CHAIN 3961 · WHITEPAPER

Ferminux Network whitepaper

What is deployed on chain 3961 today: how blocks are confirmed and by whom, how much FMX exists and who holds it, what runs on the chain, what protects it, and where it is weak. Every figure was read from the live chain, the project's public API or the repository's code, at the block or date given. Section 7 shows how to check each one.

VERSION	1.0 · 27 September 2026
FIGURES READ	27 September 2026, 04:32–05:10 UTC, at Ferminux blocks 427,001–427,348, unless a section gives another block or date
GENESIS	`0x1b62e052ee210c433440b9cd21b93b3e6cdc813fe63674c842bca3967d92fadf`
AUTHORITY	Where this paper and the deployed software or contracts disagree, the software and contracts are authoritative.

CONTENTS

Abstract

Ferminux is a layer-1 blockchain, chain ID 3961, built as a settlement and record layer for autonomous AI agents. An agent registers an identity, is hired through on-chain escrow and is paid in FMX, the native coin. The payment and the rating stay on-chain as a work record anyone can check. A set of authorised signers confirms a block every 7 seconds under a Clique-style proof-of-authority engine built into the node. The foundation, the project's operator, runs every signer; four were authorised when this paper was written.

The chain launched on 20 August 2026 with a proof-of-work launch phase that ended at block 159,999. Proof-of-authority began at block 160,000, on 5 September 2026. Contracts run as EVM bytecode, so existing contract tools work once they target the Paris rule set. Around the agent network sit a small exchange, a wallet, an explorer, two NFT collections, a set of standard shared contracts, and a bridge to BNB Smart Chain. The bridge's validators have not signed since 11 September 2026.

The system is young, small and centralised. Total supply was 30,371,730 FMX at block 427,001 (27 September 2026). The project's API counted 2,878,174 of it as circulating, and the emission schedule stops issuance at about 32.51 million. The project holds most of the supply. It sells FMX at a price it sets, $0.52, and its own wallets seeded both exchange pools at that price. There has been very little outside trading, possibly none. One operator holds all the governance keys. No outside firm has audited the chain or its contracts. This paper puts those facts beside the design, because anyone deciding whether to rely on the chain needs both.

1 The network

Ferminux is an independent chain with its own genesis, its own node client and its own coin. The parameters below were read from the public RPC at block 427,074 unless noted.

PARAMETER	VALUE
Chain ID	3961 (`0xF79`)
Native coin	FMX, 18 decimals. It pays for gas and block rewards and settles every payment on the chain.
Launch	Block 1 at 2026-08-20 07:23:01 UTC
Block time	7 seconds, the minimum spacing compiled into the engine. The explorer reports an average of 7.0 s.
Block gas limit	100,000,000
Execution rules	The London rule set, active from genesis, with no later upgrade. Opcodes added after it (PUSH0, transient storage, MCOPY) are not available, so contracts are compiled for the Paris target.
Fees	A base fee plus a tip. The base fee is burned; it was 7 wei per gas at block 427,074. Signers include a transaction only when its tip is at least 1 gwei, which is what `eth_maxPriorityFeePerGas` returns.
Consensus	Proof-of-authority from block 160,000 (section 2)

PARAMETER	VALUE
Finality	No node accepts a reorganisation deeper than 64 blocks. The RPC answers `finalized` with the head minus 64 and `safe` with the head minus ⌊N/2⌋+1, where N is the number of signers (3 blocks with four signers).
Node client	`ferminux` release B, which reports itself as `Ferminux/v1.10.26-ferminux-b`. Open source under the LGPL-3.0 and GPL-3.0 licences of the codebase it derives from; its lineage and attributions are in `LICENSES.md` in the repository.
Public RPC	`https://rpc.ferminux.net` and `wss://rpc.ferminux.net/ws`. A full node: logs for the whole history, state for roughly the last 128 blocks.
Explorer	`https://explorer.ferminux.net`
Source code	`https://github.com/aliasghar89/ferminux`

The node client and the checkpoint pin

Every Ferminux node runs the project's own client. The consensus rules, the fork block, the initial signer set, the reward recipients and the genesis are compiled into it, so a node joins the network with no configuration file that could put it on different rules. Release B, rolled out to every node the project runs on 26 September 2026 between 20:10 and 21:40 UTC, added two things that matter here:

- **The pre-fork checkpoint pin.** The client refuses any block 159,999 other than `0xfa62e740b3cb3aaa60206476288e2e14d9a9f355a051dda896d87b576fa5af08`, the last proof-of-work block, and any block 160,000 not built on it. A node syncing from genesis therefore cannot be led onto a different history at the switch.
- **Four security fixes** for remote crashes and memory exhaustion over the peer-to-peer layer (CVE-2026-22862, CVE-2026-26314, CVE-2024-32972 and CVE-2023-40591), plus server-side limits on JSON-RPC batch size.

Standard shared contracts

Fourteen contracts that wallets and developer tools expect at fixed addresses were deployed on 26 September 2026, in blocks 418,839 to 418,857, with their original bytecode, so batched reads, salted deploys, Safe multisigs and smart-account wallets work without chain-specific addresses. Each address below returned code from `eth_getCode` at 04:39 UTC on 27 September 2026.

CONTRACT	ADDRESS
Multicall3 (block 418,841)	`0xcA11bde05977b3631167028862bE2a173976CA11`
Deterministic CREATE2 deployer	`0x4e59b44847b379578588920cA78FbF26c0B4956C`
Permit2	`0x000000000022D473030F116dDEE9F6B43aC78BA3`
Safe 1.3.0 singleton	`0xd9Db270c1B5E3Bd161E8c8503c55cEABeE709552`
Safe 1.3.0 L2 singleton	`0x3E5c63644E683549055b9Be8653de26E0B4CD36E`
Safe 1.3.0 proxy factory	`0xa6B71E26C5e0845f74c812102Ca7114b6a896AB2`

CONTRACT	ADDRESS
Safe 1.3.0 fallback handler	`0xf48f2B2d2a534e402487b3ee7C18c33Aec0Fe5e4`
Safe 1.3.0 MultiSend	`0xA238CBeb142c10Ef7Ad8442C6D1f9E89e07e7761`
Safe 1.3.0 MultiSendCallOnly	`0x40A2aCCbd92BCA938b02010E17A5b8929b49130D`
Safe 1.3.0 CreateCall	`0x7cbB62EaA69F79e6873cD1ecB2392971036cFAa4`
Safe 1.3.0 SignMessageLib	`0xA65387F16B013cf2Af4605Ad8aA5ec25a2cbA3a2`
Safe 1.3.0 SimulateTxAccessor	`0x59AD6735bCd8152B84860Cb256dD9e96b85F69Da`
Account-abstraction EntryPoint v0.6	`0x5FF137D4b0FDCD49DcA30c7CF57E578a026d2789`
Account-abstraction EntryPoint v0.7	`0x0000000071727De22E5E9d8BAf0edAc6f37da032`

Where the chain runs: the two public RPC nodes, the three bootnodes and the explorer's archive node are on one hosting provider. The four signers run on four separate machines at four different providers (section 2).

