Wasta-Offline Illustrated Guide For Administrators

by Bill Martin - 10 January 2019

This illustrated guide is for branch administrators in charge of maintaining and distributing "Full" Wasta-Offline mirrors on portable USB drives that can be circulated to remote regions where teams are using Wasta Linux. The regional teams can use the portable USB drives to update the software on their Wasta Linux computers, and then return the drives to the branch center for updating and reuse again in the regions. Another illustrated guide document called "How to Use the Full Wasta-Offline Mirror to Update the Software on Your Wasta Linux Computer" is designed for use by the regional team advisors, consultants or a trained team member.

To use this Guide for Administrators, you need to have enough technical knowledge to be able to run Linux scripts. Running the scripts can be done by double-clicking the script from within the File Manager interface – the easiest method to run a script in the Terminal - or by opening a Terminal where the scripts reside and running the scripts by typing the command line: **bash <script-name>** Either way, the scripts are relatively easy to use. Once started, the scripts are menu driven, and are designed to run with minimal interaction by the administrator. They do not require a lot of command line knowledge – just an occasional selection of a menu item, or 'y' or 'n' response at a plain language prompt.

This guide has 3 Initial Setup 'How to" sections and 2 Regular Use 'How to' sections. Click on the links below to go directly to the desired illustrated section.:

Initial Setup sections (one-time-only actions):

Regular Use sections (regular updating actions):

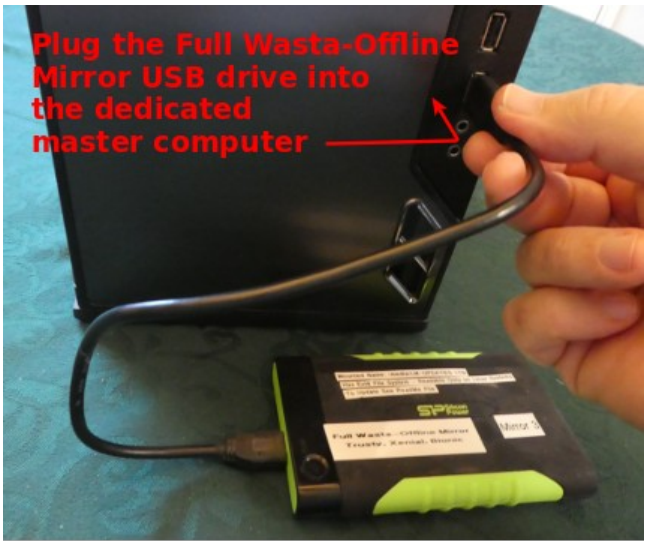
How to Make a Master Mirror by copying an Existing Mirror from an External USB Drive - using the script: *make_Master_for_Wasta-Offline.sh*

Making a master mirror is desirable since it can be used as the machine that regularly calls apt-mirror to keep its master mirror up-to-date, and can then be the source mirror from which one or more external USB drive(s) can be synchronized and brought up-to-date locally - without having to repeatedly download mirror updates for each copy of the mirror on the USB drives. The computer containing the master mirror can be set to automatically download mirror updates from a free local server (such as exists at Ukarumpa) that has up-to-date versions of the software mirrors mentioned above, or - if cheap "unlimited" Internet is available – downloading them directly from the Internet.

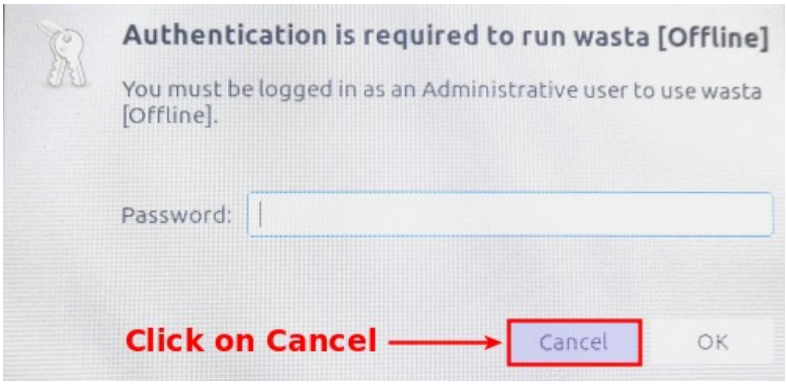
IMPORTANT: Before you create a full Wasta-Offline master mirror, be sure that the computer to host the master mirror has enough space available to hold a full Wasta-Offline mirror. The current space required exceeds 750GB. The computer should have a partition that has at least 1TB of free space for the full mirror (and future updates).

The ***make_Master_for_Wasta-Offline.sh*** script is provided to help you create the master mirror. Creating the master mirror is a one time process – you will only need to run the ***make_Master_for_Wasta-Offline.sh*** script **once**. Here are the steps for using this script:

Step 1: Plug in the portable USB drive that contains the Full Wasta-Offline Mirror (from Bill Martin) into the USB port of the master computer – the computer that is to host the "master mirror". This computer should be running Wasta Linux and have a good connection to the server or the Internet.

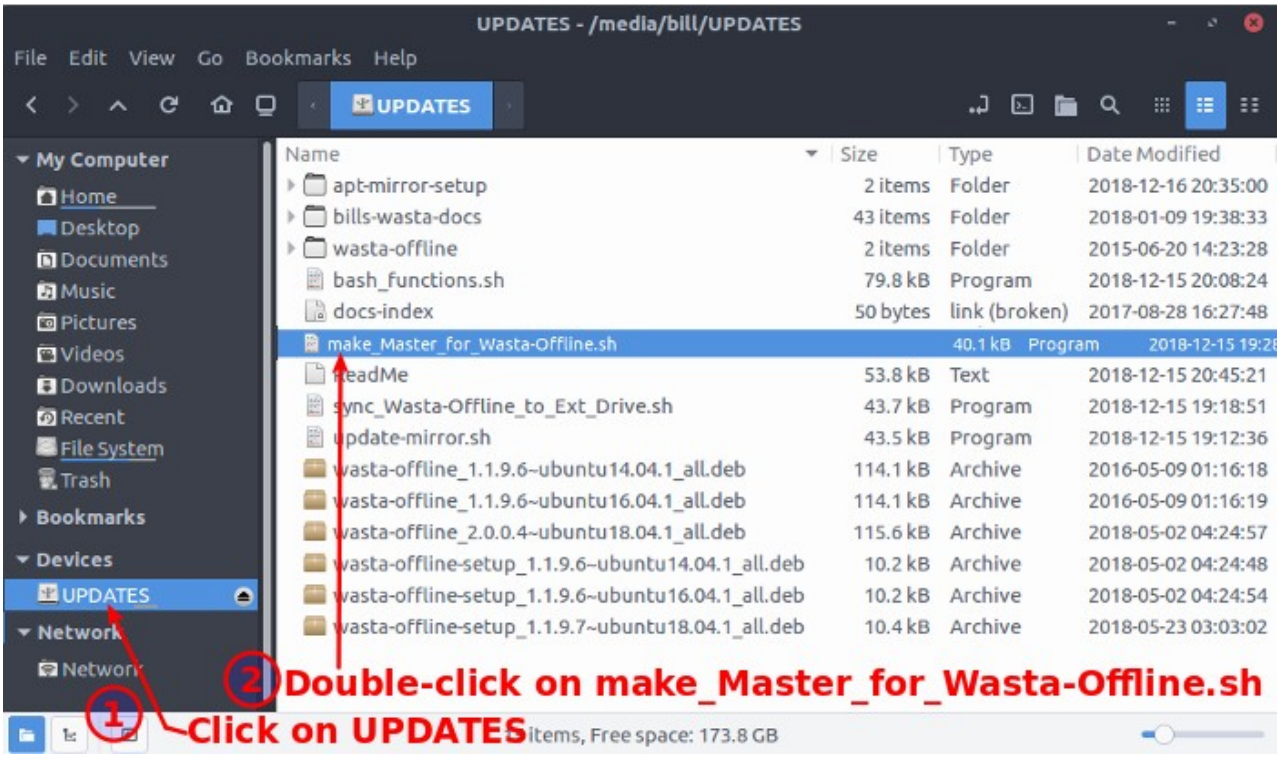


Step 2: Cancel the Wasta-Offline password dialog. After about 10 seconds the wasta-offline program will start automatically (if the computer is running Wasta Linux), and will display a dialog asking for your password.



This time click on **Cancel** to close the "Authentication" dialog and stop the wasta-offline program from running. We don't need the wasta-offline program running while we create the master mirror on the master computer.

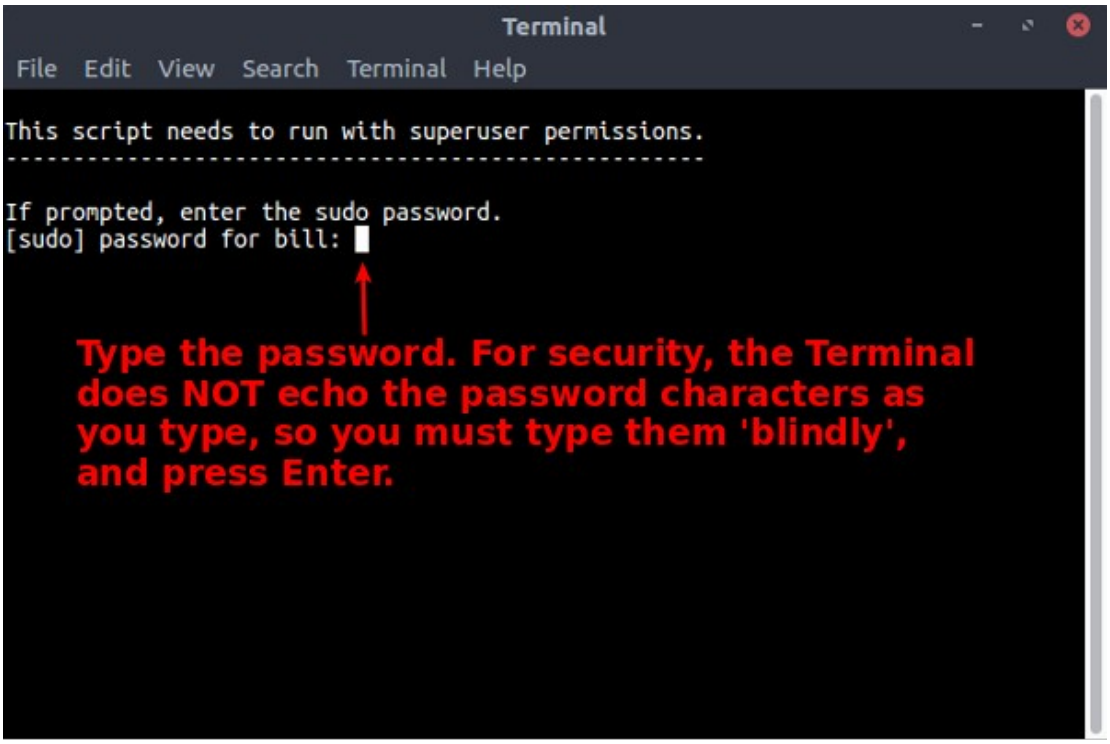
Step 3: Open the File Manager ("Files" on the panel) and locate the external USB hard drive that you just plugged in. It should be called something like **UPDATES** under the **Devices** list in the File Manager. Click on **UPDATES** to highlight the drive and display its contents in the right-hand pane of the File Manager:



Step 4: Run the *make_Master_for_Wasta-Offline.sh* script. On the UPDATES drive you should see a script there that is called: ***make_Master_for_Wasta-Offline.sh***. Double-click on the ***make_Master_for_Wasta-Offline.sh*** script (or highlight the ***make_Master_for_Wasta-Offline.sh*** file and press Enter). The File Manager will ask how the script should be run:

Select the "**Run in Terminal**" button.
The script should open a terminal window and start running.

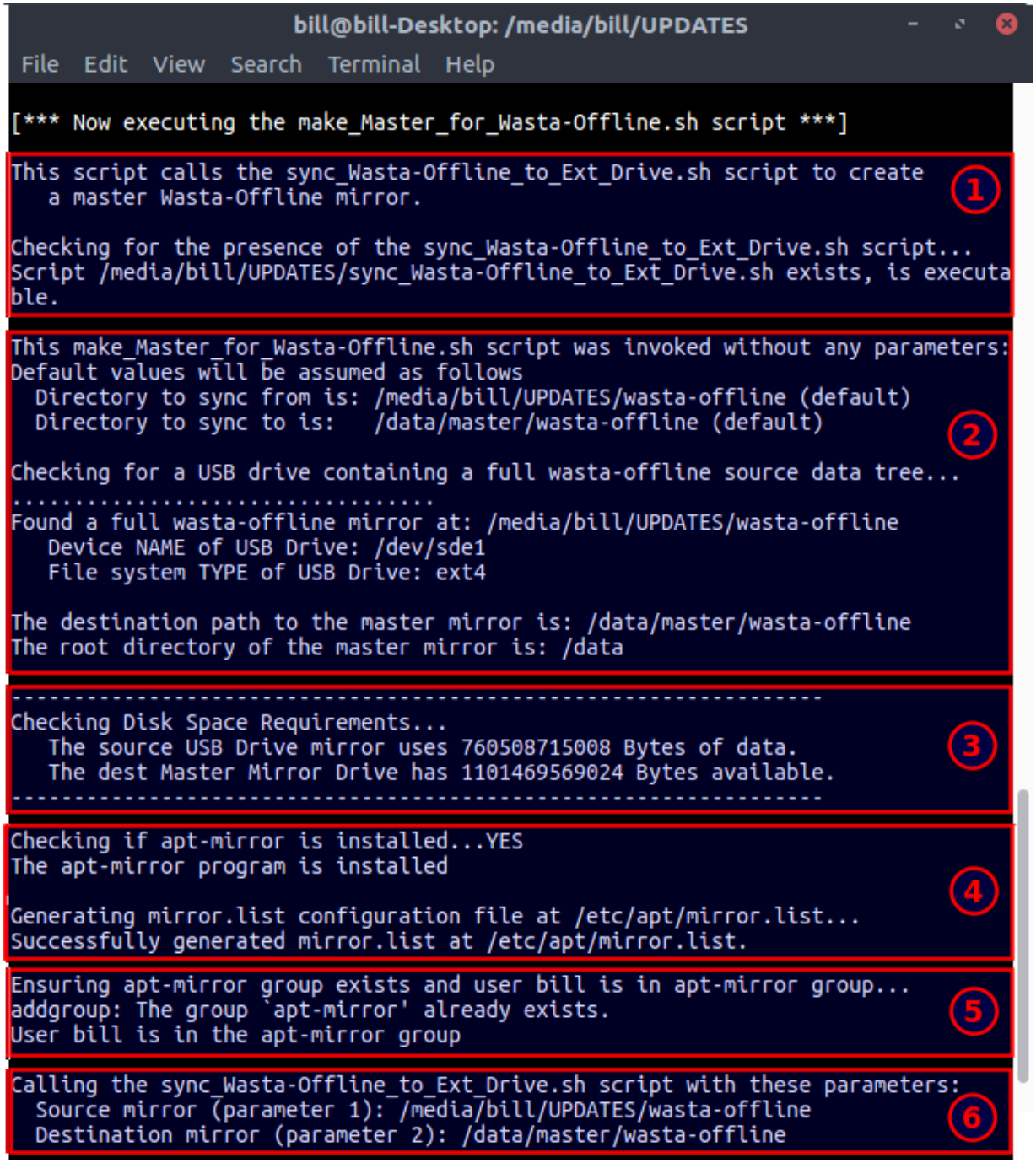
Note: If no terminal opens as illustrated below, but instead, an editor opens showing the contents of the script, see the Appendix section at end of this guide called "Setting File Manager Preferences to allow you to select *Run in Terminal* when double-clicking on the script".



Step 5: At the Terminal prompt, type the password for the user of the computer. As shown above, when you enter a password at the Terminal, the Terminal does not echo the password characters as you type, so you must type the password 'blindly', and press Enter. The script should then continue running.

