

Creating and using a live installation image

 docs.fedoraproject.org/en-US/quick-docs/creating-and-using-a-live-installation-image/index.html

Quick Docs master

This page has been converted from the Fedora Project Wiki and cleaned up for publishing here on the Fedora Docs Portal, but it has not yet been reviewed for technical accuracy. This means any information on this page may be outdated or inaccurate. Reviews for technical accuracy are greatly appreciated. If you want to help, see the [README](#) file in the source repository for instructions.

Downloading Fedora

You can download Fedora from <https://getfedora.org/>.

There are multiple desktops available for use with Fedora. Each has a slightly different look and feel and offers varying levels of customization. You can use the [Fedora Workstation](#) image, which comes with the GNOME desktop by default, and then change your environment afterwards by installing additional packages, or you can download a spin image which will give you a different environment out of the box. Visit [Fedora Spins](#) for more information.

You can also take advantage of Fedora Labs. Fedora Labs is a selection of curated bundles of purpose-driven software and content as curated and maintained by members of the Fedora Community. These may be installed as standalone full versions of Fedora or as add-ons to existing Fedora installations. Visit [Fedora Labs](#) for details.

Please refer to [f20@fedora:install-guide:index.adoc](#) for getting help on the process of installing Fedora.

Creating and using live USB

You can write all Fedora ISO images to a USB stick, making this a convenient way on any USB-bootable computer to either install Fedora or try a **live** Fedora environment without writing to the computer's hard disk. You will need a USB stick at least as large as the image you wish to write.

Using Fedora Media Writer

The official and supported tool to create a Fedora USB stick is the **Fedora Media Writer** utility, which was formerly known as **LiveUSB Creator**. To learn more about this application and how to use it, refer to the [Preparing for Installation](#) chapter of the Fedora Installation Guide.

Fedora Media Writer destroys all data on the USB stick. If you need a non-destructive write method (to preserve existing data on your USB stick) or support for 'data persistence', you can use the [livecd-iso-to-disk](#) utility on Fedora.

Using GNOME Disk Utility

This method will destroy all data on the USB stick. If you need a non-destructive write method (to preserve existing data on your USB stick) and/or support for 'data persistence', you can use the `livecd-iso-to-disk` utility on Fedora.

This method is considered unsupported. You can use it on your own risk.

This method is for people running Linux, or another unix with GNOME, Nautilus and the GNOME Disk Utility installed. Particularly, if you are using a distribution other than Fedora which does not support Flatpak, this may be the easiest available method. A standard installation of Fedora, or a standard GNOME installation of many other distributions, should be able to use this method. On Fedora, ensure the packages *nautilus* and *gnome-disk-utility* are installed. Similar graphical direct-write tools may be available for other desktops, or you may use the command-line *direct write* method.

1. Download a Fedora image, choose a USB stick that does not contain any data you need, and connect it.
2. Run Nautilus (Files), open the **Overview** by pressing the **Start/Super** key, type Files, and hit **Enter**.
3. Find the downloaded image, right-click on it, go to **Open With**, and click **Disk Image Writer**.
4. Select your USB stick as the **Destination**, and click **Start Restoring**.

Command line methods

These methods are considered unsupported. You can use them on your own risk.

Using the livecd-iso-to-disk tool

This method will destroy all data on the USB stick *if the `--format` parameter is passed*.

The `livecd-iso-to-disk` method is slightly less reliable than Fedora Media Writer and can be used reliably only from within Fedora: it does not work in Windows or OS X, and is not supported (and will usually fail) in non-Fedora distributions. However, it supports three advanced features which FMW does not include:

1. You may use a *non-destructive* method to create the stick, meaning existing files on the stick will not be destroyed. This is less reliable than the *destructive* write methods, and should be used only if you have no stick you can afford to wipe.
2. On live images, you can include a feature called a *persistent overlay*, which allows changes made to persist across reboots. You can perform updates just like a regular installation to your hard disk, except that kernel updates require manual intervention and overlay space may be insufficient. Without a *persistent overlay*, the stick will return to a fresh state each time it is booted.
3. On live images, you can also have a separate area to store user account information and data such as documents and downloaded files, with optional encryption for security and peace of mind.

