

IoT Security Institute

ENTERPRISE ARCHITECTURE SECURITY FRAMEWORK

FOR

BLOCKCHAIN APPS

Matthew Hargreaves, SCCISP ,
TOGAF9, Internet of Things Security
Institute.



Cyber
Security
Through
Education

Entire Network Security

BLOCKCHAIN APPS

- Indelible ledgers hold the value for each wallet
- Smart Contracts codify business transaction logic
- Unstoppable blockchains compute subsequent blocks regularly
- Human or machine clients provide their wallets to smart contracts for execution.

Blockchain Apps are a recent type of application built on *decentralised indelible ledgers*, first introduced by the Bitcoin cryptocurrency. Their key quality is the *trustworthiness* of the ledger. This quality allowed Bitcoin to claim its network represents a currency that could be used on the internet, which it now does.

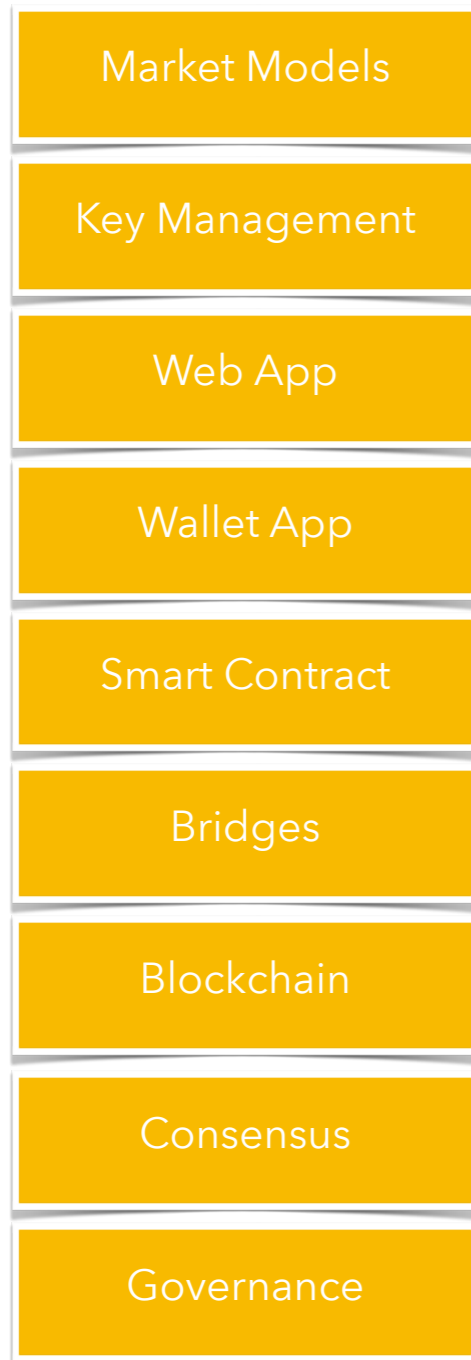
Bitcoin's ledger was implemented as a blockchain, but some newer cryptocurrencies use decentralised acyclic graphs (DAGs), so perhaps we should refer to 'Indelible Ledger' instead of 'Blockchain', but blockchain has become the terminology most closely associated with the technology. We use both interchangeably.

Ethereum introduced *computation* on the blockchain itself in the form of *smart contracts*. This led to blockchain applications (aka apps) where web applications interact with the smart contracts, which, every block, get computed alongside currency transfer transactions.

Transactions are sometimes initiated manually by people, but sometimes also *automatically* in the context of the *Internet of Things* (IoT).

The *entire network security* is assessed, from cryptocurrency wallet, web application, blockchain transaction and the blockchain itself.

Reference Technology Stack Architecture



Security vulnerabilities come from many places – source code weaknesses, systems and network fragility and the social consensus that secures a blockchain are just a few. This framework covers them all and provides guides on how to assess, plan for and remediate each of them.

Before going through each threat in detail, consider a selection of the concerns for this *technology solution stack*.

Element of Stack	Technology Area	Concerns	Questions
Key Management - IoT Admin	Keys	Are keys managed carefully?	How are the private keys secured – both during day-to-day operation and during commissioning?
Key Management - End User	Keys	Are keys managed carefully?	How have the private keys or pass phrase been archived for future reconstruction?
Wallet App (e.g. MetaMask)	Vendor Code	Coding vulnerability - are the keys safe?	Is there a zero-day vulnerability in the wallet app? Is the underlying operating system vulnerable to key-stroke loggers?
Smart Contract	Bespoke Code	Coding vulnerability – is the intended execution safe?	Can the coding of the smart contract be relied on? Can it be proved correct? Has it been peer-reviewed?
Blockchain	Infrastructure	Is the ledger reliably implemented?	To what extent may we trust the underlying ledger/blockchain's accuracy – i.e., the implementation of the blockchain?
Bridge	Infrastructure	Is the bridge implementation securely coded and maintained?	If the blockchain has implemented bridges to other blockchains, how secure and reliable are those bridges? What are the implications to the App, should the bridge be hacked?
Consensus Implementation	Network	Are miners/validators sufficiently decentralised?	What measures are in place to prevent a 51% attack? Is the distribution of the miners/validators assessed regularly?
Governance	Blockchain Governance	Does the underlying blockchain have a future?	Is the blockchain dedicated to the application or a shared platform? Who controls the future of the blockchain? What are the financial and social reasons for the blockchain to continue to prosper in the future?

THREAT LANDSCAPE

Blockchain Governance



Blockchains, starting with Bitcoin's, provide a *reliable, indelible ledger* platform *required* by the application. That

is, if the application does not *require* such a reliable, indelible ledger, it probably should be implemented instead with a relational database. To date, these ledgers have not been easily hacked or surreptitiously altered. Only new transactions may be added.

Governance ensures the future of a blockchain.

Tenancy Architecture Implications

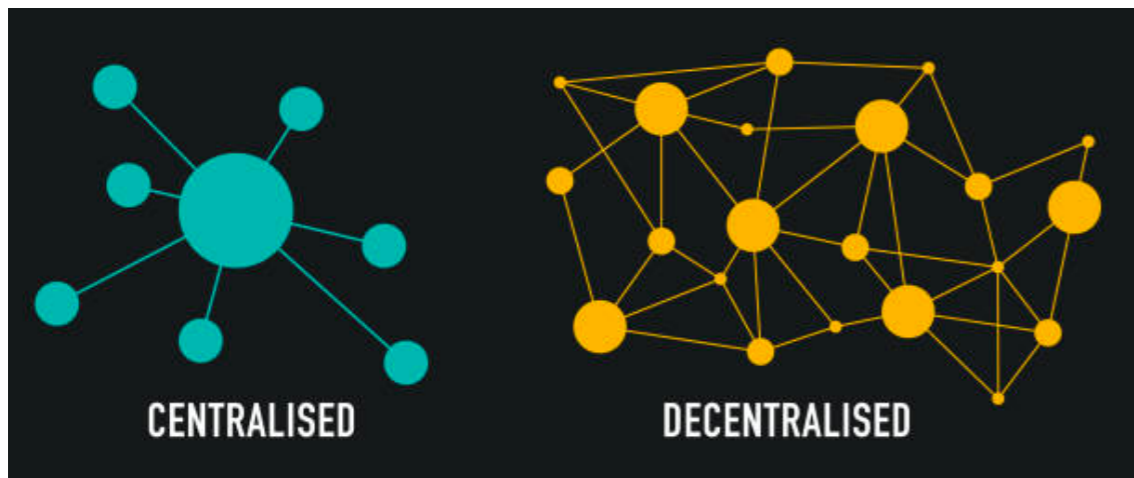
Your application may be a tenant of a blockchain such as Ethereum, where your project has created its ERC20 token type, minted them and coded smart contracts, in Selenium on Ethereum. Alternatively, your project may have implemented a *dedicated blockchain*. In either case, there is a reliance on the ongoing success of the blockchain. Will miners (proof-of-work) or validators (proof-of-stake) continue to be incentivised to continue creating new blocks? This risk needs to be assessed and contingency plans prepared.

With COSMOS ecosystem blockchains, governance is generally *on-chain*. Any wallet containing sufficient coinage may propose a change to the network configuration or deployment of funds; Every wallet participating in staking funds to secure the network then

may vote on the proposal. This process is *managed on-chain*, meaning the mechanism to achieve this has already been developed as a smart contract and web application, leading to a verifiable result for all, on-chain.

This security risk to the blockchain is more *commercial and social* than technical but needs assessing nonetheless.

Blockchain Network Decentralisation



Public blockchains, such as Bitcoin and Ethereum, *trust network decentralisation* to maintain their independence from interested parties that may otherwise attack the blockchain by perverting the production of blocks in several ways. The idea is that a *public, decentralised network* creates a situation where everyone keeps an eye

on everyone else to ensure the safety and trustworthiness of the network. If the network is compromised, those not involved in the compromise will suffer, so they are incentivised to make sure this does not occur.

Private blockchains dedicated to the application instead rely on *organisational hierarchal discipline* to ensure no shenanigans are at play in the production of blocks.

The risk of colluding network nodes

In decentralised networks, there is a risk that the degree of decentralisation will diminish when a group of miners or validators collude to attack the network. This is known as a 51% attack. This risk needs to be managed by regularly monitoring the then degree of decentralisation and putting in place measures to increase decentralisation, where it is at risk.

