한 번에 끝내는
블록체인 개발 A to Z

Chapter 2

Blockchain 2.0 - Ethereum

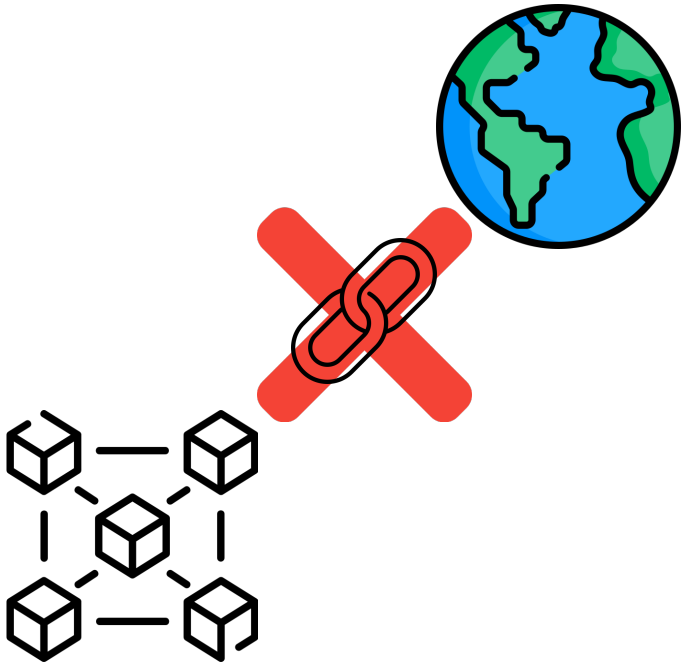
Chapter 2

Blockchain 2.0 - Ethereum

Oracle

Oracle 문제

- Ethereum 네트워크는 블록체인 내부 데이터에 대한 안전성을 보장하지만, 외부 실제 데이터에 대한 위변조에 대한 안전성을 보장할 수 없다.
- Contract가 API 등을 통해서 외부 서버와 직접적인 통신이 불가능하기 때문에, 특정 사용자가 외부 데이터를 네트워크상에 입력하여야 활용이 가능하다.



랜덤 생성 문제

❶ 동일한 결과

Smart Contract 상에서 트랜잭션을 검증하기 위해서는 모든 Smart Contract가 동일한 랜덤값을 생성해야 한다.

❷ 블록 Hash등을 활용

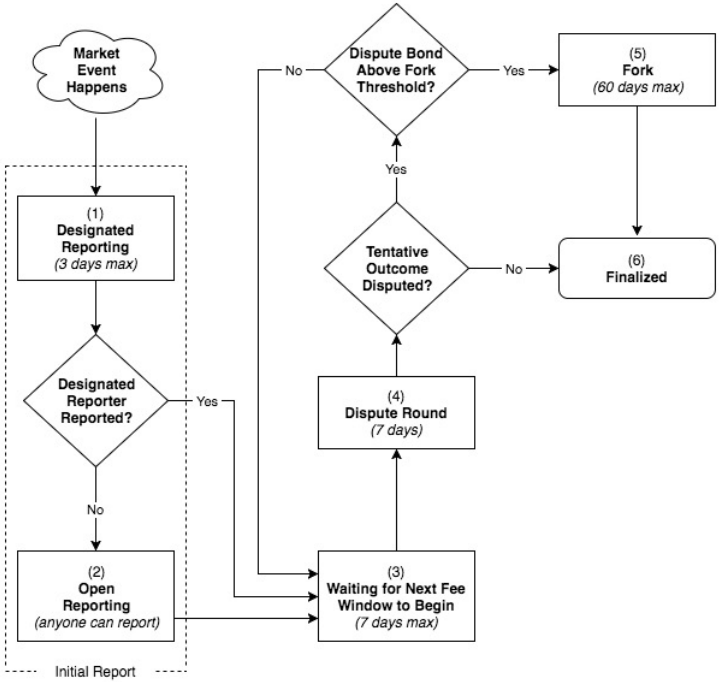
Hash 함수는 어느정도 랜덤값으로 활용이 가능하다. 하지만 동일한 블록 내에서는 동일한 Hash값을 이용할 수 있기 때문에 이를 통한 공격이 가능하다.

