[Réseau]

- [Forensic] TShark
- [Forensic] Analyse IPs / domaines
- [Forensic] NetworkMiner
- [Forensic] A-Packets

[Forensic] TShark

Introduction

Cet outil permet d'analyser en ligne de commande des fichiers PCAP un peu à la manière de Wireshark. L'avantage est que vous pouvez utiliser les avantages de bash pour filtrer et effectuer des opérations avancées assez simplement.

Cheat-sheet

IPs connexions sortantes TCP

Si vous souhaitez récupérer les IPs des connexions sortantes TCP en excluant les IPs privées et les IPs appartenant à Akaimai (connexions microsoft légitimes), vous pouvez utiliser la commande suivante :

```
tshark -r 2024-11-26-traffic-analysis-exercise.pcap -T fields -e ip.dst -Y "tcp" | sort -u | grep -Ev "^(10\.|172\.(1[6-9]|2[0-9]|3[0-1])\.|192\.168\.)" | while read ip; do nslookup "$ip" | grep -qi "akamaitechnologies.com" || echo "$ip"; done
```

> Vous pouvez remplacer l'argument -Y "tcp" par**-Y "http"** si vous souhaitez uniquement les connexions HTTP.

IPs connexions entrantes TCP

```
tshark -r 2024-11-26-traffic-analysis-exercise.pcap -T fields -e ip.src -Y "tcp" | grep -E "^(10\.|172\.(1[6-9]|2[0-9]|3[0-1])\.|192\.168\.)" | sort -u
```

> Vous pouvez remplacer l'argument -Y "tcp" par **-Y "http"** si vous souhaitez uniquement les connexions HTTP.

Connexions DNS

```
tshark -r 2024-11-26-traffic-analysis-exercise.pcap -Y "dns.qry.name" -T fields -e dns.qry.name | sort -u
```

Ports de connexion

Pour regarder les ports TCP les plus utilisés :

```
tshark -r 2024-11-26-traffic-analysis-exercise.pcap -T fields -e ip.src -e ip.dst -e tcp.dstport -Y "tcp" | sort -u
```

Pour filtrer selon certains ports spécifiques :

```
tshark -r 2024-11-26-traffic-analysis-exercise.pcap -T fields -e ip.src -e ip.dst -e tcp.dstport -Y "tcp" | sort -u | grep -iE "\s80|\s443|\s88|\s3389\s445"
```

Extraire les fichiers

Pour récupérer tous les fichiers ayant transités via **HTTP** et supprimer les fichiers sans extension (faux-positifs) :

```
tshark -r 2024-11-26-traffic-analysis-exercise.pcap --export-objects "http,extracted_files" && find extracted_files -type f ! -name "*.*" -delete
```

> Les fichiers seront disponibles dans le répertoire ./extracted_files

Scanning

Pour détecter un scan de ports, vous pouvez observer s'il y a un multitude de connexions SYN :

```
tshark -r 2024-11-26-traffic-analysis-exercise.pcap -Y "tcp.flags.syn == 1 and tcp.flags.ack == 0" -T fields -e ip.src -e tcp.dstport | sort | uniq -c | sort -nr | head -20
```

Trouver les fichiers ayant transités par SMB

```
tshark -r 2024-11-26-traffic-analysis-exercise.pcap -Y "smb2" -T fields -e smb2.filename | sort -u
```

> Vous trouverez aussi des registres modifiés via SMB !

Trouver les fichiers ayant transités par FTP

```
tshark -r 2024-11-26-traffic-analysis-exercise.pcap -Y "ftp.request.command" -T fields -e ftp.request.command -e ftp.request.arg | sort -u
```

Headers HTTP liés à une IP

```
tshark -r 2024-11-26-traffic-analysis-exercise.pcap -Y "ip.src == 194.180.191.64 || ip.dst == 194.180.191.64 && http" -V | grep -E "^[[:space:]]+[A-Za-z-]+:"  | sed '/^[[:space:]]*Host/ i\'$'\n'
```

Extraire les mots de passe en clair

```
tshark -r fichier.pcap -Y 'http.authbasic' -T fields -e http.authorization
```

```
tshark -r fichier.pcap -Y 'ftp.request.command == "USER" or ftp.request.command == "PASS"' -T fields -e ftp.request.arg
```

```
tshark -r fichier.pcap -Y 'telnet' -T fields -e telnet.data
```

> Pour extraire des secrets, **NetworkMiner** semble plus performant.

[Forensic] Analyse IPs / domaines

Introduction

Lors de l'analyse de vos trames, il vous sera utile d'identifier les IPs et les domaines notamment pour déterminer s'ils sont malveillants et s'il s'agit d'un serveur de l'attaquant.