Bridging Vulnerabilities

"No man is an Island" - John Donne.

New blockchains are appearing all the time. Many flounder; some prosper. Over time the desire to integrate blockchains increases for many reasons.

Consider -

Decentralised Finance (Defi), where financial instruments are implemented as smart contracts on decentralised blockchains, the liquidity of more established blockchains is often sought on new chains.

Centralised Cryptocurrency Exchanges (CEXs), such as Binance and Coinbase, provide facilities to exchange coins and tokens similarly to fiat currency exchanges. They achieve this on their in-house dedicated platform where the organisation has full control of the funds they make available to their clients.

Decentralised Cryptocurrency Exchanges (DEXs) need to do this in an *automated, decentralised* manner, where smart contracts are used for the exchange of funds across blockchains. For this to be possible, a cross-blockchain bridge has to be implemented. Since blockchains’ unique value is their reliability and trustworthiness in the blockchain’s issuance and transfer of funds between wallets, extending this facility to outside entities such as foreign blockchains brings with it attendant risks. Many, including Ethereum’s Vitalic Buterin, have warned against the inherent risks of cross-blockchain bridges. By

September 2022, there have already been many exploits with severe consequences.

Cryptocurrency Exchanges Hacks

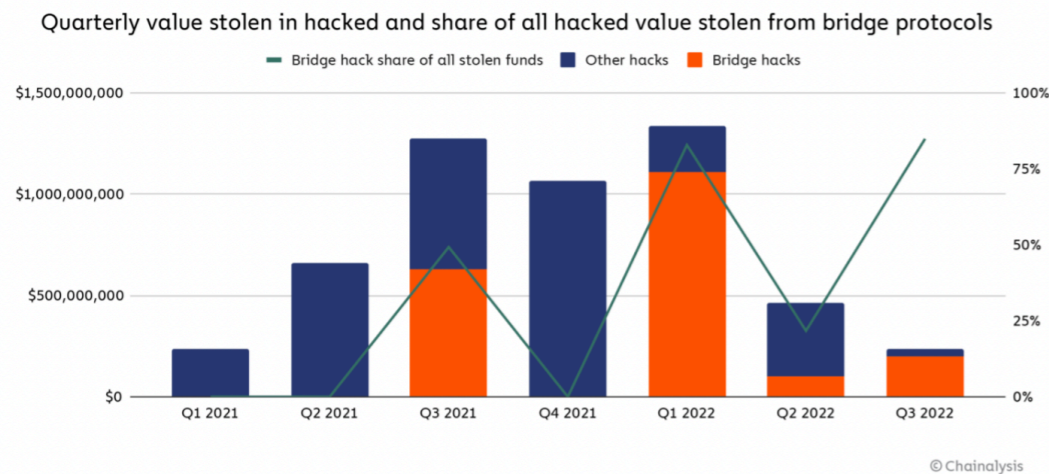
Here is a list from [HedgeWithCrypto.com](#).

List of Hacked Crypto Exchanges

A complete list of cryptocurrency exchanges and platforms that have been hacked or suffered major security breaches are listed below. If we have missed any, let us know by contacting us.

DATE	EXCHANGE	CAUSE OF HACK	AMOUNT STOLEN (USD)
2022, January 17	Crypto.com	Unknown	\$34 million
2021, December 11	AscendEX	Obtained access to hot wallet	\$80 million
2021, December 5	BitMart	Obtained access to hot wallet	\$150 million
2021, August 19	Liquid	Obtained access to hot wallet	\$97 million
2021, April 29	Hotbit	Obtained access to hot wallet	Nil
2020, December 23	Livecoin	Compromised system/servers	Unknown
2020, December 21	EXMO	Obtained access to hot wallet	\$4 million
2020, December 1	BTC Markets	Internal staff error/mistake	270,000 user's private details
2020, September 25	KuCoin	Data leak	\$275 million
2020, July 11	Cashaa	Malware	\$3.1 million
2020, June 29	Balancer	Vulnerability in protocol	\$500,000

Bridge Specific Hacks



Taken from Chainalysis' [report](#).

A cross-blockchain bridge should be considered an area of high risk and deserves a risk assessment of its own.

The technology used to achieve a bridge bears heavily on the associated risk profile. For instance, COSMOS Ecosystem has introduced Inter-Blockchain Communication (IBC) protocol, which is designed to address this very issue. This is a more reliable, better-designed approach than extant solutions that do not use IBC. Within the COSMOS, IBC has proven a reliable solution to cross-chain asset transfer, demonstrated by the Osmosis DEX and its ability in bridging assets to its OSMO chain.

Blockchain Implementation

Most blockchain implementations are modelled after Bitcoin, so let us consider Bitcoin.

Regardless of the great success of Bitcoin's blockchain in delivering a *reliable indelible ledger*, we should not neglect the possibility of *that coding itself being vulnerable*. All code may contain errors. Bitcoin's architectural solution and its cryptographic solutions are well considered and have stood the test of time with a financial bounty that has increased significantly every Bitcoin halving.

The Bitcoin blockchain itself continues to be regarded as 100% secure with no blockchain hacks. This is the basis of the technology it introduced but *all software* can contain *zero-day defects*.

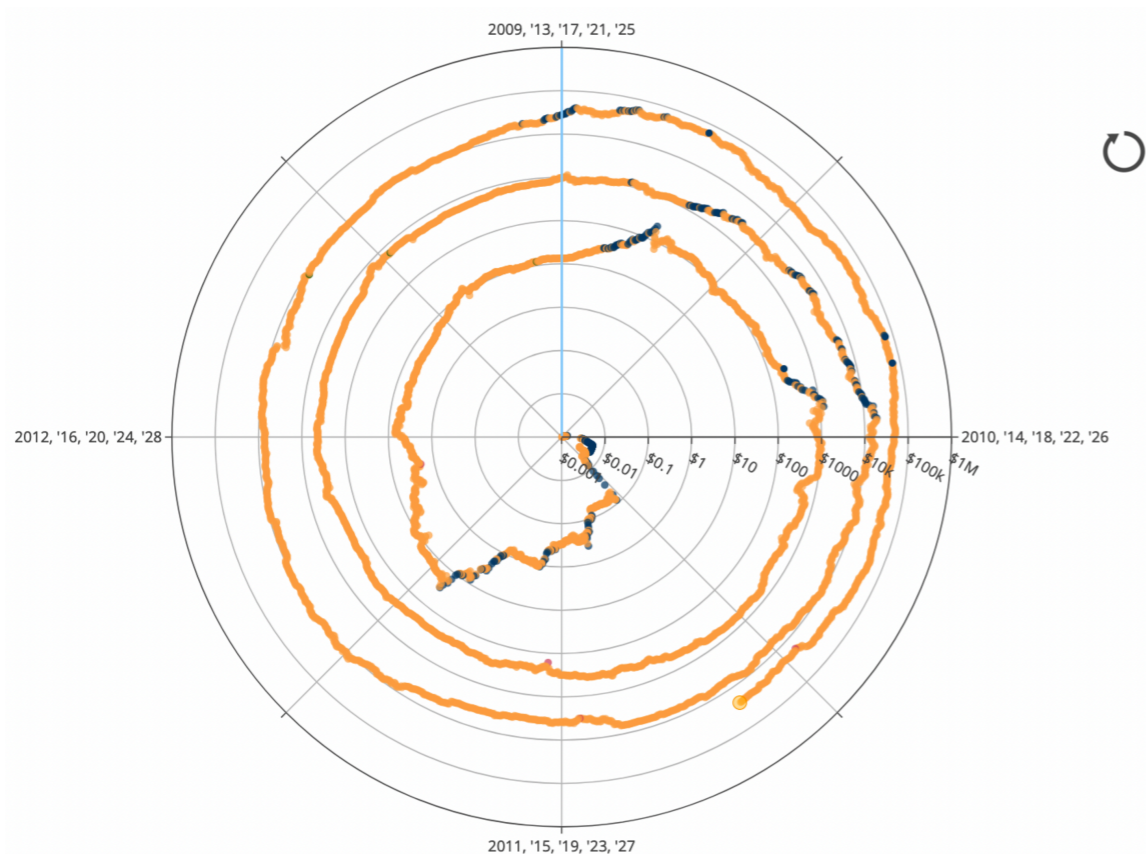


Chart - Bitcoin price on a logarithmic scale, where each rotation is 4 years, which is very nearly the Bitcoin halving cycle period. Notice how, *after 4 years, the price has always been higher.*

When Bitcoin was first deployed in January 2009, there would have been little reward for breaking the machine, other than kudos amongst the cypherpunks. Now there are billions, soon trillions, of dollars at stake. Say, for instance, if one finds a way to *derive the private key from a public wallet address*, one could help oneself to any

existing wallet. An interesting choice would be those associated with Satoshi Nakamoto which are valued at millions of dollars and have not been used for many years. This would surely attract a lot of attention, which may not be what one would want! Since this cryptographic mechanism is now so trusted, there would surely be significant activity before the Bitcoin core maintainers start working on solutions, if we accept that they believed in the hack. There are current concerns about the vulnerability of these cryptographic algorithms to quantum computers and I believe in time Bitcoin will migrate to quantum-resilient cryptographic primitives.