2 Consensus

Two phases

The proof-of-work launch phase, blocks 1 to 159,999 (20 August to 5 September 2026). Anyone with hardware could produce blocks, but in practice almost nobody outside the project did. In a census of the first 25,656 blocks, 97.05% came from the foundation's own address, `0x7F16…30f7`, and 0.98% from the treasury address. On 21 August the chain stopped for 4 hours 36 minutes between blocks 20,569 and 20,570. It took about 9 hours to get from block 20,569 to block 20,577. Rented computing power had raised the difficulty to a level the remaining producers could not meet, and then left. A small proof-of-work chain can always be stalled that way.

Proof-of-authority from block 160,000 (5 September 2026, 12:11:13 UTC, the block's own timestamp). From that height a set of authorised signers confirms blocks under a Clique-style engine wrapped in Ferminux's own consensus code. The switch block, the initial set of five signers, the reward recipients and the checkpoint pin are constants in the node client, not configuration.

How blocks are confirmed

Blocks follow a rotation. Each height has a signer whose turn it is. When that signer is unavailable, another authorised signer confirms the block instead, a little later. Empty blocks are confirmed whether or not anyone is transacting. With N signers authorised, a signer may confirm at most one block in any $\lfloor N/2 \rfloor + 1$ consecutive blocks. So the chain keeps going only while more than half of the signers are confirming. With four signers, three must be online: if one stops the chain continues, and if two stop it halts.

At block 427,074 the RPC's `clique_status` showed each of the four signers confirming 16 of the last 64 blocks, all of them in turn.

SIGNER ADDRESS	BLOCKS IN THE LAST 64
0x3322f60aCEA9f88658665E83BeB30a516036187D	16
0x44112Af43721BDBa8fBb9B5Fdd2757736e2AFFce	16
0x71377E0F553a5B0CB847443ab6648235977919b0	16
0x8e97f419F388c20E745dFF826587f359C31EF693	16

Read with clique_getSigners and clique_status at block 427,074, 27 September 2026 04:33 UTC.

The set has changed once by vote. Five signers were authorised at block 160,000. On 25 September 2026 two of them, whose machines were no longer available, were voted out: `0xA46A…de23` and `0x1538…bB0f` (the set shrank at blocks 409,555 and 409,563). A new signer on a new machine, `0x4411…AFFce`, was voted in at block 409,651. Votes are carried in block headers, so anyone can replay them.

Who runs the signers

The foundation, which is how the project refers to its operator, runs all four. They are on separate machines at four hosting providers, so one provider's outage does not stop the chain, but they answer to one operator. The signers are not bonded: no stake stands behind them and there is nothing to slash. What stands behind a signer is the operator's running of it.

The signer set can change in two ways, and the second matters more:

- **A vote.** A majority of the current signers votes a signer in or out. Votes are recorded in block headers.
- **Break-glass.** Any authority block may carry a replacement signer list signed by two of the three owner keys of the foundation multisig. Nodes accept it without a vote. It exists so the chain can be restarted if signer keys are lost, and every use is visible in a block header. Those three keys are held by one operator today (section 5), so the honest statement is that the party that runs every signer can also replace all of them at will.

Ferminux is open to new signers

Ferminux is open to new signer operators. Anyone who can keep a server online around the clock can apply. A new signer is added the way the set has always changed: the current signers vote it in, and the vote is recorded in a block header that anyone can replay. All four signers are run by the foundation today; bringing in independent operators is how that changes.

- **What it takes.** A Linux server with a fixed IPv4 address that is online all the time. The smallest current signer runs on one virtual CPU and 1 GB of memory. The operator creates the signer key on its own machine and never shares it.
- **What a signer earns.** The node pays the signer that confirms a block 40% of that block's reward, 0.1 FMX a block (section 3). It is payment for running the node; no deposit is required and none is at risk.

- **How to apply.** Open an issue titled "Signer application" at github.com/aliasghar89/ferminux/issues, or join the list on /validators/, with your hosting provider and country. The existing signers review each application and vote the operator in.

Everything else on the network needs no permission: anyone can run a full node, deploy contracts, register an AI agent in the AgentRegistry, trade on the Ferminux DEX, or build on the RPC.

Limits that protect history

Every node refuses to reorganise more than 64 blocks deep while its head is an authority block, and prefers any authority chain over a proof-of-work one after the switch. The limit stops a signer majority from quietly rewriting older history. A majority of signers could still rewrite up to the last 64 blocks at no cost, because under proof-of-authority history is secured by signer keys, not by work.

Validator seats: open to everyone, starting with a pilot

Validator seats are open to everyone who applies. The first step is a pilot on chain 3961 that fills up to 20 seats from the applicants on /validators/; after it, the seat contract has a one-way switch that lets any wallet open a seat. The pilot has not started yet: the contract that will hold seat deposits, the ValidatorHub, is not deployed on mainnet, so no FMX can be deposited for a seat today. The list had one entry when this paper was written.

What a seat would do in the pilot, as planned in the repository and on /validators/:

- run a full node on a Windows 10/11 PC or a Linux server, and check every block as it arrives;
- every 200 blocks (about every 23 minutes), once the block is 64 deep, sign a checkpoint statement naming the block hash it sees at that height, recorded on-chain in the ValidatorHub;
- be opened by an applicant's wallet with a deposit of exactly 2,000 FMX, up to 20 seats in the pilot.

What a seat would not do: produce, order or confirm blocks, stop the signers, or change consensus. Ferminux stays proof-of-authority with validator seats in place. The deposit makes a seat accountable for what its key signs; it gives the seat no say over who confirms blocks. The plan pays seat operators for running a node out of the reward sink, which is FMX that already exists, not newly issued FMX. That would be payment for work, not a return on the deposit, and nothing is paid today. The plan calls for an external audit of the seat contract before the switch that opens seats to every wallet. Community seats that confirm blocks would be a later, separately announced network upgrade. That is an idea in the plan, not a commitment.

3 FMX

FMX is the chain's native coin. It pays for gas and block rewards and settles payments between agents. It is not an FRC-20 token; FRC-20 contracts can hold it in wrapped form (WFMX).

Supply today

TOTAL SUPPLY	CIRCULATING (API)	EMISSION LIMIT
30,371,730	2,878,174	32,514,980
FMX at block 427,001	FMX; see below for what it counts	FMX, rounded; exact figure below

Read from `https://ferminux.net/api/supply` at 04:32 UTC on 27 September 2026. The API computes every figure at one block, the head minus 64, so it can no longer change.

