Your submission was sent successfully! *Close*

You have successfully unsubscribed! *Close*

How to verify your Ubuntu download

1. Overview

While we hope you can usually trust your Ubuntu download, it is definitely reassuring to be able to verify that the image you have downloaded is not corrupted in some way, and also that it is an authentic image that hasn't been tampered with.

What you'll learn

- How to use `gpg` tools to verify the authenticity of a file
- How to use `sha256` tools to verify the integrity of a file

What you'll need

- Access to the command line gpg tools
- Internet access to download the signatures

Originally authored by Canonical Web Team

---

2. Necessary software

The key executables you will require are `sha256sum`, `md5sum` and `gpg`.

For Ubuntu

These are part of the `coreutils` and `gnupg` packages, which are installed by default.

For Windows

If you are using bash on Windows 10 (why on earth not? See this tutorial <https://tutorials.ubuntu.com/tutorial/tutorial-ubuntu-on-windows>), these tools are part of the default install.

For macOS

You can install the latest GnuPG using Homebrew <https://brew.sh/> :

```
brew install gnupg
```

The `sha256sum` program and other useful utilities are provided by `coreutils`:

```
brew install coreutils
```

For other versions of Linux

Your mileage may vary, but these are standard tools included and enabled by default in most systems. If you don't have them, check with your package manager and search for the executable names given above.

All versions - check the commands are working!

You can check the commands work as expected by running the following:

```
gpg --list-keys
```

If this is the first time you have run `gpg`, this will create a trust database for the current user.

```
md5sum --version
sha256sum --version
```

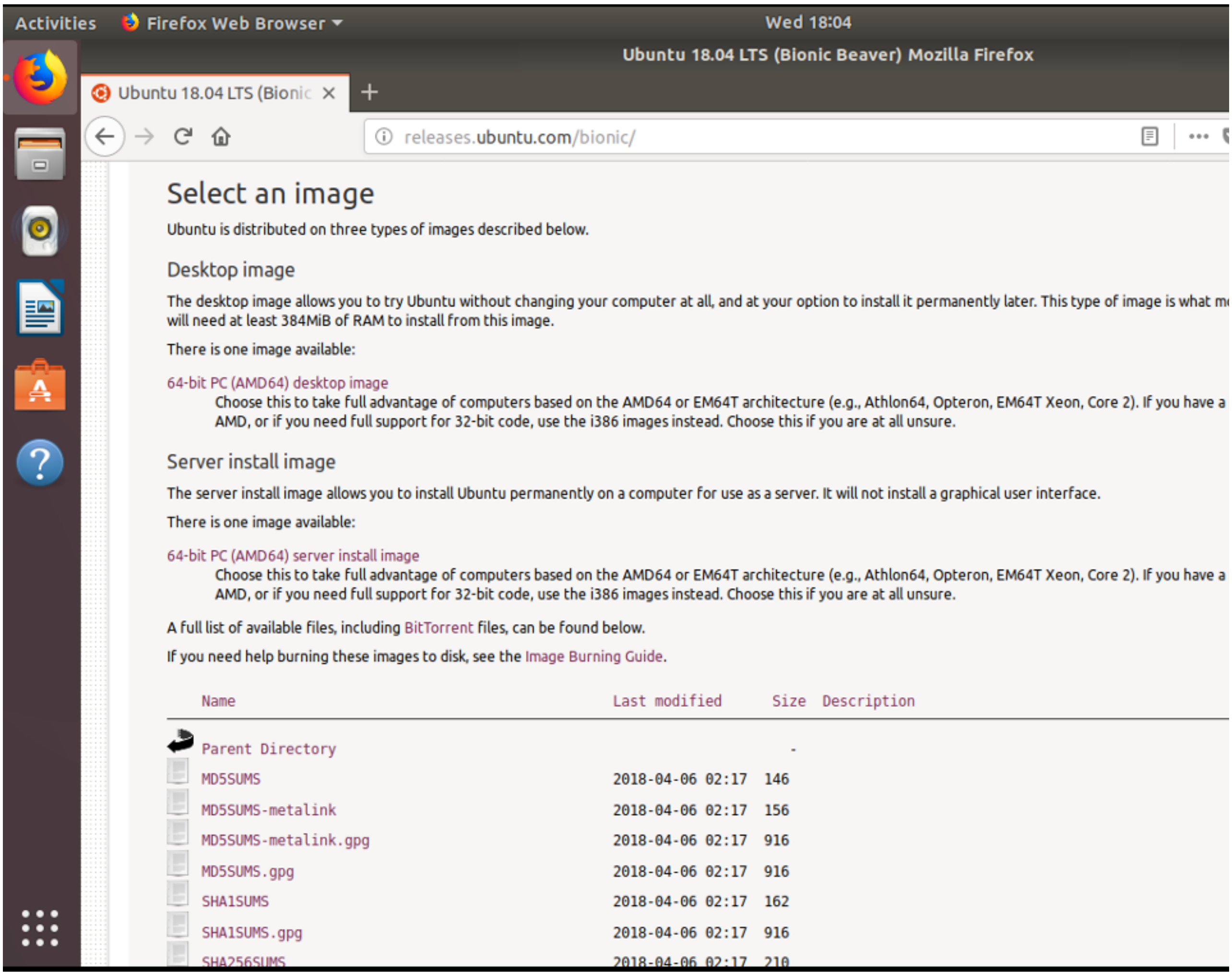
Both these commands should output some version information. Now we have the tools we need, we can move on to finding and downloading the files we need

