BitCurator Quick Start Guide v0.5.6

Last updated: December 13, 2013

UNC
SCHOOL OF INFORMATION
AND LIBRARY SCIENCE

MITH
MARYLAND INSTITUTE FOR
TECHNOLOGY IN THE HUMANITIES

What this Document Covers

- Learn how to run the BitCurator environment in a virtual machine (using VirtualBox)
- Export file system metadata from a disk image (using fiwalk)
- Locate and identify personally identifying information within digital materials (using Bulk Extractor)
- Recognize and understand the main data elements that are generated by many open source forensics tools (using DFXML)
- Generate summary reports of DFXML metadata that can be used to characterize the contents of disks (using BitCurator reporting tools)

Things You'll Need to Get Started

- A machine running a 64-bit version of Windows 7 (or newer), Mac OS 10.8 (or newer), or a 64-bit Linux variant
- 4GB of RAM (minimum; 8GB preferred).
- 10GB free hard disk space (minimum; 20GB preferred)
- The BitCurator VM or Live CD: http://wiki.bitcurator.net
- An up-to-date version of VirtualBox: https://www.virtualbox.org/wiki/Downloads
- The VirtualBox Extension Pack (to be installed on the host system – just download and double-click on the file once you've installed VirtualBox)

Unpacking the BitCurator Virtual Machine

- The BitCurator Virtual Machine is packaged as a tar archive and compressed with gzip. The file will look something like: "BitCurator-0.5.0.tar.gz"

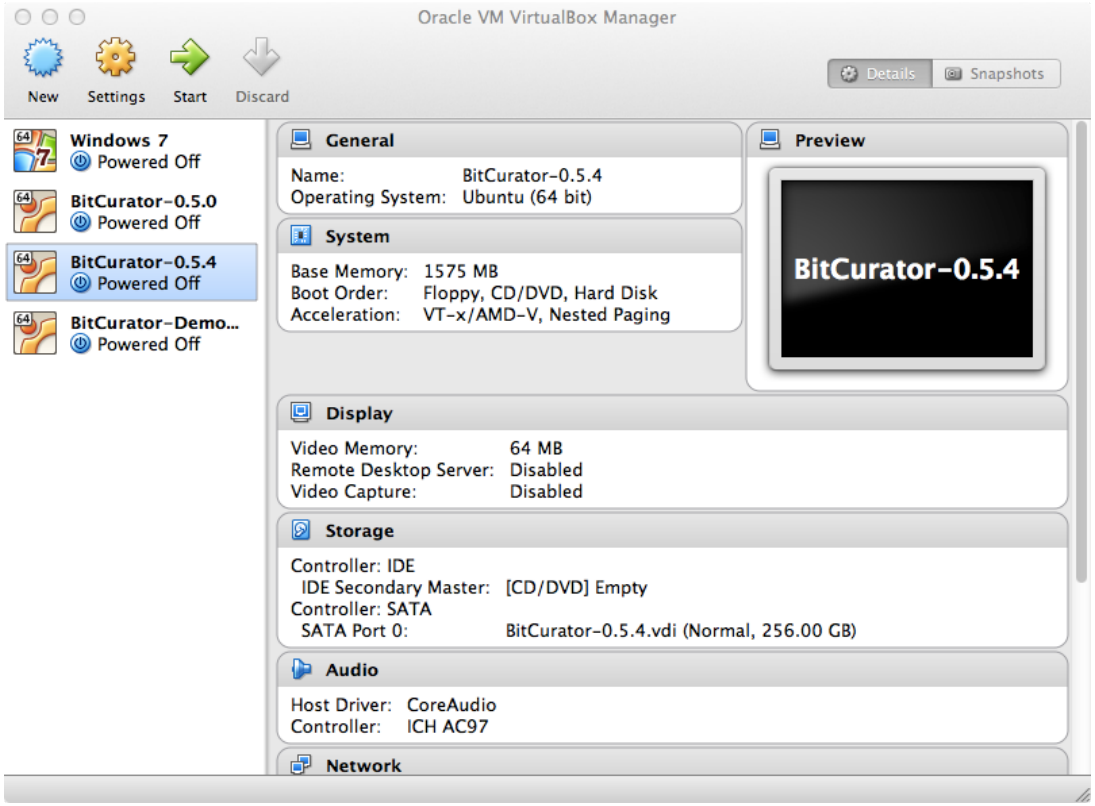
- On a Mac or Linux machine, you can simply double-click on the file to unpack the contents. On a Windows 7 machine, you'll need a 3rd party utility such as 7zip: http://www.7-zip.org/download.html

- When using 7-zip, you'll need to unpack the .tar.gz file (right click, and select "Extract here…"), and then unpack the resulting .tar file (right click, and select "Extract here…"). This will extract a directory containing the BitCurator .vdi and .vbox files.

The BitCurator Virtual Machine Files

- Once you've unpacked the archive, you'll find a directory containing two files (versions may differ from those shown here):
  - BitCurator-0.5.4.vbox (the VirtualBox configuration file)
  - BitCurator-0.5.4.vdi (the VirtualBox disk image)
- Copy this directory to a location of your choosing (inside the "VirtualBox VMs" directory in your home directory is a good place), and start up VirtualBox.
- Note: If you've never created or used a VM in VirtualBox before, you won't have a "VirtualBox VMs" directory. Don't worry – just remember where you extracted the BitCurator directory.

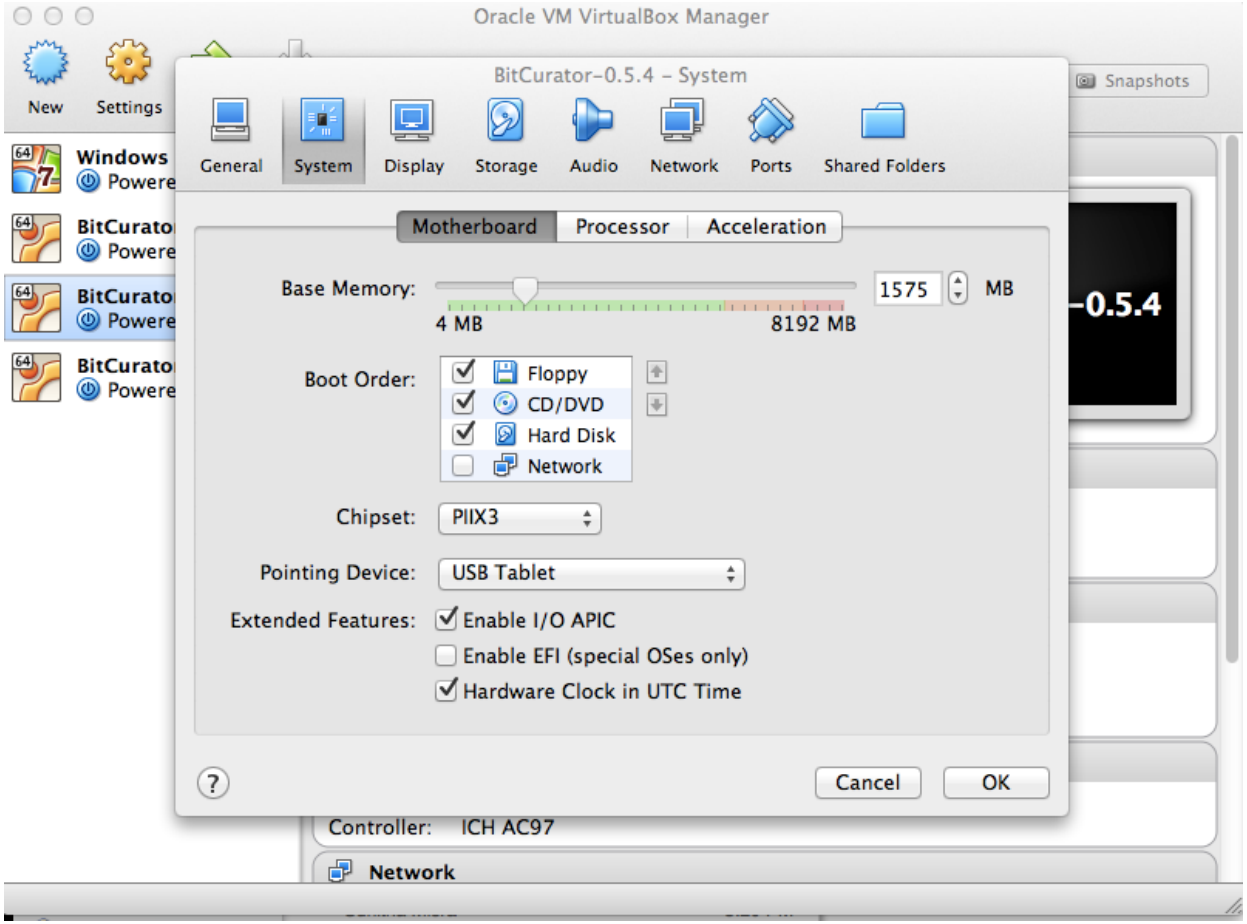
The Oracle VM VirtualBox Manager



Once you've installed VirtualBox and the VirtualBox extension pack, start up VirtualBox. **Note: Depending on how your Windows system is configured, you may need to right-click on the VirtualBox icon and select "Run as Administrator…"**

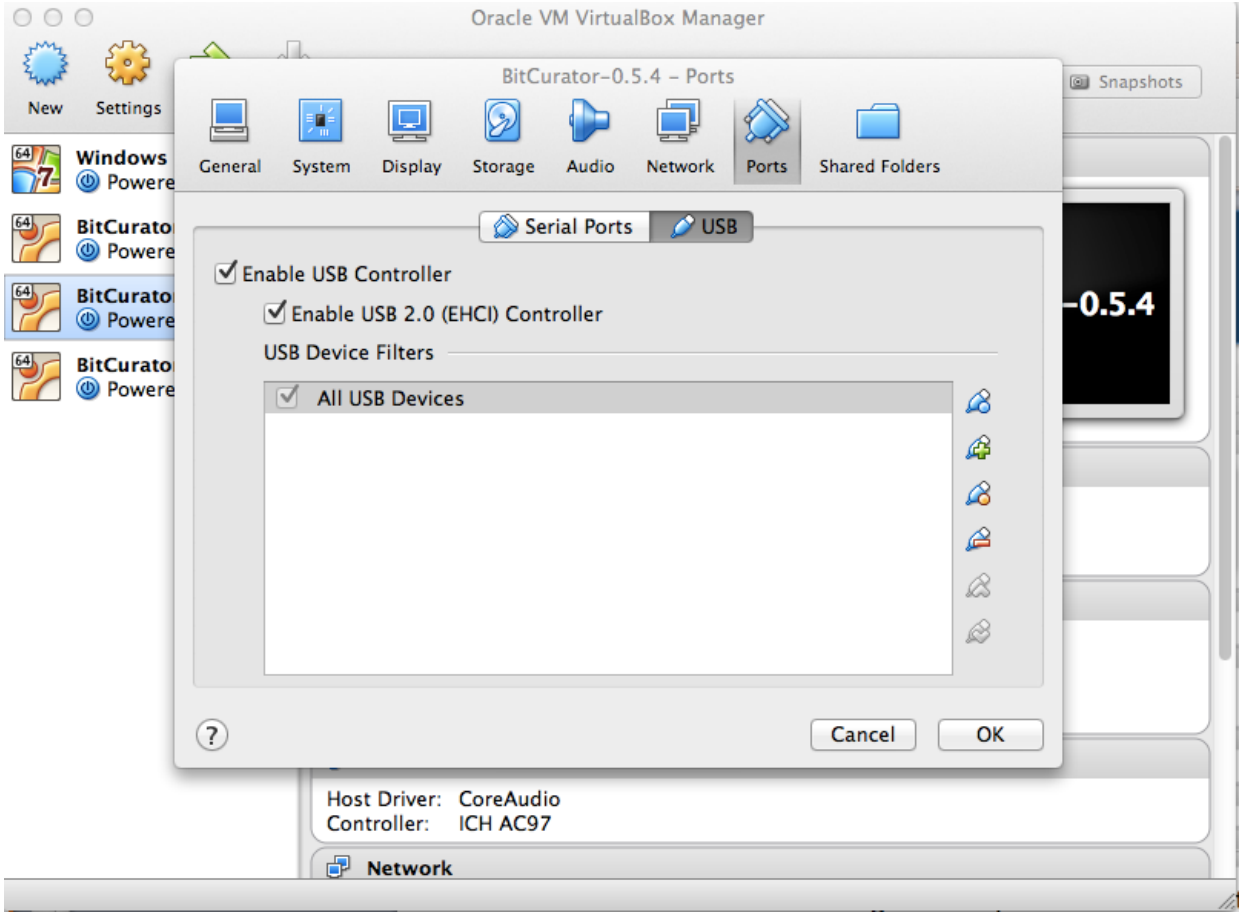
Now, select the menu item "Machine -> Add…", and navigate to the .vbox file that you extracted. The Virtual Machine should appear in the list within the manager, as shown above.

The Oracle VM VirtualBox Manager



Click on the Settings icon in the manager, and select the system tab. You may wish to increase the RAM and number of processors dedicated to the VM depending on the hardware that you're running on. For best results, select the largest number in the "green" areas for each.

The Oracle VM VirtualBox Manager



Click on the Ports icon in the Settings window, and select the USB tab. Note that there is a device filter in use that captures every USB device that is attached to the system while the VM is running. **Note: If you don't see it there, create one by clicking on the blue icon to the right. It doesn't matter what you name it.**

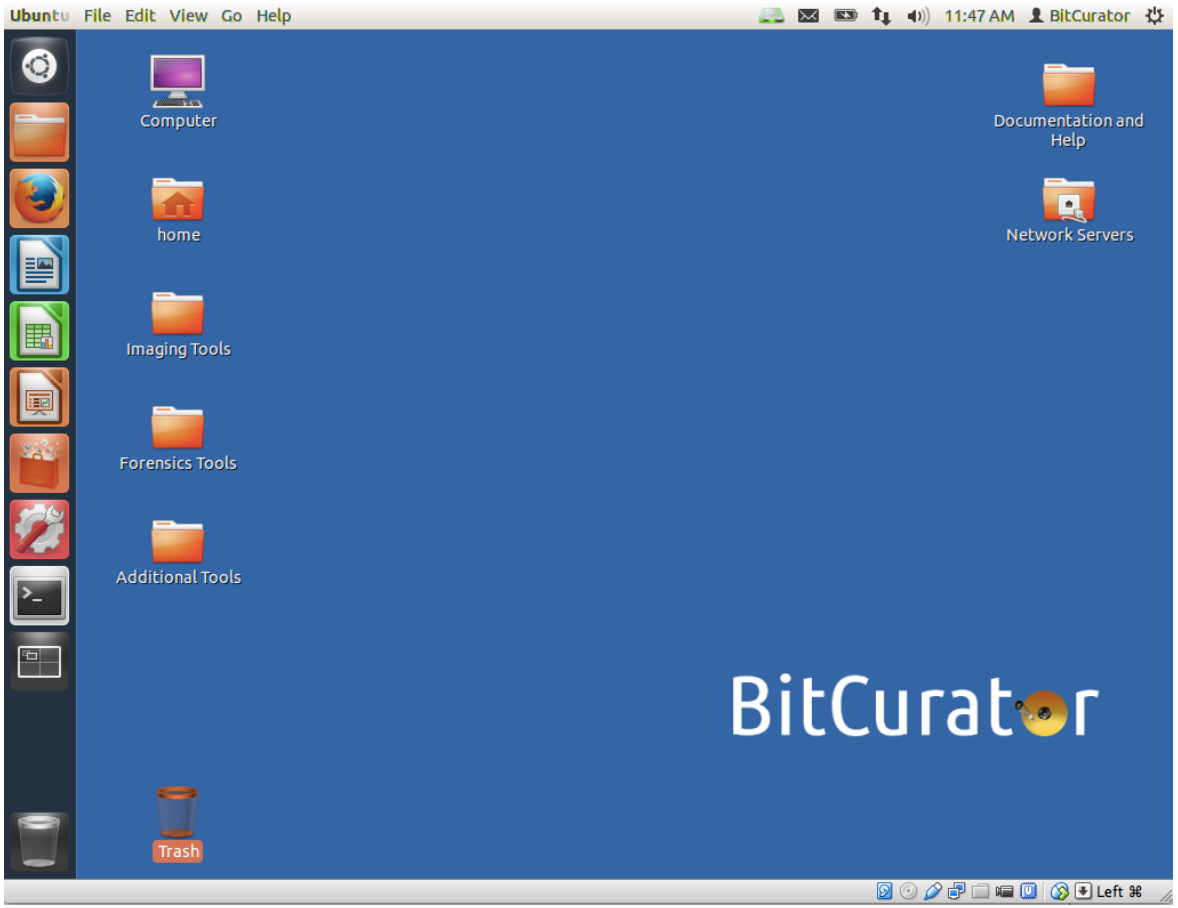
Starting the BitCurator Environment

Clicking on the green "Start" arrow in the Oracle VM VirtualBox Manager screen will start the BitCurator environment. You'll see a startup screen, and then the BitCurator environment will boot and automatically log in.

Note: If you see an error message mentioning virtualization extensions, or "Intel VT-x", your host machine's BIOS does not have the VT-x extensions enabled. You'll need to reboot your host, and hold down "Del" (or "Esc", or the "ThinkPad" button, depending on your machine) and enable the "Intel Virtualization Extensions" in the BIOS.