Step 6: Understanding the script's console output. As the script runs, it makes a number of checks to ensure that a full Wasta-Offline mirror exists on an attached USB drive, and that a "master mirror" can be created on the dedicated computer. It compares how much space the full Wasta-Offline mirror uses on the USB drive with the space available on the dedicated computer that will receive the copy of the

full mirror which becomes the "master mirror." The script also ensures that the dedicated computer is configured to run the apt-mirror program that keeps the master mirror up-to-date. Here is the first part of the ***make_Master_for_Wasta-Offline.sh*** script output as the script runs and performs the checks (no user interaction is normally required):



Below is an explanation of the 6 numbered blocks shown in the screen shot of script output above:

This part of the script checks that its companion script ***sync_Wasta-Offline_to_Ext_Drive.sh*** exists and is executable. That companion script needs to be available and executable because it does the main work of copying/syncing all of the mirror data and files from the USB drive to the dedicated computer on which the master mirror is being created. If the companion scrip is not found you will see the following warning, and the script will abort its run:

```
****** WARNING ******
Cannot find the /media/bill/UPDATES/sync_Wasta-Offline_to_Ext_Drive.sh script
This script requires that sync_Wasta-Offline_to_Ext_Drive.sh be available.
****** WARNING ******
"Aborting...
```

The script then checks for any parameters passed to the script, and determines the location path of the 'source' data and the default (or designated) path of the 'destination' data. It check to ensure that the 'source' data really has the full Wasta-Offline mirror data, and that the dedicated computer can accept the data for creating the master mirror at the root directory destination. It also checks to make sure that the 'destination' location is NOT the same as the 'source' data location (the mirror data cannot be copied to itself!). If no USB drive was found having a full Wasta-Offline Mirror on it the following warning and abort message will be given:

```
****** WARNING ******
"No USB drive was found having a full Wasta-Offline Mirror.
Have you plugged in a USB drive containing a full Wasta-Offline Mirror on it?
Cannot create a master mirror without a full mirror on a USB drive to copy from.
Please plug in an existing 'Full Wasta-Offline Mirror' USB Drive and try again!
****** WARNING ******
"Aborting...
```

If the 'destination' location is the SAME as the 'source' data location, the following warning and abort message will be given:

```
****** WARNING ******
The destination location is the SAME as the USB drive.
It makes no sense to copy mirror data from/to the same location!
Please supply a destination path for the master mirror is not the same as the
the path to the USB Drive and try again!
****** WARNING ******
Script Usage:
make_Master_for_Wasta-Offline.sh [no parameters]
  <source-dir-path> defaults to /media/$USER/<DISK_LABEL>/wasta-offline
  <destination-dir-path> defaults to /data/master/wasta-offline
make_Master_for_Wasta-Offline.sh [<destination-dir-path>]
  <source-dir-path> defaults to /media/$USER/<DISK_LABEL>/wasta-offline
make_Master_for_Wasta-Offline.sh [<source-dir-path>] [<destination-dir-path>]
Aborting...
```

The script then compares how much space the full Wasta-Offline mirror uses on the USB drive with the space available on the dedicated computer that will receive the copy of the full mirror which becomes the "master mirror." If the USB's mirror data is GREATER than the space available on the dedicated computer, the following warning and abort message will be given:

```
****** WARNING ******
The USB data used is GREATER than available space at the master mirror!
This script cannot create a master mirror in the space available.
Please allocate disk space for the master mirror of at least 1TB and try again!
****** WARNING ******
Aborting...
```

If the space available on the dedicated computer is sufficient to hold the full mirror data, but the available space is somewhat less than 1TB, the script will continue, but the following warning will be given:

```
WARNING: The destination Master Mirror Drive space is smaller than 1TB!
You have sufficient space now, but your master mirror may run out of space.
We recommend that you allocate at least 1TB for the master mirror!
```

This check ensures that apt-mirror is installed, and there is a properly configured **mirror.list** file at **/etc/apt/mirror.list** that defines (for the **apt-mirror** program) the Linux repositories composing the full Wasta-Offline mirror. If apt-mirror is not installed, you will see output here related to the installation of the apt-mirror program. If apt-mirror cannot be installed or the mirror.list path cannot be generated for some reason, the following warning(s) and abort message will be given:

```
****** WARNING ******
Error: Could not install apt-mirror/generate mirror.list at /etc/apt/mirror.list.
****** WARNING ******
Aborting...
```

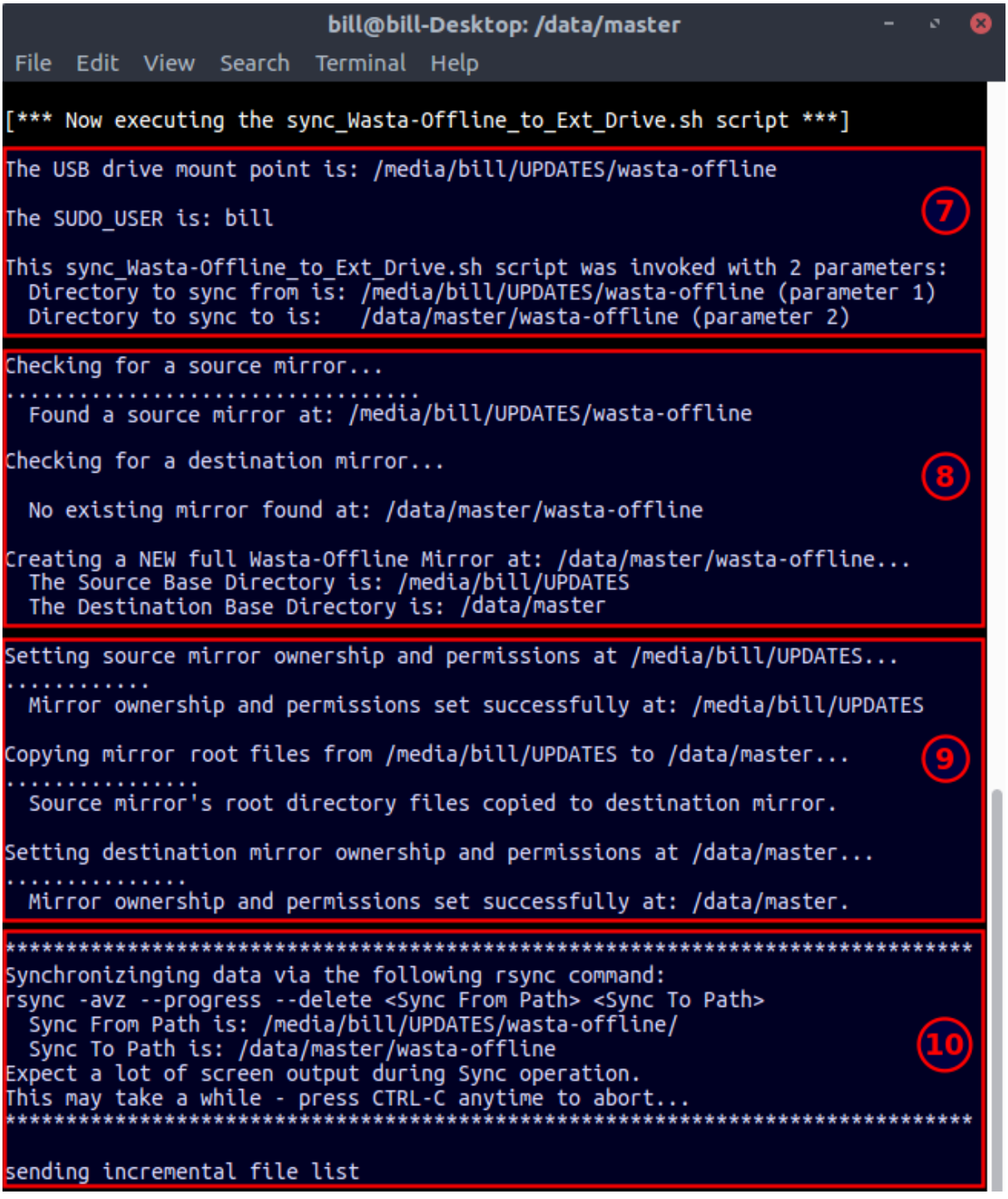
This check ensures that the dedicated computer has a group named **apt-mirror**, and that the user is a member of that apt-mirror group. If the current user could not be added to the apt-mirror group, the script will continue, but the following warning message will be given:

```
WARNING: Could not add user: <user-name> to the apt-mirror group
```

This console output indicates that the ***make_Master_for_Wasta-Offline.sh*** script is ready to call the companion script ***sync_Wasta-Offline_to_Ext_Drive.sh*** that will do the heavy lifting of copying all of the Wasta-Offline mirror data from the Source mirror (at the path shown for parameter 1), to the Destination mirror (at the path shown for parameter 2).

Step 7: Understanding the companion script's console output. If the initial checks are successful, the ***make_Master_for_Wasta-Offline.sh*** script at this point calls the companion script named: ***sync_Wasta-Offline_to_Ext_Drive.sh.*** That companion script now starts running and adding its output in the Terminal, which includes some of the same checks that the ***make_Master_for_Wasta-***

`Offline.sh` script made. Here is another part of the console output, now being produced by that companion script (again no user interaction is normally required):



Here (below) is an explanation of blocks numbered 7-10 shown in the screen shot of the continuing script output above - now coming from the companion ***sync_Wasta-Offline_to_Ext_Drive.sh*** script:

After announcing that it is now executing, the companion script ***sync_Wasta-Offline_to_Ext_Drive.sh*** makes some of the same checks that the calling script, ***make_Master_for_Wasta-Offline.sh,*** made earlier. It takes note of the two parameters that were passed to it by the calling script, interpreting them as the path to sync mirror data from (the 'source'), and the path to sync mirror data to (the 'destination').

The companion script next checks to ensure that the designated 'source' data really has the full Wasta-Offline mirror data that can be used to create the master mirror. If a full Wasta-Offline mirror is not found at the 'source' a warning and an abort message is given:

```
****** WARNING ******
Could not find a source mirror at: /media/$USER/<DISK_LABEL>/wasta-offline
Therefore, cannot update the USB mirror from this computer.
****** WARNING ******
Aborting...
```

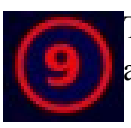
The companion script then examines the location designated as the 'destination' for the mirror data, first checking whether an existing Wasta-Offline mirror is at the 'destination' or not. If no existing mirror is found (the usual case when initially creating a master mirror), the script creates an initial (empty) directory tree for receiving the full mirror data. If an empty tree directory cannot be created for some reason a warning and an abort message is given:

```
****** WARNING ******
Cannot create mirror directories at $COPYTODIR - is the Drive writeable?
You might try rebooting the computer and running this script again.
****** WARNING ******
Aborting...
```

If (per chance) there is an existing Wasta-Offline mirror already located on the dedicated computer at the designated location, the script will examine the existing mirror to see if the existing mirror is OLDER, NEWER, or the SAME, and will prompt the user – asking whether to proceed on not. This check to see if the existing mirror is OLDER, NEWER, or the SAME, is generally bypassed when the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script is called by the ***make_Master_for_Wasta-Offline.sh*** script, so we make no more mention of it here. Normally, you should see the following output when creating a new master mirror at the designated destination:

```
Checking for a destination mirror…
  No existing mirror found at: /data/master/wasta-offline

Creating a NEW full Wasta-Offline Mirror at: /data/master/wasta-offline…
  The Source Base Directory is: /media/<user-name>/<DISK_LABEL>
  The Destination Base Directory is: /data/master
```



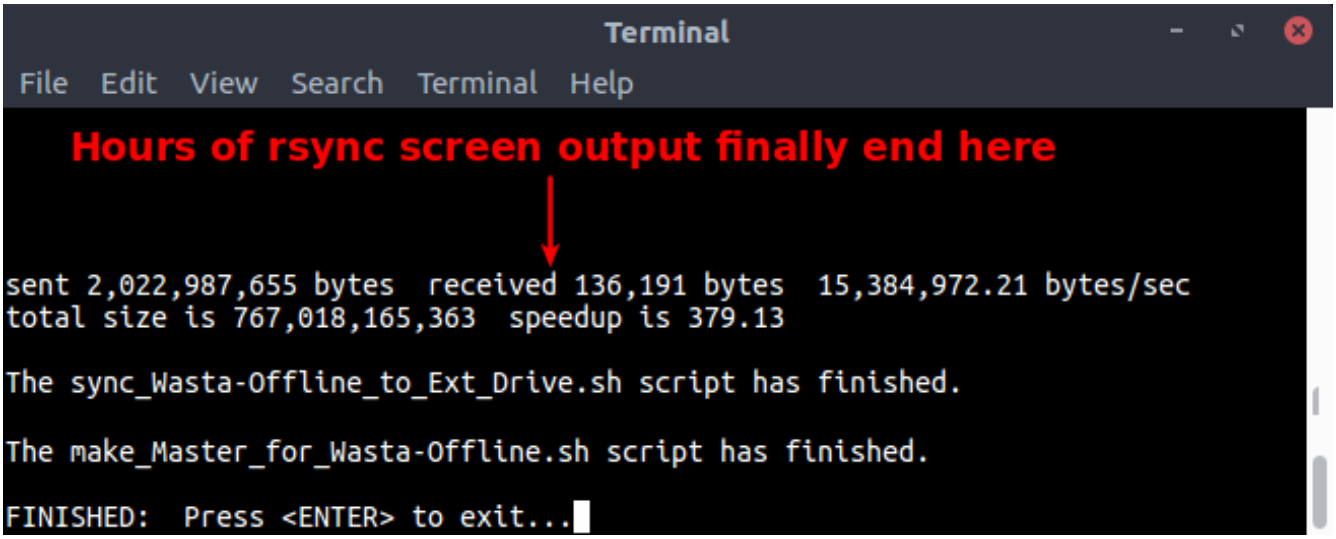
The companion script then ensures that the ownership and permissions of the source mirror are set appropriately (ownership set to apt-mirror:apt-mirror) and permissions set to read-

write for everyone. It copies the root directory files from the USB 'source' drive (/media/<user-name>/<DISK_LABEL>) to the dedicated computer's base location (/data/master). Then it ensures that the ownership and permissions of the destination mirror tree are set appropriately (ownership set to apt-mirror:apt-mirror) and permissions set to read-write for everyone.

The companion script then outputs this block of output lines, indicating how it will call the Linux **rsync** command – the command that does the actual copying/syncing of files from the source mirror to create the mirror at the destination. The options used in the call of rsync are:

```
-avz        [archive mode/recursive (-a), verbose (-v), compress (-z)]
–progress   [show progress during transfer]
--delete    [delete extraneous files from dest dirs – removes outdated files]
```

The rsync command then begins its copying and syncing process. When creating a new full Wasta-Offline mirror – amounting to over 750GB of data - rsync will generate hours and hours of console output! It will do its work unattended, so be patient and check on its output after a few hours of run time. Eventually rsync will complete its work and both the calling script and this companion scrip will finish – with this console message similar to the following:

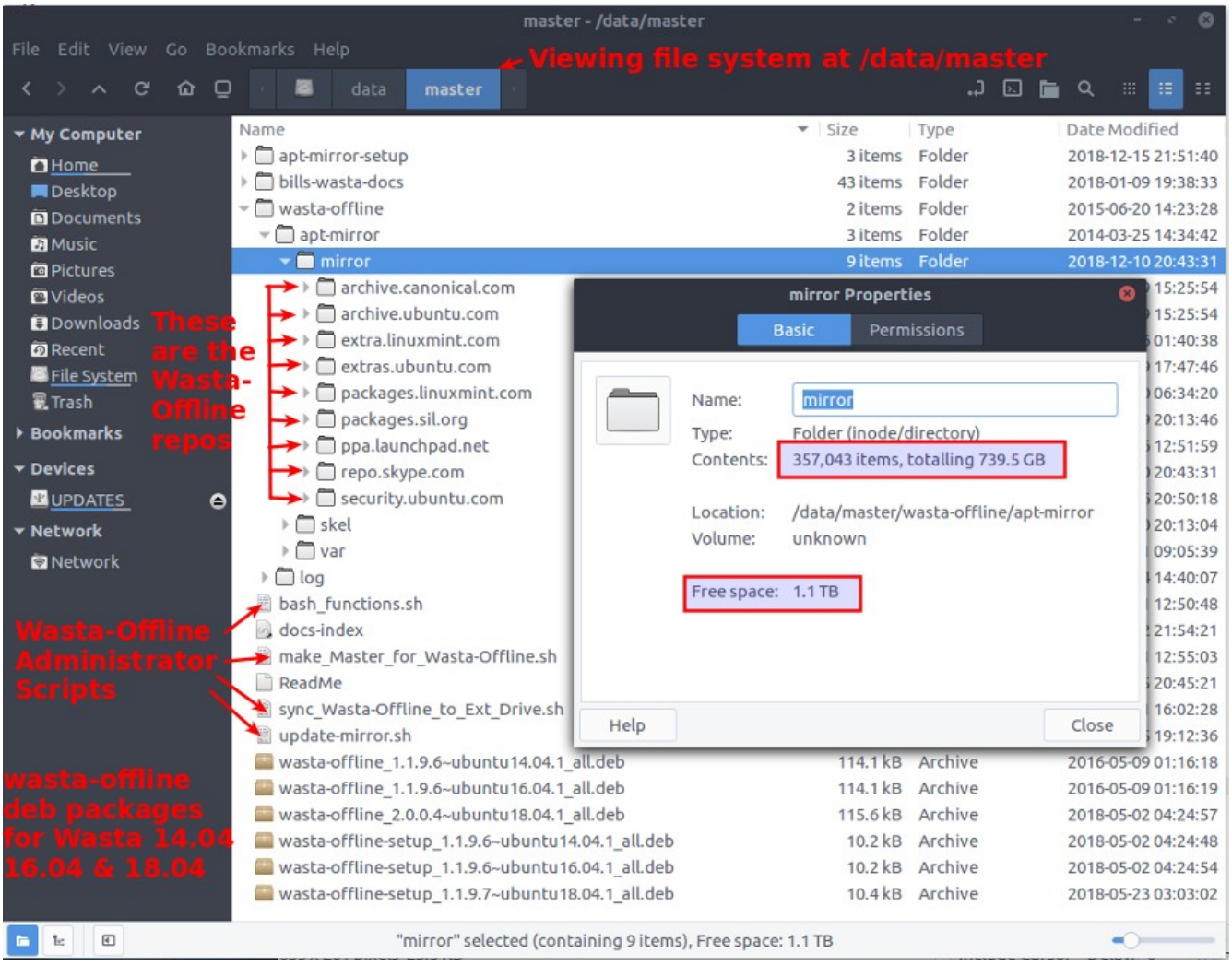


If rsync encountered an error or could not complete its copying process, you will see the following warning and abort message before the "FINISHED" message in the console output:

```
****** WARNING ******
Could not rsync the mirror data to $COPYTODIR!
****** WARNING ******
Aborting...
```

Step 8: Verify that a full Wasta-Offline Mirror was created on the dedicated computer. Run the File Manager again (if it is not still running), and navigate to the location where the master mirror was created (defaults to /data/master). Here is a screen shot of what you should see with the File Manager

viewing the contents of the /data/master folder, and examining the Properties of the 'mirror' directory (right-click on the 'mirror' directory and select Properties):



The above screen shot gives a picture of the contents and extent of the new master mirror on the dedicated computer. The **mirror Properties** dialog should indicate that there are over 350,000 items in the mirror itself - representing about 740GB of space used currently in the master mirror. The master mirror illustrated above was created on a partition that is about 2TB in size – hence the available 'Free space' shown above is 1.1TB – plenty of room for the gradual growth of the Wasta-Offline Full Mirror. For a master mirror created on a 1TB partition, you should have around 200GB of Free space remaining after the master mirror has been created.

