

Bitcoin Core version 0.19.0.1 is now available from:

<https://bitcoincore.org/bin/bitcoin-core-0.19.0.1/>

This release includes new features, various bug fixes and performance improvements, as well as updated translations.

Please report bugs using the issue tracker at GitHub:

<https://github.com/bitcoin/bitcoin/issues>

To receive security and update notifications, please subscribe to:

<https://bitcoincore.org/en/list/announcements/join/>

How to Upgrade

If you are running an older version, shut it down. Wait until it has completely shut down (which might take a few minutes for older versions), then run the installer (on Windows) or just copy over `/Applications/Bitcoin-Qt` (on Mac) or `bitcoind` / `bitcoin-qt` (on Linux).

Upgrading directly from a version of Bitcoin Core that has reached its EOL is possible, but might take some time if the datadir needs to be migrated. Old wallet versions of Bitcoin Core are generally supported.

Compatibility

Bitcoin Core is supported and extensively tested on operating systems using the Linux kernel, macOS 10.10+, and Windows 7 and newer. It is not recommended to use Bitcoin Core on unsupported systems.

Bitcoin Core should also work on most other Unix-like systems but is not as frequently tested on them.

From 0.17.0 onwards, macOS <10.10 is no longer supported. 0.17.0 is built using Qt 5.9.x, which doesn't support versions of macOS older than 10.10. Additionally, Bitcoin Core does not yet change appearance when macOS "dark mode" is activated.

Users running macOS Catalina may need to "right-click" and then choose "Open" to open the Bitcoin Core .dmg. This is due to new signing requirements imposed by Apple, which the Bitcoin Core project does not yet adhere too.

Notable changes

New user documentation

- [Reduce memory](#) suggests configuration tweaks for running Bitcoin Core on systems with limited memory. (#16339)

New RPCs

- `getbalances` returns an object with all balances (`mine` , `untrusted_pending` and `immature`). Please refer to the RPC help of `getbalances` for details. The new RPC is intended to replace `getbalance` , `getunconfirmedbalance` , and the balance fields in `getwalletinfo` . These old calls and fields may be removed in a future version. (#15930, #16239)

- `setwalletflag` sets and unsets wallet flags that enable or disable features specific to that existing wallet, such as the new `avoid_reuse` feature documented elsewhere in these release notes. (#13756)
- `getblockfilter` gets the BIP158 filter for the specified block. This RPC is only enabled if block filters have been created using the `-blockfilterindex` configuration option. (#14121)

New settings

- `-blockfilterindex` enables the creation of BIP158 block filters for the entire blockchain. Filters will be created in the background and currently use about 4 GiB of space. Note: this version of Bitcoin Core does not serve block filters over the P2P network, although the local user may obtain block filters using the `getblockfilter` RPC. (#14121)

Updated settings

- `whitebind` and `whitelist` now accept a list of permissions to provide peers connecting using the indicated interfaces or IP addresses. If no permissions are specified with an address or CIDR network, the implicit default permissions are the same as previous releases. See the `bitcoind -help` output for these two options for details about the available permissions. (#16248)
- Users setting custom `dbcache` values can increase their setting slightly without using any more real memory. Recent changes reduced the memory use by about 9% and made chainstate accounting more accurate (it was underestimating the use of memory before). For example, if you set a value of "450" before, you may now set a value of "500" to use about the same real amount of memory. (#16957)

Updated RPCs

Note: some low-level RPC changes mainly useful for testing are described in the Low-level Changes section below.

- `sendmany` no longer has a `minconf` argument. This argument was not well-specified and would lead to RPC errors even when the wallet's coin selection succeeded. Users who want to influence coin selection can use the existing `-spendzeroconfchange`, `-limitancestorcount`, `-limitdescendantcount` and `-walletpreventlongchains` configuration arguments. (#15596)
- `getbalance` and `sendtoaddress`, plus the new RPCs `getbalances` and `createwallet`, now accept an "avoid_reuse" parameter that controls whether already used addresses should be included in the operation. Additionally, `sendtoaddress` will avoid partial spends when `avoid_reuse` is enabled even if this feature is not already enabled via the `-avoidpartialspends` command line flag because not doing so would risk using up the "wrong" UTXO for an address reuse case. (#13756)
- RPCs which have an `include_watchonly` argument or `includeWatching` option now default to `true` for watch-only wallets. Affected RPCs are: `getbalance`, `listreceivedbyaddress`, `listreceivedbylabel`, `listtransactions`, `listsinceblock`, `gettransaction`, `walletcreatefundedpsbt`, and `fundrawtransaction`. (#16383)
- `listunspent` now returns a "reused" bool for each output if the wallet flag "avoid_reuse" is enabled. (#13756)
- `getblockstats` now uses BlockUndo data instead of the transaction index, making it much faster, no longer dependent on the `-txindex` configuration option, and functional for all non-pruned blocks. (#14802)

- `utxoupdatepsbt` now accepts a `descriptors` parameter that will fill out input and output scripts and keys when known. P2SH-witness inputs will be filled in from the UTXO set when a descriptor is provided that shows they're spending segwit outputs. See the RPC help text for full details. (#15427)
- `sendrawtransaction` and `testmempoolaccept` no longer accept a `allowhighfees` parameter to fail mempool acceptance if the transaction fee exceeds the value of the configuration option `-maxtxfee`. Now there is a hardcoded default maximum feerate that can be changed when calling either RPC using a `maxfeerate` parameter. (#15620)
- `getmempoolancestors`, `getmempooldescendants`, `getmempoolentry`, and `getrawmempool` no longer return a `size` field unless the configuration option `-deprecatedrpc=size` is used. Instead a new `vsize` field is returned with the transaction's virtual size (consistent with other RPCs such as `getrawtransaction`). (#15637)
- `getwalletinfo` now includes a `scanning` field that is either `false` (no scanning) or an object with information about the duration and progress of the wallet's scanning historical blocks for transactions affecting its balances. (#15730)
- `gettransaction` now accepts a third (boolean) argument `verbose`. If set to `true`, a new `decoded` field will be added to the response containing the decoded transaction. This field is equivalent to RPC `decoderawtransaction`, or RPC `getrawtransaction` when `verbose` is passed. (#16185, #16866, #16873)
- `createwallet` accepts a new `passphrase` parameter. If set, this will create the new wallet encrypted with the given passphrase. If unset (the default) or set to an empty string, no encryption will be used. (#16394)
- `getchaintxstats` RPC now returns the additional key of `window_final_block_height`. (#16695)
- `getmempoolentry` now provides a `weight` field containing the transaction weight as defined in BIP141. (#16647)
- The `getnetworkinfo` and `getpeerinfo` commands now contain a new field with decoded network service flags. (#16786)
- `getdescriptorinfo` now returns an additional `checksum` field containing the checksum for the unmodified descriptor provided by the user (that is, before the descriptor is normalized for the `descriptor` field). (#15986)
- `joinsbts` now shuffles the order of the inputs and outputs of the resulting joined PSBT. Previously, inputs and outputs were added in the order PSBTs were provided. This made it easy to correlate inputs to outputs, representing a privacy leak. (#16512)
- `walletcreatefundedpsbt` now signals BIP125 Replace-by-Fee if the `-walletrbf` configuration option is set to true. (#15911)