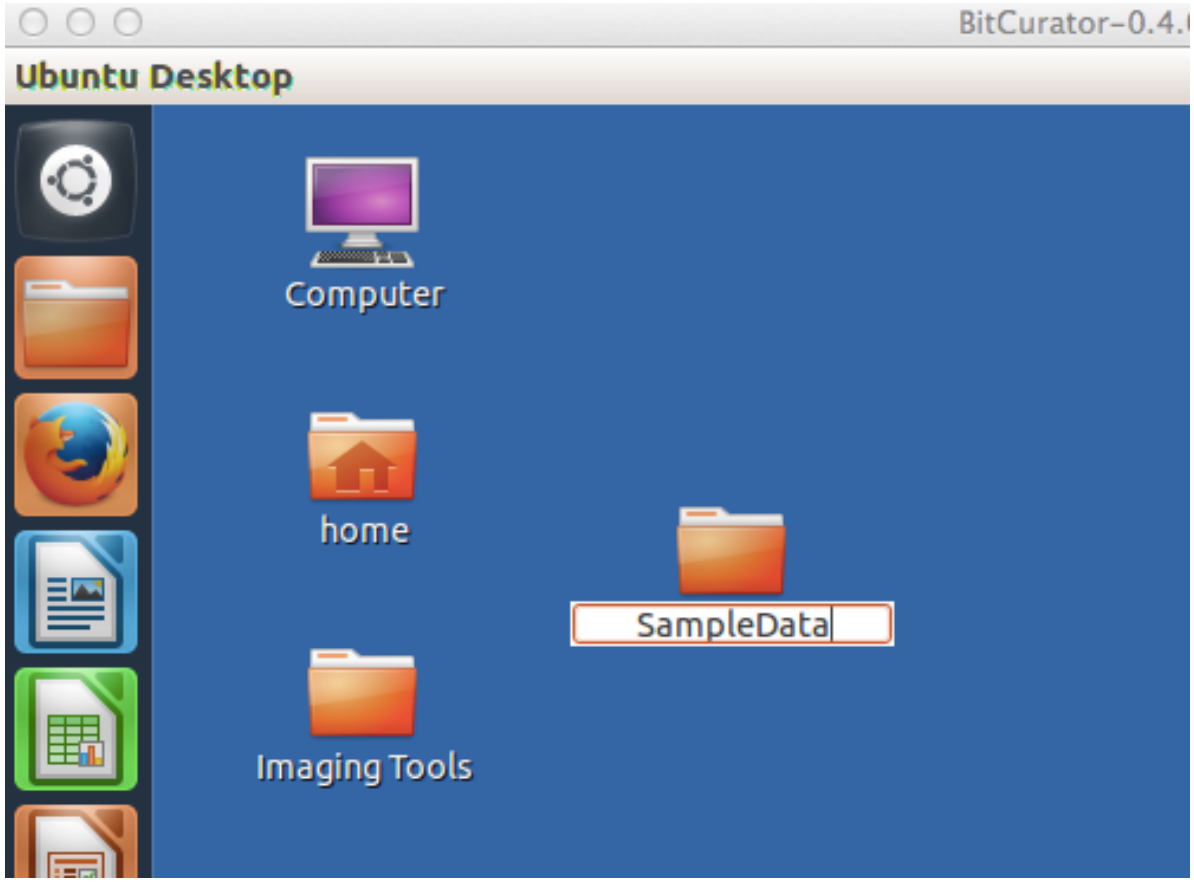
Note: If BitCurator fails to boot for other reasons, it may be due to a "non-optimal setting" detected for your particular hardware. Try powering off the virtual machine, checking your settings, and starting again. If you're still having a problem, let us know on the BitCurator users group (linked on our wiki at http://wiki.bitcurator.net/)

Starting the BitCurator Environment



The BitCurator virtual machine should log in automatically. If you log out or the machine goes to sleep, the password to log back in is “bcadmin”. You can also use this password to update installed software, if prompted.

Adding a Folder to Store Disk Images and Reports



Right-click anywhere on the desktop, and select “Create New Folder”. A folder named “Untitled Folder” will appear on the Desktop. Click on the name and rename it to “SampleData”. We’ll use this location to store the data for the rest of the exercise.

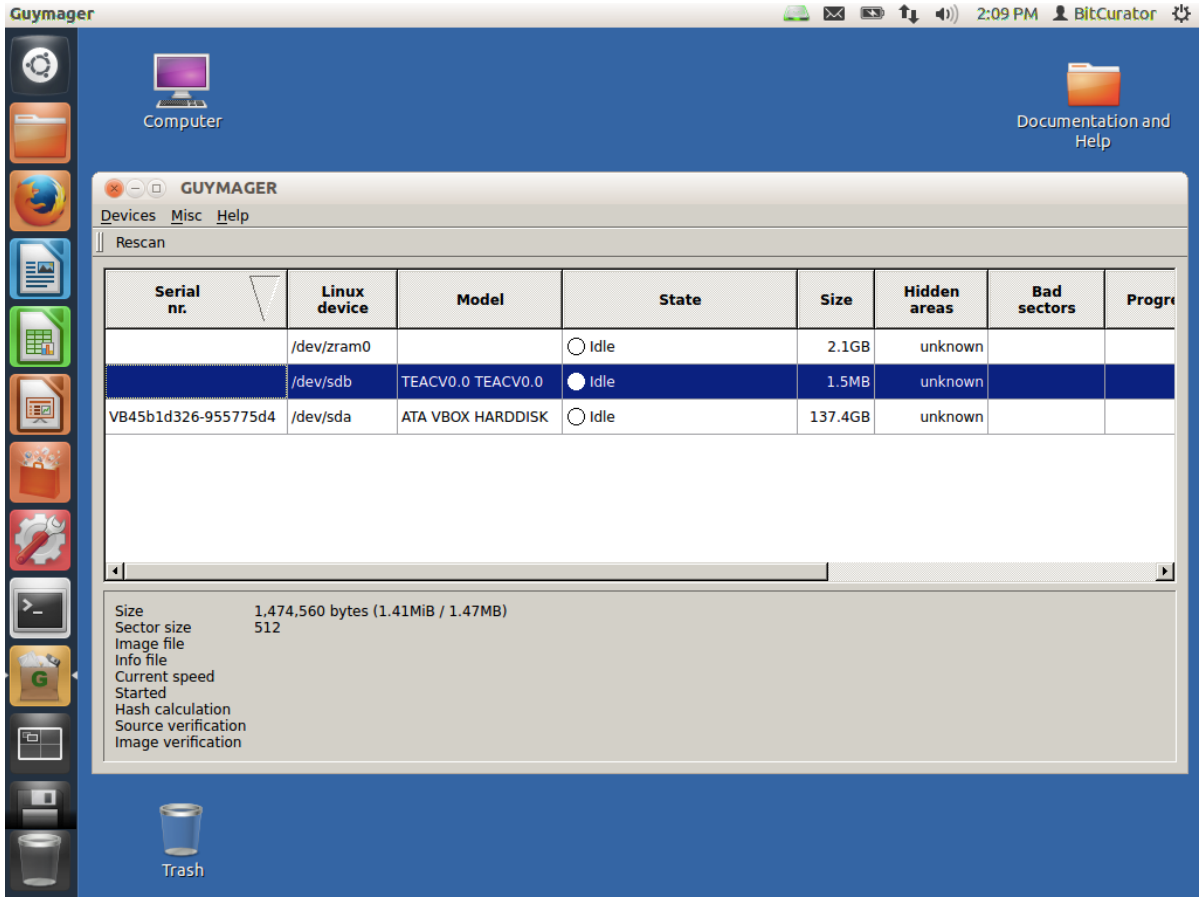
Getting Ready to Image Digital Media

BitCurator can be used to image a wide variety of digital media. For this example* we'll use an external USB floppy disk drive and a 3.5" FAT16 (DOS) formatted floppy disk.

Now that the VM is started up, the device will be automatically captured when plugged in.

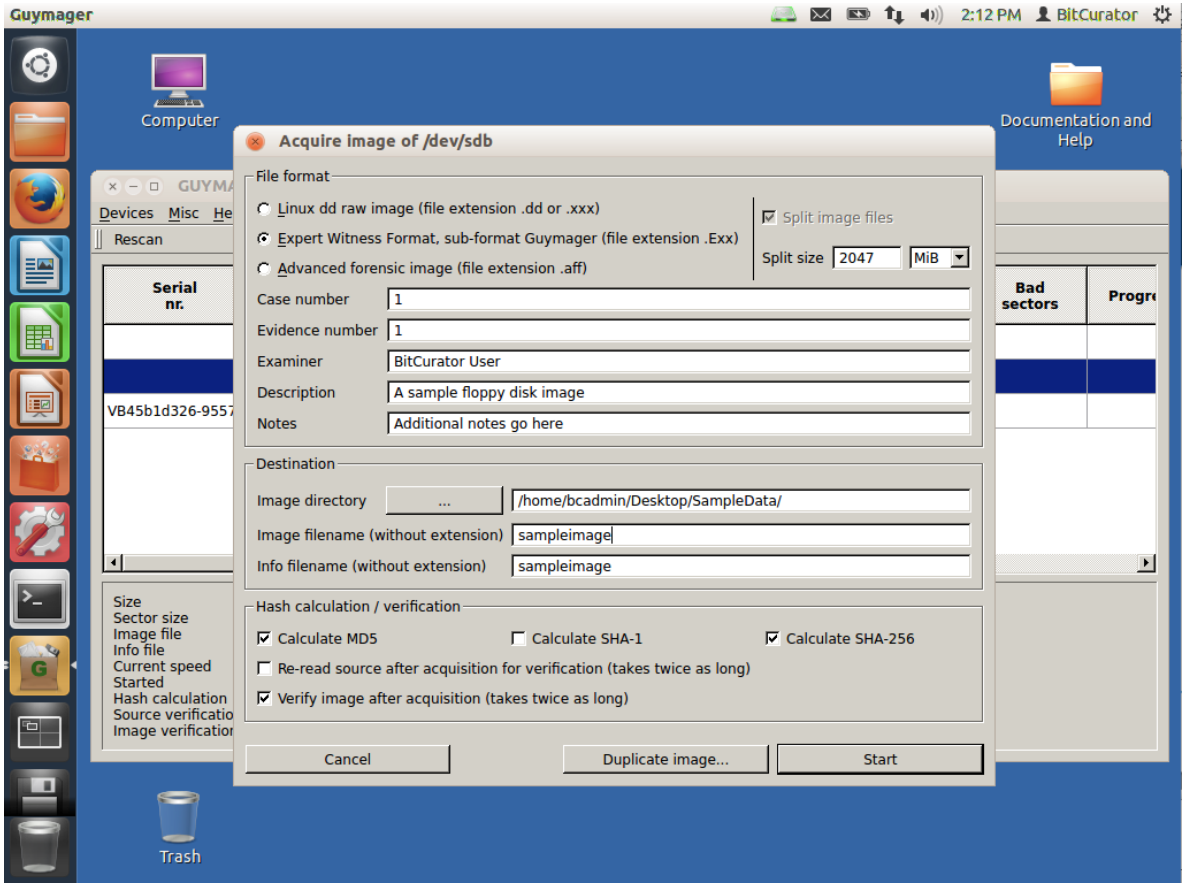
*The process from this point on will be largely the same whether we're working with data from a CD, a floppy, a hard disk, or any other media.

Imaging the Disk



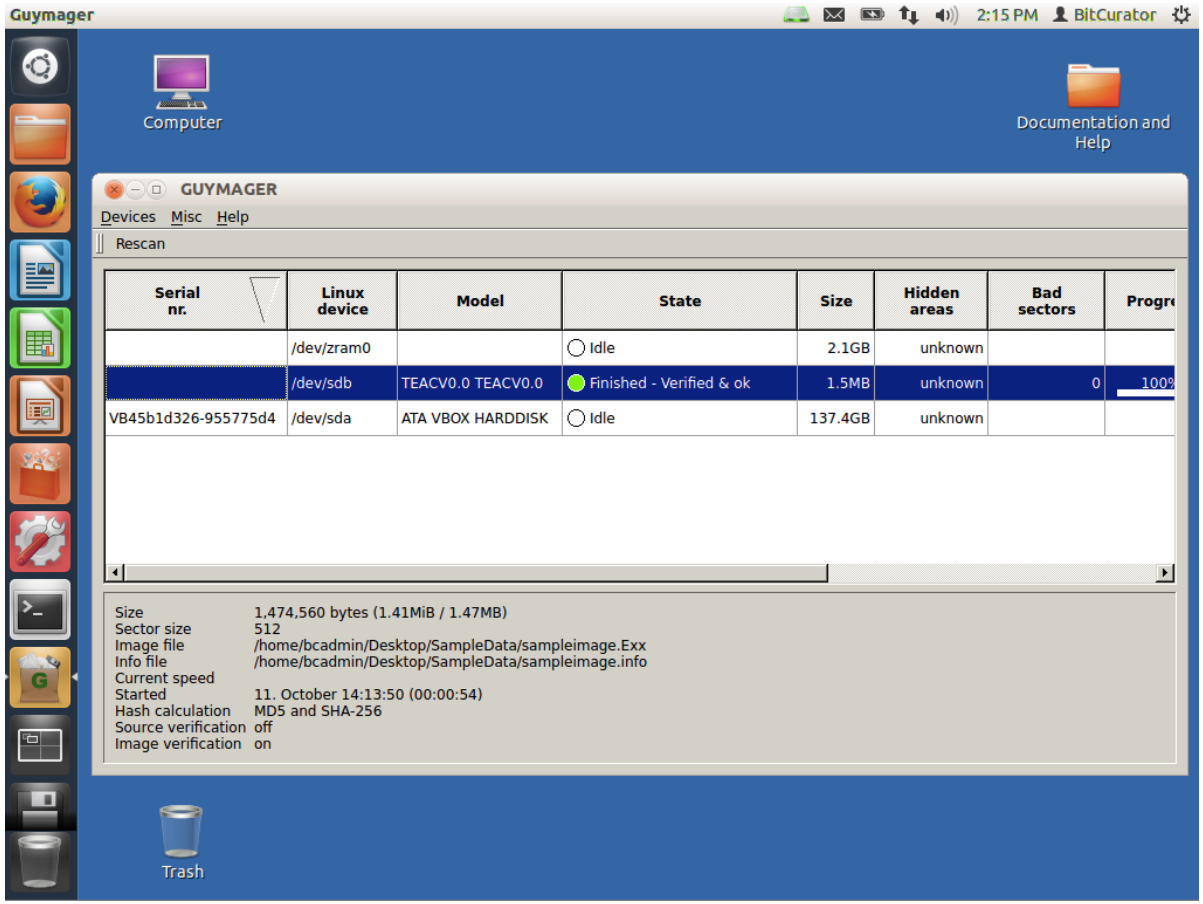
Once the drive is plugged in with the media inserted, a disk images should appear in the menubar on the right. The disk has not been mounted. This simply indicates that it has been recognized. Double-click on “Imaging Tools” on the Desktop, and then double-click on Guymager. The TEAC-brand floppy drive is selected in the picture above.

Entering Imaging Metadata



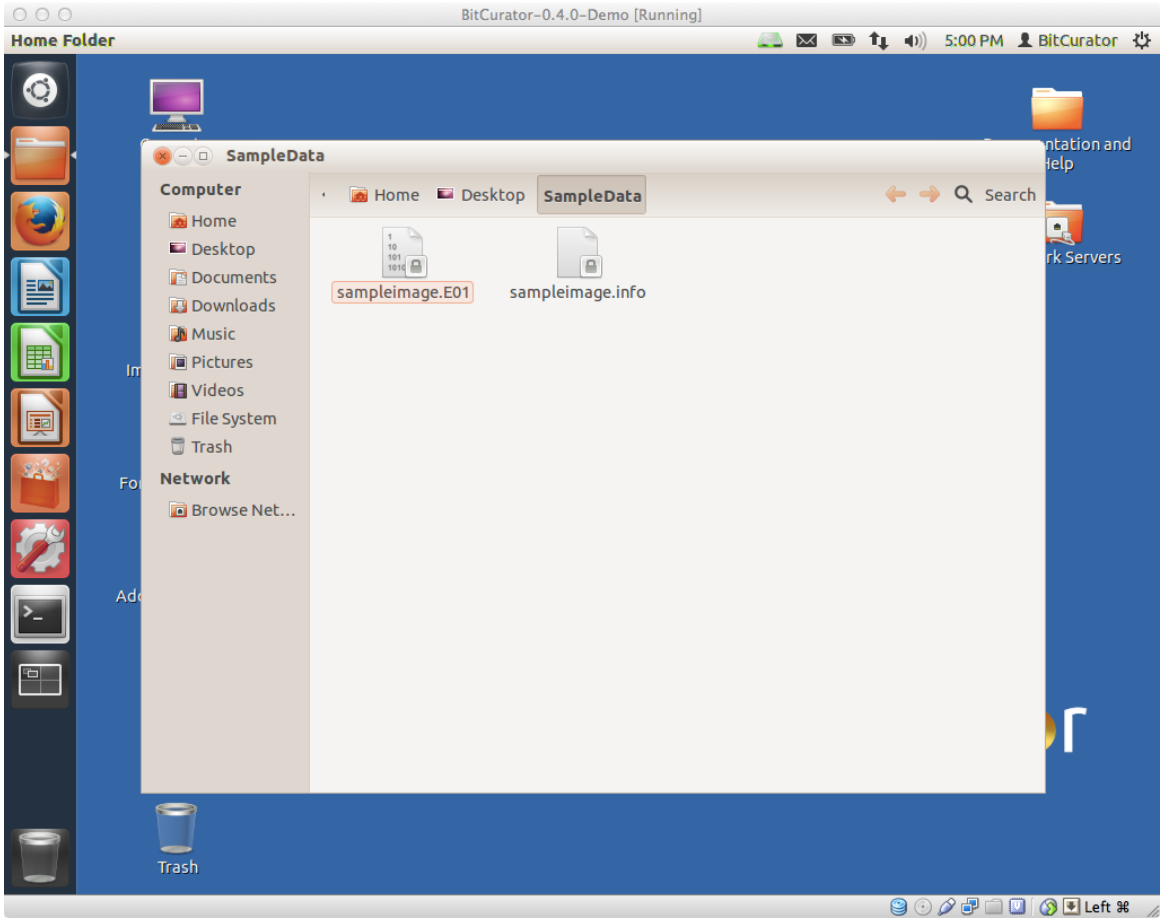
Right click on the device, and select “Acquire image”. We’ll capture the disk image in the Expert Witness Format (the second option at the top). The remaining metadata can be entered as desired or left blank. Don’t forget to select the directory we made on the desktop under “Image directory”, and name the image. Then click “OK”.

