



**POLYTECHNIQUE  
MONTRÉAL**

UNIVERSITÉ  
D'INGÉNIERIE

## Consigne

- L'ensemble des réponses doivent être documentées et appuyées de captures d'écran. Même si la réponse est correcte si elle n'est pas documentée votre note sera de **0 point**.
- Vous devez faire preuve d'analyse et de recherche dans les réponses que vous fournissez.
- Veuillez soumettre vos devoirs sous forme de fichiers PDF ou DOC et présenter vos réponses sous les questions ci-dessous en les copiant telles quelles avec leur numéro de question.

## Exercice No 1 (20p)

En utilisant les instructions du **laboratoire 1 - cours 10**, faites une analyse des PDF3 et PDF4, disponibles dans le laboratoire du cours 10

Commençons par le fichier PDF3 qui est *Research Paper on Nuclear Posture Review 2010.PDF* que j'ai renommé PDF3 pour que ce soit plus facile de le manier dans la ligne de commande. Même chose avec le pdf dans folder PDF4 je l'ai renommé PDF4

Nous commençons avec PDF3, et nous runnons la commande dans le lab pdf parser

```
.PDF > parser3.txt  
remnux@remnux:~/Cours10PDF/WARNING/PDF3$ pdf-parser.py PDF3.PDF > parser3.txt  
remnux@remnux:~/Cours10PDF/WARNING/PDF3$
```

Recent

Starred

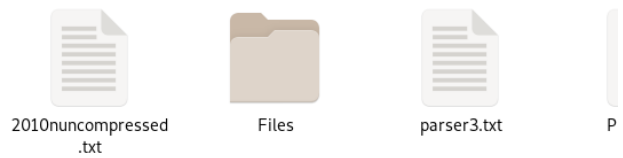
Home

Documents

Downloads

Music

Pictures



1 parser3.txt

```
PDF Comment '%PDF-1.6%'
obj 1 0
Type: EmbeddedFile
Referencing:
Contains stream
<<
  /Filter /FlateDecode
  /Length 2060
  /Type EmbeddedFile
>>

obj 2 0
Type:
Referencing: 3 0 R

<<
  /V 1
  /Kids [3 0 R]
  /T (topmostSubform[0])
>>

obj 3 0
Type:
Referencing: 2 0 R, 4 0 R

<<
  /Parent 2 0 R
  /Kids [4 0 R]
  /T (Page1[0])
>>

obj 4 0
Type: /Annot
Referencing: 3 0 R, 3 0 R

<<
  /MK
  /F
  /A [0.0 1.0]
>>
  /TP 1
  /P 5 0 R
  /FT /Btn
  /TU (ImageField1)
  /FT 0.5.5.0
  /Parent 3 0 R
  /F 4
  /DA (/CourierStd 10 Tf 0 g)
  /Subtype /Widget
  /Type /Annot
  /T (ImageField1[0])
  /Rect [157.385 755.147 188.385 759.087]
>>

obj 5 0
Type: /Page
Referencing: 5 0 R

<<
  /Rotate 0
  /CropBox [0.0 0.0 612.0 792.0]
  /MediaBox [0.0 0.0 612.0 792.0]
  /Resources
    <<
      /XObject
    >>
    /Parent 6 0 R
  /Type /Page
  /PsetInfo null
>>

obj 6 0
Type: /Pages
Referencing: 5 0 R

<<
  /Kids [5 0 R]
  /Type /Pages
  /Count 1
>>

obj 7 0
Type: /Catalog
Referencing: 6 0 R, 8 0 R

<<
  /PageMode /UseAttachments
  /Pages 6 0 R
  /MarkInfo
    <<
      /Marked true
    >>
  /Lang (en-us)
  /AcroForm 8 0 R
  /Type /Catalog
>>

obj 8 0
Type: /Catalog
Referencing: 7 0 R, 8 0 R
PDF Comment '%EOF%'
```

```
obj 8 0
Type:
Referencing: 1 0 R, 2 0 R

<<
  /DA (/Helv 0 Tf 0 g)
  /XFA ([Template] 1 0 R)
  /Fields [2 0 R]
>>
```

```
obj 9 0
Type:
Referencing:
Contains stream

<<
  /Length 76812
>>
```

xref

trailer

```
<<
  /Size 10
  /Root 7 0 R
>>
```

startxref 79801

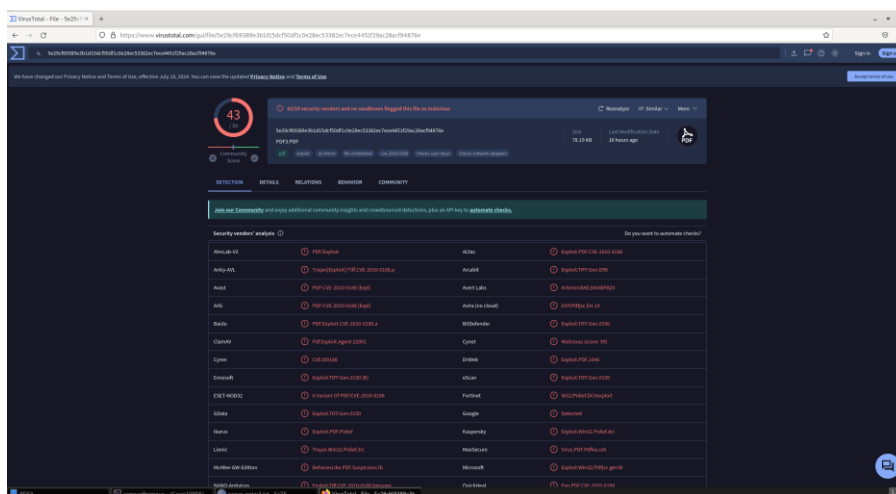
PDF Comment '%EOF%'

Jun 16 16:29
parser-water3.txt - SciTE

```
remnux@remnux: ~/Cours10PDF/WARNING/PDF3
remnux@remnux:~/Cours10PDF/WARNING/PDF3$ pdfid.py PDF3.PDF
PDFiD 0.2.8 PDF3.PDF
PDF Header: %PDF-1.6
obj          9
endobj       9
stream      2
endstream   2
xref         1
trailer      1
startxref    1
/Page        1
/Encrypt      0
/ObjStm       0
/JS           0
/JavaScript   0
/AA           0
/OpenAction   0
/AcroForm     1
/JBig2Decode  0
/RichMedia    0
/Launch       0
/EmbeddedFile 1
/XFA          1
/URI          0
/Colors > 2~24 0

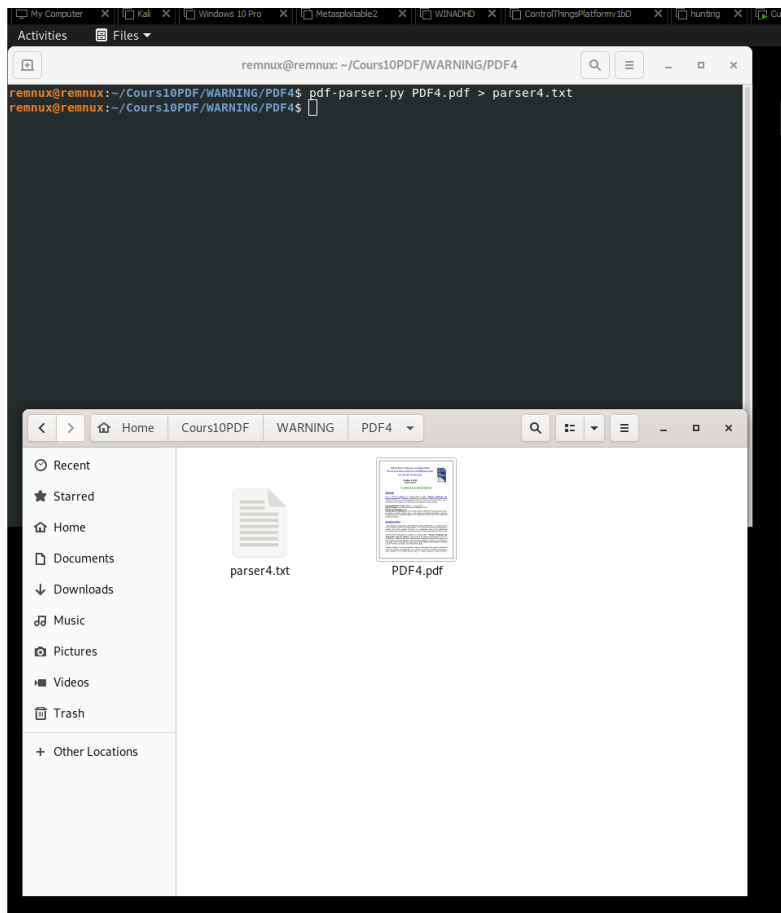
remnux@remnux:~/Cours10PDF/WARNING/PDF3$ pdf-parser.py water_update_part2.pdf > parser-water3.txt
remnux@remnux:~/Cours10PDF/WARNING/PDF3$ pdf-parser.py PDF3.PDF > parser-water3.txt
remnux@remnux:~/Cours10PDF/WARNING/PDF3$
```

C'est ce qui complete l'analyse de base par rapport au lab puisqu'il n'y a pas de code javascript. Nous pouvons aussi simplement utiliser *VirusTotal* pour les PDF qui semblent suspects.