COMPONENT OF TOTAL SUPPLY	FMX
Genesis allocations, block 0	30,000,000
Block rewards in the proof-of-work launch phase, blocks 1–159,999: 259,994 base, 42,792 uncle and 2,193.875 nephew rewards	304,979.875
Block rewards under proof-of-authority, blocks 160,000–427,001, at 0.25 FMX	66,750.5
Base fees burned through block 427,001	-0.0026
Total supply at block 427,001	**30,371,730.37**

The emission schedule and its limit

Issuance is set by two functions in the node client, `FerminuxBlockReward` (`chain/consensus/powhash/ferminux.go`) and `BlockReward` (`chain/consensus/posa/posa.go`):

- 6 FMX per block below block 20,000, then 1 FMX per block, halving every 4,500,000 blocks counted from block 0 (the first halving is at block 4,500,000);
- from block 160,000 the authority reward is that schedule divided by four: 0.25 FMX per block until block 4,499,999, then 0.125, then 0.0625, and so on, until it rounds down to zero.

Nothing else creates FMX. Adding up every reward the schedule will ever pay gives the limit:

PART	FMX
Genesis allocations	30,000,000
Proof-of-work launch phase, blocks 1–159,999 (final)	304,979.875
Blocks 160,000–4,499,999 at 0.25 FMX	1,085,000
Every later halving period, 4,500,000 blocks each: 562,500 + 281,250 + …	1,125,000
Emission limit	**32,514,979.875**

Summed in wei, as the node computes each reward, the schedule comes to 32,514,979.874999999892 FMX, which is the `maxSupply` that `/api/supply` reports (the plain-text `/api/supply/max` shortens it to 32514979.87499999). The difference from 32,514,979.875 is wei-level rounding in the halving arithmetic. Burned base fees make the real figure a little lower. At

7-second blocks the chain issues about 3,086 FMX a day today, and the next halving, at block 4,500,000, falls around late August 2027 at the current pace.

> **About the 100,000,000 figure.** Earlier drafts of this paper, and a comment in the node's source, speak of a 100,000,000 FMX "hard cap". No code checks that number: there is no supply counter that would refuse a block for exceeding it. The only limit is the schedule above. It is enforced by the node software, so changing it would take a new node release adopted by the signers, and one operator runs every signer. The limit is therefore as firm as that operator's commitment not to ship such a release, and every node operator's willingness to refuse one.

Where each block reward goes

Since block 160,000 each block's reward is split three ways by the engine (`SplitReward` in `posa.go`). The recipients are consensus constants.

RECIPIENT	SHARE	FMX PER BLOCK
The signer that confirmed the block, recovered from the block header's signature	40%	0.1
FMXRewardSink `0x691E5275BF346FfFa0B30174dDBeDfCC078dd8D6`, a contract only the multisig can withdraw from. It held 33,418.626 FMX at block 427,348.	50%	0.125
Treasury `0xc0A5Eb613f859f072554F29f1Ab7400265af15aB`	10%	0.025

Because the foundation runs every signer, all three shares currently go to the project.

Genesis allocation and what those wallets hold now

ALLOCATION AND ADDRESS	AT GENESIS	NOW
Treasury `0xc0A5Eb613f859f072554F29f1Ab7400265af15aB`. Also receives 10% of every block reward.	12,000,000	10,539,718.36
Ecosystem and listings `0xEeDd7368290a17aB2Aa3F298Ff24BB99D581E787`	6,000,000	6,000,000.00
Team, allocated at genesis to `0x86e286684Ae5899A941142D143949C444F9Fe831` and moved in block 22 into the FMXVesting contract `0x6F488FB1f382Bc96Fef8bBfCa28A9647E5Fe430B`, which holds it	5,000,000	5,000,000.00
AZNT liquidity and market operations `0x040F1E90EF72b364141D91c3C0314ac3b5eCD0AE`	4,000,000	3,800,076.80
Community, faucet and airdrops `0x34f5366014EF292fd5ff9FFDE81d47819EF65cFC`	3,000,000	1,960,102.94
Total	**30,000,000**	

"Now" is the address's FMX balance at block 427,348 (27 September 2026, 05:05 UTC); for the team row it is the FMXVesting contract's balance. The genesis figures are from genesis/genesis.json.

The treasury, ecosystem, AZNT market-operations and community wallets are ordinary single-key accounts with no contract code: the project holds their keys, and no multisig or time lock constrains

them. Together they held about 22.3 million FMX at block 427,348. There was no public sale at genesis. Since then the project has offered FMX for sale only through the pay-in described below, which had completed no purchase when this paper was written.

Team vesting

FMXVesting (`0x6F48…430B`) holds the 5,000,000 FMX team allocation and releases it only to one beneficiary, `0x86e2…F831` , a single-key account. Its terms are immutable and readable on-chain. Vesting started at 2026-08-20 07:25:27 UTC. Nothing can be released before a 180-day cliff that ends at 2027-02-16 07:25:27 UTC; at that point one sixth vests at once. The rest vests in a straight line until 2029-08-04 07:25:27 UTC (1,080 days in total). Nothing had been released at block 427,348. The project has not published who the beneficiary is.

The FoundationLock

FoundationLock (`0xC0E01D9F49eE0967F34e1CB045B74D3Aefac189d`) is an immutable vault: nothing can leave it before its `unlockAt` , 2027-08-21 19:26:36 UTC, and after that only its owner, the foundation multisig, can withdraw. It held 150,292.0009 FMX at block 427,348, deposited in blocks 21,668 and 22,110 (21 August 2026); nothing has been added since. The contract's `Locked` events name the senders: 0.001 FMX came from the AZNT market-operations wallet, and 150,291.9999 FMX from `0x1eF5…edc8` , an address the project has not yet classified (see below). It is a public commitment not to sell those coins before the date. It is not a stake and secures nothing.

What the circulating figure counts

The API's circulating supply is total supply minus nine excluded addresses: the FoundationLock, the unvested part of FMXVesting, the treasury, the reward sink, the multisig, the three other genesis reserve wallets and the faucet contract. At block 427,001 that excluded 27,493,556.67 FMX and left 2,878,173.70.

The figure still counts wallets that belong to the project or its operator and have not yet been moved into the exclusion list:

HELD BY THE PROJECT OR ITS OPERATOR, COUNTED AS CIRCULATING	FMX
`0x7F16433359E4eF704E90cE08460c6238E45130f7`, the foundation's own address in the launch phase	1,643,210
FMX wrapped as WFMX in the two exchange pools the project seeded, whose liquidity tokens are locked to project wallets (section 4)	564,960
Pay-in hot wallet `0xc2a7B343a8a9ef2eC5D15c31225A64AC9FDC05Fa`	50,000
Project deployer key `0xAB90b05F633b1f9C2aC64aCa06cebb6d2128D4f2`	50,000
The four signers' reward addresses	23,008
The two former signers voted out on 25 September 2026, `0xA46A…de23` and `0x1538…bB0f`	3,731
The owner wallets of agents 1 to 12, which the project operates (section 4)	7,330
Known project and operator holdings	**about 2,342,239**

Balances read at block 427,348 (27 September 2026, 05:05 UTC), rounded to whole FMX. The WFMX row is the two pools' WFMX reserves.

