|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **`Command** | **Parameters** | | **Description** | | **Requires unlocked wallet? (v0.4.0+)** | |
| addmultisigaddress | <nrequired> <'["key","key"]'> [account] | | Add a nrequired-to-sign multisignature address to the wallet. Each key is a bitcoin address or hex-encoded public key. If [account] is specified, assign address to [account]. | | N | |
| addnode | <node> <add/remove/onetry> | | **version 0.8** Attempts add or remove <node> from the addnode list or try a connection to <node> once. | | N | |
| backupwallet | <destination> | | Safely copies wallet.dat to destination, which can be a directory or a path with filename. | | N | |
| createmultisig | <nrequired> <'["key,"key"]'> | | Creates a multi-signature address and returns a json object | |  | |
| createrawtransaction | [{"txid":txid,"vout":n},...] {address:amount,...} | | **version 0.7** Creates a [raw transaction](https://en.bitcoin.it/wiki/Raw_Transactions) spending given inputs. | | N | |
| decoderawtransaction | <hex string> | | **version 0.7** Produces a human-readable JSON object for a [raw transaction](https://en.bitcoin.it/wiki/Raw_Transactions). | | N | |
| dumpprivkey | <bitcoinaddress> | | Reveals the private key corresponding to <bitcoinaddress> | | Y | |
| encryptwallet | <passphrase> | | Encrypts the wallet with <passphrase>. | | N | |
| getaccount | <bitcoinaddress> | | Returns the account associated with the given address. | | N | |
| ~~getaccountaddress~~ | ~~<account>~~ | | ~~Returns the current bitcoin address for receiving payments to this account.~~ | | ~~N~~ | |
| ~~getaddednodeinfo~~ | ~~<dns> [node]~~ | | **~~version 0.8~~** ~~Returns information about the given added node, or all added nodes~~  ~~(note that onetry addnodes are not listed here) If dns is false, only a list of added nodes will be provided, otherwise connected information will also be available.~~ | |  | |
| getaddressesbyaccount | <account> | | Returns the list of addresses for the given account. | | N | |
| ~~getbalance~~ | ~~[account] [minconf=1]~~ | | ~~If [account] is not specified, returns the server's total available balance.~~  ~~If [account] is specified, returns the balance in the account.~~ | | ~~N~~ | |
| getbestblockhash |  | | **recent git checkouts only** Returns the hash of the best (tip) block in the longest block chain. | | N | |
| ~~getblock~~ | ~~<hash>~~ | | ~~Returns information about the block with the given hash.~~ | | ~~N~~ | |
| ~~getblockcount~~ |  | | ~~Returns the number of blocks in the longest block chain.~~ | | ~~N~~ | |
| ~~getblockhash~~ | ~~<index>~~ | | ~~Returns hash of block in best-block-chain at <index>; index 0 is the~~ [~~genesis block~~](https://en.bitcoin.it/wiki/Genesis_block) | | ~~N~~ | |
| ~~getblocknumber~~ |  | | **~~Deprecated~~**~~.~~ **~~Removed in version 0.7~~**~~. Use getblockcount.~~ | | ~~N~~ | |
| getblocktemplate | [params] | | Returns data needed to construct a block to work on. See [BIP\_0022](https://en.bitcoin.it/wiki/BIP_0022) for more info on params. | | N | |
| ~~getconnectioncount~~ |  | | ~~Returns the number of connections to other nodes.~~ | | ~~N~~ | |
| ~~getdifficulty~~ |  | | ~~Returns the proof-of-work difficulty as a multiple of the minimum difficulty.~~ | | ~~N~~ | |
| ~~getgenerate~~ |  | | ~~Returns true or false whether bitcoind is currently generating hashes~~ | | ~~N~~ | |
| ~~gethashespersec~~ |  | | ~~Returns a recent hashes per second performance measurement while generating.~~ | | ~~N~~ | |
| ~~getinfo~~ |  | | ~~Returns an object containing various state info.~~ | | ~~N~~ | |
| getmemorypool | [data] | | **Replaced in v0.7.0 with getblocktemplate, submitblock, getrawmempool** | | N | |
| ~~getmininginfo~~ |  | | ~~Returns an object containing mining-related information:~~   * ~~blocks~~ * ~~currentblocksize~~ * ~~currentblocktx~~ * ~~difficulty~~ * ~~errors~~ * ~~generate~~ * ~~genproclimit~~ * ~~hashespersec~~ * ~~pooledtx~~ * ~~testnet~~ | | ~~N~~ | |
