SectorScope

USERS MANUAL

August 19, 2016

Authored by:
Bruce D. Allen

Contents

1 Introduction

1.1 Overview

SectorScope is a graphical interface tool for viewing blocks that match blacklist block hashes stored in a *hashdb* database. *SectorScope* includes interfaces for scanning media images, ingesting source files into *hashdb* databases, and directly viewing raw media image bytes.

1.2 Obtaining *SectorScope*

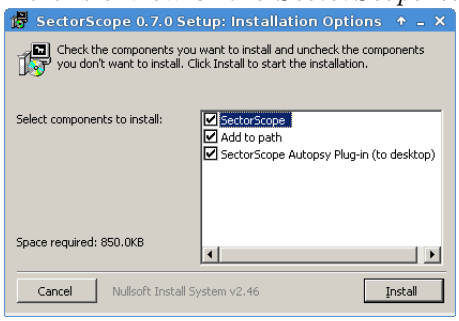
SectorScope requires *hashdb*. Both tools are readily available for Windows systems, Linux flavors, and MacOS. Windows installers are available for Windows users. Source code distributions are available. Developers may download these tools directly from source available on GitHub.

1.2.1 Installing on Windows

Download and run the latest *SectorScope* Windows installer available under `http://digitalcorpora.org/downloads/sectorscope/`. It is named `SectorScope-x.y.z-windowsinstaller.exe` where x.y.z is the latest version.

The Windows installer includes an optional *SectorScope Autopsy* `.nbm` module for running *SectorScope* from *Autopsy*. Please see **Subsubsection 1.2.4** and **Subsubsection 1.2.5** for information on configuring and installing this optional *SectorScope* module.

Here is a view of the *SectorScope* Windows installer:



1.2.2 Installing on Linux or Mac

- Download *SectorScope* files from the `.zip` file under `http://digitalcorpora.org/downloads/sectorscope/`.
- Unzip the downloaded `.zip` file to extract *SectorScope*.
- Copy these unzipped files to a directory where you can run them or set your `PATH` variable to find them. For example create directory `local/bin` in your home directory and copy the files there. Then add the following text to your `$HOME/.bash`

file and then close and reopen your command window so that these tools can be found:

```
# User specific aliases and functions
PATH=$HOME/local/bin:$PATH
```

1.2.3 Installing Other Resources

SectorScope requires other resources to run:

- **hashdb**
  SectorScope uses *hashdb* to run scans, ingest block hashes, and read media image bytes.

  For Windows: download and run the latest *hashdb* Windows installer available under `http://digitalcorpora.org/downloads/hashdb/`. It is named `hashdb-x.y.z-windowsinstallelr.exe` where x.y.z is the latest version.

  For Linux and Mac: download and build *hashdb* as described in the *hashdb* Users Manual available at `http://digitalcorpora.org/downloads/hashdb/` or on the *hashdb* Wiki at `https://github.com/NPS-DEEP/hashdb/wiki/Installing-hashdb`.

- **Python**
  SectorScope requires Python2.7 or Python3. Please see `https://www.python.org/downloads/` to install Python.

- **TSK**
  Although TSK is required for generating media image annotations, *SectorScope* can run without it. To support media image annotations, please install TSK executables and libraries from `http://www.sleuthkit.org/sleuthkit/download.php` and set your `PATH` variable to include the installed `bin` directory.

- **Autopsy**
  If you would like to run *SectorScope* from *Autopsy*, please also install *Autopsy*, `http://www.sleuthkit.org/autopsy`. Note that the *SectorScope Autopsy* plug-in module must also be installed and configured, described below.

1.2.4 Installing the *SectorScope Autopsy* Plug-in Module

The *SectorScope* Windows installer installs the `.nbm` *SectorScope Autopsy* plug-in module onto the desktop. Please follow these steps to install this module into the *Autopsy* workflow:

1. Open *Autopsy*. From the *Autopsy* menu, select `Tools | Plugins`.
2. Open the `Downloaded` tab and click the `Add Plugins...` button.
3. From the `Add Plugins` window, navigate to the `.nbm` module file that was installed onto the desktop, and open it. It may be at `C:\Users\Public\Public Desktop`.
4. Click `Install` and follow the wizard.

