

PDF



Myths vs FACTS

Oxford University 15th July 2015 Ange Albertini

ANGE ALBERTINI

reverse engineering

VISUAL DOCUMENTATION

@angealbertini

ange@corkami.com

<http://www.corkami.com>



Disclaimer:

this is my first digipres event

I come here with a very different perspective:

I might sound pessimistic (or provocative/killjoy)...

Give me hope, give me peace on earth ;)

I might be entirely **wrong** - *p/*lease let me know!

**I used to think:
“PDF is perfect”**

Complex documents,
yet uniform rendering on any system
(no wonder it's omnipresent)
⇒ I believed the myth...

Professionally, I analyse PDFs

Malware, security

(It originally happened by “accident”,
but I’ve been doing it since then...)

PORTABLE DOCUMENT FORMAT

HEADER

```
%PDF-1.1          SIGNATURE & VERSION INFORMATION
<< [D VALUE] >>
<<
  /Pages 2 0 R
  >>
endobj
```

IDENTIFIER (WITH /)

```
2 0 obj
<<
  /Type /Pages
  /Count 1
  /Kids [3 0 R]
  >>
endobj
```

ARRAY

```
3 0 obj
<<
  /Type /Page
  /Contents 4 0 R
  /Parent 2 0 R
  /Resources <<
    /Font <<
      /F1 <<
        /Type /Font
        /Subtype /Type1
        /BaseFont /Arial
      >>
    >>
  >>
endobj
```

BODY

```
4 0 obj
<< /Length 50 >>
stream
BT
  /F1 110 Tf
  10 400 Td
  (Hello World!)Tj
ET
endstream
endobj
```

STREAM PARAMETERS:
LENGTH, COMPRESSION...

BEGIN TEXT
FONT F1 (ARIAL) SET TO SIZE 110
MOVE TO COORDINATE 10, 400
OUTPUT TEXT 'HELLO WORLD!'
END TEXT

XREF TABLE

```
CROSS REFERENCE
xref
0 5
0000000000 65535 f
0000000010 00000 n
0000000047 00000 n
0000000111 00000 n
0000000313 00000 n
```

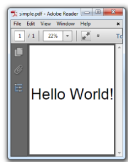
CROSS REFERENCES
5 OBJECTS, STARTING AT INDEX 0
(STANDARD FIRST EMPTY OBJECT 0
OFFSET TO OBJECT 1, REV 0
TO OBJECT 2..

TRAILER

```
trailer
<<
  /Root 1 0 R
  >>
startxref
413
%%EOF
```

PARSING

%PDF-1. ? IS CHECKED
startxref POINTS TO XREF
xref POINTS TO EACH OBJECT
trailer IS PARSED
REFERENCES ARE FOLLOWED
DOCUMENT IS RENDERED



TRAILER

ROOT

1

PAGES

2

PARENT

KIDS

3

CONTENTS

4

PDF operators

GRAPHICS STATE

general

<i>lineWidth</i>	w set line width
<i>lineCap</i>	J set line cap
<i>lineJoin</i>	j set line join
<i>miterLimit</i>	M set miter limit
<i>dashArray</i> <i>dashPhase</i>	d set line dash
<i>intent</i>	ri set color rendering intent
<i>flatness</i>	i set flatness tolerance
<i>dictName</i>	gs set graphics state

special

q	save current state
Q	restore current state
<i>a b c d e f</i>	cm modify current transformation matrix

TEXT

object

	BT begin text
	ET end text

position

<i>x y</i>	Td move to next line
<i>x y</i>	TD move to next line + set leading
<i>a b c d e f</i>	Tm text & text line matrixes
	T* move to next line + reset leading

showing

<i>string</i>	TJ show string
<i>string</i>	' move to next line + show string
<i>a_w a_c string</i>	" move to next line + show string + set word & character spacing
<i>array</i>	TJ show string(s) with glyph positioning

PATH

construction

<i>x y</i>	m move points (begin subpath)
<i>x y</i>	l line
<i>x₁ y₁ x₂ y₂ x₃ y₃</i>	c cubic bezier
<i>x₂ y₂ x₃ y₃</i>	v cubic bezier
<i>x₁ y₁ x₃ y₃</i>	y cubic bezier
	h close subpath
<i>x y width height</i>	re rectangle

painting

S	stroke the path
s	close and stroke (h S)
f	fill
F	fill (deprecated)
f*	even-odd fill
B	fill & stroke
B*	even-odd fill & stroke
b	close, fill & stroke
b*	close, even-odd fill & stroke
n	end path (no filling/stroking)

clipping

W	intersect clipping path
W*	even-odd intersect clipping path



even-odd fill

I created fact sheets about PDF

I gave
presentations
about PDF

Advanced

PDF TRICKS

**An overview of
potential leaks via PDF**

**PDF
secrets**

hiding & revealing secrets in PDF documents

Personally, I *play* with PDF

proactive, and *fun*

PDF is more than a file format.
PDF is an abuse playground!

The logo for 'PDF disaster' features a red square icon composed of four smaller squares arranged in a 2x2 grid to the left of the text. The text 'PDF' is in a large, bold, black sans-serif font, and 'disaster' is in a smaller, bold, black sans-serif font below it.

PDF
disaster

Yes, I write PDFs by hand...

[...and I open them in hex editors]

%PDF-1.

```
1 0 obj
<< /Kids [<<
    /Parent 1 0 R
    /Resources <<>>
    /Contents 2 0 R
  >>]
>>
```

```
2 0 obj
<<>>
stream
BT
  /F1 110 Tf
  10 400 Td
  (Hello World!) Tj
ET
endstream
endobj
```

```
trailer <<
  /Root << /Pages 1 0 R >>
>>
```

...like this one

%PDF-1.

truncated signature

1 0 obj

<< /Kids [<<

 /Parent 1 0 R

 /Resources <<>>

 /Contents 2 0 R

>>]

>>

2 0 obj

<<>>

stream

BT

 /F1 110 Tf

 10 400 Td

 (Hello World!) Tj

ET

endstream

endobj

trailer <<

 /Root << /Pages 1 0 R >>

>>

missing parent /Type
/Kids should be indirect
missing /Font
missing kid /Type
missing /Count

missing endobj

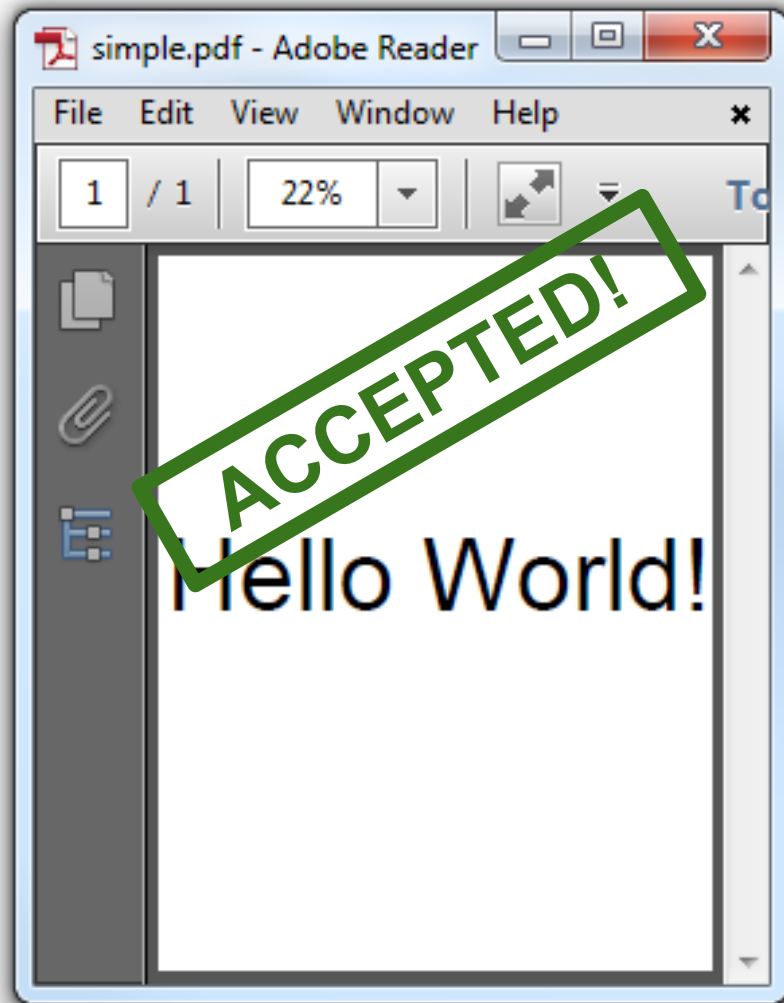
missing /Length

missing xref

/Root should be indirect, missing /Size, missing root /Type
missing startxref, %%EOF

It's *not*
standard...

...but it works
exactly as planned!
(without any reported error)



Binary art

PDF + creativity = ... ?



