Chapter 4
Balancing Security: A Moving Target

Artemij Voskobojnikov, Volker Skwarek, Atefeh Mashatan, Shin'Ichiro Matsuo, Chris Rowell, and Tim Weingärtner

4.1 Introduction

4.1.1 Security

In general, security is a non-functional but essential requirement of any IT system. It is also applicable to any IT system based on blockchain technology. However, with blockchain technology, we must differentiate between the usual security of

A. Voskobojnikov (✉)
Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, BC, Canada
e-mail: voskart@ece.ubc.ca

V. Skwarek
Department of Industrial Engineering and Management, Hamburg University of Applied Sciences, Hamburg, Germany
e-mail: volker.skwarek@haw-hamburg.de

A. Mashatan
School of Information Technology Management, Ryerson University, Toronto, ON, Canada
e-mail: amashatan@ryerson.ca

S. Matsuo
Department of Computer Science, Georgetown University, Washington, DC, USA
e-mail: Shinichiro.Matsuo@georgetown.edu

C. Rowell
Sauder School of Business, University of British Columbia, Vancouver, BC, Canada
e-mail: christopher.rowell@sauder.ubc.ca

T. Weingärtner
Lucerne School of Information Technology, Lucerne University of Applied Sciences and Arts, Lucerne, Switzerland
e-mail: tim.weingaertner@hslu.ch

V. L. Lemieux, C. Feng (eds.), *Building Decentralized Trust*,
https://doi.org/10.1007/978-3-030-54414-0_4

blockchain technology as an IT system, and the requirements for securing the "crypto-assets", such as Bitcoin, that blockchains seek to secure.

Definitions of blockchain vary and are still in formation. From the original Bitcoin paper written by Satoshi Nakamoto, the technology was explained as "*an electronic payment system based on cryptographic proof instead of trust, allowing any two willing parties to transact directly with each other without the need for a trusted third party*" (Nakamoto 2008). In this chapter, we define a blockchain as a solution to unauthorized changes to data integrity and to the double-spending problem in a distributed system, using a peer-to-peer distributed timestamp server to generate computational proof of the chronological order of transactions.

The underlying mathematics of Bitcoin is summed up by the definitions above. That is, blockchain technology assures only the chronological order of transactions without a trusted third party. This is the core and mandatory security requirement of a blockchain. On the other hand, other security requirements may be desirable; for example, protection of the privacy of identity regarding each transaction, and the confidentiality of transaction data. The former is partially satisfied by the original Bitcoin implementation, the latter however is not. Thus, these requirements are optional.

In general, security requirements depend on the specification of each IT system. Therefore, when we discuss blockchain security, we need to be careful about the relationship between what we need and what blockchain technology provides. Unfortunately, blockchain technology is not a silver bullet for security issues in our networked society. As described above, the security that this technology offers is incomplete in terms of generally accepted fundamental requirements such as confidentiality, integrity, and availability (CIA) (Stallings and Brown 2018). Thus, we need additional controls for enhanced security.

Blockchain technology implicitly has many (sometimes hidden) assumptions in its operations. Designers of cryptographic protocols do not make general claims such as "all private key (signing keys in the case of blockchains) should be kept secret" because in real-life settings it is very hard to accomplish. Many incidents have led to hundreds of million-dollar losses from cryptocurrency exchanges caused by the difficulty of key management. We need to be aware that cryptography is not the root of trust for *confidentiality*, *integrity,* and *availability*. It is merely the mathematical tool to transform the confidentiality, integrity, and availability problem to the "key management" problem.

Similarly, there are many assumptions in the core blockchain protocol itself. To make the distributed consensus secure and to avoid control by one entity, we need a well-distributed and large enough number of nodes. This number of nodes should also be sustainable. For Proof-of-Work blockchains, poorly distributed hashing power can result in arguably one of the most prominent attacks, the 51% attack. An uneven hashing distribution would allow the group of miners with 51% or more of the overall hashing power to omit new transactions or double-spend coins (see Sect. 4.2.2 for more details).

