**1. Make sure that GnuPG is installed**

For Debian/Ubuntu:

```
sudo apt install gnupg
```

For Fedora:

```
sudo dnf install gnupg
```

For Arch:

```
sudo pacman -S gnupg
```

**2. Navigate to downloads folder**

```
cd downloads
```

If the folder doesn't exist, create it with `mkdir ~/Downloads`.)

**3. Download three files**

Download the correct release file for your machine (example for aarch64)

```
wget
https://bitcoinknots.org/files/29.x/29.1.knots20250903/bitcoin-29.1.kn
ots20250903-aarch64-linux-gnu.tar.gz
```

Download the checksum file (`SHA256SUMS`, a list of hashes for each release).

```
wget https://bitcoinknots.org/files/29.x/29.1.knots20250903/SHA256SUMS
```

Download the signature file (`SHA256SUMS.asc`, the signed version of the checksums).

```
wget
https://bitcoinknots.org/files/29.x/29.1.knots20250903/SHA256SUMS.asc
```

**4. Clone the Bitcoin Knots github guix signature repository**

```
git clone https://github.com/bitcoinknots/guix.sigs.git
```

**5. Import the developers signatures**

```
gpg --import guix.sigs/builder-keys/*
```

**6. Verify the authenticity of the list of checksum hashes**

```
gpg --verify SHA256SUMS.asc
```

You should see output showing developer emails and a line that says **GOOD SIGNATURE**. You'll also see the developer's public key fingerprint, which you should cross-check against trusted sources.

(Note: A warning about the key not being certified or trusted is normal unless you've set up your own web of trust. That's an advanced step for users who assign trust levels and sign keys themselves.)

**7. Verify the integrity of the release file**

```
sha256sum --check --ignore-missing SHA256SUMS
```

This checks that your downloaded release file matches the expected hash listed in
`SHA256SUMS` (you want to ignore all the other releases)
Look for **OK** next to your file.