신뢰 실행환경 기반 블록체인 동향

https://youtu.be/VKOWF5d7QrM

서론

신뢰 실행환경이란

신뢰 실행환경 기반 블록체인

서론

- 기존 블록체인의 문제점
  - 트랜잭션이 원장에 그대로 저장됨
  - IoT 환경 내의 악의적인 노드를 탐지하기 어려움
  - 블록체인 내 데이터에 대한 신뢰성을 보장할 수 없음(오라클 문제)
  - 낮은 트랜잭션 처리량

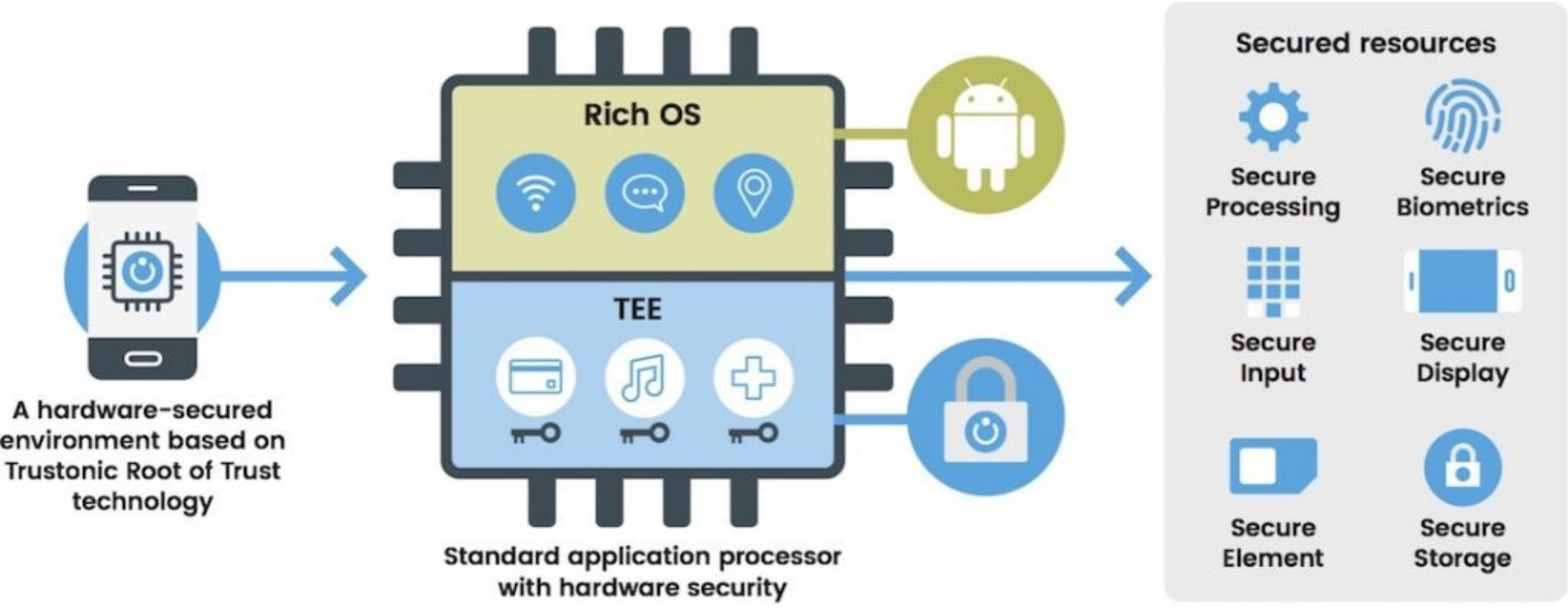
신뢰 실행환경 기반 블록체인을 통해 해결

서론

	BDTF	Truxen	Teegraph
문제점	거래에 대한 신뢰성을 보장할 수 없음	낮은 트랜잭션 처리량,악의적인 노드 탐지가 어려움	낮은 트랜잭션 처리량, 신뢰할 수 없는 중개자, 단일 노드 포크 공격
목적	데이터 거래 플랫폼	적대적 공격 방지	고효율성 합의 알고리즘, 단일 노드 포크 공격 방지
실행환경	**TEE**	**TPM**	**TEE**

