Writeup: Welcome to PWN !

HACK THE PODS CTF 第2回

作問者 : 嶋田壮志

1. 説明文

PWNへようこそ！

このジャンルでは、実行ファイルを解析して脆弱性を見つけ出し、その脆弱性を使って実際に攻撃を行うよ！

このジャンルを学ぶことで、自分が書いているプログラムでどのへんが脆弱性になりやすいかが分かるのでセキュアコーディングの癖がついたり、他人のコードの脆弱性な部分を発見できるようになり、オープンソースソフトウェア（OSS）の脆弱性を見つけたりすることができるよ！

興味がある方はぜひこのジャンルを極めていこう！

PWNの導入は、実行ファイルを実行し動作を確認してから、ソースコードを読んでバグを見つけるんだ。

その後は、見つけたバグを使って手元の環境で攻撃を再現できたら、実際に問題サーバーへ攻撃を行ってみよう！

1. 説明文（続き）

問題サーバーへのアクセス方法：

以下のコマンドをshellに入力してください

`nc xxx.xxx.xxx.xxx 60000`

2. 共有されるファイル

- welcome_pwn.bin
  - 問題のメインファイル
  - 自身の解答の確認に使う
- welcome_pwn.c
  - 上記実行ファイルのソースコード
  - これを見て脆弱性やセキュリティの穴を見つける

3. Writeup

実行ファイルを実行してみます

```
babyblue0@BayB0 /mnt/c/Users/user/Downloads
 % chmod +x ./welcome_pwn.bin
babyblue0@BayB0 /mnt/c/Users/user/Downloads
 % ./welcome_pwn.bin
Great!
You connected problem server!
Flag is here!

Input password: AAAAAAAAAA
Incorrect password...
Bye!
```

ユーザーにパスワードを入力を求めてます。

3. Writeup

ソースコードを確認してみます（次ページで細かく解説します）

```
26 int main()
25   char pass[255] = { 0 };
24
23   setup();
22
21   puts("Great!");
20   puts("You connected problem server!");
19   puts("Flag is here!");
18
17   puts("");
16   printf("Input password: ");
15   scanf("%255s", pass );       // Input password
14
13   if( strcmp( pass, "GAS_GAS_GAS" ) == 0 ){    // Compare input password with correct password
12   |  puts("Congrats!");
11   |  puts("");
10   |  //print_flag();
 9   |  system("/bin/sh\0");     // Get shell!!
 8   }
 7   else {
 6   |  puts("Incorrect password...");
 5   }
 4
 3   puts("Bye!");
 2
 1   return 0;
53 }
```

```
47 void setup(){
46   setbuf(stdin, 0);
45   setbuf(stdout, 0);
44   alarm( 180 );
43 }
42 void print_flag(){
41   FILE *fp;
40   char flag[64];
39
38   fp = fopen( "flag.txt", "r" );
37   if( fp == NULL ){
36   |  printf("Faild open file: flag.txt");
35   |  exit( 1 );
34   }
33
32   fgets( flag, sizeof( flag ), fp );
31   printf( "%s", flag );
30
29   return;
28 }
27
```

3. Writeup

main関数の流れ（黒背景でゴメンね！）

3. Writeup

ユーザーが入力した文字列（パスワード）を、
strcmp（文字列比較関数）で、”GAS_GAS_GAS”と比較している！

⇒ つまりパスワードは、”GAS_GAS_GAS” (flagではない)