1.2.5 Configuring the *SectorScope Autopsy* Plug-in Module

The path to the *hashdb* database must be configured:

1. Start a new case, `File | New Case...`, fill in the Case Information fields, and click Next.
2. Fill in Case Information and click Finish.
3. For `Add Data Source` (1 of 3), put in a media image for *Autopsy* to process and click `Next`.
4. For `Add Data Source` (2 of 3), select checkboxes as desired, then click on `SectorScope` text to configure the path to your *hashdb* database to scan against. Currently a file chooser is not available, so please type in the full path, for example: `C:\Users\me\my_hashdb.hdb`. Click `Next`.
5. For `Add Data Source` (3 of 3) click `Finish`. When the *SectorScope* module begins processing, *Autopsy* will display "NPS-SectorScope ..." as *hashdb* runs, which may take up to several hours. Unfortunately, *hashdb* progress is not currently indicated. For diagnostics: please see text in the generated log file and in the generated `stderr_hashdb.txt` file and try running the scan directly from *SectorScope*.

1.3 Starting *SectorScope*

To open *SectorScope*, type the following on the command line:

- `sectorscope.py`

If desired, *SectorScope* may be opened with alternate parameters so that it starts with a scan dataset. Type `sectorscope.py -h` for help on options.

SectorScope runs from *Autopsy*. To start *SectorScope* from *Autopsy*, click the `SectorScope` property in the *Autopsy* window.

2 Working with *SectorScope*

2.1 Managing Massive Datasets

Block hash scans can produce a large amount of matches. *SectorScope* provides filters to highlight and ignore data:

- **Highlight Sectors and Hashes**
  Entire sectors or individual hashes may be highlighted.
- **Ignore Sectors and Hashes**
  Entire sectors or individual hashes may be ignored.
- **Ignore Entropy Range**
  All hashes from blocks below or above given entropy values may be ignored.
- **Ignore Maximum Duplicate Hashes**
  All hashes with a duplicates count above a maximum may be ignored.

- **Ignore Auto-filter Labeled Hashes**
  All hashes marked with a block label may be ignored.

SectorScope is able to provide filtering by using block and source information stored in the *hashdb* database. Specifically:

- **Block Entropy**
  Low entropy blocks are frequently nonprobative.
- **Block Label**
  hashdb generates block labels for blocks possessing specific characteristics. *hashdb* generates several block labels. Defining effective block labels is an area of research.
- **Source Size**
  The source size is used to calculate the percentage of a source that was matched.
- **Count**
  The count of matches for a block and the sub-count of matches contributed by each source for a block are used to indicate how common a block is.
- **Source Label**
  Source labels may be used to infer expected entropy values. *hashdb* and *SectorScope* do not use this information. This is an area of research.

2.2 Scan Files

Scan files contain the information about the blocks in a media image that match blacklist blocks in a block hash database. *SectorScope* contains the interfaces necessary for creating *hashdb* databases, creating scan files by scanning media images, and viewing match information contained in scan files.

2.3 Histogram Graph

The histogram graph shows a histogram of frequencies of matched hashes along a scanned media image. Controls in the user interface allow the user to pan and zoom the graph, highlight or ignore regions of the histogram, correlate matches with source files, and export sectors of the media image.

2.4 Media Image Annotations

SectorScope displays media annotations below the histogam bars. The following annotation types are currently supported:

- Disk partition information obtained by running the TSK `mmls` command.
- File system information obtained by running the TSK `fsstat` command.

Annotation entries define the annotation type, the offset and length of the content being annotated, and annotation text. New annotation types may be added in the future.

SectorScope has the ability to work with sector sizes other than 512 bytes. The TSK tools that *SectorScope* use to produce annotation expect a sector size of 512 bytes. To prevent reporting incorrect units, *SectorScope* will not open annotations when the sector

size specified when opening a scan is not 512.

The first time a scan file is opened by *SectorScope*, *SectorScope* prepares annotation content in a subdirectory next to the scan file named `<scan file>.temp_annotations`. For example if the scan file is named `scanfile.json`, annotations will be generated at `scanfile.json.temp_annotations`. *SectorScope* references information in this subdirectory in order to display annotation content. This subdirectory may be deleted. *SectorScope* will generate it again if it is not there.

