Quarkslab

Securing every bit of your data

Analysing malicious documents and files with oletools

Philippe Lagadec – Pass-The-Salt 2024-07-05

WHOAMI

Philippe Lagadec

- Cybersecurity R&D engineer at Quarkslab
- Dissecting file formats and collecting malware since 2000
- Some open-source projects :
  - **olefile since 2005 (to parse MS Office documents)**
  - **exefilter since 2008 (to scan many file formats and clean them from active content like macros)**
  - **oletools since 2012 (to analyse MS Office files, detect malicious documents)**
- X/Twitter : @decalage2 - @decalage@mastodon.social – https://decalage.info

SOMMAIRE

1. Olefile / Oletools – a bit of history

2. Analysing suspicious documents

3. File format identification (ftguess)

4. MSI / MSIX

A bit of history

- 2000: building the 1st email gateway for French MoD – developed filters for **VBA macros, RTF OLE objects and PDF**
- 2005: open-sourced **olefile**, parser for OLE files (MS Office), fork from PIL (now Pillow)
- 2008: open-sourced **exefilter** (to filter and disarm many file formats)
- 2012: published **oletools**
- 2014: added **olevba** to extract and analyse VBA macros
- 2015: published **ViperMonkey**, an emulator for VBA macros and VBScript
- Since then added many features, additional file formats, etc
- With the help from dozens of contributors!

Oletools : open-source tools to analyse OLE files & MS Office documents

- Open-source project started in 2012, initially for exploring OLE files
- Several tools to analyse different file formats (legacy Office 97-2003 files, OpenXML Office 2007+ files, RTF, etc)
- Detect security issues / attack techniques:
  - **VBA Macros, XLM macros, OLE objects, DDE, Remote Templates/OLE**

- https://github.com/decalage2/oletools

Projects / Products using oletools

- oletools are used by a number of projects and online malware analysis services, including:
- ACE, ADAPT, Anlyz.io, AssemblyLine, Binary Refinery, CAPE, CinCan, Cortex XSOAR (Palo Alto), Cuckoo Sandbox, DARKSURGEON, Deepviz, DIARIO, dridex.malwareconfig.com, EML Analyzer, EXPMON, FAME, FLARE-VM, GLIMPS Malware, Hybrid-analysis.com, InQuest Labs, IntelOwl, Joe Sandbox, Laika BOSS, MacroMilter, mailcow, malshare.io, malware-repo, Malware Repository Framework (MRF), MalwareBazaar, olefy, Pandora, PeekabooAV, pcodedmp, PyCIRCLean, QFlow, Qu1cksc0pe, Tylabs QuickSand, REMnux, Snake, SNDBOX, Splunk add-on for MS O365 Email, SpuriousEmu, Strelka, stoQ, Sublime Platform/MQL, Subparse, TheHive/Cortex, ThreatBoook, TSUGURI Linux, Vba2Graph, Viper, ViperMonkey, YOMI, and probably VirusTotal, FileScan.IO.

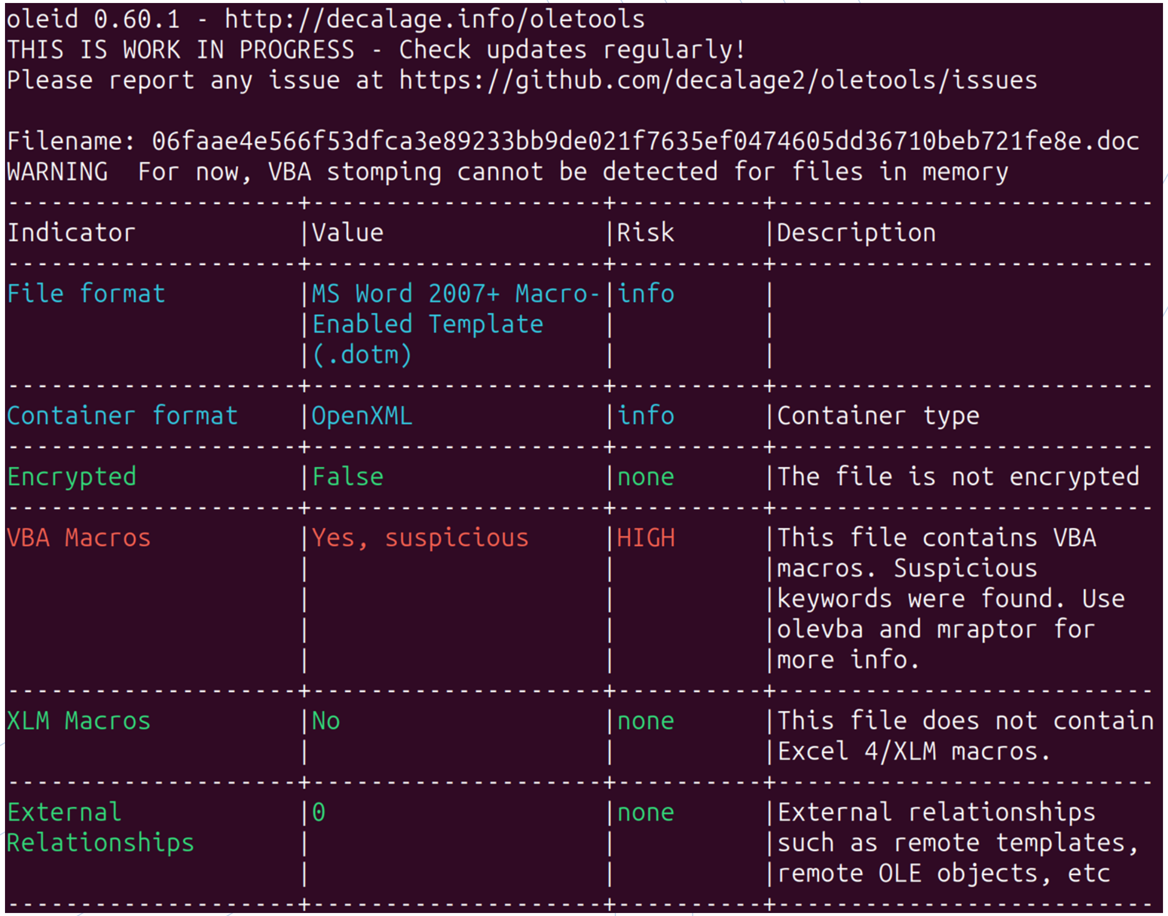
Many file formats:

OLE/CFB format	OpenXML format (ZIP+XML)	Other
DOC - Word 97-2003	DOCX/M – Word 2007+	RTF
XLS – Excel 97-2003	XLSX/M/B – Excel 2007+	Word 2003 XML
PPT – PowerPoint 97-2003	PPTX/M – PowerPoint 2007+	Word XML - FlatOPC
PUB - Publisher	VSDX - Visio	Excel 2003 XML
VSD - Visio	XPS	PowerPoint XML - FlatOPC
MPP/MPT – Project	MSIX	MHT – Word/Excel MHTML
Outlook messages		SLK
FlashPix images		CSV
StickyNotes		
MSI		

OLETOOLS

Oleid : quick summary of analysis

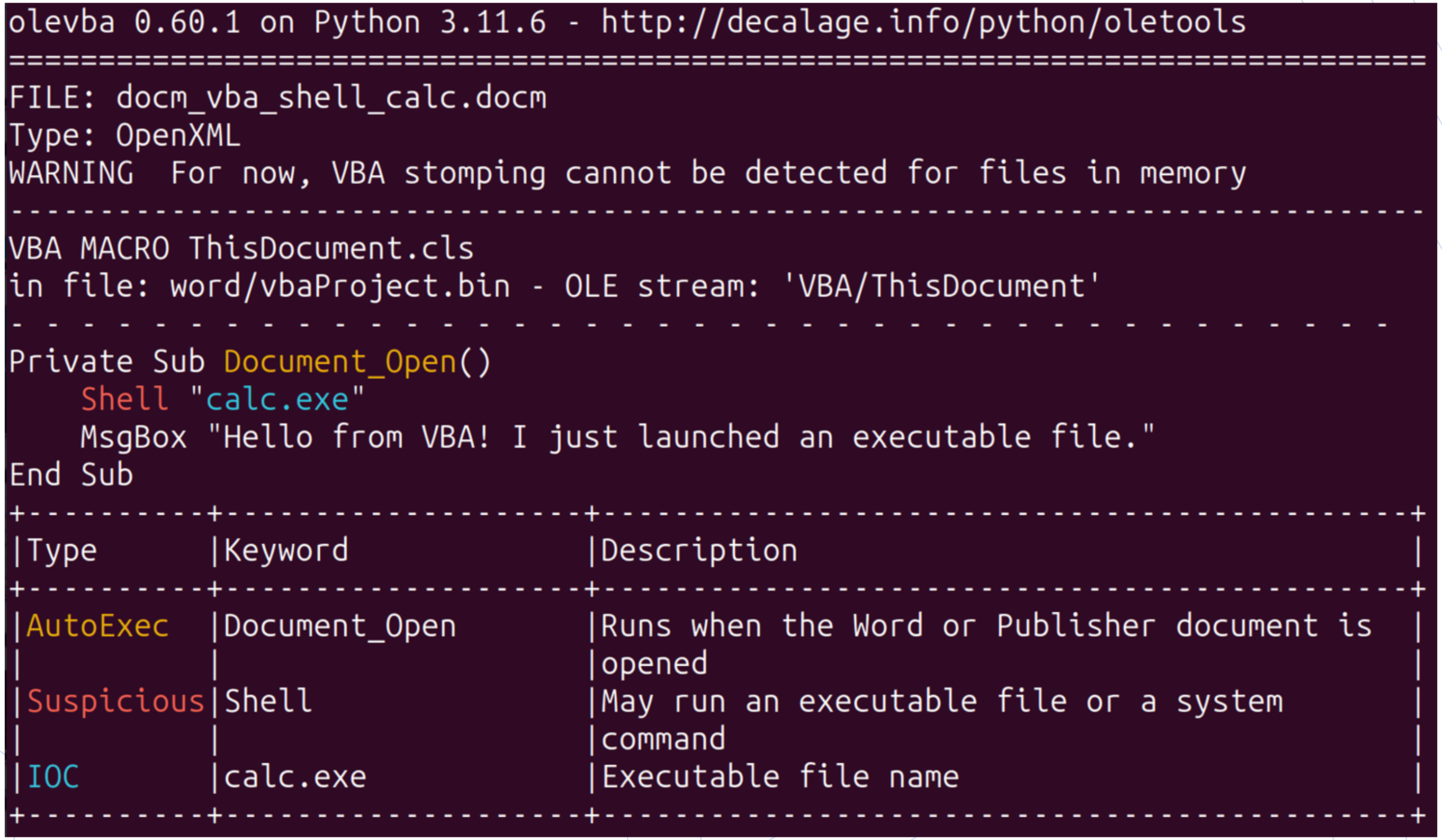
- First tool to be used:
- Identify file format
- Run all relevant oletools
- Summarize results
- Suggest tools to get more details



```
oleid 0.60.1 - http://decalage.info/oletools
THIS IS WORK IN PROGRESS - Check updates regularly!
Please report any issue at https://github.com/decalage2/oletools/issues

Filename: 06faae4e566f53dfca3e89233bb9de021f7635ef0474605dd36710beb721fe8e.doc
WARNING  For now, VBA stomping cannot be detected for files in memory
--------------------+--------------------+----------+--------------------------
Indicator           |Value               |Risk      |Description
--------------------+--------------------+----------+--------------------------
File format         |MS Word 2007+ Macro-|info      |
                    |Enabled Template    |          |
                    |(.dotm)             |          |
--------------------+--------------------+----------+--------------------------
Container format    |OpenXML             |info      |Container type
--------------------+--------------------+----------+--------------------------
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
--------------------+--------------------+----------+--------------------------
```