また、いかにもflagを出力しそうな関数、print_flag関数では、
同ディレクトリ上の「flag.txt」の内容を出力している！

⇒ つまりflagは、「flag.txt」の内容

3. Writeup

実際に手元のバイナリで動かしてみる

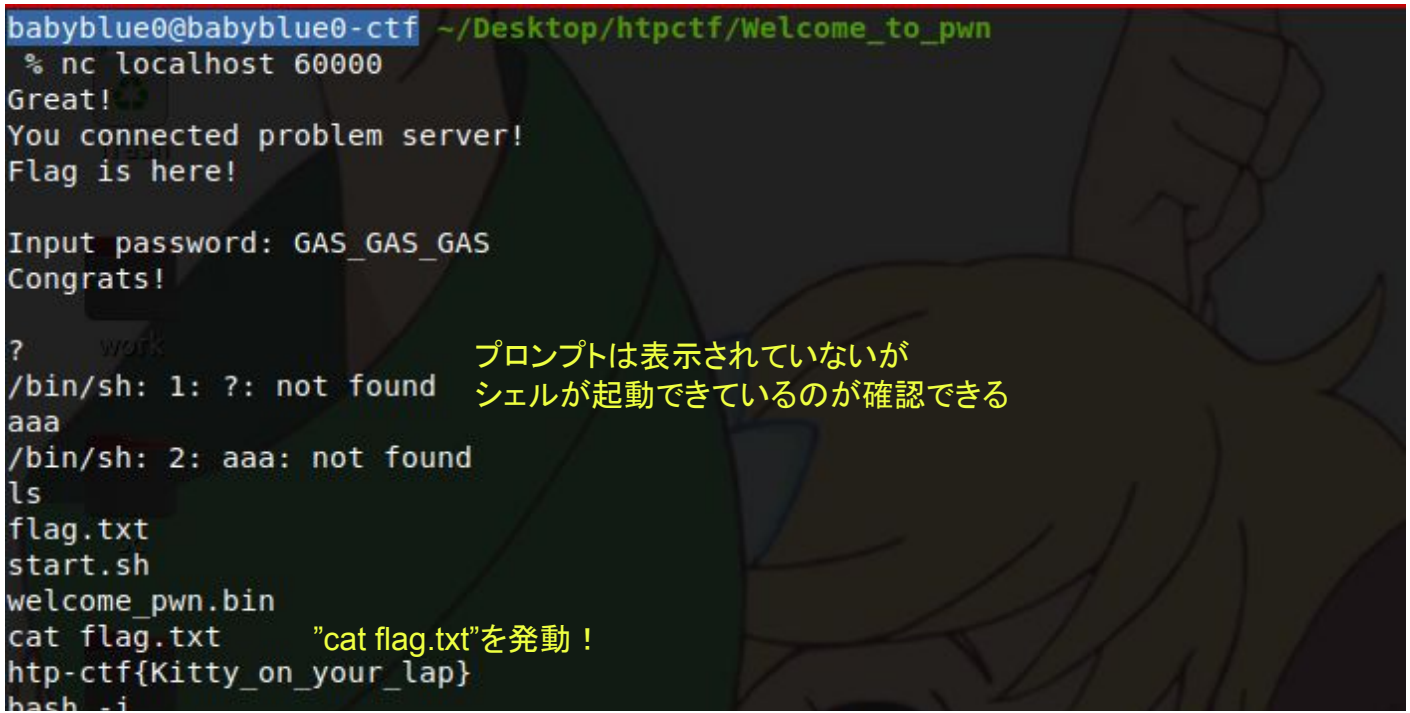
```
babyblue0@BayB0 /mnt/c/Users/user/Downloads/wp
 % ./welcome_pwn.bin
Great!
You connected problem server!
Flag is here!

Input password: GAS_GAS_GAS  パスワードの入力
Congrats!

$ ls  シェルを獲得！（$はプロンプト）
welcome_pwn.bin  welcome_pwn.c
$ cat flag.txt flagは持っていないので表示ができない
cat: flag.txt: No such file or directory
$ exit
Bye!
```

3. Writeup

手元の環境でうまく行ったので、リモートサーバー上で検証してみる



3. Writeup

flag: htp-ctf{Kitty_on_your_lap}

ちなみに、、、

flagの「Kitty on your lap」は、
PCゲーム「ひざの上の同居人（1998）」が元ネタ。
、、、なのですが、僕は知らないです。

僕がC言語を学んでいたときに、
サンプルコードで使われていたため本問題でも参考にしました。

言うなれば、ある界隈の「Hello World」的なノリだと思います。
ようこそ！PWNへ！（ ニャ～ン(ΦωΦ) ）