❸ 외부 랜덤 데이터의 필요성

Ethereum은 이러한 문제로 인하여 랜덤 함수를 제공하고 있지 않다. 따라서 랜덤을 통한 서비스를 제공하기 위해서는 안전한 랜덤값을 제공하는 Oracle 서비스를 이용해야 한다.

Augur

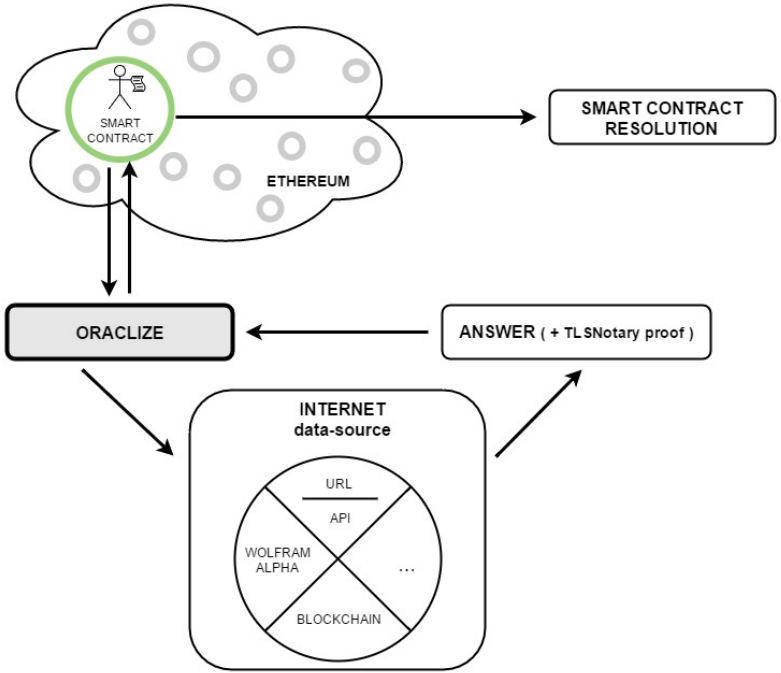
- 미래 예측 시장에 적용하는 블록체인 플랫폼이다. Ethereum 초창기 Dapp으로 Real World의 데이터를 블록체인으로 가져오는 방식을 구현하였다. (NFL 경기의 승패 등)
- Oracle 문제를 해결하기 위해서 경기 결과를 등록하고 수수료를 제공받는 Reporter를 구성하고 있으며, 잘못된 결과를 등록할 때 보유한 REP 토큰을 잃게 되는 구조로 안전성을 보장하고 있다.



(출처 : https://steemit.com/coinkorea/@piljae/augur)

Oraclize

- 외부 Real World 데이터를 대신 가져오는 기능을 제공하고 있다.
- API 형태 URL 을 통해서 Oraclize 서비스가 대신 외부 결과를 조회하고 이를 Smart Contract 상에 Call Back 형식으로 입력하는 방식이다.
- Oraclize의 중앙화 문제로 인해 데이터 위변조 시 문제가 발생할 가능성이 존재한다.



출처) oraclize.it

ChainLink

탈중앙화된 데이터 제공

OpenSource 방식으로 데이터 제공 서비스 소스를 공개하여 사용자들이 악성 코드 탐지가 가능하도록 하였다.

노드 분산

데이터 수집 노드를 분산시켜 단일 장애 위험을 제거하고, 데이터를 정해진 시간안에 제공

서명을 통한 인증

노드가 Smart Contract에 데이터 제공시에 서명을 통해서 어떤 노드가 데이터 제공하였는지 증적을 제공

평판 관리

잘못된 정보가 전달되지 경우나 데이터 전달의 지연이 발생했는지 정보를 통해 데이터 제공 노드의 평판을 관리

Defi의
Oracle 활용

- Defi에서 가장 중요한 것은 담보의 가치를 관리하는 것이다. 담보의 가치가 떨어지게 되면, 추가 담보를 요구하거나 청산을 진행해야 한다.
- 이를 위해서는 Real World에서의 담보의 가치를 알아야하기 때문에, 특정 시점마다 Oracle 서비스를 통해서 가격정보를 갱신하고 담보 가치를 재계산해야한다.