CS 5332 – Digital Forensics Assignment 1

Part 1) To be able analyze the PDF file with embedded DOC document that drops EICAR test file, first I extracted the zip file and renamed the pdf file to mal.pdf in REMnuxV6. To get basic information about the pdf I used the pdfid.py command.

```
remnux@remnux: ~/Downloads
File  Edit  Tabs  Help
remnux@remnux:~/Downloads$ sudo pdfid.py mal.pdf
PDFiD 0.2.1 mal.pdf
 PDF Header: %PDF-1.1
 obj                    9
 endobj                 9
 stream                 2
 endstream              2
 xref                   1
 trailer                1
 startxref              1
 /Page                  1
 /Encrypt               0
 /ObjStm                0
 /JS                    1
 /JavaScript            1
 /AA                    0
 /OpenAction            1
 /AcroForm              0
 /JBIG2Decode           0
 /RichMedia             0
 /Launch                0
```

```
remnux@remnux: ~/Downloads
File  Edit  Tabs  Help
PDFiD 0.2.1 mal.pdf
 PDF Header: %PDF-1.1
 obj                    9
 endobj                 9
 stream                 2
 endstream              2
 xref                   1
 trailer                1
 startxref              1
 /Page                  1
 /Encrypt               0
 /ObjStm                0
 /JS                    1
 /JavaScript            1
 /AA                    0
 /OpenAction            1
 /AcroForm              0
 /JBIG2Decode           0
 /RichMedia             0
 /Launch                0
 /EmbeddedFile          1
```

From the results we can see that pdf has JavaScript code and a file embedded to it. To get a look at the embedded file I used pdf-parser.py.

The embedded file is a doc file named eicar-dropper and it contains streams and it is compressed by the standard compression technique Flat Decode.

```
remnux@remnux: ~/Downloads
File  Edit  Tabs  Help
remnux@remnux:~/Downloads$ pdf-parser.py -s embedded
file mal.pdf
obj 1 0
 Type: /Catalog
 Referencing: 2 0 R, 3 0 R, 7 0 R, 9 0 R

  <<
    /Type /Catalog
    /Outlines 2 0 R
    /Pages 3 0 R
    /Names
      <<
        /EmbeddedFiles
          <<
            /Names [(eicar-dropper.doc) 7 0 R]
          >>
      >>
    /OpenAction 9 0 R
  >>
```

```
remnux@remnux: ~/Downloads
File  Edit  Tabs  Help
      <<
        /EmbeddedFiles
          <<
            /Names [(eicar-dropper.doc) 7 0 R]
          >>
      >>
    /OpenAction 9 0 R
  >>


obj 8 0
 Type: /EmbeddedFile
 Referencing:
 Contains stream

  <<
    /Length 8952
    /Filter /FlateDecode
    /Type /EmbeddedFile
  >>
```

To get a more detailed information about the embedded file, I used the same command and pipelined it with more.

```
remnux@remnux: ~/Downloads
File  Edit  Tabs  Help
remnux@remnux:~/Downloads$ pdf-parser.py -s embeddedfile -f
 mal.pdf | more
obj 1 0
 Type: /Catalog
 Referencing: 2 0 R, 3 0 R, 7 0 R, 9 0 R

  <<
    /Type /Catalog
    /Outlines 2 0 R
    /Pages 3 0 R
    /Names
      <<
        /EmbeddedFiles
          <<
            /Names [(eicar-dropper.doc) 7 0 R]
          >>
      >>
    /OpenAction 9 0 R
  >>

 [(1, '\r\n'), (2, '<<'), (1, '\r\n '), (2, '/Type'), (1, '
```

```
remnux@remnux: ~/Downloads
File  Edit  Tabs  Help
        /EmbeddedFiles
          <<
            /Names [(eicar-dropper.doc) 7 0 R]
          >>
      >>
    /OpenAction 9 0 R
  >>

 [(1, '\r\n'), (2, '<<'), (1, '\r\n '), (2, '/Type'), (1, '
 '), (2, '/Catalog'), (1, '\r\n '), (2, '/Outlines'), (1, '
 '), (3, '2'), (1, ' '), (3, '0'), (1, ' '), (3, 'R'), (1,
'\r\n '), (2, '/Pages'), (1, ' '), (3, '3'), (1, ' '), (3,
'0'), (1, ' '), (3, 'R'), (1, '\r\n '), (2, '/Names'), (1,
' '), (2, '<<'), (1, ' '), (2, '/EmbeddedFiles'), (1, ' '),
 (2, '<<'), (1, ' '), (2, '/Names'), (1, ' '), (2, '['), (2
, '('), (3, 'eicar-dropper.doc'), (2, ')'), (1, ' '), (3, '
7'), (1, ' '), (3, '0'), (1, ' '), (3, 'R'), (2, ']'), (1,
' '), (2, '>>'), (1, ' '), (2, '>>'), (1, '\r\n '), (2, '/O
penAction'), (1, ' '), (3, '9'), (1, ' '), (3, '0'), (1, '
'), (3, 'R'), (1, '\r\n'), (2, '>>'), (1, '\r\n')]
```

```
remnux@remnux: ~/Downloads
File  Edit  Tabs  Help
obj 8 0
 Type: /EmbeddedFile
 Referencing:
 Contains stream

  <<
    /Length 8952
    /Filter /FlateDecode
    /Type /EmbeddedFile
  >>

 '\xd0\xcf\x11\xe0\xa1\xb1\x1a\xe1\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x00\x00\x00\x00\x00>\x00\x03\x00\xfe\x
f\t\x00\x06\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x01
\x00\x00\x00&\x00\x00\x00\x00\x00\x00\x00\x00\x10\x00\x00(\
x00\x00\x00\x02\x00\x00\x00\xfe\xff\xff\xff\x00\x00\x00\x00
%\x00\x00\x00\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\x
ff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\
xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff
\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xf
f\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\xff\x
```

