

Md380tools-menu

Menu subset of the md380tools-vm Virtual Machine

Warren Merkel, KD4Z

Installation instructions to install the KD4Z md380tools-vm scripting framework on bare metal Linux.

This is not a fork of the md380tools-vm project, but merely a subset. You will be running exactly the same scripting framework, and will enjoy the automatic update features of the main VM type of installation.

If you would like to take advantage of the easy-to-use menu features of the md380tools-vm project, but would rather run the scripts natively in your preferred flavor of linux, these steps will allow you to do just that. If you are running MMDVM on a Raspberry Pi, you are almost ready to go!

Prerequisites:

You must have an existing installation of linux. You can use a stand-alone Linux machine. Or you could run Linux inside a Virtual Machine of your own choosing. Or you can use Raspian running on a Raspberry Pi.

Installing the md380tools-menu framework:

Start in a non-privileged user's home directory. (Don't miss the space "tilda" at the end)

```
cd ~
```

If you have already been using the md380tools, you might need to remove it first. Depending on how it was installed, md380tools needs to be installed as a non-privileged user. Some "pre-built" images for MMDVMHost and other md380tools VM images, install md380tools as "root". Since we will be running as a non-privileged user here, we need to just remove it using super-user elevation. When you run "glv", the md380tools and md380tools-vm folders will be completely removed. This needs to be able to occur as a non-privileged user.

So, from your current user's home directory. (*cd ~* to get there)

Run this: *ls md380tools*

If you see a bunch of files fly by in the listing, fine. We need to just remove them. This is the easiest way to clean house in case they are owned by the root user. They will be put back each time you run glv anyway.

Run this command to remove the existing folder, if it exists. (If your Linux distribution doesn't have sudo, be sure to "su -" to become root first)

```
sudo rm -rf md380tools
```

If you didn't see the md380tools folder had pre-existed in your image, you will need to perform installation of md380tools prerequisites as described here:

<https://github.com/travisgoodspeed/md380tools#preparation-of-build-environment>

Be sure to perform these steps as well:

<https://github.com/travisgoodspeed/md380tools#additional-steps-for-linux-based-installations>

Don't forget this last important step as listed on the main md380tools page.

```
cd md380tools
sudo cp 99-md380.rules /etc/udev/rules.d/
```

You will need to log back in after running the above line, and before attempting to use the flash commands. You can wait to do this until the end of the instructions however. If you forget to do this, you will get an error when attempting to access the USB device.

If you see weird errors pop up during the glv, you might need to go back to the above links and make sure you didn't skip a step!

Additional requirements for running md380tools-menu:

You will also need to install the zip and unzip programs. If you have apt-get installed, run this as root user or use sudo:

```
sudo apt-get install zip unzip
```

Otherwise, you will need to determine how to install the zip and unzip programs based on your particular linux distribution.

Pull down the md380tools-vm scripts from github:

If you su'd to root user, exit now, back to your non-privileged user. (run the exit command or Ctrl+D)

Make sure you are in the user's home directory again with:

```
cd ~
```

Then, run these next three commands:

```
git clone https://github.com/KD4Z/md380tools-vm.git
```

```
mv .bash_aliases .bash_aliases.original
```

(Ignore the file not found message if you didn't have the aliases file already. This command will make a backup of your original, just in case you had something in there you want to keep)

```
cp md380tools-vm/root/.bash_aliases ~
```

(don't miss that last space and "tilda" at the end of the command above!) Note, this will overwrite your existing `.bash_aliases` file. Now, we need to verify that the `.bash_aliases` script will get run from `.bashrc`

With your favorite text editor (vi or nano for example), open the `.bashrc` file. Scan down looking for an `if []` construct like this:

```
if [ -f ~/.bash_aliases ]; then
    ~/.bash_aliases
fi
```

If you can't find it, scroll all the way to the bottom, and add all three lines into the `.bashrc` file. Be sure to enter it exactly as listed above, including the spaces and periods! Save your changes, and exit the editor.