Cheat-sheet

Trouver le provider d'une IP

```
whois <IP> | grep -i "orgname" | cut -f2 -d':' | sed 's/^[[:space:]]*//'
```

Retirer les IPs selon le provider

```
cat ips.txt | while read ip; do if ! whois "$ip" | grep -iE "orgname|role" | grep -iq "microsoft"; then echo "$ip"; fi; done
```

> Vous pouvez filtrer sur **Microsoft** mais aussi **Akamai** ou **Cloudflare** en modifiant le grep. Cela retirera ces IPs de la liste.

> Vous pouvez remplacer le *cat ips.txt* par une commande **tshark** ou autre, cela fonctionnera parfaitement.

Analyse virustotal

Pour analyser une IP :

```
curl -s -X GET "https://www.virustotal.com/api/v3/ip_addresses/194.180.191.64" -H "x-apikey: API_KEY" | jq '.data.attributes.last_analysis_stats'
```

Pour analyser plusieurs IPs :

```
cat ips.txt| while read -r ip; do curl -s -X GET "https://www.virustotal.com/api/v3/ip_addresses/$ip" -H "x-apikey: 1e04da01b0aa70136ef46bfdf8302db049d5dbc78c9bd0f0a6f8cdf59597b6e7" | jq -r --arg ip "$ip" '"\($ip): \(.data.attributes.last_analysis_stats.malicious)"'; done
```

[Forensic] NetworkMiner

Introduction

Le logiciel **NetworkMiner** est disponible sur Windows et Linux pour analyser les fichiers PCAP. Il permet notamment d'afficher des informations comme les fichiers échangés, les secrets ou autre à travers des sous-menus.

Installation

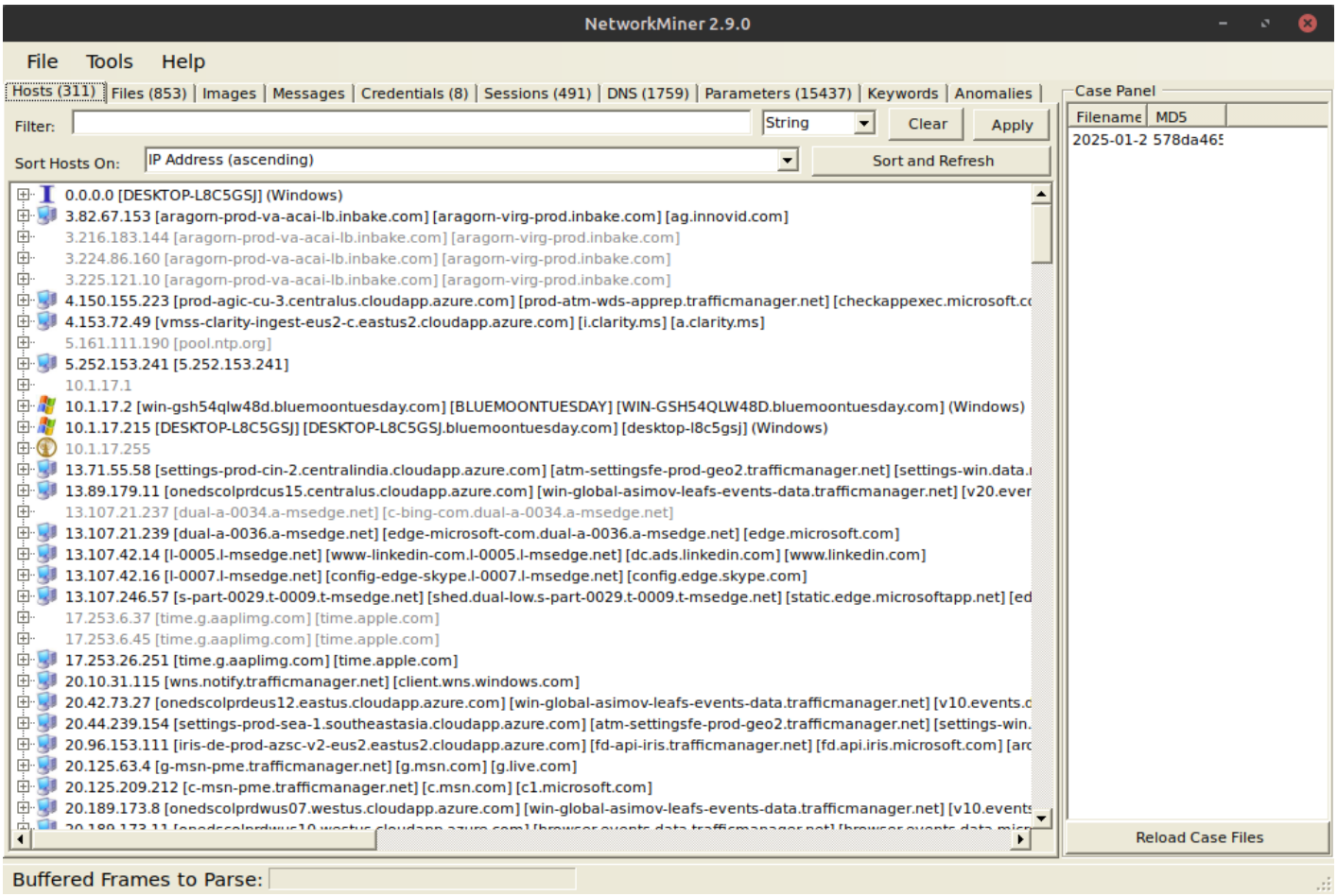
Debian

```
sudo apt install mono-devel && wget https://www.netresec.com/?download=NetworkMiner -O /tmp/nm.zip && sudo unzip /tmp/nm.zip -d /opt/ && cd /opt/NetworkMiner* && sudo chmod +x NetworkMiner.exe && sudo chmod -R go+w AssembledFiles/ && sudo chmod -R go+w Captures/
```