Running the Acquisition



You'll see the main dialog state change to "Acquisition Running". When the acquisition finishes, you'll see an "OK" message in State. **Note: The BitCurator environment runs at a resolution of 1024x768 by default. If you wish to see the whole dialog, just make the window bigger. The resolution should resize automatically.**

Examining the Image

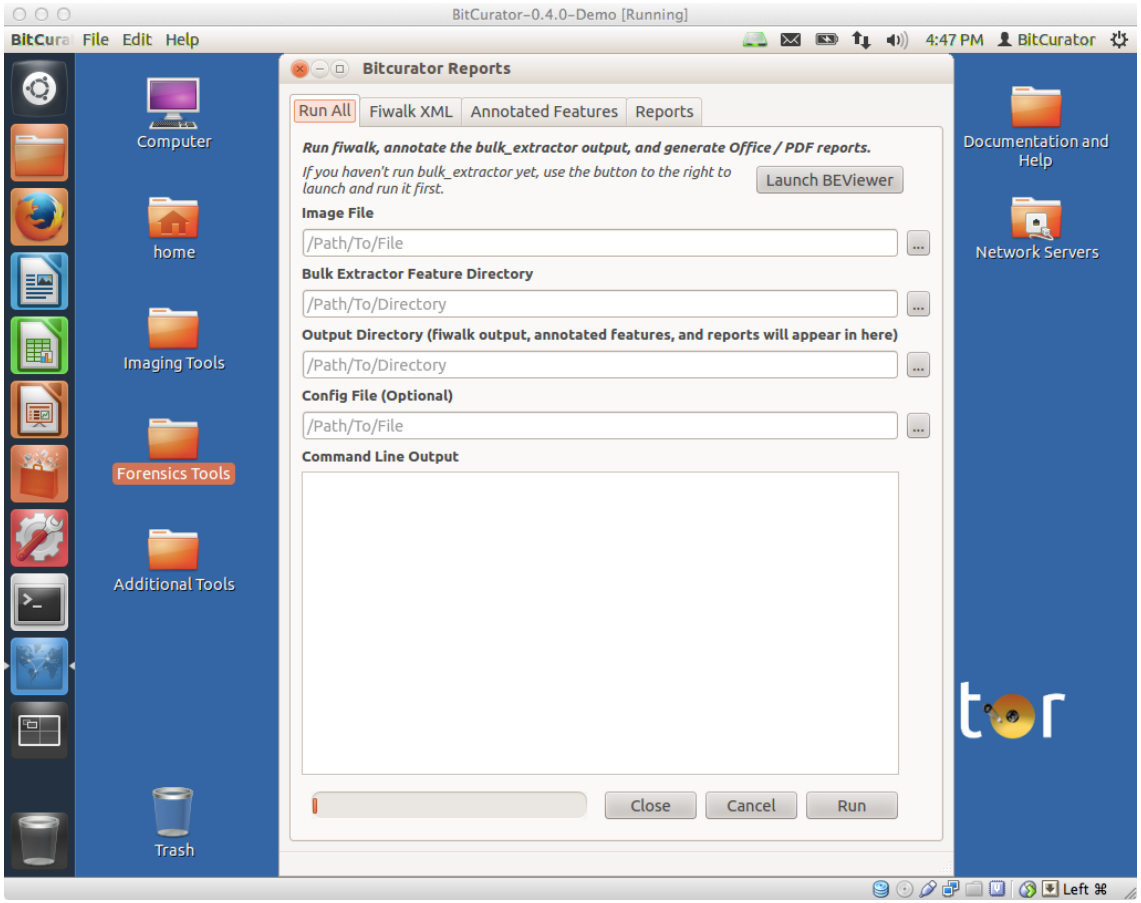


You can now exit Guymager, open up the SampleData directory we made on the desktop, and see the two files that have been produced: the .E01 image file, and a .info file specific to Guymager.

Safely removing a disk from the system

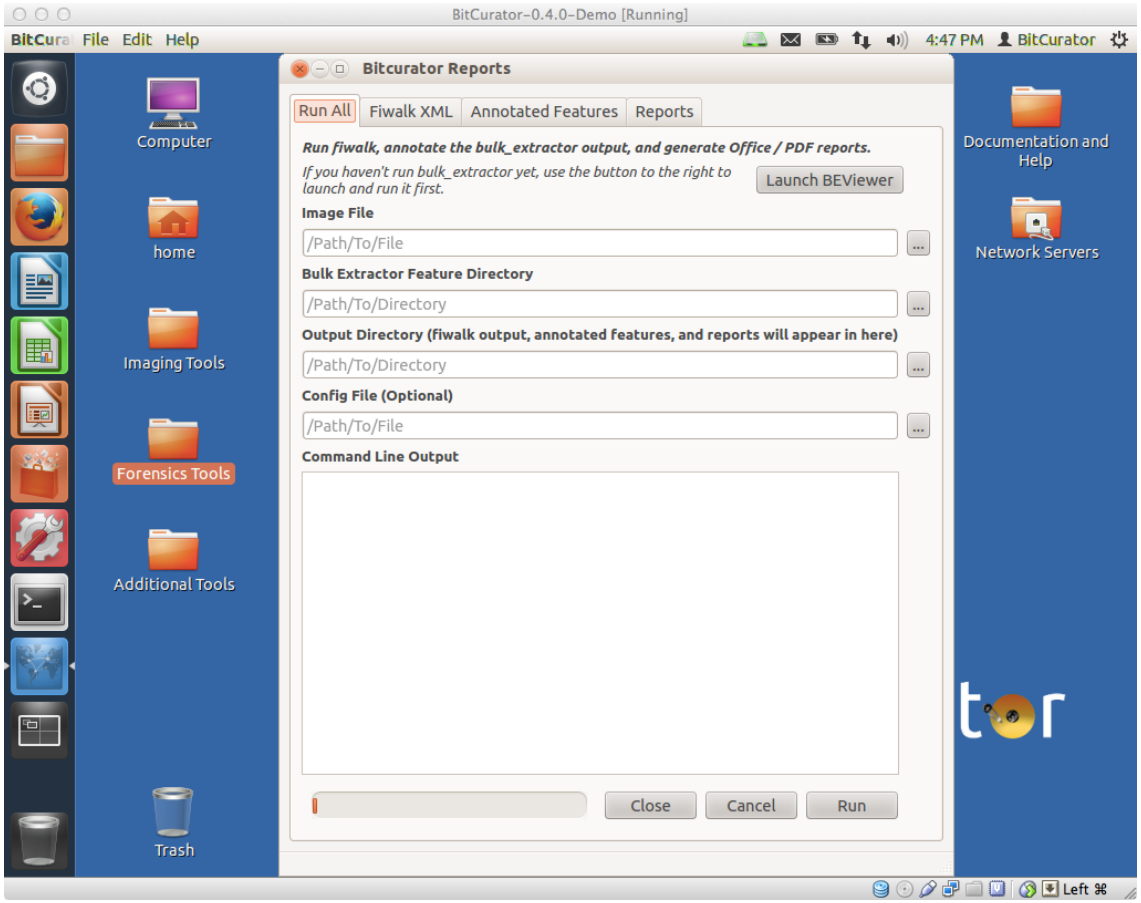
Now that the disk has been imaged, you can eject it from the system. Note that even though it's not mounted, you will still want to do this so the operating system knows it's no longer available. Right-click on the disk icon in the doc and click "Safely remove". You can now unplug your drive, or eject the disk.

Processing the file system, carving data, and generating reports



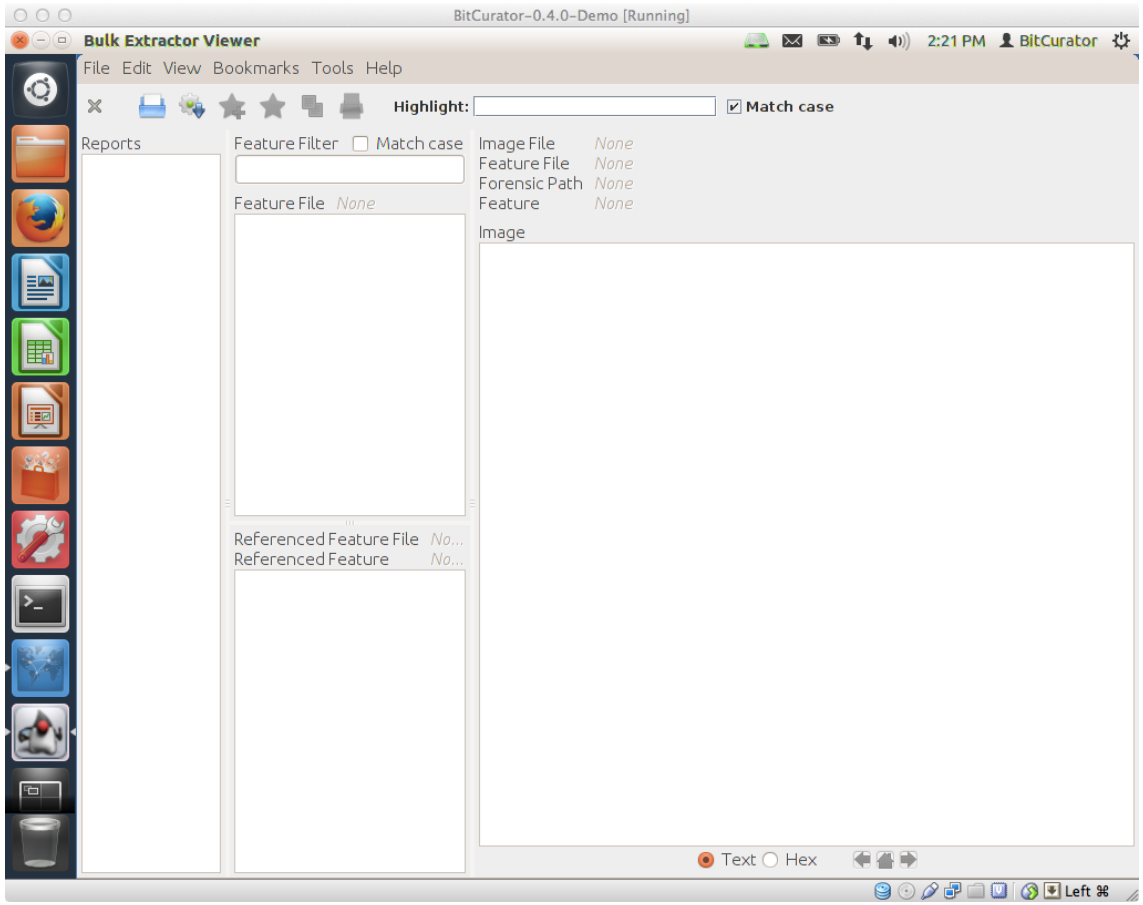
Double-click on the “Forensics Tools” folder, and then double click on the “BitCurator Reporting Tool” launcher. You’ll see a window pop up that should match the picture shown above.

Processing the file system, carving data, and generating reports



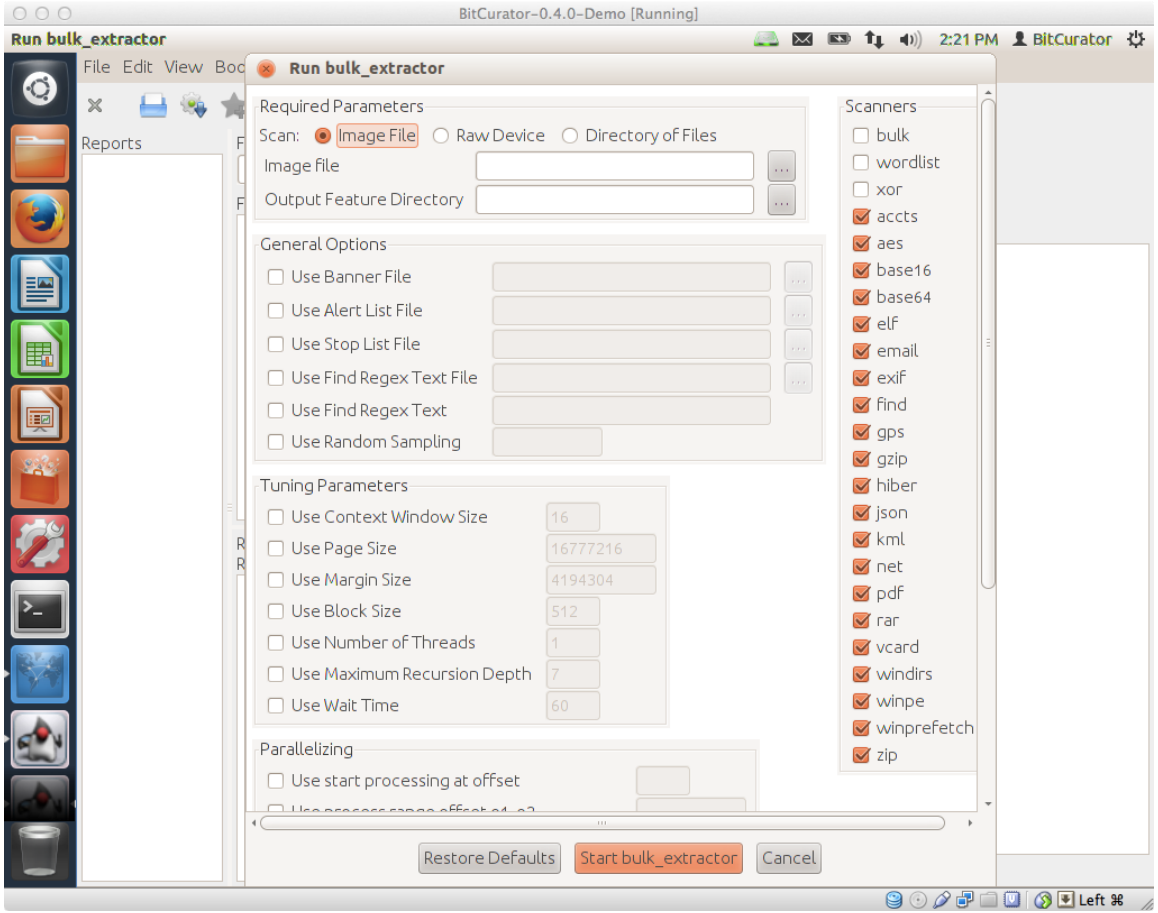
The “Run All” tab will allow you to carve the raw disk contents for features of interest, generate a DFXML listing of the file system hierarchy, links features to files, and generate high-level reports. Click on “Launch BEViewer” to run bulk_extractor before proceeding…

Generating Feature Reports with Bulk Extractor



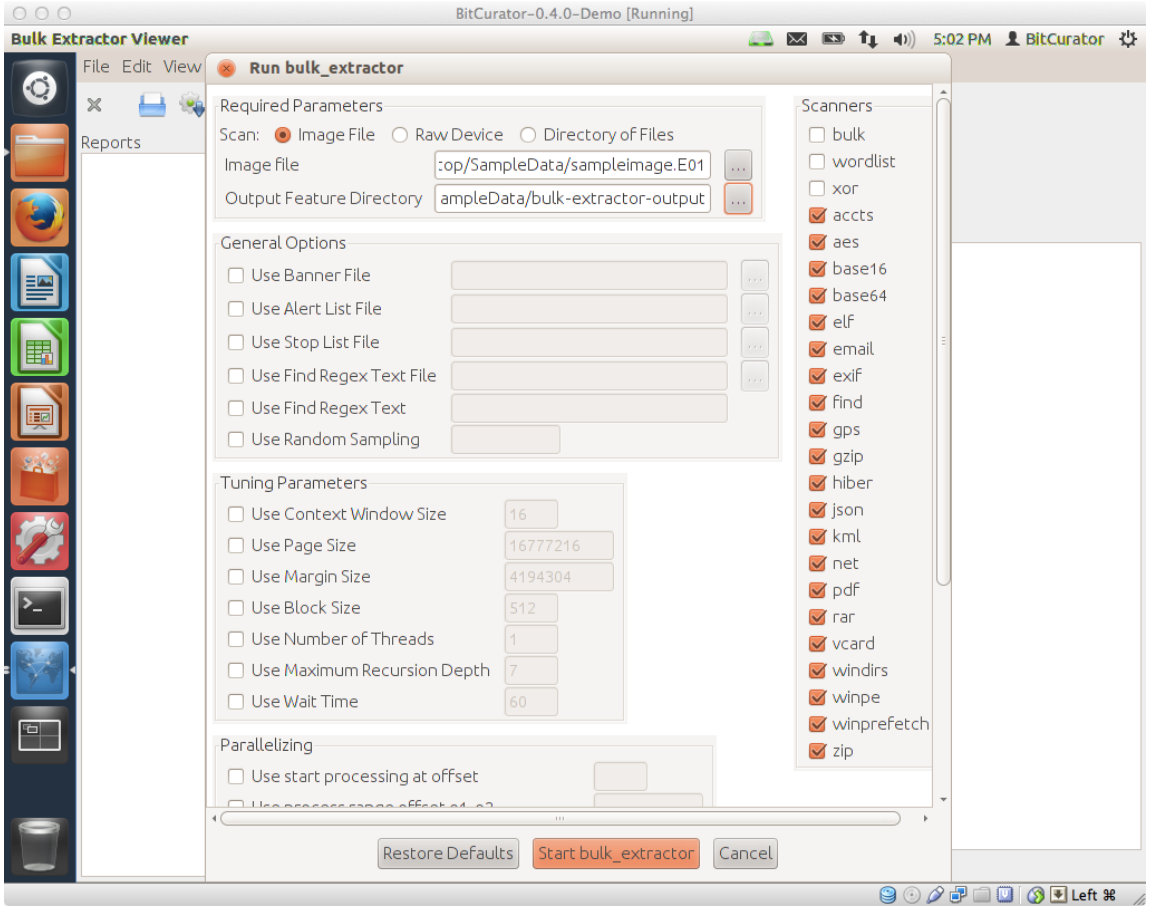
BEViewer is the GUI front-end to Bulk Extractor, a tool that allows you to identify various features of interest contained within the bitstream extracted from the source media, such as SSNs, email addresses, EXIF metadata, and others..

