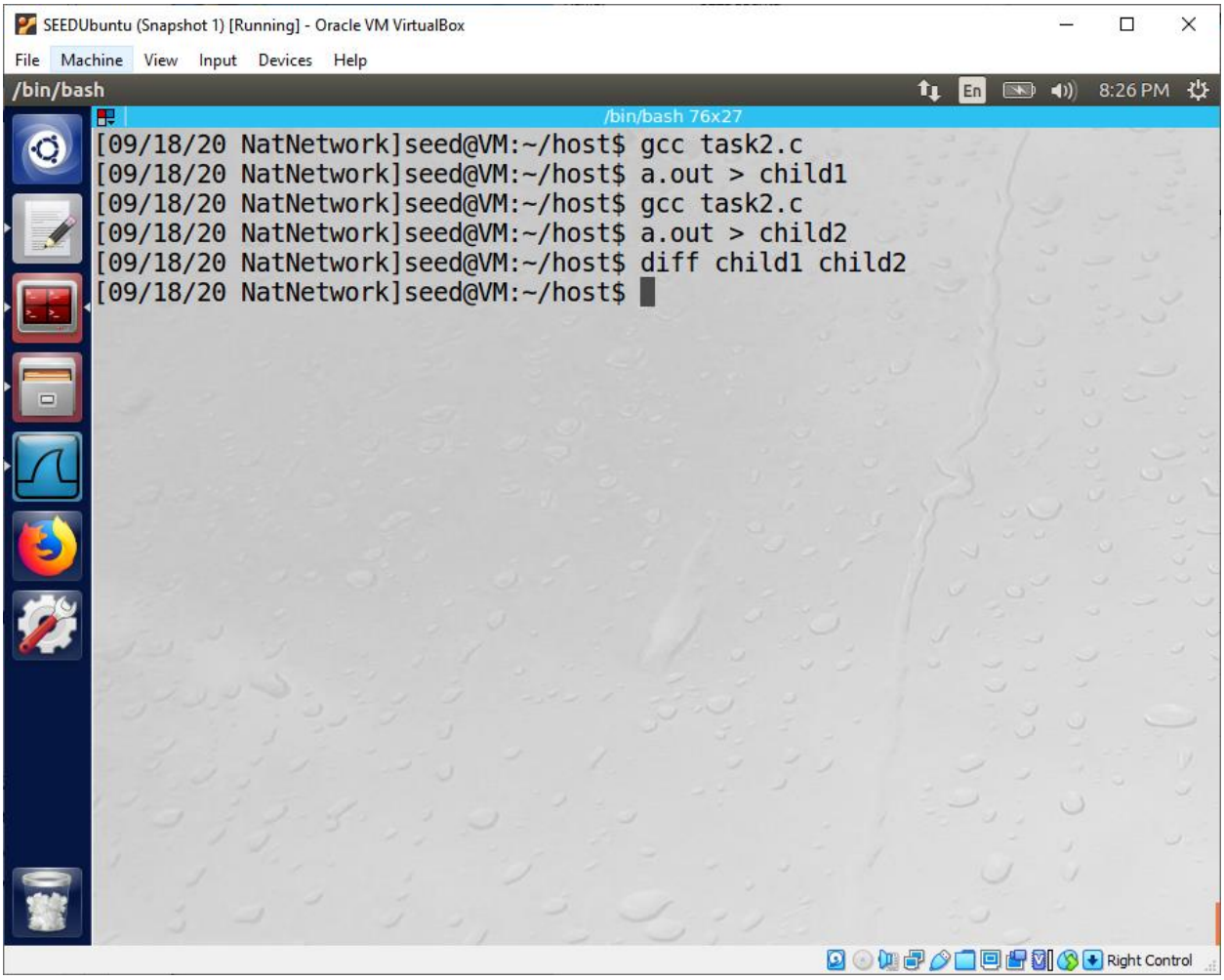
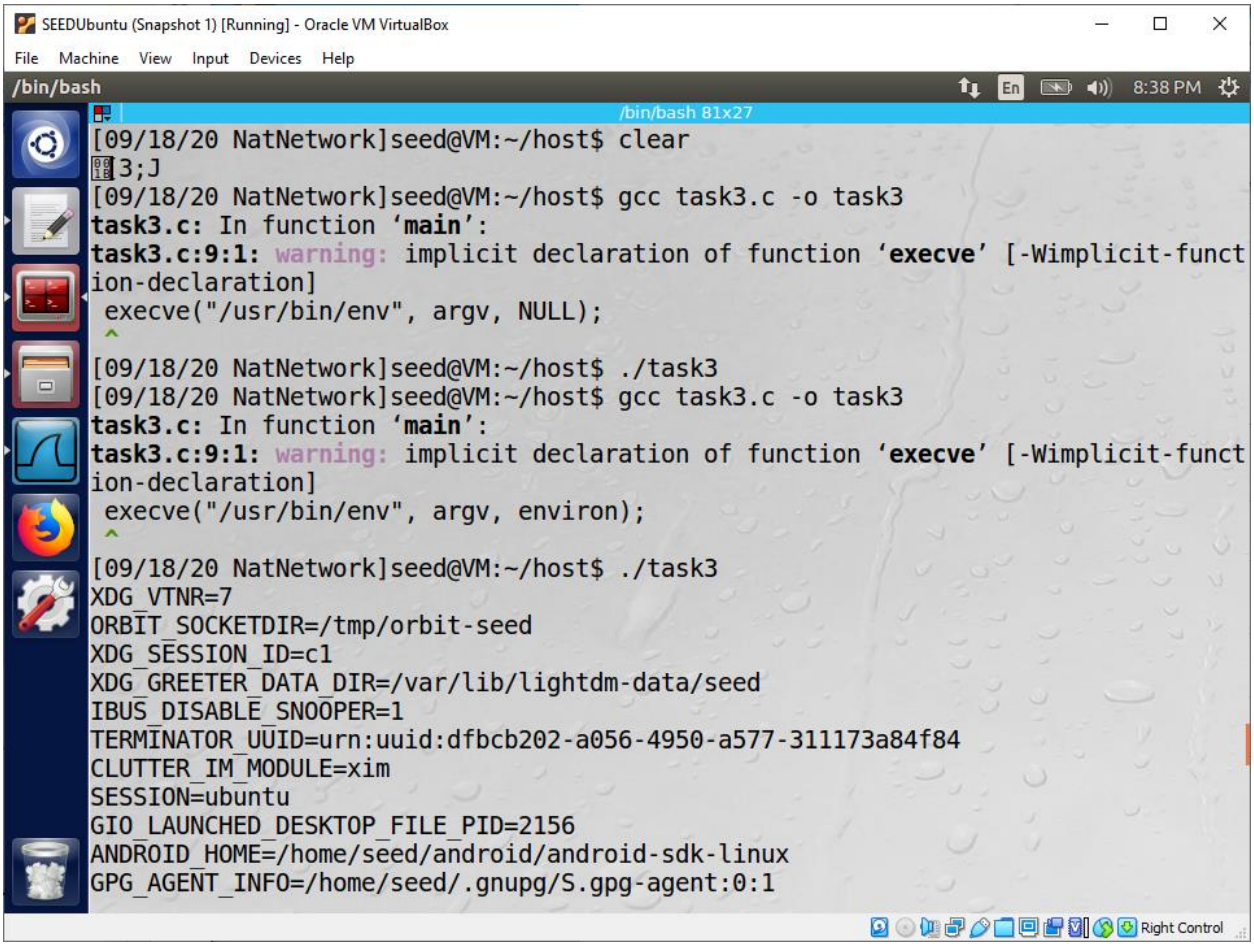
- **2.1**
- 
SEEDUbuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
/bin/bash
7:18 PM
/bin/bash 76x27
[09/18/20 NatNetwork]seed@VM:~$ printenv PWD
/home/seed
[09/18/20 NatNetwork]seed@VM:~$ env | grep PWD
PWD=/home/seed
[09/18/20 NatNetwork]seed@VM:~$ export foo="test"
[09/18/20 NatNetwork]seed@VM:~$ printenv foo
test
[09/18/20 NatNetwork]seed@VM:~$ unset foo
[09/18/20 NatNetwork]seed@VM:~$ printenv foo
[09/18/20 NatNetwork]seed@VM:~$
Right Control