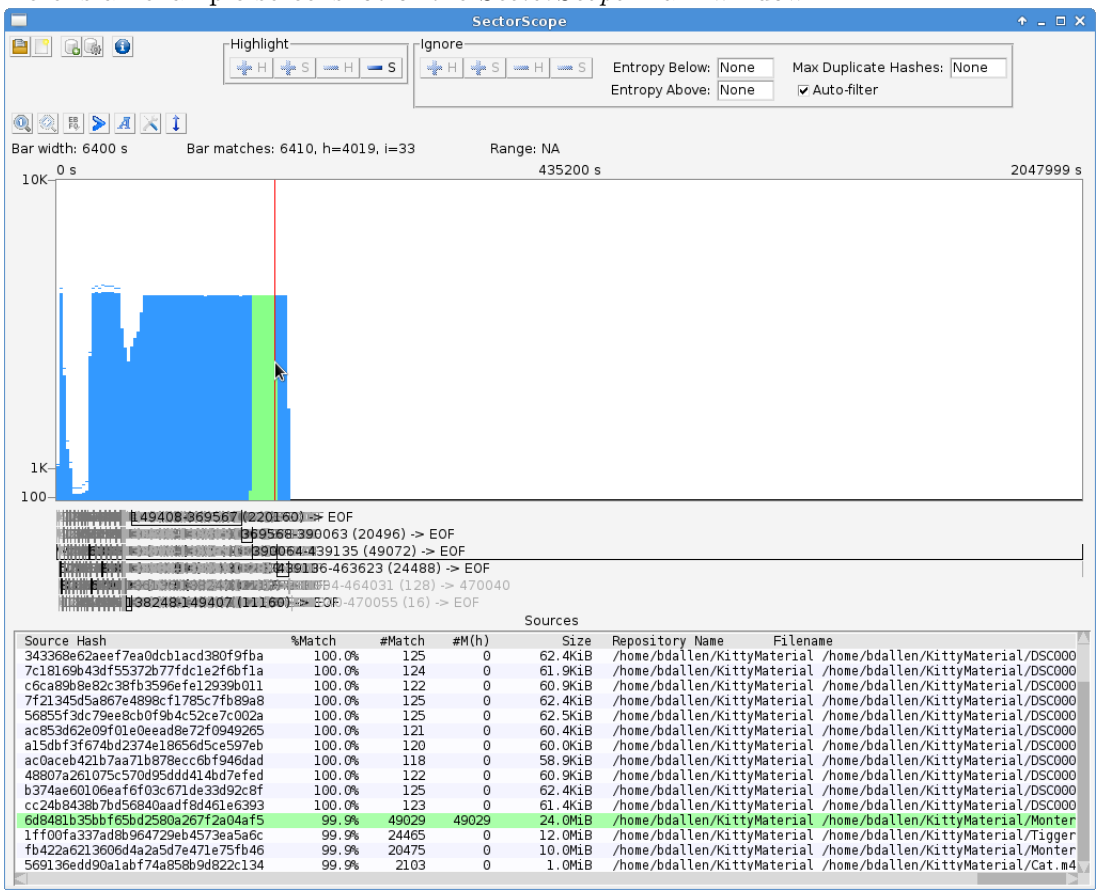
2.5 Source Files

SectorScope shows a list of all sources in a hash database that have hashes in common with the media image scan. Changing filter and range selections affect the source list by removing sources or modifying how much of a source is matched.

3 *SectorScope* User Interfaces

3.1 Main Window

Here is an example screenshot of the *SectorScope* main window:

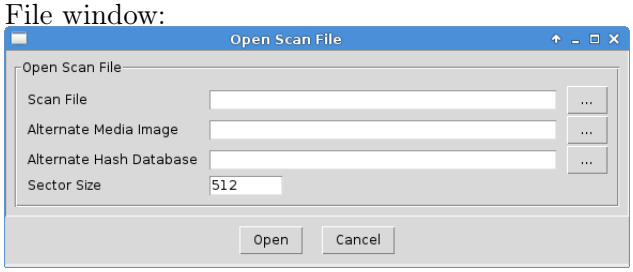


3.2 Menu Controls

Here are the menu controls:

3.2.1 Open Scan File

Use this control to open and view the output of a *hashdb* scan. Here is the Open Scan File window:



- **Scan File**
  When *hashdb* performs a scan, all matches are placed in this scan file. The scan file also contains the path to the media image scanned and the hash database scanned against. If these paths move, alternate paths may be provided.

