linuxbabe.com

How to Verify PGP Signature of Downloaded Software on Linux - LinuxBabe

Xiao Guoan (Admin)

5-6 minutes

PGP (Pretty Good Privacy) is a public key cryptography software that can be used to encrypt and sign data communication. In this tutorial, we will look at **how to verify the PGP signature of software downloaded from the Internet on Linux**.

Linux users can securely install software from their distribution’s repositories. But there are times when you need to download and install software from a website. How can you be sure that the software you downloaded wasn’t tampered with?

Some software authors sign their software using a PGP program such as **GPG** (GNU Privacy Guard), which is a free software implementation of the OpenPGP standard. In that case, you can verify the integrity of software using GPG.

The process is relatively simple:

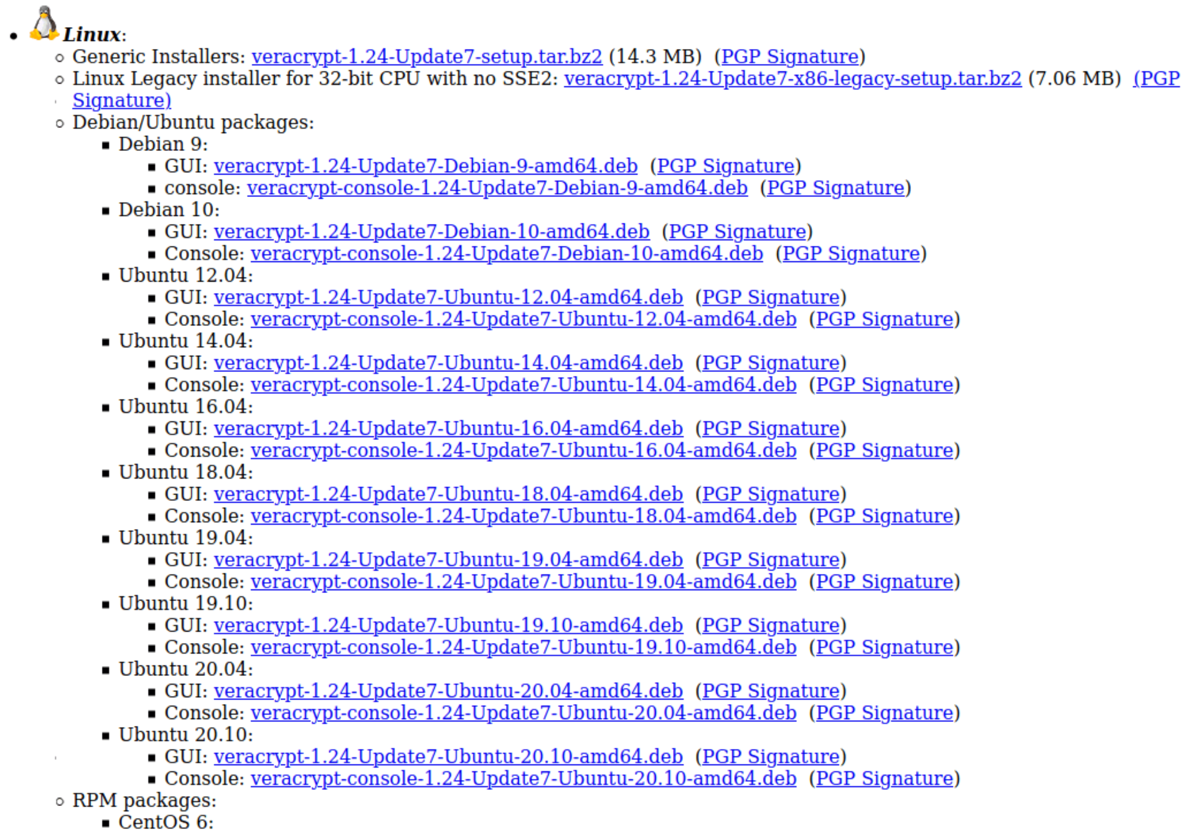
1. You download the public key (`.asc` file) of the software author.
2. Check the public key’s fingerprint to ensure that it’s the correct key.
3. Import the correct public key to your GPG public keyring.
4. Download the PGP signature file (`.sig`) of the software.
5. Use public key to verify PGP signature. If the signature is correct, then the software wasn’t tampered with.

We will use VeraCrypt as an example to show you how to verify

PGP signature of downloaded software.