From the above, when considering the security of blockchains and associated applications, we need to be cautious about what the technology offers, what the assumptions are, and how we can assure requirements that the mathematics of

blockchains do not cover. This type of security-by-design thinking should align with the existing Information Security Management System (ISMS: ISO/IEC 27000) framework.

4.1.2 Trust in an Untrusted Environment

Blockchain technology is colloquially considered as "trust-free", or "trustless", implying the lack of a central governing authority. This notion however can be misleading, as "trustlessness" is often merely a responsibility shift. Transaction partners do not have to trust one another, as long as both share the common belief that the underlying technology will perform as expected. This shifts trust from other places (e.g., identity and access management) to trust in the mathematics of a given blockchain system.

Trust by itself is multi-faceted, and definitions depend on the context and field of study. Two conceptualizations of trust are prevalent in literature, with one being an expectation of a certain behavior in relation to an interaction partner, and the other being seen as willingness to be vulnerable (Beldad et al. 2010). For offline interactions, where people or groups are the interaction partners, the concept of trust might appear obvious. For online interactions however, and blockchain-based systems in particular, the interaction partners might appear intangible and we therefore have to consider how trust is being developed and maintained. Here, social trust between interaction partners is often the result of trust in the technology.

Depending on the type of blockchain, different effects on trust can be expected. Permissioned blockchains offer a certain level of clarity when it comes to interaction partners and their responsibility. Clear governance guidelines and defined roles can alleviate concerns—such as dishonest or malicious actors—associated with public, permissionless blockchains. Here, we have to ask ourselves how to establish trust in an environment that includes actors operating in bad faith, including fraudulent start-ups and exchanges. While transparency can certainly have a positive effect on trust, other antecedents need to be explored that can help in creating the distinction between good and bad actors, which is of utmost importance. Such signals of trustworthiness are emitted by the trustees and create a context in which expectations are being formed by the trustor. Signals are categorized into symbols and symptoms (Riegelsberger et al. 2005) and vary in regard to the degree of reliability they provide. Traditionally, symbols are trust badges, but can also include reputation systems. Both can be easily mimicked by untrustworthy actors, but only if the perceived benefit exceeds the cost of emitting said symbol. Symptoms, however, are generally seen as a by-product of trustworthy actions, and are usually costly to mimic. A large, open-source code base could, for example, emit trustworthiness, as could a large customer base. The latter, however, has to be viewed with caution, as the pseudonymity of blockchain transactions can be leveraged by dishonest actors to artificially alter symptoms indicating growth. As a case in point, unregulated exchanges inflated their trading volumes by up to 95% to signal that the market was stronger than it actually was (Blockchain Transparency Institute 2018).

"Trustlessness" therefore is not a fitting term to gauge the complex trust relationships spanning over social, data/records, and technical layers of a blockchain environment. These relationships are influenced by the stakeholders, their needs, as well as operating contexts, and, moreover, interpretations of whether a system is trustworthy might vary depending on the application area.

4.1.3 Privacy on Blockchains

On a broad level, privacy is considered a basic human right, as recognized in the United Nations Universal Declaration of Human Rights (1948). The definitions are wide-ranging and often include the right to be left alone and the freedom of association. With the rise of big data and emerging technologies such as artificial intelligence and cloud computing, the focus has shifted towards appropriate use and storage of the personally identifiable data of customers. The appropriateness of data usage must be in accordance with laws and policies applicable in the jurisdictions in which a given organization (e.g., a cloud service provider) operates. Naturally, the requirement to abide by laws and policies governing privacy and data protection also holds true for blockchain technologies.

When storing or handling personally identifiable information (PII) in the blockchain context, we have to consider potential implications the underlying technological features might have. In particular, blockchain technology's decentralized nature, as well as the immutability of ledger records, might pose challenges for compliance with regulatory measures such as the General Data Protection Regulation (GDPR). GDPR includes provisions concerning the "right to be forgotten", which refers to the right of an individual to have personal information removed from public access, such as in the case of information available through an internet search (GDPR 2016; Lee 2016). Designing for, and the implications of, privacy and data protection regulations differs depending on the type of blockchain, i.e., permissioned or public, and both types have to be addressed.

