Name:Ramya Ajay

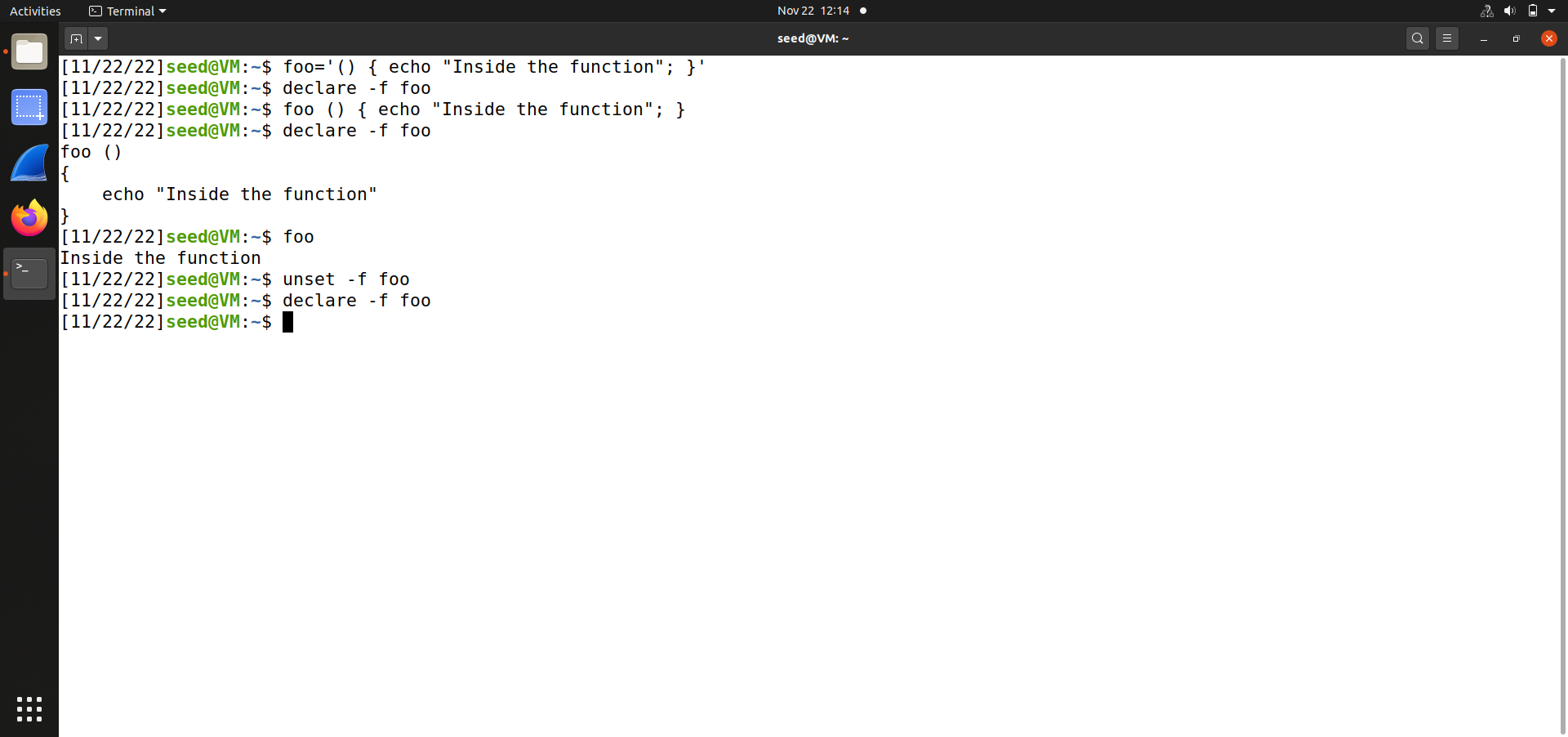
Roll No:CB.EN.P2CYS22004

Date:22/11/2022 Secure Coding Lab 10

1. ▪ Shell program is a command-line interpreter in operating systems

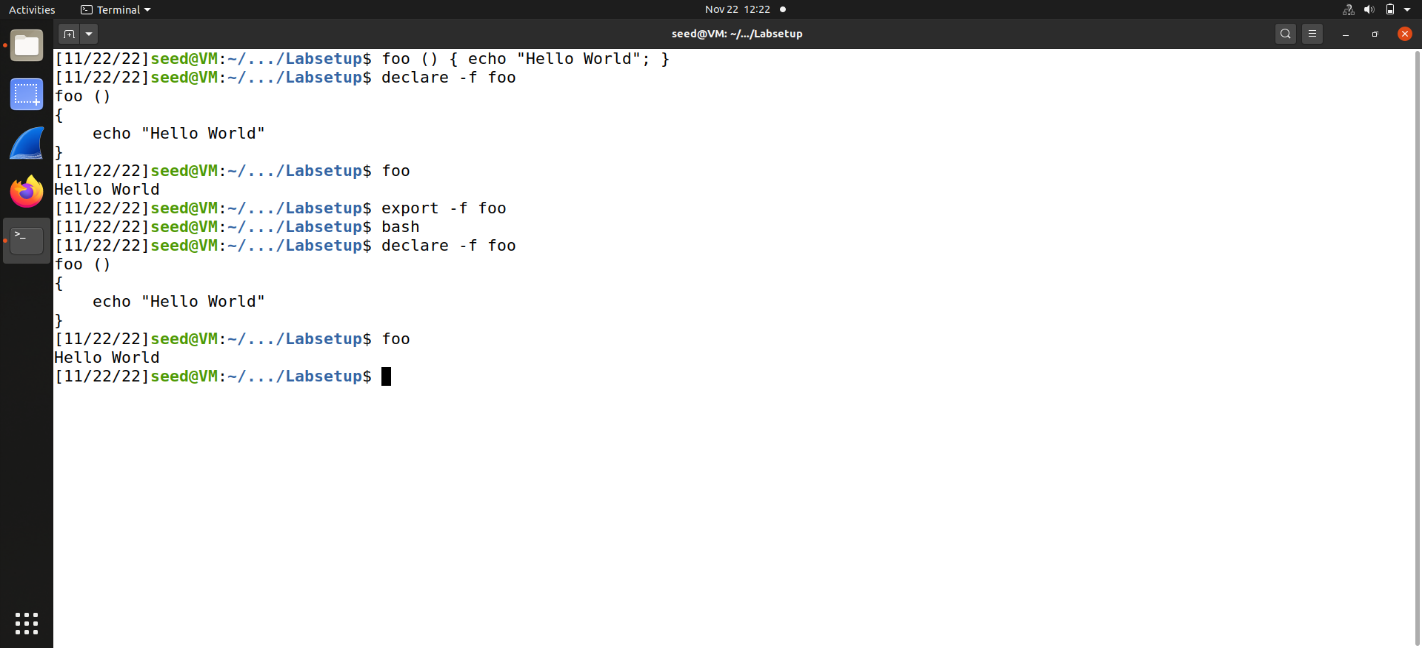
▪ Provides an interface between the user and operating system

