BitCurator

Quick Start Guide

Last updated: December 29, 2018

Release(s): 2.0.12 and later

UNC
SCHOOL OF INFORMATION
AND LIBRARY SCIENCE

BitCurator
CONSORTIUM

About BitCurator

The **BitCurator Environment** is a Ubuntu-derived Linux distribution geared towards the needs of archivists and librarians. It includes a suite of open source digital forensics and data analysis tools to help collecting institutions process born-digital materials.

The BitCurator environment is distributed as a VM that can be run in VirtualBox, and as a Live ISO that can be used to install BitCurator on a dedicated host machine.

We recommend that you run BitCurator on a dedicated machine in production environments by installing from the Live ISO image.

What this Document Covers

- How to run the BitCurator environment as a virtual machine in VirtualBox, and how to install the BitCurator environment using the Live CD image
- Generating reports on file system contents
- Using DFXML metadata to characterize the contents of file systems contained in disk images and raw media devices
- Exporting file system metadata, and files from disk images and live media
- Locating and processing sensitive and personally identifying information within digital materials (using bulk_extractor)
- Understanding the main data elements that are generated by open source forensics tools (using DFXML)
- Other useful tools included in the BitCurator environment

How To Use This Document

- **Planning on running BitCurator in VirtualBox using the existing VM?**
  - Use the instructions on Pages 5-16

- **Planning on installing BitCurator using the Live ISO image?**
  - Use the instructions on Pages 17-20

- General instructions regarding using the tools in BitCurator begin on Page 21

Getting Started with the Virtual Machine

Getting Started with the Virtual Machine

- Hardware:
  - Desktop or laptop with an Intel Core i7 processor (recommended; similar processors are also suitable)
  - 64-bit Windows 10, macOS 10.14 (or newer), or a 64-bit Linux variant.
  - 16GB RAM recommended (8GB minimum)
  - 20GB free hard disk space (minimum). The virtual machine is approximately 10GB when unpacked, and will expand to 256GB as needed. Solid-state disk (SSD) recommended.
- Software:
  - BitCurator VM or Live CD: https://github.com/BitCurator/bitcurator-distro/wiki/Releases
  - Current release of VirtualBox: https://www.virtualbox.org/wiki/Downloads
  - VirtualBox Extension Pack.
    - **Must** be installed on the host for shared folder and USB 2.0/3.0 support. Download and double-click on the file once you've installed VirtualBox. The link is **just below the main download** on the page linked above.

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-X.X.X.tar.gz”

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

- When using 7-zip, you’ll need to **unpack the .tar.gz file.** Right click on the **.tar.gz** file and select “Extract here…” to extract the **.tar** file. Then right click on the **.tar** file and select “Extract here…” again. This will extract a directory containing the BitCurator virtual machine disk image (.vbox) and configuration (.vdi) files.

BitCurator Virtual Machine Files

- Once you've unpacked the .tar.gz file, you'll find a directory containing two files (where X.X.X is your downloaded version):
  - BitCurator-X.X.X.vbox (the VirtualBox configuration file)
  - BitCurator-X.X.X.vdi (the VirtualBox disk image)

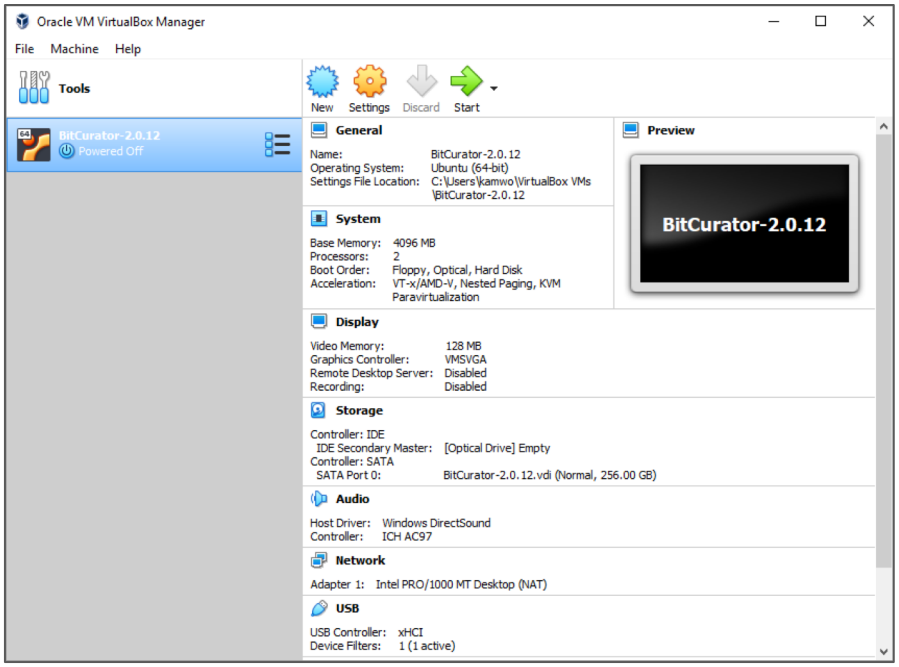
- Copy this directory to a location of your choosing (inside the "VirtualBox VMs" directory in your home directory is a good place), and start up VirtualBox.

- **Tip: 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.**

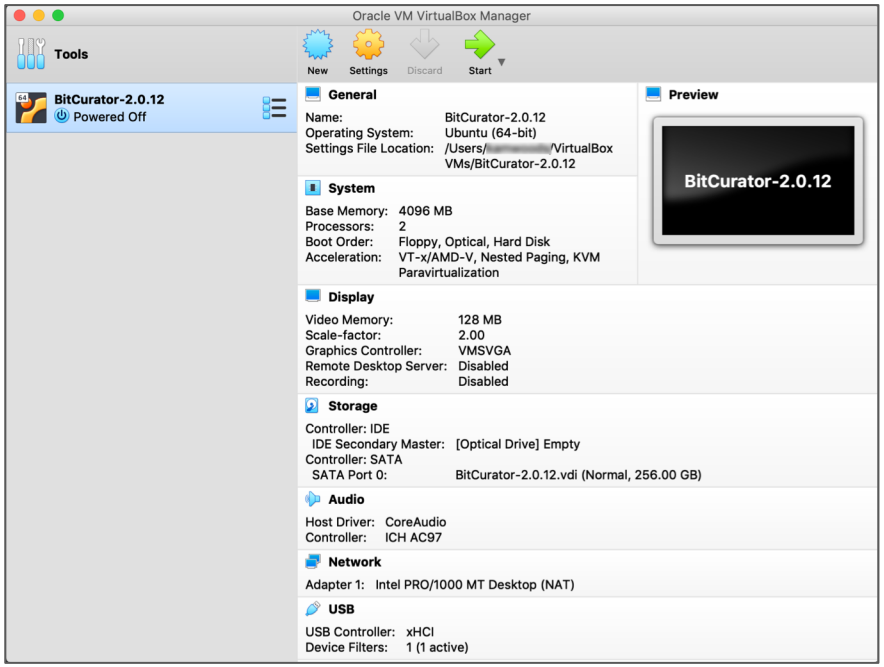
Oracle VM VirtualBox Manager

Once you've installed VirtualBox and the VirtualBox extension pack, start up VirtualBox. If you've never used VirtualBox before, your list of machines (on the left) will be blank.

Tip: You may need to right-click on the VirtualBox icon and select "Run as Administrator". Windows machines with certain administrative controls may prevent you from accessing USB devices (or lock out control of the mouse and keyboard) otherwise.



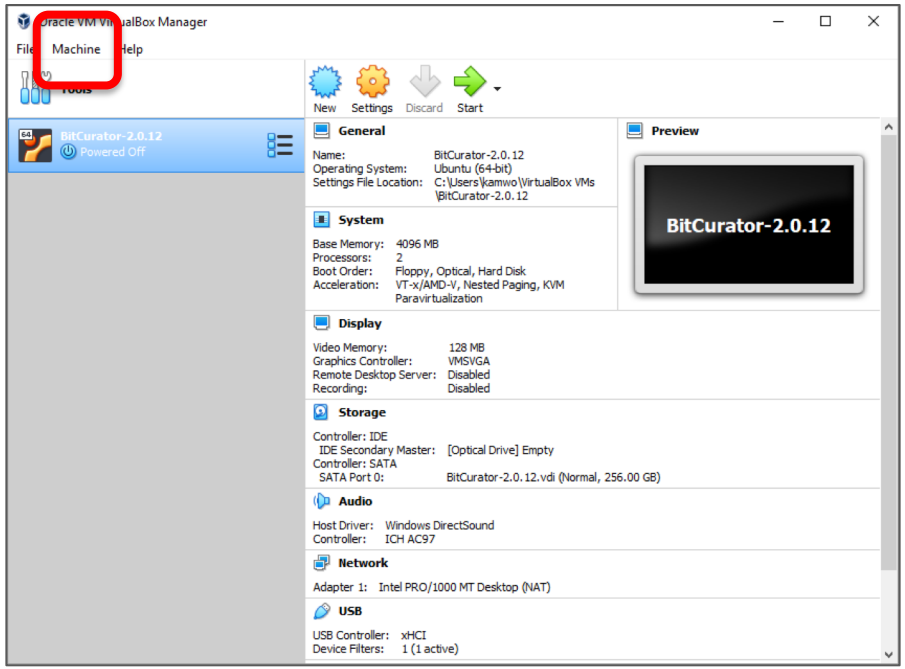
VirtualBox Manager in Windows



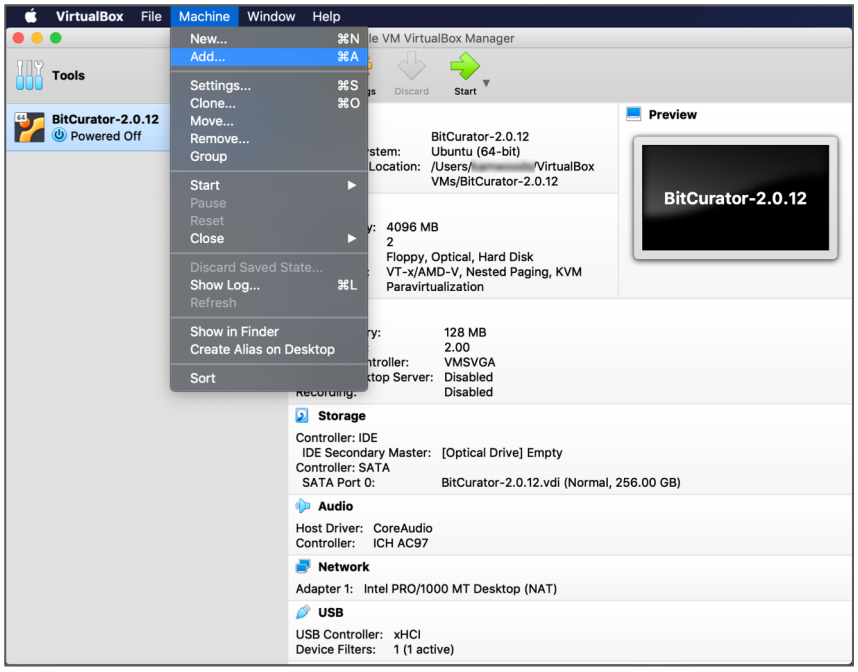
VirtualBox Manager in macOS

Oracle VM VirtualBox Manager

From the menu bar, select the menu item “Machine -> Add…”, and **navigate to the folder containing .vbox file that you extracted**. Choose that file, and the Virtual Machine should appear in the list within the manager.



Adding a machine (VirtualBox in Windows)

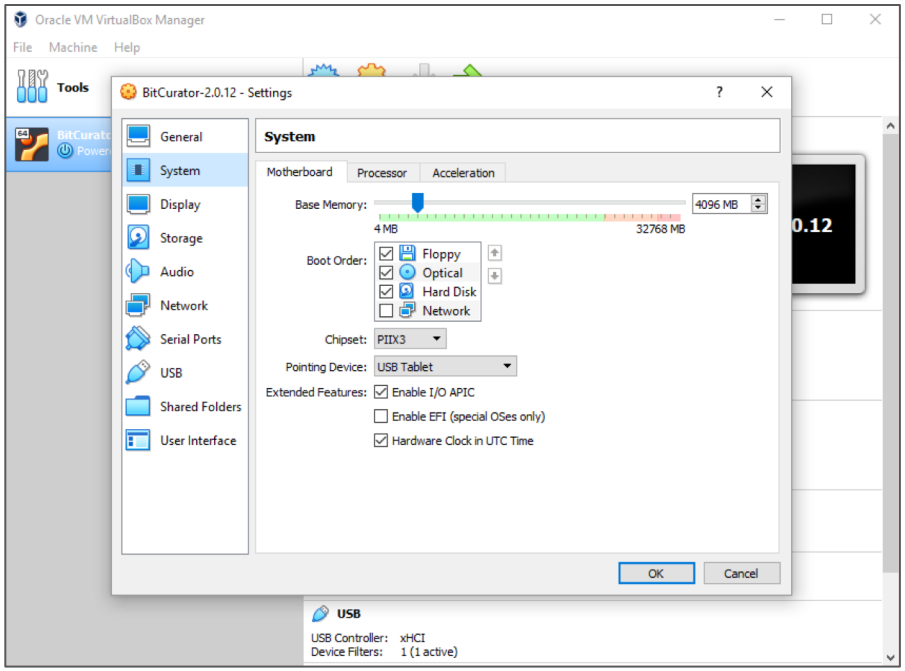


Adding a machine (VirtualBox in macOS)

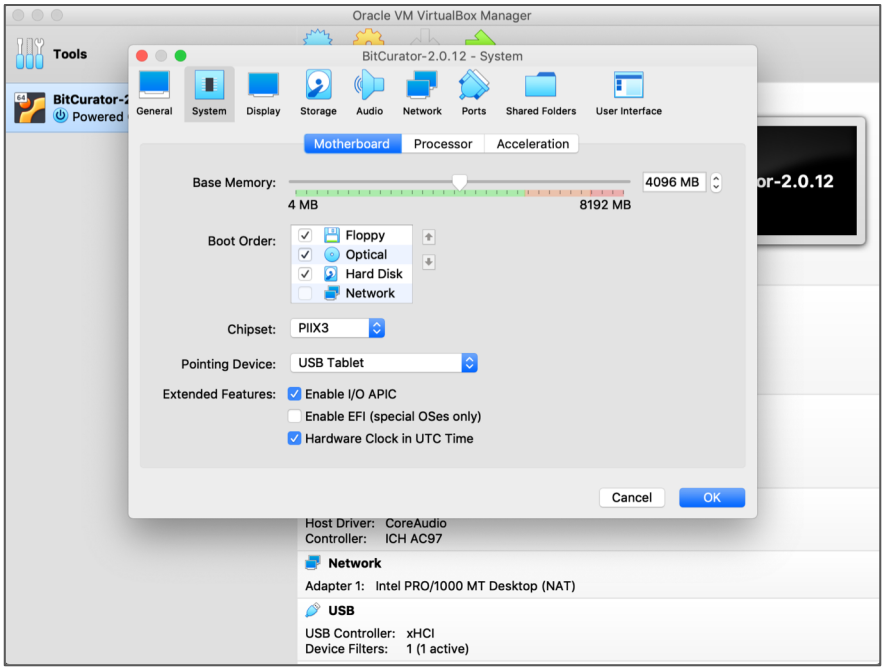
VirtualBox Manager: Configuring RAM and Processors

Click on the Settings icon, and select the system tab**. We recommend a minimum of 4096MB (4GB) RAM and 2 processors assigned to the VM. You may wish to change the RAM and number of processors depending on the hardware that you're running on. For best results, select the largest number in the "green" areas for each**.

Tip: You'll need 2 or more processors assigned for VirtualBox to support drag-and-drop.



Configuring the VM (VirtualBox in Windows)

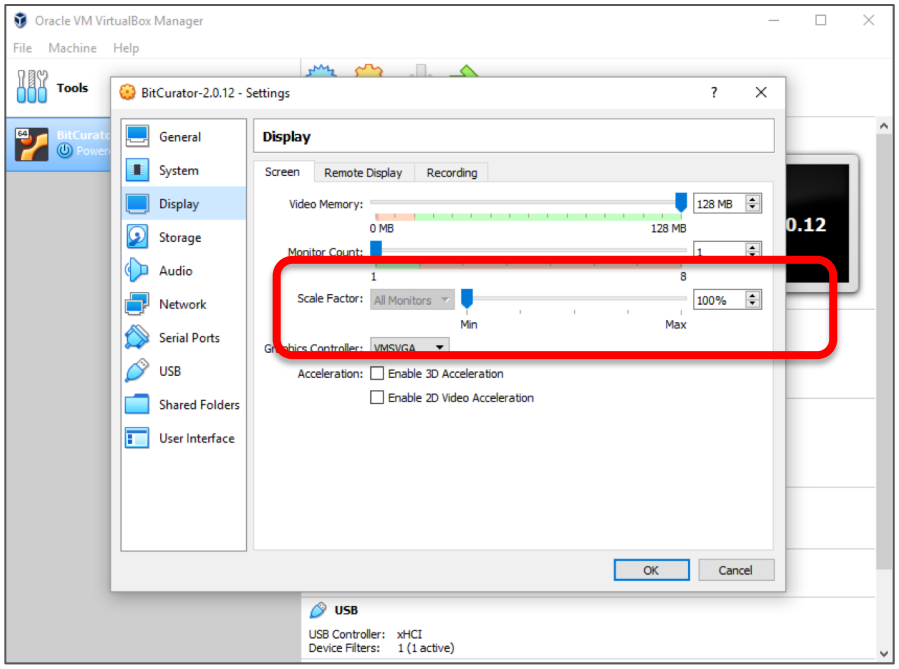


Configuring the VM (VirtualBox in macOS)

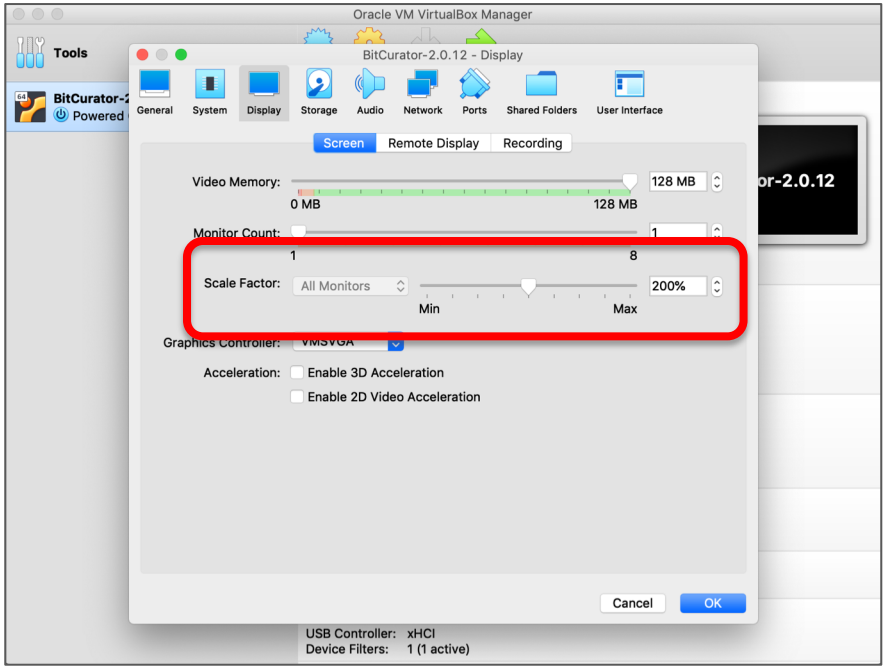
VirtualBox Manager: Configuring Video Scaling

Click on the Display icon. Note the “Scale Factor” setting. Depending on the monitor resolution of your host machine, you may need to adjust this.

Tip: A 100% scale factor will usually be appropriate for a 1080p (Full HD) screen. A 200% scale factor will usually be appropriate for a 4K screen.



Configuring the VM (VirtualBox in Windows)



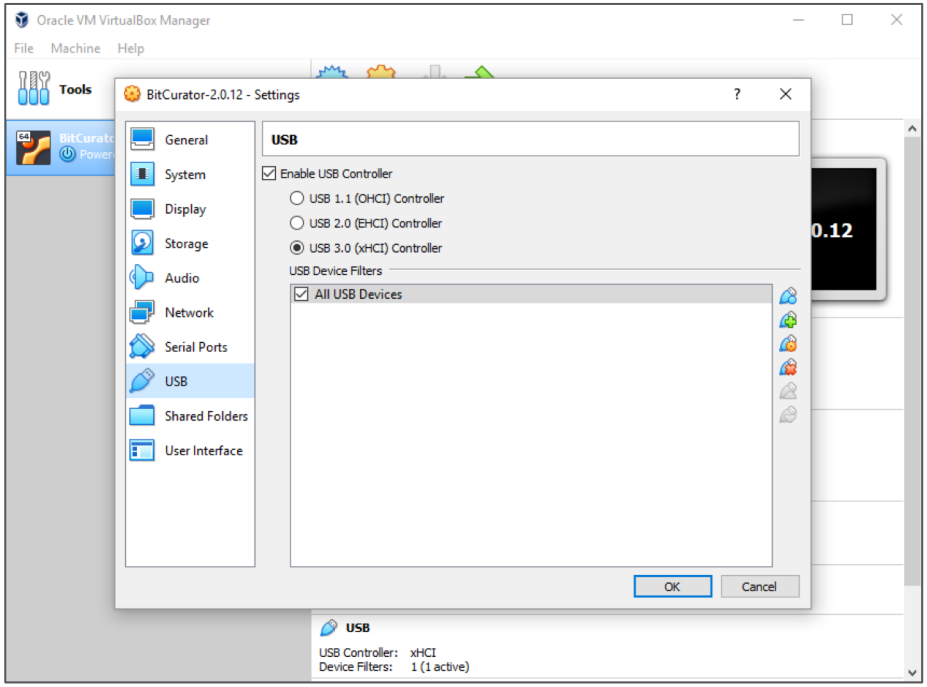
Configuring the VM (VirtualBox in macOS)

VirtualBox Manager: USB Device Capture

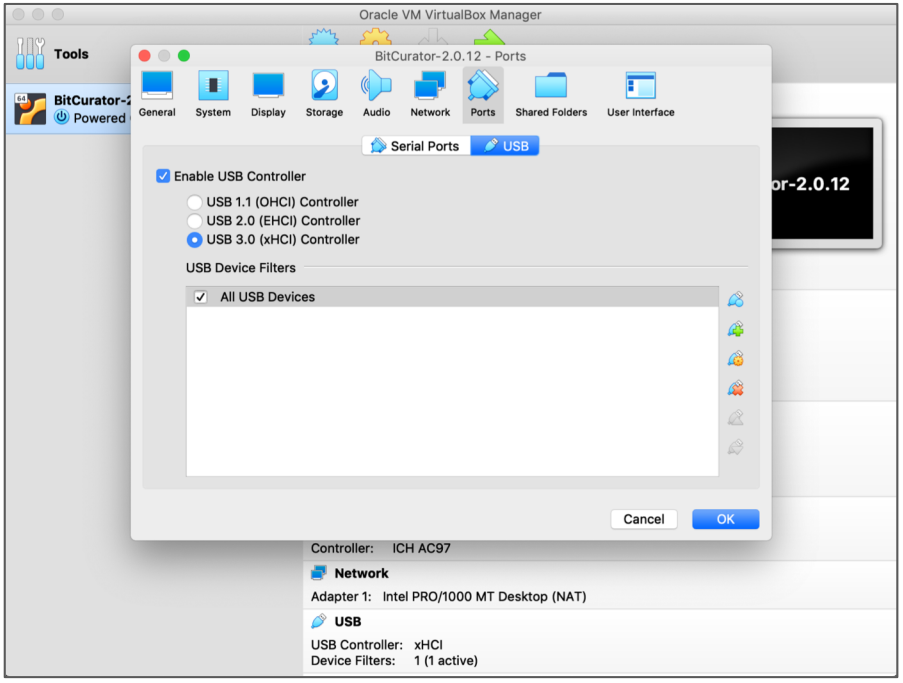
BitCurator depends on a VirtualBox device filter to capture USB devices. In OS X, you'll find this filter under the USB tab of the "Ports" icon under "Settings". In Windows, you'll find it in the "USB" tab in settings. You don't need to do anything here, unless you don't see an entry under "USB Device Filters".

If you don't see an entry, select the USB 3.0 radio button, and then create a new filter by clicking on the blue icon to the right of the list.

Tip: If you don't need access to USB devices (for example, if you plan to do analysis of existing born-digital data but no disk imaging), you can delete this filter.



USB filters (VirtualBox in Windows)

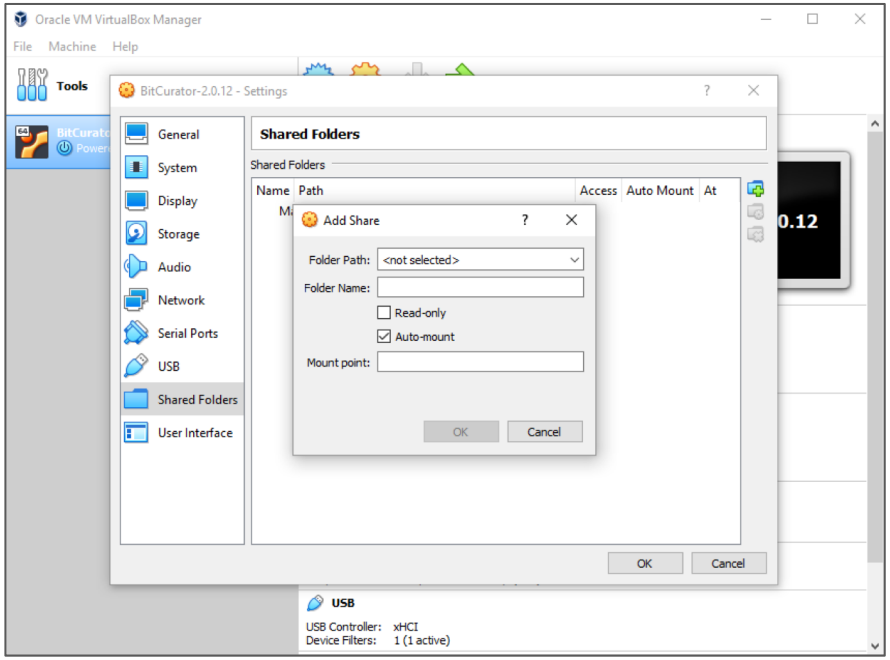


USB filters (VirtualBox in macOS)

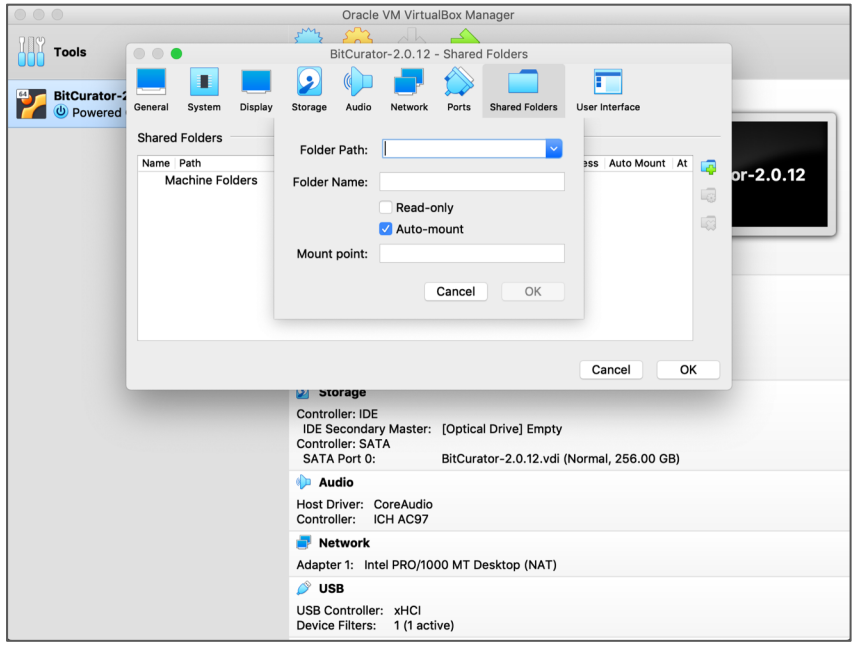
VirtualBox Manager: Shared Folders

If you wish to move processed materials **back to your host machine** from the BitCurator VM, you can set up a shared folder that both the host and the VM can write to. In the Shared Folders tab, click the folder with the green “plus” on it to choose a folder on your host machine to share.

Tip: Select “Automount” but not “Read Only”. When the machine is booted, the folder will appear in the “Shared Folders and Media” folder on the desktop in the VM.



Shared folders (VirtualBox in Windows)



Shared folders (VirtualBox in macOS)

Starting the BitCurator Environment (VM)

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.