GUI changes

- The GUI wallet now provides bech32 addresses by default. The user may change the address type during invoice generation using a GUI toggle, or the default address type may be changed with the `-addresstype` configuration option. (#15711, #16497)

- In 0.18.0, a `./configure` flag was introduced to allow disabling BIP70 support in the GUI (support was enabled by default). In 0.19.0, this flag is now **disabled** by default. If you want to compile Bitcoin Core with BIP70 support in the GUI, you can pass `--enable-bip70` to `./configure`. (#15584)

Deprecated or removed configuration options

- `-mempoolreplacement` is removed, although default node behavior remains the same. This option previously allowed the user to prevent the node from accepting or relaying BIP125 transaction replacements. This is different from the remaining configuration option `-walletrbf`. (#16171)

Deprecated or removed RPCs

- `bumpfee` no longer accepts a `totalFee` option unless the configuration parameter `deprecatedrpc=totalFee` is specified. This parameter will be fully removed in a subsequent release. (#15996)
- `bumpfee` has a new `fee_rate` option as a replacement for the deprecated `totalFee`. (#16727)
- `generate` is now removed after being deprecated in Bitcoin Core 0.18. Use the `generatetoaddress` RPC instead. (#15492)

P2P changes

- BIP 61 reject messages were deprecated in v0.18. They are now disabled by default, but can be enabled by setting the `-enablebip61` command line option. BIP 61 reject messages will be removed entirely in a future version of Bitcoin Core. (#14054)
- To eliminate well-known denial-of-service vectors in Bitcoin Core, especially for nodes with spinning disks, the default value for the `-peerbloomfilters` configuration option has been changed to false. This prevents Bitcoin Core from sending the BIP111 NODE_BLOOM service flag, accepting BIP37 bloom filters, or serving merkle blocks or transactions matching a bloom filter. Users who still want to provide bloom filter support may either set the configuration option to true to re-enable both BIP111 and BIP37 support or enable just BIP37 support for specific peers using the updated `-whitelist` and `-whitebind` configuration options described elsewhere in these release notes. For the near future, lightweight clients using public BIP111/BIP37 nodes should still be able to connect to older versions of Bitcoin Core and nodes that have manually enabled BIP37 support, but developers of such software should consider migrating to either using specific BIP37 nodes or an alternative transaction filtering system. (#16152)
- By default, Bitcoin Core will now make two additional outbound connections that are exclusively used for block-relay. No transactions or addr messages will be processed on these connections. These connections are designed to add little additional memory or bandwidth resource requirements but should make some partitioning attacks more difficult to carry out. (#15759)

Miscellaneous CLI Changes

- The `testnet` field in `bitcoin-cli -getinfo` has been renamed to `chain` and now returns the current network name as defined in BIP70 (main, test, regtest). (#15566)

Low-level changes

RPC

- `getblockchaininfo` no longer returns a `bip9_softforks` object. Instead, information has been moved into the `softforks` object and an additional `type` field describes how Bitcoin Core determines whether that soft fork is active (e.g. BIP9 or BIP90). See the RPC help for details. (#16060)
- `getblocktemplate` no longer returns a `rules` array containing `CSV` and `segwit` (the BIP9 deployments that are currently in active state). (#16060)
- `getrpcinfo` now returns a `logpath` field with the path to `debug.log`. (#15483)

Tests

- The regression test chain enabled by the `-regtest` command line flag now requires transactions to not violate standard policy by default. This is the same default used for mainnet and makes it easier to test mainnet behavior on regtest. Note that the testnet still allows non-standard txs by default and that the policy can be locally adjusted with the `-acceptnonstdtxn` command line flag for both test chains. (#15891)

Configuration

- A setting specified in the default section but not also specified in a network-specific section (e.g. testnet) will now produce an error preventing startup instead of just a warning unless the network is mainnet. This prevents settings intended for mainnet from being applied to testnet or regtest. (#15629)
- On platforms supporting `thread_local`, log lines can be prefixed with the name of the thread that caused the log. To enable this behavior, use `-logthreadnames=1`. (#15849)

Network

- When fetching a transaction announced by multiple peers, previous versions of Bitcoin Core would sequentially attempt to download the transaction from each announcing peer until the transaction is received, in the order that those peers' announcements were received. In this release, the download logic has changed to randomize the fetch order across peers and to prefer sending download requests to outbound peers over inbound peers. This fixes an issue where inbound peers could prevent a node from getting a transaction. (#14897, #15834)
- If a Tor hidden service is being used, Bitcoin Core will be bound to the standard port 8333 even if a different port is configured for cleartext connections. This prevents leaking node identity through use of identical non-default port numbers. (#15651)

Mempool and transaction relay

- Allows one extra single-ancestor transaction per package. Previously, if a transaction in the mempool had 25 descendants, or it and all of its descendants were over 101,000 vbytes, any newly-received transaction that was also a descendant would be ignored. Now, one extra descendant will be allowed provided it is an immediate descendant (child) and the child's size is 10,000 vbytes or less. This makes it possible for two-party contract protocols such as Lightning Network to give each participant an output they can spend immediately for Child-Pays-For-Parent (CPFP) fee bumping without allowing one malicious participant to fill the entire package and thus prevent the other participant from spending their output. (#15681)

- Transactions with outputs paying v1 to v16 witness versions (future segwit versions) are now accepted into the mempool, relayed, and mined. Attempting to spend those outputs remains forbidden by policy ("non-standard"). When this change has been widely deployed, wallets and services can accept any valid bech32 Bitcoin address without concern that transactions paying future segwit versions will become stuck in an unconfirmed state. (#15846)
- Legacy transactions (transactions with no segwit inputs) must now be sent using the legacy encoding format, enforcing the rule specified in BIP144. (#14039)

Wallet

- When in pruned mode, a rescan that was triggered by an `importwallet`, `importpubkey`, `importaddress`, or `importprivkey` RPC will only fail when blocks have been pruned. Previously it would fail when `-prune` has been set. This change allows setting `-prune` to a high value (e.g. the disk size) without the calls to any of the import RPCs failing until the first block is pruned. (#15870)
- When creating a transaction with a fee above `-maxtxfee` (default 0.1 BTC), the RPC commands `walletcreatefundedpsbt` and `fundrawtransaction` will now fail instead of rounding down the fee. Be aware that the `feeRate` argument is specified in BTC per 1,000 vbytes, not satoshi per vbyte. (#16257)
- A new wallet flag `avoid_reuse` has been added (default off). When enabled, a wallet will distinguish between used and unused addresses, and default to not use the former in coin selection. When setting this flag on an existing wallet, rescanning the blockchain is required to correctly mark previously used destinations. Together with "avoid partial spends" (added in Bitcoin Core v0.17.0), this can eliminate a serious privacy issue where a malicious user can track spends by sending small payments to a previously-paid address that would then be included with unrelated inputs in future payments. (#13756)

Build system changes

- Python ≥ 3.5 is now required by all aspects of the project. This includes the build systems, test framework and linters. The previously supported minimum (3.4), was EOL in March 2019. (#14954)
- The minimum supported miniUPnPc API version is set to 10. This keeps compatibility with Ubuntu 16.04 LTS and Debian 8 `libminiupnpc-dev` packages. Please note, on Debian this package is still vulnerable to [CVE-2017-8798](#) (in jessie only) and [CVE-2017-1000494](#) (both in jessie and in stretch). (#15993)

