# Nodo Bitcoin su ODROID-M1

**Parte 2**
Installazione e avvio sincronizzazione di Bitcoin Core

SatoshiSpritz Treviso
19 Novembre 2024

Dall’installazione del sistema operativo alla creazione di un nodo Bitcoin

# Download di Bitcoin core

# Dove trovare ultima versione

L’ultima versione di Bitcoin core, l’hash e le firme si trovano alla seguente pagina:

**https://bitcoincore.org/en/download/**

# Scarichiamo Bitcoin core

Eseguiamo **wget https://bitcoincore.org/bin/bitcoin-core-28.0/bitcoin-28.0-aarch64-linux-gnu.tar.gz** per scaricare il pacchetto di installazione di Bitcoin core.

# Scarichiamo la checksum

Eseguiamo **wget https://bitcoincore.org/bin/bitcoin-core-28.0/SHA256SUMS** per scaricare la checksum.

```
root@SSTreviso:/home/bitfede# wget https://bitcoincore.org/bin/bitcoin-core-28.0/SHA256SUMS
--2024-11-16 16:23:24--  https://bitcoincore.org/bin/bitcoin-core-28.0/SHA256SUMS
Resolving bitcoincore.org (bitcoincore.org)... 107.191.99.5, 198.251.83.116
Connecting to bitcoincore.org (bitcoincore.org)|107.191.99.5|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2620 (2.6K) [application/octet-stream]
Saving to: ‘SHA256SUMS’

SHA256SUMS                                          100%[=============================================================================================================>]   2.56K  --.-KB/s    in 0s

2024-11-16 16:23:25 (13.9 MB/s) - ‘SHA256SUMS’ saved [2620/2620]
```

# Scarichiamo le firme

Eseguiamo **wget https://bitcoincore.org/bin/bitcoin-core-28.0/SHA256SUMS.asc** per scaricare le firme.

```
root@SSTreviso:/home/bitfede# wget https://bitcoincore.org/bin/bitcoin-core-28.0/SHA256SUMS.asc
--2024-11-16 16:24:00--  https://bitcoincore.org/bin/bitcoin-core-28.0/SHA256SUMS.asc
Resolving bitcoincore.org (bitcoincore.org)... 198.251.83.116, 107.191.99.5
Connecting to bitcoincore.org (bitcoincore.org)|198.251.83.116|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 9109 (8.9K) [application/octet-stream]
Saving to: ‘SHA256SUMS.asc’

SHA256SUMS.asc                                  100%[=============================================================================================================>]   8.90K  --.-KB/s    in 0s

2024-11-16 16:24:00 (43.9 MB/s) - ‘SHA256SUMS.asc’ saved [9109/9109]
```

# Verifichiamo la checksum

Eseguiamo **sha256sum --ignore-missing --check SHA256SUMS** per verificare che la checksum del file scaricato corrisponda a quello elencato nel file SHA256SUMS.

Nell'output prodotto dal comando di cui sopra, assicuriamoci che venga indicato “OK” dopo il nome del file di release scaricato. Ovvero:
**bitcoin-28.0-aarch64-linux-gnu.tar.gz: OK**

```
root@SSTreviso:/home/bitfede# sha256sum --ignore-missing --check SHA256SUMS
bitcoin-28.0-aarch64-linux-gnu.tar.gz: OK
root@SSTreviso:/home/bitfede#
```

# Otteniamo le chiavi pubbliche degli sviluppatori

Si importano le chiavi pubbliche degli sviluppatori Bitcoin Core utilizzando GPG. Queste chiavi sono disponibili nel repository bitcoin-core/guix.sigs

Cloniamo il repository eseguendo il seguente comando: **git clone** https://github.com/bitcoin-core/guix.sigs

Probabilmente non avremo git installato, perciò prima installiamolo con il seguente comando: **apt-get install git**

```
root@SSTreviso:/home/bitfede# git clone https://github.com/bitcoin-core/guix.sigs
Cloning into 'guix.sigs'...
remote: Enumerating objects: 10202, done.
remote: Counting objects: 100% (770/770), done.
remote: Compressing objects: 100% (77/77), done.
remote: Total 10202 (delta 700), reused 700 (delta 693), pack-reused 9432 (from 1)
Receiving objects: 100% (10202/10202), 3.57 MiB | 2.03 MiB/s, done.
Resolving deltas: 100% (4633/4633), done.
root@SSTreviso:/home/bitfede#
```