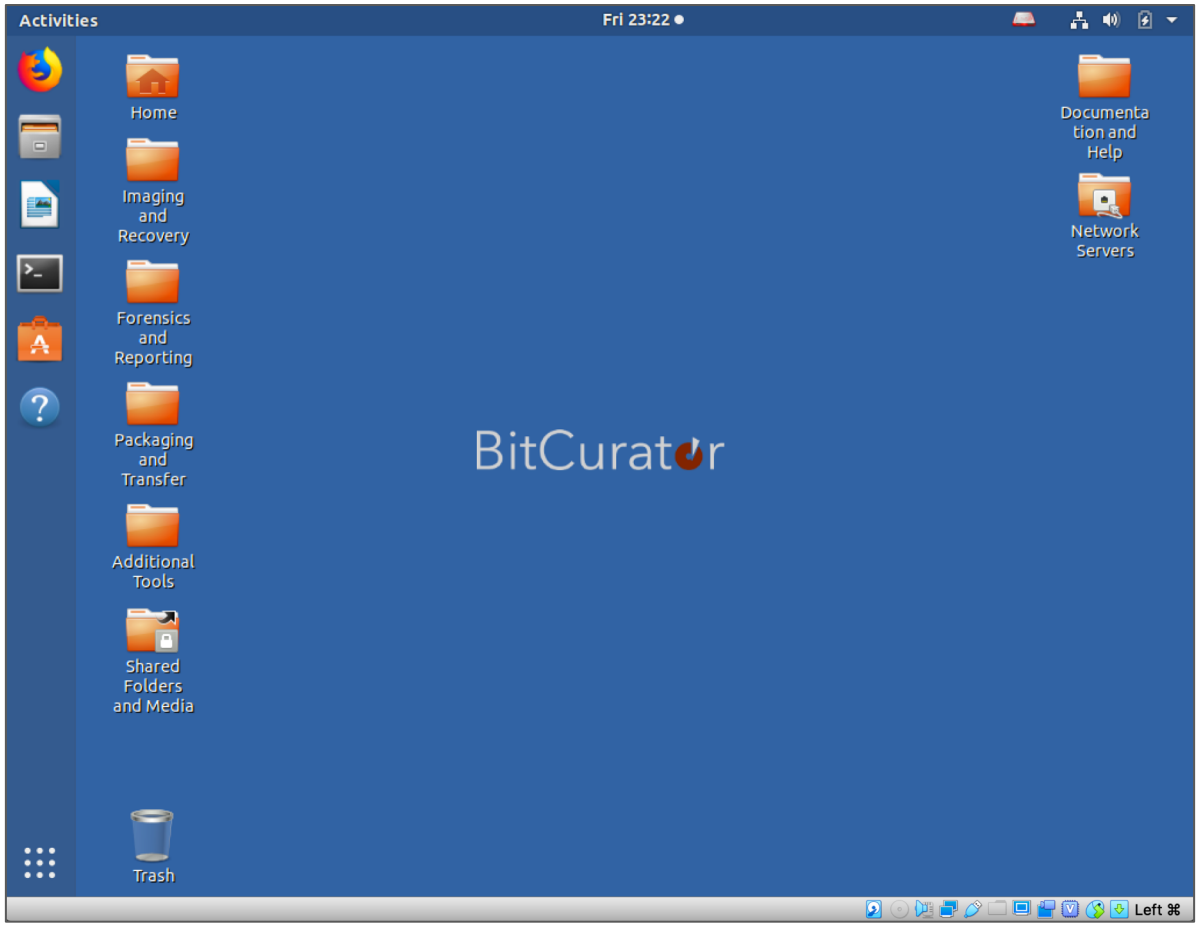
Tips:

If you encounter a menu screen on boot, simply hit the Return key to continue.

If you see an error message mentioning virtualization extensions, or "Intel VT-x", your host machine's BIOS does not have virtualization extensions enabled. You'll need to reboot your computer, holding down "Del" (or "Esc", or the "ThinkPad" button, depending on your machine). Once you're in the BIOS, locate the correct menu entry and enable the **Intel Virtualization Extensions.**

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 **https://wiki.bitcurator.net/).**

Starting the BitCurator Environment (VM)



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.

Getting Started With The Live ISO (Dedicated Install)

Getting Started With The Live ISO (Dedicated Install)

- Hardware:
  - Desktop or laptop with an Intel Core i5 or Core i7 processor (or AMD equivalent)
  - 16GB RAM recommended (8GB minimum)
  - For best performance, we recommend using an SSD (256GB or larger) when installing and running BitCurator on a dedicated machine.
- Software:
  - You will need to use an existing machine to write the ISO to a bootable medium (8GB or larger USB stick, preferably USB 3.0 or better).
  - Instructions for installation are on the following slides.

Creating a bootable USB drive (Windows and Mac)

1. Download the current BitCurator ISO image from: https://github.com/BitCurator/bitcurator-distro/wiki/Releases

2. To create a bootable USB drive using a Windows machine, follow the instructions at: https://www.ubuntu.com/download/desktop/create-a-usb-stick-on-windows

3. To create a bootable USB drive using a Mac, follow the instructions at: https://www.ubuntu.com/download/desktop/create-a-usb-stick-on-mac-osx

Installing BitCurator

- Once you have a bootable USB drive, power down the machine you wish to install BitCurator on, and plug the USB drive into a primary USB port (e.g. not a USB hub).

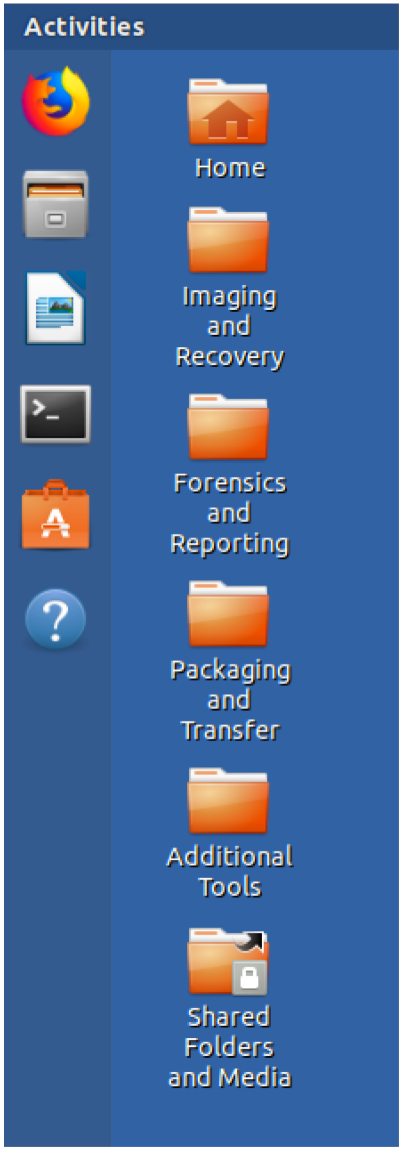
- Power on the machine. **On certain PCs, you may need to enter the BIOS to allow booting from USB drives**. Consult the documentation for your particular machine.

- Boot to the Live desktop, and then double-click on the Installer icon located on the desktop. You will be guided through the regular install procedure.

Tip: If your drive is appropriately partitioned, BitCurator can be installed alongside an existing Windows or Linux operating system. The installer will attempt to automatically identify whether an existing OS is present. Please consult the existing Ubuntu documentation regarding dual-installs for additional detail.

Essential Tools in the BitCurator Environment

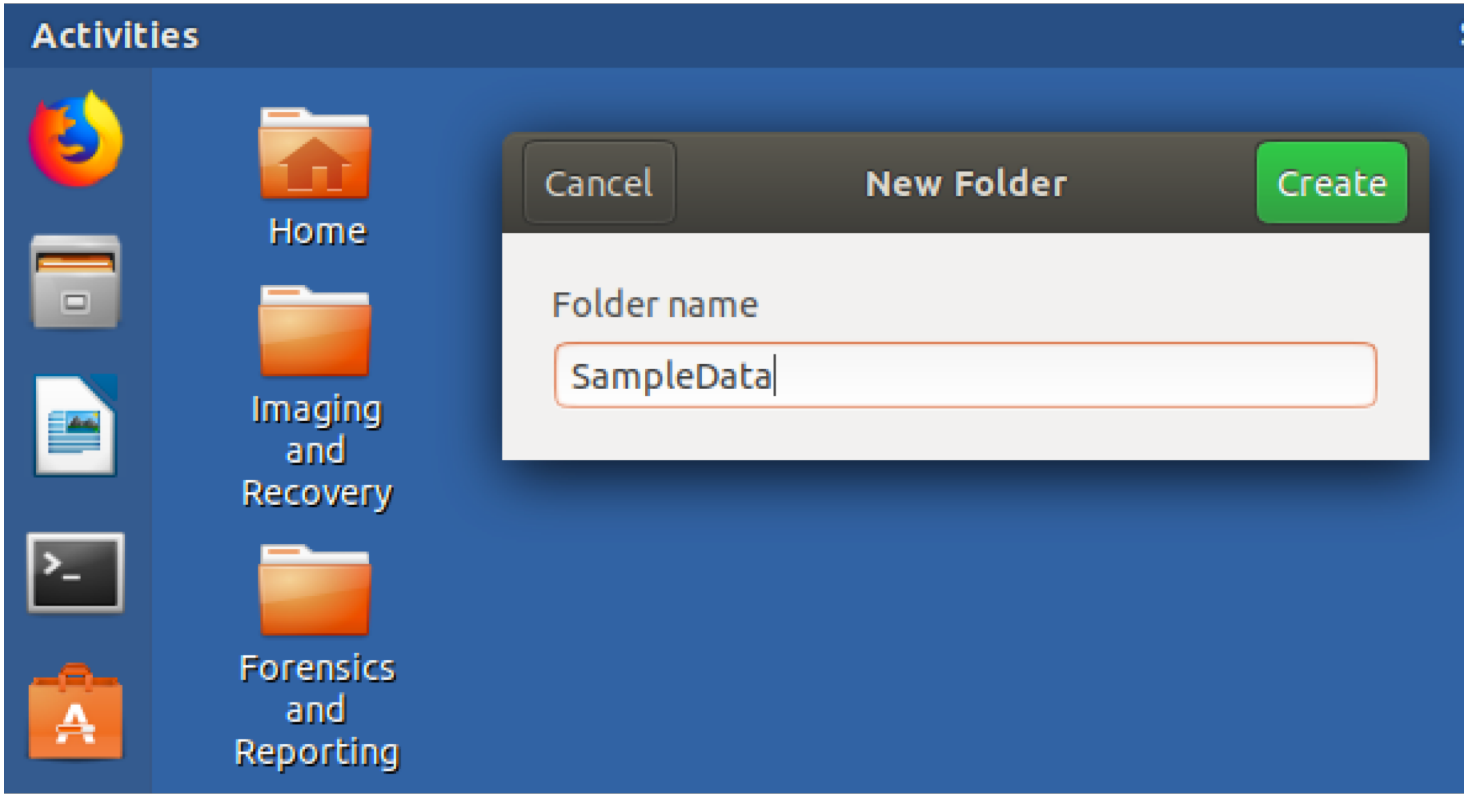
Essential Tools in the BitCurator Environment



The BitCurator Environment provides shortcuts to commonly used tools, organized into folders on the desktop:

- The **Imaging and Recovery** folder contains tools to create raw and forensically-packaged disk images from physical media.
- The **Forensics and Reporting** folder contains launchers for forensics tools, disk and file system analysis tools, and report generation utilities.
- The **Packaging and Transfer** folder points to transfer and accession tools including Bagger, Python-BagIt, and Grsync
- **Additional Tools** includes other useful software tools that may be used to inspect and process disk images and files.

Adding a Folder for Disk Images and Reports



Right-click anywhere on the desktop, and select **New Folder**. A dialog to create a folder will appear on the Desktop. Type in the name **SampleData**, and click **Create**. You will use this location to store the data produced in the steps described in the following slides.

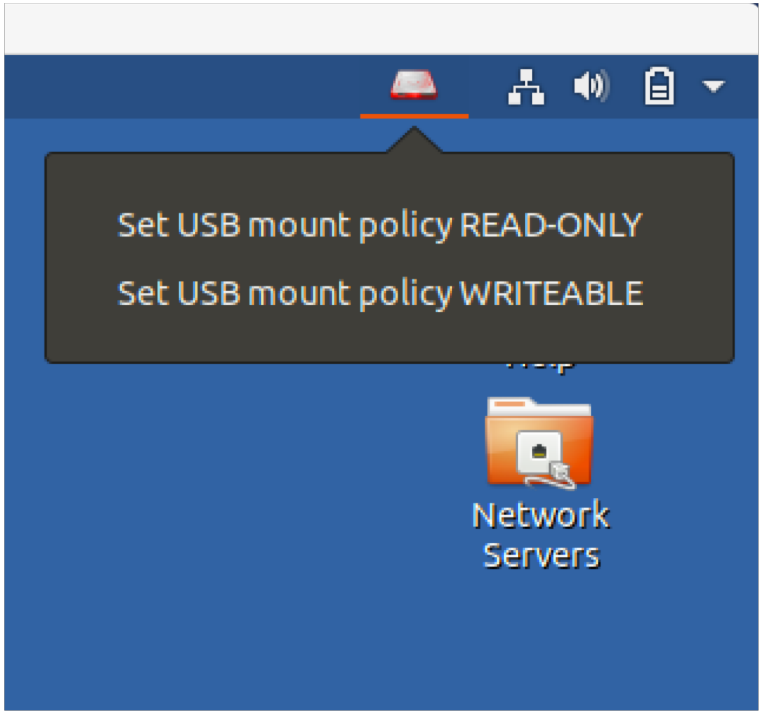
Preparing to Image Digital Media

The BitCurator environment has been customized to prevent the device management service from automatically mounting **any** media. You can **click the "Files" icon** in the dock on the left hand side to view attached devices. Clicking a device will mount it read-write in the default state.

BitCurator includes an **optional service enforcing USB drive mounts read-only**. This service is disabled by default; **any USB devices you attach and mount will be writable** unless you change the policy.

You can change the system-wide USB mount policy to enforce read-only mounts by selecting the **red disk icon** in the indicator bar at the top right and selecting **Set USB Mount Policy Read-Only.** Changing the system policy will not affect the status of any previously mounted devices.

We recommend that you do not rely on the software read-only enforcement mechanism in production situations. Always use a hardware write blocker if possible!

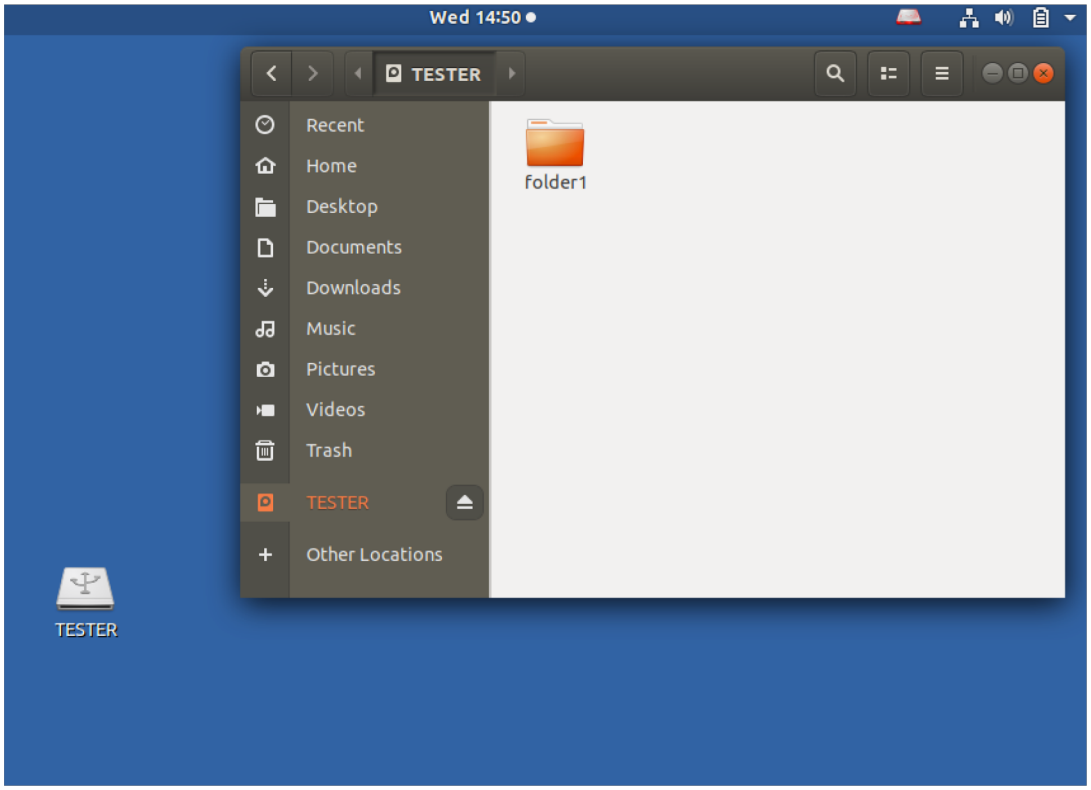


Understanding Mount Behavior (Read-Write Mode)

Read-Write (Default) Mode:
When the disk icon in the top right is red, USB mounting is performed in read-write mode. Clicking on **Files** in the dock and clicking the device name (in this case **"TESTER"**) will perform a normal read-write mount.

The contents of the device (if mountable) will appear in the **Files** window itself. Or, browse by by double clicking the disk icon that appears on the Desktop.

Click the eject button, or right-click on the disk icon on the desktop and select "Safely Remove Drive" to unmount.



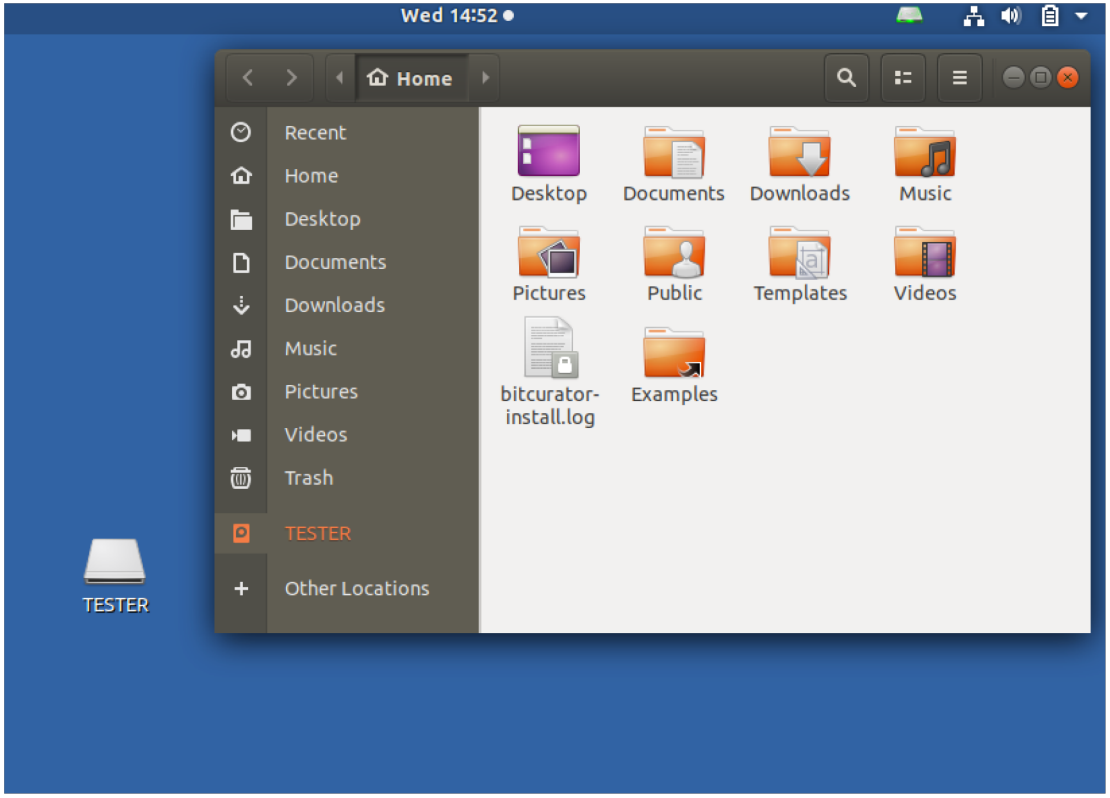
Understanding Mount Behavior (Read-Only Mode)

Read-Only Mode:

When the disk icon in the top right is green, USB mounting is performed in read-only mode. Clicking on **Files** in the dock and clicking the device name (in this case **"TESTER"**) will mount the file system via a loop device, a pseudo-device used to enforce the read-only mount.

Because the read-only mount is created using this loop device, the contents of the mount will **NOT** appear in the **Files** window itself. **You must double-click the disk icon that appears on the Desktop to browse any available file system**.

No eject button is present for this type of mount. **Right-click on the disk icon on the desktop and select "Unmount" to unmount.**

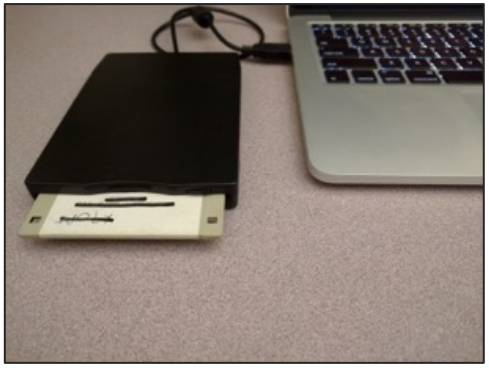


Getting Ready to Image Digital Media

This example* will use an external USB flash drive connected to a laptop running BitCurator in a VirtualBox VM.

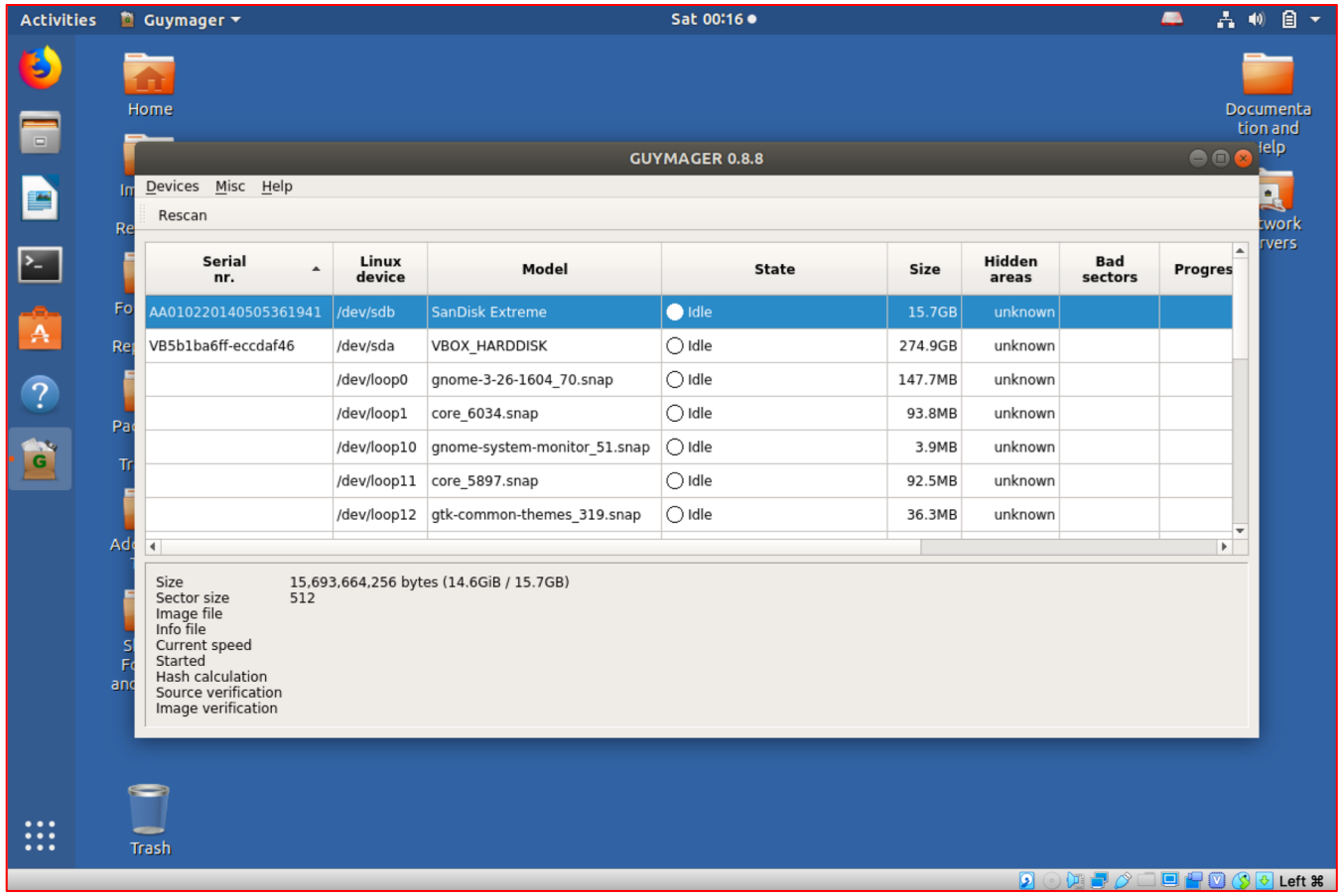
With the VM is started up, the physical device will be automatically captured when plugged in. Note: **the BitCurator environment will not automatically mount any identified file systems.**

Mounting the device is not required to image it. However, if you wish to browse the contents of the file system prior to mounting, you must click on the **Files** icon in the dock (left hand side of the screen), and click the name of the indicated volume on the device to mount. **If you are not using a hardware write blocker, or if the USB device read-only policy is not enabled, the device may be writeable!**



* The process from this point on will be effectively the same whether you are working with data from a CD, a floppy, a hard disk, or another device.

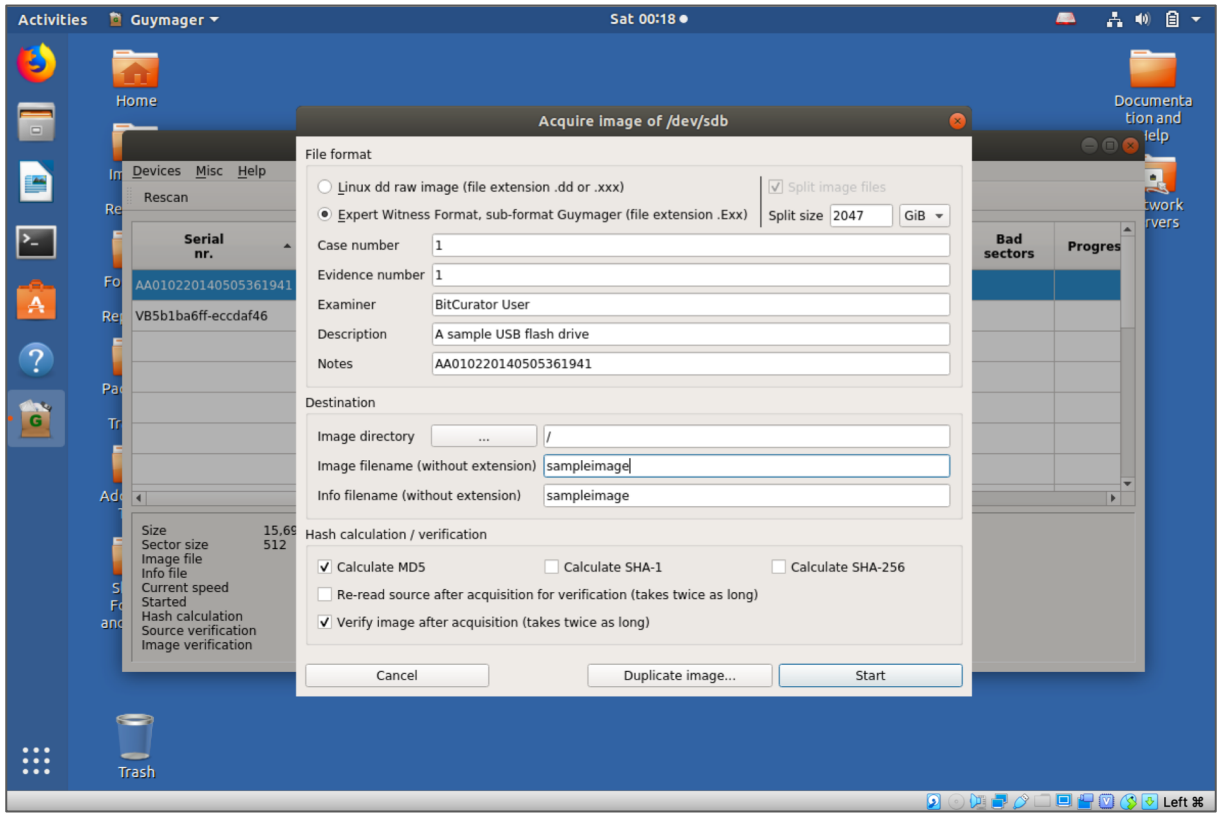
Imaging the Device



Once the device is attached, a disk icon should appear in the menu bar on the left. The device has not been mounted; this simply indicates the device is recognized by the system.

Double-click on **Imaging and Recovery** on the Desktop, and then double-click on **Guymager**. In this example, the USB flash drive is selected in the picture above.

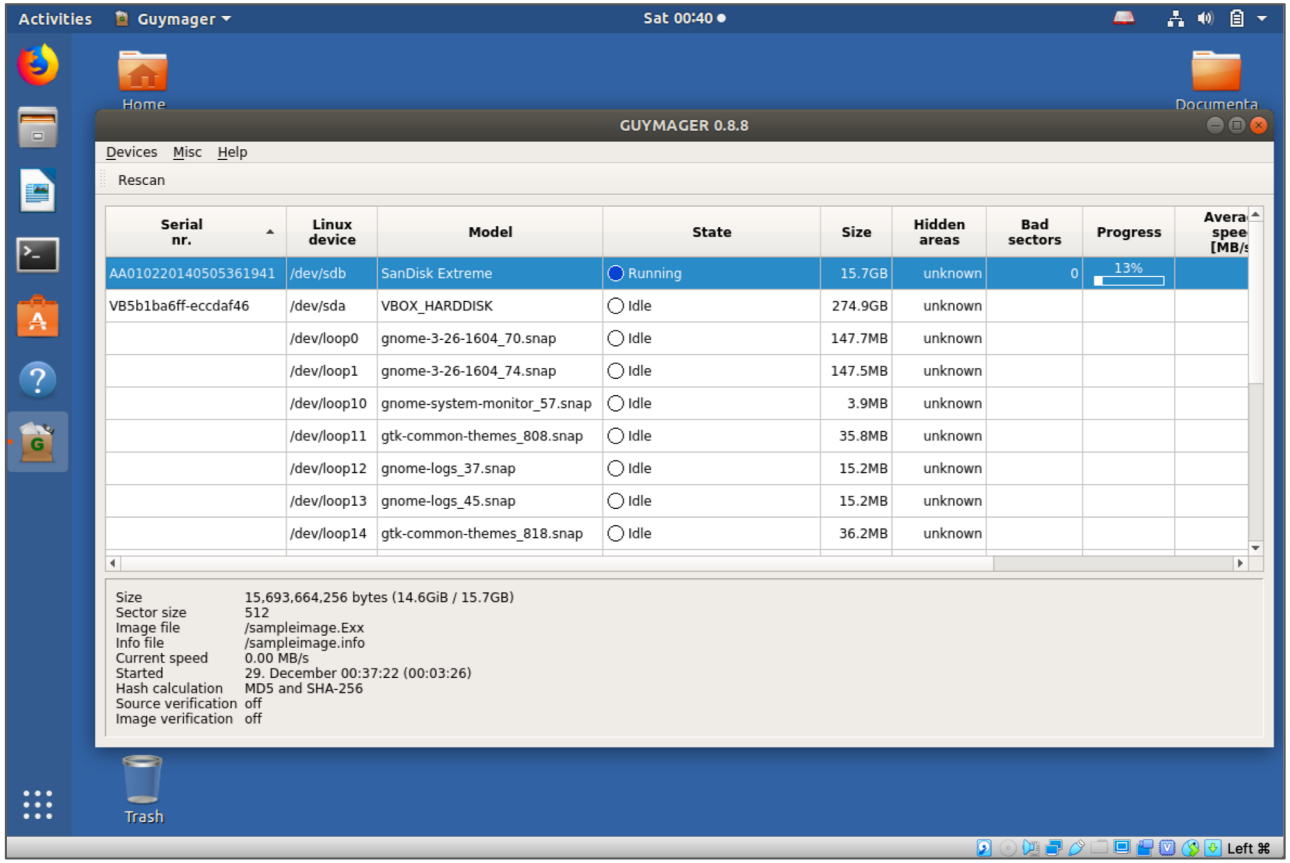
Entering Imaging Metadata



Right click on the device, and select **Acquire image**. This disk image will be acquired using the Expert Witness Format (the second option at the top). The five metadata fields starting with **Case number** are optional. Don't forget to select the directory you made on the desktop in the **Image directory** field. Finally, provide a name for the image. Then click **Start**.