For both public and permissioned blockchains, privacy considerations are imperative for system design. Decisions about what data is being stored on-chain can have grave implications for both companies and end users, depending on the application area. Blockchains are immutable by design, and in order to revert a transaction a consensus has to be reached and all participating nodes, whose number can be in the thousands, have to alter the respective local copy of the ledger. For instance, the removal of previously published PII could only be accomplished on public blockchains via achieving a consensus among all nodes, which is costly and might even be infeasible in certain cases. For permissioned blockchains however, where the number of governing nodes is comparatively small, such changes would require less effort. Similarly, updates, e.g., in the case of future regulatory restrictions, could be applied rather seamlessly in the permissioned network, given clear governing guidelines.

Privacy considerations should therefore be factored into the early design stages of the respective blockchain system, as ex-post changes often become increasingly costly and complex with time. More importantly, however, one might argue that transaction reversal, while theoretically feasible, undermines one of the core principles of blockchains, namely immutability, and should only be considered as a last resort.

4.1.4 Security as a Moving Target

Security, while widely labeled as crucial, is merely an afterthought in many cases when it comes to actual system design and implementation. Compared to centralized systems, where the attack surface is limited, blockchains can contain thousands of nodes. This dramatically increases the attack surface and makes blockchains a very attractive target. Attacks on endpoints could further have effects on the whole network, possibly resulting in the entire ledger being compromised. Reactive approaches can therefore only go so far in securing a blockchain system, especially when dealing with an ever-evolving technology; consequently, our attention has to be turned towards a security-by-design paradigm.

It cannot be overemphasized, therefore, that as important as the security-by-design approach might be for conventional systems, it is even more so for the blockchain domain. Preventing bad or vulnerable code is critical when dealing with immutability, and enforcing best practices, such as continuous testing and documentation, can help in achieving these goals. Best practices, however, can only reduce the risk of certain threats; others, such as the threat to conventional cryptography by scalable quantum computers, need to be assessed on a case-by-case basis.

While attacks on the distributed ledgers themselves are arguably more prominent, the overwhelming majority of exploits are caused by complacent end users. With rapidly evolving technologies, end users are constantly facing new challenges when interacting with blockchain-based technologies, and such challenges can result in dangerous errors leading to system failure in the worst case. Solution architects therefore not only need to account for technical vulnerabilities but also for the human factor, which is often considered as the weakest link.

4.2 Security Landscape

4.2.1 Attack Surfaces and Adversarial Goals

Prior to providing an overview of attacks and undertaking threat modelling, we have to first consider attack vectors of blockchains. Generally, an attack vector against an information system is defined as a path or means by which an attacker can gain

access to a computer or network server in order to deliver a malicious outcome (ISO 2012, 4.10). An attack surface then is defined as the combination of all attack vectors enabling the adversary in impeding the CIA security principles discussed earlier, which are extended in common security research to CIAAA (*Confidentiality, Integrity, Availability, Authenticity,* and *Accountability).*

It is an information security priority to reduce the attack surface as much as possible in order to counter the adversary. In the context of blockchain technology security, it is therefore imperative that we first examine the attack surface and clearly understand the adversarial model, to be able to better position ourselves against potential security threats (including threats to confidentiality that would negatively affect privacy).

Attack surfaces can be divided into three main categories: network-based, software-based, and user-based. Blockchain technology has vulnerabilities in all of these three categories, as described next.

4.2.1.1 Network-Based Attacks

Blockchains have an inherent peer-to-peer design and therefore are vulnerable to traditional network-based attacks such as distributed denial of service (DDoS) attacks and denial of service (DNS) attacks, e.g., DNS spoofing, where altered DNS records are used to redirect traffic (Pfleeger and Pfleeger 2002). Other blockchain-specific attacks that are carried out via the network surface include the Eclipse node isolation attack (Heilman et al. 2015); Block Withholding, referring to the decreasing of block revenue in a mining pool (Rosenfeld 2011); and Finney Attacks, which are a variation of a double-spending attack (Finney 2011).

