

Header Sync DoS Comparison

Sep 23, 2026 · @Eric

Summary

libbitcoin-node holds less per peer than satoshi (bitcoind v32.0rc2), never stores a weak branch permanently, and validates once where satoshi validates three times. Header sync now proves a peer's branch in the protocol before anything reaches the organizer, and blocks-first sync is removed. Figures below are for mainnet as of this date; satoshi figures are from its source at v32.0rc2.

How it works

Each channel proves its peer's branch before the organizer sees any of it, so every header organized is valid and storable.

flowchart LR

```

    A[Survey<br/>validate, discard, sample hash] -->|proven|
    B[Collect<br/>chain + match samples]
    A -->|not proven| X[Discard<br/>samples only]
    B --> C[Organizer<br/>strong: reorganize<br/>weak: tree]
    C --> D[Purge<br/>by work, per 144 blocks]
```

- **Survey:** each header is fully validated against a rolling chain state, then discarded. One full 32-byte hash is kept per sample interval, starting at 2,000 headers.
- **Proof:** a checkpoint on the path, or the peer exhausted with a current top at or above the proof threshold: the greater of the configured minimum work and the candidate's work less 144 blocks of its proof, queried from the organizer when the channel starts. The milestone is recorded only if seen on this path.
- **Collect:** the branch is requested again. Each segment must chain and match its sample hash, which binds it to exactly what was validated, so it is hashed but not revalidated.
- **Sample budget:** the configured header row estimate divided by 2,000 (482 on mainnet). At the budget the interval doubles, so samples stay about 15 KB per channel at any branch length.
- **Organizer:** strong headers reorganize the candidate chain; weak ones wait in the tree. Client-submitted headers are validated there instead.

- **Tree purge:** once per 144 blocks of candidate progress, whole weak branches whose top work is more than 144 blocks of work behind the candidate are removed.

Compared with satoshi

Below the proof threshold, libbitcoin holds less per peer, validates once instead of three times, and keeps nothing permanently.

Aspect (unit)	libbitcoin-node	satoshi (bitcoind v32.0rc2)
Validation passes (per header)	1	3
Salted hashes (per 1,000 headers)	0	about 3
Sample retained (bits per 1,000 headers)	128	about 1.5
Unproven chain memory (KB per peer)	about 15	up to about 650
Honest sync buffer (headers per peer)	2,000	15,400
Honest sync buffer (KB per peer)	about 340	about 740
Weak branch retention (blocks behind candidate)	144 to 288	unlimited (never erased)
Weak branch admission (blocks of work below candidate)	144, or minimum work in initial sync	144, or minimum work in initial sync
Valid low-work headers admissible by collision (headers)	0	up to about 15,400

Admission now matches satoshi's threshold; the difference is that satoshi keeps what it admits.

Attack cost and remedies

No affordable attack produces a backlog: an unproven branch never leaves the protocol, and on a synced node proof requires work within 144 blocks of the candidate.

Attack	Cost (current-difficulty blocks per header)	Memory (KB per channel)	Retention (blocks behind candidate)
Unproven chain (e.g. forked at 295,001)	about 0.00005	about 15	0 (discarded)
Weak fork near the top (ordinary stale blocks, not an attack)	about 1	up to about 340 while collecting	144 to 288
Long weak branch below candidate work	any	about 15	0 (fails proof on a synced node)

- The proof threshold is the configured minimum work during initial sync, as in satoshi; after that it is the candidate's work less 144 blocks, refreshed each time a channel starts a branch.
- Checkpoints never bounded header count, only the floor price, which falls every year.
- No mainnet observations were found of either attack; satoshi disclosed the cheap version as [CVE-2019-25220](#), costing 0.14 BTC in 2024.