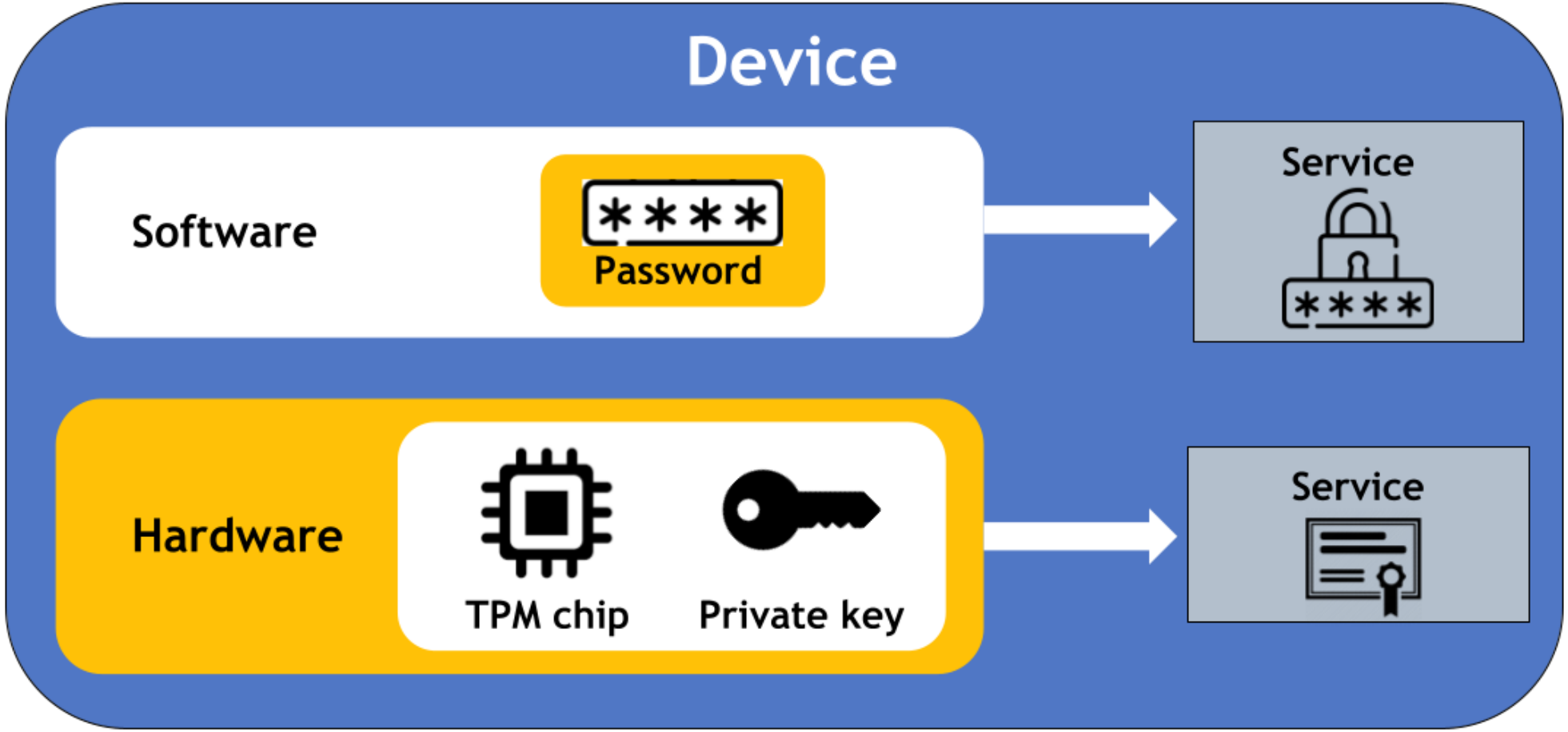
신뢰 실행환경이란

- Trusted Execution Environment (TEE)
  - Ex) Intel SGX, ARM TrustZone



신뢰 실행환경이란

- Trusted Platform Module (TPM)



TEE Blockchain-based Data Trading Framework(BDTF)

- BDTF
  - **데이터 재판매, 데이터 전송 거부** 문제
  - Trusted Data Trading Platform(TDTP)을 통한 거래
    - 합의 노드 / 거래소 노드
  - Ethereum, Intel SGX상에 구현
  - Matching 단계, Trading 단계
  - Seller, Buyer, TDTP로 구성

TABLE I
SUMMARY OF NOTATIONS

Notation	Explanation
A_{role}	Blockchain address of role
IP_{role}	The IP address of role
$E_{(role1,role2)}$	Evidence of payment from role1 to role2
K_{role}	An AES-256 key created by role
ID	Transaction id generated by trusted exchange
P	The price of data

TEE Blockchain-based Data Trading Framework(BDTF)

1. Matching

- Step 1: Buyer는 원하는 데이터에 대해 Message (P, IP_{buyer}) 브로드캐스팅한다.

- Step 2: Seller는 본인이 보유한 데이터인지, 적절한 가격인지 확인한다.

- Step 3: Seller는 IP_{buyer}를 통해 Buyer에게 A_{seller}, IP_{seller} 전송

- Step 4: Buyer는 손익과 TDTP내에 기록되어 있는 Seller의 평판에 따라 Seller 결정

- Step 5: Buyer는 IP_{seller}를 통해 Seller에게 거래에 대해 알린 후, 신뢰할 수 있는 거래소를 결정한다.

