

We open the file given using pdfid.

We see that only that the second file contains javascript and embedded file which could be possible ways for malicious content to enter.

```

kali@kali: ~/Downloads
File Actions Edit View Help
└─$ pdfid cw_pdf_files/cw_pdf_sample1.pdf
PDFID 0.2.8 cw_pdf_files/cw_pdf_sample1.pdf
PDF Header: XPDF-1.4
obj      86
endobj   86
stream   50
endstream
xref      2
trailer   2
startxref
/Page     3
/Encrypt   0
/ObjStm    0
/JS        0
/JavaScript 0
/AA        0
/OpenAction 0
/AcroForm  0
/3BIG2Decode 0
/RichMedia 0
/Launch    0
/EmbeddedFile 0
/XFA       0
/Colors > 2*24 0

(kali@kali) [~/Downloads]
└─$ pdfid cw_pdf_files/cw_pdf_sample2.pdf
PDFID 0.2.8 cw_pdf_files/cw_pdf_sample2.pdf
PDF Header: XPDF-1.6
obj      146
endobj   146
stream   55
endstream
xref      1
trailer   1
startxref
/Page     1
/Encrypt   0
/ObjStm    2
/JS        2
/JavaScript 2
/AA        2
/OpenAction 0
/AcroForm  1
/3BIG2Decode 0
/RichMedia 0
/Launch    0
/EmbeddedFile 1
/XFA       0
/Colors > 2*24 0

```

Thus we begin scanning the 2<sup>nd</sup> pdf. First searching for /JS and /Names we find no leads.

```

(kali@kali) [~/Downloads/cw_pdf_files]
└─$ pdf-parser cw_pdf_sample2.pdf | grep /Names
/Names: [{"by\x00-\x00i\x00c\x00o\x00n\x00r\x00P\x00o\x00i\x00n\x00t\x00e\x00r\x00+\x002\x005\x005\x00:\x000\x00:\x000\x00-\x00E\x00N\x00U\x00-\x000} 52 0 R ]'

(kali@kali) [~/Downloads/cw_pdf_files]
└─$ pdf-parser cw_pdf_sample2.pdf | grep /J
/J: '(AFDate_FormatEx\\(''mm/dd/yyyy\\')\;)'
/J: /JavaScript
/J: '(AFDate_KeystrokeEx\\(''mm/dd/yyyy\\')\;)'
/J: /JavaScript

(kali@kali) [~/Downloads/cw_pdf_files]
└─$

```

Finally we run a command which might contain something relevant.

```

(kali@kali) [~/Downloads/cw_pdf_files]
└─$ pdf-parser -f cw_pdf_sample2.pdf

```

so I redirect the output in a file.

```

(kali@kali) [~/Downloads/cw_pdf_files]
└─$ pdf-parser -f cw_pdf_sample2.pdf > embedded.txt

```

Skimming through the file I see some possibly obfuscated code.

[illegible]

This might contain the javascript code that is malicious.

### Task 2:

Scanning the first docx file we see that It contains external entities. Using the oleobj tool we see the external link it references.

```
kali@kali: ~/Downloads
File Actions Edit View Help
cw_pdf_files.7z  incrediblyPolishedResume.docx  sheetsForFinancial.xlsm

(kali@kali)~/Downloads
$ oleid incrediblyPolishedResume.docx
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: incrediblyPolishedResume.docx

+-----+-----+-----+
|Indicator|Value|Risk|Description|
+-----+-----+-----+
|File format|MS Word 2007+ Document (.docx)|info||
+-----+-----+-----+
|Container format|OpenXML|info|Container type|
+-----+-----+-----+
|Encrypted|False|none|The file is not encrypted|
+-----+-----+-----+
|VBA Macros|No|none|This file does not contain VBA macros.|
+-----+-----+-----+
|XLM Macros|No|none|This file does not contain Excel 4/XLM macros.|
+-----+-----+-----+
|External Relationships|1|HIGH|External relationships found: attachedTemplate - use oleobj for details|
+-----+-----+-----+

(kali@kali)~/Downloads
$ oleobj incrediblyPolishedResume.docx
oleobj 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

File: 'incrediblyPolishedResume.docx'
Found relationship 'attachedTemplate' with external link http://somtaw.warship.kuunlaan.local/macro3.dotm
```

At the bottom we can see the link.

The second file we use the same oleid tool again.

```
kali@kali: ~/Downloads
File Actions Edit View Help

(kali@kali)~/Downloads
$ oleid sheetsForFinancial.xlsm
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: sheetsForFinancial.xlsm
WARNING For now, VBA stomping cannot be detected for files in memory

Indicator | Value | Risk | Description
-----|-----|-----|-----
File format | MS Excel 2007+ Macro-Enabled Workbook (.xlsm) | info |
Container format | OpenXML | info | Container type
Encrypted | False | none | The file is not encrypted
VBA Macros | Yes, suspicious | HIGH | This file contains VBA macros. Suspicious keywords were found. Use |olevba and mraptor for more info.
XLM Macros | No | none | This file does not contain Excel 4/XLM macros.
External Relationships | 0 | none | External relationships such as remote templates, remote OLE objects, etc

(kali@kali)~/Downloads
$
```

We see it contains malicious macros.  
Running olevba and mraptor for more info.

```
kali@kali: ~/Downloads
File Actions Edit View Help

VBA MACRO Sheet1.cls
in file: xl/VbaProject.bin - OLE stream: 'VBA/Sheet1'
(empty macro)
-----
|Type|Keyword|Description|
-----|-----|-----|
|AutoExec|Workbook_Open|Runs when the Excel Workbook is opened|
|Suspicious|Open|May open a file|
|Suspicious|Write|May write to a file (if combined with Open)|
|Suspicious|Binary|May read or write a binary file (if combined with Open)|
|Suspicious|Adodb.Stream|May create a text file|
|Suspicious|SaveToFile|May create a text file|
|Suspicious|Shell|May run an executable file or a system command|
|Suspicious|Run|May run an executable file or a system command|
|Suspicious|PowerShell|May run PowerShell commands|
|Suspicious|CreateObject|May create an OLE object|
|Suspicious|Windows|May enumerate application windows (if combined with Shell.Application object)|
|Suspicious|Microsoft.XMLHTTP|May download files from the Internet|
|Suspicious|Hex Strings|Hex-encoded strings were detected, may be used to obfuscate strings (option --decode to see all)|
|Suspicious|Base64 Strings|Base64-encoded strings were detected, may be used to obfuscate strings (option --decode to see all)|
|IOC|http://srv3.wonderba|URL|
|IOC|11financial.local/ab| |
|IOC|c123.crt| |
|IOC|run.ps1|Executable file name|
|IOC|powershell.exe|Executable file name|

(kali@kali)~/Downloads
$

(kali@kali)~/Downloads
$ mraptor: sheetsForFinancial.xlsm
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
#00000000|AMX |Opt|sheetsForFinancial.xlsm

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

(kali@kali)~/Downloads
$
```