BitCurator Access

bitcurator-access-redaction Quick Start Guide

Last updated: May 8, 2018

Release(s): 0.4.2 and later

UNC
SCHOOL OF INFORMATION
AND LIBRARY SCIENCE

About bitcurator-access-redaction

The **bitcurator-access-redaction** project builds on existing disk image redaction and Digital Forensics XML tools to provide collecting institutions with software to redact strings and byte sequences identified in raw disk images.

A command-line utility (**redact-cli**) allows users to execute bulk redaction actions against raw bitstreams and file system metadata identified by fiwalk. This guide describes how to install and use this utility.

The software includes a Python API allowing institutions to develop custom redaction facilities using powerful forensics search libraries such as **liblightgrep.**

This document will help you get started: installing bitcurator-access-redaction, using the redact-cli utility, and building some simple redaction configurations.

Getting Started with bca-redtools

- **Hardware and OS:**
  - Desktop or laptop with an Intel Core i5 or Core i7 processor (or AMD equivalent) running 64-bit Linux variant (either as the host operating system or in a Virtual Machine).
  - 8GB RAM or more

- **Software:**
  - **bitcurator-access-redaction** has been tested in Ubuntu 14.04LTS and 16.04LTS, and should run on any Linux platform (including BitCurator) where the dependencies can be compiled and/or installed
  - Current release of **bitcurator-access-redaction**. **Download** the latest release (**.zip** or **.tar.gz** file) from:
    - https://github.com/bitcurator/bitcurator-access-redaction/releases

Installing the software

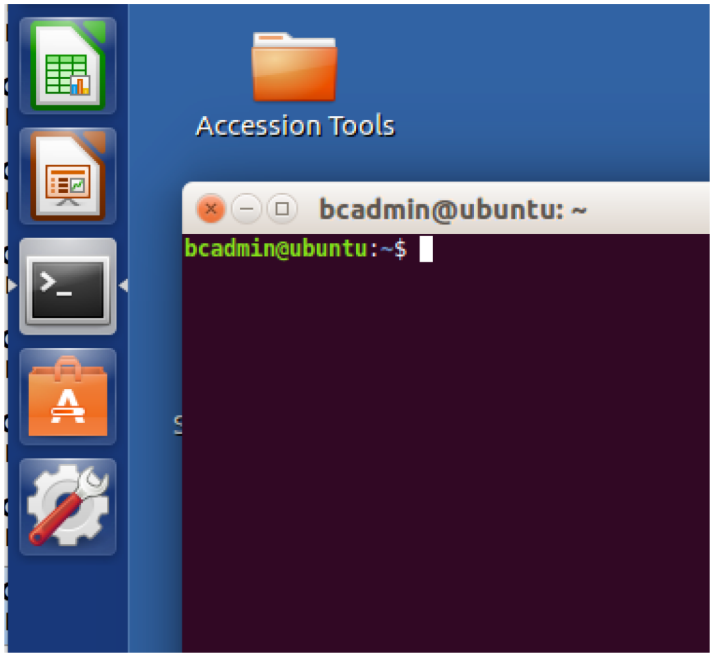
- **Note:** These instructions assume you are using BitCurator or an equivalent Ubuntu 16.04LTS environment, and that you are familiar using a terminal and navigating around the file system. If you are not familiar with the terminal, you can find a basic primer here:

  https://help.ubuntu.com/community/UsingTheTerminal

- **Open a new terminal.**

- In the BitCurator environment, you can do this by clicking on the black "Terminal" icon in the Unity launcher on the left hand side of the screen.

- **Note the "$" sign in the command prompt in the terminal window.** All commands in this tutorial include this symbol to indicate a terminal command. **Don't type the "$"!**



Installing the software

- In the terminal, and navigate to the directory where you downloaded the release. If you downloaded it using Firefox, it's probably in the "Downloads" directory:

```
$ cd /home/bcadmin/Downloads
```

(this path will be different if you're not using BitCurator)

- Move the file to your home directory (replace X.X.X with the release number):

```
$ mv /home/bcadmin/Downloads/bitcurator-access-redaction-X.X.X.zip /home/bcadmin
```

(Use this version if you downloaded the .zip file, or replace ".zip" with ".tar.gz")

- Now change directory to your home, and verify that the file is there:

```
$ cd /home/bcadmin
$ ls
```

- You should see a **list of all the files in your home directory**. If you don't see the **bitcurator-access-redaction-X.X.X.zip** (or **.tar.gz**) file, double-check your previous commands.