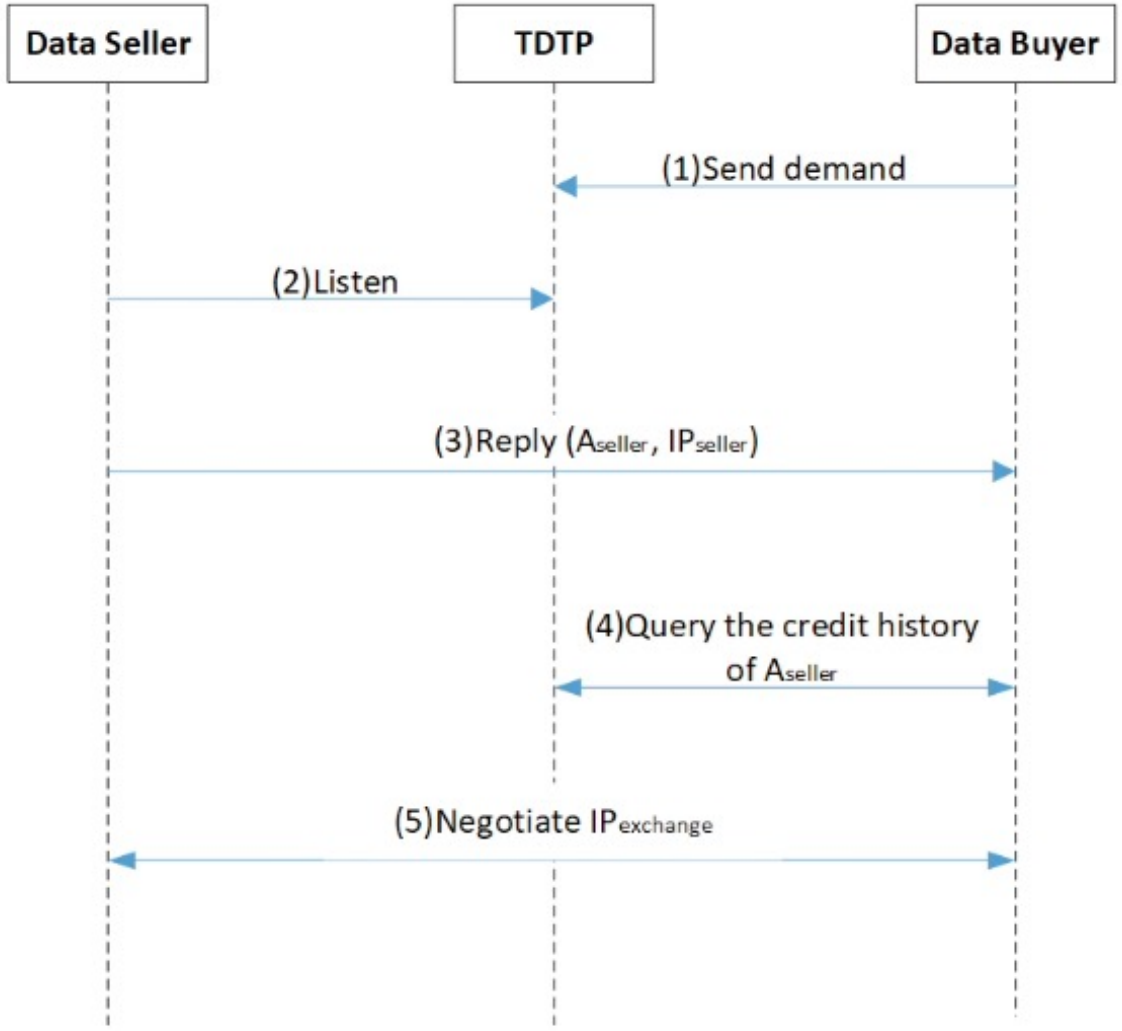


Fig. 2. The process of requirements matching in TDTP

TEE Blockchain-based Data Trading Framework(BDTF)

2. Trading

- Step 1: Buyer는 거래소 내의 거래 프로그램을 검증하기 위해 TEE enclave의 remote attestation을 수행
- Step 2: Buyer는 $A_{exchange}$를 통해 신뢰할 수 있는 거래소의 주인에게 ether를 입금한다.
- Step 3: Buyer는 이에 대한 증거로써 $E_{(buyer,\ exchange)}$를 생성한 후, enclave에 $E_{(buyer,\ exchange)}$, P, A_{buyer}, A_{seller}를 전송한다.
  - $E_{(buyer,\ exchange)}$는 입금 transaction과 block의 index로 구성된다.
- Step 4: $E_{(buyer,\ exchange)}$가 유효할 경우, enclave는 유저가 서비스를 사용하도록 허가되었는지에 대해 알려주는 id인 ID를 생성한 후 해당 ID를 Buyer에게 전송한다.
  - enclave는 $(ID,\ P,\ A_{buyer},\ A_{seller},\ IP_{buyer},\ timestamp)$를 pending transaction table에 기록한다.
- Step 5: Buyer는 Seller에게 ID와 AES-256 key인 K_{buyer}를 전송한다.