By combining these features, you can carry your computer with you in your pocket, booting it on nearly any system you find yourself using.

It is not a good idea to try and write a new Fedora release using the version of `livecd-iso-to-disk` in a much older Fedora release: it is best to only use a release a maximum of two versions older than the release you are trying to write.

Ensure the `livecd-tools` package is installed: `dnf install livecd-tools` .

Remember to identify your USB stick's device name first. In all cases, you can add the parameter `--efi` to render the stick bootable in native UEFI mode. Detailed usage information is available by running: `livecd-iso-to-disk --help` or `man livecd-iso-to-disk` .

To make an existing USB stick bootable as a Fedora image, without deleting any of the data on it, make sure that the USB drive is not mounted before executing the following, and give the root password when prompted:

```
# livecd-iso-to-disk Fedora-Workstation-Live-x86_64-20-1.1.iso /dev/sdX
```

In case it is not possible to boot from a disk created with the method shown above, before re-partitioning and re-formatting, often resetting the master boot record will enable booting:

```
# livecd-iso-to-disk --reset-mbr Fedora-Workstation-Live-x86_64-20-1.1.iso /dev/sdX
```

Using the `--format` option in the following command will erase all data on the USB drive.

If necessary, you can have `livecd-iso-to-disk` re-partition and re-format the target stick:

```
# livecd-iso-to-disk --format --reset-mbr Fedora-Workstation-Live-x86_64-20-1.1.iso /dev/sdX
```

To include a persistent filesystem for `/home`, use the `--home-size-mb` parameter. For example:

```
# livecd-iso-to-disk --home-size-mb 2048 Fedora-Workstation-Live-x86_64-20-1.1.iso /dev/sdX
```

This will create a 2 GiB filesystem that will be mounted as `/home` each time the stick is booted, allowing you to preserve data in `/home` across boots.

To enable 'data persistence' support - so changes you make to the entire live environment will persist across boots - add the `--overlay-size-mb` parameter to add a persistent data storage area to the target stick. For example:

```
# livecd-iso-to-disk --overlay-size-mb 2048 Fedora-Workstation-Live-x86_64-20-1.1.iso /dev/sdX
```

Here, `2048` is the desired size (in megabytes) of the overlay. The `livecd-iso-to-disk` tool will not accept an overlay size value greater than 4095 for VFAT, but for ext[234] filesystems it is only limited by the available space.

Due to the way it's currently implemented, every single change to this form of overlay, writes AND deletes, subtracts from its free space so it will eventually be "used up" and your USB stick will no longer boot. You can use `dmsetup status live-rw` to see how much space remains in the overlay.

The output will contain something like snapshot `42296/204800`, indicating that 4229 of 204800 512-byte sectors are allocated. Because of these limitations, it is advisable to use the `system-level` persistence sparingly, for configuration changes and important security updates only. Or, if you have sufficient disk space available, changes to the `LiveOS` root filesystem snapshot can be merged into a new copy of the root filesystem.

You can combine `--home-size-mb` and `--overlay-size-mb`, in which case data written to `/home` will not exhaust the persistent overlay.

Using a direct write method

This method will destroy all data on the USB stick. If you need a non-destructive write method, to preserve existing data on your USB stick, and/or support for `data persistence`, you can use the `livecd-iso-to-disk` utility on Fedora.

This method directly writes the image to the USB stick much like Fedora Media Writer or GNOME Disk Utility, but uses a command line utility named `dd`. Like the other *direct write* methods, it will destroy all data on the stick and does not support any of the advanced features like data

persistence, but it is a very reliable method. The `dd` tool is available on most Unix-like operating systems, including Linux distributions and OS X, and a Windows port is available. This may be your best method if you cannot use [Fedora Media Writer](#) or GNOME Disk Utility, or just if you prefer command line utilities and want a simple, quick way to write a stick.