0.19.0 change log

Consensus

- #16128 Delete error-prone CScript constructor only used with FindAndDelete (instagibbs)
- #16060 Bury bip9 deployments (jnewbery)

Policy

- #15557 Enhance `bumpfee` to include inputs when targeting a feerate (instagibbs)
- #15846 Make sending to future native witness outputs standard (sipa)

Block and transaction handling

- #15632 Remove ResendWalletTransactions from the Validation Interface (jnewbery)

- #14121 Index for BIP 157 block filters (jimpo)
- #15141 Rewrite DoS interface between validation and net_processing (sdaftuar)
- #15880 utils and libraries: Replace deprecated Boost Filesystem functions (hebasto)
- #15971 validation: Add compile-time checking for negative locking requirement in LimitValidationInterfaceQueue (practicalswift)
- #15999 init: Remove dead code in LoadChainTip (MarcoFalke)
- #16015 validation: Hold cs_main when reading chainActive in RewindBlockIndex (practicalswift)
- #16056 remove unused magic number from consistency check (instagibbs)
- #16171 Remove -mempoolreplacement to prevent needless block prop slowness (TheBlueMatt)
- #15894 Remove duplicated "Error: " prefix in logs (hebasto)
- #14193 validation: Add missing mempool locks (MarcoFalke)
- #15681 Allow one extra single-ancestor transaction per package (TheBlueMatt)
- #15305 [validation] Crash if disconnecting a block fails (sdaftuar)
- #16471 log correct messages when CFPF fails (jnewbery)
- #16433 txmempool: Remove unused default value MemPoolRemovalReason::UNKNOWN (MarcoFalke)
- #13868 Remove unused fScriptChecks parameter from CheckInputs (Empact)
- #16421 Conservatively accept RBF bumps bumping one tx at the package limits (TheBlueMatt)
- #16854 Prevent UpdateTip log message from being broken up (stevenroose)
- #16956 validation: Make GetWitnessCommitmentIndex public (MarcoFalke)
- #16713 Ignore old versionbit activations to avoid 'unknown softforks' warning (jnewbery)
- #17002 chainparams: Bump assumed chain params (MarcoFalke)
- #16849 Fix block index inconsistency in InvalidateBlock() (sdaftuar)

P2P protocol and network code

- #15597 Generate log entry when blocks messages are received unexpectedly (pstratem)
- #15654 Remove unused unsanitized user agent string CNode::strSubVer (MarcoFalke)
- #15689 netaddress: Update CNetAddr for ORCHIDv2 (dongcarl)
- #15834 Fix transaction relay bugs introduced in #14897 and expire transactions from peer in-flight map (sdaftuar)
- #15651 torcontrol: Use the default/standard network port for Tor hidden services, even if the internal port is set differently (luke-jr)
- #16188 Document what happens to getdata of unknown type (MarcoFalke)
- #15649 Add ChaCha20Poly1305@Bitcoin AEAD (jonasschnelli)
- #16152 Disable bloom filtering by default (TheBlueMatt)
- #15993 Drop support of the insecure miniUPnPc versions (hebasto)
- #16197 Use mockable time for tx download (MarcoFalke)
- #16248 Make whitebind/whitelist permissions more flexible (NicolasDorier)
- #16618 [Fix] Allow connection of a noban banned peer (NicolasDorier)
- #16631 Restore default whitelistrelay to true (NicolasDorier)
- #15759 Add 2 outbound block-relay-only connections (sdaftuar)
- #15558 Don't query all DNS seeds at once (sipa)
- #16999 0.19 seeds update (laanwj)

Wallet

- #15288 Remove wallet -> node global function calls (ryanofsky)
- #15491 Improve log output for errors during load (gwillen)
- #13541 wallet/rpc: sendrawtransaction maxfeerate (kallewoof)
- #15680 Remove resendwallettransactions RPC method (jnewbery)
- #15508 Refactor analyzepsbt for use outside RPC code (gwillen)
- #15747 Remove plethora of Get*Balance (MarcoFalke)

- #15728 Refactor relay transactions (jnewbery)
- #15639 bitcoin-wallet tool: Drop libbitcoin_server.a dependency (ryanofsky)
- #15853 Remove unused import checkpoints.h (MarcoFalke)
- #15780 add cachable amounts for caching credit/debit values (kallewoof)
- #15778 Move maxtxfee from node to wallet (jnewbery)
- #15901 log on rescan completion (andrewtoth)
- #15917 Avoid logging no_such_file_or_directory error (promag)
- #15452 Replace CScriptID and CKeyID in CTxDestination with dedicated types (instagibbs)
- #15870 Only fail rescan when blocks have actually been pruned (MarcoFalke)
- #15006 Add option to create an encrypted wallet (achow101)
- #16001 Give WalletModel::UnlockContext move semantics (sipa)
- #15741 Batch write imported stuff in importmulti (achow101)
- #16144 do not encrypt wallets with disabled private keys (mrwhythat)
- #15024 Allow specific private keys to be derived from descriptor (meshcollider)
- #13756 "avoid_reuse" wallet flag for improved privacy (kallewoof)
- #16226 Move ismine to the wallet module (achow101)
- #16239 wallet/rpc: follow-up clean-up/fixes to avoid_reuse (kallewoof)
- #16286 refactoring: wallet: Fix GCC 7.4.0 warning (hebasto)
- #16257 abort when attempting to fund a transaction above -maxtxfee (Sjors)
- #16237 Have the wallet give out destinations instead of keys (achow101)
- #16322 Fix -maxtxfee check by moving it to CWallet::CreateTransaction (promag)
- #16361 Remove redundant pre-TopUpKeypool check (instagibbs)
- #16244 Move wallet creation out of the createwallet rpc into its own function (achow101)
- #16227 Refactor CWallet's inheritance chain (achow101)
- #16208 Consume ReserveDestination on successful CreateTransaction (instagibbs)
- #16301 Use CWallet::Import* functions in all import* RPCs (achow101)
- #16402 Remove wallet settings from chainparams (MarcoFalke)
- #16415 Get rid of PendingWalletTx class (ryanofsky)
- #15588 Log the actual wallet file version and no longer publicly expose the "version" record (achow101)
- #16399 Improve wallet creation (fjahr)
- #16475 Enumerate walletdb keys (MarcoFalke)
- #15709 Do not add "setting" key as unknown (Bushstar)
- #16451 Remove CMerkleTx (jnewbery)
- #15906 Move min_depth and max_depth to coin control (amitiuttarwar)
- #16502 Drop unused OldKey (promag)
- #16394 Allow createwallet to take empty passwords to make unencrypted wallets (achow101)
- #15911 Use wallet RBF default for walletcreatefundedpsbt (Sjors)
- #16503 Remove p2pEnabled from Chain interface (ariard)
- #16557 restore coinbase and confirmed/conflicted checks in SubmitMemoryPoolAndRelay() (jnewbery)
- #14934 Descriptor expansion cache clarifications (Sjors)
- #16383 rpcwallet: default include_watchonly to true for watchonly wallets (jb55)
- #16542 Return more specific errors about invalid descriptors (achow101)
- #16572 Fix Char as Bool in Wallet (JeremyRubin)
- #16753 extract PubKey from P2PK script with Solver (theStack)
- #16716 Use wallet name instead of pointer on unload/release (promag)
- #16185 gettransaction: add an argument to decode the transaction (darosior)
- #16745 Translate all initErrors in CreateWalletFromFile (MarcoFalke)
- #16792 Assert that the HRP is lowercase in Bech32::Encode (meshcollider)
- #16624 encapsulate transactions state (ariard)
- #16830 Cleanup walletinitinterface.h (hebasto)