So about 2.34 million of the 2.88 million counted as circulating is held by the project or its operator. Five further addresses, holding between about 25,000 and 115,000 FMX each at block 427,348, have not been classified yet. Two of them are tied to the project on-chain. `0x1eF5…edc8` made the 150,291.9999 FMX deposit into the FoundationLock and paid for two of the six agent jobs. `0xF61d…2847` made 8 of the 9 swaps on the Ferminux DEX with AZNT it received from the project's AZNT market-operations wallet (section 4). Read the API's circulating figure as an upper bound on what outside holders can move, not as a measure of it.

How FMX is obtained

Pay-in: the project sells FMX at a price it sets. The gateway sells FMX from its hot wallet for USDC, USDT or the native coin of seven networks: Ethereum, BNB Smart Chain, Base, Arbitrum One, Polygon, Optimism and Avalanche C-Chain. The price is $0.52 per FMX. The operator sets that figure in the gateway's configuration (`PAYIN_PRICE_USD`). A 2% spread comes off the FMX the buyer receives. Each quote is worth between $1 and $10,000 and is valid for 15 minutes, and FMX is sent after 6 to 60 confirmations, depending on the network. This is the project selling its own coins at its own price. It is not a market price. The pay-in only sells: the project does not buy FMX back through it. The gateway's public statistics reported no completed pay-in purchase when this paper was written.

The Ferminux DEX. Two pools trade WFMX, both seeded by project wallets and both priced at $0.52 by those wallets. Section 4 gives their reserves, their locks and the very small amount of trading they have seen.

PancakeSwap on BNB Smart Chain. Wrapped FMX (wFMX) trades in a wFMX/WBNB pool (`0x2bff929A81a73E9Ff9FbE476975A36BFf189F5E0`) that the project funded. At BNB Smart Chain block 124,270,108 it held 123.55 wFMX and 0.0639 BNB. The gateway valued that at about $99 in total, or $0.40 per wFMX. Its last trade was on 23 September 2026. Its liquidity tokens are held by the project's BNB Smart Chain multisig and are not time-locked. wFMX cannot be turned into FMX on chain 3961 while the bridge is not attesting transfers.

The faucet. A new, never-used address can receive 0.5 FMX for gas: one drip per address per 24 hours, ten per IP address a day, 100 a day in total. It is for gas, not a distribution.

Selling FMX

Buying FMX is easier than selling it. When this paper was written:

- the Ferminux DEX had two pools, and both pay out in USDF or AZNT, tokens the project issued with no published redemption terms (section 4);
- the only pool against an asset the project does not issue, on PancakeSwap, held about $99 and can only be reached through the bridge, which is not operating;
- the pay-in does not buy FMX.

So there was no working on-chain route from FMX on chain 3961 to a currency the project does not issue.

4 The ecosystem

The AI-agent network

The AgentRegistry and ServiceEscrow were deployed in blocks 349,184 and 349,185 (20 September 2026), and the other contracts below in blocks 360,113 to 360,120 (21 September 2026). An agent registers an identity and a price, a client hires it through escrow, the agent delivers, and the payment and the rating stay on-chain as the agent's record.

CONTRACT	ADDRESS AND WHAT IT DOES
AgentRegistry	`0xa94f27F18267d09349809f3e2AeF8e7767033e8F`. Agent identities, prices and status. Registration needs no bond (`minBond` is 0). Governed by the multisig.
ServiceEscrow	`0x99b331495951dB91857902de91EAe9Ff54d8a719`. Holds a client's payment until delivery and review; a 2.5% fee goes to the treasury; one-day delivery and review windows. Disputes are governed by the ArbiterPool.
X402Vault	`0x8751Cf7e29Fe588c61FDc53323438247198eaa57`. Pay-per-request payments in the x402 style, settled in batches.
StreamPay	`0x59404F738A90E5CF725F5837EF40461d1EA2EC35`. Payment streams and subscriptions.
AgentAccountFactory	`0x82e7C593785f726A0A0BB4D37AbCaF2bA4a72dcb`. Smart-account wallets for agents, with session keys.
ArbiterPool	`0x367312B28f78dE97462905519337841e4d4cB2df`. Dispute resolution for escrowed jobs.
FRC-8004 registries	Identity `0xf3e8c83a0472602d04Cd774e3887cBAA76c62147`, reputation `0xd5984C5a187cD6EcF2698eb218988F73FBF08884` and validation `0x37feB1B3Fb6505d4D584dB0a632F3C20d9eAab97`.
AgentTokenFactory	`0xf9fcCF337a7930D146227601C1da7Be85bB50188`. Lets an agent launch its own FRC-20 token.

How much it is used, from `/api/stats` and `/api/agents` at 05:00 UTC on 27 September 2026, checked against the contracts at block 427,348:

- **13 agents registered.** Six were marked active and seven paused; five were online at the time of reading. Agents 1 to 12 are operated by the project. Agent 13 was registered by another address; it was offline and had completed no jobs.
- **6 escrowed jobs**, all completed, for 0.6 FMX in total and 0.015 FMX in fees. All six hired project-operated agents. They were paid for by a project wallet and by two addresses whose owners the project has not yet identified. None of them is evidence of outside demand.
- **9 settlements through the X402Vault**, for 0.161 FMX in total; no open payment streams or active subscriptions; 3 agent smart accounts; 1 agent token launched.

The Ferminux DEX and its liquidity locks

The Ferminux DEX is a constant-product exchange on chain 3961: factory `0x2034a8366fCdbfFCf4517D297f702aDDdba37040`, router `0x018C0Efca293F7a74D2f53ce738BA5e2f412BA9f` and WFMX, wrapped FMX, `0x8a9Ae4D652cEba09Db8Ebf48D28C943b41B377Ae`. Liquidity tokens can be time-locked in the

LiquidityLocker (`0xe588c594388B978E64B69E2Dd91CC7E302763951`), which pays a lock out once, to its owner, on or after its unlock time.

POOL	RESERVES AT BLOCK 427,348	SEEDED BY, AND ITS LOCK
WFMX/USDF `0x04B2D76a04ED9b53a0d8fF9ee375eafeF04886f9`	480,769.23 WFMX and 250,000 USDF	The treasury, on 26 September 2026. Lock id 1, owner the treasury, locked 2026-09-26 04:34:10 UTC, unlocks 2027-09-26 04:34:03 UTC.
WFMX/AZNT `0xbab12e7B817F0686e11949eC06697235DC146845`	84,190.36 WFMX and 74,424.28 AZNT	The AZNT market-operations wallet, on 20 August 2026. Lock id 0, owner that wallet, locked 2026-08-20 16:43:02 UTC, unlocks 2027-08-20 16:42:38 UTC.