1. Identify the name of the USB drive partition. If using this method on Windows, with the port linked above, the `dd --list` command should provide you with the correct name.
2. **Unmount all mounted partition from that device.** This is very important, otherwise the written image might get corrupted. You can unmount all mounted partitions from the device with `umount /dev/sdX*`, where `X` is the appropriate letter, e.g. `umount /dev/sdc*`.

3. Write the ISO file to the device:

```
# dd if=/path/to/image.iso of=/dev/sdX bs=8M status=progress oflag=direct
```

4. Wait until the command completes.

If you see `dd: invalid status flag: 'progress'`, your `dd` version doesn't support the `status=progress` option and you'll need to remove it. In this case, you won't see writing progress.

Using UNetbootin for Windows, OS X, and Linux

This method is considered unsupported. You can use it on your own risk.

UNetbootin may work in some cases but not others - for instance, it will likely create a stick that is bootable in BIOS mode, but not UEFI mode. Fedora cannot guarantee support for UNetbootin-written images.

While your results may vary, it is usually the case that the Fedora Media Writer, `livecd-iso-to-disk`, GNOME, and `dd` methods give better results than UNetbootin. If you encounter problems with UNetbootin, please contact the UNetbootin developers, not the Fedora developers.

[UNetbootin](#) is a graphical, bootable USB image creator. Using it will allow you to preserve any data you have in the USB drive. If you have trouble booting, however, you may wish to try with a blank, cleanly FAT32-formatted drive.

If you are running a 64-bit Linux distribution, UNetbootin may fail to run until you install the 32-bit versions of quite a lot of system libraries.

1. Download the latest UNetbootin version from the [official site](#) and install it. On Linux, the download is an executable file: save it somewhere, change it to be executable using `chmod ugo+x filename` or a file manager, and then run it.

2. Launch UNetbootin. On Linux, you might have to type the root password.
3. Click on **Diskimage** and search for the ISO file you downloaded.
4. Select Type: USB drive and choose the correct device for your stick.
5. Click OK.

If you do not see *sdX* listed, you might have to reformat the drive. You can do this from most file manager or disk utility tools, e.g. the GNOME disk utility ("Disks") on Fedora. The FAT32 format is most likely to result in a bootable stick. This will cause you to lose all data on the drive.

Creating a USB stick from a running live environment

If you are already running a live CD, DVD, or USB and want to convert that into a bootable USB stick, run the following command:

```
# livecd-iso-to-disk /run/initramfs/livedev /dev/sdX"
```

Booting from USB sticks

Almost all modern PCs can boot from USB sticks. However, how you tell the system to boot from a USB stick varies substantially from system to system. Initially, you can try this:

1. Power off the computer.
2. Plug the USB drive into a USB port.
3. Remove all other portable media, such as CDs, DVDs, floppy disks or other USB sticks.
4. Power on the computer.
5. If the computer is configured to automatically boot from the USB drive, you will see a screen that says "Automatic boot in 10 seconds..." with a countdown.

If you do a native UEFI boot, where you will see a rather more minimal boot menu.

If the computer starts to boot off the hard drive as normal, you'll need to manually configure it to boot off the USB drive. Usually, that should work like this:

1. Wait for a safe point to reboot.
2. As the machine starts to reboot, watch carefully for instructions on which key to press. Usually a function key, **Escape** , **Tab** or **Delete** is to be pressed to enter the boot device selection menu, **BIOS setup** , **firmware** , or **UEFI** . Press and hold that key. If you miss the window of opportunity, often only a few seconds, then reboot and try again.

3. Use the firmware, `BIOS`, interface or the boot device menu to put your USB drive first in the boot sequence. It might be listed as a hard drive rather than a removable drive. Each hardware manufacturer has a slightly different method for doing so.

Your computer could become unbootable or lose functionality if you change any other settings. Though these settings can be reverted, you'll need to remember what you changed in order to do so.

4. Save the changes, exit, and the computer should boot from the USB drive.

If your system has a UEFI firmware, it will usually allow you to boot the stick in UEFI native mode or BIOS compatibility mode. If you boot in UEFI native mode and perform a Fedora installation, you will get a UEFI native Fedora installation. If you boot in BIOS compatibility mode and perform a Fedora installation, you will get a BIOS compatibility mode Fedora installation.