4.2.1.2 Software-Based Attacks

The attacks that are made possible due to vulnerabilities introduced by the components of the actual blockchain structure are grouped here. This includes vulnerabilities of the consensus algorithm, as well as the underlying cryptographic primitives used in the implementation of the software. The most well-known attack against blockchain technology so far has been the 51% attack, which is carried out to manipulate the consensus mechanism by controlling more than half of the voting power, e.g., half of the mining power in Proof-of-Work consensus. All permissionless blockchains in operation so far are suffering from their consensus mechanism's weakness against this attack.

Attacks against the underlying cryptographic techniques used in blockchains are also considered a software-based attack. Quantum computing is going to reduce the effective security level of hash functions by a factor of two by means of Grover's search algorithm, which allows searching unsorted databases efficiently (Grover 1996). More importantly, Shor's algorithm (Shor 1994), which addresses the factorization problem, is going to catastrophically break the security of digital signature

schemes in current use. Hence the cryptographic techniques and digital signature schemes currently used in blockchain technology must be examined and redesigned to be made quantum resistant. Proposals are currently being examined by the National Institute of Standards and Technology (NIST).

4.2.1.3 User-Based Attacks

Blockchains are very attractive targets for attackers when ordinary human beings, without a lot of security training, are sitting at the endpoints. Many blockchain users do not adhere to proper key management given it is very cumbersome to do so, resulting in, for example, thefts of cryptocurrencies from cryptocurrency exchanges or loss of access to crypto-assets (Voskobojnikov et al. 2020). Cryptojacking is a common attack vector used to exploit the computational power of a target's computer for mining purposes. The open aspect of some platforms that accept smart contracts can allow for malicious code to be introduced and executed. In this case, the immutability of the code on the blockchain can be problematic. Introduced vulnerabilities cannot be fixed as smart contracts are immutable by design. Code audits therefore become of utmost importance prior to deployment.

4.2.2 Technical Weak Points

Having defined the attack surfaces, we can now cover common attacks in more detail. As for all systems carrying and transferring assets [defined as anything that has value to an individual, an organization or a government (ISO 2012)], blockchain technology may be subject to malicious attacks. For a better understanding of attack types and techniques, in the following section we classify and structure known attacks and vulnerabilities. The structure can be made according to different dimensions, using different models: For security considerations, it is rather common to differentiate attack levels by their protocol layer, according to the ISO/OSI-7-layer-model (Zimmermann 1980), or to the more straightforward internet protocol suite (Braden 1989). The internet protocol suite is more abstract than the ISO/OSI-7-layer-model and the latter's seven layers to four (see Fig. 4.1).

However, as blockchains are currently organized as internet applications, although there is potential for much deeper integration it is nearly impossible to assign blockchain functionality over more than the upper protocol layers. Therefore, it is more convenient to use another architectural model for blockchain technology. While many approaches for layered architectures exist (e.g., iFour Technolab Pvt. Ltd. 2019; Javeri 2019; Er-Rajy et al. 2017), some of these architectures are problem-specific, others model workflows instead of layered architectures. Consequently, none of the known approaches is suited as a reference structure for security consideration. Nevertheless, a comparison of these models leads to a consensual number of structural components, as illustrated in Fig. 4.2.

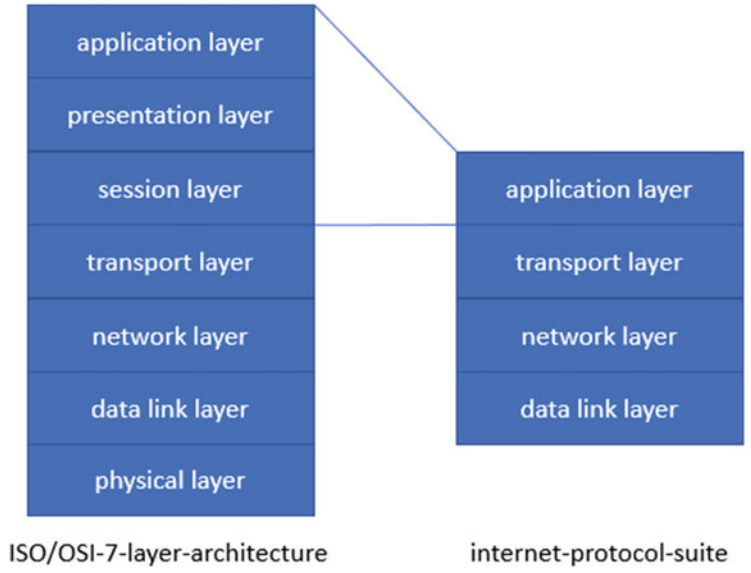

Fig. 4.1 Correlation between the ISO/OSI-7-layer-architecture and the internet-protocol-suite (adapted from Braden 1989)

