Blockchain Security
A Different Perspective

Shahid Sharif

Secunoid Systems Inc.
http://www.secunoid.com

Carlos Dominguez

Alto Azul Consulting Inc.
https://blocksec.ca

Disclaimer & License

Disclaimer

The views and opinions expressed in this presentation are those of the authors. They do not purport to reflect the policies, views, opinions or positions of any other agency, entity, organization, employer or company.

Topics

- Intro
- Cyber Security Considerations
  - Design
  - Architecture
  - Application Security
  - Identity and Access Management
  - Privacy & Confidentiality
  - Data Governance
  - Security Operations
  - Compliance and Audit
- Conclusions

BIO's

Shahid Sharif

CCSK, CISA, CISSP, CRISC, CSSP, PCIP, PMP

IT Security professional with over 20 years of experience in various industries in different roles. Some of the highlights include:

- Application Security Assessments
- Security Architecture advisory pertaining to Mobile Devices, Networks, Applications, & Systems
- Extensive experience in managing and supporting audits related to PCI-DSS, SSAE16. CSAE3416, 5025, SOC2, and SOX
- In-depth knowledge and experience on implementing Governance, Risk, & Compliance and supporting frameworks like COBIT 5, NIST, ISO, etc
- Extensive experience in creating procedures, policies and standards
- In-depth knowledge of Business Continuity Management which includes BCP, DRP, and Crisis Management.
- Private/Public Blockchain technologies

LinkedIn: https://www.linkedin.com/in/shahidsharif/

Carlos Dominguez

CISSP, CISA, CCBP, CBPro

Security professional with over 20 years of IT experience across several industries including financial and technology firms, performing as a subject matter expert in Security Architecture, Information Risk and Security Governance. Currently researching Blockchain and Distributed Ledger Technologies, and their Cybersecurity impacts.

Highlights

• Information Security Threat and Risk Assessments.
• Information Risk Management.
• Security Governance
• Security Architecture, Enterprise Security Architecture
• Information Security Policies, Standards and Procedures
• Blockchain and Distributed Ledger Technologies

LinkedIn: https://www.linkedin.com/in/carlos-l-dominguez/

Intro - Audience

Q:Target Audience?

A:Anybody who designs and implements:

- New blockchain technology
- dApps
- Oracles
- Exchanges
- Enterprise Applications
- Public Facing Applications
- Mining Farms
- Blockchain Technology Integrations

Intro – The Basics

Types of blockchains:

- Public
- Permissioned
- Hybrid

Note: This presentation is focusing on Public Blockchains

Intro – The Basics

Inherent Security Attributes (**CIA-R**):

- **Confidentiality**: Assurance that information is disclosed only to authorized parties.
- **Integrity**: Assurance for the accuracy and completeness of data
- **Availability**: Assurance that the system will be available for use when required
- **Non-Repudiation**: Assurance on parties not being able to deny having participated in a transaction.

Intro – Role of Cyber

Despite its inherent security attributes, even properly designed and implemented blockchain technology is still susceptible to other factors that could compromise its security

Also, trust-less operation of a blockchain does not extend to components residing outside the consensus network. Those out-of-chain components don’t benefit from any of the inherent or emergent blockchain attributes and can be susceptible to **STRIDE** threats (**S**poofing, **T**ampering, **R**epudiation, **I**nformation Disclosure, **D**enial of service and **E**levation of privilege)

Caution: Designers and Implementers can be held accountable for assurances on the Cyber Security of the system, its design and operation

Design – Security Architecture

Security Architecture is the collection of design artifacts that describe how the security controls work and the system CIA-R attributes.