OLETOOLS

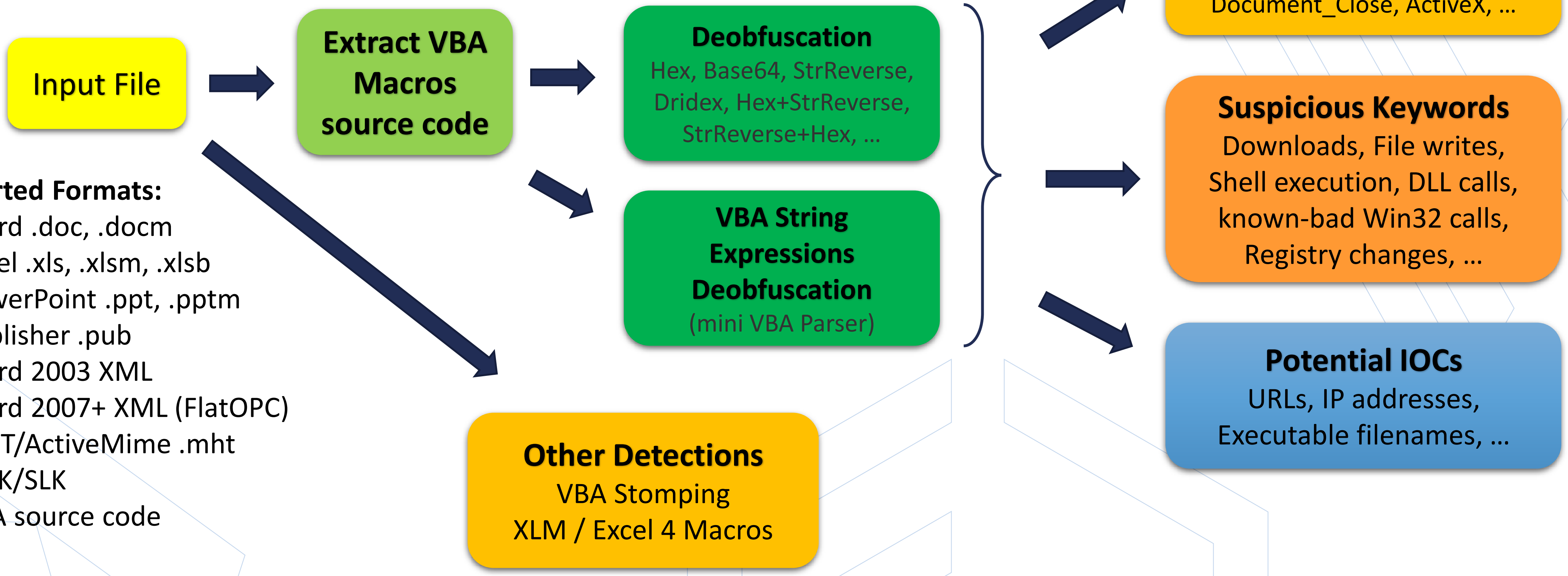
Olevba : Extract and analyze VBA Macros and Excel 4 Macros (XLM)



```
olevba 0.60.1 on Python 3.11.6 - http://decalage.info/python/oletools
===============================================================================
FILE: docm_vba_shell_calc.docm
Type: OpenXML
WARNING  For now, VBA stomping cannot be detected for files in memory
-------------------------------------------------------------------------------
VBA MACRO ThisDocument.cls
in file: word/vbaProject.bin - OLE stream: 'VBA/ThisDocument'
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Private Sub Document_Open()
    Shell "calc.exe"
    MsgBox "Hello from VBA! I just launched an executable file."
End Sub
+----------+--------------------+---------------------------------------------+
|Type      |Keyword             |Description                                  |
+----------+--------------------+---------------------------------------------+
|AutoExec  |Document_Open       |Runs when the Word or Publisher document is  |
|          |                    |opened                                       |
|Suspicious|Shell               |May run an executable file or a system       |
|          |                    |command                                      |
|IOC       |calc.exe            |Executable file name                         |
+----------+--------------------+---------------------------------------------+
```

OLETOOLS

Olevba : Extract and analyze Macros



Input File

Extract VBA Macros source code

Deobfuscation
Hex, Base64, StrReverse, Dridex, Hex+StrReverse, StrReverse+Hex, …

VBA String Expressions Deobfuscation
(mini VBA Parser)

Other Detections
VBA Stomping
XLM / Excel 4 Macros

Auto Execution Triggers
AutoOpen, Document_Open, Document_Close, ActiveX, …

Suspicious Keywords
Downloads, File writes, Shell execution, DLL calls, known-bad Win32 calls, Registry changes, …

Potential IOCs
URLs, IP addresses, Executable filenames, …

Supported Formats:

- Word .doc, .docm
- Excel .xls, .xlsm, .xlsb
- PowerPoint .ppt, .pptm
- Publisher .pub
- Word 2003 XML
- Word 2007+ XML (FlatOPC)
- MHT/ActiveMime .mht
- SYLK/SLK
- VBA source code

