

# Project Presentation

# Cyborg Network

Decentralizing the Edge for everyone

Built with **substrate**





# About Us

- ▶ Cyborg Network is a blockchain project aims at developing a trusted platform for edge computing. It addresses the challenge of minimizing communication latency, and efficiently utilizing idle computing resources in the growing demand for edge computing.

# Problem

- ▶ Centralized providers in edge computing pose a risk of extracting excess value and hoarding user data, leading to market distortion and a concentration of power and control. To ensure the fair and secure distribution of computing resources, it is crucial to develop decentralized systems that prioritize user control and transparency.

# Solution

Unlock the potential of the Edge  
with Decentralized Computing

1

**Decentralized Governance**  
Decentralized governance framework for empowering user participation and preventing concentration in edge computing.

2

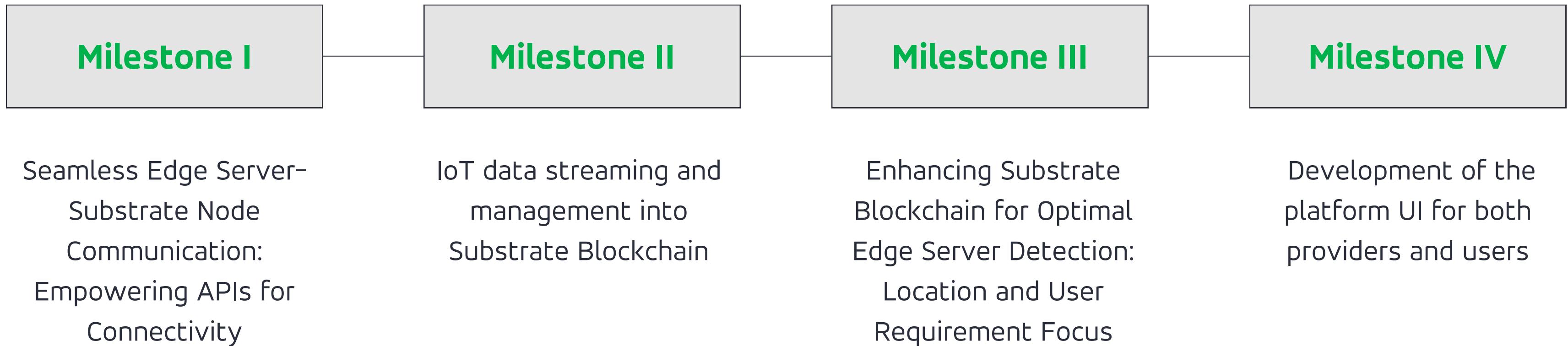
**Transparency**  
Transparent mechanisms for resource allocation, ensuring users have visibility into the distribution and utilization of resources, promoting fairness and user control.

3

**User Empowerment**  
Prioritizing user control and data ownership to foster an equitable and secure edge computing ecosystem.

# Milestones

## Unwavering Commitment to Our Long-Term Goals



# User Roles

Our users are categorized into the following



## Borgs

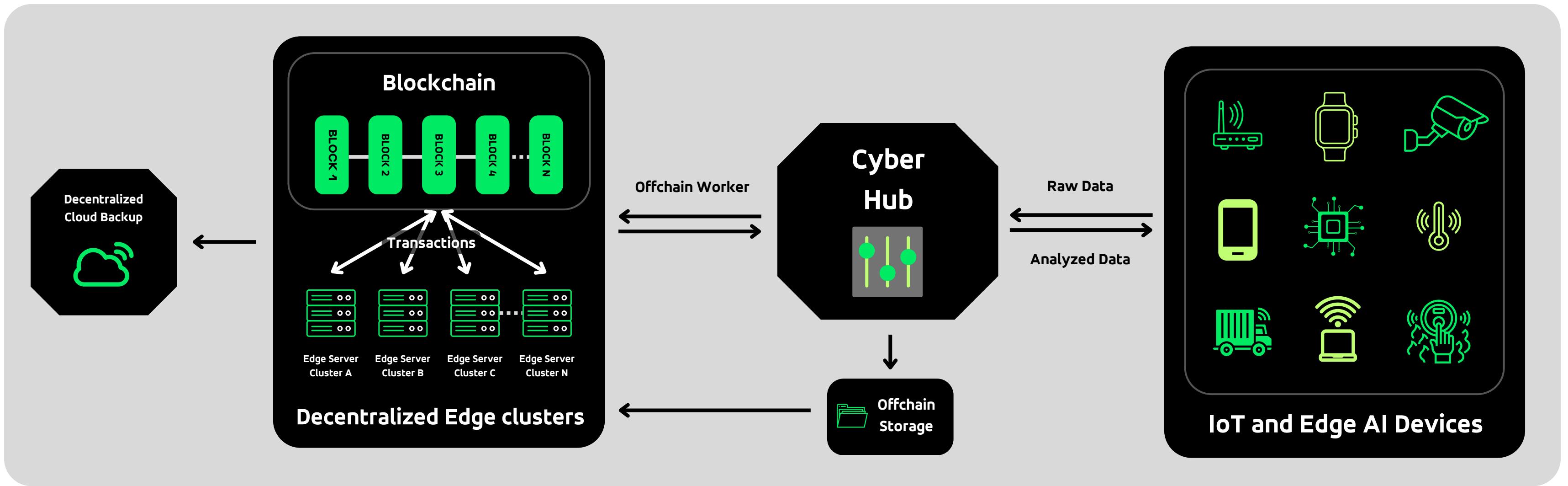
These are providers, who connect their spare computers and stable internet connections at their homes as edge servers to our network, have the opportunity to earn rewards based on our smart allocation algorithms. Playing a crucial role as the backbone of the Cyborg network, they form the foundation of seamless communication



