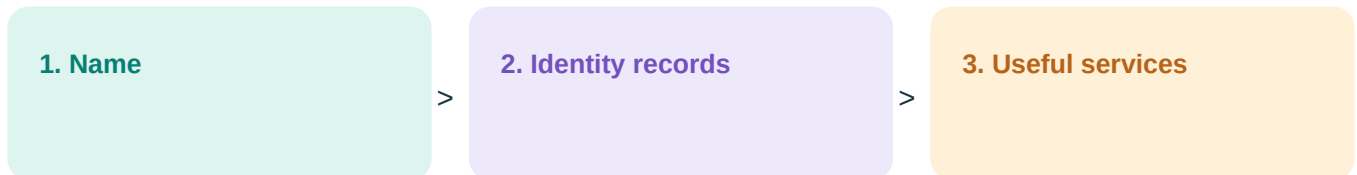


Learn in ELI5 Style

A NamID is a readable name with rules and records. Think of a name as a labelled drawer: the label helps people find it, the records say what is inside, and the owner's key controls changes. A name can describe a person, project, app, place or service. This guide walks through the test environment, one small task at a time.



- 1 Start with pages 2-5 to choose the right network and connect a wallet.
- 2 Use pages 6-10 for auctions and refunds; pages 11-15 for names, records and custody.
- 3 Explore calls, mail, signing and recovery on pages 16-18. Pages 19-20 help when something goes wrong.

YOU KNOW IT WORKED WHEN

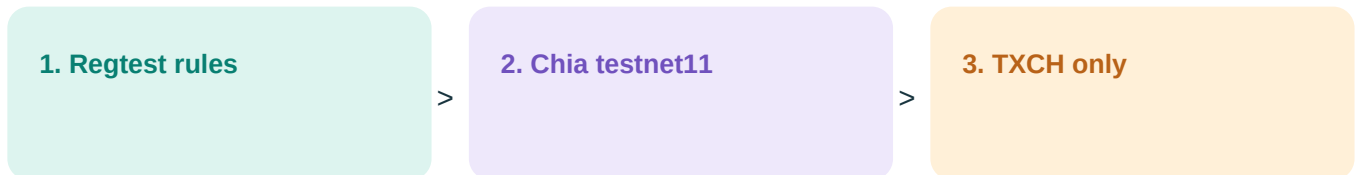
You need the NamID app and wallet, a backed-up recovery phrase and testnet TXCH for paid tests. Reading this guide costs nothing.

KEEP THIS IN MIND

Edition: 26 September 2026. This is a testnet guide, not a statement of mainnet readiness. Buttons and services may evolve.

Two clocks, one test chain

Protocol mode controls the rehearsal clock. Blockchain network controls which coins and signatures work. They are different settings. NamID has main and regtest modes; the public rehearsal currently uses regtest on Chia testnet11. TXCH belongs to that test chain. XCH on mainnet cannot pay a testnet bill.



- 1 Look for testnet11 in the website banner and in your wallet's selected network.
- 2 The rehearsal releases the name batch across five days. Each auction runs ten hours: 6 hours 40 minutes to bid, then 3 hours 20 minutes to reveal.
- 3 TLD registration lasts three days in this rehearsal. Read the actual expiry shown for your asset; changing settings does not rewrite an existing auction or lease.

YOU KNOW IT WORKED WHEN

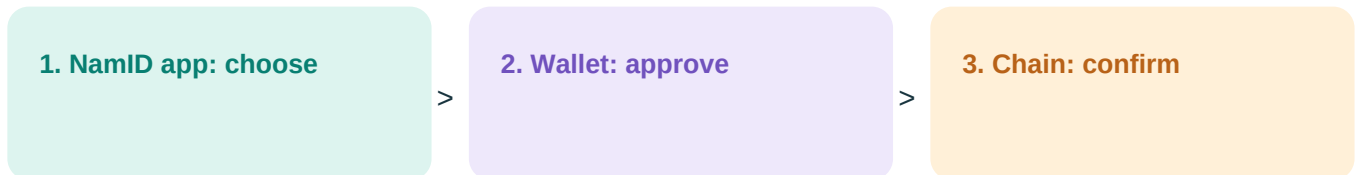
The website, wallet and payment address agree. Testnet receive addresses start with txch1.