Example: Verify PGP Signature of VeraCrypt

Although VeraCrypt is open source software, it isn’t included in Ubuntu repository. We can download VeraCrypt Linux installer from official website. I use Ubuntu 20.04 desktop, so I download the `.deb` file for Ubuntu 20.04.



- ***Linux***:
  - Generic Installers: veracrypt-1.24-Update7-setup.tar.bz2 (14.3 MB) (PGP Signature)
  - Linux Legacy installer for 32-bit CPU with no SSE2: veracrypt-1.24-Update7-x86-legacy-setup.tar.bz2 (7.06 MB) (PGP Signature)
  - Debian/Ubuntu packages:
    - Debian 9:
      - GUI: veracrypt-1.24-Update7-Debian-9-amd64.deb (PGP Signature)
      - console: veracrypt-console-1.24-Update7-Debian-9-amd64.deb (PGP Signature)
    - Debian 10:
      - GUI: veracrypt-1.24-Update7-Debian-10-amd64.deb (PGP Signature)
      - Console: veracrypt-console-1.24-Update7-Debian-10-amd64.deb (PGP Signature)
    - Ubuntu 12.04:
      - GUI: veracrypt-1.24-Update7-Ubuntu-12.04-amd64.deb (PGP Signature)
      - Console: veracrypt-console-1.24-Update7-Ubuntu-12.04-amd64.deb (PGP Signature)
    - Ubuntu 14.04:
      - GUI: veracrypt-1.24-Update7-Ubuntu-14.04-amd64.deb (PGP Signature)
      - Console: veracrypt-console-1.24-Update7-Ubuntu-14.04-amd64.deb (PGP Signature)
    - Ubuntu 16.04:
      - GUI: veracrypt-1.24-Update7-Ubuntu-16.04-amd64.deb (PGP Signature)
      - Console: veracrypt-console-1.24-Update7-Ubuntu-16.04-amd64.deb (PGP Signature)
    - Ubuntu 18.04:
      - GUI: veracrypt-1.24-Update7-Ubuntu-18.04-amd64.deb (PGP Signature)
      - Console: veracrypt-console-1.24-Update7-Ubuntu-18.04-amd64.deb (PGP Signature)
    - Ubuntu 19.04:
      - GUI: veracrypt-1.24-Update7-Ubuntu-19.04-amd64.deb (PGP Signature)
      - Console: veracrypt-console-1.24-Update7-Ubuntu-19.04-amd64.deb (PGP Signature)
    - Ubuntu 19.10:
      - GUI: veracrypt-1.24-Update7-Ubuntu-19.10-amd64.deb (PGP Signature)
      - Console: veracrypt-console-1.24-Update7-Ubuntu-19.10-amd64.deb (PGP Signature)
    - Ubuntu 20.04:
      - GUI: veracrypt-1.24-Update7-Ubuntu-20.04-amd64.deb (PGP Signature)
      - Console: veracrypt-console-1.24-Update7-Ubuntu-20.04-amd64.deb (PGP Signature)
    - Ubuntu 20.10:
      - GUI: veracrypt-1.24-Update7-Ubuntu-20.10-amd64.deb (PGP Signature)
      - Console: veracrypt-console-1.24-Update7-Ubuntu-20.10-amd64.deb (PGP Signature)
  - RPM packages:
    - CentOS 6:

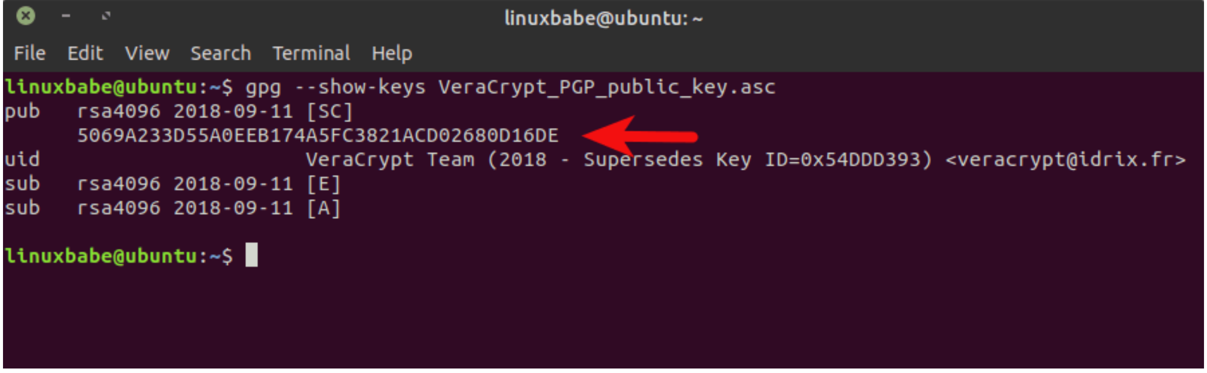
On the VeraCrypt download page, you can also find the **PGP public key** and **PGP signature** download link. Click the links to download these two files. You can run the following command to download PGP public key of VeraCrypt.

```
wget https://www.idrix.fr/VeraCrypt/VeraCrypt_PGP_public_key.asc
```