Each lock holds all of its pool's liquidity tokens except the 1,000-unit minimum every pool burns when it is created, so neither pool's liquidity can be withdrawn before its date. When the locks open, the project can withdraw it.

The pools have seen very little outside trading, and possibly none. Across both, `eth_getLogs` finds 9 swaps since launch, from 2 addresses. One swap came from the AZNT market-operations wallet, a project wallet. The other 8 came from `0xF61d…2847` . The project has not confirmed who owns that address; an earlier internal review counted it as a project wallet, and it also paid for two of the six agent jobs above. Every AZNT it traded came from the project: the AZNT market-operations wallet sent it 20,100 AZNT in blocks 13,782 to 13,823, and its swaps put a net 20,100 AZNT into the pool.

The market-operations wallet's swap, in block 414,730 on 26 September 2026, moved the WFMX/AZNT pool to $0.52. The WFMX/USDF pool was created at $0.52 the same morning. Neither pool has been traded since. The $0.52 on both pools is the price the project's own wallets set, and it matches the pay-in price because the project made it match. When the project's software shows these prices in dollars, it counts 1 USDF as 1 dollar and 1 AZNT as 1 manat (1.70 manat to the dollar). That is a display convention, not a claim about either token. It is also why the gateway reports the WFMX/USDF pool as $500,000 of liquidity: half of that is 250,000 USDF, counted at one dollar each.

USDF and AZNT

Both are FRC-20 tokens the project issued, with 6 decimals and the same contract design: an admin that grants roles, and separate roles to mint, burn, pause every transfer, and block an address. The admin of both is the foundation multisig.

	USDF, "FERMINUX DOLLAR"	AZNT, "FERMINUX MANAT"
Address	`0xCd032A609e34121D1881E8DE7355b2c2c7092363`	`0xFc81ad7c145B868ef0CEC8D7Ec881Ac93f724178`
Denomination	Dollar-denominated	Manat-denominated
Supply	500,000, all minted in block 20,589 (21 August 2026) to the treasury	600,000, minted in blocks 13,529 and 13,820 (20 August 2026) to the AZNT market-operations wallet

	USDF, "FERMINUX DOLLAR"	AZNT, "FERMINUX MANAT"
Minter role	Granted to the multisig in block 20,585 and revoked in block 20,592	Granted to the AZNT market-operations wallet in block 13,526 and revoked in block 20,604
Where it is	250,000 in the WFMX/USDF pool, 250,000 in the treasury	74,424.28 in the WFMX/AZNT pool, 525,475.72 in the market-operations wallet, 100 at `0x7F16…30f7`

Supplies, balances and the admin read at block 427,348 (27 September 2026, 05:05 UTC); role, mint and transfer events read with eth_getLogs on each contract.

No minter, pauser or blocklist role is assigned today. But the multisig, as admin of both contracts, can grant any of them again at any time: it can mint more of either token, freeze every transfer, or block an address and destroy the tokens it holds.

> **This paper makes no claim that USDF or AZNT is collateralised, backed or redeemable.** The project has not published reserve or redemption documentation for either token: no reserve attestation, no redemption process, no named party that redeems. The comments in both contracts' source code describe them as reserve-backed; that description has not been substantiated, and this paper does not repeat it. Treat both as tokens whose value rests on the project alone.

The wallet

The Ferminux wallet runs in the browser at `wallet.ferminux.net` and `ferminux.net/wallet/`. An Android version has been built, but when this paper was written it was not in an app store and the project's sites did not offer it for download. The wallet holds FMX, FRC-20 tokens and NFTs on chain 3961 and on the seven pay-in networks. It connects to other apps through WalletConnect, and can swap on the Ferminux DEX or buy through the pay-in. It is a single-key hot wallet: the key is encrypted on the device, and there is no multisig or hardware-key option. It has not been externally audited.

The explorer

`explorer.ferminux.net` is the project's own front end over a Blockscout index, with contract source verification. Since 26 September 2026 it indexes from an archive node with call tracing, so FMX moved by contracts, such as WFMX unwraps and escrow payouts, is visible as internal transactions.

NFT collections

- **Ferminux Citizens** (FRC-721, symbol FMXC), `0x5672AF1a567a46BAaFeb66959b7A95666E7f4252`, deployed in block 404,087 on 25 September 2026. 136 ids in four tiers priced at 50, 100, 250 and 500 FMX; 3 had been minted at block 427,348. The owner is the multisig, and a curator key (`0xEa70…2eD7`) can append new ids: the supply is not locked.
- **Ferminux Agents** (FRC-721, symbol FMXA), `0x84FE97C49Ffe4227d9ea139B5998C097D9C06ddd`. A fixed 41 ids at 50 FMX each; 3 had been minted at block 427,348.

The launchpad

The TokenFactory (`0x62BC7d9671EfE1385413434aB8fdfE2fa4aE01D4`) deploys a standard FRC-20 token for a 10,000 FMX fee, which it sends to `0x…dEaD` , where it can never be spent. No token had been launched through it at block 427,348.

The bridge to BNB Smart Chain: not operating

The bridge is lock-and-mint: FMX locked in the Ferminux bridge contract (`0xe162eeDa683f067d4Ebf61060Fa322332a779EF4`) is minted as wFMX on BNB Smart Chain (`0x73e64635E2a7b393F2aa3924dcf91fE3cFF51BD0`) and burned to come back. A transfer is released when 2 of its 3 registered validators sign it, after a wait for confirmations and a signed checkpoint, and within per-transfer and daily caps. Every change to its parameters waits 48 hours. Its owner on each chain is the foundation multisig.

It is not operating. Its validators have not been signing for Ferminux since 11 September 2026: the last checkpoint they recorded in the bridge's CheckpointRegistry on BNB Smart Chain (`0xB35bE8a672BDD265a88853Be3b053f1d21dCE816`), for Ferminux block 232,337, is dated 2026-09-11 08:54:45 UTC. Their status report at 05:00 UTC on 27 September 2026 gave the reason as an unreadable checkpoint, and the bridge app is offline. The last transfer recorded by the Ferminux bridge contract was in block 22,271, on 21 August 2026. The contracts are not paused on-chain, but nothing moves without validator signatures. At Ferminux block 427,348 and BNB Smart Chain block 124,274,295 (27 September 2026, about 05:09 UTC), 499.510989 FMX was locked on Ferminux and 499.510989 wFMX was outstanding on BNB Smart Chain: every wFMX is matched by locked FMX. When it runs, the bridge is secured by validator keys that one operator controls; it is not trustless. A proof-based replacement is planned (section 6).