Tip: Guymager will split images into 2GB segments by default. You may wish to change the "Split size" setting to something larger.

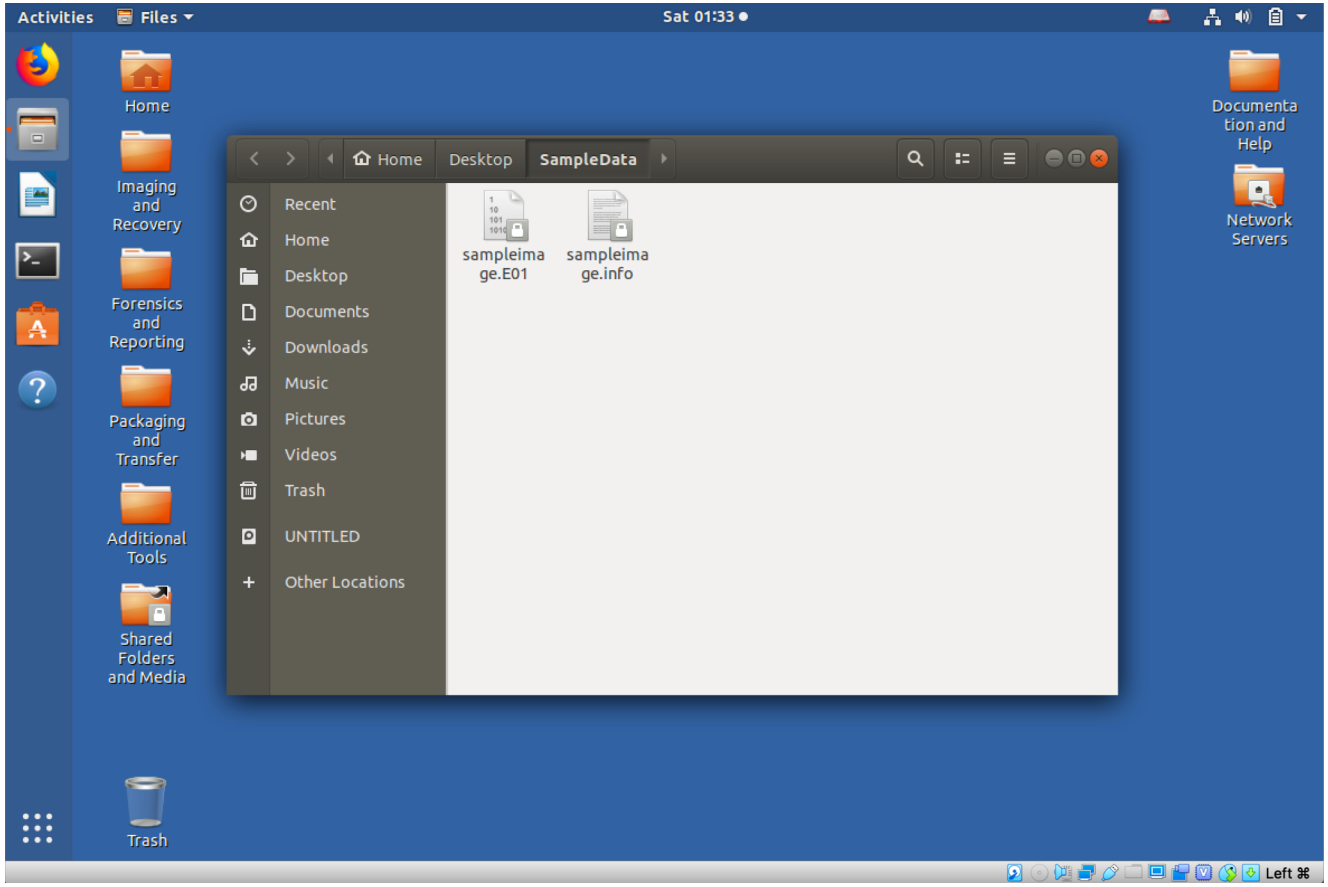
Acquiring the Disk Image



You will see the main dialog state change to **Running**. When the acquisition finishes, you will see an **OK** message in the State column.

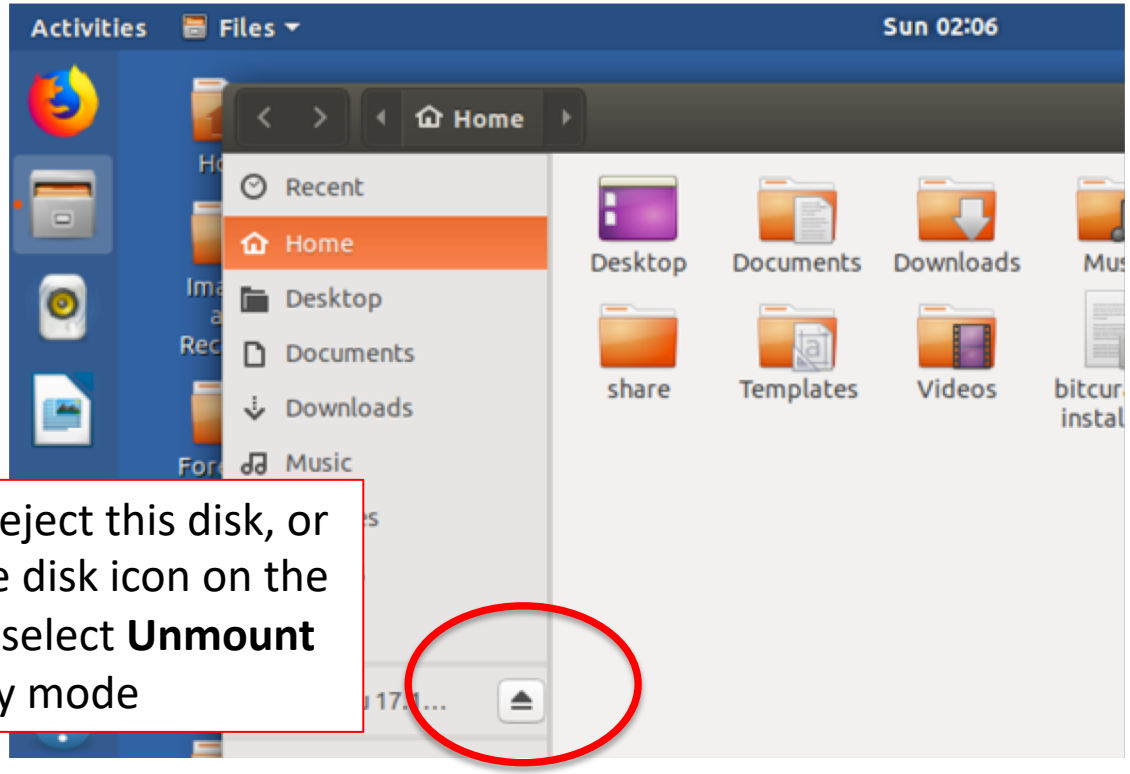
Tip: The BitCurator environment runs at a resolution of 1024x768 by default. If you can't see the whole dialog, increase the size of your VM window by dragging a side or corner. The desktop should adjust automatically.

Examining the Image



Quit Guymager, open up the SampleData directory on the desktop, and observe the two files that have been produced: the **.E01** image file, and a **.info** file with log information produced by Guymager.

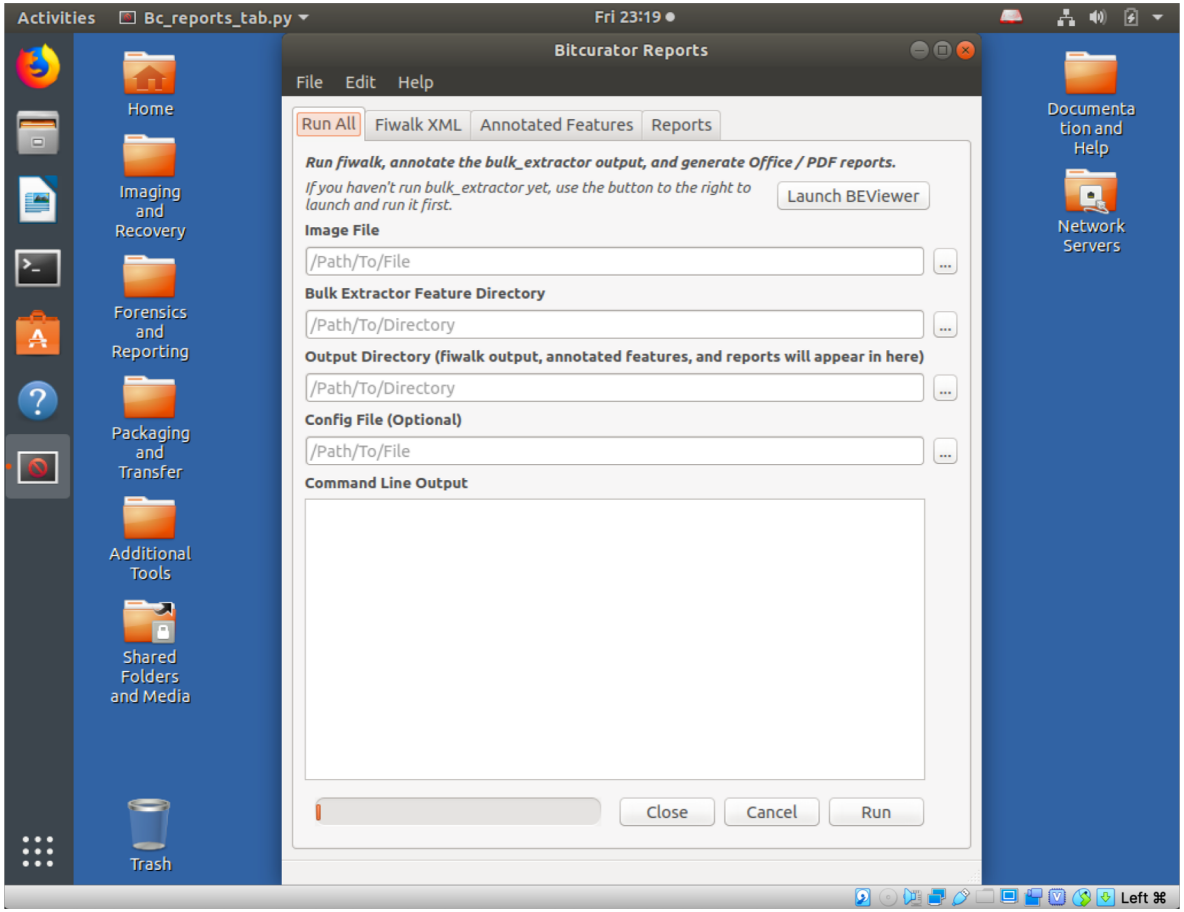
Safely removing a device from the system



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. **When in read-write mode**, click the "Files" icon in the dock, and in the new dialog that appears, click the eject button for the appropriate disk. **When in read-only mode,** right-click the disk icon on the Desktop and select **Unmount**.

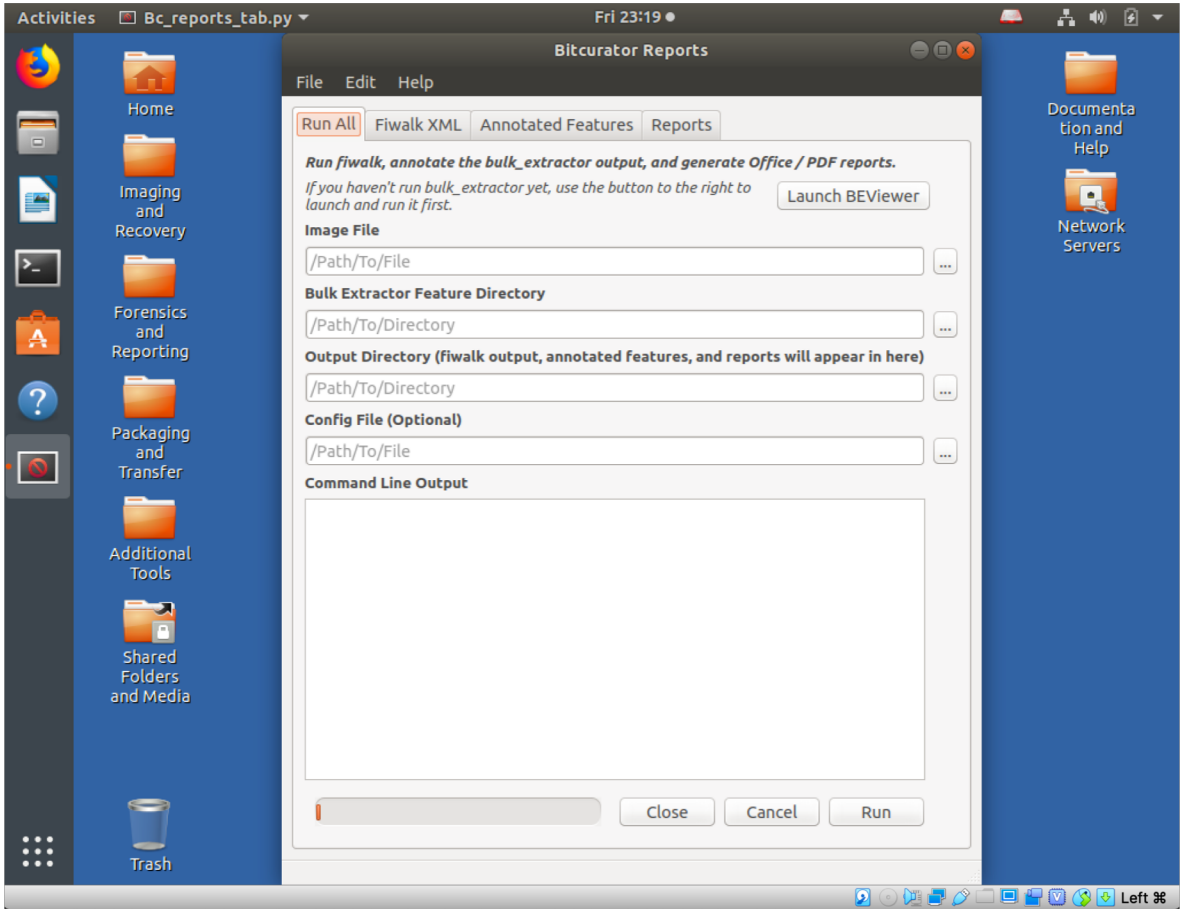
Tip: Depending on Ubuntu configuration, the disk may also appear in the dock. You can right-click on any disk icon and select "Safely Remove Drive" or "Unmount" to eject it, depending on the current mode.

Processing the file system, carving data, and generating reports



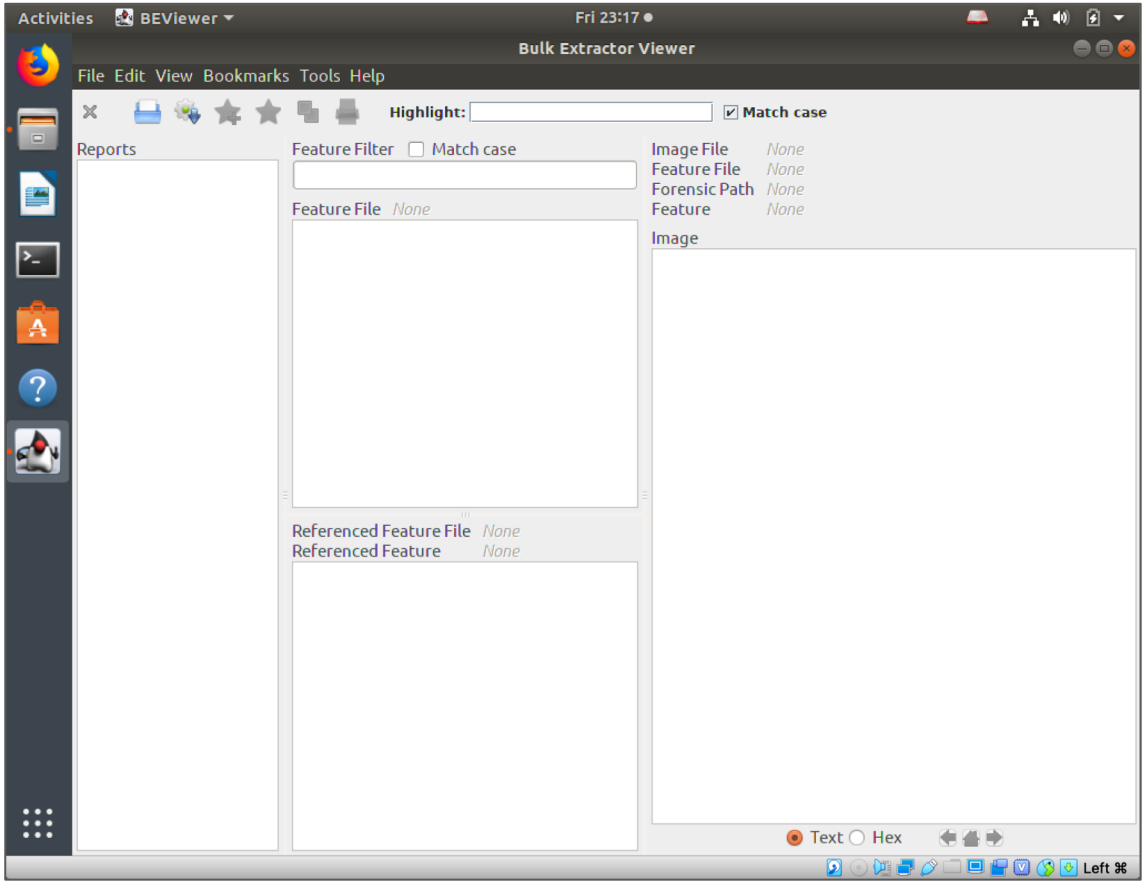
Double-click on the **Forensics and Reporting** folder, and then double click on the **BitCurator Reporting Tool** launcher. You'll see a window pop up that should match the picture shown above.

Processing the file system, carving data, and generating reports



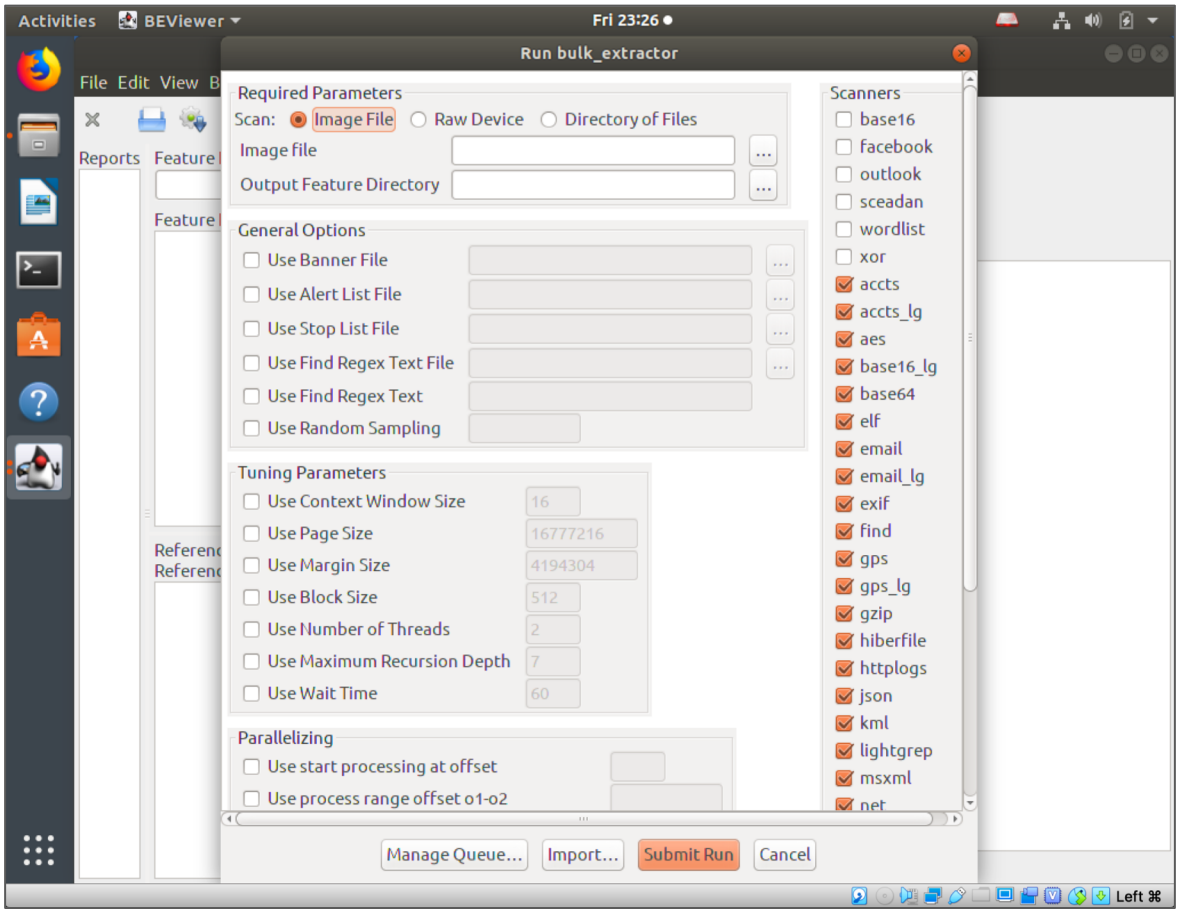
The **Run All** tab will allow you to carve the raw disk contents for features of interest, generate a DFXML listing of the file system hierarchy, match features to files within the file system, and generate high-level reports. Click on **Launch BEViewer** to run **bulk_extractor** before proceeding.

Generating Feature Reports with bulk_extractor



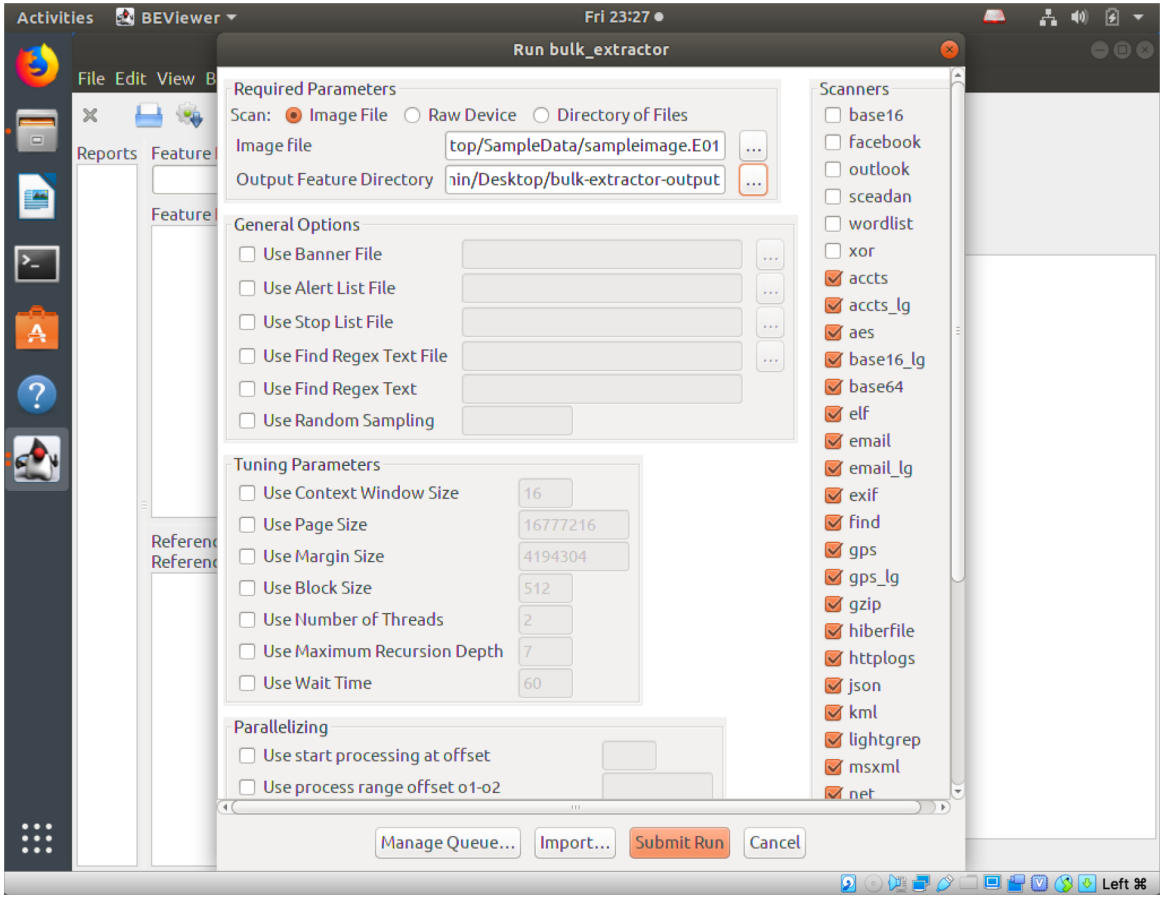
BEViewer is the graphical front-end to **bulk_extractor**. Together, these tools allow you to identify various features of interest contained within the bitstream extracted from the source media, such as SSNs, email addresses, EXIF metadata, and others.

Generating Feature Reports with bulk_extractor



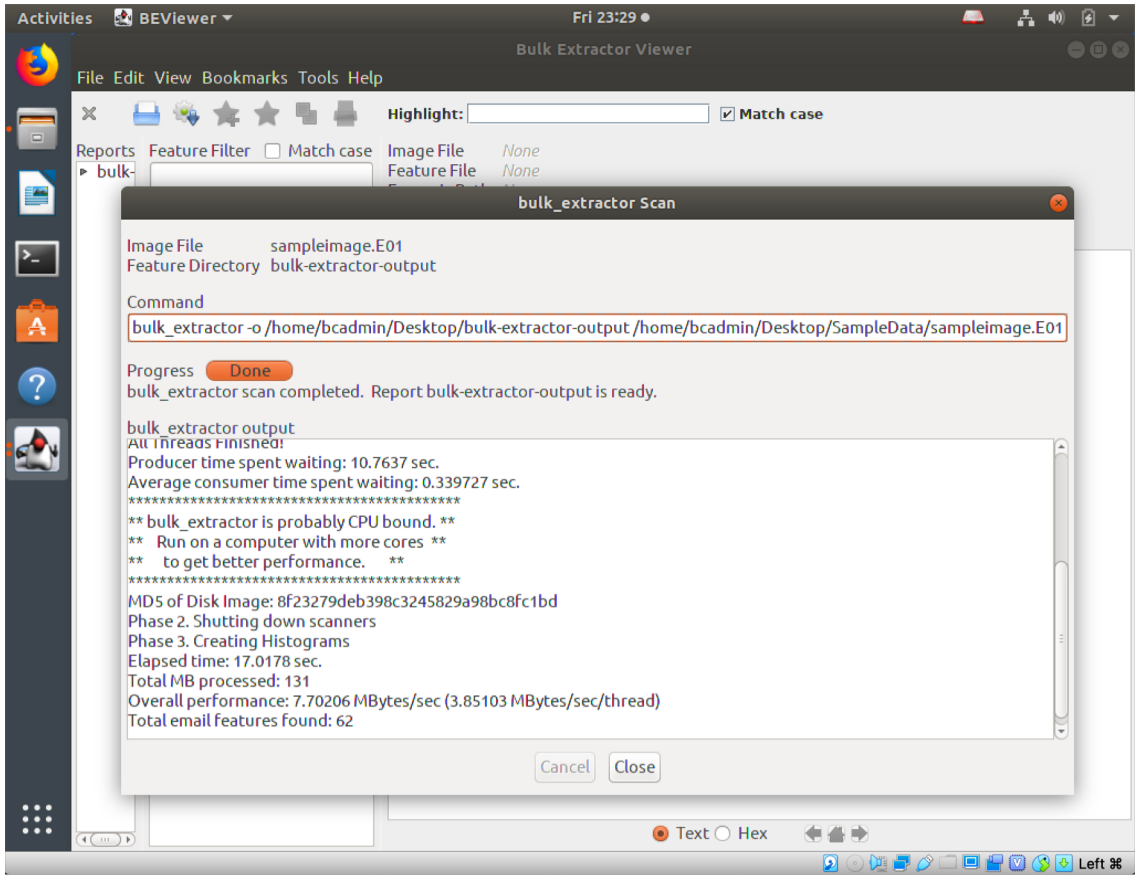
Click on the **Tools** menu in the top of the window, and select **Run bulk_extractor**. This will bring up a dialog that allows you to select which scanners to run, and where to generate the report directory.

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 you generated earlier and tell bulk_extractor to output the report in a new directory **bulk-extractor-output**, within the SampleData directory on the desktop.