KEEP THIS IN MIND

Main mode is the longer-term protocol configuration, not an announcement that mainnet sales are open. Do not send mainnet XCH to test anything here.

Meet the three places

The NamID protocol defines the naming system. The NamID app and wallet is where you bid, register and manage names, and where you approve signatures and transactions. MyMojos is the app built on NamID where you use your name. namid.net explains the project and links to evidence and documentation.



- 1 Open the NamID app on testnet for the current application rehearsal.
- 2 Use the NamID app's TLD/Auctions area for bids, My bids for follow-up, and Portfolio for owned assets.
- 3 Read each root's rules in the NamID app: issuance, pricing and renewal need not follow NalandaX.

YOU KNOW IT WORKED WHEN

You can tell whether you are choosing a name, signing a request, or checking a confirmed result.

KEEP THIS IN MIND

A listing on any site is not proof that the site holds your asset. Check the name in the NamID app, which reads the chain.

Prepare your wallet

A wallet contains the keys used to approve actions. Unlocking it makes those keys available to that wallet, not to every website. Back up the recovery phrase privately before using valuable assets. A wallet fingerprint is a convenient account label; it is not a payment address.



- 1 Open the installed NamID app and wallet and select the intended wallet. Check its name or fingerprint before every test.
- 2 Choose testnet11 and wait for peers and sync to become healthy. Confirm a receive address and spendable TXCH balance are visible.
- 3 Open the NamID TLD area. If a site needs external RPC, enable RPC in the wallet's settings; desktop wallet buttons use the app's own backend.

YOU KNOW IT WORKED WHEN

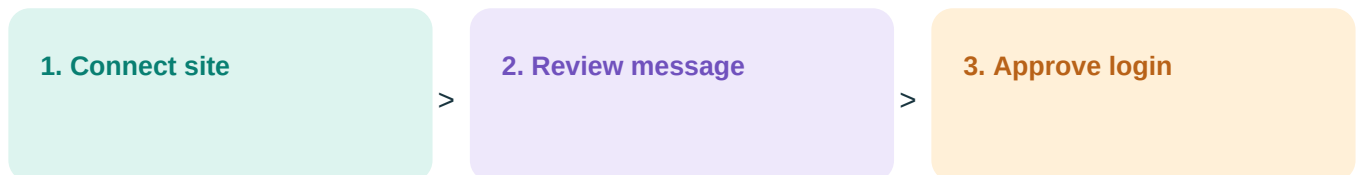
The wallet can show your current receive address and connected peers. A displayed old balance alone does not prove sync.

KEEP THIS IN MIND

Never paste your seed, private key, wallet database or RPC certificate into a website, support chat or this guide. Guardian public keys are a different thing.

Connect without giving away keys

Connecting lets a site ask your wallet for public account information. Signing in proves control of a key. Sending coins is a separate action. Treat these as three different doors, even if a website puts them next to each other.



- 1** Extension: open the wallet's browser extension, unlock or connect it, then choose it in the website sign-in menu. Approve only the intended site's request.
- 2** WalletConnect: choose the WalletConnect sign-in page, generate a pairing QR or copy the pairing link, then pair it from the wallet's WalletConnect settings.
- 3** Review the requested message and account. After approval, return to the website and check the displayed identity and network.

YOU KNOW IT WORKED WHEN

The site reports a signed-in identity. This does not by itself prove a particular name belongs to that identity.

KEEP THIS IN MIND

An unpacked extension update needs a reload in the browser's Extensions page, then a refresh of the website. Do not remove the extension or clear storage just to troubleshoot.

Choose an auction

A TLD is the final label in a name: x in yourname.x. An auction chooses who gets a root namespace under its recorded policy. Reserved company, DAO, gift and restricted names are not ordinary public auction inventory.



