OLETOOLS – HERRAMIENTAS PARA EL ANÁLISIS DE MALDOCS

oletools es un paquete de herramientas python para analizar archivos Microsoft OLE2 (también llamados Structured Storage, Compound File Binary Format o Compound Document File Format), como documentos de Microsoft Office o mensajes de Outlook, principalmente para análisis de malware, forense y depuración. Se basa en el analizador sintáctico olefile.

INSTALACIÓN

Para instalar la herramienta solamente necesitamos tener Python3 instalado en nuestro sistema y el gestor de paquetes de este. Es decir, pip3. Si cumplimos estos requisitos, podemos instalar las oletools de la siguiente manera:

```
sudo -H pip install -U oletools[full]
```

Sustituya pip por pip3 o pip2 para instalar en una versión específica de Python.

Instalación del paquete oletools

De no cumplir con los requisitos, siempre queda la opción de descargar el software directamente desde el repositorio de GitHub de la herramienta.

Una vez se ha instalado el paquete, tenemos disponibles una serie de nuevos comandos:

- **oleid**: permite analizar ficheros OLE para detectar características que normalmente se encuentran en ficheros maliciosos.
- **olevba**: dispone de la capacidad de extraer y analizar las macros VBA de los ficheros de MS Office (OLE y OpenXML).
- **MacroRaptor**: sirve para detectar las Macros VBA maliciosas.
- **msodde**: proporciona la capacidad de detectar enlaces DDE/DDEAUTO de los ficheros de MS Office, RTF y CSV.
- **pyxswf**: Detecta, analiza y extrae los objetos Flash (SWF) que pueden estar embebidos en ficheros con formato de MS Office y RTF.
- **oleobj**: Extrae los ficheros embebidos de los ficheros OLE.
- **rtfobj**: Lo mismo que el anterior pero con ficheros RTF.
- **olebrowse**: Proporciona una interfaz gráfica simple para navegar por los ficheros OLE. Este permite visualizar y extraer partes concretas del fichero.

- **olemeta**: Consigue los metadatos de los ficheros OLE.
- **oletimes**: Extrae las marcas de tiempo del fichero como la fecha de creación, la fecha de modificación, etc.
- **oledir**: Muestra todas las entradas de directorio de un archivo OLE.
- **olemap**: Pinta una tabla con todos los sectores, y sus atributos, del fichero OLE.

PROBANDO LAS HERRAMIENTAS

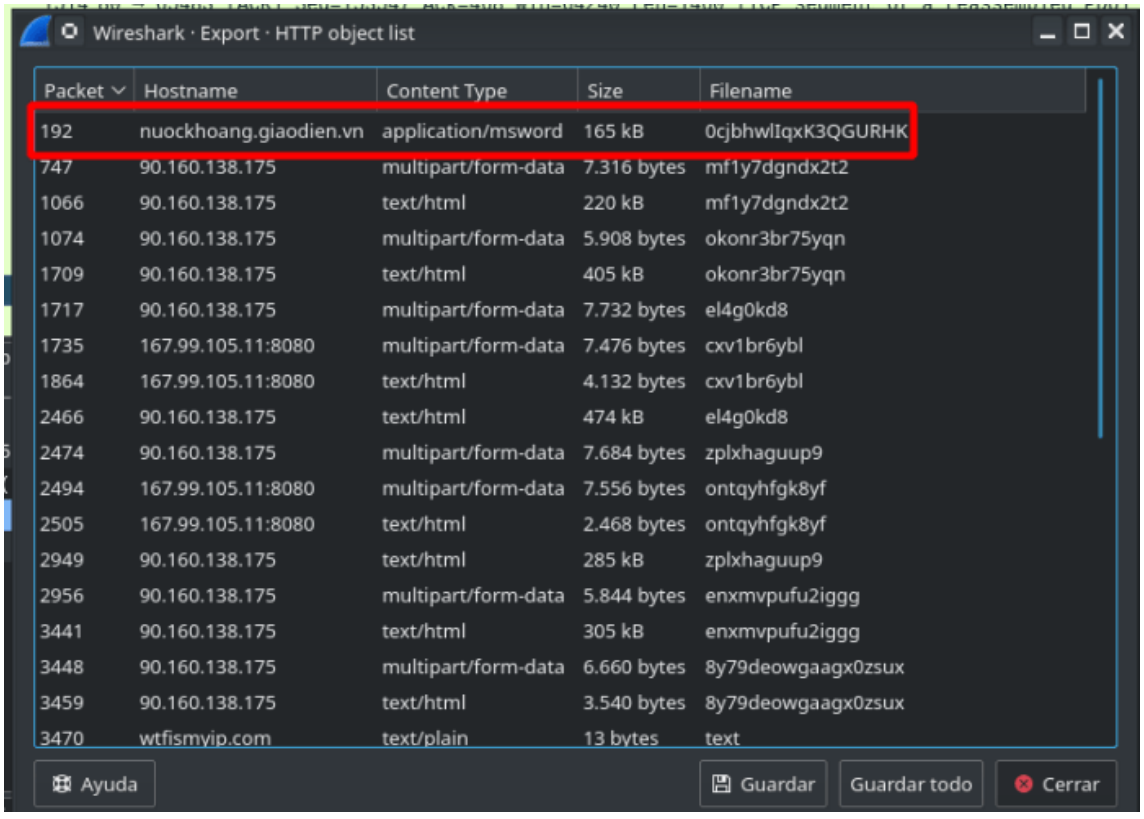
Para poder probar la herramienta necesitamos disponer de un Maldoc el cual analizar. Para ello podemos visitar la web malware-traffic-analysis.net y descargar alguna muestra de allí. **¡OJO! Se trata de Malware que de verdad puede infectar y estropear tú equipo. ¡No lo hagas si no sabes lo que estás haciendo! (No me hago responsable de lo que hagas con la información que aquí se proporciona)**.

