AGENDA

- Blockchain Background:
- Blockchain Projects and FHE:
  - DERO
  - NuCypher
Blockchain Applications

The intersection of fully homomorphic encryption and blockchains
BLOCKCHAIN BACKGROUND:

It’s NOT (just) this:
BLOCKCHAIN BACKGROUND:

**WHAT**
Secure Digital Spreadsheet
Disruptive Technology that Powers Bitcoin

**HOW IT WORKS**
- New info is added to end of chain
- Existing blocks of info can't be edited

**BENEFITS**
- No middle man
- Trust
- Security

**USES BEYOND CRYPTO**
- Contracts
- Health care
- Voting

*Napkin Finance*
BLOCKCHAIN BACKGROUND:

• Stuart Haber and W. Scott Stornetta described what is currently known as blockchain, in 1991. (Haber, S.; Stornetta, W. S., 1991).

• Their first work involved working on a cryptographically secured chain of blocks whereby no one could tamper with timestamps of documents.

• They then built on an earlier thesis concept of David Chaum’s blockchain-like system for maintaining computer systems in a mutually suspicious group (Chaum, D. 1982).

• Chaum would go on to describe the first digital currency a year later after his thesis (Chaum, D. 1983), though it is important to note this was not the first conceptualization of “electronic cash”.

BLOCKCHAIN BACKGROUND:
BLOCKCHAIN BACKGROUND:
BLOCKCHAIN BACKGROUND:

\[
\text{Enc}_{pk_1}(x_1) \quad \text{Enc}_{pk_2}(x_2) \quad +
\]

\[
\text{Dec}_{sk_2}(E_\sigma(v) \oplus v_s) \quad = \quad ????
\]
Blockchain Projects and FHE

State of the art in Blockchain/FHE - DERO
BLOCKCHAIN PROJECTS AND FHE: DERO

• “DERO is the first crypto project to combine a Proof of Work blockchain with a DAG (Directed Acyclic Graph) block structure and fully anonymous transactions based on Homomorphic Encryption.”

• The distributed ledger has a 60 second average block time (time to process transactions) and “is secure against majority hashrate attacks”.

• DERO uses Homomorphic Encryption and has smart contracts on its native chain without any extra layers or secondary blockchains.

• At present DERO has Smart Contracts on the first version of the CryptoNote protocol testnet.
# BLOCKCHAIN PROJECTS AND FHE: DERO

This page does not represent the entire state of the dero network - listing a node on this page is a voluntary process.

<table>
<thead>
<tr>
<th>Node</th>
<th>Latency</th>
<th>Height/Topo</th>
<th>Block Propagation</th>
<th>Peers INC</th>
<th>Peers OUT</th>
<th>Version</th>
<th>Last 10 blocks propagation</th>
<th>Avg</th>
<th>Last update</th>
</tr>
</thead>
<tbody>
<tr>
<td>NL4</td>
<td>108 ms</td>
<td>96573 / 96687</td>
<td>466 ms</td>
<td>0</td>
<td>0</td>
<td>2.0.0-1.alpha.atlantis+27062018</td>
<td></td>
<td>146 ms</td>
<td>5s ago</td>
</tr>
<tr>
<td>Europools.tech</td>
<td>109 ms</td>
<td>96573 / 96687</td>
<td>318 ms</td>
<td>0</td>
<td>0</td>
<td>2.0.0-1.alpha.atlantis+27062018</td>
<td></td>
<td>803 ms</td>
<td>5s ago</td>
</tr>
<tr>
<td>bailout2k18</td>
<td>110 ms</td>
<td>96573 / 96687</td>
<td>197 ms</td>
<td>0</td>
<td>0</td>
<td>2.0.0-1.alpha.atlantis+27062018</td>
<td></td>
<td>565 ms</td>
<td>5s ago</td>
</tr>
<tr>
<td>Dank Node</td>
<td>82 ms</td>
<td>96573 / 96687</td>
<td>0 ms</td>
<td>0</td>
<td>0</td>
<td>2.0.0-1.alpha.atlantis+27062018</td>
<td></td>
<td>454 ms</td>
<td>5s ago</td>
</tr>
<tr>
<td>sq.deo.miner.rocks - 1</td>
<td>254 ms</td>
<td>96573 / 96687</td>
<td>1944 ms</td>
<td>0</td>
<td>0</td>
<td>2.0.0-1.alpha.atlantis+27062018</td>
<td></td>
<td>1529 ms</td>
<td>3s ago</td>
</tr>
</tbody>
</table>
BLOCKCHAIN PROJECTS AND FHE: DERO
BLOCKCHAIN PROJECTS AND FHE: DERO

• Specifically, DERO uses Homomorphic Encryption to do arithmetic operations and “settle balances with data being always encrypted”. They claim that “balances are never decrypted before/during/after operations in any form”.