When creating a master mirror using a portable USB drive from Bill Martin containing the Full Wasta-Offline Mirror, the base directory of the master mirror (visible above at /data/master) will contain a number of useful items in addition to the repositories composing the wasta-offline mirror. Those items can be seen in the above screen shot. They include all of the wasta-offline administrator scripts (as a git

repository clone), deb packages for wasta-offline and wasta-offline-setup for Wasta 14.04, Wasta 16.04 and Wasta 18.04 (the update-mirror.sh keeps these updated to the latest version), and a directory called 'bills-wasta-docs' that contains over 160MB of Bill's documents related to Wasta Linux and Wasta-Offline.

Step 9: Move on to the next steps.

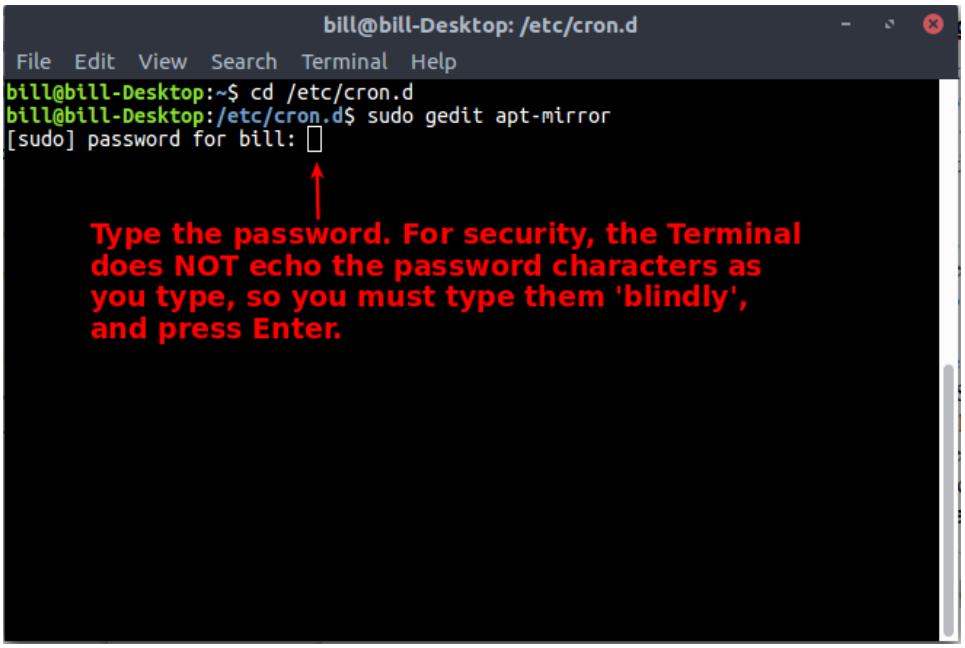
- **Configure the master computer to get daily updates from the local server**. Now that you have created a master mirror containing the Full Wasta-Offline Mirror, you may want to configure the master computer to get daily updates from the local server. The configuration process is illustrated in the next How to section called "How to Configure the Master Mirror computer to Get Mirror Updates from the Ukarumpa Server (or Internet) on a Scheduled Daily or Periodic Basis."
- **Periodically Update the Master Mirror from the local server**. Whether or not you configure the master computer to get daily updates from the local server, you can always get updates from the local server on demand using the **update-mirror.sh** script as described in the Regular How to section of this document called "How to Keep the Master Mirror Up-To-Date from the branch's Intranet (or the Internet) - using the script: update-mirror.sh".
- **Periodically Update the full mirrors on the portable USB drives, syncing them from the master mirror.** The real payoff of having a master mirror comes with the ease and efficiency of keeping an arsenal of portable USB drives containing full Wasta-Offline mirrors up-to-date. The portable drive updating process is described in the Regular How to section called "How to Update the Mirrors on Portable USB drives – synchronizing them with the master mirror - using the script: ***sync_Wasta-Offline_to_Ext_Drive.sh***".

How to Configure the Master Mirror computer to Get Mirror Updates from the Ukarumpa Server (or Internet) on a Scheduled Daily or Periodic Basis

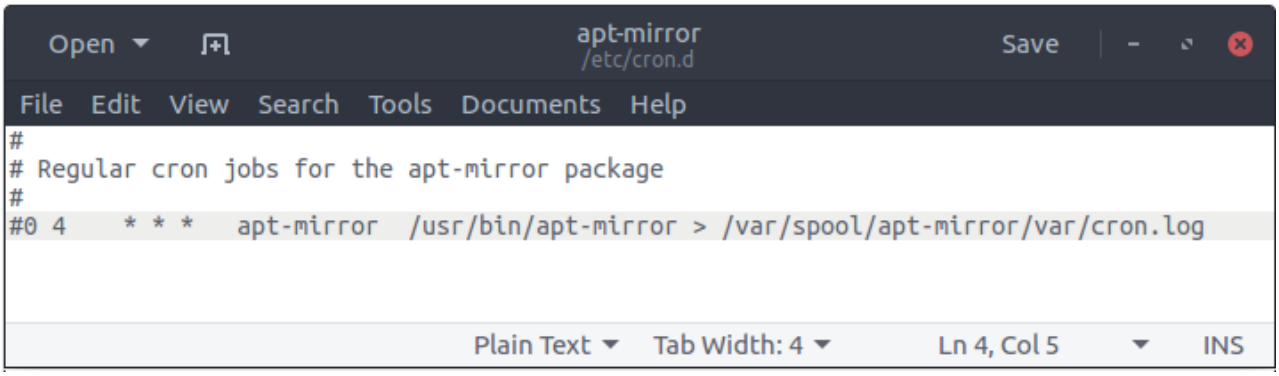
You should follow this 'How to' **only AFTER** you have successfully followed the "How to Make a Master Mirror by copying an Existing Mirror from an External USB Drive" section above to create a Master Mirror. Once a master mirror exists on a Linux computer that is attached to the network, updates to that mirror can be scheduled to happen daily as a **cron** job. This is a one-time task on the master computer.

It is best to schedule the downloading of software updates on a daily basis. It should be scheduled at a time when the master computer is already turned on and all booted up. If the network is often busy or overloaded during the day, automatically updating the master mirror can be scheduled during the night hours when the server is not so busy – as long as the dedicated master computer is left running. When the **apt-mirror** program is installed on a computer, it creates a sample cron job template file at: `/etc/cron.d/apt-mirror`. Here are the steps showing how to tweak that apt-mirror template file to cause the cron program to call apt-mirror at a certain time every day:

1. Open a Terminal window (`CTRL+ALT+T`).
2. Change to the `/etc/cron.d/` directory: **`cd /etc/cron.d`**
3. Edit the apt-mirror file as root by calling: **`sudo gedit apt-mirror`**
   Type your **password** and press **Enter** (the password is not echoed, you must type it blindly and then press the Enter key).



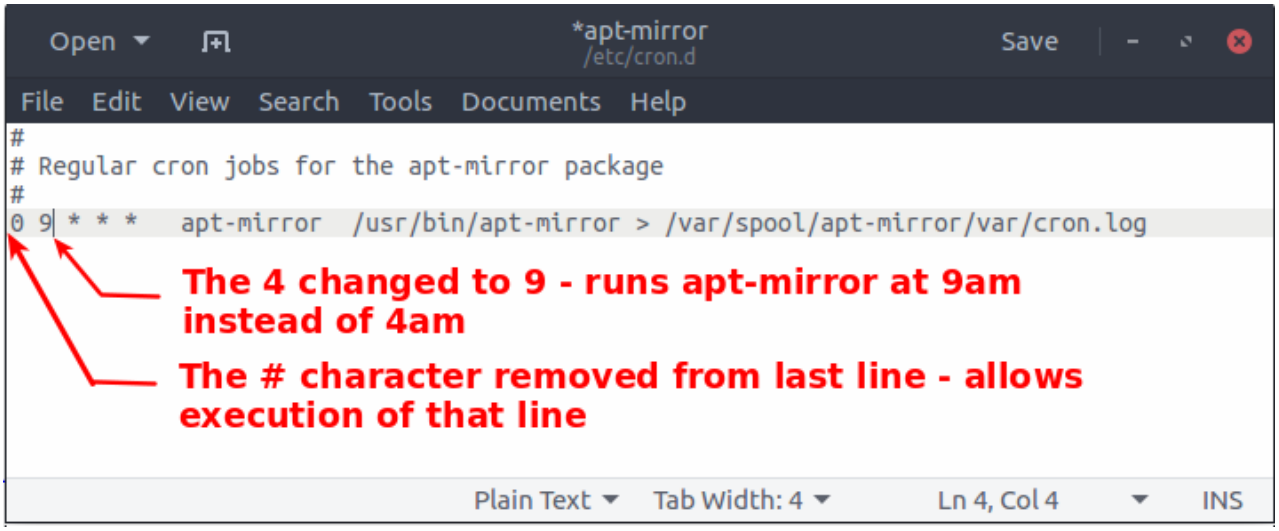
The content of the apt-mirror file will initially look like this **within the gedit editor**:



4. **Editing Instructions:**

Delete the # (comment) character from the beginning of the last line

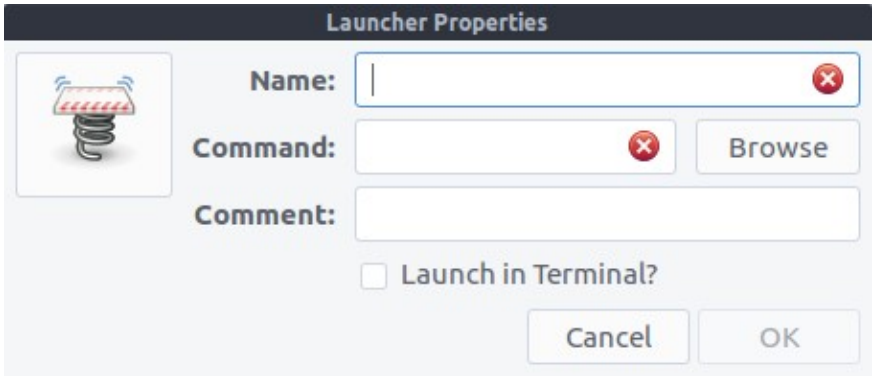
Change the **4** to an hour of the day the would be a good time for cron to run the apt-mirror program – I'll assume that might be something like **9** for 9am. The default of **4** represented 4:00am as the time to run apt-mirror. A 4:00am run time would be OK if you regularly (or prefer to) leave the computer running overnight. But, you may want to change the **4** to a different time such as **9** which will represent 9:00am - a time more suitable for a computer that is generally running during the work day but not at night. The hour you enter should be based on a 24 hour clock, so 3:00pm would be 15. The initial 0 at the beginning of the line represents the seconds past the hour, which you can leave at 0. After editing the whole file should look like this screen shot:



5. After doing the above edits to the apt-mirror file, **save** the file and **quit gedit**.

Notes:

- Editing/Activating the cron job as illustrated above need only be done once on a given master computer. Cron will attempt to run the apt-mirror program at the designated time each day the computer is on and booted up.
- When cron triggers the running of the apt-mirror program at the designated time of day, apt-mirror runs entirely in the background as it updates the master mirror. You should regularly check the apt-mirror cron job's log to make sure that it is doing its updates as scheduled. The log file is at: **`/var/spool/apt-mirror/var/cron.log`** If you run the File Manager (Nemo) and double-click on the **cron.log** file at the above location, it should open in gedit and you can check the date of the last run of apt-mirror, and see the console output of the apt-mirror run as well as the output of the **postmirror.sh** script - checking for any reported errors.
  To make it easy to check the cron.log, create a launcher on the Desktop by right-clicking on the Desktop and selecting "Create a new launcher here...". The Create Launcher dialog appears:



  Type a name for the desktop launcher. Something like **Apt-Mirror Cron Log.**
  Type the following command into the "Command" edit box. To make it easy you can just select and copy, then paste the following line into the "Command:" edit box:
  `/usr/bin/gedit /var/spool/apt-mirror/var/cron.log`
  Note that there is a **space** between gedit and /var… Click **OK** to save the desktop launcher. You should now be able to double-click on the desktop launcher to open the cron.log in the gedit editor from the desktop.
- Each time the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script is run, it calls the Linux **chown** command to ensure that all Wasta-Offline Mirror files are set to have **apt-mirror** as their 'owner', and also **apt-mirror** as their 'group' membership (designated together as **apt-mirror:apt-mirror**). This apt-mirror designation on mirror files is necessary for the cron job to run apt-mirror on the schedule you have configured in the /etc/cron.d/apt-mirror configuration file. When you first create a master mirror using the ***make_Master_for_Wasta-Offline.sh*** script, it in turn invokes the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script to copy the mirror

contents from the external USB drive to the destination location of the master mirror on the master computer.

- With a cron job updating a master Wasta-Offline Mirror on a regular schedule, it is then a fairly quick job to synchronize one or more external USB drive(s) to bring them up-to-date, following the steps in the "How to" section of this guide called: "How to Update the Mirrors on Portable USB drives – synchronizing them with the master mirror".

How to Add or Remove repositories from the Master Mirror

NOTE: Bill keeps the Wasta-Offline related scripts up-to-date, adding new distributions as they appear, and removing old distributions that are no longer supported. So, generally speaking, if you desire to change the content of the "full" Wasta-Offline Mirror, it would be best to ask Bill Martin to make the adjustments necessary, and he will adjust the scripts at their central location on GitHub where they are freely available to anyone wishing to use them (at: https://github.com/pngbill-scripts/wasta-scripts). Just communicate with him what Linux repositories and/or distributions you feel should be added or removed from the full Wasta-Offline Mirror, and he can make the necessary adjustments to the scripting function that configures the composition of the full mirror. You can then use the git program to retrieve any revisions Bill has made to those scripts by following the Appendix section called: "Using git to keep the wasta-offline scripts up-to-date from GitHub".

This "How to" tells how to temporarily tweak the script to add or remove repositories that compose the "full" Wasta-Offline Mirror, in case you want to experiment with the composition of the mirror.