- **2.2**
- 

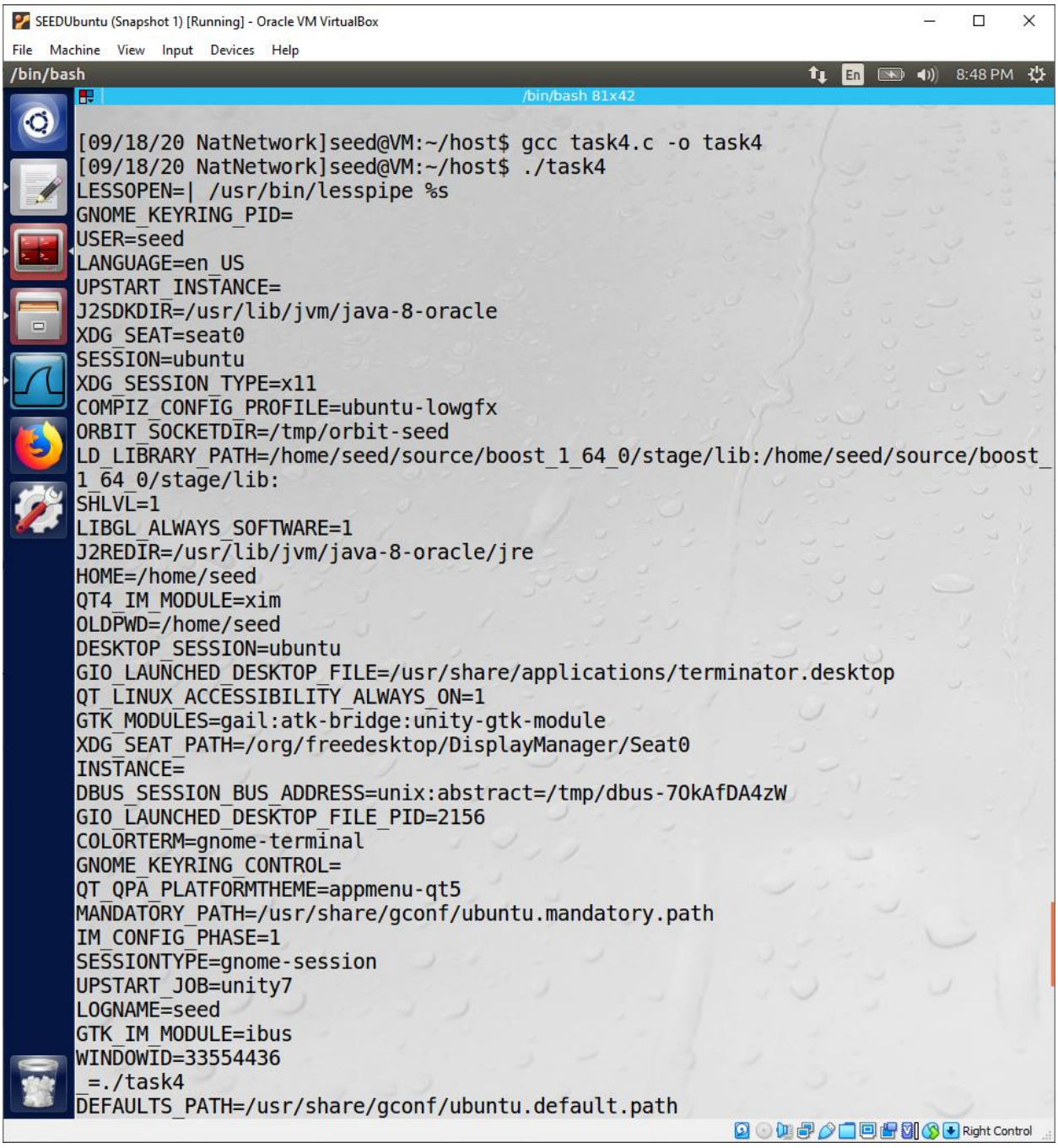
- They are the same. The only difference is the process that they call the function.

- **2.3**



- Step 1: does not print out the environment variables
- Step 2: prints out the environment variables