I remember an interview with Bitcoin core maintainers at a conference discussing the maintenance of Bitcoin core code. It was shared that, sometime earlier, a maintainer had noticed that there was no check in place for some aspect of introducing new transactions on the blockchain. This meant that transactions could be submitted that should fail to be introduced, but wouldn't be. This, of course, was fixed in the following release, but it was clear until that was the case, the blockchain itself was very vulnerable. This vulnerability seems so unlikely now but the point is that *zero-day vulnerabilities* tend to exist in all code.

Smart Contract



The application's core *transactional logic* is likely written as a *smart contract* in the blockchain's scripting language. This is what enables contractual actions to *run without intermediaries*. This is the key innovation of smart contracts. "Rules without Rulers" as someone said. Those who would interact with the smart contract inherently are trusting it. Trusting that it reliably does what it was designed to do but also trusting that it is not itself vulnerable to attack. So this is another point of vulnerability within the technology stack. Smart contract code could have zero-day vulnerabilities, so past performance may not be a guarantee of future security.

The Dao Dao Debacle



One of the first smart contract Centralised Autonomous Organisations (DAOs) was "The Dao (of DAOs)" on the Ethereum blockchain. Someone, it is assumed on reading the code, figured a way to game the contract into providing his/her wallet with all the funds held in The Dao's wallet. Ethereum was well embarrassed. What to do?

Much discussion and disagreement ensued. Some pushed for 'Code Is Law', so hard luck to those who invested; Others believed in restoring the funds via a change pushed to the miners. In the end, the Ethereum chain forked on the issue into **Ethereum** (ETH), where a *fix* was *put in place* to return the funds and **Ethereum Classic** (ETC), where no fix was put in place. There have been many hacks since then of smart contracts and bridges in particular, but pushing changes to restore funds is not in vogue. Either the funds stay stolen, or law enforcement chases down the culprits.



The DAO



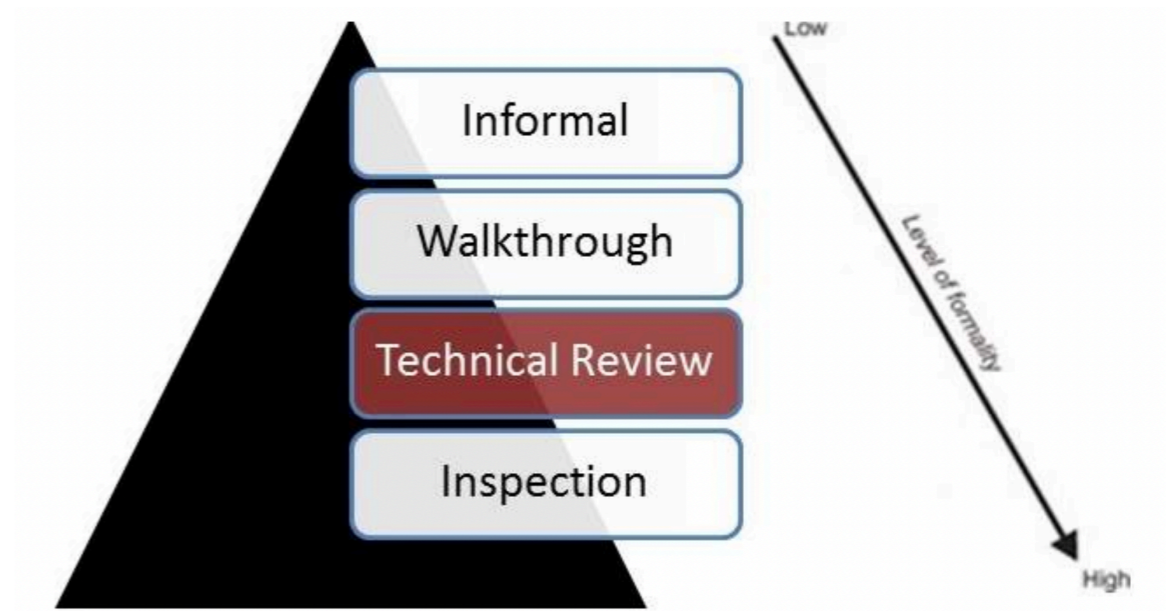
Heist: 3.6 million ETH

Value Then: ~\$50 million

Value Now: \$6.6 billion

not boring

Code Review



Proving code correct is non-trivial and automating code review is probably not even possible. Alan Turing proved in the Halting Problem that it is impossible to write a program that would detect if another program will halt or not (i.e. does not enter an infinite loop). We take this as an indication that the generic case of *program proving* is not possible. The best practice in software engineering is *peer code review* ("*code inspection*"). IBM are the 'OGs' here, though they are probably not who are thought of initially. A case in point is that IBM have shown *Code Inspections* are more *efficient* and *effective* than creating dynamic unit test code. As the programmable blockchain space matures, code review and audits are a developing business. Providing this service include IBM and Consensus. Writing

smart contract code is specialised and therefore so is reviewing same.

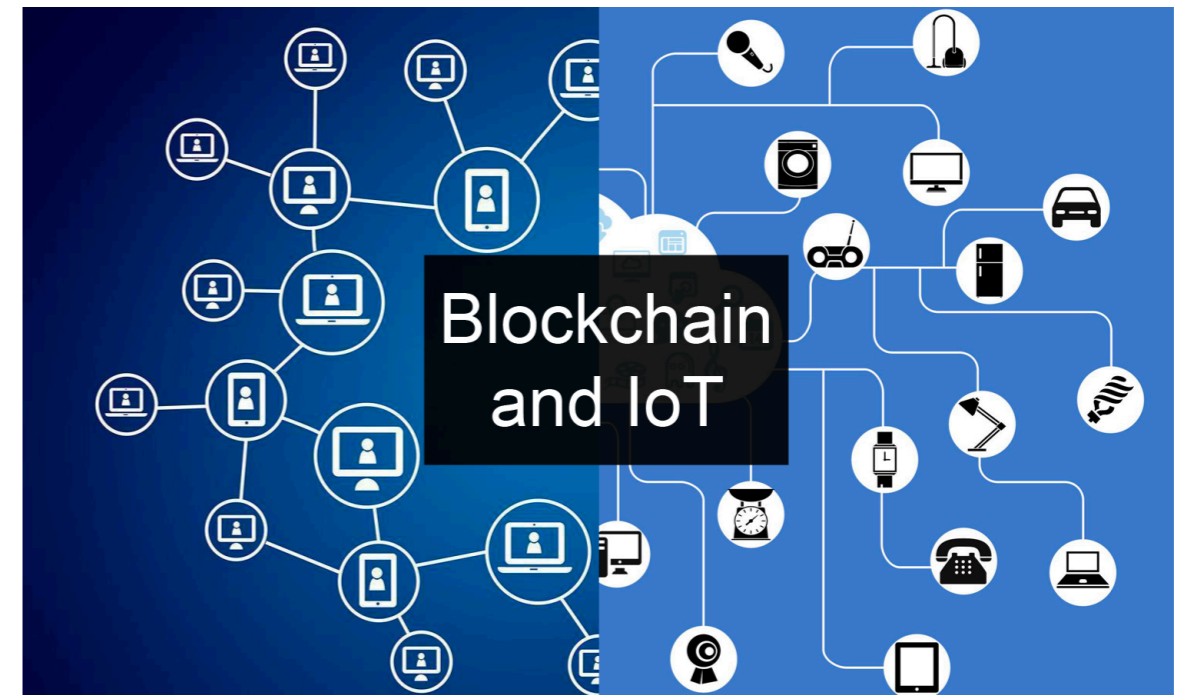
Smart Contract code, whether executed by devices in the IoT or by human interaction, need risk assessment and the engagement of a third party for code review should be considered.

Web Application

Particularly with human interaction, but sometimes where IoT devices are involved, access to the interaction with the smart contract is front-ended by a web application (web app). This is another vulnerability. The web app is being trusted not to act maliciously. The web server hosting the web app also may get compromised and a substitute, malicious web app planted. The web app and its hosting needs risk assessment.