Generating Feature Reports with Bulk Extractor



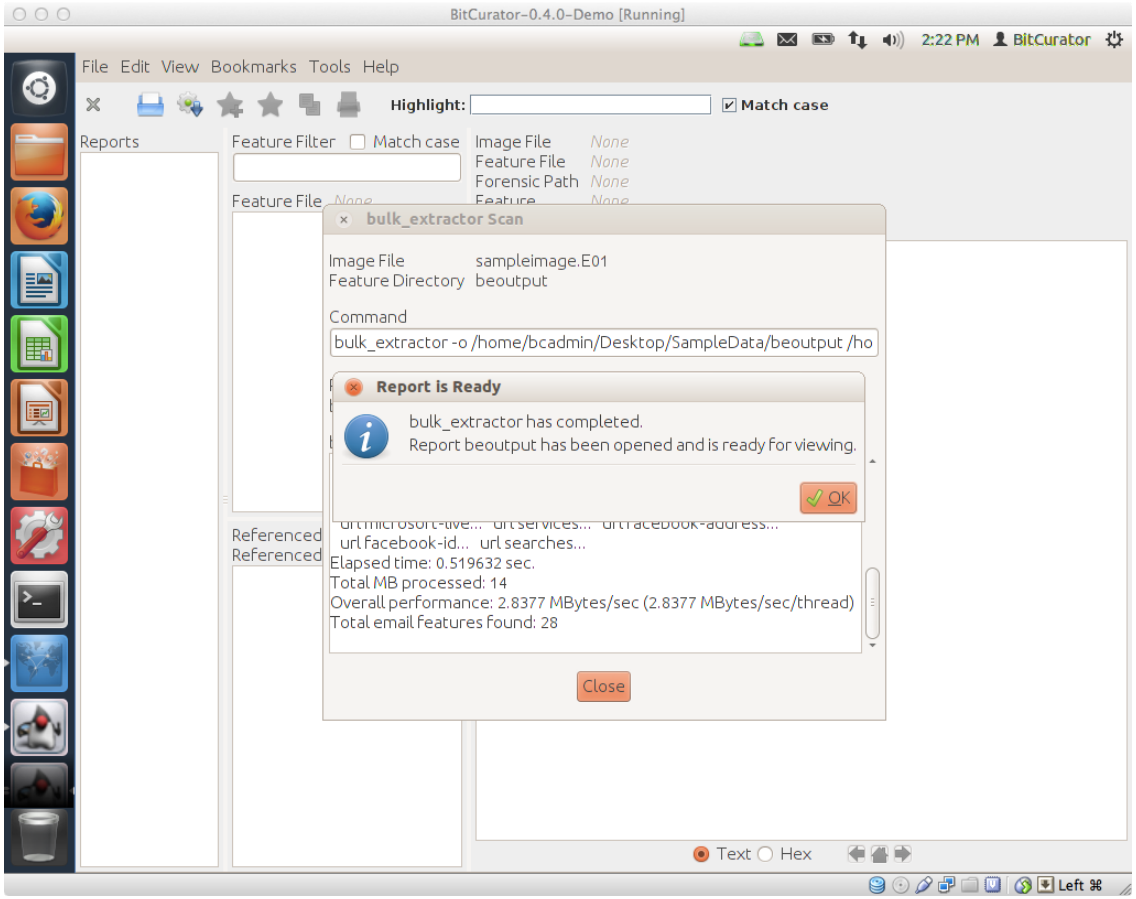
Click on the “Tools” menu in the top of the window, and select “Run Bulk Extractor”. This will bring up a dialog that allows you to select which scanners to run, and where to generate the report directory.

Generating Feature Reports with Bulk Extractor



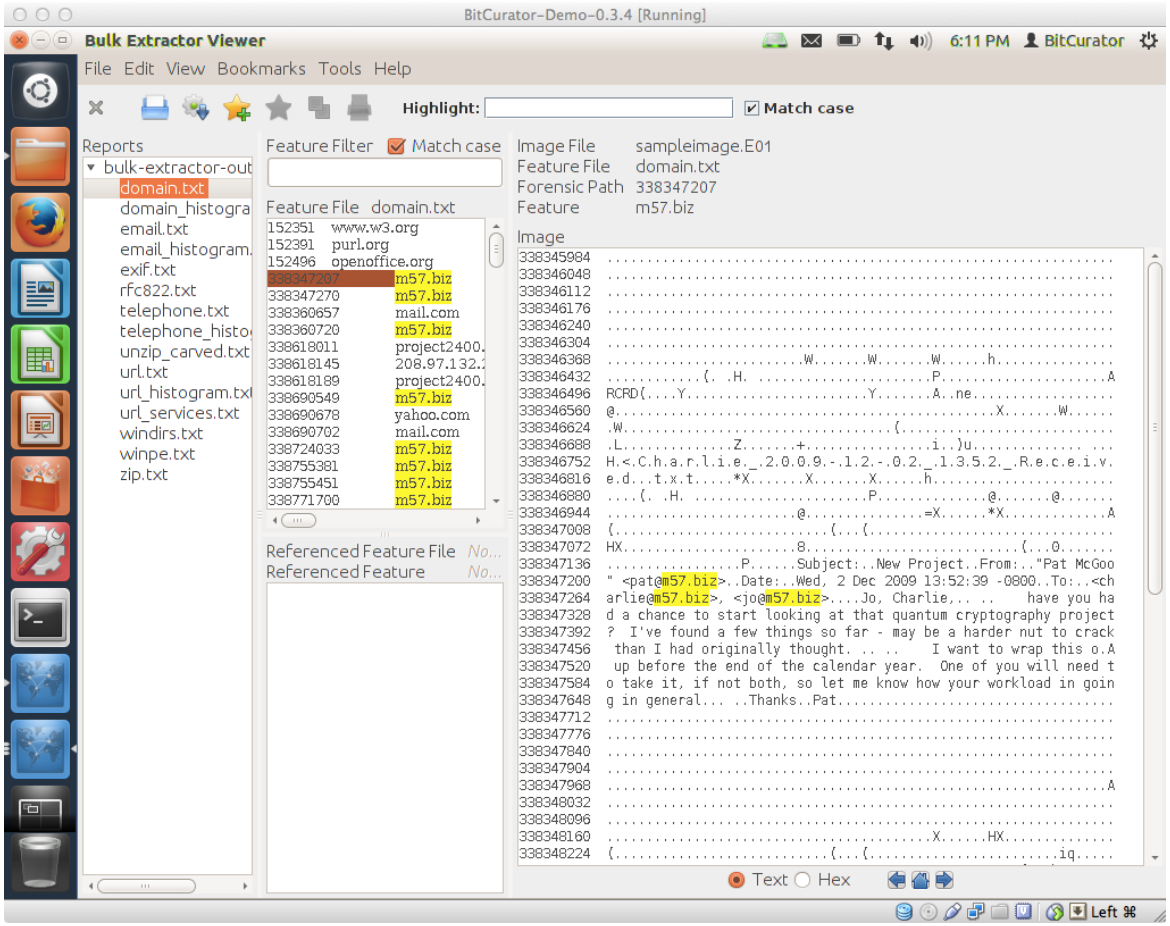
Using the "…" icons to the right of the "Image File" and "Output Feature Directory" text boxes, select the image file we previously produced and tell Bulk Extractor to output the report in a new directory "bulk-extractor-output", within the SampleData directory we made previously on the desktop.

Generating Feature Reports with Bulk Extractor



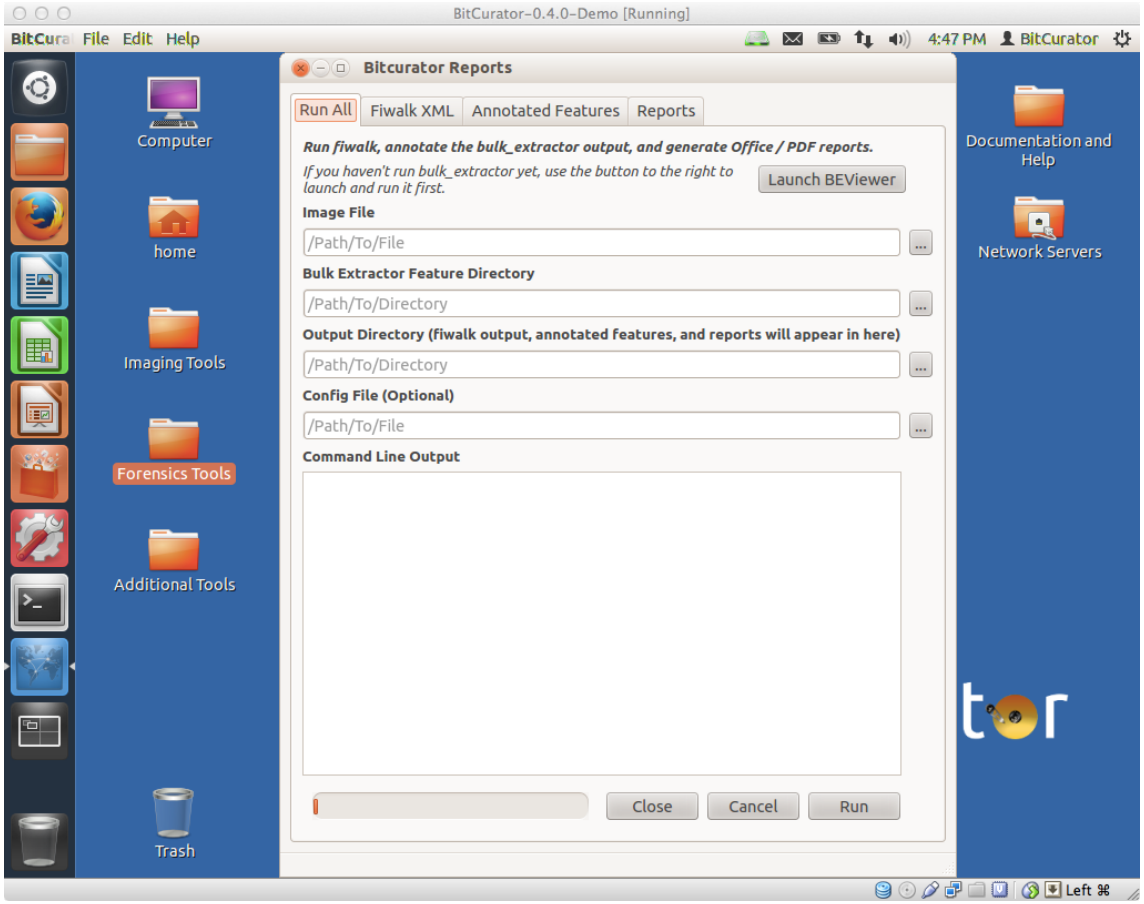
Click on “Start Bulk Extractor” at the bottom of the dialog, and you will see a new dialog appear, indicating the progress made so far. This may take a while for large images. Be patient! **NOTE: Additional processors assigned to the VM will improve performance.**

Viewing the Bulk Extractor Report



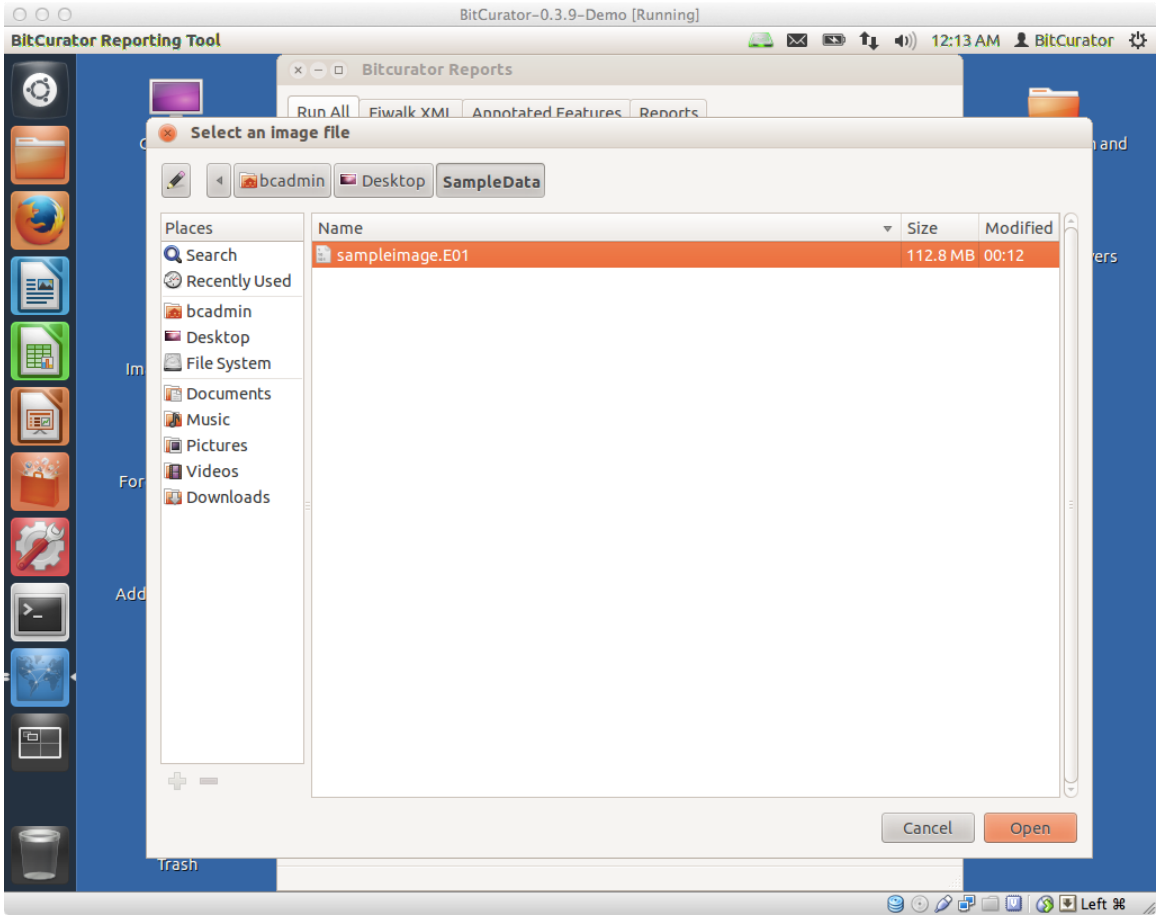
Once the process has completed, the report directory will be available in the relevant location (in our case, the directory “beoutput” within SampleData). The features identified can also be viewed in the main Bulk Extractor Viewer window, by clicking on the report name in the “Reports” subwindow. **Note: For the small disk image shown here, relatively few of the possible reports are shown. Your list may include a range of additional reports.**

Processing the file system, carving data, and generating reports



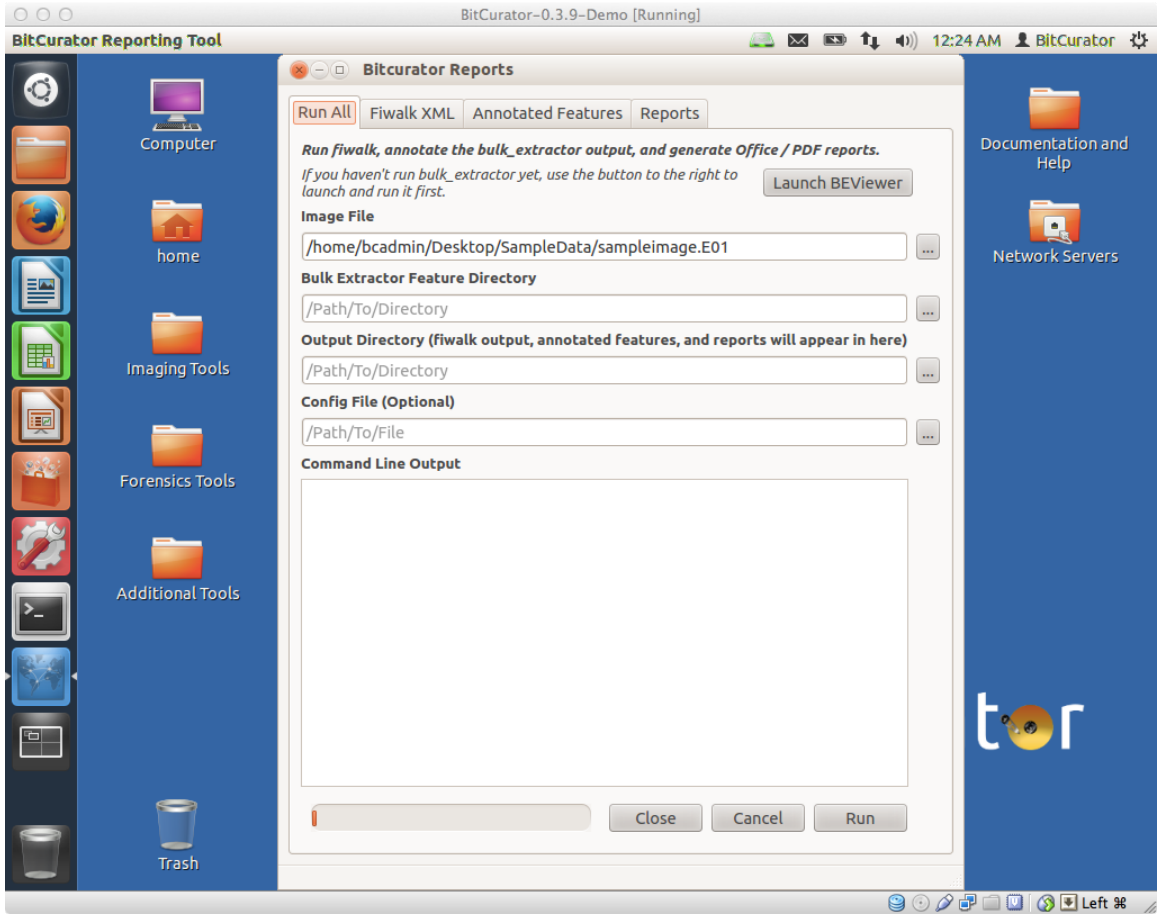
With the bulk_extractor output in place, we can now run fiwalk, the annotation tool that will link bulk_extractor features to files within the file system, and the BitCurator reports, using the “Run All” tab. (Appendix A shows how to run these tools individually using the other tabs).