- **2.4**

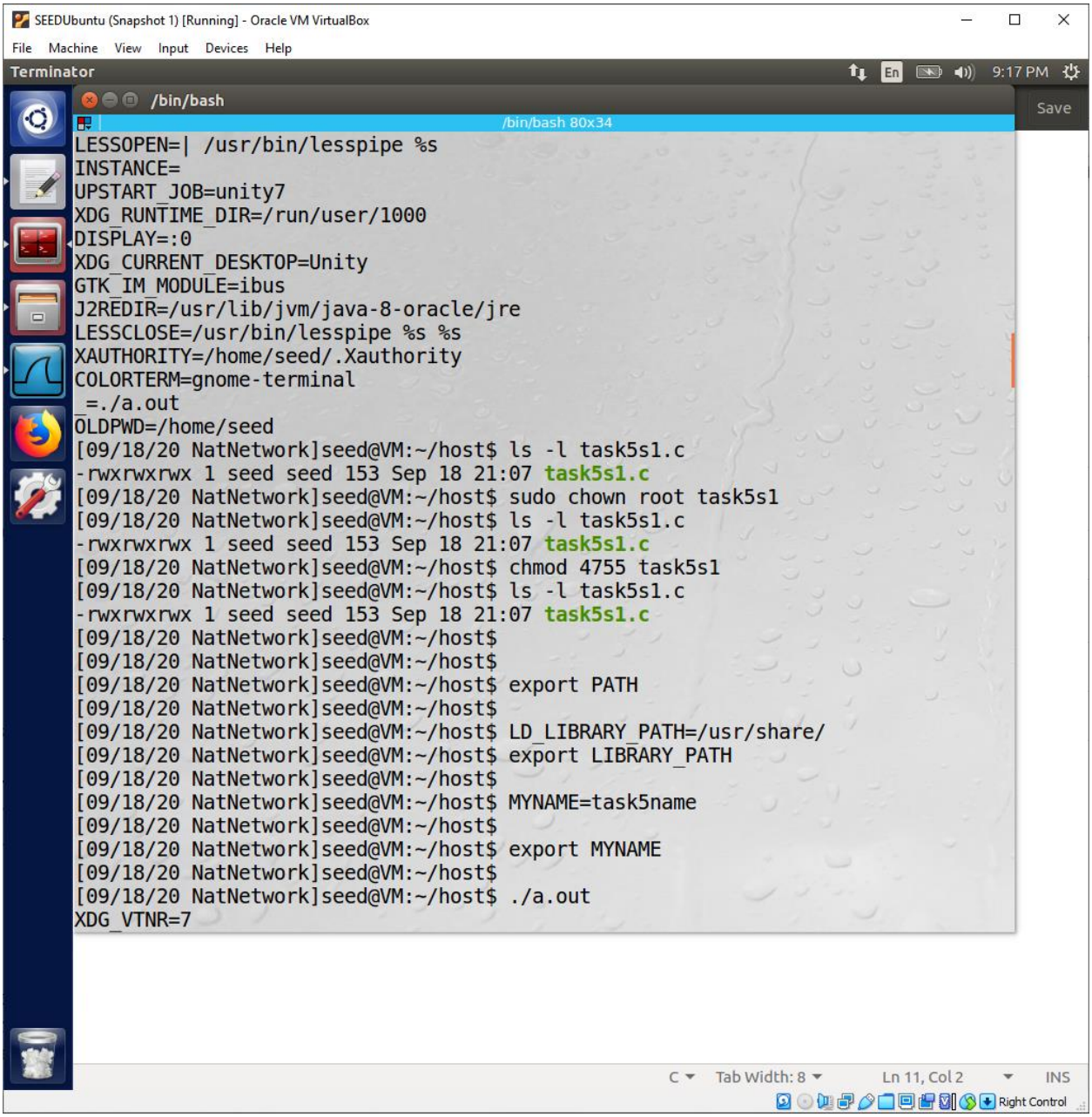
- 

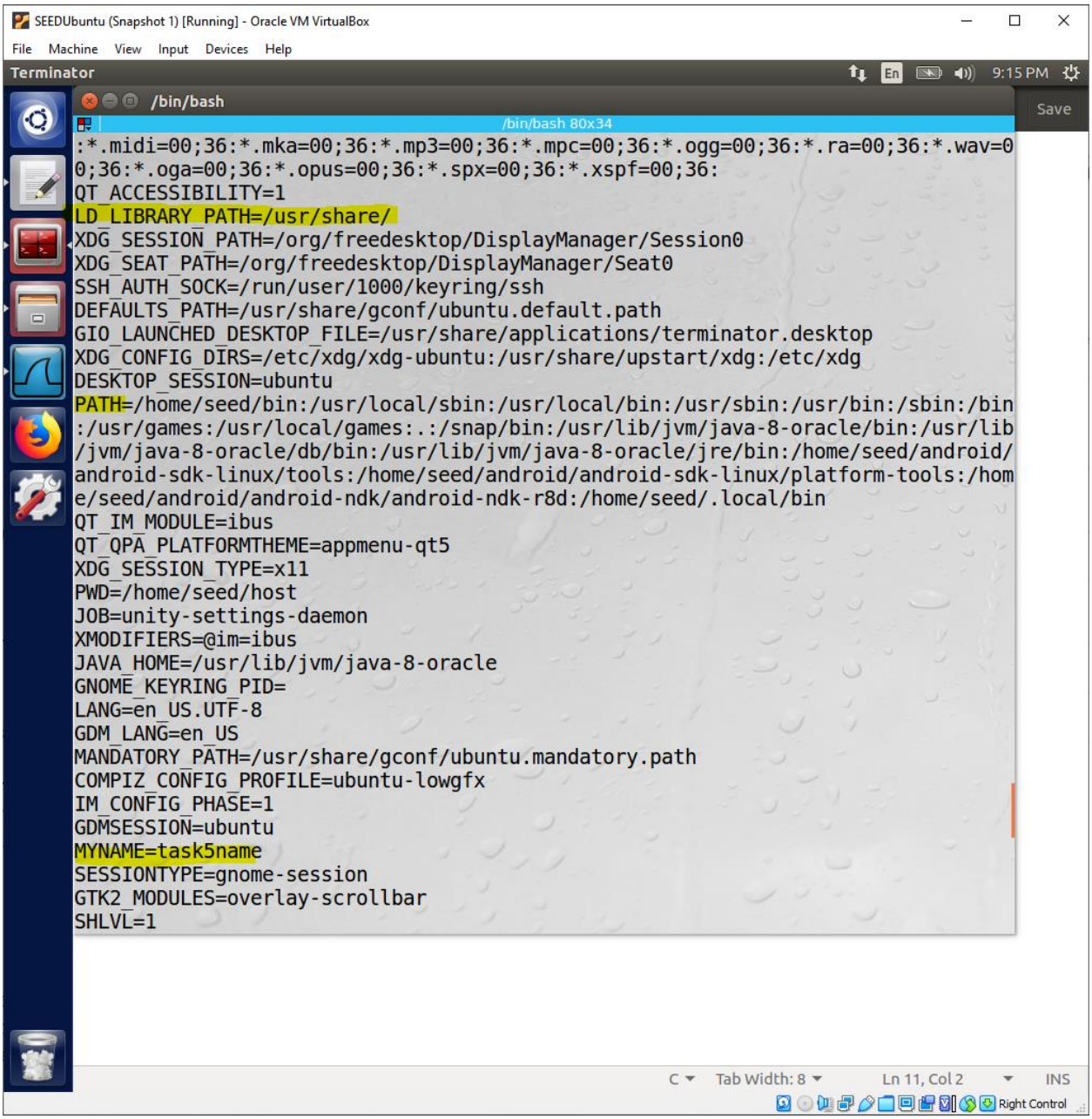
- It prints out environment variable under usr/bin. Therefore, it is verified that the system() uses excel to execute /bin/sh.

