CTF Hw0 writeup

b05902127 劉俊緯, user ID: a127000555

Pwn

FLAG{B0fs_4re_th3_Bas1cs}

- 可以在ld那一行看到，我們需要24個垃圾char + hidden 就可以覆蓋ret位置 '\x66\x05\x40\x00\x00\x00\x00\x00'。
- 利用objdump --disassemble 即可看到位置在哪裡。
- 24個char = [rbp-0x10] (16) + (saved rbp)(8) 接下來就是ret addr.
- 最後你會看到是個shell，`cat flag` 即可。
- code: `python3 OAO.py` 再打cat flag即可看到flag。

Web

FLAG{easy_302_web_challenge}

- `curl -i` 戳一戳看有沒有。例如/flag, /hw0 , /flag，都會戳到一些奇怪的東西。
- 我記的原本有hint好像被拿掉了？我本來是靠這個hint知道是/hw0的。
- 最後戳到 /hw0.php，發現location有東西：Location: 9a0fe27c8bcc9aad51eda55e1b735eb5.php
- 直接戳 /9a0fe27c8bcc9aad51eda55e1b735eb5.php就有flag了。

Crypto

flag{how2decrypt}

- 把code一行一行讀下去，會發現在後段m==A之前，可以一直往回推去。而反推回去就是一個很冗的過程，直到需要解 `pow(m, 65537, b)` ，都是一股腦的回推回去就好了。
- 至於解pow，是用歐拉解的，詳細如下。
  - 前提：`pow(m,65537,p*q)` 求m，p,q質數。
  - $\phi(b) = \phi(pq) = (p-1)(q-1)$
  - target: find x that $m^{x(65537)} \equiv m^1 (\mod b)$
  - $m^{x(65537)} = m^{y\phi(b)+1} \equiv m^1 (\mod b)$
  - 因此，我們只要用基本RSA會用到的find mode reverse就可以找到$65537^{-1} (\mod b)$，也就可以找到x。
  - 最後 `pow(x,上面pow出來的結果,b)` 就可以找到m。
- 找完m之後，我們看到一開始的function會將chr拆解並丟到md5裡面。因此建個小型chr彩虹表就可以逆推回去了。
- code: `python OAO.py` 即可看到flag。

Stego

- 因為我不想在練一次音聽我就沒存flag了。

- 一開始就是套路。拿到一張圖先XOR/&各種數字在每個channel上，並看看有沒有神奇的東西。最後發現在blue channel &1 有神奇編碼。
- 看到上面有0101，大致上就可以猜到它是個binary file。但是後面連續的固定pattern很奇怪，所以我們就會忍不住想要先一個一個bytes的轉成chr解讀。
- 最後發現是一連串'CS 2018 Fall'，因此可以找到binary file跟字串的中斷點。萃取出來後dump到一個file就好了。
- dump到一個file，但是不知道是什麼東西，通常就會先用file/binwalk等指令去戳戳看，發現是個影片/音訊檔: mpeg。
- 我們用audicity打開它，(靠經驗或者靠波形，一般來說聲音都是漸弱，但這個mpeg是漸強。)聽一下就可以知道這是reverse的音訊檔。因此把它reverse回來之後，一個字一個字聽，在從hex轉回str，就可以得到flag。
- code: `python3 OAO.py` 可生成XD，用file拿到副檔名後append，最後你會看到XD.mpeg，接下來就考驗音聽了。

Rev

> flag{baby_java_anti_rev}

- 聽說直接丟進去一般的JD不work(or JD-GUI)，因為簽證的關係(？)
- 不幸的是，我發現有網站他的decompiler會直接略過這個問題。
  - https://jdec.herokuapp.com/ 雖然我上次看它突然down了？
- 總而言之，我們用了掠過這樣一個問題的decompiler，使的我們輕易的就拿到了code。
- 拿到了code，直接對這個程式碼反解，就可以拿到flag了。（直接拿數字丟去python貌似有byte轉型不一的問題，因此直接用java做他了。）
- code: `javac ctf.java && java ctf`

Hello CTF

- Hello CTF ：）.
- code: Hi!