| ~~getnewaddress~~ | ~~[account]~~ | | ~~Returns a new bitcoin address for receiving payments. If [account] is specified payments received with the address will be credited to [account].~~ | | ~~N~~ | |
| ~~getpeerinfo~~ |  | | **~~version 0.7~~** ~~Returns data about each connected node.~~ | | ~~N~~ | |
| getrawchangeaddress | [account] | | **recent git checkouts only** Returns a new Bitcoin address, for receiving change. This is for use with raw transactions, NOT normal use. | | Y | |
| ~~getrawmempool~~ |  | | **~~version 0.7~~** ~~Returns all transaction ids in memory pool~~ | | ~~N~~ | |
| **getrawtransaction** | **<txid> [verbose=0]** | | **version 0.7 Returns** [**raw transaction**](https://en.bitcoin.it/wiki/Raw_Transactions) **representation for given transaction id.** | | **N** | |
| ~~getreceivedbyaccount~~ | ~~[account] [minconf=1]~~ | | ~~Returns the total amount received by addresses with [account] in transactions with at least [minconf] confirmations. If [account] not provided return will include all transactions to all accounts. (version 0.3.24)~~ | | ~~N~~ | |
| **getreceivedbyaddress** | | **<bitcoinaddress> [minconf=1]** | | **Returns the amount received by <bitcoinaddress> in transactions with at least [minconf] confirmations. It correctly handles the case where someone has sent to the address in multiple transactions. Keep in mind that addresses are only ever used for receiving transactions. Works only for addresses in the local wallet, external addresses will always show 0.** | | **N** | |
| **gettransaction** | | **<txid>** | **Returns an object about the given transaction containing:**   * **"amount" : total amount of the transaction** * **"confirmations" : number of confirmations of the transaction** * **"txid" : the transaction ID** * **"time" : time associated with the transaction[1].** * **"details" - An array of objects containing:**   + **"account"**   + **"address"**   + **"category"**   + **"amount"**   + **"fee"** | | **N** | |
| gettxout | | <txid> <n> [includemempool=true] | Returns details about an unspent transaction output (UTXO) | | N | |
| gettxoutsetinfo | |  | Returns statistics about the unspent transaction output (UTXO) set | | N | |
| [getwork](https://en.bitcoin.it/wiki/Getwork) | | [data] | If [data] is not specified, returns formatted hash data to work on:   * "midstate" : precomputed hash state after hashing the first half of the data * "data" : block data * "hash1" : formatted hash buffer for second hash * "target" : little endian hash target   If [data] is specified, tries to solve the block and returns true if it was successful. | | N | |
| help | | [command] | List commands, or get help for a command. | | N | |
| importprivkey | | <bitcoinprivkey> [label] [rescan=true] | Adds a private key (as returned by dumpprivkey) to your wallet. This may take a while, as a [rescan](https://en.bitcoin.it/wiki/How_to_import_private_keys#Import_Private_key.28s.29) is done, looking for existing transactions. **Optional [rescan] parameter added in 0.8.0.** Note: There's no need to import public key, as in [ECDSA](https://en.bitcoin.it/wiki/Elliptic_Curve_Digital_Signature_Algorithm) (as opposed to RSA) this can be computed from private key. | | Y | |
| keypoolrefill | |  | Fills the keypool, requires wallet passphrase to be set. | | Y | |
| ~~listaccounts~~ | | ~~[minconf=1]~~ | ~~Returns Object that has account names as keys, account balances as values.~~ | | ~~N~~ | |
| listaddressgroupings | |  | **version 0.7** Returns all addresses in the wallet and info used for coincontrol. | | N | |
| listreceivedbyaccount | | [minconf=1] [includeempty=false] | Returns an array of objects containing:   * "account" : the account of the receiving addresses * "amount" : total amount received by addresses with this account * "confirmations" : number of confirmations of the most recent transaction included | | N | |