3. Download checksums and signatures

Alongside the actual ISO files containing the Ubuntu image you downloaded, all Ubuntu mirrors publish some extra files. The ones we are interested in are called:

```
SHA256SUMS
SHA256SUMS.gpg
```

It is usually convenient to download these at the same time as downloading the distro. However, if you didn't, not to worry - the checksums and the signature are consistent for the image, so even if you downloaded your ISO file from a different source, as long as it is fresh and hasn't been updated in the interim, you can fetch these files from the http://releases.ubuntu.com <http://releases.ubuntu.com> page for the relevant release. You will usually find the relevant files on the top of the directory listing.


Activities
Firefox Web Browser
Wed 18:04
Ubuntu 18.04 LTS (Bionic Beaver) Mozilla Firefox
Ubuntu 18.04 LTS (Bionic
releases.ubuntu.com/bionic/

Select an image

Ubuntu is distributed on three types of images described below.

Desktop image

The desktop image allows you to try Ubuntu without changing your computer at all, and at your option to install it permanently later. This type of image is what m will need at least 384MiB of RAM to install from this image.

There is one image available:

64-bit PC (AMD64) desktop image
Choose this to take full advantage of computers based on the AMD64 or EM64T architecture (e.g., Athlon64, Opteron, EM64T Xeon, Core 2). If you have a AMD, or if you need full support for 32-bit code, use the i386 images instead. Choose this if you are at all unsure.

Server install image

The server install image allows you to install Ubuntu permanently on a computer for use as a server. It will not install a graphical user interface.

There is one image available:

64-bit PC (AMD64) server install image
Choose this to take full advantage of computers based on the AMD64 or EM64T architecture (e.g., Athlon64, Opteron, EM64T Xeon, Core 2). If you have a AMD, or if you need full support for 32-bit code, use the i386 images instead. Choose this if you are at all unsure.

A full list of available files, including BitTorrent files, can be found below.

If you need help burning these images to disk, see the Image Burning Guide.

Name	Last modified	Size	Description
Parent Directory		-	
MD5SUMS	2018-04-06 02:17	146	
MD5SUMS-metalink	2018-04-06 02:17	156	
MD5SUMS-metalink.gpg	2018-04-06 02:17	916	
MD5SUMS.gpg	2018-04-06 02:17	916	
SHA1SUMS	2018-04-06 02:17	162	
SHA1SUMS.gpg	2018-04-06 02:17	916	
SHA256SUMS	2018-04-06 02:17	210	

Note - some people question that if the site they are downloading from is not secure (many archive mirrors do not use SSL), how can they trust the signatures? The gpg fingerprint is checked against the Ubuntu keyserver, so if the signature matches, you know it is authentic no matter where/how it was downloaded!

The `SHA256SUMS` file contains checksums for **all** the available images (you can check this by opening the file) where a checksum exists - development and beta versions sometimes do not generate new checksums for each release.

The `SHA256SUMS.gpg` file is the GnuPG signature for that file. In the next step we will use this signature file to **verify** the checksum file.

4. Retrieve the correct signature key

Depending on your platform, you may or may not need to download the public key used to authenticate the checksum file (Ubuntu and most variants come with the relevant keys pre-installed). The easiest way to find out if you need the key is to run the authentication command:

```
gpg --keyid-format long --verify SHA256SUMS.gpg SHA256SUMS
```

We use GnuPG's "long" (64-bit) key IDs throughout this tutorial, since "short" (32-bit) key IDs are insecure.

If you don't have the keys...

If there is no public key for Ubuntu already present, you will get an error message similar to the following:

```
gpg: Signature made Thu Apr  5 22:19:36 2018 EDT
                    using DSA key ID 46181433FBB75451
gpg: Can't check signature: No public key
gpg: Signature made Thu Apr  5 22:19:36 2018 EDT
                    using RSA key ID D94AA3F0EFE21092
gpg: Can't check signature: No public key
```

This is actually a really useful message, as it tells us which key or keys were used to generate the signature file. Knowing these ID numbers (46181433FBB75451 and D94AA3F0EFE21092 in the example), means we can request them from the Ubuntu key server.

