

**De:** Alejandro Martínez Martínez  
**Enviado:** viernes, 25 de marzo de 2022 17:03  
**Para:** Alejandro Martínez  
**Asunto:** bb

Bitcoin Core version v22.0.0  
Usage: bitcoin-qt [command-line options]

**Options:**

-?	Print this help message and exit
-alertnotify=<cmd>	Execute command when a relevant alert is received or we see a really long fork (%s in cmd is replaced by message)
-assumevalid=<hex>	If this block is in the chain assume that it and its ancestors are valid and potentially skip their script verification (0 to verify all, default: 00, testnet: 00, signet: 0000000187d4440e5bff91488b700a140441e089a8aaea707414982460edbf54)
-blockfilterindex=<type>	Maintain an index of compact filters by block (default: 0, values: basic). If <type> is not supplied or if <type> = 1, indexes for all known types are enabled.
-blocknotify=<cmd>	Execute command when the best block changes (%s in cmd is replaced by block hash)
-blockreconstructionextratxn=<n>	Extra transactions to keep in memory for compact block reconstructions (default: 100)
-blocksdir=<dir>	Specify directory to hold blocks subdirectory for *.dat files (default: <datadir>)
-blocksonly	Whether to reject transactions from network peers. Automatic broadcast and rebroadcast of any transactions from inbound peers is disabled, unless the peer has the 'forcerelay' permission. RPC transactions are not affected. (default: 0)
-coinstatsindex	Maintain coinstats index used by the gettxoutsetinfo RPC (default: 0)
-conf=<file>	Specify path to read-only configuration file. Relative paths will be prefixed by datadir location. (default: bitcoin.conf)
-datadir=<dir>	Specify data directory
-dbcache=<n>	Maximum database cache size <n> MiB (4 to 16384, default: 450). In addition, unused mempool memory is shared for this cache (see -maxmempool).
-debuglogfile=<file>	Specify location of debug log file. Relative paths will be prefixed by a net-specific datadir location. (-nodebuglogfile to disable; default: debug.log)

-includeconf=<file>	Specify additional configuration file, relative to the -datadir path (only useable from configuration file, not command line)
-loadblock=<file>	Imports blocks from external file on startup
-maxmempool=<n>	Keep the transaction memory pool below <n> megabytes (default: 300)
-maxorphantx=<n>	Keep at most <n> unconnectable transactions in memory (default: 100)
-mempoolexpiry=<n>	Do not keep transactions in the mempool longer than <n> hours (default: 336)
-par=<n>	Set the number of script verification threads (-8 to 15, 0 = auto, <0 = leave that many cores free, default: 0)
-persistmempool	Whether to save the mempool on shutdown and load on restart (default: 1)
-pid=<file>	Specify pid file. Relative paths will be prefixed by a net-specific datadir location. (default: bitcoind.pid)
-prune=<n>	Reduce storage requirements by enabling pruning (deleting) of old blocks. This allows the pruneblockchain RPC to be called to delete specific blocks, and enables automatic pruning of old blocks if a target size in MiB is provided. This mode is incompatible with -txindex, -coinstatsindex and -rescan. Warning: Reverting this setting requires re-downloading the entire blockchain. (default: 0 = disable pruning blocks, 1 = allow manual pruning via RPC, >=550 = automatically prune block files to stay under the specified target size in MiB)
-reindex	Rebuild chain state and block index from the blk*.dat files on disk
-reindex-chainstate	Rebuild chain state from the currently indexed blocks. When in pruning mode or if blocks on disk might be corrupted, use full -reindex instead.
-settings=<file>	Specify path to dynamic settings data file. Can be disabled with -nosettings. File is written at runtime and not meant to be edited by users (use bitcoin.conf instead for custom settings). Relative paths will be prefixed by datadir location. (default: settings.json)
-startupnotify=<cmd>	Execute command on startup.
-txindex	Maintain a full transaction index, used by the getrawtransaction rpc call (default: 0)
-version	Print version and exit