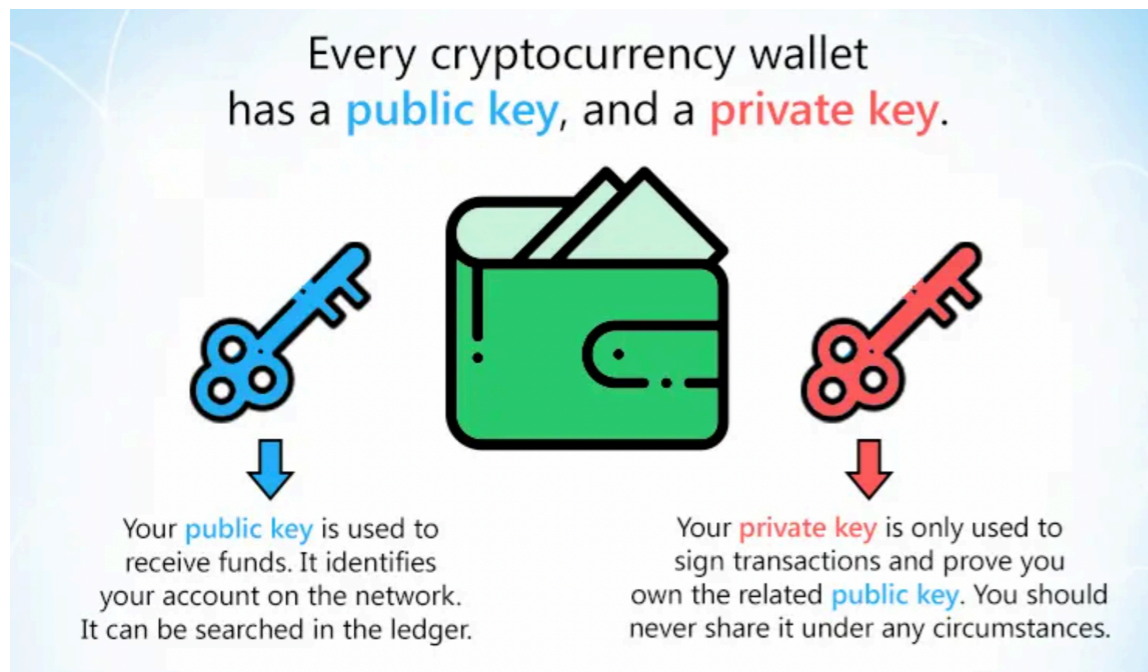
Wallet Application

IoT Wallets



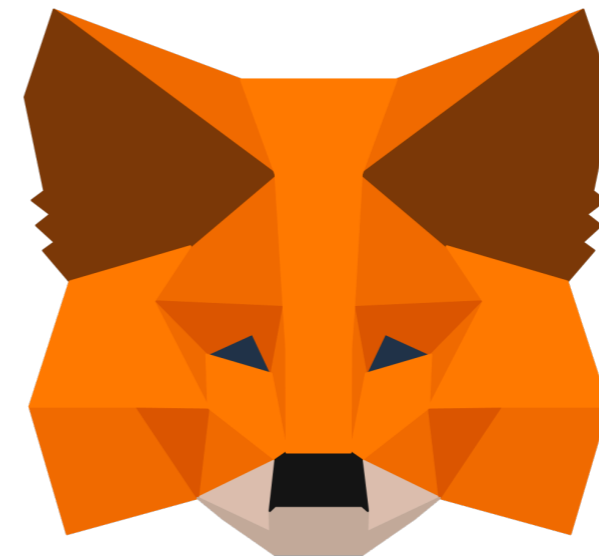
With IoT devices, the crypto wallet, specifically the wallet's *private key*, is usually embedded *within the device*, along with the wallet logic to *create and sign transactions*. This is what it means to have an **IoT blockchain**.

Wallet - another meaning



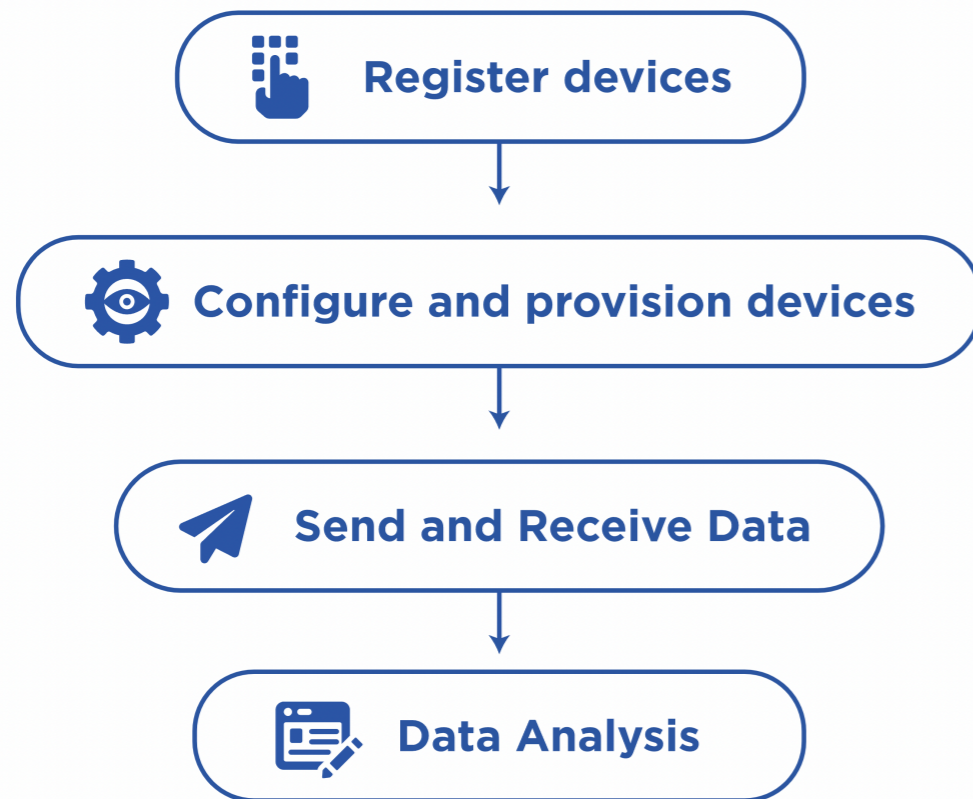
Essentially, from one perspective, a 'wallet' is simply *the numeric identity* used on the blockchain for holding (attributing) the value (coins) that the blockchain manages. From another perspective, a 'wallet' is the software that *prepares transactions to be submitted to the blockchain's miners/validators*.

End User Wallets



For web apps, a wallet is likely a browser extension such as **MetaMask**, **Keplr** or similar. On mobile phone devices, many crypto wallet apps are available. In all cases, the funds (coins) are *held on the blockchains*, not within the wallets. The software wallets simply interrogate the blockchains to discover balances and transaction histories and *prepare transactions* to be sent to miners/validators. What provides access to the funds held at the wallet addresses is the private key for the associated wallet. Software wallets need access to that key to be able to sign transactions.

IoT Wallet Administration



IoT devices generally have the required private key embedded in the device itself, though other discovery schemes are possible.



Browser extensions and software wallet apps can either (i) have the private key held accessible to it, usually in encrypted form or (ii) they can interact with a hardware wallet such as **Ledger** or **Trezor** which stores the private key within the device but never reveals it, rather they simply sign transactions when the user authorises them through the hardware wallet's user interface.

Particularly where the private key is held within the browser extension software or the mobile app, there is a risk of private key loss, along with the attendant loss of funds.

This is another case needing risk assessment. Also, with IoT devices, a risk assessment is required for the possible loss of private keys from the device by the capture and analysis of the device itself. This leads to the next topic of *private key management*. Does each IoT device have its own private key and do clusters of devices share a pattern that might be used to guess private keys?

Private Keys Management

The most onerous and perilous aspect of cryptocurrency is the *maintenance of private keys*. This dynamic was introduced with the Bitcoin blockchain and is inherent in all cryptocurrencies. Corporate custodianship is replaced by the *self-custody* of the private keys that action wallets

transfer on the blockchain ledger. Never before had *cryptographic keys been used to entitle access to funds in such an absolute manner*. Make no mistake. It is a quite different situation from passwords being used as an authentication scheme, or account numbers being used at high street banks. In both those situations, funds are not lost if you lose your password or your bank account number. They are with cryptocurrency private keys. Forever.

Public Key Infrastructure (PKI)

The invention of public key cryptography, where a key pair, one private, one public, is used for private communications and non-repudiation of messages led to the introduction of public key infrastructure (PKI), securing all manner of *communications*. In PKI, if keys are lost, or certificates expire, a new key pair is generated and used in their place, without any *loss of infrastructure*. It is true that if an individual, or organisation, encrypts a file with a key that is later lost, then the data held in that file will likely then be lost forever, but data often can be recreated and this situation rarely leads to the absolute loss of funds. Consider PKI's use in HTTPS protocol. The browser clients do not have any key-loss risk. They hardly have any key management to contend with either. Not so with

cryptocurrency. Bitcoin recognises a wallet address as a number (public key) where it can recognise if that number's corresponding private key has been used to sign wallet transfer messages (transactions). If the private key has been used and there are sufficient funds at the wallet address, the transaction is allowed; otherwise, it isn't. Bitcoin also issues new currency every block (about every 4 minutes) into the successful miner's wallet address. The amount of new currency is a fixed amount that, according to the protocol, halves approximately every four years. The chain of these blocks *is the Bitcoin ledger*, which is unencrypted and viewable to everyone with an internet connection in its entirety.

Types of Loss

If the private key to a wallet is lost, so are the funds held at that wallet address. This is true in several ways. Not only in the sense that if the private key cannot be recovered, then neither can the funds in the corresponding wallet address, but also if the private key gets discovered by other parties, then those other parties may withdraw the funds from the wallet into another one that the perpetrator alone controls.

The Custodianship Option

One may consider relinquishing control of the funds in favour of engaging in a *banking-style relationship* with an organisation holding the private keys to your cryptocurrency. You *give over custody* of your funds for the bank to custody them instead. This requires a high degree of trust that sometimes proves ill-advised. Hence also the cryptocurrency slogan, “Not your keys; not your crypto”.

There is no getting around it. Where there is a blockchain-based application, there is also the task of cryptographic private key management. This attendant process needs documenting, rehearsing and an assessment of *every type of risk*.

End-User Key Management

A single end-user holding cryptocurrency is a common use case and well documented, though not a situation without risks and common frequent failures.