For more information on all this, see the [UEFI page](#). USB sticks written from x86_64 images with [Fedora Media Writer](#), [GNOME Disk Utility](#), `dd`, other `dd`-style utilities should be UEFI native bootable. Sticks written with other utilities may not be UEFI native bootable, and sticks written from i686 images will never be UEFI bootable.

Identifying a stick on Linux

Most of the writing methods will require you to know the `/dev` name for your USB stick, e.g. `/dev/sdc`, when using them on Linux. You do not need to know this in order to use Fedora Media Writer. To find this out:

1. Insert the USB stick into a USB port.
2. Open a terminal and run `dmesg`.
3. Near the end of the output, you will see something like:

```
[32656.573467] sd 8:0:0:0: [sdX] Attached SCSI removable disk
```

`sdX` will be `sdb`, `sdc`, `sdd`, etc.

This is the name of the disk you will use. We'll call it `sdX` from now on. If you have connected more than one USB stick to the system, be careful that you identify the correct one, often you will see a manufacturer name or capacity in the output which you can use to make sure you identified the correct stick.

Troubleshooting a live USB

livecd-iso-to-disk problems

Partition isn't marked bootable

If you get the message **Partition isn't marked bootable!** , you need to mark the partition bootable. To do this, run **parted /dev/sdX** , and use the **toggle N** boot command, where **X** is the appropriate letter, and **N** is the partition number. For example:

```
$ parted /dev/sdb
GNU Parted 1.8.6
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) print
Model: Imation Flash Drive (scsi)
Disk /dev/sdX: 1062MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
```

Number	Start	End	Size	Type	File system	Flags
1	32.3kB	1062MB	1062MB	primary	fat16	

```
(parted) toggle 1 boot
(parted) print
Model: Imation Flash Drive (scsi)
Disk /dev/sdX: 1062MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
```

Number	Start	End	Size	Type	File system	Flags
1	32.3kB	1062MB	1062MB	primary	fat16	boot

```
(parted) quit
Information: Don't forget to update /etc/fstab, if necessary.
```

Partitions need a filesystem label

If you get the message **Need to have a filesystem label** or **UUID** for your USB device, you need to label the partition: **dosfslabel /dev/sdX LIVE** .

Partition has different physical/logical endings

If you get this message from fdisk, you may need to reformat the flash drive when writing the image, by passing **--format** when writing the stick.

MBR appears to be blank

If your test boot reports a corrupted boot sector, or you get the message **MBR appears to be blank.** , you need to install or reset the master boot record (MBR), by passing **--reset-mbr** when writing the stick.

livecd-iso-to-disk on other Linux distributions

livecd-iso-to-disk is not meant to be run from a non-Fedora system. Even if it happens to run and write a stick apparently successfully from some other distribution, the stick may well fail to boot. Use of **livecd-iso-to-disk** on any distribution other than Fedora is unsupported and not expected to work: please use an alternative method, such as [Fedora Media Writer](#).

Testing a USB stick using qemu

You can test your stick using QEMU.

```
# umount /dev/sdX1
$ qemu -hda /dev/sdX -m 1024 -vga std
```

Mounting a Live USB filesystem

You can use the [liveimage-mount](#) script in the [livecd-tools](#) package to mount an attached Live USB device or other LiveOS image, such as an ISO or Live CD. This is convenient when you want to copy in or out some file from the LiveOS filesystem on a Live USB, or just examine the files in a Live ISO or Live CD.

Creating and using live CD

FIXME Needs modularization

Getting started

To create a live image, the `livecd-creator` tool is used. For this, super user privileges are needed.

The `livecd-creator` tool is part of the `_livecd-tools_package`. If it is not installed on your system, add it with DNF:

```
# dnf install livecd-tools spin-kickstarts
```

If you are interested in localized (i.e. translated into other languages) live CD files, install also *l10n-kickstarts*.

