

Visualisation de traces réseaux

Thibault LENGAGNE et Nicolas NGÔ-MAÏ

Centrale Supélec - Campus de Rennes

4 février 2016

- 1 Retour sur les choix technologiques
- 2 Détails techniques
- 3 Démonstration
- 4 Travail à venir
- 5 Conclusion

- Le but est de créer un outil destiné aux pentesters, permettant d'analyser efficacement une trace réseau.
- L'attaquant dispose d'une trace réseau mais n'a aucune connaissance de ce réseau.
- Un fichier pcap devient rapidement très volumineux et son analyse est laborieuse, nécessitant l'utilisation de plusieurs outils.

L'attaquant espère soutirer des informations sur le réseau :

- Les adresse IP identifiées (+ géolocalisation, résolution DNS)
- Les protocoles utilisés (en particulier non chiffrés ou mal configurés)
- Les noeuds importants du réseau (serveur de fichier, DNS, LDAP...)
- Extraire des traces réseaux filtrées, extraire les données sensibles

- Nous voulions à l'origine interfacier plusieurs outils (Ettercap, ChaosReader, tcptrace...)
- Finalement, Scapy nous permet de manipuler la trace réseau de façon satisfaisante.
- Enfin, le choix initial de PyQt a été remplacé par une interface web.

- Pour stocker les résultats, nous avons choisi d'utiliser une base de donnée PostGreSQL
- Nous avons trouvé un outil très puissant pour faire de la visualisation : D3.js
- Nous avons donc choisi de mettre en place un serveur web en python (Flask)

Nous avons d'ores et déjà remplis les objectifs suivants :

- Extraction de sessions, des utilisateurs, et des protocoles
- Création de statistiques, extraction des données en clair
- Visualisation en barre parallèle

- 1 Retour sur les choix technologiques
- 2 Détails techniques**
- 3 Démonstration
- 4 Travail à venir
- 5 Conclusion

Quatres tables dans la base de données :

Packet
<input type="checkbox"/> id
<input type="checkbox"/> hostSrc
<input type="checkbox"/> hostDest
<input type="checkbox"/> portSrc
<input type="checkbox"/> portDest
<input type="checkbox"/> protocol
<input type="checkbox"/> data
<input type="checkbox"/> timestamp
<input type="checkbox"/> sessionId

Session
<input type="checkbox"/> id
<input type="checkbox"/> hostSrc
<input type="checkbox"/> hostDest
<input type="checkbox"/> portSrc
<input type="checkbox"/> portDest
<input type="checkbox"/> protocol

Stat
<input type="checkbox"/> id
<input type="checkbox"/> name
<input type="checkbox"/> value
<input type="checkbox"/> comment

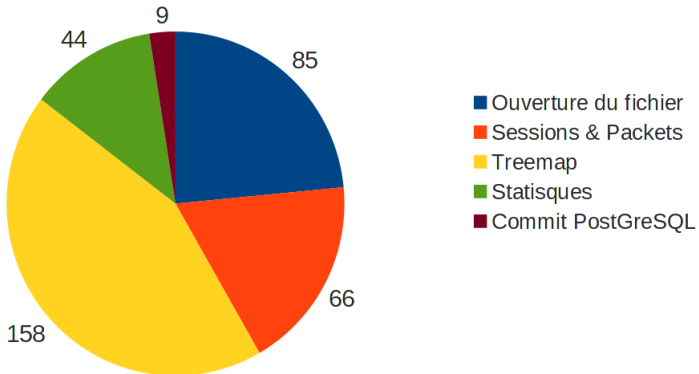
User
<input type="checkbox"/> id
<input type="checkbox"/> address
<input type="checkbox"/> exchanged

- La librairie Scapy permet de lire nos fichiers, puis de manipuler les packets.
- Une fonction parcourt tous les paquets. Les tables User et Stats sont alimentées
- Une fonction parcourt tous les sessions. Les tables Session et Trames sont alimentées

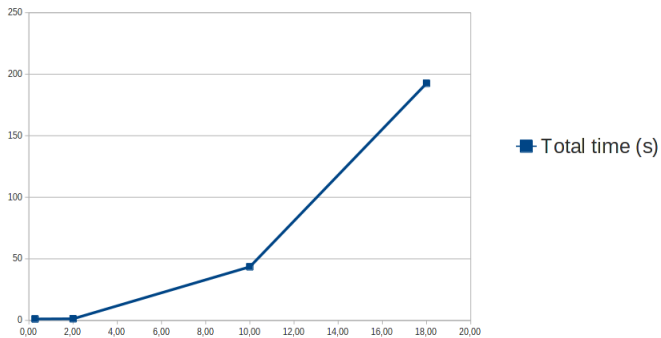
Performances du parser

- Les fichiers pcap peuvent atteindre plusieurs Giga assez rapidement.
- L'écriture dans la base de donnée étant couteuse, le programme écrit dans la base de donnée une seule fois, à la fin de la collecte des données

Analyse d'un fichier de 18 MO



Temps de Parsing = $f(\text{Taille})$



- 1 Retour sur les choix technologiques
- 2 Détails techniques
- 3 Démonstration**
- 4 Travail à venir
- 5 Conclusion

Le projet est disponible sur

<https://github.com/lechinois/Pcap-visualization-project>

Après avoir suivi la procédure d'installation, on démarre le serveur pour accéder à l'interface web

- 1 Retour sur les choix technologiques
- 2 Détails techniques
- 3 Démonstration
- 4 Travail à venir**
- 5 Conclusion

Objectifs à remplir

- Ajouter les différents filtres possibles
- Extraire d'un pcap a partir des trames filtrées
- Extraire les données non chiffrées des protocoles SMTP,IMAP,POP,LDAP
- Ajouter la résolution DNS
- Ajouter d'autres mode de visualisation (carte des IP,..)

- 1 Retour sur les choix technologiques
- 2 Détails techniques
- 3 Démonstration
- 4 Travail à venir
- 5 Conclusion**

Merci de votre attention !