Key Management with Hierarchical Deterministic (HD) Wallets and Seed Phrases

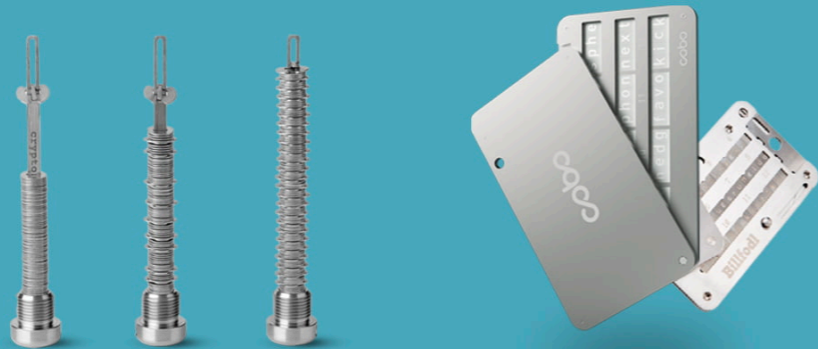
Improvements to Bitcoin are proposed and managed through “Bitcoin Improvement Proposals” (BIPs) held on GitHub. One such is BIP32, “Hierarchical Deterministic (HD) Wallets” attributed to Pieter Wuille in 2012 from the idea of

Gregory Maxwell. BIP32 introduces the scheme where one *seed number* can be used to deterministically generate a tree of a maximum of 256 x (private, public) key pairs to be used as wallets. For an individual, this means only having to manage the *single seed number*, which simplifies key management. Another innovation is BIP39 where the seed number is represented as a list of natural, say English, language words from a predetermined numbered list of 2048 words. 2048 (decimal) is exactly 10 bits (2^{10}), so a sequence of 12 words is exactly 120 bits, or 15 x 8-bit bytes, which can be used as the seed number for the HD wallet. Now, these 12 words can be used as the seed phrase for all the wallet addresses. This is the scheme used by hardware and software HD wallets in common use.

This now reduces the challenge to only securing a single 12-word phrase, rather than multiple long strings of digits.

Private Key Physical Security

best crypto metal plates



protect your seed phrase.

- Multiple-signature (multi-sig) arrangements, where an “n of m” allows access.

Size Considerations

Bitcoin’s original design targeted, perhaps a proof-of-concept, a ledger for making small online payments to websites. I doubt Satoshi was considering wallets to hold the high-value funds that Bitcoin does today. The cyberpunk triumph of securing funds using cryptography surely did not take into account the perilous situation of having those funds *forever at risk of private key management*. We have improved the key management situation from the very early days of Bitcoin, but we are still in a difficult place with our private key management, which either has to be impeccable or face the total loss of funds to lost wallets.

Market Models

Finally, where the application implements or makes use of tokens (or coins) as the digitisation of assets and the market for the tokens are traded on exchanges, then there exist *market risks* to the *liquidity* and *price* of the tokens.

There remain the same physical storage challenges and risks. Some techniques to consider are

- Indelible inks used to write down seed phrases on archival paper, protected by plastic wrap.
- Seed phrases stamped out on stainless steel sheets
- Safes and personal deposit boxes.
- Splitting phrases across multiple locations.

This might be considered a *financial risk only* and outside the scope of security, but that is short-sighted. Where the application makes use of a dedicated asset say, which is traded on exchanges, vulnerabilities of the market model for the asset can create situations that create arbitrage opportunities that are exploited to devastating effect. This can severely affect liquidity, bringing an application down.

Flash Loan Attacks

Ever since finance went online and online currency exchanges were introduced, there have been risks from the combination of

1. Large pools of capital
2. Fast automated transactions
3. Arbitrage opportunities
4. Price fluctuation, caused by liquidity changes brought on by large transactions

Decentralised exchanges (DEX) and Defi increase this risk because of the opportunities for large, very short-term (unsecured) loans, known as *flash loans*.

Speed is at the essence here. Flash loans occur in a *single transaction block*. They can, in the space of a single block,

- Take out a loan
- create an arbitrage opportunity
- exploit the opportunity
- Repay the loan

That's a profit to the flash loan arbitrager and a loss to the market. Since this is automated, it is straightforward to repeat the automation to devastating effects.

Bear in mind that the transaction block is the *entirety of financial activity* occurring on its blockchain for that time period and *all transactions* in the block can be sequenced.

As decentralised financial instruments get more complex and numerous, the opportunity for these types of attacks increases.

These arbitrages are generally known as *attacks* since they incorporate a degree of *market manipulation* which is a criminal activity in the regulated financial industry.

Decentralised Apps can reduce the risk by becoming more sophisticated about *price discovery*, viz.

1. Using *decentralised, reliable* price oracles
2. Having *more frequent* price discoveries
3. Using *time-weighted average* pricing.

Maximum Extractable Value (MEV)

It is a similar situation with MEV, which was initially an abbreviation for Miner Extractable Value (MEV), but little changes from proof-of-work (miners) to proof-of-stake (validators), so is largely considered to mean Maximum Extractable Value (MEV) now.

Within a single block, there is an opportunity for those creating the block to *front-run trades*. Say a large DEX purchase (swap) transaction is part of a block, reducing the amount of asset X. This will cause the price of X to increase. A *sandwich attack* places an X buy order immediately before the large transaction and an equivalent X sell order immediately after it. This way the creator of the sandwich profits from the increased price. It is a way the

miner or validator can *maximise* the *value* that they *extract* from the blocks they mine/validate, hence, MEV.

MEV is not an immediate security risk to a blockchain application but should be considered a *financial risk* that can be managed by the choice of technology deployed to miners/validators. For instance, if miners were not able to detect the number of transactions, they would not be able to create sandwich front running.

If you are interested in MEV, the [MEV-Explore](#) website provides some great information and charts, even though it is just for the Ethereum blockchain and just limited to on-chain, using lower-bound estimates, for a limited number of protocols and not including bandwidth exploits.

\$675,524,491

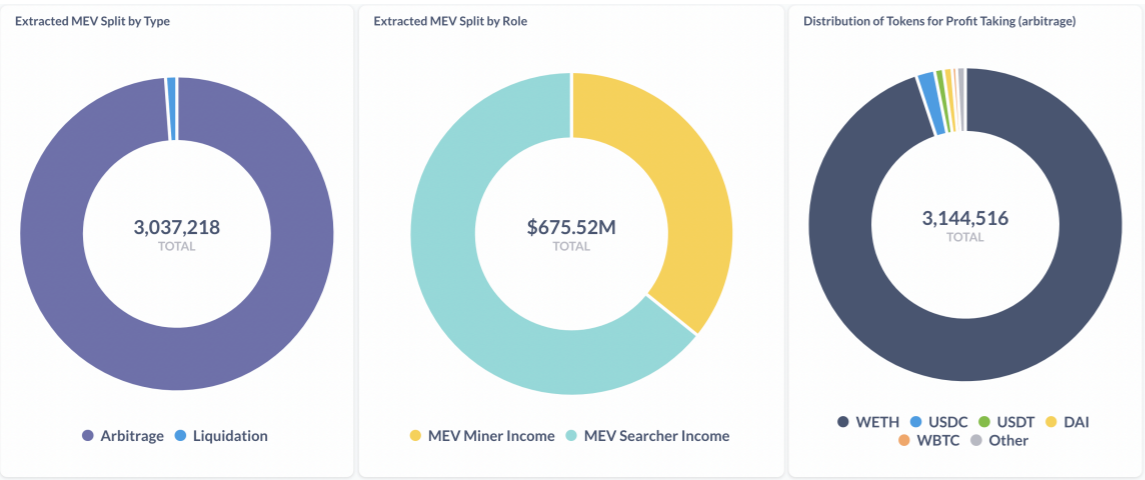
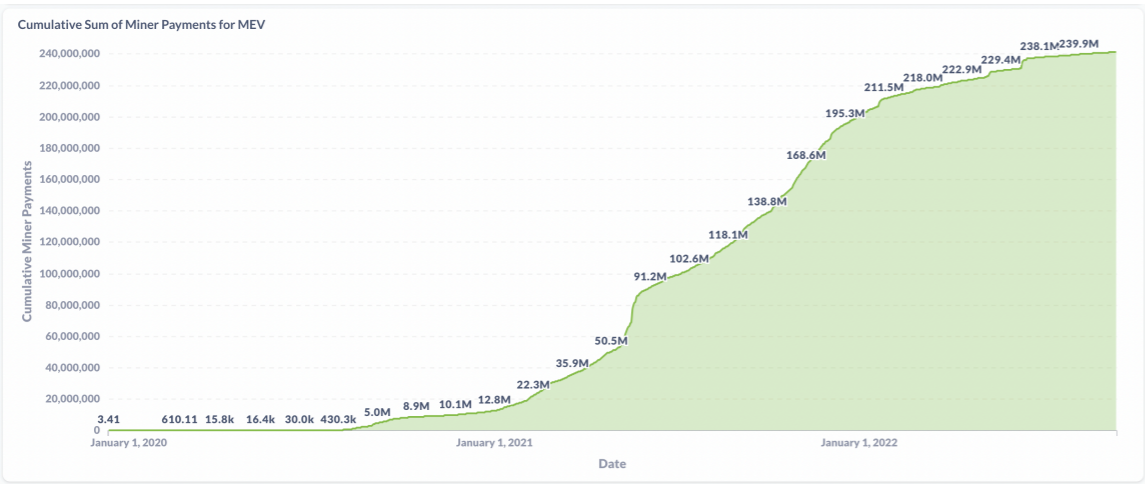
Total Extracted MEV ⓘ

