

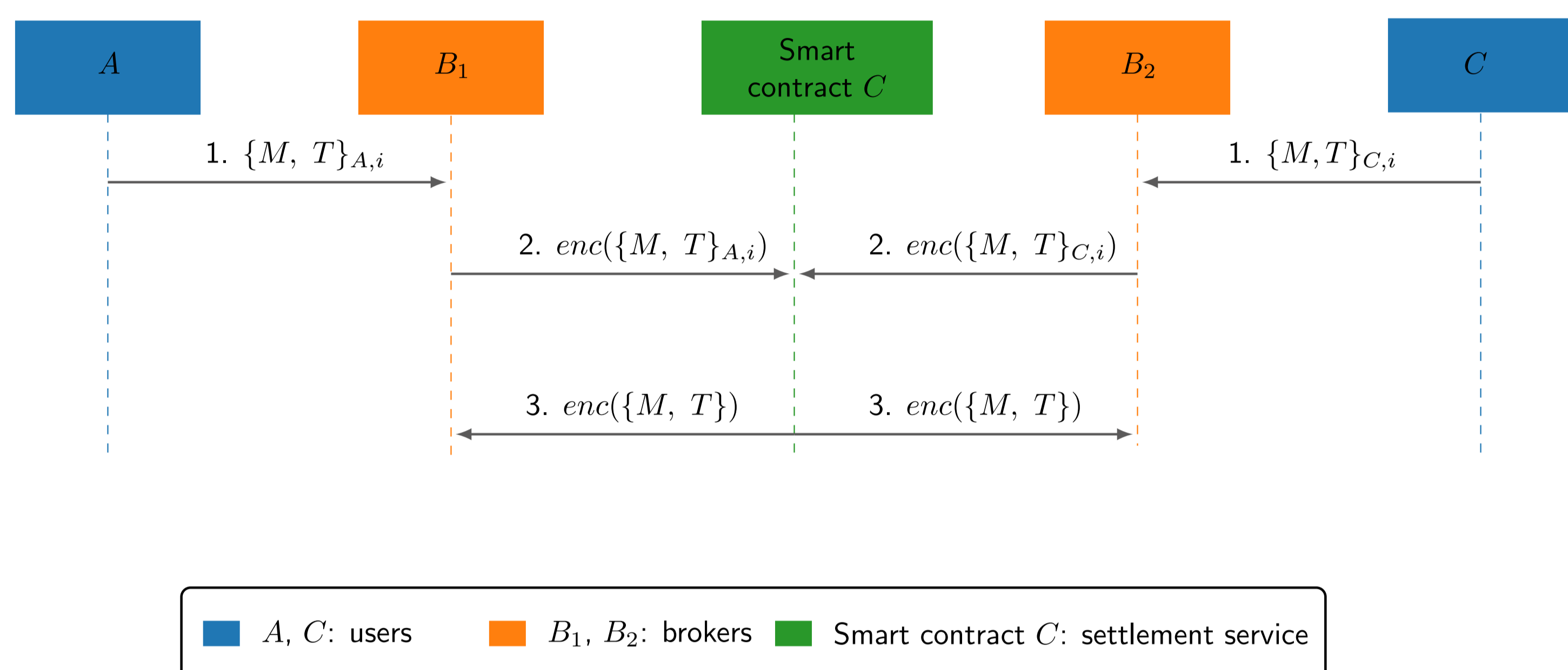
Abstract

- **FHE**: Enables computations on encrypted data without decryption
- **Smart contracts**: Ensure transparency and immutability in transactions
- **FHE + Smart contracts**: Enable secure and privacy-preserving decentralized applications