The document starts with '\xd0\xcf\x11\xe0…' which is the signature of ole files. That is why I continue with oledump.py command.

```
remnux@remnux:~/Downloads$ pdf-parser.py -s embeddedfile -f
 -d - mal.pdf | oledump.py
  1:        114 '\x01CompObj'
  2:       4096 '\x05DocumentSummaryInformation'
  3:       4096 '\x05SummaryInformation'
  4:       6509 '1Table'
  5:        409 'Macros/PROJECT'
  6:         65 'Macros/PROJECTwm'
  7: M     3716 'Macros/VBA/Module1'
  8: m      924 'Macros/VBA/ThisDocument'
  9:       2601 'Macros/VBA/_VBA_PROJECT'
 10:        563 'Macros/VBA/dir'
 11:       4096 'WordDocument'
remnux@remnux:~/Downloads$
```

We can see that the embedded word document contains Macros. That is why I ran oledump with two plugins, first dridex plugin for decoding and http_heuristics for checking if there is an http request in embedded doc. We can see that the file is not trying make an http request.

```
remnux@remnux: ~/Downloads
File  Edit  Tabs  Help
remnux@remnux:~/Downloads$ pdf-parser.py -s  embeddedfile -
f -d - mal.pdf | oledump.py -p plugin_dridex.py,plugin_http
_heuristics.py
  1:       114 '\x01CompObj'
  2:      4096 '\x05DocumentSummaryInformation'
  3:      4096 '\x05SummaryInformation'
  4:      6509 '1Table'
  5:       409 'Macros/PROJECT'
  6:        65 'Macros/PROJECTwm'
  7: M    3716 'Macros/VBA/Module1'
               Plugin: Dridex decoder
               Plugin: HTTP Heuristics plugin
                 Module1
                 '\xb5\xe9\xa9'
  8: m     924 'Macros/VBA/ThisDocument'
               Plugin: Dridex decoder
               Plugin: HTTP Heuristics plugin
                 'N\x18\xac\x0e\x87.\x99\xe9\xed'
  9:      2601 'Macros/VBA/_VBA_PROJECT'
 10:       563 'Macros/VBA/dir'
 11:      4096 'WordDocument'
```

To analyze the JavaScript I used pdf-parser again. The JavaScript code is not obfuscated and what it does is exports the data in the embedded word document.

```
remnux@remnux:~/Downloads$ pdf-parser.py -s /javascript mal
.pdf
obj 9 0
 Type: /Action
 Referencing:

  <<
    /Type /Action
    /S /JavaScript
    /JS (this.exportDataObject({ cName: "eicar-dropper.doc"
, nLaunch: 2 });)
  >>
```

Part 2) I documented my walkthrough of each level through screenshots:

Level 0:

ssh bandit0@bandit.labs.overthewire.org -p 2220

password: bandit0

Level 0 to Level 1:

```
bandit0@bandit:~$ ls
readme
bandit0@bandit:~$ cat readme
boJ9jbbUNNfktd78OOpsqOltutMc3MY1
```

Level 1 to Level 2:

```
bandit1@bandit:~$ ls
-
bandit1@bandit:~$ cat "./-"
CV1DtqXWVFXTvM2F0k09SHz0YwRINYA9
bandit1@bandit:~$ exit
```