Considerations:

- System Architecture (DApp vs Traditional)
- Layers and Tiers
- Components (on-chain, off-chain)
- Type of blockchain (public/permissioned)
- Actors/Threats

Caution: Control design by Cryptoeconomics is a new field still being developed

Security Architectural View

Blockchain technology could be described by the type or architecture under consideration:

- Primary Architecture (Business): the business capabilities being fulfilled, as per the inclusion of specific business roles, business processes and business functions being addressed by the blockchain. It could also include the business cases and business requirements. This is the architecture that exposes the blockchain to high level legal or requirements

- Secondary Architecture (Application): the blockchain as a business application that support the primary architecture. Includes the role and function of the layers, the logical components involved, the interfaces and their static and dynamic behaviors. This architecture also includes the cryptographic techniques as logical components of the implementation. It clearly defines what is “on-chain” vs “off-chain”

- Tertiary Architecture (Infrastructure): The IT systems that support the blockchain as an application, and described in terms of infrastructure. Also includes the IT systems that provide operational support, including security operations, as well as the off-chain components that interact with the blockchain

Crypto Economics

- Cryptographic primitives and techniques have a unique attribute among all other security controls in the technical domain: is it cheaper for the defender to apply them than for an attacker to defeat the controls.
- This cost asymmetry is an inherent property of current algorithms, and is supported by formal proofs and mathematical analysis of the techniques leading to adoption as technical standards.
- Open financial instruments that rely on Blockchain Technology are making use of cryptographic techniques to drive economic incentives, implemented as design mechanisms in those instruments. This usage has been branded “Cryptoeconomics”.
- Cryptoeconomics can be defined as the application of game theory costs models to cryptographic techniques to determine economic incentives in decentralized systems that is assumed to have adversarial actors, with the objective of leverage cryptography cost asymmetry to provide a level of fault tolerance and resolve conflicts. This field is not exclusive to blockchains but is a built-in property for blockchains.
- Cryptoeconomics plays a role in Blockchain Technology Threat Modeling as the functions that inhabit the boundary between on-chain and off-chain could upset the design balance of the system.
- While security controls costs are usually consider during their design, the application of game theory for developing security controls in the form of incentives by levering cryptography is a novel approach.

Threat Modeling Considerations

Given the diversity of potential configurations and applications for blockchains it is not possible to build a generic threat model that applies to all circumstances. Threat modeling can still be applied, but with special consideration for the blockchain specific architecture as a distributed system.

When performing a threat model for a blockchain implementation the practitioner should pay special attention to:

- Functions that rely on data and transactions recorded in the blockchain (on-chain) and benefit from its inherent attributes
- Functions that are executed outside the blockchain (off-chain) and therefore do not inherit any of the blockchain inherent attributes
- Functions that cross the boundary between on-chain and off-chain processing. While the blockchain may be able to provide transaction trust among untrusted parties by means of a consensus mechanism, parties that are not participating in the consensus mechanism (off-chain) can not be said to be part of the transaction trust
- Layering effect due to encapsulation of protocols that could result on expected attributes from one layer not to be present in the layers above

Off-chain components vulnerabilities

The trust-less operation of a blockchain does not extend to components residing outside the consensus network. Those out-of-chain components don't benefit from any of the inherent or emergent blockchain attributes and can be susceptible to STRIDE threats (Spoofing, Tampering, Repudiation, Information disclosure, Denial of service and Elevation of privilege).

These are examples of functions that are executed off-chain:

- Oracles acting as an appointed trusted source providing real world data to a blockchain
- Applications that connect to the blockchain via APIs
- IoT devices feeding data to an Oracle or an application
- Inter-node messages that are not recorded in the ledger
- Third parties such as Wallets and exchanges acting on behalf of a user that does not interact directly with the blockchain via a node participating in the consensus

Application Security – Smart Contracts

Smart Contract Pitfalls

- Designing with Immutability in mind (kill switch, ownership)
- Versioning
- Coding Language
- Frameworks (external dependencies)
- Run-away processes protection