Fedora note:

If you are using Fedora, you might need to create a small bash script to redirect the SHA256 calculations to a different binary file than normal. Fedora seems to have removed the "shasum" program, and replaced it with multiple versions that calculate different size SHA hashes.

Run this to check to see if you have (or are missing) the shasum binary file.

```
ls /usr/bin/shasum
```

If you see a message "no such file or directory", then you need to follow these next simple steps to create a bash script replacement for the missing file. Otherwise, if you have the shasum binary file, skip to "Ready-Set-Go" section

You should also check to see if you have the sha256sum binary, as you will need it to be present!

```
ls /usr/bin/sha256sum
```

Hopefully, you will see a listing returned for this file.

As root user, use your favorite text editor, create a file named `/usr/bin/shasum`

You can use `sudo nano` or `sudo vi` for this. Add these two lines in the file and save it.

```
#!/usr/bin/bash
/usr/bin/sha256sum $3
```

You will need to set this new file to be executable with this:

```
sudo chmod +x /usr/bin/shasum
```

That should now allow the md380tools scripts to run on Fedora. That bash script becomes a replacement for the missing shasum binary, and just redirects it to the newer sha256sum program.

One more thing! The current instructions for installing md380tools on Fedora are missing something quite basic! It appears that Fedora 26 (the current version as of this document) does not have Python installed by default. You will need to install Python too!

```
sudo dnf install python
```

Ready-Set-Go

Logout of your terminal by running the exit command.

```
exit
```

Login again or open a terminal window again.

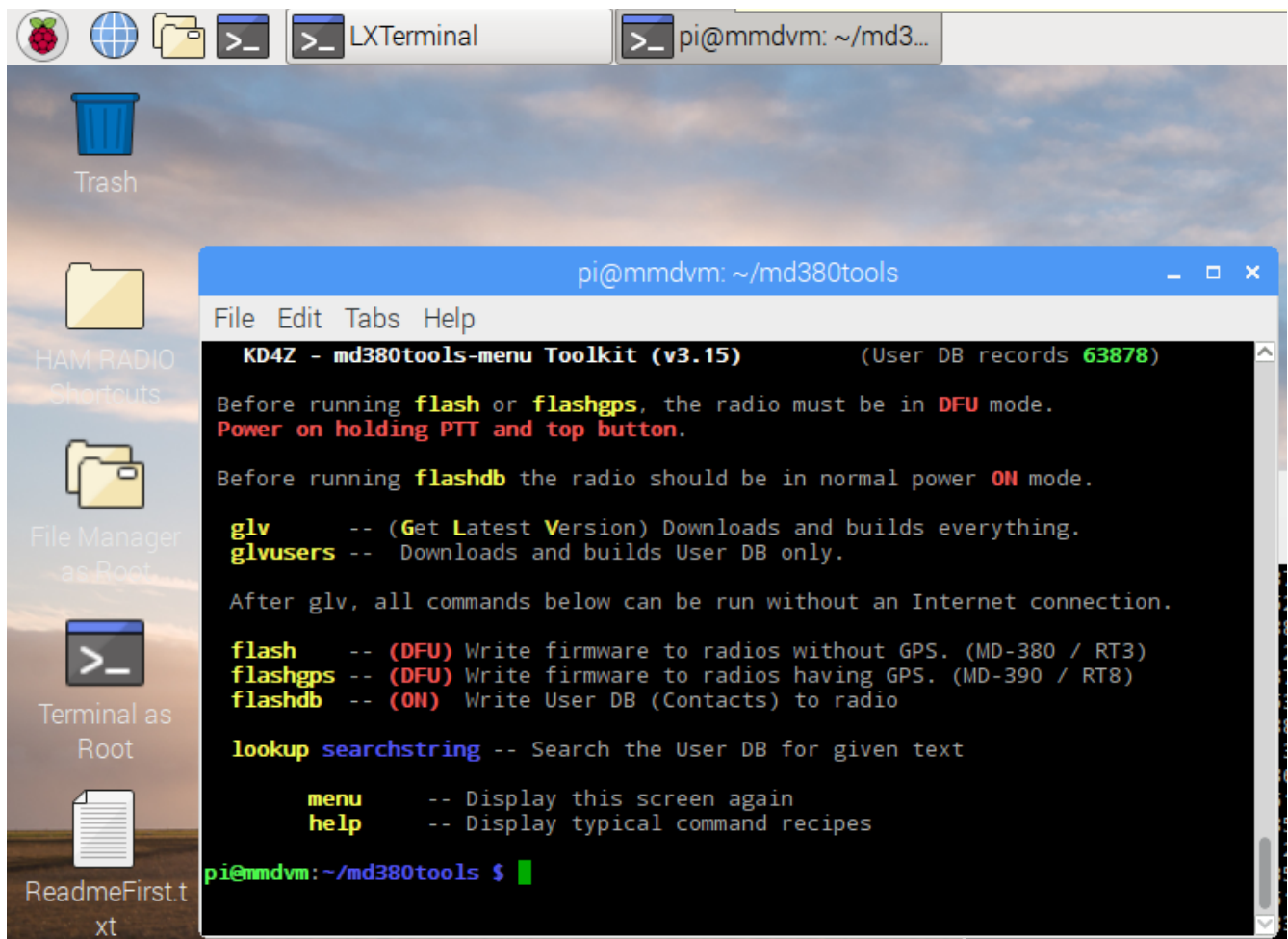
Run the main glv command in the KD4Z Toolkit:

```
glv
```