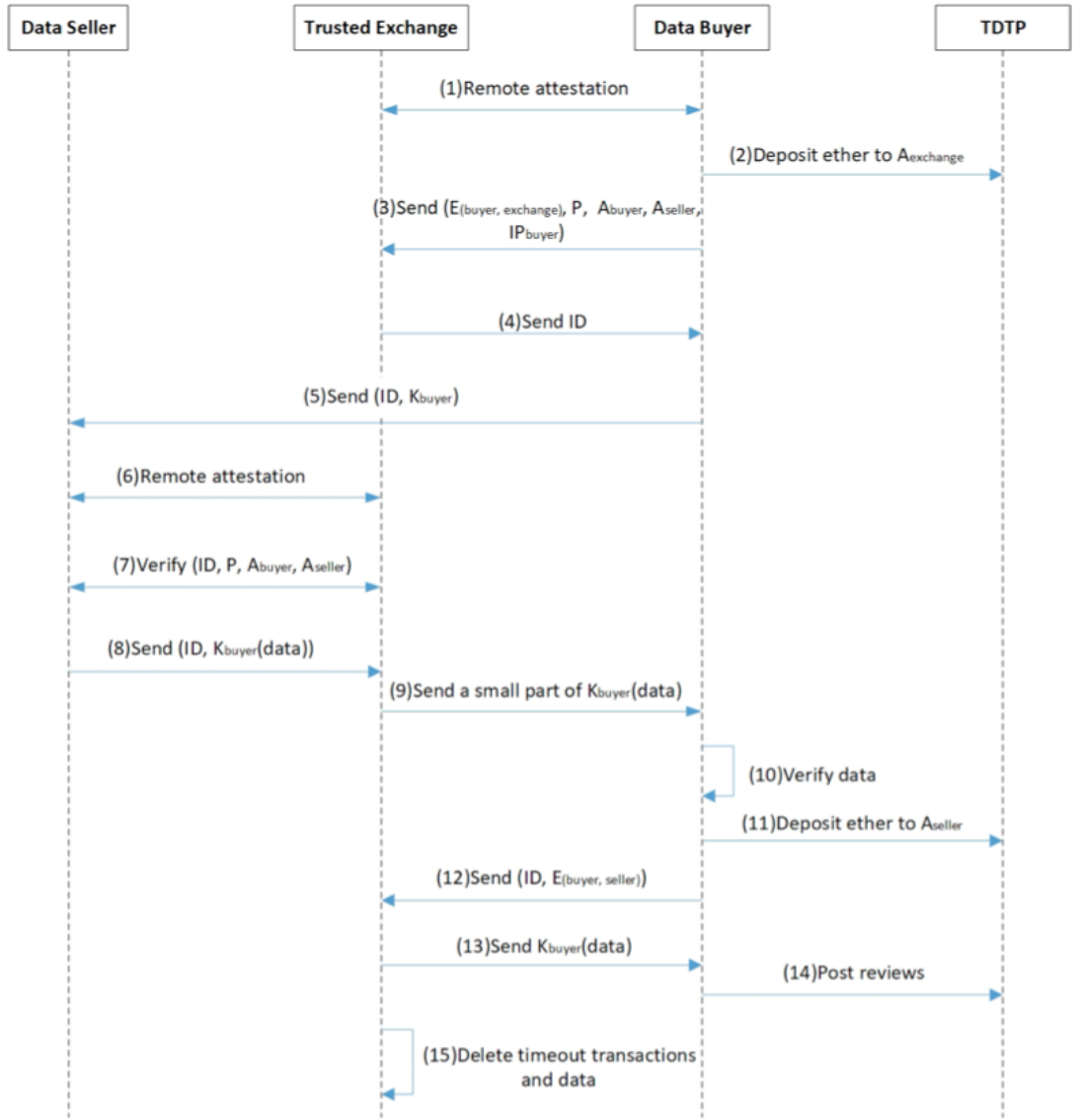


Fig. 3. The process of trading in TDTP

TEE Blockchain-based Data Trading Framework(BDTF)

2. Trading

- Step 6: Seller는 거래 프로그램을 검증하기 위해 TEE enclave의 remote attestation을 수행
- Step 7: Seller는 $(ID,\ P,\ A_{buyer},\ A_{seller},\ IP_{buyer})$가 올바른지 검증한다.
- Step 8: Seller는 K_{buyer}를 사용하여 데이터를 암호화한 후, 해당 데이터를 ID와 함께 enclave에 전송한다.
- Step 9: enclave는 받은 ID를 통해 pending transaction table을 참조하고, IP_{buyer}를 통해 데이터의 일부분을 Buyer에게 전송한다.
- Step 10: Buyer는 수신된 데이터를 복호화한 후, 원하는 데이터인지 확인한다.