- Even encrypted with password

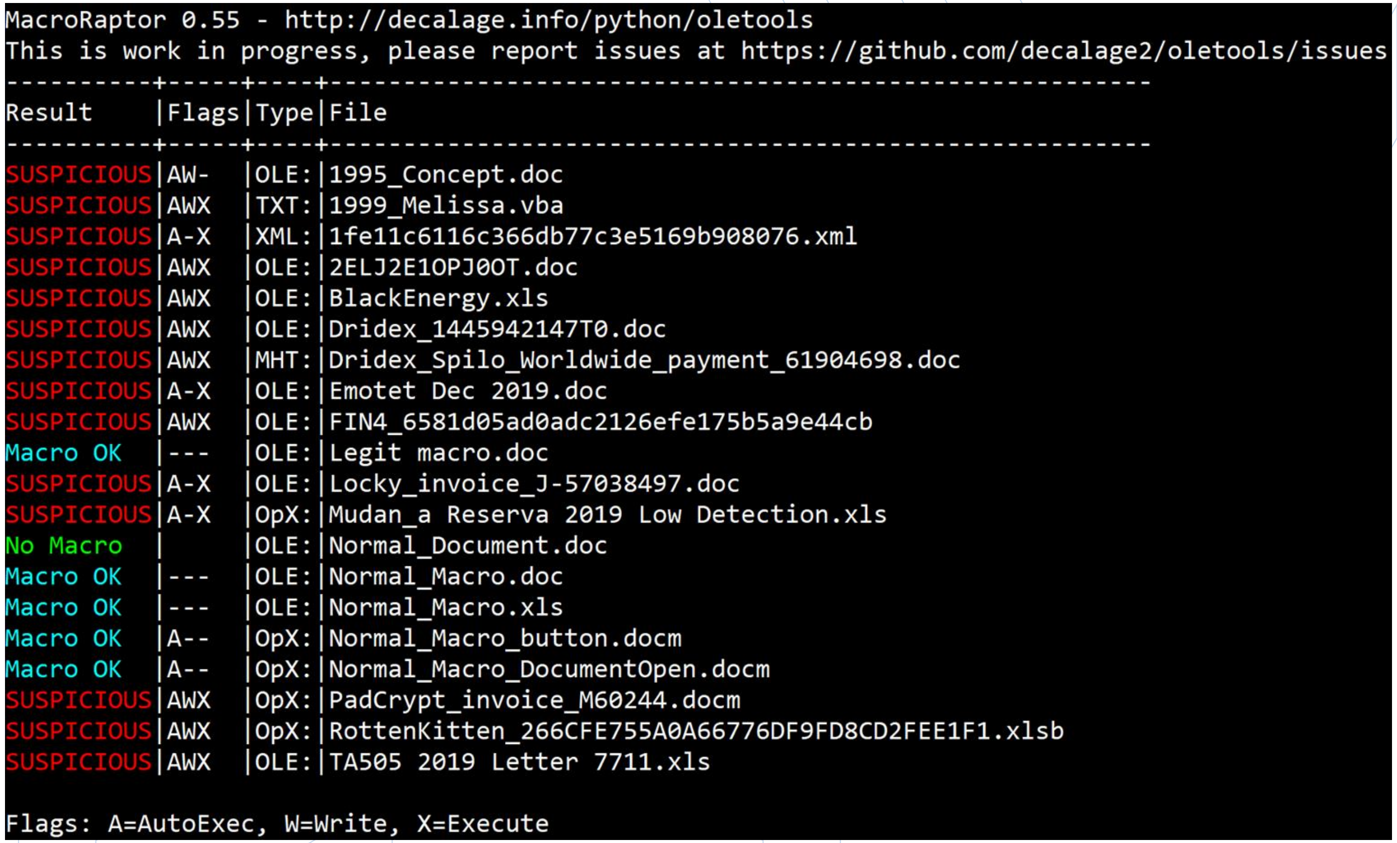
OLETOOLS

Mraptor : Detect suspicious VBA macros

- Can distinguish legitimate and suspicious macros

- MacroRaptor algorithm:
  - **A: Automatic triggers**
  - **W: Any write operation that may be used to drop a payload**
  - **X: Any execute operation**

- **Suspicious = A and (W or X)**



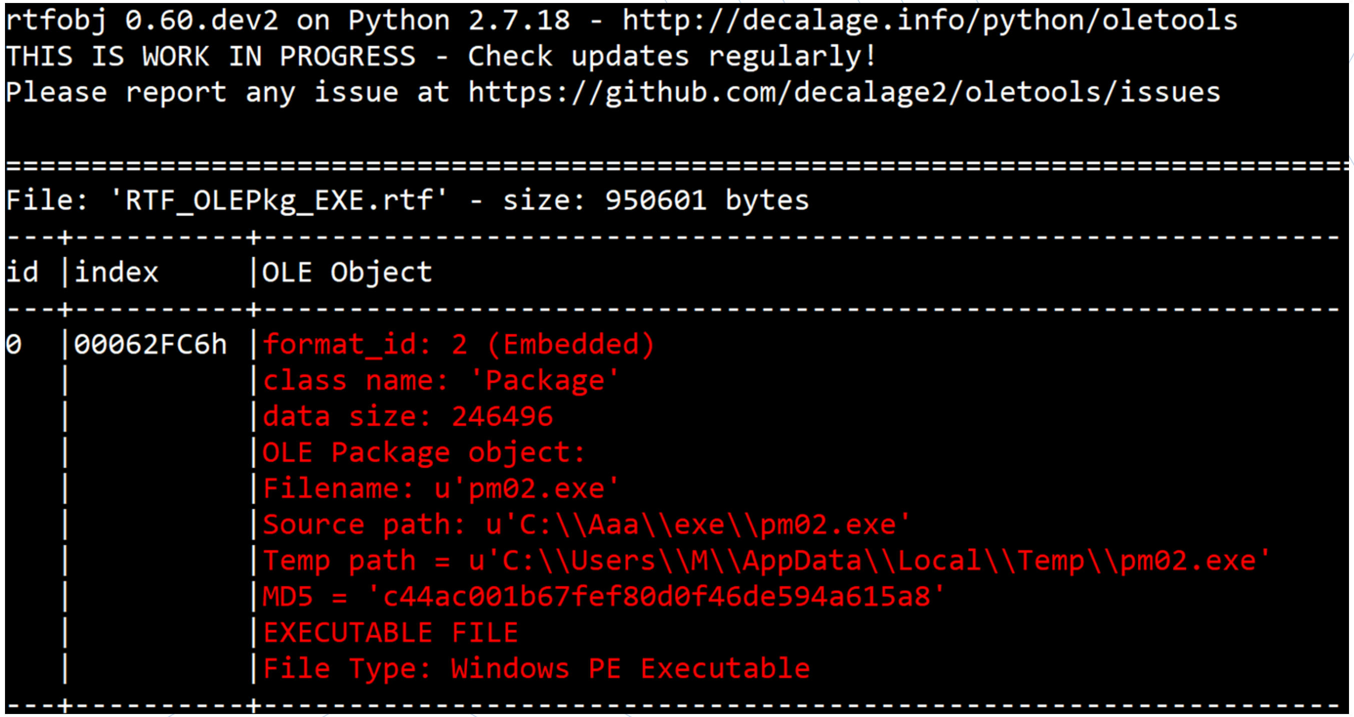
```
MacroRaptor 0.55 - http://decalage.info/python/oletools
This is work in progress, please report issues at https://github.com/decalage2/oletools/issues
----------+-----+----+--------------------------------------------------------
Result    |Flags|Type|File
----------+-----+----+--------------------------------------------------------
SUSPICIOUS|AW-  |OLE:|1995_Concept.doc
SUSPICIOUS|AWX  |TXT:|1999_Melissa.vba
SUSPICIOUS|A-X  |XML:|1fe11c6116c366db77c3e5169b908076.xml
SUSPICIOUS|AWX  |OLE:|2ELJ2E1OPJ0OT.doc
SUSPICIOUS|AWX  |OLE:|BlackEnergy.xls
SUSPICIOUS|AWX  |OLE:|Dridex_1445942147T0.doc
SUSPICIOUS|AWX  |MHT:|Dridex_Spilo_Worldwide_payment_61904698.doc
SUSPICIOUS|A-X  |OLE:|Emotet Dec 2019.doc
SUSPICIOUS|AWX  |OLE:|FIN4_6581d05ad0adc2126efe175b5a9e44cb
Macro OK  |---  |OLE:|Legit macro.doc
SUSPICIOUS|A-X  |OLE:|Locky_invoice_J-57038497.doc
SUSPICIOUS|A-X  |OpX:|Mudan_a Reserva 2019 Low Detection.xls
No Macro  |     |OLE:|Normal_Document.doc
Macro OK  |---  |OLE:|Normal_Macro.doc
Macro OK  |---  |OLE:|Normal_Macro.xls
Macro OK  |A--  |OpX:|Normal_Macro_button.docm
Macro OK  |A--  |OpX:|Normal_Macro_DocumentOpen.docm
SUSPICIOUS|AWX  |OpX:|PadCrypt_invoice_M60244.docm
SUSPICIOUS|AWX  |OpX:|RottenKitten_266CFE755A0A66776DF9FD8CD2FEE1F1.xlsb
SUSPICIOUS|AWX  |OLE:|TA505 2019 Letter 7711.xls

Flags: A=AutoExec, W=Write, X=Execute
```

OLETOOLS

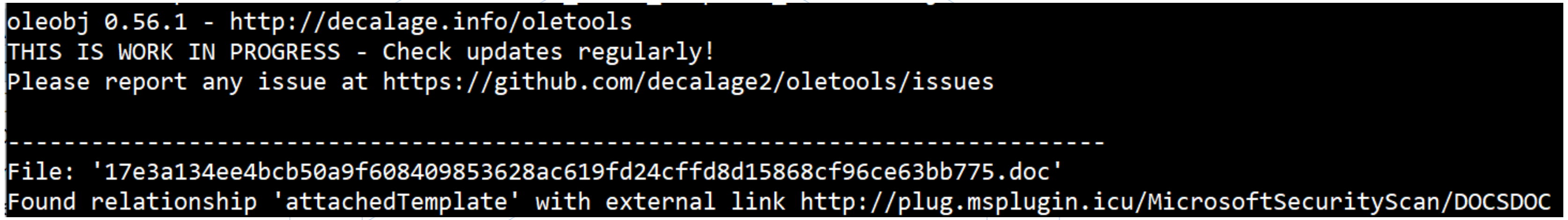
rtfobj/oleobj: detect suspicious OLE objects in RTF and MS Office files

Examples :

- **OLE Package** objects containing executable files
- **Exploits for vulnerabilities**, such as Equation Editor
- **Remote Attached Templates** with Macros
- **Remote OLE objects**, such as Follina

