

Post Mortem Analysis Techniques of Fake Invoices

Manipulated PDF documents



CIRCL

Computer Incident
Response Center
Luxembourg

Team CIRCL
G rard Wagener
TLP:WHITE

<http://www.circl.lu/>
Twitter: @circl.lu

16-17 May, 2019

Reported fraud

Detoured invoices

- Supplier sends payment reminders to customers
- Customer answers that he paid, showing a proof of payment
- Supplier says that it is not his bank account details

Reported fraud

Detoured invoices

Open questions

- Was the invoice created from scratch?
 - By the accounting system itself?
 - By a third party tool?
- By a manipulation of an existing invoice
 - By the accounting system itself?
 - By a third party tool?
 - Where was the original invoice created?
 - Where was it intercepted?
 - Under which form was it intercepted? (scan, office documents)

PDF internals

PDF data structure

| |
|------------|
| %PDF-1.5 |
| 1 0 obj |
| ... |
| endobj |
| 2 0 obj |
| ... |
| endobj |
| obj |
| ... |
| endobj |

| |
|------------------------|
| obj |
| /Type /XRef |
| /Index [0 113] |
| /Size 113 |
| /W [1 3 1] |
| /Root 110 0 R |
| /ID [<C173A17AE5> ...] |
| startxref offset |
| %%EOF |

PDF internals

Why bothering with these details?

because of ...

- Many different PDF format variants
- `www.adobe.com/devnet/pdf/pdf_reference_archive.html`
- Not all tools interpret them correctly
- Tools strip potential valuable information
 - Comments left by the creator software
 - Generation IDs → track original files
 - Manipulation left overs of the "attacker"

Detoured invoices

Practical invoice.pdf analysis

Data to be analyzed

| | |
|-----------------|----------------------------------|
| Filename | invoice.pdf |
| Number of bytes | 27758 |
| MD5 hash | 04a18e4a2b3baf08bd5cb33121842b22 |

Questions

- What version has the PDF?
- How many objects the PDF has?
- What value has is the startxref offset?
- What is at is location?
- How many objects are in the xref table?

Detoured invoices

Practical invoice.pdf analysis

Data to be analyzed

| | |
|-----------------|----------------------------------|
| Filename | invoice.pdf |
| Number of bytes | 27758 |
| MD5 hash | 04a18e4a2b3baf08bd5cb33121842b22 |

Getting PDF version with standard unix tools

```
file invoice.pdf
```

```
head -c 9 invoice.pdf
```

Using pdfid.py from Didier Stevens

```
pdfid.py invoice.pdf
```

Detoured invoices

Practical invoice.pdf analysis

Data to be analyzed

| | |
|-----------------|----------------------------------|
| Filename | invoice.pdf |
| Number of bytes | 27758 |
| MD5 hash | 04a18e4a2b3baf08bd5cb33121842b22 |

Counting objects with standard unix tools

```
strings invoice.pdf | grep "endobj" | wc -l
```

Using pdfid.py from Didier Stevens

```
pdfid.py invoice.pdf
```


Detoured invoices

Practical invoice.pdf analysis

Data to be analyzed

| | |
|-----------------|----------------------------------|
| Filename | invoice.pdf |
| Number of bytes | 27758 |
| MD5 hash | 04a18e4a2b3baf08bd5cb33121842b22 |

Getting the startxref offset with standard unix tools

```
OFFSET='strings invoice.pdf | grep -A 1 "startxref" |  
tail -n 1'
```

Detoured invoices

Practical invoice.pdf analysis

Data to be analyzed

| | |
|-----------------|----------------------------------|
| Filename | invoice.pdf |
| Number of bytes | 27758 |
| MD5 hash | 04a18e4a2b3baf08bd5cb33121842b22 |

Determining xref table with standard unix tools

```
OFFSET='strings invoice.pdf | grep -A 1 "
    startxref" | tail -n 1'
dd if=invoice.pdf bs=1 skip=$OFFSET | less
```

Detoured invoices

Practical invoice.pdf analysis

Data to be analyzed

| | |
|-----------------|----------------------------------|
| Filename | invoice.pdf |
| Number of bytes | 27758 |
| MD5 hash | 04a18e4a2b3baf08bd5cb33121842b22 |

Determining the number of items in the xref table with standard unix tools

```
OFFSET='strings invoice.pdf | grep -A 1 "
    startxref" | tail -n 1'
dd if=invoice.pdf bs=1 skip=$OFFSET | head -n 2 |
    tail -n 1 | cut -d ' ' -f2
```

Detoured invoices

Extracting PDF metadata with pdftinfo

```
pdftinfo invoice.pdf
```

```
Title: SSMILE_prin19041715230
```

```
Creator: SMILE_printer
```

```
Producer: KONICA MINOLTA bizhub C458
```

```
CreationDate: Wed Apr 17 16:23:17 2019 CEST
```

```
ModDate: Wed Apr 17 16:23:17 2019 CEST
```

```
Page size: 595 x 841 pts
```

```
File size: 27758 bytes
```

```
PDF version: 1.4
```

```
...
```

Detoured invoices

Extracting PDF metadata with pdftinfo

Open questions

- Is the creator known?
- Is the producer known?
- Are the timestamps in a valid time frame?
- Does the file size correspond?

