

blogs: a log with a witness nobody controls

A log you keep yourself can be edited by whoever holds the machine; a log someone else keeps can be edited by them. **blogs** gives a log a witness nobody controls. Producers append signed, numbered batches; BSV Layered Multicast delivers each batch once to every archive at the same moment; and one small anchor per interval, mined on the ledger, lets any archive prove its copy complete and any auditor prove when a line was committed.

Author: Jeff Harris · Lightweb Inc. · jeff@lightweb.net

Pattern 005 · Version 01

Audience: application developers, founders and business leaders. No Bitcoin or networking background assumed; every term is explained on first use.

We = Lightweb Inc., designer of the bstack patterns and operator of the 1BSV network; **you** = a builder or decision maker evaluating them.

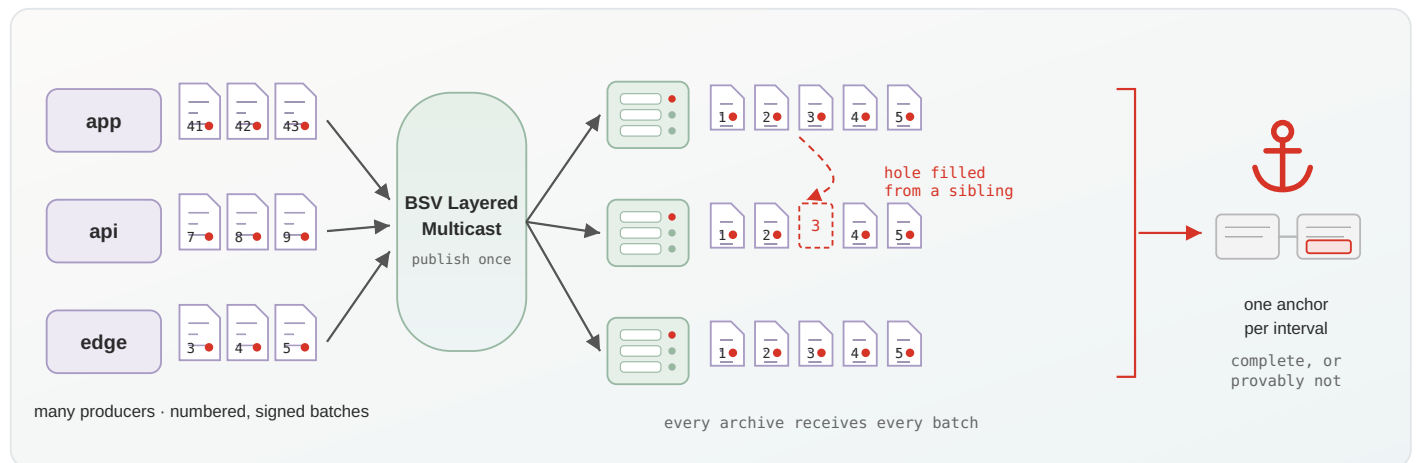


Figure 1: many producers, every archive, one anchor per interval. A hole in one archive is filled from a sibling.

The series, and where this fits

The bstack pattern series rebuilds the building blocks nearly every product needs so that the data defends itself: state proves itself on the BSV ledger, BSV Layered Multicast delivers each publication once to every subscriber, and readers verify instead of trusting whoever served them.

