TP 07 : Petit hack entre amis

Introduction: Ce TP revient sur les principes du linkage en C.

Édition de liens : linking

On considère le fragment suivant d'un programme de C :

```
x=y+z;
printf("x=%d\n",x);
```

En analysant le code assembleur produit par le compilateur on trouve que la première instruction va directement être réalisée par des opérations sur les registres et la mémoire tandis que la deuxième fait appel à une fonction externe.

`printf` est en effet réalisée par une fonction fournie par une *bibliothèque partagée*. Le modèle de compilation préféré de Linux/Unix et d'autres systèmes est de stocker les programmes exécutables dans les fichiers sous une forme incomplète, tout en spécifiant quelles fonctions partagées seront nécéssaires à son exécution. Lors de l'exécution, le programme sera chargé dans la mémoire, puis un *éditeur de liens* établit les connexions entre le programme et les bibliothèques partagées : le «*linking*» en anglais.

Normalement, le système sait où trouver les bonnes librairies. Le comportement de l'éditeur de liens peut être modifié par deux variables :

— La variable `LD_LIBRARY_PATH` (qui peut être modifié sur la ligne de commande) donne une liste de répertoires où l'éditeur de liens pourra trouver des bibliothèques.

— La variable `LD_PRELOAD` définit quelques librairies dont l'utilisation sera prioritaire.

Dans cet exercise nous explorerons surtout des applications de l'utilisation de `LD_PRELOAD`.

1. On considère le programme `password1` (disponible sous forme binaire) qui demande un mot de passe. Trouvez le bon mot de passe.
2. Faites de même pour `password2`.
3. Comment éviter les failles de sécurité présentes dans ces deux programmes ?
4. Le programme `game` vous demande une numéro entre 1 et 100 choisi aléatoirement lors de chaque exécution. Comment tricher pour gagner toujours au premier tour ?