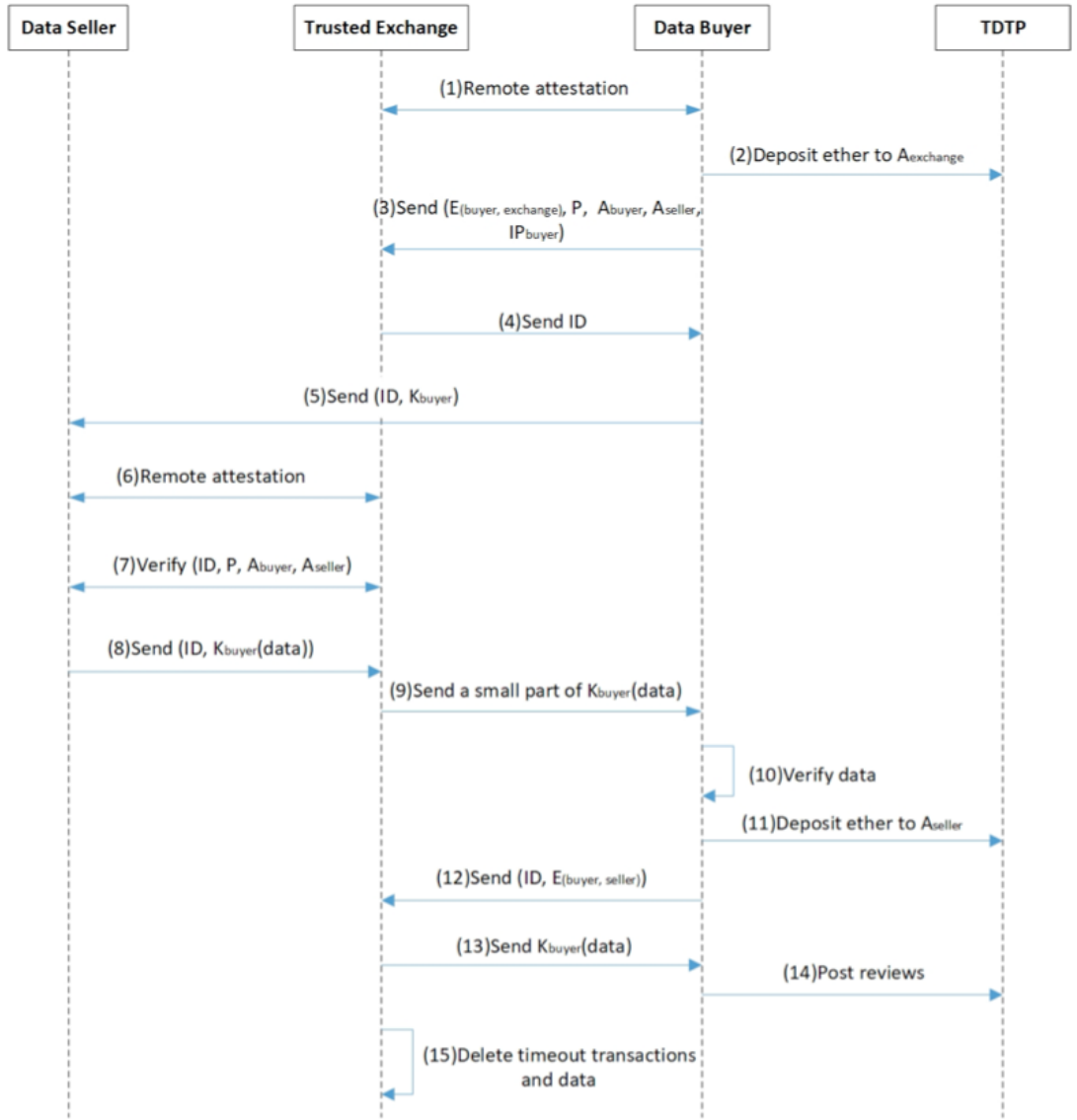


Fig. 3. The process of trading in TDTP

TEE Blockchain-based Data Trading Framework(BDTF)

2. Trading

- Step 11: Buyer는 A_{seller}를 통해 Seller에게 입금한다.
- Step 12: 입금이 완료된 후, Buyer는 enclave에게 ID와 $E_{(buyer,\ seller)}$를 보낸다.
- Step 13: enclave는 P, A_{buyer}, A_{seller}가 pending transaction table와 동일한지 검증한 후, IP_{buyer}에게 데이터를 전송한다.
- Step 14: 거래가 끝난 후, Buyer는 Seller와 거래소에 대한 리뷰를 남길 수 있으며 TDTP에 기록된다.
- Step 15: enclave는 데이터와 완료된 거래를 pending transaction table 상에서 삭제한다.