-addnode=<ip>	Add a node to connect to and attempt to keep the connection open (see the addnode RPC help for more info). This option can be specified multiple times to add multiple nodes; connections are limited to 8 at a time and are counted separately from the -maxconnections limit.
-asmap=<file>	Specify asn mapping used for bucketing of the peers (default: ip_asn.map). Relative paths will be prefixed by the net-specific datadir location.
-bantime=<n>	Default duration (in seconds) of manually configured bans (default: 86400)

-bind=<addr>[:<port>][=onion]	Bind to given address and always listen on it (default: 0.0.0.0). Use [host]:port notation for IPv6. Append =onion to tag any incoming connections to that address and port as incoming Tor connections (default: 127.0.0.1:8334=onion, testnet: 127.0.0.1:18334=onion, signet: 127.0.0.1:38334=onion, regtest: 127.0.0.1:18445=onion)
-connect=<ip>	Connect only to the specified node; -noconnect disables automatic connections (the rules for this peer are the same as for -addnode). This option can be specified multiple times to connect to multiple nodes.
-discover	Discover own IP addresses (default: 1 when listening and no -externalip or -proxy)
-dns	Allow DNS lookups for -addnode, -seednode and -connect (default: 1)
-dnsseed	Query for peer addresses via DNS lookup, if low on addresses (default: 1 unless -connect used)
-externalip=<ip>	Specify your own public address
-fixedseeds	Allow fixed seeds if DNS seeds don't provide peers (default: 1)
-forcednsseed	Always query for peer addresses via DNS lookup (default: 0)
-i2pacceptincoming	If set and -i2psam is also set then incoming I2P connections are accepted via the SAM proxy. If this is not set but -i2psam is set then only outgoing connections will be made to the I2P network. Ignored if -i2psam is not set. Listening for incoming I2P connections is done through the SAM proxy, not by binding to a local address and port (default: 1)
-i2psam=<ip:port>	I2P SAM proxy to reach I2P peers and accept I2P connections (default: none)
-listen	Accept connections from outside (default: 1 if no -proxy or -connect)
-listenonion	Automatically create Tor onion service (default: 1)
-maxconnections=<n>	Maintain at most <n> connections to peers (default: 125). This limit does not apply to connections manually added via -addnode or the addnode RPC, which have a separate limit of 8.
-maxreceivebuffer=<n>	Maximum per-connection receive buffer, <n>*1000 bytes (default: 5000)
-maxsendbuffer=<n>	Maximum per-connection send buffer, <n>*1000 bytes (default: 1000)
-maxtimeadjustment	Maximum allowed median peer time offset adjustment. Local perspective of time may be influenced by peers

	forward or backward by this amount. (default: 4200 seconds)
-maxuploadtarget=<n>	Tries to keep outbound traffic under the given target (in MiB per 24h). Limit does not apply to peers with 'download' permission. 0 = no limit (default: 0)
-natpmp	Use NAT-PMP to map the listening port (default: 0)
-networkactive	Enable all P2P network activity (default: 1). Can be changed by the setnetworkactive RPC command
-onion=<ip:port>	Use separate SOCKS5 proxy to reach peers via Tor onion services, set -noonion to disable (default: -proxy)
-onlynet=<net>	Make outgoing connections only through network <net> (ipv4, ipv6, onion, i2p). Incoming connections are not affected by this option. This option can be specified multiple times to allow multiple networks. Warning: if it is used with non-onion networks and the -onion or -proxy option is set, then outbound onion connections will still be made; use -noonion or -onion=0 to disable outbound onion connections in this case.
-peerblockfilters	Serve compact block filters to peers per BIP 157 (default: 0)
-peerbloomfilters	Support filtering of blocks and transaction with bloom filters (default: 0)
-permitbaremultisig	Relay non-P2SH multisig (default: 1)
-port=<port>	Listen for connections on <port>. Nodes not using the default ports (default: 8333, testnet: 18333, signet: 38333, regtest: 18444) are unlikely to get incoming connections. Not relevant for I2P (see doc/i2p.md).
-proxy=<ip:port>	Connect through SOCKS5 proxy, set -noproxy to disable (default: disabled)
-proxyrandomize	Randomize credentials for every proxy connection. This enables Tor stream isolation (default: 1)
-seednode=<ip>	Connect to a node to retrieve peer addresses, and disconnect. This option can be specified multiple times to connect to multiple nodes.
-timeout=<n>	Specify socket connection timeout in milliseconds. If an initial attempt to connect is unsuccessful after this amount of time, drop it (minimum: 1, default: 5000)
-torcontrol=<ip>:<port>	Tor control port to use if onion listening enabled (default: 127.0.0.1:9051)
-torpassword=<pass>	Tor control port password (default: empty)
-upnp	Use UPnP to map the listening port (default: 0)
-whitebind=<[permissions@]addr>	Bind to the given address and add permission flags to the peers connecting to it. Use [host]:port notation for IPv6.