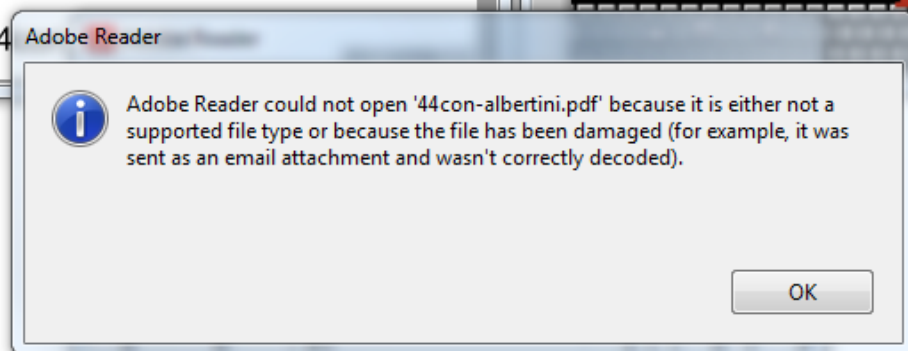
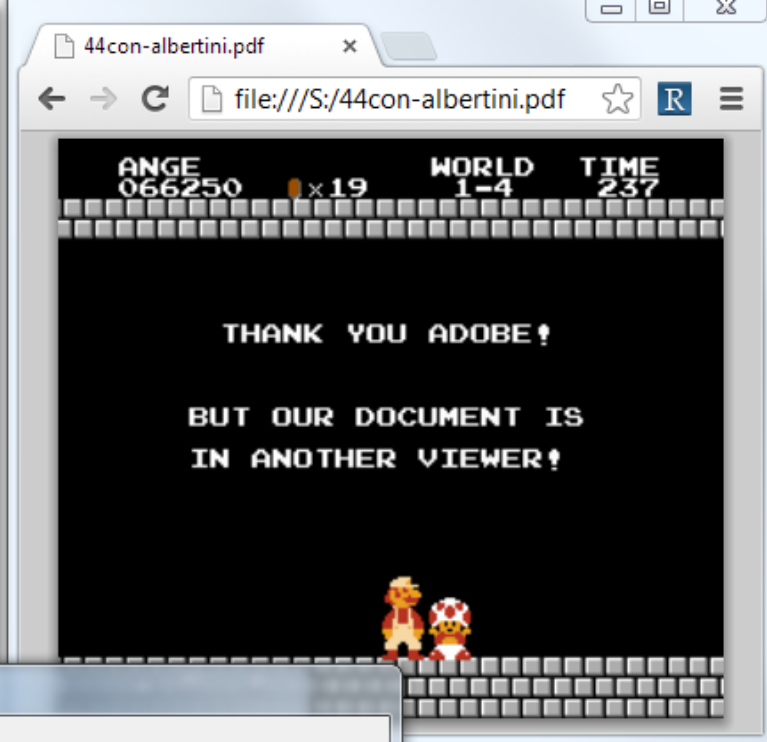
PDF slides



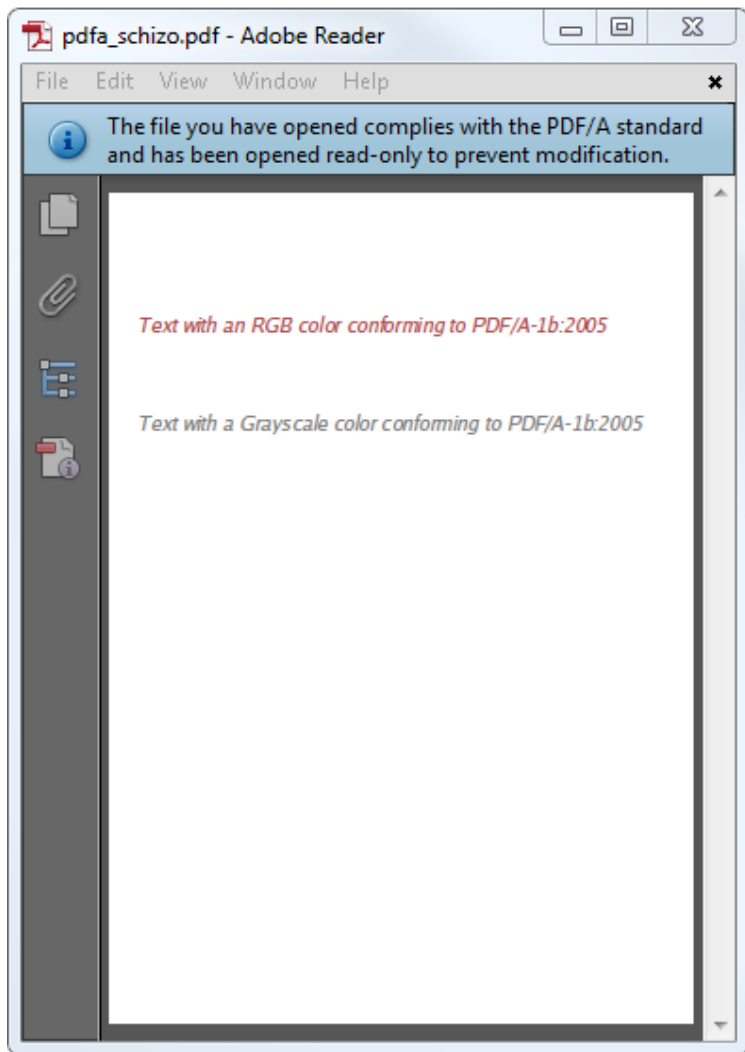
PDF viewer



the slides for my talk at 44Con
are distributed as a file that is
simultaneously
a PDF *and* a PE (a PDF viewer)
so that the slides can view themselves
(oh, and it's also HTML + Java)...



...and it's also schizophrenic (PDF documents appear different with different readers)



(Also available in PDF/A flavour)