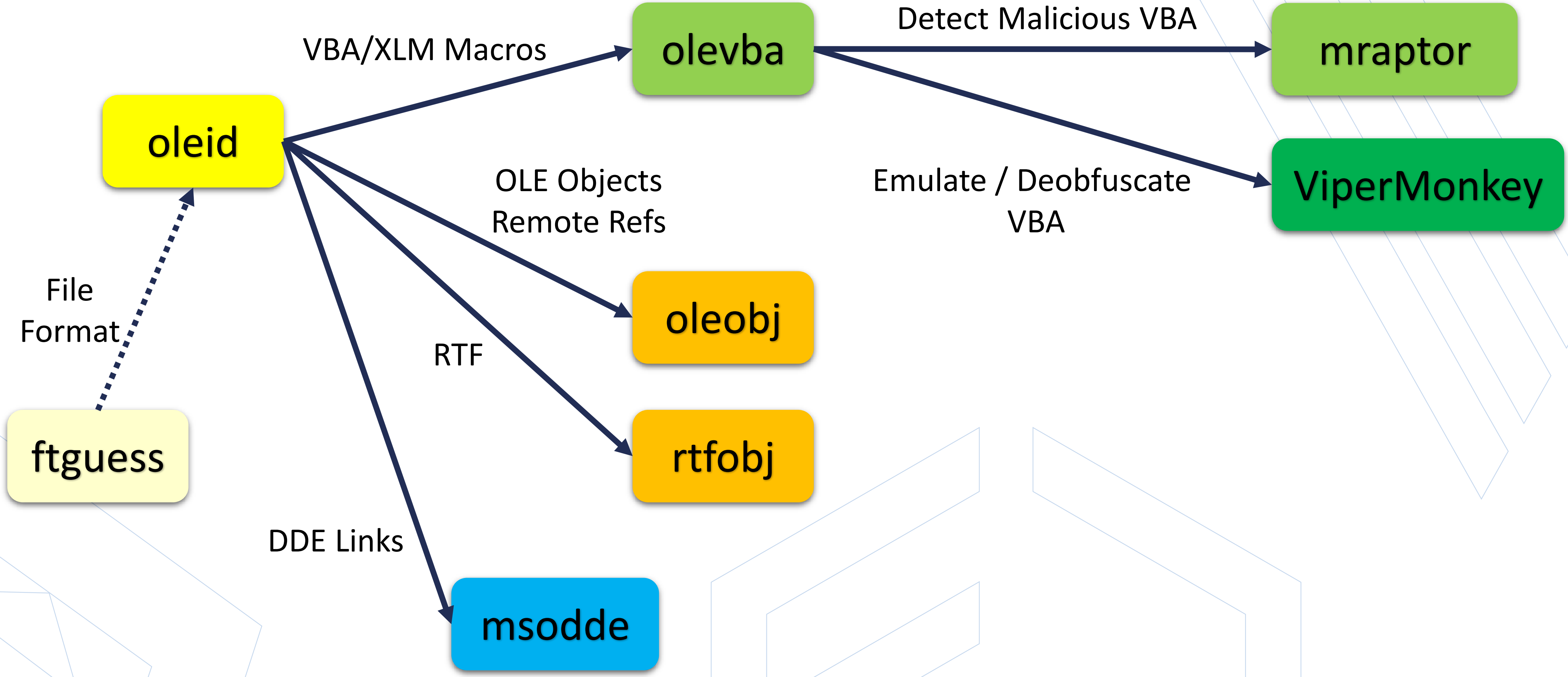


```
rtfobj 0.60.dev2 on Python 2.7.18 - http://decalage.info/python/oletools
THIS IS WORK IN PROGRESS - Check updates regularly!
Please report any issue at https://github.com/decalage2/oletools/issues

===============================================================================
File: 'RTF_OLEPkg_EXE.rtf' - size: 950601 bytes
---+----------+---------------------------------------------------------------
id |index     |OLE Object
---+----------+---------------------------------------------------------------
0  |00062FC6h |format_id: 2 (Embedded)
   |          |class name: 'Package'
   |          |data size: 246496
   |          |OLE Package object:
   |          |Filename: u'pm02.exe'
   |          |Source path: u'C:\\Aaa\\exe\\pm02.exe'
   |          |Temp path = u'C:\\Users\\M\\AppData\\Local\\Temp\\pm02.exe'
   |          |MD5 = 'c44ac001b67fef80d0f46de594a615a8'
   |          |EXECUTABLE FILE
   |          |File Type: Windows PE Executable
---+----------+---------------------------------------------------------------
```



```
oleobj 0.56.1 - http://decalage.info/oletools
THIS IS WORK IN PROGRESS - Check updates regularly!
Please report any issue at https://github.com/decalage2/oletools/issues

-------------------------------------------------------------------------------
File: '17e3a134ee4bcb50a9f608409853628ac619fd24cffd8d15868cf96ce63bb775.doc'
Found relationship 'attachedTemplate' with external link http://plug.msplugin.icu/MicrosoftSecurityScan/DOCSDOC
```

Oletools usage



Why are documents still a threat today ?

ATTACK TECHNIQUES

Techniques to run malicious code from MS Office documents

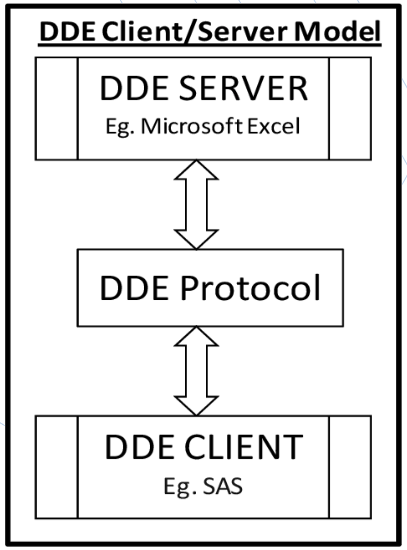
Microsoft® Visual Basic® for Applications™

- VBA Macros
- Excel 4 Macros
- DDE
- OLE Package objects
- Vulnerabilities

About Microsoft Excel

Microsoft

Microsoft Excel Version 4.0
Copyright © 1985-1992
This copy of Microsoft Excel is licensed to:
User
Company


DDE Client/Server Model

DDE SERVER
Eg. Microsoft Excel

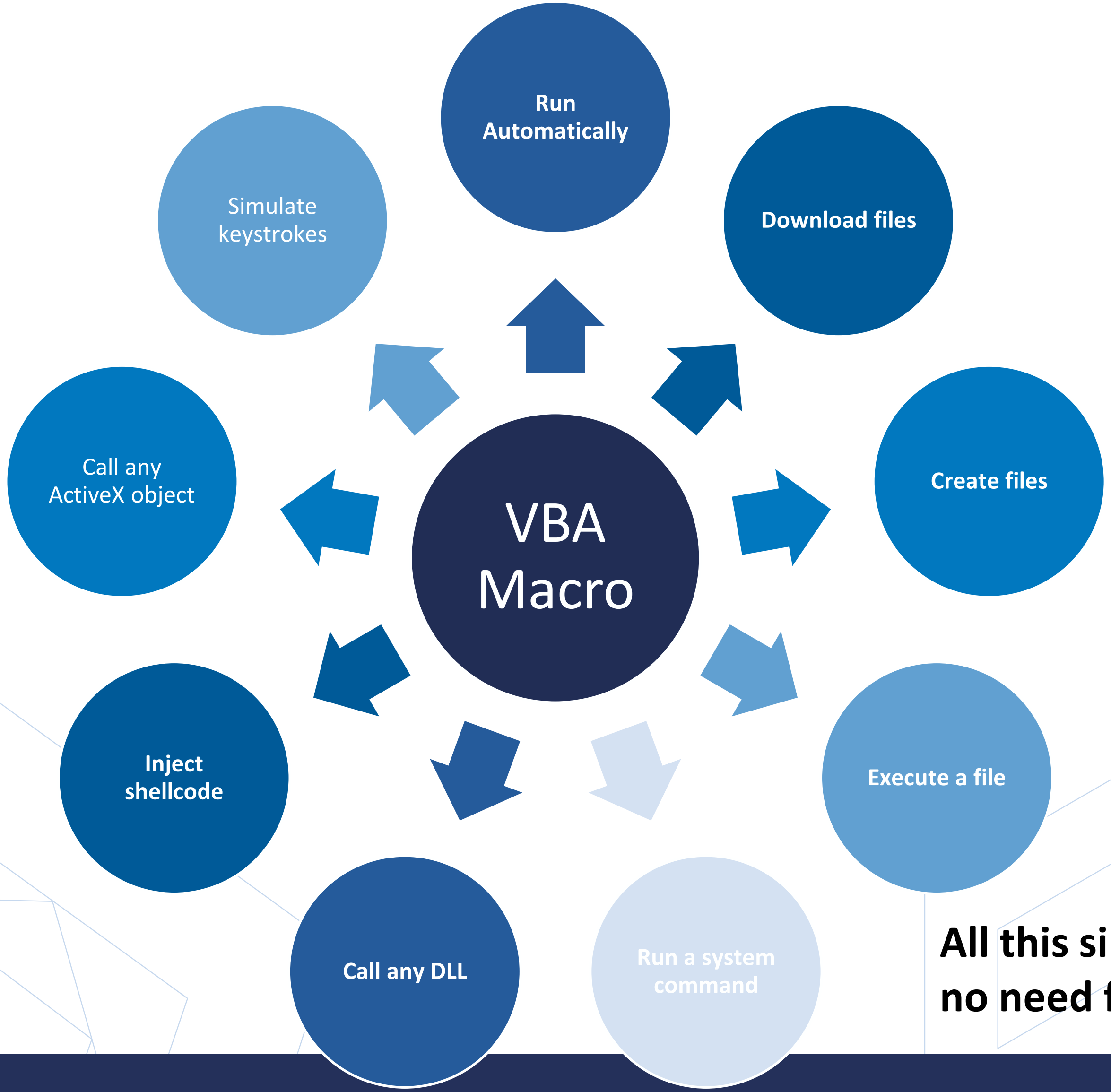
DDE Protocol

DDE CLIENT
Eg. SAS

CVE-2023-36884

MS Office Zero-Day Vulnerability Exploited For Espionage

WHAT CAN A MALICIOUS MACRO DO?



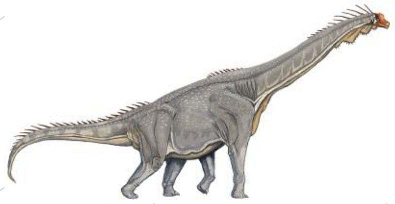
Note: It is possible to write malware completely in VBA.
But in practice, VBA macros are mostly used to write **Droppers** or **Downloaders**, to trigger other stages of malware.