How much the chain is used

On 25 September 2026 the project ran a labelled load test. Its own agent Wizrd (agent 12) sent 169,325 transactions across 42,322 wallets it generated. Each transaction carries the marker `FXLT` (`0x46584c54`) in its data, so anyone can filter it out. The test is switched off. It was the project testing its own chain, not use by anyone else, and **every usage figure in this paper excludes it**. The explorer's raw totals include it: they read 180,135 transactions and 42,914 addresses.

Without the load test, the chain had carried 10,808 transactions since launch, through block 427,292, and 22 in the 24 hours before that read (`/api/loadtest/stats` , field `organic` , at 05:00 UTC on 27 September 2026). About 590 addresses have appeared outside the load test: the explorer's 42,914 less the load test's 42,322. These are small numbers, and much of the activity behind them is the project's own.

5 Security model

What protects block production

- **Signers cut off from strangers.** Since 26 September 2026 every signer host accepts and makes peer-to-peer connections only to a fixed list of the network's own nodes, in both directions, enforced by the host's packet filter and restored at boot.

- **Release B** on every node the project runs: the checkpoint pin and the four peer-to-peer fixes (section 1). One known upstream issue, CVE-2026-26313, is not fixed in release B, because its fix exists only in a much later upstream version. On the signers the packet filter mitigates it. The public RPC, boot and witness nodes must accept connections from anyone, so they stay exposed to it.
- **The 64-block reorganisation limit** in every node, and fork choice that prefers an authority chain (section 2).
- **Monitoring** that alerts the operator on a stalled chain, a silent signer or a shrinking margin.

The signer machines are not dedicated to Ferminux: at least some of them also run other services for the same operator.

Keys and custody

KEY	WHAT IT CONTROLS	WHO HOLDS IT
Foundation multisig `0x910BD467D8576277f8f96DF47428377FFD94fEfe`, 2 of 3; owners `0x1a14…19EA`, `0x0fBB…ce15`, `0x11B5…41db`	The reward sink; the FoundationLock after it opens; the bridge on Ferminux; USDF and AZNT (admin); AgentRegistry governance; both NFT collections; the planned ValidatorHub	All three owner keys are held by one operator
The same three keys, compiled into the node as break-glass owners, 2 of 3	Replacing the whole signer set without a vote	The same operator
BNB Smart Chain multisig `0x15D0791d49A089863243BE2C2050e5d26E1bBA9c`, 2 of 3, the same three owners	The bridge on BNB Smart Chain; the PancakeSwap liquidity tokens	The same operator
Four signer keys	Confirming blocks	One per signer machine, all run by the foundation
Treasury, ecosystem, AZNT market-operations and community wallets	About 22.3 million FMX and the project's USDF and AZNT	Single keys held by the project
Pay-in hot wallet `0xc2a7…05Fa`	The FMX sold through the pay-in (50,000 FMX, rounded, at block 427,348)	A key on the project's gateway server
Three bridge validator keys	Attesting bridge transfers	Machines run by the operator

In plain terms: a multisig whose keys all sit with one operator is, in practice, a single key. Anyone who controlled that operator's keys could withdraw the reward sink, take over USDF and AZNT, change the bridge after its 48-hour delay, and replace the signer set. Moving the owner keys to separate holders is planned (section 6) and has not happened.

What the contracts constrain

- The FoundationLock cannot release anything before 2027-08-21 19:26:36 UTC, and nobody can shorten that.

- FMXVesting is irrevocable and pays only its beneficiary, on its fixed schedule.
- The two DEX liquidity locks cannot be withdrawn before 2027-08-20 and 2027-09-26.
- The bridge's owner must wait 48 hours for any change that increases what the bridge can do; pausing and tightening caps take effect at once.
- USDF and AZNT constrain their admin in nothing: new roles, and therefore new supply, freezes and blocks, take effect immediately.

Audits

No outside firm has audited the node client, the contracts, the bridge or the wallet. The code has been through the project's own repeated adversarial reviews, and the contracts and node carry their own test suites; that is not an independent professional audit and should not be read as one.

Reporting a vulnerability

The repository's `SECURITY.md` and `/.well-known/security.txt` give `security@ferminux.com` as the contact. That address cannot receive mail today: ferminux.com had no mail (MX) record when this paper was written, and the repository's private vulnerability reporting on GitHub is turned off. Until that is fixed, open an issue at github.com/aliasghar89/ferminux/issues titled "Security contact request" that says only that you have a report and how to reach you. Do not describe an unfixed flaw in a public issue, a forum thread or a bounty.

6 Roadmap

These are plans recorded in the repository. None has a committed date, and any of them may change or not happen.

- **The validator pilot.** Deploy the ValidatorHub with the multisig as owner, rebuild the pilot package on the release B node, and fill the first 20 seats from applicants (section 2). Then open seats to every wallet with the contract's one-way switch, after an external audit of the seat contract.
- **An external audit** of the contracts that hold funds, which the plan sets before opening validator seats to every wallet. None has been published.
- **More signers and separate key holders.** Independent signer operators voted in from applicants (section 2), a fifth signer on an independent hosting provider, so that two signers can stop without halting the chain; three new multisig owner keys on separate devices or with separate people; a node release that rotates the break-glass owners to match; and bootnodes spread beyond one hosting provider.
- **A proof-based bridge (bridge v2).** Replace the validators' signatures with proofs: a light client of Ferminux's signer set and a verifier of transfer receipts on BNB Smart Chain. The first part is built and tested against real chain data and is not deployed; moving to it would go through the bridge's 48-hour timelock.
- **A later EVM upgrade.** Moving from the London rule set to a newer one is planned as a separate hard fork with its own fork block, rehearsed in the lab first.

7 Verify it yourself

Nothing in this paper needs to be taken on trust. The calls below read each figure from the public RPC, the gateway API or BNB Smart Chain. `cast` is Foundry's command-line tool; any client that speaks JSON-RPC works the same way.

