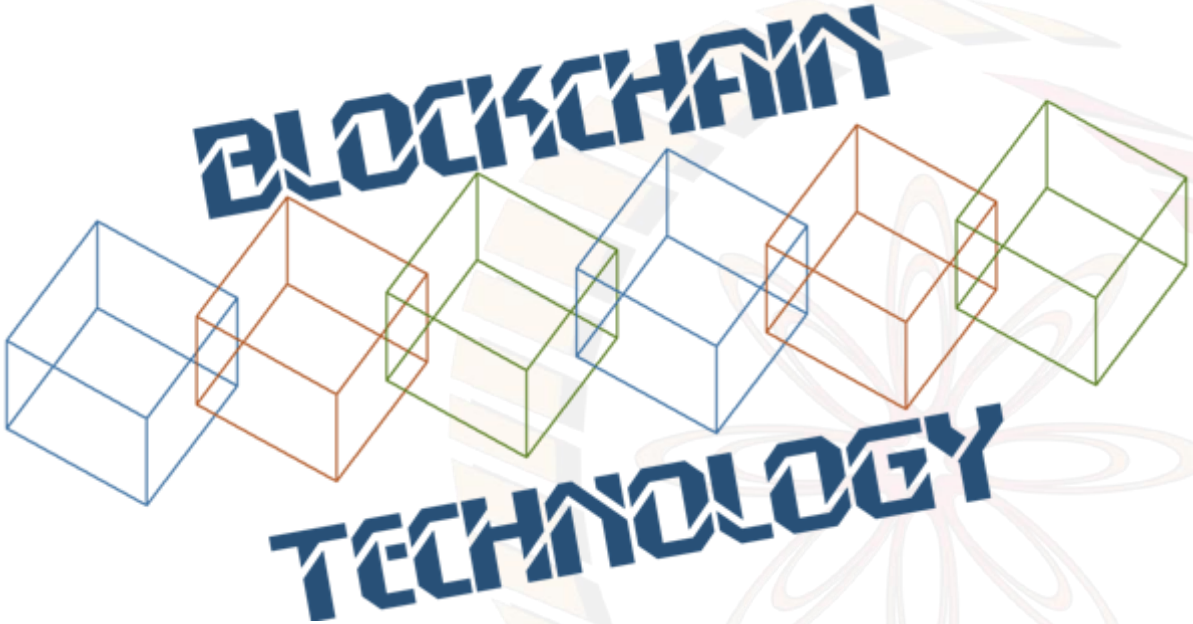
BLOCKCHAINS

ARCHITECTURE, DESIGN AND USE CASES

SANDIP CHAKRABORTY
COMPUTER SCIENCE AND ENGINEERING,
IIT KHARAGPUR

PRAVEEN JAYACHANDRAN
IBM RESEARCH,
INDIA

*Image courtesy: http://beetfusion.com/



SECURITY FEATURES OVERVIEW

Open Network: Security Properties

Identity	Transactions	Transaction Validation	Transaction Ordering
(what defines system participants)	(network messages)	("correctness" of network messages)	(protocols to order transactions)

Security:
- Correct transaction validation
- Ledger immutability

Privacy:
- Pseudonymity, in some cases anonymity

Assumptions:
- > 50% computing power complies with protocol
- User wallet is safely maintained
- All contracts are deterministic (Bitcoin and Ethereum achieve this by restricting set of permissible operations)

"Attack the assumptions" & .. human error

Compromise user-wallets by
- Attacking online wallet services

Compromise ledger immutability by
- Motivating alternate behavior
- Attacks shown at even ~25% compute power

Exploiting smart contract vulnerabilities to arbitrarily change ownership of coins

