Data Retrieval from Slack Space

March 27, 2025

Contents

1 Foremost setup

The lab calls for us to edit our foremost configuration file. We find a line in /etc/foremost.conf that reads like so: ~ pdf y 5000000 %PDF- %EOF~ This will tell foremost to find files that have the .pdf file extension, of at most 5MB, with file signatures %PDF~ and %EOF for beginning of file and end of file respectively.

2 Run

Running foremost on out $L0_{\text{Documents.dd}}$ file now will output a pdf that in a text editor, will look something like:

```
%PDF-1.6
%
2285 0 obj <</Linearized 1/L 3176275/O 2290/E 503535/N 4/T 3130509/H [ 7861 958]>>
endobj

xref
```

```
2285 370
0000000016 00000 n
0000008819 00000 n
0000008960 00000 n
...
0000502995 00000 n
0000503056 00000 n
0000503117 00000 n
0000503178 00000 n
0000503272 00000 n
0000503328 00000 n
0000503423 00000 n
0000503479 00000 n
0000007861 00000 n
trailer
<</Size 2655/Prev 3130496/Root 2287 0 R/Encrypt 2286 0 R/Info 2284 0 R/ID[<E4B41E4F063
startxref
0
%%EOF
```

This is not the whole of a pdf.

3 The structure of a pdf

Kessler:File Signatures A pdf may contain more than one %EOF marker, so when our config file only goes up to the first instance we lose data following subsequent %EOF markers. Our file carving needs to be a bit more lenient.

4 Conclusion

Given the shenaniganns using the `-t` flag is important to properly carve files using foremost.

4.1 Excursus: "Slack space"

Due to the nature of file carving as we have seen above it is a byte to byte function, so when we're copying the carve out into our hard drive it will write over the blocks the Filesystem designates for the "new" carved file, thus the slack space is from your kali box, rather than from the image itself.