This is done with the following command. Note that the ID numbers are hexadecimal, so we prefix them with `0x`:

```
gpg --keyid-format long --keyserver hkp://keyserver.ubuntu.com --recv-keys 0x46181433FBB75451 0xD94AA3F0EFE21092
```

This command should retrieve the keys we want and add them to your keyring. You should see a message like this:

```
gpg: requesting key 46181433FBB75451 from hkp server keyserver.ubuntu.com
gpg: requesting key D94AA3F0EFE21092 from hkp server keyserver.ubuntu.com
gpg: /root/.gnupg/trustdb.gpg: trustdb created
gpg: key 46181433FBB75451: public key "Ubuntu CD Image Automatic Signing Key <cdimage@ubuntu.com>" imported
gpg: key D94AA3F0EFE21092: public key "Ubuntu CD Image Automatic Signing Key (2012) <cdimage@ubuntu.com>" imported
gpg: no ultimately trusted keys found
gpg: Total number processed: 2
gpg:               imported: 2  (RSA: 1)
```

You can now inspect the key fingerprints by running:

```
gpg --keyid-format long --list-keys --with-fingerprint 0x46181433FBB75451 0xD94AA3F0EFE21092
```

…which should produce the following output:

```
pub   dsa1024/46181433FBB75451 2004-12-30 [SC]
      Key fingerprint = C598 6B4F 1257 FFA8 6632  CBA7 4618 1433 FBB7 5451
uid                  Ubuntu CD Image Automatic Signing Key <cdimage@ubuntu.com>

pub   rsa4096/D94AA3F0EFE21092 2012-05-11 [SC]
      Key fingerprint = 8439 38DF 228D 22F7 B374  2BC0 D94A A3F0 EFE2 1092
uid                  Ubuntu CD Image Automatic Signing Key (2012) <cdimage@ubuntu.com>
```

5. Verify the SHA256 checksum

Now you can verify the checksum file using the signature.

```
gpg --keyid-format long --verify SHA256SUMS.gpg SHA256SUMS
```

This time the command should return something like this:

```
gpg: Signature made Fri 25 Mar 04:36:20 2016 GMT
                    using DSA key ID 46181433FBB75451
gpg: Good signature from "Ubuntu CD Image Automatic Signing Key <cdimage@ubuntu.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: C598 6B4F 1257 FFA8 6632  CBA7 4618 1433 FBB7 5451
gpg: Signature made Fri 25 Mar 04:36:20 2016 GMT
                    using RSA key ID D94AA3F0EFE21092
gpg: Good signature from "Ubuntu CD Image Automatic Signing Key (2012) <cdimage@ubuntu.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 8439 38DF 228D 22F7 B374  2BC0 D94A A3F0 EFE2 1092
```

A **Good** signature means that the checked file was definitely signed by the owner of the keyfile stated (if they didn't match, the signature would be reported as **BAD**). The warning message is simply there to let you know that you have not countersigned the key and it isn't in your list of trusted sources. If you want to know more about signing keys and trust, you can check out the Ubuntu community GPG wiki page <https://help.ubuntu.com/community/GnuPrivacyGuardHowto> .

Now that we have verified the checksum file was created by Ubuntu, we can check that the ISO file we downloaded matches the checksum.

6. Check the ISO

Now you need to generate a sha256 checksum for the downloaded ISO and compare it to the one you downloaded in your `SHA256SUM` file.

Make sure the downloaded the `SHA256SUMS` and `SHA256SUMS.gpg` files are in the same directory as the Ubuntu ÌSO file. Then run the following commands in a terminal.

```
sha256sum -c SHA256SUMS 2>&1 | grep OK
```

The output you want will look similar to the following:

```
ubuntu-18.04-desktop-amd64.iso: OK
```

If you get no results (or any result other than that shown above) then the ISO file does not match the checksum. This could be because the ISO has been altered, or it downloaded incorrectly - either way you should download a fresh ISO from a known good source.

7. What’s next?

Now you know you have a good ISO image, you can burn it to a DVD or copy it to a bootable USB stick to install or try Ubuntu!

Here are a few more tutorials you may want to look at:

- How to burn a DVD on Ubuntu <https://tutorials.ubuntu.com/tutorial/tutorial-burn-a-dvd-on-ubuntu>, Windows <https://tutorials.ubuntu.com/tutorial/tutorial-burn-a-dvd-on-windows> or macOS <https://tutorials.ubuntu.com/tutorial/tutorial-burn-a-dvd-on-macos>
- How to create a bootable USB stick on Ubuntu <https://tutorials.ubuntu.com/tutorial/tutorial-create-a-usb-stick-on-ubuntu>, Windows <https://tutorials.ubuntu.com/tutorial/tutorial-create-a-usb-stick-on-windows> or macOS <https://tutorials.ubuntu.com/tutorial/tutorial-create-a-usb-stick-on-macos>
- Trying out Ubuntu <https://tutorials.ubuntu.com/tutorial/try-ubuntu-before-you-install> before you install
- How to install Ubuntu <https://tutorials.ubuntu.com/tutorial/tutorial-install-ubuntu-desktop>

Finding help

If you get stuck, help is always at hand.

- Ask Ubuntu <https://askubuntu.com/>
- Ubuntu Forums <https://ubuntuforums.org/>
- IRC-based support <https://wiki.ubuntu.com/IRC/ChannelList>

Was this tutorial useful?

-
-
-