[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.

Revision #16
Created 12 March 2025 10:30:20 by Elieroc
Updated 12 March 2025 18:00:00 by Elieroc