XI'AN · HYPERSCALE FOR RADIX

WEEK #14

10–16 Aug 2026

Hyperscale Weekly

Split the work, not the world

CURRENT MILESTONE

Milestone 2 · Radix Engine

Porting the Radix Engine so Scrypto runs across shards

Week 2

Previously: flightofthefox had unified the codebase onto one engine with zero external Radix dependencies and begun wiring the fee system and cross-shard settlement into the new foundation.

This was a big, wide-ranging week. flightofthefox overhauled how transactions prove they're allowed to act, brought post-quantum signatures online, and carefully spelled out exactly what happens to cross-shard transactions when a shard terminates. Milestone 2's Radix Engine feature-parity work is hitting its stride.

206	+21k −5.5k	427	1
-16% vs last wk	-17% vs last wk		
COMMITS	LINES CHANGED	FILES TOUCHED	DAY COMMIT STREAK

COMMITS / DAY



peak 60/day · 6/7 days active

WHAT FLIGHTOFTHEFOX BUILT

Badges take over from signatures

The VM's authorization model was rebuilt: holding a badge now grants permission where a raw signature used to, and accounts mint a proof they can present later. This brings the Radix Engine's access control to the sharded VM.

'see "The Journey" on hyperscale.rs

Post-quantum crypto goes live

ML-DSA-65 post-quantum signatures and secp256k1 are now wired through the transaction path, and the address system was rebuilt with protocol-hash-pinned derivations. flightofthefox noted parity with industry post-quantum efforts.

'see "The System" on hyperscale.rs

The dead shard's will, fully spelled out

Dozens of commits hardened the rules for cross-shard transactions caught when a shard terminates — evidence windows measured in beacon epochs, settled-set serving windows, and boundary records that rebuild accounts a replay never reached.

'see "The Will" on hyperscale.rs

THE BOTTOM LINE

STANDOUT CHANGE

The authorization model now uses badges and proofs instead of raw signatures — a fundamental shift bringing the Radix Engine's access control to the sharded VM.

WHAT'S NEXT

Likely next: continuing to port Radix Engine resource types like non-fungible tokens and collections, and wiring the fee system deeper into the now-badge-aware authorization path.

Written by AI from public commits, the community chat and hyperscale.rs — may contain mistakes.

From the community

What people are saying, and the words behind the work.

HEARD IN THE CHAT

The mood: Chat buzzed about AI-powered security auditing and whether binary-field cryptography might change hash-function choices, with flightofthefox reassuring that a hash swap is trivial and red-teaming shou

> *"it will be a very fun exercise at the first testnet, for everyone to point their favourite frontier models at the code and try to hack their way to king of the hill 😅"*
>
> — FLIGHTOFTHEFOX, IN THE COMMUNITY TELEGRAM

THIS WEEK'S CONCEPT

The Journey — *One payment crosses a sharded world.* The authorization model is the heart of Milestone 2's Babylon feature-parity goal — badges, proofs, and stored rules are how the Radix Engine controls who can do what, and this week that system took shape on the sharded foundation.

hyperscale.rs/journey

JARGON, DECODED

ML-DSA-65 — A post-quantum signature standard approved by NIST, designed to resist future quantum computers.

Badge — A digital credential stored on-chain that grants permission to perform operations, replacing the need for a raw signature each time.

Securify — Upgrading an account so stored logic rules govern its authorization instead of a simple signature check.

Written by AI from public commits, the community chat and hyperscale.rs — may contain mistakes.

ZOOM OUT

The bigger picture

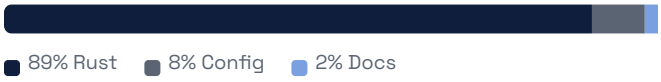
Where the work landed, and the map to read along on hyperscale.rs.

MOMENTUM · LINES CHANGED / WEEK



last 8 weeks · newest right

THIS WEEK'S WORK



89% Rust 8% Config 2% Docs

WHERE THE WORK LANDED

types · The System
address system and signature scheme overhaul

node · The Will
straddler settlement and terminal handoff hardening

vm/manifest-builder · The Journey
transaction building and authorization wiring

THE FULL PICTURE

The System
The whole sharded design at a glance.

The Journey
One transaction's trip through the network.

The Clock
Consensus-attested time across independent shards.

The Census
The leaderless beacon of validators, stake and shards.

The Overlap
Why two conflicting blocks can never both commit.

The Generals
All-or-nothing cross-shard commits, computed not voted.

The Archive
Every published byte is preserved or provably expired.

The Library
All state in one merkle tree; a shard is a subtree.

The Will
How in-flight transactions settle when a shard dies.

The Lottery
Random, ever-moving committees; proven cheats are jailed.

The Triage
Under overload, urgent traffic never waits for bulk.

The Governor
Validator entry is re-priced every epoch, like a market.

The Crash Lab
The simulator that replays any failure byte-for-byte.

The Proof
Proving safety with maths, not just tests.

The Asterisks
Every design's trade-offs — including Hyperscale's own.

THE ROAD TO MAINNET

M1	M2	M3	M4	M5	M6
Adaptive Sharding done	Radix Engine ~5 mo	Gateway & API ~3 mo	Validator GUI ~3 mo	Migration ~3 mo	Live support 12 mo

Now: Milestone 2 · Radix Engine — Week 2. Since day one: 3,174 commits — one developer, fully in public.

LINKS

WEBSITE
hyperscale.rs

DISCUSS THIS ISSUE ON X
x.com/i/status/2089216311196872951

CODE (PUBLIC)
github.com/hyperscalers/hyperscale-rs

MILESTONE PROPOSAL (PDF)
hyperscale.rs/xian_proposal.pdf

COMMUNITY (TELEGRAM)
t.me/hyperscale_rs

AUTHOR
flightofthefox

scan for hyperscale.rs

Questions? flightofthefox is always happy to discuss in the community Telegram — t.me/hyperscale_rs.

Written by AI from public commits, the community chat and hyperscale.rs — may contain mistakes.