Hard forks are sometimes inevitable…

Blockchain for the Enterprise World

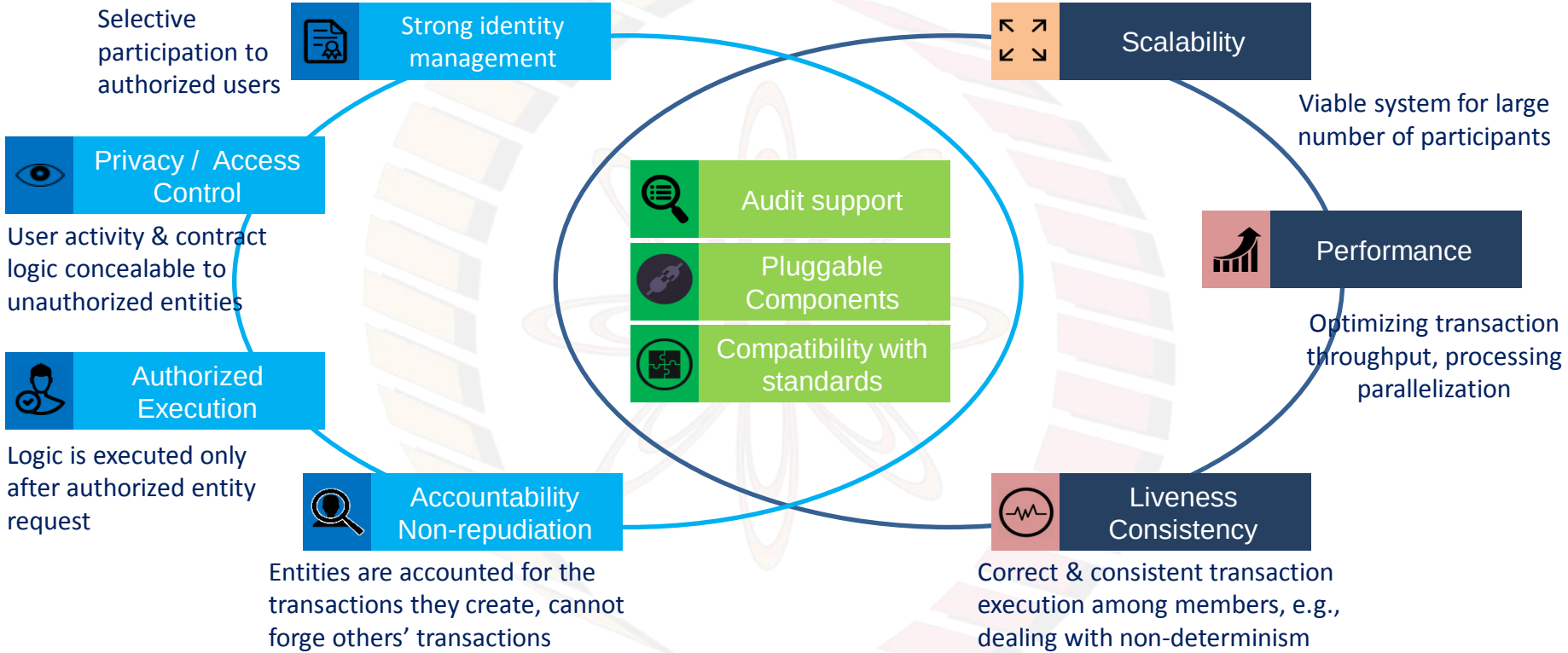
Blockchain = Shared replicated ledger + Consen-sus + Business logic + Crypto-graphic protocols

- Cost-effective
- Discoverability
- Trusted record-keeping
- Automation of trusted processes

- Strong identity management
- Accountability/ non-repudiation
- Auditability
- Privacy
- Performance

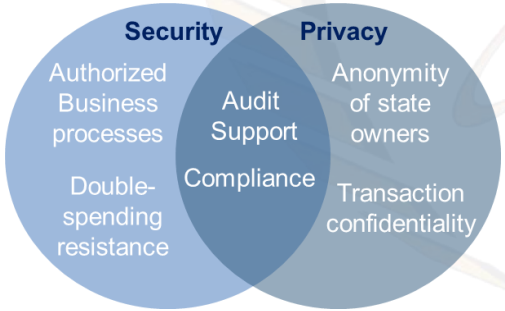
Permissioned blockchain platforms & HYPERLEDGER PROJECT