[La otra alternativa es que crees tú mismo un documento con alguna macro inofensiva y lo pruebes. Esta es, sin lugar a dudas, la alternativa más segura.]

En este caso se ha optado, por ejemplo, por descargar la siguiente muestra, que en este caso está relacionada con el Malware Emotet:

https://malware-traffic-analysis.net/2021/01/04/index.html

Por tanto, descargamos el fichero ZIP y extraemos el PCAP. Desde la herramienta Wireshark podemos obtener el documento fácilmente:



Maldoc identificado en Wireshark

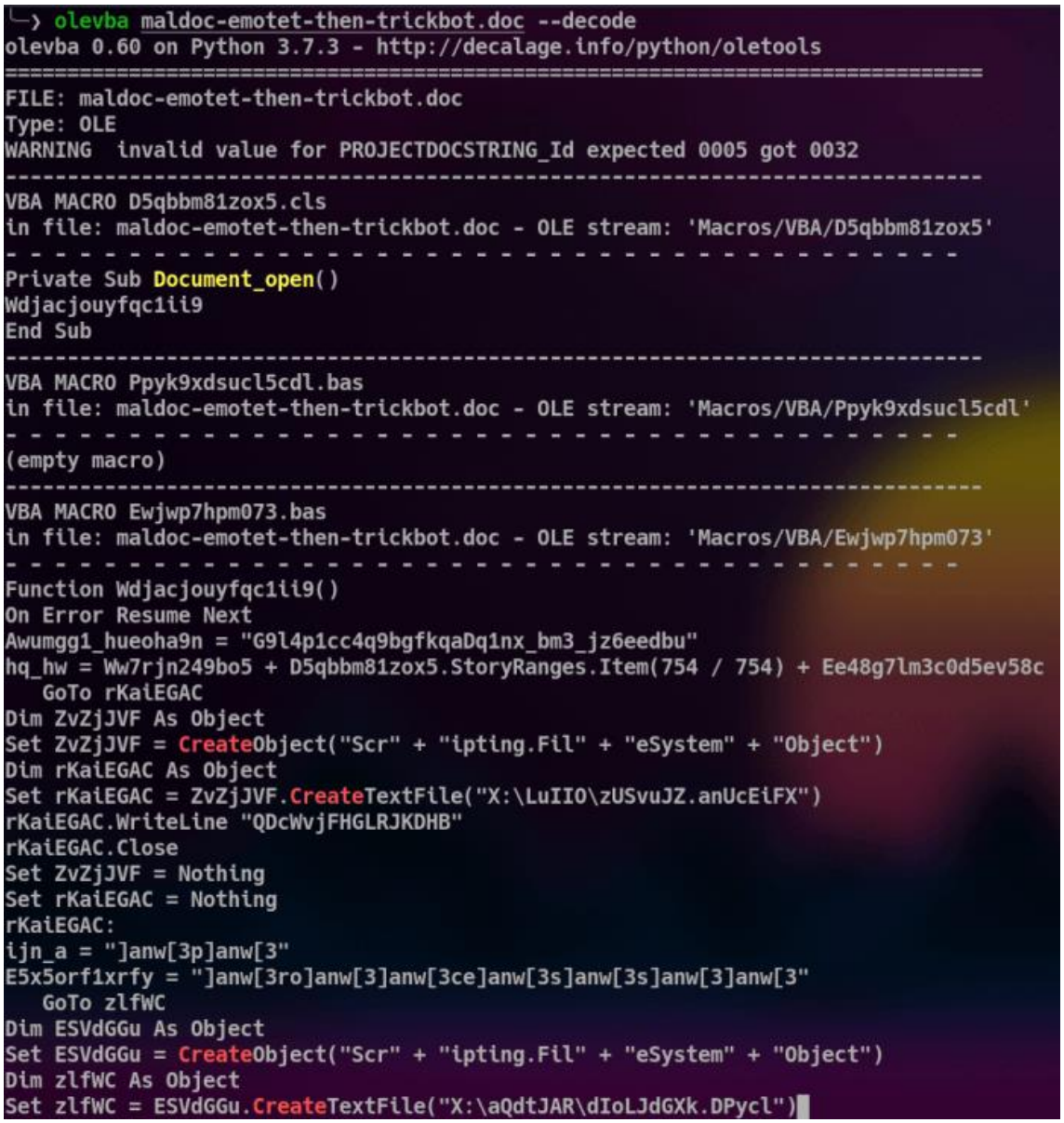
Una vez nos hemos hecho con el fichero malicioso, podemos empezar a trastear con él haciendo uso de las herramientas. Por ejemplo, podemos ejecutar el primer comando listado anteriormente:

```
└─> oleid maldoc-emotet-then-trickbot.doc
oleid 0.60.dev1 - http://decalage.info/oletools
THIS IS WORK IN PROGRESS - Check updates regularly!
Please report any issue at https://github.com/decalage2/oletools/issues

Filename: maldoc-emotet-then-trickbot.doc
+--------------------+--------------------+----------+--------------------------
|Indicator           |Value               |Risk      |Description
+--------------------+--------------------+----------+--------------------------
|File format         |MS Word 97-2003     |info      |
|                    |Document or Template|          |
+--------------------+--------------------+----------+--------------------------
|Container format    |OLE                 |info      |Container type
+--------------------+--------------------+----------+--------------------------
|Application name    |Microsoft Office    |info      |Application name declared
|                    |Word                |          |in properties
+--------------------+--------------------+----------+--------------------------
|Properties code page|1252: ANSI Latin 1; |info      |Code page used for
|                    |Western European    |          |properties
|                    |(Windows)           |          |
+--------------------+--------------------+----------+--------------------------
|Author              |Louise Chevalier    |info      |Author declared in
|                    |                    |          |properties
+--------------------+--------------------+----------+--------------------------
|Encrypted           |False               |none      |The file is not encrypted
+--------------------+--------------------+----------+--------------------------
|VBA Macros          |Yes, suspicious     |HIGH      |This file contains VBA
|                    |                    |          |macros. Suspicious
|                    |                    |          |keywords were found. Use
|                    |                    |          |olevba and mraptor for
|                    |                    |          |more info.
+--------------------+--------------------+----------+--------------------------
|XLM Macros          |No                  |none      |This file does not contain
|                    |                    |          |Excel 4/XLM macros.
+--------------------+--------------------+----------+--------------------------
|External            |0                   |none      |External relationships
|Relationships       |                    |          |such as remote templates,
|                    |                    |          |remote OLE objects, etc
+--------------------+--------------------+----------+--------------------------
```