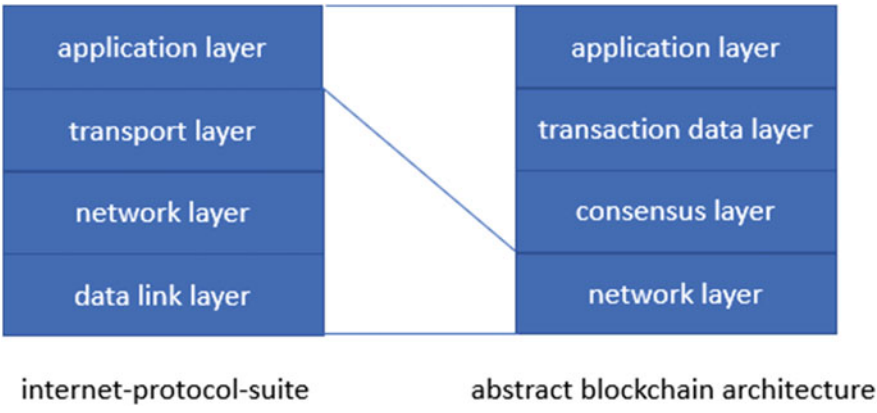

Fig. 4.2 Correlation between the internet-protocol-suite and a potential abstract blockchain architecture (adapted from Fig. 3, Wu and Tran 2018)

The four abstract blockchain architecture layers shown in Fig. 4.2 may be described as follows:

- **Application layer**: applications from in- or outside a blockchain system that are creating or working with transaction data on the blockchain, including blockchain programs such as smart contracts;

- **Transaction data layer**: all kinds of network data as created, transmitted, and received by users of a blockchain system, including code or binary data for blockchain programs such as smart contracts;
- **Consensus layer**: a mechanism for and communication about the correctness of the data–block data layer: all data required for the operation of a blockchain system including user-generated network load data such as addresses (=the value that identifies accounts participating in a transaction), transaction data, as well as system data such as hashes, block numbers or timestamps; and
- **Network layer**: the underlying peer-to-peer-network and associated package data units (PDUs).

This proposed structure offers a framework for the classification of blockchain attacks in the following paragraphs.

Weaknesses can be generally distinguished into those from inside and from outside of the blockchain system. Internal weaknesses, as in all software systems, might be due to software errors. Even in a distributed software system like a blockchain, the source code comes from one single source. Undiscovered errors are distributed to all nodes and therefore are active on all nodes. This might also be used as an attack vector. A number of the recent exploits associated with smart contracts can be attributed to this weakness (Batista and Lemieux 2019).

The more unpredictable technological weaknesses come from outside of the blockchain. If we assume that information added to the blockchain is secure, who guarantees the truth of the origin of this information? Especially if this information comes from an insecure source such as single sensors or manually added data. This can be especially problematic given the immutability of such data once recorded in the blockchain and the assumptions about trust that such recording can imbue.

The most famous attacks affecting the general public occur on the application layer. Security breaches at the interface level, such as breaches of blockchain wallets, are outside of the blockchain but affect the functionality and security of the whole system. Another weakness might come from software running inside the blockchain such as smart contracts. Though smart contracts are not directly linked to the code of the blockchain and should not affect the execution of the blockchain, smart contracts themselves can hold severe errors. A prominent example is the Decentralized Autonomous Organization (DAO) attack from June 2016, which became famous because Ether worth several million US dollars were transferred from a smart contract to another by exploiting a vulnerability in the source code (Atzei et al. 2017). Other examples affect outside-blockchain applications such as trading platforms by attacking their databases and key storage. A very comprehensive work that examines security issues of public blockchains was conducted by Li et al. (2018). This chapter extends their findings and sorts them systematically according to the layered structure as proposed above.