Level 2 to Level 3:

```
bandit2@bandit:~$ ls
spaces in this filename
bandit2@bandit:~$ cat "spaces in this filename"
UmHadQclWmgdLOKQ3YNgjWxGoRMb5luK
```

Level 3 to Level 4:

```
bandit3@bandit:~$ ls
inhere
bandit3@bandit:~$ cd inhere
bandit3@bandit:~/inhere$ ls
bandit3@bandit:~/inhere$ ls -la
total 12
drwxr-xr-x 2 root    root    4096 Dec 28 14:34 .
drwxr-xr-x 3 root    root    4096 Dec 28 14:34 ..
-rw-r----- 1 bandit4 bandit3   33 Dec 28 14:34 .hidden
bandit3@bandit:~/inhere$ cat .hidden
pIwrPrtPN36QITSp3EQaw936yaFoFgAB
```

Level 4 to Level 5:

```
bandit4@bandit:~$ file inhere/*
inhere/-file00: data
inhere/-file01: data
inhere/-file02: data
inhere/-file03: data
inhere/-file04: data
inhere/-file05: data
inhere/-file06: data
inhere/-file07: ASCII text
inhere/-file08: data
inhere/-file09: data
bandit4@bandit:~$ cd inhere
bandit4@bandit:~/inhere$ cat "-file07"
cat: invalid option -- 'f'
Try 'cat --help' for more information.
bandit4@bandit:~/inhere$ cat "./-file07"
koReBOKuIDDepwhWk7jZC0RTdopnAYKh
bandit4@bandit:~/inhere$
```

Level 5 to Level 6:

```
bandit5@bandit:~$ ls
inhere
bandit5@bandit:~$ find inhere/ -size 1033c
inhere/maybehere07/.file2
bandit5@bandit:~$ cat inhere/maybehere07/.file2
DXjZPULLxYr17uwoI01bNLQbtFemEgo7
```

Level 6 to Level 7:

```
bandit6@bandit:~$ find / -user bandit7 -group bandit6 -size 33c -type f 2>/dev/n
ull
/var/lib/dpkg/info/bandit7.password
bandit6@bandit:~$ cat /var/lib/dpkg/info/bandit7.password
HKBPTKQnIay4Fw76bEy8PVxKEDQRKTzs
bandit6@bandit:~$
```

Level 7 to Level 8:

```
bandit7@bandit:~$ ls
data.txt
bandit7@bandit:~$ grep "millionth" data.txt
millionth       cvX2JJa4CFALtqS87jk27qwqGhBM9plV
bandit7@bandit:~$
```

Level 8 to Level 9:

```
bandit8@bandit:~$ sort data.txt | uniq -u
UsvVyFSfZZWbi6wgC7dAFyFuR6jQQUhR
bandit8@bandit:~$
```

Level 9 to Level 10:

```
bandit9@bandit:~$ strings data.txt | grep =
nfZ=
U=R*q
=-VW+
========== theP`
        =uN
\<P5J7=^
========== password
L='.
L========== isA
G&eB_=
9T=8?
9=!/"
========== truKLdjsbJ5g7yyJ2X2R0o3a5HQJFuLk
```

Level 10 to Level 11:

```
bandit10@bandit:~$ cat data.txt
VGhlIHBhc3N3b3JkIGlzIElGdWt3S0dzRlc4TU9xM0lSRnFyeEUxaHhUTkViVVBSCg==
bandit10@bandit:~$ echo VGhlIHBhc3N3b3JkIGlzIElGdWt3S0dzRlc4TU9xM0lSRnFyeEUxaHhUTkViVV
BSCg==
VGhlIHBhc3N3b3JkIGlzIElGdWt3S0dzRlc4TU9xM0lSRnFyeEUxaHhUTkViVVBSCg==
bandit10@bandit:~$ echo VGhlIHBhc3N3b3JkIGlzIElGdWt3S0dzRlc4TU9xM0lSRnFyeEUxaHhUTkViVV
BSCg== | base64 --decode
The password is IFukwKGsFW8MOq3IRFqrxE1hxTNEbUPR
bandit10@bandit:~$
```

Level 11 to Level 12:

```
bandit11@bandit:~$ cat data.txt | tr 'A-Za-z' 'N-ZA-Mn-za-m'
The password is 5Te8Y4drgCRfCx8ugdwuEX8KFC6k2EUu
```