All this simply using native MS Office features available since 1997, no need for any exploit !

A HISTORY OF MACROS

Office 95/97

- 95: WordBasic
- 97: VBA - **simple Yes/No prompt** to enable macros



1995-2003

- **Macrovirus era**
- Concept, Laroux, Melissa, Lexar

Office 2000/XP/2003

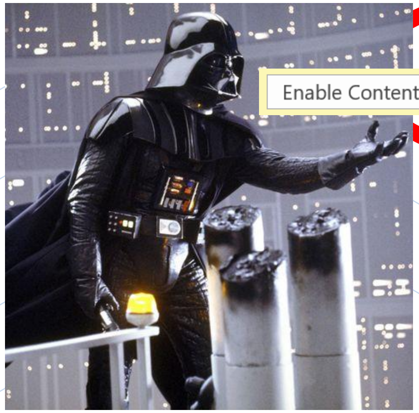
- Unsigned macros are **DISABLED BY DEFAULT**

2004-2013

- **VBA winter**
- Attackers prefer exploits

Office 2010 / 2013 / 2016 / 365

- **Single "Enable Content" button** AFTER seeing the document (Lures)…
- Sandbox against exploits (Protected View)



2014-2023

VBA Macros come back

- Used as first stage to deliver malware
- 100,000s of phishing e-mails per day
- Banking Trojans, Ransomware, APTs, …

ATTACK TECHNIQUES

All those techniques have been « patched » by Microsoft, right ?

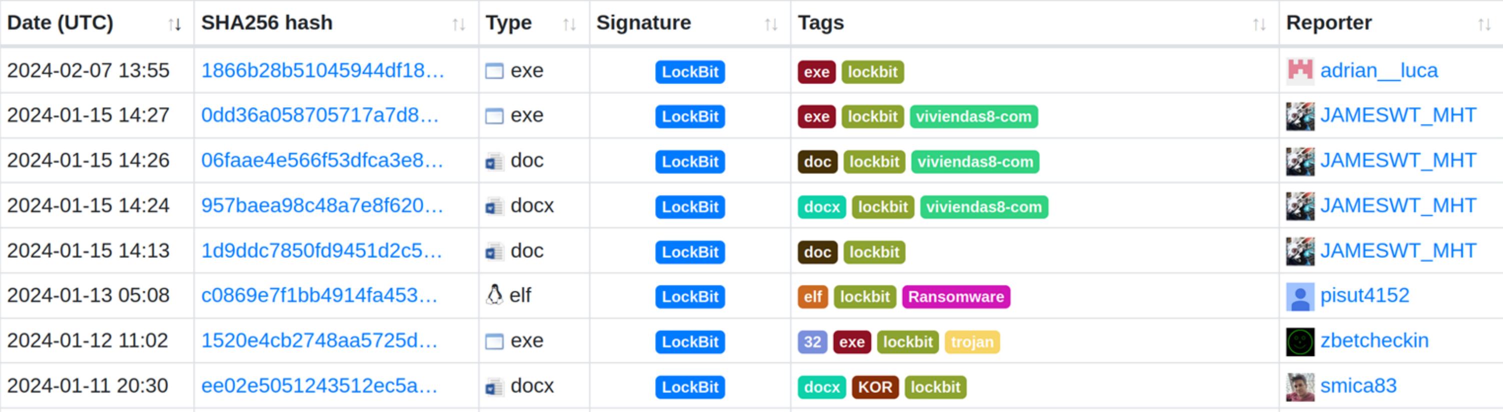
YES! :

- VBA Macros
  - 2023 : Blocked by default for files coming from the Internet
- Excel 4 Macros
  - 2022 : Disabled by default
- DDE
  - 2017 : disabled by default in Word
  - 2019 : disabled by default in Excel
- OLE Package objects
  - 2020 : Blocked for executable files
- Vulnerabilities
  - CVE-2017-11882 (Equation Editor), CVE-2021-40444, CVE-2022-30190 (Follina), CVE-2023-36884, ...

ATTACK TECHNIQUES

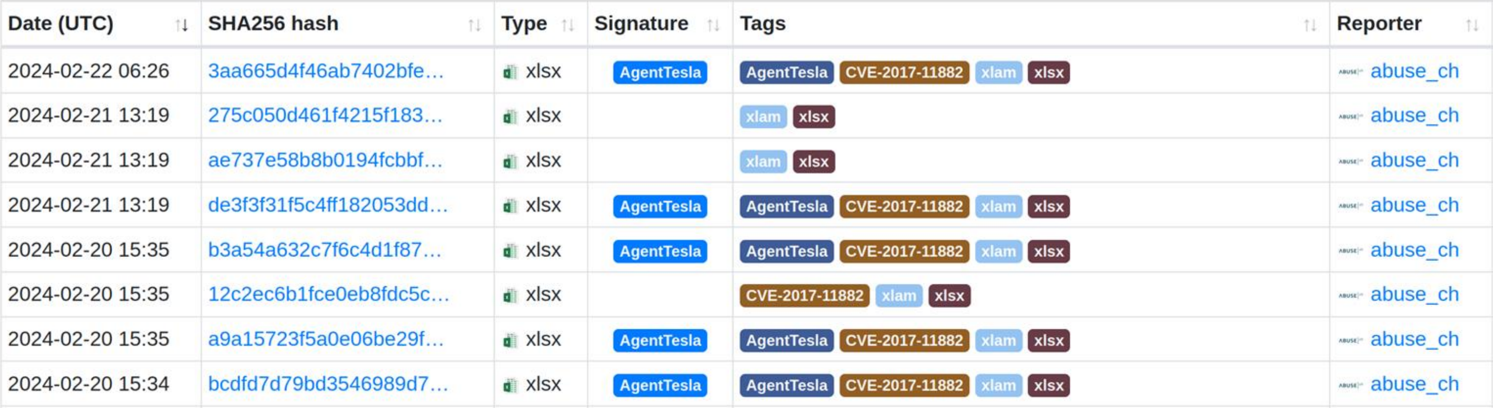
But… Threat Actors are still using Malicious Documents in 2024

- LockBit ransomware using Remote Templates with VBA Macros :



Date (UTC)	SHA256 hash	Type	Signature	Tags	Reporter
2024-02-07 13:55	1866b28b51045944df18…	exe	LockBit	exe lockbit	adrian__luca
2024-01-15 14:27	0dd36a058705717a7d8…	exe	LockBit	exe lockbit viviendas8-com	JAMESWT_MHT
2024-01-15 14:26	06faae4e566f53dfca3e8…	doc	LockBit	doc lockbit viviendas8-com	JAMESWT_MHT
2024-01-15 14:24	957baea98c48a7e8f620…	docx	LockBit	docx lockbit viviendas8-com	JAMESWT_MHT
2024-01-15 14:13	1d9ddc7850fd9451d2c5…	doc	LockBit	doc lockbit	JAMESWT_MHT
2024-01-13 05:08	c0869e7f1bb4914fa453…	elf	LockBit	elf lockbit Ransomware	pisut4152
2024-01-12 11:02	1520e4cb2748aa5725d…	exe	LockBit	32 exe lockbit trojan	zbetcheckin
2024-01-11 20:30	ee02e5051243512ec5a…	docx	LockBit	docx KOR lockbit	smica83

- AgentTesla using XLAM – Excel Add-ins with VBA Macros



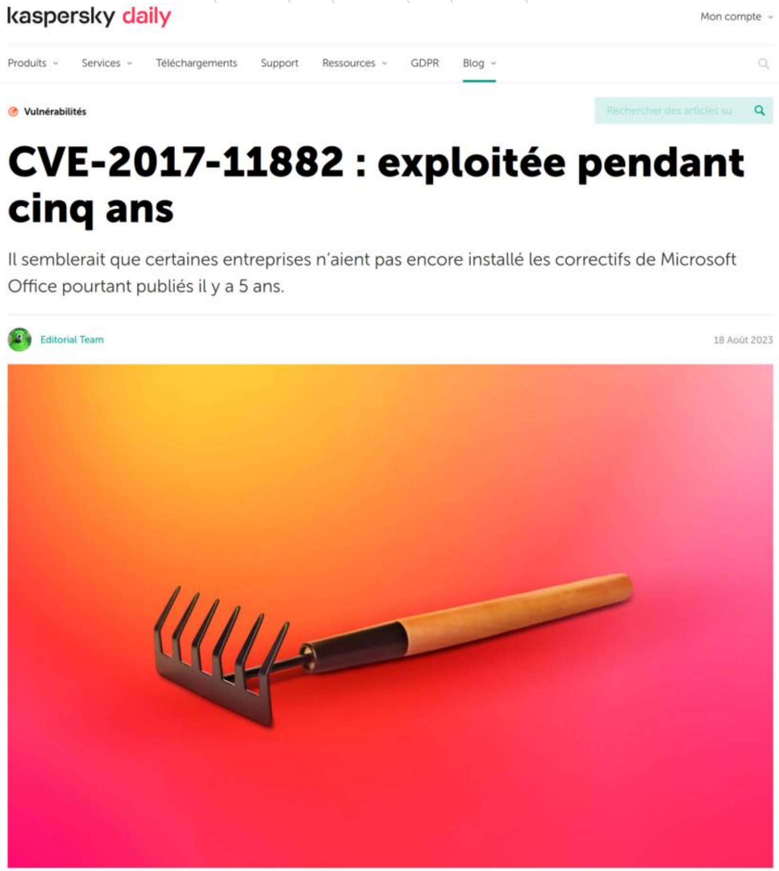
Date (UTC)	SHA256 hash	Type	Signature	Tags	Reporter
2024-02-22 06:26	3aa665d4f46ab7402bfe…	xlsx	AgentTesla	AgentTesla CVE-2017-11882 xlam xlsx	abuse_ch
2024-02-21 13:19	275c050d461f4215f183…	xlsx		xlam xlsx	abuse_ch
2024-02-21 13:19	ae737e58b8b0194fcbbf…	xlsx		xlam xlsx	abuse_ch
2024-02-21 13:19	de3f3f31f5c4ff182053dd…	xlsx	AgentTesla	AgentTesla CVE-2017-11882 xlam xlsx	abuse_ch
2024-02-20 15:35	b3a54a632c7f6c4d1f87…	xlsx	AgentTesla	AgentTesla CVE-2017-11882 xlam xlsx	abuse_ch
2024-02-20 15:35	12c2ec6b1fce0eb8fdc5c…	xlsx		CVE-2017-11882 xlam xlsx	abuse_ch
2024-02-20 15:35	a9a15723f5a0e06be29f…	xlsx	AgentTesla	AgentTesla CVE-2017-11882 xlam xlsx	abuse_ch
2024-02-20 15:34	bcdfd7d79bd3546989d7…	xlsx	AgentTesla	AgentTesla CVE-2017-11882 xlam xlsx	abuse_ch