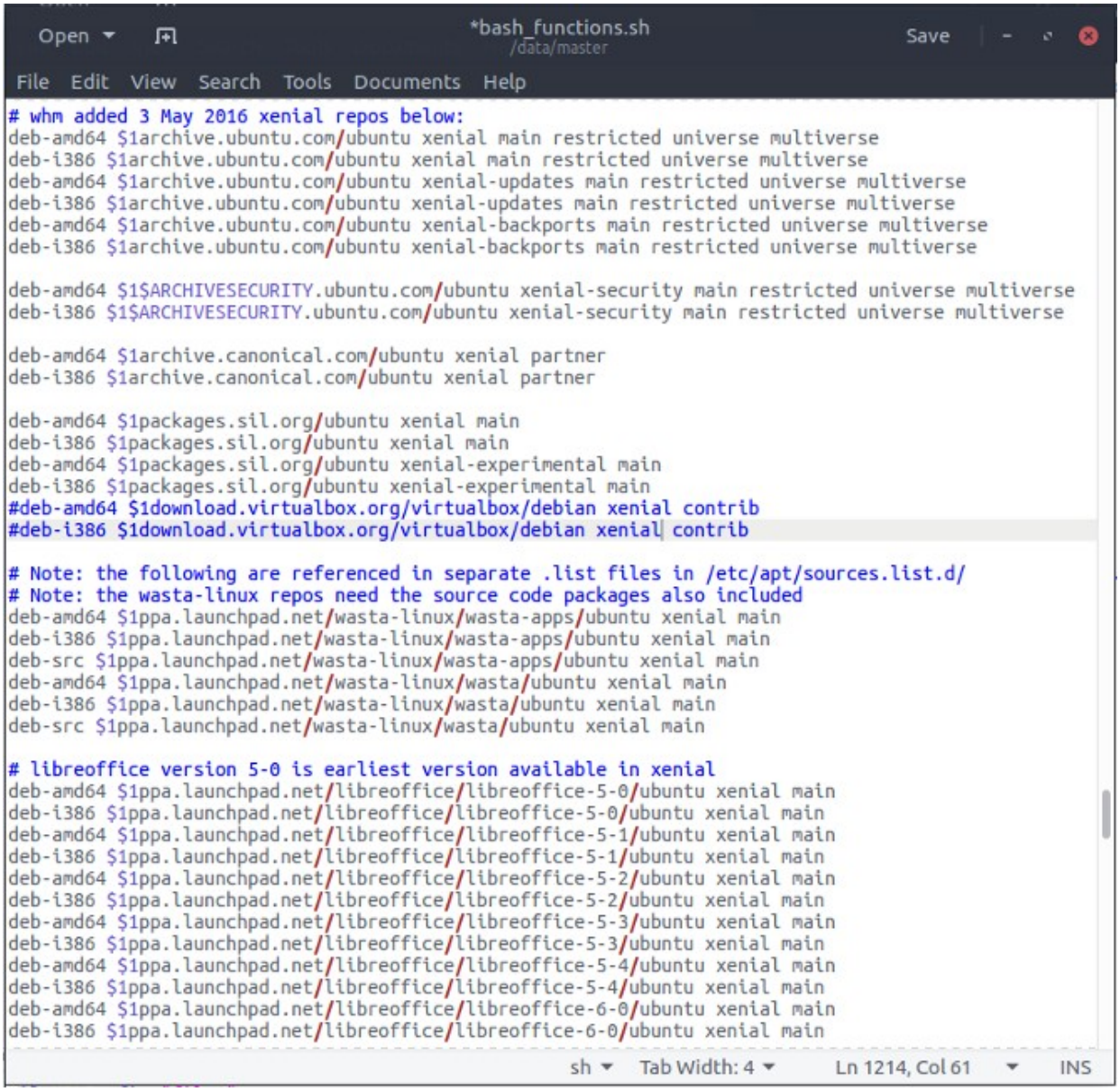
The overall content of an apt-mirror generated software mirror is determined by the contents of a configuration file called **`mirror.list`** that is located at: **`/etc/apt/mirror.list`** on the computer that runs the **apt-mirror** program. The provided script called ***update-mirror.sh*** takes care of adjusting the computer's **`mirror.list`** file - ensuring that it points to the appropriate external software mirrors. The actual location where the **`mirror.list`** file's software sources point to, depends on the user's choice at the following menu when the ***update-mirror.sh*** script is run:

```
MAIN MENU:

********************************************************************************
Where should the Wasta-Offline Mirror get its software updates?
     1) Get software updates from the SIL Ukarumpa local server.
     2) Get software updates directly from the Internet (might be expensive!)
     3) Get software updates from a custom network path that I will provide.
     4) Quit - I don't want to get any software updates at this time.
   Please press the 1, 2, 3, or 4 key, or hit any key to abort - countdown 60
********************************************************************************
```

The **update-mirror.sh** script actually uses a **bash function** called **generate_mirror_list_file ()** which is located in the **bash_functions.sh** file included with the other scripts in this distribution. It is in the **generate_mirror_list_file ()** function that tweaks can be made to determine what content gets generated into the computer's **mirror.list** file. That **generate_mirror_list_file ()** function utilizes a "here-document" form of redirection that tells the shell to read from the current script's source-text until the line containing **EOF** is seen – saving the redirected output directly to the **mirror.list** file. This

effectively duplicates the textual content of that segment of the function between its "**cat < < EOF...**" statement and its final **EOF** line copying that into a fresh version of the mirror.list file on the user's computer. Here is a screen shot of a portion of the **generate_mirror_list_file ()** function that determines the content of the **xenial** repositories:



Variables within the "here document" text (only **$1** and **$ARCHIVESECURITY** are seen above) are expanded into their current text values before the **mirror.list** file is written out to disk.

Since the Wasta-Offline full mirror supports both 32-bit (i386) and 64-bit (amd64) architectures, we have pairs of lines for most repository entries – one line that begins with **deb-amd64**… and a second

line that begins with **deb-i386**… as you can see in the screen shot above. The **wasta** and **wasta-apps** repositories also include a third line **deb-src** to include the source code in the repositories.

All repository lines have a variable **$1** prefixed to each repository location. For example, shown below are two lines excerpted from the middle of that "here-document" shown above – with the variable **$1** prefixes highlighted in red:

```
deb-amd64 $1packages.sil.org/ubuntu xenial-experimental main
deb-i386 $1packages.sil.org/ubuntu xenial-experimental main
```

When the bash function **generate_mirror_list_file ()** is called by a script to create the **mirror.list** file, the **$1** variable prefixed to each repository will be substituted with a string of characters that was used as a parameter when the **generate_mirror_list_file ()** function was called from the ***update-mirror.sh*** script. That parameter normally is either **`http://linuxrepo.sil.org.pg/mirror/, or http://`**, depending on which menu item (1 or 2 in the MAIN MENU) the administrator selects when running the ***update-mirror.sh*** script – thus the **$1** variable points to either the local Ukarumpa server, or the Internet.

After the script generates the mirror.list file, the two lines shown above will actually look like this in the computer's mirror.list file – when getting updates from the Ukarumpa local server (MAIN MENU selection 1):

```
deb-amd64 http://linuxrepo.sil.org.pg/mirror/packages.sil.org/ubuntu xenial-experimental main
deb-amd64 http://linuxrepo.sil.org.pg/mirror/packages.sil.org/ubuntu xenial-experimental main
```

The two lines would look like this – when getting updates directly from the Internet (MAIN MENU selection 2):

```
deb-amd64 http://packages.sil.org/ubuntu xenial-experimental main
deb-amd64 http://packages.sil.org/ubuntu xenial-experimental main
```

If one wanted to remove this SIL **xenial-experimental** repository from the mirror, one could put a # (comment marking) character at the beginning of each of the above two lines. That effectively "comments out" those two lines from being read by the apt-mirror program, and the pair of lines in the script function would look like this:

```
#deb-amd64 $1download.virtualbox.org/virtualbox/debian xenial contrib
#deb-i386 $1download.virtualbox.org/virtualbox/debian xenial contrib
```

Including more repositories is straightforward once you know the URL location of the repository. You just add them to the "here-document" in the form illustrated above. In one special case, that of VirtualBox, I have commented out that repository in the "here document" screen shot earlier in this section. If one wished to include the VirtualBox repositories for the xenial distribution in the "full" Wasta-Offline Mirror, one could uncomment the following two lines:

```
#deb-amd64 $1download.virtualbox.org/virtualbox/debian xenial contrib
#deb-i386 $1download.virtualbox.org/virtualbox/debian xenial contrib
```

by removing the initial # comment characters from both lines, making them become "active":

```
deb-amd64 $1download.virtualbox.org/virtualbox/debian xenial contrib
deb-i386 $1download.virtualbox.org/virtualbox/debian xenial contrib
```

Changes to the "here document" in the **generate_mirror_list_file ()** will not be effective until the next time you call the ***update-mirror.sh*** script – since the ***update-mirror.sh*** script re-generates the mirror.list file each time the ***update-mirror.sh*** script is run.

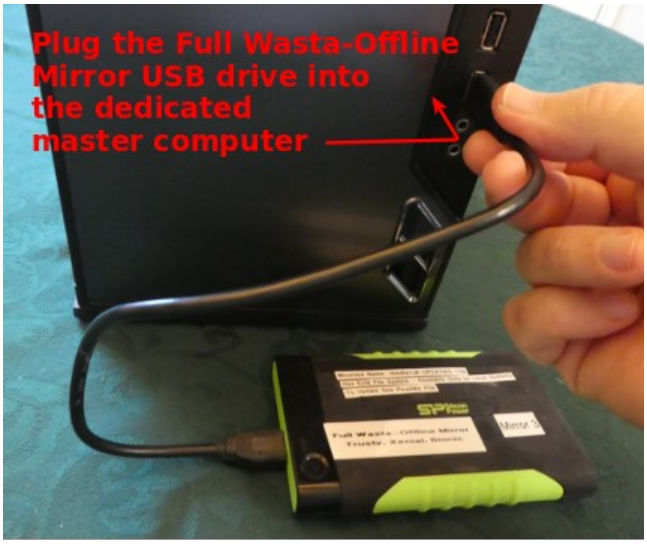
Remember: If you remove repositories by commenting them out of the resulting mirror.list configuration file, those repositories will be purged from the mirror itself the next time **apt-mirror** is run (directly or via running the ***update-mirror.sh*** script), and if you later want those repositories back in the mirror, they would have to be downloaded again (from the local server or the Internet).

How to Keep the Master Mirror Up-To-Date from the branch's Intranet (or the open Internet) - using the script: *update-mirror.sh*

This "How to" tells how to update the data on a dedicated Linux computer that hosts a copy of the Full Wasta-Offline Mirror. The initial creation of such a master mirror is described in a previous "How to" procedure, called "How to Make a Master Mirror by copying an Existing Mirror from an External USB Drive". As described in that section, the dedicated computer that hosts the master mirror should ideally be getting automatic updates automatically from the local server on a daily basis – assuming that a cron process was enabled on the dedicated computer hosting the master mirror (Enabling the cron job process is described in another "How to" section above, called "How to Configure the Master Mirror computer to Get Mirror Updates from the Ukarumpa Server (or Internet) on a Scheduled Daily or Periodic Basis"). This present "How to" section describes how to fetch updates to the master mirror on demand from the local server – whether or not a cron job process has been enabled. The ***update-mirror.sh*** script is provided for this purpose.

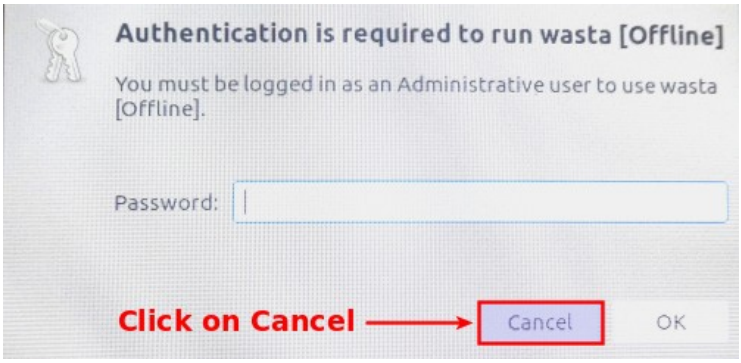
CAUTION: The process of updating a full mirror could involve downloading many gigabytes of data, so be sure you have access to a free local server that contains the data, or cheap access for downloading data from the Internet. Depending on the speed of your connection to the local server (or to the Internet), and how long since the full mirror was last updated, the update process could take just a few minutes, or it could take many hours (or even days). Hence, you should consider doing the updating process on a dedicated computer that can be left to run unattended for a long time - perhaps even overnight.

Even if you have no dedicated computer hosting a copy of the master mirror, the ***update-mirror.sh*** script described here can be run to update a connected USB drive that contains the full Wasta-Offline Mirror directly from the local server – without having created a master mirror on the computer the ***update-mirror.sh*** script is invoked on. Here are the steps for using this ***update-mirror.sh*** script:

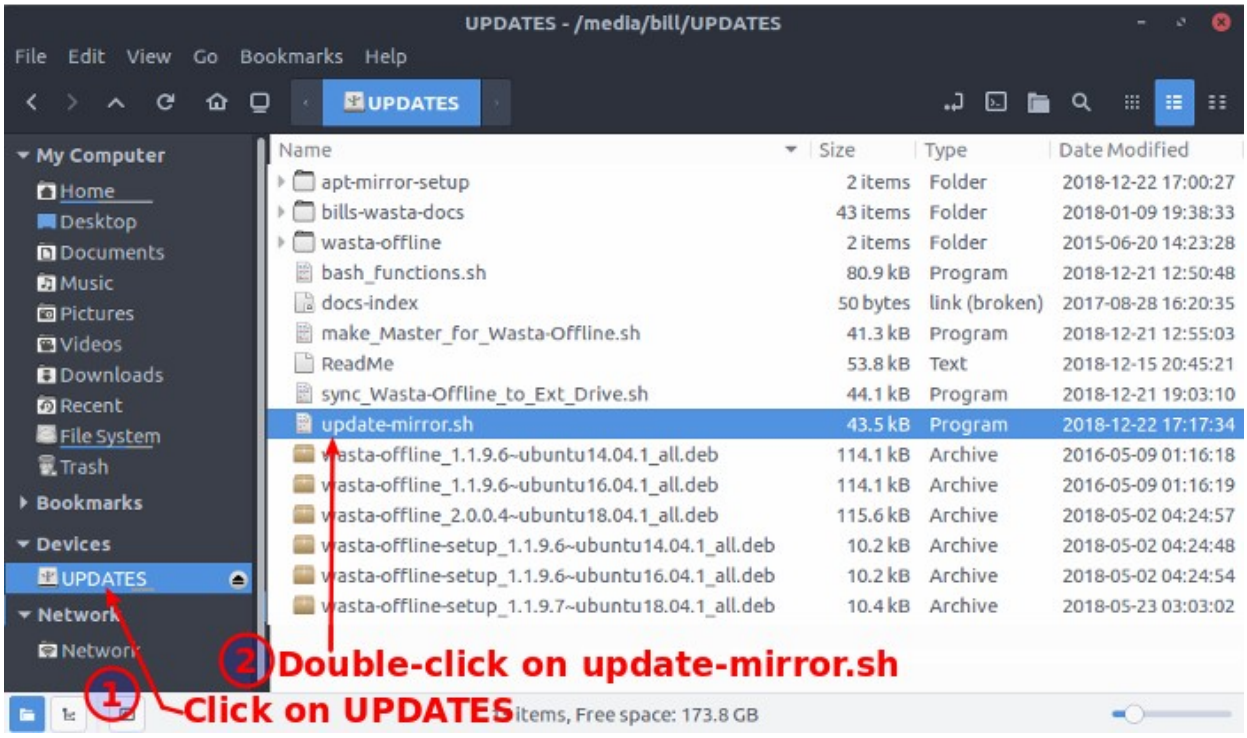


Step 1: Plug in the portable USB drive that contains the Full Wasta-Offline Mirror (from Bill Martin) into the USB port of the master computer – the computer that hosts the "master mirror". This computer should be running Wasta Linux and have a good connection to the server or the Internet.

Step 2: Cancel the Wasta-Offline password dialog. After about 10 seconds the wasta-offline program will start automatically (if the computer is running Wasta Linux), and will display a dialog asking for your password:

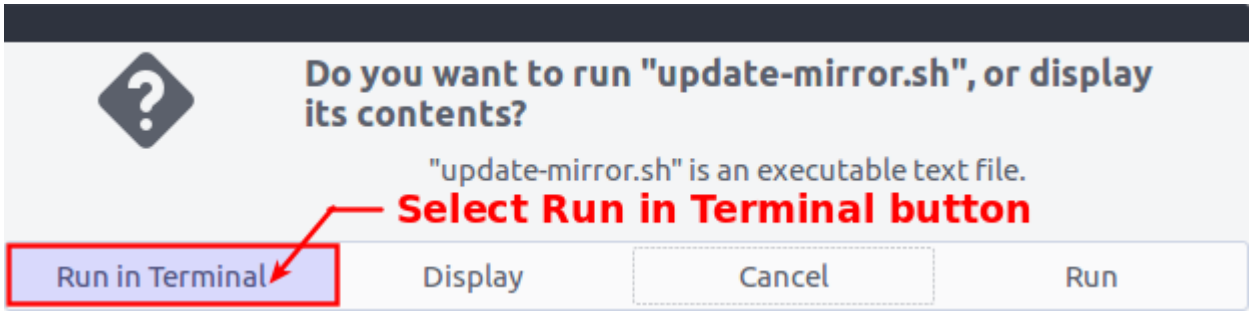


This time click on **Cancel** to close the "Authentication" dialog and stop the wasta-offline program from running. We don't need the wasta-offline program running while updating the master mirror on the master computer.