Installing the software

- In your terminal, unzip the file as follows (replace X.X.X with the release number):

  `$ unzip bitcurator-access-redaction-X.X.X.zip` (if you have the .zip file)

  `$ tar zxvf bitcurator-access-redaction-X-X-X.tar.gz` (if you have the .tar.gz file)

- Navigate to the libredact directory:

  `$ cd bitcurator-access-redaction-X.X.X/libredact`

- Build and install the software using pip:

  `$ pip install –e .`

- The following step is **NOT REQUIRED** in the BitCurator environment, but may be needed in other Linux installs:

  PIP builds an executable script in **/home/youruser/.local/bin/redact-cli**. You can add this script (temporarily) to your path using the following command:

  `$ PATH=$PATH:/home/`whoami`/.local/bin`

Intro to running redact-cli

- The **redact-cli** tool automates the process of performing bulk redactions on raw disk images. This includes the ability to overwrite specific strings or regular expression matches, byte runs that match specific files or directory entries (by consulting **fiwalk** output), and others. The remaining sections of this guide discuss these options, how to enable and configure them, and demonstrate a simple redaction using a test image.

- Before we get started, take a quick look at the output of the following commands:

  The command line tool can be run with a help (-h) flag to view the available options:

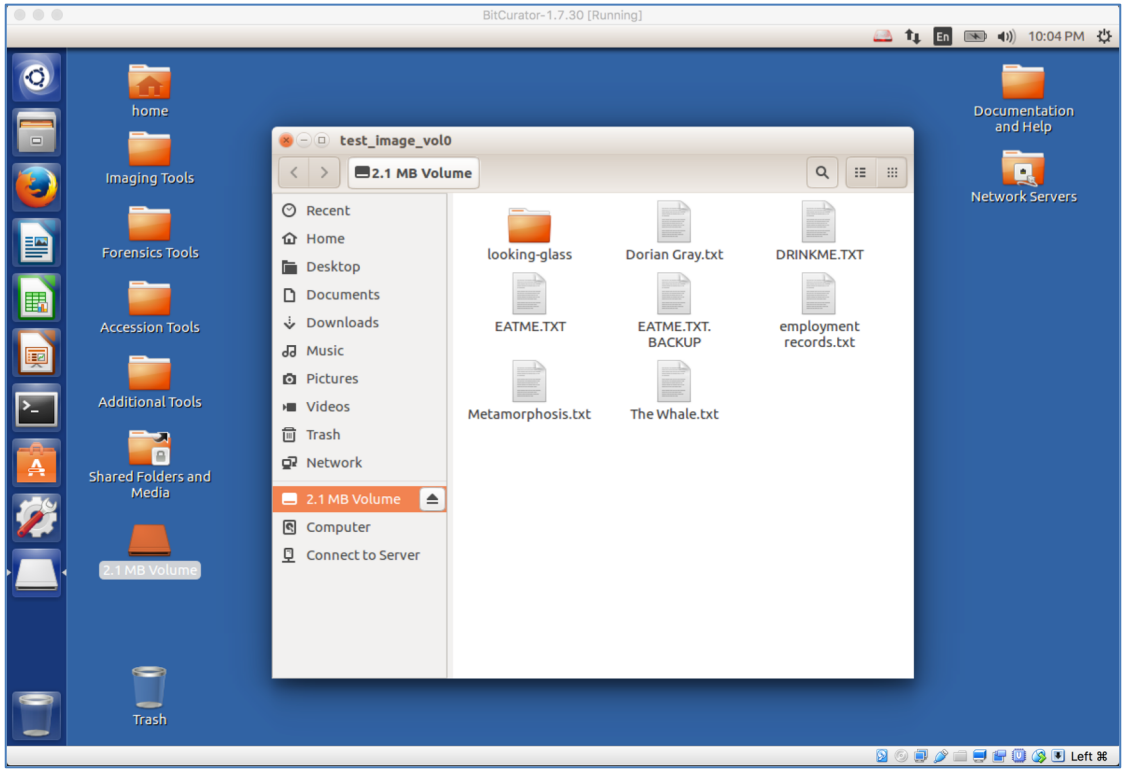
  ```
  $ redact-cli -h
  ```

  Running the tools with the capitalized (-H) flag provides additional detail on how to create configuration files to perform specific redactions.

  ```
  $ redact-cli -H
  ```

Selecting a disk image for redaction

- The **redact-cli** tool operates only on raw disk images. The disk image **should not be mounted** when running the tool.
- A small sample disk image (**test_image.raw**) has been included in the **test** directory (bca-redtools-X.X.X/libredact/test). In the BitCurator environment, you can view the contents of this image in a read-only mount by right-clicking on the file and selecting **Scripts->Mount Disk Image.** Don't forgot to unmount the image when you're done examining it!