Level 12 to Level 13:

```
bandit12@bandit:/tmp/dfs12$ mv data.out data.gz
bandit12@bandit:/tmp/dfs12$ gzip -d data.gz
bandit12@bandit:/tmp/dfs12$ file data
data: POSIX tar archive (GNU)
bandit12@bandit:/tmp/dfs12$ tar -xf data
bandit12@bandit:/tmp/dfs12$ ls
data  data.bin  data.txt  data5.bin
bandit12@bandit:/tmp/dfs12$ file data5.bin
data5.bin: POSIX tar archive (GNU)
bandit12@bandit:/tmp/dfs12$ tar -xf data5.bin
bandit12@bandit:/tmp/dfs12$ ls
data  data.bin  data.txt  data5.bin  data6.bin
bandit12@bandit:/tmp/dfs12$ file data6.bin
data6.bin: bzip2 compressed data, block size = 900k
bandit12@bandit:/tmp/dfs12$ bzip2 -d data6.bin
bzip2: Can't guess original name for data6.bin -- using data6.bin.out
bandit12@bandit:/tmp/dfs12$ ls
data  data.bin  data.txt  data5.bin  data6.bin.out
bandit12@bandit:/tmp/dfs12$ file data6.bin.out
data6.bin.out: POSIX tar archive (GNU)
bandit12@bandit:/tmp/dfs12$ tar -xf data6.bin.out
bandit12@bandit:/tmp/dfs12$ ls
data  data.bin  data.txt  data5.bin  data6.bin.out  data8.bin
bandit12@bandit:/tmp/dfs12$ file data8
data8: cannot open `data8' (No such file or directory)
bandit12@bandit:/tmp/dfs12$ file data8.bin
data8.bin: gzip compressed data, was "data9.bin", last modified: Thu Dec 28 13:34:36
017, max compression, from Unix
bandit12@bandit:/tmp/dfs12$ mv data8.bin data8.gz
bandit12@bandit:/tmp/dfs12$ gzip -d data8.gz
bandit12@bandit:/tmp/dfs12$ ls
data  data.bin  data.txt  data5.bin  data6.bin.out  data8
bandit12@bandit:/tmp/dfs12$ cat data8
The password is 8ZjyCRiBWFYkneahHwxCv3wb2a1ORpYL
```

Level 13 to Level 14:

```
bandit13@bandit:~$ ls
sshkey.private
bandit13@bandit:~$ cat sshkey.private
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEAxkkOE83W2cOT7IWhFc9aPaaQmQDdgzuXCv+ppZHa++buSkN+
gg0tcr7Fw8NLGa5+Uzec2rEg0WmeevB13AIoYp0MZyETq46t+jk9puNwZwIt9XgB
ZufGtZEwWbFWw/vVLNwOXBe4UWStGRWzgPpEeSv5Tb1VjLZIBdGphTIK22Amz6Zb
ThMsiMnyJafEwJ/T8PQO3myS91vUHEuoOMAzoUID4kN0MEZ3+XahyK0HJVq68KsV
DbefXG1vvA3GAJ29kxJaqvRfgYnqZryWN7w3CHjNU4c/2Jkp+n8L0SnxaNA+WYA7
jiPyTF0is8uzMlYQ4l1Lzh/8/MpvhCQF8r22dwIDAQABAoIBAQC6dWBjhyEOzjeA
D3j/RWmap9M5zfJ/wb2bfidNpwbB8rsJ4sZIDZQ7XuIh4LfygoAQSS+bBw3RXvzE
bvJt3SmU8hIDuLsCjL1VnBY5pY7Bju8g8aR/3FyjyNAqx/TLfzlLYfOu7i9Jet67
xAh0tONG/u8FB5I3LAI2Vp6OviwvdWeC4nOxCthldpuPKNLA8rmMMVRTKQ+7T2VS
hXmwYckKUcUgzoVSpiNZaS0zUDypdpy2+tRH3MQa5kqN1YKjvF8RC47woOYCktsD
b3FFpGNFec9Taa3Msy+DfQQhHKZFKIL3bJDONtmrVvtYK40/yeU4aZ/HA2DQzwhe
bl1AfiEhAoGBAOnVjosBkm7sblK+n4IEwPxs8sOmhPnTDUy5WGrpSCrXOmsVIBUf
laL3ZGLx3xCIwtCnEucB9DvN2HZkupc/h6hTKUYLqXuyLD8njTrbRhLgbC9QrKrS
M1F2fSTxVqPtZDlDMwjNR04xHA/fKh8bXXyTMqOHNJTHHNhbh3McdURjAoGBANkU
1hqfnw7+aXncJ9bjysr1ZWbqOE5Nd8AFgfwaKuGTTVX2NsUQnCMWdOp+wFak40JH
PKWkJNdBG+ex0H9JNQsTK3X5PBMAS8AfX0GrKeuwKWA6erytVTqjOfLYcdp5+z9s
3DtVCxDuVsM+i4X8UqIGOlvGbtKEVokHPFXP1q/dAoGAcHg5YX7WEehCgCYTzpO+
xysX8ScM2qS6xuZ3MqUWAxUWkh7NGZvhe0sGy9iOdANzwKw7mUUFViaCMR/t54W1
GC83sOs3D7n5Mj8x3NdO8xFit7dT9a245TvaoYQ7KgmqpSg/ScKCw4c3eiLava+J
3btnJeSIU+8ZXq9XjPRpKwUCgYA7z6LiOQKxNeXH3qHXcnHok855maUj5fJNpPbY
iDkyZ8ySF8GlcFsky8Yw6fWCqfG3zDrohJ5l9JmEsBh7SadkwsZhvecQcS9t4vby
9/8X4jS0P8ibfcKS4nBP+dT81kkkg5Z5MohXBORA7VWx+AChcDEkprsQ+w32xeD
qT1EvQKBgQDKm8ws2ByvSUVs9GjTilCajFqLJ0eVYzRPaY6f++Gv/UVfAPV4c+S0
kAWpXbv5tbkkzbS0eaLPTKgLzavXtQoTtKwrjpolHKIHUz6Wu+n4abfAIRFubOdN
/+aLoRQ0yBDRbdXMsZN/jvY44eM+xRLdRVyMmdPtP8belRi2E2aEzA==
-----END RSA PRIVATE KEY-----
bandit13@bandit:~$ ssh -i sshkey.private bandit14@localhost -p 2220
ssh: connect to host localhost port 2220: Connection refused
bandit13@bandit:~$ ssh -i sshkey.private bandit14@localhost
Could not create directory '/home/bandit13/.ssh'.
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:98UL0ZWr85496EtCRkKlo20X3OPnyPSB5tB5RPbhczc.
```