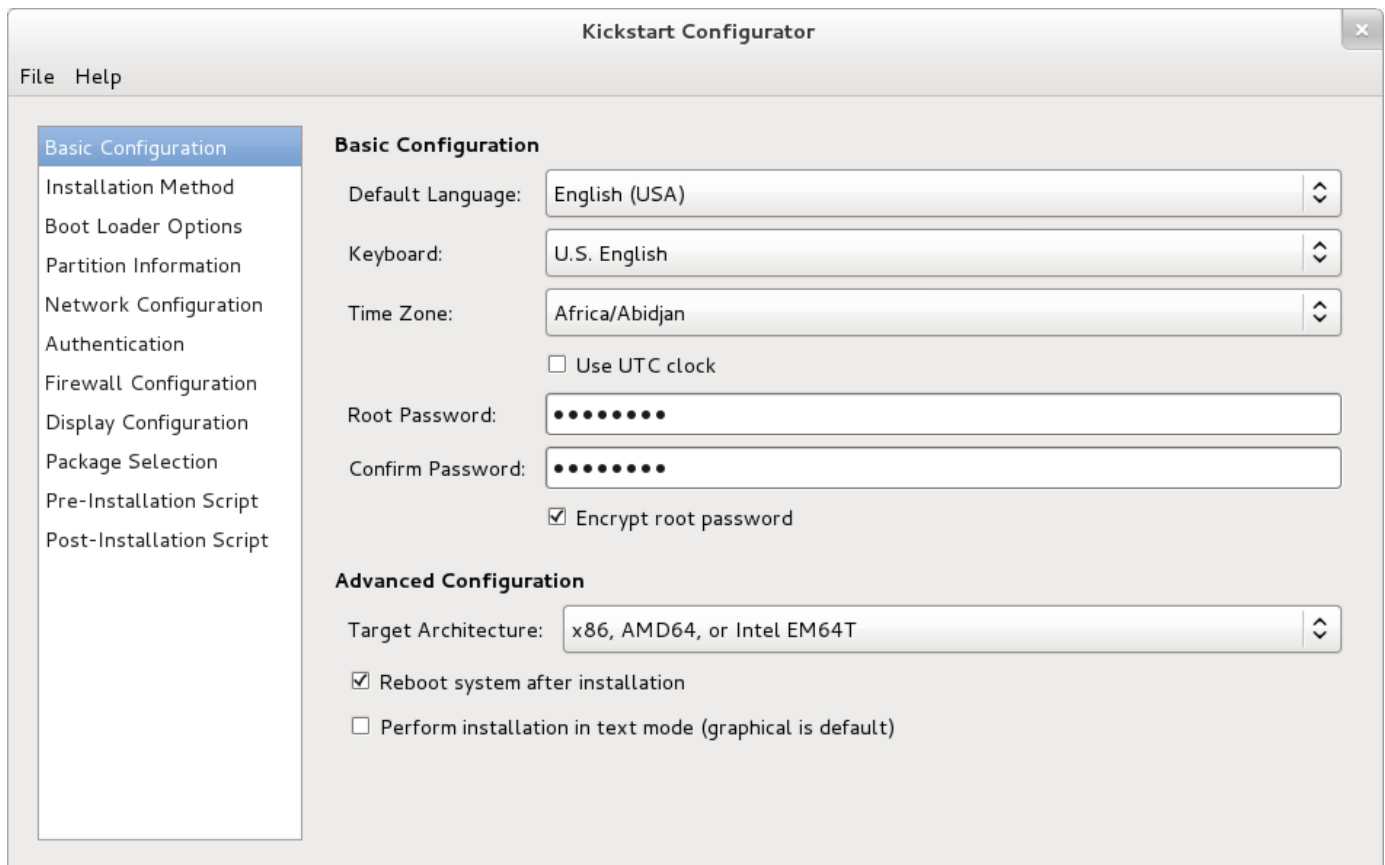
Configuring the image

The configuration of the live image is defined by a file called *kickstart*. It can include some basic system configuration items, the package manifest, and a script to be run at the end of the build process.