Extracting file system metadata from the disk image

- You must generate a Digital Forensics XML file for the disk image using **fiwalk** if you wish to perform redactions that match specific types of file system metadata (for example, filenames, or entire files matching particular MD5 hashes)

- We'll copy the small test image to the Desktop for convenience in this example:

  ```
  $ cp ~/bca-redtools-X.X.X/libredact/test/test_image.raw ~/Desktop
  ```

  (Note: The "~" here is simply a shortcut for the home directory, /home/bcadmin)

- In BitCurator, you can run **fiwalk** in the terminal you have open. For this example, we'll simply send the output to the Desktop:

  ```
  $ fiwalk -f -X ~/Desktop/test_image_fw.xml ~/Desktop/test_image.raw
  ```

- You should now see both the disk image and the fiwalk output XML file on the Desktop.

The Redaction Configuration File, Explained

- The configuration file can specify complete instructions for how **redact-cli** runs. Arguments given on the command-line or in calls to the **redact-cli** API method will override settings in the configuration file.

- For this example, we'll put everything in the configuration file for simplicity. First, an overview of the syntax for this file.

  - The readaction configuration file consists of commands, one per line.
  - Order of the commands does not matter.
  - **Simple Commands:**

INPUT_FILE <file path>	path to disk image file to redact
OUTPUT_FILE <file path>	path to write the redacted disk image
DFXML_FILE <file path>	optional path to previously generated
DFXML REPORT_FILE <file path>	optional path to write audit report file
IGNORE <pattern>	ignore files whose names match regex (repeatable)
COMMIT	perform rule actions

The redaction configuration file, explained

- The redaction actions themselves are specified using a set of **Rule Commands**, each of which is triggered by a specific condition, and performs a given action:

- **Rule Command** Format: [target condition] [action]

  Target Conditions:

  FILE_NAME_EQUAL <filename> - target a file with the given filename

  FILE_NAME_MATCH <pattern> - target any file with a given filename pattern

  FILE_DIRNAME_EQUAL <directory> - target all files in the directory

  FILE_MD5 <md5> - target any file with the given md5

  FILE_SHA1 <sha1> - target any file with the given sha1

  FILE_SEQ_EQUAL <string> - target any file that contains <a string>

  FILE_SEQ_MATCH <pattern> - target any file that contains a sequence matching <a pattern>

  SEQ_EQUAL <string> - target any sequences equal to <a string>

  SEQ_MATCH <pattern> - target any sequences matching <a pattern>

  Actions:

SCRUB	overwrite the bytes in the target with zeroes
FILL 0x44	overwrite by filling with a given character (here, 0x44, or ASCII 'D')
FUZZ	fuzz the binary, but not the strings

The redaction configuration file, explained

- Finally, the target conditions shown in the previous slide may be triggered by matching a given pattern.
  - Patterns can be a wildcard expression. For example, '*.txt' to match files with the **.txt** extension
  - They can also be regular expressions. If you're not familiar with regular expressions, or need help building a regex to match a particular pattern, try this helpful online tool:

    http://regexr.com/
  - As an example, the following regular expression will target social security numbers:

    '\d{3}-?\d{2}-?\d{4}'
- **Be cautious with patterns and the SEQ_MATCH condition. It is easy to write a pattern that will match the whole file.**

Writing our own configuration file

- Configuration files for **redact-cli** are text files, with the **Simple** and **Rule** commands separated by newlines. Now, we'll create a new text file to hold our rules.

- In the BitCurator environment, you can right-click on the Desktop and select **New Document->Empty Document** to create un "Untitled Document" file. Double-click on this file to open it in **gedit**, a simple text editor.

  - This configuration file will include lines for an INPUT_FILE (our disk image), a DFXML_FILE (our DFXML metadata export) and an OUTPUT_FILE (our redacted disk image copy).
  - These **Simple** commands are followed by a series of **Rule** commands that specify a series of redactions.
  - The following slide includes comments describing each of the **Rule** command. Lines starting with a "**#**" are comments, and will be ignored by the program.

- **Copy or retype the contents of the next slide into the text file. When you are done, save the file as "test_image_config.txt" on the Desktop.**

Configuration file rules

```
INPUT_FILE /home/bcadmin/Desktop/test_image.raw
DFXML_FILE /home/bcadmin/Desktop/test_image_fw.xml
OUTPUT_FILE /home/bcadmin/Desktop/test_image_redacted.raw

# Targets The Whale.txt
FILE_NAME_MATCH *Whale.txt FUZZ

# Targets Dorian Gray.txt
FILE_MD5 114583cd8355334071e9343a929f6f7c FILL 0x44

# Targets DRINKME.TXT
FILE_SHA1 7f9f0286e16e9c74c992e682e27487a9eb691e86 FILL 0x44

# Fill Kafka sequences in Metamorphsis.txt with K
SEQ_EQUAL Kafka FILL 0x4B
SEQ_MATCH \d{3}-?\d{2}-?\d{4} FILL 0x44

# Scrub EATME.TXT
FILE_SEQ_EQUAL pineapple-upside-down-cake SCRUB

# Scrub Alice in Wonderland
FILE_DIRNAME_EQUAL looking-glass SCRUB

# Ignore EATME.TXT.BACKUP
IGNORE *.BACKUP

# Commit the redaction (write out a redacted disk image)
COMMIT
```