# Importiamo le chiavi pubbliche degli sviluppatori

Importiamo le chiavi pubbliche degli sviluppatori eseguendo questo comando: **gpg --import guix.sigs/builder-keys/***

```
root@SSTreviso:/home/bitfede# gpg --import guix.sigs/builder-keys/*
gpg: key 188CBB2648416AD5: 6 signatures not checked due to missing keys
gpg: /root/.gnupg/trustdb.gpg: trustdb created
gpg: key 188CBB2648416AD5: public key ".0xB10C <b10c@b10c.me>" imported
gpg: key 17565732E08E5E41: 3 signatures not checked due to missing keys
gpg: key 17565732E08E5E41: public key "Ava Chow <me@achow101.com>" imported
gpg: key D7CC770B81FD22A8: 2 signatures not checked due to missing keys
gpg: key D7CC770B81FD22A8: public key "Ben Carman <benthecarman@live.com>" imported
gpg: key 1C2491FFEB0EF770: 2 signatures not checked due to missing keys
gpg: key 1C2491FFEB0EF770: public key "Cory Fields <cfields@bitcoinfoundation.org>" imported
gpg: key A5E0907A0380E6C3: public key "CoinForensics (SigningKey) <59567284+coinforensics@users.noreply.github.com>" imported
gpg: key E13FC145CD3F4304: 15 signatures not checked due to missing keys
gpg: key E13FC145CD3F4304: public key "Antoine Poinsot <darosior@protonmail.com>" imported
gpg: key 3B6305FA06DE51D5: public key "David Gumberg <davidzgumberg@gmail.com>" imported
gpg: key C37B1C1D44C786EE: public key "Duncan Dean <duncangleeddean@gmail.com>" imported
gpg: key 2EBB056FD847F8A7: 12 signatures not checked due to missing keys
gpg: key 2EBB056FD847F8A7: public key "Stephan Oeste (it) <it@oeste.de>" imported
gpg: key 944D35F9AC3DB76A: 18 signatures not checked due to missing keys
gpg: key 944D35F9AC3DB76A: public key "Michael Ford (bitcoin-otc) <fanquake@gmail.com>" imported
gpg: key 8F617F1200A6D25C: 8 signatures not checked due to missing keys
gpg: key 8F617F1200A6D25C: public key "Gloria Zhao <gloriazhao@berkeley.edu>" imported
gpg: key 8E4256593F177720: 1 signature not checked due to a missing key
gpg: key 8E4256593F177720: public key "Oliver Gugger <gugger@gmail.com>" imported
gpg: key 410108112E7EA81F: public key "Hennadii Stepanov (GitHub key) <32963518+hebasto@users.noreply.github.com>" imported
gpg: key D11BD4F33F1DB499: public key "jackielove4u <jackielove4u@hotmail.com>" imported
gpg: key 8ADCB558C4F33D65: public key "josibake@protonmail.com <josibake@protonmail.com>" imported
gpg: key F62711DBDCA8AE56: public key "Dimitri <kvaciral@protonmail.com>" imported
gpg: key 74810B0123346C9A6: 104 signatures not checked due to missing keys
gpg: key 74810B0123346C9A6: public key "Wladimir J. van der Laan <laanwj@protonmail.com>" imported
gpg: key A291A2C45D0C504A: public key "Luke Dashjr (Codesigning) <luke-jr+git@utopios.org>" imported
gpg: key B66D427F873CB1A3: public key "m3dwards <me@maxedwards.me>" imported
gpg: key E7E2984B6289C93A: 1 signature not checked due to a missing key
gpg: key E7E2984B6289C93A: public key "Matthew Zipkin (GitHub Signing Key) <pinheadmz@gmail.com>" imported
gpg: key 747A7AE2FB0FD25B: public key "satsie <staciewaleyko@gmail.com>" imported
gpg: key 860FEB804E669320: 61 signatures not checked due to missing keys
gpg: key 860FEB804E669320: public key "Pieter Wuille <pieter@wuille.net>" imported
gpg: key 57FF9BDBCC301009: 42 signatures not checked due to missing keys
gpg: key 57FF9BDBCC301009: public key "Sjors Provoost <sjors@sprovoost.nl>" imported
gpg: key 476E74C8529A9006: public key "Sebastian van Staa <sebastian.van.staa@gmail.com>" imported
gpg: key 9303B33A305224CB: 15 signatures not checked due to missing keys
gpg: key 9303B33A305224CB: public key "Sebastian Kung (TheCharlatan) <seb.kung@gmail.com>" imported
gpg: key C2371D91CB716EA7: public key "Sebastian Falbesoner (theStack) <sebastian.falbesoner@gmail.com>" imported
gpg: key A7BEB2621678D37D: public key "vertiond <vertion@protonmail.com>" imported
gpg: key 3B8F814A784218F8: 1 signature not checked due to a missing key
gpg: key 3B8F814A784218F8: public key "Will Clark <will@256k1.dev>" imported
gpg: key 8E3A8F3247DBCBBF: public key "Willy Ko <willyk@syscoin.org>" imported
gpg: Total number processed: 29
gpg:               imported: 29
gpg: no ultimately trusted keys found
```