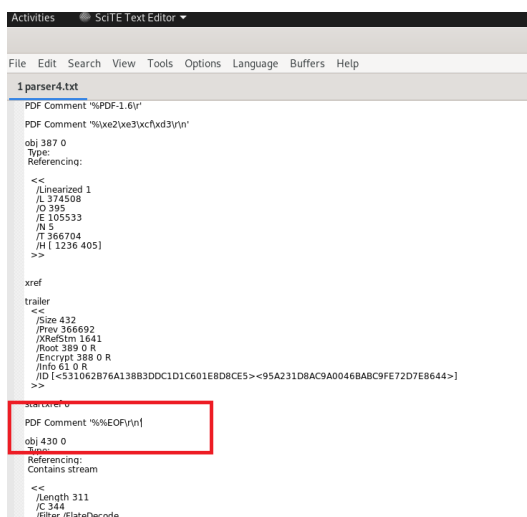


## Maintenant le PDF4.

Nous allons commencer par faire la même chose et parser le pdf à l'aide de la commande .py



En ouvrant le `parser4.txt` fichier, on peut constater que dans le trailer il y a déjà un comment `%%EOF`, qui est censé être plus bas dans le fichier, mais celui-ci en contient une couple, ce qui pourrait potentiellement être suspicieux selon la documentation dans le labo. Et il y a plusieurs hexa decimaux, beaucoup de caractère non lisible dans un mauvais format, possiblement du javascript, donc heureusement nous avons `pdfid.py` pour pouvoir voir s'il contient du javascript :





Et effectivement, nous avons du javascript dans celui-ci :

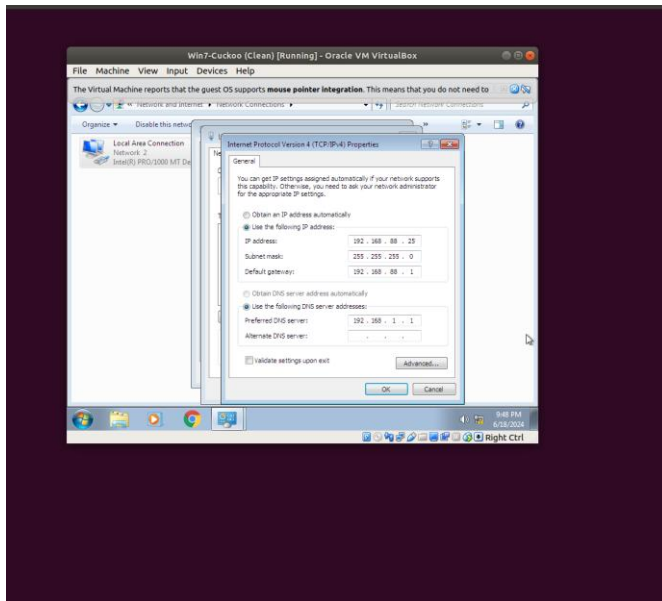
```
remnux@remnux: ~/Cours10PDF/WARNING/PDF4
remnux@remnux:~/Cours10PDF/WARNING/PDF4$ pdf-parser.py PDF4.pdf > parser4.txt
remnux@remnux:~/Cours10PDF/WARNING/PDF4$ pdftd.py PDF4.pdf
PDFiD 0.2.8 PDF4.pdf
PDF Header: %PDF-1.6
obj 106
endobj 106
stream 52
endstream 52
xref 2
trailer 2
startxref 2
/Page 5
/Encrypt 2
/ObjStm 4
/JS 1
/JavaScript 2
/AA 0
/OpenAction 0
/AcroForm 0
/JBIG2Decode 0
/RichMedia 0
/Launch 0
/EmbeddedFile 0
/XFA 0
/URI 0
/Colors > 2^24 0

remnux@remnux:~/Cours10PDF/WARNING/PDF4$
```

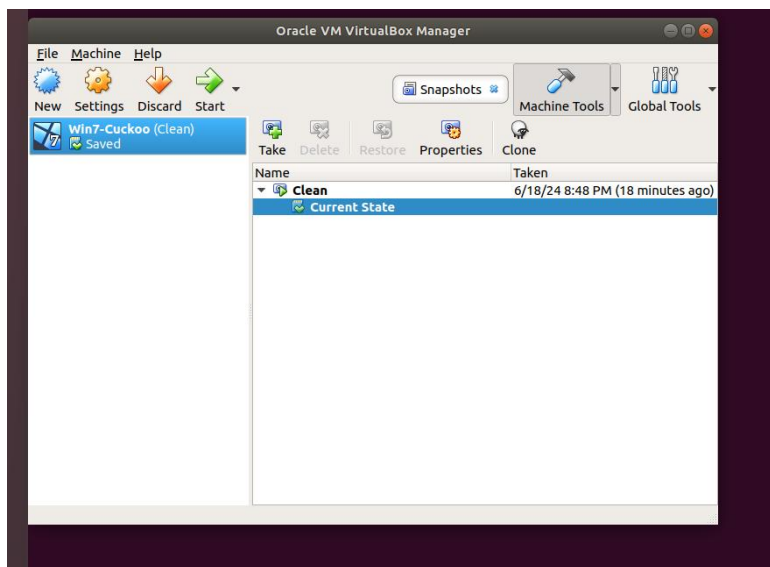
Donc tout ce qu'il reste à faire c'est effectuer les commandes pour extraire le javascript. Voir images finales ci-dessous

[illegible]

Nous commençons par changer les adresses IP de notre VM Win7-Cuckoo à ceux indiqués dans le lab.



On fait ensuite un snapshot qu'on intitule Clean



On modifie nos fichiers virtualbox.conf et cuckoo.conf

```
cuckoo@ubuntu02: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /home/cuckoo/.cuckoo/conf/cuckoo.conf Modified

# Cuckoo is capable of sending "developer feedback" to the developers so that
# they can more easily improve the project. This functionality also allows the
# user to quickly request new features, report bugs, and get in touch with
# support in general, etc.
enabled = no
name =
company =
email =

[resultserver]
# The Result Server is used to receive in real time the behavioral logs
# produced by the analyzer.
# Specify the IP address of the host. The analysis machines should be able
# to contact the host through such address, so make sure it's valid.
# NOTE: if you set resultserver IP to 0.0.0.0 you have to set the option
# 'resultserver ip' for all your virtual machines in machinery configuration.
ip = 192.168.88.1

# Specify a port number to bind the result server on.
```

```
[remotecontrol]
# Enable for remote control
enabled = yes
```

```
cuckoo@ubuntu02: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /home/cuckoo/.cuckoo/conf/virtualmachines/cuckoo1.conf Modified

# Virtualbox will bind the VRDP interface to the first
# controlports = 5000-5050

[cuckoo1]
# Specify the label name of the current machine as specified in
# VirtualBox configuration.
label = Win7-Cuckoo

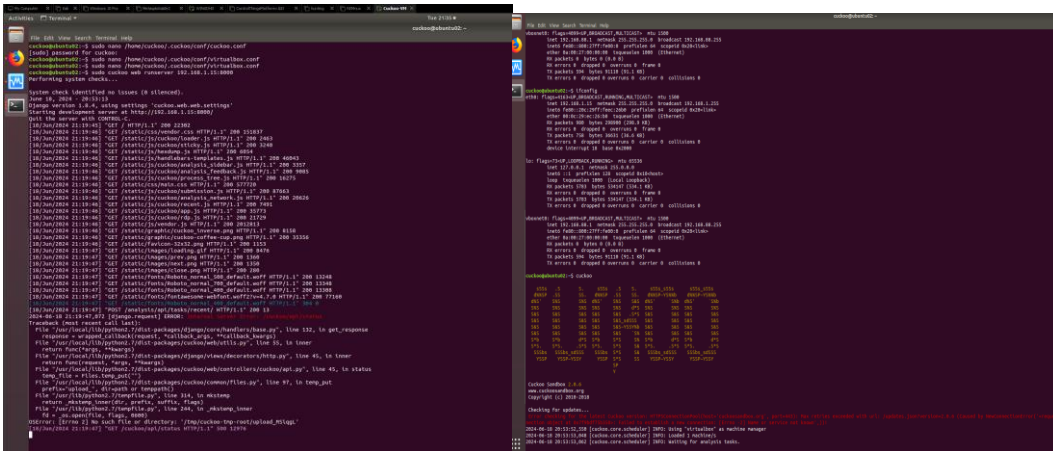
# Specify the operating system platform used by current machine
# [windows/darwin/linux].
platform = windows

# Specify the IP address of the current virtual machine. The IP
# address is valid and that the host machine is able to reach it,
# the analysis will fail.
ip = 192.168.88.25

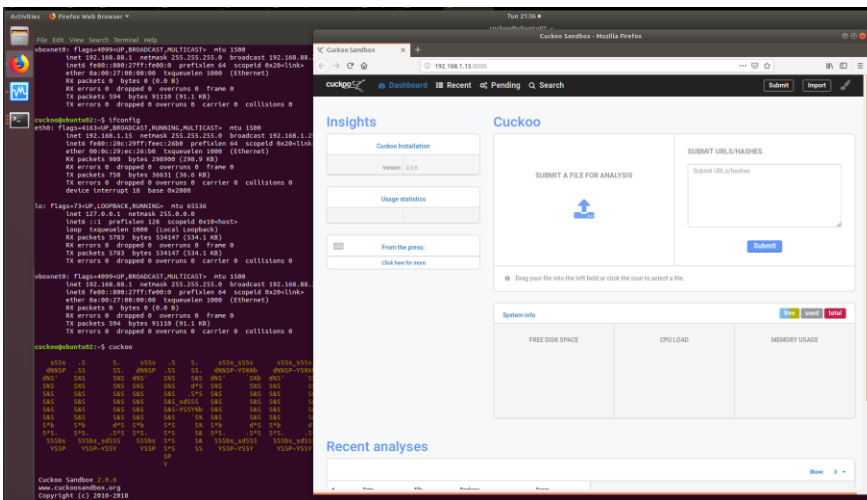
# (Optional) Specify the snapshot name to use. If you have a snapshot
```