Resultado del comando oleid

Efectivamente, el comando nos devuelve en los resultados que el fichero contiene Macros VBA entre otras cosas. A partir de este punto, podemos ejecutar el comando que queramos. No obstante, el más interesante de todos es olevba, ya que sabemos que el fichero tiene Macros en él:



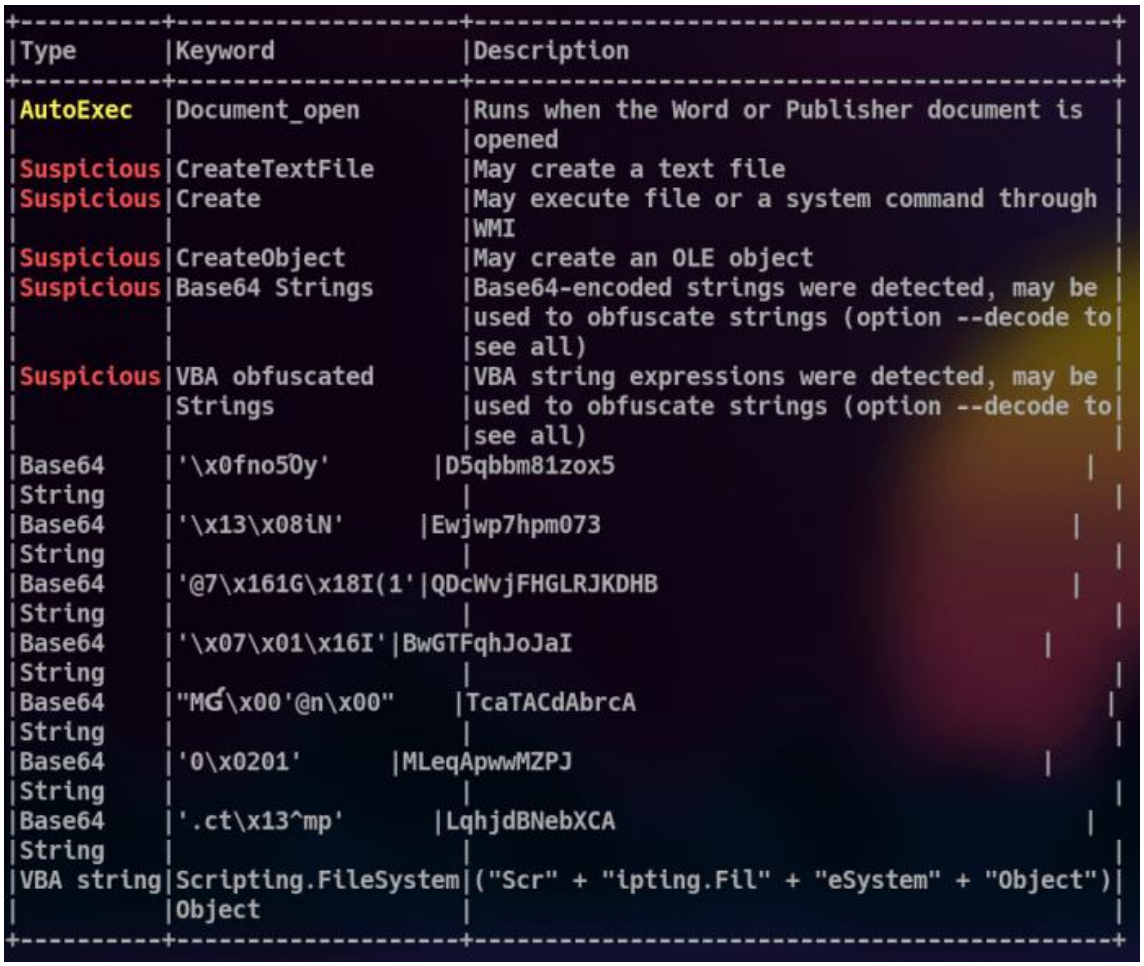
Ejecución del comando

Como se puede ver en la imagen, al lanzar el comando, la herramienta se encarga de extraer la Macro o Macros del fichero y a su vez se dedica a analizarlas. Si nos fijamos, la palabra «Create» aparece destacada en rojo, ya que es un indicativo de que la Macro intenta llevar a cabo acciones como por ejemplo crear un fichero, un objeto, etc.

Por otro lado, comentar que la flag `--decode` sirve para indicarle a la herramienta que si encuentra cadenas codificadas en base64 las decodifique automáticamente. Podemos descubrir todos los parámetros que se pueden utilizar con la herramienta si ejecutamos el siguiente comando: `olevba -h`.

Al tratarse de un caso real, la Macro está lo suficientemente ofuscada como para complicarle la vida lo máximo posible al analista. No obstante, se pueden apreciar algunas cosas a simple vista.

Finalmente, tras el código obtenido de la Macro, el comando proporciona una tabla con un resumen del análisis que ha realizado:



Type	Keyword	Description
AutoExec	Document_open	Runs when the Word or Publisher document is opened
Suspicious	CreateTextFile	May create a text file