	Allowed permissions: bloomfilter (allow requesting BIP37 filtered blocks and transactions), noban (do not ban for misbehavior; implies download), forcerelay (relay transactions that are already in the mempool; implies relay), relay (relay even in -blocksonly mode, and unlimited transaction announcements), mempool (allow requesting BIP35 mempool contents), download (allow getheaders during IBD, no disconnect after maxuploadtarget limit), addr (responses to GETADDR avoid hitting the cache and contain random records with the most up-to-date info). Specify multiple permissions separated by commas (default: download,noban,mempool,relay). Can be specified multiple times.
-whitelist=<[permissions@]IP address or network>	Add permission flags to the peers connecting from the given IP address (e.g. 1.2.3.4) or CIDR-notated network (e.g. 1.2.3.0/24). Uses the same permissions as -whitebind. Can be specified multiple times.

-addresstype	What type of addresses to use ("legacy", "p2sh-segwit", or "bech32", default: "bech32")
-avoidpartialspends	Group outputs by address, selecting many (possibly all) or none, instead of selecting on a per-output basis. Privacy is improved as addresses are mostly swept with fewer transactions and outputs are aggregated in clean change addresses. It may result in higher fees due to less optimal coin selection caused by this added limitation and possibly a larger-than-necessary number of inputs being used. Always enabled for wallets with "avoid_reuse" enabled, otherwise default: 0.
-changetype	What type of change to use ("legacy", "p2sh-segwit", or "bech32"). Default is same as -addresstype, except when -addresstype=p2sh-segwit a native segwit output is used when sending to a native segwit address)
-disablewallet	Do not load the wallet and disable wallet RPC calls
-discardfee=<amt>	The fee rate (in BTC/kvB) that indicates your tolerance for discarding change by adding it to the fee (default: 0.0001). Note: An output is discarded if it is dust at this rate, but we will always discard up to the dust relay fee and a discard fee above that is limited by the fee estimate for the longest target
-fallbackfee=<amt>	A fee rate (in BTC/kvB) that will be used when fee estimation has insufficient data. 0 to entirely disable the fallbackfee feature. (default: 0.00)
-keypool=<n>	Set key pool size to <n> (default: 1000). Warning: Smaller sizes may increase the risk of losing funds when restoring