From task 2.5,

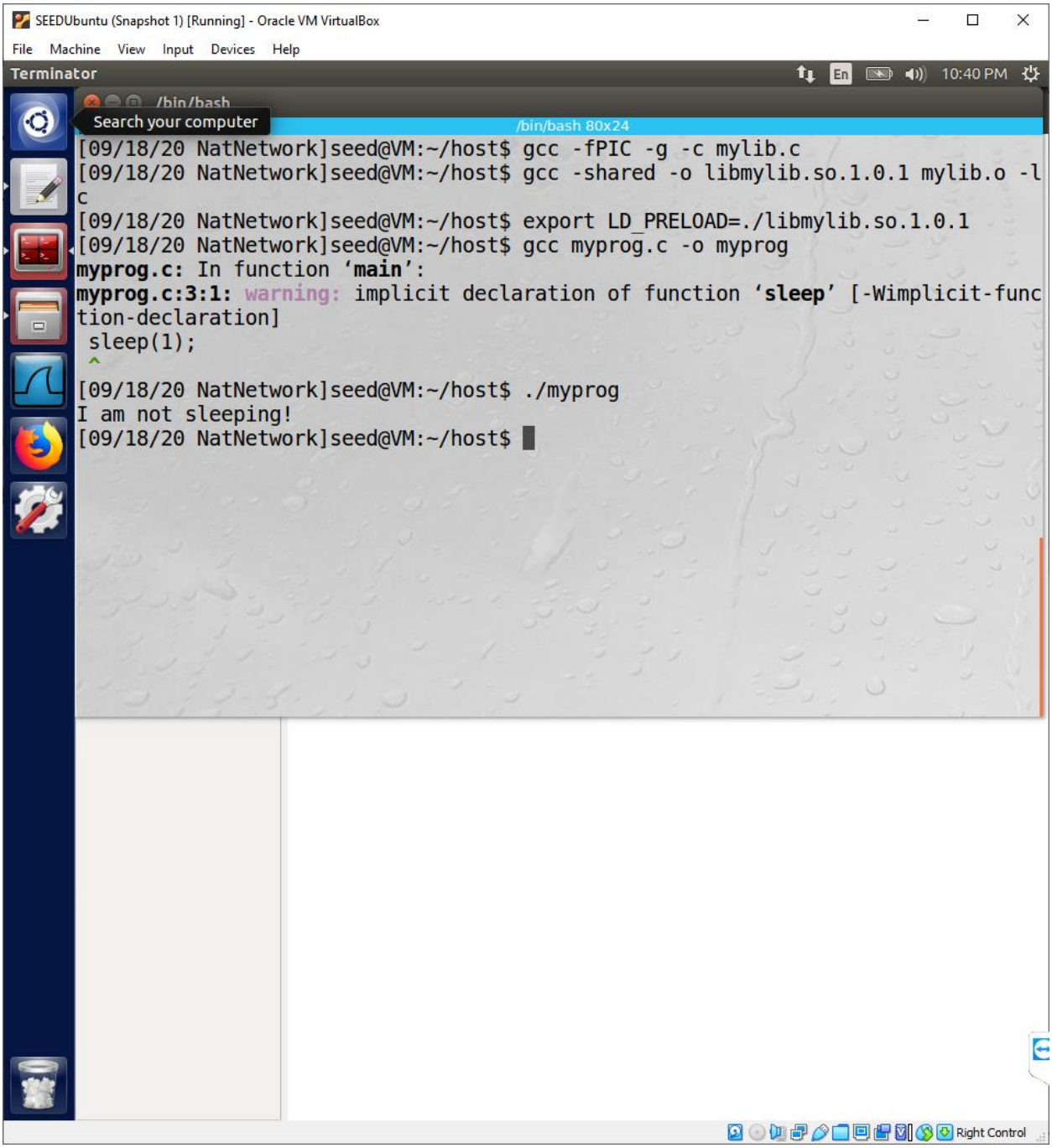
At this point, I'd notice that 'chown' does not change the owner of the files for some reason, so my results might be different from what it is supposed to be. However, I followed the lab instructions, and will attach what I've done and state what I think it is supposed to be.