▪ Different types of shell : sh, bash, csh, zsh, windows powershell etc

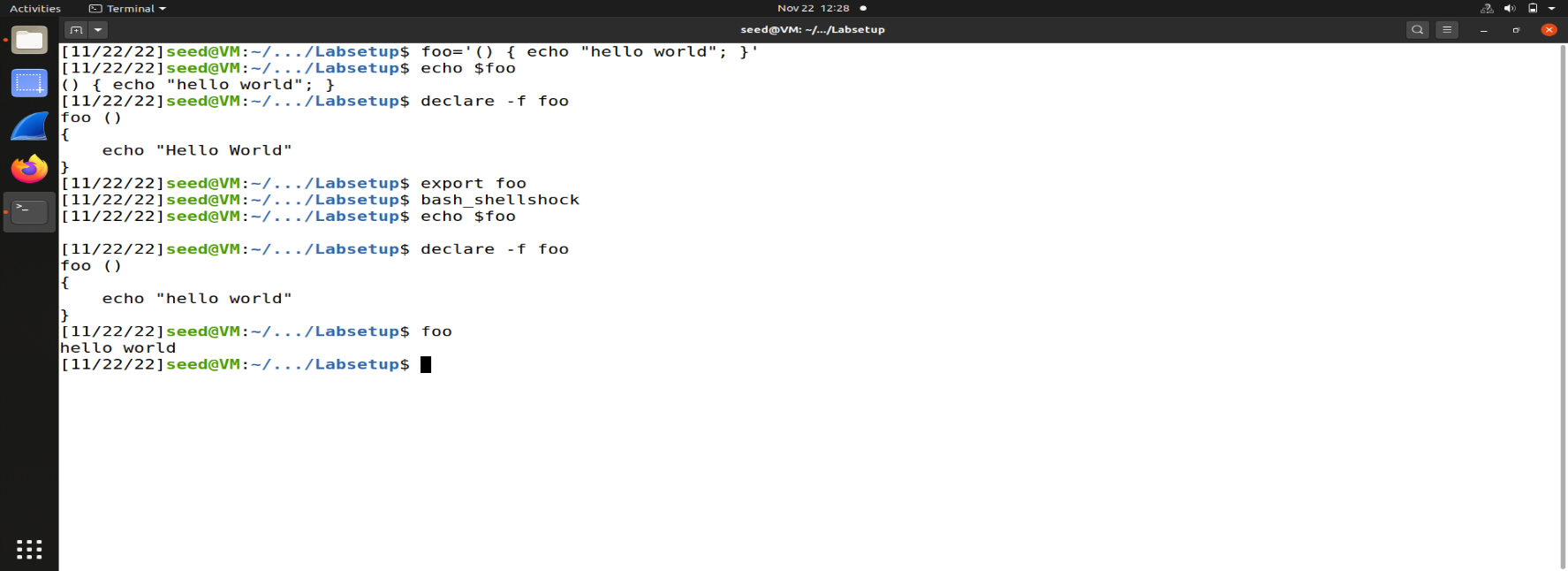


2)Passing shell function to child process

Approach 1: Define a function in the parent shell, export it, and then the child process will have it.



Approach 2: Define an environment variable. It will become a function definition in the child bash process