- 1** In the NamID app and wallet, open the TLD area and select a live auction or search a label. A scheduled name is not necessarily open yet.
- 2** Read the bid/reveal timeline and reserve. Your real bid must meet the displayed minimum; the blind does not count toward it.
- 3** Check whether the name is reserved or already owned. Use the DAO/reservations view for reserved names rather than trying to pay an auction address.

YOU KNOW IT WORKED WHEN

The auction is in the bidding phase, payment intake is available, and the wallet uses testnet11.

KEEP THIS IN MIND

No one should promise you a reserved gift merely because your preferred spelling appears in a list. Recipient verification and the authorized allocation process still apply.

Bid plus blind, explained

The bid is the most you are willing to pay for the name. The blind is extra money locked beside it to hide that bid's size. The public deposit is their sum. The blind does not help you win and is not an extra price charged for the name.



- 1 Enter your maximum price in Bid. Start with an amount you can leave locked until the auction is resolved.
- 2 Enter a blind amount, or zero. Read the total and network fee before you approve anything.
- 3 For illustration: a 0.03 TXCH deposit with a 0.015 TXCH clearing price returns 0.015 TXCH to the winner at settlement. Fees are separate.

YOU KNOW IT WORKED WHEN

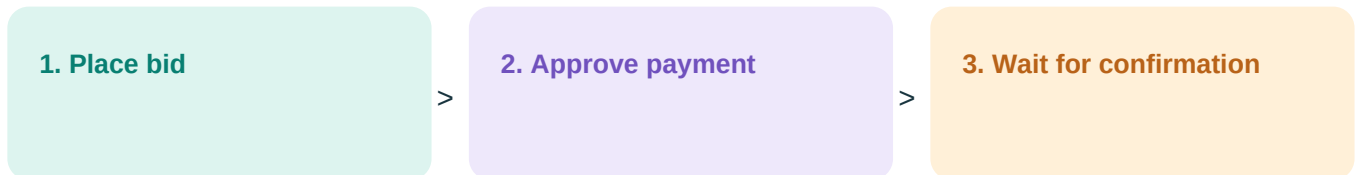
You understand the difference between maximum price, public locked total and eventual clearing price.

KEEP THIS IN MIND

These are examples, not suggested spending amounts. On testnet the auction service still picks the winner from the revealed bids; the coins limit where the money can go.

Place the bid once

The wallet prepares a protected bid deposit and shows a transaction review. The lock address exists because blockchain rules must hold the deposit until an allowed spend. You do not need to invent, manage or manually copy that address in the normal desktop flow. The refund destination is bound to your bid's wallet key.



- 1** Enter Bid and optional Blind, then choose Place bid. Read the selected name, amounts, network and fee in the confirmation.
- 2** Approve once. Open My bids and wait for the deposit's chain status to become live. Submitted and confirmed are different states.
- 3** If you cancelled before payment, Finish placing bid can resume an unpaid draft. If a payment attempt was recorded, inspect the wallet transaction and chain status before retrying.

YOU KNOW IT WORKED WHEN

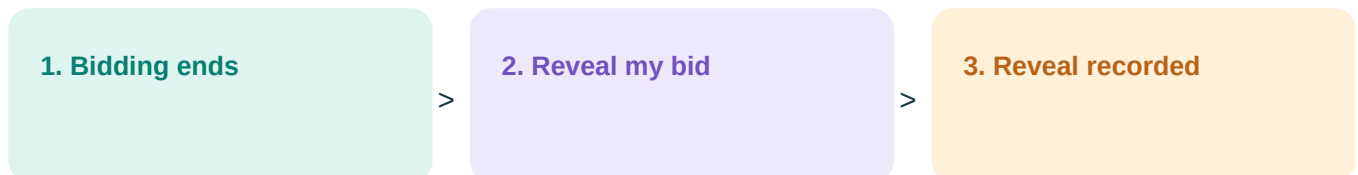
My bids reports a live deposit and the corresponding wallet transaction is confirmed.

KEEP THIS IN MIND

Do not send a second payment merely because the website was slow. A 502 can be a server fault; first check whether the wallet actually submitted a transaction.