| listreceivedbyaddress | | [minconf=1] [includeempty=false] | Returns an array of objects containing:   * "address" : receiving address * "account" : the account of the receiving address * "amount" : total amount received by the address * "confirmations" : number of confirmations of the most recent transaction included   To get a list of accounts on the system, execute bitcoind listreceivedbyaddress 0 true | | N | |
| listsinceblock | | [blockhash] [target-confirmations] | Get all transactions in blocks since block [blockhash], or all transactions if omitted. **[target-confirmations] is** [**ignored (bug)**](https://github.com/bitcoin/bitcoin/issues/1153) **up to (at least) v0.8.5** | | N | |
| listtransactions | | [account] [count=10] [from=0] | Returns up to [count] most recent transactions skipping the first [from] transactions for account [account]. If [account] not provided will return recent transaction from all accounts. | | N | |
| listunspent | | [minconf=1] [maxconf=999999] | **version 0.7** Returns array of unspent transaction inputs in the wallet. | | N | |
| listlockunspent | |  | **version 0.8** Returns list of temporarily unspendable outputs | |  | |
| lockunspent | | <unlock?> [array-of-objects] | **version 0.8** Updates list of temporarily unspendable outputs | |  | |
| move | | <fromaccount> <toaccount> <amount> [minconf=1] [comment] | Move from one account in your wallet to another | | N | |
| sendfrom | | <fromaccount> <tobitcoinaddress> <amount> [minconf=1] [comment] [comment-to] | <amount> is a real and is rounded to 8 decimal places. Will send the given amount to the given address, ensuring the account has a valid balance using [minconf] confirmations. Returns the transaction ID if successful (not in JSON object). | | Y | |
| sendmany | | <fromaccount> {address:amount,...} [minconf=1] [comment] | amounts are double-precision floating point numbers | | Y | |
| sendrawtransaction | | <hexstring> | **version 0.7** Submits [raw transaction](https://en.bitcoin.it/wiki/Raw_Transactions) (serialized, hex-encoded) to local node and network. | | N | |
| sendtoaddress | | <bitcoinaddress> <amount> [comment] [comment-to] | <amount> is a real and is rounded to 8 decimal places. Returns the transaction ID <txid> if successful. | | Y | |
| setaccount | | <bitcoinaddress> <account> | Sets the account associated with the given address. Assigning address that is already assigned to the same account will create a new address associated with that account. | | N | |
| setgenerate | | <generate> [genproclimit] | <generate> is true or false to turn generation on or off.  Generation is limited to [genproclimit] processors, -1 is unlimited. | | N | |
| settxfee | | <amount> | <amount> is a real and is rounded to the nearest 0.00000001 | | N | |
| signmessage | | <bitcoinaddress> <message> | Sign a message with the private key of an address. | | Y | |
| signrawtransaction | | <hexstring> [{"txid":txid,"vout":n,"scriptPubKey":hex},...] [<privatekey1>,...] | **version 0.7** Adds signatures to a [raw transaction](https://en.bitcoin.it/wiki/Raw_Transactions) and returns the resulting raw transaction. | | Y/N | |
| stop | |  | Stop bitcoin server. | | N | |
| submitblock | | <hex data> [optional-params-obj] | Attempts to submit new block to network. | | N | |
| validateaddress | | <bitcoinaddress> | Return information about <bitcoinaddress>. | | N | |
| verifymessage | | <bitcoinaddress> <signature> <message> | Verify a signed message. | | N | |
| walletlock | |  | Removes the wallet encryption key from memory, locking the wallet. After calling this method, you will need to call walletpassphrase again before being able to call any methods which require the wallet to be unlocked. | | N | |
| walletpassphrase | | <passphrase> <timeout> | Stores the wallet decryption key in memory for <timeout> seconds. | | N | |
| walletpassphrasechange | | <oldpassphrase> <newpassphrase> | Changes the wallet passphrase from <oldpassphrase> to <newpassphrase>. | | N | |