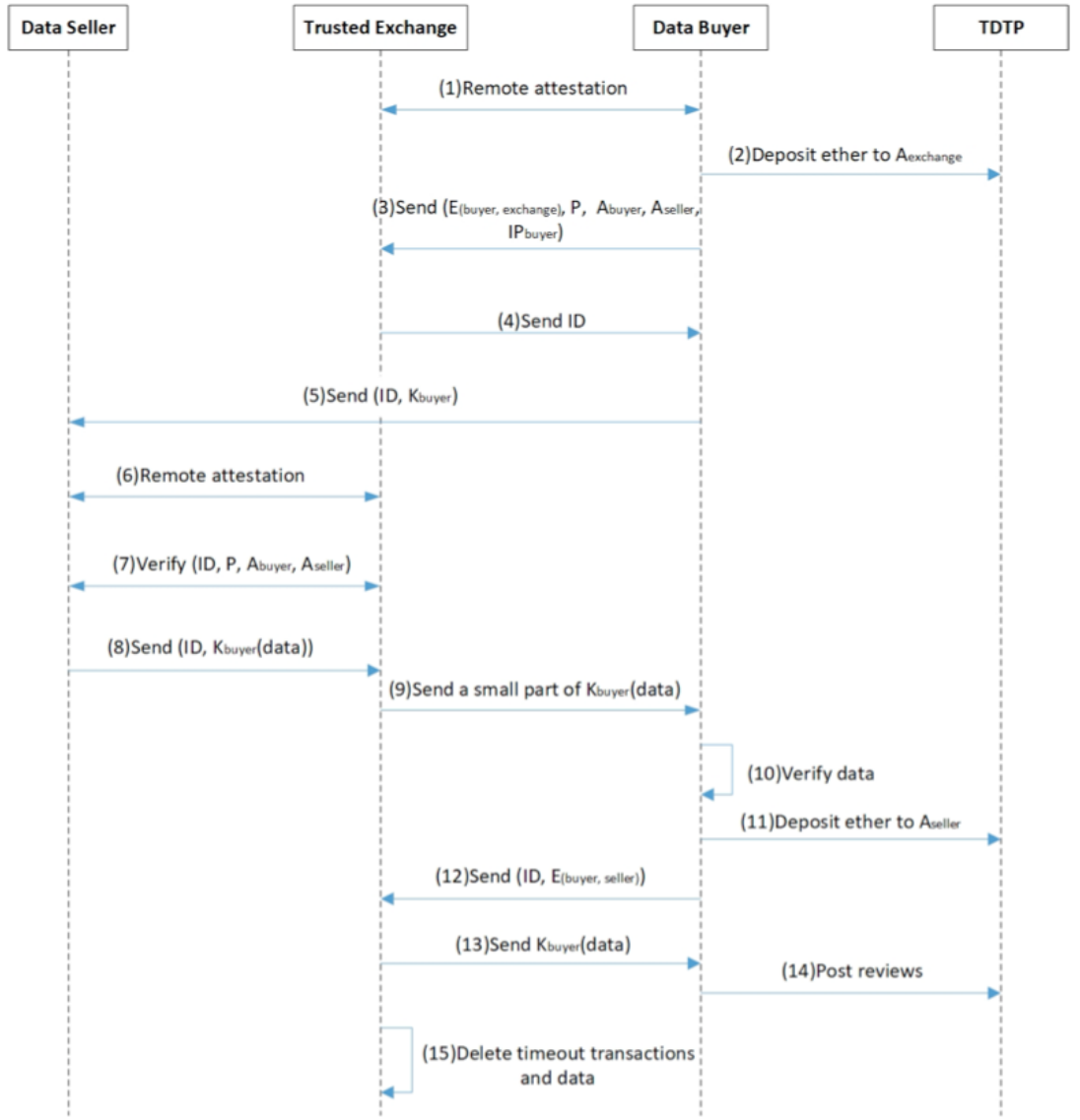


Fig. 3. The process of trading in TDTP

TEE Blockchain-based Data Trading Framework(BDTF)

- **보안**
  - 악의적인 데이터 판매자 → 불일치하는 데이터 판매 방지
  - 악의적인 데이터 구매자 → 결제 거부 방지
  - 악의적인 거래소 → 데이터 재판매 방지