## Titans

These end users will have the ability to leverage the network and take advantage of our decentralized web services, such as storage, computation, and data transmission at the edge. This will empower them to efficiently access and utilize our services while benefiting from the decentralized nature of the network, ensuring seamless and decentralized access to their data and computing needs.

# Architecture Diagram



# Project Demo

## Posting Edge server data into Substrate Blockchain

### Components

We are going to use 3 software elements to demonstrate how we can post data from an Edge server into a Substrate Blockchain using off chain worker module:

**Cyber Hub** - A hybrid server implementation supporting both web socket and HTTP. We will send commands to the Edge server which is connected as web socket client to the Cyber Hub.

**Cyborg Smart Client (CSC)** -A low level web socket client built with Rust to enable connection between Edge servers and Blockchain. We currently have 2 API methods (get server specifications and CLI) which can be called from Cyber Hub.

**Cyborg Node** - The node implementation of the Cyborg Blockchain with a new pallet called Edge Connect to listen to a HTTP endpoint and publish data on chain. (Our Current Submission)

# Project Demo

Posting Edge server data into  
Substrate Blockchain

## Workflow

- Compile and execute Cyborg node (Instructions in readme)
- Run Cyber Hub using Cargo run
- Compile and execute CSC (instructions in readme)
- Open localhost 9000 in any web browser
- Execute any of the 2 API methods
- Open localhost 8080/block to see the result of the command
- Check the logs of the Cyborg node to see the data being published to the Blockchain