Après, on peut simplement entrer pour vérifier dans le browser si le sandbox a bel et bien fonctionné, même si nous sommes encore sous l'adresse statique 192.168.1.15



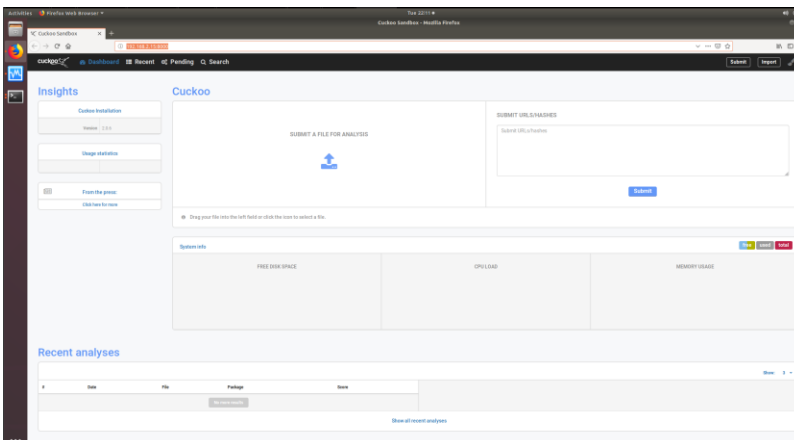


Maintenant, lorsqu'on va a 192.168.1.15:8000 dans notre browser, nous sommes prompté à l'interface web de cuckoo sandbox. Voir image ci-dessous



Maintenant, on doit simplement faire sûr qu'on peut communiquer avec mes autres VM pour qu'elles puissent accéder ce même interface. Pour cela, on doit changer notre adresse static dans /etc/network/interfaces.

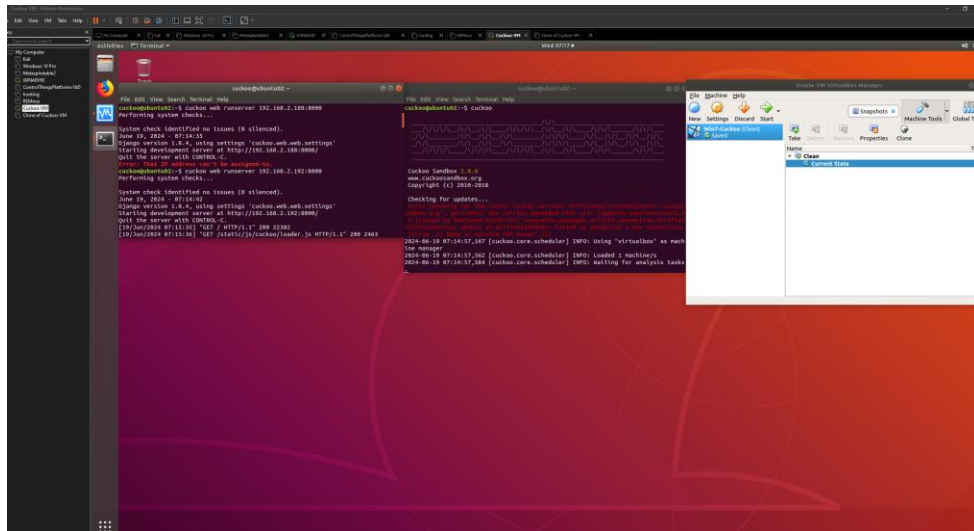
Voici le nouvel interface lorsque j'ai changé mon adresse IP a 192.168.2.15



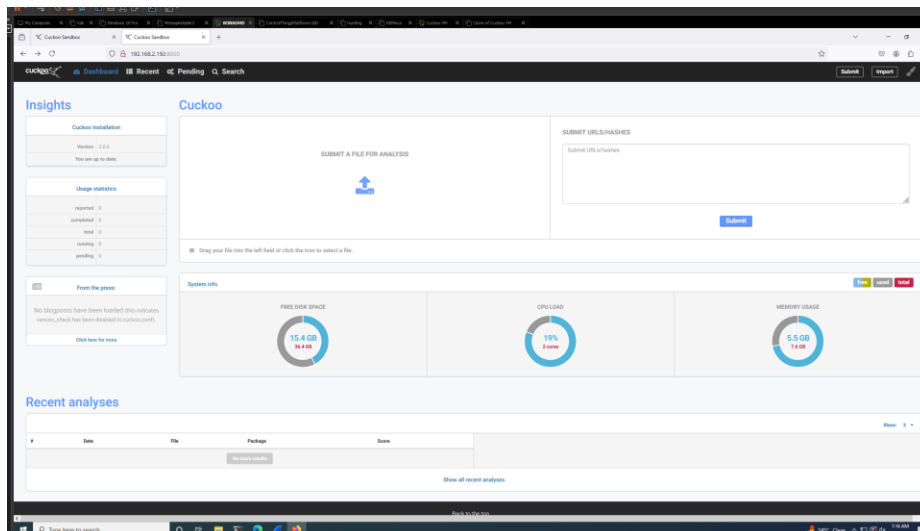
Maintenant, il suffit simplement d'aller dans la vm ADHD et accéder à l'interface web de cuckoo depuis notre serveur qu'on a créé dans la Cuckoo-VM.

**\*\*J'ai eu quelques problemes et j'ai dû reprendre un snapshot de ma VM, j'ai nommé le nouvel IP address à 192.168.2.192 (suivant le 2.187 de mon home network)\*\***

Voici mon terminal cuckoo qui relance le sandbox avec la nouvelle adresse IP, ainsi qu'une autre photo de ma WINADHD dans ma sandbox à partir du internet browser :



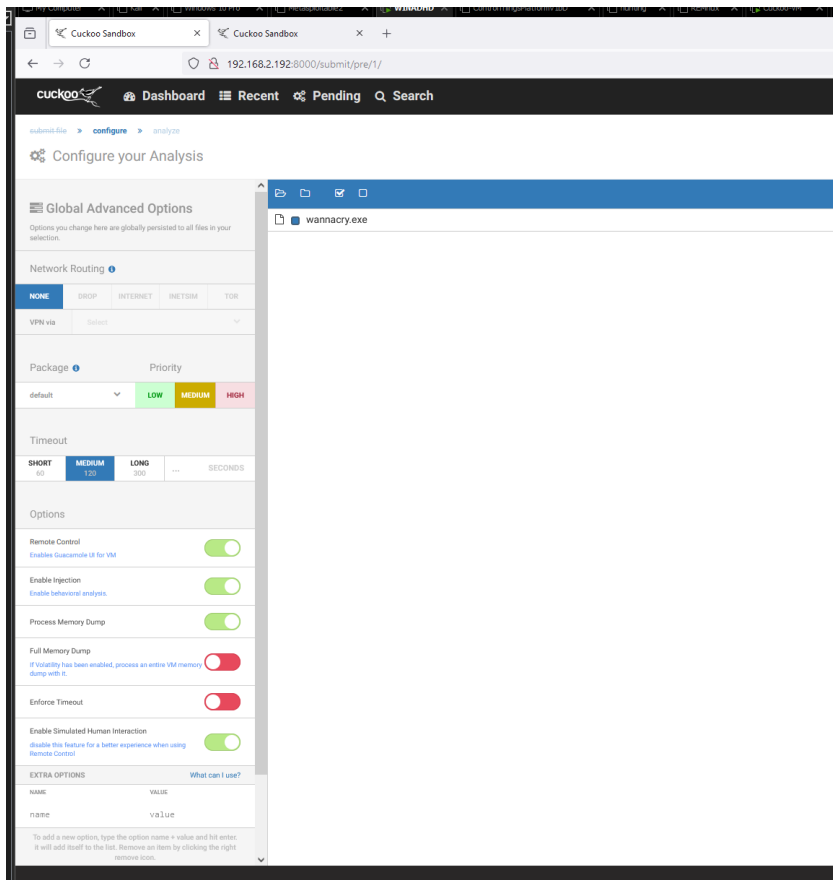
Et la WINADHD dans l'interface web :



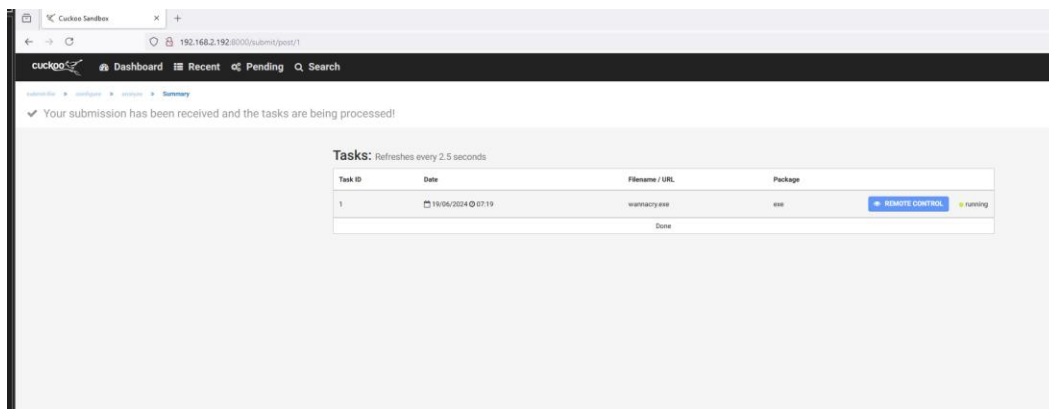
b) Similaire à l'exercice du laboratoire, analysez le malware wannacry.exe (5p)