In the following subsections, potential attacks on different layers are listed without a claim of completeness. Ongoing research is needed, therefore, to extend and validate our taxonomy.

4.2.2.1 Attacks on the Application Layer

The most famous are *wallet or exchange hacks*, where attackers gain access to private keys and initiate malicious transactions. Alternatively, they at least manipulate the software behavior so that they can use it on behalf of the user. Additionally, intentionally or accidentally *erroneous smart contracts* may lead to unintended vulnerabilities such as investigated by Atzei et al. (2017).

4.2.2.2 Attacks on the Transaction Layer

Attacks on the transaction layer can be considered as sending corrupted net data. However, at least for data created at the user level, such an attack can be either easy or nearly impossible (=hard) to detect:

- **Simple**: an asset transfer on the blockchain refers either to a direct change (world-state-model such as in Ethereum) or to a relative change (UTXO-model) of the user state. In the case of double-spending or malicious data access, more than only incorrect data is sent; such an attack also requires that user signatures or mining processes be manipulated.
- **Hard**: in the case of user-generated data, this data is a trust anchor in that it is assumed to be accurate (which may be a false assumption) and there is no way for the blockchain to determine the correctness of the data. A singular attack on this level may affect oracles, transferring data from outside to inside the blockchain system. A corrupt oracle may manipulate the data while moving them between systems.

4.2.2.3 Attacks on the Consensus Layer

Forking attacks are most prominent at the consensus layer. 51%-attacks lead to the same consequence as *feather forking* or *punitive forking*: they change a once-met and fixed consensus within the blockchain and directly attack a trust-creating property—its immutability. Also, *selfish mining*, by producing valid blocks without publishing them, can be added to this category. As it is the goal of selfish mining to achieve the longest chain, it is hard to prevent another chain from being created in parallel as long as it is shorter than the secretly selfish mined chain. As soon as the selfish chain is ensured to be (by far) the longest chain, it may also force a fork and invalidate the parallel chain.

4.2.2.4 Attacks on the Block Data Layer

Rare but known attacks on the transaction layer are related to stolen or recovered private keys to create valid address-signature pairs for malicious transactions. In a

non-peer-reviewed but often-cited research paper by Mayer (2016), potential vulnerabilities in the ECDSA-key-generation scheme on a theoretical mathematical level were covered. Neither real attacks nor even their attempts have been shown or proven by this paper, but they leave at least a potential of recovering private keys more efficiently than brute force attacks. Most problematic is the immutability-by-concept of the transaction layer. Although the length of the keys is still considered secure, it may be hacked in some near future, and this opens the full history of a blockchain without the chance to recode it to a higher level of security.

Although *double-spending*—creating multiple independent transactions referring to the same base transaction—should be detected by the consensus layer, it is nevertheless considered as an attack because it takes some time before an invalid transaction can be filtered out. Additionally, it is hard to determine which of the multiple transactions should be accepted as authentic and be processed: within the same chain, probably the second transaction would be considered as invalid. However, if both are recorded in parallel chains, the longer chain will win. This attack generally affects multiple layers. However, to be successful, it requires some specialized knowledge about block-building, wallets, and structural overhead information.

Smart contracts may also be subject to attacks on the block data layer in terms of *transaction order attacks*: As a transaction changes the blockchain to a new state, their order of execution on the same smart contract may affect the outcome of the smart contract execution. For example, the ownership of a smart contract may be hijacked by deploying a smart contract to a blockchain. This can occur within the same block-epoch but at an earlier point of transaction handling when an asset-transaction is started from the same address that the smart contract will obtain after its deployment. In this case, the user with the earlier transaction officially owns the address of the smart contract and, therefore, also the smart contract itself.

4.2.2.5 Attacks on the Network Layer

As blockchain systems require a network for their distributed communication, they usually use internet protocol-based systems. Therefore, they are generally vulnerable to all attacks that can damage and exploit internet communication. Most famous attacks based on the routing hierarchy either abuse a monopoly position of a central routing/service instance or reciprocally damage it.

