Incentivization of correct behavior in a decentralized computing marketplace

Henry MONT - Internship update 1

RedChainLab
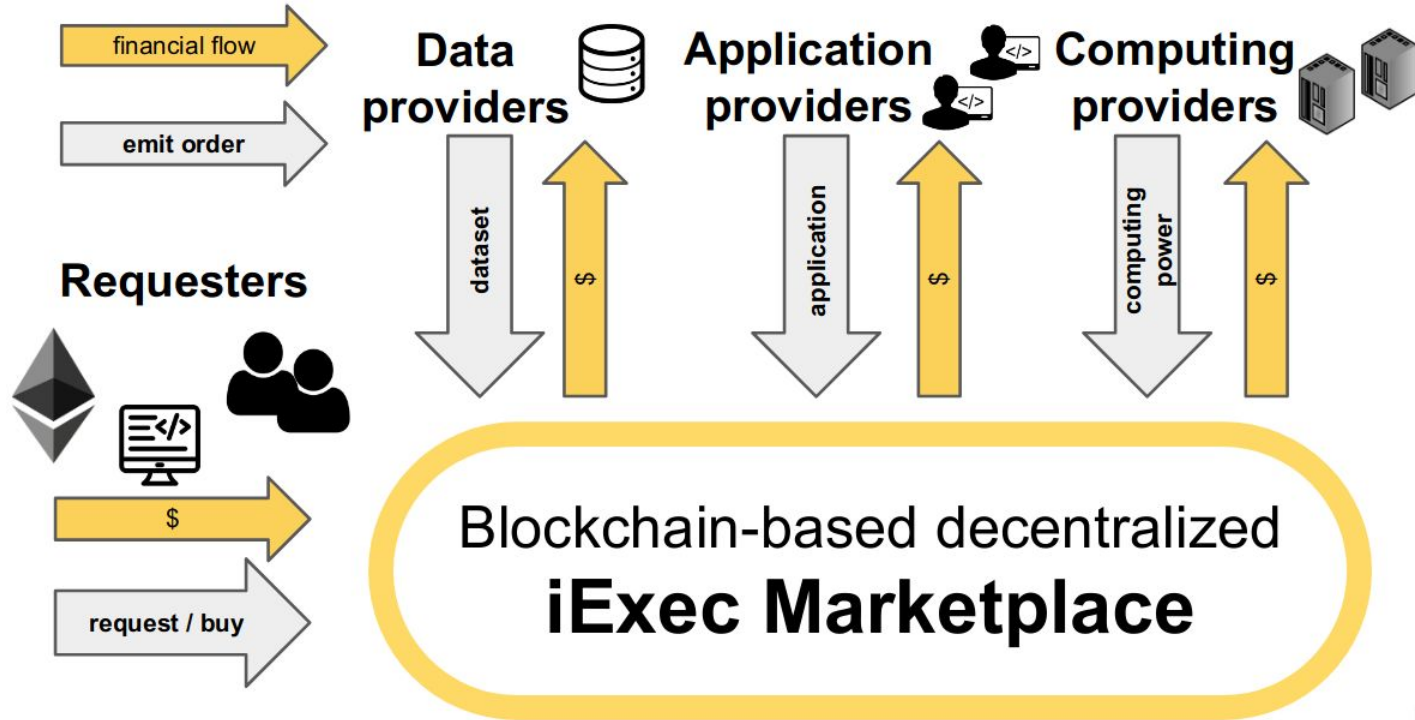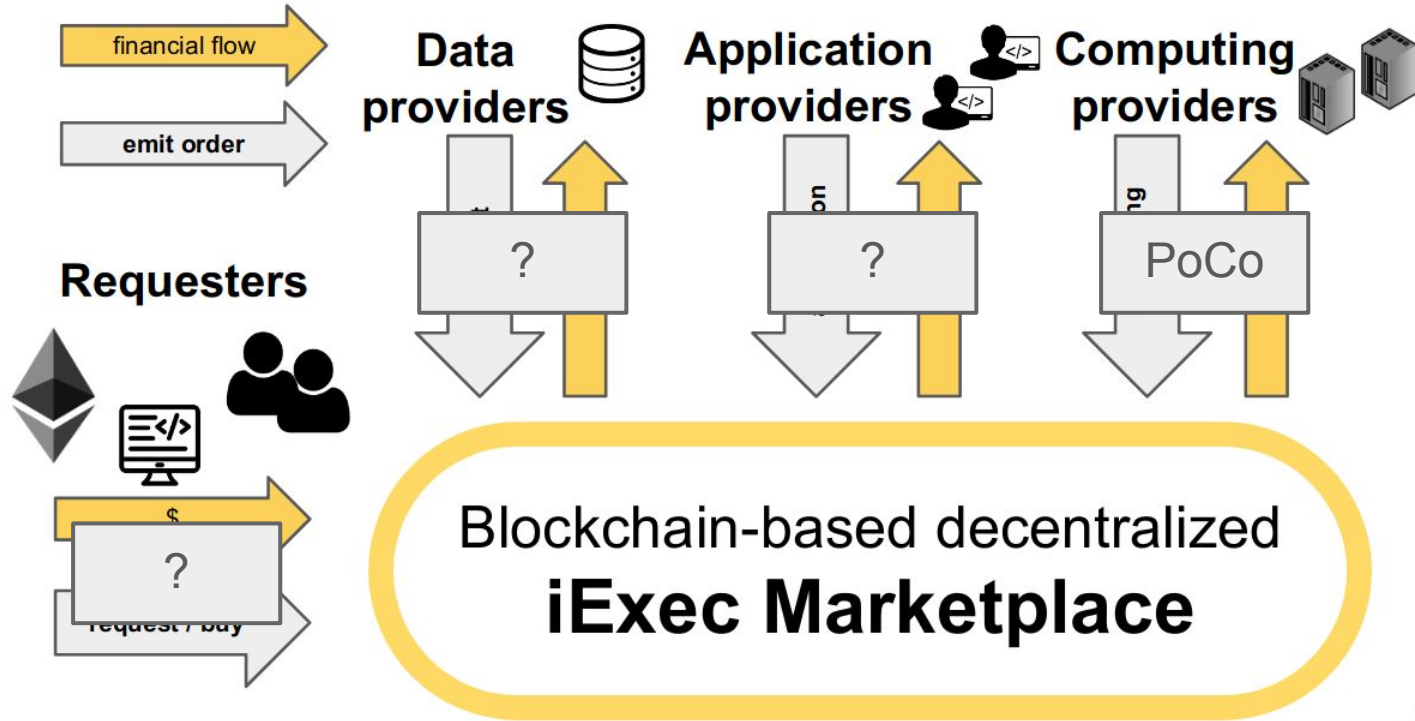
Quick summary of my internship topic