\$1,376,528

Last 30 days Extracted MEV

k

Last 24h Extracted MEV

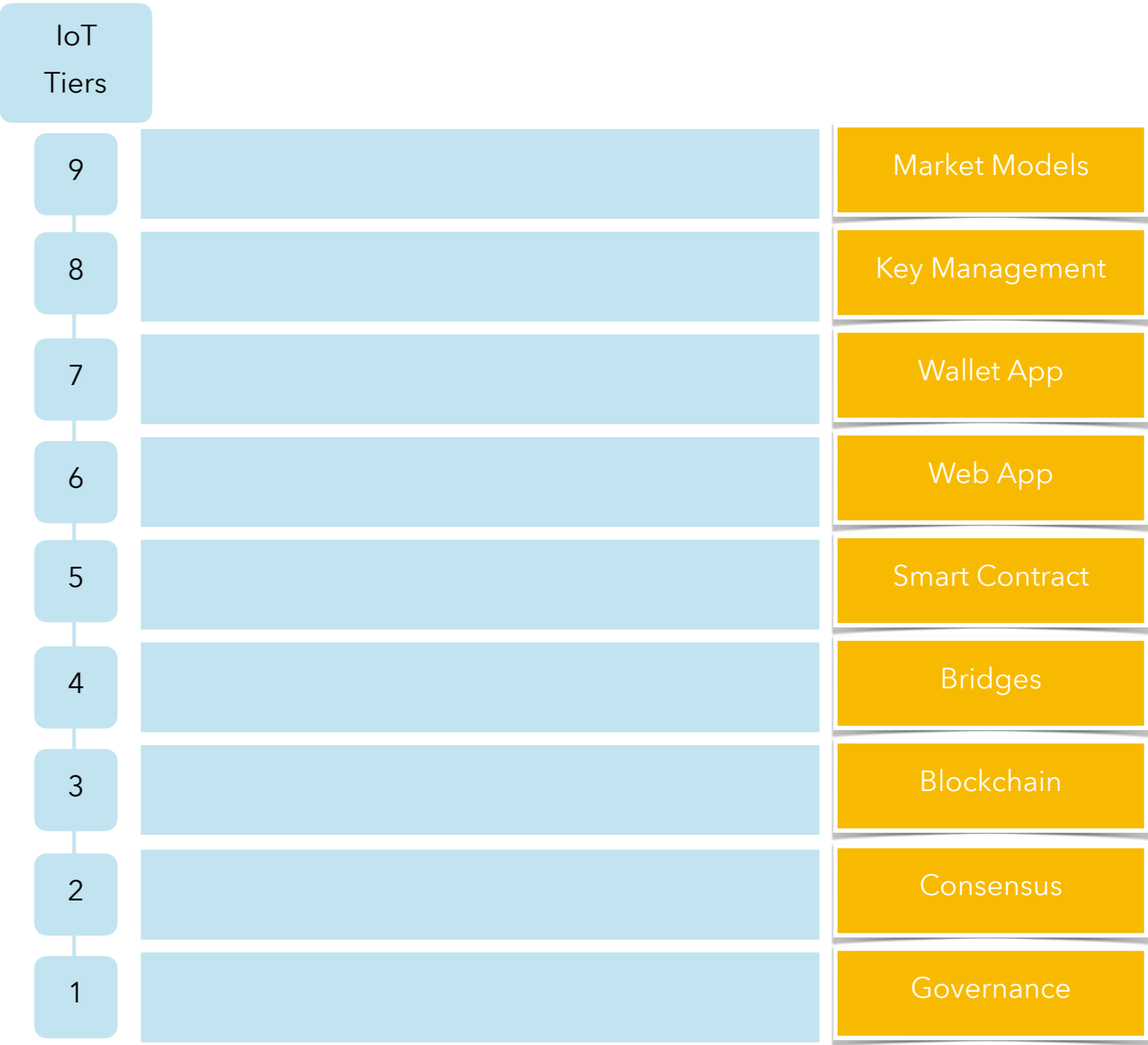


REFERENCE ARCHITECTURE

Layers most important to Blockchain Apps security are considered.

The nature of public blockchain technology leads to different risks profile to traditional TCP/IP network architecture. There are more in the less mature areas of blockchain architecture.

Blockchain IoT Reference Architecture



Tier	Threat	Cyber Procedures	Cyber Defence Solution	Advanced Features
1. Governance	Public Shared-Tenancy Blockchain - Loss of miners'/validators' interest/incentives, leading to abandonment.	Regular (say annual) blockchain viability assessments		
1. Governance	Industry Consortium Dedicated Blockchain - Lack of participation	Pre-deployment assessment. Establish and maintain governance forums. Consider implementing on-chain voting of governance proposals, which leads to better efficiency.		
2. Concensus	51% Attack of public blockchains - i.e. colluding miners'/validators' creating false or tampered blocks that benefit	Regularly monitor <i>Miner/Validator centralisation</i> - i.e. the voting power		

Tier	Threat	Cyber Procedure	Cyber Defence Solution	Advanced Features
3. Blockchain	Core Blockchain software insecurities - i.e. the underlying <i>trustworthiness of the ledger</i> is at risk of software insecurities being exploited by miners/validators or other parties.	Run source code audits for existing and new/changes to the core blockchain code. Engage specialists to review every change made to the code base.		Where any changes are made to cryptography and hashing algorithms or their implementations, engage professional cryptographers to review the changes.
3. Blockchain	Core Blockchain Encryption broken by Quantum Computers - New quantum computers will be able to <i>search the available number space</i> much quicker than is currently possible and therefore discover funded wallets (private, public) key pairs to exploit.	Quantum-safe encryption algorithms have been and continue to be deployed. If this risk is a concern, ensure only quantum-safe algorithms are used in the blockchain logic.		
4. Bridges	Bridge implementation is financially attacked using flash loans or other novel technologies or services.	Complete a risk assessment for the deployment of any bridging	Assess attacks by flash loans and other off-blockchain financial	

Tier	Threat	Cyber Procedure	Cyber Defence Solution	Advanced Features
5. Smart Contracts	Smart Contract implementation has zero-day vulnerabilities that can be exploited – see “The Dao Dao” hack.	<p>Have all smart contract source-code reviewed internally and then externally.</p> <p>Create internal security networks (test-nets) and penetration test smart contracts.</p> <p>For all smart contracts, create a risk assessment and complete contingency plans should the smart contract be exploited - i.e. decide what action will be taken should this occur.</p>		
6. Web Apps	The web application is compromised, in a similar manner to other website	Assess the extent of the security threat. This is specific to the <i>blockchain application</i> . If an attacker		

Tier	Threat	Cyber Procedure	Cyber Defence Solution	Advanced Features
7. Wallet Apps	1. Browser Wallet - Loss of private keys and therefore coins/tokens	<p>Adopt hardware (HW) wallets where end-users need to use web apps, browsers and browser wallets. This has cost implications and may not be <i>economically possible</i> in some cases.</p> <p>If HW wallets are not practical, protect against loss of funds due to loss of private keys - i.e. protect against the situation where a private key can no longer be retrieved, ever. Consider that where keys are <i>stored in any online form</i>, they are particularly vulnerable to the many highly motivated hackers online. Consider schemes where private keys are <i>never stored online</i>, or if they must be, use <i>zero-knowledge cloud services</i> as a minimum.</p>		
7. Wallet Apps	2. IoT Device Wallet - Loss of private keys and therefore coins/tokens	<p>Ensure IoT devices adopt IoT device security best practices. - See the IoTSI's "IoT Security Reference Architecture". Additionally, ensure that the private keys held on the devices are <i>held in encrypted form only</i> and use hard-to-reverse engineer encryption methods.</p>		

Tier	Threat	Cyber Procedure	Cyber Defence Solution	Advanced Features
8. Key Management	Weak Private Key Generation - Generating, <i>weak, easily guessed/ discovered</i> private keys.	A private key is simply a number – any number (other than zero). A public key is generated from it. What keeps private keys secure is their <i>very large search space</i> to identify them. Using any <i>non-random</i> scheme to be included in their generation makes them more vulnerable to discovery. Folk are using these non-random schemes all the time and robots are continually searching for occurrences of the same and removing funds as soon as they are detected. Ensure the private key, probably more likely a seed phrase is generated <i>truly randomly</i> .		
8. Key Management	Private Key Accidental Disclosure Online - Storing private keys online and having third parties discover them and exploit them (withdraw funds to their wallets)	Do not store online private keys , or probably more likely, seed phrases. Rather store them on <i>archival safe media</i> in safes and safe deposit boxes. Consider that safes and safe deposit boxes may fail, so store duplicates at multiple locations. Manage the risk of bad actors getting access to the archived private keys - Use tamper-proof packaging. Private key management with cryptocurrencies is inherently hard. There is no easy solution. The higher the value of the asset, the more measures should be taken. Complete a risk assessment, of having private keys discovered.		
8. Key Management	Private Key Lost Forever - Losing a private key and therefore access to the funds that it protects, <i>forever</i> .	This is the complementary risk to accidental disclosure. Complete a risk assessment of how the private keys could be lost forever and consider all mitigations and contingency plans. Using HD wallets and seed phrases is safer than using raw private keys. Many hardware wallets and software wallets implement the same BIP032 HD wallet scheme, reducing the dependency on any particular hardware or software wallet. Private key management with cryptocurrencies is inherently hard.		

