

+91 8310902220
UTC +5:30
aaryamannchallani@gmail.com

Aaryamann Challani

Applied Cryptography Engineer

x.com/p1ge0nh8er
github.com/rymnc
linkedin.com/in/aaryamannc

Robust, well-written, performant and maintainable code is something I strive to work towards.

SKILLS

Languages	Rust, Huff, Solidity, Circom, Nim, TypeScript, SQL, GraphQL
Frameworks/Libraries	Libp2p, Circomlib, Arkworks, Gnark, Foundry, Hardhat, Terraform, ethers.js
Infra	Kubernetes, Docker, Sqlite, Postgres, Github Actions
Communication	English, Hindi, Kannada

TECHNICAL EXPERIENCE

Senior Protocol Engineer

2024 — Current

Fuel Labs

Remote

- Lead maintainer of [zkvm-primitives](#), the **Rust** implementation of primitives required to make Fuel Ignition a Stage 2 rollout.
- Implemented Block execution proofs and DA compression proofs in **SP1** and **Risc0** to make Fuel Ignition a Stage 2 rollout. While doing so, discovered several bugs in **SP1** and worked with their team to resolve them.
- Maintainer of [fuel-core](#), the **Rust** implementation of Fuel Ignition.
- Performed several benchmarks and increased transaction throughput for specific operations by upto **40%** using **SIMD** and modern CPU pipelining.
- Developed a custom synchronization primitive (**SeqLock**) that optimized concurrent thread read performance by reducing lock acquisition and release time, by upto **20%**.
- Developed and designed a dynamic gas pricing mechanism based on DA costs, thus reducing Fuel's costs to post blobs to DA by nearly **80%**.
- Designed a **Snap Sync** mechanism to allow node operators to **significantly** reduce sync time, taking inspiration from BitTorrent.
- Several high-severity bug fixes that would prevent blocks from being produced.
- Resolved **194** incidents, **166** MTTA (seconds), **24003** Average response effort (seconds), always on call.
- Responsible for the **EigenDA** integration, increasing throughput of the network by **8.3x**. Built an internal indexer since Eigen's dashboard did not provide sufficient data for our team.
- Lead maintainer of the backend for **O2**, A CLOB built on Fuel. Primary challenges faced here are data indexation and supporting high throughput.
- Designed an **overhaul of the p2p** network to move away from libp2p and to use [Iroh](#) instead to allow high throughput. See a demo [here](#) that streams fuel block headers straight to the browser.

Applied Crypto Engineer + Team Lead

2022 — 2024

Vac Research, Unit in Status

Remote

- Worked on enhancements and optimizations in [nwaku](#), the **Nim** implementation of Waku.
- Researched and Engineered the anonymous rate limiting protocol, **RLN** for use in Waku.
- Lead maintainer of the [zerokit](#) Rust library, using **Arkworks**.
- **Implemented** the **Stealth Address** protocol for 8+ curves in Rust.

Software Engineer

2021 - 2022

Connect Financial

Remote

- Wrote and Deployed an ERC-20 Staking Platform in **Solidity**.
- Architected, Engineered and Deployed a system of **50+** microservices to GKE required for advanced risk management and credit card settlements.
- Managed the above infrastructure using **Terraform**, GCP, and Github Actions, ensuring that there would be **no downtime** between upgrades, abiding by our SLAs.

Contract Software Engineer

2021

ZeroDao

Remote

- Wrote an **SDK** which utilized **libp2p** and **RenVM** to facilitate 0 confirmation multichain swaps

Junior Software Engineer

2020 — 2021

Framework Ventures

Remote

- Wrote Integrations for Popular DeFi protocols during DeFi Summer, like Compound, Balancer, Synthetix, etc in Js which were consumed by market making strategies.
- Managed Ethereum Node Infrastructure on GCP
- Wrote and Handled the Infrastructure for deployment of various services that took part in market making, using GKE and GCB.
- Wrote a highly efficient and lightweight data ingestion system in **Rust** to obtain market data from **10+** CEX's, with **100+** tickers each, which was later used by analysts for backtesting of strategies developed by Quants.
- Wrote an off-chain MultiSig that was used to prevent excess gas usage, bringing down fees by up to **66.66%**

+91 8310902220
UTC +5:30
aaryamannchallani@gmail.com

Aaryamann Challani

Applied Cryptography Engineer

x.com/p1ge0nh8er
github.com/rymnc
linkedin.com/in/aaryamannc

Contract Software Engineer

DIA Association

2020

Remote

- Wrote a **EVM-compatible bridge node** in JS, which is currently used in DIA's Oracle Network.
- Wrote on a **smart contract monitor** that is used by DIA to monitor the health and status of their contracts, which is used on their status page.

EDUCATION

Polkadot Blockchain Academy, Cohort 0, Cambridge

2022

Bachelor of Technology in Robotics and Cryptography, Manipal Institute of Technology

ACTIVITIES

Paper author: **Message Latency in Waku Relay with Rate Limiting Nullifiers**

2024

Presented RLN @ ProgCrypto Istanbul

2023

ZK Hack Istanbul (Winner): **Reinforced Concrete Implementations**

2023

Bitcoin Grants Round 10 Hackathon (Winner): **dodo-trading-monitor**

2021

Secretary General, International Society of Automation (ISA), Manipal

2021

Head of Web Development, Leaders of Tomorrow

2020

PROJECTS

- **hessian-rs**: **Rust** implementation of the paper: cryptography over twisted hessian curves of the ring $F_q[\epsilon]$
- **orderbook-rs**: A low-latency, high-throughput orderbook implementation in **Rust** for trading systems and exchange infrastructure. Can achieve 1.5M+ ops/sec which makes it competitive with professional implementations. Uses modern CPU techniques to achieve this performance.
- **fuel-core-inspector**: Tool to quickly visualize data stored by **fuel-core** in rocksdb.
- **fuel-core-backup-cli**: Tool to perform portable backups of rocksdb in a performant way.
- **poseidon-huff**: Highly gas-efficient implementation of the Poseidon hash function in Huff. 10k gas cheaper than the industry standard implementation.
- **reinforced-concrete-huff**: Highly gas-efficient implementation of the Reinforced Concrete hash function in Huff.
- **reinforced-concrete-impls**: Implementation of the Reinforced Concrete hash function in **Circom**, **Solidity** & **01js**.
- **bloom-filter-ts**: Ergonomic implementation of Bloom filters in **TypeScript**