For the Fedora project, the most important live image configurations files are:

- [fedora-live-base.ks](#) : The base live image system, included in the *livecd-tools* package.
- For *Fedora 20 and earlier*: [fedora-livecd-desktop.ks](#) : Complete desktop with applications and input/output support for all supported locales in Fedora. This one is part of the `spin-kickstarts` package. Despite the name, this is the kickstart that generates the ~1GB-sized images for recent releases.
- For *Fedora 21 and later*: [fedora-live-workstation.ks](#). This is the Workstation product configuration.

kickstart files for other spins, e.g. Fedora Electronics Lab, can be found in `/usr/share/spin-kickstarts/` after installing the `spin-kickstarts` package. These pre-made configuration files can be a great place to start, as they already have some useful pre and post-installation scripts.



The screenshot shows the 'Kickstart Configurator' window. On the left is a sidebar with a list of configuration categories: Basic Configuration (selected), Installation Method, Boot Loader Options, Partition Information, Network Configuration, Authentication, Firewall Configuration, Display Configuration, Package Selection, Pre-Installation Script, and Post-Installation Script. The main area is divided into two sections. The 'Basic Configuration' section includes fields for 'Default Language' (English (USA)), 'Keyboard' (U.S. English), 'Time Zone' (Africa/Abidjan), a checkbox for 'Use UTC clock' (unchecked), 'Root Password' (masked with dots), 'Confirm Password' (masked with dots), and a checkbox for 'Encrypt root password' (checked). The 'Advanced Configuration' section includes a 'Target Architecture' dropdown (x86, AMD64, or Intel EM64T), a checkbox for 'Reboot system after installation' (checked), and a checkbox for 'Perform installation in text mode (graphical is default)' (unchecked).

You can create a customized *kickstart* file by running `system-config-kickstart`.

You might have to install the package first with `dnf install system-config-kickstart` in Fedora 22 and beyond or in earlier versions of Fedora. This tool is mainly intended for generating kickstart files for automated installs, not live images, so the output will probably not be usable without editing, but it may help you to generate particular kickstart directives. Remember to add the line `%include /usr/share/spin-kickstarts/fedora-live-base.ks` at the beginning of your *kickstart* file to include the base live configuration.

Making the image

To make the image, simply issue the following command:

```
livecd-creator --verbose \  
--config=/path/to/kickstart/file.ks \  
--fslabel=Image-Label \  
--cache =/var/cache/live
```

The name given by `--fs-label` is used:

- As a file system label on the ext3 and iso9660 file systems. As such, it's visible on the desktop as the CD name.
- In the *isolinux* boot loader.

If you have the repositories available locally and don't want to wait for the download of packages, just substitute the URLs listed in the configuration file to point to your local repositories.

If you have an x86_64 machine you're building on but you want a 32-bit happy iso image, add the following before your livecd-creator command:

```
setarch i686 livecd-creator [...]
```

Examples

Spinning the Fedora desktop

The following command:

```
livecd-creator --verbose \  
--config=/usr/share/spin-kickstarts/fedora-live-workstation.ks \  
--fslabel=Fedora-LiveCD \  
--cache=/var/cache/live
```

This will create a live CD called **Fedora-LiveCD** using the `fedora-live-workstation.ks` configuration file.