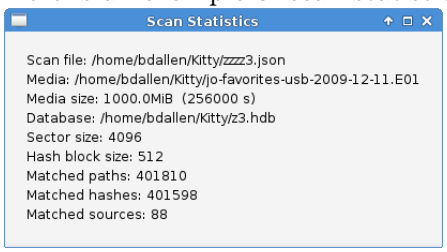
- **Alternate Media Image**
  An alternate path to the media image to use if the path in the scan file is incorrect. Keep blank to use the default.

- **Alternate Hash Database**
  An alternate path to the hash database to use if the path in the scan file is incorrect. Keep blank to use the default.

- **Sector Size**
  The sector size for this media, typically 512 bytes. *SectorScope* will display sectors of this size. *SectorScope* will zoom in to this size.

3.2.2 Scan Statistics

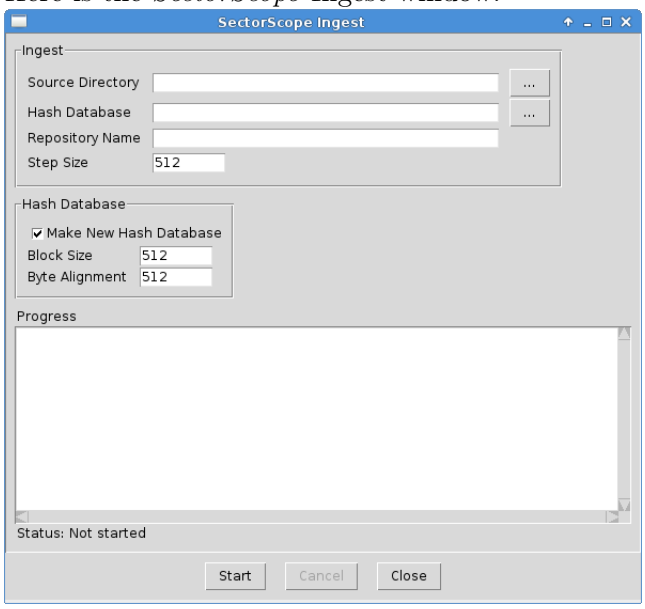
Here is an example of scan statistics:



Statistics for the opened scan include the path to the scan file, media image, and hash database, the image size, sector size, hash block size, and number of matched paths, hashes, and sources.

3.2.3 Ingest

Here is the *SectorScope* Ingest window:

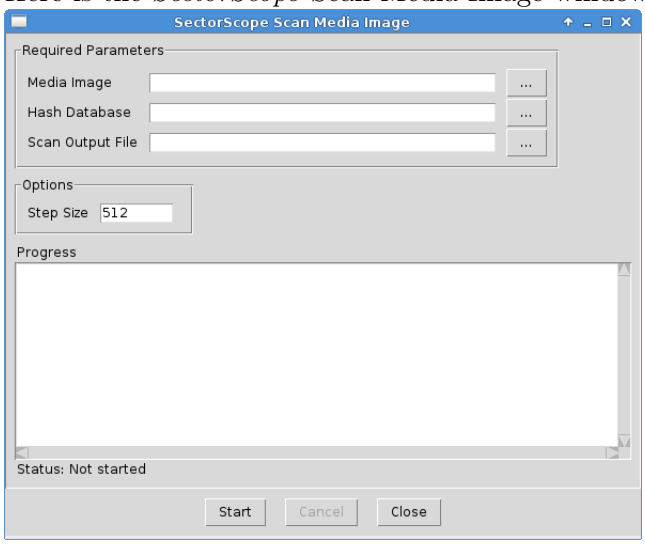


Use this window to ingest files or media images into a hash database.