- #16796 Fix segfault in CreateWalletFromFile (MarcoFalke)
- #16866 Rename 'decode' argument in gettransaction method to 'verbose' (jnewbery)
- #16727 Explicit feerate for bumpfee (instagibbs)
- #16609 descriptor: fix missed m_script_arg arg renaming in #14934 (fanquake)

RPC and other APIs

- #15492 remove deprecated generate method (Sjors)
- #15566 cli: Replace testnet with chain and return network name as per bip70 (fanquake)
- #15564 cli: Remove duplicate wallet fields from -getinfo (fanquake)
- #15642 Remove deprecated rpc warnings (jnewbery)
- #15637 Rename size to vsize in mempool related calls (fanquake)
- #15620 Uncouple non-wallet rpcs from maxTxFee global (MarcoFalke)
- #15616 Clarify decodescript RPCResult doc (MarcoFalke)
- #15669 Fix help text for signtransactionwithXXX (torkelrostad)
- #15596 Ignore sendmany::minconf as dummy value (MarcoFalke)
- #15755 remove unused var in rawtransaction.cpp (Bushstar)
- #15746 RPCHelpMan: Always name dictionary keys (MarcoFalke)
- #15748 remove dead mining code (jnewbery)
- #15751 Speed up deriveaddresses for large ranges (sipa)
- #15770 Validate maxfeerate with AmountFromValue (promag)
- #15474 rest/rpc: Make mempoolinfo atomic (promag)
- #15463 Speedup getaddressesbylabel (promag)
- #15784 Remove dependency on interfaces::Chain in SignTransaction (ariard)
- #15323 Expose g_is_mempool_loaded via getmempoolinfo (Empact)
- #15932 Serialize in getblock without cs_main (MarcoFalke)
- #15930 Add balances RPC (MarcoFalke)
- #15730 Show scanning details in getwalletinfo (promag)
- #14802 faster getblockstats using BlockUndo data (FelixWeis)
- #14984 Speedup getrawmempool when verbose=true (promag)
- #16071 Hint for importmulti in help output of importpubkey and importaddress (kristapsk)
- #16063 Mention getwalletinfo where a rescan is triggered (promag)
- #16024 deriveaddresses: Correction of descriptor checksum in RPC example (ccapo)
- #16217 getrawtransaction: inform about blockhash argument when lookup fails (darosior)
- #15427 Add support for descriptors to utxoupdatepsbt (sipa)
- #16262 Allow shutdown while in generateblocks (pstratem)
- #15483 Adding a 'logpath' entry to getrpcinfo (darosior)
- #16325 Clarify that block count means height excl genesis (MarcoFalke)
- #16326 add new utxoupdatepsbt arguments to the CRPCCCommand and CPRCCvertParam tables (jnewbery)
- #16332 Add logpath description for getrpcinfo (instagibbs)
- #16240 JSONRPCRequest-aware RPCHelpMan (kallewoof)
- #15996 Deprecate totalfee argument in bumpfee (instagibbs)
- #16467 sendrawtransaction help privacy note (jonatack)
- #16596 Fix getblocktemplate CLI example (emilengler)
- #15986 Add checksum to getdescriptorinfo (sipa)
- #16647 add weight to getmempoolentry output (fanquake)
- #16695 Add window final block height to getchaintxstats (leto)
- #16798 Refactor rawtransaction_util's SignTransaction to separate prevtx parsing (achow101)
- #16285 Improve scantxoutset response and help message (promag)
- #16725 Don't show addresses or P2PK in decoderawtransaction (NicolasDorier)
- #16787 Human readable network services (darosior)

- #16251 Improve signrawtransaction error reporting (ajtowns)
- #16873 fix regression in gettransaction (jonatack)
- #16512 Shuffle inputs and outputs after joining psbts (achow101)
- #16521 Use the default maxfeerate value as BTC/kB (Remagpie)
- #16817 Fix casing in getblockchaininfo to be inline with other fields (dangershony)
- #17131 fix -rpcclienttimeout 0 option (fjahr)
- #17249 Add missing deque include to fix build (jbeich)
- #17368 cli: fix -getinfo output when compiled with no wallet (fanquake)

GUI

- #15464 Drop unused return values in WalletFrame (promag)
- #15614 Defer removeAndDeleteWallet when no modal widget is active (promag)
- #15711 Generate bech32 addresses by default (MarcoFalke)
- #15829 update request payment button text and tab description (fanquake)
- #15874 Resolve the qt/guiutil <-> qt/optionsmodel CD (251Labs)
- #15371 Uppercase bech32 addresses in qr codes (benthecarman)
- #15928 Move QRImageWidget to its own file-pair (luke-jr)
- #16113 move coin control "OK" to the right hand side of the dialog (fanquake)
- #16090 Add vertical spacer to peer detail widget (JosuGZ)
- #15886 qt, wallet: Revamp SendConfirmationDialog (hebasto)
- #16263 Use qlInfo() if no error occurs (hebasto)
- #16153 Add antialiasing to traffic graph widget (JosuGZ)
- #16350 Remove unused guard (hebasto)
- #16106 Sort wallets in open wallet menu (promag)
- #16291 Stop translating PACKAGE_NAME (MarcoFalke)
- #16380 Remove unused bits from the service flags enum (MarcoFalke)
- #16379 Fix autostart filenames on Linux for testnet/regtest (hebasto)
- #16366 init: Use InitError for all errors in bitcoind/qt (MarcoFalke)
- #16436 Do not create payment server if -disablewallet option provided (hebasto)
- #16514 Remove unused RPCConsole::tabFocus (promag)
- #16497 Generate bech32 addresses by default (take 2, fixup) (MarcoFalke)
- #16349 Remove redundant WalletController::addWallet slot (hebasto)
- #16578 Do not pass in command line arguments to QApplication (achow101)
- #16612 Remove menu icons (laanwj)
- #16677 remove unused PlatformStyle::TextColorIcon (fanquake)
- #16694 Ensure transaction send error is always visible (fanquake)
- #14879 Add warning messages to the debug window (hebasto)
- #16708 Replace obsolete functions of QSslSocket (hebasto)
- #16701 Replace functions deprecated in Qt 5.13 (hebasto)
- #16706 Replace deprecated QSignalMapper by lambda expressions (hebasto)
- #16707 Remove obsolete QModelIndex::child() (hebasto)
- #16758 Replace QFontMetrics::width() with TextWidth() (hebasto)
- #16760 Change uninstall icon on Windows (GChuf)
- #16720 Replace objc_msgSend() function calls with the native Objective-C syntax (hebasto)
- #16788 Update transifex slug for 0.19 (laanwj)
- #15450 Create wallet menu option (achow101)
- #16735 Remove unused menu items for Windows and Linux (GChuf)
- #16826 Do additional character escaping for wallet names and address labels (achow101)
- #15529 Add Qt programs to msvc build (updated, no code changes) (sipsorcery)
- #16714 add prune to intro screen with smart default (Sjors)

