Reverse - HackMyResources - 75 points

Waïl TAHMAOUI, Kévin DUVERGER

Table des matières

1 Résolution HackMyResources :

1.1 Avec un outil à télécharger :

On peut encore une fois tenter d'utiliser `strings` mais ça ne donne rien.

En passant ce fichier dans un décompilateur la seule chose que l'on peut récupérer comme code est :

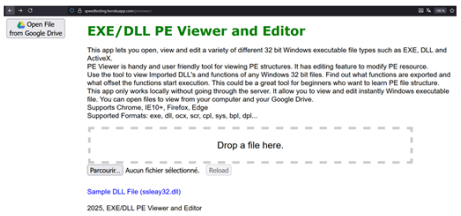
```
int __cdecl main(int argc, const char **argv, const char **envp)
{
    __main();
    puts("Hello, I do nothing !");
    return 0;
}
```

Donc il faut probablement aller chercher ailleurs !

Le challenge se nomme `HackMyResources` et nous avons un fichier exécutable Windows entre les mains (un fichier PE), peut être qu'il y a un lien entre `Resources` et les fichiers PE ? En cherchant `"resources in PE files"` sur un navigateur, vous pouvez trouver de nombreuses pages qui parlent d'une section de ressources `.rsrc` dans ces fichiers. Nous pouvons donc en conclure qu'il y a potentiellement quelque chose de caché dans cette table des ressources.
On peut ensuite aller chercher des outils pour analyser cette table des ressources, sous Windows `ResourceHacker` est un très bon choix et sous Linux vous pouvez prendre `PE-bear`. Une fois cette table des ressources analysée, voici ce que l'on peut obtenir :

1.2 Avec un outil en ligne :

Pour ce challenge, on peut aussi utiliser le site `https://speedtesting.herokuapp.com/peviewer/` pour analyser les fichiers exécutables windows (PE files). En fouillant un peu, on trouve dans les ressources une mignonne photo de chat avec un joli flag !



(a) herokuapp.com

(b) PE files.

(c) Ressource/RCData/.