Processing the file system, carving data, and generating reports



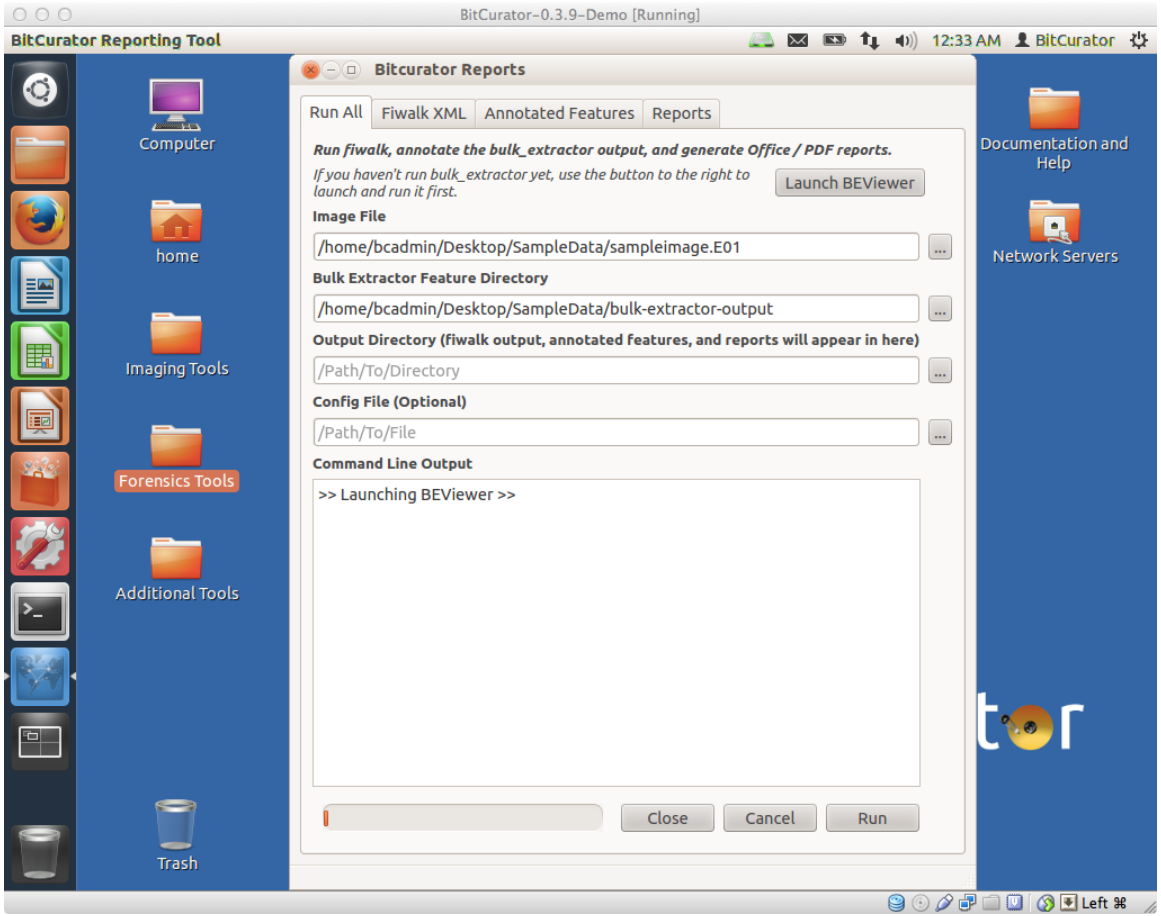
Click on the box with three dots next to the “Image File” entry, and navigate to the sample image (sampleimage.E01) we created in our SampleData directory on the Desktop earlier.

Processing the file system, carving data, and generating reports



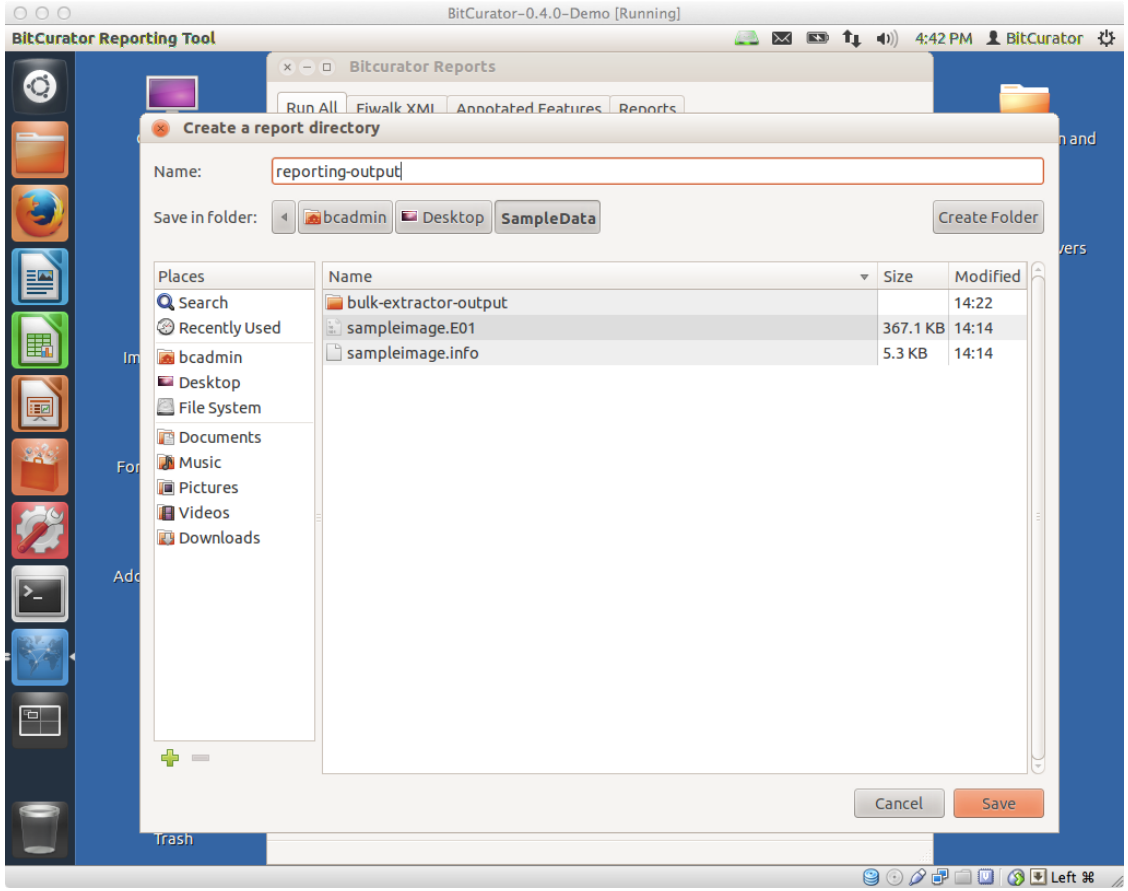
The image file you selected should now appear under the “Image File” entry.

Processing the file system, carving data, and generating reports



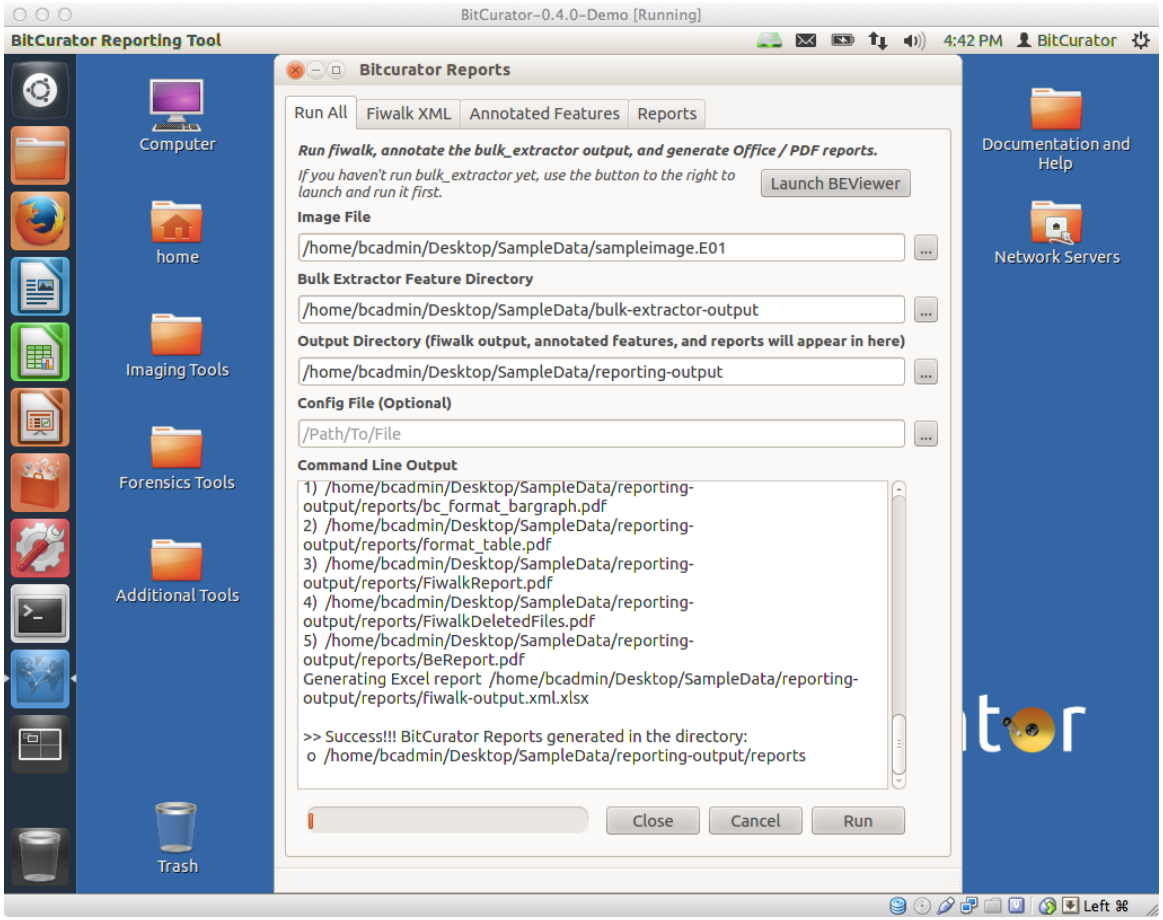
Follow the same process for the “Bulk Extractor Feature Directory” entry. We previously created the “bulk-extractor-output” directory within the “SampleData” directory on the desktop.

Processing the file system, carving data, and generating reports



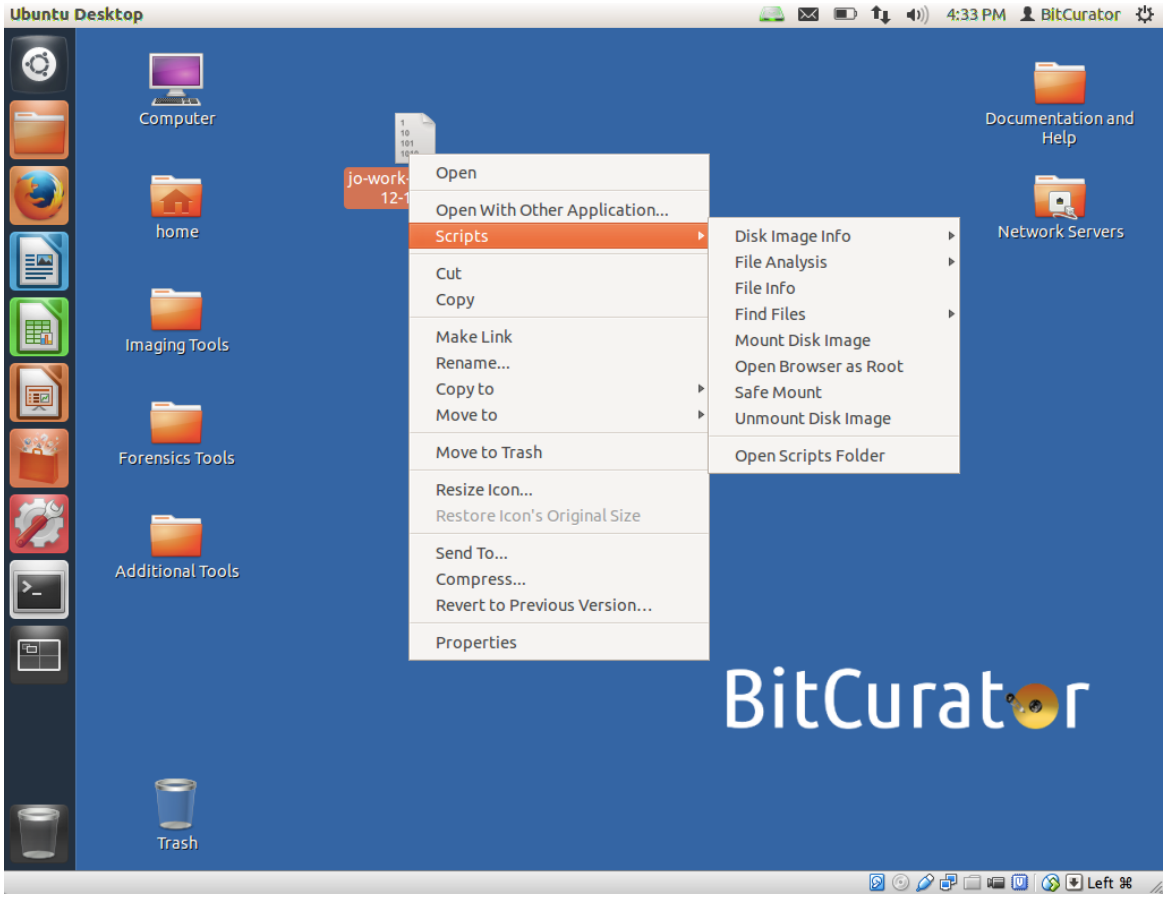
Finally, assign an output directory for the reports that will be generated. Note that you do not need to click “Create Folder” when selecting this location. Simply navigate to the desired location (in this case, Desktop/SampleData) and type in the name of the folder you wish to store the reports in. Then, click “Save”.

Processing the file system, carving data, and generating reports



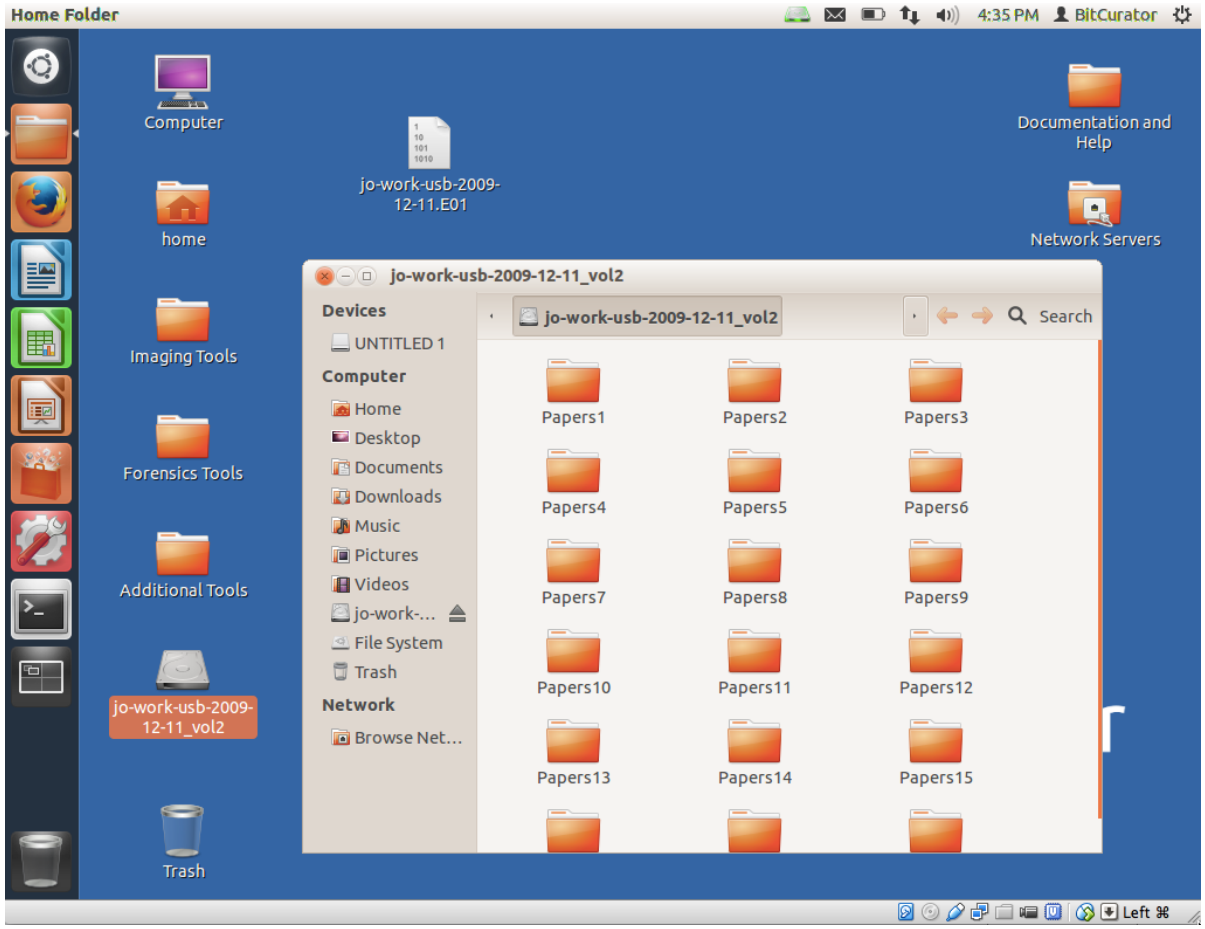
Now, click “Run”. Be patient! It may take some time for each of the steps to run on larger images.

Mounting a disk image to browse the contents



BitCurator includes scripts in the context (right-click) menu that allow you to mount and unmount disk **images** on the Desktop. Simply right click on the image file, and select “Mount Disk Image” or “Unmount Disk Image”.