- **Source Directory**
  The path to the source files to ingest recursively from.
- **Hash Database**
  The path to the hash database to import block hashes into.
- **Repository Name**
  The repository name to associate the imported sources with. Leave blank to use the default, which is the source directory path.
- **Step Size**
  The step size to move along while calculating block hashes. The byte alignment must be compatible with the step size, specifically, byte alignment must be divisible by step size.
- **Make New Hash Database**
  If this is selected, a new database will be created instead of using an existing databases. The new database will be created with the following:
  - **Block Size**
    The block size to use for calculating the block hash.
  - **Byte Alignment**
    An optimization parameter, typically the smaller of the block size and the step size.

3.2.4 Scan

Here is the *SectorScope* Scan Media Image window:



Use this window to scan a media image for matching block hashes.

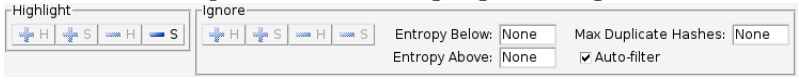
- **Media Image**
  The path to the media image to scan.
- **Hash Database**
  The path to the hash database to scan against.
- **Sector Size**
  The sector size for this media, typically 512 bytes. *SectorScope* will display sectors of this size. *SectorScope* will zoom in to this size.

3.2.5 Information

This button opens a window showing the version of *SectorScope* and *hashdb*.

3.3 Highlight and Ignore

Here is an example view of the highlight and ignore controls:



These inputs control which hashes and sources are highlighted or ignored in the histogram bar and in the source list. Highlighted items are shown in green. Ignored items are not displayed.

3.3.1 Highlight

- **+H**
  Select a range and highlight hashes in this range.
- **+S**
  Select a range and highlight all sources containing hashes in this range.
- **-H**
  Un-highlight all highlighted hashes.

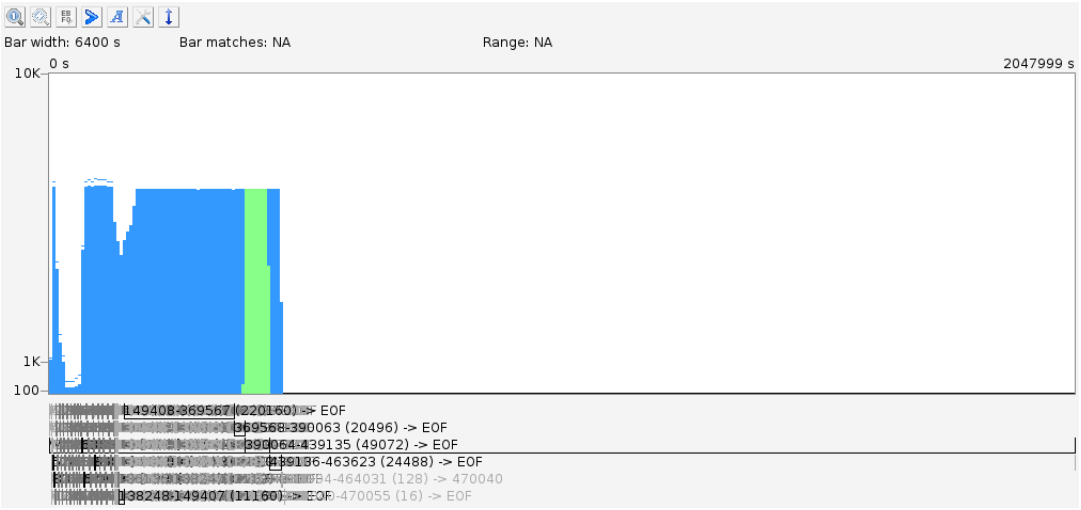
- **-S**
  Un-highlight all sources.

3.3.2 Ignore

- **+H**
  Select a range and ignore hashes in this range.
- **+S**
  Select a range and ignore all sources containing hashes in this range.
- **-H**
  Stop ignoring all ignored hashes.
- **-S**
  Stop ignoring all ignored sources.
- **Entropy Below**
  Ignore hashes with an entropy value below this threshold.
- **Entropy Above**
  Ignore hashes with an entropy value above this threshold.
- **Max Duplicate Hashes**
  Ignore hashes matched more than a maximum number of times.
- **Auto-filter**
  Ignore hashes if they have been flagged as potentially non-probative based on experimental entropy calculations.