The abuse of a monopoly position is, for example, an *eclipse attack*. Here, an attacker isolates network participants by blocking their internet messages. Principally, this attack should be impossible in a hierarchy-less, distributed network; however, major blockchains such as Bitcoin or Ethereum have been known to suffer from these attacks. As an initialization- and fallback-method for message-routing via peer-nodes fixed neighbor-node tables were used (Bitcoin Core 0.11 (ch 4): P2P network 2018; Chen 2018; Leffew 2019). This was replaced with a regular update of this table and the application of the Kademlia protocol (Maymounkov and Mazières 2002), creating more information about the network neighborhood and distributing

the traffic more randomly. Nevertheless, this vulnerability still exists as the list of the initial communication nodes is still required. If the nodes are corrupt, they can divert communications to a corrupt subnetwork, blocking new node traffic.

A counterattack to a single routing node may occur by *flooding*. Independent of the actual technique and protocol layer (data flooding, syn/ack-flooding), this leads to a temporary *denial of service (DoS)* of the participant nodes until routing tables, syn/ack-lists or similar are recovered by time-outs.

4.2.3 Records Weak Points

Blockchains are designed to operate to create trust between social actors (or technical components operated on behalf of social actors, e.g., Internet of Things (IoT) devices) through enabling the creation of trustworthy records of transactions (e.g., ledger records). Underlying weaknesses in the operation of blockchain systems, such as those discussed in the previous section, can compromise the trustworthiness of ledger records. However, as records are different than data, whilst at the same time being comprised of data, in that they often serve to convey important societal rights and entitlements and provide evidence of significant social and business decisions and actions, there are additional specific requirements needed to assure that blockchains are designed to produce trustworthy records. In archival science, records are said to be trustworthy if they are accurate, reliable, and authentic (Lemieux et al. 2019).

Accuracy concerns precision, correctness, truthfulness, and pertinence (Pearce-Moses 2018, s.v. Accuracy). As noted previously, these properties can all be adversely affected if, for example, an inaccurate external data source is used in the creation of a ledger record, such as in the case of a corrupt oracle.

Reliability relates to adherence to formal procedures in the creation of records, completeness of the records in relation to those procedural rules, and the competence of the creator to create the records (Pearce-Moses 2018, s.v. Reliability). A number of aspects of records reliability depend on a determination of how reliably a blockchain system was operating at time of creation, but other aspects of records reliability can only be determined with reference to the legal, administrative and procedural context of the application of the blockchain system to a given use case.

Finally, *authenticity* concerns the ability to determine that the ledger record is what it purports to be (Pearce-Moses 2018, s.v. Authenticity), and requires an unambiguous identity of the record and its creator, and the ability to ascertain that the record has integrity (remains unchanged from its original instantiation). Blockchains excel at integrity, but very often fail to deliver identity of records and their creators. While unique transactions can be used as content addresses for blockchain transactions, they seldom create a bond between the data comprising the transaction and the legal, administrative or procedural purpose of that transaction. When an immutable bond [the "archival bond" (Duranti 1998; Lemieux and Sporny 2017)] is not instantiated in blockchain systems, over time it may become

impossible to prove that a given ledger record serves as proof that an ownership right or entitlement was conferred by the record, or for such records to serve as proof of a decision or action. This is because knowledge of the *context* of the ledger record will only be known to and determinable by those who created the records in the first place and will likely recede with the mists of time (and the failings of human memory). Additionally, confirming the identity of records creators is challenging in public, permissionless blockchains that do not require identity to carry out transactions (e.g., they operate pseudonymously), thus making it hard to determine that a ledger record authentically represents the will of a given social actor (e.g., the social actor's will to transfer a certain amount of cryptocurrency, or a cryptoasset, such as ownership of land). Both of these challenges can prevent the realization of accountability in the CIAAA model.

In addition, all of these features needed to instantiate and secure records must be made to persist over time, which requires the application of techniques of digital records preservation. Not only are these techniques not designed with decentralized technologies in mind, they also require frequent migrations to new software that can interfere with the bit-wise integrity checks of blockchain systems. New approaches, such as that being developed by the UK's ARCHANGEL project in collaboration with the UK national archives that uses AI-approaches to determine "allowable" changes in bit-wise integrity of records, could point the way to possible solutions to this conundrum (Collomosse et al. 2018).

