Uncensorable SNS

NonceNS

김성준
김수민
엄지용
이현민

기획 의도

- 인터넷은 의사소통의 기본 수단이 됨
- 네트워크는 어떠한 정치적인 입장도 갖고 있지 않지만 이를 관리하는 주체에 의해 언제든 정치적인 도구로 활용될 수 있음
- 최근 정부나 기업이 "좋은" 혹은 "나쁜" 정보를 분류하여 사람들에게 전달되는 것 자체를 검열, 차단하는 일이 잦음
- 처음에는 이러한 필터링이 편리하게 느껴질 수도 있지만 결국 정보를 처리하는 개개인의 능력을 약화시킴

기획 의도

- 특정 주체가 네트워크 상에서 정보의 흐름을 통제하는 것은 옳지 않음
- 특히 그것이 표현의 자유와 관련된 것일 경우 정보를 통제하는 사람은 언제든 본인의 견해와 다른 주장을 검열하고 차단하고자 하는 유혹을 받을 수 있음
- 개인은 자신에게 전달되는 정보를 수용하거나 거부할지 주체적으로 판단할 기회를 가져야 함
- 표현의 자유를 침해 받는 상황에서 블록체인의 투명성을 활용하여 검열 불가 SNS 기획

NonceNS

- No + Censorship + SNS

  포브스 “트럼프 대통령의 SNS 규제, 블록체인 SNS가 대안”

  中 코로나 19 검열에 성난 언론인, ‘이더리움’에 기록했다

- 사용자가 작성하는 글은 스마트 컨트랙트를 통해 이더리움 네트워크에 영구적으로 저장되고, 누구나 해당 데이터를 활용하여 소셜 네트워크 서비스를 개발할 수 있음

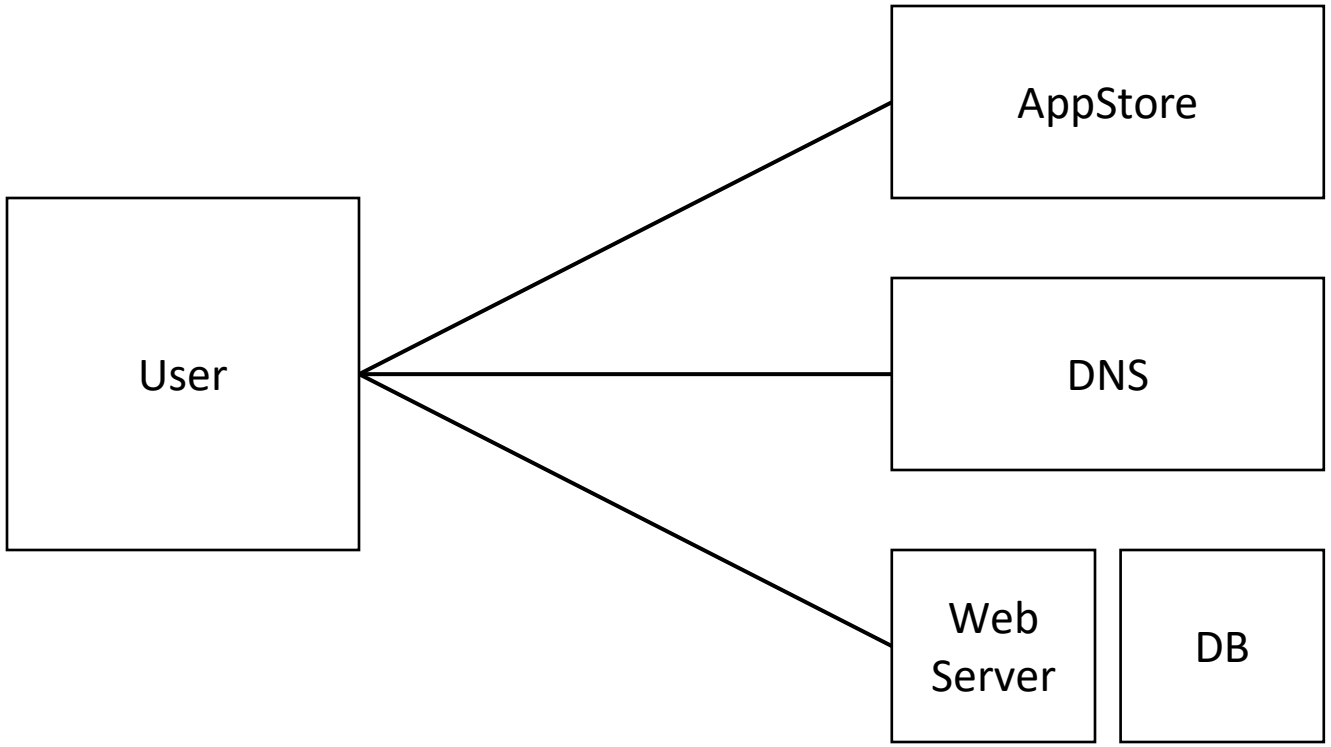
DApp

- 탈중앙화 애플리케이션 (Decentralized Application)
- 블록체인 같은 분산 시스템 위에서 작동하는 애플리케이션
- 사람에 따라 DApp이란 용어가 가리키는 범위가 조금씩 다르나 블록체인 분야에서는 스마트 컨트랙트 자체 또는 컨트랙트와 연동해서 작동하는 소프트웨어를 가리킴

DApp

- DApp은 사전에 작성된 규칙(Protocol)에 의해 예외없이 작동하며 특수한 권한을 가진 관리자(Middleman)가 동작에 관여할 수 없으므로 특정한 제 3자(Trusted 3rd Party)를 신뢰하지 않고도 사용 가능
- 설계에 따라 다르나 서비스 제공 프로세스 중간에 있는 단일 실패 지점(Single point of failure)을 제거하여 검열 불가능한(Uncensorable) 무중단 서비스를 제공할 수 있음

기존 SNS 작동 방식

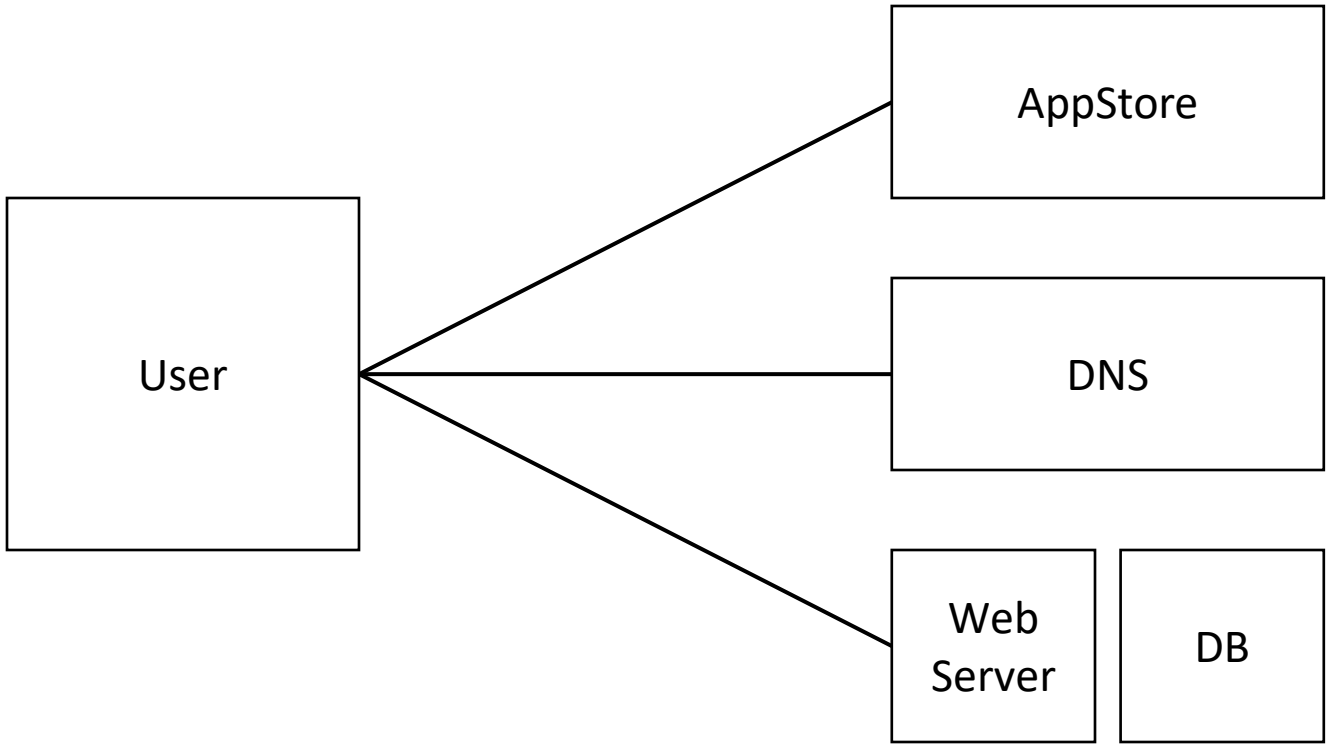


1. 앱스토어에서 App 다운로드
(Web 서비스의 경우 생략 가능)

2. www.facebook.com 과
같은 도메인을 157.240.11.35
같은 IP 주소로 변환

3. SNS 회사 서버 접속
- Web Server에서 Frontend
다운 (Web 서비스의 경우)
- App 또는 Web Frontend가
DB와 통신

기존 SNS 작동 방식

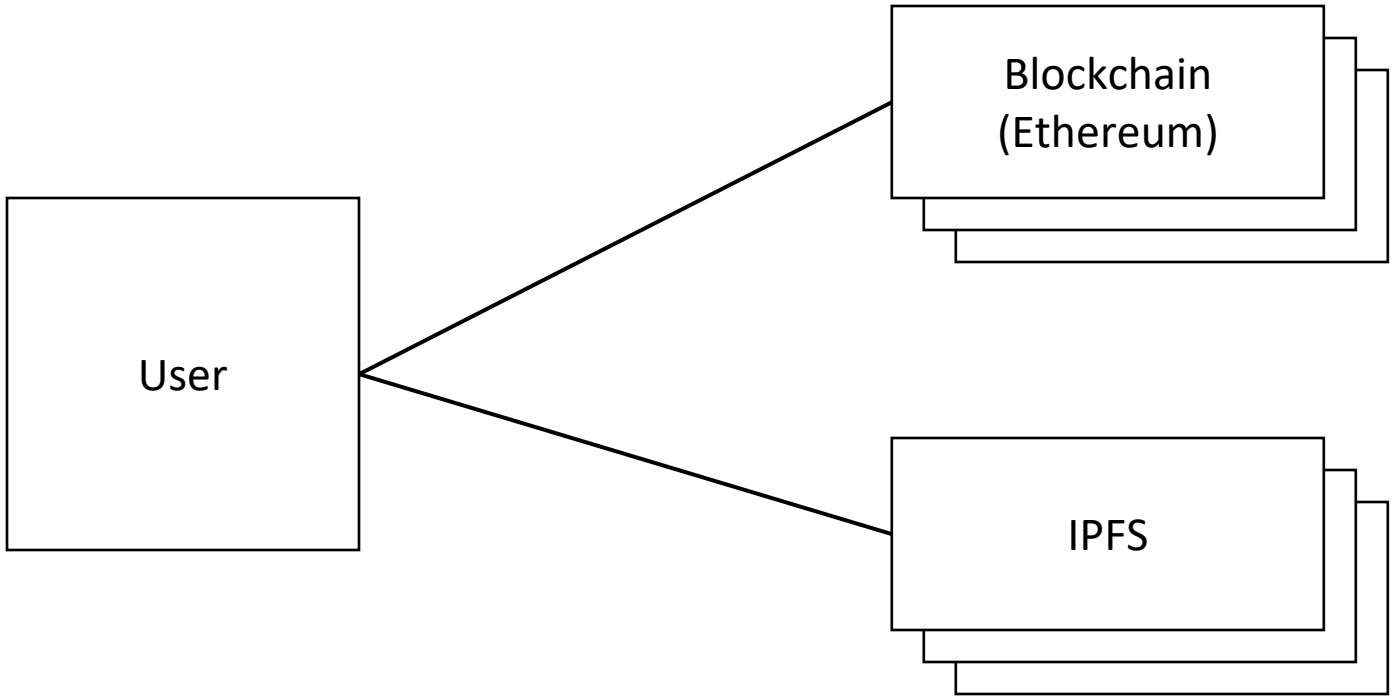


앱스토어 목록에서 삭제하거나
다운로드 차단

도메인 질의가 들어왔을 때
엉뚱한 IP 주소를 반환하거나
접속 차단

회사 측에서 정보를
검열/통제하거나 회사를 폐쇄

NonceNS 작동 방식



1. 스마트 컨트랙트에서 Web Frontend를 다운받기 위한 IPFS FileHash 조회

3. Web Frontend가 블록체인에 저장된 글을 보여주고 새로 작성할 수 있는 인터페이스 제공

2. FileHash 이용하여 Web Frontend 다운로드

1. Ethereum과 IPFS는 접속 가능한 지점이 무수히 많으므로 차단 불가능
2. 스마트 컨트랙트에는 누구나 글을 작성할 수 있음

추가 고려사항

- 글을 작성할 때마다 수수료(Ethereum을 사용하였으므로 Gas 비용 필요)를 내야 하므로 차단 불가능의 가치를 중요하게 생각하는 사용자가 지불할 수도 있고 프론트엔드를 작성하는 서비스 제공자가 이를 대신 지불할 수 있는 구조를 채택할 수도 있음
- 검열 불가능한 SNS가 갖는 목적, 활용 방식에 따라 비용 부담, 인센티브 제공 구조에 대한 별도의 고민 필요
- 글의 내용이 매우 긴 경우 글 자체도 IPFS로 배포하고 해당 FileHash를 글 내용에 기록하는 방식 사용 가능

추가 고려사항

- 스마트 컨트랙트에 누구나 글을 저장할 수 있으므로 정제되지 않은 글이 무분별하게 올라올 수 있음
- 따라서 프론트엔드 제공자는 특정한 주제나 규칙을 따른 글만 필터링해서 보여줄 수 있음 (그렇게 해도 검열/차단이라고 볼 수 없는 것은 해당 방식이 마음에 들지 않는 사람은 스마트 컨트랙트를 직접 조회하거나 다른 방식으로 작동하는 프론트엔드를 작성하면 됨)

NonceNS 스마트 컨트랙트

```
// 글 하나를 가리키는 자료 구조
struct postOne {
    address author;
    bool isImmutable;

    uint64 created;
    uint64 last_update;

    uint64 depth;
    uint children;
    uint parentId;

    string title;
    string body;
    string metadata;
}

// 새로운 글이 작성되거나 수정되면 발생하는 이벤트
event PostUpdated(address indexed author, uint indexed id, uint indexed parentId);

postOne[] public post;
mapping(address => uint[]) byAuthorIndex;
mapping(uint => uint[]) byParentIdIndex;
```

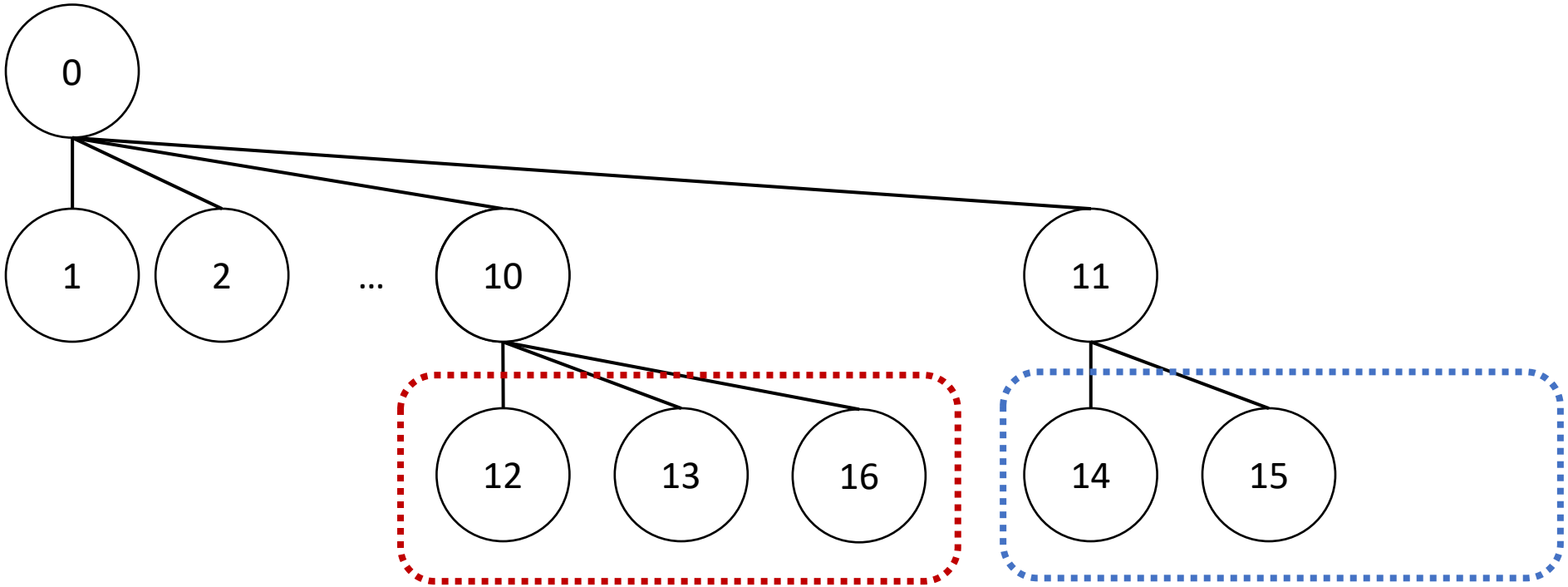
NonceNS 스마트 컨트랙트

```
// Frontend를 다운로드 받을 수 있는 IPFS FileHash 기록
mapping(address => string) public endpoint;
```

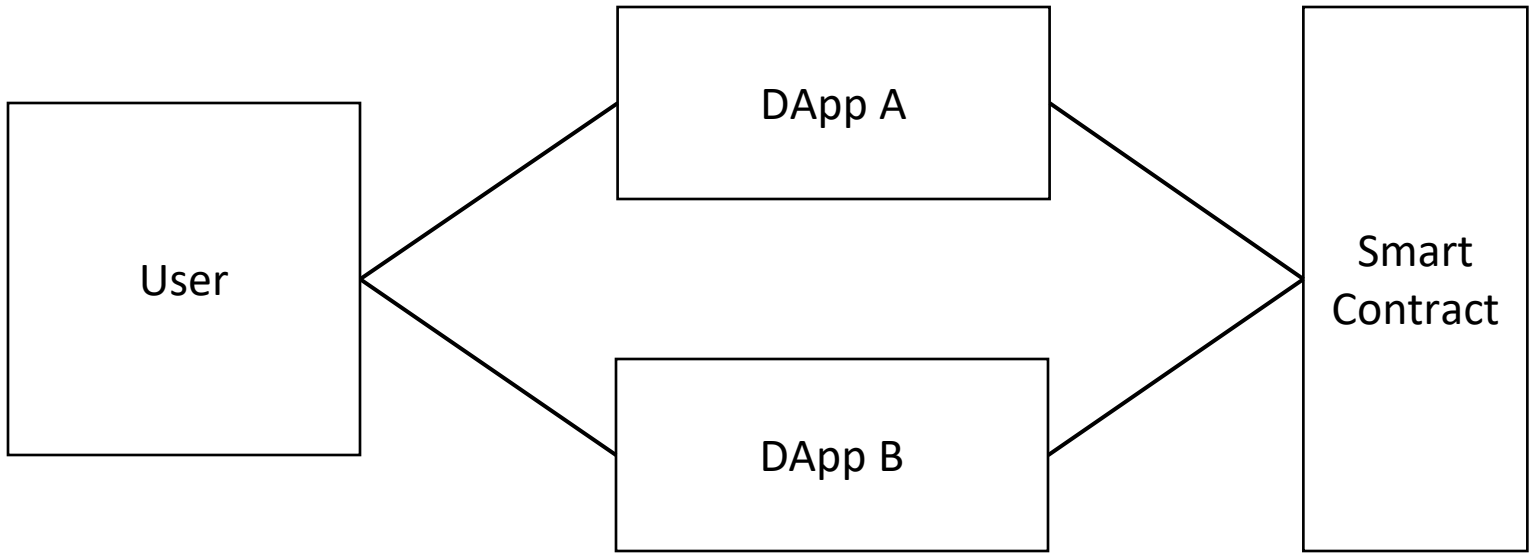
```
// 새로운 글을 작성하는 함수
// 컨트랙트를 배포하면 0번 글이 자동으로 생성되며 모든 글은 0번 글 또는 다른 글의 자식 글이 된다.
// 글의 부모-자식 관계를 활용하여 글-댓글 관계를 구현하거나 특정 주제의 글만 모아놓은 커뮤니티 기능을 구현할 수 있다.
function newPost(string memory body, uint parentId, string memory title, string memory metadata, bool isImmutable) public {
    require((parentId == 0) || (post[parentId].author != address(0x0)), "parentId doesn't exist");

    postOne memory p;

    p.author = msg.sender;
    p.created = uint64(now);
    p.last_update = uint64(now);

    p.title = title;
    p.body = body;
    p.metadata = metadata;

    p.isImmutable = isImmutable;

    post[parentId].children++;
```

isImmutable 값을 true로 설정하면 영구적으로 수정 불가능한 글로 작성한다

스마트 컨트랙트 공동 활용



스마트 컨트랙트 공동 활용



Demo

https://bit.ly/noncense1

https://bit.ly/noncense2

Q & A