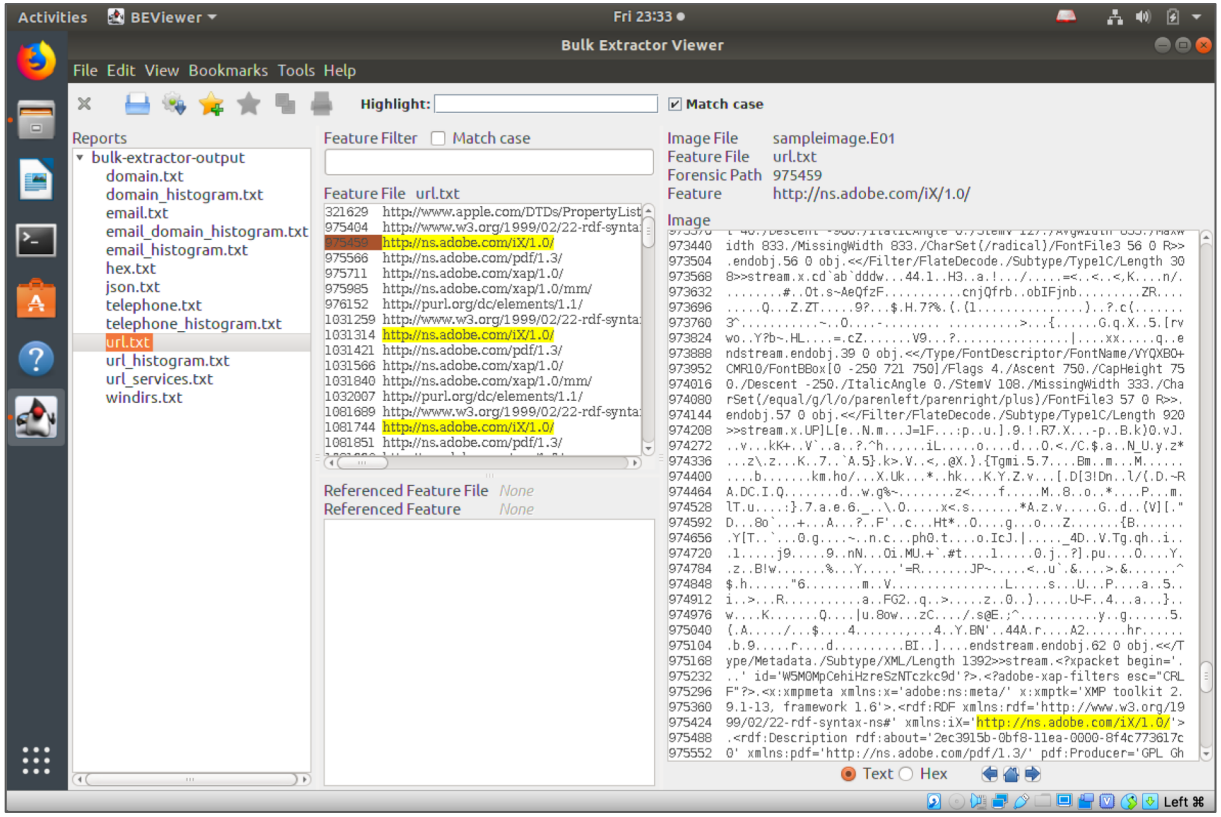
Generating Feature Reports with bulk_extractor



Click on **Submit Run** 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!

Tip: Additional processors assigned to the VM will improve performance.

Viewing the bulk_extractor Report

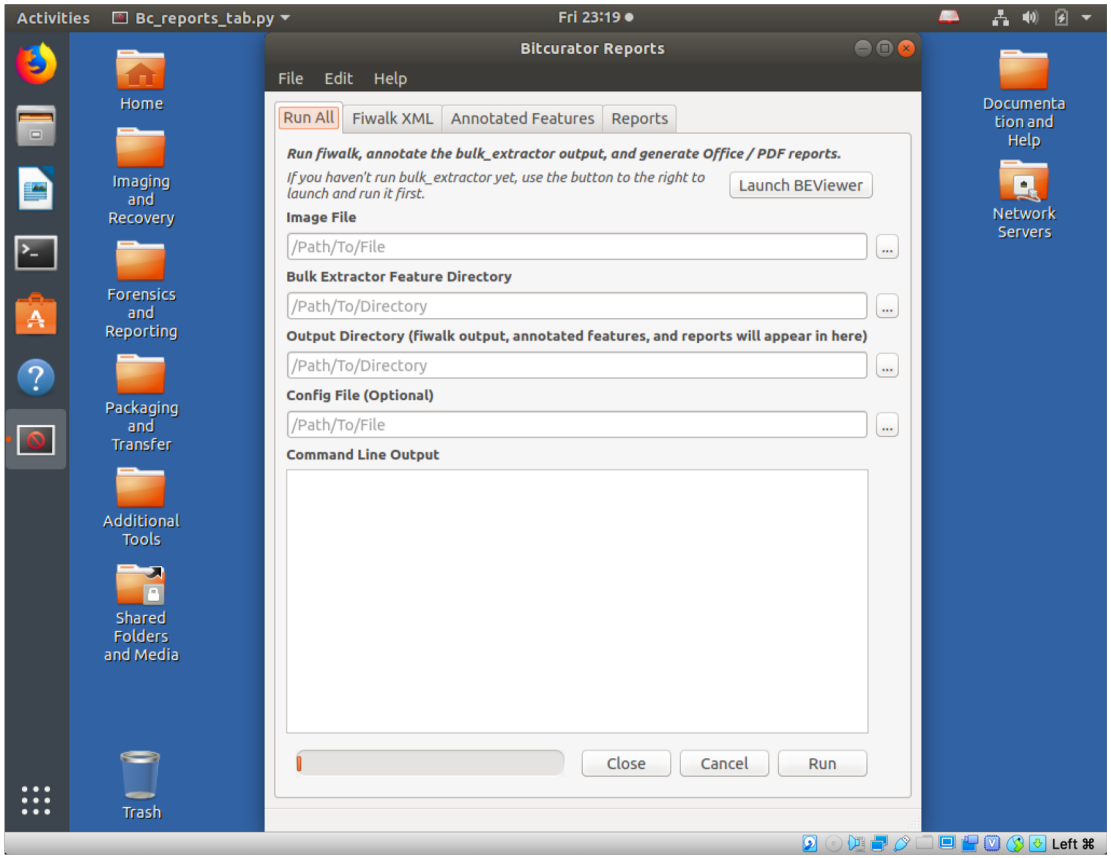


Once the process has completed, the report directory will be available in the relevant location (in our case, the directory "bulk-extractor-output" within SampleData). The features identified can also be viewed in the main Bulk Extractor Viewer window, by clicking on the report name in the "Reports" subwindow.

Tip: For the small disk image shown here, relatively few of the possible reports are shown. Your list may include a range of additional reports.

Tip: The next time you run bulk_extractor viewer (BEViewer), existing reports should remain available in the left hand column. If you need to open a report you created in a previous run, simply select "Open Report" in the "File" menu, and navigate to the "report.xml" file in the appropriate bulk_extractor output directory.

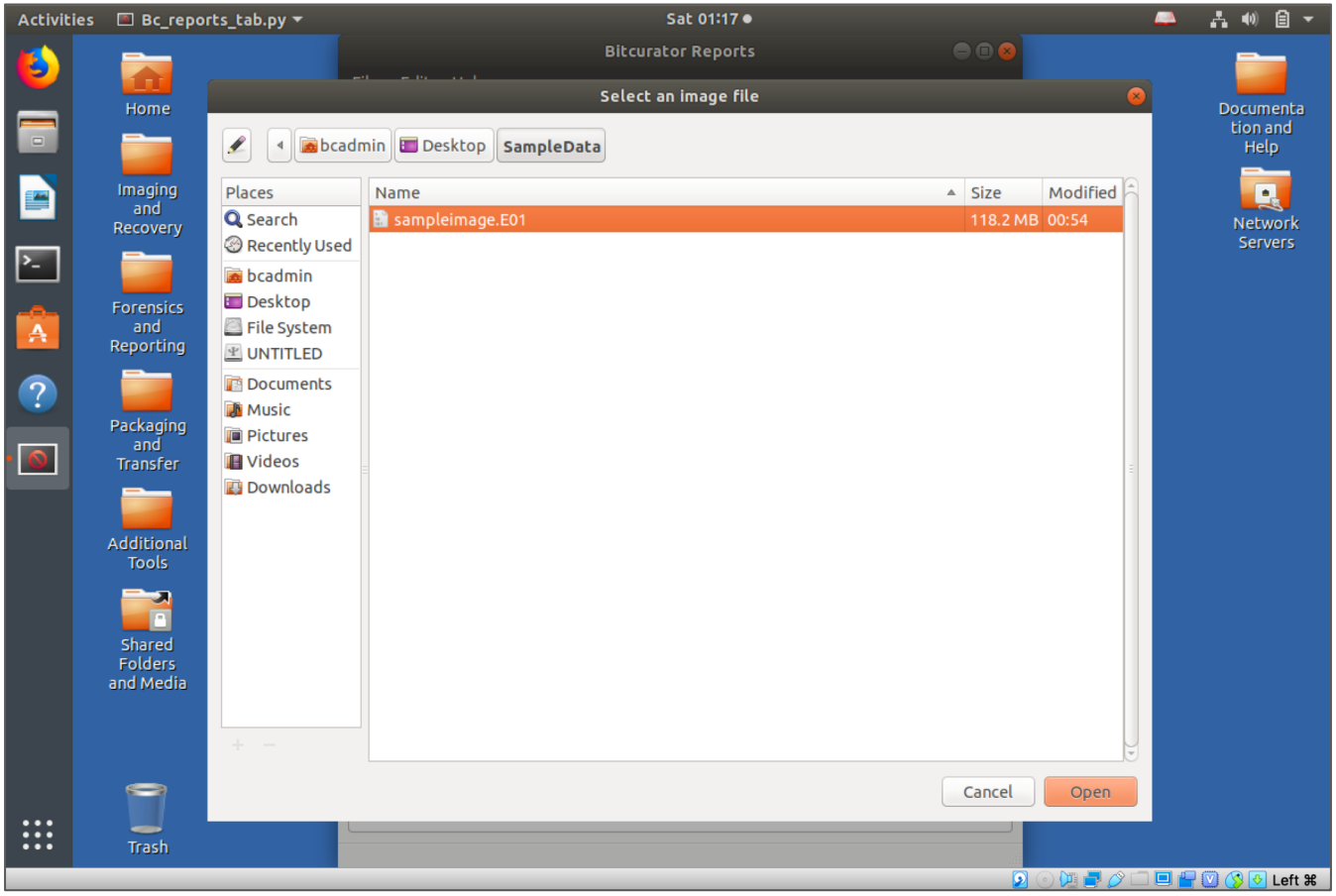
Processing the file system, carving data, and generating reports



With the **bulk_extractor** run completed, the remaining entries in the **Run All** tab in the BitCurator Reporting tool allow us to automate the process of (1) running **fiwalk**, (2) running the annotation tool to link bulk_extractor features to files within the file system, and (3) generating the BitCurator reports. We'll step through the process in the next few slides.

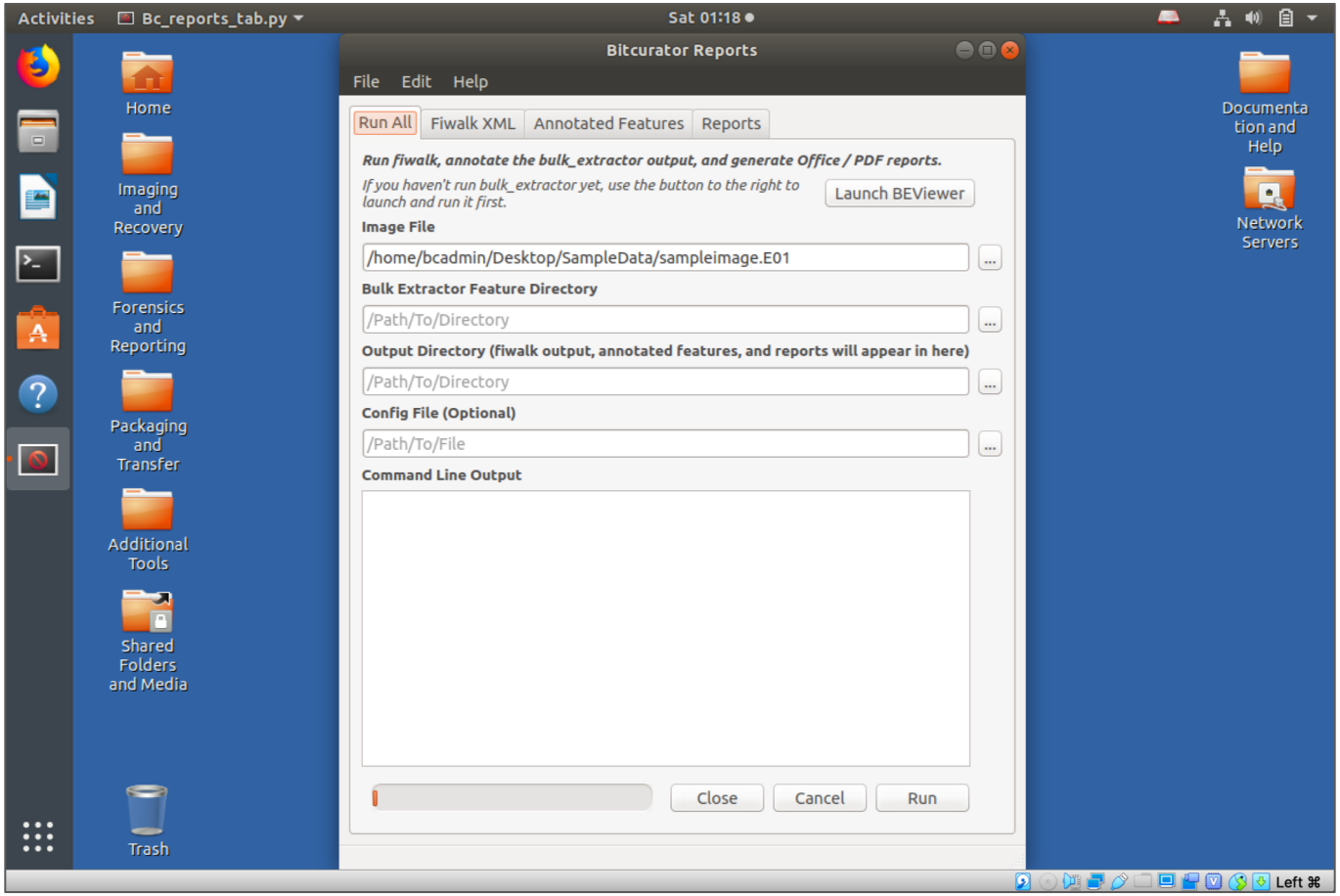
Tip: Appendix A shows how to run these tools individually using the other tabs.

Processing the file system, carving data, and generating reports



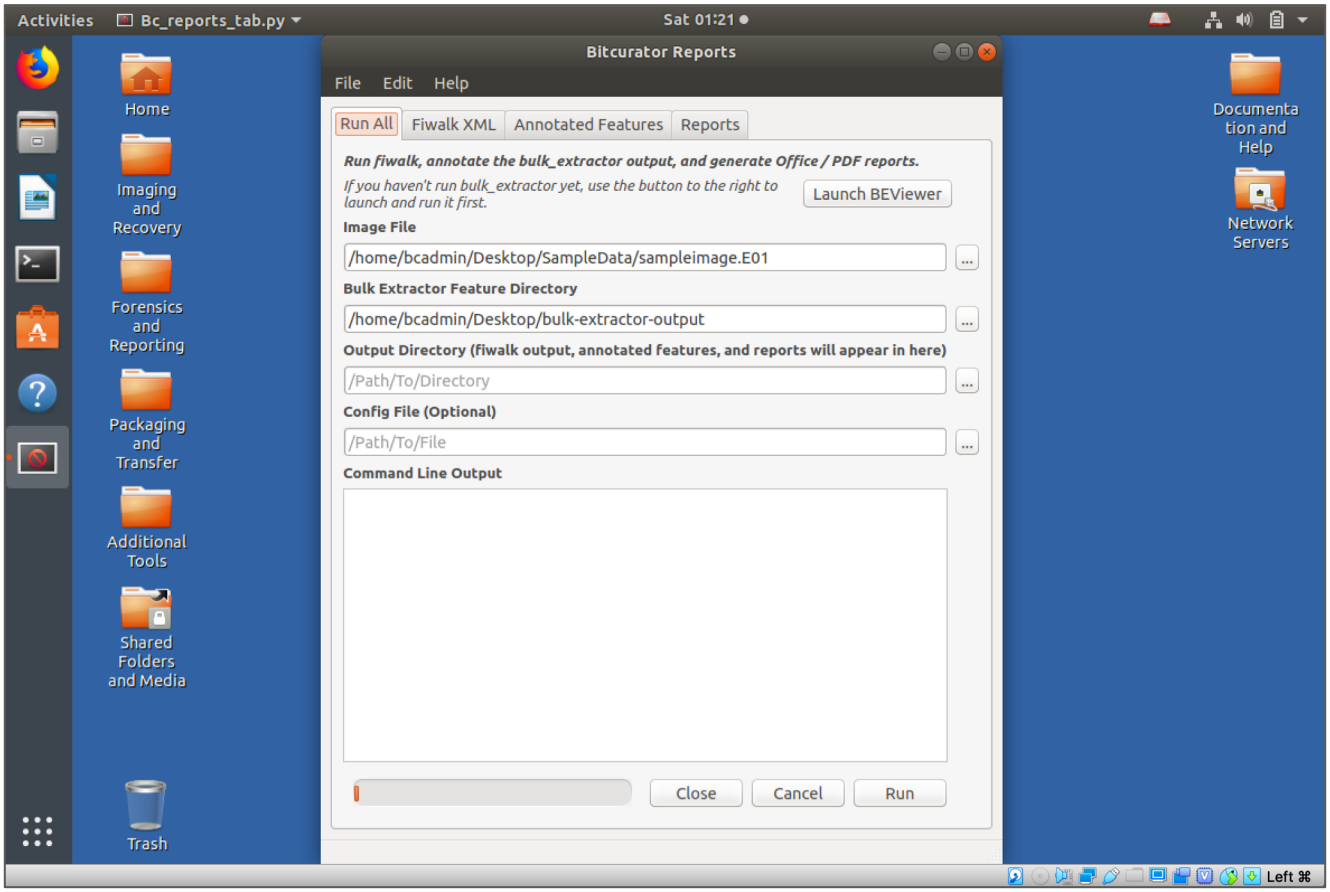
Click on the box with three dots next to the “Image File” entry, and navigate to the sample image (sampleimage.E01) we created in our SampleData directory on the Desktop earlier.

Processing the file system, carving data, and generating reports



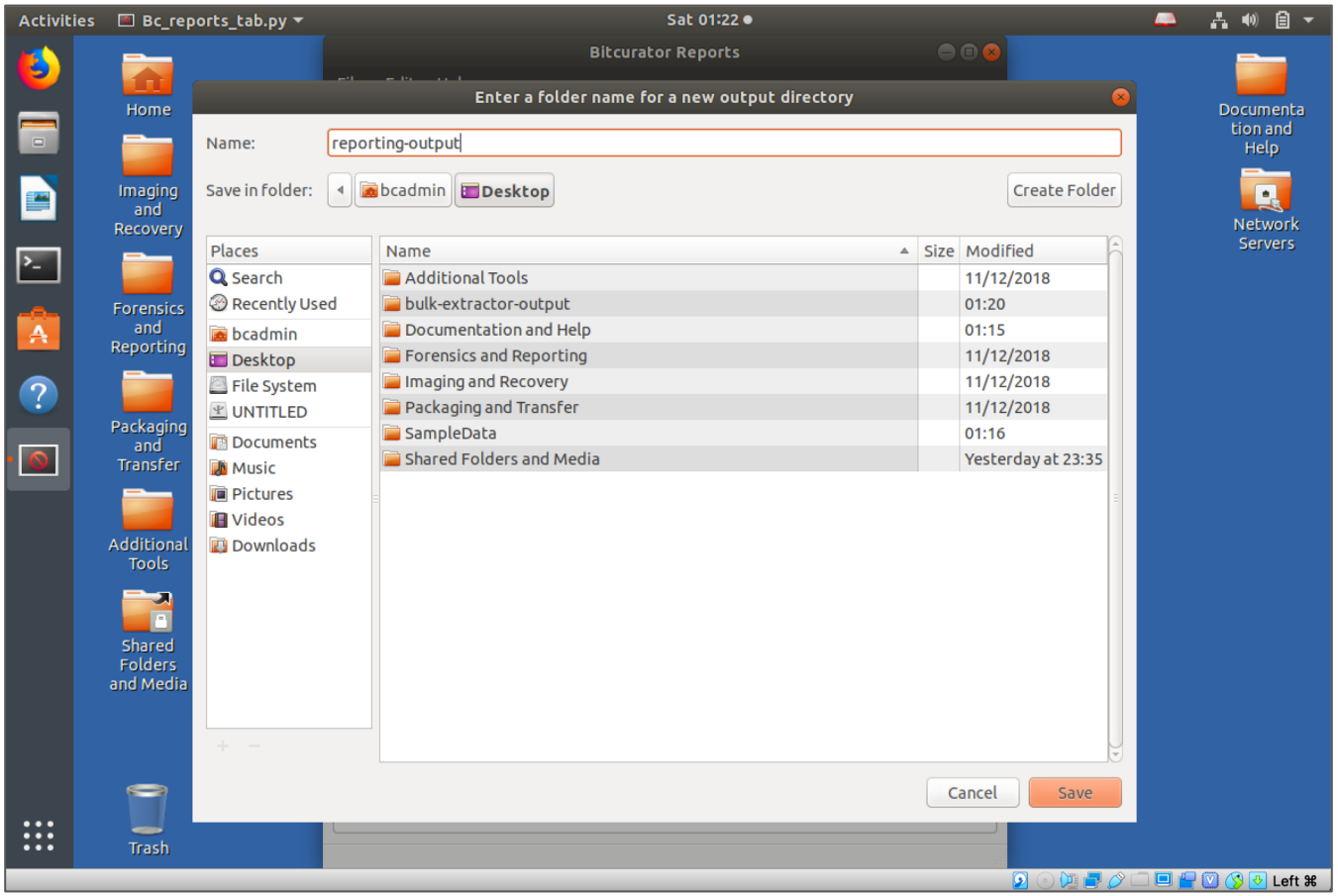
The image file you selected should now appear under the “Image File” entry.

Processing the file system, carving data, and generating reports



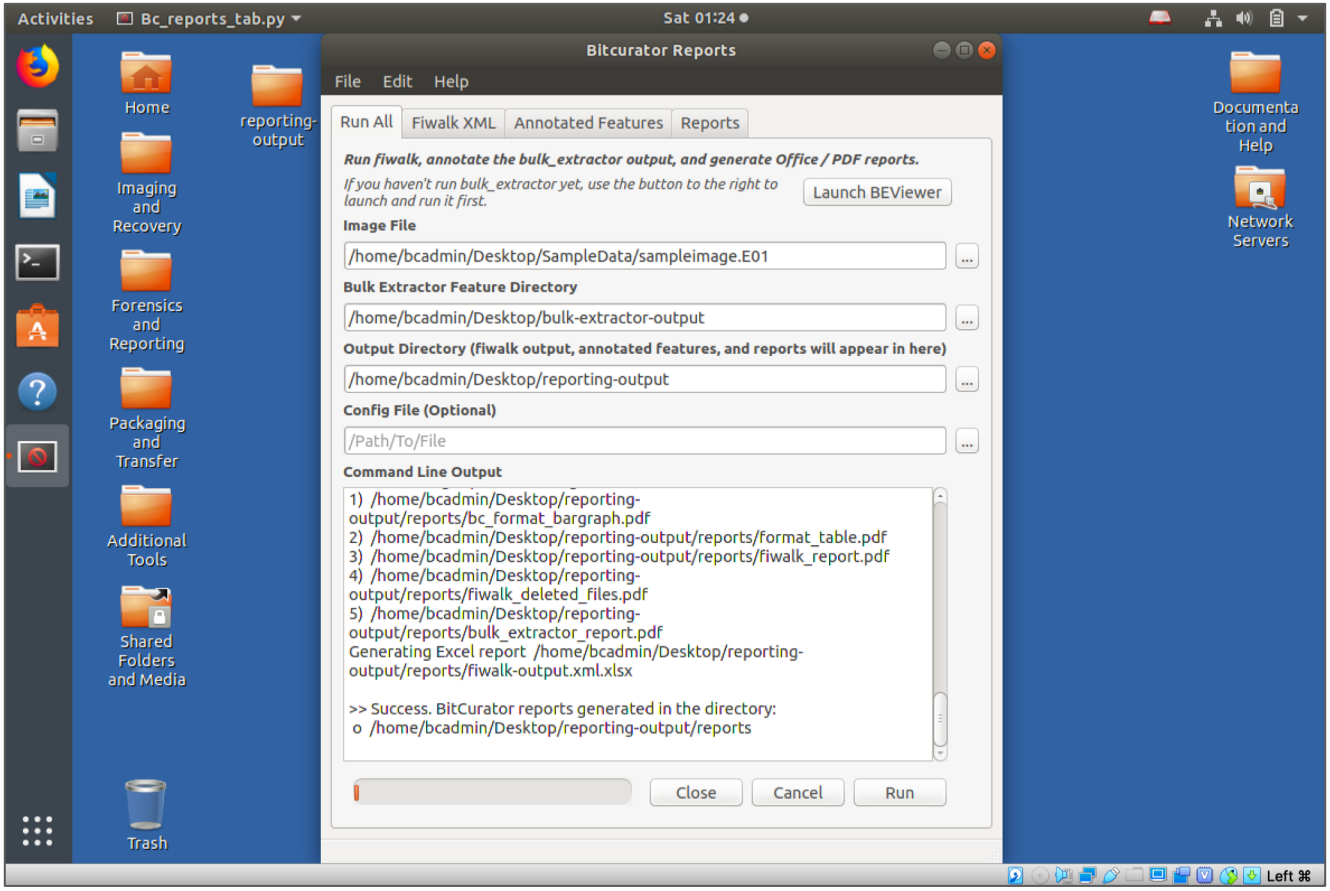
Follow the same process for the “Bulk Extractor Feature Directory” entry. We previously created the “bulk-extractor-output” directory within the “SampleData” directory on the desktop.

Processing the file system, carving data, and generating reports



Finally, assign an output directory for the reports that will be generated. **Note that you should not click "Create Folder".** Simply navigate to the desired base directory (in this case, **/home/bcadmin/Desktop/SampleData**) and type in the name of a new folder 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 the process to complete on larger images.

Tip: If you’re analyzing a raw disk image rather than a forensically-packaged one, BitCurator will **not** currently produce PREMIS output.

