

kip to main content [Accessibility help](#)

AI Mode

All

Images

Shopping

Videos

More

Move to GWS from SneakerNet Dev05

Here is the log for **Sneakernet Development Chats - v0.5**. This log details the design discussions around an offline, mesh-based communication protocol designed to route encrypted messages through physical data storage handoffs (flash drives, SD cards).



Stack Overflow +1

Project Architecture Overview

The core objective of the **Sneakernet Protocol** is to establish async message routing between isolated nodes ("Islands") without any traditional internet infrastructure. It relies on transient physical couriers ("Mules") carrying encrypted data packets.



Reddit +3

```

[ Island A ]                [ Mule / Courier ]                [ Island B ]
(Data Storage) ---> [Plugs in USB / Writes Bundles] ---> (Physically Travels) --->
[Plugs in USB / Unloads]

```

Chat Log: Version 0.5

Topic 1: Packet Lifecycle and De-duplication

- **Dev_Alice:** We need to update the bundle specification for v0.5. Right now, if a Mule visits the same node twice, we are re-indexing the exact same payload.
- **Dev_Bob:** We should implement a light-weight SQLite state tracker on the Island side. When a device mounts, the daemon reads the `manifest.json` header hash. If the hash exists in the local ledger, skip the read operation entirely to save disk write wear.
- **Dev_Alice:** Agreed. But what if the data bundle was only partially copied because the courier yanked the USB drive out too early?
- **Dev_Bob:** We will append a trailing `.done` checksum file to the bundle directory. The daemon won't treat the bundle as parseable unless the cryptographic checksum matches the index.
-

Topic 2: Routing Priority Algorithms (The "Carry Limit" Problem)

- **Dev_Charlie:** Mules are walking around with limited storage (e.g., cheap 32GB thumb drives), but our Island output spool is sitting at 200GB+. How are we deciding what gets loaded onto the courier's drive?



Stack Overflow +1

- **Dev_Alice:** For v0.5, let's implement an **Epidemic Routing Layer** with a TTL (Time-To-Live) and a priority flag:
 - **Priority 0 (Critical System/Health alerts):** Placed at the front of the queue, small byte footprint.
 - **Priority 1 (Direct User Messages):** Sorted by the closest known geographic proximity of the recipient's last seen node.
 - **Priority 2 (Bulk Data / Sync logs):** Populated using remaining drive space, sliced into 50MB chunks using a standard chunking utility.



Stack Overflow

- **Dev_Charlie:** Sounds solid. Let's make sure the chunking mechanism uses parity pieces (like old-school RAR/PAR archives). If a Mule loses one thumb drive on a multi-drive transit, the recipient island can still reconstruct the dataset using the parity chunks.
-

Topic 3: Security & Sanitization at the Air-Gap Boundary

- **Dev_Bob:** I'm still highly concerned about physical security. If someone drops a corrupted filesystem or a malicious payload onto a Mule's thumb drive, plugging it straight into the main Island terminal could compromise the node.
- **Dev_Alice:** For v0.5, we are introducing a mandatory hardware isolation gate. The thumb drive must first connect to a low-cost, disposable microcontroller running a hardened, read-only kernel.
- **Dev_Bob:** Good call. The micro-gate will:
 1. Verify the drive signature.
 2. Extract only valid file types matching our exact NNCP structured payload format.
 3. Block all raw dd system changes or hidden metadata custom OS folders (like .DS_Store or desktop.ini).
-

v0.5 Milestone Deliverables

- Define standardized manifest.json schema.
- Implement .done trailing cryptographic validation.
- Integrate Epidemic priority queuing algorithm into the core daemon.
- Construct the first hardware micro-gate prototype for untrusted media validation.
-

Would you like me to generate the **technical specifications** for the manifest.json payload structure, or would you prefer to explore the code framework for the **Epidemic routing queue**?
AI can make mistakes, so double-check responses

-  devnonsense.com

sneakernet - /dev/nonsense
Dec 26, 2024 — So transmission of binary data became physical, expressed in terms of SKUs, street addresses, shelf space. Software could be touch...
-  Hacker News

Dead USB Drives Are Fine: Building a Reliable Sneakernet
Sep 4, 2022 — We actually made a great solution for this in the. We made AOL bots that would take a file, chunk it into 1.5MB sections, and send...
-  Stack Overflow

Automatic message routing on a sneakernet

Feb 20, 2015 — Automatic message routing on a sneakernet. When it's plugged in, a piece of software starts up and reads the data from it, and can...

Show all