Challenge2 : TomyRobot CTF

Pour la catégorie des challenges moyenne/difficile nous avons choisi TomyRobot CTF qui se fait en deux machines Debian. Pour parvenir à nos fins, il vous faudra capturer 6 flags à différents niveaux au cours de la résolution du challenge. En vous aidant par des hints qui permettent de vous aider afin de trouver les flags. Pour la première machine héberge un site web avec base de données et un site wordpress (un système de gestion de contenus (CMS)) que le développeur a oublié à l'héberger. Le site web est vulnérable au SQL Injection et le site de blog partage le mot de passe de admin login avec le site, qui peut être piraté par SQLi. Dans cette machine, nous cachons les flags de 1 à 3, et aussi l'information nécessaire pour accéder à la deuxième machine.

Pour la machine 2, vous pouvez obtenir dans le fichier known_hosts de la première machine mot de passe haché. Dans cette machine, il existe plusieurs utilisateurs, respectivement existe le flag.

Les étapes de la création :

La création de ce challenge s'est basée sur un ensemble des étapes cohérents :

1. création de deux machines Debian avec une adresse privé et une autre flottante qui sont disponible sur open stack.

2. Création d'un site Web hébergé par python dans le port 8000 contenant une fail de SQL injection qui permet l'accès à la base de donne et récupérer les logins et les mots de passe ainsi que le deuxième flag (flag2 comme nom de l'utilisateur et son contenu comme mot de passer).

3. Création d'un blog Wordpress hébergé dans apache à l'aide de Docker dont le login est le mot de passe sont ceux trouvés dans l'attaque SQL Injection précédente. Pour pouvoir se connecter, l'hacker dois aller à /login. Par suite, l'attaquant dois aller à la corbeille du site pour trouver le 3éme flag.

4. L'attaquant dois ensuite aller à la soit à la page de 404 ou celle d'ajout d'un plugin pour pouvoir uploader un script de reverse Shell dedans. L'hacker doit ensuite utiliser le reverse Shell pour avoir l'accès au serveur WordPress (machine 1) et trouver le 4éme flag.
Hachage du mot de passe en MD5(challenge@tomygrpCTF1_2022/2023 ==> 9b4d2280a5c530a3efbaae9c00eb0a3a).

5. Ensuite, l'attaquant cherche dans la machine un fichier nommé « backup-mdp.txt » qui contient le mot de passe de user2(dans machine 2), le 5éme flag ainsi qu'un indice pour aller chercher dans known_hosts. Cependant ce fichier est propriétaire au user1 et ne peut être lit que par lui. L'attaquant doit ensuite trouver un fichier nommé « normalFile.txt » qui contient le mot de passe de user1 haché en md5. Après avoir accédé au fichier backup-mdp.txt, l'attaquant doit aller chercher dans known_hosts pour avoir l'adresse IP de la deuxième machine et y accéder à l'aide du mot de passe trouvé auparavant.

6. L'hacker doit ensuite chercher dans les programmes cron et trouver une tâche lancée par user3 qui lance une sauvegarde de /home/user3/data (contenant flag6.txt) chaque 5 minutes dans /home d'après un script backup.sh qui se trouve dans /home/user3. Le fichier flag6.txt ne peut être lit que par user3, la solution sera de modifier le fichier backup.sh pour pouvoir copier le contenu du flag6.txt dans un autre fichier que l'hacker peut y accéder.

Les flags

Flag 1

Md5(This is the first one) = f30c6e73b90818e134aec6d3fa28927d

Flag 2

Md5(Ohoh! The second one) = b1d2914

Flag 3

Md5(Here we go!) = 2afe596473c537a867ab734afa00b0fe

Flag4

Md5(This is the fourth one) = 8f3dbcbb884d161fb6eade138ed8c32e

Flag 5

Md5(You are genius!) = 08a5c806df767c60c7c42caf0e8f7ab8

Flag 6

Md5(Finally the last one) = fb826bcae2e33538bc3e3d430430ae32