- **2.5**

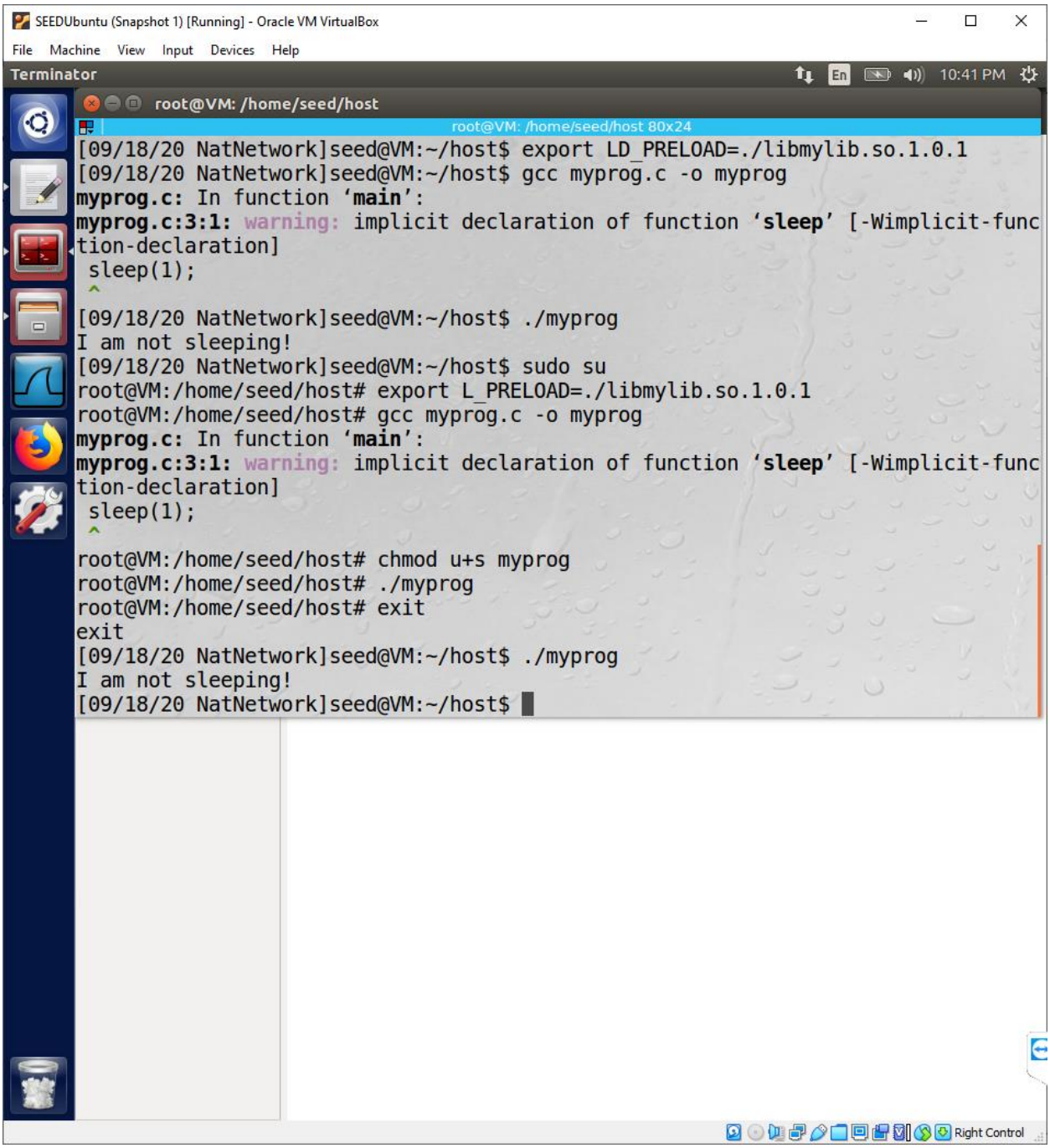

SEEDUbuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Terminator
9:17 PM
/bin/bash
Save
/bin/bash 80x34
LESSOPEN=| /usr/bin/lesspipe %s
INSTANCE=
UPSTART_JOB=unity7
XDG_RUNTIME_DIR=/run/user/1000
DISPLAY=:0
XDG_CURRENT_DESKTOP=Unity
GTK_IM_MODULE=ibus
J2REDIR=/usr/lib/jvm/java-8-oracle/jre
LESSCLOSE=/usr/bin/lesspipe %s %s
XAUTHORITY=/home/seed/.Xauthority
COLORTERM=gnome-terminal
_=./a.out
OLDPWD=/home/seed
[09/18/20 NatNetwork]seed@VM:~/host$ ls -l task5s1.c
-rwxrwxrwx 1 seed seed 153 Sep 18 21:07 task5s1.c
[09/18/20 NatNetwork]seed@VM:~/host$ sudo chown root task5s1
[09/18/20 NatNetwork]seed@VM:~/host$ ls -l task5s1.c
-rwxrwxrwx 1 seed seed 153 Sep 18 21:07 task5s1.c
[09/18/20 NatNetwork]seed@VM:~/host$ chmod 4755 task5s1
[09/18/20 NatNetwork]seed@VM:~/host$ ls -l task5s1.c
-rwxrwxrwx 1 seed seed 153 Sep 18 21:07 task5s1.c
[09/18/20 NatNetwork]seed@VM:~/host$
[09/18/20 NatNetwork]seed@VM:~/host$