ATTACK TECHNIQUES

All those techniques have been « patched » by Microsoft, right ?

Yes, but :

- Not all computers are fully patched.
  - Threat actors still use exploits from 2017 !
  - In 2024 malicious macros are still coming in.
  - Attackers only need one unpatched computer to enter.
- Some users need to use macros.
  - Finance department getting XLS files every day from partner
- Every year, new vulnerabilities are discovered
  - Example : the new MS Outlook CVE-2024-21413 allows a link to launch a document without Protected View
- Some techniques still work for lateral movement
  - Documents on SharePoint/OneDrive can still run macros
  - Documents sent by email within a company can still run macros
- Attackers sometimes find ways to bypass fences
  - Blocked features are not 100 % blocked
  - Example : Document with macros in an ISO image file, OLE Package with drag and drop



Vulnérabilités

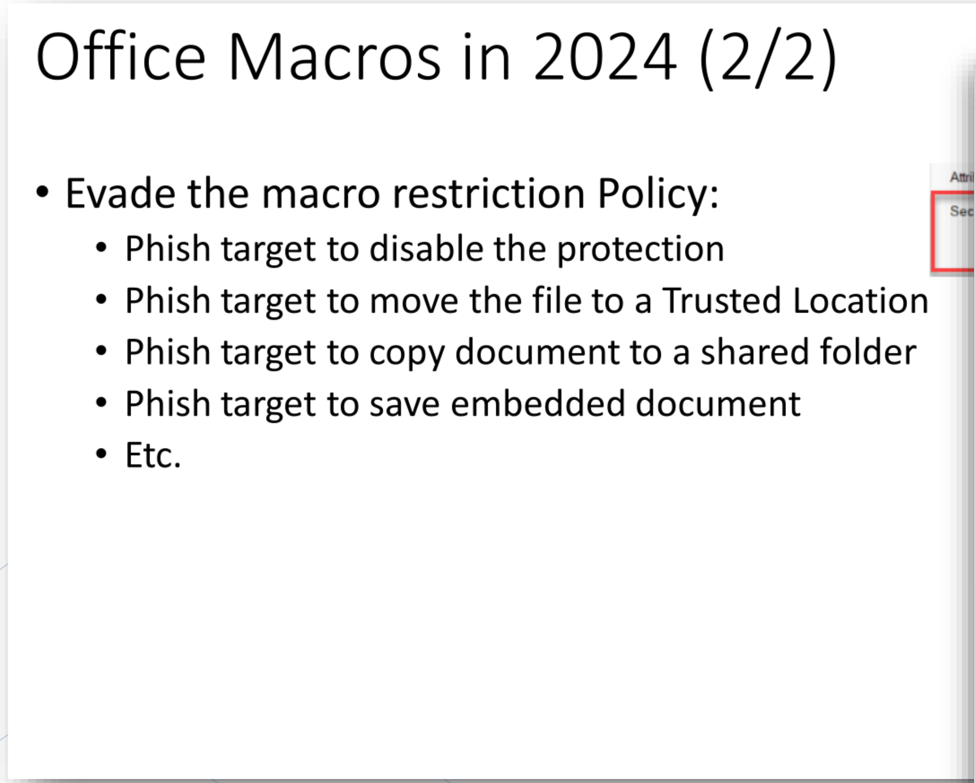
CVE-2017-11882 : exploitée pendant cinq ans

Il semblerait que certaines entreprises n'aient pas encore installé les correctifs de Microsoft Office pourtant publiés il y a 5 ans.

Editorial Team

18 Août 2023

Some Red Teamers still use VBA Macros

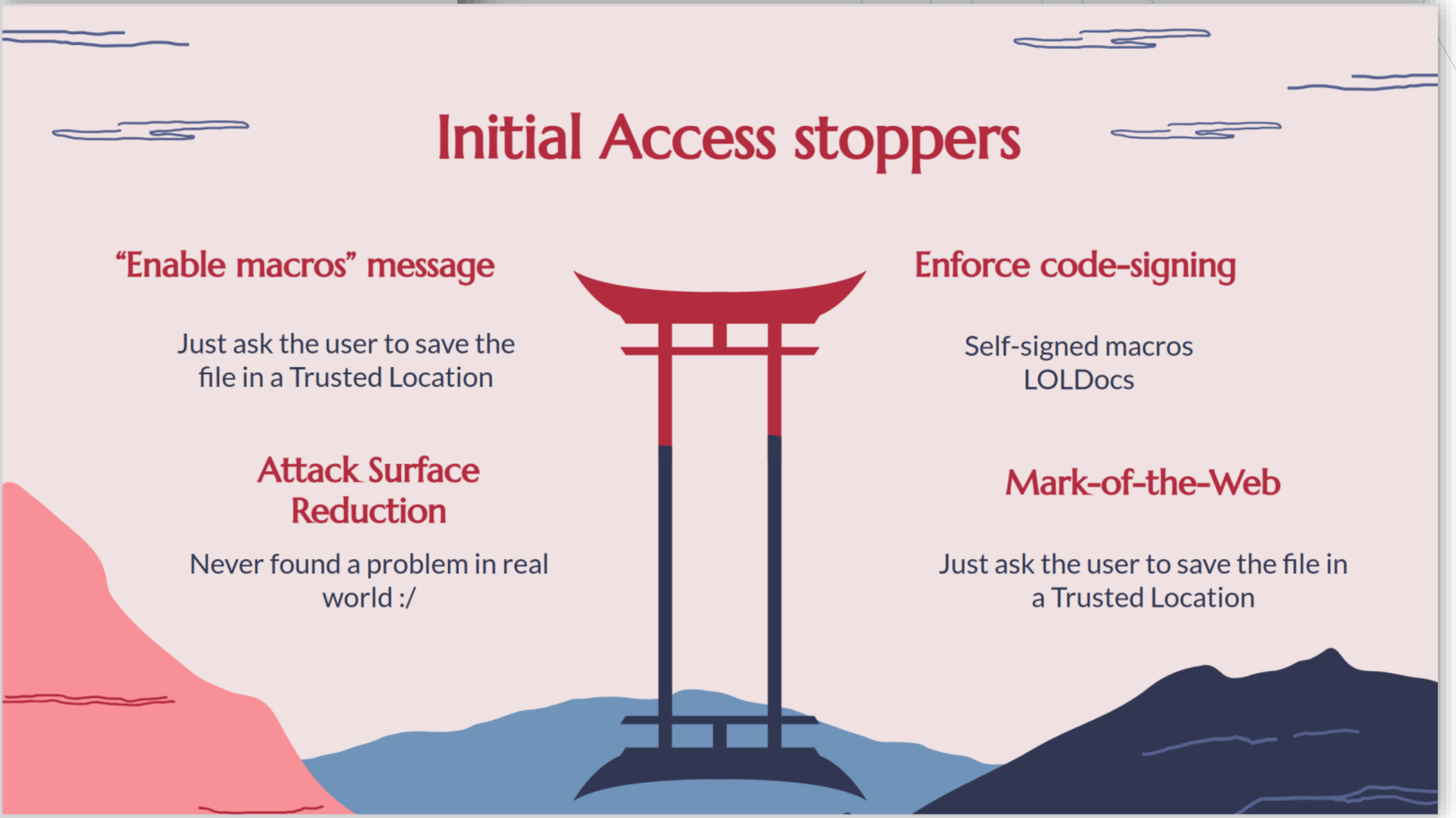


Office Macros in 2024 (2/2)

- Evade the macro restriction Policy:
  - Phish target to disable the protection
  - Phish target to move the file to a Trusted Location
  - Phish target to copy document to a shared folder
  - Phish target to save embedded document
  - Etc.