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.

bc_format_bargraph.pdf – file format histogram

bulk_extractor_report.pdf – high-level overview of feature locations on disk

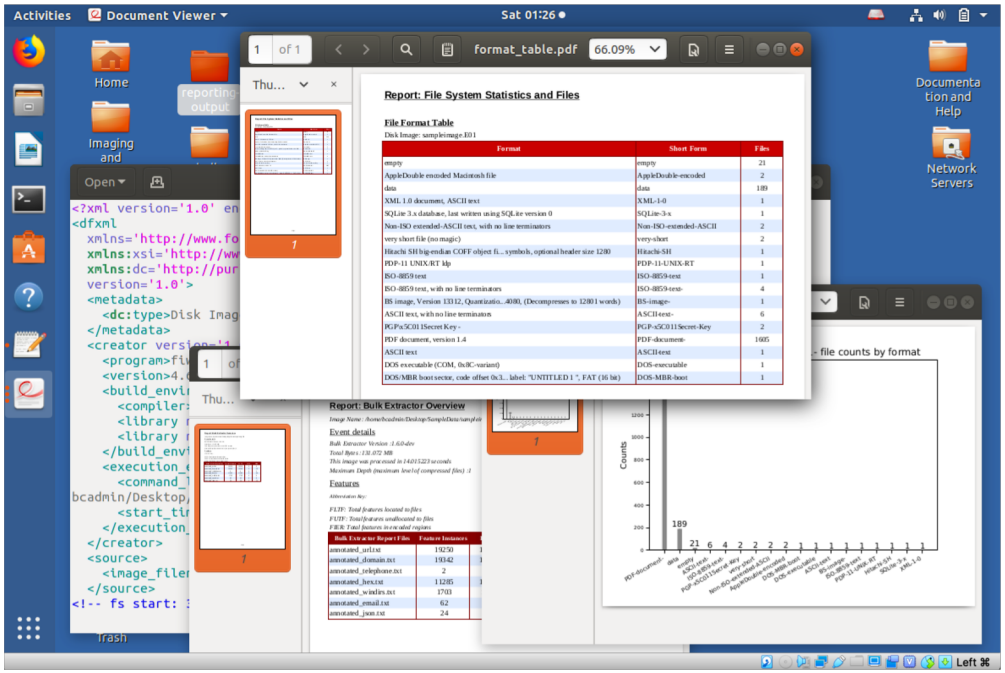
fiwalk_deleted_files.pdf – shows paths to any deleted materials found in a given partition

fiwalk-output.xml.xlsx - Excel version of the DFXML output (file system metadata)

fiwalk_report.pdf – high-level overview of file system characteristics

format_table.pdf – long-form file format names for formats shown in bargraph

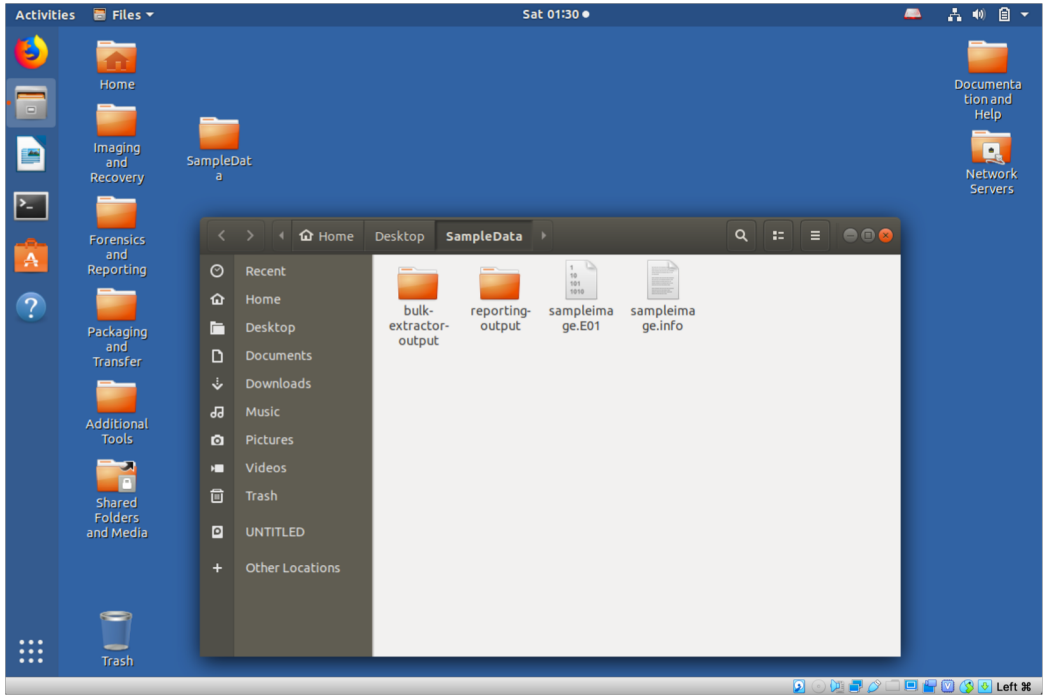
premis.xml – PREMIS preservation metadata



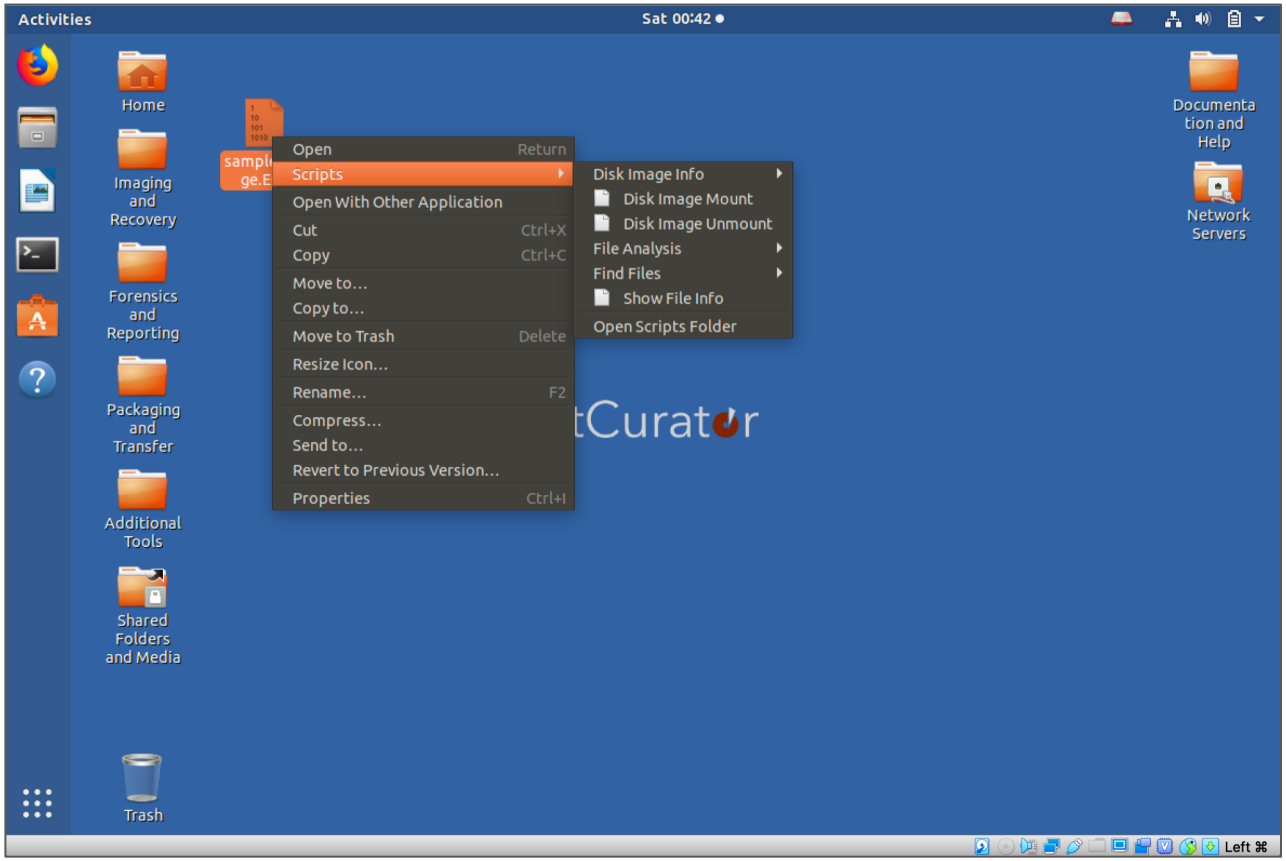
What We've Done So Far

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 bulk-extractor-output directory)
- The annotated output, linking bulk extractor features to files (in the annotated-features directory)
- A set of human-readable reports for our sample image (in the reporting-output directory)

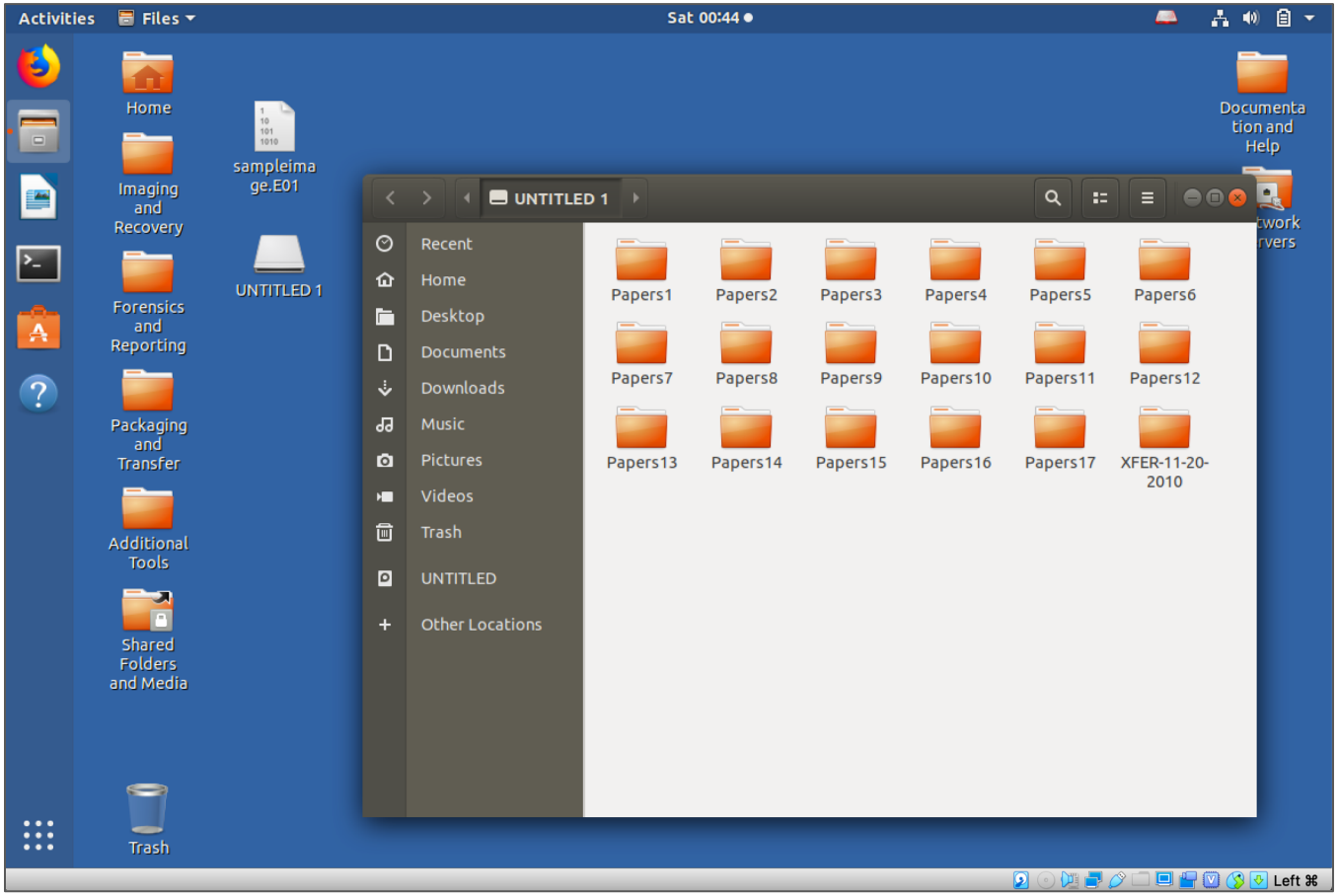


Mounting a disk image to browse the contents



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

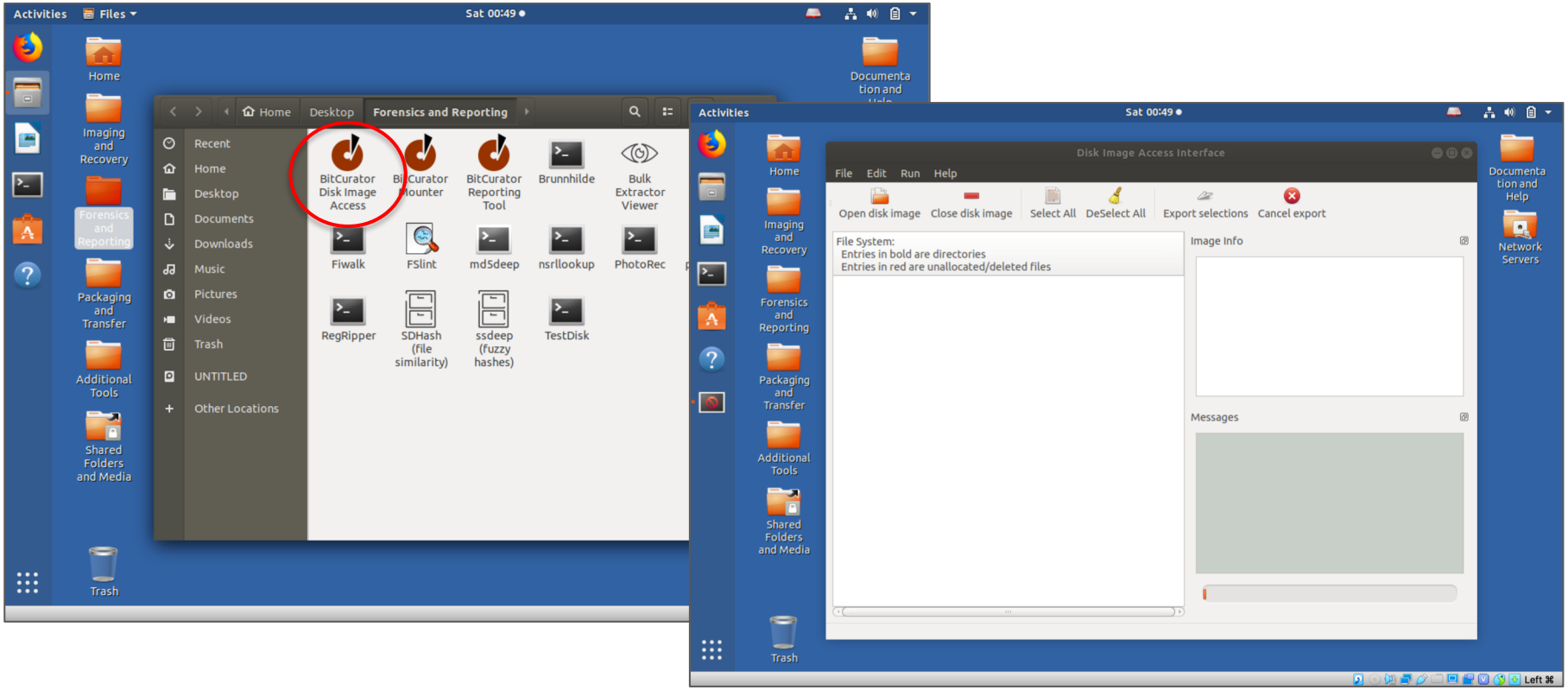
Mounting a disk image to browse the contents



You'll see a disk icon appear on the desktop corresponding to the mounted image. If a disk image does not appear on the desktop, one of two things occurred: no file system was found on the disk image, or an unrecognized file system was encountered.

Tip: To unmount, right-click on the disk image, not the mounted disk icon on the desktop.

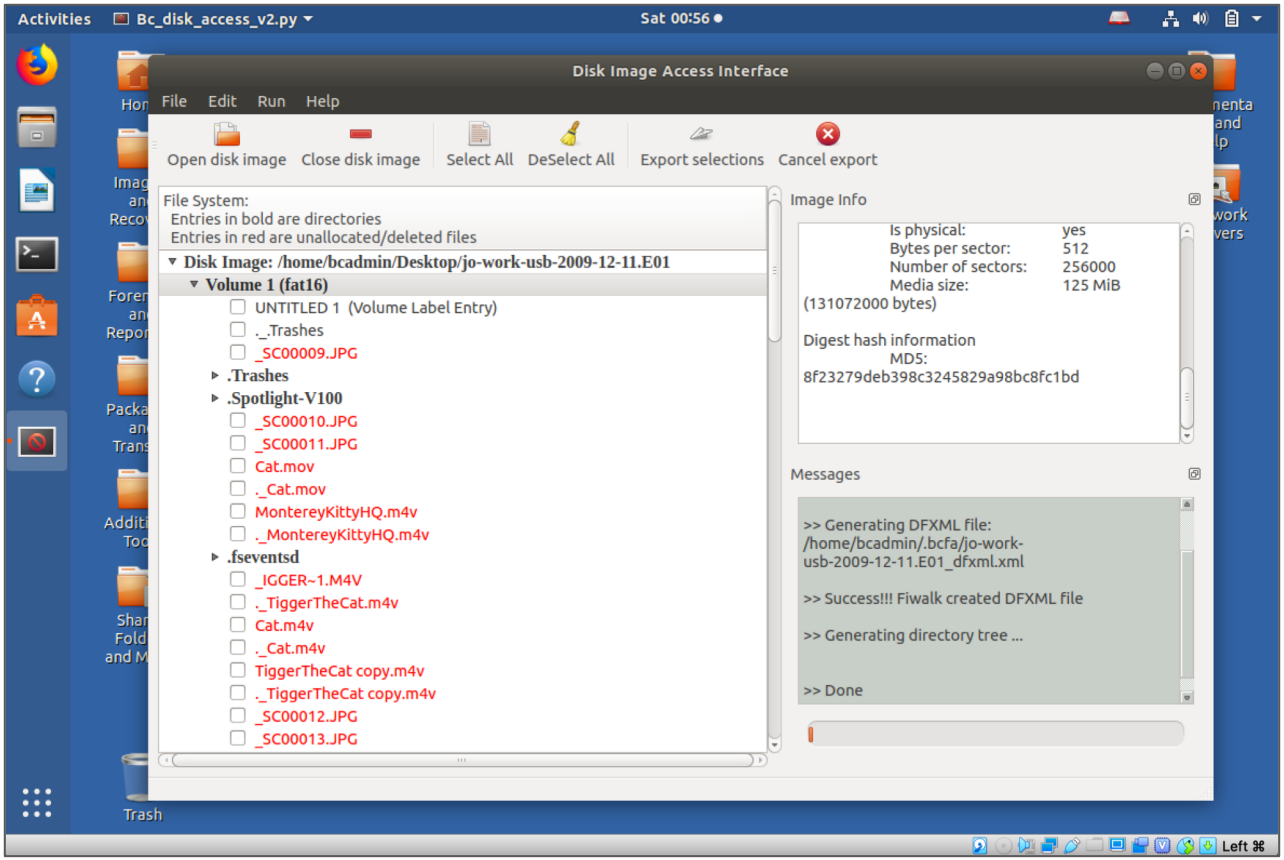
Exporting Files from a Disk Image Using the BitCurator Disk Image Access Tool



The BitCurator environment includes a standalone tool for access to the contents of file disk images, the **BitCurator Disk Image Access** tool. This tool can be found in the **Forensics and Reporting** folder on the desktop.

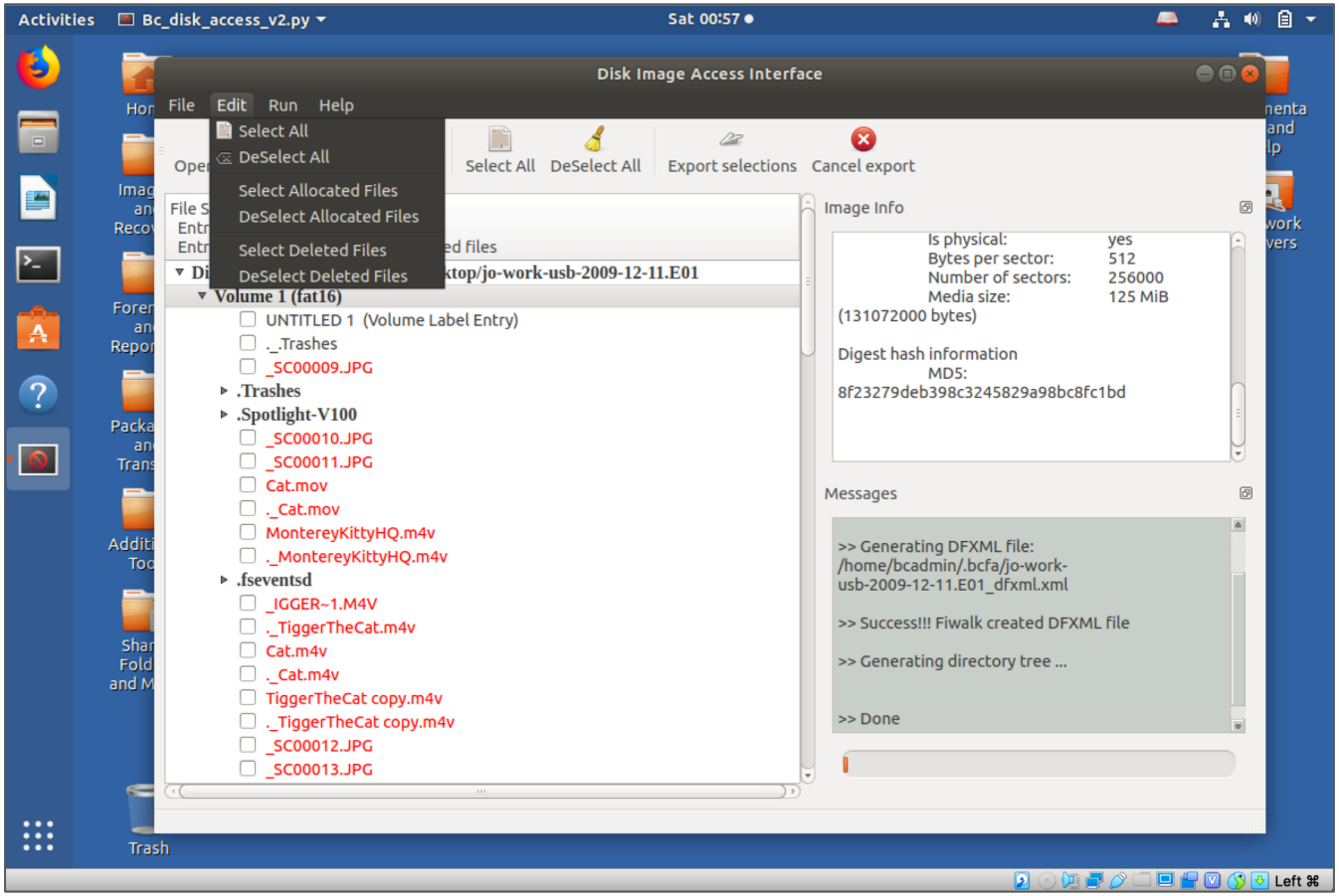
Tip: This tool uses the icat utility provided by The Sleuth Kit to extract files directly from raw and forensic disk image formats. Extracting lots of files (or large files) may take some time!

Exporting Files from a Disk Image Using the BitCurator Disk Image Access Tool



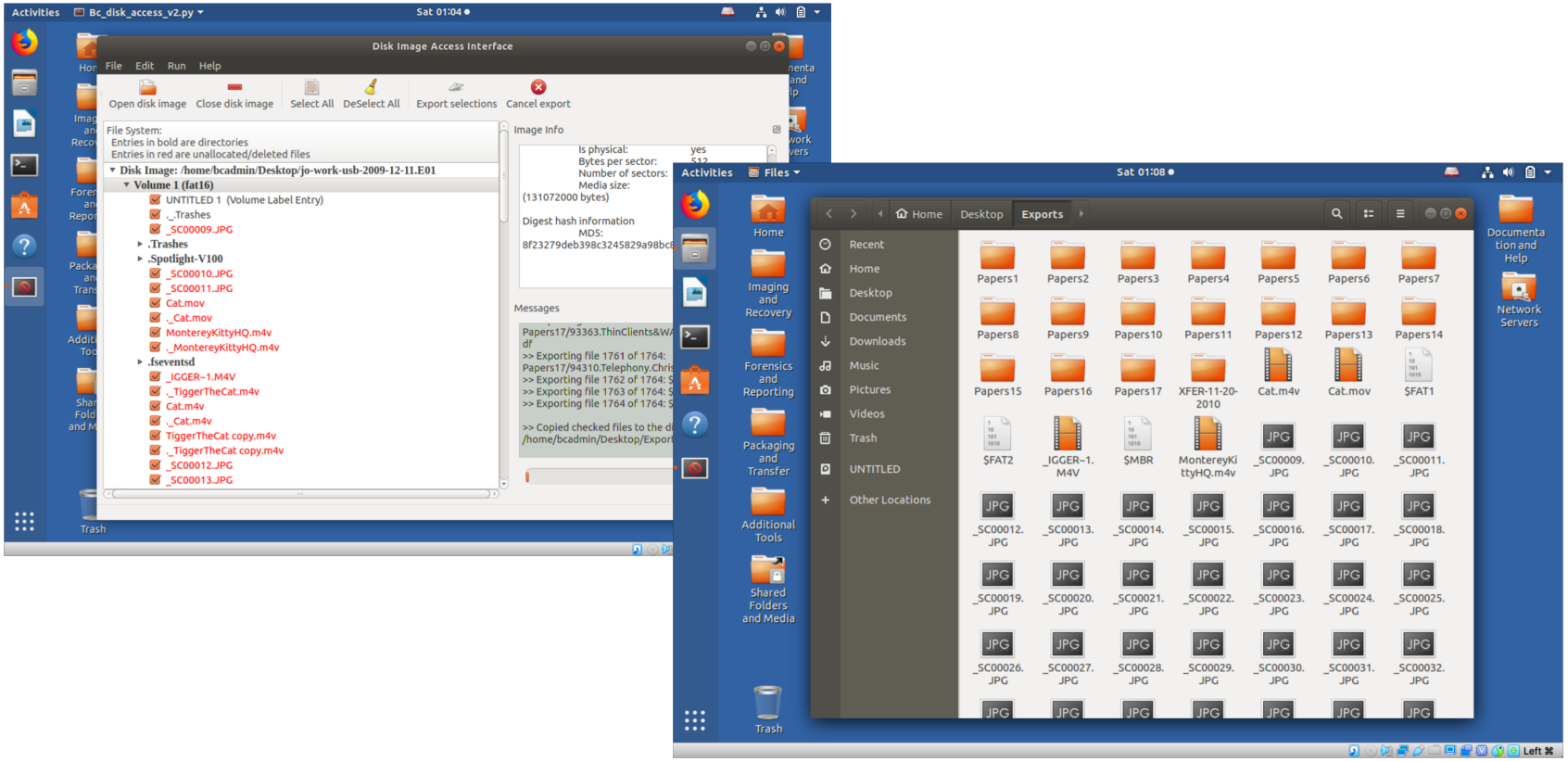
Clicking on the folder icon in the toolbar will bring up a prompt to load a disk image. The disk image will be automatically processed, and image information (if available) will be presented in the top-right dock widget. Files in the disk image may be selected manually or by clicking the **Select All** icon in the toolbar. Files may be exported using the **Export** icon (the **paper plane** to the left of the **Cancel** icon).

Exporting Files from a Disk Image Using the BitCurator Disk Image Access Tool



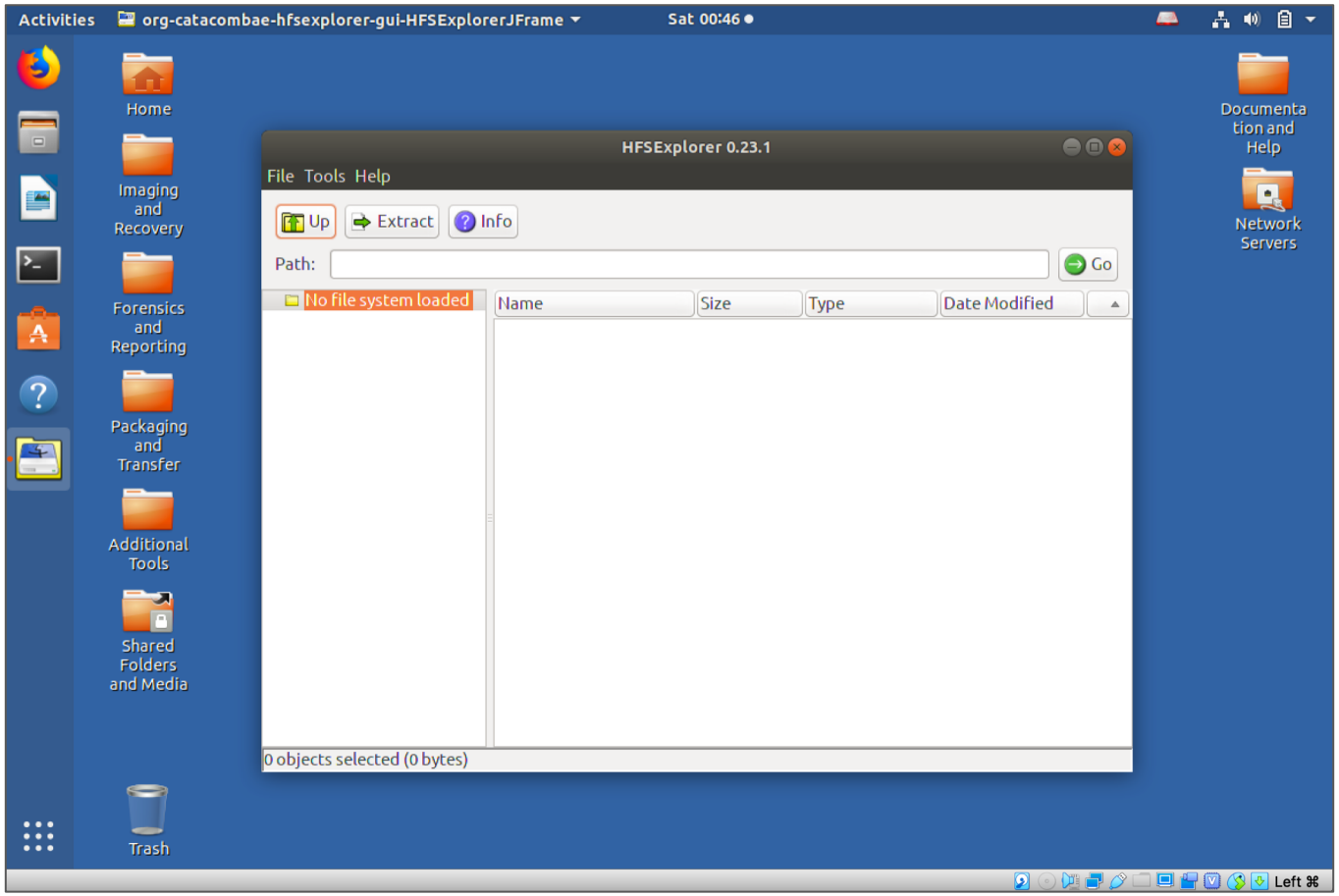
The **Edit** menu includes options to select and deselect only the allocated or only the deleted files. Click the “Export selections” button to export the selected items to a folder.

Exporting Files from a Disk Image Using the BitCurator Reporting Tool



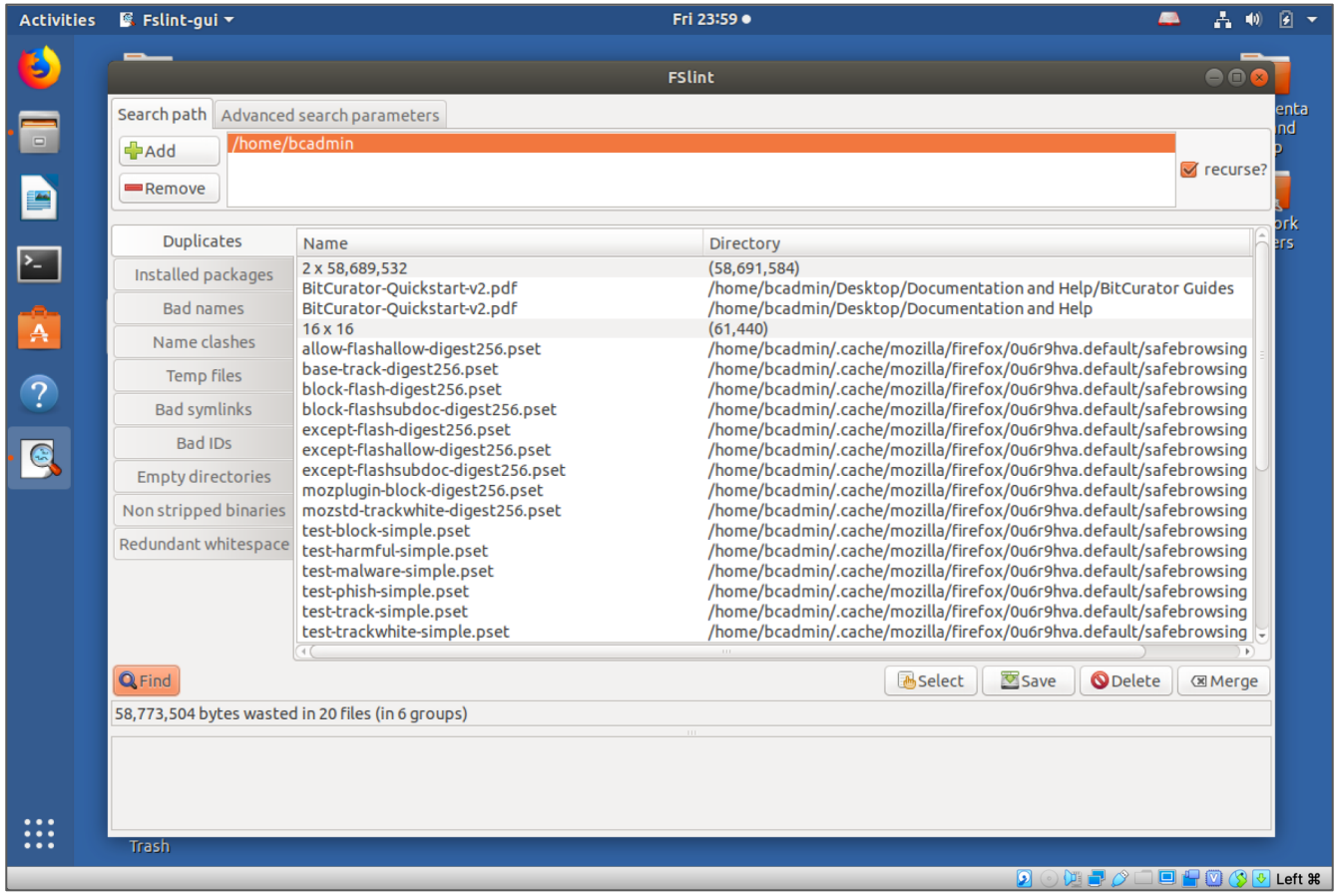
Tip: It is not always be possible to recover files marked as deleted. If the forensic software library used by this tool is unable to carve out any data associated with a deleted or damaged file, that file will not appear in the selected exports directory.