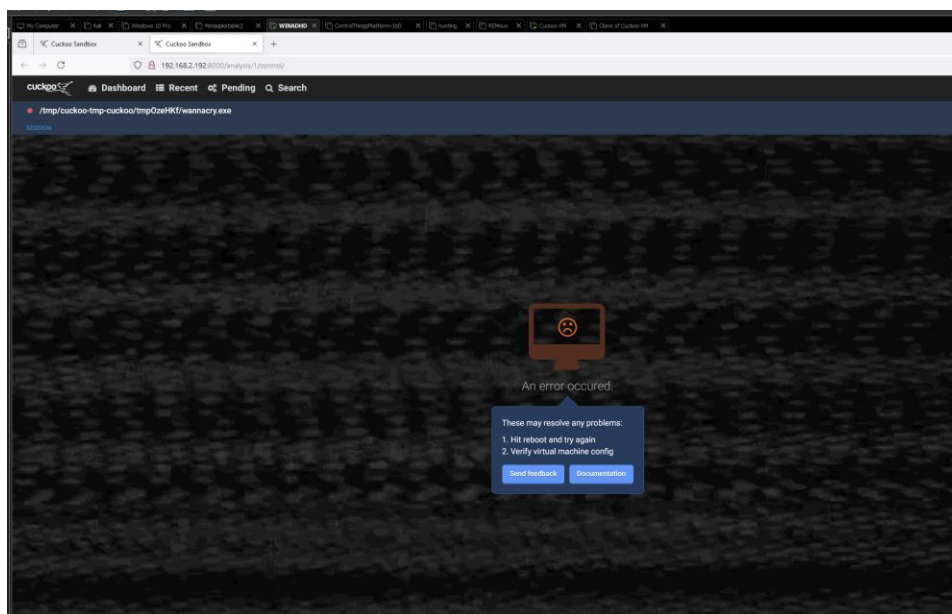
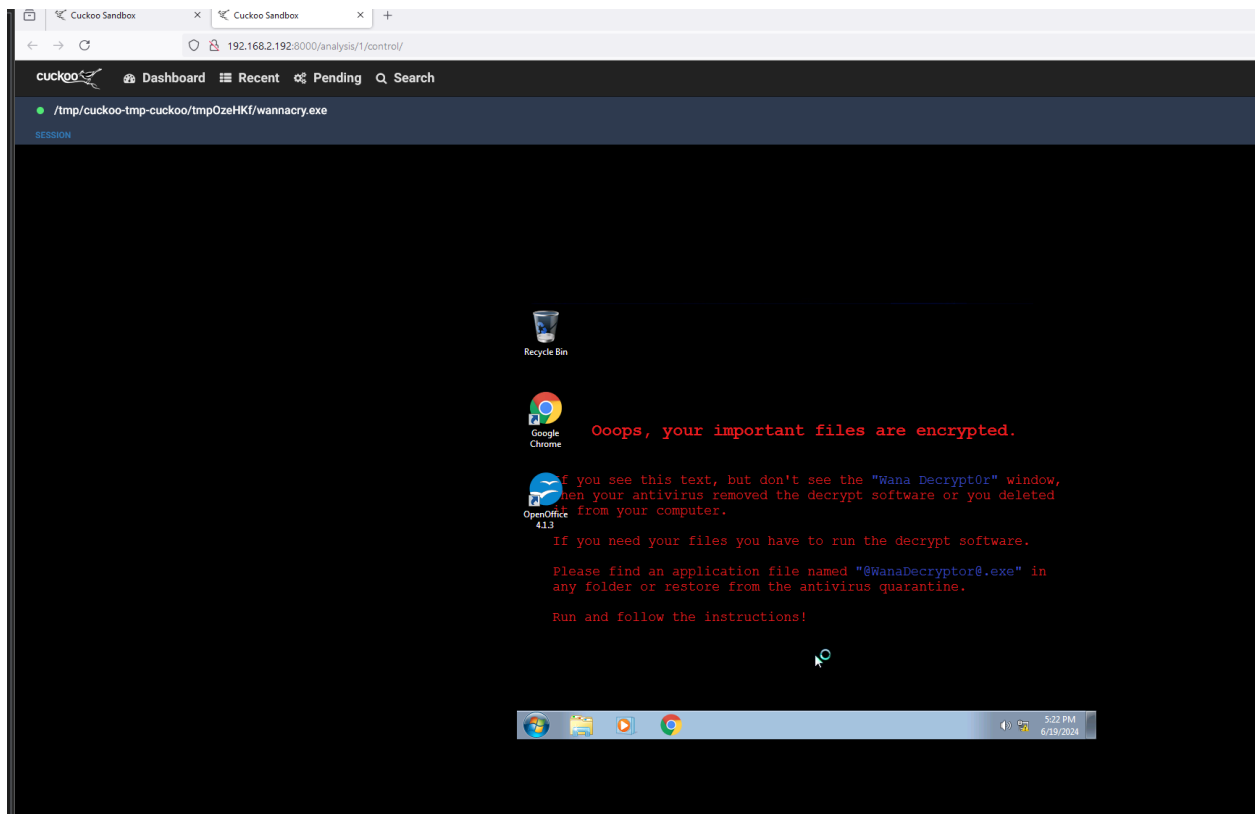
On load le fichier wannacry.exe dans la boîte à sable. On coche le remote control.

Voici le resultat initial :

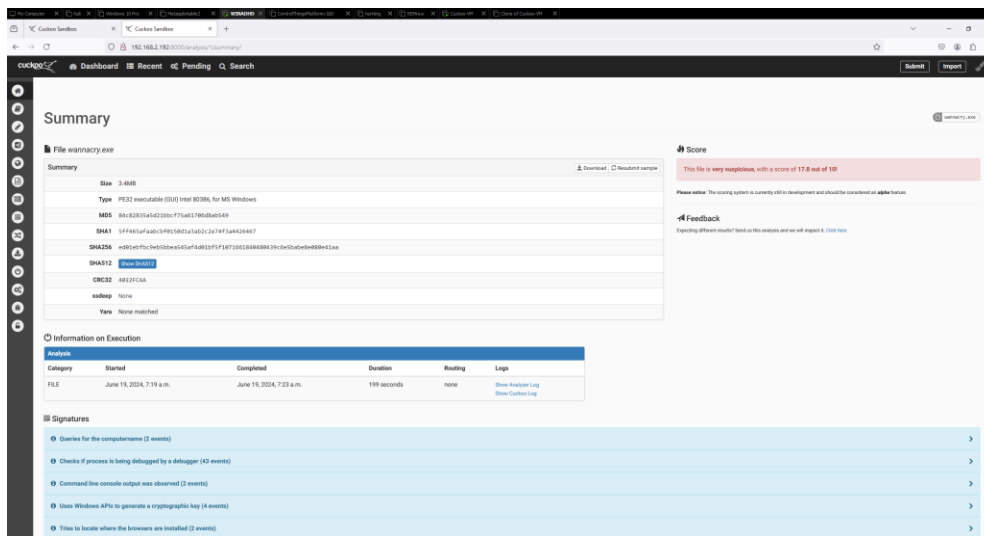


On clique sur remote control et on va etre prompt dans le simulateur avec le virus