- #16858 advise users not to switch wallets when opening a BIP70 URI (jameshilliard)
- #16822 Create wallet menu option follow-ups (jonatack)
- #16882 Re-generate translations before 0.19.0 (MarcoFalke)
- #16928 Rename address checkbox back to bech32 (MarcoFalke)
- #16837 Fix {C{,XX},LD}FLAGS pickup (dongcarl)
- #16971 Change default size of intro frame (emilengler)
- #16988 Periodic translations update (laanwj)
- #16852 When BIP70 is disabled, get PaymentRequest merchant using string search (achow101)
- #16952 make sure to update the UI when deleting a transaction (jonasschnelli)
- #17031 Prevent processing duplicate payment requests (promag)
- #17135 Make polling in ClientModel asynchronous (promag)
- #17120 Fix start timer from non QThread (promag)
- #17257 disable font antialiasing for QR image address (fanquake)

Build system

- #14954 Require python 3.5 (MarcoFalke)
- #15580 native_protobuf: avoid system zlib (dongcarl)
- #15601 Switch to python3 (take 3) (MarcoFalke)
- #15581 Make less assumptions about build env (dongcarl)
- #14853 latest RapidCheck (fanquake)
- #15446 Improve depends debuggability (dongcarl)
- #13788 Fix --disable-asm for newer assembly checks/code (luke-jr)
- #12051 add missing debian contrib file to tarball (puchu)
- #15919 Remove unused OpenSSL includes to make it more clear where OpenSSL is used (practicalswift)
- #15978 .gitignore: Don't ignore depends patches (dongcarl)
- #15939 gitian: Remove windows 32 bit build (MarcoFalke)
- #15239 scripts and tools: Move non-linux build source tarballs to "bitcoin-binaries/version" directory (hebasto)
- #14047 Add HKDF_HMAC256_L32 and method to negate a private key (jonasschnelli)
- #16051 add patch to common dependencies (fanquake)
- #16049 switch to secure download of all dependencies (Kemu)
- #16059 configure: Fix thread_local detection (dongcarl)
- #16089 add ability to skip building zeromq (fanquake)
- #15844 Purge libtool archives (dongcarl)
- #15461 update to Boost 1.70 (Sjors)
- #16141 remove GZIP export from gitian descriptors (fanquake)
- #16235 Cleaned up and consolidated msbuild files (no code changes) (sipsorcery)
- #16246 MSVC: Fix error in debug mode (Fix #16245) (NicolasDorier)
- #16183 xtrans: Configure flags cleanup (dongcarl)
- #16258 [MSVC]: Create the config.ini as part of bitcoind build (NicolasDorier)
- #16271 remove -Wall from rapidcheck build flags (fanquake)
- #16309 [MSVC] allow user level project customization (NicolasDorier)
- #16308 [MSVC] Copy build output to src/ automatically after build (NicolasDorier)
- #15457 Check std::system for -[alert|block|wallet]notify (Sjors)
- #16344 use #if HAVE_SYSTEM instead of defined(HAVE_SYSTEM) (Sjors)
- #16352 prune dbus from depends (fanquake)
- #16270 expat 2.2.7 (fanquake)
- #16408 Prune X packages (dongcarl)
- #16386 disable unused Qt features (fanquake)
- #16424 Treat -Wswitch as error when --enable-werror (MarcoFalke)

- #16441 remove qt libjpeg check from bitcoin_qt.m4 (fanquake)
- #16434 Specify AM_CPPFLAGS for ZMQ (domob1812)
- #16534 add Qt Creator Makefile.am.user to .gitignore (Bushstar)
- #16573 disable building libsecp256k1 benchmarks (fanquake)
- #16533 disable libxcb extensions (fanquake)
- #16589 Remove unused src/obj-test folder (MarcoFalke)
- #16435 autoconf: Sane `--enable-debug` defaults (dongcarl)
- #16622 echo property tests status during build (jonatack)
- #16611 Remove src/obj directory from repository (laanwj)
- #16371 ignore macOS make deploy artefacts & add them to clean-local (fanquake)
- #16654 build: update RapidCheck Makefile (jonatack)
- #16370 cleanup package configure flags (fanquake)
- #16746 msbuild: Ignore linker warning (sipsorcery)
- #16750 msbuild: adds bench_bitcoin to auto generated project files (sipsorcery)
- #16810 guix: Remove ssp spec file hack (dongcarl)
- #16477 skip deploying plugins we dont use in macdeployqtplus (fanquake)
- #16413 Bump QT to LTS release 5.9.8 (THETCR)
- #15584 disable BIP70 support by default (fanquake)
- #16871 make building protobuf optional in depends (fanquake)
- #16879 remove redundant sed patching (fanquake)
- #16809 zlib: Move toolchain options to configure (dongcarl)
- #15146 Solve SmartOS FD_ZERO build issue (Empact)
- #16870 update boost macros to latest upstream for improved error reporting (fanquake)
- #16982 Factor out qt translations from build system (laanwj)
- #16926 Add OpenSSL termios fix for musl libc (nmarley)
- #16927 Refresh ZeroMQ 4.3.1 patch (nmarley)
- #17005 Qt version appears only if GUI is being built (ch4ot1c)
- #16468 Exclude depends/Makefile.in .gitignore (promag)

Tests and QA

- #15296 Add script checking for deterministic line coverage in unit tests (practicalswift)
- #15338 ci: Build and run tests once on freebsd (MarcoFalke)
- #15479 Add .style.yapf (MarcoFalke)
- #15534 lint-format-strings: open files sequentially (fix for OS X) (gwillen)
- #15504 fuzz: Link BasicTestingSetup (shared with unit tests) (MarcoFalke)
- #15473 bench: Benchmark mempooltojson (MarcoFalke)
- #15466 Print remaining jobs in test_runner.py (stevenroose)
- #15631 mininode: Clearer error message on invalid magic bytes (MarcoFalke)
- #15255 Remove travis_wait from lint script (gkrizek)
- #15686 make pruning test faster (jnewbery)
- #15533 .style.yapf: Set column_limit=160 (MarcoFalke)
- #15660 Overhaul p2p_compactblocks.py (sdaftuar)
- #15495 Add regtests for HTTP status codes (domob1812)
- #15772 Properly log named args in authproxy (MarcoFalke)
- #15771 Prevent concurrency issues reading .cookie file (promag)
- #15693 travis: Switch to ubuntu keyserver to avoid timeouts (MarcoFalke)
- #15629 init: Throw error when network specific config is ignored (MarcoFalke)
- #15773 Add BitcoinTestFramework::sync_* methods (MarcoFalke)
- #15797 travis: Bump second timeout to 33 minutes, add rationale (MarcoFalke)
- #15788 Unify testing setups for fuzz, bench, and unit tests (MarcoFalke)