Other Tools: Browse Legacy HFS (early Macintosh file system) images



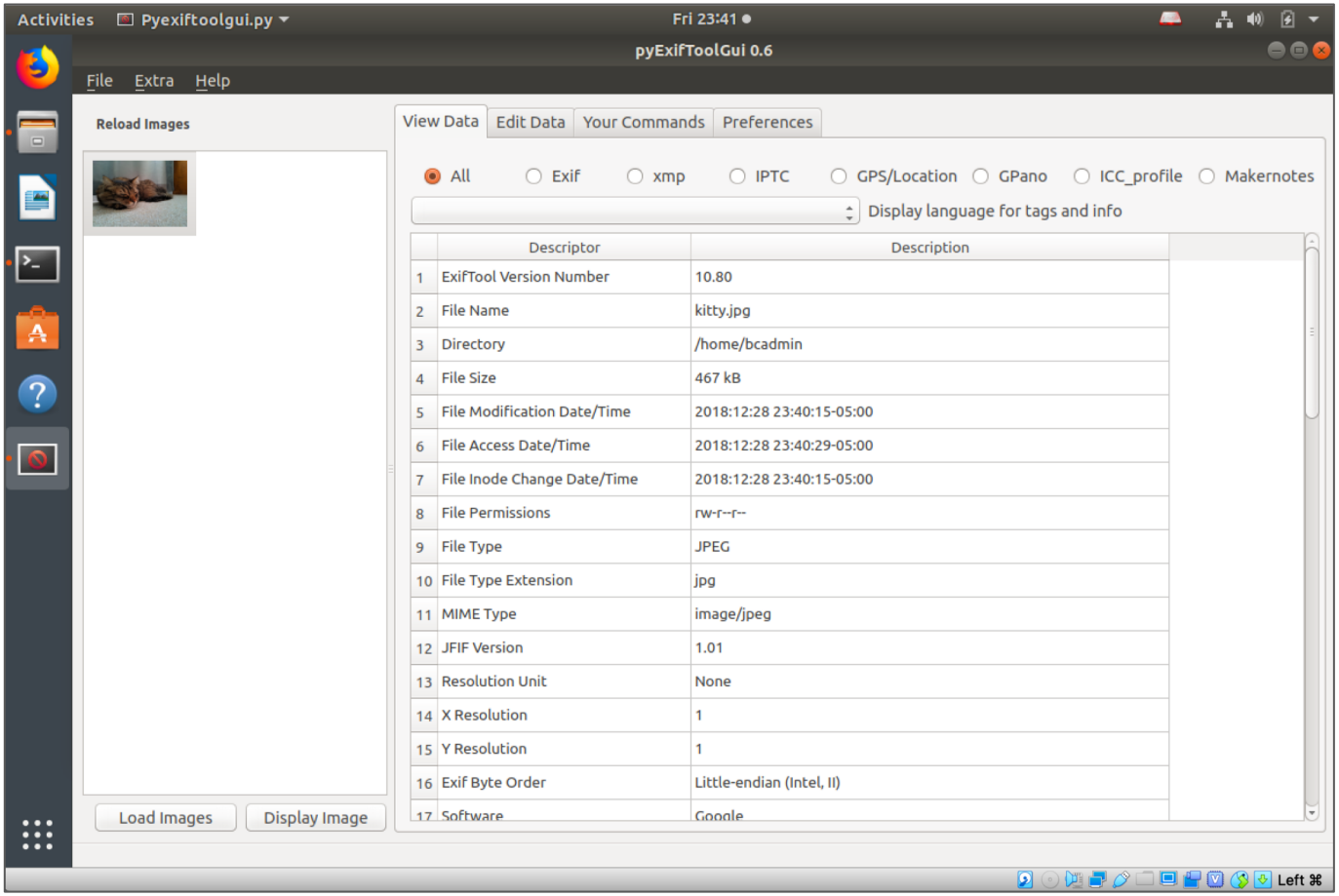
Many of the tools in BitCurator (including the BitCurator Reporting Tool and Disk Image Access Tool) use The Sleuth Kit to process disk images. TSK cannot process legacy HFS (early Macintosh) file systems. BitCurator includes **HFSExplorer** to support browsing and file export from HFS file systems.

Other Tools: Identify and Delete Duplicates with FSLint



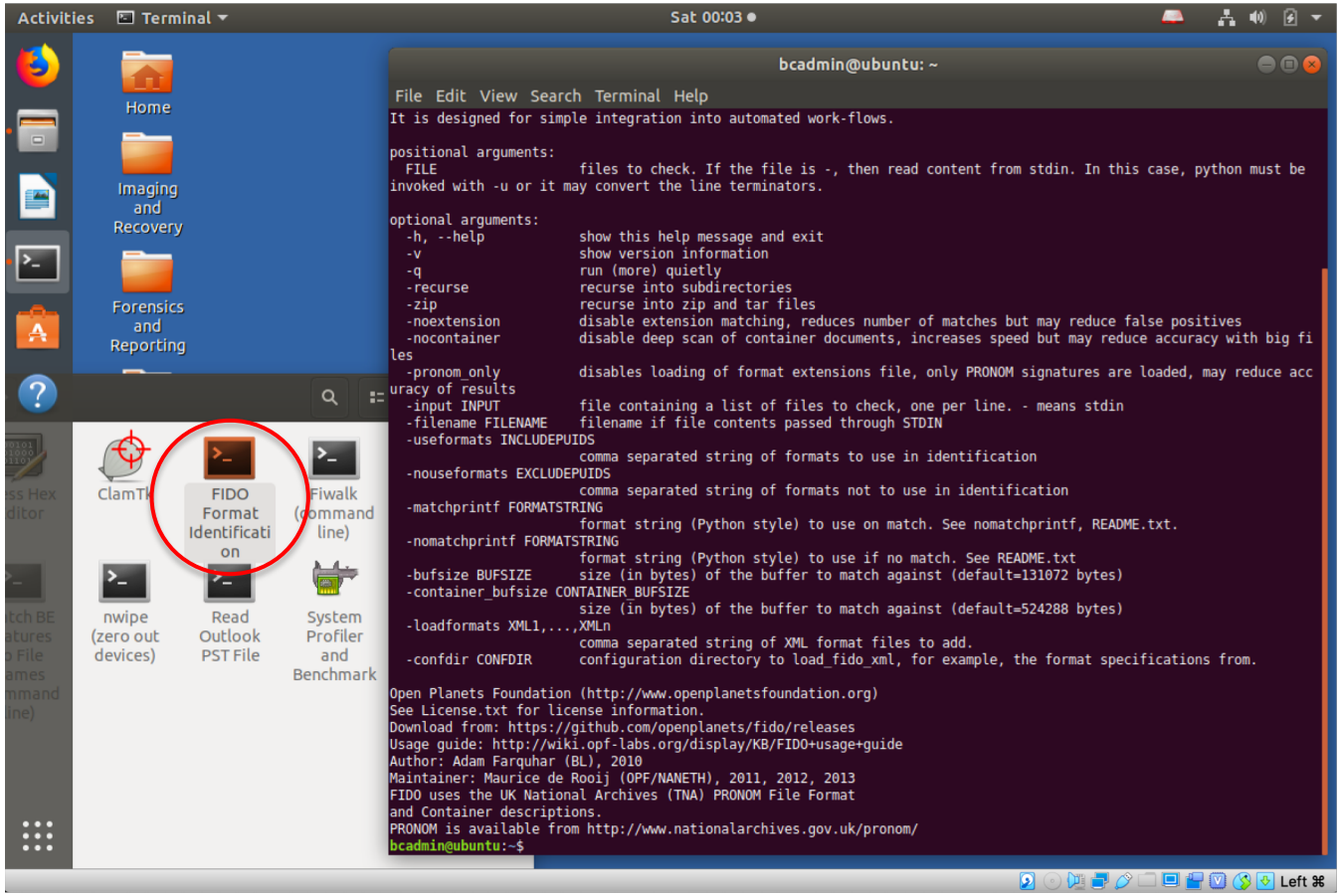
BitCurator includes **FSLint**, which allows you to rapidly scan directory contents to identify duplicates, and delete selected items from the subsequent duplicate lists.

Other Tools: pyExifToolGUI



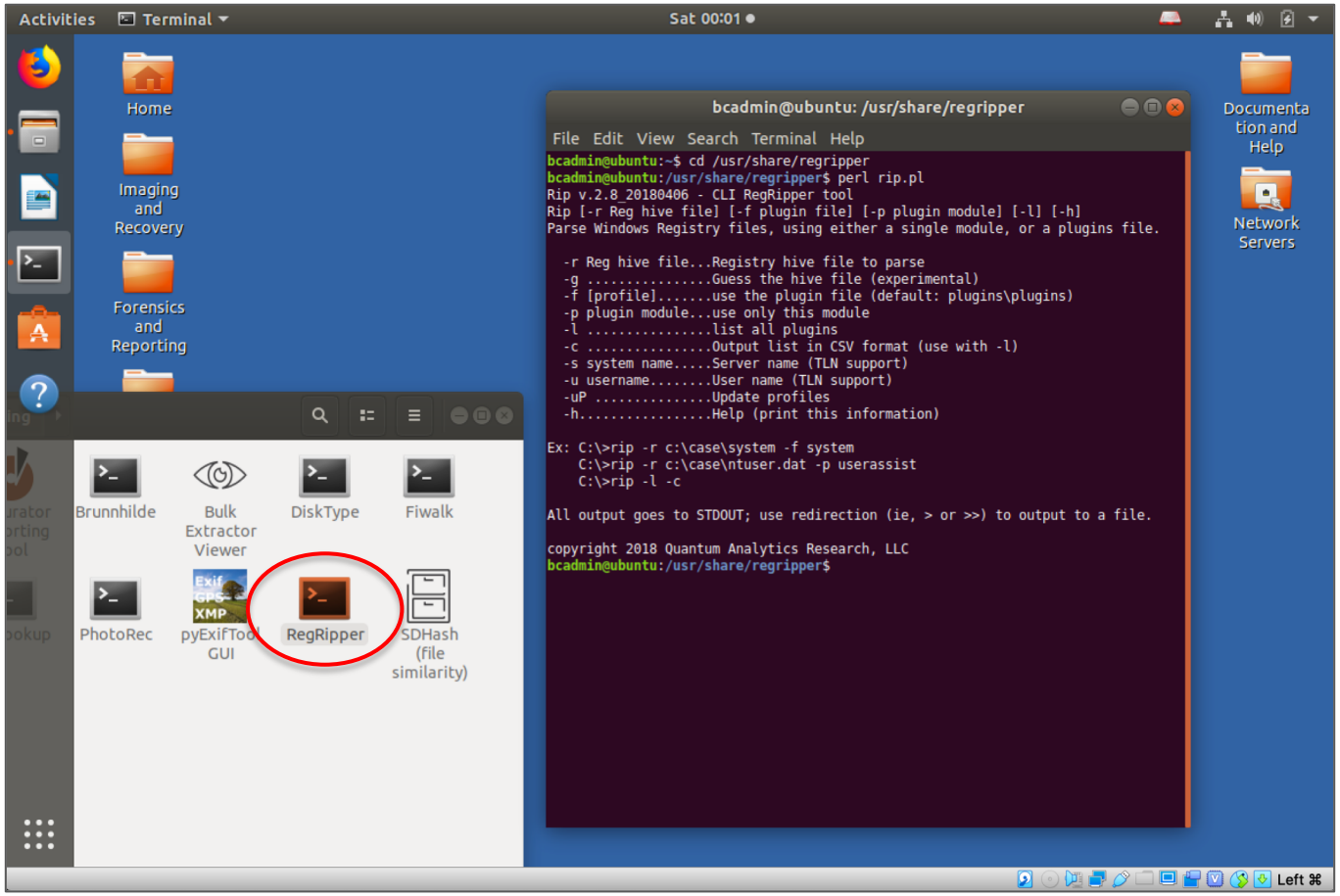
pyExifToolGUI is a graphical front-end for **ExifTool**. It allows you to view, edit, and manually export data from many graphical file formats.

Other Tools: FIDO



In the “Additional Tools” directory on the desktop, you will find a launcher for the (command line only) **FIDO** tool for file format analysis.

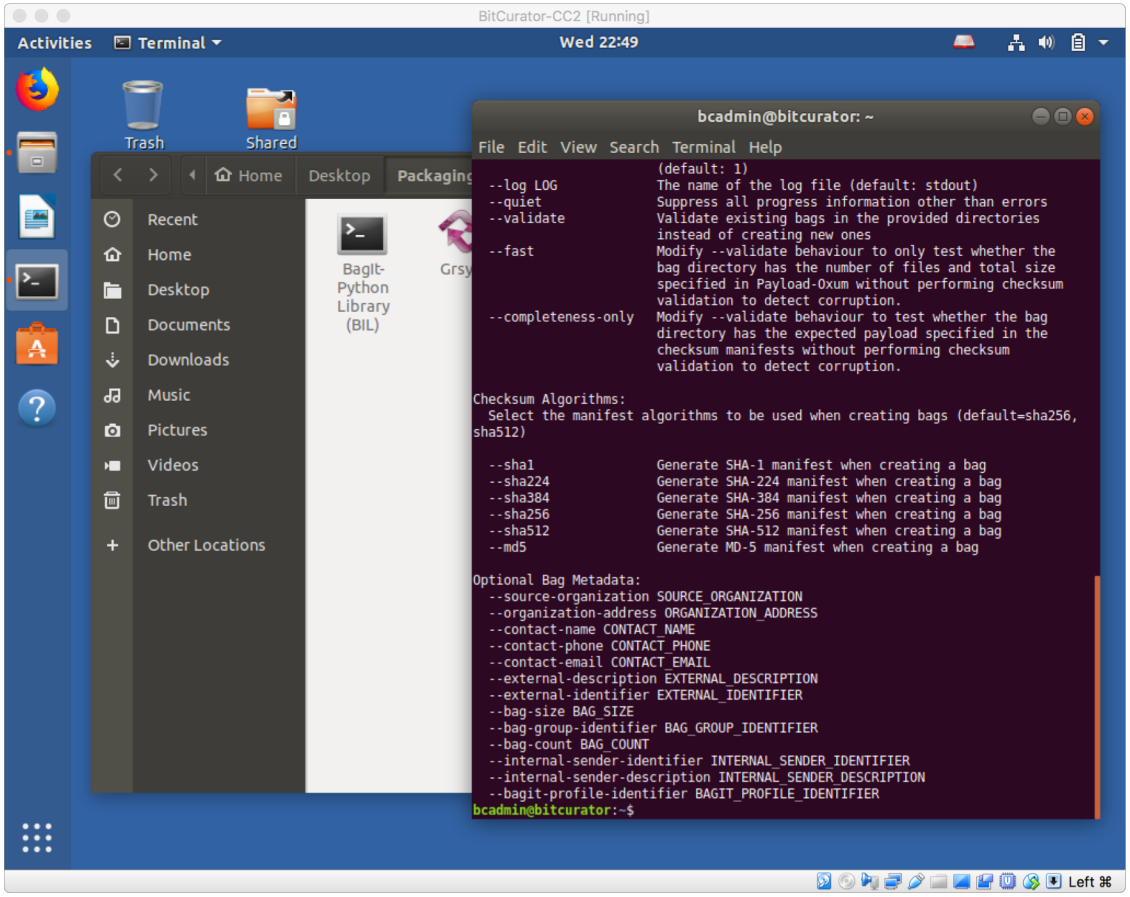
Other Tools: RegRipper



RegRipper extracts information and create reports from Microsoft Windows registry files. RegRipper is a command-line tool written in Perl.

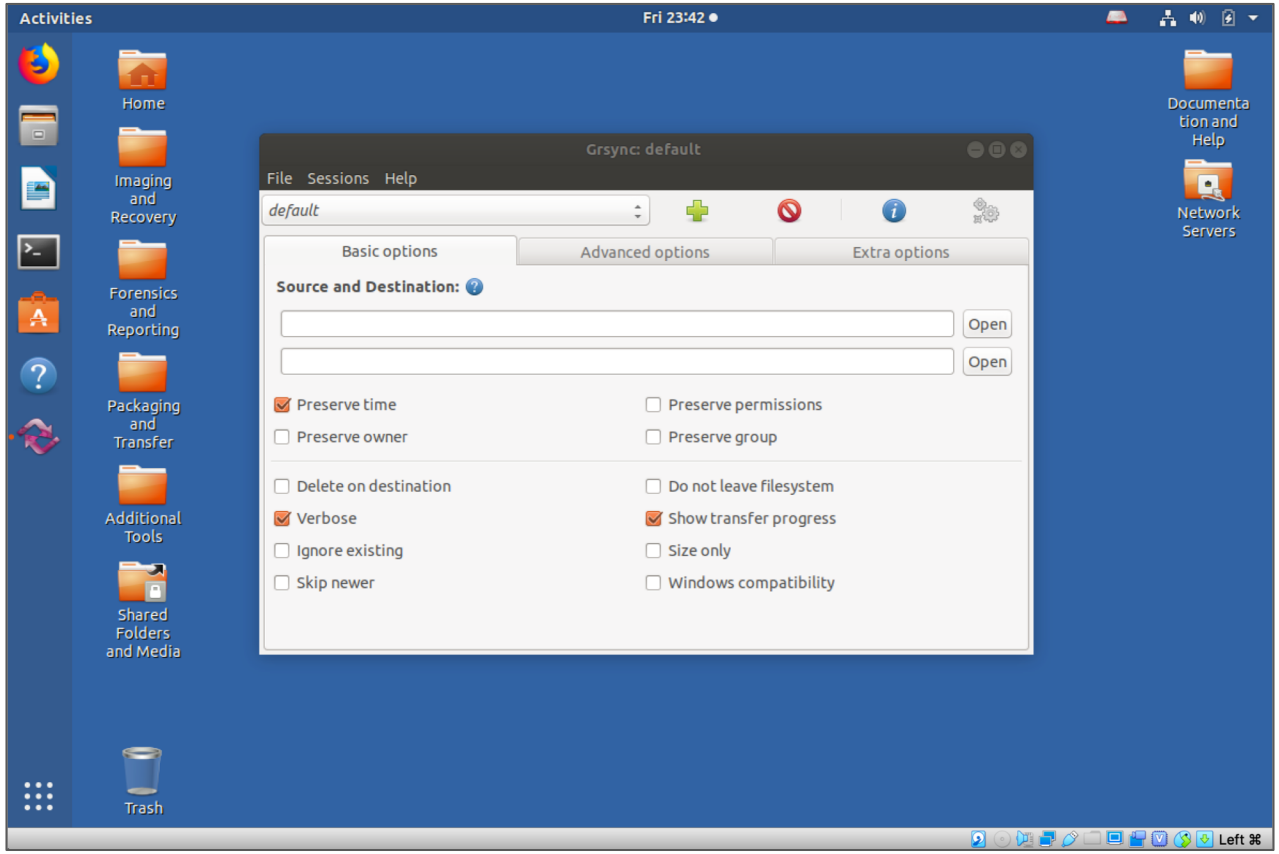
You can find it in **/usr/share/regripper**, and view options for running it using the command "perl rip.pl" within that directory (or use the desktop launcher in **Forensics and Reporting**).

Other Tools: Bagger (and BagIt Library)



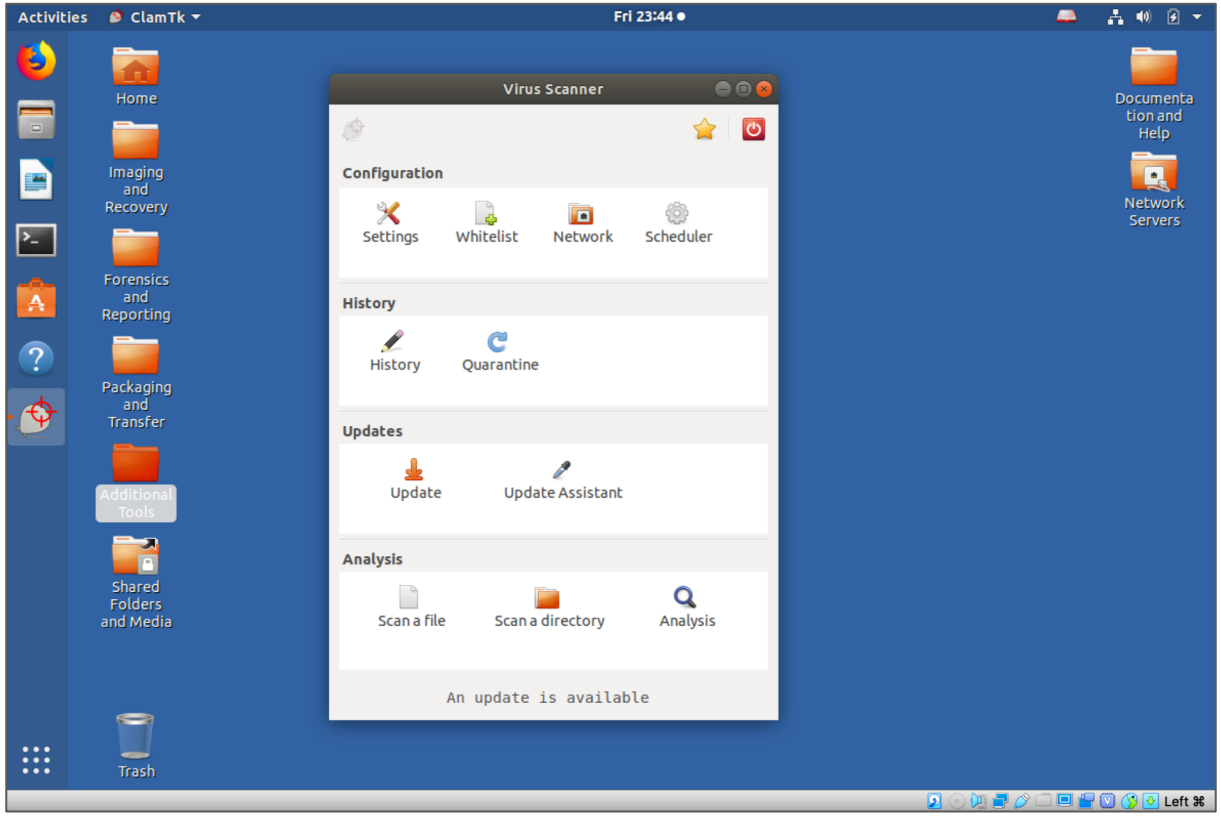
In the **Packaging and Transfer** directory on the desktop, you will find a launcher for the Python **Bagit** library used to manage and package collections of files and associated metadata.

Other Tools: Grsync



In the **Packaging and Transfer** directory on the desktop, you will find a launcher for **Grsync**, a graphical interface for the rsync utility. **Grsync** provides facilities for **permissions-preservation**, **halt-on-failure**, and many other advanced options.

Other Tools: ClamTK (GUI for Clam Antivirus)



In the **Additional Tools** directory on the desktop, you will find a launcher for **ClamTK**, the GUI front-end to the ClamAV antivirus service.

The **Scan** menu (and the related icons under “Analysis”) will allow you to scan specific files, directories, mounted disks, and other devices.

Other Tools: Running fiwalk with the ClamAV plugin to identify viruses and malware (command-line only)

The fiwalk tool has a plugin architecture allowing you to run external tools over file items it identifies. In a terminal, enter the ficlam directory by typing:

cd /home/bcadmin/.ficlam

This directory contains a fiwalk ClamAV plugin from The Sleuth Kit (ficlam.sh) and the associated configuration file. You can generate fiwalk output that includes virus scan results by typing:

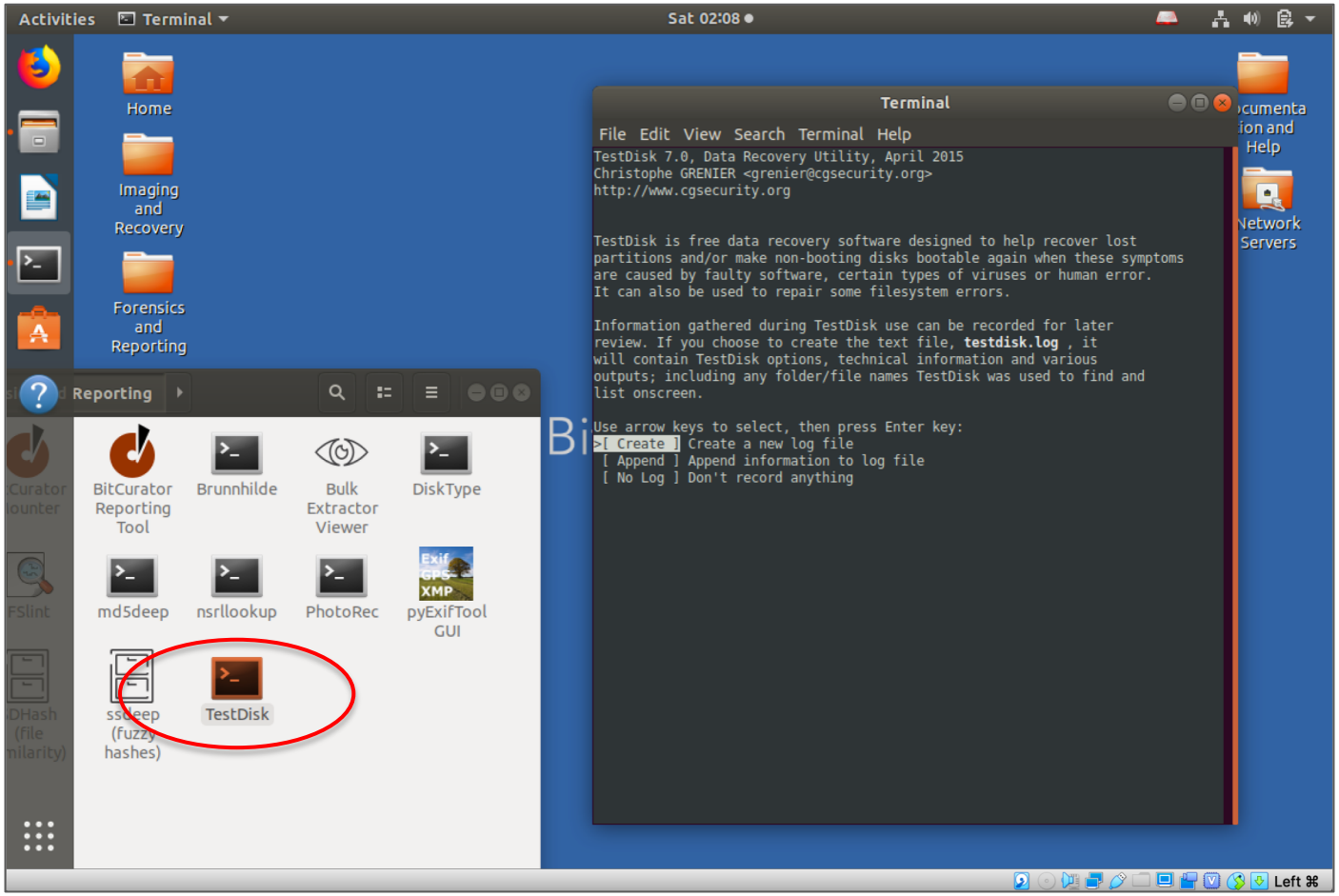
fiwalk –c clamconfig.txt –X /home/bcadmin/Desktop/myoutput.xml /home/bcadmin/Desktop/myimage.E01

In this example, "myoutput.xml" is where you'd like the DFXML output to appear, and myimage.E01 is the name of your disk image. Change these names and directory paths as needed.