Level 14 to Level 15:

```
bandit14@bandit:~$ cat /etc/bandit_pass/bandit14
4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e
bandit14@bandit:~$ telnet localhost 30000
Trying ::1...
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
4wcYUJFw0k0XLShlDzztnTBHiqxU3b3e
Correct!
BfMYroe26WYalil77FoDi9qh59eK5xNr

Connection closed by foreign host.
bandit14@bandit:~$
```

Level 15 to Level 16:

```
  For support, questions or comments, contact us through IRC on
  irc.overthewire.org #wargames.

  Enjoy your stay!

bandit15@bandit:~$ cat /etc/bandit_pass/bandit15
BfMYroe26WYalil77FoDi9qh59eK5xNr
bandit15@bandit:~$ openssl s_client -connect localhost:30001 -quiet
depth=0 CN = bandit
verify error:num=18:self signed certificate
verify return:1
depth=0 CN = bandit
verify return:1
BfMYroe26WYalil77FoDi9qh59eK5xNr
Correct!
cluFn7wTiGryunymYOu4RcffSxQluehd

bandit15@bandit:~$
```

Level 16 to Level 17:

```
bandit16@bandit:~$ chmod 600 /tmp/key17/sshkey.pem
bandit16@bandit:~$ ssh -i /tmp/key17/sshkey.pem bandit17@localhost
```

Level 17 to Level 18:

```
bandit17@bandit:~$ diff passwords.new passwords.old
42c42
< kfBf3eYk5BPBRzwjqutbbfE887SVc5Yd
---
```

Level 18 to Level 19:

```
Byebye !
Connection to bandit.labs.overthewire.org closed.
sevgi@arcadia:~$ ssh bandit18@bandit.labs.overthewire.org -p 2220 cat readme
This is a OverTheWire game server. More information on http://www.overthewire.org/warg
ames
bandit18@bandit.labs.overthewire.org's password:
IueksS7Ubh8G3DCwVzrTd8rAVOwq3M5x
sevgi@arcadia:~$
```

Level 19 to Level 20:

```
  Enjoy your stay!

bandit19@bandit:~$ ls
bandit20-do
bandit19@bandit:~$ file bandit20-do
bandit20-do: setuid ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamica
lly linked, interpreter /lib/ld-linux.so.2, for GNU/Linux 2.6.32, BuildID[sha1]=1c05d8
0e62cd205a3497b870e8294402424a4f7c, not stripped
bandit19@bandit:~$ ls -l
total 8
-rwsr-x--- 1 bandit20 bandit19 7408 Dec 28 14:34 bandit20-do
bandit19@bandit:~$ ./bandit20-do
Run a command as another user.
  Example: ./bandit20-do id
bandit19@bandit:~$ ./bandit20-do cat /etc/bandit_pass/bandit20
GbKksEFF4yrVs6il55v6gwY5aVje5f0j
bandit19@bandit:~$
```

