ODNCA-STD-009 · Paymail Compatibility Profile

Process web3 names exactly like web2 paymail — one extra line of code

Status: v1.0 — Adopted

Audience: wallet and application builders who already speak bsvalias/Paymail.

Companion documents: ODNCA-STD-001 (Resolution), ODNCA-STD-006 (Data Channels), ODNCA-STD-002 (the live TLD lists).

Reference implementation: the ORDnet bsvalias bridge v1.1.0, live behind the profile endpoint.

1. The promise

Your code already speaks paymail: capability discovery, `pki`, `paymentDestination`, the P2P pair. This profile lets that exact code pay **native web3 names** — `info@earthlog.web3`, `pay@spiek.web3` — with **one change: where you fetch the capability document.** Every request and response after that line is byte-compatible bsvalias 1.0. You do not implement a new resolver, a new signature scheme, or a new wire format. You add an `if`.

2. The one extra line

Standard paymail discovery starts with a DNS SRV lookup on the handle's domain. A web3 domain (`earthlog.web3`) does not exist in ICANN DNS, so that lookup fails before anything else can happen. The profile's rule replaces that first step — and only that step:

> **If the handle's TLD is a recognised web3 TLD, skip DNS discovery and fetch the capability document from the profile endpoint. Everything else is your existing paymail flow, unchanged.**

In code:

```
const ODNCA_TLDS = await odncaTlds();          // §4 — cached daily
const WELL_KNOWN  = "https://sns.ordnet.io/.well-known/bsvalias";

// --- the one extra line ---
const capUrl = ODNCA_TLDS.has(tldOf(domain))
  ? WELL_KNOWN                                 // web3: fixed endpoint
  : await srvDiscover(domain);                 // web2: your existing code

// from here on: your existing bsvalias client, byte-for-byte
const caps = await fetchJson(capUrl);
```

3. What you get back (nothing new)

The capability document is standard bsvalias 1.0 with the standard keys — `pki`, `paymentDestination`, the P2P pair (`2a40af698840`, `5f1323cddf31`), verify-pubkey, public-profile — each a URI template with the literal `{alias}@{domain.tld}` placeholders your client already substitutes. Substituting `info@earthlog.web3` just works:

YOUR EXISTING CALL	ANSWER
`POST …/address/info@earthlog.web3`	`{ "output": "76a914…88ac" }` — the current on-chain holder's locking script
`GET …/id/info@earthlog.web3`	`{ "bsvalias": "1.0", "handle": "info@earthlog.web3", "pubkey": "02…" }`
`POST …/p2p-payment-destination/…`	`{ "outputs": […], "reference": "ordnet-…" }` — then submit the tx as always

Behind the profile endpoint the answers come from the on-chain name state (ODNCA-STD-001) instead of a customer database — your client cannot tell the difference, and does not need to. Mailbox semantics follow the name standard: the holder of a mailbox is by definition the holder of the domain; an unknown mailbox still pays the domain holder. Unknown or unrecognised domains answer a clean `404 unknown_domain` — exactly the failure shape your error handling already expects.

4. The TLD list

The recognised TLD set is published, live, and small:

- **Machine source:** the live parameter lists (ODNCA-STD-002), served by the registry and mirrored at the profile endpoint's health route.
- **Snapshot at adoption:** `web3, bitcoin, crypto, blockchain, ordnet` (+ resolve-only: `bsv, bitcoinsv`).
- **Caching:** refresh daily; the list changes rarely and only ever by published, chain-anchored decision (ODNCA-STD-004).

A hardcoded copy of the snapshot is acceptable as a fallback; the live fetch keeps you current when TLDs activate.

5. What deliberately did not change

- **No new trust asked.** The profile endpoint discloses only what the public resolver already discloses. Wallets wanting more than paymail offers can upgrade any time to the native signed answers with outpoint proof (ODNCA-STD-001 §9) — but that is an upgrade, never the entry fee.
- **No house-domain aliasing.** There is no `name@ordnet.io` form: a web3 name is addressed as itself, never as an alias under someone else's domain. The profile endpoint's hostname is infrastructure — it never appears in anyone's address.
- **web2 paymail is untouched.** Handles on ordinary DNS domains follow your existing SRV path; this profile only claims the TLDs it publishes.

---

6. Conformance

A client conforms to this profile when: (a) for handles whose TLD is in the published set, it fetches the capability document from the profile endpoint instead of DNS discovery; (b) it treats the responses as standard bsvalias 1.0, which they are; and (c) it refreshes the TLD set at least daily. Test with one call: resolve `info@earthlog.web3` through your existing client with the one line added — if payment resolution returns a script, you are done. That is the entire certification.
