

NOSTR as technical infrastructure.

Relay <https://github.com/fiatjaf/pyramid>

Blossom <https://github.com/hzrd149/blossom>

Ngit <https://ngit.dev/install>

GRASP <https://ngit.dev/protocol/grasp>

Nsite <https://nsite.run/>

Napps <https://github.com/eskema/nostrapps>

FIPS <https://github.com/jmcorgan/fips>

This guide covers self-hosting the core components of the Nostr technical stack, from relays and media servers to Git hosting and mesh networking. You can run everything on a single VPS, a home server, or a hybrid setup.

Core Infrastructure Components

Relay: Pyramid

Pyramid is a community relay written in Go with a hierarchical invite system. Members invite others, creating a self-organizing ladder that scales membership to thousands.

- VPS Install: The simplest method is a one-line script: `curl -s https://raw.githubusercontent.com/fiatjaf/pyramid/refs/heads/master/easy.sh | bash`. A Docker image is also available.
- Home Install: Build from source with just build after cloning the repository. A systemd service ensures it starts on boot.
- Config: Environment variables set DOMAIN, RELAY_NAME, and your RELAY_PUBKEY. Membership is managed via users.json.

Blossom: Media Server

Blossom is a content-addressed blob storage server that uses Nostr keys for identity and serves files by their SHA-256 hash.

- Docker: Copy config.example.yml to config.yml, set the storage backend, then run `docker compose up --build`.
- From Source: Requires Deno 2.x. Clone the repo and run `deno task start`.
- Features: Supports S3-compatible backends, automatic media optimization (image resize, video transcode), and Nostr-signed authentication (BUD-11).

Git Hosting: ngit + GRASP

GRASP (Git Relays Authorized via Signed-Nostr Proofs) lets you host Git repositories where your Nostr keys define project identity, not a central forge.

- Install ngit: Use the one-line installer (`curl -fsSL https://ngit.dev/install.sh | bash`) or install via Cargo/Nix. Both ngit and git-remote-nostr must be on your PATH.
- Install GRASP Server: ngit-grasp is a single Rust binary. It combines a Git Smart HTTP backend with a Nostr relay for authorization.
- Workflow: Initialize a repo with ngit init, which publishes a signed announcement. Pushes are validated against signed state events, eliminating the need for external Git hooks.

Static Sites: nsite

nsite publishes static websites as content-addressed blobs on Blossom, indexed by a signed Nostr event. Any gateway can serve your site, and your npub is its address.

- Gateway: Run the nsite-gateway with Docker (`docker run -p 3000:3000 ghcr.io/hzrd149/nsite-gateway`).
- Config: Set LOOKUP_RELAYS, BLOSSOM_SERVERS, and PUBLIC_DOMAIN in the .env file.
- Deploy: Use the web deployer at nsite.run or the @nsyte/cli tool to publish your site.

Napps: Browser Launcher

Nostrapps is a browser-based launcher that runs static Nostr apps (nsites) in sandboxed windows. It fetches apps, caches them, and supports WebAssembly napps.

- Run Locally: Clone the repository and serve it with any static file server (e.g., `python3 -m http.server 8080`).

Networking: FIPS Mesh

FIPS (Free Internetnetworking Peering System) is a self-organizing encrypted mesh network built on Nostr identities. Your machine becomes a node, and existing IPv6 software runs over it unchanged.

- Install: Use the binary installer for your platform (.deb, .pkg, .ipk) or build from source with Rust.
- Config: Edit /etc/fips/fips.yaml to set a bootstrap peer, then start the service with `sudo systemctl start fips`.
- Overlay Mode: FIPS can tunnel IPv6 traffic over existing IP networks, allowing you to reach peers behind NAT or on different ISPs.



Maximizing Self-Hosting: Home vs. VPS

For a VPS Setup

- Minimum Specs: 2 GB RAM and 2 vCPU are sufficient for a relay, Blossom, and nsite gateway.
- Security: Use Nginx as a reverse proxy, obtain SSL/TLS certificates (Let's Encrypt), and configure a firewall (UFW).
- Persistence: Use systemd services for all daemons so they restart on boot. Store users.json, Blossom data, and Git repositories on persistent volumes.

For a Home Server

- Network: A static IP or dynamic DNS (DDNS) is needed for external access. Open ports 80/443 (or use a reverse tunnel like Cloudflare Tunnel) for the relay and Blossom.
- Storage: Blossom and Git repositories benefit from SSDs for speed. Consider a NAS-mounted volume for larger media blobs.
- Resilience: Run FIPS in overlay mode to create a private mesh between your home server and a VPS, enabling secure, encrypted access without exposing services directly.

Hybrid Optimization

- Split Services: Run the relay and Blossom on a VPS for high availability, and host Git repositories (ngit-grasp) and nsite gateways at home for storage-heavy workloads.
- Backup: Regularly back up users.json, Blossom's SQLite database, and Git repositories. GRASP's state is derived from Nostr events, so your keys are the ultimate recovery tool.

Integration & Maximizing Your Stack

- Unified Identity: Use the same Nostr keypair for Pyramid relay, Blossom, ngit, and nsite. Your npub becomes your universal identifier across all services.
- Media Pipeline: Upload images/videos to your Blossom server, then reference the Blossom URI in Nostr notes and nsite manifests.
- Git + Relay: Run ngit-grasp alongside Pyramid. Repository announcements and state events are stored on your relay, while Git data is served by GRASP.
- Mesh Security: Use FIPS to create a private network between your home and VPS. Access your relay, Blossom, and Git services over the mesh without exposing them to the public internet.
- Napp Hosting: Publish your own napps as nsites, then load them in Nostrapps. This creates a fully self-hosted app ecosystem where you control the launcher, the apps, and the data.

The key to maximizing self-hosting is redundancy: run multiple relays, mirror Blossom blobs across servers, and list multiple GRASP servers for your Git repositories. This ensures that no single point of failure can take your infrastructure offline.