You see a lot of scripts running and source code flying by! When all of that is done, you will be left back at the linux prompt. Remember, to display the menu again, at anytime, run:

```
menu
```

You should see this screen.

The image shows a Raspberry Pi desktop environment. On the left sidebar, there are icons for 'Trash', 'HAM RADIO Shortcuts', 'File Manager as Root', 'Terminal as Root', and 'ReadmeFirst.txt'. The main desktop area has a blue sky background. A terminal window titled 'pi@mmdvm: ~/md380tools' is open, displaying the 'KD4Z - md380tools-menu Toolkit (v3.15)' menu. The menu text includes instructions for using 'flash', 'flashgps', 'flashdb', 'lookup', 'menu', and 'help' commands, and notes about DFU mode and internet connectivity. The prompt 'pi@mmdvm:~/md380tools \$' is visible at the bottom of the terminal window.

```
pi@mmdvm: ~/md380tools
File Edit Tabs Help
KD4Z - md380tools-menu Toolkit (v3.15) (User DB records 63878)
Before running flash or flashgps, the radio must be in DFU mode.
Power on holding PTT and top button.
Before running flashdb the radio should be in normal power ON mode.
glv -- (Get Latest Version) Downloads and builds everything.
glvusers -- Downloads and builds User DB only.
After glv, all commands below can be run without an Internet connection.
flash -- (DFU) Write firmware to radios without GPS. (MD-380 / RT3)
flashgps -- (DFU) Write firmware to radios having GPS. (MD-390 / RT8)
flashdb -- (ON) Write User DB (Contacts) to radio
lookup searchstring -- Search the User DB for given text
menu -- Display this screen again
help -- Display typical command recipes
pi@mmdvm:~/md380tools $
```

In this example, I installed the KD4Z Toolkit scripting in Russell's excellent KB5RAB MMDVMHost raspberry Pi image. He already had the stock md380tools installed, but I wanted to have full use of the KD4Z Toolkit as well.

The commands are exactly as described in the installation documentation for the KD4Z/md380tools-vm project found here:

<https://github.com/KD4Z/md380tools-vm>

If you want to have the menu automatically display when the terminal window is opened, you can add this line at the end of your .bashrc script using your favorite editor. Don't miss the dot at the beginning of the filename. If you have nano: (or use vi if you prefer)

```
nano .bashrc
```

Go all the way down to the end of the script, and add this line:

```
~/md380tools-vm/menuopts
```

Then save and exit with Control+X, press y, then press Enter.

Enjoy.

73 de KD4Z
Warren Merkel

revision 1.0	3 February 2017
revision 1.1	18 June 2017
revision 1.2	27 July 2017