Level 20 to Level 21:

```
  For more information rega  Enjoy your stay!
  http://www.overthewire.or
                             bandit20@bandit:~$ ./connect 32100
  For support, questions or-bash: ./connect: No such file or directory
  irc.overthewire.org #warg bandit20@bandit:~$ ./suconnect 32100
                             Read: GbKksEFF4yrVs6il55v6gwY5aVje5f0j
  Enjoy your stay!           Password matches, sending next password
                             bandit20@bandit:~$
bandit20@bandit:~$ ls
suconnect
bandit20@bandit:~$ ./suconnect
Usage: ./suconnect <portnumber>
This program will connect to the given port on localhost using TCP. If it receives the
 correct password from the other side, the next password is transmitted back.
bandit20@bandit:~$ nc -l 32100 < /etc/bandit_pass/bandit20
gE269g2h3mw3pwgrj0Ha9Uoqen1c9DGr
bandit20@bandit:~$
```

Level 21 to Level 22:

```
bandit21@bandit:~$ cd /etc/cron.d/
bandit21@bandit:/etc/cron.d$ ls -l
total 16
-rw-r--r-- 1 root root 120 Dec 28 14:34 cronjob_bandit22
-rw-r--r-- 1 root root 122 Dec 28 14:34 cronjob_bandit23
-rw-r--r-- 1 root root 120 Dec 28 14:34 cronjob_bandit24
-rw-r--r-- 1 root root 190 Oct 31 13:21 popularity-contest
bandit21@bandit:/etc/cron.d$ cat cronjob_bandit22
@reboot bandit22 /usr/bin/cronjob_bandit22.sh &> /dev/null
* * * * * bandit22 /usr/bin/cronjob_bandit22.sh &> /dev/null
bandit21@bandit:/etc/cron.d$ cat /usr/bin/cron_job_bandit22.sh
cat: /usr/bin/cron_job_bandit22.sh: No such file or directory
bandit21@bandit:/etc/cron.d$ cat /usr/bin/cronjob_bandit22.sh
#!/bin/bash
chmod 644 /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv
cat /etc/bandit_pass/bandit22 > /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv
bandit21@bandit:/etc/cron.d$ cat /tmp/t7O6lds9S0RqQh9aMcz6ShpAoZKF7fgv
Yk7owGAcWjwMVRwrTesJEwB7WVOiILLI
bandit21@bandit:/etc/cron.d$
```

Level 22 to Level 23:

```
bandit22@bandit:~$ ls -l
total 0
bandit22@bandit:~$ cd /etc/cron.d/
bandit22@bandit:/etc/cron.d$ ls -l
total 16
-rw-r--r-- 1 root root 120 Dec 28 14:34 cronjob_bandit22
-rw-r--r-- 1 root root 122 Dec 28 14:34 cronjob_bandit23
-rw-r--r-- 1 root root 120 Dec 28 14:34 cronjob_bandit24
-rw-r--r-- 1 root root 190 Oct 31 13:21 popularity-contest
bandit22@bandit:/etc/cron.d$ cat cronjob_bandit23
@reboot bandit23 /usr/bin/cronjob_bandit23.sh  &> /dev/null
* * * * * bandit23 /usr/bin/cronjob_bandit23.sh  &> /dev/null
bandit22@bandit:/etc/cron.d$ cat /usr/bin/cronjob_bandit23.sh
#!/bin/bash

myname=$(whoami)
mytarget=$(echo I am user $myname | md5sum | cut -d ' ' -f 1)

echo "Copying passwordfile /etc/bandit_pass/$myname to /tmp/$mytarget"

cat /etc/bandit_pass/$myname > /tmp/$mytarget
bandit22@bandit:/etc/cron.d$ cat /tmp/$mytarget
cat: /tmp/: Permission denied
bandit22@bandit:/etc/cron.d$ /usr/bin/cronjob_bandit23.sh
Copying passwordfile /etc/bandit_pass/bandit22 to /tmp/8169b67bd894ddbb4412f91573b38db
3
bandit22@bandit:/etc/cron.d$ cat /tmp/8169b67bd894ddbb4412f91573b38db
cat: /tmp/8169b67bd894ddbb4412f91573b38db: No such file or directory
bandit22@bandit:/etc/cron.d$ cat /tmp/8169b67bd894ddbb4412f91573b38db3
Yk7owGAcWjwMVRwrTesJEwB7WVOiILLI
bandit22@bandit:/etc/cron.d$ echo I am user bandit23 | md5sum | cut -d ' ' -f 1
8ca319486bfbbc3663ea0fbe81326349
bandit22@bandit:/etc/cron.d$ cat /tmp/8ca319486bfbbc3663ea0fbe81326349
jc1udXuA1tiHqjIsL8yaapX5XIAI6i0n
bandit22@bandit:/etc/cron.d$
```

