Note de vulnérabilité - Reverse challenge

Kerchouche Fatiha - Haral Dylan - Francio François - Clément Lyonnet

Lors de l'analyse d'exécutables, il a été possible de retrouver le mot de passe demandé par celui-ci.

1 Vulnérabilité(s)

- Les codes exécutables contiennent des informations sur le mot de passe à entrer.
- Le flag ainsi retrouvé est "3004".

2 Méthode(s)

- En désassemblant le premier exécutable, il a été possible d'analyser le comportement de celui-ci.
- Il a été possible de voir que `exe1` fait appel à `exe2` (voir Annexes plus bas).
- En désassemblant `exe2`, les caractères du mot de passe demandé sont ainsi retrouvés.
- Une fois le mot de passe correct entré à `exe1`, celui-ci télécharge un autre exécutable qui génère le fichier contenant le flag.

3 Outils utilisés

- Des logiciels comme Ghidra ou IDA permettent de retrouver le logique du code source ainsi que le mot de passe nécessaire.

4 Annexes

- La routine principale lancée par `exe1` :

```
void processEntry entry(undefined8 param_1,undefined8 param_2)

{
  undefined auStack_8 [8];

  routine(routine2,param_2,&stack0x00000008,0,0,param_1,auStack_8);
  do {
                    /* WARNING: Do nothing block with infinite loop */
  } while( true );
}
```

- La routine d'`exe1` fait appel à `exe2` :

```
password = argv[1];
canary = *(long *)(in_FS_OFFSET + 0x28);
if (password != (char *)0x0) {
  if (*password == 'h') {
    if (((password[1] == 'i') && (password[2] == 'n')) && (password[3] == 't')) {
      printf("Ahoy :)\nI accept a password of 12 characters, good luck !");
      returnVal = 0;
      goto CANARY_CHECK_LABEL;
    }
  }
  else if (*password == '\0') goto PRINTF_LABEL;
  passwordLen = thunk_FUN_00410ac0(password + 1);
  if ((passwordLen == 11) && (argc == 2)) {
    passwordLen = execve("./exe2",argv,0);
    if (passwordLen != -1) {
      returnVal = 0;
      goto CANARY_CHECK_LABEL;
    }
```

- Les caractères du mot de passe sont retrouvés dans `exe2` :

```
undefined8 routine2(int argcCpy,char **argv)

{
  undefined8 returnVal;
  long in_FS_OFFSET;
  long canary;
  char *password;

  returnVal = 1;
  canary = *(long *)(in_FS_OFFSET + 0x28);
  password = argv[1];
  if (((((*password == 'Y') && (password[5] == '1')) && (password[3] == 'Z')) &&
      ((((password[7] == '.') && (password[9] == 'D')) &&
        (((password[0xb] == ',') && ((password[1] == '~') && (password[2] == '4')))))) &&
       (password[4] == 'C')))) &&
     (((password[6] == 'T') && (password[8] == 'S')) && (password[10] == 'r')))) {
```

- L'analyse d'`exe2` révèle l'exécution d'`exe3` :

```
FUN_00404b40("chmod +x ./exe3");
FUN_00404b40("./exe3");
```

- Le flag est retrouvé sous forme de GIF :