[09/18/20 NatNetwork]seed@VM:~/host$ export PATH
[09/18/20 NatNetwork]seed@VM:~/host$
[09/18/20 NatNetwork]seed@VM:~/host$ LD_LIBRARY_PATH=/usr/share/
[09/18/20 NatNetwork]seed@VM:~/host$ export LIBRARY_PATH
[09/18/20 NatNetwork]seed@VM:~/host$
[09/18/20 NatNetwork]seed@VM:~/host$ MYNAME=task5name
[09/18/20 NatNetwork]seed@VM:~/host$
[09/18/20 NatNetwork]seed@VM:~/host$ export MYNAME
[09/18/20 NatNetwork]seed@VM:~/host$
[09/18/20 NatNetwork]seed@VM:~/host$ ./a.out
XDG_VTNR=7
C
Tab Width: 8
Ln 11, Col 2
INS
Right Control



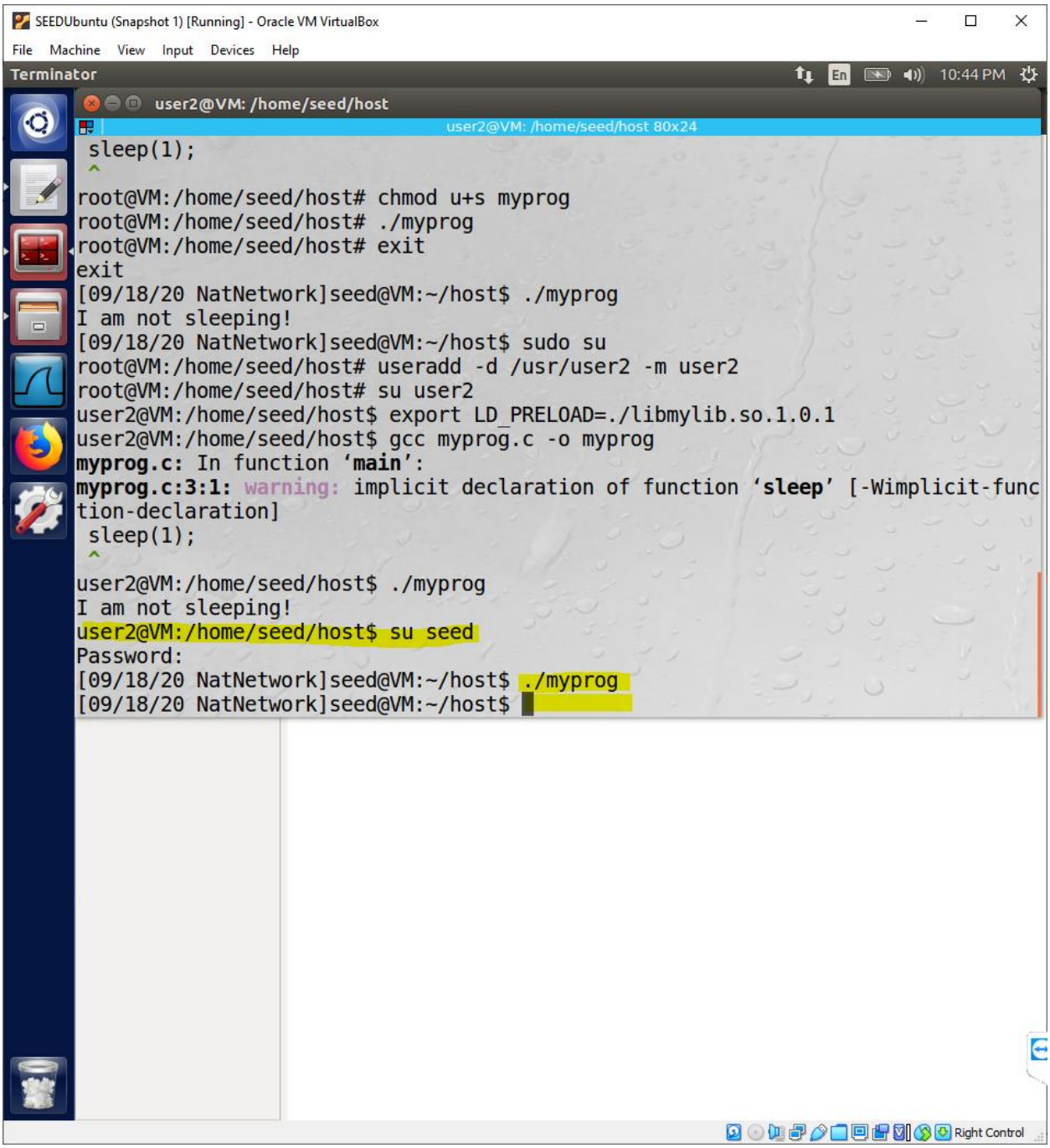
- What I expect: I am accessing the file as a non-root user. At that point, a new process was forked with root permission and the file was executed under the permission in the process. The command ‘export’ sets the environmental variable for all the processes in the system, which the root user also would be accessing. Then, this is how it is getting the values what we have set in the command.