# Project Demo

## Posting Edge server data into Substrate Blockchain

### Workflow

- Compile and execute Cyborg node (Instructions in readme)

```
cyborg-node on ✨ working-poc4 via 🚦 v1.69.0-nightly
❯ ./target/release/cyborg-node --dev --offchain-worker always
2023-05-28 19:34:27 Substrate Node
2023-05-28 19:34:27 🎉 version 0.1.0-dev-944fa30eb0e
2023-05-28 19:34:27 ❤️ by Cyborg Network <https://github.com/Cyborg-Network>, 2017-2023
2023-05-28 19:34:27 📜 Chain specification: Cyborg DevNet
2023-05-28 19:34:27 🏷️ Node name: female-table-0681
2023-05-28 19:34:27 🎖️ Role: AUTHORITY
2023-05-28 19:34:27 💾 Database: RocksDb at /tmp/substrate7HmtrM/chains/dev/db/full
2023-05-28 19:34:27 🛡️ Native runtime: cyborg-node-100 (cyborg-node-1.tx1.au1)
2023-05-28 19:34:27 🚀 Initializing Genesis block/state (state: 0xb44c...0de2, header-hash: 0x4c77...4dcc)
2023-05-28 19:34:27 🧑 Loading GRANDPA authority set from genesis on what appears to be first startup.
2023-05-28 19:34:28 Using default protocol ID "sup" because none is configured in the chain specs
2023-05-28 19:34:28 🗂️ Local node identity is: 12D3KooWQEzsUnZrNk71HQ7Cd6sxQ4hLKEkr4PNEbJXJyNrVvdge
2023-05-28 19:34:28 🖥️ Operating system: linux
2023-05-28 19:34:28 🖥️ CPU architecture: x86_64
2023-05-28 19:34:28 🖥️ Target environment: gnu
2023-05-28 19:34:28 🖥️ CPU: AMD Ryzen 7 5800H with Radeon Graphics
2023-05-28 19:34:28 🖥️ CPU cores: 8
2023-05-28 19:34:28 🌐 Memory: 31954MB
2023-05-28 19:34:28 🌐 Kernel: 5.19.0-42-generic
2023-05-28 19:34:28 🌐 Linux distribution: Ubuntu 22.04.2 LTS
2023-05-28 19:34:28 🌐 Virtual machine: no
2023-05-28 19:34:28 📦 Highest known block at #0
2023-05-28 19:34:28 🎉 Prometheus exporter started at 127.0.0.1:9615
2023-05-28 19:34:28 Running JSON-RPC HTTP server: addr=127.0.0.1:9933, allowed origins=["*"]
2023-05-28 19:34:28 Running JSON-RPC WS server: addr=127.0.0.1:9944, allowed origins=["*"]
2023-05-28 19:34:30 🎉 Starting consensus session on top of parent 0x4c77eb39220e20105608769cc2e176611edd9c2bfcfd1bf07935c5335f7864dcc
2023-05-28 19:34:30 🎁 Prepared block for proposing at 1 (0 ms) [hash: 0x6975497b7584eec549fb568b448c93e58ec24566e51ceaf4f86a199333c7547; parent_hash: 0x4c77...4dcc; extrinsics (1): [0x8948...f4e8]]
2023-05-28 19:34:30 🚚 Pre-sealed block for proposal at 1. Hash now 0x669d289ac3531f7681015bd1d2fbca394ccf230073efdfc8ddcf96a2098b97d2, previously 0x6975497b7584eec549fb568b448c93e58ec24566e51ceaf4f86a199333c7547.
2023-05-28 19:34:30 🎉 Imported #1 (0x669d...97d2)
2023-05-28 19:34:30 Hello from offchain workers!
2023-05-28 19:34:30 Requested started id=1816 method=GET uri=http://127.0.0.1:8080/block
```

# Project Demo

## Posting Edge server data into Substrate Blockchain

### Workflow

- Run Cyber Hub using Cargo run

```
/bin/bash 270x12
2023-05-28T11:24:49.825260244+00:00 INFO cyberhub::ws_handler:Received message: pong
2023-05-28T11:24:54.006584364+00:00 INFO cyberhub::http_handler:Received HTTP request: Request { method: GET, uri: /block, version: HTTP/1.1, headers: {"host": "127.0.0.1:8080"}, body: Body(Empty) }
2023-05-28T11:25:00.005366529+00:00 INFO cyberhub::http_handler:Received HTTP request: Request { method: GET, uri: /block, version: HTTP/1.1, headers: {"host": "127.0.0.1:8080"}, body: Body(Empty) }
2023-05-28T11:25:04.826465780+00:00 INFO cyberhub::ws_handler:Received message: pong
2023-05-28T11:25:06.008278734+00:00 INFO cyberhub::http_handler:Received HTTP request: Request { method: GET, uri: /block, version: HTTP/1.1, headers: {"host": "127.0.0.1:8080"}, body: Body(Empty) }
2023-05-28T11:25:12.007799352+00:00 INFO cyberhub::http_handler:Received HTTP request: Request { method: GET, uri: /block, version: HTTP/1.1, headers: {"host": "127.0.0.1:8080"}, body: Body(Empty) }
2023-05-28T11:25:18.008544921+00:00 INFO cyberhub::http_handler:Received HTTP request: Request { method: GET, uri: /block, version: HTTP/1.1, headers: {"host": "127.0.0.1:8080"}, body: Body(Empty) }
2023-05-28T11:25:18.009955585+00:00 INFO cyberhub::http_handler:Received HTTP request: Request { method: GET, uri: /block, version: HTTP/1.1, headers: {"host": "127.0.0.1:8080"}, body: Body(Empty) }
2023-05-28T11:25:19.827944854+00:00 INFO cyberhub::ws_handler:Received message: pong
2023-05-28T11:25:24.009457259+00:00 INFO cyberhub::http_handler:Received HTTP request: Request { method: GET, uri: /block, version: HTTP/1.1, headers: {"host": "127.0.0.1:8080"}, body: Body(Empty) }
2023-05-28T11:25:30.006663619+00:00 INFO cyberhub::http_handler:Received HTTP request: Request { method: GET, uri: /block, version: HTTP/1.1, headers: {"host": "127.0.0.1:8080"}, body: Body(Empty) }
```

