octra

git x discord telegram

gitxdiscordtelegram

- welcome to octra
- user docs
- generating a wallet
- installing the client
- using the client
- sending transactions
- encrypting and decrypting the balance
- stealth transactions
- circles
- tokens
- network fees
- role of oct
- octranomics
- wrapped oct
- bridging
- contract addresses
- developer docs
- introduction to applied
- programs
- setting up IDE
- building your first program
- building circles
- RPC scheme
- applied cheatsheet
- validator docs
- running a node
- tech docs
- ▼ HFHE

  modular arithmetic over hypergraphs in HFHE

  ▼ subtraction in hypergraph space

  hypergraph stability after subtraction

  minimizing deviations in hypergraph matrices

  multiplication of integers

Archived copy · cryp2nova.com

integer division and range narrowing mechanics
HFHE key sharding
starting vector generation
sbox module
transformation mechanism
noise basis determination
noise assessment in ciphertext elements

welcome to octra

user and developer documentation for the octra network

documentation is updated continuously as the network grows and evolves, and some materials may become momentarily outdated. always double-check octra labs' official github.

octra is inherently a collaborative project. if you have any suggestions, you are welcome to send a pull request or join the discussion in our telegram or in the discord development channel.

for specific technical questions, write to:

what is octra

octra is an FHE blockchain network with support for isolated execution environments. it can function both as a standalone L1 blockchain and as decentralized encrypted middleware for existing networks and applications.

octra was founded in 2021 and has been in active development since late 2022. an internal prototype was released in october 2023, a public testnet was launched in june 2025, and mainnet alpha was launched in december 2025.

the majority of octra's code is written in OCaml and C++. octra's public codebase and tooling are progressively open-sourced through official repositories.

what is FHE

fully homomorphic encryption, or FHE, allows computation on encrypted data without decrypting it.

in the context of octra, this means that the network can process all kinds of data, execute arbitrary logic, and return results, all while the underlying data remains encrypted.

the FHE scheme that is used in octra network is called HFHE, or Hypergraph Fully Homomorphic Encryption. it is an in-house development based on hypergraphs and designed to enable consumer-grade applications with the help of parallel encrypted computations.

Archived copy · cryp2nova.com

some octra libraries can be easily ported to other projects. for example, the HFHE library can be compatible with any project and can operate independently.

how octra works

octra combines a blockchain network, a HFHE-based encrypted compute layer, and isolated execution environments called **circles**.

circles can host **programs** with dedicated application logic and encrypted data storage. programs can be built in native AppliedML, as well as Rust, C++, OCaml, or WASM.

octra accepts all kinds of data, which is transformed into vectors, then encrypted and processed using HFHE. encryption and decryption require R1CS proof generation.

the network automatically distributes storage and validation between nodes of three types: bootstrap nodes, standard nodes, and light nodes.

further information can be found in the octra network litepaper (2024).

octra virtual machine

OVM supports AppliedML as the native language for programs. it also supports a diverse array of languages through WASM. an IDE is built-in for the current version of the web client.

generating a wallet ›

Last updated 11 days ago

- welcome to octra
- user docs
- generating a wallet
- installing the client
- using the client
- sending transactions
- encrypting and decrypting the balance
- stealth transactions
- circles
- tokens
- network fees
- role of oct
- octranomics
- wrapped oct
- bridging

CRYP2NOVA

Archived copy · cryp2nova.com

- contract addresses
- developer docs
- introduction to applied
- programs
- setting up IDE
- building your first program
- building circles
- RPC scheme
- applied cheatsheet
- validator docs
- running a node
- tech docs
- ▼ HFHE
  - modular arithmetic over hypergraphs in HFHE
  - ▼ subtraction in hypergraph space
    - hypergraph stability after subtraction
    - minimizing deviations in hypergraph matrices
    - multiplication of integers
    - integer division and range narrowing mechanics
    - HFHE key sharding
    - starting vector generation
    - sbox module
    - transformation mechanism
    - noise basis determination
    - noise assessment in ciphertext elements



CRYP2NOVA

Archived copy · cryp2nova.com