- #15352 Reduce noise level in test_bitcoin output (practicalswift)
- #15779 Add wallet_balance benchmark (MarcoFalke)
- #15843 fix outdated include in blockfilter_index_tests (jamesob)
- #15866 Add missing syncwithvalidationinterfacequeue to wallet_import_rescan (MarcoFalke)
- #15697 Make swap_magic_bytes in p2p_invalid_messages atomic (MarcoFalke)
- #15895 Avoid re-reading config.ini unnecessarily (luke-jr)
- #15896 feature_filelock, interface_bitcoin_cli: Use PACKAGE_NAME in messages rather than hardcoding Bitcoin Core (luke-jr)
- #15897 QA/mininode: Send all headers upfront in send_blocks_and_test to avoid sending an unconnected one (luke-jr)
- #15696 test_runner: Move feature_pruning to base tests (MarcoFalke)
- #15869 Add settings merge test to prevent regressions (ryanofsky)
- #15758 Add further tests to wallet_balance (MarcoFalke)
- #15841 combine_logs: append node stderr and stdout if it exists (MarcoFalke)
- #15949 test_runner: Move pruning back to extended (MarcoFalke)
- #15927 log thread names by default in functional tests (jnewbery)
- #15664 change default Python block serialization to witness (instagibbs)
- #15988 Add test for ArgsManager::GetChainName (ryanofsky)
- #15963 Make random seed logged and settable (jnewbery)
- #15943 Fail if RPC has been added without tests (MarcoFalke)
- #16036 travis: Run all lint scripts even if one fails (scravy)
- #13555 parameterize adjustment period in versionbits_computeblockversion (JBaczkuk)
- #16079 wallet_balance.py: Prevent edge cases (stevenroose)
- #16078 replace tx hash with txid in rawtransaction test (LongShao007)
- #16042 Bump MAX_NODES to 12 (MarcoFalke)
- #16124 Limit Python linting to files in the repo (practicalswift)
- #16143 Mark unit test blockfilter_index_initial_sync as non-deterministic (practicalswift)
- #16214 travis: Fix caching issues (MarcoFalke)
- #15982 Make msg_block a witness block (MarcoFalke)
- #16225 Make coins_tests/updatecoins_simulation_test deterministic (practicalswift)
- #16236 fuzz: Log output even if fuzzer failed (MarcoFalke)
- #15520 cirrus: Run extended test feature_pruning (MarcoFalke)
- #16234 Add test for unknown args (MarcoFalke)
- #16207 stop generating lcov coverage when functional tests fail (asood123)
- #16252 Log to debug.log in all unit tests (MarcoFalke)
- #16289 Add missing ECC_Stop() in GUI rpcnestedtests.cpp (jonasschnelli)
- #16278 Remove unused includes (practicalswift)
- #16302 Add missing syncwithvalidationinterfacequeue to wallet_balance test (MarcoFalke)
- #15538 wallet_bumpfee.py: Make sure coin selection produces change (instagibbs)
- #16294 Create at most one testing setup (MarcoFalke)
- #16299 bench: Move generated data to a dedicated translation unit (promag)
- #16329 Add tests for getblockchaininfo.softforks (MarcoFalke)
- #15687 tool wallet test coverage for unexpected writes to wallet (jonatack)
- #16267 bench: Benchmark blocktojson (fanatid)
- #14505 Add linter to make sure single parameter constructors are marked explicit (practicalswift)
- #16338 Disable other targets when enable-fuzz is set (qmma70)
- #16334 rpc_users: Also test rpcauth.py with password (dongcarl)
- #15282 Replace hard-coded hex tx with class in test framework (stevenroose)
- #16390 Add --filter option to test_runner.py (promag)
- #15891 Require standard txs in regtest by default (MarcoFalke)

- #16374 Enable passing wildcard test names to test runner from root (jonatack)
- #16420 Fix race condition in wallet_encryption test (jonasschnelli)
- #16422 remove redundant setup in addrman_tests (zenosage)
- #16438 travis: Print memory and number of cpus (MarcoFalke)
- #16445 Skip flaky p2p_invalid_messages test on macOS (fjahr)
- #16459 Fix race condition in example_test.py (sdaftuar)
- #16464 Ensure we don't generate a too-big block in p2sh sigops test (sdaftuar)
- #16491 fix deprecated log.warn in feature_dbcrash test (jonatack)
- #15134 Switch one of the Travis jobs to an unsigned char environment (-funsigned-char) (practicalswift)
- #16505 Changes verbosity of msbuild from quiet to normal in the appveyor script (sipsorcery)
- #16293 Make test cases separate functions (MarcoFalke)
- #16470 Fail early on disconnect in mininode.wait_for_* (MarcoFalke)
- #16277 Suppress output in test_bitcoin for expected errors (gertjaap)
- #16493 Fix test failures (MarcoFalke)
- #16538 Add missing sync_blocks to feature_pruning (MarcoFalke)
- #16509 Adapt test framework for chains other than "regtest" (MarcoFalke)
- #16363 Add test for BIP30 duplicate tx (MarcoFalke)
- #16535 Explain why -whitelist is used in feature_fee_estimation (MarcoFalke)
- #16554 only include and use OpenSSL where it's actually needed (BIP70) (fanquake)
- #16598 Remove confusing hash256 function in util (elichai)
- #16595 travis: Use extended 90 minute timeout when available (MarcoFalke)
- #16563 Add unit test for AddTimeData (mzumsande)
- #16561 Use colors and dots in test_runner.py output only if standard output is a terminal (practicalswift)
- #16465 Test p2sh-witness and bech32 in wallet_import_rescan (MarcoFalke)
- #16582 Rework ci (Use travis only as fallback env) (MarcoFalke)
- #16633 travis: Fix test_runner.py timeouts (MarcoFalke)
- #16646 Run tests with UPnP disabled (fanquake)
- #16623 ci: Add environment files for all settings (MarcoFalke)
- #16656 fix rpc_setban.py race (jonasschnelli)
- #16570 Make descriptor tests deterministic (daveikher)
- #16404 Test ZMQ notification after chain reorg (promag)
- #16726 Avoid common Python default parameter gotcha when mutable dict/list:s are used as default parameter values (practicalswift)
- #16739 ci: Pass down \$makejobs to test_runner.py, other improvements (MarcoFalke)
- #16767 Check for codespell in lint-spelling.sh (kristapks)
- #16768 Make lint-includes.sh work from any directory (kristapks)
- #15257 Scripts and tools: Bump flake8 to 3.7.8 (Empact)
- #16804 Remove unused try-block in assert_debug_log (MarcoFalke)
- #16850 `servicesnames` field in `getpeerinfo` and `getnetworkinfo` (darosior)
- #16551 Test that low difficulty chain fork is rejected (MarcoFalke)
- #16737 Establish only one connection between nodes in rpc_invalidateblock (MarcoFalke)
- #16845 Add notes on how to generate data/wallets/high_minversion (MarcoFalke)
- #16888 Bump timeouts in slow running tests (MarcoFalke)
- #16864 Add python bech32 impl round-trip test (instagibbs)
- #16865 add some unit tests for merkle.cpp (soroosh-sdi)
- #14696 Add explicit references to related CVE's in p2p_invalid_block test (lucash-dev)
- #16907 lint: Add DisabledOpcodeTemplates to whitelist (MarcoFalke)
- #16898 Remove connect_nodes_bi (MarcoFalke)
- #16917 Move common function assert_approx() into util.py (fridokus)
- #16921 Add information on how to add Vulture suppressions (practicalswift)