	from an old backup, if none of the addresses in the original keypool have been used.
-maxapsfee=<n>	Spend up to this amount in additional (absolute) fees (in BTC) if it allows the use of partial spend avoidance (default: 0.00)
-mintxfee=<amt>	Fee rates (in BTC/kvB) smaller than this are considered zero fee for transaction creation (default: 0.00001)
-paytxfee=<amt>	Fee rate (in BTC/kvB) to add to transactions you send (default: 0.00)
-rescan	Rescan the block chain for missing wallet transactions on startup
-signer=<cmd>	External signing tool, see doc/external-signer.md
-spendzeroconfchange	Spend unconfirmed change when sending transactions (default: 1)
-txconfirmtarget=<n>	If paytxfee is not set, include enough fee so transactions begin confirmation on average within n blocks (default: 6)
-wallet=<path>	Specify wallet path to load at startup. Can be used multiple times to load multiple wallets. Path is to a directory containing wallet data and log files. If the path is not absolute, it is interpreted relative to <walletdir>. This only loads existing wallets and does not create new ones. For backwards compatibility this also accepts names of existing top-level data files in <walletdir>.
-walletbroadcast	Make the wallet broadcast transactions (default: 1)
-walletdir=<dir>	Specify directory to hold wallets (default: <datadir>/wallets if it exists, otherwise <datadir>)
-walletnotify=<cmd>	Execute command when a wallet transaction changes. %s in cmd is replaced by TxID, %w is replaced by wallet name, %b is replaced by the hash of the block including the transaction (set to 'unconfirmed' if the transaction is not included) and %h is replaced by the block height (-1 if not included). %w is not currently implemented on windows. On systems where %w is supported, it should NOT be quoted because this would break shell escaping used to invoke the command.
-walletrbf	Send transactions with full-RBF opt-in enabled (RPC only, default: 0)

-zmqpubhashblock=<address>	Enable publish hash block in <address>
-zmqpubhashblockhwm=<n>	Set publish hash block outbound message high water mark (default: 1000)
-zmqpubhashtx=<address>	Enable publish hash transaction in <address>

-zmqpubhashtxhwm=<n>	Set publish hash transaction outbound message high water mark (default: 1000)
-zmqpubrawblock=<address>	Enable publish raw block in <address>
-zmqpubrawblockhwm=<n>	Set publish raw block outbound message high water mark (default: 1000)
-zmqpubrawtx=<address>	Enable publish raw transaction in <address>
-zmqpubrawtxhwm=<n>	Set publish raw transaction outbound message high water mark (default: 1000)
-zmqpubsequence=<address>	Enable publish hash block and tx sequence in <address>
-zmqpubsequencehwm=<n>	Set publish hash sequence message high water mark (default: 1000)

-debug=<category>	Output debugging information (default: -nodebug, supplying <category> is optional). If <category> is not supplied or if <category> = 1, output all debugging information. <category> can be: net, tor, mempool, http, bench, zmq, walletdb, rpc, estimatefee, addrman, selectcoins, reindex, cmpctblock, rand, prune, proxy, mempoolrej, libevent, coindb, qt, leveldb, validation, i2p, ipc. This option can be specified multiple times to output multiple categories.
-debugexclude=<category>	Exclude debugging information for a category. Can be used in conjunction with -debug=1 to output debug logs for all categories except the specified category. This option can be specified multiple times to exclude multiple categories.
-help-debug	Print help message with debugging options and exit
-logips	Include IP addresses in debug output (default: 0)
-logsourcelocations	Prepend debug output with name of the originating source location (source file, line number and function name) (default: 0)
-logtimestamps	Prepend debug output with timestamp (default: 1)
-maxtxfee=<amt>	Maximum total fees (in BTC) to use in a single wallet transaction; setting this too low may abort large transactions (default: 0.10)
-printtoconsole	Send trace/debug info to console (default: 1 when no -daemon. To disable logging to file, set -nodebuglogfile)
-shrinkdebugfile	Shrink debug.log file on client startup (default: 1 when no -debug)
-uacomment=<cmt>	Append comment to the user agent string

-chain=<chain>	Use the chain <chain> (default: main). Allowed values: main, test, signet, regtest
-signet	Use the signet chain. Equivalent to -chain=signet. Note that the network is defined by the -signetchallenge parameter
-signetchallenge	Blocks must satisfy the given script to be considered valid (only for signet networks; defaults to the global default signet test network challenge)
-signetseednode	Specify a seed node for the signet network, in the hostname[:port] format, e.g. sig.net:1234 (may be used multiple times to specify multiple seed nodes; defaults to the global default signet test network seed node(s))
-testnet	Use the test chain. Equivalent to -chain=test.