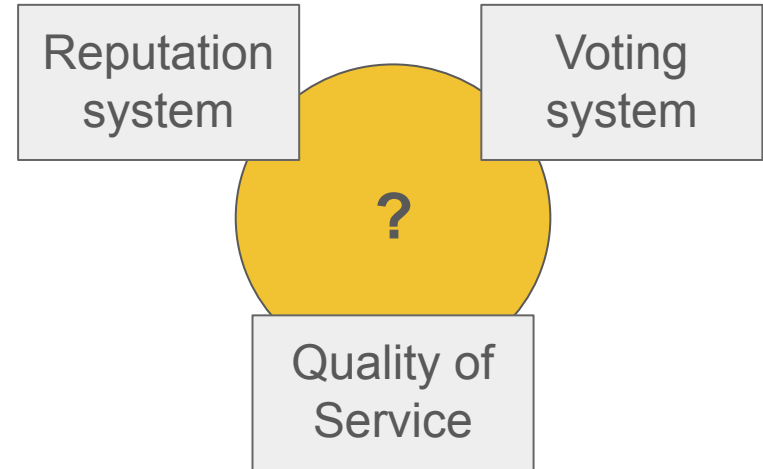Diagram from: Matthieu BETTINGER, Comité de suivi de thèse D3

Quick summary of my internship topic

In case of problem, who is at fault ?

We need:

- A way to check other actors' contributions.
- It needs to be as objective as possible.
- The system should deter each actor from malicious activities.

Reputation system

Voting system

?

Quality of Service

Actors and threats

Requesters:

- Access datasets without proper compensation.
- Access computing resources without proper compensation.
- Damage the reputation/finance of the marketplace.
- Alter the reputation/finance of other actors unethically.
- Earn money unethically.
- Use the platform for illegal purposes.

dApp providers:

- Access datasets without proper compensation.
- Access computing resources without proper compensation.
- Damage the reputation/finance of the marketplace.
- Alter the reputation/finance of other actors unethically.
- Earn money unethically.

Data providers:

- Damage the reputation of the marketplace.
- Alter the reputation/finance of other actors unethically.
- Earn money unethically.

RedChainLab

Vectors of attack

- Bound to fail tasks (with flawed app, flawed dataset or flawed request).
- Easy (virtually free) tasks to increase one's reputation.
- Denial of Service attack.
- Inserting Backdoors in dApps.
- Malicious app leaking information (in the result or through outside calls).
- Malicious app not executing the right code (using computing power for something else).
- Collusion Between Participants.
- Data Poisoning (providing fabricated or wrong data).
- Delaying computation on purpose to slow down task.

RedChainLab

Possible research

- Creating a voting process to audit or test code / data.
- Analyzing patterns across different requests to determine which component is most likely responsible for failures.
- Defining for which case the computing provider can be held responsible for failure. Otherwise, consider that the requester is responsible for task failure.
- Creating a trusted tier for Data and dApp which the user know for sure should work as intended (based on reputation/review/failure rate).

Possible research related to Requesters

- Detecting wrong / malformed request.
- Finding all the different kind of incentive mechanisms that could be applied in this context.
  - Moderation / voting system.
  - Reputation system.
  - Economic incentives

RedChainLab

Possible research related to dApps

- Characterizing app execution:
  - Defining Quality Metrics for dApps.
  - Tracking sensitive data during application execution.
  - Tracking I/O Operations.
  - Predicting Execution Time of Computer Programs.
  - Finding programming errors and bugs.
  - Establishing a testing pipeline for each type of dApps.
- Finding all the different kind of incentive mechanisms that could be applied in this context.
  - Moderation / voting system.
  - Reputation system.
  - Economic incentives.

RedChainLab

Possible research related to Data

- Characterizing data content:
  - Defining Quality Metrics for data.
  - Detecting fabricated or modified data.
  - Collecting user feedback on data.
  - Establishing a testing pipeline for each type of data.
- Finding all the different kind of incentive mechanisms that could be applied in this context.
  - Moderation / voting system.
  - Reputation system.
  - Economic incentives.

RedChainLab

Conclusion