- Compile and execute CSC (instructions in readme)

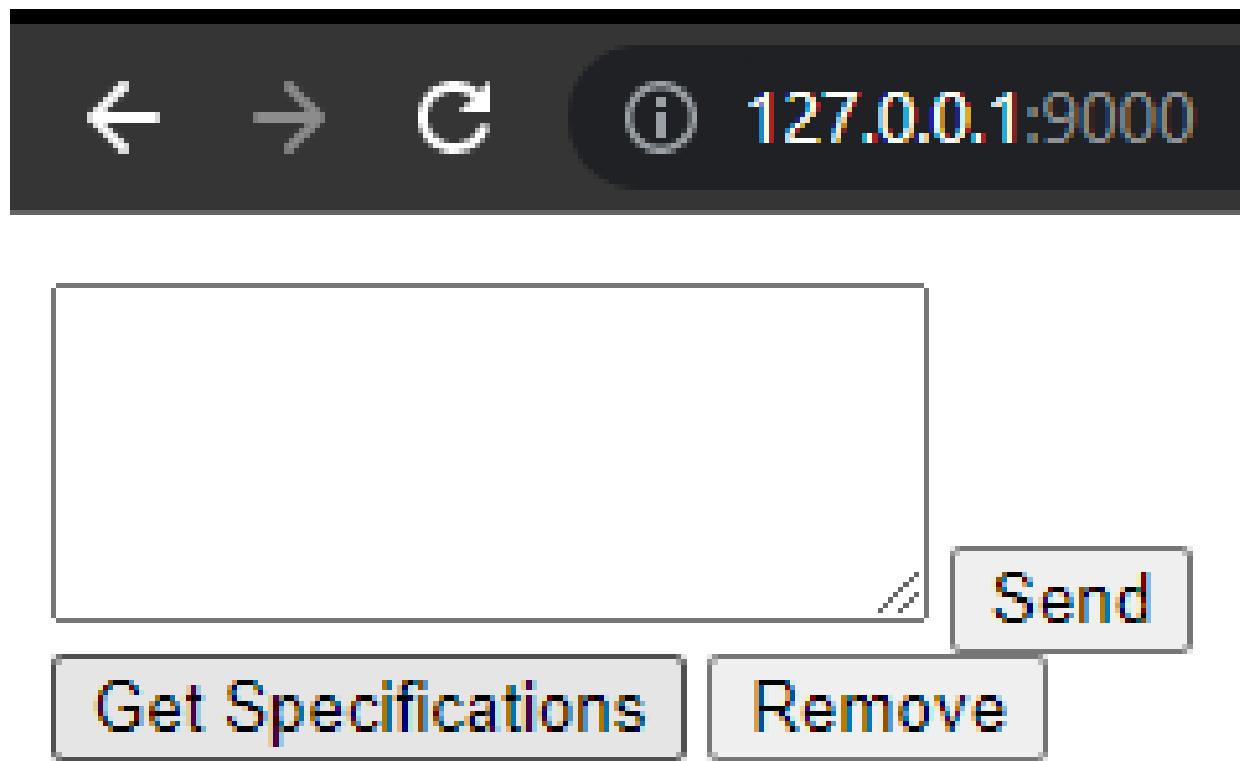
```
root@LLU333:/home/sha888/cyborg/Cyborg-smart-client 270x9
2023-05-28T11:24:49.824941005+00:00 INFO bin::client:Processed message: pong
2023-05-28T11:24:49.824953858+00:00 INFO bin::client:Processed message
2023-05-28T11:25:04.826067361+00:00 INFO bin::client:Received message: ping
2023-05-28T11:25:04.826112693+00:00 INFO bin::client:Processed message: pong
2023-05-28T11:25:04.826125056+00:00 INFO bin::client:Processed message
2023-05-28T11:25:19.827535143+00:00 INFO bin::client:Received message: ping
2023-05-28T11:25:19.827582009+00:00 INFO bin::client:Processed message: pong
2023-05-28T11:25:19.827595769+00:00 INFO bin::client:Processed message
```