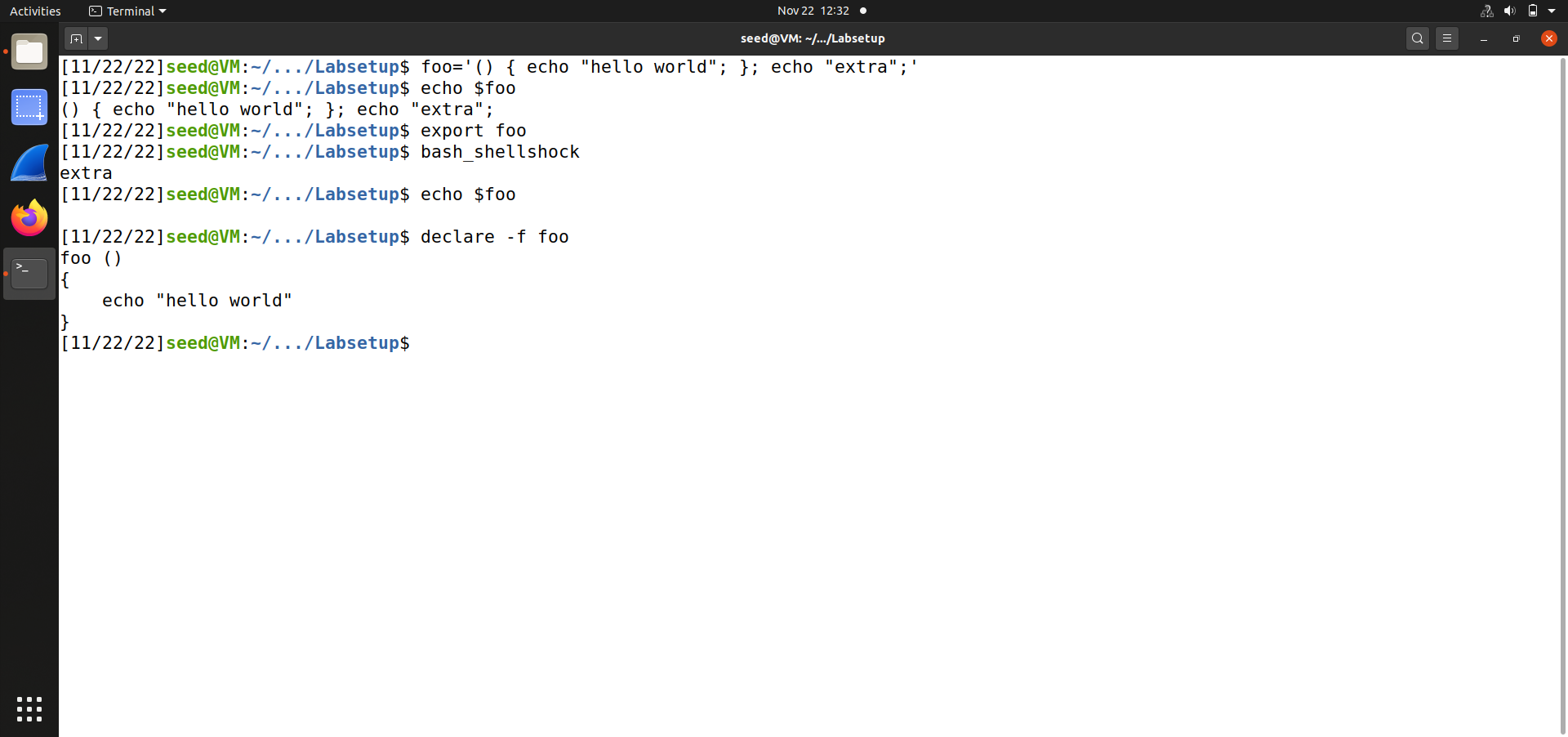
Both approaches are similar. They both use environment variables. In the first method, When the parent shell creates a new process, it passes each exported function definition as an environment variable. If the child process runs bash, the bash program will turn the environment variable back to a function definition, just like what is defined in the second method.