- #16920 Fix extra_args in wallet_import_rescan.py (MarcoFalke)
- #16918 Make PORT_MIN in test runner configurable (MarcoFalke)
- #16941 travis: Disable feature_block in tsan run due to oom (MarcoFalke)
- #16929 follow-up to rpc: default maxfeerate value as BTC/kB (jonatack)
- #16959 ci: Set \$host before setting fallback values (MarcoFalke)
- #16961 Remove python dead code linter (laanwj)
- #16931 add unittests for CheckProofOfWork (soroosh-sdi)
- #16991 Fix service flag comparison check in rpc_net test (luke-jr) (laanwj)
- #16987 Correct docstring param name (jbampton)
- #17015 Explain QT_QPA_PLATFORM for gui tests (MarcoFalke)
- #17006 Enable UBSan for Travis fuzzing job (practicalswift)
- #17086 Fix fs_tests for unknown locales (carnhofdaki)
- #15903 appveyor: Write @PACKAGE_NAME@ to config (MarcoFalke)
- #16742 test: add executable flag for wallet_watchonly.py (theStack)
- #16740 qa: Relax so that the subscriber is ready before publishing zmq messages (#16740)

Miscellaneous

- #15335 Fix lack of warning of unrecognized section names (AkioNak)
- #15528 contrib: Bump gitian descriptors for 0.19 (MarcoFalke)
- #15609 scripts and tools: Set 'distro' explicitly (hebasto)
- #15519 Add Poly1305 implementation (jonasschnelli)
- #15643 contrib: Gh-merge: include acks in merge commit (MarcoFalke)
- #15838 scripts and tools: Fetch missing review comments in github-merge.py (nkostoulas)
- #15920 lint: Check that all wallet args are hidden (MarcoFalke)
- #15849 Thread names in logs and deadlock debug tools (jamesob)
- #15650 Handle the result of posix_fallocate system call (lucajepa)
- #15766 scripts and tools: Upgrade gitian image before signing (hebasto)
- #15512 Add ChaCha20 encryption option (XOR) (jonasschnelli)
- #15968 Fix portability issue with pthreads (grim-trigger)
- #15970 Utils and libraries: fix static_assert for macro HAVE_THREAD_LOCAL (orientye)
- #15863 scripts and tools: Ensure repos are up-to-date in gitian-build.py (hebasto)
- #15224 Add RNG strengthening (10ms once every minute) (sipa)
- #15840 Contrib scripts: Filter IPv6 by ASN (abitfan)
- #13998 Scripts and tools: gitian-build.py improvements and corrections (hebasto)
- #15236 scripts and tools: Make --setup command independent (hebasto)
- #16114 contrib: Add curl as a required program in gitian-build.py (fanquake)
- #16046 util: Add type safe gettime (MarcoFalke)
- #15703 Update secp256k1 subtree to latest upstream (sipa)
- #16086 contrib: Use newer config.guess & config.sub in install_db4.sh (fanquake)
- #16130 Don't GPG sign intermediate commits with github-merge tool (stevenroose)
- #16162 scripts: Add key for michael ford (fanquake) to trusted keys list (fanquake)
- #16201 devtools: Always use unabbreviated commit IDs in github-merge.py (laanwj)
- #16112 util: Log early messages (MarcoFalke)
- #16223 devtools: Fetch and display ACKs at sign-off time in github-merge (laanwj)
- #16300 util: Explain why the path is cached (MarcoFalke)
- #16314 scripts and tools: Update copyright_header.py script (hebasto)
- #16158 Fix logic of memory_cleanse() on MSVC and clean up docs (real-or-random)
- #14734 fix an undefined behavior in uint::SetHex (kazcw)
- #16327 scripts and tools: Update ShellCheck linter (hebasto)
- #15277 contrib: Enable building in guix containers (dongcarl)

- #16362 Add bilingual_str type (hebasto)
- #16481 logs: add missing space (harding)
- #16581 sipsorcery gitian key (sipsorcery)
- #16566 util: Refactor upper/lowercase functions (kallewoof)
- #16620 util: Move resolveerrmsg to util/error (MarcoFalke)
- #16625 scripts: Remove github-merge.py (fanquake)
- #15864 Fix datadir handling (hebasto)
- #16670 util: Add join helper to join a list of strings (MarcoFalke)
- #16665 scripts: Move update-translations.py to maintainer-tools repo (fanquake)
- #16730 Support serialization of `std::vector<bool>` (sipa)
- #16556 Fix systemd service file configuration directory setup (setpill)
- #15615 Add log output during initial header sync (jonasschnelli)
- #16774 Avoid unnecessary "Synchronizing blockheaders" log messages (jonasschnelli)
- #16489 log: harmonize bitcoind logging (jonatack)
- #16577 util: Cbufferedfile fixes and unit test (LarryRuane)
- #16984 util: Make thread names shorter (hebasto)
- #17038 Don't rename main thread at process level (laanwj)
- #17184 util: Filter out macos process serial number (hebasto)
- #17095 util: Filter control characters out of log messages (laanwj)
- #17085 init: Change fallback locale to C.UTF-8 (laanwj)
- #16957 9% less memory: make SaltedOutpointHasher noexcept (martinus)
- #17449 fix uninitialized variable nMinerConfirmationWindow (bitcoinVBR)

Documentation

- #15514 Update Transifex links (fanquake)
- #15513 add "sections" info to example bitcoin.conf (fanquake)
- #15530 Move wallet lock annotations to header (MarcoFalke)
- #15562 remove duplicate clone step in build-windows.md (fanquake)
- #15565 remove release note fragments (fanquake)
- #15444 Additional productivity tips (Sjors)
- #15577 Enable TLS in link to chris.beams.io (JeremyRand)
- #15604 release note for disabling reject messages by default (jnewbery)
- #15611 Add Gitian key for droark (droark)
- #15626 Update ACK description in CONTRIBUTING.md (jonatack)
- #15603 Add more tips to productivity.md (gwillen)
- #15683 Comment for seemingly duplicate LIBBITCOIN_SERVER (Bushstar)
- #15685 rpc-mining: Clarify error messages (MarcoFalke)
- #15760 Clarify sendrawtransaction::maxfeerate==0 help (MarcoFalke)
- #15659 fix findFork comment (r8921039)
- #15718 Improve netaddress comments (dongcarl)
- #15833 remove out-of-date comment on pay-to-witness support (r8921039)
- #15821 Remove upgrade note in release notes from EOL versions (MarcoFalke)
- #15267 explain AcceptToMemoryPoolWorker's coins_to_uncache (jamesob)
- #15887 Align code example style with clang-format (hebasto)
- #15877 Fix -dustrelayfee= argument docs grammar (keepkeyjon)
- #15908 Align MSVC build options with Linux build ones (hebasto)
- #15941 Add historical release notes for 0.18.0 (laanwj)
- #15794 Clarify PR guidelines w/re documentation (dongcarl)
- #15607 Release process updates (jonatack)
- #14364 Clarify -blocksdir usage (sangaman)