Puis lancez NetworkMiner :

```
cd /opt/NetworkMiner* && mono NetworkMiner.exe --noupdatecheck
```

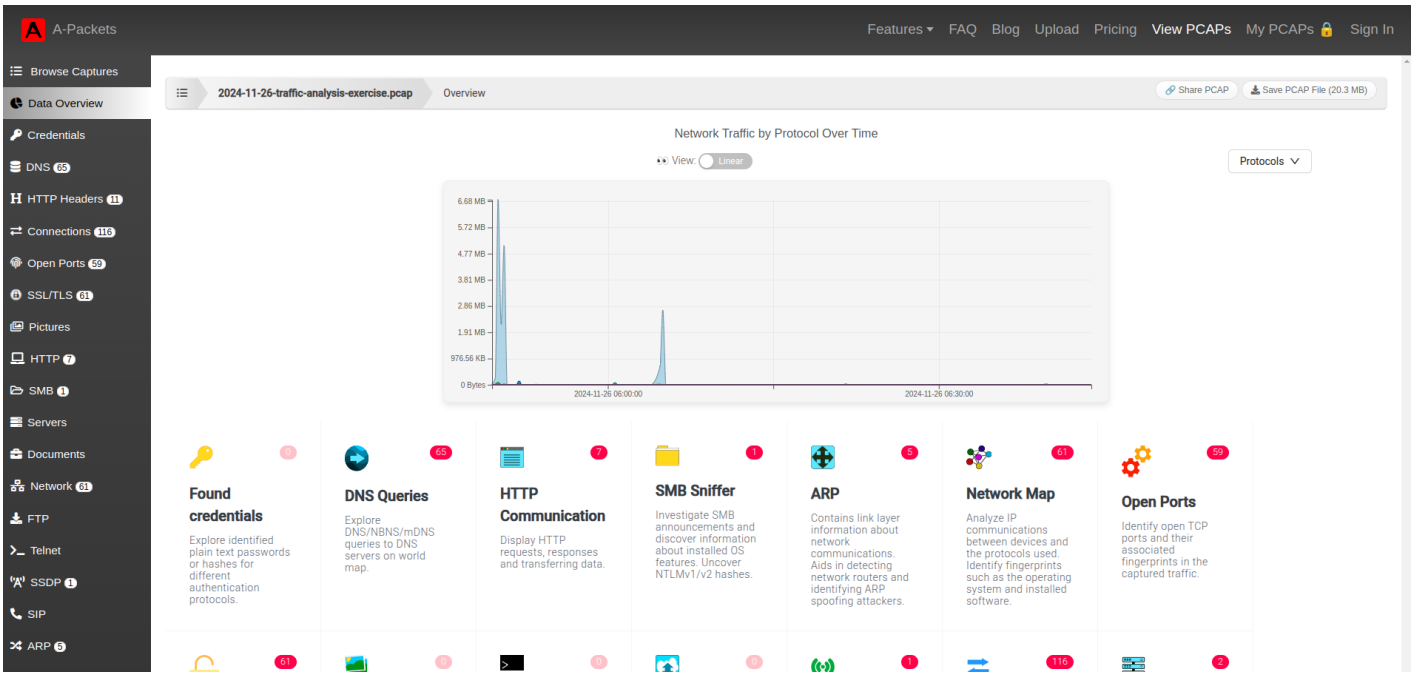
Interface



[Forensic] A-Packets

Introduction

L'outil en ligne **A-Packets** permet d'analyser des fichiers PCAP de moins de **25Mo**. La navigation pour trouver des indicateurs de compromission est très pratique et peut faire gagner énormément de temps pour une investigation contrairement à des outils comme Wireshark.



Lien

- https://apackets.com/