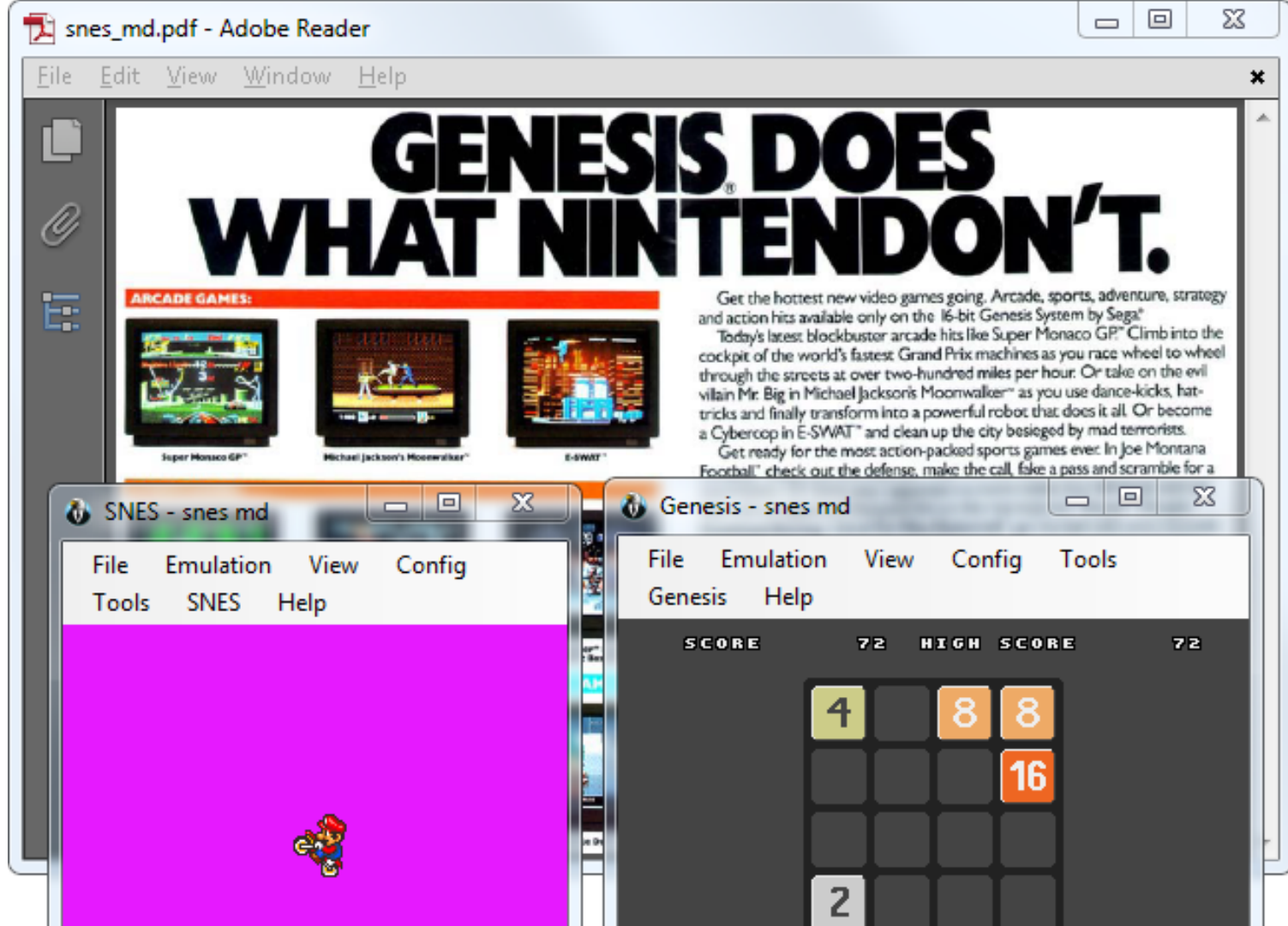
Keyboard

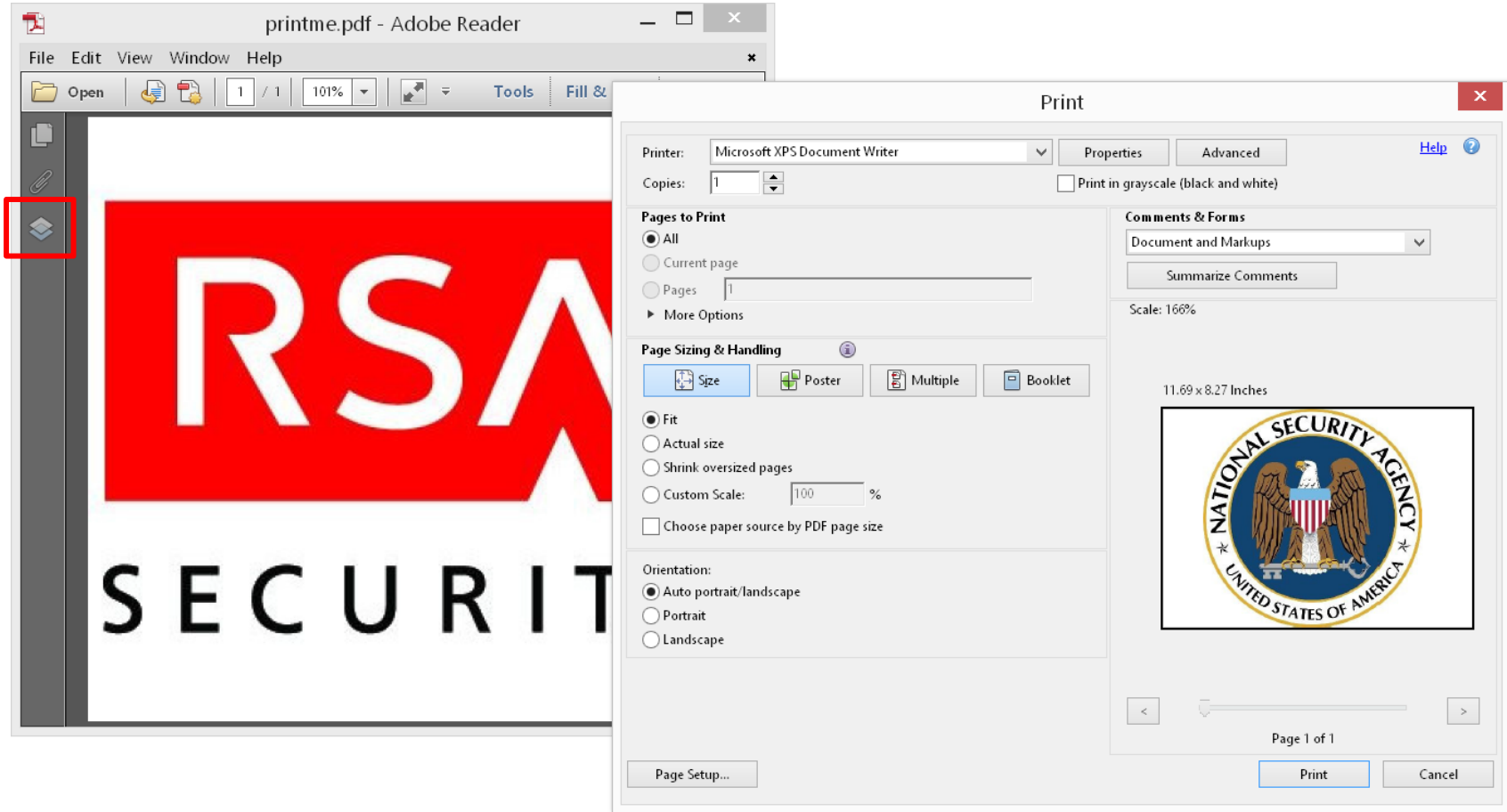
Wave View

NES Music

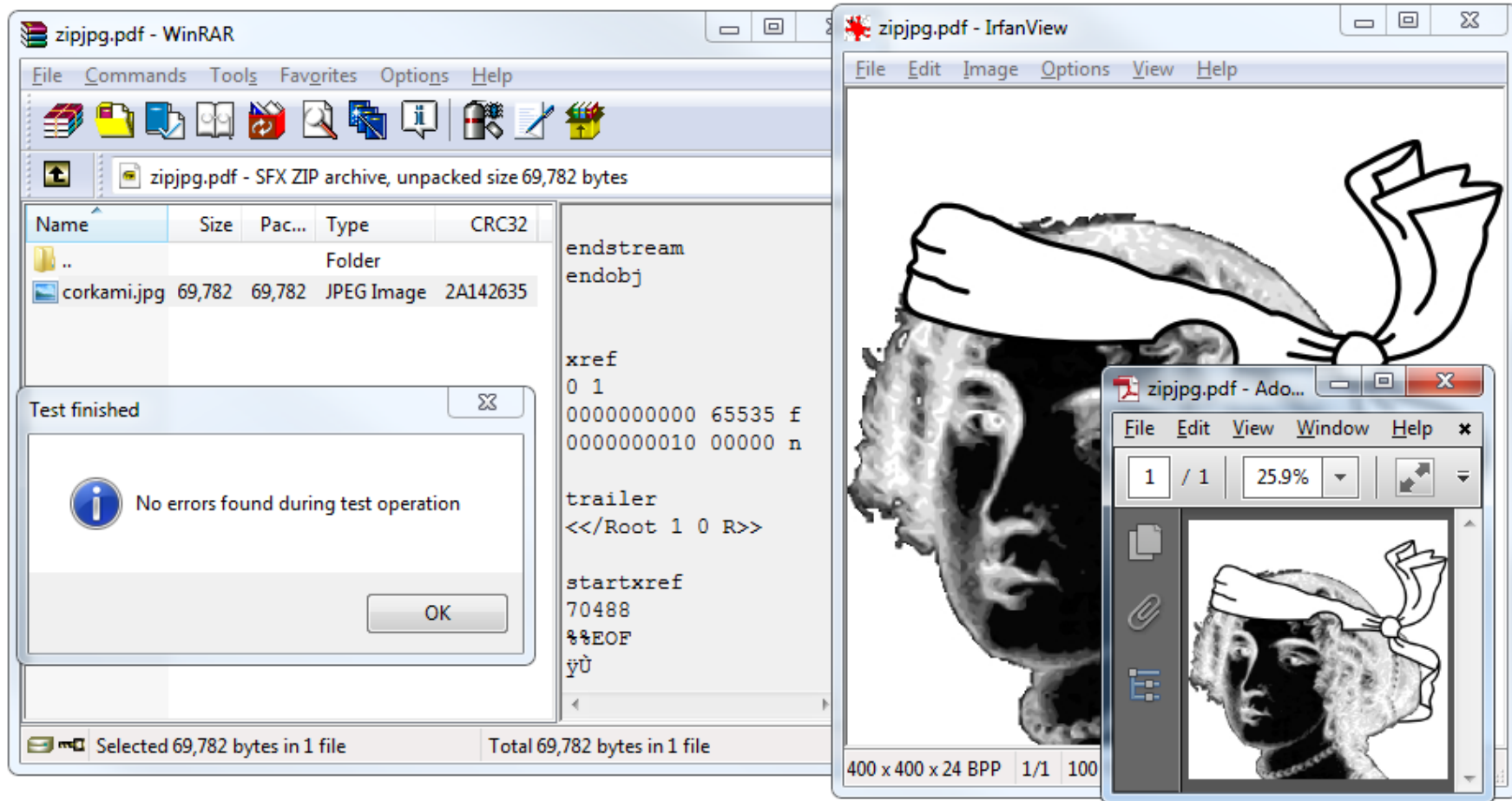
[illegible]