Before you do anything with the public key, you must always check the key’s fingerprint to see if it’s the correct key. Display the fingerprint of the key using the command below.

```
gpg --show-keys VeraCrypt_PGP_public_key.asc
```

The second line of the output is the key’s fingerprint.



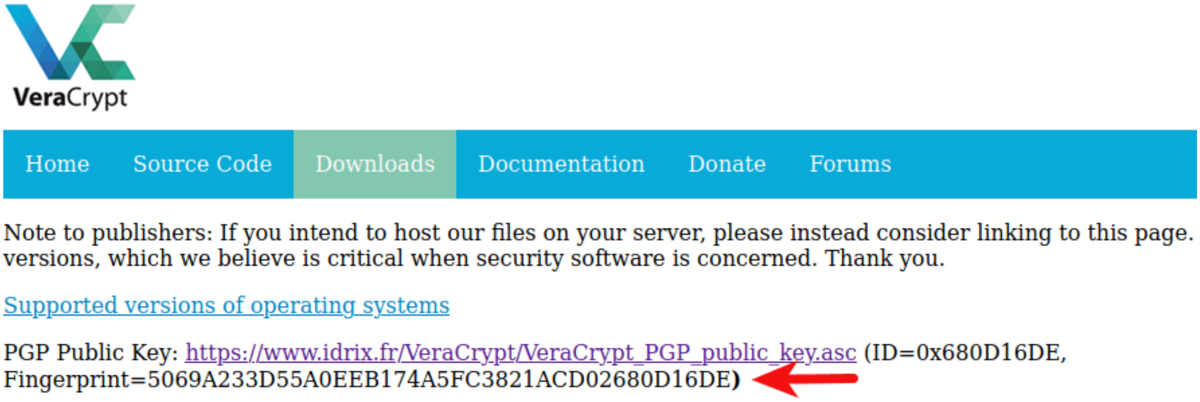
```
linuxbabe@ubuntu:~$ gpg --show-keys VeraCrypt_PGP_public_key.asc
pub   rsa4096 2018-09-11 [SC]
      5069A233D55A0EEB174A5FC3821ACD02680D16DE
uid                      VeraCrypt Team (2018 - Supersedes Key ID=0x54DDD393) <veracrypt@idrix.fr>
sub   rsa4096 2018-09-11 [E]
sub   rsa4096 2018-09-11 [A]

linuxbabe@ubuntu:~$ 
```

If you are using a very old version of GPG (`gpg --version`) like 1.4.20, then use the following command to display the fingerprint.

gpg --with-fingerprint VeraCrypt_PGP_public_key.asc

Compare it with the fingerprint published on VeraCrypt website.



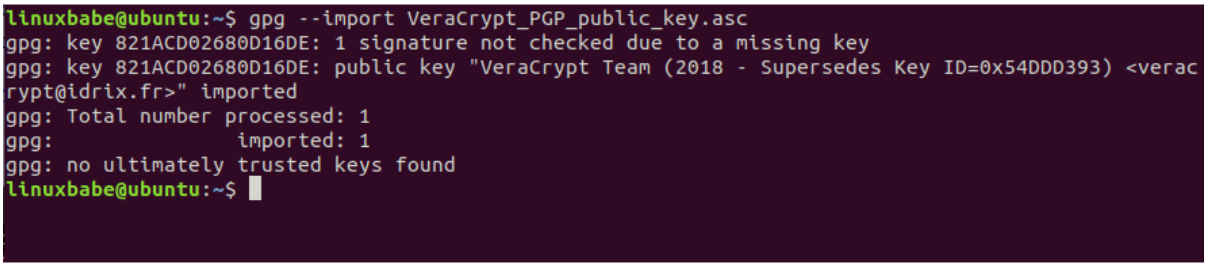
Note to publishers: If you intend to host our files on your server, please instead consider linking to this page. versions, which we believe is critical when security software is concerned. Thank you.