The second method does not require the parent process to be a shell process.

Any process that needs to pass a function definition to the child bash process can simply use environment variables

3)ShellShock Vulnerability

Parent process can pass a function definition to a child shell process via an environment variable . Due to a bug in the parsing logic, bash executes some of the command contained in the variable



4)Shell shock on setuid programs

vul.c

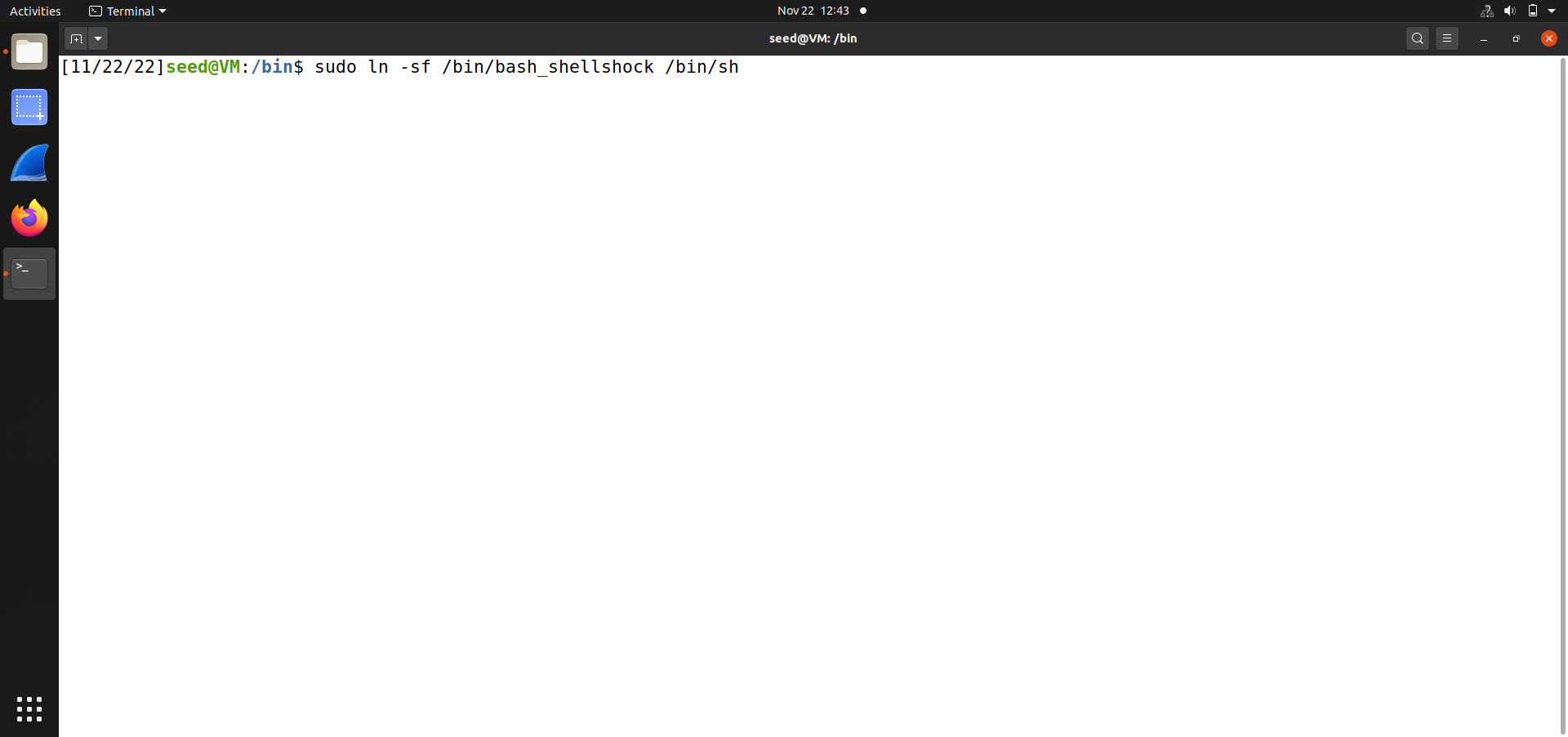
void main()