Caution: there is a lot code out there that is neither smart nor contracts

Smart Contract Security Tools

- Various organizations are now bringing out blockchain security tools to address this void. Here is the list:
- A platform should support conventional programming languages to ensure existing secure coding practices can be leveraged.
- OpenZeppelin has created an open framework of reusable and secure smart contracts in the Solidity language.
- Trailofbits has created following tools:
    - A repository that contains examples of common Ethereum smart contract vulnerabilities, including real code.
    - "Slither combines a set of proprietary static analyses on Solidity that detect common mistakes such as bugs in reentrancy, constructors, method access, and more. Run Slither as you develop, on every new checkin of code."
    - "Echidna applies next-generation smart fuzzing to EVM bytecode. Write Echidna tests for your code after you complete new features. It provides simple, high coverage unit tests that discover security bugs. Until your app has 80+% coverage with Echidna, don't consider it complete".
    - "Manticore uses symbolic execution to simulate complex multi-contract and multi-transaction attacks against EVM bytecode. Once your app is functional, write Manticore tests to discover hidden, unexpected, or dangerous states that it can enter. Manticore enumerates the execution states of your contract and verifies critical functionality".
    - They have also created some reversing tools just like Porosity below which converts bytecode to solidity code.
- NCC Group had compiled a DASP (Decentralized Application Security Project) Top10 list
- Porosity, a decompiler and smart contract auditing tool for Ethereum smart-contracts.

Application Security – Best Practices

Recommended Best Practices

- Code Management (Code supply chain security: Github, CVS)
- Code Sources: Open Source, Proprietary
- Security Testing: Environment Isolation, DevSecOps pipeline)
- Code assurances: Signing, fingerprint, distribution, packaging
- Tools: DAST, SAST, Smart Contract proofing, Test Driven Development
- Industry Standard Cryptographic primitives

Caution: Open Blockchain and Open Source are not the same thing.

Identity and Access Management

Consider who has access, how and why:

- End user
- Developers
- Administrative Access
- MFA/2FA
- Segregation of Duties/Roles in Conflict

Identity Sources:

- Centralized & Decentralized
- Identity Providers, Trust, and Consumption

Caution: Identity underpins Access. Bad Identity results on Bad Access

Privacy & Confidentiality Considerations 1/2

- Due to its inherent security attributes, blockchain technology would seem to be a good fit for data security solutions where high integrity and availability are required.
- A review of use cases, either in public or consortium domains, would reveal that Database technology is not challenged by blockchain technology when considering transaction performance, confidentiality and data retention requirements.
- Use of blockchain technology for data security, or to store or manage data, requires to pay attention to the data security requirements. Not doing so could lead to miss-applications of blockchain technology.

Privacy & Confidentiality Considerations 2/2

The first consideration would be regarding the classification of the data, specifically around the inclusion of data elements that require High Confidentiality. A subset of this would include Personal Identifiable Information (PII).

Public and Consortium Blockchains may not suitable for sharing data across parties, when the data needs to remain confidential to those parties.

Consortiums could rely on off-chain compensating controls to address Confidentiality issues, as part of Consortium Governance. Public blockchains currently lack the controls required to provide forward looking Confidentiality requirements, and any data stored in a public blockchain could be considered to be at risk and potentially exposed in the future.

Use of blockchain technology for storing PII is strongly not recommended, as it is not likely to comply with privacy legislation:

- Once the data is shared, it can not be unshared
- Data encryption, as a confidentiality control, is exposed to technology obsolesce
- Data that has been shared, and protected with current encryption standards, could be exposed in the future
- Current technologies don’t address data retention requirements
- Data stored in a blockchain is immutable and cant be updated as per new encryption standards

Immutability & Data Retention

- While it may seem that blockchain immutability is an all-around positive attribute, it does creates some challenges with data management. Data that has been stored in a blockchain, specially public ones, is not susceptible to data retention policies. Once recorded, the data would remain as is for the lifetime of the blockchain.
- The Immutability security attribute of blockchain technology creates additional challenges for Confidentiality, as it makes it virtually impossible to address legacy data protection controls: it can neither be deleted or re-encrypted.