Come back to reveal

During bidding, your real amount is hidden. During reveal, your wallet authorizes opening that bid. A payment alone does not complete the auction. Keep the wallet and its saved bid information available, and put the reveal deadline in your calendar.



- 1 Open the same wallet on testnet11 when the auction enters the reveal phase. Select the same name.
- 2 Choose Reveal my bid and review the message request. This proves control of the bid's key; it is not a request to fund another bid.
- 3 Check that the reveal was accepted. Keep the confirmation and return after the reveal phase ends to see the outcome.

YOU KNOW IT WORKED WHEN

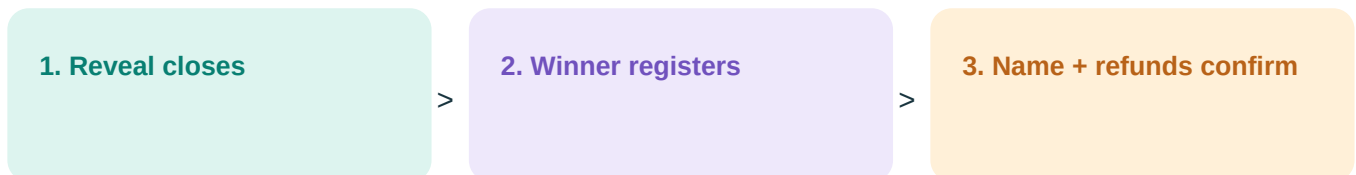
A reveal is recorded before the deadline. The deposit remains locked while the auction is live.

KEEP THIS IN MIND

Do not assume an unattended wallet reveals automatically. Missed or invalid reveals may require the documented abort/refund or timeout recovery path; they cannot be fixed by changing the old deadline.

Winner: register and receive change

For newly opened auctions requiring registration, the winner signs a registration request after reveal closes. The settlement is intended to create the name and return change together. The price is the greater of the reserve and the eligible second bid, subject to the current auction's rules.



- 1 After the reveal deadline, open the auction outcome. If you won, choose Register winning name & refund.
- 2 Review the name, price and refund destination. Approve the message, then wait for the settlement transaction to confirm.
- 3 Check Portfolio for the name and your wallet for the refund. Losing funded bids receive their refund through the settlement path; there is no separate refund button to press.

YOU KNOW IT WORKED WHEN

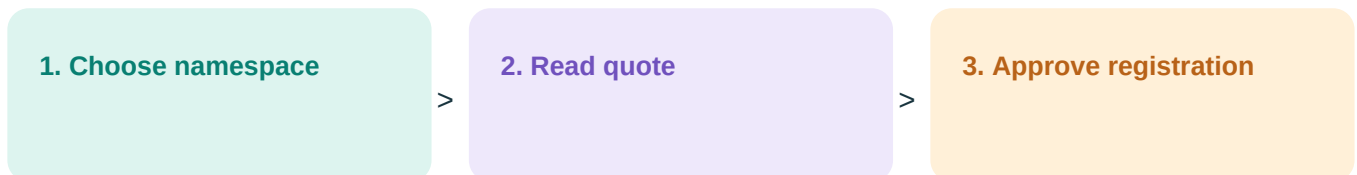
Confirmed chain evidence matches the name mint and refunds. A pending registration request is not a completed transfer.

KEEP THIS IN MIND

Older auctions may use the earlier automatic-settlement policy. If the winner never registers or the operator fails, recovery depends on the deposit's recorded timeout. Do not promise instant refunds.

Get a name under a TLD

A second-level name, or SLD, sits inside a root namespace. You do not need to win the root auction to register an available SLD offered by its operator. The root's issuance policy decides who can register, what it costs, and how long it lasts.



- 1 Open the NamID app's registration area or Get a NamID. Choose an available namespace and type the label you want.
- 2 Read the complete quote: price, network fee, term, transfer rules and any eligibility requirement. Free allocation does not necessarily mean zero network fee.
- 3 Use the supported wallet approval flow and check the resulting registration status. Keep the transaction receipt and verify the new name appears under the correct wallet.