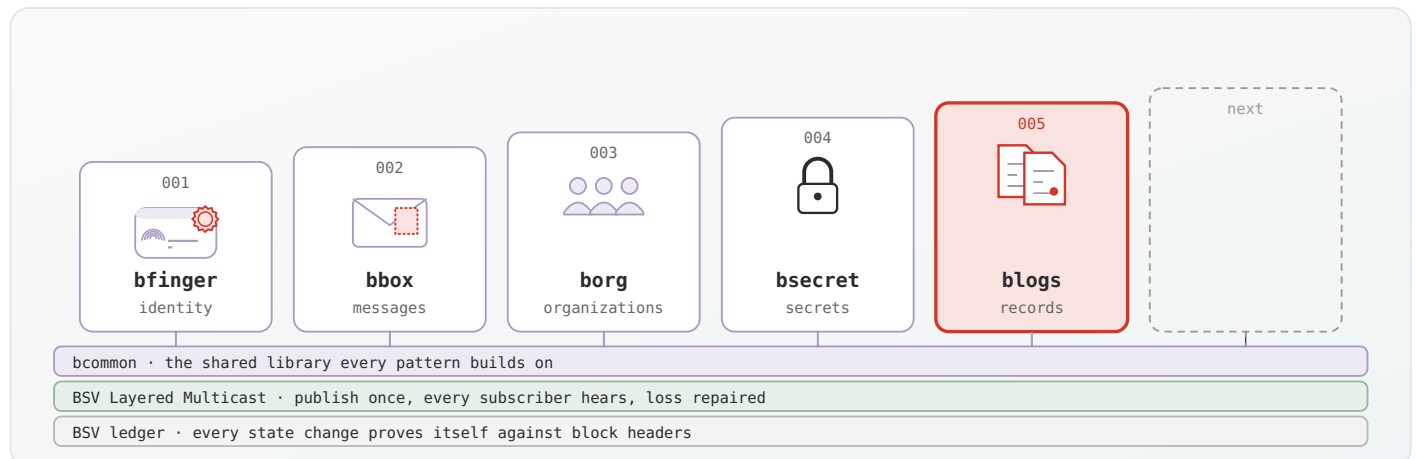


Figure 2: the series as one build. **blogs**, in red, is the fifth pattern.

So far. **bfinger** (001) made an identity prove itself and made it payable. **bbox** (002) delivered privately to one party, payment inside. **borg** (003) gave teams a roll and keys that change when someone leaves, and **bsecret** (004) built a vault on it whose hosts never hold a key. Each of them changes state now and then: one record, one envelope, one membership change, one version at a time.

This one. **blogs** turns the same foundation toward volume. A log is thousands of lines a minute from many machines at once, so the question changes from "what does this say now?" to "is this everything, and when was it written?"

The problem

Every regulated business, every security team and every platform with a dispute desk relies on logs, and every log has the same weakness: it is exactly as trustworthy as whoever holds it. An administrator on the box can edit it. A vendor that stores it can lose, alter or withhold it. A copy shipped to a second destination is only as good as the path it traveled, and proving two copies match, or that neither is missing a line, takes a reconciliation project rather than a check.

The usual remedies add more of the same: a second vendor, a replication tier, a signed export, a compliance vault. Each one moves trust around; none removes it. And none of them can tell an auditor, by arithmetic, that the record in front of them is complete.

What blogs does

A producer pipes its log into blogs. Lines are cut into **batches**, each signed by the producer's key and numbered (1, 2, 3, ...), each naming the batch before it. Every archive subscribed to the stream receives every batch. At the end of each **interval**, ten minutes by default, the producer mines one small **anchor** that commits to every batch in the interval. From then on, three facts about any line can be proven by anyone holding block headers:

- **who signed it:** the producer's key;
- **where it sat:** its position in the numbered stream;
- **when, at the latest:** committed no later than the anchor's block.

Nobody, the producer included, can insert, drop or reorder a line inside an anchored interval without the proof failing.

```
$ tail -F /var/log/app.log | blogs produce app_qzxkvbwmtr
$ blogs verify-interval app_qzxkvbwmtr 02c0ffee... 1
VERIFIED interval 1 of app_qzxkvbwmtr by 02c0ffee...: seqs 1 to 4, root 5d7f...3999,
committed no later than block 912430
```

Figure 3: produce and verify. Readers and auditors need no node and no coin.