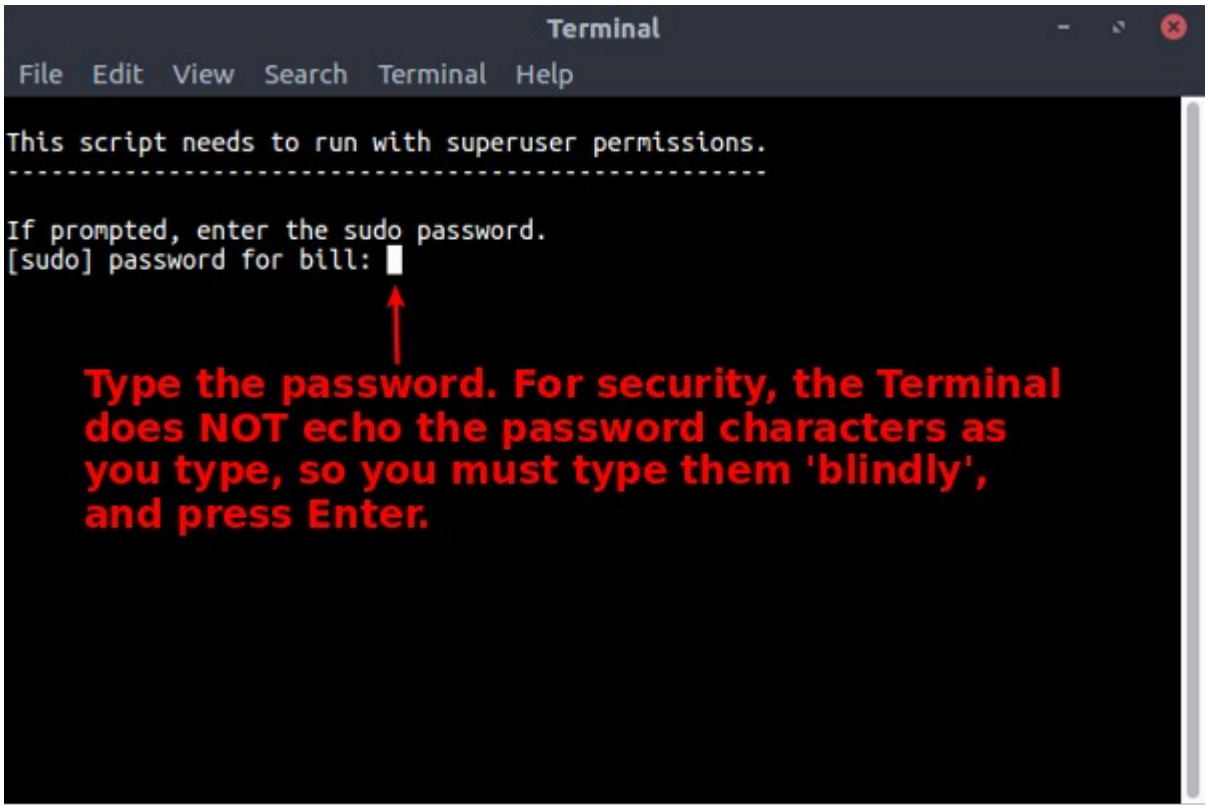


Step 3: Open the File Manager ("Files" on the panel) and locate the external USB hard drive that you just plugged in. It should be called something like **UPDATES** under the **Devices** list in the File Manager. Click on **UPDATES** to highlight the drive and display its contents:

Step 4: Run the *update-mirror.sh* script. On the UPDATES drive you should see a script there that is called: ***update-mirror.sh***. **Double-click** on the ***update-mirror.sh*** script (or highlight the ***update-mirror.sh*** file and press Enter). The File Manager will ask how the script should be run:



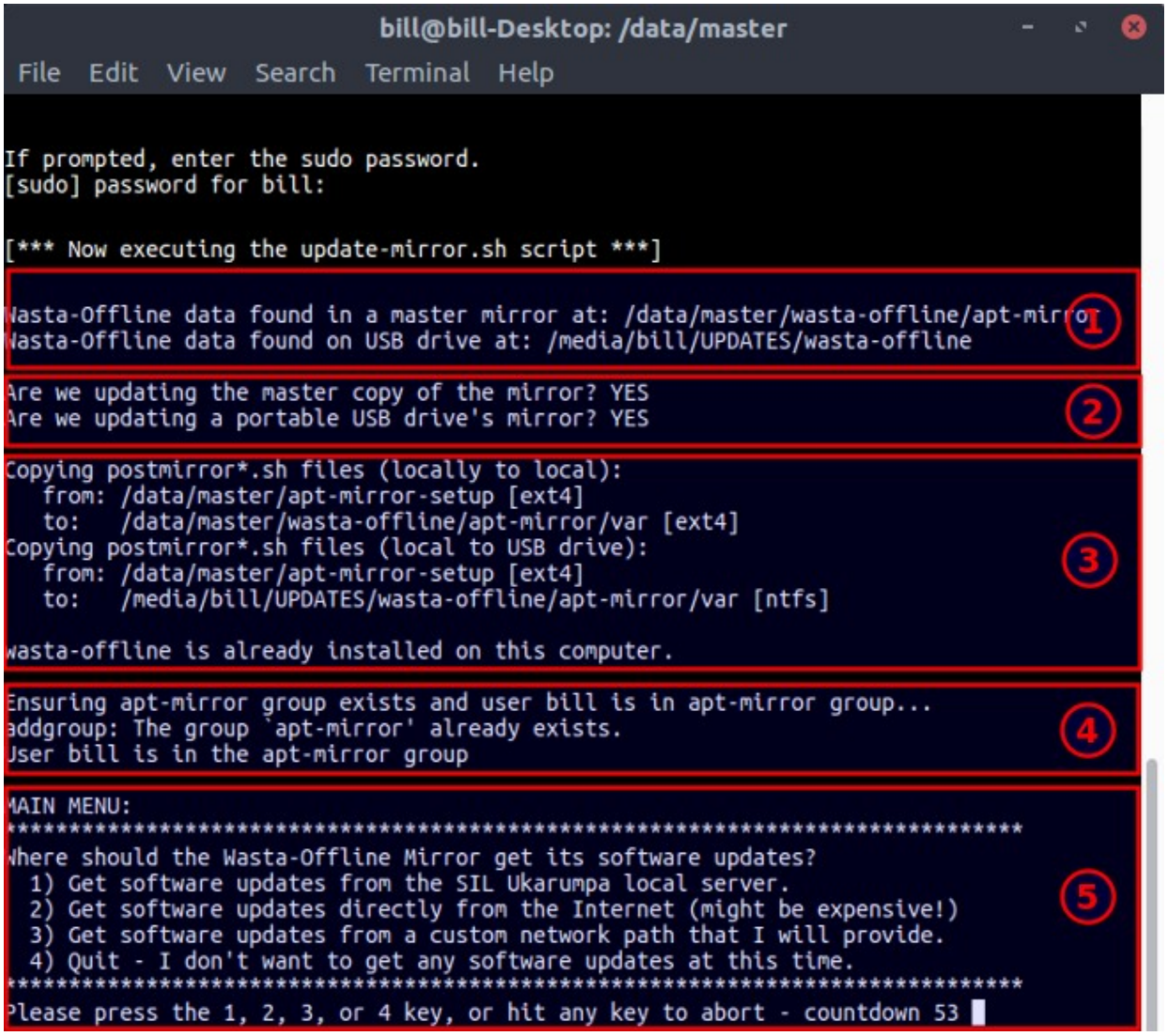
Select the "**Run in Terminal**" button.
The script should open a terminal window and start running:

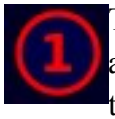


Note: If no terminal opens as illustrated below, but instead, an editor opens showing the contents of the script, see the Appendix section at end of this guide called "Setting File Manager Preferences to allow you to select *Run in Terminal* when double-clicking on the script".

Step 5: At the Terminal prompt, type the password for the user of the computer. As shown above, when you enter a password at the Terminal, the Terminal does not echo the password characters as you type, so you must **type the password 'blindly'**, and press **Enter**. The script should then continue.

Step 6: Understanding the script's console output. As the script runs, it makes a number of checks to ensure that a full Wasta-Offline mirror exists on an attached USB drive, and looks for a "master mirror" on the dedicated host computer. It ensures that the dedicated computer is configured to run the **apt-mirror** program that keeps the master mirror up-to-date. The script then presents a **MAIN MENU** with 4 numbered options, and waits for your response in a countdown prompt. The Terminal output up to this point – including the **MAIN MENU** looks like this:



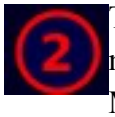


The script checks for the existence of a master mirror on the host computer, and also for any attached USB drive containing a full Wasta-Offline Mirror. If no existing mirror can be found the following warning and abort message will be given:

```
****** WARNING ******
"A USB drive with wasta-offline data was not found.
Cannot update Wasta-Offline Mirror.
****** WARNING ******
"Aborting...
```

Two supplemental scripts need to be present in the apt-mirror-setup directory. If either of these supplemental scripts cannot be found a warning and abort message will be given:

```
****** WARNING ******"
The postmirror.sh [or postmirror2.sh] file was not found. It should be at:
  <current-directory>/apt-mirror-setup/postmirror.sh [or postmirror2.sh]
  in the apt-mirror-setup subfolder of the <current-directory> directory.
Cannot continue update-mirror.sh processing! Please try again…
****** WARNING ******
Aborting...
```



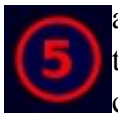
The ***update-mirror.sh*** script is designed to update the full Wasta-Offline Mirror residing as a master mirror on the computer on which the script is executing, or the full Wasta-Offline Mirror of an attached USB drive, or both. This output indicates which mirrors will be updated during the running of this script, indicating **YES** or **NO** in either case.

The script ensures that the two scripts ***postmirror.sh*** and ***postmirror2.sh*** scripts are copied from the apt-mirror-setup directory to a .../var directory where the apt-mirror program expects to find them, and that they are set to be executable. The script also check whether the **wasta-offline** program is installed (the normal case for Wasta Linux). If wasta-offline is not installed the script attempts to find an appropriate local **wasta-offline….deb** package and installs it. If wasta-offline is not installed and a deb package cannot be found the script continues, but the following warning is given:

```
Cannot install wasta-offline. A local deb package was not found.
You will need to install wasta-offline before you can use the mirror.
```

This check ensures that the computer has a group named **apt-mirror**, and that the user is a member of that apt-mirror group. If the current user could not be added to the apt-mirror group, the script will continue, but the following warning message will be given:

```
WARNING: Could not add user: <user-name> to the apt-mirror group
```

Step 7: The MAIN MENU. The ***updata-mirror.sh*** script now presents the **MAIN MENU** to the 
administrator. In order to update the master mirror and/or a mirror on an attached USB drive, the script needs to know where it can get the software updates from. The **MAIN MENU** has **4** choices as presented in this screen output, and prompts the administrator to select one of the 4 numbered choices. It is a timed countdown prompt giving the user 60 seconds to make a selection. After starting the ***update-mirror.sh*** script, the screen output proceeds quickly up to the **MAIN MENU** then pauses there waiting for you to respond.

For convenience below is a representation of the **MAIN MENU**.

```
MAIN MENU:

********************************************************************************
   Where should the Wasta-Offline Mirror get its software updates?
     1) Get software updates from the SIL Ukarumpa local server.
     2) Get software updates directly from the Internet (might be expensive!)
     3) Get software updates from a custom network path that I will provide.
     4) Quit - I don't want to get any software updates at this time.
********************************************************************************
   Please press the 1, 2, 3, or 4 key, or hit any key to abort - countdown 60
```

If you don't respond within 60 seconds the **4) Quit** option will automatically happen when the countdown reaches 0, and the script will stop without doing anything more. If it quits before you can read the options and decide, just press **Enter** to finish the script, and start running ***update-mirror.sh*** again (step 3 above), and in a matter of a few seconds you will be back at the **MAIN MENU** countdown prompt again.

Normally an administrator dwelling at a center like Ukarumpa would select option **1** since the local network already maintains a mirror of Linux software. In a location (such as within the USA) that has very low or no cost Internet access, an administrator would select option **2**. An administrator that has access to a different kind of local network that maintains a Linux mirror (such as an FTP network mirror) might select option **3** and provide the script with a path to that mirror. Option **4** quits the execution of the ***update-mirror.sh*** script – as does responding with any key press (other than 1, 2, or 3) at the countdown prompt.

If the administrator selects option **1** at the **MAIN MENU**, the script will try to connect directly with the Ukarumpa local server's mirror via the URL prefix of **`http://linuxrepo.sil.org.pg/mirror/`**
If no connection can be made with the server at that URL, this warning and an abort message is given:

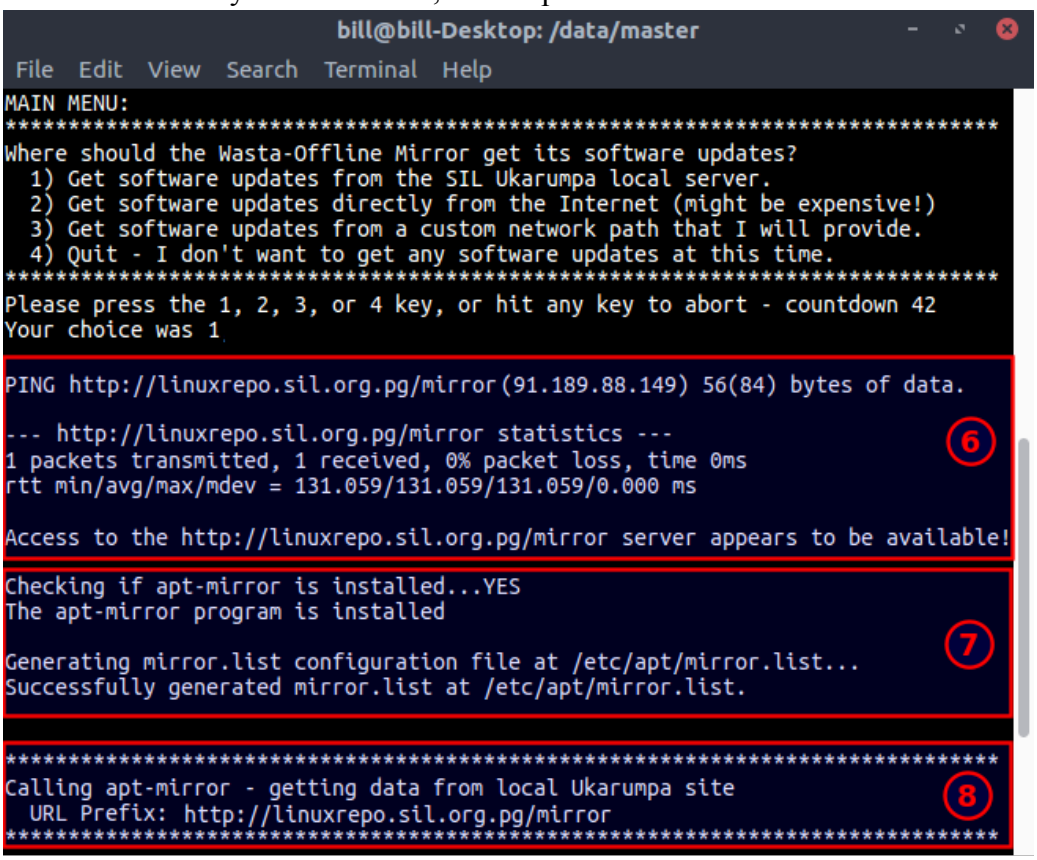
```
****** WARNING ******
Access to the http://linuxrepo.sil.org.pg/mirror server is not available.
This script cannot run without access to the SIL server.
Make sure the computer has access to the server, then try again.
****** WARNING ******
Aborting…
```

If the above warning and abort occurs, make sure the computer has access to the local server/intranet, and try running the ***update-mirror.sh*** script again from Step 4 above.