YOU KNOW IT WORKED WHEN

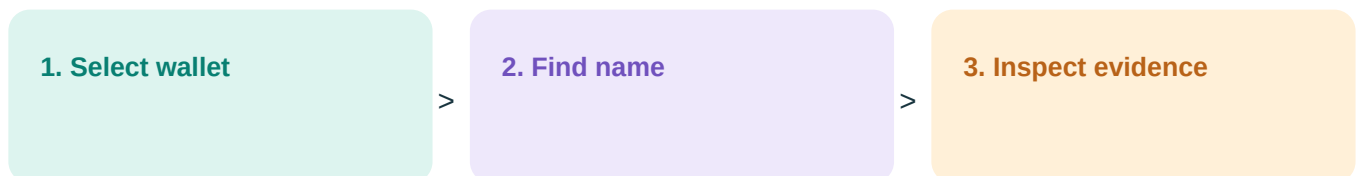
The UI shows the requested full name, an explicit policy and a confirmed outcome, rather than only a draft or reservation.

KEEP THIS IN MIND

A universal \$10 soulbound name and a universal three-year variable-price name are not both established across every live namespace. Do not infer these terms from an old pitch or note.

Read your portfolio

Portfolio collects names associated with the selected wallet. Some entries come from registry ownership records; others have independently recognisable on-chain asset data. Labels such as unverified, transferable and soulbound should mean different things, not different colours for the same claim.



- 1 Confirm the selected wallet and network, then refresh Portfolio. Separate TLD roots from names issued underneath them.
- 2 Open the asset details. Compare its launcher or current coin information with the transaction receipt when available.
- 3 Use Verify a name on-chain or Recognize for supported assets to let the wallet inspect their on-chain rules. Recognition is not a new purchase or a change of ownership.

YOU KNOW IT WORKED WHEN

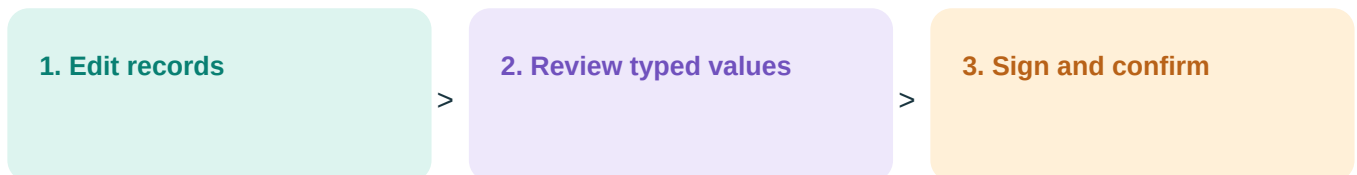
The displayed rules match the asset's actual covenant. A synthetic registry identifier is not presented as an on-chain transaction.

KEEP THIS IN MIND

If an asset is absent, check sync and the registry response before attempting to mint it again. Some custody/indexing paths still need verification; an unverified badge is an honest limitation.

Give your name useful records

Records connect a name to things people use: a wallet address, a website service, DNS data or a communication key. Editing records is an owner action. Typing someone else's name into a form does not give you permission to edit it.



- 1 Choose an owned name in Portfolio, then Edit records. Pick a supported record type and fill its specific fields.
- 2 For a wallet address, use the correct network. For website or nameserver data, enter a service you actually control; example values are not working infrastructure.
- 3 Review and sign the change. After confirmation, resolve the name again and compare the returned record and proof information.

YOU KNOW IT WORKED WHEN

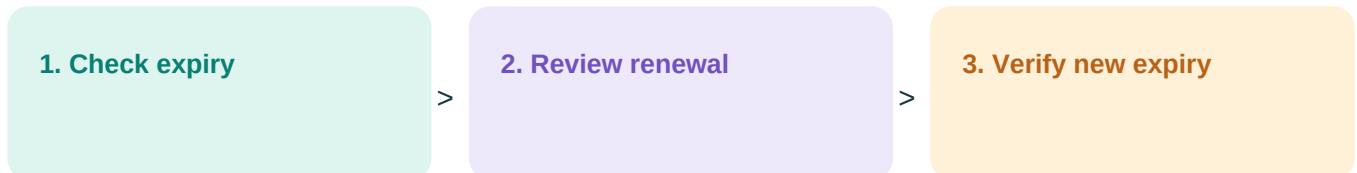
The new record resolves to the value you intended, with evidence for the current asset state.