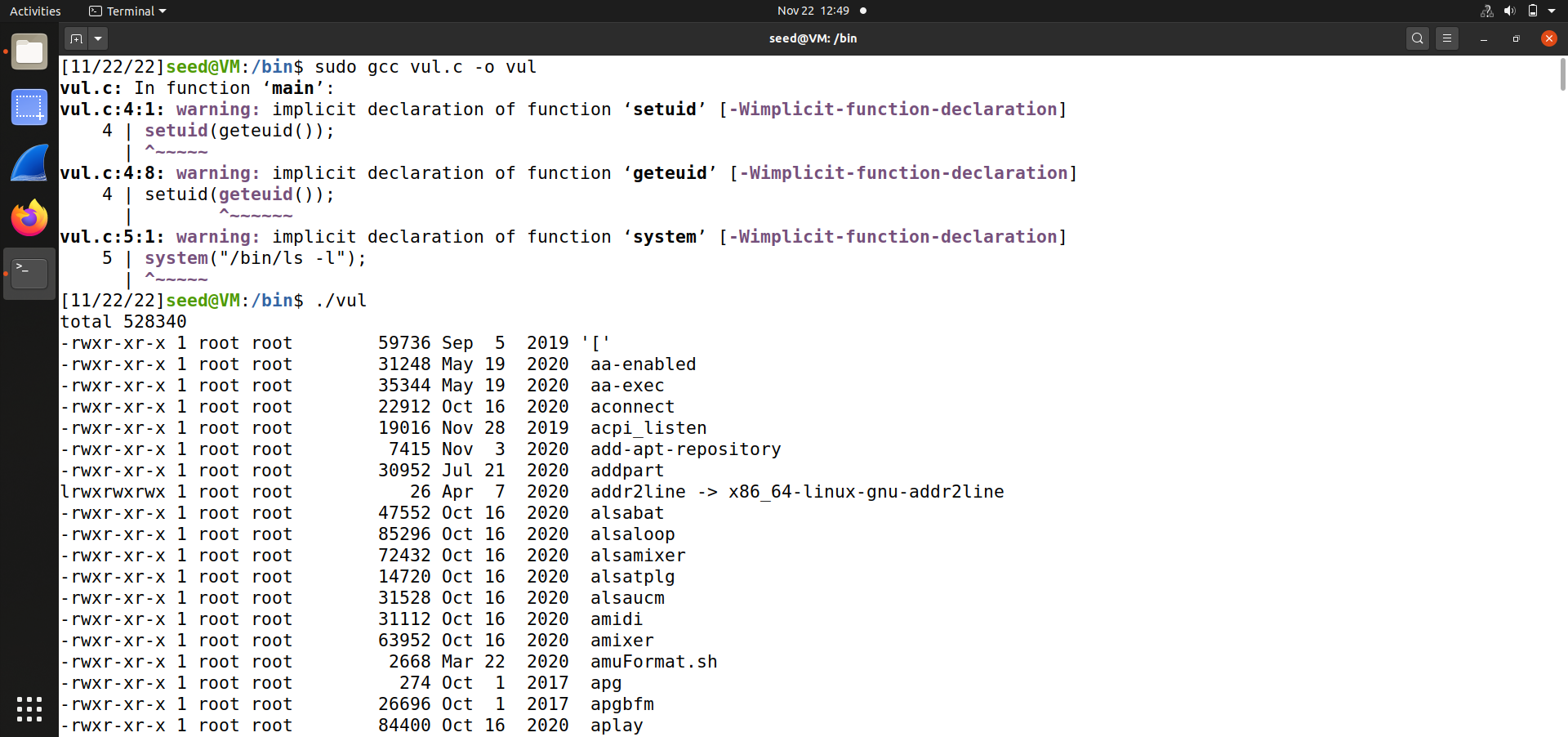
{