# Project Demo

## Posting Edge server data into Substrate Blockchain

### Workflow

- Open localhost 9000 in any web browser
- Execute any of the 2 API methods
- Open localhost 8080/block to see the result of the command



```
127.0.0.1:8080/block
{
  "args": [],
  "data": {
    "cpus": [
      "AMD Ryzen 7 5800H with Radeon Graphics"
    ],
    "csc_connected": true,
    "csc_version": "0.1.0",
    "ip": "202.58.194.22",
    "kernel": "5.19.0-42-generic",
    "linux_version": "22.04",
    "memory": "31G",
    "networks": [
      "wlp4s0",
      "lo",
      "en0"
    ],
    "os": "Linux 22.04 Ubuntu",
    "security": 90,
    "serverhostname": "LLU333",
    "timezone": "Asia/Makassar"
  }
}
```

A screenshot of a terminal window showing the JSON output of the API call to "127.0.0.1:8080/block". The output is a nested JSON object with fields like "args", "data", "cpus", "csc\_connected", "csc\_version", "ip", "kernel", "linux\_version", "memory", "networks", "os", "security", "serverhostname", and "timezone".

# Project Demo

## Posting Edge server data into Substrate Blockchain

### Workflow

- Check the logs of the Cyborg node to see the data being published to the Blockchain

```
root@LLU333: /home/sha888/cyborg/Cyborg-smart-client
/bin/bash 270x29
2023-05-28 19:25:18 fetch_response: {"args":[],"data":{"cpus":["AMD Ryzen 7 5800H with Radeon Graphics"],"csc_connected":true,"csc_version":"0.1.0","ip":"202.58.194.22","kernel":"5.19.0-42-generic","linux_version":"22.04","memory":" 31G","networks":["wlp4s0","lo","eno1"],"os":"Linux 22.04 Ubuntu","security":90,"serverhostname":"LLU333","timezone":"Asia/Makassar"}, "id":"0fe743b6-8f83-42bd-b333-e148f5161545", "ref":"68f369a0-c77f-4f89-8d81-9464ce59a376", "status_code":200, "timestamp":1685273080576, "type":"ack"}]
2023-05-28 19:25:18 [d43593c715fdd31c61141abd04a99fd6822c8558854ccde39a5684e7a56da27d (5GrwvaEF...)] Submitted response: {"args":[],"data":{"cpus":["AMD Ryzen 7 5800H with Radeon Graphics"],"csc_connected":true,"csc_version":"0.1.0","ip":"202.58.194.22","kernel":"5.19.0-42-generic","linux_version":"22.04","memory":" 31G","networks":["wlp4s0","lo","eno1"],"os":"Linux 22.04 Ubuntu","security":90,"serverhostname":"LLU333","timezone":"Asia/Makassar"}, "id":"0fe743b6-8f83-42bd-b333-e148f5161545", "ref":"68f369a0-c77f-4f89-8d81-9464ce59a376", "status_code":200, "timestamp":1685273080576, "type":"ack"}]
2023-05-28 19:25:22 🚧 Idle (0 peers), best: #16 (0xb69a..b6ef), finalized #14 (0x4893..e845), ↓ 0 ↑ 0
2023-05-28 19:25:24 🎉 Starting consensus session on top of parent 0xb69a0358ab261ab463d10ee79aae993b4a8a130f97830630db6cd8844a5bb6ef
2023-05-28 19:25:24 🎉 Prepared block for proposing at 17 (1 ms) [hash: 0x5564328f16f8b3d8c44064b52155fc2882a4206a8caa5b6d3d88f88eec1b9fea; parent_hash: 0xb69a..b6ef; extrinsics (2): [0x4bb2..568d, 0xed0a..83f7]]
2023-05-28 19:25:24 🎉 Pre-sealed block for proposal at 17. Hash now 0x0f143d32f2d763868815b5d5dc62f6a6ce46c28650b9417de85fdc591017f3e0, previously 0x5564328f16f8b3d8c44064b52155fc2882a4206a8caa5b6d3d88f88eec1b9fea.
2023-05-28 19:25:24 🎉 Imported #17 (0x0f14..f3e0)
2023-05-28 19:25:24 Hello from offchain workers!