File entries in the DFXML that are clean will appear as follows (at the end of each **fileobject** entry):

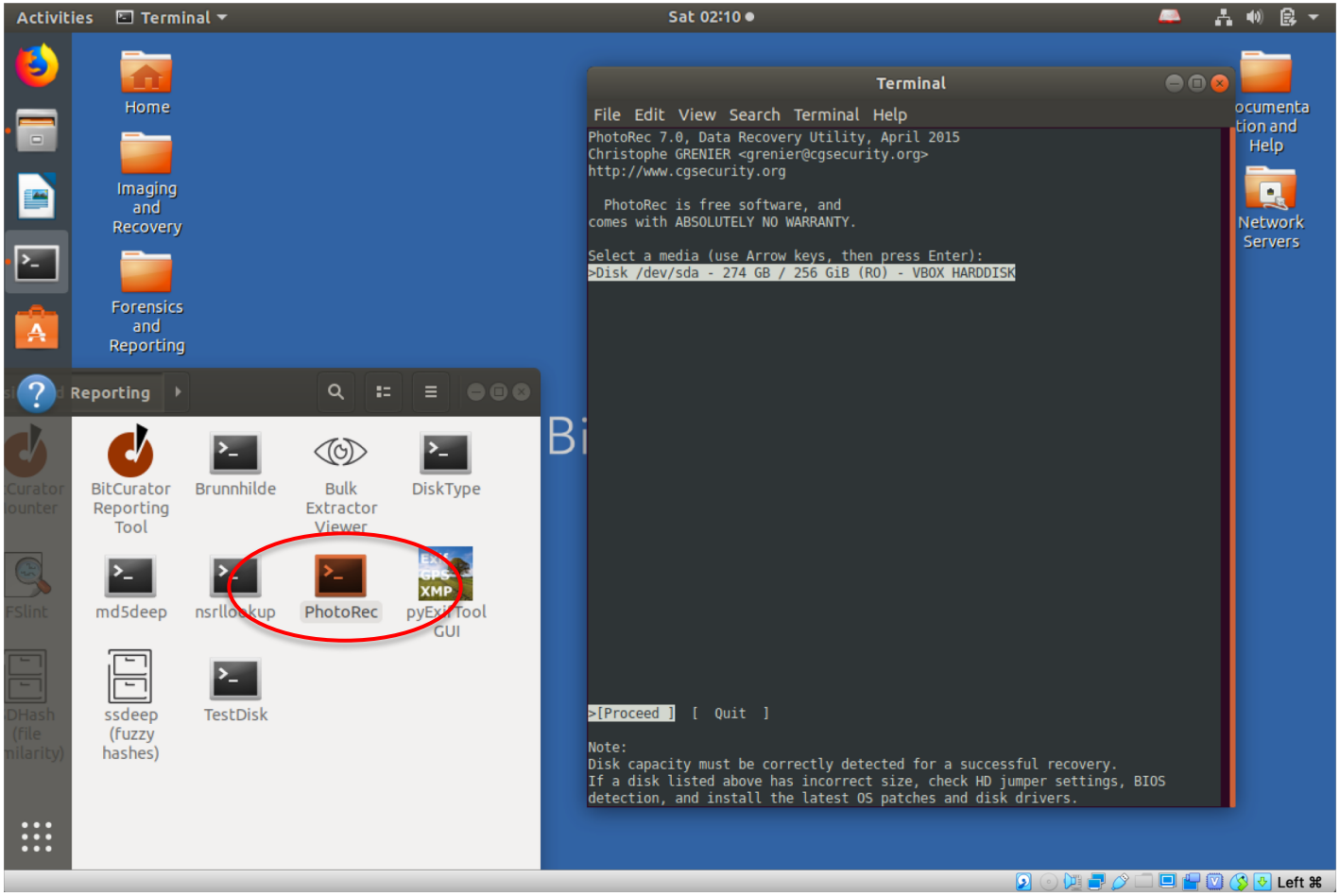
```
<!-- plugin_process -->
   <clamav_infected>0</clamav_infected>
```

Other Tools: TestDisk



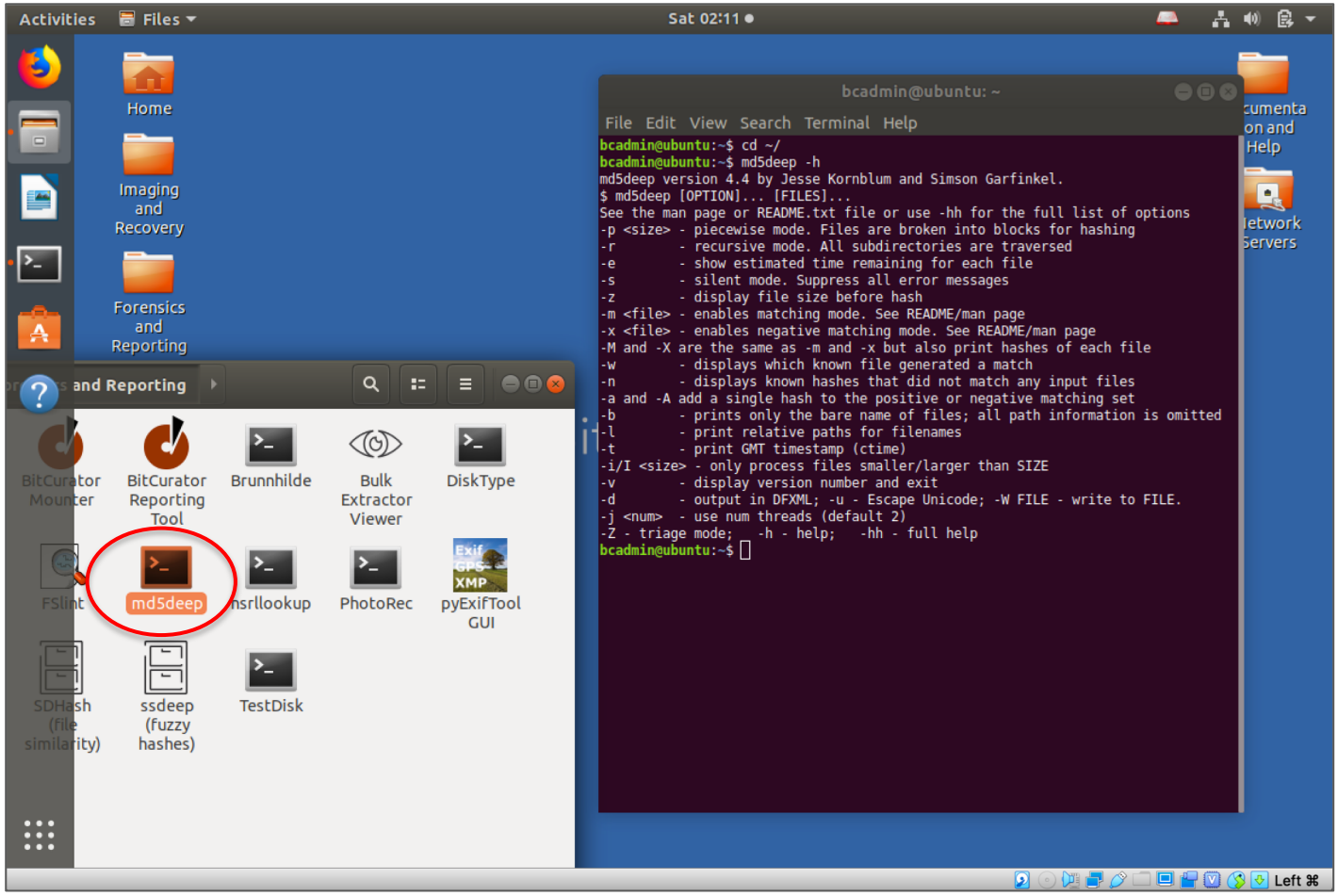
TestDisk is an tool used to recover “lost” partitions and analyze damaged disks. More info can be found at http://www.cgsecurity.org/wiki/TestDisk.

Other Tools: PhotoRec



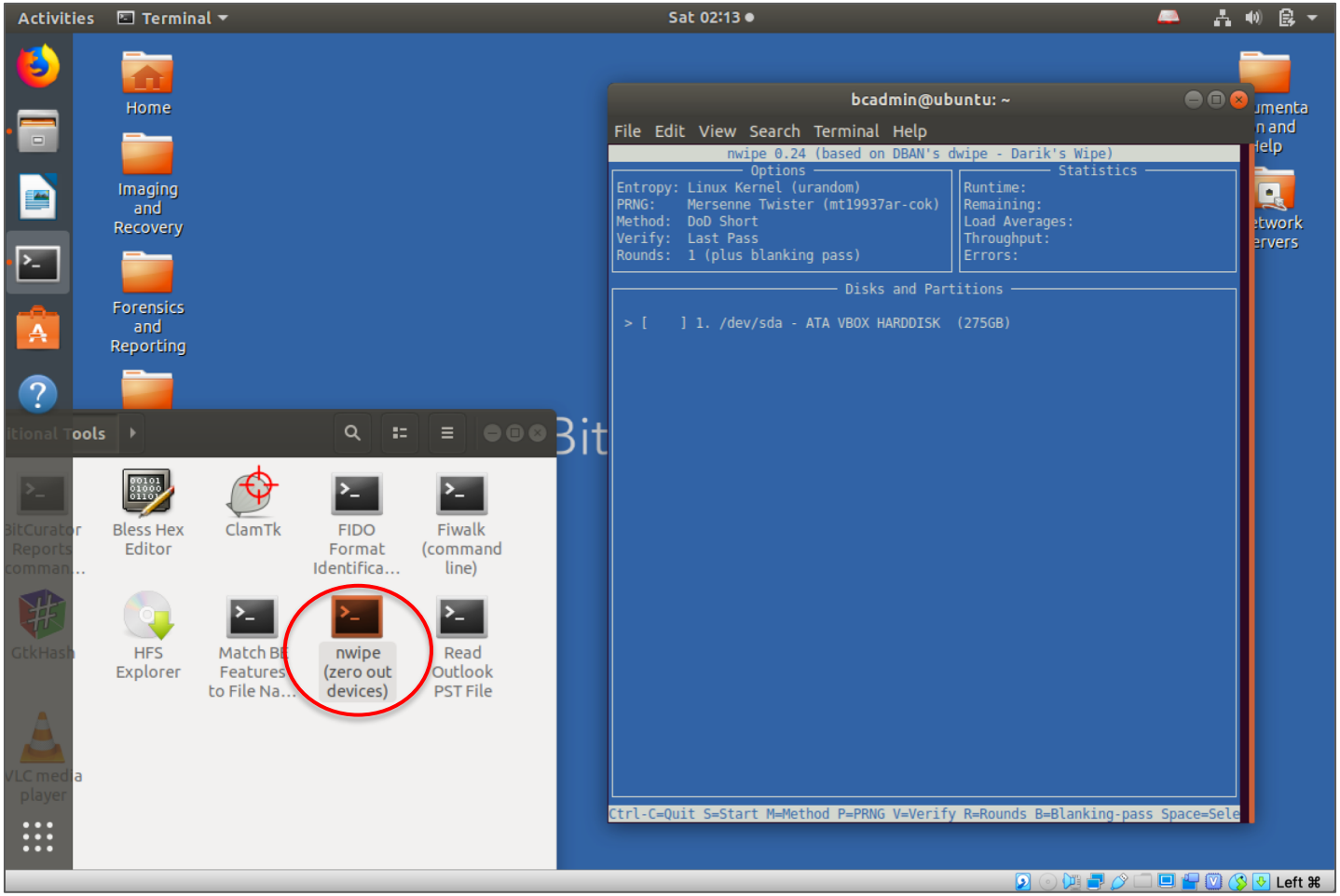
PhotoRec is a tool used to recover damaged and “lost” files. More info can be found at http://www.cgsecurity.org/wiki/PhotoRec.

Other Tools: md5deep (and hashdeep)



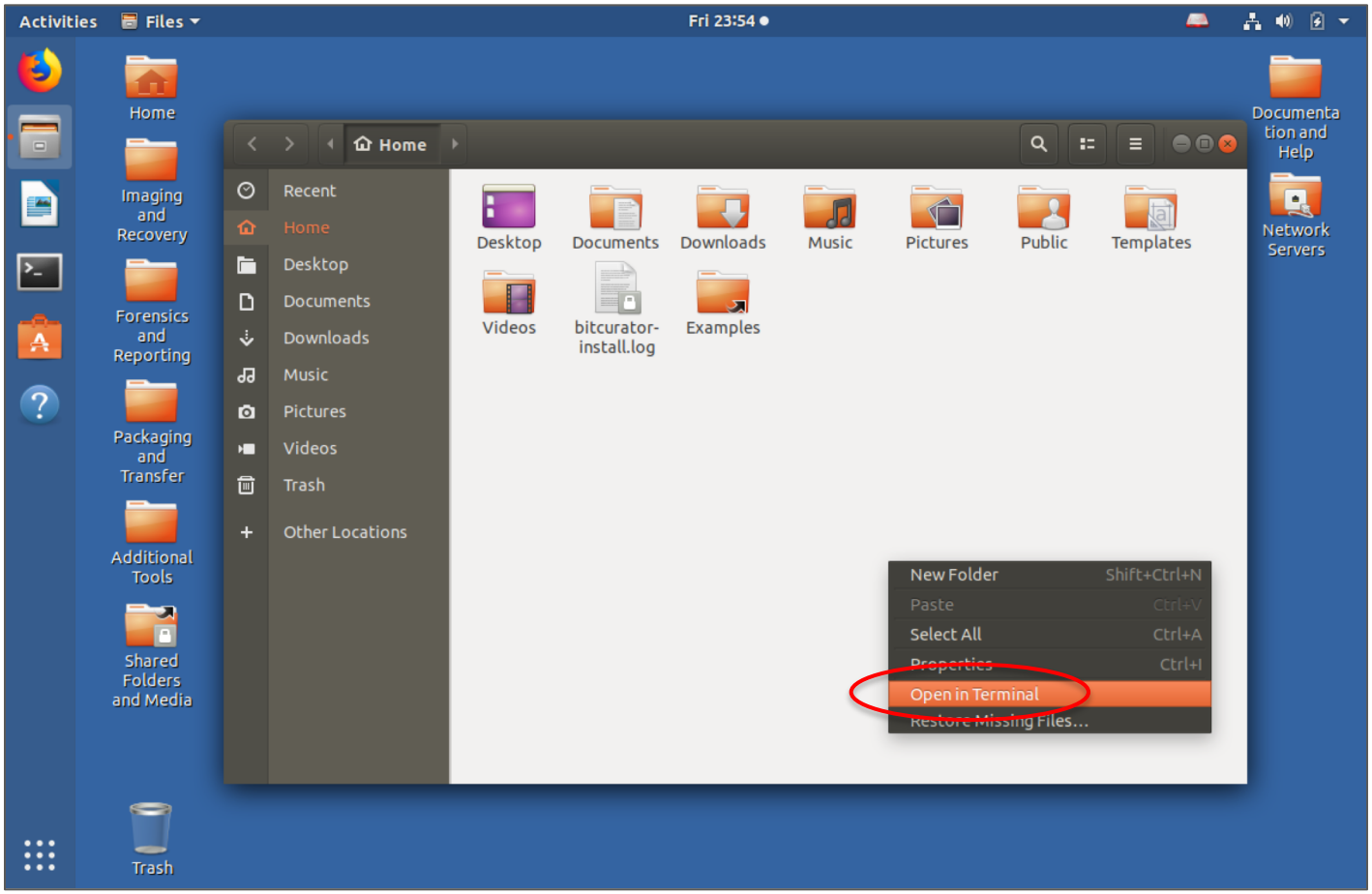
The md5deep tools may be used to compute MD5, SHA-1, SHA-256, Tiger, and Whirlpool message digests over any set of files. The toolset supports recursive operation, comparison to existing hashsets, and time estimation when running over large sources.

Other Tools: nwipe (securely wipe media)



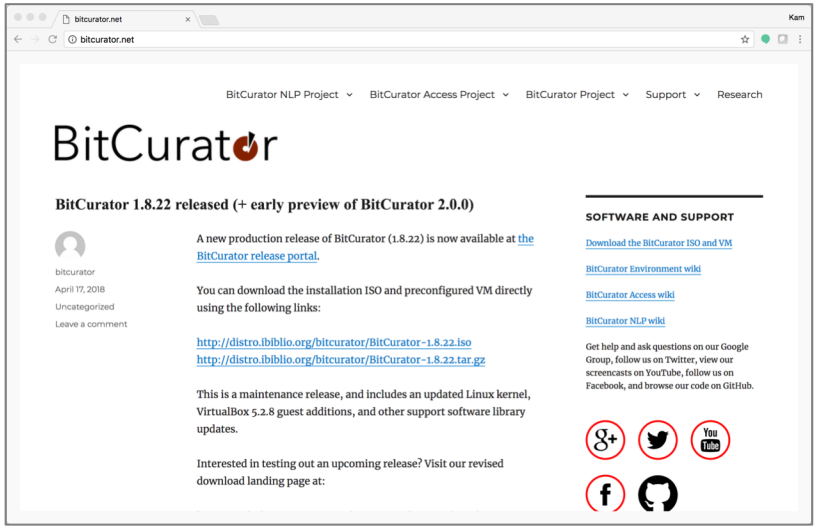
The **nwipe** tool in "Additional Tools" allows you to securely wipe (overwrite with zeros or random data) physical devices. **WARNING! This tool runs as root and allows you to wipe any device, including the disk the system is running on. Use with caution!**

Navigation Tips: Open a location in a terminal



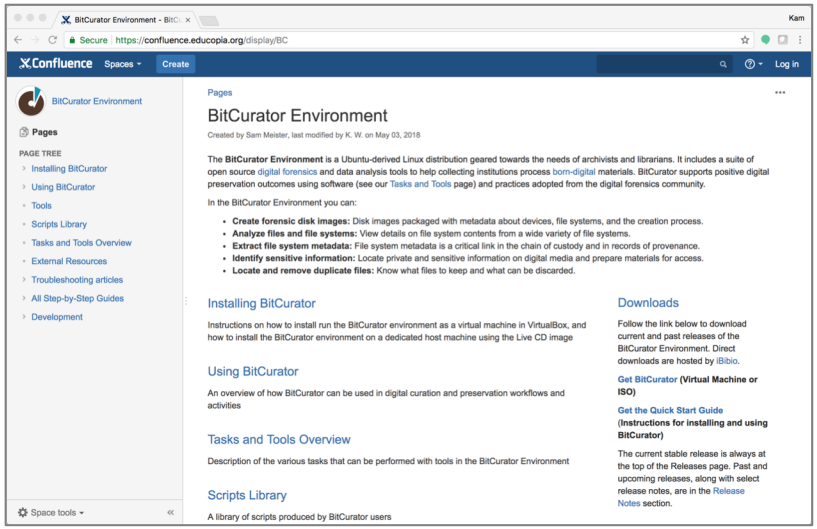
When browsing the file system using Nautilus, you can open any location in a terminal simply by right-clicking anywhere (not on a file or folder) in the browser window and selecting **Open in Terminal**.

Find Updated BitCurator Information and Documentation Online



- News and social media updates on BitCurator projects
- Project histories
- Personnel and contributors

https://bitcurator.net

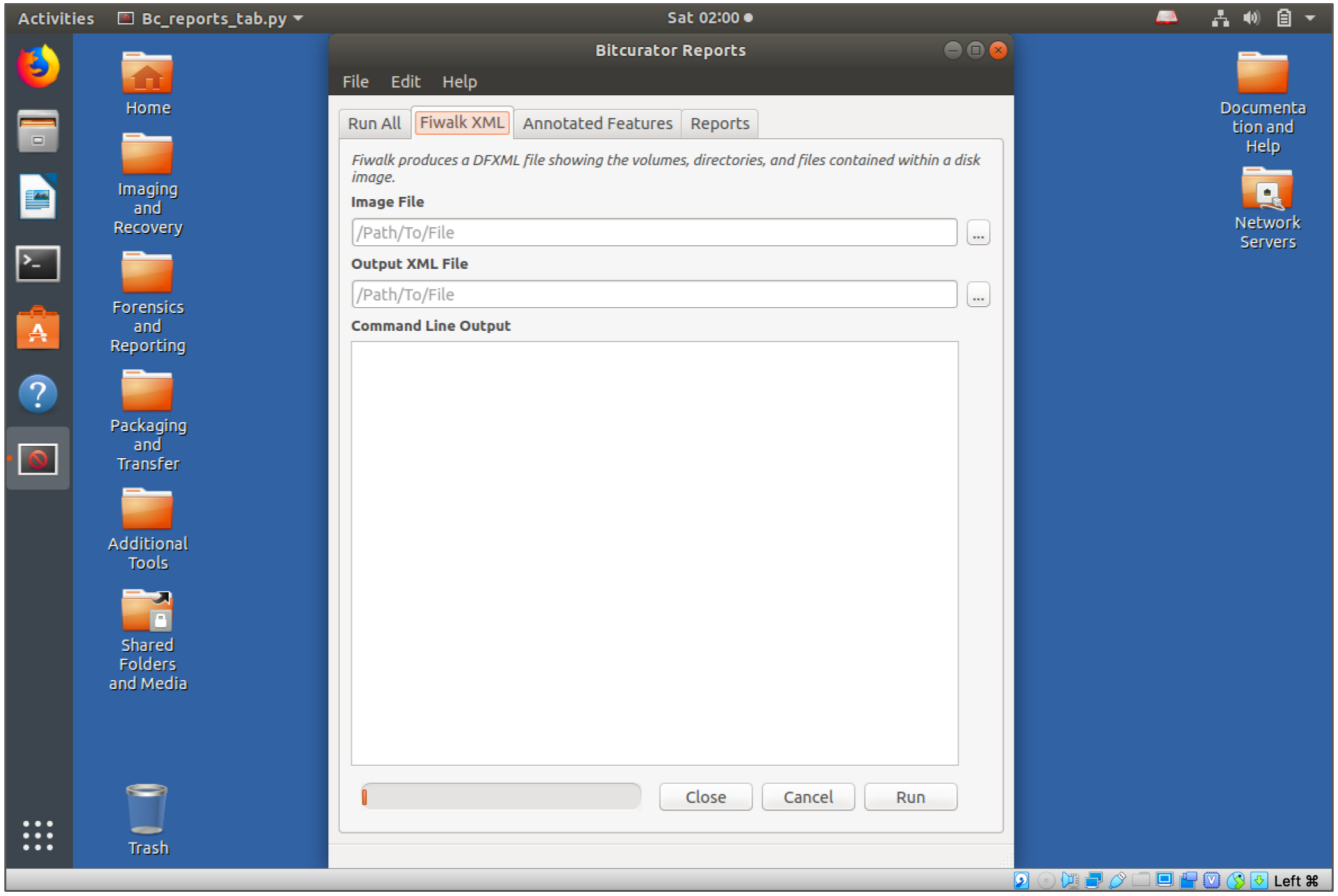


- Documentation and walkthroughs
- Workflows, scripts and BitCurator Consortium member contributions

https://confluence.educopia.org/display/BC

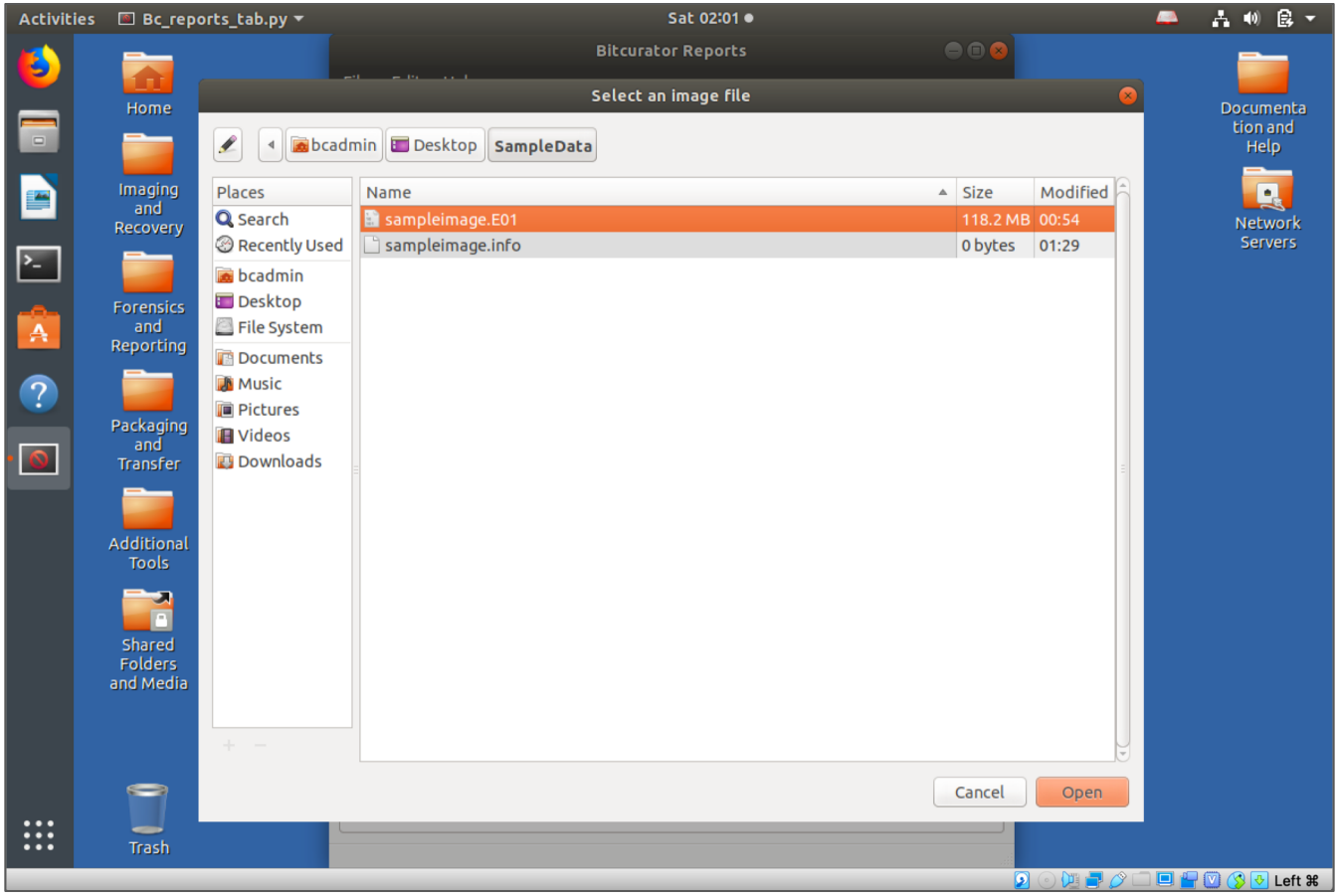
APPENDIX A: Running BitCurator reporting tools individually

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



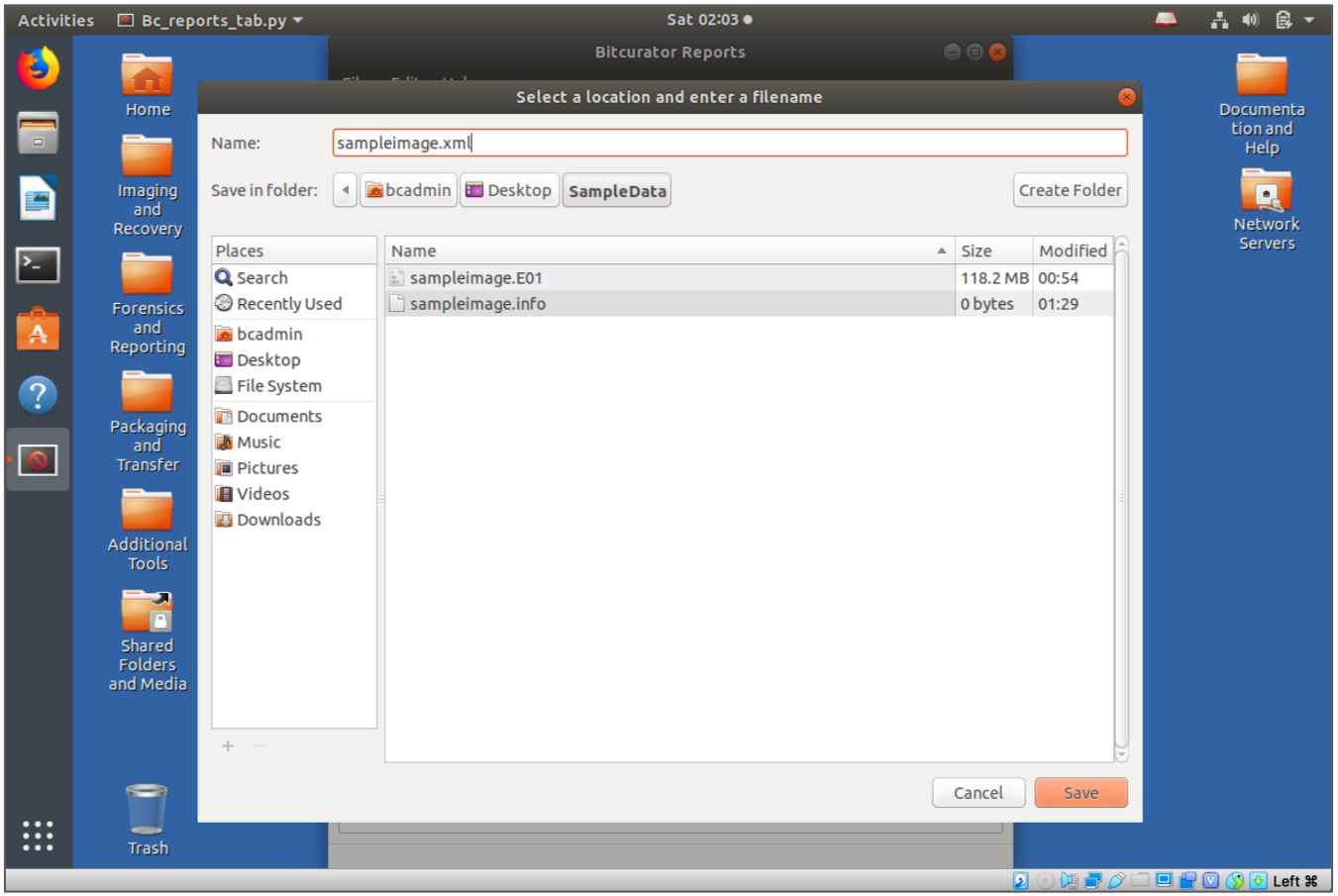
Double-click on the **Forensics and Reporting** folder, and then double click on the **BitCurator Reports** launcher. You’ll see a window pop up that should match the picture shown above. Select the **Fiwalk XML** tab.