Super NES
Megadrive

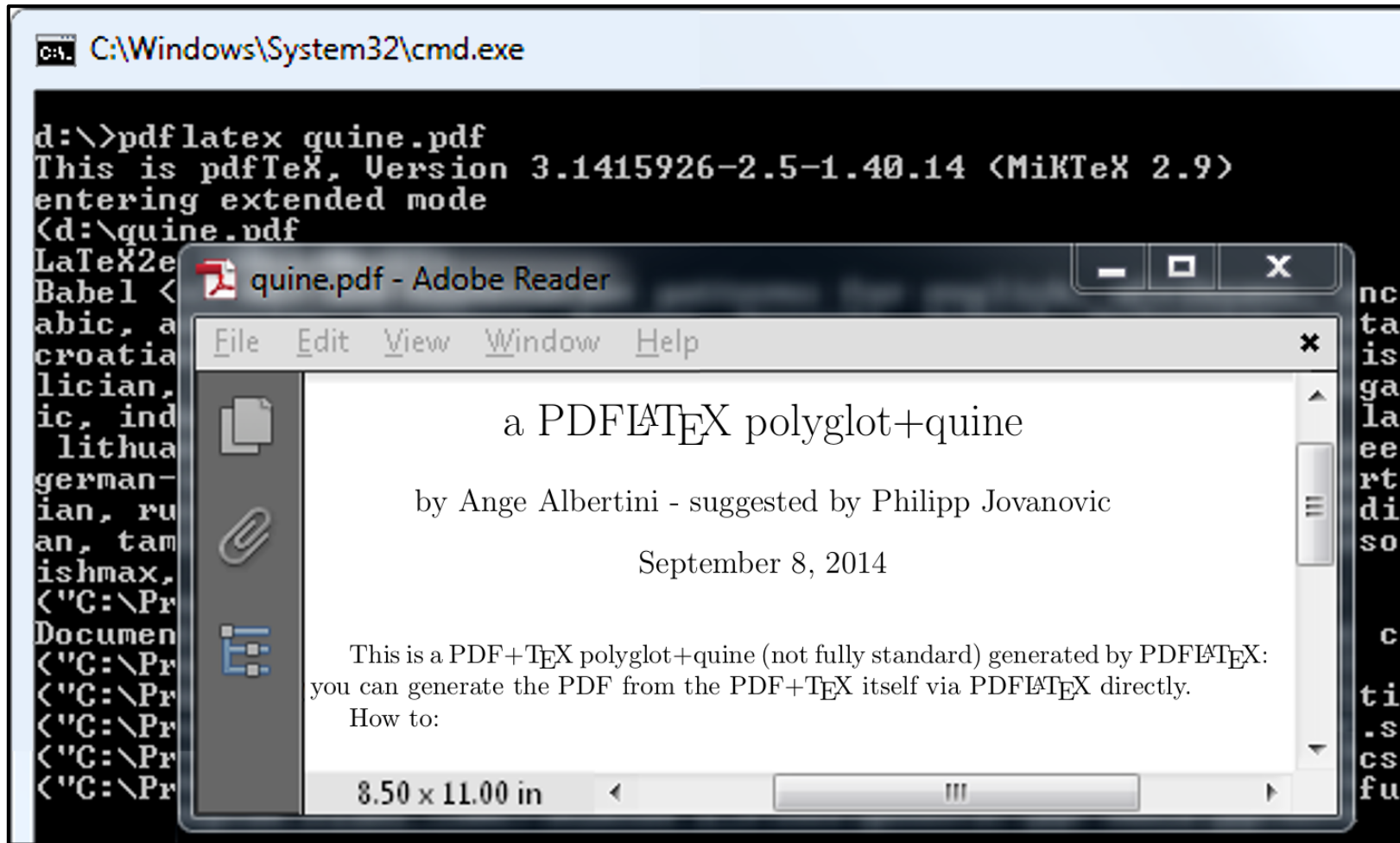




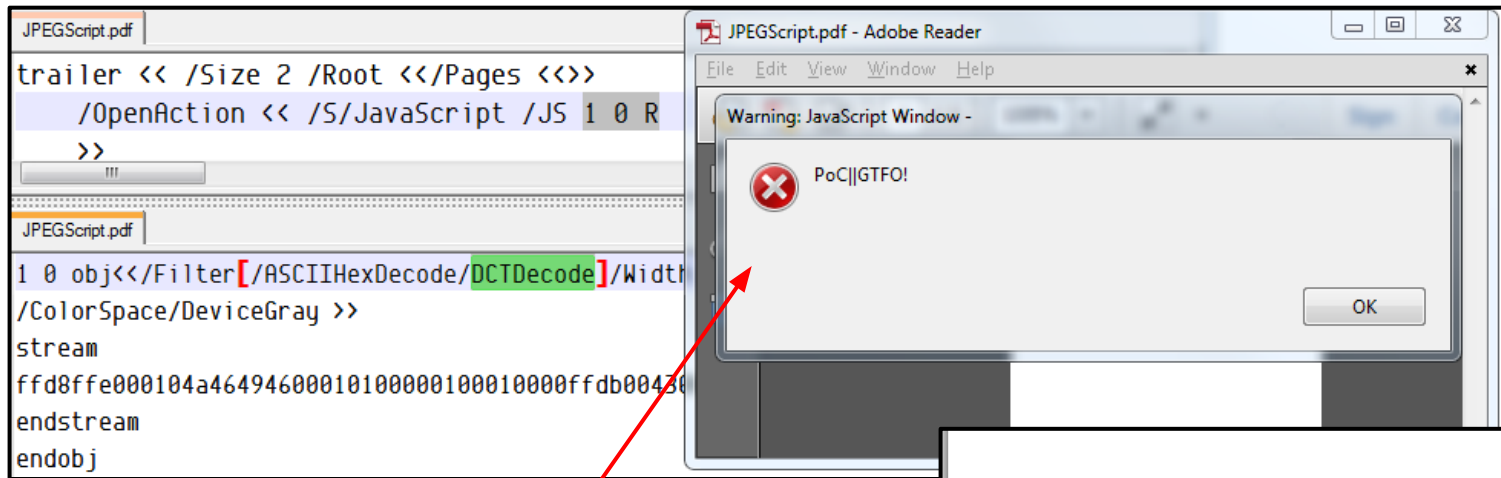
What you see is not always what you print - when you use Layers [**O**ptional **C**ontent **G**roups]!
Fun fact: you **can't** change the printing output with Adobe Reader ;)



JPEG + ZIP + PDF Chimera (3 headers but only 1 image data)



PDFLaTeX quine (the document is its own source)



script == picture

JPEGScript

`app.alert("PoC||GTFO!");`

JPEG-encoded JavaScript
(deprecated)

PoC||GTFO

International Journal of Proof-of-Concept or Get The F*** Out

the “new” 2600 / Phrack...

Distributed as **PDF** ⇒ each issue is a PoC

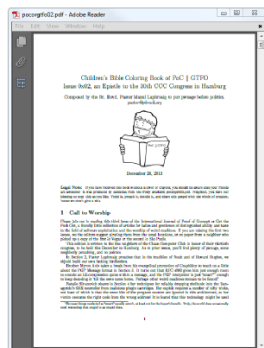
```
QEMU
[0000:7C000]
Berliner Spargel Operating System
Mein Deutsch is nicht so gut, aber es ist Spargel zeit!
by Travis Goodspeed

m -- Memory Viewer
a -- About

This is a minimal operating system by Travis Goodspeed for 16-bit Real
Mode 8086 on an IBM PC. It was written in order to learn about the
8086, and it quite likely will serve no use for you. It is free
without any strings attached, but please give credit were credit is
due if you fork it.

Also, and this is very important, you should use the included hex viewer
to poke around this machine's memory. The boot sector at 0000:7C000
is likely a good place to start.
Press the 'any' key to continue._
```

MBR (bootable)
+ PDF + ZIP



Archive: pocorgtfo02.pdf
warning [pocorgtfo02.pdf]: 8016414 extra bytes at beginning or within zipfile
(attempting to process anyway)

Length	EAs	ACLs	Date	Time	Name
852	0	0	12/06/13	16:25	README.txt
6794	0	0	12/06/13	16:25	coda.txt
20164	0	0	12/06/13	16:25	feeling.txt
12618	0	0	12/06/13	16:25	harrison.txt
0	0	0	12/06/13	16:25	pqpquine/
275	0	0	12/06/13	16:25	pqpquine/Makefile
1006	0	0	12/06/13	16:25	pqpquine/inflate.c
5323	0	0	12/06/13	16:25	pqpquine/quine.c
203706	0	0	12/06/13	16:25	rffc4880.txt
2046109	0	0	12/06/13	16:25	tamagotch1.zip
15565	0	0	12/06/13	16:25	thewub.txt
278598	0	0	08/05/13	13:06	pocorgtfo00.pdf
3790438	0	0	10/13/13	02:47	pocorgtfo01.pdf
6381448	0	0			13 files

FILE

```

0000: ff d8
0002: ff e0 <size.16> <content>
0014: ff fe <size.16>
      : %PDF-1.5
      999 0 obj
      <<>
      stream

0039: ...
xx : ff d9
xx+2 : endstream
      endobj
xx+14: %PDF-1.5 ...

```

JPEG

"START OF IMAGE" MARKER

"APP0" MARKER (REQUIRED | HEADER)

"COMMENT" MARKER

COMMENT CONTENT

(OTHER MARKERS, ORIGINAL JPEG DATA)...

