Reverse - RewriteMe - 100 points

Kévin DUVERGER

Table des matières

1 Résolution RewriteMe :

Comme d'habitude, nous pouvons essayer `strings` pour tenter d'avoir des informations. On peut trouver des chaînes du genre `"Please enter the password : "` mais rien de beaucoup plus intéressant.

On peut encore une fois le passer dans un décompilateur en essayant de récupérer le code du main :

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
    int v3; // eax
    char Str[48]; // [esp+18h] [ebp-68h] BYREF
    char Str2[48]; // [esp+48h] [ebp-38h] BYREF
    FILE *v7; // [esp+78h] [ebp-8h]

    __main();
    printf("Please enter the password : ");
    gets(Str);
    if ( Str[strlen(Str) - 1] == 10 )
        Str[strlen(Str) - 1] = 0;
    Str[47] = 0;
    v3 = strcmp(Str, Str2);
    if ( v3 ) {
        puts("Awwww ... too bad, that is not the password !");
    } else {
        v7 = fopen("flag.txt", "r");
        fgets(Str, 40, v7);
        printf("GG ! The flag is : %s\n", Str);
        fclose(v7);
    }
    return 0;
}
```

En regardant ce code, vous voyez qu'un mot de passe est demandé à l'utilisateur et que c'est la fonction `gets` qui s'en charge en mettant le résultat dans la variable `Str`. La variable `Str2` contient probablement le mot de passe, ce dernier étant stocké dans une autre section de l'exécutable, c'est pourquoi sa valeur n'apparaît pas ici.
La fonction `gets` ne vérifie pas la taille de ce qui est entré par l'utilisateur, aucune autre vérification n'est faite dans le code, les variables pour la chaîne de l'utilisateur et le mot de passe sont côte à côté sur la pile (ce sont des variables locales) et le challenge se nomme `RewriteMe`, tout nous fait donc penser à une possible **buffer overflow** !
L'idée serait donc de ré-écrire la chaîne `Str2` de telle sorte qu'elle soit identique à `Str`. Pour ce faire, vous pouvez voir que le code nous aide en plus : il met bien le dernier caractère de `Str` à 0 et il change le `\n` à la fin de la chaîne par un caractère nul également. L'idée va donc être de rentrée une chaîne en 2 parties : 48 premiers caractères où vous mettez ce que vous voulez, et les 47 suivants où il faudra mettre l'identique des 47 premiers.

En envoyant :
`aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa`
on arrive donc à faire cette ré-écriture et à obtenir le flag qui est : `cryptisCTF{buff3r_0v3rfl0ws_ar3_fun}`.