Mounting a disk image to browse the contents



You'll be asked for the administrative password ("bcadmin") and (for those file systems that BitCurator recognizes) you'll see a disk icon appear on the desktop corresponding to the mounted image.

What We've Done So Far

Closing any open windows, let's open the "SampleData" directory on the desktop and review what we've produced so far:

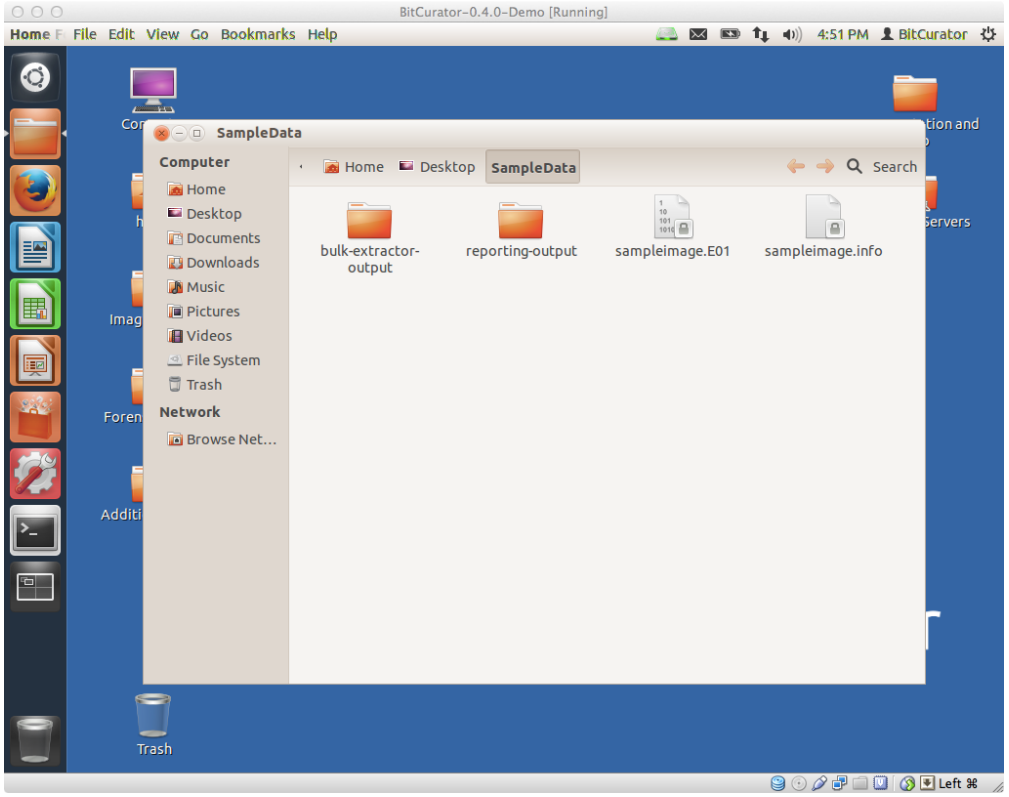
A sample image (sampleimage.E01)

A fiwalk XML report (sampleimage.xml)

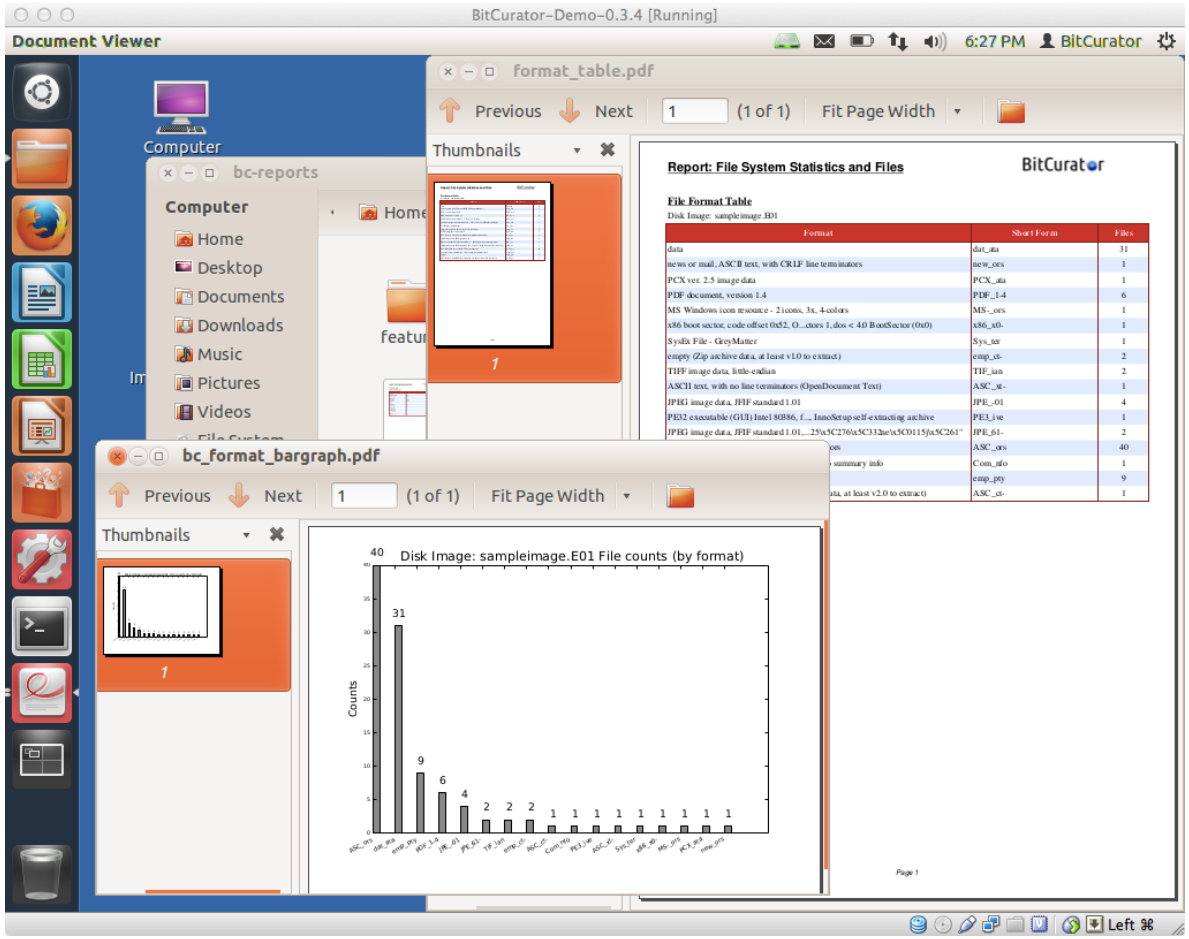
Bulk extractor output (in the beoutput directory)

The annotated output, linking bulk extractor features to files (in the beannotated directory)

A set of human-readable reports for our sample image (in the bcsamplereports directory)

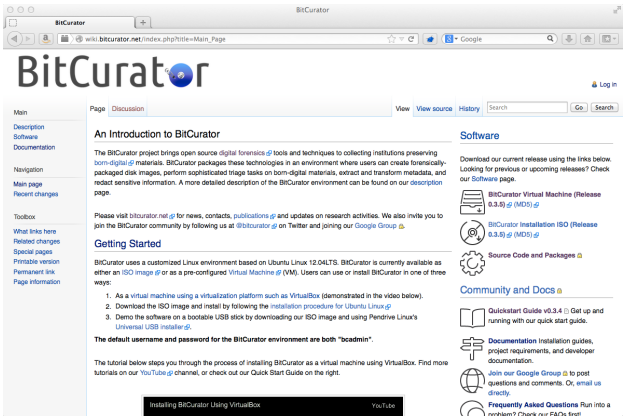


Examining Some of the Reports

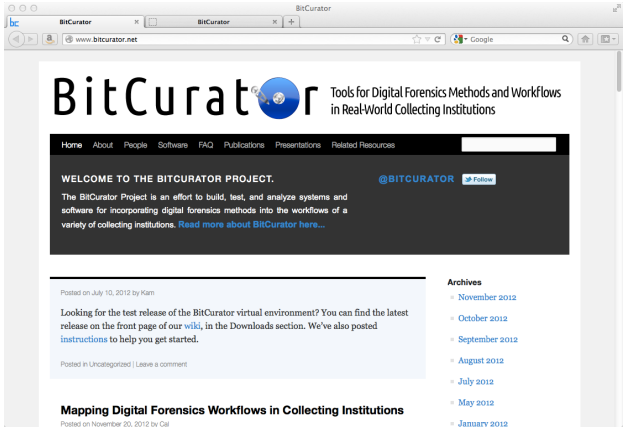


Open the BitCurator reports directory, and examine some of the files. You'll find visualizations, .xlsx transcriptions of file system metadata, high level reports on file types, and overviews of features identified by bulk extractor.

Find Updated BitCurator Information and Documentation Online



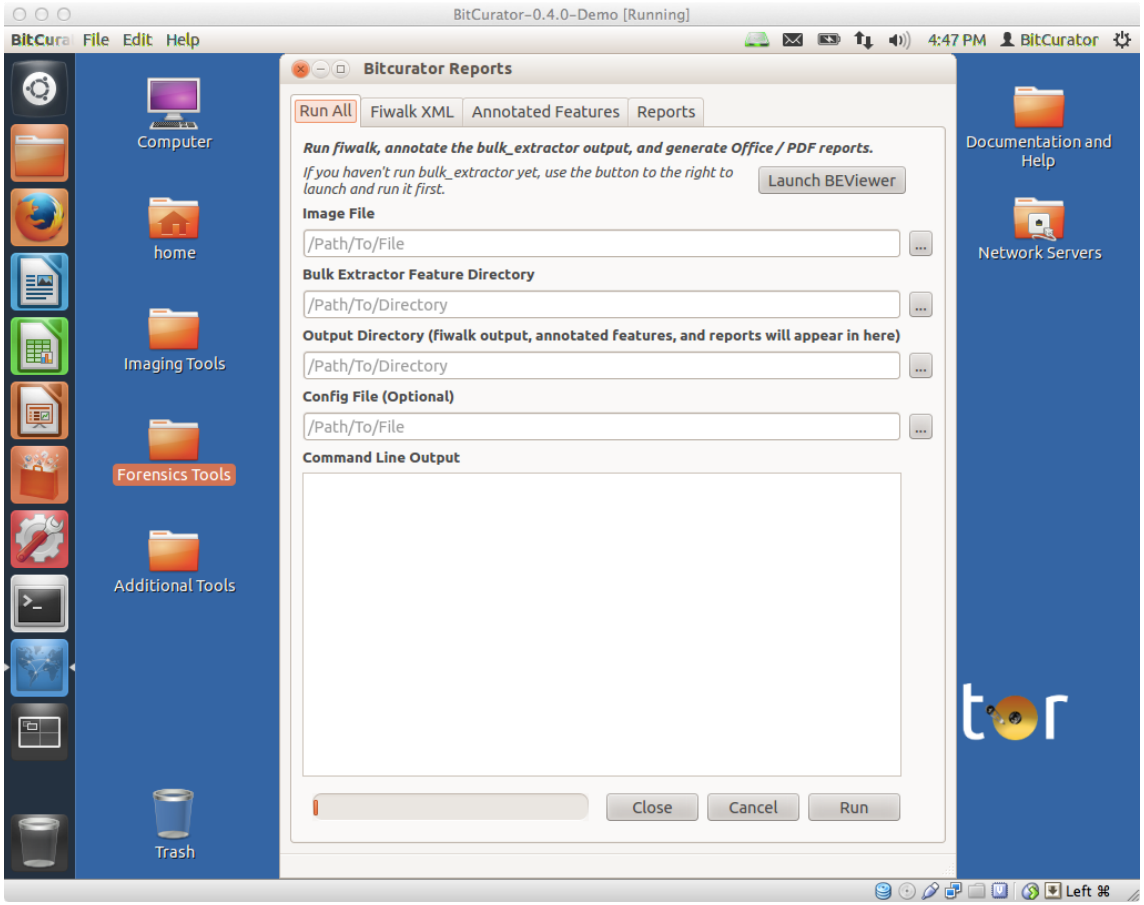
Get the software
Documentation and technical specifications
Google Group
http://wiki.bitcurator.net/



People
Project overview
News
http://www.bitcurator.net/

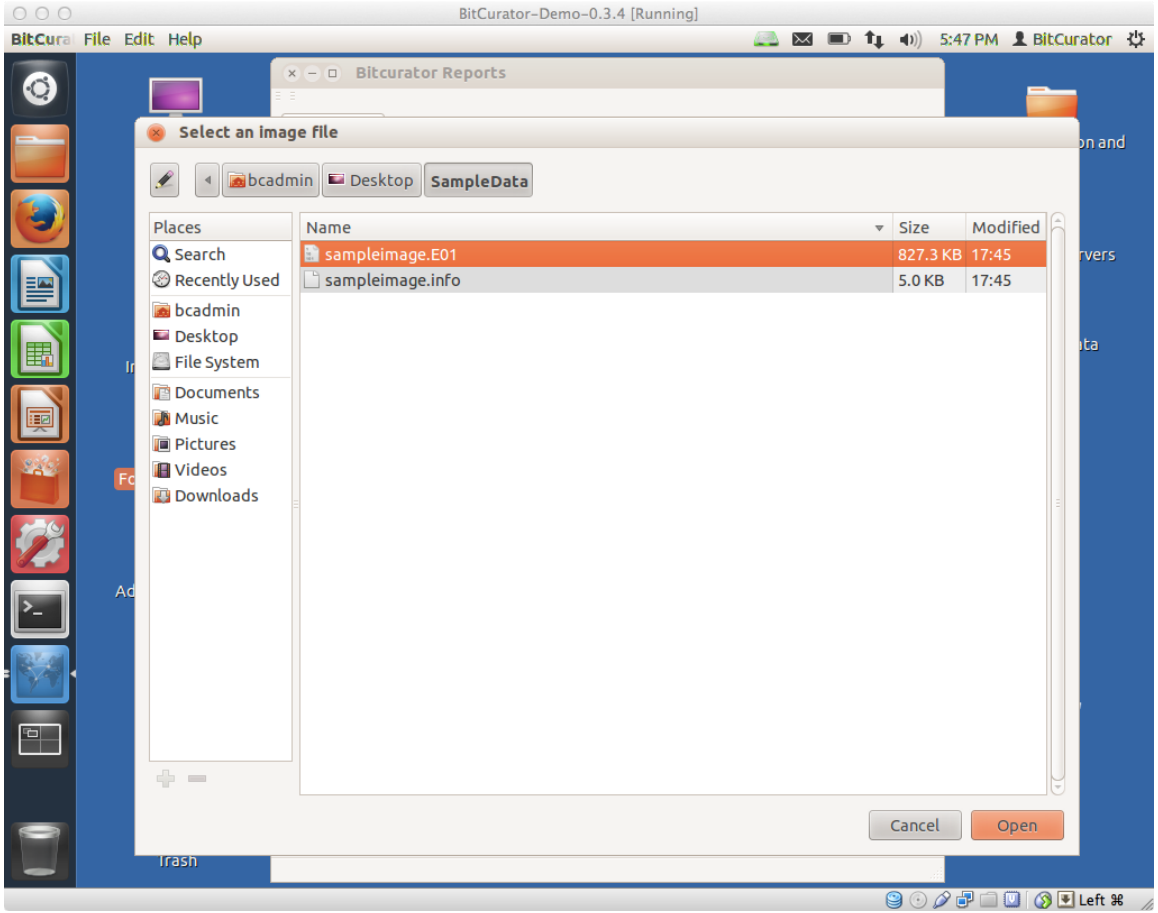
APPENDIX A: Running tools individually

Producing a DFXML report of the file system contents using the ‘fiwalk’ tab.



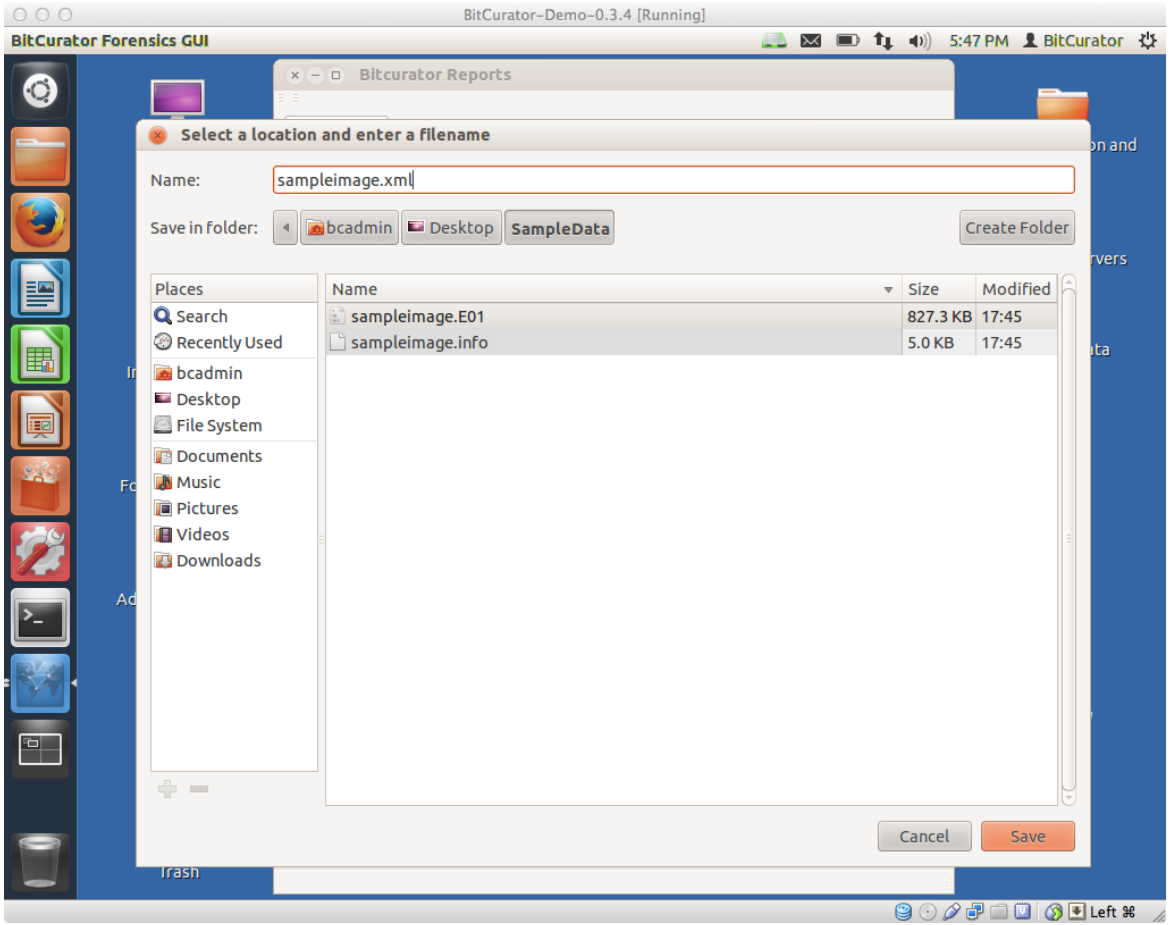
Double-click on the “Forensics Tools” folder, and then double click on the “BitCurator Forensics GUI” launcher. You’ll see a window pop up that should match the picture shown above. Select the “Fiwalk XML” tab…

Producing a DFXML report of the file system contents using the ‘fiwalk’ tab.