"END OF IMAGE" MARKER

PDF

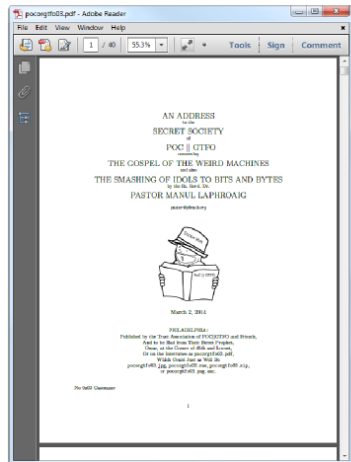
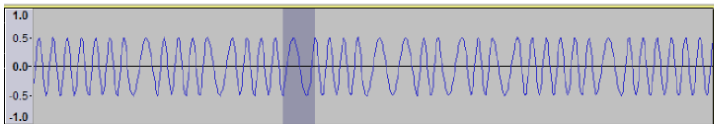
PDF SIGNATURE

STARTING A DUMMY BINARY OBJECT

CLOSING THE DUMMY OBJECT

ORIGINAL PDF CONTENTS (MULTIPLE SIGNATURES ARE IGNORED)

*REPLACED WITH 00 00 TO BYPASS ADOBE FILTER



AngeCrypto: getting valid files after encryption

1 CONTROLLING FIRST ENCRYPTED BLOCK

PLAINTEXT BLOCKS: P1, P2

CIPHER BLOCKS: C1, C2

ENC (P1) = C1

DEC(C1) = P1

IV = DEC(C1) * P1

ENC(C2) = C2

DEC(C2) = P2

EXAMPLE WITH AES:

KEY: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

IV: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

ENC(P1) = 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

DEC(C1) = 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

IV = DEC(C1) * P1 = 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

3 SKIPPING UNCONTROLLED BLOCKS

(1) PNG SIGNATURE

STARTING A DUMMY CHUNK

RANDOM ENCRYPTED DATA

ENDING DUMMY CHUNK

(2) STARTING CONTROLLED DATA

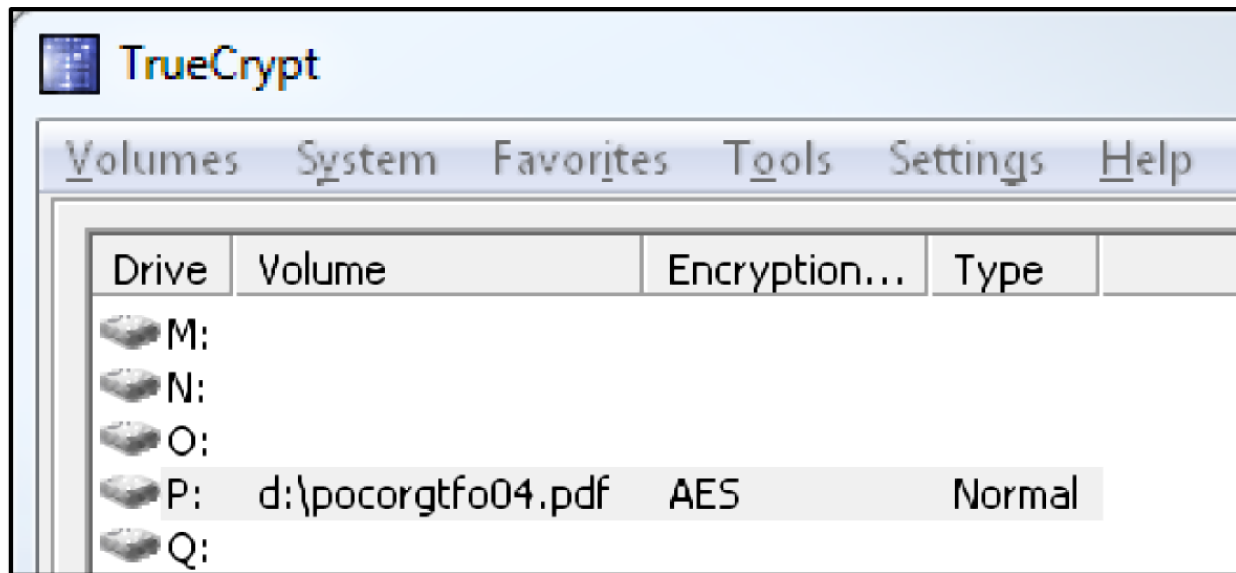
END OF IMAGE

ANGE ALBERTIN
WITH THE HELP OF JEAN-PHILIPPE AUMASSON

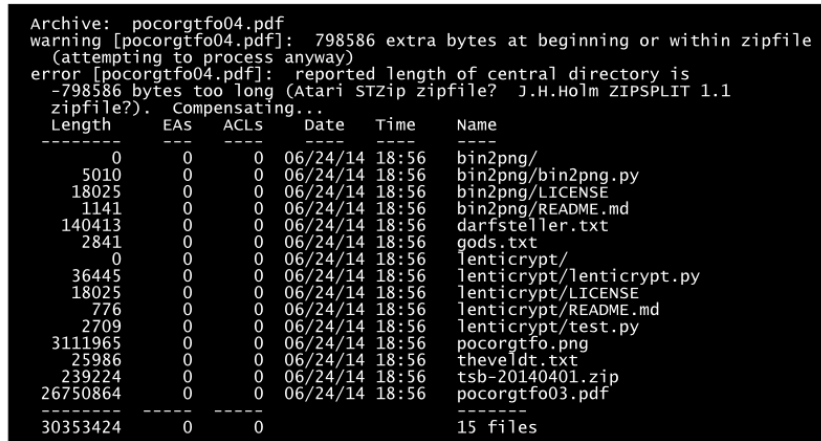
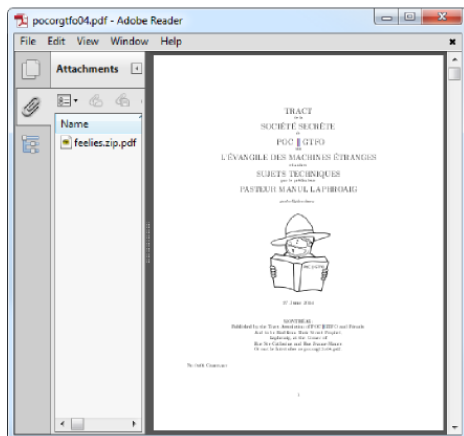
raw audio +
JPG + AES(PNG)
+ PDF + ZIP

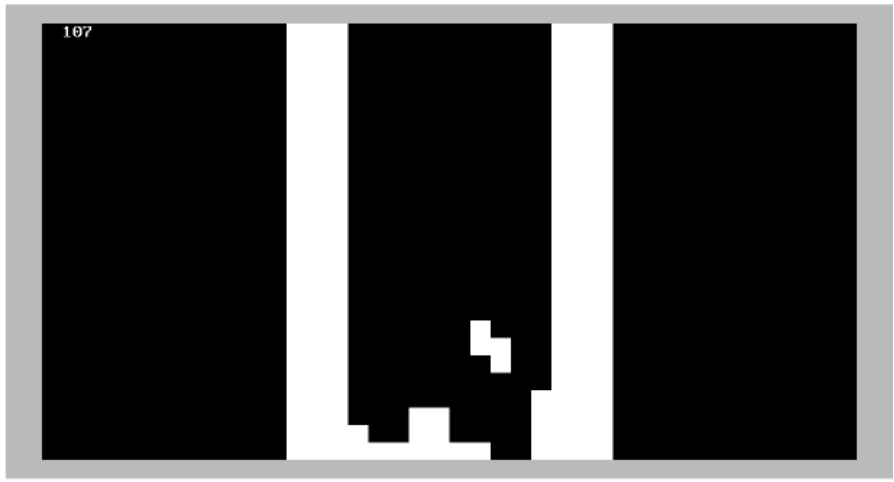
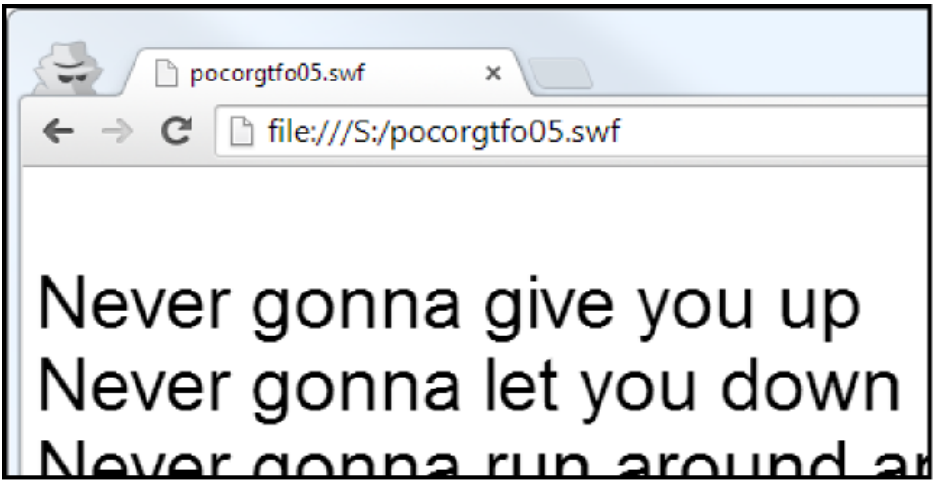
Archive: pocorgtfo03.pdf
warning [pocorgtfo03.pdf]: 12224072 extra bytes at beginning or within zipfile (attempting to process anyway)

Length	EAS	ACLs	Date	Time	Name
2561	0	0	02/10/14	06:23	alexander.txt
7848	0	0	02/08/14	20:20	bochs-2.6.2.patch
6135	0	0	02/08/14	20:21	bochs-20140203.patch
7248	0	0	02/09/14	08:35	defusing.zip
4830	0	0	12/01/13	15:48	despair.txt
14892	0	0	11/27/13	19:03	lasta.txt
26325	0	0	02/07/14	21:06	lastq.txt
473449	0	0	02/07/14	21:06	netwatch-337f8b1.tar.gz
131930	0	0	02/24/14	20:32	nokiacypher.png
14645	0	0	02/17/14	18:52	packed
2129	0	0	02/07/14	21:06	saucers.txt
3144	0	0	02/07/14	21:06	tamadec.txt
6227	0	0	02/07/14	21:06	tetranglix.tar.bz2
14109425	0	0	02/07/14	21:06	pocorgtfo02.pdf
322	0	0	03/03/14	01:28	pocorgtfo03-encrypt.py
14811110	0	0			15 files