If the administrator selects **2** at the MAIN MENU, the script will try to connect directly with the actual (outside) Internet via the regular Internet URL prefix protocol of **`http://...`**
If no Internet connection is available to the **www.archive.ubuntu.com** site on the open Internet, this warning and an abort message is given:

```
****** WARNING ******
Internet access to www.archive.ubuntu.com not currently available.
This script cannot continue without access to the Internet.
Make sure the computer has access to the Internet, then try again.
Or, alternately, run wasta-offline and install software without Internet access
****** WARNING ******
Aborting…
```

If the above warning and abort occurs – and you really want to download possibly a lot of data - potentially gigabytes - from the Internet, make sure the computer has access to the Internet, and try running the ***update-mirror.sh*** script again. Once a viable **MAIN MENU** selection is made and the script can access the necessary Linux mirrors, the script continues to run as shown below:



This check is to determine if access to the server (or Internet if option 2 is selected) is available from the computer running the script. If server access is not available, the following warning and abort message appears:

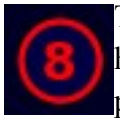
```
****** WARNING ******
"Access to the http://linuxrepo.sil.org.pg/mirror server is not available.
This script cannot run without access to the SIL server.
Make sure the computer has access to the server, then try again.
****** WARNING ******
Aborting…
```

or, for option 2 ("Get software updates directly from the Internet") this warning and abort message:

```
****** WARNING ******
"Internet access to www.archive.ubuntu.com not currently available.
This script cannot run without access to the SIL server.
Make sure the computer has access to the Internet, then try again.
****** WARNING ******
Aborting…
```

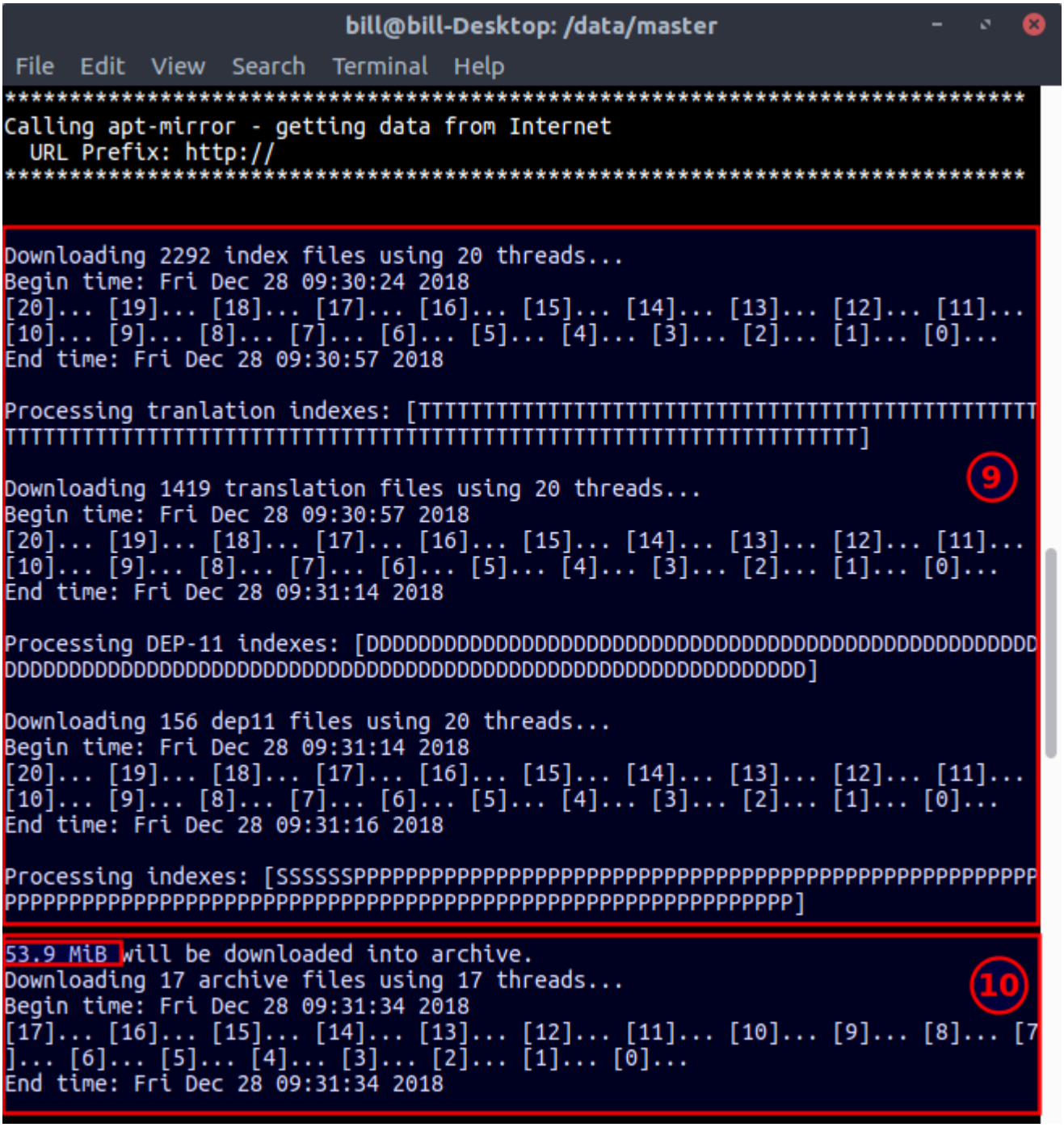
This check ensures that apt-mirror is installed, and there is a properly configured **mirror.list** file at **/etc/apt/mirror.list** that defines (for the **apt-mirror** program) the Linux repositories composing the full Wasta-Offline mirror. If apt-mirror is not installed, you will see output here related to the installation of the apt-mirror program. If apt-mirror cannot be installed or the mirror.list cannot be generated for some reason, the following warning(s) and abort message will be given:

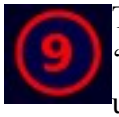
```
****** WARNING ******
Error: Could not install apt-mirror/generate mirror.list at /etc/apt/mirror.list.
****** WARNING ******
Aborting...
```



This output block announces that the **apt-mirror** program will now be executed. The output here also indicates the location where it will be downloading its data from, and the URL prefix/protocol being used. Executing the **apt-mirror** program is the main method we use to create and update the full Wasta-Offline Mirror. The repositories that are to be updated have been defined by the newly re-generated **mirror.list** file at **/etc/apt/mirror.list** (see check 7 above).

Most of the remaining console output from this point onward is produced by the apt-mirror program as shown in the screen shot below:





The **apt-mirror** program first downloads and processes a number of small files which it calls 'index files' 'translation files', and 'dep11 files' - using up to 20 process threads. This data is used to determine how much total data needs to be downloaded in order to update the mirror on the local machine to bring it up-to-date with what is in the remote mirror (on the server or Internet). Downloading these small files and the "Processing indexes" step usually takes 1-3 minutes.

This block of output announces the **amount of data and the number of files/packages** that apt-mirror needs to download in order to bring the local mirror up-to-date with the remote mirror. The important item to note is the **amount** that "will be downloaded into archive". In the sample above only a small amount of data – **53.9MiB** will be downloaded – because the local mirror is very nearly up to date with the remote mirror. **This figure may range from zero to tens of Gigabytes!** As apt-mirror proceeds with the download process it will successively show how many threads are left in the download process. The amount of time required to get to the **End time** line can vary from just a few minutes to an hour or more – depending on how up-to-date, or out-of-date the local mirror is compared to the remote mirror, and how fast the connection is to the server (or Internet).

Note: It would be possible to use this ***update-mirror.sh*** script to create a whole new local (master) mirror from scratch, but that would possibly put a burden on the local server for many hours or days of downloading time. That is why we have provided the ***make_Master_for_Wasta-Offline.sh*** script which is designed to be used together with an external USB drive containing a relatively up-to-date full Wasta-Offline Mirror (supplied by Bill Martin) – as a "kick start" for creating a master mirror. That "kick start" process is described and illustrated in the section called "How to Make a Master Mirror by copying an Existing Mirror from an External USB Drive" of this guide. That 'How to' is designed to create the master mirror relatively quickly – by syncing/copying it through the USB port, rather than downloading over 750GB of data over a potentially much slower connection with the local server.

When **apt-mirror** finishes downloading the data, it automatically runs the ***postmirror.sh*** script located in the local mirror's **`/wasta-offline/apt-mirror/var/`** directory. This script does some more checks, and automatically calls a ***clean.sh*** script (located in the same directory), and then prompts the user with this menu of choices:

```
Run the postmirror2.sh script to correct Hash Sum mismatches errors?
  1) No, don't run the script. There are no Hash Sum mismatches (default)
  2) Yes, run the script and get (75MB) of metadata from the Internet, or
  3) Yes, run the script and get (75MB) of metadata from the local FTP site
Please press the 1, 2, or 3 key, or hit any key to abort - countdown 60
```

The "countdown" will take 60 seconds to count down. If you don't respond within 60 seconds the **1) No, don't run the script...** selection will automatically happen and the script will **not** check for "Hash Sum mismatches". Not checking for "Hash Sum mismatches" – option **1** – should be the usual choice - unless you or others have experienced "Hash Sum mismatch" errors when trying to update the software on computers while using the wasta-offline program against this mirror.

After a default "No" response to the above prompt, some apt-mirror post-processing is done, and, if downloading was being done directly from the Internet, the script uses the **git** program (installs git if

necessary) to update any of the scripts from the external **wasta-scripts** repository, and also update the documentation from the **bills-wasta-docs** repository, producing the following screen output:

```
No selection made, or no response within 60 seconds. Assuming response of 1
The postmirror2.sh script was not called. Script completed.

Post Mirror script has completed. See above output for any possible errors.

Checking if git is installed...YES
The git program is installed
The local wasta-scripts repo .git file exists
Pull in any updates
Already up-to-date.
Create a .gitignore file for wasta-scripts
The bills-wasta-docs path is: /data/master/bills-wasta-docs
Already up-to-date.
Make /data/master/wasta-offline/apt-mirror owner be apt-mirror:apt-mirror
[*** End of apt-mirror post-processing ***]
```

After the ***postmirror.sh*** script (and optionally ***postmirror2.sh***) complete their cleanup work, the original ***update-mirror.sh*** script will keep running - and if an external USB drive containing a full Wasta-Offline Mirror is attached - the script will also automatically update the mirror on the external USB drive also, as a convenience to the administrator. To sync data to the USB drive the ***update-mirror.sh*** script will automatically call the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script which syncs the newly updated master mirror out to the external USB drive, updating its mirror too.

When the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script starts running you will see this announcement in the console output:

```
[*** Now executing the sync_Wasta-Offline_to_Ext_Drive.sh script ***]
```

NOTE: For detailed screen shots of the typical console output of the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script, see the last 'How to' section below called "How to Update the Mirrors on Portable USB drives – synchronizing them with the master mirror".

One of the first things the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script will do is to check time stamp of the mirror on the external USB drive and compare it with the time stamp of the local master mirror that was just updated by apt-mirror. Depending on the time stamps, you will be asked to confirm whether or not you want the mirror at the external USB drive destination to be updated. The information and prompt will look something like this:

```
    Comparing time stamps of the destination and source mirrors...
      Timestamp of mirror at destination is: 1415071069
      Timestamp of mirror at source is: 1415059606
**************************************************************************
    An OLDER copy of the wasta-offline mirror already exists at the destination!
```

```
    Replace it with the NEWER mirror from the source location? [y/n] (default='y')
*****************************************************************************
    Please press the y or n key, or hit any key to abort - countdown 60
```

Here again, the "countdown" will take 60 seconds to count down. If you don't respond within 60 seconds the "y" (yes) response will be assumed if the destination mirror is "OLDER", or if "there is no time stamp" on the destination mirror. In all other situations, a "n" (no) response will be assumed, and automatically selected if the countdown reaches zero.

All prompts requiring administrator response during their execution have prompts countdown times so that the administrator can allow the most common actions to take place unattended - that is, not checking for "Hash Sum mismatches", and the usual updating of a mirror on an attached external USB drive which most likely will be **OLDER** than the just-updated local master mirror.

The ***sync_Wasta-Offline_to_Ext_Drive.sh*** script (if called by the **update-mirror.sh** script) will produce a lot of output as it updates and syncs the files in the destination mirror on the USB drive. When the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script finishes, it returns execution control to its calling script ***update-mirror.sh***.

As with most scripts, if you need to interrupt the script for any reason, you can do so by typing **CTRL+C** within the terminal session window (you may need to do CTRL+C repeatedly to truly stop a busy script). If apt-mirror was still running when a CTRL+C interrupt signal is given, the apt-mirror program will stop. You can start up the ***update-mirror.sh*** script again at any time and apt-mirror will continue downloading the software changes from the point where it left off. It won't have to re-download any data that has already been downloaded. Hence, you can repeatedly run the ***update-mirror.sh*** script and interrupt it as many times as it takes to get all the available software updates - or choose an alternate menu response that was bypassed by a countdown prompt shown above. The same rule applies to the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script. It can be interrupted at any time via CTRL+C, and can be restarted by running the script by name, and it will continue syncing data from the point it left off.

When the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script is finishes the terminal will display this line:

```
    The sync_Wasta-Offline_to_Ext_Drive.sh script has finished.
```

When the ***update-mirror.sh*** script finishes (and all called scripts have finished) the terminal will display these two lines:

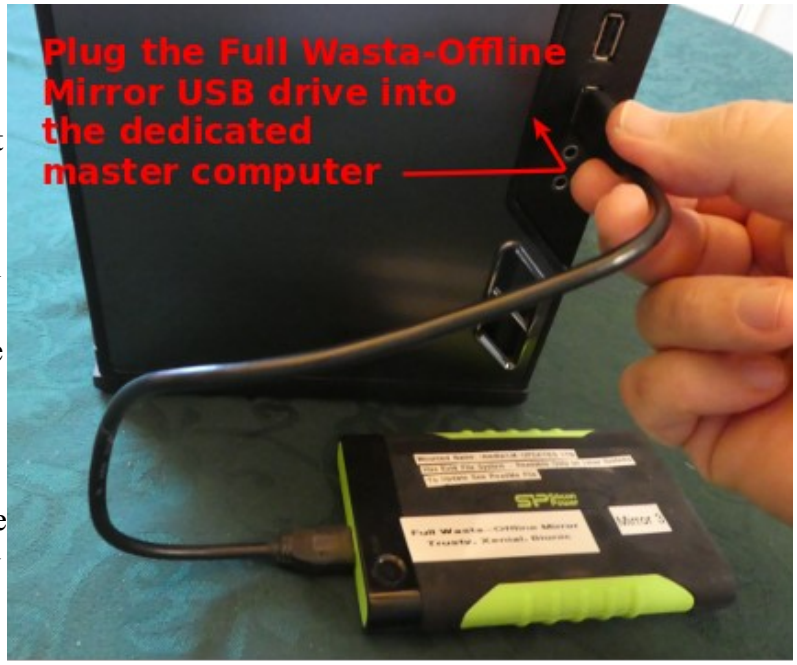
```
    The updata-mirror.sh script has finished.
    FINISHED:  Press <ENTER> to exit...
```

Press the **Enter** key to end the terminal session and close its window. You should respond to this last prompt by pressing **Enter**, otherwise the USB external hard drive may indicate it is "Busy" when you try to safely remove it.

How to Update the Mirrors on Portable USB drives – synchronizing them with the master mirror - using the script: sync_Wasta-Offline_to_Ext_Drive.sh

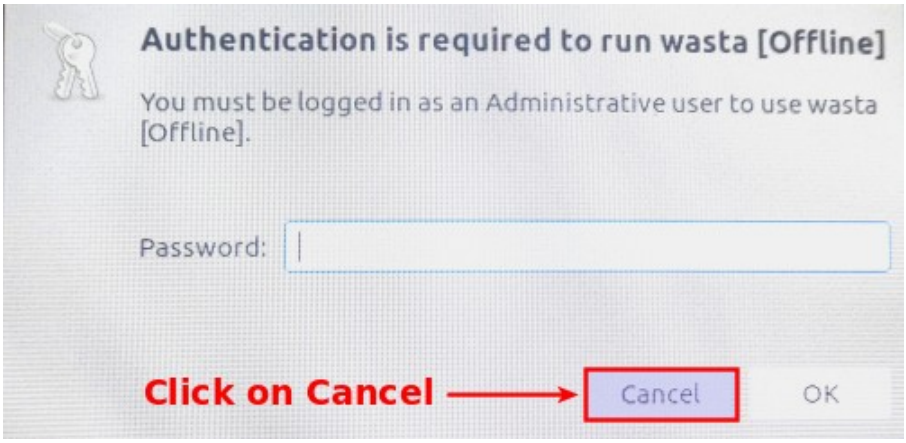
NOTE: The ***sync_Wasta-Offline_to_Ext_Drive.sh*** script is a versatile script that can be run as a stand-alone script to copy/sync a mirror from a source location to a destination location. The ***sync_Wasta-Offline_to_Ext_Drive.sh*** script is also automatically called by the ***make_Master_for_Wasta-Offline.sh*** script to create (kick start) a master mirror from an external USB drive, and may (optionally) be called by the ***update-mirror.sh*** script to copy/sync data from the just updated master mirror to an external USB drive. The ***sync_Wasta-Offline_to_Ext_Drive.sh*** script is the script that you will most frequently call to update the full Wasta-Offline mirrors of an arsenal of USB drives that are being circulated to and from the remote regions.

This "How to" tells how to synchronize a slightly out-of-date Wasta-Offline Mirror that resides on an external USB drive and bring it up-to-date against an up-to-date **master mirror** that resides on a local computer:



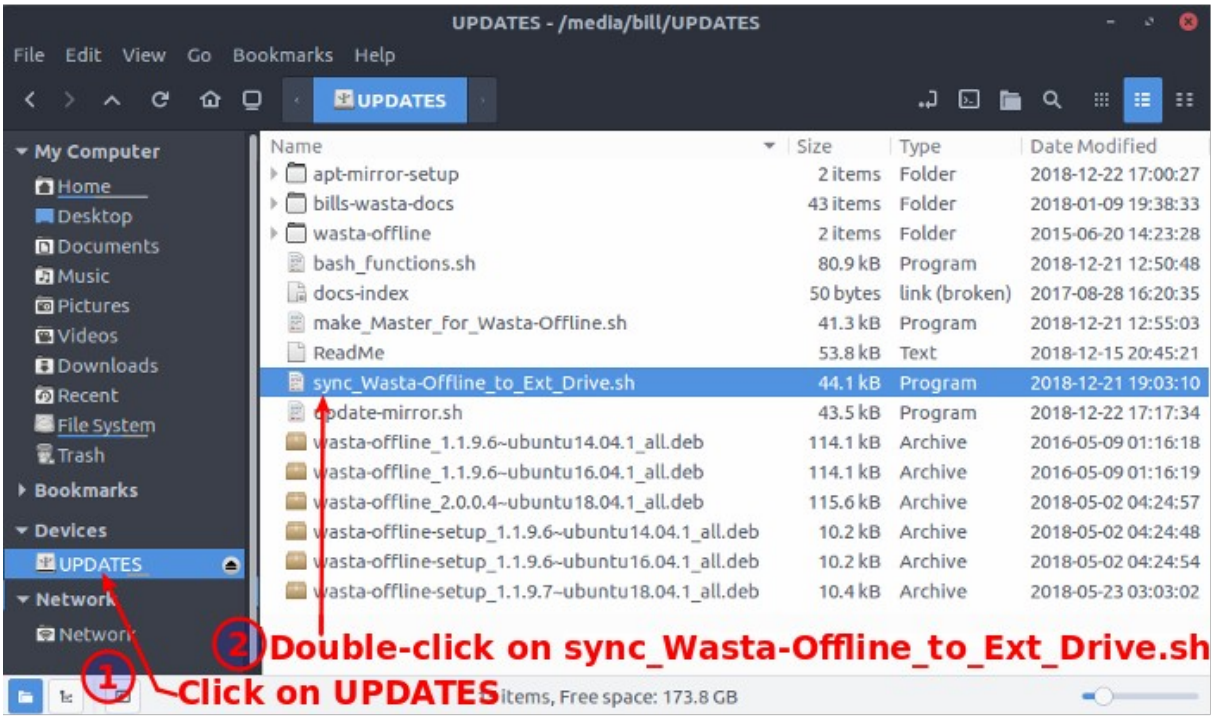
Step 1: Plug in the portable USB drive that has (or will contain) the Full Wasta-Offline Mirror into the USB port of the master computer – the computer that hosts the "master mirror". This computer should be running Wasta Linux and have a good connection to the server or the Internet. If you do not already have a **master mirror**, see the How to called "How to Make a Master Mirror by copying an Existing Mirror from an External USB Drive". If you already have a master mirror, and want to create a **new** full mirror (or update any out-of-date mirror) on a USB drive, you can plug in the USB drive and continue with the following steps.