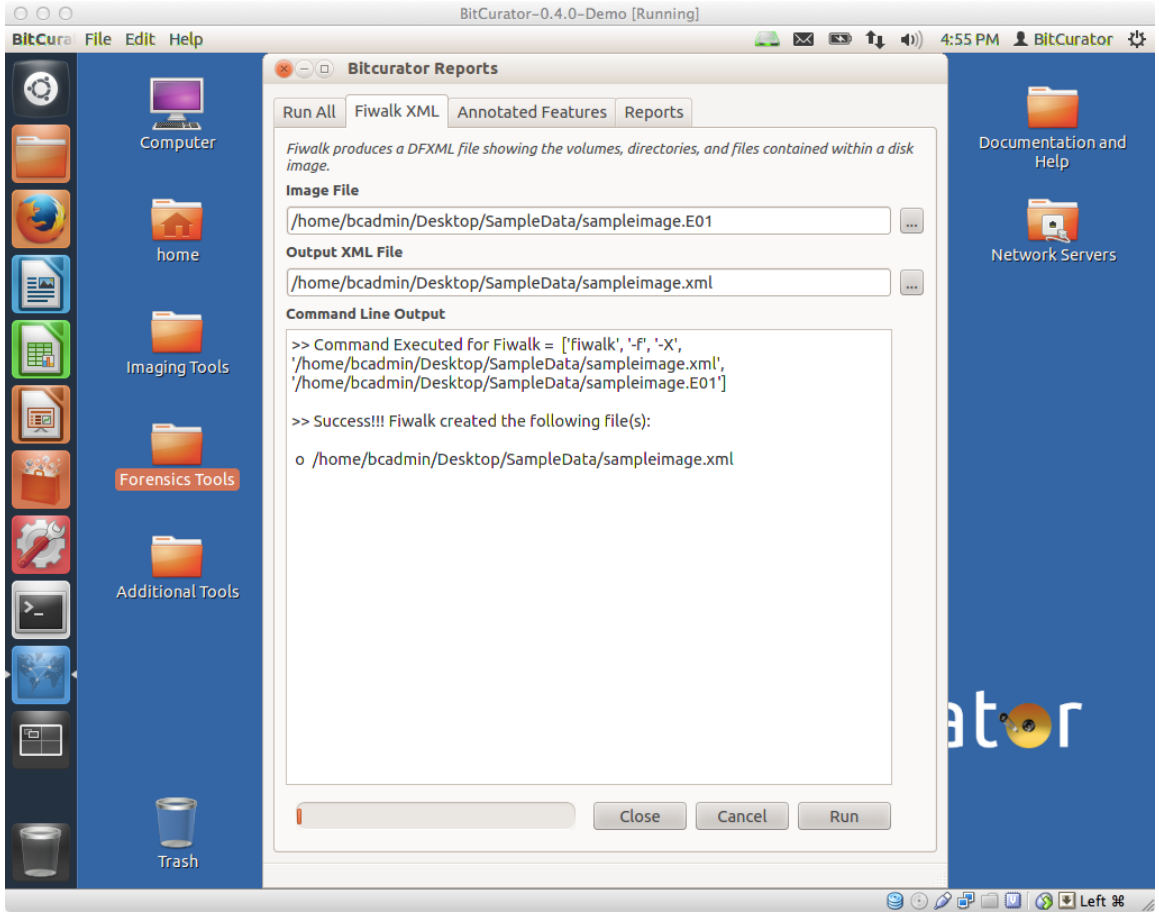
Fiwalk needs to know where the disk image file you created is, and it needs to know where to create the DFXML output file. Click on the box with the three dots to the right of the Image File text edit box, and navigate to the directory containing the image we just created. Select ‘sampleimage.E01’ and click ‘OK’.

Producing a DFXML report of the file system contents using the ‘fiwalk’ tab.



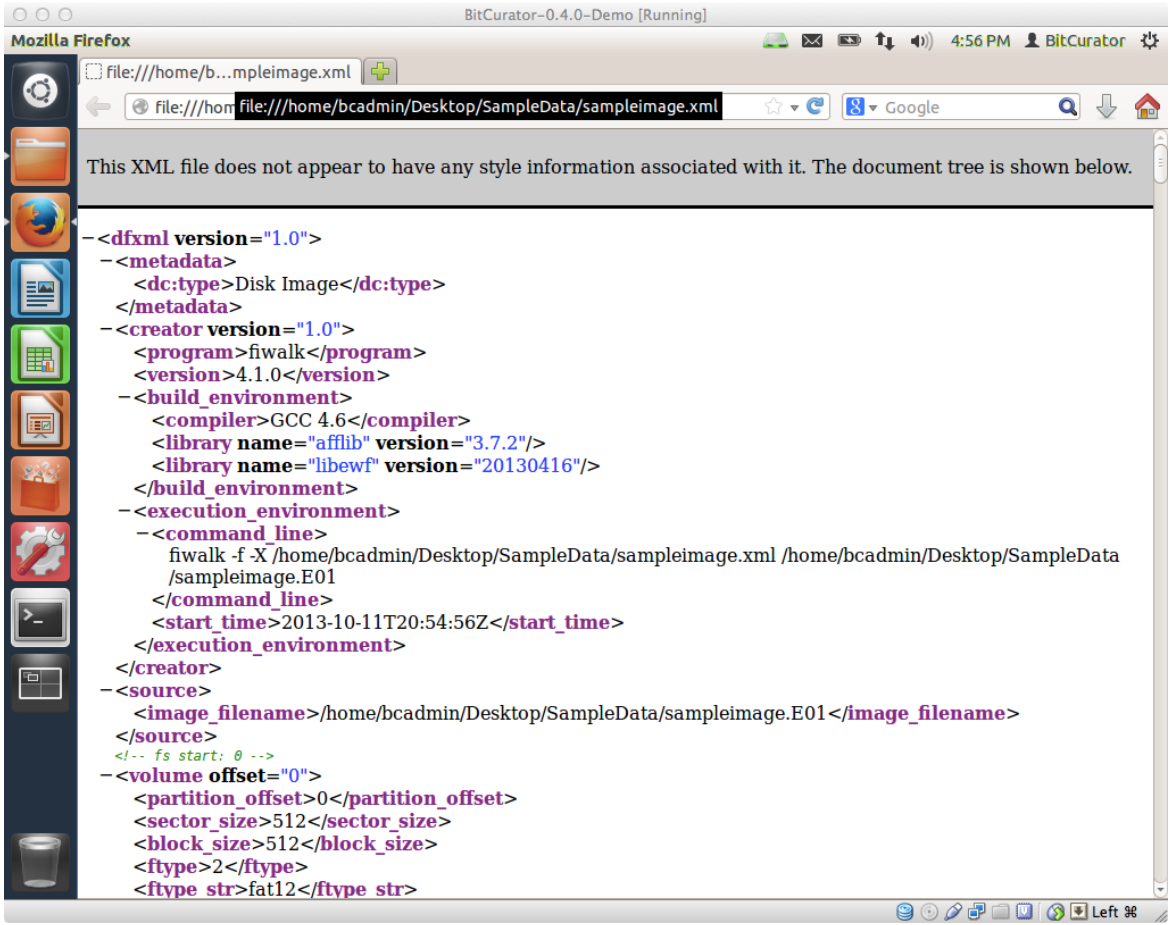
Now click on the box with three dots to the right of the ‘Output XML File’ text area, and navigate to the same directory on the desktop. Type in “sampleimage.xml” under ‘Name’ at the top, and click ‘OK’.

Producing a DFXML report of the file system contents using the ‘fiwalk’ tab.



Your main window should now have both the Image File and the Output XML File fields filled with the appropriate locations. Click ‘Run’, and fiwalk will run.

Producing a DFXML report of the file system contents using the ‘fiwalk’ tab.



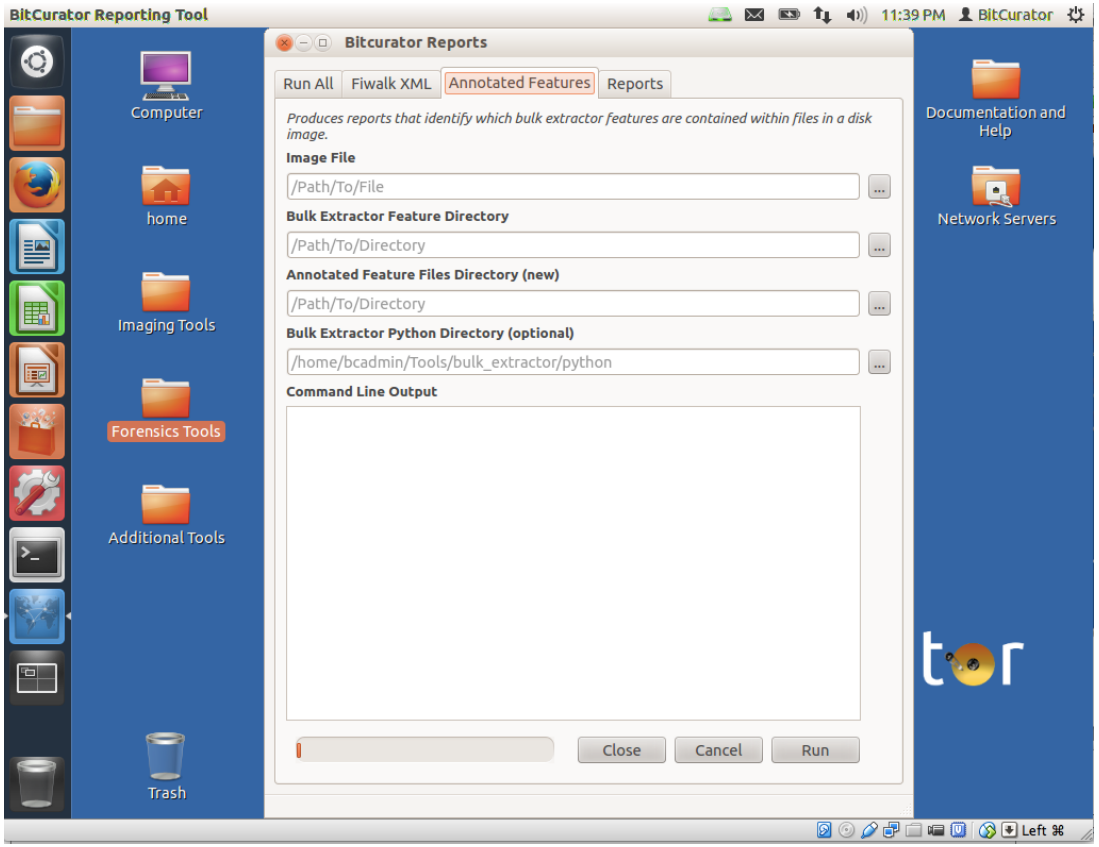
The resulting DFXML file can be found in the Sample Data directory we created earlier on the desktop. You can examine the contents by double-clicking on it.

Matching Features to Files

Bulk Extractor extracts these features from a disk image by scanning the raw bitstream – not by parsing the file system.

In order to determine which files these features appear within (or if they appear on an area of the disk not associated with the file system), we need to run an additional tool.

For the next step, either maximize the BitCurator GUI you minimized earlier, or restart it from the 'Forensics Tools' directory on the desktop. Click on the 'Annotated Features' tab.



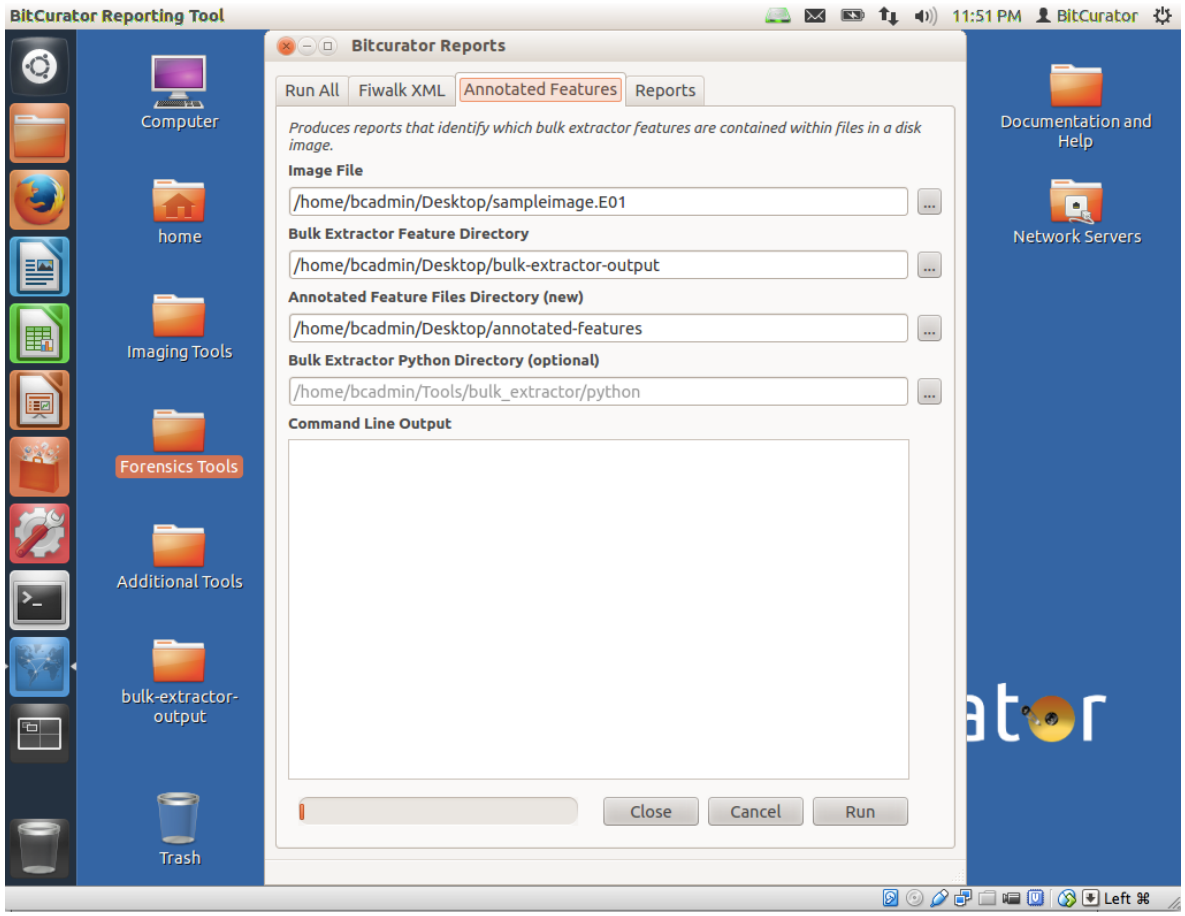
Matching Features to Files

In order to annotate the features – that is, identify which features belong to which files within the file system – we need to know about four things:

1. Which feature reports to work from (the GUI uses all of them; if you'd like to be more selected, there is a command-line option)
2. Where the image file is
3. Where the fiwalk output is
4. Where the bulk extractor output we just created is
5. Where we want to generated the output. In this case, we're telling it to make a new directory called "beannotated" in our SampleData directory on the desktop.