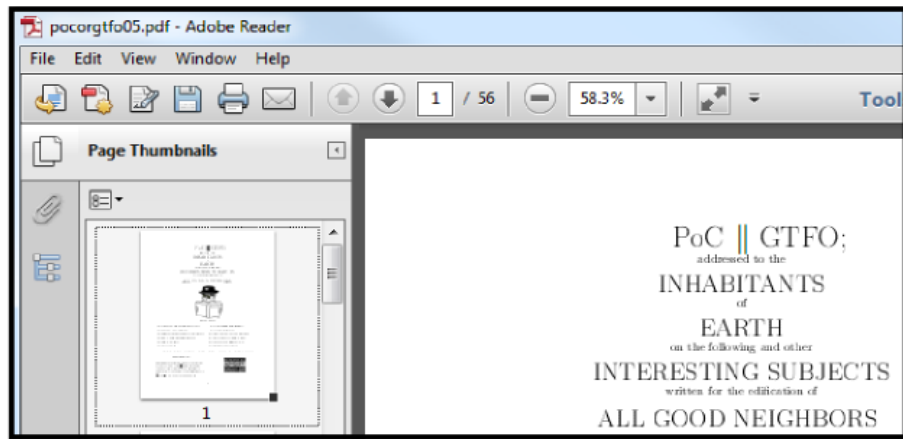


TrueCrypt
+ PDF + ZIP





Flash + bootable ISO + PDF + ZIP



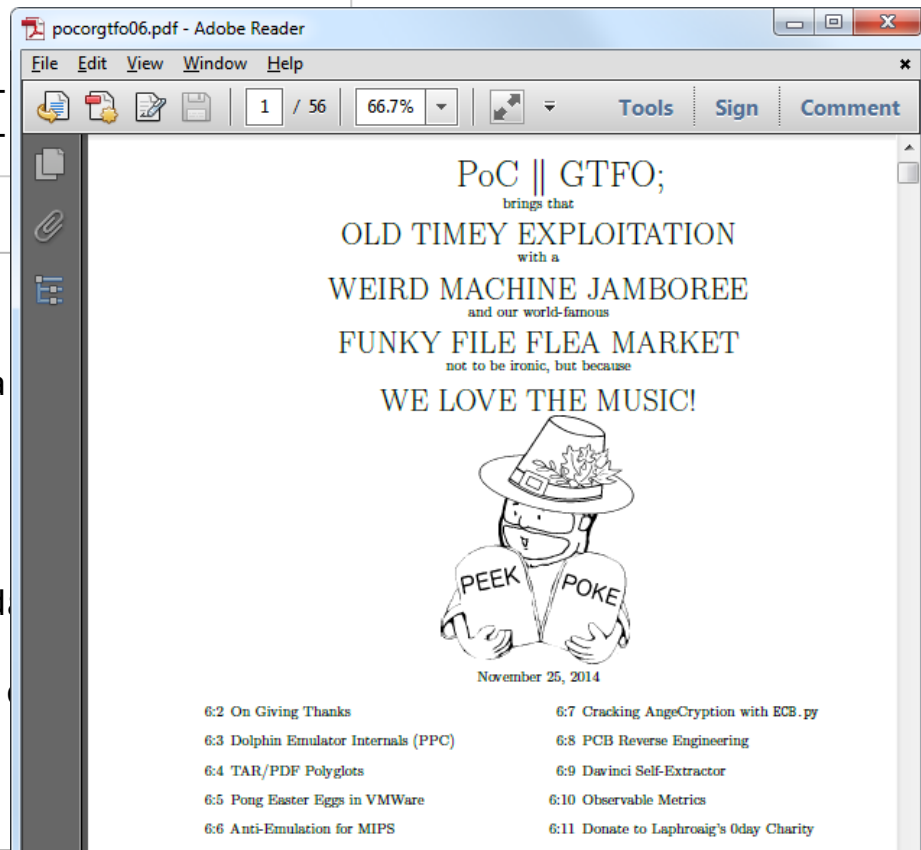
```
Archive: pocorgtfo05.pdf
warning [pocorgtfo05.pdf]:
  (attempting to process an
    creating: PEXternalizer/
    creating: PEXternalizer/
    inflating: PEXternalizer/
    inflating: PEXternalizer/
    inflating: PEXternalizer/
    inflating: PEXternalizer/
    inflating: PEXternalizer/
```

TAR + PDF + ZIP

```
$ tar -tvf pocorgtfo06.pdf
-rw-r--r-- Manul/Laphroaig    0 2014-10-06
-rw-r--r-- Manul/Laphroaig 525849 2014-10-
-rw-r--r-- Manul/Laphroaig 273658 2014-10-
```

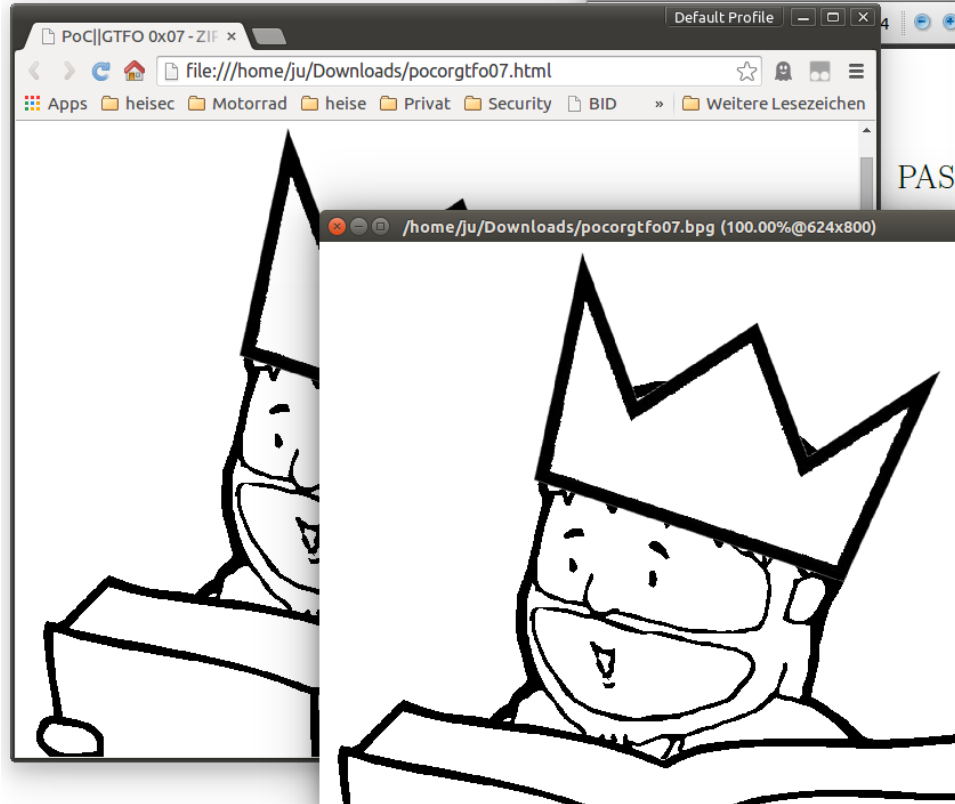
```
$ unzip -l pocorgtfo06.pdf
Archive:  pocorgtfo06.pdf
warning [pocorgtfo06.pdf]:  10672929 extra
(attempting to process anyway)
```

Length	Date	Time	Name
-----	-----	-----	----
4095	11/24/2014	23:44	64k.txt
818941	08/18/2014	23:28	acsac13_zadd
4564	10/05/2014	00:06	burn.txt
342232	11/24/2014	23:44	davinci.tgz.
3785	11/24/2014	23:44	davinci.txt
5111	09/28/2014	21:05	declare.txt
0	08/23/2014	19:21	ecb2/

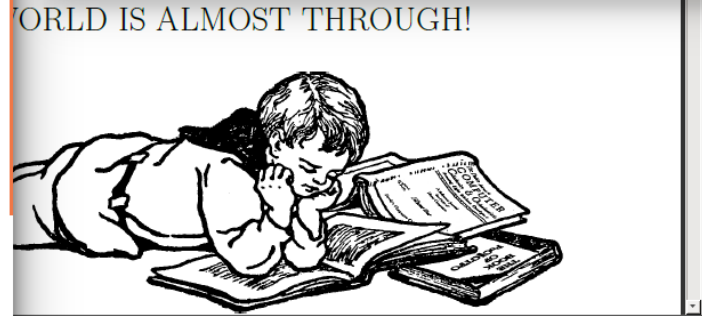


BPG + HTML (incl. a BPG viewer in JS)

+ PDF + ZIP



```
ju@loki:
Datei Bearbeiten Ansicht Suchen Terminal Hilfe
Archive: pocorgtfo07.zip
***** PWNED *****
dumping credentials...
*****
Length      Date      Time      Name
-----
6325      2015-02-02  20:56      500miles.txt
0         2015-03-19  15:51      abusing_file_formats/
370375    2015-03-06  21:51      abusing_file_formats/3in1.png
512       2015-03-06  21:51      abusing_file_formats/abstract.tar
143360    2015-03-06  21:51      abusing_file_formats/AppleII.pdf
116       2015-03-06  21:51      abusing_file_formats/asciizip
420852    2015-03-06  23:27      abusing_file_formats/cameron.png
41902     2015-03-19  15:51      abusing_file_formats/chimera.png.zip
122880    2015-03-07  19:16      abusing_file_formats/chimera.tar.pdf
590538    2015-03-06  21:51      abusing_file_formats/Corkami_spectrogram.bmp
537654    2015-03-06  21:51      abusing_file_formats/dr_nsa.bmp
10213     2015-03-06  21:51      abusing_file_formats/I love reliable forensics sof
ware.png
```