Tier	Threat	Cyber Procedure	Cyber Defence Solution
9. Market Models	Flash Loan Attack reduces the liquidity of coins/tokens.	Monitor all places where your applications coins/tokens are held, such as CEX, DEX and other financial instruments and perform a risk assessment of the then-known attacks. Consider reserving a portion of the initial coin release solely for recovery from flash-loan attacks.	
9. Market Models	Economic loss through Maximum Extractable Value (MEV) exploits.	Design encrypted transactions into the blockchain model so that MEV is less likely to occur.	

CASE STUDY

Background

Delivery Stage

Below we present a *blockchain application security case study* as an example of the framework in use. A full detailed analysis is out of scope; A document, to the level we present, would be used at the *initial solution stage* within a project. Once accepted, it would be detailed further as the solution design progresses.

Application Choice

We have chosen the establishment of a *new supply chain management application* that uses IoT devices that follow the goods being produced throughout their entire supply chain lifecycle. The application will *establish its own public blockchain* instead of sharing tenancy with others. This provides the case study with good coverage both of IoT and blockchain aspects.

Disclaimers

We have taken as *inspiration* an existing (October 2022) successfully deployed project. Any similarity should not be

construed to imply the authors had any involvement with any existing projects.

Application

Business Model - The development organisation is creating a generic blockchain-based solution for one of its customers. The plan is to make the technology and platform available to other customers as well. So the design is for multiple customers but initially, just one customer will use the solution.

Business Domain - Supply Chain Management, where, for each asset, the supply chain lifecycle details for that asset are captured as asset transactions on the blockchain, including date/time stamping, counterparties identification and weights and measures.

Workflow - A unique IoT device dongle is attached to the device at the earliest stage in the supply chain. At each subsequent stage of the supply chain, the blockchain is automatically updated with information captured at the source, on encountering the dongle.

Benefits - The trustworthiness of the blockchain ledger increases business confidence and assurance of activities and provides timely information for all parties.

Web Portal - An internet-facing, corporate *web portal* provides authenticated access to inspecting the supply of goods. It inspects the blockchain to discover goods, including their location and associated information.

Blockchain Economic Model - The *initial VVV coin mint* has allocations for development funds, treasury, customer (initial) funding and staking rewards. Blockchain transactions and smart contracts consume VVV coins. The customer has an ongoing cost to acquire VVV coins to fund the wallets of its dongles. VVV coin liquidity has been established on three public cryptocurrency exchanges. VVV coins are also required by the staking nodes to secure the network.

Blockchain - The blockchain code is developed in-house from the open-source COSMOS code-base. This is a mature, highly regarded project that has seen adoption by Binance, crypto.com and many others.

IoT Dongle - An always-on, IP-enabled device with a unique identity, which is its *blockchain wallet address*. The

dongle holds the *associated private key* of its wallet address encrypted and embedded within the device itself. The dongle is responsible for signing blockchain transactions from its wallet. The *generic asset update pattern* is this - the supply chain stations read the wallet address from the IoT dongle, may discover its status thus far from the device itself and/or the blockchain and prepares a blockchain transaction to update the then status, which the *dongle needs to sign* before the station submits the transaction to the blockchain. This proves the station encountered that dongle at that time at that point in the supply chain.

Solution Architecture

1. Governance

A blockchain governance model has been established which includes the customer organisation and key industry bodies and the validating nodes. An on-chain voting system will be established for governing changes to the blockchain. New customers will be added to governance.

2. Consensus Algorithms

A proof-of-authority consensus algorithm has been designed. Nodes are incentivised through staking rewards. Partners have been established.

3. Blockchain Implementation

The new blockchain is being developed in-house from COSMOS SDK open-source codebase with experienced software development staff.

4. Blockchain Bridges

No bridges to other blockchains are part of the solution, initially.

5. Smart Contracts

The solution includes the development several smart contracts that *implement bespoke transactional behaviour* between various actors along the supply chain. This protects the interests of all parties involved in the solution and increases confidence in the reliability of the blockchain ledger solution. The core set of smart contracts will be extended as a result of consultations with all the parties involved. The smart contract code will be available for review by all parties.

6. Web App

The web portal app is developed by the development organisation. It includes the core logic and facilities of the generic solution plus tailoring for each of the parties involved. Authenticated access is required for some aspects, while others are public. It uses industry-standard language, development techniques and technical infrastructure.

A public blockchain explorer is available via the web portal app.

7. Wallet App

Generally, the application's main blockchain wallets are controlled by the private keys held within the IoT dongle, so there are no traditional web wallet apps or mobile wallet apps for these devices.

To fund the wallets of the IoT dongles an administrative part of the web app interacts with an administrative wallet, implemented as a browser wallet to acquire funds from exchanges and deposit same into the IoT dongle wallets.

8. Key Management

IoT dongles are manufactured with initial state (wallet address, private key) pairs. These are randomly created.

Maintenance of the dongle can introduce a newly refreshed key pair, but cannot retrieve existing private keys. Thus the dongle's private key is never revealed, much as is the case with hardware wallets.

The web app includes management of the dongles, including a wallet address inventory.

Administrative funding wallets have their business processes defined, which include the use of hardware wallets and archiving of seed phrases.

9. Market Models

The VVV coin is the base currency introduced by the application. Initially, these coins are traded on three public exchanges to provide liquidity for their customer(s). Even though there was no business plan to develop and promote the coin, speculators are expected to purchase and hold the coin for expected gain.

The business holds a large treasury of VVV coins for future allocation of work on the blockchain and as a contingency for unexpected liquidity issues.

The solution allows for customers' tailorings to include the creation of on-chain tokens and NFTs. This does not form part of the initial customer's solution.

Case Study Security Assessment

Tier	Threat	Cyber Procedure	Cyber Defence Solution Planned	Remaining Actions and/or residual risks
1. Governance	Public Shared-Tenancy Blockchain - Loss of miners'/validators' interest/incentives, leading to abandonment.	Regular (say annual) blockchain viability assessments	Solution has the business developer controlling the tenancy of new customers as business progresses. Risk is reduced as the blockchain is not shared with general public, rather instead other IoT supply chain customers. A governance model has been established to provide ongoing governance of the blockchain. Incentives and on-chain governance have been planned.	
2. Consensus	51% Attack of public blockchains - i.e. colluding miners'/validators' creating false or tampered blocks that benefit only them.	Regularly monitor Miner/Validator centralisation - i.e. the voting power of each validator or the hashing power of each miner. Instigate anti-centralisation measures with miners/validators when predetermined risk levels are reached. - e.g apply limits to the voting power of any one validator.	A large collection of validator nodes is part of the solution. The novel business model will be protected by the <i>regular risk assessment of consensus attacks</i> . The blockchain is not a public good, so only business partners with the core development organisation will make use of the blockchain. Hacks by colluding parties to fabricate the blockchain is a possibility but a low one as the prospect of detection is high and penalties severe.	
3. Blockchain	Core Blockchain software insecurities - i.e. the underlying <i>trustworthiness of the ledger</i> is at risk of software insecurities being exploited by miners/validators or other parties.	Run source code audits for existing and new/changes to the core blockchain code. Engage specialists to review every change made to the code base.	An industry leading blockchain SDK is planned (COSMOS SDK). This, along with the cyber procedures of source code audits for changes, including block chain specialists, will ensure best practice.	Where any changes are made to cryptography and hashing algorithms or their implementations, engage professional cryptographers to review the changes.
3. Blockchain	Core Blockchain Encryption broken by Quantum Computers - New quantum computers will be able to <i>search the available number space</i> much quicker than is currently possible and therefore discover funded wallets (private, public) key pairs to exploit.	Quantum-safe encryption algorithms have been and continue to be deployed. If this risk is a concern, ensure only quantum-safe algorithms are used in the blockchain logic.	Since COSMOS SDK does not currently use quantum-safe encryption, monitor this risk as part of an annual operations security risk assessment and consider upgrading algorithms when they become available.	