Step 2: Cancel the Wasta-Offline password dialog. If the USB drive already contains a wasta-offline mirror, after about 10 seconds the wasta-offline program will start automatically (if the computer is running Wasta Linux), and will display a dialog asking for your password.



This time click on **Cancel** to close the "Authentication" dialog and stop the wasta-offline program from running. We don't need the wasta-offline program running while we sync the USB drive's mirror with the master mirror on the master computer.

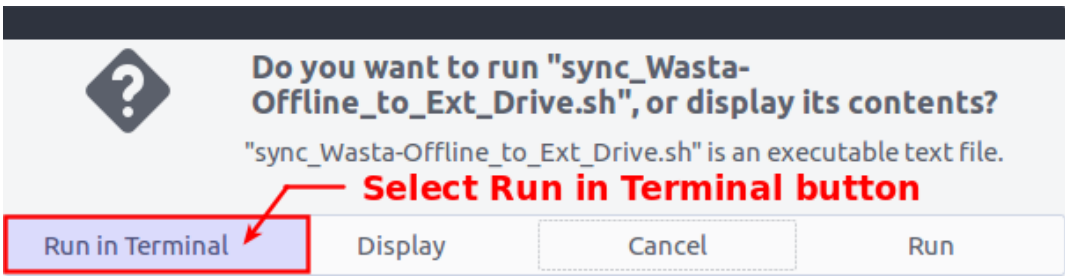
Step 3: Open the File Manager ("Files" on the panel) and locate the external USB hard drive that you just plugged in. It should be called something like **UPDATES** under the **Devices** list in the File Manager. Click on **UPDATES** to highlight the drive and display its contents in the right-hand pane of the File Manager (or locate your **master mirror** and highlight that folder to see similar contents):



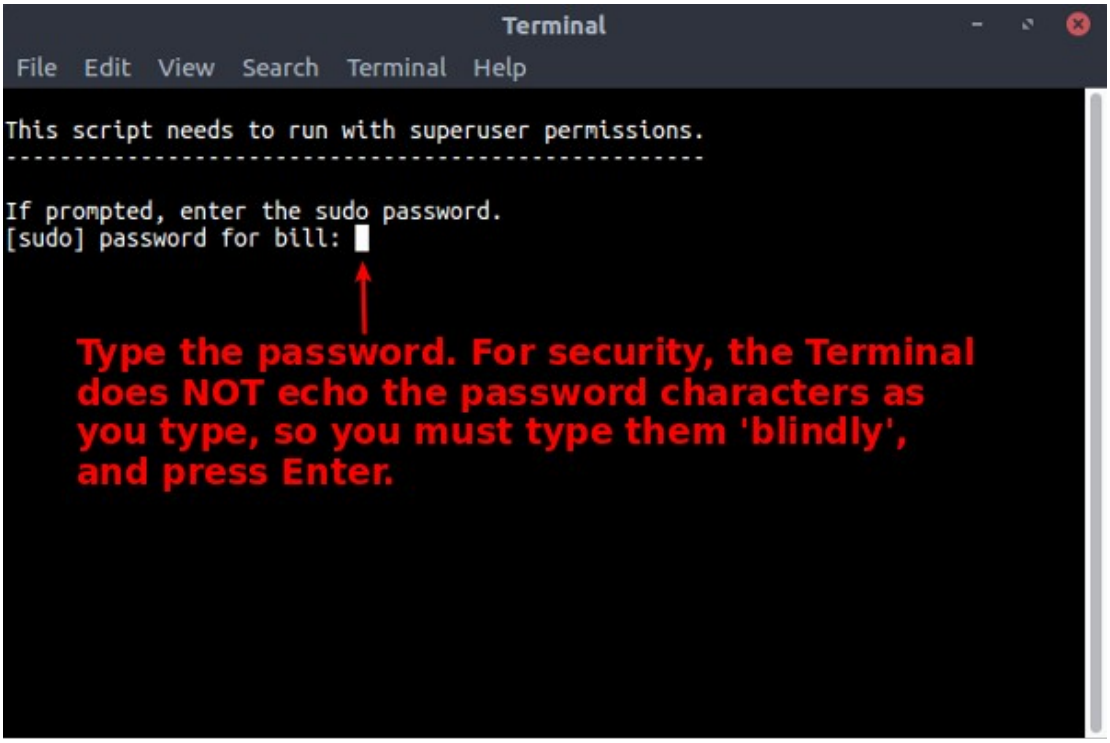
Step 4: Run the *sync_Wasta-Offline_to_Ext_Drive.sh* script. On the UPDATES drive (or in your master mirror folder) you should see a script there that is called: ***sync_Wasta-Offline_to_Ext_Drive.sh***.

Double-click on the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script (or highlight the ***sync_Wasta-Offline_to_Ext_Drive.sh*** file and press **Enter**). The File Manager will ask how the script should be run:

Select the "**Run in Terminal**" button.
The script should open a terminal window and start running.

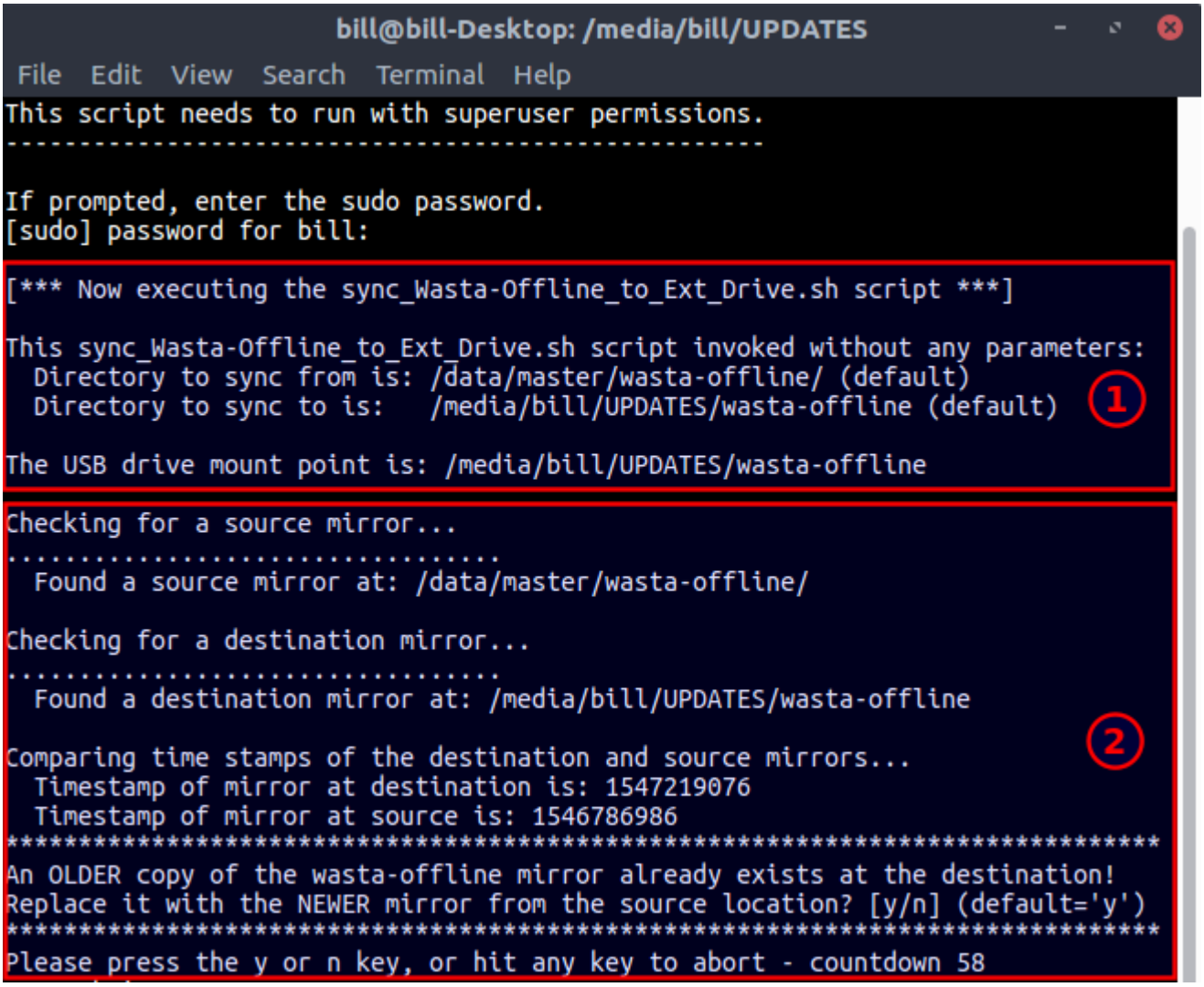


Note: If no terminal opens as illustrated below, but instead, an editor opens showing the contents of the script, see the Appendix section at end of this guide called "Setting File Manager Preferences to allow you to select *Run in Terminal* when double-clicking on the script".



Step 5: At the Terminal prompt, type the password for the user of the computer. As shown above, when you enter a password at the Terminal, **the Terminal does not echo the password characters as you type, so you must type the password 'blindly', and press Enter**. The script should then continue running.

Step 6: Understanding the script's console output. As the script runs, it makes a number of checks to ensure that a full Wasta-Offline mirror exists on an attached USB drive, and that a **master mirror** exists on the dedicated computer that we can copy/sync from. It then compares the time stamp of the master mirror with the time stamp of the mirror on the external USB drive. Ordinarily you will be updating the older data on the external USB drive with newer data in the master mirror, so the time stamp on the external USB drive will be **OLDER** than the time stamp of the master mirror on the dedicated computer, and the following console output will be displayed:



Below is an explanation of the two numbered blocks shown in the screen shot of script output above:

1 After announcing that it is now executing, the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script takes note that no parameters were passed into the script, and determines the default path to sync mirror data **from** (the 'source'), and the default path to sync mirror data **to** (the 'destination'). It also locates the mount point of the external USB drive. In the normal case the

'Directory to sync from' is a master mirror at: **`/data/master/wasta-offline/`** (default), and the 'Directory to sync to' is a USB drive at: **`/media/<User-Name>/<DISK_LABEL>/wasta-offline`** (default) as illustrated in the console output above.

If the script could not find a mount point for a **USB drive** containing a warning message and abort notification will be displayed as follows:

```
Wasta-Offline data was NOT found at /media/…
****** WARNING ******"
No USB drive found to update or create a wasta-offline mirror."
Please plug in a suitable USB drive and try running this script again."
****** WARNING ******"
Aborting..."
```

The script next checks to ensure that the designated 'source' data really has a **full** Wasta-Offline mirror that can be used to sync from. If a full Wasta-Offline mirror is not found at the 'source' a warning and an abort message is given:

```
****** WARNING ******
Could not find a source mirror at: /data/master/wasta-offline
Therefore, cannot update the USB mirror from this computer.
****** WARNING ******
Aborting...
```

The script then examines the location designated as the 'destination' for the mirror data, first checking whether an existing Wasta-Offline mirror is at the 'destination' or not. If no existing mirror is found (might be the case if a new USB drive is being prepared with a full Wasta-Offline Mirror), the script creates an initial (empty) directory tree for receiving the full mirror data. If an empty tree directory cannot be created for some reason a warning and an abort message is given:

```
****** WARNING ******
Cannot create mirror directories at /media/<User-Name>/<DISK_LABEL>/wasta-offline -
is the Drive writeable?
You might try rebooting the computer and running this script again.
****** WARNING ******
Aborting...
```

If there is a USB plugged in that does not already have Wasta-Offline Mirror data on it, it may be that you wish to create a Wasta-Offline Mirror on a **new USB drive**. The script will detect all attached USB drives that might be candidates, and present you with some information about the drive(s) and ask you to verify whether you want to create a new mirror on a USB drive, by selecting a USB drive from a numbered list. Normally you would only have one USB drive attached, so the "list" is likely to contain only one candidate USB drive, as shown by the **1)** line below:

```
Found potential USB Drive(s) for creating a new wasta-offline mirror…
Select a USB partition NUMBER from this list to create a new mirror:
NUMBER PARTITION TYPE MOUNTPOINT
  1)   /dev/sde1 ntfs /media/<User-Name>/<DISK_LABEL>
Type the NUMBER of the USB drive to use, or hit any key to abort - countdown 60
```

If you want the script to create a new Wasta-Offline Mirror on an attached USB drive shown above, just type the **number** shown under the **NUMBER** column of the above prompt – to the left of the USB drive description. For example, if you want to create a new copy of the mirror on the drive listed as **`1) /dev/sde1 ntfs /media/`** above, simply type the **`1`** key. The script will respond with:

```
Your choice was 1
Creating initial wasta-offline tree at: /media/<User-Name>/<DISK_LABEL>/wasta-offline
```

and it will continue and create a new full Wasta-Offline Mirror on the selected USB drive.

Note: If you have attached more than one USB drive (none having a wasta-offline mirror), then there would be more lines to select from in the listing of potential USB drives presented above – 1)..., 2)..., 3)... etc.

If you fail to respond to the countdown prompt within 60 seconds, or type a number or other key that is not a listed number, the script will abort. If that happens simply ensure that a USB drive is plugged in that you want to sync to, and run the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script again.

Step 7: Verify that you want to sync the master mirror to the USB drive. The script then compares the time stamp values of the destination and source mirrors. As mentioned above, you ordinarily will be updating the older data on the external USB drive with newer data in the master mirror, so the time stamp on the external USB drive will be **OLDER** than the time stamp of the master mirror on the dedicated computer, and you would see the prompt as shown in the screen shot of the script's console output above, and repeated here below:

```
********************************************************************************
An OLDER copy of the wasta-offline mirror already exists at the destination!
Replace it with the NEWER mirror from the source location? [y/n] (default='y')
********************************************************************************
Please press the y or n key, or hit any key to abort - countdown 60
```

This is the usual case. The assumed default response would be '**y**' for **YES**, and the script would continue with the sync process – even if you failed to respond to the countdown prompt within 60 seconds.

Similarly, if there is **no time stamp** on the destination USB drive, then it is assumed that a new mirror is being created on the destination USB drive, and the following prompt would be displayed (instead of the **`An OLDER...`** prompt above):

```
********************************************************************************
No Time stamp file found at destination, assuming the mirror there is older.
Replace it with the mirror from the source location? [y/n] (default is 'y')
********************************************************************************
Please press the y or n key, or hit any key to abort - countdown 60
```

In this case the default response would also be '**y**' for **YES**, and the script would continue with the copy/sync process – even if you failed to respond to the countdown prompt within 60 seconds. REMEMBER copying/syncing to an empty drive and creating a new mirror will take a long time to complete, since you would be copying over 750GB of mirror data!