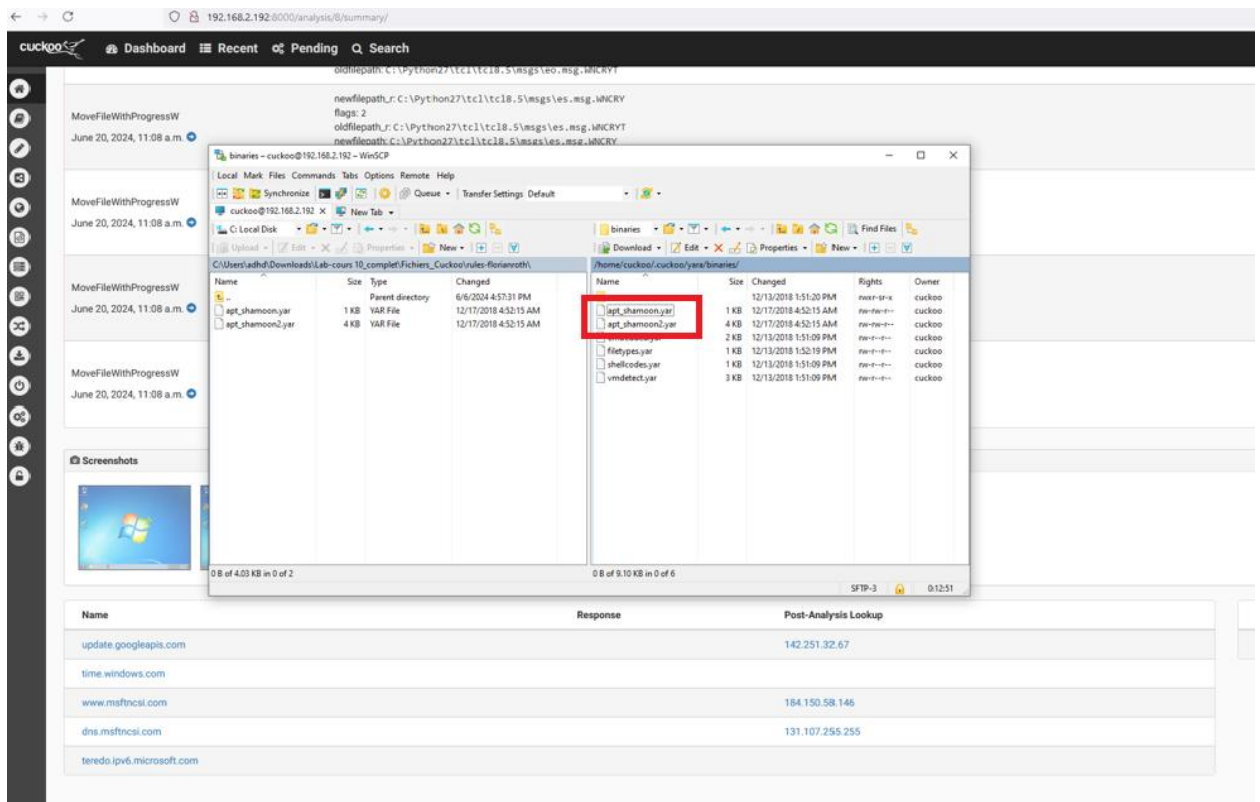


On fait "Show Report" et voici ce qu'on obtient :



c) Similaire à l'exercice du laboratoire, analysez le malware Shamoon, incluant la règle Yara (5p)

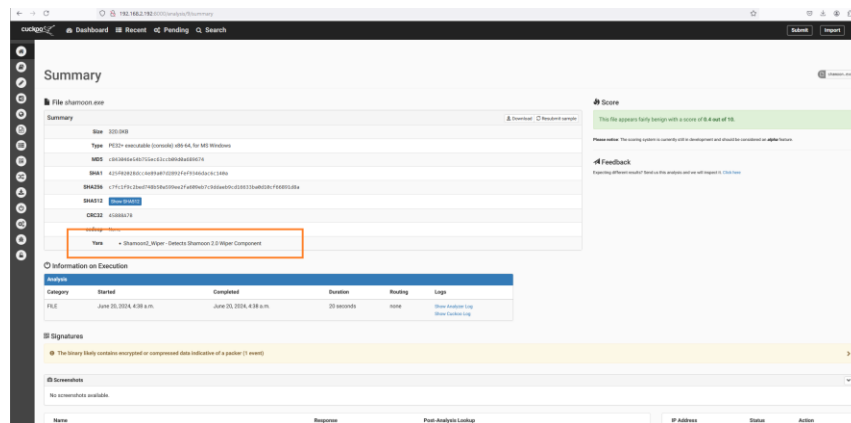
On commence par télécharger WinSCP, et on se connecte remotely a droite avec notre compte et IP de notre VM cuckoo. Par la suite, on transfère les fichiers dans notre répertoire de cuckoo comme indiqué dans le lab



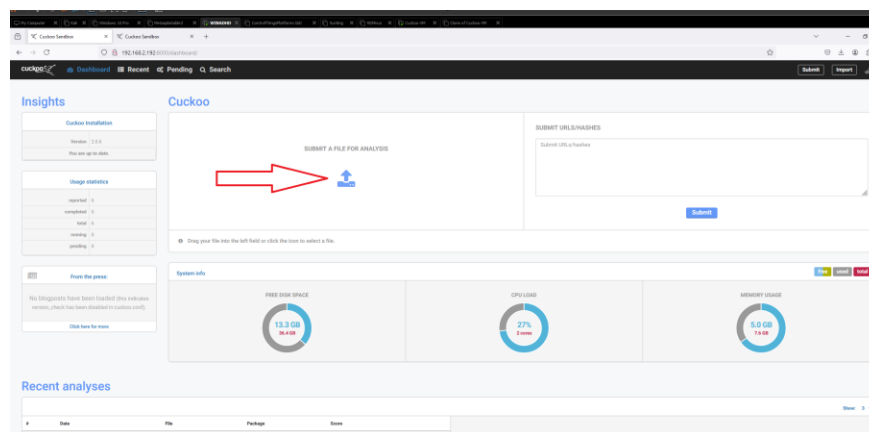
On redémarre par la suite notre serveur cuckoo qu'on a créé dans la VM-cuckoo.

Nous runnons à nouveau le malware, et nous pouvons voir que le score est relativement low comparé à avant.

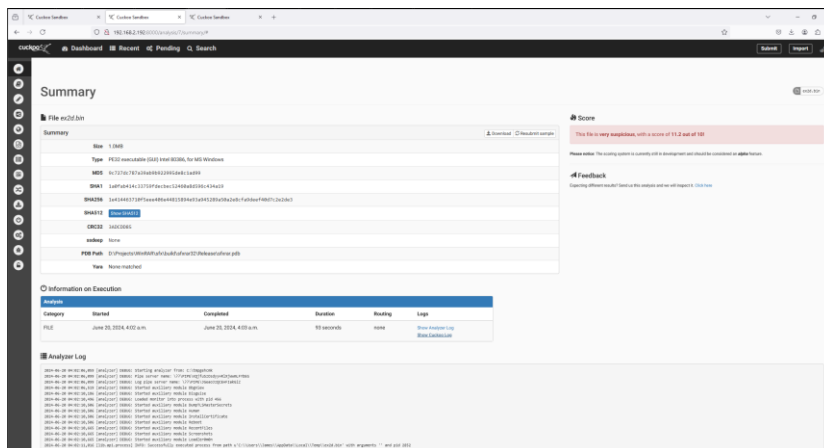
Pourquoi? Parce qu'avec cette règle, ce n'est pas suitable for proper sandbox operations. Mais, nous pouvons quand même voir dans le summary view que Yara a fait un hit. *Voir image ci-dessous.*



d) Analysez dans Cuckoo sandbox le fichier 2d.bin. Documentez vos trouvailles. (5p)

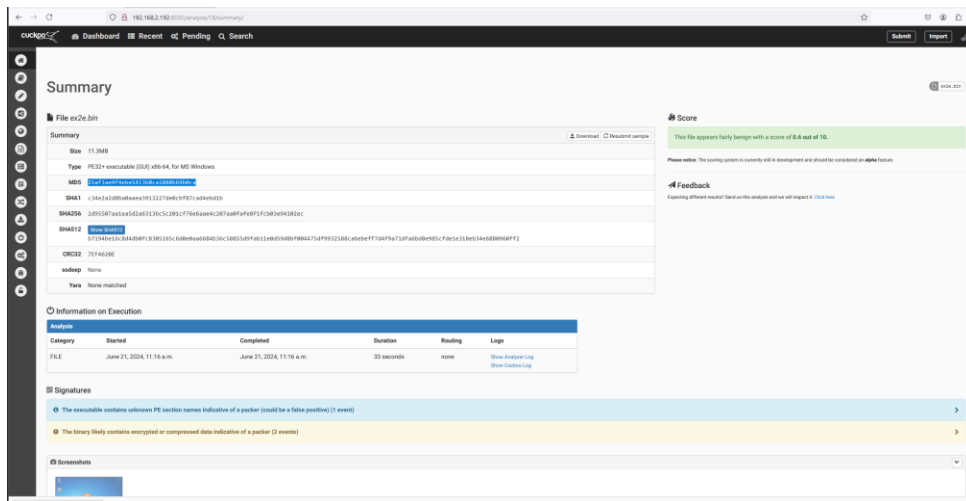


On upload le fichier ex2d.bin, on attend environ 1 minute et on a les résultats. Voici ce que nous avons.