Security Operations - Processes

Consider “who, where, when & how” for these processes

- Blockchain initiation Ceremony (genesis block)
- Incident Response (before it becomes a crisis)
- Crisis Management (when things go really bad)
- Third party validations and assessment (trust no one)
- Vulnerability Management (and Remediation)
- Capacity Management
- Disaster Recovery
- Physical Security

Security Operations – Key Management

The complexity of **Key Management** is usually underestimated up front.

- Type of Wallet supported (HD Wallets recommended)
- Type of Key (do not invent, follow)
- Safe key storage (cold storage, physical)
- Key issuance
- Key transmitting
- Key Revocation
- Use of MultiSig

Caution: Keys have limited lifetime. Frequent usage means exposure.

Security Operations – Monitoring

Consider the W's (who, where, how) for these ongoing processes:

- Security Logs (Where are those and for how long)
- DoS Attacks
- Health of nodes (and Mempools monitoring)
- Transaction processing (detect availability issues)
- Master Nodes Changes
- Measure Centralization (network, miners, …)

Caution: Often the most invisible of components has the biggest effect on security: mind network concentration.

Compliance and Audit

Any hints of ownership (financial or operational) over a blockchain system dealing with real assets usually ends with someone being on the receiving end of Compliance and Audit. Eventually.

The best way to deal with Audit and Compliance is pro-actively self-audit and having an independent party provide audit reports.

Caution: mind the jurisdiction of the owners, operators, management and clients. Blockchain projects have a knack for crossing jurisdictions.

Conclusion

- Requirements, Requirements, Requirements
- Avoid assumptions
- Document the design for:
    - Infrastructure
    - Data structures
    - dApp
    - Frontend App

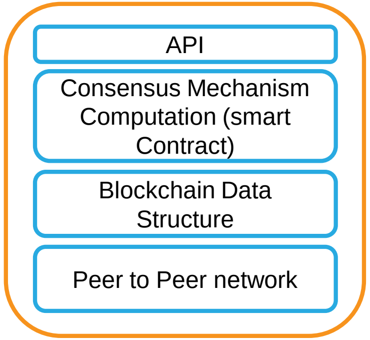
Hint: Consult with qualified Security SMEs if you need help.

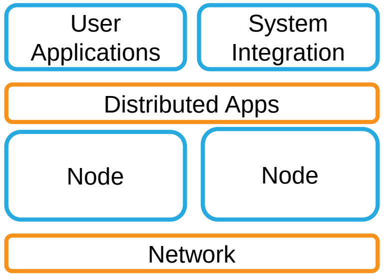
Questions

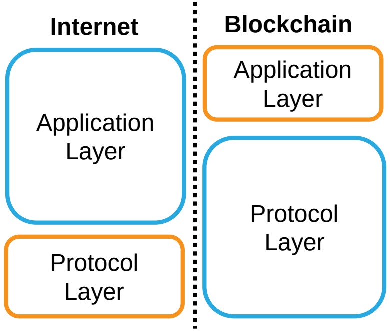
Thank You!

Backup Slides

Three views







Node

- By necessity, most technologies share a similar node structure
- The Peer to Peer network can use any protocols
- Key Management is done outside the node
- There could be specialized nodes, and lightweight nodes
- Enterprise Blockchain could have pluggable Consensus
- Current technologies don't limit the size of the Data Structure
- There are a number of components that reside outside the node

Platform (Web 3.0, Distributed Applications)

- Blockchain proposes an platform (internet of money)
- This stack includes a number of natively distributed components
- The model promotes decentralization and disintermediation
- It is based on Cryptoeconomic designs (consensus, incentive/penalty driven)
- Is driving Interoperability across the landscape
- Lacks Command and Control points (nobody is in charge)
- This notion is present in Enterprise Blockchain (Distributed Applications Marketplace)

Protocol

- Blockchain can be considered to be a "fat protocol"
- As a protocol, Blockchain moves "value" instead of "information"
- Unlike internet protocols where the value is in applications
- Blockchain protocols create value directly (participation and investment)
- The protocol value drives adoption and standardization
- The notion applies to public Blockchain, but could cross over