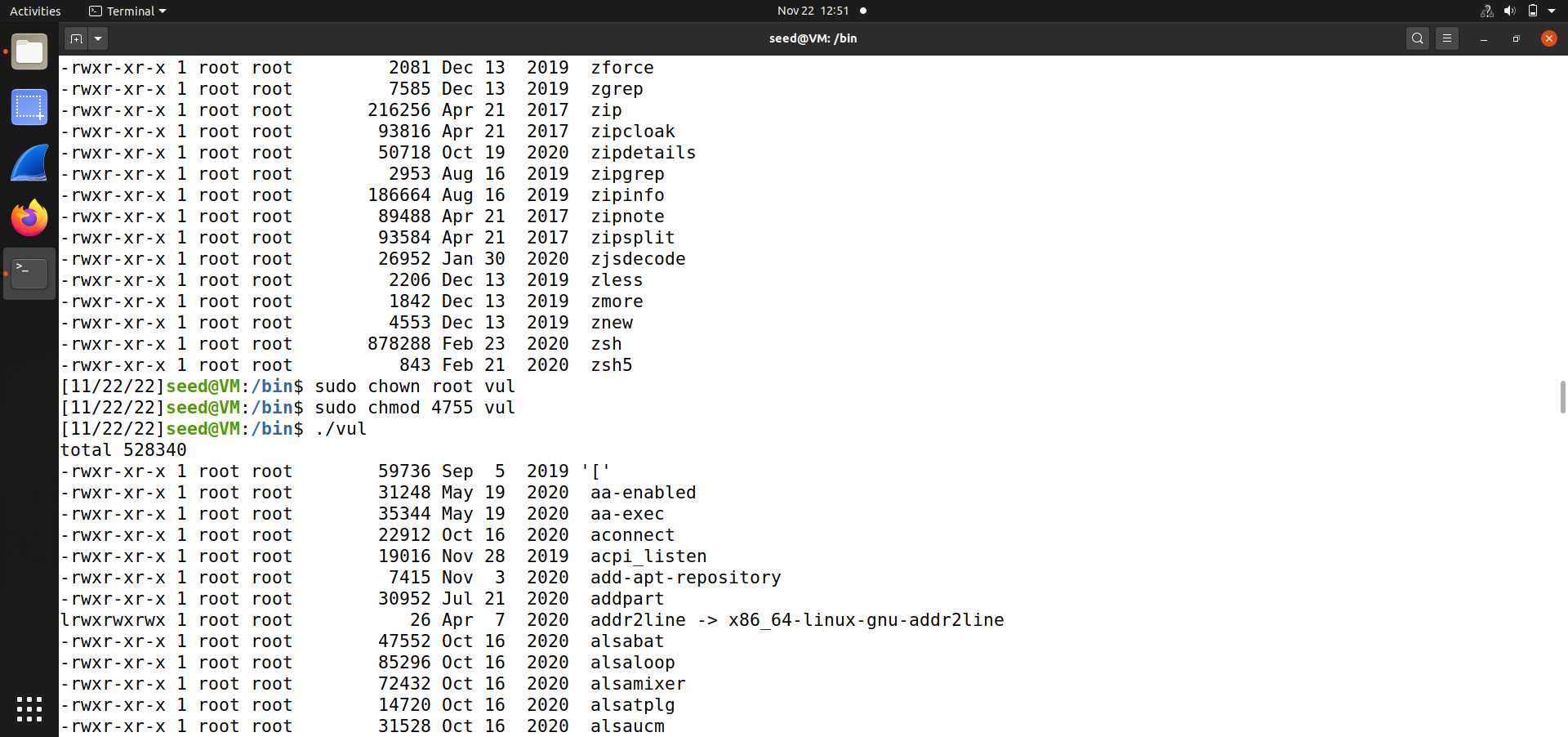
setuid(getuid());