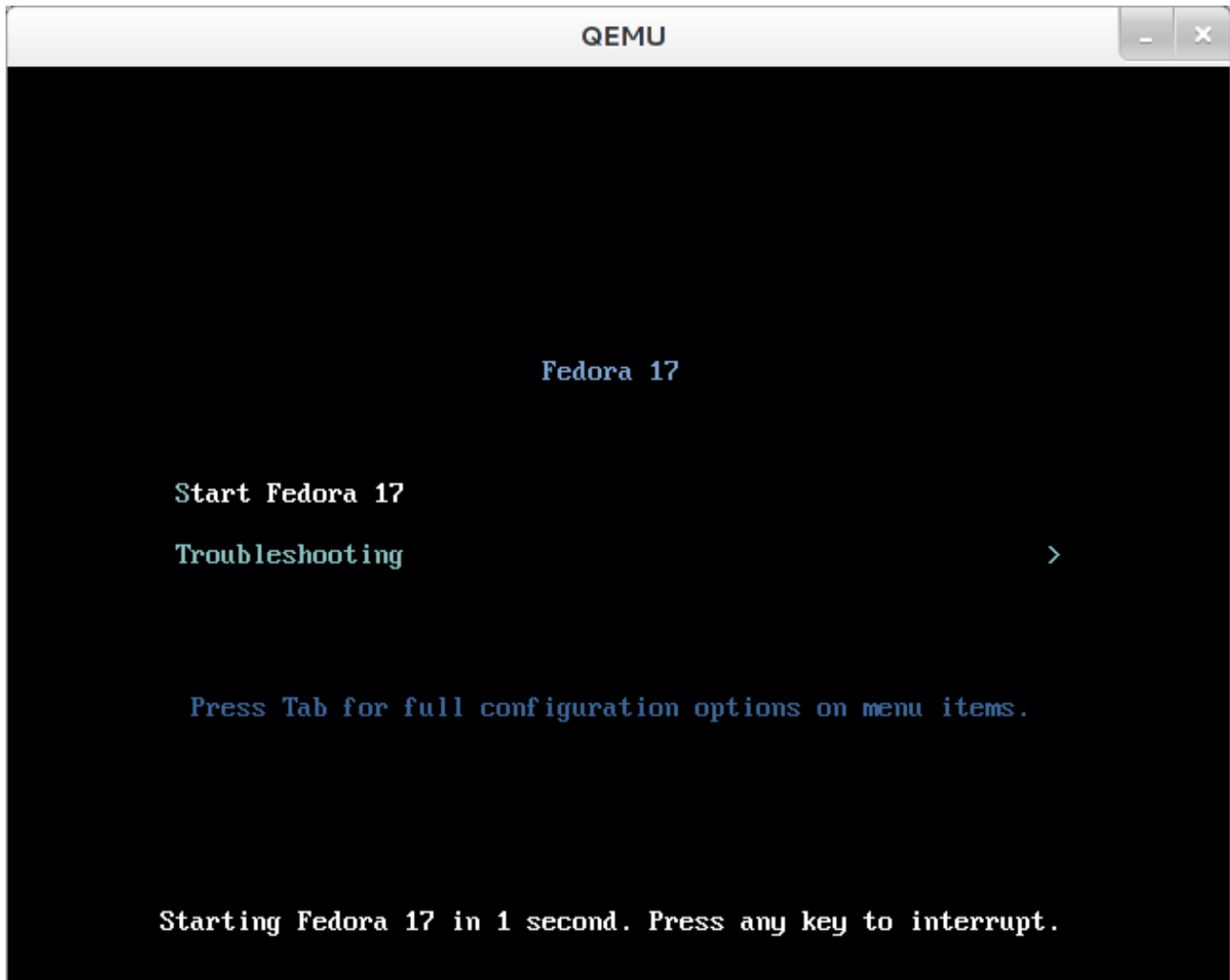
A Barebones live CD

The following command:

```
livecd-creator --verbose \  
--config=/usr/share/doc/livecd-tools-$(rpm -q livecd-tools --qf "%{VERSION}")/livecd-fedora-minimal.ks \  
--cache=/var/cache/live
```

This will create a live CD that will boot to a login prompt.

Testing your live CD using KVM or qemu



As root:

```
# qemu-kvm -m 2048 -vga qxl -cdrom filename.iso
```

If you do not have KVM support, you have to use `qemu` instead.

```
# qemu-system-x86_64 -m 2048 -vga qxl -cdrom filename.iso
```

Replace `filename.iso` with the name of your created Live CD image and `qemu-system-x86_64` with an appropriate qemu binary for the target system, e.g. `qemu-system-i386`.

Using your new live image

You can burn your image directly to a CD or a DVD if it fits, or you can write it to a USB stick.

Live image media verification

The live image can incorporate functionality to verify itself. To do so, you need to have `isomd5sum` installed both on the system used for creating the image and installed into the

image. This is so that the `implantisomd5` and `checkisomd5` utilities can be used. These utilities take advantage of embedding an *md5sum* into the application area of the iso9660 image. This then gets verified before mounting the real root filesystem.

Want to help? [Learn how to contribute to Fedora Docs.](#)