How it works

1. **Batches are never mined.** Each batch rides a transaction built so it can never be mined, spending one small output of a mined **funding tree**. It verifies by SPV (simplified payment verification: checking a transaction against block headers alone) without paying for block space, and the producer can retract it.
2. **One anchor per interval is mined.** The anchor holds the Merkle root (a single fingerprint of a whole list) over the interval's batches, using the same RFC 6962 tree construction Certificate Transparency uses. Each anchor spends the previous one, so a producer's anchors form one chain covering every batch exactly once.
3. **Completeness becomes a check.** An archive holding every numbered batch of an interval recomputes the root; if it equals the anchor's root, the copy is complete and proven. If not, the gap is a numbered hole, not a mystery.
4. **Holes are repaired from siblings.** On the plane, loss is repaired in transit. Without it, one backfill command copies a missing batch from the first sibling whose answer verifies. No producer needs to be online, and no archive has to trust another.
5. **Many producers, no coordinator.** Any number of producers write to one stream, each with its own sequence and its own anchor chain. The stream's topic is the meeting point; nothing else is shared.

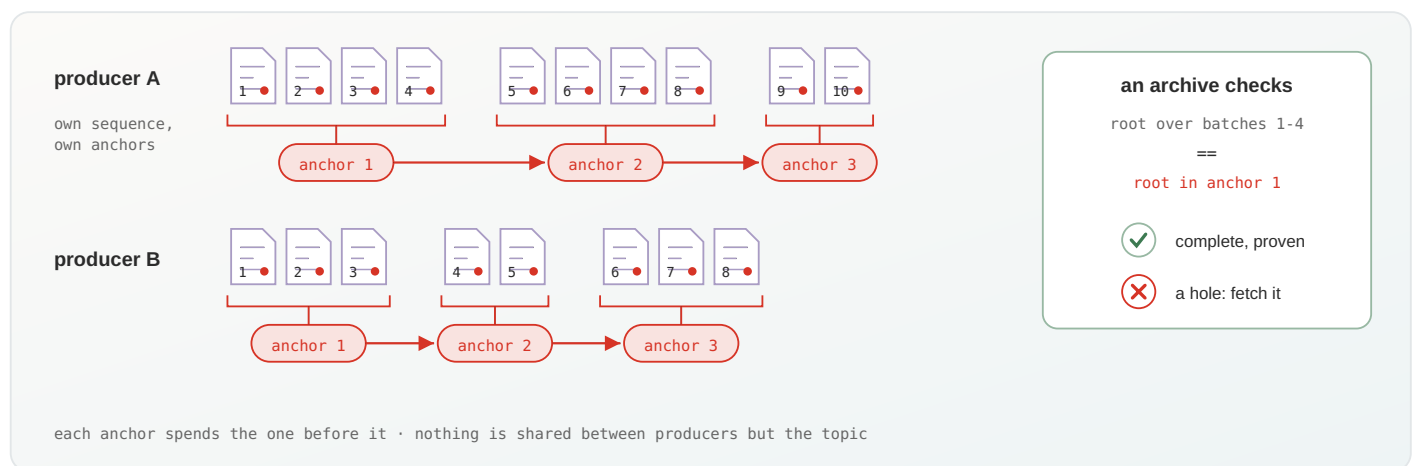


Figure 4: each producer numbers its own batches and anchors its own intervals; an archive proves completeness by recomputing one root.

Two profiles cover two kinds of data. **Full bytes** carries the log content in the record and is the archiving default. **Digest only** puts a fingerprint on the network and moves the bytes out of band, for content that must stay in one jurisdiction. Profiles can be mixed batch by batch.

Words in this brief. *Producer:* a machine or service writing a stream. *Archive:* an overlay host that keeps a stream. *Interval:* the span one anchor covers. *Merkle root:* one short fingerprint that commits to a whole numbered list. *Funding tree:* one mined transaction with many small outputs, one spent per batch.