Case Study Security Assessment - Continued

Tier	Threat	Cyber Procedure	Cyber Defence Solution Planned	Remaining Actions and/or residual risks
4. Bridges	Bridge implementation is financially attacked using flash loans or other novel technologies or services.	<p>Complete a risk assessment for the deployment of any bridging technology, when required.</p> <p>If appropriate, put in place upper limits on the transaction amounts/time period for the application.</p> <p>Deploy financial market monitors to generate alarms on unusual transactions behaviours.</p>	No bridges are part of the solution, but, since public exchanges are used, they may introduce risks associated with bridges.	<p>Assess attacks by flash loans and other off-blockchain financial instruments.</p> <p>Monitor the applications coins/tokens liquidity in all places that they are traded - CEXs, DEXs, etc.</p>
5. Smart Contracts	Smart Contract implementation has zero-day vulnerabilities that can be exploited – see “The Dao Dao” hack.	<p>Have all smart contract source-code reviewed internally and then externally.</p> <p>Create internal security networks (test-nets) and penetration test smart contracts.</p> <p>For all smart contracts, create a risk assessment and complete contingency plans should the smart contract be exploited - i.e. decide what action will be taken should this occur.</p>	<p>Smart Contracts are a significant part of the planned solution, so the cyber procedures identified are key.</p> <p>From the viewpoint of a customer of the development organisation, the blockchain infrastructure and coin economy is shared. However, each customer’s risk is limited to the smart contracts they interact with.</p> <p>Other organisations’ smart contracts do not put the customers’ smart contracts and coin economy at risk.</p>	
6. Web Apps	The web application is compromised , in a similar manner to other website compromises, such as network intrusion or database injection.	<p>Assess the extent of the security threat. This is specific to the <i>blockchain application</i>. If an attacker could change the web app code, what business damage could they do?</p> <p>Adopt enterprise systems and network security best practices - firewalls, pen tests, off-server log file collection, etc. i.e treat the web server and the hosted application as a commercial website.</p>	<p>By design, should the web portal app go offline, the core supply chain blockchain transactions will be unaffected. However, visibility of the supply chain would be significantly reduced, until the service is restored.</p> <p>Role based access controls (RBAC) and authentication is an aspect of the solution that needs including in security controls testing.</p>	

Case Study Security Assessment - Continued

Tier	Threat	Cyber Procedure	Cyber Defence Solution Planned	Remaining Actions and/or residual risks
7. Wallet Apps	1. Browser Wallet - Loss of private keys and therefore coins/tokens	<p>Adopt hardware (HW) wallets where end-users need to use web apps, browsers and browser wallets. This has cost implications and may not be <i>economically possible</i> in some cases.</p> <p>If HW wallets are not practical, protect against loss of funds due to loss of private keys - i.e. protect against the situation where a private key can no longer be retrieved, ever.</p>	<p>The administrative wallet has a <i>defined process</i> for using hardware wallets and maintaining archives of the seed phrase. Since there is a limited number of administration wallets, it is possible to use hardware wallets.</p> <p>Effective management of seed phrases is a necessary cost and the process is defined and will be risk assessed.</p>	
7. Wallet Apps	2. IoT Device Wallet - Loss of private keys and therefore coins/tokens	<p>Ensure IoT devices adopt IoT device security best practices. - See the IoTSI's "IoT Security Reference Architecture". Additionally, ensure that the private keys held on the devices are <i>held in encrypted form only</i> and use hard-to-reverse engineer encryption methods.</p>	<p>The IoT dongle is a bespoke implementation of a hardware wallet, since it follows the pattern of holding, but never revealing the private key.</p>	<p>Ensure that the implementation of the IoT dongle is reviewed and risk assessed to ensure the private keys are only ever stored in encrypted form, that strong encryption is used for this.</p> <p>The key risk is one where the IoT dongle protection could be easily bypassed for all dongles, thereby bypassing application security designs.</p>
8. Key Management	Weak Private Key Generation - Generating, <i>weak, easily guessed/discovered</i> private keys.	<p>A private key is simply a number – any number (other than zero). A public key is generated from it. What keeps private keys secure is their <i>very large search space</i> to</p>	<p>Ensure a <i>code and encryption review</i> is performed for the generation of the initial random IoT dongle private keys.</p>	<p>Create a challenge to hacking researches to be able to hack the IoT dongle device and either extract the private key or otherwise compromise the device.</p>

Case Study Security Assessment - Continued

Tier	Threat	Cyber Procedure	Cyber Defence Solution Planned	Remaining Actions and/or residual risks
8. Key Management	Private Key Accidental Disclosure Online - Storing private keys online and having third parties discover them and exploit them (withdraw funds to their wallets)	Do not store online private keys , or probably more likely, seed phrases. Rather store them on <i>archival safe media</i> in safes and safe deposit boxes. Consider that safes and safe deposit boxes may fail, so store duplicates at multiple locations. Manage the risk of bad actors getting access to the archived private keys - Use tamper-proof packaging. Private key management with cryptocurrencies is inherently hard. There is no easy solution. The higher the value of the asset, the more measures should be taken. Complete a risk assessment, of having private keys discovered.	The solution does not store private keys or seed phrases online. The <i>administrative seed phrase</i> is stored with defined processes, including archival media, tamper-proof containers and multiple safe deposit boxes.	Still, this remains one of the weakest aspects of the solution, inherent with cryptocurrency solutions. No short-cuts are to be taken in the management of the administrative seed phrase.
8. Key Management	Private Key Lost Forever - Losing a private key and therefore access to the funds that it protects, <i>forever</i> .	This is the complementary risk to accidental disclosure. Complete a risk assessment of how the private keys could be lost forever and consider all mitigations and contingency plans. Using HD wallets and seed phrases is safer than using raw private keys. Many hardware wallets and software wallets implement the same BIP032 HD wallet scheme, reducing the dependency on any particular hardware or software wallet. Private key management with cryptocurrencies is inherently hard.	This very situation has occurred for a number of high profile cryptocurrency projects such as Polkadot and Harmony One, where private keys (or seed phrases) to high value wallets have been lost and therefore the associated funds as well. This is why such emphasis has been placed on a mature defined process and ongoing diligence.	Remain diligent.
9. Market Models	Flash Loan Attack reduces the liquidity of coins/tokens.	Monitor all places where your applications coins/tokens are held, such as CEX, DEX and other financial instruments and perform a risk assessment of the then-known attacks. Consider reserving a portion of the initial coin release solely for recovery from flash-loan attacks.	No risks have been assessed initially, but most of the risk is outside the control of the development organisation and its customers. So, review this risk regularly and constantly. A treasury has been established as a contingency against unforeseen losses.	
9. Market Models	Economic loss through Maximum Extractable Value (MEV) exploits.	Design encrypted transactions into the blockchain model so that MEV is less likely to occur.	Wherever there is a market (trading occurs) there will be exposure to the MEV risk. This is the case here. MEV must be managed and therefore measured as well.	When the COSMOS SDK includes provisions for encrypted transactions, adopt this to limit MEV exploit. This is particularly important since the platform supports multiple clients / customers.

IoT SI Issue Reporting

All IoT SI documents are subject to continuous review and improvement. As part of this process, we encourage readers to report any ambiguities, inconsistencies or inaccuracies they may find in this document or other IoT SI materials by sending an email to *admin@iotsecurityinstitute.com*

Creative Commons Licensing Agreement

Attribution-NoDerivs CC BY-ND



This license allows for redistribution, commercial and non-commercial, as long as it is passed along unchanged and in whole, with credit to the IoT Security Institute.



Abbreviations and Terminology

Acronyms and terms	Description
DAG	Directed Acyclic Graph. An alternative data structure to the blockchain with a graph structure rather than a simple sequence of blocks. Blockchains using DAGs are badly named, but the term blockchain has come to mean much more than a data structure topology.
Defi	Decentralised Finance. A financial system without the usual intermediary of a bank. Instead, smart contracts implement an algorithmic contract that anybody with a wallet can make use of. Typical financial products are loans (lending and borrowing).
Cefi	Centralised Finance. The traditional finance of banks, central banks, hedge funds, insurance companies, etc. all with their usual <i>financial products</i> . Examples - Black Rock, Barclays Bank, National Australia Bank (NAB).
DEX	Decentralised Exchange. Cryptocurrency exchange <i>without</i> a <i>centralised</i> organisation owning and running it. Instead, it is usually built and maintained by volunteers. <i>Liquidity</i> is provided by external parties by depositing funds (various cryptocurrencies) into <i>liquidity pools</i> . Users can then exchange cryptocurrencies using smart contracts. Sometimes an exchange may introduce its cryptocurrency asset and provide rewards in that asset. Examples - SushiSwap, Osmosis.
CEX	Centralised Cryptocurrency Exchange. A cryptocurrency exchange ran by a commercial organisation. Examples - Binance, Kucoin
PKI	Public Key Infrastructure. The collection of <i>technologies</i> developed from <i>public key encryption</i> , providing facilities such as HTTPS encryption, signing and non-repudiation of messages.
HD Wallet	Hierarchical Deterministic Wallets. A wallet in cryptocurrency is a <i>pair of keys</i> , one public, one private. The public one is essentially the <i>public identifier</i> for the wallet, sometimes referred to as the <i>wallet address</i> ; the private one is what is used to sign withdrawal transactions for the wallet. An HD wallet provides a scheme to generate, deterministically, a sequence of key pairs from a single long integer number, thereby providing access to many wallets from a single long integer number. This number is most often represented as a sequence of 12 (or more) words chosen from a dictionary of 2048 words. This system is used by hardware and software wallets.
HW Wallet	Hardware Wallet. These are HD wallets that contain many wallets key pairs (wallet address, private key) but will never reveal the private key in any form, instead they will only sign transactions with the private key. This protects the private keys from the many hacks attempting to capture private keys.
IBC	Inter blockchain communication. This is a standard and technology development by the COSMOS ecosystem, readily available to COSMOS technology chains, using Tendermint consensus. It can also be adopted by other chains that meet its criteria.

Abbreviations and Terminology - continued

Acronyms and terms		Description
IoT SI	Internet of Things Security Institute.	

References and Resources

[illegible]

References and Resources - continued

Domain	Area	Article	Comment
Financial Models	Financial Attacks	Deep Dive into Flash Loans	Understanding the mechanism
Financial Models	Financial Attacks	Analysis of Flash Loans	Understanding the mechanism
Private Key Management	Hierarchical Deterministic (HD) Wallets	BIP32 - HD Wallet Specification	Understanding the mechanism
Private Key Management	Hierarchical Deterministic (HD) Wallets	BIP39 - 2048 English Words	The words used in seed phrases