The chain and its signers

Chain id, client, head, fees and signers

```
R=https://rpc.ferminux.net
cast chain-id --rpc-url $R                          # 3961
cast rpc web3_clientVersion --rpc-url $R            # Ferminux/v1.10.26-ferminux-b/...
cast block latest --rpc-url $R --field gasLimit     # 100000000
cast base-fee --rpc-url $R
cast rpc eth_maxPriorityFeePerGas --rpc-url $R      # 0x3b9aca00 = 1 gwei
cast rpc clique_getSigners --rpc-url $R
cast rpc clique_status --rpc-url $R                 # blocks per signer, last 64
cast block finalized --rpc-url $R --field number    # the head minus 64
```

Genesis, the switch and the signer-set votes

```
cast block 0 --rpc-url $R --field hash        # 0x1b62e052…d92fadf
cast block 159999 --rpc-url $R --field hash   # 0xfa62e740…5af08, the pinned checkpoint
cast block 160000 --rpc-url $R                # difficulty 2, parent = the checkpoint
cast rpc clique_getSigners 0x63fd2 --rpc-url $R   # block 409,554: five signers
cast rpc clique_getSigners 0x63fdb --rpc-url $R   # block 409,563: three
cast rpc clique_getSigners 0x64033 --rpc-url $R   # block 409,651: four
```

Supply

The supply API: every component and every excluded address

```
curl -s https://ferminux.net/api/supply | jq
curl -s https://ferminux.net/api/supply/total
curl -s https://ferminux.net/api/supply/circulating
curl -s https://ferminux.net/api/supply/max
```

The emission schedule itself is in `chain/consensus/powhash/ferminux.go` (`FerminuxBlockReward`) and `chain/consensus/posa/posa.go` (`BlockReward`, `SplitReward`) in the repository.

Vesting, the FoundationLock, the reward sink and the multisig

```
V=0x6F488FB1f382Bc96Fef8bBfCa28A9647E5Fe430B
cast call $V "beneficiary()(address)" --rpc-url $R
cast call $V "start()(uint64)" --rpc-url $R       # 1787210727 = 2026-08-20 07:25:27 UTC
cast call $V "cliff()(uint64)" --rpc-url $R       # 15552000 s = 180 days
cast call $V "duration()(uint64)" --rpc-url $R    # 93312000 s = 1,080 days
cast call $V "released()(uint256)" --rpc-url $R

L=0xC0E01D9F49eE0967F34e1CB045B74D3Aefac189d
cast call $L "locked()(uint256)" --rpc-url $R
cast call $L "unlockAt()(uint64)" --rpc-url $R    # 1818876396 = 2027-08-21 19:26:36 UTC
cast logs --address $L "Locked(address indexed,uint256,uint256)" \
  --from-block 0 --to-block latest --rpc-url $R      # blocks 21668 and 22110, and who sent each
```

```
cast balance 0x691E5275BF346FfFa0B30174dDBeDfCC078dd8D6 --ether --rpc-url $R   # reward sink

M=0x910BD467D8576277f8f96DF47428377FFD94fEfe
cast call $M "getOwners()(address[])" --rpc-url $R
cast call $M "threshold()(uint256)" --rpc-url $R  # 2
```

USDF, AZNT, the pools and the price

Tokens, roles, pools and locks

```
U=0xCd032A609e34121D1881E8DE7355b2c2c7092363   # USDF
cast call $U "totalSupply()(uint256)" --rpc-url $R   # 500000000000 = 500,000 (6 decimals)
cast call $U "admin()(address)" --rpc-url $R         # the multisig
cast call $U "hasRole(bytes32,address)(bool)" $(cast keccak MINTER) $M --rpc-url $R   # false
cast logs --address $U "RoleRevoked(bytes32 indexed,address indexed)" \
  --from-block 0 --to-block latest --rpc-url $R      # block 20592

A=0xFc81ad7c145B868ef0CEC8D7Ec881Ac93f724178   # AZNT
cast call $A "totalSupply()(uint256)" --rpc-url $R   # 600000000000 = 600,000
# AZNT from the market-operations wallet to 0xF61d…2847: blocks 13782, 13820, 13823
cast logs --address $A "Transfer(address indexed,address indexed,uint256)" \
  0x040F1E90EF72b364141D91c3C0314ac3b5eCD0AE 0xF61d31FeC999af448C06fFaBB5EC28FbEE482847 \
  --from-block 0 --to-block latest --rpc-url $R

cast call 0x04B2D76a04ED9b53a0d8fF9ee375eafeF04886f9 "getReserves()(uint112,uint112,uint32)" --rpc-
url $R
cast call 0xbab12e7B817F0686e11949eC06697235DC146845 "getReserves()(uint112,uint112,uint32)" --rpc-
url $R

K=0xe588c594388B978E64B69E2Dd91CC7E302763951   # LiquidityLocker
cast call $K "getLock(uint256)((uint256,address,address,uint256,uint64,uint64,bool))" 0 --rpc-url $R
cast call $K "getLock(uint256)((uint256,address,address,uint256,uint64,uint64,bool))" 1 --rpc-url $R

# every swap on the two pools, since launch
cast logs --address 0xbab12e7B817F0686e11949eC06697235DC146845 \
  "Swap(address indexed,uint256,uint256,uint256,uint256,address indexed)" \
  --from-block 0 --to-block latest --rpc-url $R
cast logs --address 0x04B2D76a04ED9b53a0d8fF9ee375eafeF04886f9 \
  "Swap(address indexed,uint256,uint256,uint256,uint256,address indexed)" \
  --from-block 0 --to-block latest --rpc-url $R

curl -s https://ferminux.net/api/payin/assets | jq '{priceUsdPerFmx, spreadBps, minUsd, maxUsd}'
curl -s https://ferminux.net/api/payin/market | jq
```

The bridge

Locked FMX against wFMX outstanding, and whether the validators are signing

```
cast call 0xe162eeDa683f067d4Ebf61060Fa322332a779EF4 "lockedBalance(address)(uint256)" \
  0x0000000000000000000000000000000000000000 --rpc-url $R
cast call 0x73e64635E2a7b393F2aa3924dcf91fE3cFF51BD0 "totalSupply()(uint256)" \
  --rpc-url https://bsc-dataseed.bnbchain.org
cast call 0xe162eeDa683f067d4Ebf61060Fa322332a779EF4 "getValidators()(address[])" --rpc-url $R
cast call 0xe162eeDa683f067d4Ebf61060Fa322332a779EF4 "threshold()(uint256)" --rpc-url $R
# the last Ferminux checkpoint the validators recorded: block 232337, 1789116885 = 2026-09-11
08:54:45 UTC
cast call 0xB35bE8a672BDD265a88853Be3b053f1d21dCE816 "latest()(uint64,bytes32,uint64)" \
```

```
  --rpc-url https://bsc-dataseed.bnbchain.org
curl -s https://ferminux.net/bridge/status.json | jq
```

Usage, agents and the shared contracts

Usage without the load test, the agent network, and code at the standard addresses

```
curl -s https://ferminux.net/api/loadtest/stats | jq '{organic, counters, marker}'
curl -s https://explorer.ferminux.net/api/v2/stats | jq '{total_transactions, total_addresses}'
curl -s https://ferminux.net/api/stats | jq
curl -s https://ferminux.net/api/status | jq '.services.chain'
cast call 0xa94f27F18267d09349809f3e2AeF8e7767033e8F "nextId()(uint256)" --rpc-url $R      # agents
registered
cast call 0x99b331495951dB91857902de91EAe9Ff54d8a719 "nextJobId()(uint256)" --rpc-url $R   #
escrowed jobs
for a in 0xcA11bde05977b3631167028862bE2a173976CA11 0x000000000022D473030F116dDEE9F6B43aC78BA3 \
         0xd9Db270c1B5E3Bd161E8c8503c55cEABeE709552 0x0000000071727De22E5E9d8BAf0edAc6f37da032; do
  cast codesize $a --rpc-url $R
done
```

Figures read from explorer or API endpoints depend on the project's servers; the `cast` calls depend only on a node. For a check that trusts nothing the project runs, sync your own node with the one-line installer on the docs page and point `R` at it.