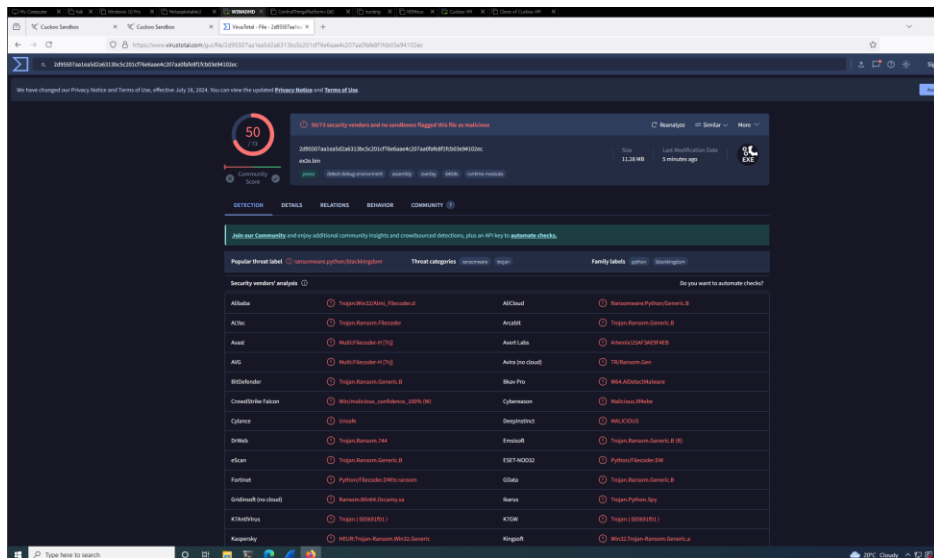
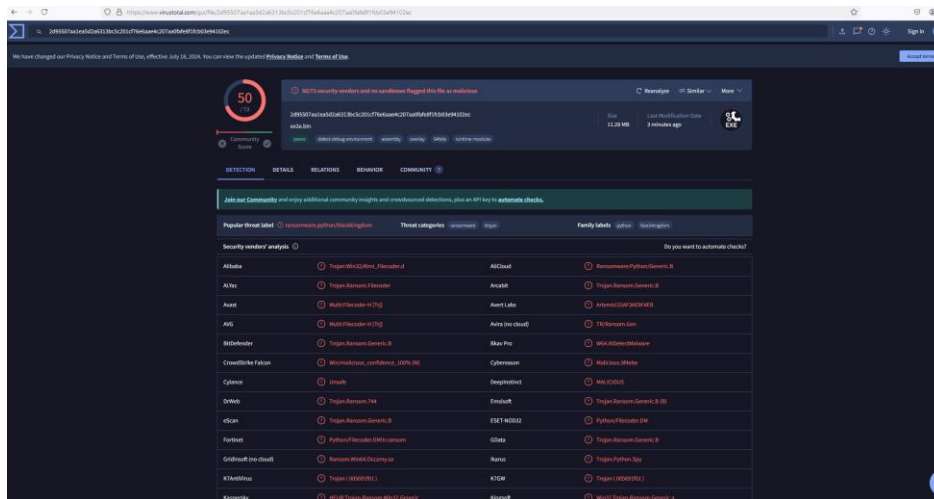


En cherchant les liens virustotal et github qui nous sont fournis, on peut déier très rapidement que c'est un Trojan appelé LokiBot qui est un malware Trojan/ransomware.

- e) Analysez dans Cuckoo sandbox le fichier 2e.bin. Documentez vos trouvailles. Quel est le message affiche dans Windows 7, vu via remote control, pendant que le virus est exécuté? (5p)



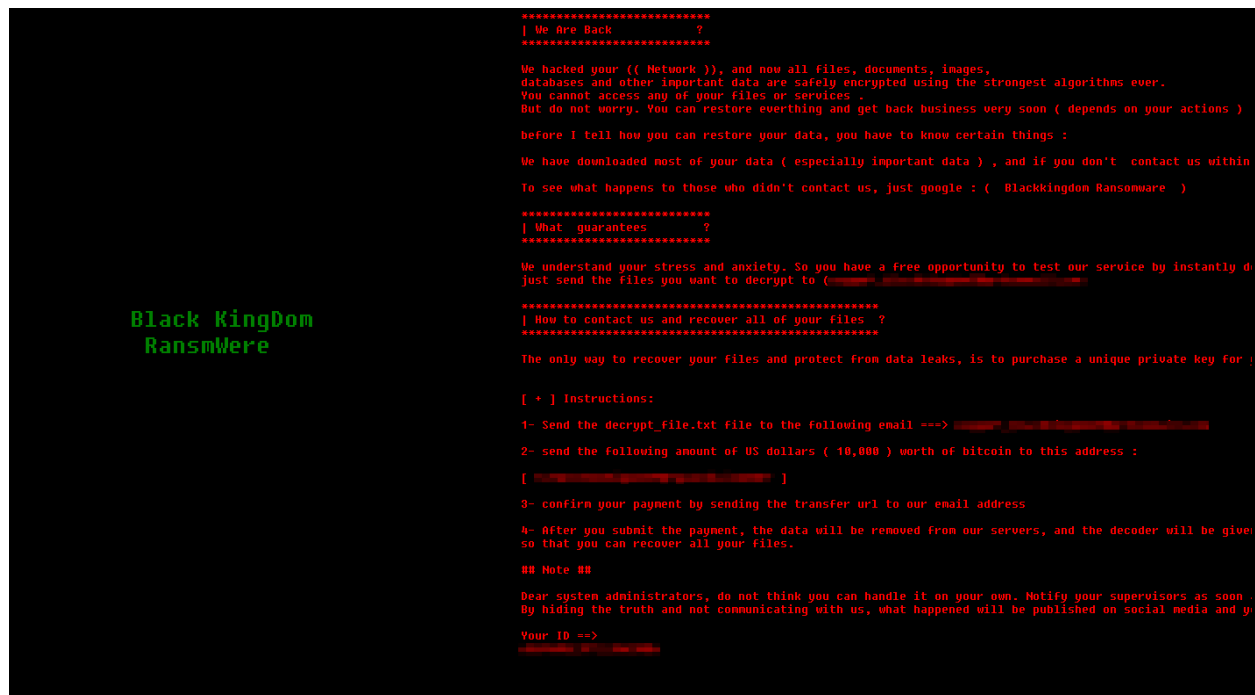
Avec les 2 règles ajoutées de Yara pour shamoon, et les defaults rules, nous n'avons pas vraiment rien capté dans le sommaire avec ex2e.bin, mais nous avons tous les hash, et nous pouvons analyser ceux-ci par exemple. Voir image ci-dessous



Nous pouvons voir que c'est un ransomware trojan nommé BlackKingdom

Et voici le message prompted par le virus ransomware Black Kingdom.

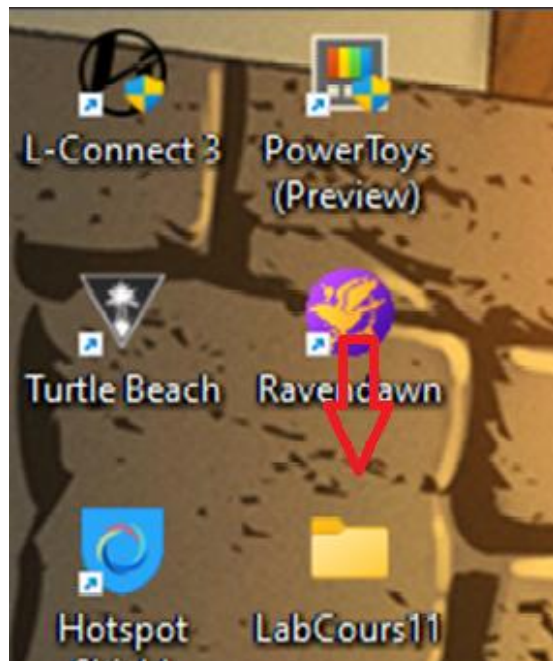




## Exercice No 3

Effectuez et documentez les **laboratoires du cours 11**

Nous commençons avec le lab 1 (Les IOC en action) et ensuite créer un fichier Cours11Lab sur notre Desktop



[illegible]

The screenshot shows a Windows File Explorer window titled 'LabCours11'. The address bar displays the path 'LabCours11' with a search icon on the right. The left sidebar shows the navigation pane with 'Home' and 'Gallery' selected. The main area displays a table of files and folders:

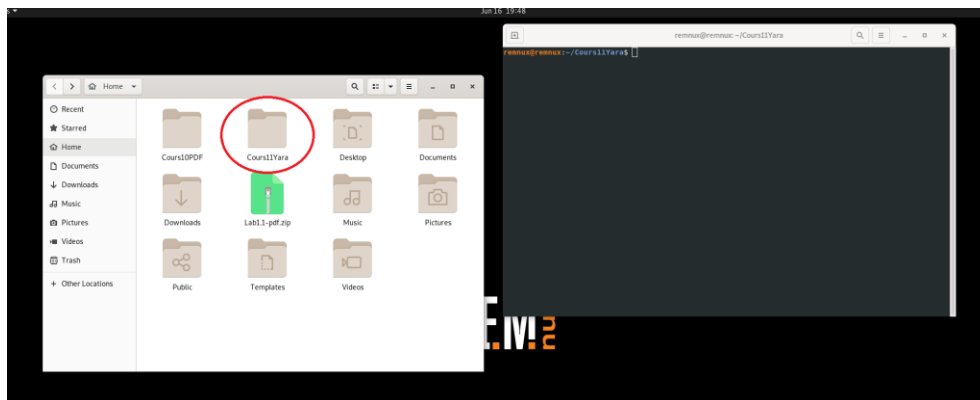
Name	Date modified	Type	Size
67e1924e-50e8-40ce-bf0f-80c3d54b99bc...	2024-06-16 7:18 PM	IOC File	2 KB



