PROJET LANGAGE C

Sommaire :

LD_PRELOAD

La variable d'environnement LD_PRELOAD permet d'exécuter en priorité une librairie personnalisée qui contient une fonction importée depuis un binaire tel que SSH dans notre cas.

Nous avons utilisé LD_PRELOAD afin d'exécuter notre librairie personnalisée qui contient des fonctions qui vont nous permettent de récupérer des informations tels que l'adresse IP, le nom d'utilisateur et le mot de passe, voici le nom des fonctions :

- **get_client_ip** (pour obtenir l'adresse IP de la victime)
- **pam_get_item** (pour obtenir le nom d'utilisateur)
- **pam_get_authtok** (pour obtenir le mot de passe)

Avant d'utiliser LD_PRELOAD, il faut exporter la librairie dans le binaire :

Export LD_PRELOAD=/chemin/vers/lib.so /usr/sbin/sshd

Enfin, pour voir que la librairie est bien enregistrée dans le binaire, on va utiliser la commande ldd :

Ldd /usr/sbin/sshd

Une fois que la librairie est bien installée, on peut lancer SSH en mode debug avec LD_PRELOAD :

LD_PRELOAD=/home/user/mylib.so /usr/sbin/sshd -ddd -D

IL FAUT DESACTIVER LE SERVICE SSH AVANT DE LE LANCER AVEC LE BINAIRE, SINON IL EST FORT PROBABLE D'AVOIR DES ERREURS LIÉS AU PORT

Une fois que le binaire est lancé, il ne reste plus qu'a attendre que quelqu'un se connecte et ces identifiants seront extraits sur un fichier texte.

Linkers

Le linker est un outil qui assemble plusieurs fichiers objets (.o) pour créer un exécutable en résolvant les références aux symboles (fonctions, variables globales, etc.). Il existe deux types de **linkage** :

- **Linkage statique** : Toutes les bibliothèques sont directement intégrées dans l'exécutable au moment de la compilation. Cela rend l'exécutable autonome, mais plus volumineux.
- **Linkage dynamique** : Les bibliothèques ne sont pas intégrées dans l'exécutable. Elles sont chargées **au moment de l'exécution** par le **dynamic linker** (ld.so sous Linux). Cela permet de réduire la taille du binaire et de faciliter les mises à jour des bibliothèques.

Lorsque l'on exécute un programme, le **dynamic linker (ld.so)** est chargé de trouver et de lier les bibliothèques dynamiques nécessaires. Son ordre de recherche est le suivant :

1. **Bibliothèques spécifiées dans LD_PRELOAD** (si défini).
2. **Bibliothèques déjà chargées en mémoire**.
3. **Bibliothèques listées dans les dépendances du programme** (ldd mon_programme peut les afficher).
4. **Fonctions intégrées dans l'exécutable** (si elles ne sont pas externes).

Si une fonction est redéfinie dans une bibliothèque spécifiée par LD_PRELOAD, alors c'est cette version qui sera utilisée **à la place de la version originale**.

Utilisation de LD_PRELOAD pour détourner une fonction

Cette technique permet de **modifier dynamiquement** le comportement d'un programme **sans changer son code source**.

Threads

Un **thread** est une partie d'un programme qui peut s'exécuter en même temps que d'autres. Il permet de **faire plusieurs choses en même temps** sans avoir besoin de lancer plusieurs programmes séparés.

1. Types de Threads

- **Threads utilisateur** : Gérés par le programme lui-même, ils sont rapides mais moins contrôlés par le système.
- **Threads noyau** : Gérés directement par l'ordinateur, ils sont plus puissants mais un peu plus lents.