Applications

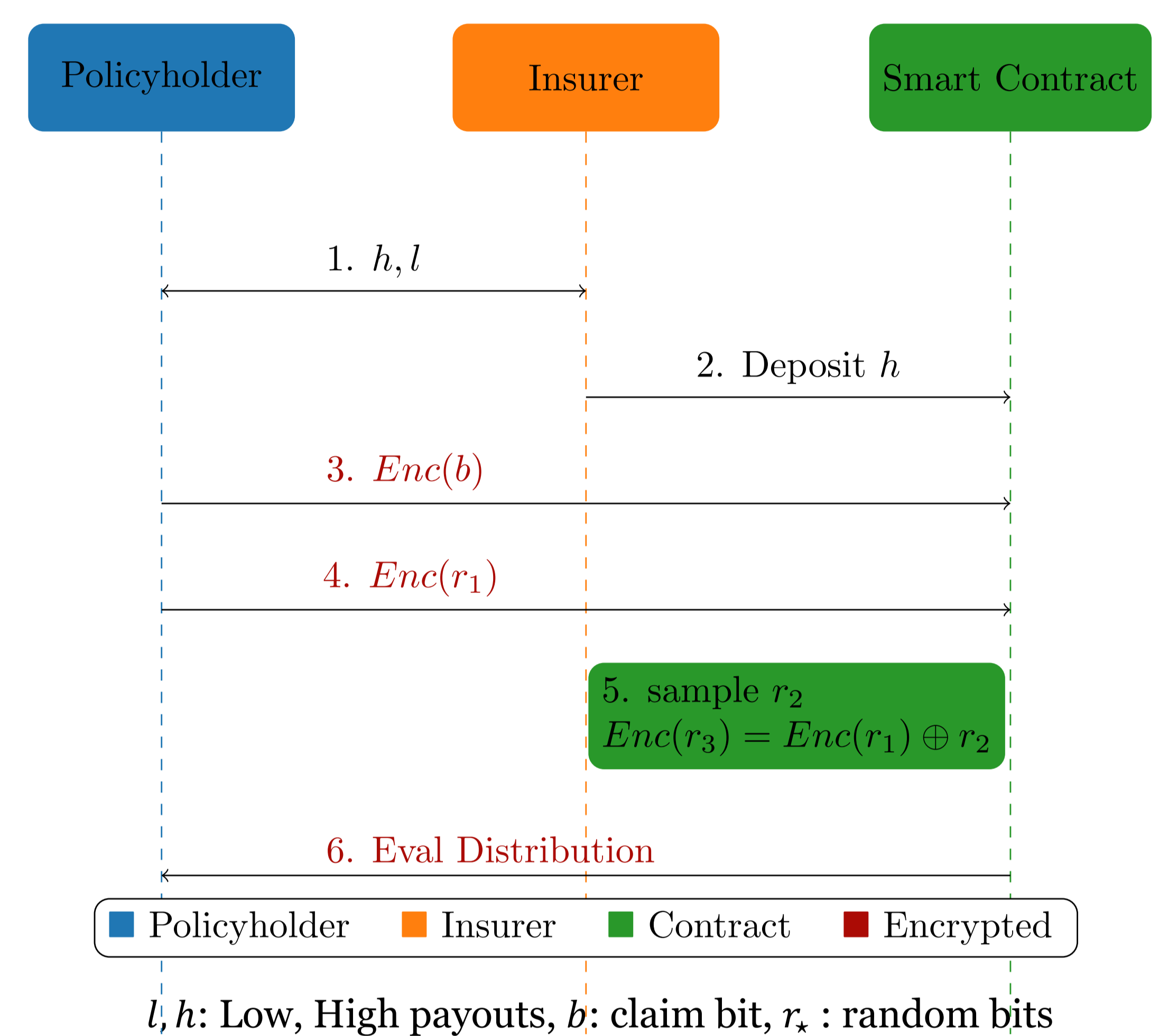
- Privacy-preserving token operations (ERC 1543)
- Economic applications
 1. Bilateral Insurance
 2. Intermediated Markets Coordination
 3. Repo Markets with Private Balance Sheets