3.4 Histogram and Media Image Annotation Graph

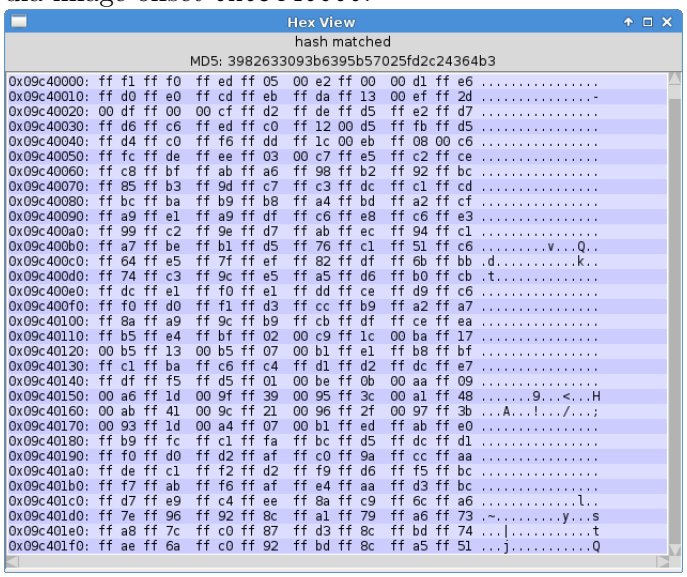
Histogram and media image annotations include button controls, cursor controls, graph annotations, and media image annotations:



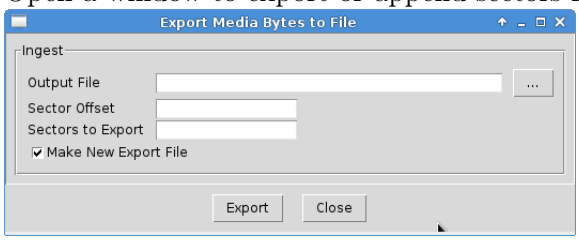
3.4.1 Graph Button Controls

Here are the histogram and annotation graph button controls:

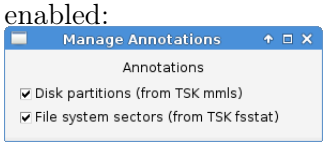
- **Zoom to Full Scale**
  Zoom out so the view spans the entire media iamge.
- **Zoom to Range Selection**
  Zoom in on the selected range to fit the view.
- **Show Hex View**
  Open a window to show the hexadecimal bytes of the block under the cursor. The following example hex view shows bytes for matched block hash `0x3982...` at media image offset `0x09c40000`.



- **Export Media Bytes**
  Open a window to export or append sectors from the media image into a file.



- **Manage Annotations**
  Open a window to enable or disable annotation categories. Here is an example window showing that annotations from disk partitions and file system sectors are enabled:



- Disk partition annotations are obtained by running the TSK mmls command.
- File system sectors are obtained by running the TSK fsstat command.

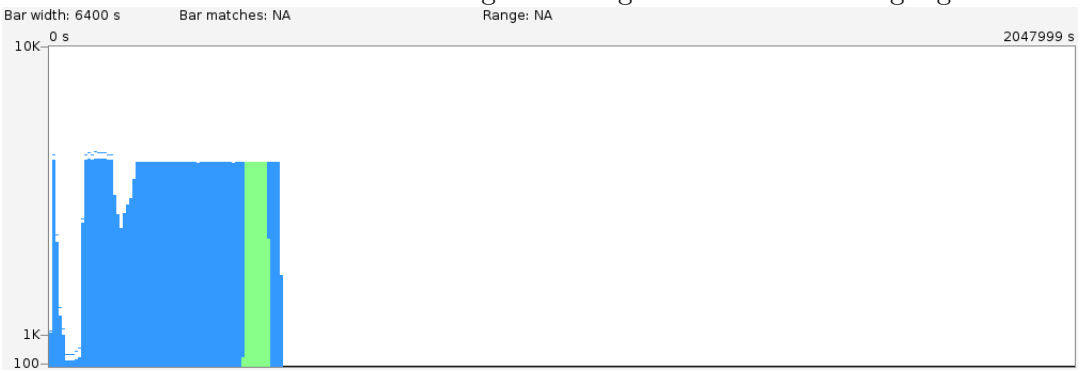
Please see **Subsection 2.4** for information about media image annotations.