Now, it is possible that the destination mirror could be the **SAME** or **NEWER** than the master mirror being copied from. In such cases the countdown prompt would look like one of these:

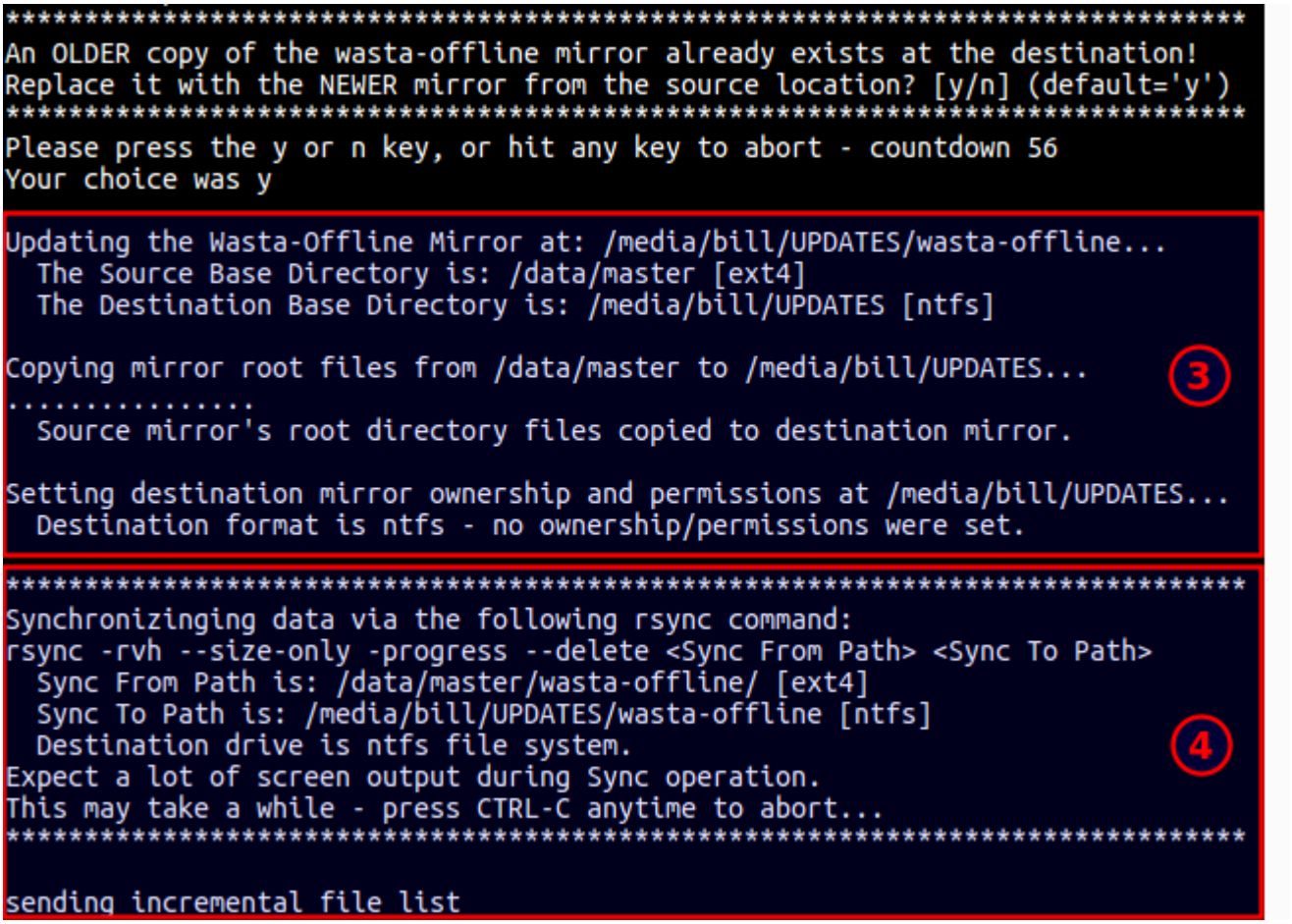
```
********************************************************************************
The SAME copy of the wasta-offline mirror already exists at the destination!
Update/Sync it anyway from the source location? [y/n] (default='n')
********************************************************************************
Please press the y or n key, or hit any key to abort - countdown 60
```

```
********************************************************************************
A NEWER copy of the wasta-offline mirror already exists at the destination!
Replace it with the OLDER mirror from the source location? [y/n] (default='n')
********************************************************************************
Please press the y or n key, or hit any key to abort - countdown 60
```

Notice that the default response for either of the above cases is '**n**' for **NO**. Updating or syncing the SAME mirror over itself would not make much sense when both source and destination mirrors have the same time stamp. Doing so might only be recommended if you think some mirror data on the destination drive was corrupted. The process of syncing the SAME data would not take very long since the rsync command is smart and will only change files that have actually changed.

Replacing NEWER data with OLDER data also would not make much sense,. You might only want to do that in case you wanted to roll back the data to an an earlier state for some reason. But, such rollbacks are not likely to ever be needed, so the default response is '**n**' for **NO** in such cases.

Step 8: Understand the script's console output.
Having countdown prompts with default responses as explained above, allow the administrator to allow the scripts to do their work mostly unattended once the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script has been started. If the explicit or default response to the prompt is **y** for **YES**, the script now continues to run. Here is a sample of what the continuing console output would look like:



Here (below) is an explanation of block numbers 3 and 4 shown in the screen shot above of the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script's output:

The script announces that it is "Updating the wasta-offline Mirror...". Before copying actual mirror data, the script copies certain files from the root/base directory of the source location to the root/base directory of the destination location, and sets the destination mirror's ownership and group permissions if appropriate. If the file system of the destination drive is not formatted as a Linux file system (i.e., it is vfat or ntfs as in the illustration above), no ownership/permissions will be set. On Linux file systems the ownership and group permissions of mirror files are set to **apt-mirror:apt-mirror**, in order for the apt-mirror cron process to be able to update the mirror files. When

setting ownership and permissions on a USB destination (with a Linux file system format) the process may take a few minutes without visible activity on the console. The setting of ownership and permissions is mainly for a master mirror residing on a dedicated Linux computer, but will also be done for external USB drives that are formatted with a Linux file system.

The script then outputs this block of lines, indicating how it will call the Linux **rsync** command and the rsync options it will be using while rsync does the actual copying/syncing of files from the source mirror to the mirror at the destination. The rsync command and options used in the call illustrated above are:

```
********************************************************************************
Synchronizinging data via the following rsync command:
rsync -rvh --size-only -progress --delete <Sync From Path> <Sync To Path>
  Sync From Path is: /data/master/wasta-offline/ [ext4]
  Sync To Path is: /media/bill/UPDATES/wasta-offline [ntfs]
  Destination drive is ntfs file system.
Expect a lot of screen output during Sync operation.
This may take a while - press CTRL-C anytime to abort...
********************************************************************************
```

The type of file system of the destination drive (Sync To Path) determines the rsync options that are used. In the illustration above the destination's file system is **ntfs**.
For ntfs or vfat, the rsync options used are (as shown above):

```
-rvh        [recursive (-r), verbose (-v), human-readable (-h)]
--size-only [skip files that match in size]
--progress  [show progress during transfer]
--delete    [delete extraneous files from dest dirs – removes outdated files]
```

For a Linux (ext2, ext3 or ext4) file system, the rsync options used would be the following:

```
-avh        [archive mode/recursive (-a), verbose (-v), human-readable (-h)]
--progress  [show progress during transfer]
--delete    [delete extraneous files from dest dirs – removes outdated files]
```

Note: rsync option **-a** by itself is equivalent to these 7 rsync options: **-rlptgoD** (see rsync's man pages if you are interested). Syncing files to non-Linux formatted drives – especially FAT32 (vfat) formatted drives – can be problematic and requires that different rsync options be used as indicated above.

NOTE: The **--delete** option used with the rsync command means that rsync will delete any obsolete files and packages that no longer exist in the source (master) mirror location. Recall that the apt-mirror program will do the same when it runs to update the master mirror. Using this **--delete** option with the rsync command will ensure that the destination mirror accurately reflects the exact content of the source (master) mirror each time the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script is run.

The rsync command then begins its copying and syncing process with the output line:

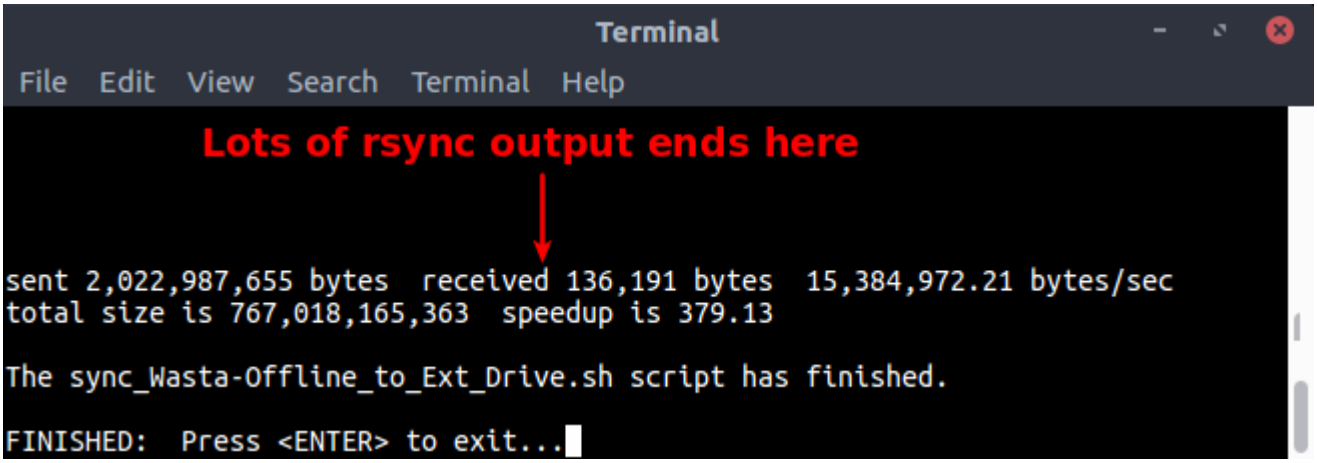
```
sending incremental file list
```

At this point there will be a lot of console output as rsync copies and/or syncs the data from the source mirror to the destination mirror. The amount of output depends, of course, on how much the source and destination mirrors differ from each other. If the destination mirror is being created from scratch, or has not been updated in a long time there will be a lot of output in the console – possibly hours and hours, especially if a **new** USB drive is being initialized with over 750GB of new mirror data!

If rsync encountered an error or could not complete its copying process, you will see the following warning and abort message before the "FINISHED" message in the console output:

```
****** WARNING ******
Could not rsync the mirror data to /media/<User-Name>/<DISK_LABEL>/wasta-offline
****** WARNING ******
Aborting...
```

Eventually when rsync completes its work, the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script finishes with a console message similar to the following:



Step 9: Press the Enter key to end the terminal session and close its window. This should be done, otherwise the USB external hard drive will indicate it is "Busy" when you try to safely remove it.

The external USB drive's mirror should now be up-to-date with an exact duplicate of the master mirror, and can be sent out to a remote region where team advisors can use it to update team members' Wasta Linux computers.

APPENDIX

Setting File Manager Preferences to allow you to select *Run in Terminal* when double-clicking on the script

Note: If the script opens in the gedit editor instead of asking you if you want to **Run in Terminal**, you can try making the following adjustment in the File Manager's settings:

- Quit the gedit editor, and start the **File Manager** (Nemo) by clicking on the **Files** icon on the panel.
- Select **Preferences** from Nemo's **Edit** menu.
- Select the **Behavior** tab at the top of the **Preferences dialog**.
- Under the **Executable Text Files** section, select the **Ask each time** button.
- Click on the **Close** button. Now when you double-click on an executable script file, Nemo should give you the option of **Run in Terminal**.

If, after following sub-steps above, Nemo still opens the script in the gedit editor instead of offering to "Run" it, it may be that the script is not set to have executable permissions.
To check and set **executable** permissions on the script, you can make the following adjustment to the script file's permissions:

- Start the (Nemo) **File Manager** (click the **Files** icon on the panel)
- Navigate to the device or folder containing the script that you want to check if it is **executable**.
- Right-click on the script's name in the list, and select **Properties**.
- In the **Properties** dialog select the **Permissions** tab at the top of the dialog.
- Make sure there is a tick in the check box to the right of **Execute** that says **Allow executing file as program**.
- Click on the **Close** button. Now when you double-click on that executable script file within File Manager, it should give you the option of **Run in Terminal**.

If, Nemo refuses to make the script executable, it is likely that the script is residing on a USB drive that is formatted with a FAT32 (vfat) file system. FAT32 doesn't support the direct execution of Linux scripts. If that is the case, and if you have created a master mirror (using the ***make_Master_for_Wasta-Offline.sh*** script), you can try running the File Manager and navigate to the master mirror's directory (by default it is `/data/master`) and try double-clicking on the script within the File Manager pointed to that location. If your external USB drive is a FAT32 format drive, and you have not created a master mirror, you can copy the script files in the root directory of the USB drive over to a suitable location on the Linux computer from which you are running these scripts. Then you can make the scripts executable by following the second set of points above.

Using git to keep the wasta-offline scripts up-to-date from GitHub

Bill Martin maintains all of the scripts described in this guide in a GitHub repository at:

https://github.com/pngbill-scripts/wasta-scripts

If you want to **clone** a copy of that GitHub repository to your Desktop you can open a Terminal (**CTRL+ALT+T**), and type this command (assuming you have Internet access):

```
git clone https://github.com/pngbill-scripts/wasta-scripts Desktop/wasta-scripts
```

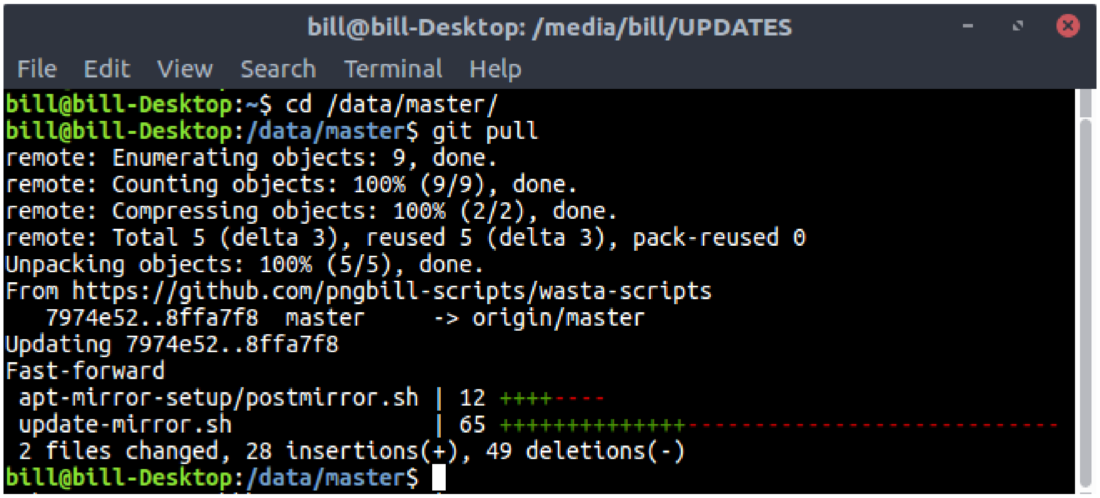
to create a folder called **wasta-scripts** on your Desktop and clone the scripts into that directory.

Please note that, as a convenience to administrators, all of the scripts are included in the root directory of any USB drives containing the full Wasta-Offline Mirror as supplied by Bill Martin. The scripts are also maintained in any master mirror that is created via the ***make_Master_for_Wasta-Offline.sh*** script, and thereafter are updated whenever the ***update-mirror.sh*** script or the ***sync_Wasta-Offline_to_Ext_Drive.sh*** script are executed. In other words, the scripts also help to ensure that the latest scripts get propagated along and are updated during their routine use.

You can ensure that you have the latest versions of Bill's scripts by simply opening a Terminal and changing the working directory to the root directory of an existing full Wasta-Offline Mirror, and executing the **git pull** command. For example (assuming you have a master mirror at **/data/master** on a dedicated computer), you can press **CTRL+ALT+T** to open a Terminal, and then type the two commands:

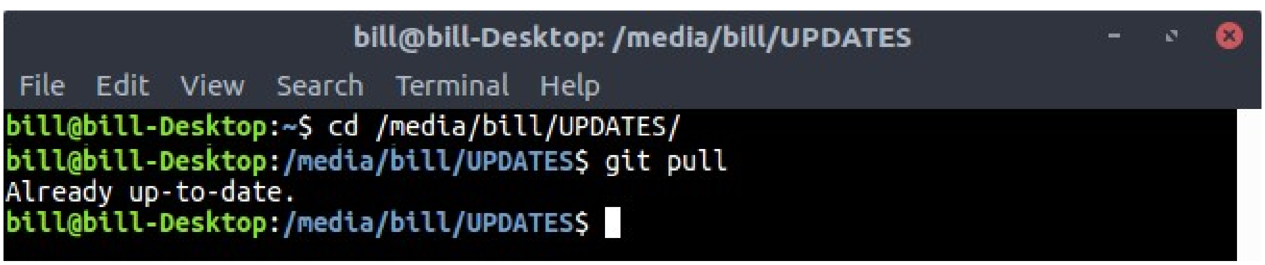
```
cd /data/master
git pull
```

as shown graphically below:



In the Terminal above we gave the **`cd /data/master`** command to change to a directory the master mirror that has the full Wasta-Offline Mirror, which includes a git repository of Bill's scripts. Then the **`git pull`** command was executed from that directory. There were two scripts that had changes and git merged them into the working copy in the **`/data/master`** folder.

If you call **`git pull`** and there were no changes to the scripts in the external repository, the git command will report **`Already up-to-date`**, as seen in the following screen shot:



If you run into problems with conflicts while using git, it is probably easiest to **clone** the **wasta-scripts** repo (see above) to some temporary folder, then copy its contents (including the normally hidden **`.git`** and **`.gitignore`** files) back to their location at your master mirror (**`/data/master`**) or the root directory of an external USB drive (**/media/<User-Name>/<DISK_LABEL>**).

Note: A git repository will have a hidden file named **.git** in the directory where the repository resides. The same directory may also have another hidden file named **.gitignore**. If you look for these files using the File Manager, you will need to select **`Show Hidden Files (Ctrl+H)`** from File Manager's **`View`** menu to see them.