- **성능**
  - Ethereum

  : 15~25TPS

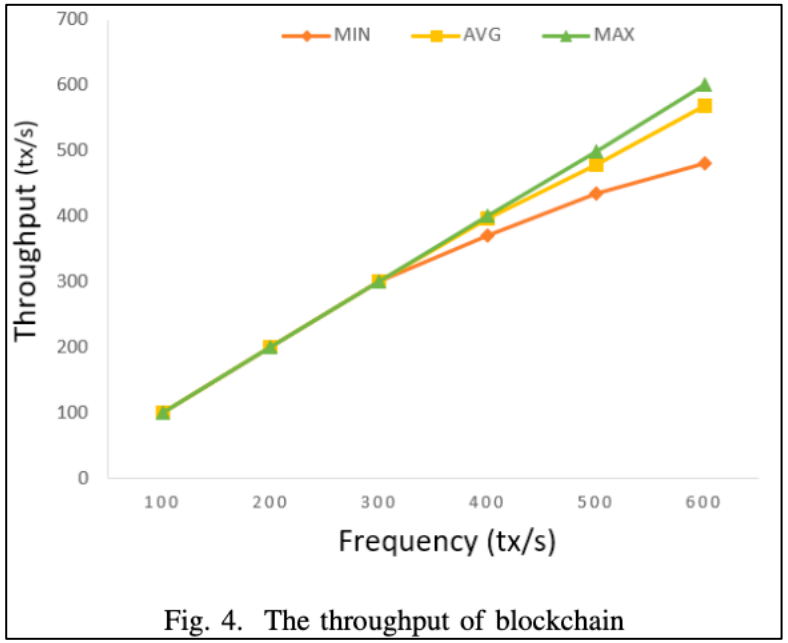


Fig. 4. The throughput of blockchain

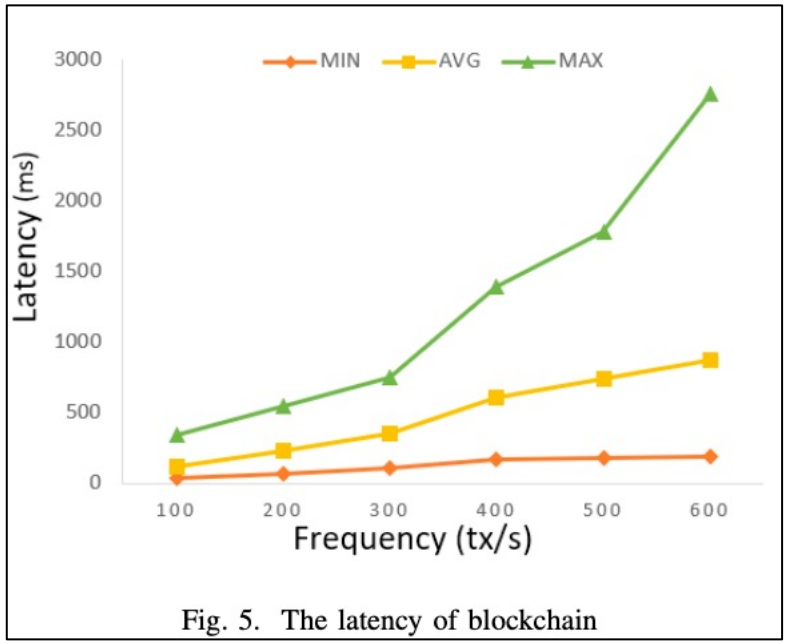


Fig. 5. The latency of blockchain

Truxen: A Trusted Compution Enhanced Blockchain

- Truxen
  - 신뢰 컴퓨팅 기반 프로토콜인 Proof-of-Integrity(PoI)

    → 무결성 제공
  - 블록체인의 효율성 문제를 해결하기 위한 "Single Execution Model"
  - **오프체인 호출 및 비결정적 컴퓨팅을 허용**

    → Enterprise Application 도입에 중요한 역할

    → 블록체인의 효율성 문제 해결

Truxen: A Trusted Compution Enhanced Blockchain

- **Blockchain Enhancement**
  - Mining Process
  - Proof-of-Integrity(PoI) Protocol
    - Miner Join Method
    - Miner Electing Method
    - Mining

- **Blockchain Efficiency Enhancement**
  - Transaction
  - Smart Contract
  - Single Execution Model

Truxen: A Trusted Compution Enhanced Blockchain

- Blockchain Enhancement
  - **Mining Process**
  - Proof-of-Integrity(PoI) Protocol
    - Miner Join Method
    - Miner Electing Method
    - Mining

- Blockchain Efficiency Enhancement
  - Transaction
  - Smart Contract
  - Single Execution Model

Truxen: A Trusted Compution Enhanced Blockchain

- Mining Process
  - 기존의 Proof-of-Work는 에너지 소모적
  - 신뢰 컴퓨팅을 통해 해결
    1) 설계대로 일정하게 동작
       → 변조가 일어날 경우 remote attestation에 실패
       → 해당 채굴자의 블록은 무시되고 제거됨
    2) 무작위 또는 라운드 로빈 방식을 통한 채굴자 선정 방식 사용
       → 고유 ID를 통해 Sybil 공격에 효율적으로 대응
    3) 블록과 Remote attestation이 함께 전달하여 트랜잭션을 실행하지 않음
       → 블록체인에 새로운 클라이언트가 들어올 때 실행되는 연산 감소
       → 각 노드의 성능 향상
    4) 거래 내역을 수정할 수 없으므로 51% 공격 내성

Truxen: A Trusted Compution Enhanced Blockchain

- Blockchain Enhancement
  - Mining Process
  - Proof-of-Integrity(PoI) Protocol
    - Miner Join Method
    - Miner Electing Method
    - Mining

- Blockchain Efficiency Enhancement
  - Transaction
  - Smart Contract
  - Single Execution Model

Truxen: A Trusted Compution Enhanced Blockchain

- PoI – Miner Join Method

  1) 참가를 원하는 채굴자가 네트워크에 참여 요청

     → Integrity Report 전송

  2) 기존의 채굴자들은 요청을 블록에 추가

  3) 올바른 Integrity Report인지 검증

     → 올바른 발급자에 의한 것인지

     → 올바른 서명인지

  4) 채굴자 리스트에 추가

Truxen: A Trusted Compution Enhanced Blockchain

- Blockchain Enhancement
  - Mining Process
  - Proof-of-Integrity(PoI) Protocol
    - Miner Join Method
    - Miner Electing Method
    - Mining

- Blockchain Efficiency Enhancement
  - Transaction
  - Smart Contract
  - Single Execution Model

Truxen: A Trusted Compution Enhanced Blockchain

- PoI – Miner Electing Method
  - **무작위**
    1) 채굴자의 해시값을 오름차순으로 정렬
    2) Verifiable Random Functions(VRF) 를 통해 블록의 해시값에 대한 hash와 proof 생성
    3) $hash\ (mod\ n)$번 째의 채굴자 선택
       - n은 전체 채굴자의 수
  - **라운드 로빈**
    1) 채굴자의 해시값을 오름차순으로 정렬
    2) $blockHeight\ (mod\ n)$번 째의 채굴자 선택

Truxen: A Trusted Compution Enhanced Blockchain

- Blockchain Enhancement
  - Mining Process
  - Proof-of-Integrity(PoI) Protocol
    - Miner Join Method
    - Miner Electing Method
    - Mining

- Blockchain Efficiency Enhancement
  - Transaction
  - Smart Contract
  - Single Execution Model