Intermediated Markets Coordination



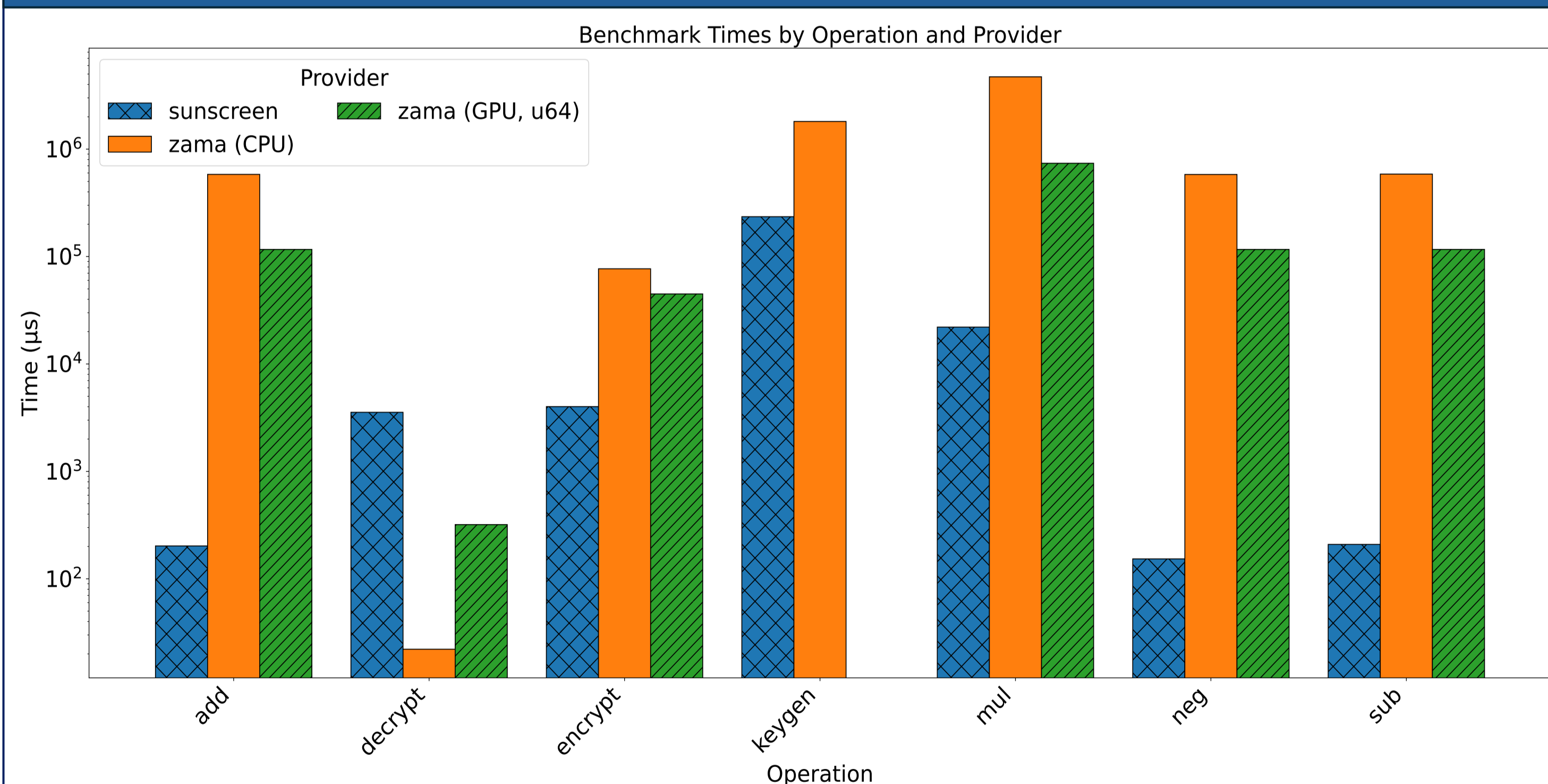
- Use-case:
 - Markets want to trade **without revealing** bids
 - Assets transferred **only on matching** bids
 - Privacy and Public Verifiability of FHE and smart contracts enable a trustless implementation
- Technique
 - Users (A, B) send bids to brokers (B₁, B₂)
 - Brokers sort bids and submit to smart contract
 - Smart contracts matches bids
 - Brokers transfer assets of the bids