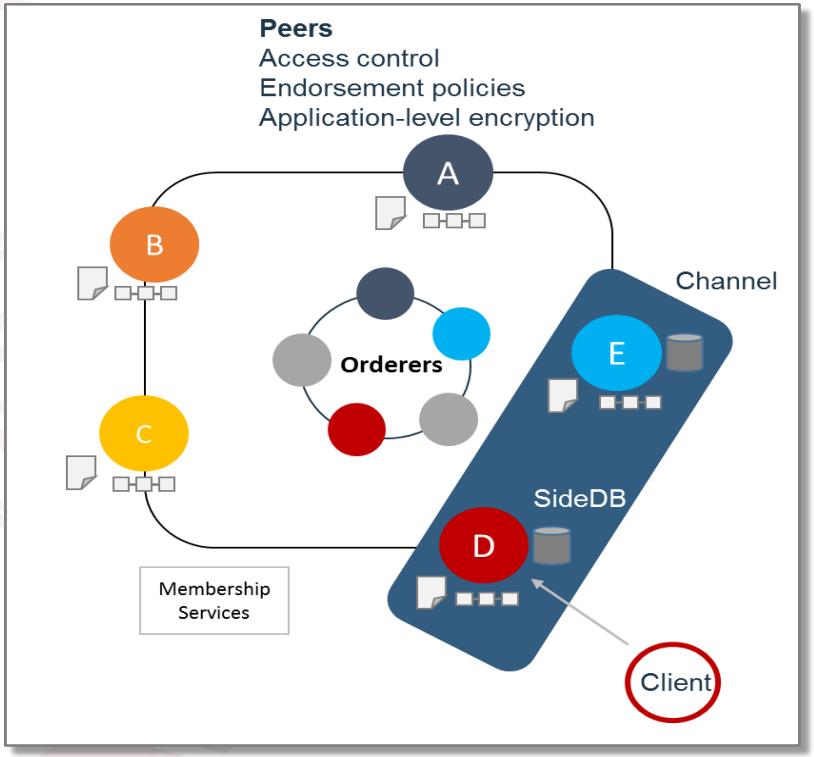
Enterprise Blockchain Applications: Security Considerations



Security and Privacy: Key Differentiation of Fabric

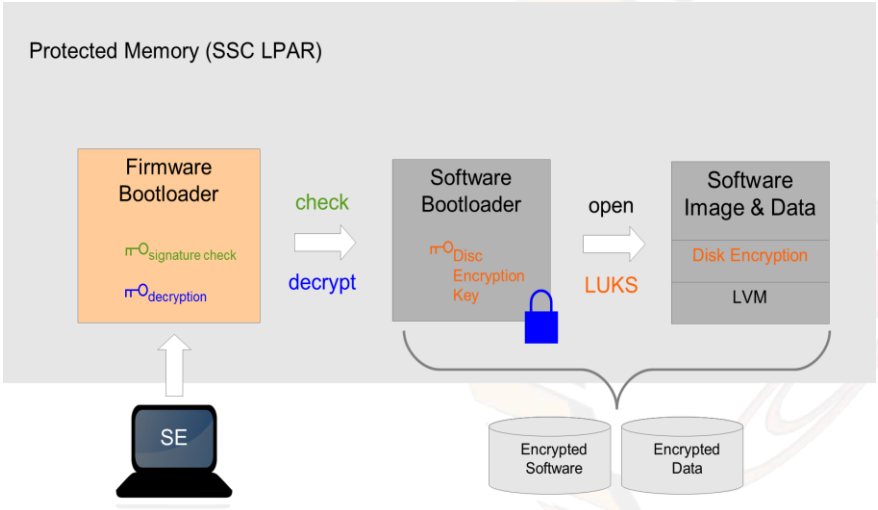
- Existing security/privacy controls
  - Membership and access control
  - Endorsement policies
  - Application-level encryption
  - Channels
  - SideDB
  - Trusted chaincode execution (secure containers)
- New security/privacy controls
  - Anonymous and unlinkable transactions (Identity Mixer)





Security in Cloud / Hardware

- IBM Blockchain Platform: All components run inside Secure Services Container



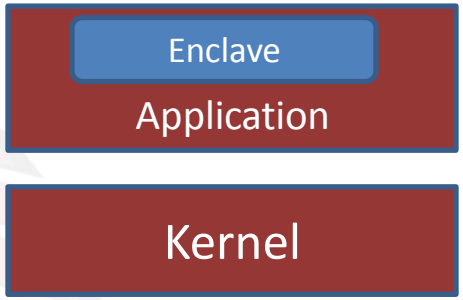
Complete isolation and encryption of code and data