# Verifichiamo le firme

Verifichiamo le firme del file SHA256SUMS.asc eseguendo: **gpg --verify SHA256SUMS.asc**

Per ogni sviluppatore dovrebbe comparire la scritta **gpg: Good signature from "...."** a significare che la firma è valida. (Possiamo ignorare gli eventuali warning.)

```
root@SSTreviso:/home/bitfede# gpg --verify SHA256SUMS.asc
gpg: assuming signed data in 'SHA256SUMS'
gpg: Signature made Thu 03 Oct 2024 18:25:02 CEST
gpg:                using RSA key 101598DC823C1B5F9A6624ABA5E0907A0380E6C3
gpg: Good signature from "CoinForensics (SigningKey) <59567284+coinforensics@users.noreply.github.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: 1015 98DC 823C 1B5F 9A66  24AB A5E0 907A 0380 E6C3
gpg: Signature made Thu 03 Oct 2024 23:36:16 CEST
gpg:                using RSA key 6A8F9C266528E25AEB1D7731C2371D91CB716EA7
gpg:                issuer "sebastian.falbesoner@gmail.com"
```

# Importanza della procedura

Questa procedura serve a garantire due aspetti fondamentali

**Integrità**: La verifica della checksum assicura che il file scaricato non sia stato corrotto o modificato durante il download.

**Autenticità**: La verifica delle firme conferma che il file proviene effettivamente dagli sviluppatori di Bitcoin Core e non da una fonte malevola.

# Installazione di Bitcoin core

# Scompattiamo il pacchetto scaricato

Eseguiamo **tar xzvf bitcoin-28.0-aarch64-linux-gnu.tar.gz** per scompattare il pacchetto scaricato.

```
root@SSTreviso:/home/bitfede# tar xzvf bitcoin-28.0-aarch64-linux-gnu.tar.gz
bitcoin-28.0/
bitcoin-28.0/README.md
bitcoin-28.0/bin/
bitcoin-28.0/bin/bitcoin-cli
bitcoin-28.0/bin/bitcoin-qt
bitcoin-28.0/bin/bitcoin-tx
bitcoin-28.0/bin/bitcoin-util
bitcoin-28.0/bin/bitcoin-wallet
bitcoin-28.0/bin/bitcoind
bitcoin-28.0/bin/test_bitcoin
bitcoin-28.0/bitcoin.conf
bitcoin-28.0/share/
bitcoin-28.0/share/man/
bitcoin-28.0/share/man/man1/
bitcoin-28.0/share/man/man1/bitcoin-cli.1
bitcoin-28.0/share/man/man1/bitcoin-qt.1
bitcoin-28.0/share/man/man1/bitcoin-tx.1
bitcoin-28.0/share/man/man1/bitcoin-util.1
bitcoin-28.0/share/man/man1/bitcoin-wallet.1
bitcoin-28.0/share/man/man1/bitcoind.1
bitcoin-28.0/share/rpcauth/
bitcoin-28.0/share/rpcauth/README.md
bitcoin-28.0/share/rpcauth/rpcauth.py
```

# Installiamo Bitcoin core

Eseguiamo **install -m 0755 -o root -g root -t /usr/local/bin bitcoin-28.0/bin/*** per installare Bitcoin core.