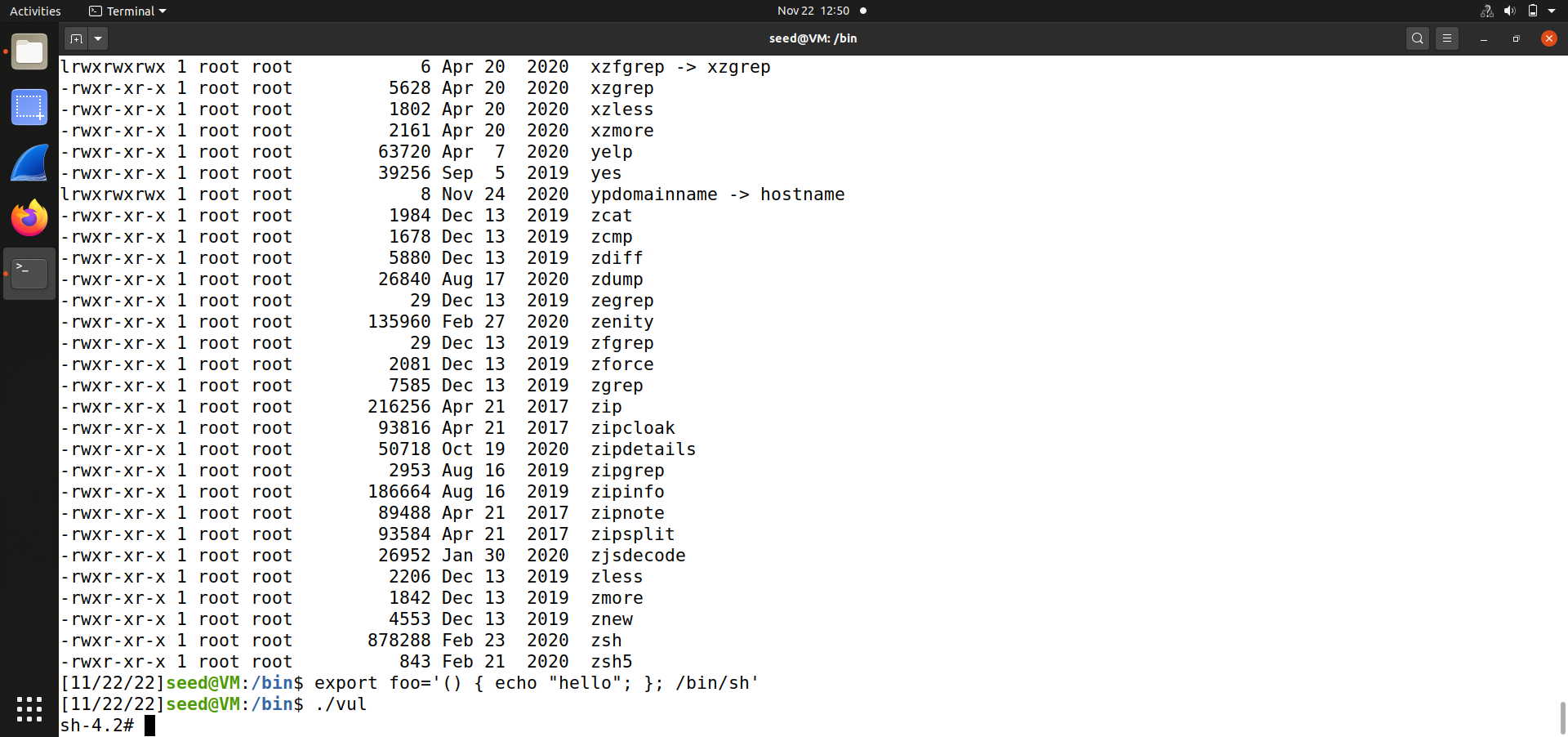
system(“/bin/ls -l”);

}









Program uses the system() function to run the /bin/ls command

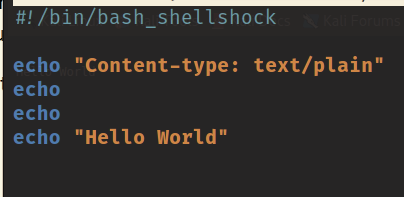
This program is a Set-UID root program

The system function actually uses fork() to create a child process, then uses execl() to execute the /bin/sh program.

The program is going to invoke the vulnerable bash program. Based on the shellshock vulnerability, we can simply construct a function declaration.

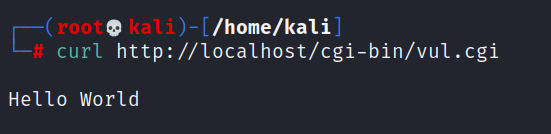
**SHELLSHOCK ATTACK ON CGI PROGRAMS –**

Here we write a very simple cgi program (vul.cgi) which just prints out the contents “Hello World” & store it in the location /usr/lib/cgi-bin/.

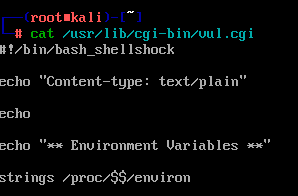


After saving the file , we need to start the apache2 server by the command systemctl start apache2.

