RSW-0008 — Network Events

Revision 0.01 | Protocol rsw/1 | Format 1 | Experimental proposal

Developed by Pablo Murad — November 23, 2026 — pablo@pablomurad.com — https://rsw.works

Experimental Project Specification. This document is an experimental specification of the Really Small Web project. It is a work in progress and has not been published as an RFC or approved as an Internet Standard.

License: specifications CC0 1.0 (public domain); future reference implementation and official tools MIT.

RSW-0008 — Network Events

Revision 0.01 | Protocol rsw/1 | Format 1

Post-MVP — experimental draft; not enabled in basic rsw/1

Developed by Pablo Murad

Pablo Murad — November 23, 2026

pablo@pablomurad.com

MUST and MUST NOT indicate requirements of the described profile; SHOULD indicates a recommendation; MAY indicates an option. Experimental document, not yet ratified or implemented. The rules of RSW-0001 through RSW-0003 remain applicable.

01. Status and Objective

Post-MVP proposal; it is not part of basic rsw/1 and does not modify PNGs. It introduces an optional transport of signed events over HTTPS. There is no consensus, guaranteed delivery, central authority, or automatic relay discovery. Clients and nodes opt into the experimental events/1 profile independently of cartridge support.

02. Envelope

Strict UTF-8 JSON, without duplicates, with the integer subset and canonicalization of RSW-0002/0003. Maximum 32 KiB per envelope, depth 16. Exact fields: id, payload, signature. id = "sha256:" + lowercase hex of SHA-256(JCS(payload)); signature = 128 lowercase hex of Ed25519 over ASCII("RSW/1:network-event") || NUL || JCS(payload). Apply strict RSW-0003 point validation. The payload has protocol="rsw/1", stream="events/1", type, public_key, created_at, nonce, and body. type is an RSW-0002 identifier up to 32 bytes. public_key and created_at follow RSW-0002, both mandatory. nonce is 32 lowercase hex from 16 CSPRNG bytes. body is a typed object. Do not include id or signature in the payload. Do not sign a hash in place of the message.

Proposed types: announce and update (body = {name, version, sha256, domain}, RSW-0006 rules, public_key must match the publication); reference (body = {uri}, RSW-0004 URI); follow, comment, reaction, message, and encrypted-message (RSW-0009/0014); mirror (RSW-0010); key-binding, key-rotation, key-revocation (RSW-0015). body fields are exact per type. Unknown types are neither executed nor applied; nodes may refuse them as UNSUPPORTED_EVENT. An announcement event is a location hint, not an authorized index nor a confirmation of a new edition. Verify the object through normal resolution.

03. Experimental Transport

Explicitly configured HTTPS node, origin validated as RSW-0005. The POST /rsw-events/v1/events endpoint receives one JSON envelope, responding 201 if stored, 200 if an identical one is already known, 400 invalid, 413 too large, 429 quota, 403 policy. Responses do not prove delivery to recipients. GET /rsw-events/v1/events?after=<cursor>&limit=<n> returns {events:[...], next:<cursor>, more:<boolean>}; opaque ASCII cursor [A-Za-z0-9_-]{1,128}, omitted on the first query; limit 1..50, default 20. Response up to 2 MiB. The cursor belongs to the node, not to the author's clock; an expired cursor returns 410 and requires an explicit restart. Each query has limits and can be cancelled. GET does not grant posting permissions. No implicit cookies; administrative authentication, if necessary, is outside cartridge content.

04. Validation and Storage

Limit bytes before parsing, validate the schema, recompute id, verify the signature and type rules before persisting. Deduplicate by id; the same id with different canonical content is a conflict and requires rejection. Local ingestion sequence is independent of created_at, which does not prove global order. Clients store cursors per node and seen ids, process idempotently, and do not interpret repeated receipt as a new action. Events may arrive out of order. The node may expire events per its published policy; do not promise universal delivery or retention. Quotas per origin, key, rate, and disk; moderate spam and excessive loads. Events are not RPC calls and enable no execution.

05. Consent and Privacy

No event traffic in the MVP. Adding a node, sending an event, or syncing requires explicit configuration/consent. Public events may be copied by third parties. TLS protects transport; it does not hide node metadata; the profile does not protect against IP or traffic observation. Do not transmit reading history, private keys, or library data. Sending private messages depends on RSW-0014; a plaintext body is not secret.

06. Tests and Blockers

JCS/Ed25519 vectors per domain, nonce, duplication, altered id, unknown type, wrong clock, pagination, expired cursor, response limit, and abuse. Pending: private node authentication, retention policies and signature filters, pagination under deletions, DoS protection, implementation, and interoperability. Do not activate before review. References: RSW-0003/0005/0009/0010/0014/0015/0016; RFC 8785, RFC 8032, RFC 9110 (https://www.rfc-editor.org/).

License: CC0 1.0 (public domain dedication; see CC0-1.0.txt in the downloads area). Developed by Pablo Murad | pablo@pablomurad.com END — RSW-0008

Author's Address

Pablo Murad
Independent
pablo@pablomurad.com
https://pablomurad.com
Official project website: https://rsw.works

Small files. Independent worlds.