- #15777 Add doxygen comments for keypool classes (jnewbery)
- #15820 Add productivity notes for dummy rebases (dongcarl)
- #15922 Explain how to pass in non-fundamental types into functions (MarcoFalke)
- #16080 build/doc: update bitcoin_config.h packages, release process (jonatack)
- #16047 analyze psbt description in doc/psbt.md (jonatack)
- #16039 add release note for 14954 (fanquake)
- #16139 Add riscv64 to outputs list in release-process.md (JeremyRand)
- #16140 create security policy (narula)
- #16164 update release process for SECURITY.md (jonatack)
- #16213 Remove explicit mention of versions from SECURITY.md (MarcoFalke)
- #16186 doc/lint: Fix spelling errors identified by codespell 1.15.0 (Empact)
- #16149 Rework section on ACK in CONTRIBUTING.md (MarcoFalke)
- #16196 Add release notes for 14897 & 15834 (MarcoFalke)
- #16241 add rapidcheck to vcpkg install list (fanquake)
- #16243 Remove travis badge from readme (MarcoFalke)
- #16256 remove orphaned header in developer notes (jonatack)
- #15964 Improve build-osx document formatting (giulio92)
- #16313 Fix broken link in doc/build-osx.md (jonatack)
- #16330 Use placeholder instead of key expiration date (hebasto)
- #16339 add reduce-memory.md (fanquake)
- #16347 Include static members in Doxygen (dongcarl)
- #15824 Improve netbase comments (dongcarl)
- #16430 Update bips 35, 37 and 111 status (MarcoFalke)
- #16455 Remove downgrading warning in release notes, per 0.18 branch (MarcoFalke)
- #16484 update labels in CONTRIBUTING.md (MarcoFalke)
- #16483 update Python command in msvc readme (sipsorcery)
- #16504 Add release note for the deprecated totalFee option of bumpfee (promag)
- #16448 add note on precedence of options in bitcoin.conf (fanquake)
- #16536 Update and extend benchmarking.md (ariard)
- #16530 Fix grammar and punctuation in developer notes (Tech1k)
- #16574 Add historical release notes for 0.18.1 (laanwj)
- #16585 Update Markdown syntax for bdb packages (emilengler)
- #16586 Mention other ways to conserve memory on compilation (MarcoFalke)
- #16605 Add missing contributor to 0.18.1 release notes (meshcollider)
- #16615 Fix typos in COPYRIGHT (gapeman)
- #16626 Fix spelling error chache -> cache (nilswloewen)
- #16587 Improve versionbits.h documentation (ariard)
- #16643 Add ZMQ dependencies to the Fedora build instructions (hebasto)
- #16634 Refer in rpcbind doc to the manpage (MarcoFalke)
- #16555 mention whitelist is inbound, and applies to blocksonly (Sjors)
- #16645 initial RapidCheck property-based testing documentation (jonatack)
- #16691 improve depends prefix documentation (fanquake)
- #16629 Add documentation for the new whitelist permissions (NicolasDorier)
- #16723 Update labels in CONTRIBUTING.md (hebasto)
- #16461 Tidy up shadowing section (promag)
- #16621 add default bitcoin.conf locations (GChuf)
- #16752 Delete stale URL in test README (michaelfolkson)
- #14862 Declare BLOCK_VALID_HEADER reserved (MarcoFalke)
- #16806 Add issue templates for bug and feature request (MarcoFalke)
- #16857 Elaborate need to re-login on Debian-based after usermod for Tor group (clashicly)

- #16863 Add a missing closing parenthesis in the bitcoin-wallet's help (darosior)
- #16757 CChainState return values (MarcoFalke)
- #16847 add comments clarifying how local services are advertised (jamesob)
- #16812 Fix whitespace errs in .md files, bitcoin.conf, and Info.plist.in (ch4ot1c)
- #16885 Update tx-size-small comment with relevant CVE disclosure (instagibbs)
- #16900 Fix doxygen comment for SignTransaction in rpc/rawtransaction_util (MarcoFalke)
- #16914 Update homebrew instruction for doxygen (Sjors)
- #16912 Remove Doxygen intro from src/bitcoind.cpp (ch4ot1c)
- #16960 replace outdated OpenSSL comment in test README (fanquake)
- #16968 Remove MSVC update step from translation process (laanwj)
- #16953 Improve test READMEs (fjahr)
- #16962 Put PR template in comments (laanwj)
- #16397 Clarify includeWatching for fundrawtransaction (stevenroose)
- #15459 add how to calculate blockchain and chainstate size variables to release process (marcoagner)
- #16997 Update bips.md for 0.19 (laanwj)
- #17001 Remove mention of renamed mapBlocksUnlinked (MarcoFalke)
- #17014 Consolidate release notes before 0.19.0 (move-only) (MarcoFalke)
- #17111 update bips.md with buried BIP9 deployments (MarcoFalke)

Credits

Thanks to everyone who directly contributed to this release:

- 251
- Aaron Clauson
- Akio Nakamura
- Alistair Mann
- Amiti Uttarwar
- Andrew Chow
- andrewtoth
- Anthony Towns
- Antoine Riard
- Aseem Sood
- Ben Carman
- Ben Woosley
- bpay
- Carl Dong
- Carnhof Daki
- Chris Capobianco
- Chris Moore
- Chuf
- clashic
- clashicly
- Cory Fields
- Daki Carnhof
- Dan Gershony
- Daniel Edgecumbe
- Daniel Kraft
- Daniel McNally
- darosior

- David A. Harding
- David Reikher
- Douglas Roark
- Elichai Turkel
- Emil
- Emil Engler
- ezegom
- Fabian Jahr
- fanquake
- Felix Weis
- Ferdinando M. Ametrano
- fridokus
- gapeman
- GChuf
- Gert-Jaap Glasbergen
- Giulio Lombardo
- Glenn Willen
- Graham Krizek
- Gregory Sanders
- grim-trigger
- gwillen
- Hennadii Stepanov
- Jack Mallers
- James Hilliard
- James O'Beirne
- Jan Beich
- Jeremy Rubin
- JeremyRand
- Jim Posen
- John Bampton
- John Newbery
- Jon Atack
- Jon Layton
- Jonas Schnelli
- Jonathan "Duke" Leto
- João Barbosa
- Joonmo Yang
- Jordan Baczuk
- Jorge Timón
- Josu Goñi
- Julian Fleischer
- Karl-Johan Alm
- Kaz Wesley
- keepkeyjon
- Kirill Fomichev
- Kristaps Kaupe
- Kristian Kramer
- Larry Ruane
- Lenny Maiorani
- LongShao007

- Luca Venturini
- lucash-dev
- Luke Dashjr
- marcoagner
- MarcoFalke
- marcuswin
- Martin Ankerl
- Martin Zumsande
- Matt Corallo
- MeshCollider
- Michael Folkson
- Miguel Herranz
- Nathan Marley
- Neha Narula
- nicolas.dorier
- Nils Loewen
- nkostoulas
- NullFunctor
- orient
- Patrick Strateman
- Peter Bushnell
- Peter Wagner
- Pieter Wuille
- practicalswift
- qmma
- r8921039
- RJ Rybarczyk
- Russell Yanofsky
- Samuel Dobson
- Sebastian Falbesoner
- setpill
- shannon1916
- Sjors Provoost
- soroosh-sdi
- Steven Roose
- Suhas Daftuar
- tecnovert
- THETCR
- Tim Ruffing
- Tobias Kaderle
- Torkel Rogstad
- Ulrich Kempken
- whythat
- William Casarin
- Wladimir J. van der Laan
- zenosage

As well as everyone that helped translating on [Transifex](#).