KEEP THIS IN MIND

Namespace registration proceeds belong in its issuance or vault policy, not an ordinary name's contact records. Publishing a record alone does not make every browser resolve a non-ICANN name.

Renew before the clock runs out

A name can have a registration term even when it cannot be sold. Soulbound describes transferability; it does not automatically mean the lease lasts forever. The regtest clock is deliberately short so expiry and renewal problems show up quickly.



- 1** Read the actual expiry on the asset or namespace. The current rehearsal's TLD term is three days; read the SLD's own policy rather than assuming it matches.
- 2** Start renewal from the appropriate namespace or vault view. Review the amount, beneficiary and requested extension before paying or signing.
- 3** Wait for confirmation and recheck the expiry. Save the receipt if the balance changed but the displayed term did not.

YOU KNOW IT WORKED WHEN

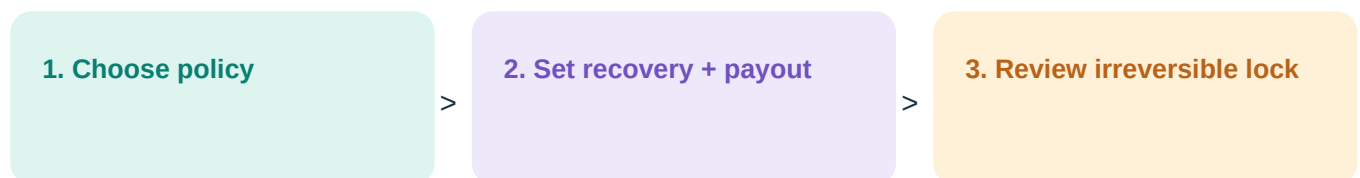
The authoritative lease or chain state shows the intended extension. A scheduled job or a success toast alone is insufficient.

KEEP THIS IN MIND

Automatic and perpetual renewal require working, funded execution. Do not rely on an unattended keeper until its specific path has been demonstrated. Keep a manual reminder during testing.

Vaults are optional commitments

There are two related ideas: a wallet vault for custody and a TLD vault for a namespace's rules. Locking a TLD can be an irreversible commitment that changes how its ownership is controlled. Ordinary SLD registration should not silently force you through a permanent TLD lock.



- 1 Open TLD Vault only when you intend to lock a supported owned root. Read its permanent/non-transferable behaviour before proceeding.
- 2 Set the namespace's SLD registration payout address in the lock/issuance policy. Check its network and spelling carefully.
- 3 Provide the required independent guardian public keys, review the complete policy and approve only after understanding recovery. Confirm the exact successor vault coin before treating the lock as complete.

YOU KNOW IT WORKED WHEN

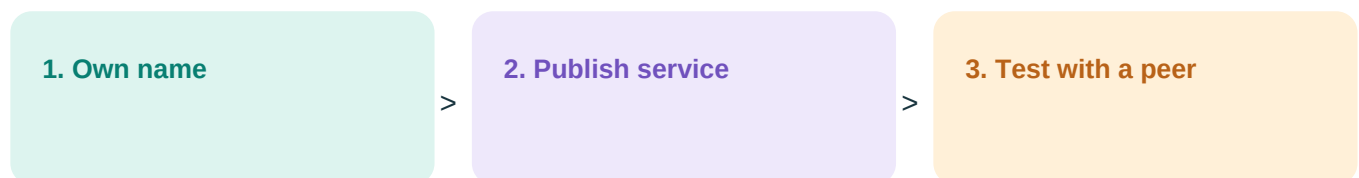
The lock is confirmed on-chain and the displayed policy, payout and recovery keys match your review.

KEEP THIS IN MIND

Two keys derived from the same wallet seed are not independent recovery. The existing .x test mint is being left unlocked; a recoverable lock test needs suitable independent public keys.

