

CIPHRA

AI that computes on ciphertext

An FHE coprocessor network for LLMs · Whitepaper v0.1 · October 2026

Token: \$CPH **Category:** Confidential computing **X:** @ciphra_fhe

This draft describes planned technology and products. Performance figures and dates are targets. Token launch details are published by the launchpad. This is not an offer of securities or investment advice.

Contents

1. Every prompt is a confession
2. Why today's options fall short
3. What CIPHRA is
4. How it works
5. The technology
6. Products
7. Who it is for
8. The \$CPH token
9. Roadmap
10. Security model and honest limits
11. Team and partners
12. Risks and disclaimer

1. Every prompt is a confession

A doctor pastes a patient's scan report into an AI assistant and asks for a second opinion. A lawyer uploads a draft settlement to check the wording. A trader asks a model to stress-test a position nobody else is supposed to know about. A teenager types something into a chatbot at 2 a.m. that they have never told anyone.

Each of these is a confession. And in almost every AI product today, that confession arrives at someone else's server in plain text. The model has to read it to answer it. So does the infrastructure around the model: logs, caches, monitoring, the engineers on call.

Most of the time nothing bad happens. But "most of the time" is not good enough for a hospital, a bank or a government, which is why the organizations with the most valuable data are often the slowest to use AI. They are not afraid of the model. They are afraid of the moment their data becomes readable.

| *What if the model could answer without ever being able to read the question?*

That is not science fiction. It is a branch of cryptography called fully homomorphic encryption (FHE), first shown to be possible in 2009. FHE lets a computer run calculations on encrypted data and produce an encrypted result that only the data owner can unlock. The server does the work. It never sees the input, and it never sees the answer.

For years FHE was too slow for anything but toy problems. That is changing. Better schemes, open-source libraries and dedicated hardware have cut the gap by orders of magnitude, and government and industry programs are now building FHE accelerator chips. CIPHRA exists to turn that progress into something developers can use today and scale tomorrow: a network of FHE coprocessors that run AI inference on data that stays encrypted from the moment it leaves the user's device.

2. Why today's options fall short

Teams that want private AI today choose between four imperfect options:

Option	What it protects	What it leaves open
Standard AI API with a privacy policy	Legal promises, contracts	Data is readable by the provider and its infrastructure
Self-hosting a model on-premise	Data stays inside the company	Expensive hardware, limited model quality, heavy operations
Trusted execution environments (TEE)	Data is isolated inside secure hardware	Trust in the chip vendor; history of side-channel attacks
Anonymizing data before sending	Removes obvious identifiers	Context can re-identify people; much useful detail is lost

FHE is the only approach where the security does not depend on trusting the operator, the hardware vendor or a policy document. The data is protected by mathematics. Its weakness has always been speed. CIPHRA focuses on that weakness.

3. What CIPHRA is

CIPHRA is an FHE coprocessor network for language models. Developers send encrypted prompts and receive encrypted outputs. Independent node operators run the encrypted computation on accelerated hardware and are paid per job. The encryption keys never leave the user.

- **For developers:** an SDK and API that look like a normal AI inference API, with encryption and decryption handled on the client.
- **For node operators:** software that turns GPU servers, and later dedicated FHE accelerators, into paid coprocessors.
- **For end users:** AI features where "we can't see your data" is a technical fact, not a promise.

4. How it works

1. **Key generation.** The client SDK creates a secret key on the user's device and a public evaluation key that lets servers compute on ciphertext without decrypting it.
2. **Encryption.** The prompt is tokenized and embedded locally, then encrypted.
3. **Encrypted inference.** A CIPHRA node runs the model's layers on the ciphertext. It sees only encrypted numbers in, encrypted numbers out.
4. **Verification.** The client checks that the agreed model was run. At launch this uses redundant execution on independent nodes with matching results; succinct cryptographic proofs for FHE computation will be added as that research matures.
5. **Decryption.** Only the client's secret key can turn the result into readable text.

At no step does the network, the node operator or CIPHRA itself hold a key that can read the user's data.

5. The technology

5.1 Schemes

CIPHRA builds on established, peer-reviewed FHE schemes rather than inventing new cryptography. CKKS supports approximate arithmetic on real numbers and suits the matrix multiplications that dominate neural networks. TFHE supports fast evaluation of non-linear functions through programmable bootstrapping. The CIPHRA compiler chooses the scheme per layer. The implementation builds on mature open-source libraries such as OpenFHE and TFHE-rs, and every CIPHRA-specific component will be open-sourced and audited.

5.2 Making models FHE-friendly

Neural networks contain operations that are cheap in plain text but expensive under encryption, especially non-linear functions such as softmax and GELU and the normalization layers. CIPHRA uses three techniques:

- **Polynomial approximation** of non-linear functions, tuned for each model so accuracy loss stays within a stated tolerance.