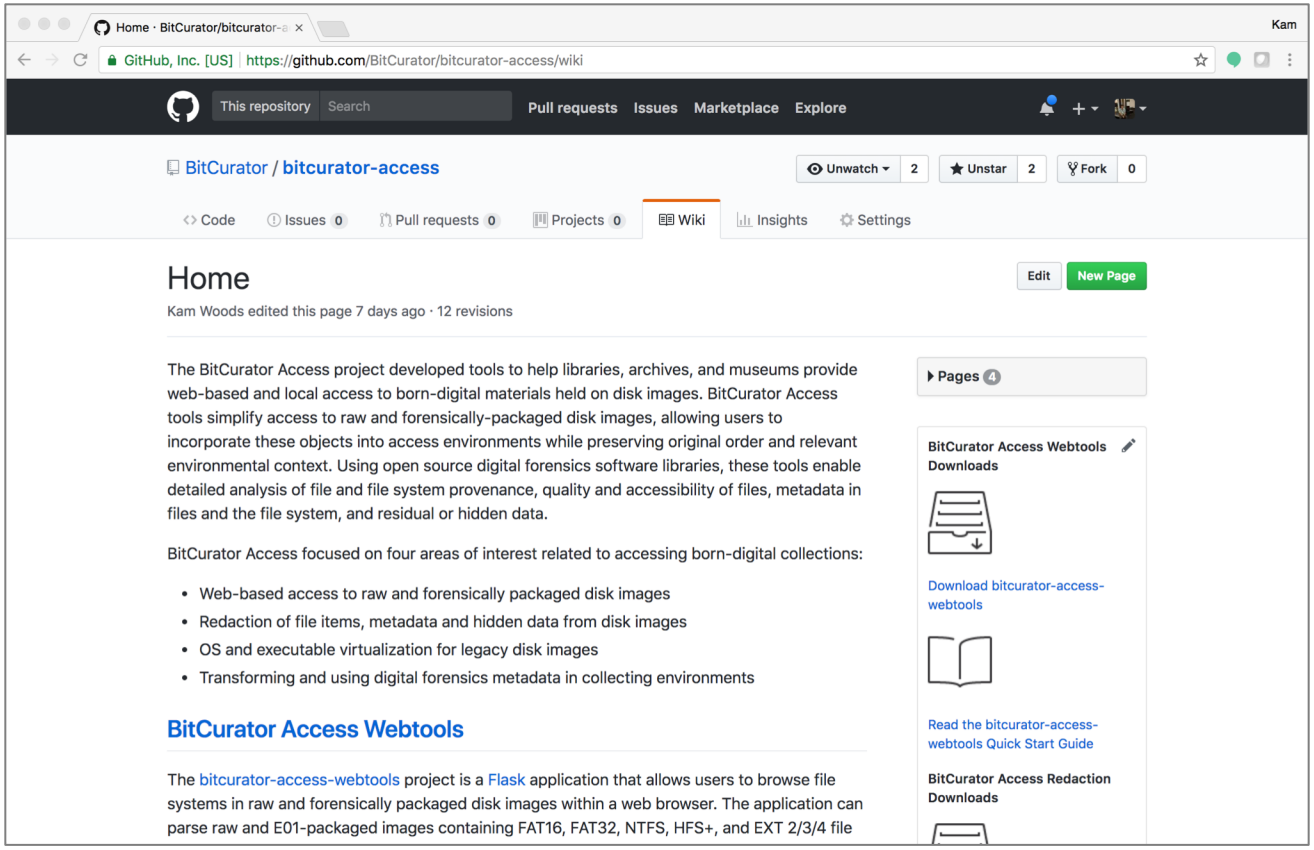
Running the redaction

- In the terminal, type the following:

  $ redact-cli -c ~/Desktop/test_image_config.txt

- A new file, **test_image_redacted.raw**, will be written to the Desktop. Mount this image by right-clicking and selecting **Scripts->Mount Disk Image** to examine the files and directories targeted by the redaction rules in our configuration file.

Further Information



More detailed information can be found on the project wiki at https://github.com/BitCurator/bitcurator-access/wiki.

Source code and releases can be found on GitHub at https://github.com/bitcurator/bitcurator-access-redaction.