Set up a call by name

Calls need more than a name. They need a communication key, a reachable relay/service and a verified record tying that service to the name. The web Calls page is a setup flow; it does not prove that every relay or peer can already complete a call.



- 1 Sign in, open Calls and resolve your owned name. Supply a real configured relay endpoint rather than an example hostname.
- 2 Generate the communication key and register the service. Keep its private half on the device that will receive calls; publishing only the public half is intentional.
- 3 Commit the service record through the supported wallet flow, then test a call from another device or identity in the NamID app. Grant microphone access only for the intended call.

YOU KNOW IT WORKED WHEN

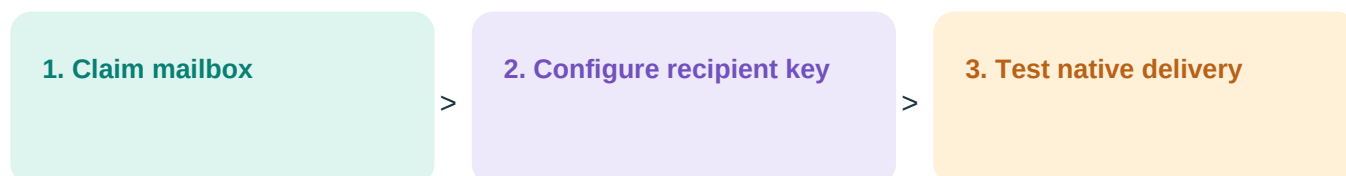
Both peers can discover the current service, establish a call and end it cleanly. A saved directory entry alone is not a successful call.

KEEP THIS IN MIND

Recovery of your naming wallet does not necessarily restore device-local communication keys. Back up or rotate those keys deliberately; do not publish their private values.

Mail: reservation versus delivery

NamID mail associates a mailbox with a name. The web claim page can reserve it; the NamID app and wallet has native name-mail interfaces. Ordinary internet email delivery, bridge aliases and IMAP/POP clients are separate rollout capabilities. A reservation is not evidence that external email works.



- 1** Sign in as the name's owner, open Mail and claim the mailbox. Read the returned state: reserved, pending and active are different.
- 2** Before sending sensitive native mail, verify the recipient has the expected communication key and understand the privacy shown by that client.
- 3** Test a short non-sensitive message between controlled names. Confirm receipt and reading on the recipient side; then test the reply.

YOU KNOW IT WORKED WHEN

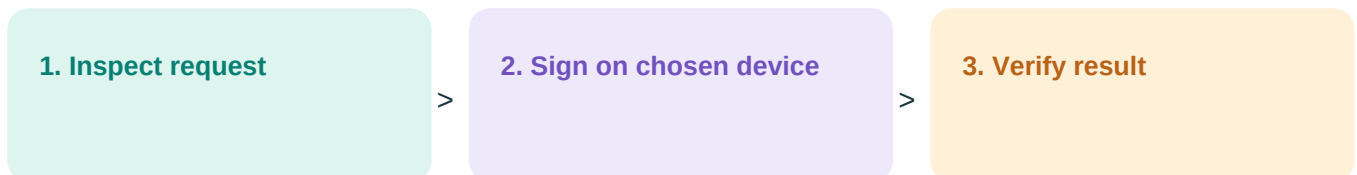
The recipient can actually read the message. A proposed bridge alias should not be advertised as a working public email address without a delivery test.

KEEP THIS IN MIND

Some current paths can fall back to plaintext when a recipient has no encryption key. Do not assume end-to-end encryption from the word mail. Service metadata may remain visible.

Sign, rotate and recover

Signing approves a particular message or transaction. Rotation replaces a control key. Recovery uses a prearranged guardian path when ordinary control fails. A separate signing device does not make an unreviewed transaction safe by itself.



- 1** For ordinary login, inspect the site and message. For a spend, inspect the actual destination, amount, network and fees.
- 2** For a supported offline flow, export the signing request, review it on the offline device, then return the signed result for submission.
- 3** For rotation or recovery, use the asset's supported action and respect its delay/cancel window. Verify the new key on the current successor before retiring the old setup.