2023-05-28 19:25:24 Requested started id=218 method=GET uri=http://127.0.0.1:8080/block
2023-05-28 19:25:24 fetch_response: {"args":[],"data":{"cpus":["AMD Ryzen 7 5800H with Radeon Graphics"],"csc_connected":true,"csc_version":"0.1.0","ip":"202.58.194.22","kernel":"5.19.0-42-generic","linux_version":"22.04","memory":" 31G","networks":["wlp4s0","lo","eno1"],"os":"Linux 22.04 Ubuntu","security":90,"serverhostname":"LLU333","timezone":"Asia/Makassar"}, "id":"0fe743b6-8f83-42bd-b333-e148f5161545", "ref":"68f369a0-c77f-4f89-8d81-9464ce59a376", "status_code":200, "timestamp":1685273080576, "type":"ack"}]
2023-05-28 19:25:24 Response: {"args":[],"data":{"cpus":["AMD Ryzen 7 5800H with Radeon Graphics"],"csc_connected":true,"csc_version":"0.1.0","ip":"202.58.194.22","kernel":"5.19.0-42-generic","linux_version":"22.04","memory":" 31G","networks":["wlp4s0","lo","eno1"],"os":"Linux 22.04 Ubuntu","security":90,"serverhostname":"LLU333","timezone":"Asia/Makassar"}, "id":"0fe743b6-8f83-42bd-b333-e148f5161545", "ref":"68f369a0-c77f-4f89-8d81-9464ce59a376", "status_code":200, "timestamp":1685273080576, "type":"ack"}]
2023-05-28 19:25:27 🚧 Idle (0 peers), best: #17 (0x0f14..f3e0), finalized #15 (0x9cc7..5f23), ↓ 0 ↑ 0
2023-05-28 19:25:30 🎉 Starting consensus session on top of parent 0x0f143d32f2d763868815b5d5dc62f6a6ce46c28650b9417de85fdc591017f3e0
2023-05-28 19:25:30 🎉 Prepared block for proposing at 18 (0 ms) [hash: 0x7291c5b21226cc0505bbcbb5b6ba7178db60ebc35a5388fd742322567c6ac83; parent_hash: 0x0f14..f3e0; extrinsics (1): [0x4048..19b8]]
2023-05-28 19:25:30 🎉 Pre-sealed block for proposal at 18. Hash now 0xcff79fe624fed70a325233492fb94e6857065cd47a7a8iae29a1c2775f27603, previously 0x7291c5b21226cc0505bbcbb5b6ba7178db60ebc35a5388fd742322567c6ac83.
2023-05-28 19:25:30 🎉 Imported #18 (0xcff7..7603)
2023-05-28 19:25:30 Hello from offchain workers!
2023-05-28 19:25:30 Requested started id=1336 method=GET uri=http://127.0.0.1:8080/block
2023-05-28 19:25:30 fetch_response: {"args":[],"data":{"cpus":["AMD Ryzen 7 5800H with Radeon Graphics"],"csc_connected":true,"csc_version":"0.1.0","ip":"202.58.194.22","kernel":"5.19.0-42-generic","linux_version":"22.04","memory":" 31G","networks":["wlp4s0","lo","eno1"],"os":"Linux 22.04 Ubuntu","security":90,"serverhostname":"LLU333","timezone":"Asia/Makassar"}, "id":"0fe743b6-8f83-42bd-b333-e148f5161545", "ref":"68f369a0-c77f-4f89-8d81-9464ce59a376", "status_code":200, "timestamp":1685273080576, "type":"ack"}]
2023-05-28 19:25:30 Response: {"args":[],"data":{"cpus":["AMD Ryzen 7 5800H with Radeon Graphics"],"csc_connected":true,"csc_version":"0.1.0","ip":"202.58.194.22","kernel":"5.19.0-42-generic","linux_version":"22.04","memory":" 31G","networks":["wlp4s0","lo","eno1"],"os":"Linux 22.04 Ubuntu","security":90,"serverhostname":"LLU333","timezone":"Asia/Makassar"}, "id":"0fe743b6-8f83-42bd-b333-e148f5161545", "ref":"68f369a0-c77f-4f89-8d81-9464ce59a376", "status_code":200, "timestamp":1685273080576, "type":"ack"}]
2023-05-28 19:25:32 🚧 Idle (0 peers), best: #18 (0xcff7..7603), finalized #16 (0xb69a..b6ef), ↓ 0 ↑ 0
```



# Connect with us.



info@cyborgnetwork.io



Cyborg Network



@Cyborg\_network\_



Cyborg Network Official