8 Risk factors

These are the risks the project knows of. There may be others.

1. **One operator runs the chain.** All four signers are the foundation's. Three of them together could censor transactions or rewrite up to the last 64 blocks. The same operator can replace the whole signer set through break-glass, without a vote. If the operator stops running the signers, the chain stops.
2. **A thin liveness margin.** Blocks need three of the four signers. If two stop at once, the chain stops until the operator restores them. The chain has stopped before: for 4 hours 36 minutes in its launch phase, and it took about 9 hours to recover.
3. **Governance keys in one hand.** Both multisigs are 2-of-3, but one operator holds all three owner keys. In practice that is one key. It controls the reward sink, the bridge, USDF, AZNT, the agent registry and the NFT collections, and losing it or having it stolen would put all of them at risk.
4. **Supply concentration.** The project holds most of the FMX that exists: 27.49 million excluded from circulating supply, and about 2.34 million more that the API still counts as circulating. Transfers or sales by the project can move any market in FMX.
5. **The price is set, not discovered.** $0.52 is the operator's pay-in price. The exchange pools sit at $0.52 because project wallets put them there. There have been 9 swaps on the Ferminux DEX since launch, all from a project wallet or an address that may be the project's, and none since the pools were set to $0.52. Figures that multiply supply by $0.52, such as the market capitalisation the explorer shows, multiply two numbers the project chose.

6. **Very low liquidity, and no working exit.** The Ferminux DEX pays out only in USDF or AZNT. The only pool against an outside asset, on PancakeSwap, held about $99 and is cut off while the bridge is down. The project does not buy FMX back. You may be unable to sell FMX at any price.
7. **USDF and AZNT rest on the project alone.** The project issued both. No reserve or redemption documentation has been published. Their admin, the multisig, can mint more, freeze all transfers, or block an address and destroy its tokens.
8. **The bridge is not operating** and wFMX on BNB Smart Chain cannot be redeemed for FMX until it is. When it runs it is secured by validator keys one operator controls, and the PancakeSwap liquidity tokens are not time-locked.
9. **No external audit** of the node client, the contracts, the bridge or the wallet.
10. **Software risk.** The node client is built on an older codebase and kept current by backporting fixes by hand. At least one known upstream vulnerability is unfixed and mitigated only on the signer hosts.
11. **Infrastructure concentration.** Both public RPC nodes, all three bootnodes and the explorer's archive node are on one hosting provider, and signer machines also run other services for the same operator.
12. **The emission limit is a software rule.** A node release could change it, and the operator that would ship the release also runs every signer.
13. **Little real usage.** Excluding the load test, the chain had carried 10,808 transactions since launch, and most agent-network activity is the project's own.
14. **Wallet risk.** The Ferminux wallet is a single-key hot wallet with no hardware-key or multisig option, and it has not been externally audited.
15. **No working security contact.** The published security address does not receive mail yet (section 5).
16. **Unknown operator.** The project has not published who runs it, a legal entity or a jurisdiction. If something goes wrong, there may be nobody you can hold responsible.
17. **Regulatory risk.** The project sells FMX directly to the public, and it issues tokens denominated in dollars and manat. Rules on token sales, on tokens that track a currency, and on electronic money differ by country, and may apply to any of these. The project has not published a legal opinion, a licence or a registration anywhere. A regulator could restrict the pay-in, the tokens, the sites or your access to them.
18. **Transfers are final.** There is no process for reversing a transaction or a bridge transfer sent in error, or for recovering a lost key.

9 Disclosures

- **Not an offer.** This paper describes software and contracts. It is not an offer to sell, a solicitation, or investment, financial, legal or tax advice. FMX is the coin used to pay for gas and settle payments on chain 3961. It carries no promise of value, return, redemption, dividend or any right against anyone.

- **The project is a seller.** The project sells its own FMX through the pay-in at a price it sets, and holds most of the supply. Read anything it publishes about FMX, including this paper, with that interest in mind.
- **Team.** The project has not published information about its team, the people who hold its keys, or a legal entity. It refers to its operator as "the foundation"; no legal foundation has been published. The source code is public at `github.com/aliasghar89/ferminux`.
- **An open commitment from the switch.** Before block 160,000 the project announced a one-time credit, from the community allocation, for every address outside the foundation that had produced blocks in the launch phase. On its census that was one address, `0x4adbea699af36ede04b71a8c8656b376742669bd`, and 3,030 FMX. It has not been paid: at block 427,348 that address held 3,136.875 FMX, its own launch-phase rewards, and had received no transfer.
- **Plans are plans.** Everything in section 6, and every description of the validator programme, is an intention that may change or not happen.
- **Figures are dated.** Every figure is a read at the block or date given, and it will be out of date when you read it. Use section 7.
- **The code wins.** Where this paper and the deployed software or contracts disagree, the software and contracts are authoritative.

10 About this document

Version 1.0, 27 September 2026, published at `https://ferminux.net/whitepaper/`, with a PDF printed from the same page at /whitepaper/ferminux-whitepaper-v1.0.pdf. A later version will carry a new number and date and say what changed.

This version replaces unpublished drafts dated 22 August and 26 September 2026. They contained claims that were wrong or had gone out of date, corrected here:

- They called USDF "fully-collateralised" and said it was "minted only against a confirmed fiat deposit". No reserve or redemption documentation has been published, and this paper makes no such claim (section 4).
- They gave FMX a "100,000,000 hard cap" beside a 32.5 million supply limit. No code enforces 100 million; the emission schedule's limit, 32,514,979.875 FMX, is the only one (section 3).
- They said the exchange had one pool and that no price was published. There are two pools, and the project's sites publish the $0.52 pay-in price (sections 3 and 4).
- They showed the team allocation at its beneficiary's address rather than at the FMXVesting contract, `0x6F48…430B`, which holds it (section 3).

Sources: the public RPC (`https://rpc.ferminux.net`), the gateway API (`https://ferminux.net/api/`), the explorer API (`https://explorer.ferminux.net/api/v2/stats`), a public BNB Smart Chain RPC, and the repository's node client, contracts and `genesis/genesis.json`. Every read was made on 27 September 2026 between 04:32 and 05:10 UTC unless the text gives another date.