-bytespersigop	Equivalent bytes per sigop in transactions for relay and mining (default: 20)
-datacarrier	Relay and mine data carrier transactions (default: 1)
-datacarriersize	Maximum size of data in data carrier transactions we relay and mine (default: 83)
-minrelaytxfee=<amt>	Fees (in BTC/kvB) smaller than this are considered zero fee for relaying, mining and transaction creation (default: 0.00001)
-whitelistforcerelay	Add 'forcerelay' permission to whitelisted inbound peers with default permissions. This will relay transactions even if the transactions were already in the mempool. (default: 0)
-whitelistrelay	Add 'relay' permission to whitelisted inbound peers with default permissions. This will accept relayed transactions even when not relaying transactions (default: 1)

-blockmaxweight=<n>	Set maximum BIP141 block weight (default: 3996000)
-blockmintxfee=<amt>	Set lowest fee rate (in BTC/kvB) for transactions to be included in block creation. (default: 0.00001)

-rest	Accept public REST requests (default: 0)
-rpccallowip=<ip>	Allow JSON-RPC connections from specified source. Valid for <ip> are a single IP (e.g. 1.2.3.4), a network/netmask (e.g. 1.2.3.4/255.255.255.0) or a network/CIDR (e.g. 1.2.3.4/24). This option can be specified multiple times
-rpcauth=<userpw>	Username and HMAC-SHA-256 hashed password for JSON-RPC connections. The field <userpw> comes in the format: <USERNAME>:<SALT>\$<HASH>. A canonical python script is included in share/rpcauth. The client then connects



	normally using the rpcuser=<USERNAME>/rpcpassword=<PASSWORD> pair of arguments. This option can be specified multiple times
-rpcbind=<addr>[:port]	Bind to given address to listen for JSON-RPC connections. Do not expose the RPC server to untrusted networks such as the public internet! This option is ignored unless -rpcallowip is also passed. Port is optional and overrides -rpcport. Use [host]:port notation for IPv6. This option can be specified multiple times (default: 127.0.0.1 and ::1 i.e., localhost)
-rpccookiefile=<loc>	Location of the auth cookie. Relative paths will be prefixed by a net-specific datadir location. (default: data dir)
-rpcpassword=<pw>	Password for JSON-RPC connections
-rpcport=<port>	Listen for JSON-RPC connections on <port> (default: 8332, testnet: 18332, signet: 38332, regtest: 18443)
-rpcserialversion	Sets the serialization of raw transaction or block hex returned in non-verbose mode, non-segwit(0) or segwit(1) (default: 1)
-rpcthreads=<n>	Set the number of threads to service RPC calls (default: 4)
-rpcuser=<user>	Username for JSON-RPC connections
-rpcwhitelist=<whitelist>	Set a whitelist to filter incoming RPC calls for a specific user. The field <whitelist> comes in the format: <USERNAME>:<rpc 1>,<rpc 2>,...,<rpc n>. If multiple whitelists are set for a given user, they are set-intersected. See -rpcwhitelistdefault documentation for information on default whitelist behavior.
-rpcwhitelistdefault	Sets default behavior for rpc whitelisting. Unless rpcwhitelistdefault is set to 0, if any -rpcwhitelist is set, the rpc server acts as if all rpc users are subject to empty-unless-otherwise-specified whitelists. If rpcwhitelistdefault is set to 1 and no -rpcwhitelist is set, rpc server acts as if all rpc users are subject to empty whitelists.
-server	Accept command line and JSON-RPC commands
-choosedatadir	Choose data directory on startup (default: 0)
-lang=<lang>	Set language, for example "de_DE" (default: system locale)
-min	Start minimized
-resetguisettings	Reset all settings changed in the GUI
-splash	Show splash screen on startup (default: 1)