Why it is new

- **No BRC defined a provable log stream before.** The nearest Bitcoin standards keep log records off chain and put at most a root on chain. blogs defines the batch, the stream and the interval anchor, and makes each producer's completeness, not only integrity, something an auditor proves.
- **The proof costs one anchor per interval, not one per line.** Batches are never mined: a batch's only ledger cost is its small share of a funding tree. Sequence, timing and completeness for up to thousands of batches cost one mined anchor.
- **Retraction that keeps the record honest.** A producer can withdraw batches with one mined sweep per funding tree. Honest archives stop serving them, but the numbered position stays, so the anchor still proves the interval and nobody can pretend the gap was never there.
- **The audit needs nothing but files and headers.** An archive's own files plus block headers are enough; no host, no producer, no vendor needs to cooperate. The tooling distinguishes a forgery from an outage in its exit codes.

Built to scale: BSV Layered Multicast

A log with one archive is a log someone can lose. The value is in several independent archives, in different organizations or jurisdictions, holding the same bytes at the same moment, and that is where point-to-point delivery hurts: every producer uploads every batch to every archive, so cost multiplies by producers times archives.



Figure 5: one publish per batch, whatever the number of archives.

On BSV Layered Multicast a producer submits each batch once and every subscribed archive receives it at the same moment, with loss detected and repaired by the network. Adding an archive is a subscription the producer never needs to learn about. blogs also runs without the network, each producer uploading to a handful of archives and repairing by backfill; past a few archives, the plane is what removes the cost.

What it means for the business

Who	Pays	Gets
A producer	miner fees (a small share of a funding tree per batch, one anchor per interval) and access to a node	a log nobody can quietly edit, including itself
An archive operator	its storage and the delivery it elects	a copy it can prove complete; a seam to sell retention and history
An auditor or reader	nothing: no node, no coin	proof of who, where and when, from headers alone

- **Compliance becomes a check.** "Show me the log is complete" stops being a reconciliation project and becomes a command.
- **Multi-cloud and multi-jurisdiction by subscription.** A second region, a regulator's archive or a customer's own copy is one more subscriber, not one more pipeline.
- **The analyst and the auditor hold the same bytes.** Security telemetry, audit trails and application logs can ride one stream, each producer's record proven by its own anchors.

Honest limits

- **Batching sets the latency floor,** and a batch is unproven until its interval's anchor is mined.
- **It proves a line was committed, not that it is true,** and not when it was written before its anchor.

- **Metadata is public:** who writes to which stream, and how much.
- **A producer's key is its authority.** A stolen key writes in its name; key rotation is not part of this version.
- **Retraction is not erasure,** and an auditor sees an empty position without learning why it is empty.

What comes next

Five patterns in, the series has an identity, a private message with payment inside, a team, its secrets, and a record of everything that happened. The patterns that follow compose these into the products people use every day, each standing on the ones before it, which is why each gets smaller to build and easier to trust.

Run it on 1BSV

The 1BSV network is the commercial operation of BSV Layered Multicast, run by Lightweb Inc. For an application built on blogs:

- **Producers ride free.** Publishing into the network is never billed, and neither are the network's repair retransmissions.
- **Archives pay for what is delivered to them,** and for nothing else: overlay objects with their proofs on an application account, which also carries the live header feed to verify against. No subscriptions, no minimums, no commitments.
- **Early access.** The overlay lane is onboarding its first users. Onboarding is self-service with one approval step.

Go deeper

- *The bstack: publish once, prove everything, and let users ride free* (Lightweb Inc., Paper 009): <https://1bsv.net/papers/bstack-patterns.pdf>
- *The overlay object plane* (Lightweb Inc., Paper 006): <https://1bsv.net/papers/overlay-object-plane.pdf>
- The pattern series: <https://1bsv.net/patterns.html>

1BSV - BSV Layered Multicast - cash moves fast on 1BSV. · 1bsv.net

Lightweb Inc. · Jeff Harris · jeff@lightweb.net · github.com/lightwebinc

Unbounded Solutions for a Small World™ · © Lightweb Inc.