Caution

- All elements in a PDF could be manipulated
- The integrity is not guaranteed

PDF dissection

Getting an overview with the tool `pdfid.py`

```
pdfid.py invoice.pdf
```

```
PDFiD 0.2.1 invoice.pdf
```

```
PDF Header: %PDF-1.4
```

```
obj 37
```

```
endobj 37
```

```
stream 16
```

```
endstream 16
```

```
xref 1
```

```
trailer 1
```

```
startxref 1
```

```
/Page 1
```

```
/JavaScript 0
```

```
/OpenAction 1
```

```
/AcroForm 0
```

Checking active components

Items frequently used to load malware

- OpenAction
- JavaScript
- AcroForm

Checking active components

OpenAction

```
python pdf-parser.py -s openaction invoice.pdf
obj 37 0
Type: /Catalog
Referencing: 2 0 R, 34 0 R, 1 0 R

<<
  /Type /Catalog
  /Pages 2 0 R
  /Metadata 34 0 R
  /OpenAction [ 1 0 R /Fit ]
>>
```


Checking active components

OpenAction

```
/OpenAction [ 1 0 R /Fit ]
```

| | |
|--------------------|----------------------|
| Object number | 1 |
| Generation number | 0 |
| Indirect reference | R |
| Fit | Display instructions |

Checking active components

OpenAction

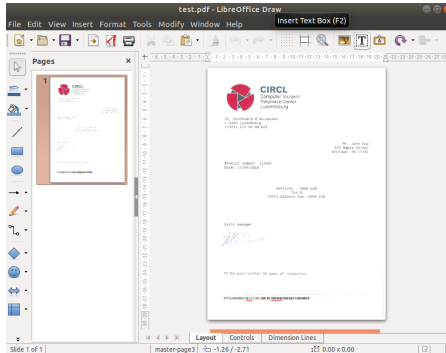
What is at object 1?

```
python pdf-parser.py invoice.pdf -o 1
obj 1 0
Type: /Page
Referencing: 2 0 R, 3 0 R, 4 0 R
<<
  /Type /Page
  /Parent 2 0 R
  /MediaBox [ 0 0 595.000 841.000 ]
  /Resources
    <<
      /ProcSet [ /PDF /Text /ImageB /ImageC /ImageI ]
      ...
```

Detoured invoices

Checking document modifications

- Tools for manipulating PDF documents: LibreOffice, Preview on MacOS, Adobe Acrobat
- Low skills are needed for doing these manipulations



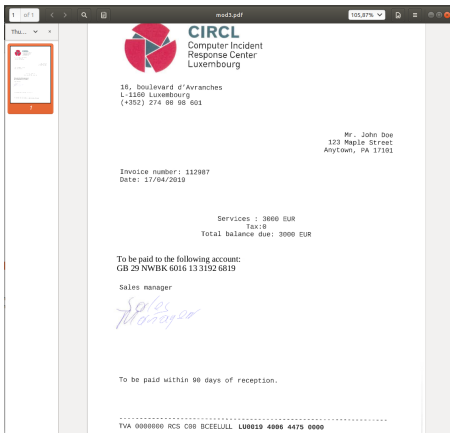
Detoured invoices

Checking document modifications

- Insert text boxes (add new bank account details, delivery addresses, ...)
- Adding overlays in the picture → hide some parts
- Add some signature scans
- ...

Detoured invoices

Checking document modifications



Detoured invoices

Checking document modifications

Checking for added text boxes

```
pdf-parser.py -s /fontfile mod1.pdf
```

```
    obj 56 0
Type: /FontDescriptor
Referencing: 54 0 R
<<
  /Type /FontDescriptor
  /FontName /CAAAAA+LiberationSerif-Bold
  /Flags 4
  /FontFile2 54 0 R
>>
```

Detoured invoices

Checking document modifications

- Which font descriptor corresponds to what?
- Dump the font file
- Display the glyphs
- Check the coordinates
- or ...
- Deactivate it and visualize

Detoured invoices

Checking document modifications

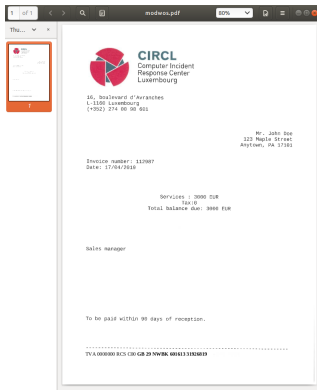
```
cat mod1.pdf | sed 's/58_0_obj/99_0_obj/g' > out.pdf
```

To be paid within 90 days of reception.

TVA 0000000 RCS C00

Detoured invoices

Adding signature scans



Adding signature scans



Mr. John Doe
123 Maple Street
Anytown, PA 17280

Services : 3000 EUR
Tax:0
Total balance due: 3000 EUR

Alber

TVA 0000000 RCS C00 R0000000 0000000000 0000000000

Detoured invoices

Adding signature scans

Search for included images

```
pdf-parser.py -s /image invoice2.pdf
```

```
obj 5 0
```

```
  Type: /XObject
```

```
  Referencing: 7 0 R
```

```
  Contains stream
```

```
    <<
```

```
      /Type /XObject
```

```
      /Subtype /Image
```

```
      /Width 433
```

```
      /Height 180
```

Detoured invoices

Adding signature scans

Extract the image from the pdf document

```
pdf-parser.py -o 5 invoice2.pdf -d signature.png
```

Check the image

```
display signature.png
```

What can be shared?

- File meta information
 - Did other recipients received it?
 - Is it in a backups?
 - Was it in mailboxes?
 - Is it in shadow copies
 - ...
- Timestamps → get a time range of operations
- Bank account details
 - Prevent other transfers
 - Correlate cases