BitCurator: Generating PDF Reports

1. Purpose:

The Python script generate_report.py depends on the output files generated by the utilities "fiwalk", "bulk_extractor" and identify_files. fiwalk generates a text file with detailed information on each file residing on the image file. bulk_extractor generates a number of text files, one for each feature found on the media image. generate_report.py uses these text files to generate reports in the form of PDF files. A configuration file is provided for the user to control and configure the reports that need to be generated. The following are some of the reports it generates:

2. Reports

2.1.1. Feature reports:

The bulk_extractor utility generates a number of text files, one per feature that it finds in the image file. Examples of features are email, url, domain, etc. The "identify_filenames" script takes these text files as inputs and generates one text file for each of these features by reorganizing the per-file information in a more orderly fashion. generate_reports script generates one PDF file per feature with the per-feature information tabulated. A user can indicate which feature reports and how many lines of each report to generate. Another report, beReport.pdf, is generated, which has the statistical information for each feature file, extracted from the annotated files.

2.1.2. Filesystem statistics and files:

The filwalk utility generates a text file which contains per-file information of all the files present in the image. Using this data, generate_reports script produces a PDF file with two tables: First one is a table containing the technical metadata, with information on the number of files, directories, deleted files, etc., and another table with each row dedicated to a file, with information like partition, size, filetype, etc. A user can choose to generate this report or not, using the configuration file.

2.1.3. Deleted files:

A table with all the deleted files

2.1,4. File format bar-graph and table:

generate_reports script generates a bar-graph describing the number of files of a particular file format. It also generates a table that goes with this graph, where it tabulates the same information. A user can choose to generate this report or not, using the configuration file.

2.1.5. File format tables

For each format type, a report is generated giving the filenames in this format. There could be a large number of formats, depending on the image file. A user can specify how many format reports to be generated.

3. Usage

generate_report is a python script, which takes as input, the directory where the annotated files generated by identify_filenames script reside and also the text file generated by the bulk_extractor utility. The output is a bunch of PDF files that will be placed in the directory specified by the “--outdir” parameter of the script. The script mandates the option "--pdf_report". The fiwalk-generated text file is specified with the option "--fiwalk_txtfile". Example invocation:
Command: python3 generate_report.py --pdf_report --fiwalk_txtfile <filepath> --annotated_dir <dirpath> --outdir <dirpath>

```
python3 generate_report.py --pdf_report --fiwalk_txtfile ~/Research/TestData/BEOutputs_131/
charlie_fi_FT.txt
  --annotated_dir ~/Research/TestData/BEOutputs_131/annotated_charlie_output
 --outdir ~/Research/TestData/BEOutputs_131/charlie_rep_outdir
```

3.1 Configuration file

By default, the configuration file bc_report_config.txt is defined. But the user is free to provide any filename. The script asks the user if a configuration file is going to be used. If the answer is "No", the default configuration is chosen. It generates first 5 feature files, all the non-feature files by default. If the answer is “Yes”, the user is prompted to either enter a filename, or press return to use the default file, bc_report_config.txt. If the file is not found, it puts out a warning and reverts to default config.

The following options can be configured using this file:

- The first character on a line tells what to expect from the subsequent text on that line. All the lines starting with a "#" are considered comments and the line is ignored.

- L: Logo: A .png image can be specified for the Logo. By default, the Bitcurator logo is used.
- F: Feature file. A feature file can be specified on this line, telling the program to generate the report file corresponding to this feature. The third field is the number of lines to be reported.

```
#F:rfc822:30
F:domain:0
F:email:100
F:exif:0
```

- R: Fiwalk and Bulk Extractor Report files. The following are the four files in this category. User can choose not to report them by commenting the lines out:
R:bc_format_bargraph:0
R:FiwalkReport:200
R:FiwalkDeletedFiles:0
R:BeReport:0

- Miscellaneous options:

- Don't display special files - those starting from "."

  S:REPORT_SPECIAL_FILES:YES
- Set maximum format files to report: The utility generates a one file for each format type, where it lists all the files of that format type. If the number of formats is huge, the number of files generated could be very large. So user can specify the number here to control the number of files generated.
  S:MAX_FILE_FORMAT_FILES_TO_REPORT:5
- Set a maximum number of feature files to report:
  One can specify an upper limit of feature files to generate the report for, by specifying the number with MAX_FEATURE_FILES_TO_REPORT
  S:MAX_FEATURE_FILES_TO_REPORT:3