Bilateral Insurance



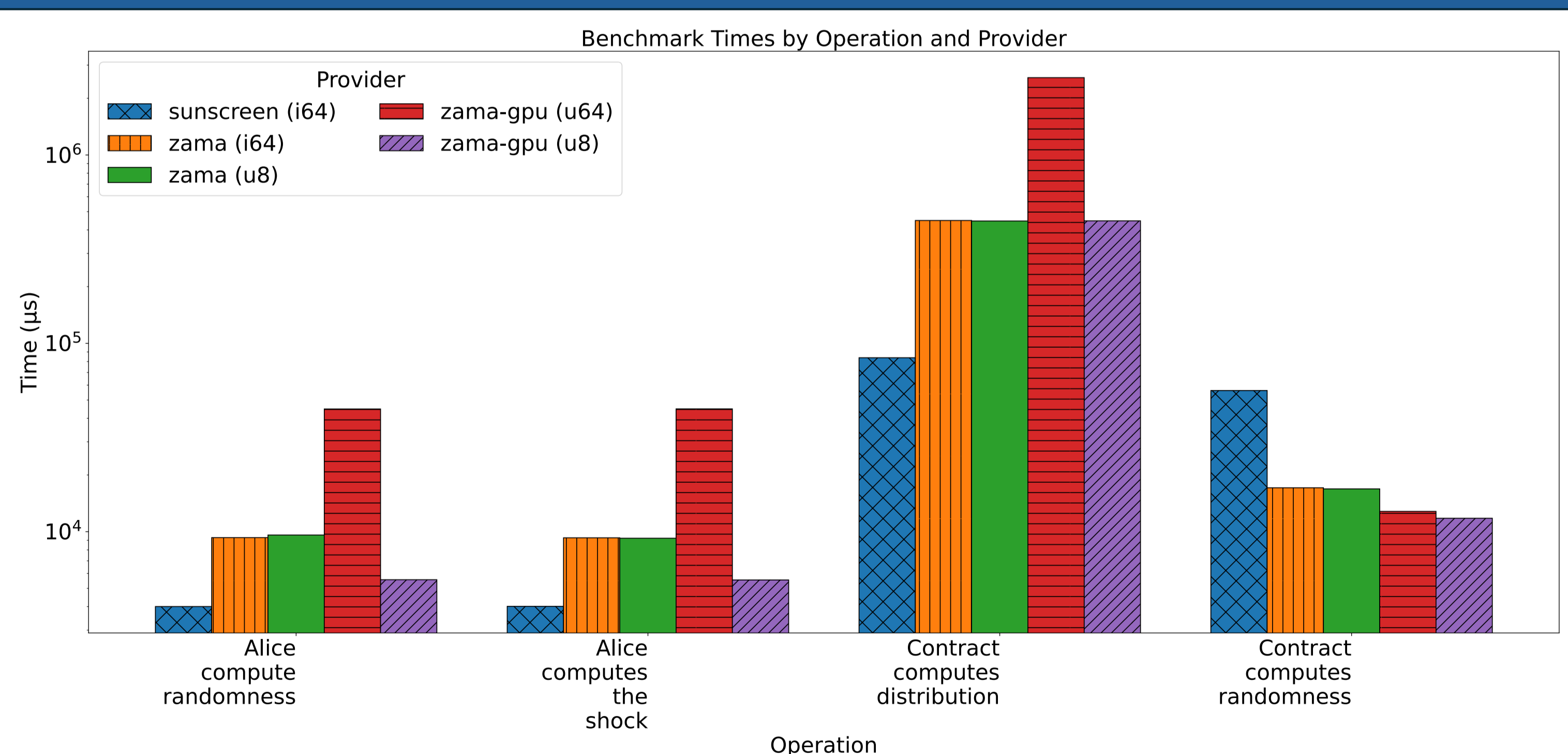
- Use-case:
 - The insurance mechanism incentivizes policyholders to report unverifiable claims truthfully
 - Privacy and Public Verifiability of FHE and smart contracts enable a trustless implementation
- Technique: Randomly payout (h) to non-claims
- Distribution = $l + (h - l)b \cdot r_3$

Operations Benchmarks



- Focus Points:
 - GPU vs. CPU: GPU is faster for 64-bit unsigned integers
 - Integer-based FHE vs. Boolean-based FHE: Integer-based is faster for small circuits

Bilateral Insurance Benchmarks



- Focus Points:
 - GPU vs. CPU: GPU is faster for larger-width integers
 - Integer-based FHE vs. Boolean-based FHE: Integer-based gets worse as the circuit grows larger

Research Gaps for FHE in Smart Contracts

- Validator scalability (beyond $n = 1, t = 0$)
- Computational scalability of FHE
- Hardware acceleration for FHE
- Key management for FHE
- Ciphertext and key sizes of FHE
- Verifiable inputs to contracts
- Rational security of FHE protocols
- Incentive designs for FHE operations