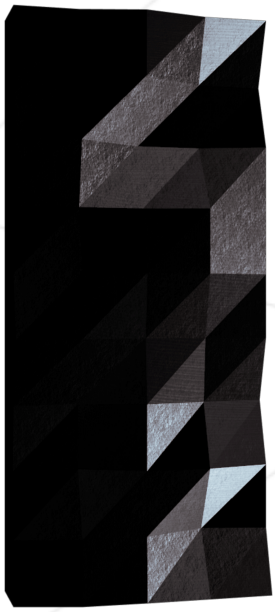

Bootcamp Cybersecurité | 42

Inquisitor

Résumé: Empoisonnement ARP.

Version: 1

Table des matières

Chapitre I

Prologue

Une innocente victime d'empoisonnement ARP.

Chapitre II

Introduction

Le *modèle OSI* est l'architecture suivie par les réseaux informatiques de toute la planète. Il se compose de 7 couches, chacune d'entre elles comportant des risques et étant exposée à différents types de vulnérabilités et de formes d'exploitation.

Au niveau du réseau, il existe des éléments chargés de décider où diriger le trafic. Chaque réseau local possède une passerelle par défaut, qui reçoit le trafic externe et le distribue parmi ses noeuds. Cette passerelle est généralement connue sous le nom de *router*.

Si un nœud du réseau est capable de se faire passer pour la passerelle, il peut prendre le contrôle du trafic, l'intercepter et décider à qui le transmettre, tout en étant capable de le modifier ou de le bloquer.

L'usurpation ARP peut également être utilisée de manière légitime, par exemple pour rediriger les nouvelles connexions vers une page d'enregistrement avant d'utiliser un réseau, comme cela est courant dans les réseaux portes ouvertes des aéroports, des cafétérias et d'autres lieux publics.

Chapitre III

Partie obligatoire

Puisque travailler avec des *sockets bruts* nécessite des autorisations de bas niveau, dans ce projet, vous travaillerez dans un conteneur ou une machine virtuelle.

En cas d'utilisation d'une machine virtuelle, vous n'inclurez dans le référentiel de livraison qu'un fichier `signature.txt` avec la somme de contrôle du .vdi de votre machine. Pendant l'évaluation, la signature du référentiel sera comparée à la signature réelle de votre machine, et si elles ne correspondent pas, votre note ce sera un 0.

Dans le cas où vous travaillez avec un ou plusieurs conteneurs, en plus du code de votre programme vous inclurez le Dockerfile ou le docker-compose.yaml ainsi qu'un script Bash appelé `start.sh` qui démarre l'ensemble de l'environnement sans intervention de l'utilisateur.

Vous allez créer un programme appelé `inquisiteur` avec les caractéristiques suivantes :

- Recevra quatre paramètres : <IP-src> <MAC-src> <IP-target> <MAC-target>
- Sera capable d'effectuer un empoisonnement ARP dans les deux sens (full duplex)
- Lorsque l'attaque est arrêtée (CTRL+C), les tables ARP sont restaurées.
- Ne fonctionne qu'avec les adresses IPv4.
- Le programme sera capable d'intercepter le trafic résultant de la connexion à un serveur FTP.
- Les noms des fichiers échangés entre le client et le serveur FTP s'affichent en temps réel.
- Le programme ne s'arrêtera jamais inopinément et traitera toutes les erreurs de saisie.

Vous utiliserez la bibliothèque `libpcap` pour renifler les paquets. Par conséquent, vous pouvez utiliser n'importe quel langage de programmation qui l'implémente (C, C++, Python, etc.).

Chapitre IV

Partie bonus

L'évaluation des bonus se fera `SI ET SEULEMENT SI` la partie obligatoire est `PARFAITE`. Dans le cas contraire, les bonus seront totalement `IGNORÉES`.

Vous pouvez améliorer votre projet grâce aux fonctionnalités suivantes :

- Mode "Verbose" (-v) qui montre tout le trafic FTP et pas seulement les noms de fichiers.

Chapitre V

Évaluations peer 2 peer

Ce projet sera corrigé par d'autres étudiants. Remettez les fichiers dans le dépôt Git et assurez-vous que tout fonctionne comme prévu.