YOU KNOW IT WORKED WHEN

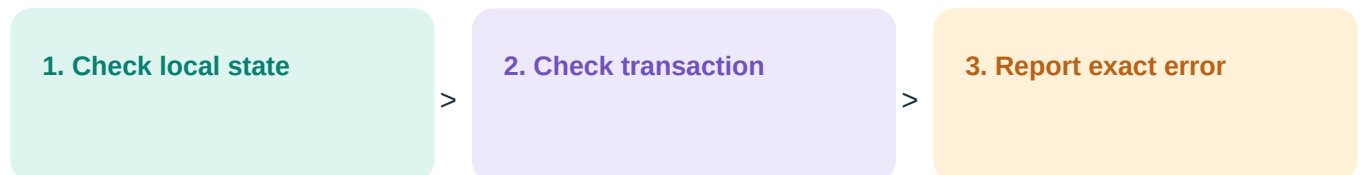
The signed bytes correspond to your intended action. Device separation never requires typing an offline seed into a website.

KEEP THIS IN MIND

Compatibility varies by asset and device. This guide does not claim every NamID covenant has completed an offline signing or guardian recovery test. Stop when the action cannot be decoded for review.

When a button fails

An error is information, not an instruction to pay again. Work from the smallest check toward the service. Keep enough evidence to distinguish a wallet problem, a browser connection problem and a transaction already sent to the chain.



- 1** No wallet detected: unlock the intended wallet, check testnet11 and sync, then reconnect. Extension absent: reload the installed extension and refresh the page.
- 2** WalletConnect stuck: make a fresh pairing, inspect the wallet approval and check the site's relay/configuration error. RPC certificate errors need a correct paired configuration, not disabled TLS checks.
- 3** 502 or payment uncertainty: inspect My bids and wallet history before retrying. Record the TLD, phase, time, public transaction ID and exact error. Never attach secrets.

YOU KNOW IT WORKED WHEN

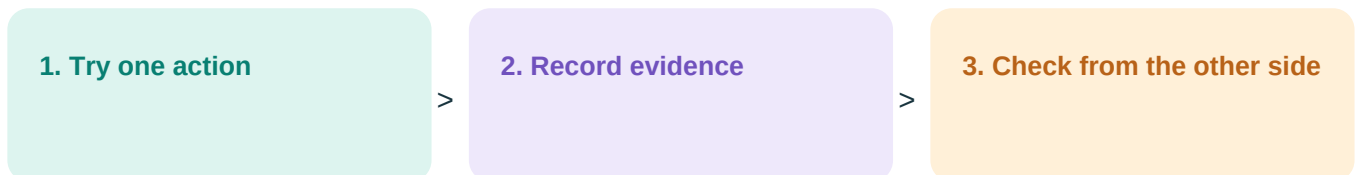
You can describe whether the request failed before signing, after submission or after confirmation.

KEEP THIS IN MIND

The September 26 bid-draft incident was a server thread-safety defect. Its repair does not guarantee that every future error has the same cause. Report the observed evidence.

Your first complete rehearsal

A useful test ends with evidence, not just a green button. Start with a single small scenario, record its outcome and then broaden it. The same discipline helps builders integrating NamID with another application.



- 1** Auction checklist: correct network, reserve visible, one confirmed bid, timely reveal, winner registration, confirmed name and expected refund.
- 2** Name checklist: clear issuance rules, verified ownership, a records edit that resolves correctly, and renewal with a verified new expiry.
- 3** Service checklist: a second peer receives a call or message. Identity integrations must verify current ownership and bind signatures to the app, network and fresh challenge; a display name is not authentication.

YOU KNOW IT WORKED WHEN

Keep public receipts and a list of failures. Read namid.net's validation notes and the repository's MAINNET-GAP.md before presenting readiness claims.

KEEP THIS IN MIND

Funding can support independent review, recovery testing and integration work. Passing local tests or a single funded demo is not an exhaustive audit. Guidance and reproducible evidence matter as much as a polished screen.