Supported versions of operating systems

PGP Public Key: https://www.idrix.fr/VeraCrypt/VeraCrypt_PGP_public_key.asc (ID=0x680D16DE, Fingerprint=5069A233D55A0EEB174A5FC3821ACD02680D16DE)

As you can see, the two fingerprints are identical, which means the public key is correct. So you can import the public key to your GPG public keyring with:

gpg --import VeraCrypt_PGP_public_key.asc

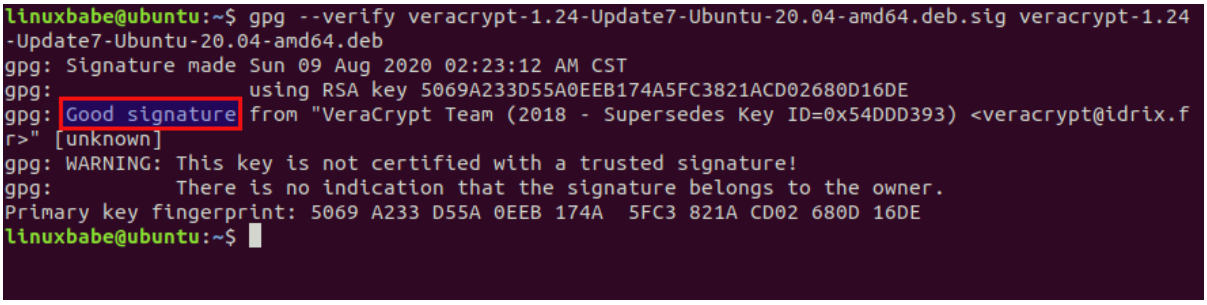


```
linuxbabe@ubuntu:~$ gpg --import VeraCrypt_PGP_public_key.asc
gpg: key 821ACD02680D16DE: 1 signature not checked due to a missing key
gpg: key 821ACD02680D16DE: public key "VeraCrypt Team (2018 - Supersedes Key ID=0x54DDD393) <verac
rypt@idrix.fr>" imported
gpg: Total number processed: 1
gpg:               imported: 1
gpg: no ultimately trusted keys found
linuxbabe@ubuntu:~$ 
```

Now verify the signature of the software installer file using the command below. You need to specify the **signature file** (`.sig`) and the **software installer file**. This is a detached signature, meaning that the signature and software are in separate files.

gpg --verify veracrypt-1.24-Update7-Ubuntu-20.04-amd64.deb.sig veracrypt-1.24-Update7-Ubuntu-20.04-amd64.deb

The output should say “Good Signature”.



```
linuxbabe@ubuntu:~$ gpg --verify veracrypt-1.24-Update7-Ubuntu-20.04-amd64.deb.sig veracrypt-1.24
-Update7-Ubuntu-20.04-amd64.deb
gpg: Signature made Sun 09 Aug 2020 02:23:12 AM CST
gpg:                using RSA key 5069A233D55A0EEB174A5FC3821ACD02680D16DE
gpg: Good signature from "VeraCrypt Team (2018 - Supersedes Key ID=0x54DDD393) <veracrypt@idrix.f
r>" [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: 5069 A233 D55A 0EEB 174A  5FC3 821A CD02 680D 16DE
linuxbabe@ubuntu:~$ 
```

How signature verifications works:

- The signature is a hash value, encrypted with the software author’s private key.
- GPG uses the public key to decrypt hash value.
- GPG calculates the hash value of VeraCrypt installer and compare the two.
- If these two hash values match, then the signature is good and the software wasn’t tampered with.

If GPG tells you it’s a bad signature, then the software installer was tampered with or corrupted.

Importing Public Key from a Trusted Source

Note that if the software author tells you his/her public key ID on the website, then you can import the public key with the following command, so you don’t have to manually download the PGP public key and import it to your keyring.

gpg --recv-keys <key-ID>

Then display the fingerprint with:

gpg --fingerprint <key-ID>

And compare the fingerprint from output with the one published on website. This is more secure because the public key is imported from a public key server, which by default is set to `hkp://keys.gnupg.net` in `~/.gnupg/gpg.conf` file. There’re hundreds of public keyservers around the world. Ubuntu has their own key server. MIT also has one.

If you see the following error,

```
gpg: keyserver receive failed: No data
```

then you can try a different key server, like this:

```
gpg --keyserver hkps://keyserver.ubuntu.com --recv-keys 0x680D16DE
```

That's it!

I hope this tutorial helped you verify PGP signature of software downloads. As always, if you found this post useful, then subscribe to our free newsletter or follow us on Twitter or like our Facebook page.