https://www.linkedin.com/posts/emeric-nasi-84950528_extended-slides-activity-7211761335305437184-9nG_?utm_source=share&utm_medium=member_desktop

https://github.com/X-C3LL/congresos-slides/blob/master/Offensive%20VBA.pdf



Initial Access stoppers

“Enable macros” message

Just ask the user to save the file in a Trusted Location

Attack Surface Reduction

Never found a problem in real world :/

Enforce code-signing

Self-signed macros
LOLDocs

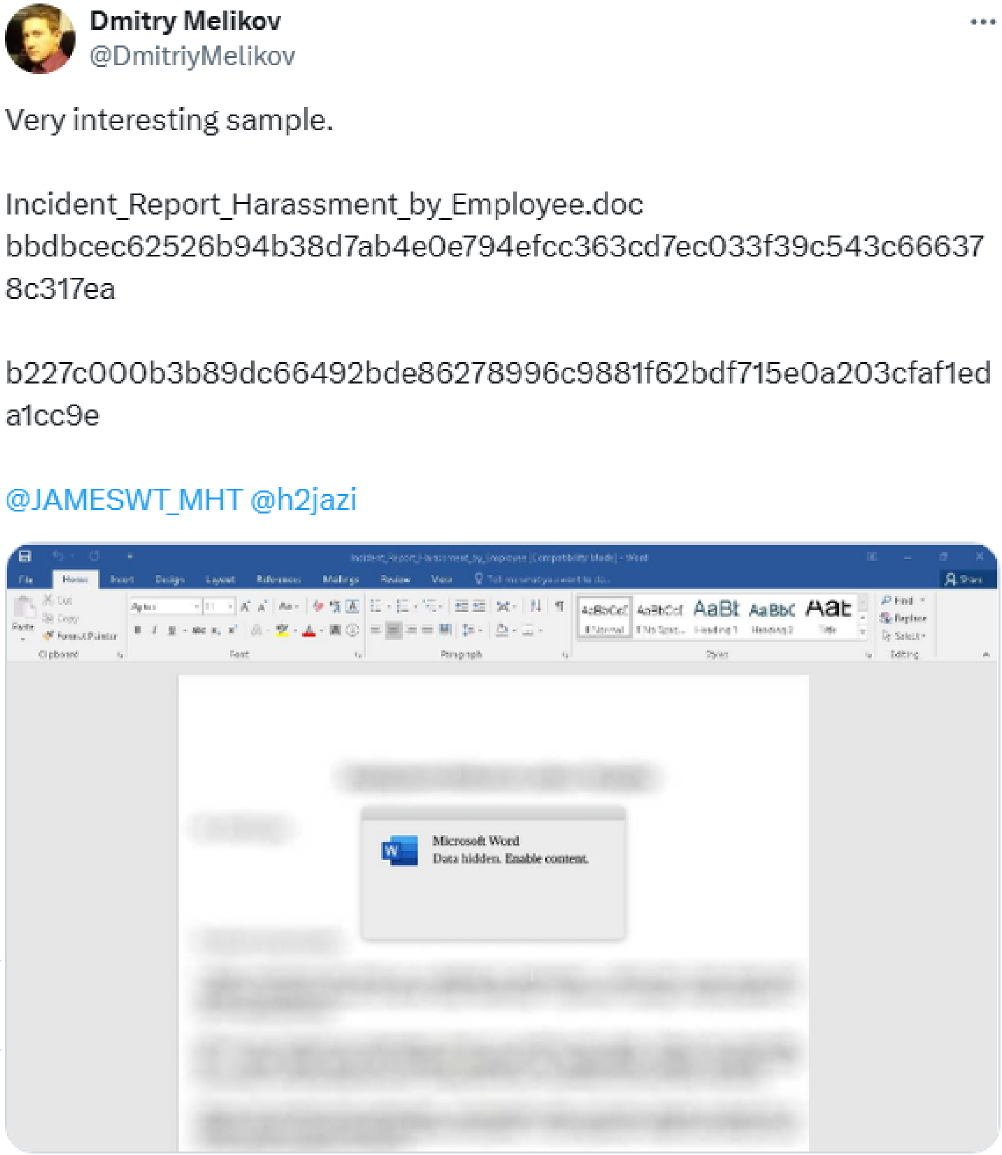
Mark-of-the-Web

Just ask the user to save the file in a Trusted Location

DEMOS

Demo 1

- Recent sample with VBA macro



Dmitry Melikov
@DmitriyMelikov

Very interesting sample.

Incident_Report_Harassment_by_Employee.doc
bbdbcec62526b94b38d7ab4e0e794efcc363cd7ec033f39c543c66637
8c317ea

b227c000b3b89dc66492bde86278996c9881f62bdf715e0a203cfaf1ed
a1cc9e

@JAMESWT_MHT @h2jazi

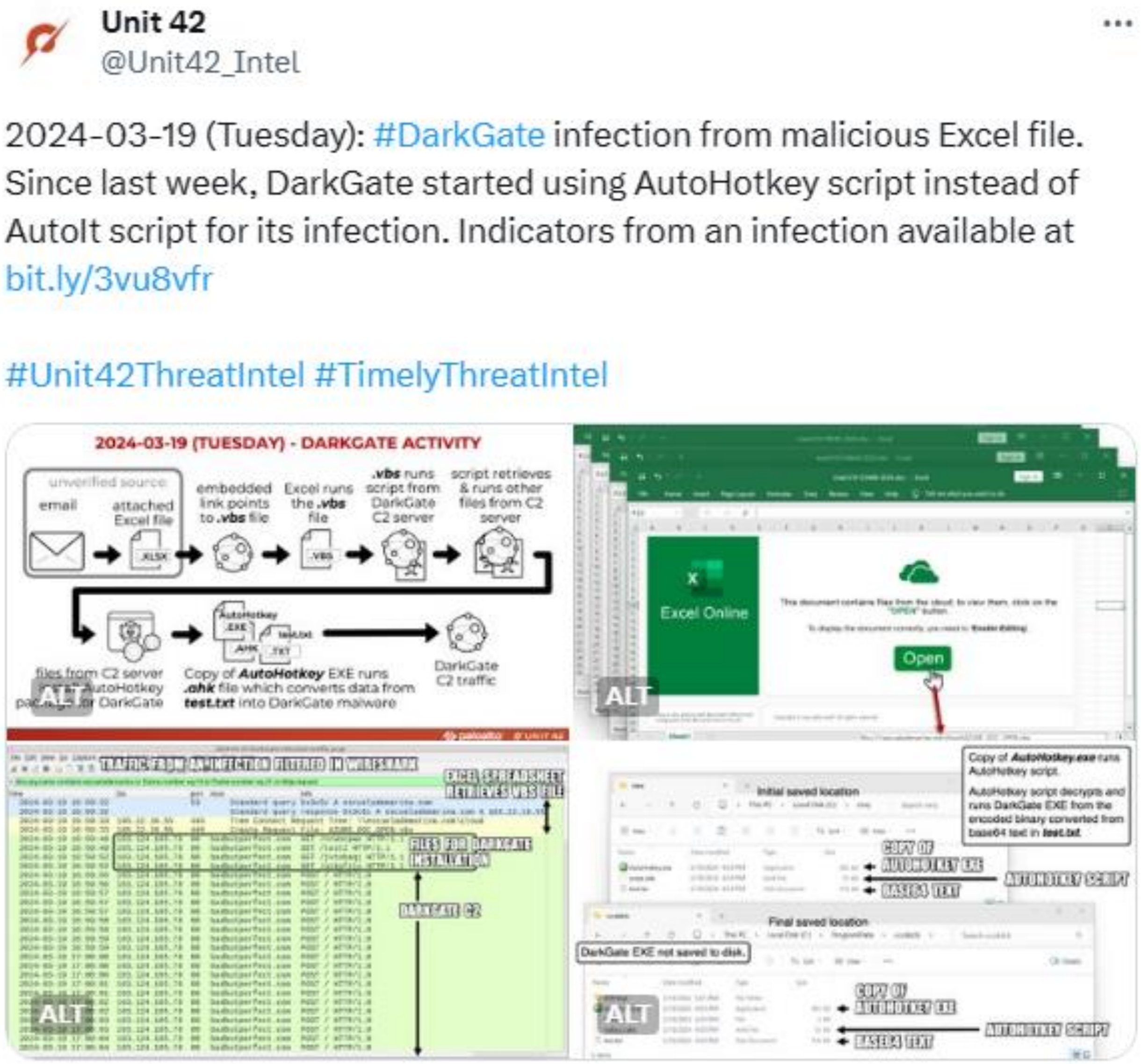
Microsoft Word
Data hidden. Enable content.