# Avviamo la sincronizzazione

Eseguiamo **bitcoind** per avviare la sincronizzazione.

```
root@SSTreviso:/home/bitfede# bitcoind
2024-11-16T15:46:36Z Bitcoin Core version v28.0.0 (release build)
2024-11-16T15:46:36Z Script verification uses 3 additional threads
2024-11-16T15:46:36Z Using the 'arm_shani(1way,2way)' SHA256 implementation
2024-11-16T15:46:36Z Default data directory /root/.bitcoin
2024-11-16T15:46:36Z Using data directory /root/.bitcoin
2024-11-16T15:46:36Z Config file: /root/.bitcoin/bitcoin.conf (not found, sk
ipping)
2024-11-16T15:46:36Z Using at most 125 automatic connections (1024 file desc
riptors available)
2024-11-16T15:46:36Z scheduler thread start
2024-11-16T15:46:36Z Binding RPC on address ::1 port 8332
2024-11-16T15:46:36Z Binding RPC on address 127.0.0.1 port 8332
2024-11-16T15:46:36Z Using random cookie authentication.
2024-11-16T15:46:36Z Generated RPC authentication cookie /root/.bitcoin/.coo
kie
2024-11-16T15:46:36Z Permissions used for cookie: rw-------
2024-11-16T15:46:36Z Starting HTTP server with 4 worker threads
2024-11-16T15:46:36Z Using wallet directory /root/.bitcoin/wallets
2024-11-16T15:46:36Z init message: Verifying wallet(s)…
2024-11-16T15:46:36Z Using /16 prefix for IP bucketing
2024-11-16T15:46:36Z init message: Loading P2P addresses…
2024-11-16T15:46:36Z Creating peers.dat because the file was not found ("/ro
ot/.bitcoin/peers.dat")
2024-11-16T15:46:36Z init message: Loading banlist…
2024-11-16T15:46:36Z Recreating the banlist database
2024-11-16T15:46:36Z SetNetworkActive: true
```

```
2024-11-16T15:46:37Z Loading addresses from DNS seed dnsseed.bluematt.me.
2024-11-16T15:46:37Z Loading addresses from DNS seed seed.bitcoin.jonasschne
lli.ch.
2024-11-16T15:46:37Z Loading addresses from DNS seed dnsseed.bitcoin.dashjr-
list-of-p2p-nodes.us.
2024-11-16T15:46:38Z Loading addresses from DNS seed dnsseed.emzy.de.
2024-11-16T15:46:40Z Loading addresses from DNS seed seed.bitcoin.sipa.be.
2024-11-16T15:46:40Z Loading addresses from DNS seed seed.mainnet.achownodes
.xyz.
2024-11-16T15:46:41Z Loading addresses from DNS seed seed.btc.petertodd.net.
2024-11-16T15:46:41Z Loading addresses from DNS seed seed.bitcoin.wiz.biz.
2024-11-16T15:46:41Z New outbound-full-relay v1 peer connected: version: 700
16, blocks=870579, peer=0
2024-11-16T15:46:42Z 254 addresses found from DNS seeds
2024-11-16T15:46:42Z dnsseed thread exit
2024-11-16T15:46:45Z Pre-synchronizing blockheaders, height: 2000 (~0.24%)
2024-11-16T15:46:47Z Pre-synchronizing blockheaders, height: 4000 (~0.48%)
2024-11-16T15:46:48Z New outbound-full-relay v1 peer connected: version: 700
16, blocks=870579, peer=1
2024-11-16T15:46:52Z Pre-synchronizing blockheaders, height: 6000 (~0.72%)
2024-11-16T15:46:55Z New outbound-full-relay v1 peer connected: version: 700
16, blocks=870579, peer=2
2024-11-16T15:46:59Z Pre-synchronizing blockheaders, height: 8000 (~0.96%)
2024-11-16T15:47:03Z New outbound-full-relay v1 peer connected: version: 700
16, blocks=870579, peer=3
```

# Configurazione casalinga

# Modifichiamo il file bitcoin.conf

Eseguiamo **nano ~/.bitcoin/bitcoin.conf** per aprire il file bitcoin.conf

Aggiungiamo le seguenti righe:
daemon=1
blocksonly=1
maxconnections=20
maxuploadtarget=500
txindex=1
blockfilterindex=1

Premiamo **CTRL+x** per uscire, poi **y** per salvare e **invio** per sovrascrivere il file.

# Spiegazione

**daemon=1**

Fa sì che bitcoind venga eseguito come un daemon in background ovvero  che il processo continuerà a funzionare anche dopo aver chiuso il terminale.

**blocksonly=1**

Fa sì che vengano scaricati solo i blocchi e non la mempool (transazioni non confermate)

**maxconnections=20**

Limita il numero massimo di connessioni in entrata e in uscita a 20

**maxuploadtarget=500**

Imposta un limite di upload giornaliero di 500 MB.

# Spiegazione

**txindex=1**

Mantiene un indice completo delle transazioni (esplorazione completa della blockchain)

**blockfilterindex=1**

Crea e mantiene un indice dei filtri di blocco (permette una verifica più efficiente delle transazioni)

# Sitografia

- https://bitcoincore.org/en/download/
- https://officinebitcoin.it/lezioni/fulhar/
- https://git-scm.com/downloads/linux
- https://jlopp.github.io/bitcoin-core-config-generator/