Challenge FTP

March 21, 2022

1 Objectif de la SAE

Maintenant que vous êtes aguerris avec la manipulation des trames grâce aux outils Python et Scapy, nous allons pouvoir mettre en évidence les vunérabilités des protocoles réseau couremment utilisés.

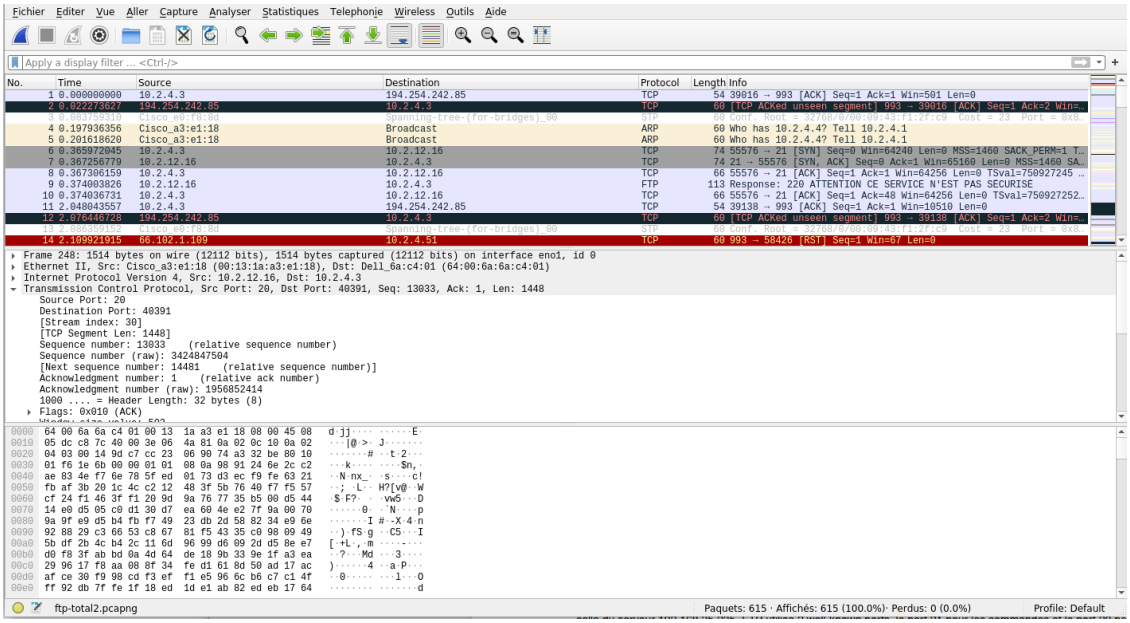
En cela nous allons travailler sur des protocoles comme FTP, TELNET, HTTP, … c'est à dire des protocoles qui n'utilisent pas de chiffrement, contrairement à SFTP, FTPS, SHTTP, HTTPS, SSH, … dont le 'S' (pour "Secure") signifie que les communications sont chiffrées … vous devriez assez vite en comprendre les avantages.

Ces connaissances devraient vous permettre de résoudre quelques énigmes proposées assez fréquemment dans le cadre de challenges de cybersécurité de type CTF "Capture The flag".

De temps à autre, ne soyez pas étonné si certains challenges "bonus" vous sont proposés … ils vous permettront d'acquérir quelques compétences supplémentaires tout en glanant quelques points en plus pour votre note de SAE.

2 Analyse d'une capture Wireshark d'un transfert FTP

Cette capture a enregistré le transfert de fichier entre un client et un serveur utilisant le protocole FTP. Vous trouverez suffisamment d'informations sur Internet pour comprendre le fonctionnement de FTP, somme toute assez simple:



2.0.1 1) Filtrage du fichier

Un premier travail sur ce fichier devrait vous permettre de ne filtrer que les trames correspondant à cet échange (attention, le filtre “FTP” n’est pas suffisant … ne serait-ce que pour conserver l’affichage des ouvertures de connexions SYN, SYN/ACK, ACK)

2.0.2 2) Analyse visuelle de la capture

Dans un deuxième temps, en parcourant la capture, vous devriez arriver à retrouver un certain nombre d’informations dans les différents en-têtes. Les adresses IP bien-sûr, mais également les ports (FTP utilise deux connexions différentes sur deux ports distincts. Vous expliquerez pourquoi il procède ainsi). Vous expliquerez également à quoi correspondent les requêtes applicatives que vous trouverez sans doute: USER, PASS, SYST, PORT, LIST … ainsi que les valeurs qu’elles prennent dans cette capture. Normalement, vous devriez retrouver un certain nombre d’informations intéressantes, voire … indiscrètes !

2.0.3 3) Extraction du fichier transféré

En utilisant les fonctionnalités de Wireshark, vous allez très probablement réussir à reconstituer le fichier qui a été tranféré. Vous expliquerez la méthode sur votre compte-rendu.
En lisant ce fichier, il est fort probable que vous compreniez très vite l’intérêt d’utiliser du SFTP ou FTPS au lieu du FTP classique … (SFTP et FTPS chiffrent les données avant de les tranférer, si elles sont interceptées … les choses deviennent plus compliquées pour le hacker :-)).
Conservons ce premier challenge “bonus” pour plus tard …

2.0.4 4) Développement d’un “Exploit” en Python

Pour terminer cette manipulation sur FTP, l’objectif final est d’écrire un programme python qui sera capable, en sniffant le réseau, d’intercepter une connexion FTP entre un client et un serveur, de récupérer les identifiants de connexion et d’enregistrer les données tranférées. Ce n’est pas

si compliqué que cela, vous avez toutes les billes pour le faire si vous avez fait correctement les premières étapes proposées au début de cette SAE.

Voici une petite astuce cependant: l'adresse IP et le numéro de port suggérés par le client apparaissent sur six octets, 4 pour l'adresse IP et 2 pour le port: $P1$ et $P2$. La lecture est simple pour l'adrese IP (1 octet par valeur), mais pour le port, étant donné que $P1$ représente le poids fort, un petit calcul est nécessaire pour obtenir sa valeur. En résumé:

Addresse IP : IP1.IP2.IP3.IP4, Numéro de port TCP : $P1 \times 256 + P2$

2.0.5 Bonus: Dechiffrement du message

Dans le cadre d'un transfert SFTP ou FTPS, **toutes** les données sont chiffrées, y compris les identifiants qui, comme vous venez de le constater, sont transférés en clair avec FTP. Bien évidemment, le chiffrement utilisé (AES ou Triple DES) est autrement plus résistant que celui qui est proposé ici … néanmoins il faut un début à tout !
En vous penchant sur ce 1er challenge "bonus", vous pourrez acquérir les bases des algorithmes de chiffrement dits "par substitution".