- **Toggle offset Units**
  Toggle the displayed offset units between sectors, decimal bytes, and hexadecimal bytes.

- **Toggle Auto-y-axis**
  Enable or disable auto-y-axis histogram bar scaling.

3.4.2 Histogram

The histogram graph area shows the frequency of hash matches at given offsets in the media image. Bar width depends on the zoom level. Bar color indicates hashes present within the region of the bar. Blue indicates hashes present. A blue tick above the bar indicates total hashes when hashes are ignored. A green bar indicates highlighted hashes.



3.4.3 Histogram Bar Annotation

- **Bar width**
  The number of sectors or bytes spanned by each bar, depending on the offset format.

- **Bar matches**
  The number of matches under the bar that the cursor is hovering over. Three values are shown: total matches, highlighted matches (H), and ignored matches (I).

- **Range**
  The amount of the media image that the selected range spans.

- **Bounds**
  The first and last offset across the histogram is shown.

- **Y-axis scale**
  The horizontal scale, indicating the number of hash matches required to reach that height. This scale may be toggled between auto-scaling and no scaling.

3.4.4 Histogram Cursor Controls

Cursor movement, click, drag, and wheel motions manipulate the graph view:

- Move the mouse to position the cursor. Enable hex view to see the bytes under the cursor.
- Drag the mouse to select a range. Zoom to expand the graph to the selected range. Ignore or highlight hashes or sources in this selected range.
- Right-click drag to pan the graph.
- Roll the wheel to zoom in and out.

3.4.5 Media Image Annotation Graph

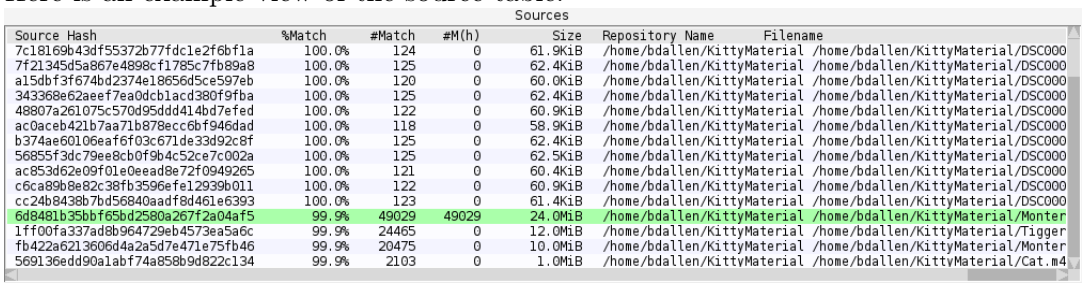
Here is an example view of a media image annotation graph:

Disk partitions and file system sectors are shown. They run over the top of each other because the histogram view is fully zoomed out and there are so many annotations present. To help preserve readability, only annotations that refer to a range that spans a whole histogram bar or more are shown in black, annotations that span less than one histogram bar in width are shown in gray, and annotations that span less than one tenth of a histogram bar are shown in light gray.

Please see **Subsection 2.4** for information about media image annotations.

3.5 Source Table

Here is an example view of the source table:



3.5.1 Source Table Contents

The source table shows information about sources in the hash database that have hashes in common with the media image scan. For each source shown, this information includes the source file hash, how much of the file matched, the file size, and one repository name, filename pair matching this file hash.

The set of sources and the percent matched depend on the range selection in the histogram view and filter settings:

- If a range is not selected in the histogram view, all matched sources are shown except for sources that are filtered as ignored and sources where all matched hashes are filtered as ignored..
- If a range is selected in the histogram view, only sources with hashes in this range may be shown.
- If all the hashes that match a source are ignored, the source will not be shown.
- If some of the hashes that match a source are ignored, the source will be shown, but the percent matched value will be lower.
- If all the hashes that match a source are ignored, the source will not be shown.

Rows in the source table contain the following information:

- **Source Hash**
  The hash hexcode of the source file.
- **%Match**
  The percent of this source that was matched by the scan.
- **#Match**
  The number of matches matched by the scan.
- **#M(h)**
  The number of matches that are highlighted.
- **Size**
  The size of the source file matched.
- **Repository Name**
  A repository name associated with this source file.
- **Filename**
  A filename associated with this source file.