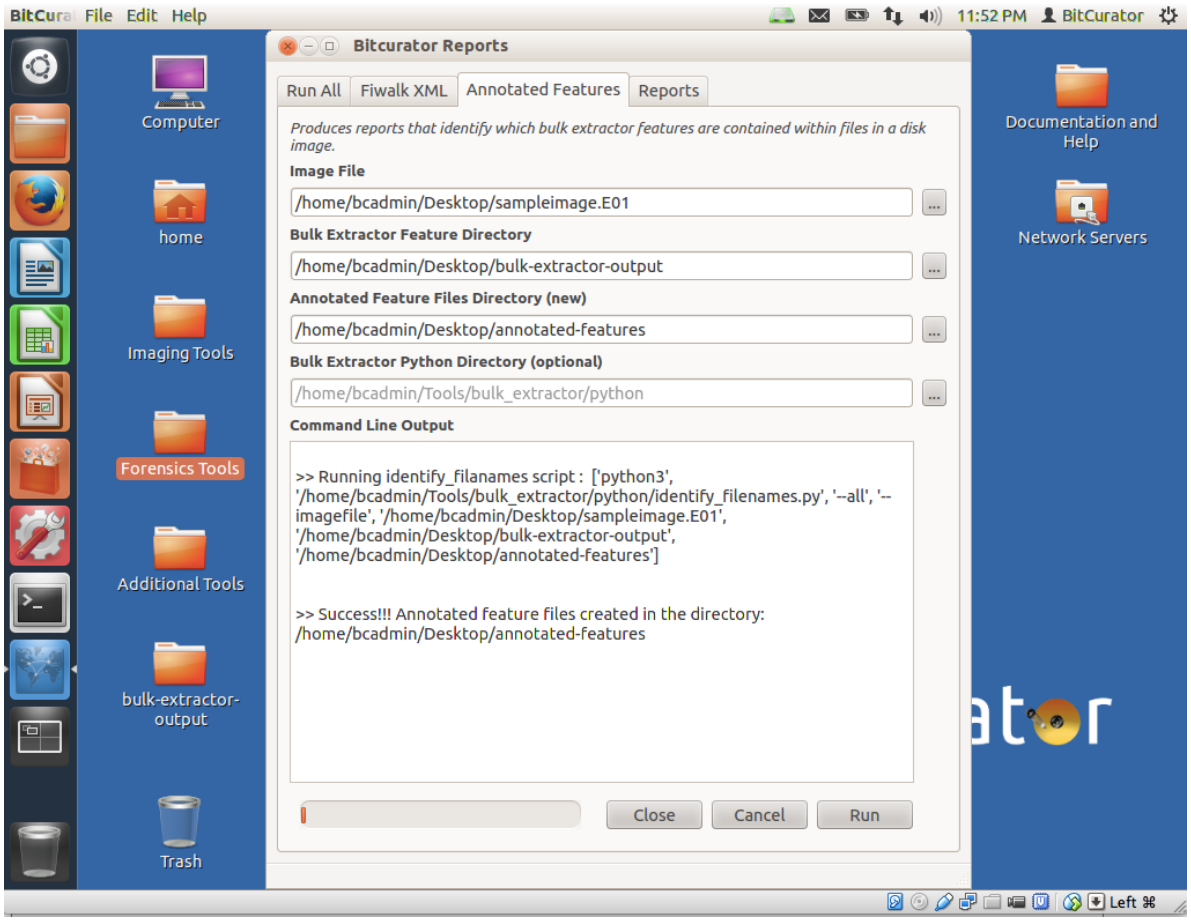
In the following slide, we follow the same procedure for selecting these items as we did for the fiwalk tab.

Matching Features to Files



In the screenshot shown above, we've selected the existing image file, the existing bulk extractor output directory, and named a new directory within the SampleData directory on the desktop for the annotated features. (All of these steps were performed by bringing up the relevant file dialogues for each selection by clicking on the great boxes with three dots to the right of each text box). **Note: The Bulk Extractor python directory can be left unmodified.**

Matching Features to Files



Click ‘Run’, and the tool will run. Scroll down in the ‘Command Line Output’ window and you should see a ‘Success’ message as indicated above.

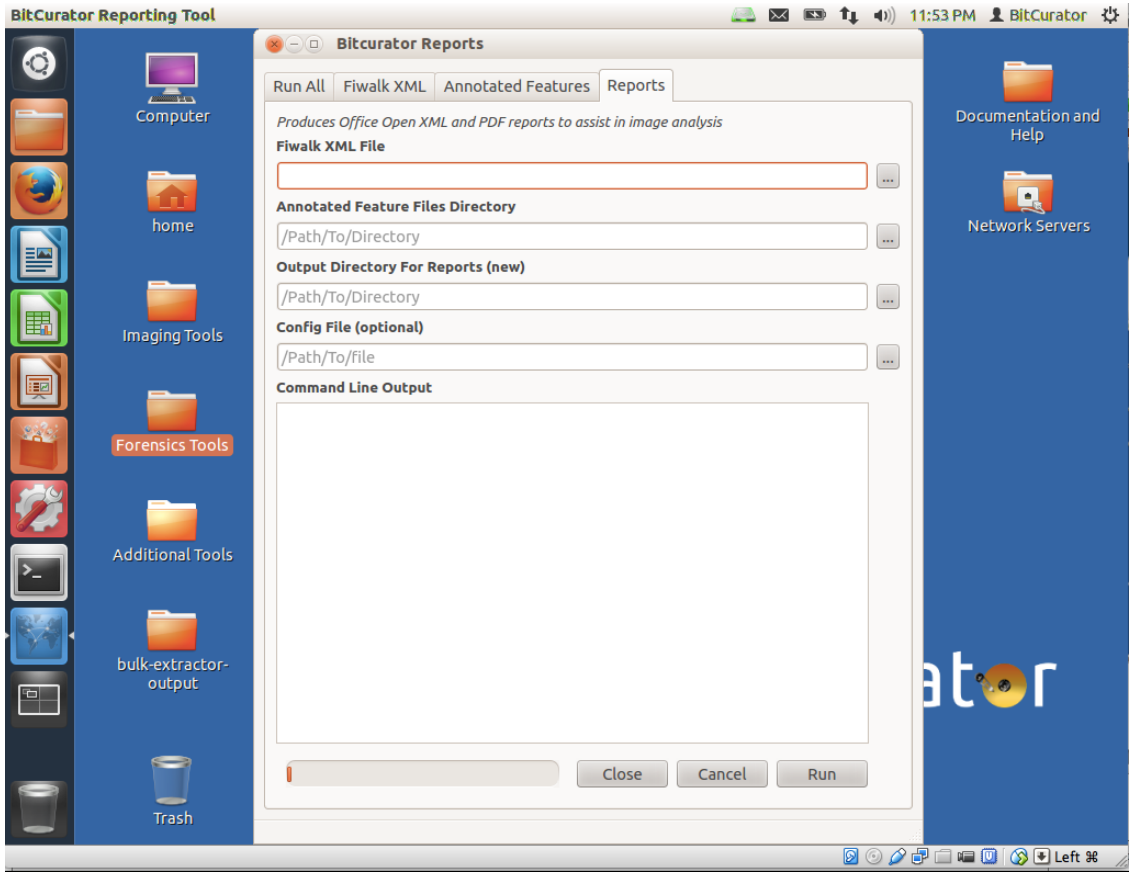
Generating BitCurator Forensic Reports

Now that we have a disk image, an XML representation of the file system contents, a directory of feature files, and a set of reports that match features to filenames, we can run the BitCurator reporting tool.

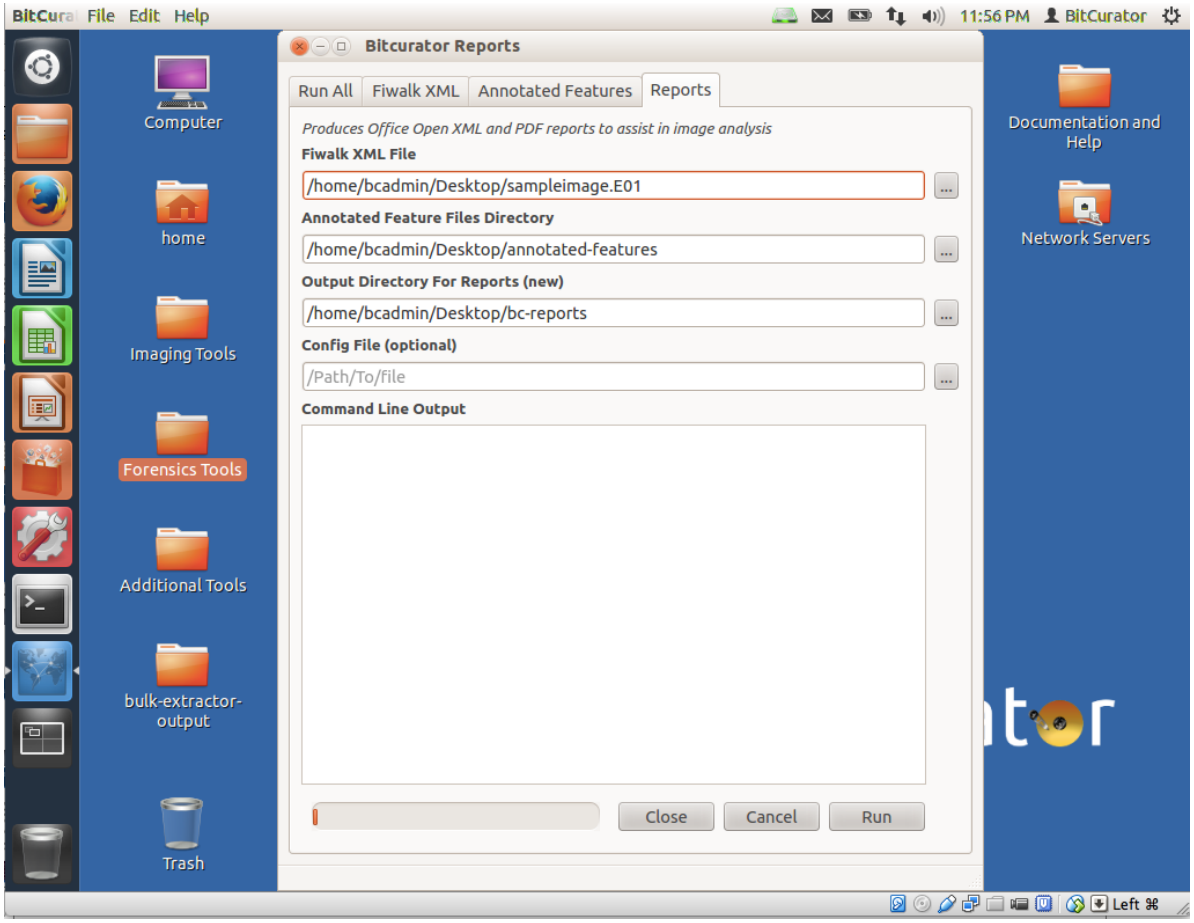
Click on the ‘Reports’ tab in the BitCurator GUI.

The “Generate Report” program needs to know about four things:

1. Where the fiwalk output is
2. Where the annotated bulk extractor report directory is (we generated this in the previous step)
3. Where we want to generate the output.

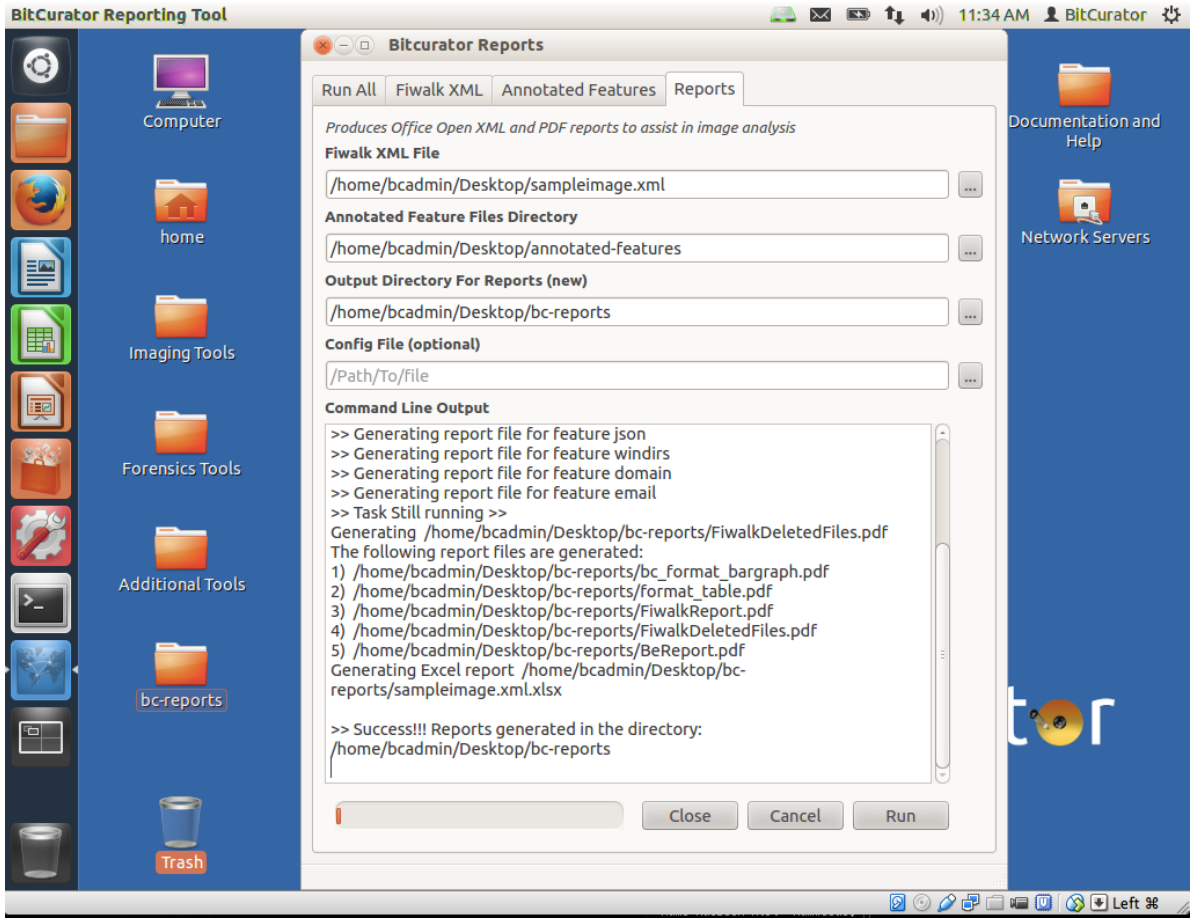


Generating BitCurator Forensic Reports



As in steps for the previous tabs, use the grey squares to select the fiwalk XML file that we created in the SampleData directory, the annotated features directory, and – finally – to specify a new output directory for the BitCurator reports. In the image above, we've chosen to place this new directory in SampleData, and call it "bc-reports".

Generating BitCurator Forensic Reports



Click ‘Run’, and you will see output appear in the ‘Command Line Output’ box indicating success or notifying you of an error.

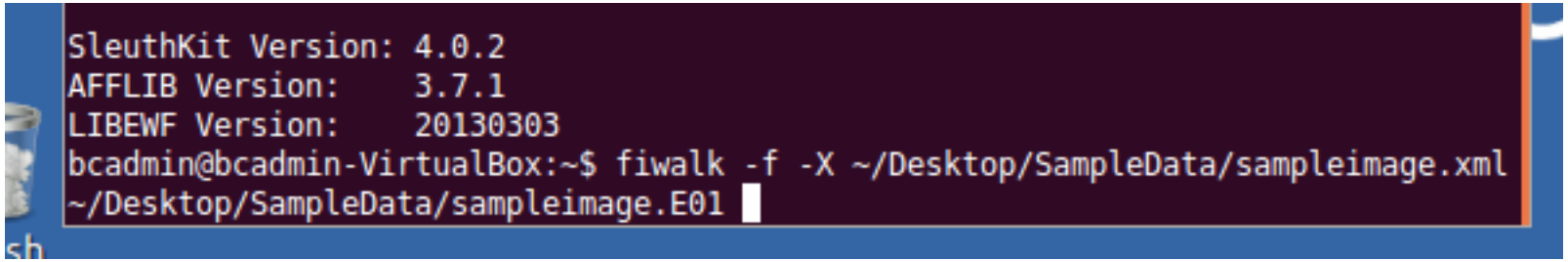
APPENDIX B: Using these tools via the command-line

B1. Fiwalk: Producing a DFXML report of the File System Contents

The fiwalk program really only needs to know three things:

1. Whether you want to run “file” to identify the file formats in the file system (the ‘- f’ option)
2. The name of the DFXML file that will be produced (‘-X’, followed by the file path)
3. The name of the image to process.

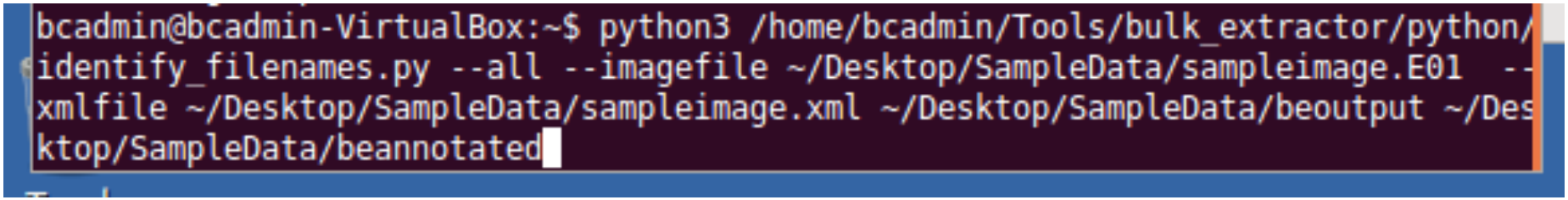
The command to run is shown below (the ‘~/’ at the beginning of each path just tells the program to start looking for these folders in the user’s home directory)



B2. Identify_filenames.py: Matching Features to Files

The “Identify Filenames” program needs to know about four things:

1. Which feature reports to work from (here we’ve used the “all” flag to tell it to use all of them)
2. Where the image file is (“—image file [FILE LOCATION]”)
3. Where the fiwalk output is (“—xmlfile [FILE LOCATION]”)
4. Where the bulk extractor output is (just the location)
5. Where we want to generated the output. In this case, we’re telling it to make a new directory called “beannotated” in our SampleData directory on the desktop.



B3. BitCurator reporting: Running the Report Generator

The “Generate Report” program needs to know about four things:

1. Where the fiwalk output is (“—fiwalk_xmlfile [FILE LOCATION]”)
2. Where the annotated bulk extractor report directory (the one we generated in the last step) is (“—annotated_dir [DIRECTORY LOCATION]”)
3. Where we want to generated the output. In this case, we’re telling it to make a new directory called “bcsamplereports” in our SampleData directory on the desktop.

Finally, we’ll get a couple of prompts for configuration. We’ll use the defaults for now (typing “Y” and enter for the first prompt, and simply hitting enter for the second)

```
bcadmin@bcadmin-VirtualBox:~$ python3 /home/bcadmin/Tools/bitcurator/python/gene
rate_report.py --fiwalk_xmlfile ~/Desktop/SampleData/sampleimage.xml --annotated
_dir ~/Desktop/SampleData/beannotated --outdir ~/Desktop/SampleData/bcsamplerepo
rts
>>> Do you want to specify the configuration file?: [Y/N]:Y
>>> Please specify the configuration file[/etc/bitcurator/bc_report_config.txt]:
```