Bitcoin Core version 0.10.1 is now available from:

https://bitcoin.org/bin/bitcoin-core-0.10.1/

This is a new minor version release, bringing bug fixes and translation updates. It is recommended to upgrade to this version.

Please report bugs using the issue tracker at github:

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

Upgrading and downgrading

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).

Downgrade warning

Because release 0.10.0 and later makes use of headers-first synchronization and parallel block download (see further), the block files and databases are not backwards-compatible with pre-0.10 versions of Bitcoin Core or other software:

- Blocks will be stored on disk out of order (in the order they are received, really), which makes it incompatible with some tools or other programs. Reindexing using earlier versions will also not work anymore as a result of this.

- The block index database will now hold headers for which no block is stored on disk, which earlier versions won't support.

If you want to be able to downgrade smoothly, make a backup of your entire data directory. Without this your node will need start syncing (or importing from bootstrap.dat) anew afterwards. It is possible that the data from a completely synchronised 0.10 node may be usable in older versions as-is, but this is not supported and may break as soon as the older version attempts to reindex.

This does not affect wallet forward or backward compatibility.

Notable changes

This is a minor release and hence there are no notable changes. For the notable changes in 0.10, refer to the release notes for the 0.10.0 release at https://github.com/bitcoin/bitcoin/blob/v0.10.0/doc/release-notes.md

0.10.1 Change log

Detailed release notes follow. This overview includes changes that affect external behavior, not code moves, refactors or string updates.

RPC:

- `7f502be` fix crash: createmultisig and addmultisigaddress

- `eae305f` Fix missing lock in submitblock

Block (database) and transaction handling:

- `1d2cdd2` Fix InvalidateBlock to add chainActive.Tip to setBlockIndexCandidates
- `c91c660` fix InvalidateBlock to repopulate setBlockIndexCandidates
- `002c8a2` fix possible block db breakage during re-index
- `a1f425b` Add (optional) consistency check for the block chain data structures
- `1c62e84` Keep mempool consistent during block-reorgs
- `57d1f46` Fix CheckBlockIndex for reindex
- `bac6fca` Set nSequenceId when a block is fully linked

P2P protocol and network code:

- `78f64ef` don't trickle for whitelisted nodes
- `ca301bf` Reduce fingerprinting through timestamps in 'addr' messages.
- `200f293` Ignore getaddr messages on Outbound connections.
- `d5d8998` Limit message sizes before transfer
- `aeb9279` Better fingerprinting protection for non-main-chain getdatas.
- `cf0218f` Make addrman's bucket placement deterministic (countermeasure 1 against eclipse attacks, see http://cs-people.bu.edu/heilman/eclipse/)
- `0c6f334` Always use a 50% chance to choose between tried and new entries (countermeasure 2 against eclipse attacks)
- `214154e` Do not bias outgoing connections towards fresh addresses (countermeasure 2 against eclipse attacks)
- `aa587d4` Scale up addrman (countermeasure 6 against eclipse attacks)
- `139cd81` Cap nAttempts penalty at 8 and switch to pow instead of a division loop

Validation:

- `d148f62` Acquire CCheckQueue's lock to avoid race condition

Build system:

- `8752b5c` 0.10 fix for crashes on OSX 10.6

Wallet:

- N/A

GUI:

- `2c08406` some mac specifiy cleanup (memory handling, unnecessary code)
- `81145a6` fix OSX dock icon window reopening
- `786cf72` fix a issue where "command line options"-action overwrite "Preference"-action (on OSX)

Tests:

- `1117378` add RPC test for InvalidateBlock

Miscellaneous:

- `c9e022b` Initialization: set Boost path locale in main thread
- `23126a0` Sanitize command strings before logging them.
- `323de27` Initialization: setup environment before starting Qt tests

- `7494e09` Initialization: setup environment before starting tests
- `df45564` Initialization: set fallback locale as environment variable

Credits

Thanks to everyone who directly contributed to this release:

- Alex Morcos
- Cory Fields
- dexX7
- fsb4000
- Gavin Andresen
- Gregory Maxwell
- Ivan Pustogarov
- Jonas Schnelli
- Matt Corallo
- mrbandrews
- Pieter Wuille
- Ruben de Vries
- Suhas Daftuar
- Wladimir J. van der Laan

And all those who contributed additional code review and/or security research:

- 21E14
- Alison Kendler
- Aviv Zohar
- Ethan Heilman
- Evil-Knievel
- fanquake
- Jeff Garzik
- Jonas Nick
- Luke Dashjr
- Patrick Strateman
- Philip Kaufmann
- Sergio Demian Lerner
- Sharon Goldberg

As well as everyone that helped translating on Transifex.