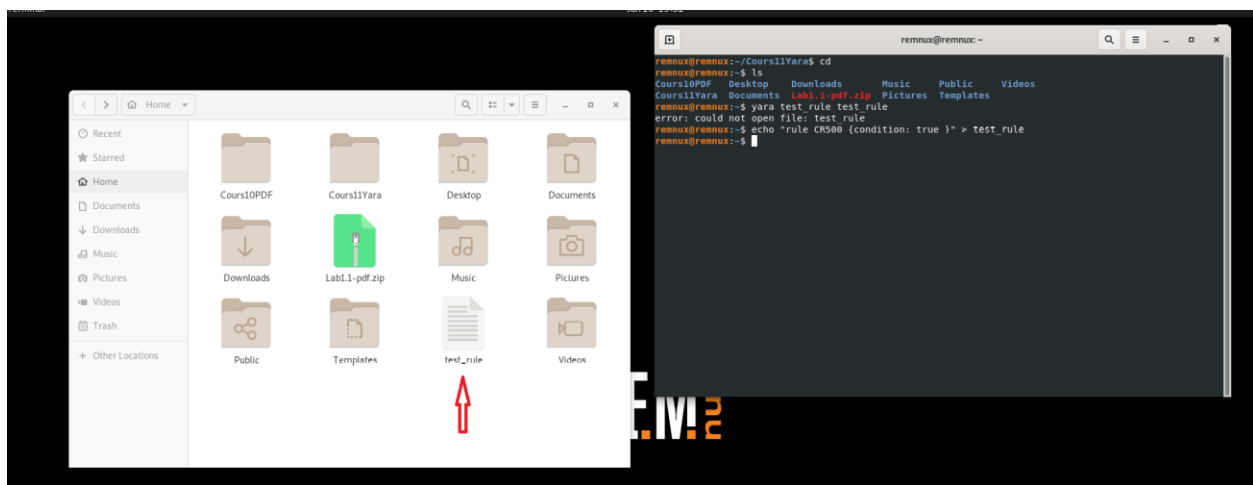
déterminer l'étendue de la menace. Il est également important de donner leur avis aux personnes impliquées dans le partage et la création du IOC. Les équipes ne parviennent souvent pas à communiquer efficacement leurs besoins et leurs défis, ce qui pourrait autrement réduire les problèmes. La transmission de ce type de retour d'informations au personnel chargé de la manipulation des menaces et de l'environnement ainsi qu'au personnel chargé de la consommation des renseignements sur les menaces pourrait générer de meilleurs IOC et leur permettre de comprendre ce qui fonctionne, ce qui est nécessaire et quelles informations d'accompagnement pourraient aider les intervenants en cas d'incident.

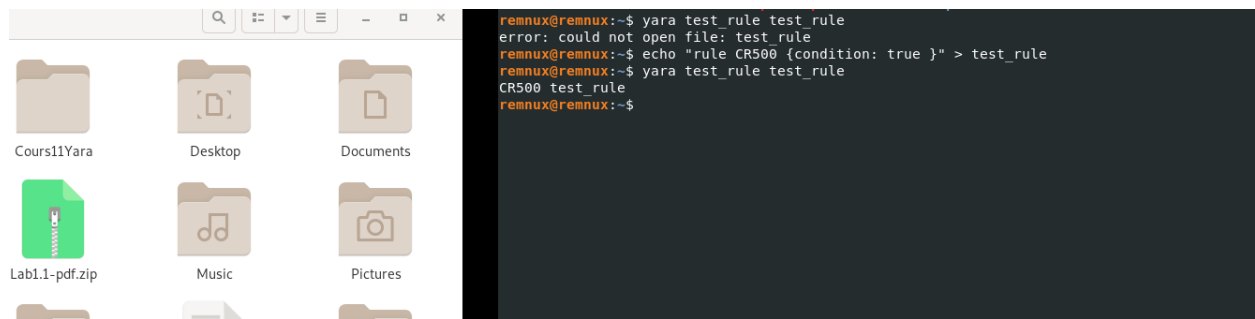
## Maintenant, Laboratoire 2 – Yara

On commence par unzip le fichier Cours11Yara. Pour que nous ayons un simple fichier Cours11Yara avec les 2 fichiers memdumps à l'intérieur.

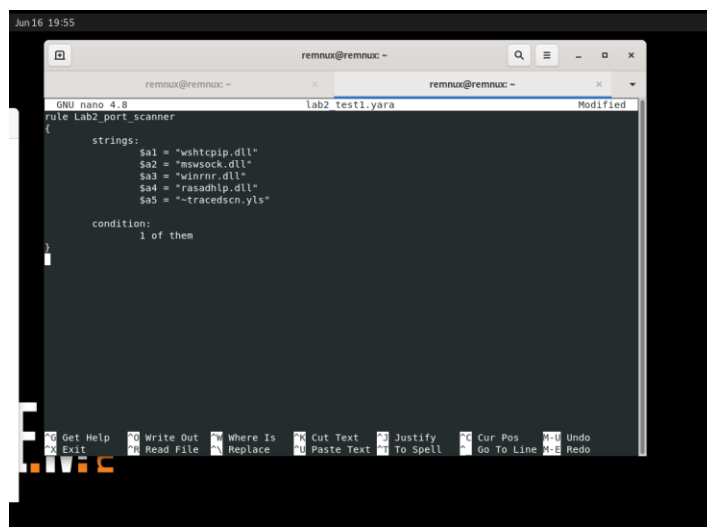


On exécute la commande **echo "rule CR500 {condition: true }" > test\_rule** et on teste la règle par rapport à elle-même avec **yara test\_rule test\_rule**

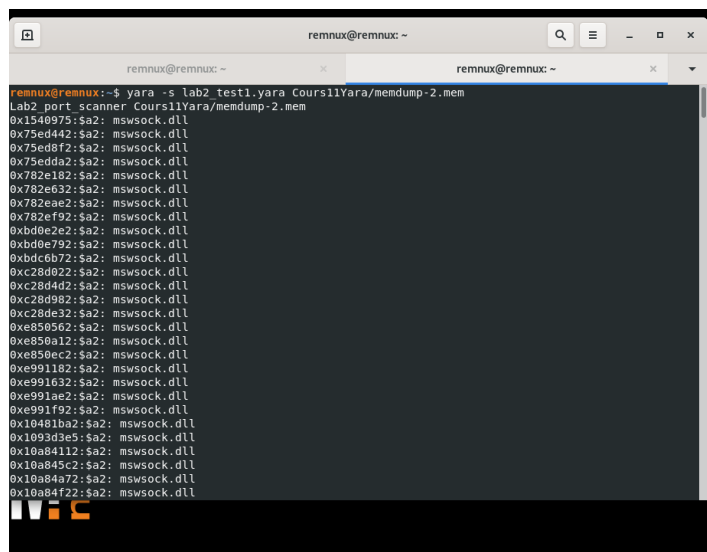




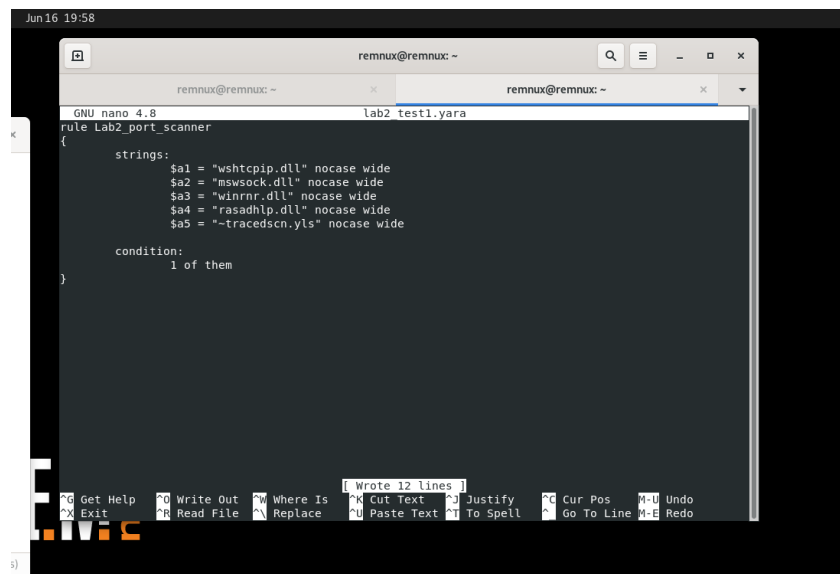
On insère maintenant les indicateurs dans la règle YARA comme indiqué dans le lab. Voir image ci-dessous pour la règle dans le txt editor.



On exécute la règle sur une des images mémoire.



On ajoute ensuite “nocase” et “wide” après chaque chaîne comme ci-dessous :



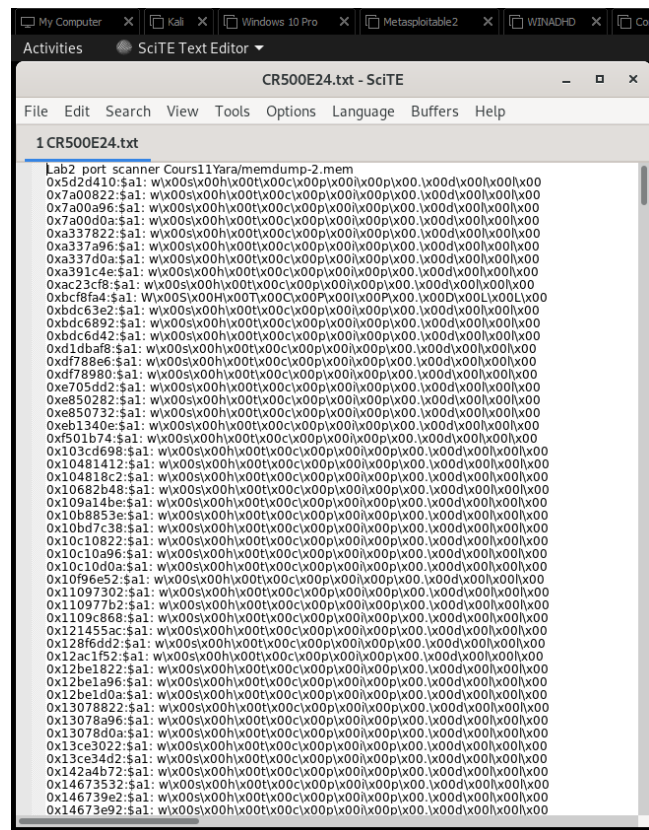
```
GNU nano 4.8 lab2_test1.yara
rule Lab2_port_scanner
{
  strings:
    $a1 = "wshtcpip.dll" nocase wide
    $a2 = "mwssock.dll" nocase wide
    $a3 = "winnr.dll" nocase wide
    $a4 = "rasadhip.dll" nocase wide
    $a5 = "-tracedscn.yls" nocase wide

  condition:
    1 of them
}
```

On enregistre et on quitte l’éditeur avec **CTRL+O** et **CTRL+X**.

