Hecate2

Electrical and Computer Engineering

github.com/Hecate2 @ hecate2@qq.com SECRET_PHONE

Education

2016-2020 **Fudan University**, Bachelor of Science, Electronic and Information Science and Technology
2020-2021 **Georgia Institute of Technology**, Master, Electrical and Computer Engineering
(2-year program finished in 1 year)

Competences

Languages	Python, C#, Go, Solidity, JavaScript(TypeScript)(Vue3), Rust, C(++), MATLAB, verilog, Java...
Fields of Competence	blockchain (VM, DeFi, ...), c100k concurrency, web scraping, data-intricate web backend
Insania Fast Learner	Never fear knowledge! I swallow docs & source codes & papers and learn insanely fast!

Experiences

March 2025 – July 2025

Senior Smart Contract Engineer, mETH Ethereum Staking Infrastructure, github.com/mantle-lsp

- mETH backend service maintenance : A41, blockdaemon, Kraken, P2P, Stakefish; SSV validators; oracles; Pectra upgrade.
- mETH contract security : blocking transfers of sanctioned accounts; reviews of bug bounty reports.
- Simulation infrastructures of Ethereum validator depositing, attesting and exiting, on private or public networks, with fully automatic signature service.
- Optimization of Ethereum validator exit sweep delay timing.
- Researches on ethereum-package, Omnichain Fungible Token (OFT) and Lido v3.

Solidity | foundry | golang | Ethereum staking | mETH

May 2021 – February 2025

Blockchain developer, Smart Contracts, Infrastructures & Security Audits, neo.org

- Revolutionized testing & debugging tools for Neo blockchain : Fairy (chain node plug-in for RPC-based testing & debugging) with an automated client. Similar to but more powerful than hardhat & truffle.
- Neo smart contract assembly analyzer, optimizer, and vulnerability scanner, supporting a mixture of try-catch-finally and ordinary control flow.
- Researches on LLM, RAG and agent frameworks like LangGraph, Dify, ChromaDB and MongoDB. Preliminary construction for SpoonOS, a solution of MCP tools and LLM agents.
- Relational database implemented by smart contract and key-value database. Secondary index & range query supported by splay tree.
- Basic backend in Rust for cryptographic operations of Neo account. Contributions of Neo implementation in Rust.
- Miscellaneous docs & tutorials for debugging Neo blockchain at deepest source-code level.
- VM-like game skill engine, for arbitrary skill implementation.
- Miscellaneous DeFi & DAO smart contracts : NeoBurger, ruler protocol, Divisible NFT standard & loan, etc. Researched on MINLP algorithms to maximize profits, and zero-knowledge-proof airdrop contracts using Merkle tree to save GAS fees.
- Misc cryptography programs & tutorials : Ethereum ECrecover instruction in C# using Bouncy Castle; RSA keys from XML to PEM; Neo wallet to WIF format; zero-knowledge tutorial; helping design Ethereum-compatible anti-mev dBFT consensus protocol, using threshold encryption & distributed key generation.
- Helped analyze Neo's VM, garbage collection, etc. , which lead to reveals of security vulnerabilities; Helped find VM instruction-level issues in Neo's smart contract compilers.
- Misc Solidity contract tests using hardhat.
- Performed on-chain data analysis; monitored some popular DeFi protocols on Ethereum.
- Helped build a webpage-based functional demo utilizing Neo's features : NeoFS, digital ID system, oracle & smart contracts. Analyzed the front end (using React) and smart contract codes.

Neo blockchain | smart contract | DeFi | neo-vm | compiler | tester | debugger | security audit | NeoFS | Solidity | hardhat | C# | node.js | React | cryptography | consensus protocol | zero knowledge proof | MCP | LangGraph

June 2019

Ignareo, HTTP Spider Core of Ultimate Concurrency, 150+ stars

- Spiders packed in asynchronous tornado servers
- Distributed broker architecture : aspects packaged as micro–service nodes
- Covering shortages of scrapy : non-blocking sleep, user codes of better cohesion, freedom and tidiness; scalability with customized load–balancing
- Seamless compatibility with anything based on asyncio or gevent
- Combat-proven performance and stability

distributed | web spider | asyncio | aiohttp | gevent | requests | tornado

February 2020
June 2020

Intern backend developer, Federated Learning Backend with Data Privacy, www.points.org

A web service for cross–organization federated machine learning. Based on distributed datasets from different organizations, the training process should still keep strict data privacy of each data owner. The service is principally based on gRPC instead of HTTP.

- Basic but complete experience of commercial development flow : designing, coding, deploying, testing, pull request and code review. 111052++, 86888– lines of code commits in the main repository.
- Designed and wrote a user and API authorization backend. Wrote a data-usage approval backend.
- Referring to horizontal learning APIs, developed vertical learning APIs. These APIs mostly involve data manipulation, but do not include core encryption and secret–sharing algorithms for learning.
- Wrote a basic service to retrieve real-time logs from docker containers and feed them to the frontend.
- Developed basic automatic cross-system process tests for the APIs described above.
- Implemented a mechanism to stop distributed vertical or horizontal training.
- Built a basic client for on-line model inference service stress test, using Ignareo.
- Tentatively designed a RESTful service for data providers to manage data, using flask and swagger.
- Participated in software copyright registration and federated learning product standards.
- Wrote docs and user manuals.

protobuf | gRPC | sqlalchemy | redis (message queue) | tensorflow-federated | dockerfile | docker-compose | git | Flask | swagger | Go

August 2021
September 2021

Intern backend developer, Web Spider & Flask Backend for OpenMMLab Operation Service, SenseTime × Shanghai AI Lab

- Maintained a system to fetch information from OpenMMLab's Github repositories and display info on a web page; send weekly e-mail reports based on fetched Feishu docs. Fixed miscellaneous bugs and implemented back end features.
- Designed a process to migrate data from SQLite to MySQL (using DataGrip).
- Designed code for runtime multi-thread operations in Flask.

DataGrip | docker-compose | Flask | SQLAlchemy | apscheduler | requests

July 2019
August 2019

Intern developer, High Concurrency Web Spider for Monitoring, Johnson Controls

- Monitored tens of thousands of IoT devices with help of Ignareo.
- Analyzed anomaly and output results to Excel sheets.
- Provided a web service to browse and download the Excel files.

Ignareo | requests | tornado | xlwings