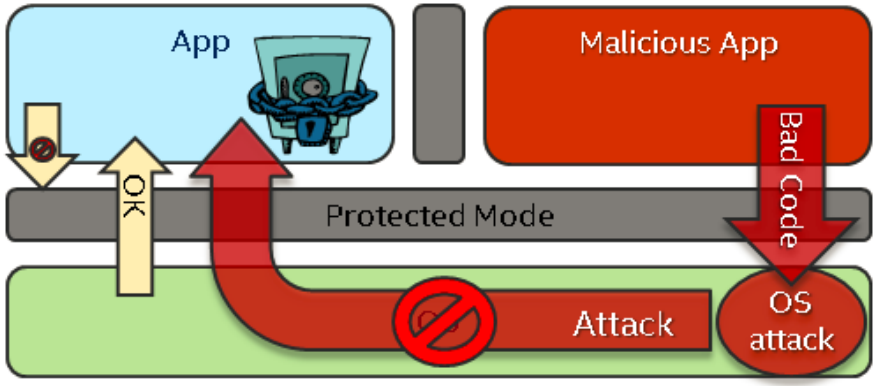
Boot sequence

1. Firmware bootloader is loaded in memory
2. Firmware loads the software bootloader from disk
   - Check integrity of software bootloader
   - Decrypt software bootloader
3. Software bootloader activate encrypted disks
   - Key stored in software bootloader (encrypted)
   - Encryption/decryption done in flight when accessing appliance code and data
4. Appliance designed to be managed by remote APIs only
   - REST APIs to configure Linux and apps
   - No ssh (allowed in dev mode)

Intel Software Guard Extensions (SGX)

- Trusted computing base
  - SGX Hardware (silicon chip + CPU microcode)
  - Code running inside the enclave
- Isolation of user-level code, protected from processes running at higher privilege levels
- Remote attestation of the enclave
- Reverse sandbox for applications

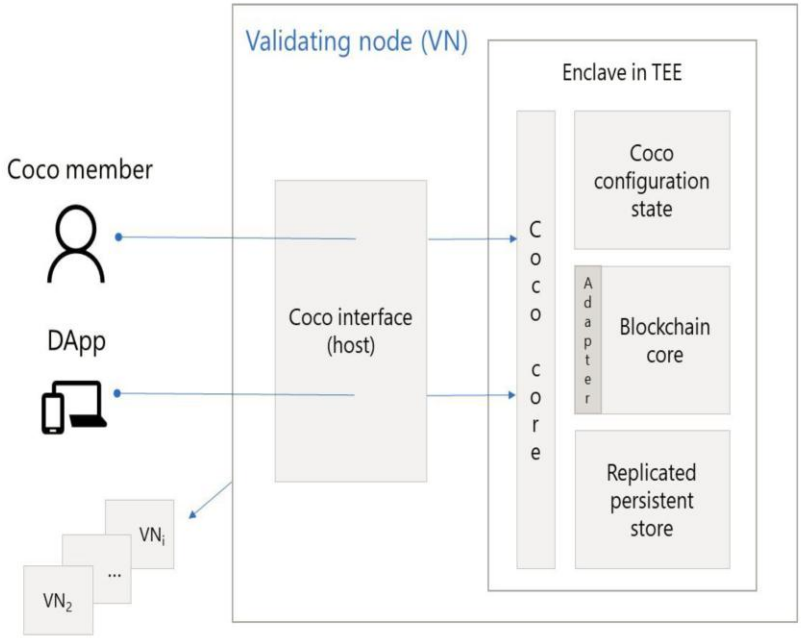




SGX Protection Model

Coco Framework

- A framework that allows creation of blockchain networks based on different blockchain protocols that supports
  - Multiple consensus algorithms
  - Trusted execution environments (isolation and strong confidentiality)
  - Permissioned identity management
  - Network management through a voting policy



Fun Reading

- “Majority is not enough: Bitcoin mining is vulnerable”, I Eyal, EG Sirer, International Conference on Financial Cryptography, 2014. Available at: https://www.cs.cornell.edu/~ie53/publications/btcProcFC.pdf
- “A Survey on Security and Privacy Issues in Bitcoin”, M Conti, S Kumar, C Lal, S Ruj: https://arxiv.org/pdf/1706.00916
- Understanding the DAO Attack, Coindesk blog: https://www.coindesk.com/understanding-dao-hack-journalists/
- Intel SGX Details: https://software.intel.com/en-us/sgx/details
- Coco framework whitepaper: https://github.com/Azure/coco-framework/blob/master/docs/Coco Framework whitepaper.pdf


thank you!