meritr

BUIDL CTC 2026 · AI / RWA

Prove the history.
Keep the collateral.

Autonomous DeAI debt restructuring and cross-chain credit memory, on Creditcoin.

usemeritr.vercel.app · github.com/mrnetwork0001/Meritr · Apache 2.0

THE PROBLEM

Onchain credit is amnesiac, and it is brutal.

Amnesiac

A borrower with three years of flawless Aave repayments on Ethereum arrives on a new chain as a stranger. The history is real and public. No contract on the destination chain can verify it without trusting somebody to report it faithfully.

Brutal

Every major lending market answers distress with exactly one action: liquidation. A temporary drawdown ends the borrower's equity, dumps collateral into a falling market, and pays a bonus to a bot. Traditional finance restructures distressed debt every day. DeFi seizes it.

WHAT MERITR IS

Credit a chain can verify. Distress a borrower can survive.

01	**Prove**	Read a borrower's real repayment, collateral and borrow history from Ethereum through Creditcoin's Attestcoin precompile. A Merkle-inclusion and continuity proof the runtime itself validates.
02	**Score**	Fold proven facts into a portable 300–900 score that sets the interest rate and borrowing capacity, recomputed onchain rather than supplied as an argument.
03	**Restructure**	When a position enters distress, an autonomous agent cuts the rate, extends the term and retires debt from a reserve — instead of liquidating. The borrower keeps their collateral.

THE INTEGRATION

Attestcoin is the only way a fact can enter.

ETHEREUM MAINNET	Aave V3 repayment, supply, borrow	chainKey 3
PROOF BUILDER	Merkle inclusion + continuity proof	Creditcoin service
PRECOMPILE 0x…0FD2	the Creditcoin runtime verifies it	not an oracle
MERITRATTESTOR	decoded against an onchain event schema	CreditFactAttested

MeritrAttestor inherits ASCBase and resolves chain keys from the ChainInfo precompile at 0x…0FD3 rather than hardcoding them — a wrong key fails silently, so the deployment script aborts on a mismatch. Remove Attestcoin and the protocol has no inputs at all.

THE SECURITY CLAIM, MADE CHECKABLE

Change one byte. The runtime refuses it.

GENUINE PROOF	ACCEPTED
ONE BYTE ALTERED	REJECTED — Merkle proof validation failed

npm run verify:proof — a view call against the live precompile. No gas, no wallet, no account. Anyone can run it, including a judge, against the deployed contracts. Nobody can insert a credit fact: not the team, not a key, not a committee.

EVIDENCE

Real borrowers. Not a fixture.

363	174	$71.4M	0
CREDIT FACTS PROVEN	REAL ETHEREUM BORROWERS	OF PROVEN AAVE ACTIVITY	ORACLE OPERATORS

Counted from chain logs, not written down. A relayer holding no roles at all adds to this every four minutes, whether anyone is watching or not — which is a harder claim to fake than a recorded demo. Read live at /api/attestations.

WHY IT MATTERS

The score is not a badge. It is the terms.

NO PROVEN HISTORY

300

Bronze

24.00%	30%
APR	MAX LTV

29 PROVEN AAVE FACTS

785

Platinum

7.84%	70.4%
APR	MAX LTV

Same protocol, same collateral, same day. The only difference is history a chain could verify. That is the product.

THE LOAD-BEARING DECISION

restructure takes one argument: an address.

```
function restructure(address borrower)
    external
    returns (uint256, uint256, uint256);
```

~~No rate.~~ ~~No amount.~~ ~~No score.~~ ~~No signature over off-chain numbers.~~

The vault re-reads the borrower's Attestcoin-derived score and recomputes every term through the same library the agent used. The AI decides whether and whom. The chain decides how much. A fully compromised agent key can trigger restructurings the protocol would already have approved, and nothing else.

AUTONOMY, DEMONSTRATED

The agent found it, ranked it, and acted. Unattended.

```
Book  3 position(s): healthy=1, stressed=2
Triage: 2 position(s) actionable, ranked by loss averted.
  Restructuring 0x27e7c47f…: HF 1.079 -> 1.350, retiring 1321.01
  Confirmed: 0xfc72b3449fd384a331bf…
  Restructuring 0xc5818BFB…: HF 1.079 -> 1.350, retiring 550.42
  Confirmed: 0x75bc20b274853f0e1f01…
```

Both transactions signed by 0xC06B6015…, a key holding RISK_AGENT_ROLE and no other role on any Meritr contract. Health factor 1.079 → 1.350 on both. No collateral seized — this vault has never emitted a Liquidated event. Check triggeredBy on either transaction.

TRUST MODEL

Assume the agent key is stolen.

It can

Trigger restructurings the protocol would already have approved

It cannot

Invent a score

Grant itself a rate

Choose an amount

Drain the reserve

Seize collateral

The agent is not a liveness dependency either: anyone may trigger the identical restructuring six hours after a position is flagged. Borrower protection does not hinge on a server staying up. agents/scoring.py mirrors CreditMath.sol to the wei, and 168 parity vectors generated from the deployed library assert it.

STATED PLAINLY

What is real, and what is not.

Credit facts	proof-verified — nobody can create one without the precompile
The restructurings	real onchain state, two of them agent-signed
Contracts	all five source-verified on Blockscout
Collateral pricing	governance-fed behind PRICE_ROLE — the single trusted input
Demo market tokens	synthetic, open mint, worth nothing — and verified, so checkable
“ZK-Credit”	a commitment scheme today, not a SNARK
Contracts	unaudited. A CertiK audit is a prize, not a completed step

The credit layer holds no reference to the market layer: grep -ci 'vault' on the attestor and passport returns 0. Swapping the demo market for real assets would leave every score unchanged.

JUDGING CRITERIA

Against the five CEIP pillars.

Technical alignment	Attestcoin is not decoration — it is the only input path. ASCBase, the 0x…0FD2 BlockProver, the 0x…0FD3 ChainInfo registry.
Market & technical relevance	Cross-chain credit portability and restructuring-over-liquidation are live problems. The source data is real Aave V3 mainnet activity.
Product vision	History should follow a borrower between chains; distress should be survivable. The passport makes the first portable, the vault the second real.
Execution capability	122 commits across the window, 50 Solidity and 39 Python tests, wei-level agent/contract parity, a written limitations section.
User-base expansion	Honest status: 174 proven borrowers have not opted in. A real distribution channel, but converting it is future work — not a shipped result.

WHAT COMES NEXT

From proven history to a credit market.

Real assets	The vault's asset and collateral are immutable, so production means a fresh deployment against canonical stablecoins and a real price feed rather than PRICE_ROLE.
Writeability	Creditcoin's write path is in final development. When it ships, a restructuring decided here can settle on the source chain.
The SNARK	factsCommitment is the intended substitution point: prove a score band without publishing the facts behind it.
Onboarding the proven	174 addresses already carry standing here. Reaching them is the go-to-market.

meritr

Cross-chain credit a chain can verify. Distress a borrower can survive.

Live console	usemeritr.vercel.app
Source	github.com/mrnetwork0001/Meritr
Risk API	meritr.38.49.216.120.sslip.io/api/attestations
MeritrVault	0x233D2aE279230fBFFbe61e6dF2A9DC6bF6ff3e84

Creditcoin CC3 testnet, chain 102031 · Attestcoin precompile 0x…0FD2 · figures read from chain on 2026-09-12