Cette fois, nous nous attendons à beaucoup plus de résultats, nous les afficherons donc dans un fichier texte pour une visualisation plus facile. Nous exécutons la commande suivante : **yara -s lab2\_test1.yara Cours11Yara/memdump-2.mem > CR500E24.txt**

Nous sommes ensuite présentés avec ces résultats dans le fichier texte :



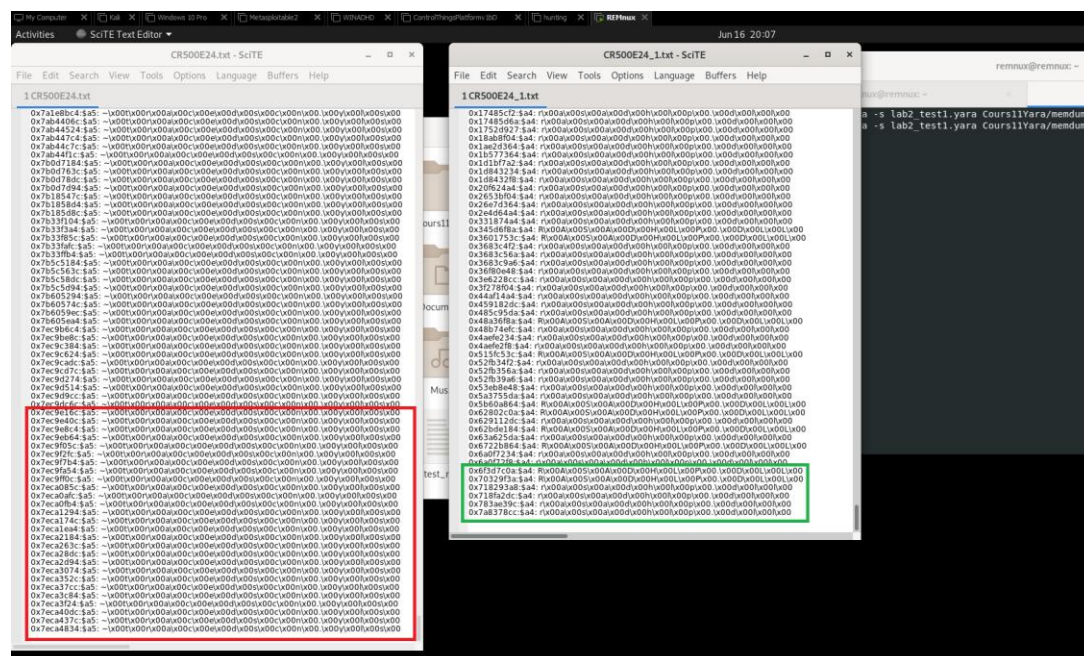
```
1 CR500E24.txt
lab2 port_scanner Cours11Yara/memdump-2.mem
0x5d2410:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x7a00822:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x7a00a96:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x7a00d0a:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xa337822:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xa337a96:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xa337d0a:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xa391c4e:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xac23cf8:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xbcf8fa4:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xbdc63e2:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xbdc6892:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xbdc6d42:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xd1dba8:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xd778e6:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xdf78980:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xe705dd2:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xe850282:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xe850732:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xeb1340e:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0xf501b74:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x103cd698:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x10481412:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x104818c2:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x10682b48:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x109a14be:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x10b8853e:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x10bd7c38:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x10c10822:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x10c10a96:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x10c10d0a:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x10f96e52:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x11097302:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x110977b2:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x1109c868:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x121455ac:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x128f6dd2:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x12ac1f52:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x12be1822:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x12be1a96:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x12be1d0a:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x13078822:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x13078a96:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x13078d0a:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x13ce3022:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x13ce34d2:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x142a4b72:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x14673532:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x146739e2:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
0x14673e92:$a1: wlx00sx00hx00txx00cx00px00lx00dx00lx00lx00
```

Nous avons désormais des hits sur chaque chaîne. Les résultats sont en HEX à cause de l'option "wide" puisqu'il ne s'agit plus de simples chaînes ASCII observables. Mais les deux options "wide" et "nocase" ont toutes deux permis de trouver des instances de chaînes que vous n'auriez pas vues autrement.

Il est important de valider la règle YARA par rapport à une bonne image connue pour supprimer toutes les chaînes fortement faussement positives dans le IOC.

On exécute la règle lab2\_test1.yara sur l'image mémoire de base avec la commande suivante :  
**yara -s lab2\_test1.yara Cours11Yara/memdump-1.mem > CR500E24\_1.txt**

Nous avons maintenant nos 2 fichiers que l'on peut comparer, comme dans le lab nous avons les même résultats



Si on souhaite plus d'informations pour rédiger des règles plus complexes, on peut consulter la documentation YARA à l'adresse :

<https://yara.readthedocs.io/en/stable/writingrules.html>

## Exercice No 4 (25p)

En utilisant la capture Redline suivante

<https://drive.google.com/file/d/1HZRBXRlxqSUiVojKeyhDir5C499NwPAF/view?usp=sharing> (mot de passe infected) répondez aux question suivantes (similaire au lab cours 9)

- a) Quel et le nom, système d'opération, logged on user et l'adresse IP du système victime (5p)

Nom: WIN7X64

Machine Name:	WIN7X64
Host Name:	Win7x64
System Date:	2023-06-17 16:54:24Z
Time Zone DST:	Eastern Daylight Time
Time Zone Standard:	Eastern Standard Time
Product Name:	Microsoft Windows [Version 6.0.6002.18005] Copyright (c) 2009 Microsoft Corporation. All rights reserved.

OS: Windows 7 Professional 7600

Operating System Information	
Operating System:	Windows 7 Professional 7600
Product Name:	Windows 7 Professional

logged on user: Win7x64\Lucian,WORKGROUP\WIN7X64\$

Logged on User:	Win7x64\Lucian,WORKGROUP\WIN7X64\$
BIOS Information	


Adresse IP: 192.168.1.208



System Information	
Operating System:	Windows 7 Professional 7600
Domain:	WORKGROUP
Host:	Win7x64
Primary IP Address:	192.168.1.208

- b) Pendant le réponse aux incidents vous identifiez une machine avec l'adresse IP 192.168.1.105.
- a. En regardant les services, ports ... ouverts, pourriez-vous identifier une connexion et, si le cas, si cela est suspectieuse? (5p)

Oui nous pouvons observer une connexion depuis un serveur http 8000 (basic http server).

Évidemment, un trojan.exe est extrêmement suspectieux et devrait être investigué/intervenir.

	trojan.exe	864	C:\Users\Lucian\Downloads\trojan.exe	ESTABLISHED	192.168.1.208	49343	192.168.1.105	4444	TCP
---	------------	-----	--------------------------------------	-------------	---------------	-------	---------------	------	-----

	Download Type	Source URL	Target Directory	File Name	File Size	Bytes Downloaded	State	Start Date	End Date	Last Accessed
	Manual	http://192.168.1.105:8000/trojan	C:\Users\Lucian\Downloads	trojan	7 Kilobytes	7 Kilobytes	Finished	2023-06-17 16:37:48Z	2023-06-17 16:37:49Z	

- b. Pourriez-vous identifier la provenance du fichier qui communique avec cette machine? (5p)

C'est un exécutable dans le fichiers Downloads. Donc le fichier a été téléchargé et puis exécuté à distance. Depuis l'adresse et source URL **192.168.1.105:8000/trojan**



Download Type	Source URL	Target Directory	File Name	File Size	Bytes Downloaded	State	Start Date	End Date	Last Access
Manual	http://192.168.1.105:8000/trojan	C:\Users\Lucian\Downloads	trojan	7 Kilobytes	7 Kilobytes	Finished	2023-06-17 16:37:48Z	2023-06-17 16:37:49Z	

c) Il y a une connexion du qModMaster vers une autre machine. Pourriez-vous identifier :

a. Le nom et l'id du processus (2.5p)

**qModMaster.exe / PID : 3880**

qModMaster.exe	3880	C:\Users\Lucian\Desktop\qModMaster	ESTABLISHED	192.168.1.208	49588	192.168.1.44	1502	TCP
----------------	------	------------------------------------	-------------	---------------	-------	--------------	------	-----

b. La date du lancement (5p)

**2023-06-17**

URL	file:///C:/Users/Lucian/Desktop/qModMaster-Win32-...	0	2023-06-17 16:42:12Z
-----	--	---	----------------------

c. L'ip et le port de destination (2.5p)

**IP est 192.168.1.44 / Port de destination est 1502**

qModMaster.exe	3880	C:\Users\Lucian\Desktop\qModMaster	ESTABLISHED	192.168.1.208	49588	192.168.1.44	1502	TCP
----------------	------	------------------------------------	-------------	---------------	-------	--------------	------	-----