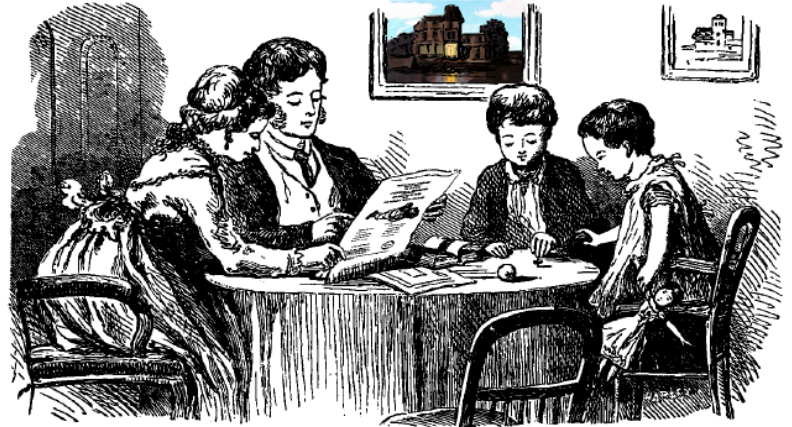


Shell script + PDF + ZIP

```
$ echo "terrible raccoons achieve their escapades" | ./pocorgtfo08.pdf -d 4321  
good neighbors secure their communications
```

```
$ unzip -l pocorgtfo08.pdf  
Archive: pocorgtfo08.pdf
```

Length	EAS	ACLs	Date	Time	Na
-----	---	----	----	----	-
988446	0	0	08/06/15	22:46	EC
440648	0	0	09/06/15	20:36	ai
522633	0	0	09/06/15	19:18	ai
1546	0	0	08/06/15	22:46	al
118696	0	0	08/06/15	22:46	br
31337	0	0	08/06/15	22:46	ex
38109	0	0	08/06/15	22:46	ge
303926	0	0	08/06/15	22:46	if
160225	0	0	08/06/15	22:46	jt
3149	0	0	08/06/15	22:46	le
2244652	0	0	08/06/15	22:46	ma
4662	0	0	08/06/15	22:46	ni



AS EXPLOITS SIT LONELY,
FORGOTTEN ON THE SHELF
YOUR FRIENDLY NEIGHBORS AT
PoC || GTFO
PROUDLY PRESENT

... and others

Bootable quine in assembly,
2 switchable PDFs via ROT13,
hash collisions,
GameBoy + Sega Master System...

You get the idea...

The worst case for preservation?

I explore corner cases, before attackers do it

How is it possible?

- signature offset not enforced
- stream object (containing anything)
- comments can contain binary data
- appended data
- objects tolerated between XREF and startxref

and a few specific abuses (some are fixed now)

What is PDF ?

I asked online...

Postscript Derived Failure

Potential Disaster Forever

Practically Destructive File

Pretty Demented Format

Preservation Dooming Format

POC||GTFO Demonstration Format

Paper Dimensions Fixed

Payload Deployment File

Posterity Depends on Forensics

Preserving Document Forever

Pathetic & Dangerous Format

Posthoc Depression Format

Perversely Designed Format

Penile Dysfunction Format

Proprietary Document Fee

Public Domain Farce

Postscript Didn't Fit

Penetrate Dodgy Firewall

Proven Dysfunctional Format

Please Don't Fail / Again

Polyglot (Definition|Deployment|Delivery) Framework

Pants-Down Format

PDF is a Disaster for the Future

Perpetually Disagreeable Format

...and I wasn't disappointed :)

More seriously...

(from my *personal* point of view)

A miracle?

Fonts are embedded in the document

Rendering is following complex rules
(overly-complex, from a security standpoint)



An open format?

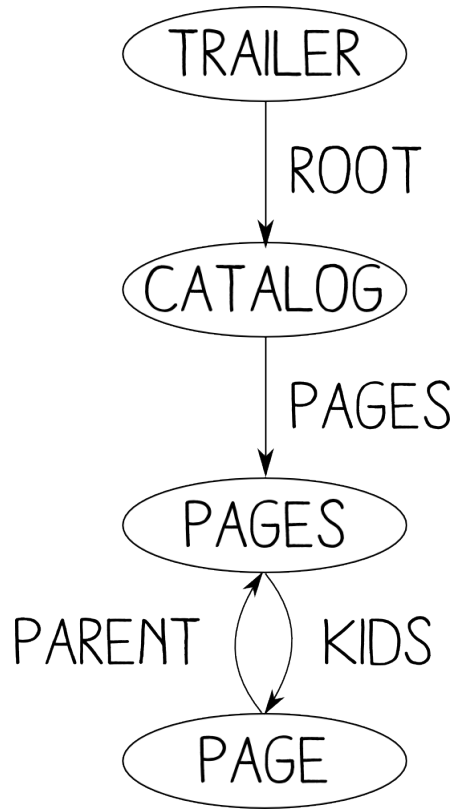
ISO \$pec\$ = 200\$

These specs only cover the main part :(

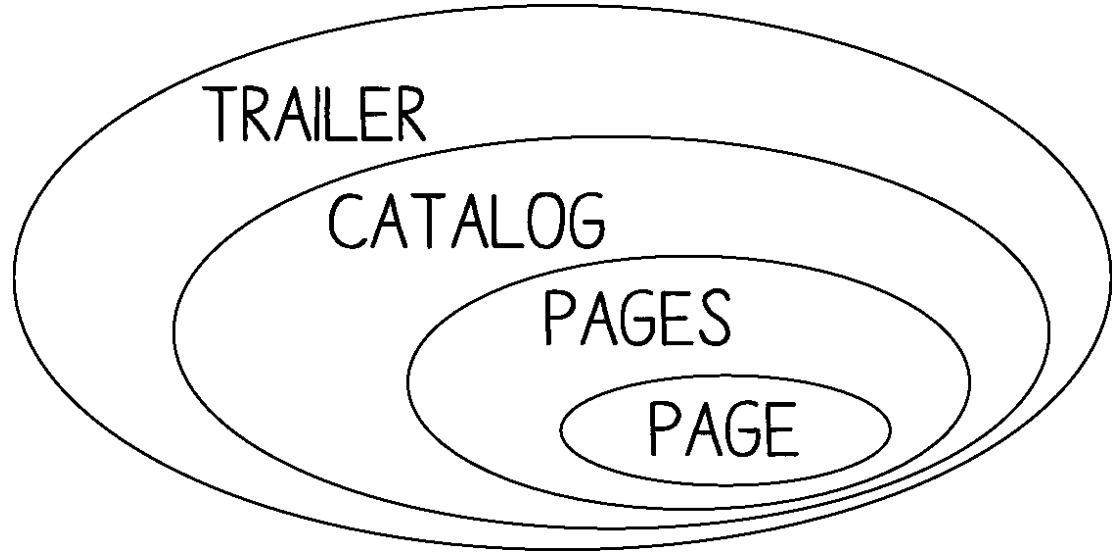
They are unclear - no formal guarantee :(

A strict format ?

No reader completely enforces the specs
⇒ recovery mode (sometimes 'explicit')
signature, stream length, XREF...



standard structure
(each object should be distinct)



non-standard but tolerated structure
(inlined objects)

Many possible malformations handled specifically by each reader (high level)...