3.5.2 Source Table Controls

- If a range is selected, only sources with hashes in that range are shown. Otherwise, all hashes are shown.
- If all hashes of a source are ignored, that source will not be shown.
- Click on a source to highlight it. It will show up green. Portions of the histogram bars contributed by that source will show up green. Re-click to un-highlight.
- Left-click on a source to ignore it. It will be removed from the sources table. Portions of the histogram bars contributed by that source will be removed. Ignored sources may be un-ignored using the **-S** Ignore control.

4 Examples

4.1 Create and Scan using *SectorScope*

In this example, we create a blacklist database of block hashes, then scan a media image for matches:

1. Identify a directory containing your blacklist source data. In this example, we use the Kitty Material demo source files available at `http://digitalcorpora.org/corpora/scenarios/2009-m57-patents/KittyMaterial` and scan for matches in the demo media image available at `http://digitalcorpora.org/corpora/scenarios/2009-m57-patents/drives-redacted/jo-favorites-usb-2009-12-11.E01` which contains blacklist block hashes from the Kitty demo:

2. Start *SectorScope* by typing the following at a command prompt:

   - `sectorscope.py`

3. Click on the Ingest menu icon to open the *SectorScope* Ingest window.

4. Fill in the fields:

   - Set Source Directory to the directory containing your blacklist source data.
   - Set Hash Database to your new hash database.
   - Set the Repository Name to the name of this case dataset, or leave blank to use the source directory as the repository name.
   - Use the default step size to ingest along sector intervals.
   - Keep Make New Hash Database checked since you will not be ingesting block hashes into an existing database.
   - Use the default block size to calculate sector-sized block hashes.
   - Use the default byte alignment since the blocks are sector-aligned.

5. Click the `Start` button to begin the process of ingesting block hashes from files under the source directory. Progress information will be displayed. When done, status will indicate `Done`. Close the window.

6. Click on the scan media image icon to open the *SectorScope* Scan Media Image window.

7. Fill in the fields:

   - Set the path to the media image to be scanned.
   - Set the path to the newly created hash database.
   - Set the path to the scan output file that will be generated.
   - Use the default step size to scan along sector intervals.

8. Click the `Start` button to begin the process of scanning for matching block hashes. Progress information will be displayed. When done, status will indicate `Done`. Close the window.

9. Click on the Open scanned output icon to open the Open Scan File window which opens the scan data into *SectorScope*.

10. Fill in the fields:
    - Set the path to newly created scan file.
    - Leave the alternate media image field blank since the path to the media image defined in the scan file is correct.
    - Leave the alternate hash database field blank since the path to the hash database defined in the scan file is correct.
    - Use the default sector size to view sector offsets for 512-byte-sized sectors.
11. Click the `Open` button to load this scan dataset into *SectorScope*. It can take 30 seconds or more to load large scan datasets. *SectorScope* currently does not show progress during open. Close the window.
12. Manipulate *SectorScope* controls to examine matched data.

4.2 Scan using *Autopsy*

1. Obtain a populated hash database and a media image to scan. This example uses the database and media image described in the previous example.
2. Start *Autopsy* and select `Add Data Source` if this window is not already open.
3. In `Add Data Source` Step 1, select the path to media image `jo-favorites-usb-2009-12-11.E01` and click `Next`.
4. In `Add Data Source` Step 2, configure ingest modules, select the SectorScope ingest module and type in the full path to the `kitty_blacklist.hdb` block hash blacklist database (file chooser capability is not available yet). Deselect other modules as desired. Click `Next`. Autopsy will remember these settings.
5. In `Add Data Source` Step 3, click finish.
6. When processing completes, click on Reports in the tree on the left to show the Reports table. In the Reports table, click on the `Block Hash Blacklist` cell to open *SectorScope*.

5 Alternate Configurations

- **Alternate Hash Algorithm**
  SectorScope calculates MD5 hashes when showing block hash values in the hex view. If your use-case requires a hash algorithm other than MD5, please see source code file `NPS-SectorScope/python/media_hex_window.py` for instructions on changing this algorithm.