- **2.6**
- 

- Case 1: uses LD_PRELOAD environment variable to override the sleep function.

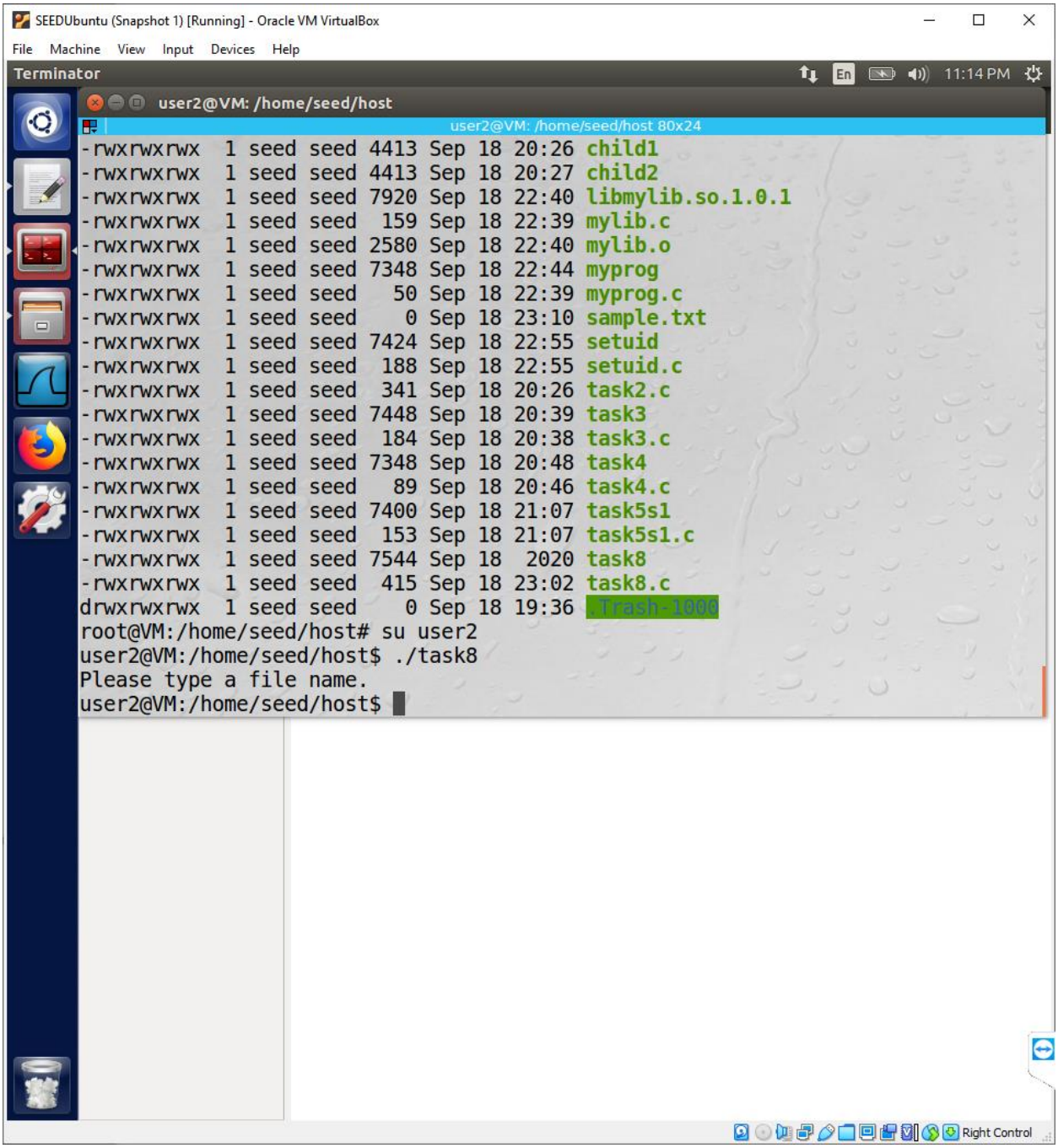


- Case 2(before exit): ignores the environment, so it is not overloaded. It just uses the system's own sleep function
- Case 3(after exit): uses the environment variable, and uses the overloaded sleep function



- Case 4: the sleep function is not overloaded.
- Conclusion: Only the program created by the user itself runs the LD_PRELOAD environment variable, overloading the sleep function.

- **2.7** (At this point, I noticed the ownership has not been changed. I tried to fix it rather to continue working on it since I could not change the ownership of any files.



- 2.8
- 2.9