Truxen: A Trusted Compution Enhanced Blockchain

- PoI – Mining

1) 선출된 채굴자는 Transactions, Merkle root, Block header를 포함하여 **후보 블록 생성**

2) TPM을 통해 블록에 대한 **integrity value와 attestation quote를 생성하여 변조 방지**

3) 후보 블록에 integrity value와 attestation quote를 추가하여 최종 블록으로 구성한 후 브로드캐스팅

4) 일반 노드들은 해당 블록의 **integrity value와 올바른 Miner Electing Method를 따라 선택된 채굴자인지 확인하여 검증**

Truxen: A Trusted Compution Enhanced Blockchain

- Blockchain Enhancement
  - Mining Process
  - Proof-of-Integrity(PoI) Protocol
    - Miner Join Method
    - Miner Electing Method
    - Mining

- Blockchain Efficiency Enhancement
  - **Transaction**
  - **Smart Contract**
  - Single Execution Model

Truxen: A Trusted Compution Enhanced Blockchain

- Transaction
  - 신뢰 컴퓨팅을 통해 채굴자에 대한 검증이 끝났으므로 트랜잭션 결과에 대해 서명, 잔액에 대한 **추가적인 검증을 하지 않음**
  - 트랜잭션에 대한 시간 복잡도가 **O(n) 에서 O(1)로 감소**
- Smart Contract
  - Transaction과 유사
  - 채굴자가 생성한 블록을 브로드캐스팅하는 동안 계약 실행
  - 일반 노드가 블록을 수신한 후 무결성을 검증하여 블록체인에 저장

Truxen: A Trusted Compution Enhanced Blockchain

- Blockchain Enhancement
  - Mining Process
  - Proof-of-Integrity(PoI) Protocol
    - Miner Join Method
    - Miner Electing Method
    - Mining

- Blockchain Efficiency Enhancement
  - Transaction
  - Smart Contract
  - **Single Execution Model**

Truxen: A Trusted Compution Enhanced Blockchain

- Single Execution Model (SEM)
  - **블록체인 내에서 트랜잭션과 스마트 컨트랙트를 실행하는 단 하나의 노드**
  - 엔트로피 소스, 시드가 변경되지 않으므로 **난수 사용 가능**

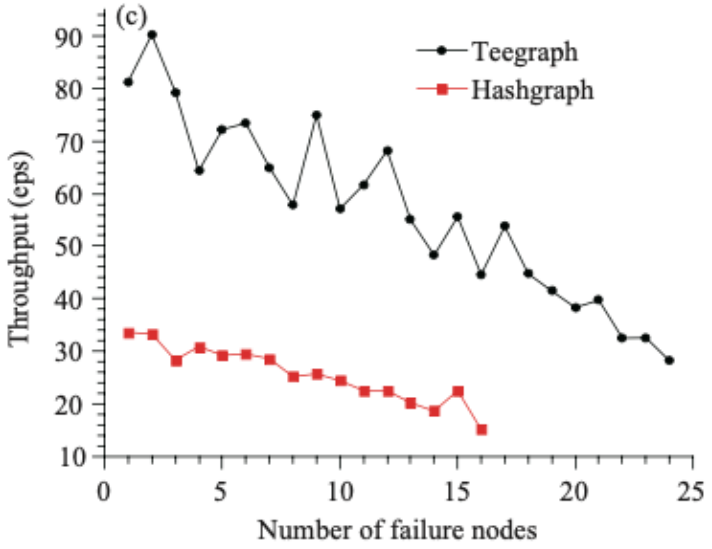
    → 서명, 키의 시드값, ID 등으로 사용 가능
  - 채굴자들의 **컴퓨팅 자원 절약**

Teegraph

- **고효율 합의 알고리즘**을 위한 Directed Acyclic Graph(DA) 기반 데이터 구조를 생성하기 위한 **가십 프로토콜 기반** 메시지 통신 매커니즘
- "Single-use of self-parent" 매커니즘을 통한 **단일노드 포크 공격 방지**
- 기존의 프로토콜은 전체의 2/3 이상의 투표를 요구

  → 1/2만 요구

  → ex) 50개의 노드



- 동적 매커니즘을 통한 합의 주체 변경

  → 다수의 하위 그룹이 생길 경우, 그룹 내의 노드에 대한 리스트를 포함하는 특별 이벤트를 생성하여 하위 그룹을 분리할 수 있음

Teegraph

- Teegraph는 가십 프로토콜을 사용
- 동시에 두 가지의 이벤트를 생성하여 Fork Attack 수행 가능
- **TEE를 통해 Fork Attack 방지**
  - 각 이벤트가 단 한 번만 자체적으로 부모가 될 수 있는 "Single-use of Self-parent" 사용
    1) 이벤트를 TEE에게 전송
    2) TEE는 이벤트의 부모해시를 메모리에 저장된 n-1번 째의 해시와 비교
    3) 두 해시가 동일하면 이벤트에 서명한 후 노드에게 돌려보냄

       → 동일하지 않을 경우, 프로세스 종료
    4) TEE는 이벤트의 해시를 메모리에 저장하여 **n-1번째의 이벤트를 대체**

       → **동일한 자체 부모로 두 개의 다른 이벤트를 생성할 수 없음**

       → **포크 공격 발생 불가**

Q & A