Level 23 to Level 24:

```
bandit23@bandit:~$ cd /tmp/bnt23
bandit23@bandit:/tmp/bnt23$ ls
getpass.sh
bandit23@bandit:/tmp/bnt23$ cat getpass.sh
#!/bin/bash
cat /etc/bandit_pass/bandit24 > tmp/bnt23/pass.txt
bandit23@bandit:/tmp/bnt23$ vim getpass.sh
bandit23@bandit:/tmp/bnt23$ chmod 777 getpass.sh
bandit23@bandit:/tmp/bnt23$ cp getpass.sh /var/spool/bandit24
bandit23@bandit:/tmp/bnt23$ cat pass.txt
cat: pass.txt: No such file or directory
bandit23@bandit:/tmp/bnt23$ ls
getpass.sh
bandit23@bandit:/tmp/bnt23$ cd ..
bandit23@bandit:/tmp$ cd ..
bandit23@bandit:/$ tmp/bandit24
-bash: tmp/bandit24: Is a directory
bandit23@bandit:/$ cd tmp/bandit24
bandit23@bandit:/tmp/bandit24$ ls
a.py  b.sh  passw
bandit23@bandit:/tmp/bandit24$ cat b.sh
#!/bin/bash

pass="UoMYTrfrBFHyQXmg6gzctqAwOmw1IohZ"
```

Level 24 to Level 25:

```
bandit24@bandit:/tmp/bnt24$ cat nano getpin.sh
cat: nano: No such file or directory
#!/bin/bash

START=0000
LAST_PIN=9999
HOST="localhost"
PORT=30002
PASSWD="UoMYTrfrBFHyQXmg6gzctqAwOmw1IohZ"

for i in {0..9}{0..9}{0..9}{0..9}
do
        echo $PASSWD' '$i | nc $HOST $PORT >> r &
done
bandit24@bandit:/tmp/bnt24$ chmod 777 ./getpin.sh
bandit24@bandit:/tmp/bnt24$ ./getpin.sh
```

After the execution the command: sort r | uniq -u will return the password, however I think there is a problem with this level because it does not work.

Level 25 to Level 26:

```
  irc.overthewire.org #wargames.

  Enjoy your stay!

bandit25@bandit:~$ cat /usr/bin/showtext
#!/bin/sh

export TERM=linux

more ~/text.txt
exit 0
bandit25@bandit:~$
```

```
 _                    _ _ _   ___   __
| |                  | (_) | |__ \ / /
| |__   __ _ _ __   __| |_| |_   ) / /_
| '_ \ / _` | '_ \ / _` | | __| / / '_ \
"~/text.txt" [readonly] 6L, 258C
```

ssh.exe

Before running ‘ssh -i bandit26.sshkey bandit26@localhost’ command we need to minimize the screen then open the vim editor and type ‘:r /etc/bandit_pass/bandit26’ to get the password.

Level 26 to Level 27: This level does not exist yet.

Part 3)

For the analysis I used the following tutorial**.**

https://countuponsecurity.com/2014/09/22/malicious-documents-pdf-analysis-in-5-steps/

To analyze the PDF file, I first rename it to hw1.pdf (to make the analysis easier)

1) To find the number of objects I used pdfid command. The number of objects in the file is 6.

```
remnux@remnux: ~/Downloads
File  Edit  Tabs  Help
remnux@remnux:~/Downloads$ pdfid hw1.pdf
PDFiD 0.2.1 hw1.pdf
 PDF Header: %PDF-1.7
 obj                    6
 endobj                 6
 stream                 2
 endstream              2
 xref                   1
 trailer                1
 startxref              1
 /Page                  1
 /Encrypt               0
 /ObjStm                0
 /JS                    0
 /JavaScript            0
```

2) To see if the file is compressed I use pdf-parser tool. It is seen that the file is encoded with flat encode (Filter/F1which is a compression algorithm.

```
remnux@remnux: ~/Downloads
File  Edit  Tabs  Help
remnux@remnux:~/Downloads$ pdf-parser.py hw1.pdf
PDF Comment '%PDF-1.7\n'

PDF Comment '%\xc0\xff\xee\xfa\xba\xda\n'

obj 1 0
 Type:
 Referencing:
 Contains stream

  <<
    /Filter [ /Fl /Fl ]
    /L 544
  >>


obj 2 0
 Type:
 Referencing: 1 0 R
```