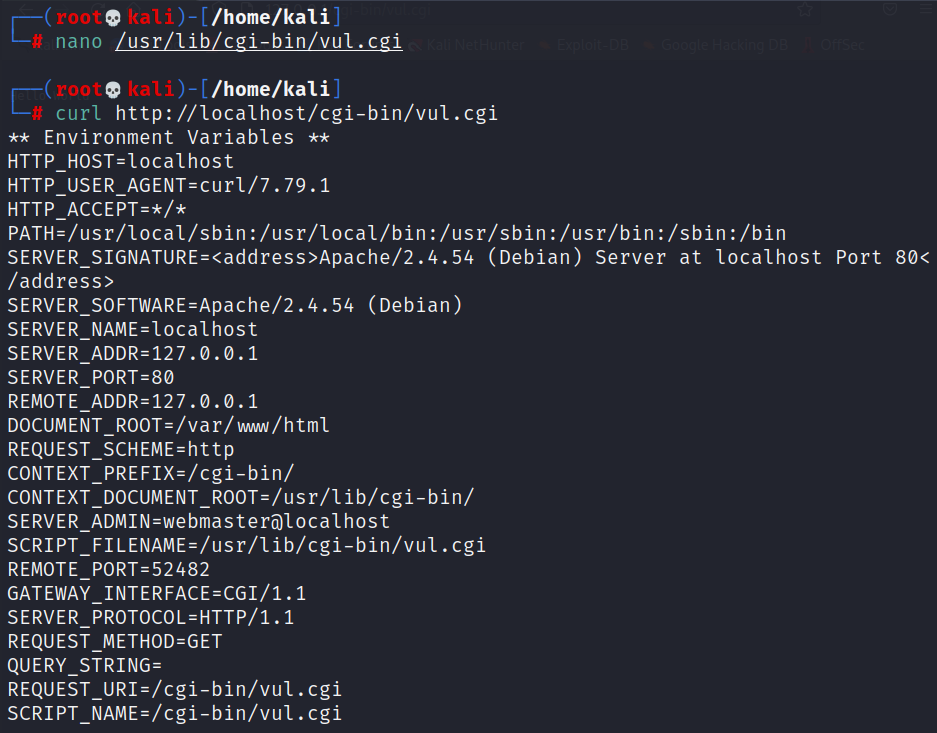
Now when we use curl command to the url in localhost to run the cgi script. We get the output Hello World as below.



Here we modify the vul.cgi to this program which will print out all the env variables of the server running vulnerable version of shellshock .

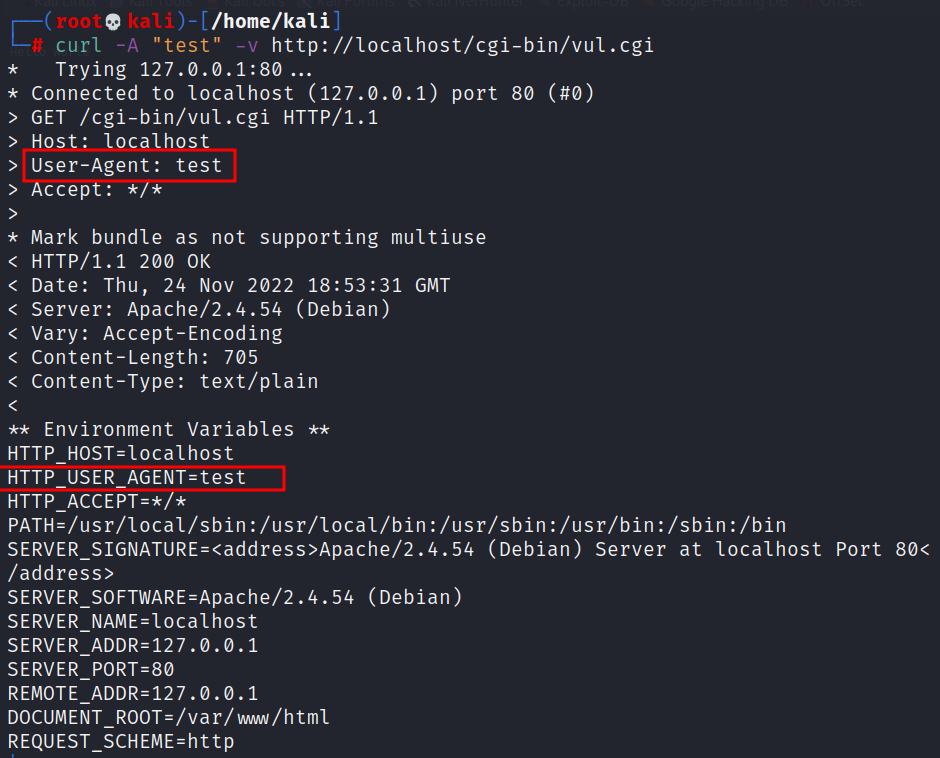


Now if we use curl command on the url ,it executes the cgi script and sends us the environmental variables. Here the **user agent is curl/7/79/1 .**



In curl command , we can use the **–A flag to change the user-agent field** to whatever we want. So we give **–A “test”.**

As seen below in the request and response, we see that the USER\_AGENT is set to test.



**Launching the shellshock attack**

Now with the below payload , we were able to execute /bin/ls command on the server using shellshock vulnerability thus leading to remote code execution on the server which is pretty dangerous.