Demo 2

- Hancitor sample with VBA macro and OLE package object

Demo 3

- Recent XLSX with malicious link



Unit 42
@Unit42_Intel

2024-03-19 (Tuesday): #DarkGate infection from malicious Excel file. Since last week, DarkGate started using AutoHotkey script instead of AutoIt script for its infection. Indicators from an infection available at bit.ly/3vu8vfr

#Unit42ThreatIntel #TimelyThreatIntel

2024-03-19 (TUESDAY) - DARKGATE ACTIVITY
unverified source
email
attached Excel file
.XLSX
embedded link points to .vbs file
Excel runs the .vbs file
.VBS
.vbs runs script from DarkGate C2 server
script retrieves & runs other files from C2 server
files from C2 server AutoHotkey package for DarkGate
AutoHotkey .EXE
.AHK
test.txt .TXT
Copy of AutoHotkey EXE runs .ahk file which converts data from test.txt into DarkGate malware
DarkGate C2 traffic
paloalto UNIT 42
ALT
Excel Online
Open
ALT
TRAFFIC FROM AN INFECTION FILTERED IN WIRESHARK
EXCEL SPREADSHEET RETRIEVES VBS FILE
FILES FOR DARKGATE INSTALLATION
DARKGATE C2
ALT
Initial saved location
Copy of AutoHotkey.exe runs AutoHotkey script.
AutoHotkey script decrypts and runs DarkGate EXE from the encoded binary converted from base64 text in test.txt
COPY OF AUTOHOTKEY EXE
AUTOHOTKEY SCRIPT
BASE64 TEXT
Final saved location
DarkGate EXE not saved to disk.
COPY OF AUTOHOTKEY EXE
AUTOHOTKEY SCRIPT
BASE64 TEXT
ALT

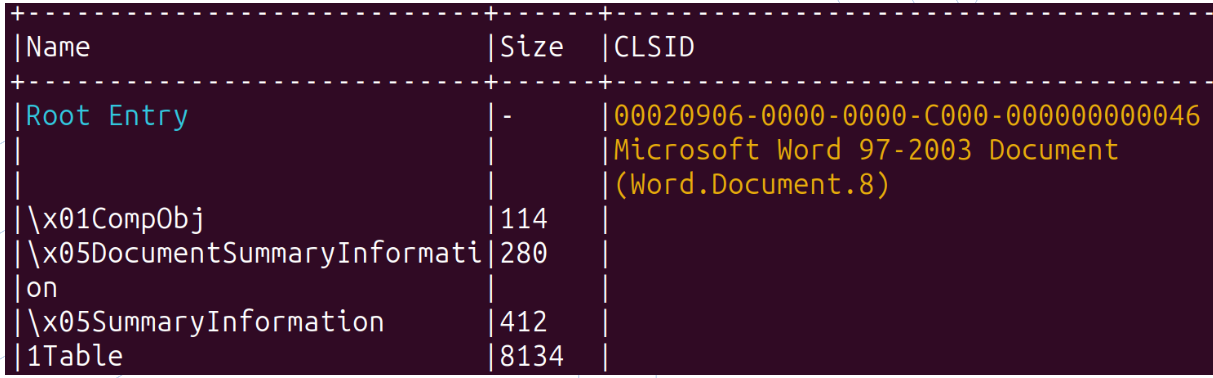
Ftguess – File Format Identification

Ftguess – why a new file type guessing tool

- **file/libmagic** are good most of the time, but not very accurate for MS Office formats and some others like MSIX, PowerShell, JavaScript, VBScript.
- **TrID** is a bit better.
- **Magika** (recent tool from Google based on ML) works well on script formats, but version 1 does not know many formats yet.

- Ftguess implements my custom algorithm to detect file formats more precisely and **only by content**, especially MS Office formats.

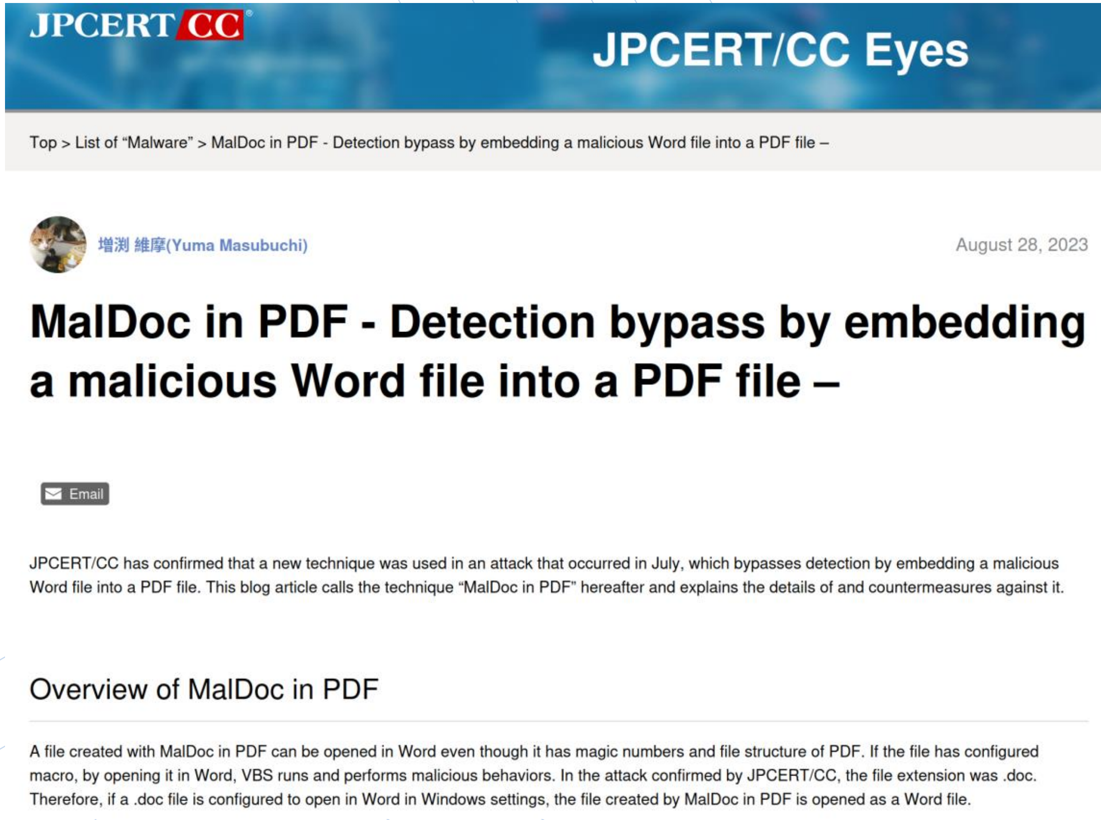
- For example:
  - If file starts by OLE Magic “D0CF”: parse OLE structure
    - Get CLSID from root storage
    - Each CLSID maps to a specific file format:
      - Word 97, Excel 97, etc
  - If file is a ZIP archive: look for specific XML files
    - Check URL of main relationship
    - Each URL maps to a specific OpenXML file format
      - Word 2007, Excel 2007, etc



```
+----------------------------+------+-------------------------------------
|Name                        |Size  |CLSID
+----------------------------+------+-------------------------------------
|Root Entry                  |-     |00020906-0000-0000-C000-000000000046
|                            |      |Microsoft Word 97-2003 Document
|                            |      |(Word.Document.8)
|\x01CompObj                 |114   |
|\x05DocumentSummaryInformati|280   |
|on                          |      |
|\x05SummaryInformation      |412   |
|1Table                      |8134  |
```

Ftguess - future

- But OS like Windows and Linux use **first the file extension** and second the file content to decide which app should open the file.
- This is abused by **polyglots** - example: a MHT file with VBA macros, with a fake PDF header:
  - Most tools see the file as PDF
  - On Windows, it is opened by Word due to the .doc extension
- Ftguess will add a new **strict mode** to be more accurate:
  1. Identify potential formats based on file extension
  2. For each, verify that the content matches the extension
- Example: if file extension = .doc
  - Check if content is Word 97-2003
  - Or Word 2007+
  - Or RTF
  - Or MHT
  - Or plain text
- Only relevant if file has not been renamed.



JPCERT CC

JPCERT/CC Eyes

Top > List of "Malware" > MalDoc in PDF - Detection bypass by embedding a malicious Word file into a PDF file –

増渕 維摩(Yuma Masubuchi)

August 28, 2023

MalDoc in PDF - Detection bypass by embedding a malicious Word file into a PDF file –

Email

JPCERT/CC has confirmed that a new technique was used in an attack that occurred in July, which bypasses detection by embedding a malicious Word file into a PDF file. This blog article calls the technique "MalDoc in PDF" hereafter and explains the details of and countermeasures against it.

Overview of MalDoc in PDF

A file created with MalDoc in PDF can be opened in Word even though it has magic numbers and file structure of PDF. If the file has configured macro, by opening it in Word, VBS runs and performs malicious behaviors. In the attack confirmed by JPCERT/CC, the file extension was .doc. Therefore, if a .doc file is configured to open in Word in Windows settings, the file created by MalDoc in PDF is opened as a Word file.

MSI / MSIX

MSI / MSIX

- MSI and MSIX: installer packages for MS Windows
- Both abused in the recent years to deliver malware
- **MSI: OLE format** (like .doc, .xls)
  - Introduced in 1999 (with Office 2000)
  - Custom Actions can be added to run scripts, EXE or DLL
  - A legitimate MSI can be backdoored
  - Can be parsed by olefile/oletools but needs additional processing to extract useful info and detect malware
  - Undocumented file format
    - Custom encoding algorithm for stream names
    - Custom database format inside some streams

MSI / MSIX

- **MSIX: OpenXML** format (like .docx, .xlsx)
  - Not exactly OpenXML, just a subset
  - Very recent format (2018), not well known and not well supported by security tools
  - Could also be parsed and analysed by oletools
  - Requires digital signature (but some threat actors use stolen keys or buy them)
  - Bonus point: not yet in the list of MS Outlook blocked attachments!
  - Undocumented file format
    - But XML files are easier to parse

olemsi

- New tool under development to parse MSI / MSIX and extract useful information + embedded executable files and scripts
- **Existing tools are not sufficient**:
  - **Msidump is great to analyse malicious MSI but requires Windows**
  - **Lessmsi also requires Windows**
  - **Msitools require Linux, not security oriented**
  - **Not found any suitable tool or parser for MSIX**
- **Work in progress**:
  - **Done: decode OLE stream names from MSI**
  - **MSI: Extract embedded files from streams and CAB files**
  - **MSI: Parse databases, extract custom actions**
  - **MSI: Extract scripts**
  - **MSIX: Parse manifest**
  - **MSIX: Parse config.json**
  - **MSIX Extract embedded files and scripts**
  - **Detect suspicious MSI / MSIX**

Any questions:

@decalage2 on X

@decalage@mastodon.social