- **Hybrid splitting:** the client runs the cheap first and last steps (tokenization, embedding, decoding) locally, and the network runs the heavy middle layers encrypted.
- **FHE-aware fine-tuning** of open-weight models so they keep their quality after approximation.

5.3 Hardware acceleration

FHE workloads are dominated by large polynomial multiplications and number-theoretic transforms, which parallelize well. CIPHRA's acceleration path has three stages:

Stage	Hardware	Purpose
Now	Data-center GPUs	Custom CUDA kernels for NTT and key switching; runs on hardware node operators already own
Next	FPGA cards	Lower latency and better energy use for production workloads
Later	FHE ASICs	Dedicated accelerator chips, sourced through hardware partnerships as they reach the market

5.4 Performance, stated honestly

Encrypted inference is still much slower and more expensive than plain inference, and it will remain so for some time. CIPHRA does not claim to run frontier-size models fully encrypted at chat speed today. The first production targets are small and mid-size models where encrypted inference is already practical: classification, embeddings for private search, extraction from documents and compact language models. Larger models follow as acceleration improves. CIPHRA will publish benchmarks for every supported model, including latency, cost per request and accuracy against the unencrypted baseline.

6. Products

Product	Description
CIPHRA SDK	Client libraries (Python, TypeScript) that handle keys, encryption, decryption and proof checking
Private Inference API	A familiar API for encrypted calls to supported models, priced per request
Private RAG	Encrypted embeddings and similarity search, so companies can search private documents with AI without exposing them
CIPHRA Node	Software for operators to run encrypted workloads on GPU servers and earn fees
Model Registry	A public list of supported models with their FHE parameters, accuracy reports and benchmarks

7. Who it is for

- **Healthcare:** triage, coding and summarization of patient records without exposing them to the AI provider.
- **Legal:** contract review and e-discovery on privileged documents.

- **Finance:** fraud scoring and risk analysis on customer data that cannot leave the bank in readable form.
- **Enterprise copilots:** internal assistants that can read sensitive documents without the vendor seeing them.
- **On-chain agents:** AI agents that act on private inputs, such as a trading strategy, without revealing them to the network.

8. The \$CPH token

\$CPH is the utility token of the CIPHRA network. It is designed to be used, not just held.

- **Paying for compute:** encrypted inference jobs are priced in \$CPH. Enterprises that prefer stable pricing can pay in stablecoins, which are converted at settlement.
- **Node staking:** operators stake \$CPH to receive jobs. A node that returns results that fail verification loses part of its stake.
- **Priority and capacity:** stakers can reserve guaranteed capacity for production workloads.
- **Governance:** holders vote on supported models, fee parameters, grants and protocol upgrades.

Launch: \$CPH will be launched through a launchpad. Supply, allocation, vesting and sale terms are set and published by the launchpad and are not part of this document.

9. Roadmap

Milestones are targets and depend on funding, audits and hardware availability.

Phase	Target timing	Deliverables
1 · Proof	Months 0–6	Open-source compiler alpha; encrypted classification and embedding models on GPU; public benchmarks; developer testnet
2 · Private RAG	Months 6–12	Private RAG and Inference API beta; first external node operators; security audit of SDK and node software; \$CPH launch on launchpad
3 · Language models	Months 12–24	Encrypted inference for compact language models with hybrid splitting; FPGA nodes; enterprise pilots in healthcare and finance
4 · Scale	Month 24+	Larger models as accelerators mature; ASIC partnerships; full on-chain governance of the network

10. Security model and honest limits

- **What is protected:** prompt content, model outputs and retrieved documents, against node operators, CIPHRA and network observers.
- **What is not hidden:** the fact that a request was made, its size and timing, and which model was called.
- **Model weights:** in the initial design weights are known to nodes; encrypted inputs are protected, not the model itself.

- **Key safety:** if a user's device or secret key is compromised, encryption cannot protect their data.
- **Accuracy:** approximated models may differ slightly from the original; every supported model ships with a published accuracy report.

11. Team and partners

To be completed before publication: core team and advisors with verifiable backgrounds in cryptography and systems, the legal entity, audit firms, hardware and launchpad partners. Investors and users judge confidential-computing projects heavily on the team's credibility, so this section should not be published empty.

12. Risks and disclaimer

- **Technical:** performance targets may take longer than planned; cryptographic implementations may contain bugs despite audits.
- **Hardware:** availability and cost of FPGA and ASIC accelerators are outside the project's control.
- **Adoption:** the network depends on developers and enterprises choosing encrypted inference over cheaper alternatives.
- **Regulatory:** rules on digital assets differ by country and change over time.
- **Market:** the price of \$CPH may be volatile, and holders may lose part or all of its value.

This document describes planned technology. It is not an offer to sell or a solicitation to buy any security, and nothing in it is investment, legal or tax advice. All figures and timelines are subject to change.