Many possible abuses

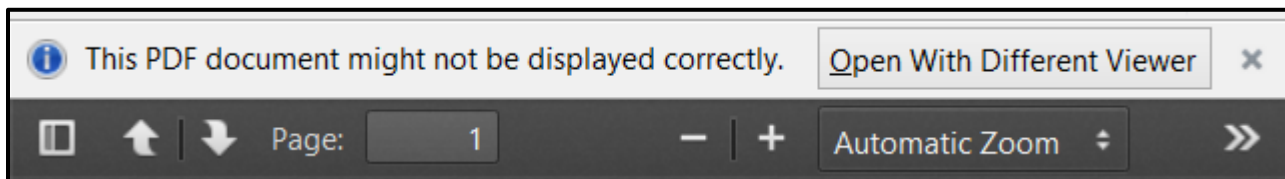
follows the specs

corruption tolerated

absence tolerated

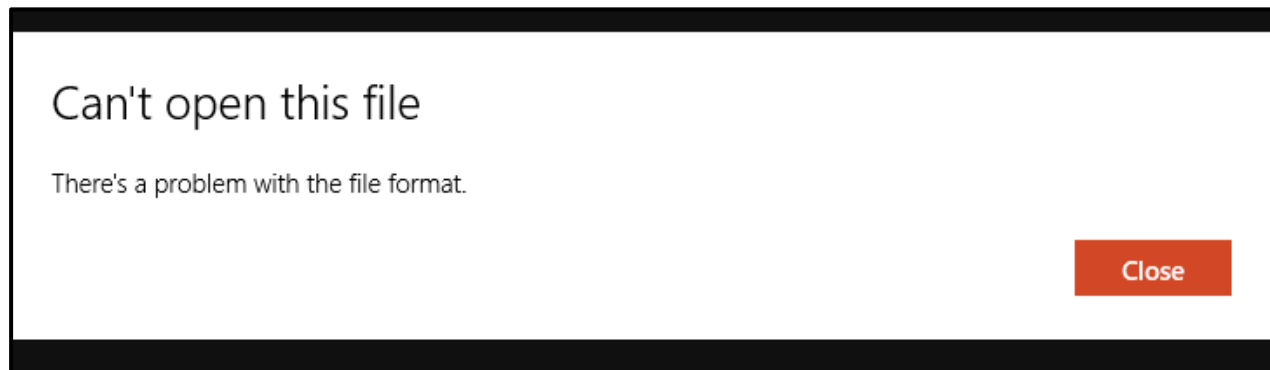
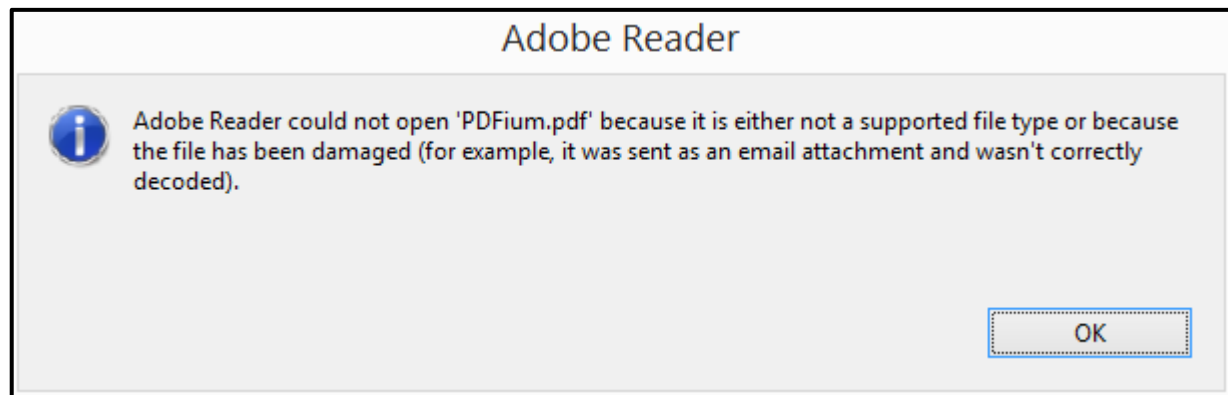
	signature	endobj	/Count	text operators	/Font	font use	xref	/Resources	trailer
Adobe Reader	corruption tolerated	corruption tolerated	absence tolerated	follows the specs	absence tolerated	follows the specs	absence tolerated	follows the specs	follows the specs
MuPDF	absence tolerated	absence tolerated	follows the specs	corruption tolerated	follows the specs	follows the specs	absence tolerated	follows the specs	absence tolerated
PDF.js	absence tolerated	follows the specs	follows the specs	absence tolerated	follows the specs	follows the specs	corruption tolerated	follows the specs	corruption tolerated
PDFium	corruption tolerated	absence tolerated	absence tolerated	absence tolerated	absence tolerated	absence tolerated	absence tolerated	absence tolerated	follows the specs
Poppler	absence tolerated	absence tolerated	follows the specs	absence tolerated	follows the specs	follows the specs	absence tolerated	follows the specs	follows the specs

... different readers have different tolerances ...



Failed to load PDF document

Error loading D:\FireFox.pdf



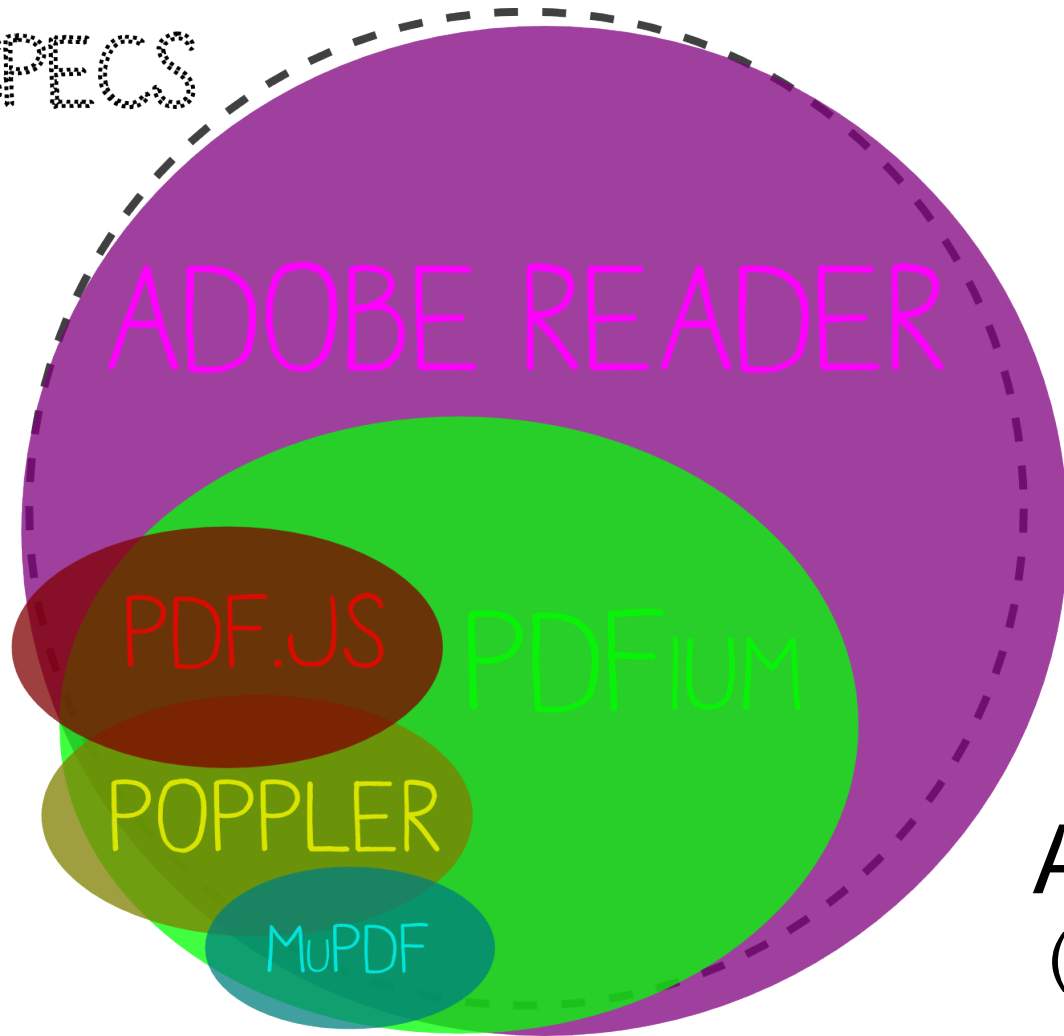
...so a PDF specifically crafted for one reader, may fail with all other readers.

A uniform format?

Many free readers, but...

- Many (useful) features only available in Adobe Reader:
forms, signature, layers...
(it's Adobe's business model)
- Other readers just aim to support “standard” PDFs

SPECS



A beautiful mess!
(an artist's interpretation)

A consistent format?

Adobe Reader is closing security issues.

This is good, but...

⇒ Some features are not supported anymore

⇒ Potential lack of backward compatibility

It's a complex patchwork!

JPGs are stored *entirely* as-is, but PNG have to be *converted* to raw

Forms as XML

PostScript Transfer function

Web (Flash, JavaScript...)

3D objects

A coherent format?

- text + line comments, yet binary
- unusual whitespace, binary also in comments
- different escaping
- read forward+no separator and object reference
- hex as nibbles and odd-numbered
- bottom up but also possibly top down (who wins?)
- corrupted ZLIB still tolerated
- image compression for non-images

What if...

...Adobe would stop supporting PDF ?
We're just left with the 'specs' ?

After all...

...Flash is being killed for security reasons,
after becoming progressively redundant.

PDF could be converted to something else.

PDF & preservation

- JPG + OCR'ed text = simple
...so simple that we wouldn't need PDF ?
other PDFs = complex (Adobe-dependent)

Is PDF/A the solution? more \$pec\$

“Backward compatibility”

...is a beautiful utopia!

And it leads to saying
“we've always done it this way”
even after several generations :(

“Backward compatibility”

...can be incompatible with security fixes

JPEG-encoded JavaScript
PDF polyglots

Brace yourself...

PDF 2.0 is coming!

It's not improving stability and preservability
Will Adobe adhere to it ? Since it's distinct now...

*<https://www.youtube.com/watch?v=wGmcTf-uMrE>

Conclusion

“a *complex* puzzle because the original picture is messy”



Adobe

WWW.ADOBE.COM/SECURITY

Conclusion

- PDF is very useful - omnipresent for a reason
- it's still involved in computer security
 - recent complete takeover of Windows 8.1 by @j00ru
- it's quite a monster
 - I'm merely scratching the surface
 - its specs were messy from the beginning
- it's far from perfect
 - "if only Adobe Reader was open"

ACK

Paul Wheatley

@doegox @pdfkunfoo @newsoft

@internot @insertscript @avlidienbrunn

@foxgrrl @chrisjohnriley @travisgoodspeed

and everybody for the PDF suggestions :)

@angealbertini
corkami.com