• They utilize “Homomorphic Rings” for confidential transactions: This provides “untraceability, privacy, and fungibility” while making sure that the system is stable and secure.
OTHER DERO (FHE BASED) FEATURES:

• Homomorphic account based model ([Transaction Execution, lines 82-95]).

• Instant account balances[ Need to get 66 bytes of data only from the blockchain].

• No more chain scanning or wallet scanning to detect funds, no key images etc.

• Fixed per account cost of 66 bytes in blockchain[Immense scalability].

• **Perfectly anonymous transactions with many-out-of-many proofs** [bulletproofs and sigma protocol]

• Deniability
OTHER DERO (FHE BASED) FEATURES:

- Fixed transaction size say ~2.5KB (ring size 8) or ~3.4 KB (ring size 16) etc based on chosen anonymity group size[ logarithmic growth]

- Anonymity group can be chosen in powers of 2.

- Allows homomorphic assets ( programmable SCs with fixed overhead per asset ), with open Smart Contract but encrypted data.
OTHER DERO (FHE BASED) FEATURES:

- Allows chain pruning on daemons to control growth of data on daemons.
- Transaction generation takes less than 25 ms.
- Transaction verification takes even less than 25ms time.
- No trusted setup, no hidden parameters.
- Pruning chain/history for immense scalability[while still secured using merkle proofs].
- Example disk requirements of 1 billion accounts (assuming it does not want to keep history of transactions, but keeps proofs to prove that the node is in sync with all other nodes)
**DERO SIZES:**

<table>
<thead>
<tr>
<th>Ring Size</th>
<th>DEROHE TX Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1553 bytes</td>
</tr>
<tr>
<td>4</td>
<td>2013 bytes</td>
</tr>
<tr>
<td>8</td>
<td>2605 bytes</td>
</tr>
<tr>
<td>16</td>
<td>3461 bytes</td>
</tr>
<tr>
<td>32</td>
<td>4825 bytes</td>
</tr>
<tr>
<td>64</td>
<td>7285 bytes</td>
</tr>
<tr>
<td>128</td>
<td>11839 bytes</td>
</tr>
<tr>
<td>512</td>
<td>~35000 bytes</td>
</tr>
</tbody>
</table>
Blockchain Projects and FHE

State of the art in Blockchain/FHE - NuCypher
A GPU implementation of fully homomorphic encryption on torus

This library implements the fully homomorphic encryption algorithm from TFHE using CUDA and OpenCL. Unlike TFHE, where FFT is used internally to speed up polynomial multiplication, nufhe can use either FFT or purely integer NTT (DFT-like transform on a finite field). The latter is based on the arithmetic operations and NTT scheme from cuFHE. Refer to the project documentation for more details.

Usage example

```python
import random
import nufhe

size = 32
bits1 = [random.choice([False, True]) for i in range(size)]
bits2 = [random.choice([False, True]) for i in range(size)]
reference = [not (b1 and b2) for b1, b2 in zip(bits1, bits2)]

cxt = nufhe.Context()
secret_key, cloud_key = cxt.make_key_pair()

ciphertext1 = cxt.encrypt(secret_key, bits1)
ciphertext2 = cxt.encrypt(secret_key, bits2)

vm = cxt.make_virtual_machine(cloud_key)
result = vm.gate_nand(ciphertext1, ciphertext2)
result_bits = cxt.decrypt(secret_key, result)

assert all(result_bits == reference)
```
BLOCKCHAIN PROJECTS AND FHE: NuCypher

• Created GPU Optimized version of TFHE (GSW based), called called nuFHE. It is optimized for CUDA and OpenCL.

• There is an optimization on the FHE operations using FFT and non-integer NTT (number theoretic transform) . The latter functionality is available in cuFHE, the CUDA port of TFHE, and both are also available in PALISADE (DFT's using FFT, but CRT is an integer-only NTT in EVALUATION mode).
BLOCKCHAIN PROJECTS AND FHE: NuCypher

• Miners can then compute directly on any subset of the users’ encrypted inputs. *No off-chain coordination needed. No interaction necessary for the computation. No need for the users to even be online.”*
THANK YOU!

https://palisade-crypto.org