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



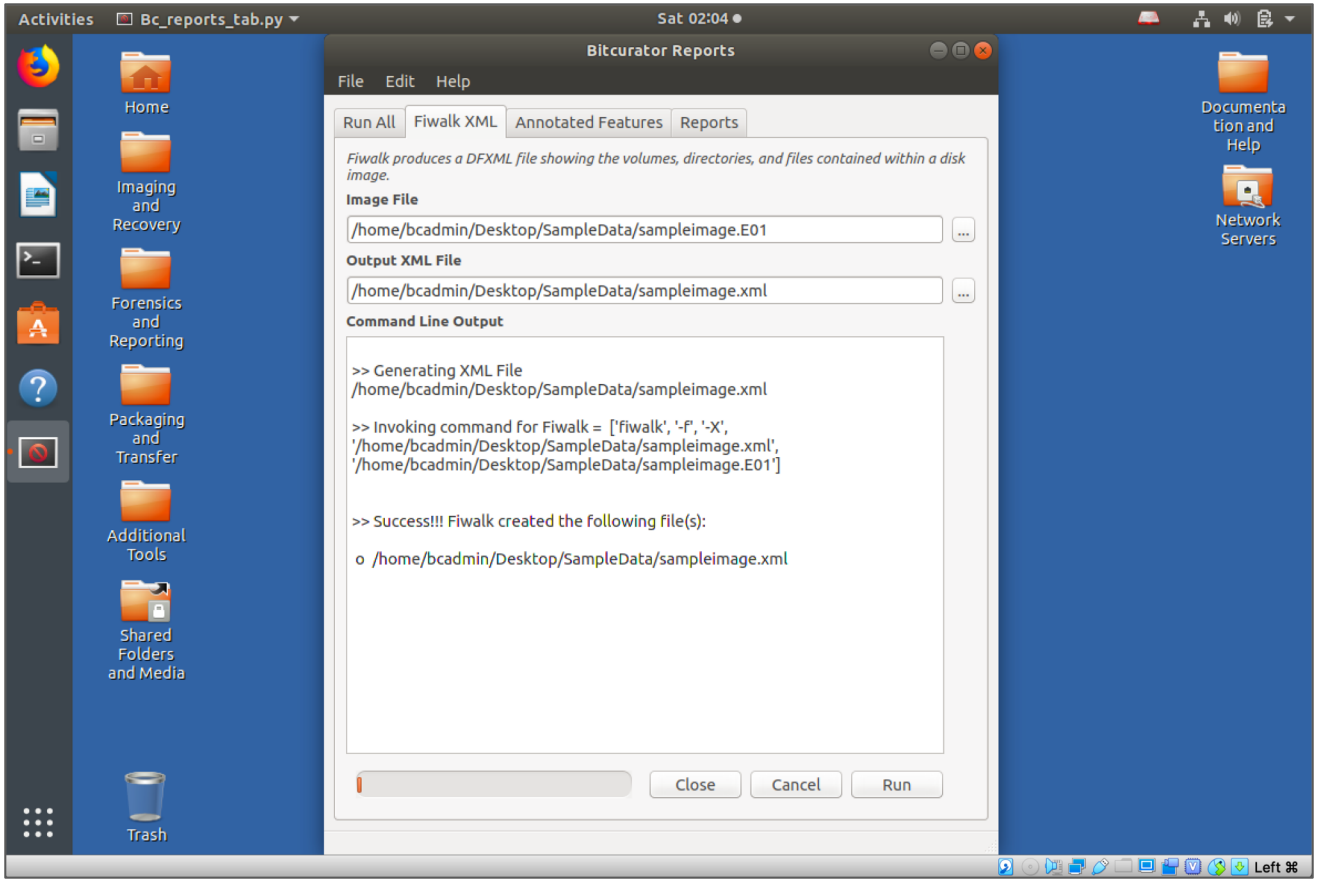
Fiwalk needs to know where the disk image file is. It also needs to know where to create the DFXML output file. Click on the **‘…’ box** to the right of the Image File text edit box, and navigate to the directory containing the image we just created. Select ‘sampleimage.E01’ and click **Open**.

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



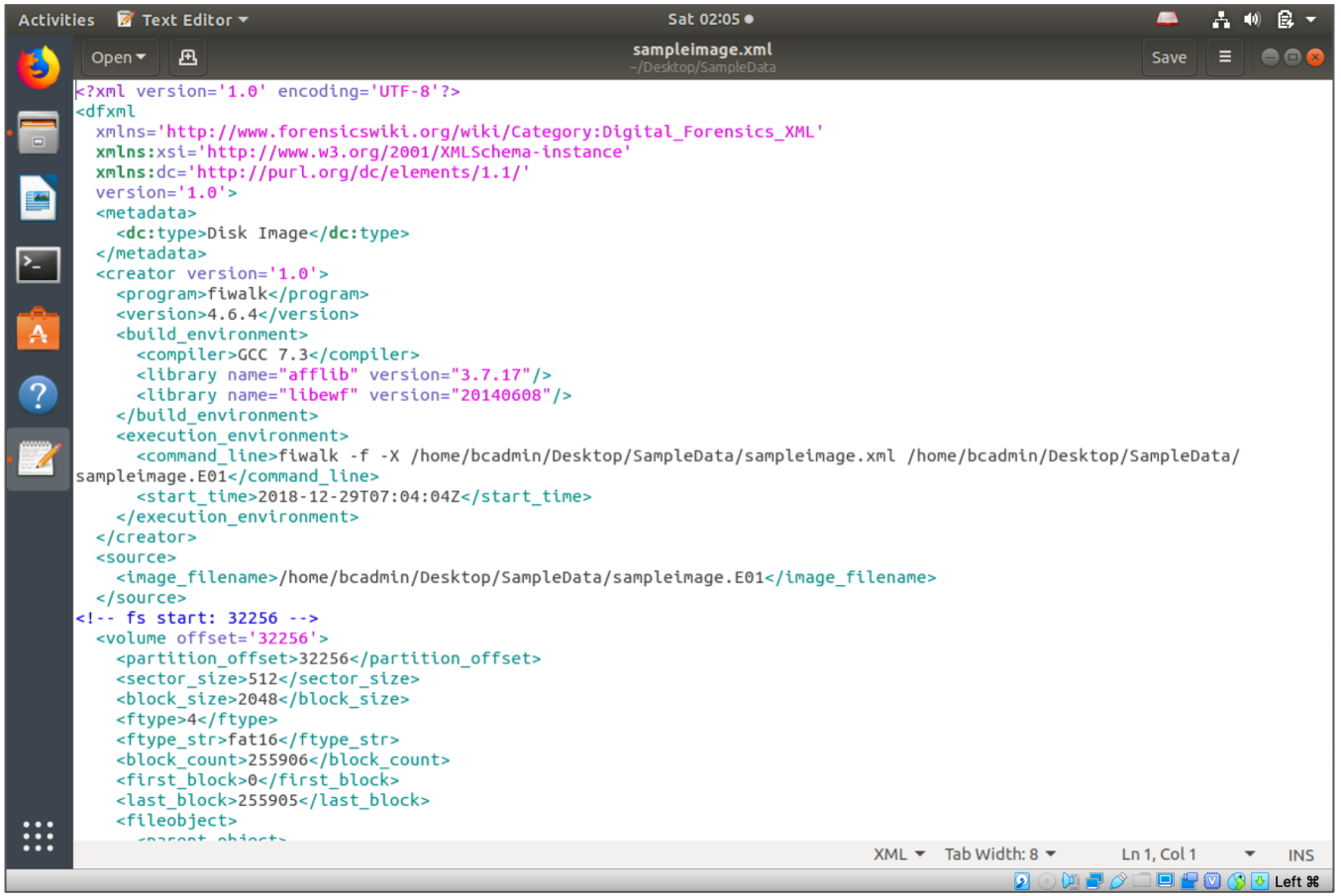
Now click on the box with three dots to the right of the **Output XML File** text area, and navigate to the same directory on the desktop. Type in “sampleimage.xml” under **Name** at the top, and click **Save**.

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



The 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.



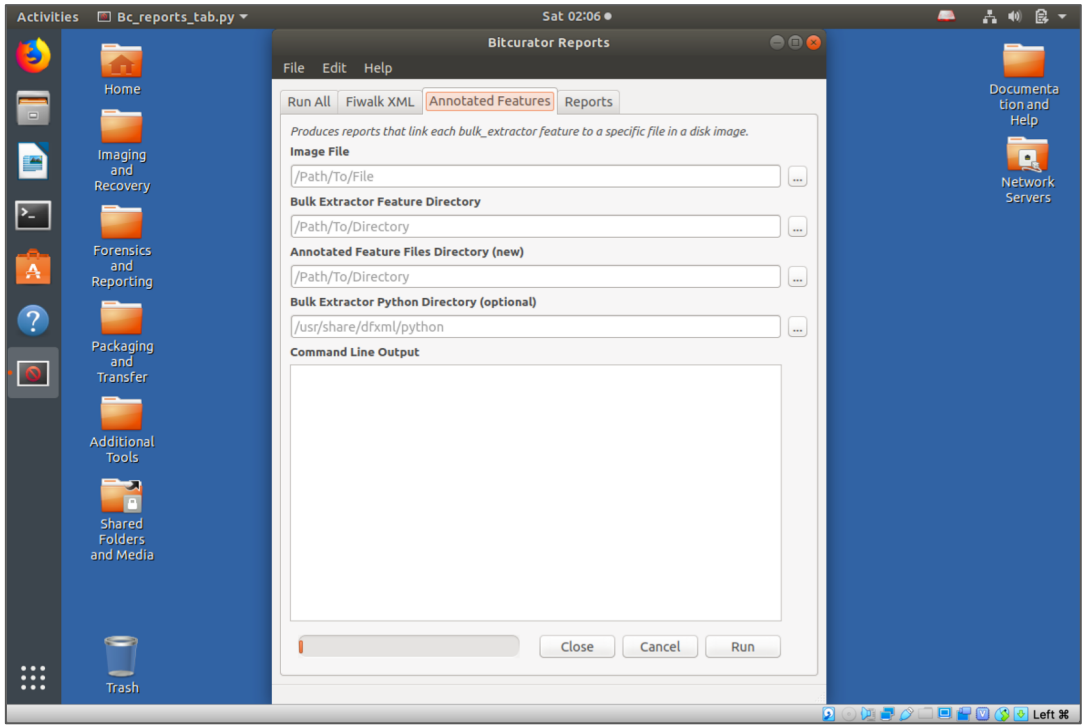
The resulting DFXML file can be found in the **SampleData** directory created earlier on the desktop. Examine the contents by double-clicking on it.

Matching Features to Files

Bulk Extractor extracts 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 Reports GUI you minimized earlier, or restart it from the **Forensics and Reporting** directory on the desktop. Click on the **Annotated Features** tab.



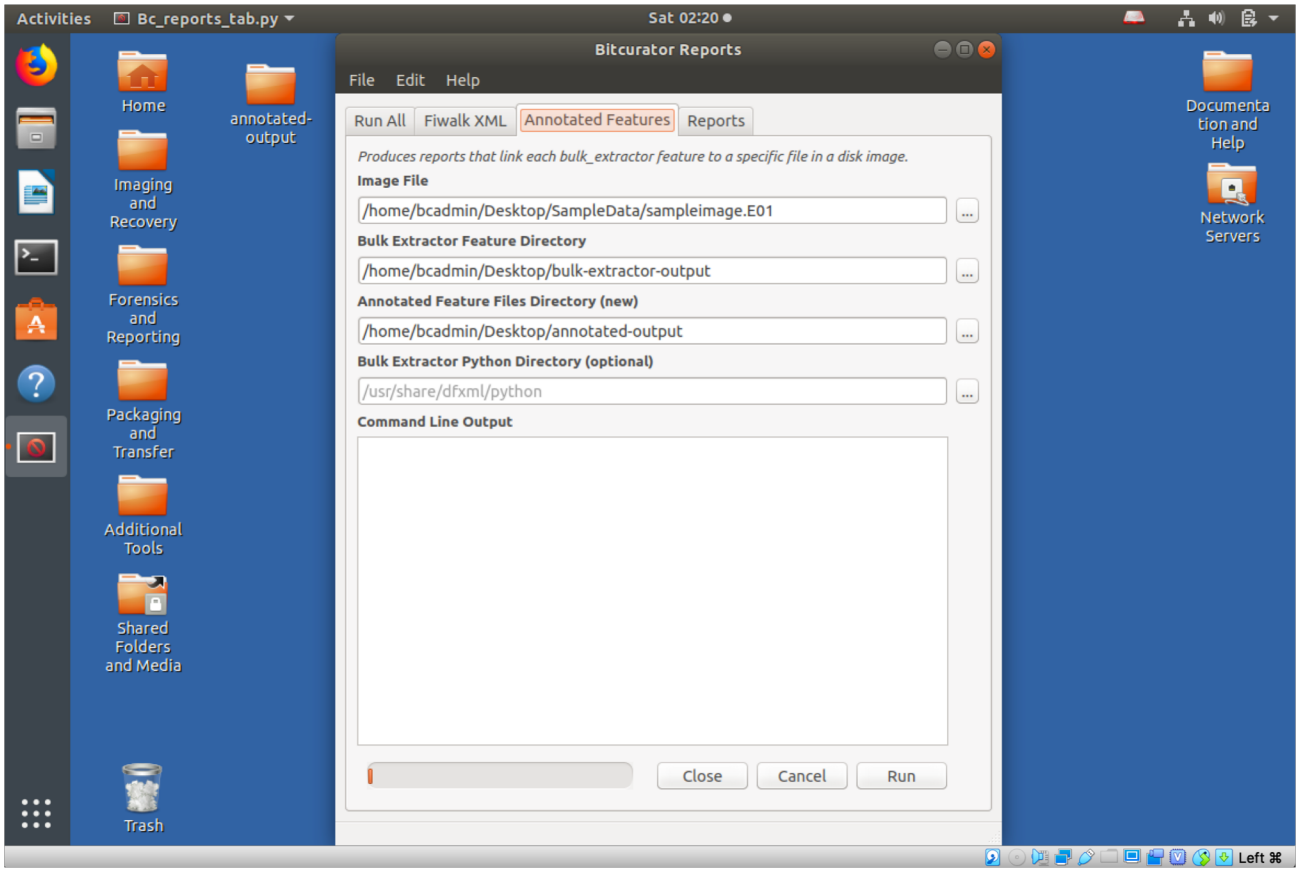
Matching Features to Files

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

1. Which feature reports to work from (the BitCurator Reports GUI uses all of them; to select specific reports, there is a command-line option.
2. The location of the **image file**.
3. The location of the **fiwalk** output.
4. The location of the **bulk_extractor** output.
5. Where to generate the reporting output. In this case, a new directory called **annotated-files** will be created in the SampleData directory on the desktop.

Selection of these items is shown in the following slides.

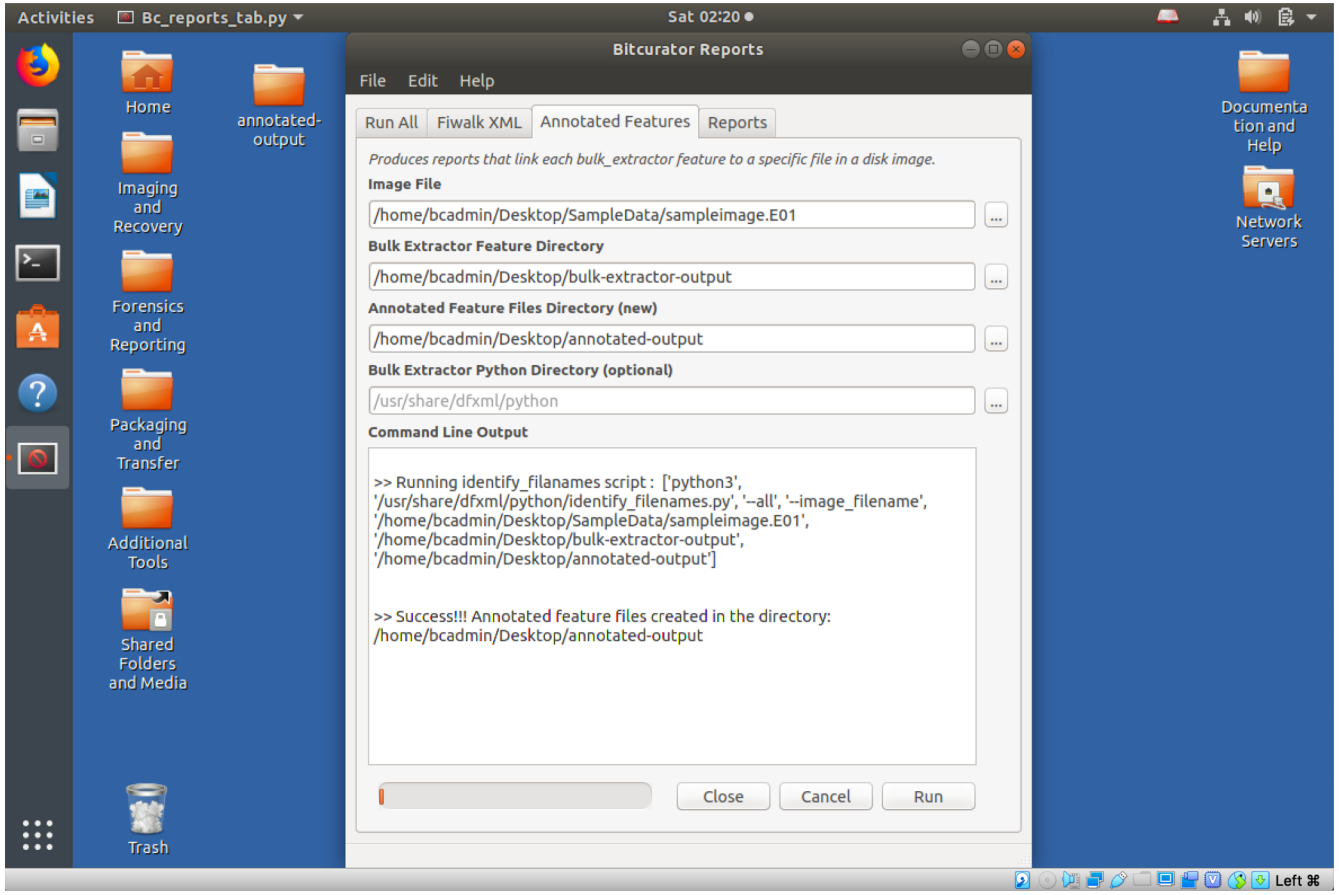
Matching Features to Files



In the screenshot shown here, the image file and bulk extractor output directory have been selected. A new directory has been specified inside the SampleData directory for the annotated features.

Tip: The Bulk Extractor python directory is optional.

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.

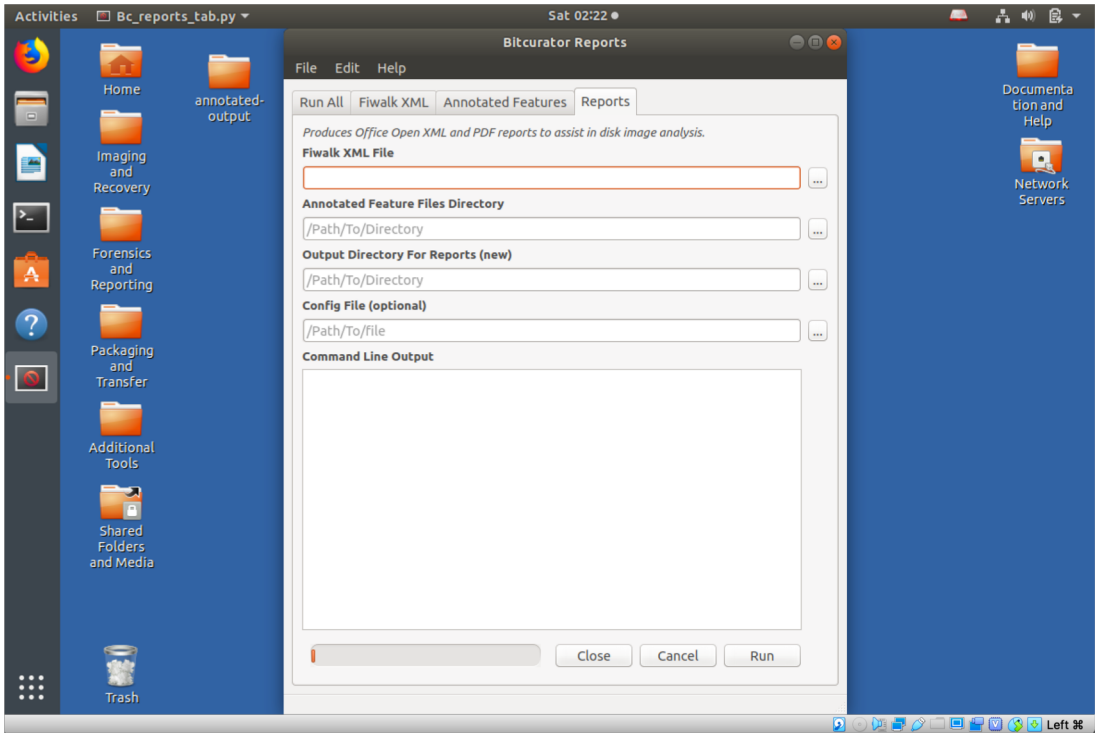
Generating BitCurator Forensic Reports

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.

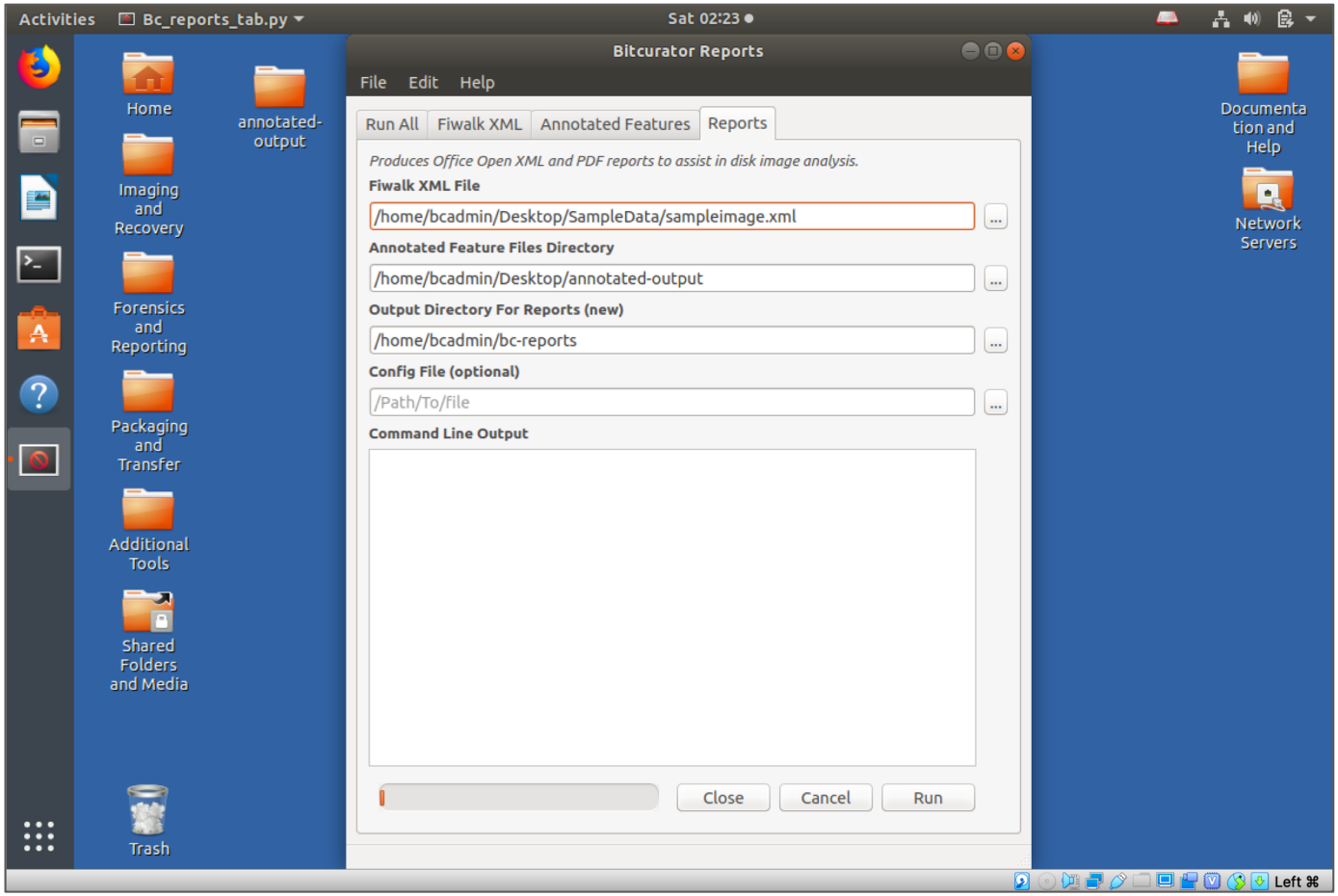
Click on the ‘Reports’ tab in the BitCurator GUI.

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.

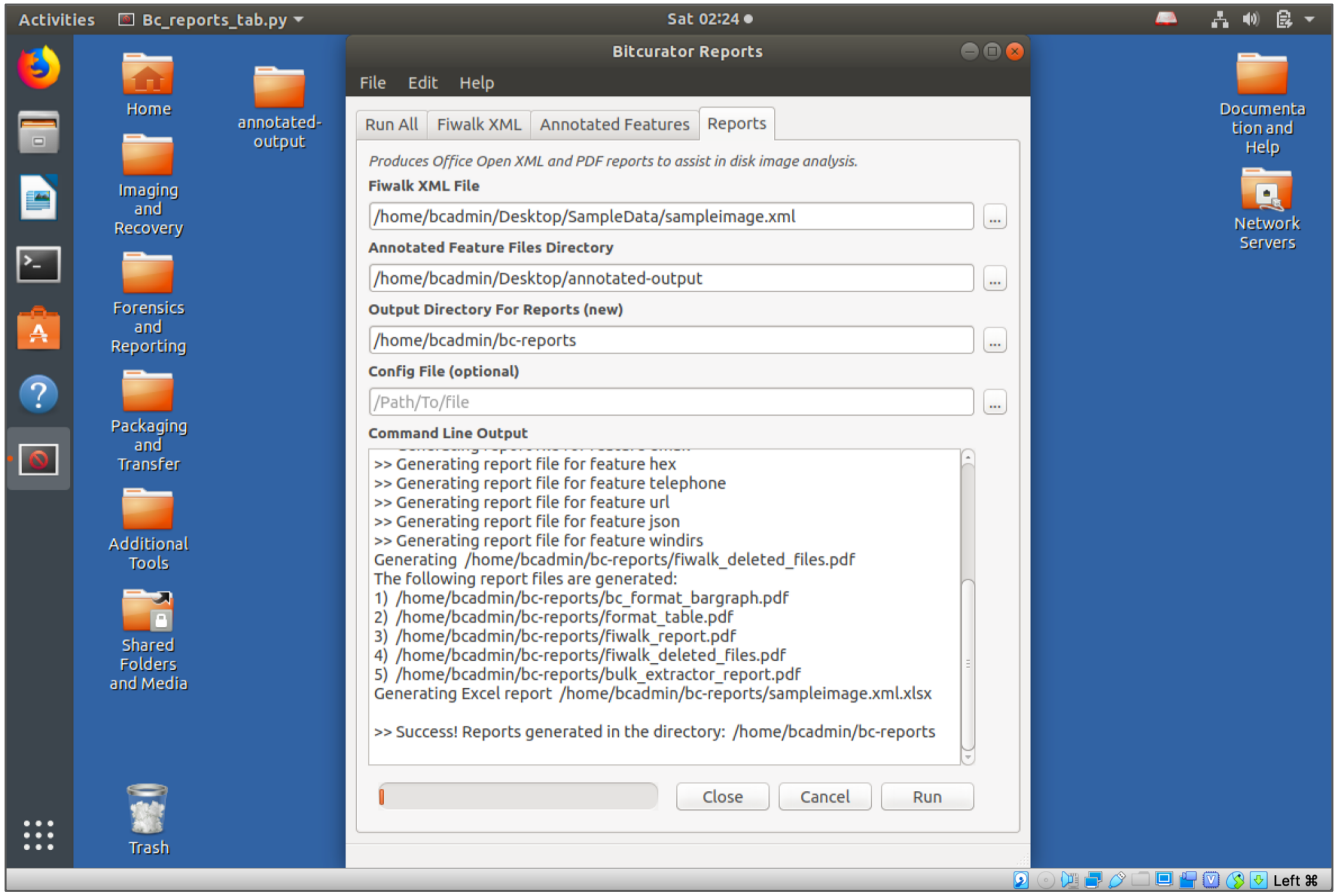


Generating BitCurator Forensic Reports



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 if any error occurs.

APPENDIX B: Using BitCurator tools on the command line

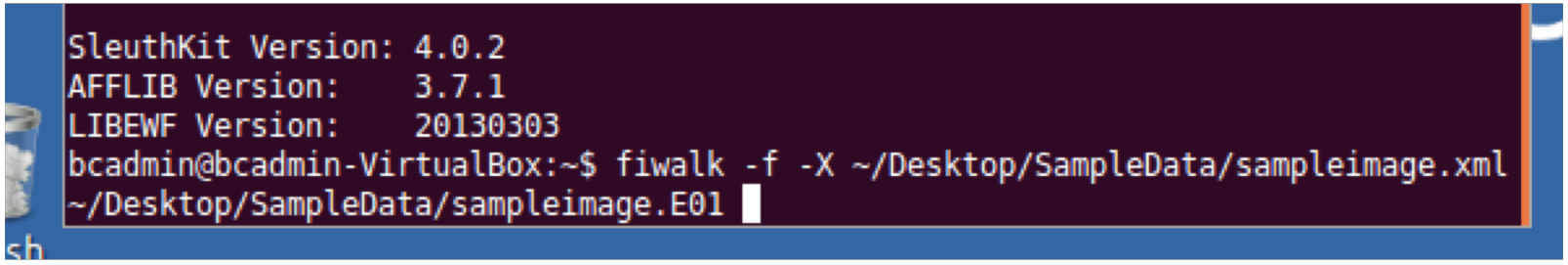
B1. Fiwalk: Producing a DFXML report of the File System Contents

The fiwalk tool 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.

Tip: "~/" at the beginning of each path 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 five 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_filename [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.

```
bcadmin@ubuntu:~$ python3 /home/bcadmin/Tools/bulk_extractor/python/identify_filenames.py
 --all --image_filename ~/Desktop/SampleData/sampleimage.E01 --xmlfile ~/Desktop/SampleDa
ta/sampleimage.xml ~/Desktop/SampleData/bulk-extractor-output ~/Desktop/SampleData/beannc
tated
Reading file map from XML file /home/bcadmin/Desktop/SampleData/sampleimage.xml
Processed 1000 fileobjects in DFXML file
feature_file: domain.txt
feature_file: email.txt
feature_file: hex.txt
feature_file: json.txt
feature_file: telephone.txt
feature_file: url.txt
feature_file: windirs.txt
******************************
** Total Features:    32343 **
** Total Located:     32343 **
******************************
bcadmin@ubuntu:~$
```

B3. BitCurator reporting: Running the Report Generator

The “Generate Report” program needs to know three 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.

You will see a several prompts for configuration. For now, use the defaults (typing “Y” and enter for the first prompt, and 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]:
```