Suspicious	Create	May execute file or a system command through WMI
Suspicious	CreateObject	May create an OLE object
Suspicious	Base64 Strings	Base64-encoded strings were detected, may be used to obfuscate strings (option --decode to see all)
Suspicious	VBA obfuscated Strings	VBA string expressions were detected, may be used to obfuscate strings (option --decode to see all)
Base64 String	'\x0fno5Oy'	D5qbbm81zox5
Base64 String	'\x13\x08iN'	Ewjwp7hpm073
Base64 String	'@7\x161G\x18I(1'	QDcWvjFHGLRJKDHB
Base64 String	'\x07\x01\x16I'	BwGTFqhJoJaI
Base64 String	"MG\x00'@n\x00"	TcaTACdAbrcA
Base64 String	'0\x0201'	MLeqApwwMZPJ
Base64 String	'.ct\x13^mp'	LqhjdBNebXCA
VBA string	Scripting.FileSystem Object	("Scr" + "ipting.Fil" + "eSystem" + "Object")

Tabla final con los resultados del análisis

En esta tabla podemos encontrar cosas interesantes como que la Macro se ejecutará en el momento se abra el documento. Además, tal y como se ha comentado previamente, también ha encontrado palabras clave como «Create» y derivadas, así como cadenas codificadas en base64 o cadenas ofuscadas.

Cabe destacar que en el caso de encontrar un comando codificado en base64 de Powershell, la herramienta es capaz de extraerlo y decodificar dicho código en base64 de manera automática. De hecho, la herramienta es capaz de detectar técnicas tan complejas como VBA Stomping (https://attack.mitre.org/techniques/T1564/007/). En conclusión, es una herramienta muy completa para el análisis de este tipo de documentos.