3) The pdfid command that I used for object detection also shows that the file has AcroForm which indicates that the file may contain JavaScript that is obfuscated. It also ha XFA forms which shows that this PDF has high possibility of containing malicious elements.

```
remnux@remnux: ~/Downloads
File  Edit  Tabs  Help
obj                     6
endobj                  6
stream                  2
endstream               2
xref                    1
trailer                 1
startxref               1
/Page                   1
/Encrypt                0
/ObjStm                 0
/JS                     0
/JavaScript             0
/AA                     0
/OpenAction             0
/AcroForm               1
/JBIG2Decode            0
/RichMedia              0
/Launch                 0
/EmbeddedFile           0
/XFA                    1
```

4) From pdf-parser tool's output we can see that object 2 is referencing object 1 which seems to have compressed malicious elements. However, to make it clearer I also used peepdf tool. At first peepdf did not work but I add -f at the end of the command to force the analysis. The output shows that object 1 has JavaScript code which was not visible on the other tools' output.

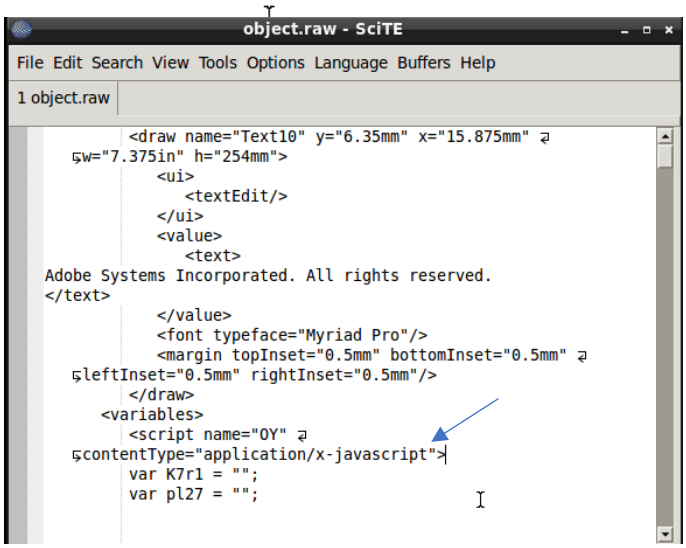
```
Updates: 0
Objects: 6
Streams: 2
Comments: 0
Errors: 0

Version 0:
        Catalog: 3
        Info: No
        Objects (6): [1, 2, 3, 4, 5, 6]
                Errors (2): [1, 6]
        Streams (2): [1, 6]
                Encoded (1): [1]
        Objects with JS code (1): [1]
        Suspicious elements:
                /AcroForm: [3]
                /XFA: [2]
                BMP/RLE heap corruption (CVE-2013-2729):
[1]
```

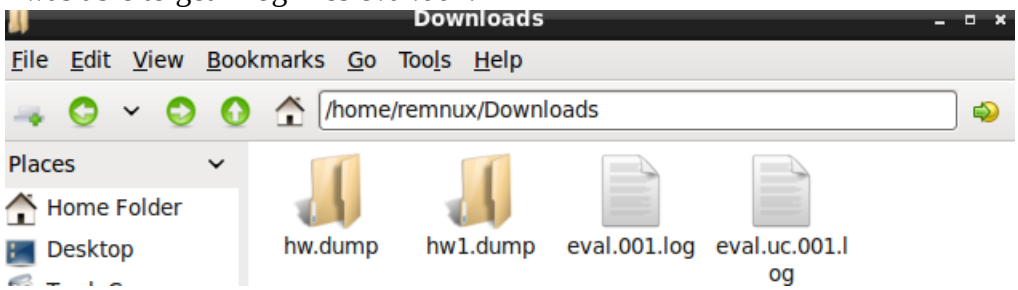
To output the object 1's stream I used the following command:

```
remnux@remnux:~/Downloads$ pdf-parser.py -c hw1.pdf -
-object 1 --filter --raw > object.raw
remnux@remnux:~/Downloads$
```

Then looking into the object.raw revealed the obfuscated JavaScript code.



5) To de-obfuscate the JavaScript code, I manually extract the JavaScript part of the stream and clean the code a little bit to be able to use js-didier tool to interpret and execute the extracted code.
   Then I ran js.didier h1.pdf
   I was able to get 2 log files eval.001:



However, both of the .log files have only one line of entry on them which is not enough to continue the analysis with creating a shellcode. That is why I could not go further with the analysis. The content of the .log files are shown below: