Roll-DPoS

https://youtu.be/FySzgxCSNyU

PoS & DPoS

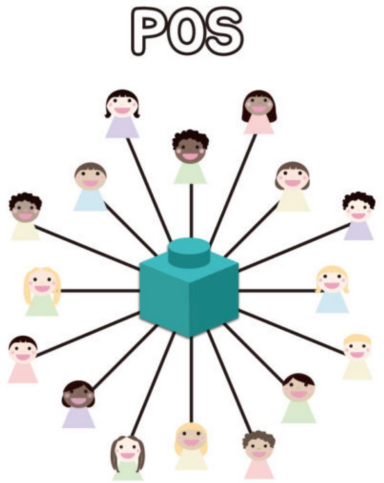
DKG

BLS 임계값 서명

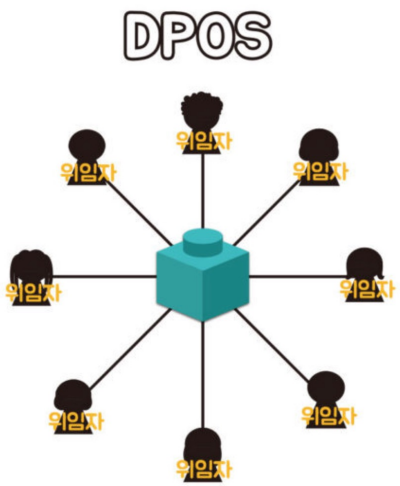
Roll-DPoS

PoS & DPoS

- **지분증명(Proof of Stake : PoS)**
  - 해당 암호화폐를 보유하고 있는 **지분율**에 비례하여 의사결정 권한을 주는 합의 알고리즘
    → 주주총회에서 주식 지분율에 비례하여 의사결정 권한을 가지는 것과 유사
  - 작업증명(PoW)와 다르게 채굴 과정 필요 X
    → 에너지 소모 X



- **위임지분증명(Delegated Proof of Stake : DPoS)**
  - 참여자들이 가진 코인의 지분율에 비례하여 투표권을 행사하여 **대표자 선출**
    → 선출된 대표자들이 새로운 블록의 유효성을 검증하는 과정에 참여
  - 대표자들만이 트랜잭션에 대한 검증에 참여
    → 속도 향상
  - 대표자가 블록을 성공적으로 생성할 경우 보상을 받음
    → 대표자는 본인에게 투표한 유권자들에게 보상을 분배
  - 대표자 선출은 라운드마다 진행



분산형 키 생성 (Distributed Key Generation)

- **DKG (Distributed Key Generation)**
  - 여러 참여자들이 다 같이 개인키와 공개키를 생성
  - 다른 암호화 모델과 달리 신뢰할 수 있는 제 3자가 존재 X
  - 한 명의 참여자가 개인키를 통째로 가지고 있을 수 없음
  - 공유된 암호문을 해독할 때 사용

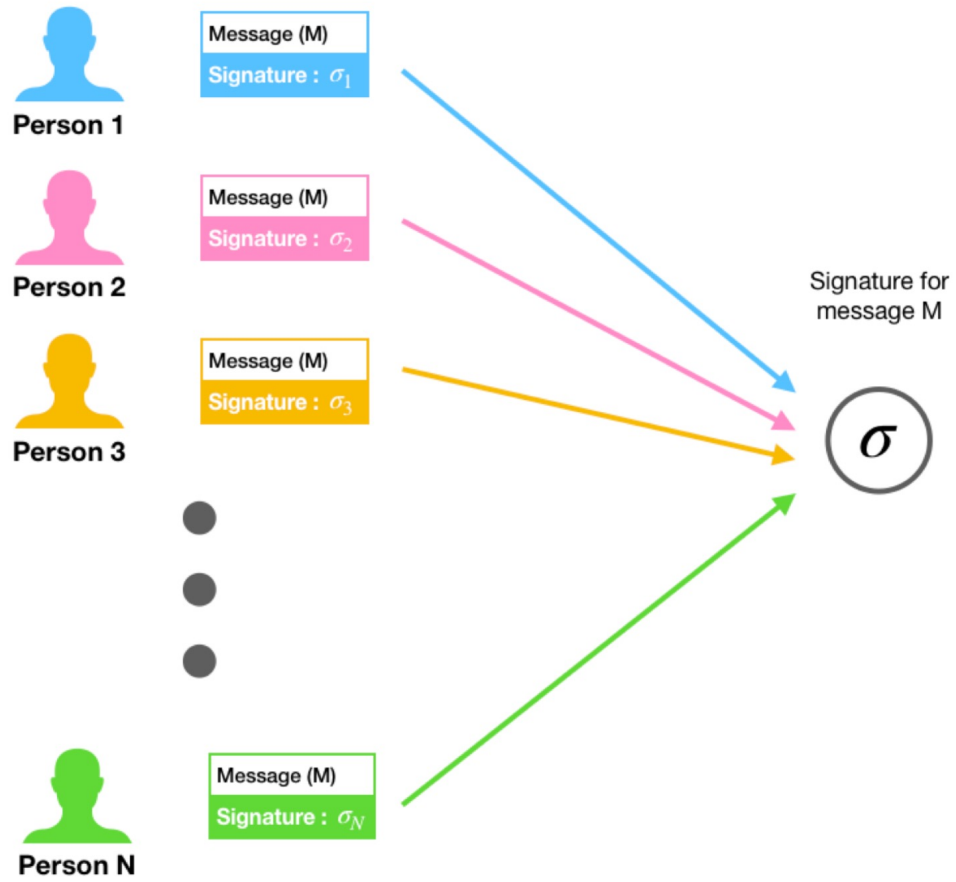
BLS 임계값 서명

- **BLS threshold signature**
  - N명의 그룹 참여자들이 개인키 S를 나눠서 소유
  - 각 참여자들 S의 일부분만 소유하고 있음
  - 그룹 서명이 필요할 경우 참여자들이 자신이 가지고 있는 키를 가지고 서명
  - N명 중 k명 이상이 서명했을 경우에만 유효한 서명

- **BLS 임계값 서명을 통한 난수 생성할 때 필요한 SEED 생성**
  - 난수 생성을 위해 사용할 특정 값이 있음 (seed)
  - 난수 생성의 의무가 있는 참여자들이 각각 해당 값에 대해 서명을 진행
  - K < =서명 수

    → 해당 값을 난수 생성에 이용 O
  - K > 서명 수

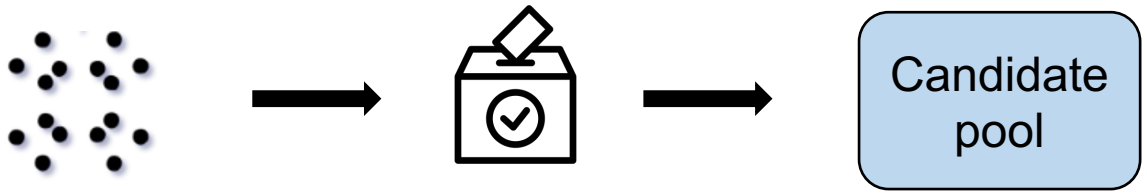
    → 해당 값을 난수 생성에 이용 X



임계값 서명(threshold signatures) 개념도, 이미지=디콘 제작

Roll-DPoS 과정

- **후보자 풀에 들어갈 노드 선택**
  1. **후보자 등록 :** 후보자들은 투표를 받기 위해 홍보 웹사이트 생성
  2. **투표 시작 :** 이더리움 투표 트랜잭션을 지지하는 후보에게 보냄으로써 투표함
  3. **투표 마감 :** 투표 시간이 끝나면 투표를 얼마나 받았는지 확인하기 위해 스냅샷을 찍음
     → 투표를 많이 받은 상위 N명의 노드가 후보자 풀에 들어감



Roll-DPoS 과정

- **BLS 임계값 서명을 통한 랜덤 비콘 생성**
  - DKG 방식으로 MNT 곡선으로부터 BLS 키를 생성
  - 블록 생성자가 이전 에폭의 랜덤 비콘과 BLS 키를 통해 현재 에폭의 랜덤 비콘 생성

$$s_j^{(i)} = \mathbf{SignShareGen}(sk_i^{(j)}, s_{j-1} \| j)$$

i : 블록 생성자 i

j : 현재 에폭

s : 랜덤 비콘

sk : MNT 곡선으로부터 생성한 키

Roll-DPoS 과정

- **랜덤 비콘으로 블록생성자 순서 선택**
  - DKG 방식으로 MNT 곡선으로부터 BLS 키를 생성
  - 블록 생성자가 이전 에폭의 랜덤 비콘과 BLS 키를 통해 현재 에폭의 랜덤 비콘 생성
  - 해당 랜덤 비콘을 SEED로 사용하여 DRBG를 통한 L-bit의 난수 생성

$$s_j^{(i)} = \mathbf{SignShareGen}(sk_i^{(j)}, s_{j-1} \| j)$$

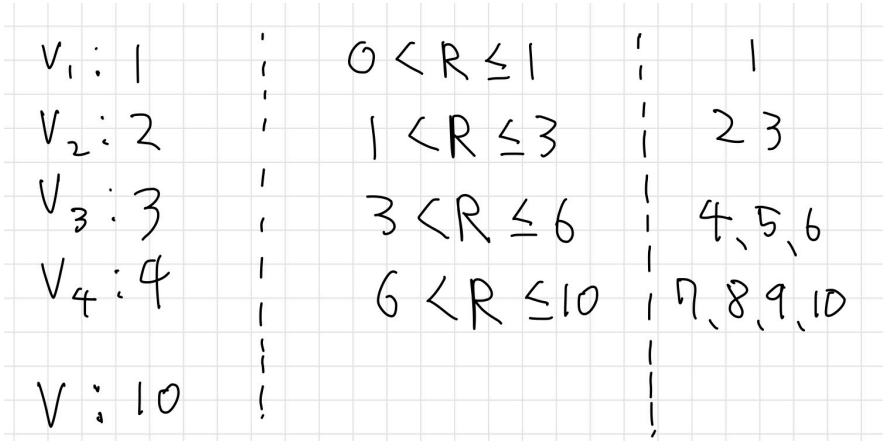
$$\mathbf{DRBG}(s_0, pk_1^{(1)}, \ldots, pk_N^{(1)}, 1)$$

$$R \bmod \mathrm{V}^{(1)} \in \begin{cases} \left(0, \mathrm{V}_1^{(1)}\right] & \text{if } j = 1 \\ \left(\sum_{i=1}^{j-1} \mathrm{V}_i^{(1)}, \sum_{i=1}^{j} \mathrm{V}_i^{(1)}\right] & \text{if } j > 1 \end{cases}$$

$V^{(1)}$: 총 투표 수

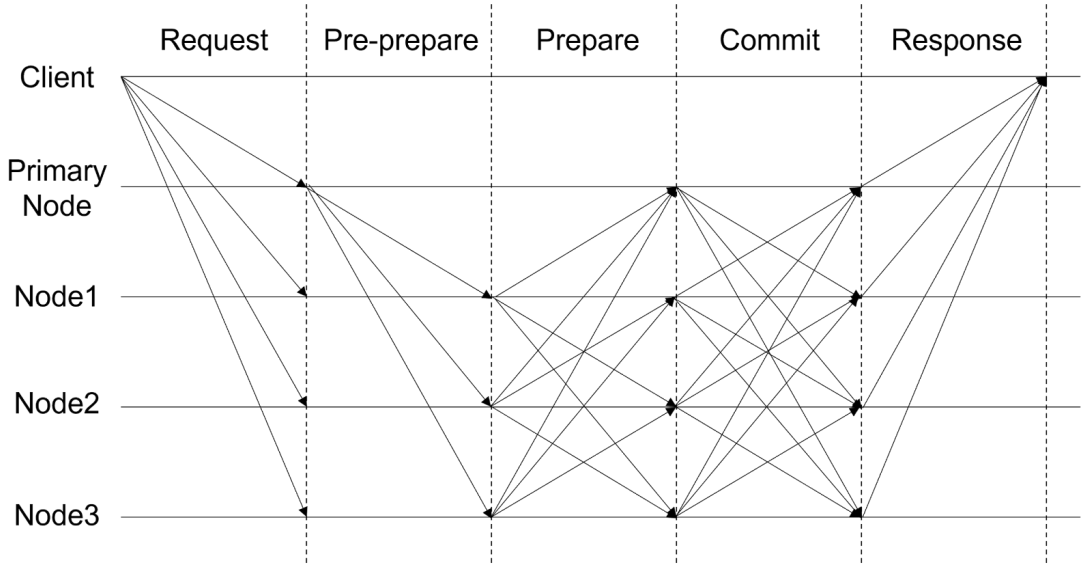
$V_i^{(j)}$: j에폭에서의 후보자 i 의 득표수

R : 난수



Roll-DPoS 과정

- **ECDSA 서명을 사용하여 PBFT 합의**
  - PBFT는 라운드로 실행되고 각 라운드는 3단계로 구성
  - **Pre-prepare 단계** : 리더 노드가 특정 수 만큼의 트랜잭션을 포함하고 있는 블록을 서명 후 제안 (signed Pre-prepare message)
  - **Prepare 단계 :** 리더로부터 받은 블록이 유효한지 확인하고 서명 후 브로드캐스트 (signed prepare message)
  - **Commit 단계 :** 다른 노드로부터 받은 블록이 유효한지 확인하고 서명 후 브로드캐스트 (signed commit message)
  - 최종적으로 충분한 수의 노드가 동의했을 경우 제안된 블록은 블록체인에 커밋



$$f = \lfloor (n-1)/3 \rfloor$$

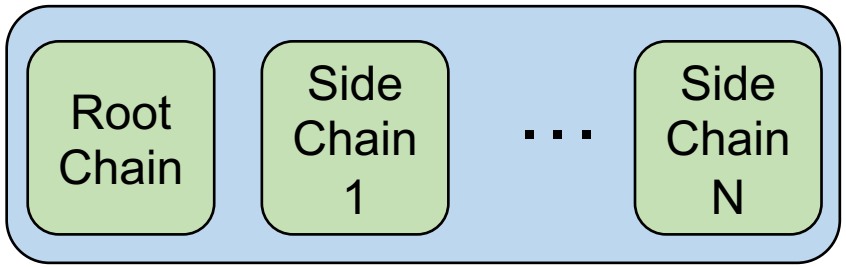
if $n = 10$

$f = 3$

$2f + 1$ digital signatures

Roll-DPoS 과정

- **복잡한 블록체인을 위한 자동 스케일링 후보자 풀**
  - 사이드체인의 개수에 따라 후보자 풀의 크기를 조절
  - 후보자들은 루트 체인의 블록 생성자가 될지 사이드 체인의 블록 생성자가 될지 선택
    → 결과적으로 후보자 풀은 여러 하위 그룹으로 나뉨 (루트체인, 사이드체인1 … 사이드체인 N)
  - 만약 블록 생성을 안 하고 쉬고 있는 노드가 임계값보다
    → 많으면 특정 수 만큼의 노드를 후보자 풀에서 내쫓음
    → 적으면 특정 수 만큼의 노드를 후보자 풀에 넣음
  - 합의 프로세스에 참여하는 노드를 증가 → 확장성⬆
  - 많은 노드들이 블록 생성자가 되어 보상을 받을 수 있게 할 수 있음



Candidate Pool

Q & A