4.2.4 Social Weak Points

The blockchain domain is rapidly evolving and is predominantly driven by technological innovation. It is therefore not surprising that both the data/records and technical layers receive far more attention than the social layer in the context of security considerations—leading to users having to adapt to existing software—and less so to software being designed with the users' needs in mind. User-induced errors are prevalent and are often exploited by attackers, more so than the underlying technology itself including the provably secure cryptography.

In traditional online systems the user is exposed to a wide range of threats as discussed above, including but not limited to *phishing*, *malware*, or *man-in-the-middle attacks*. The relevance of these threats becomes evident whenever the confidentiality of credentials is at stake. For instance, in the case of authentication/authorization schemes, such credentials are commonly used for access control, whether the asset in question is an online banking account or a cryptocurrency wallet. Focusing solely on commonalities would however be unjust, as blockchains present unique risks and challenges with which end users are directly or indirectly confronted.

Key features of blockchains, such as immutability and decentralization, are perceived as favorable by many, but can also lead to dangerous errors at the same time. Transactions are irreversible by design, implying that given no centralized

authority, the user is fully responsible for their actions. Comparable systems, such as online banking or e-commerce, provide support in case of self-induced errors, and nowadays this is considered as a norm. For blockchains, however, there is no safety net. The user is solely responsible for lost seed phrases or wrongfully-addressed transactions. Errors on the social layer therefore become of utmost importance and the responsibility shift needs to be conveyed clearly to end users. Given the severe impact such errors might have, many companies resort to designing centralized solutions, thus making themselves at least partially responsible in case of user mistakes.

Depending on the user base and application area, finding such a balance between the degree of decentralization and responsibility might become critical. Advanced users might be able to withstand a higher cognitive load when interacting with a system, whereas novice users might surrender when facing even the smallest usability challenges. Other members of society may be incapable of the cognitive effort needed, and regulatory frameworks may be needed to address such situations. Leveraging technological innovation without the users in mind will fail and user experience should no longer be considered a secondary goal, even in technology-driven domains such as blockchain technology-driven innovation.

4.2.5 Failure in Governance: Regulations and Regulatory Goals

In the real world, the use of blockchain technology could be against the social order. For example, many cryptocurrencies were and are currently used for money laundering. This raises the question of how to promote use of blockchain technology that improves the social order rather than undermining it. This is where regulations can prove to be warranted.

Originally, regulations are decided from regulatory goals. According to economic theories, regulatory goals prevent "market failure", which entails preventing crime or enabling consumer protection, and financial stability. These goals are general and, of course, applicable to blockchain-based IT systems. Regulators, however, have faced challenges in responding to the pace of blockchain innovation. The original Satoshi Nakamoto paper was published in 2008, and right after that the reference source code was provided to the public in an open source development style. This caused issues in coordinating regulatory goals and applying regulations to actual implementations of blockchain-based IT systems.

In the history of the development of internet technology, the underlying technology and mathematics come from academic research. The development of internet technology involved a wide range of expertise in order to make the technology suitable for society. Then, companies created actual implementations. After standardization, which arose from multi-stakeholder discussions, the real business

started. This sequence of steps created harmonizations among technology and social order (including regulations).

But, in the case of Bitcoin, the actual business started without verification backed by academia and multi-stakeholders. Security of cryptocurrency exchanges is one of the issues caused by such shortcomings. It touches one of the core issues of concern to regulators; that is, consumer protection. Blockchain engineers have not always considered the requirement for consumer protection. Regulators do not have a common language to talk with the open-source engineers. Business entities try to use immature technology to handle hundreds of millions of dollars, and venture capitalists force companies to start their business as early as possible without a truly mature technology. For consumer protection, transparency to the consumer is the crucial aspect; however, it is very difficult for the average consumer, and even many experienced investors and engineers, to critically review many so-called "white paper" documents to determine whether they should cast their money into Initial Coin Offerings (ICO). In general, it is too difficult to judge if specific source code is sufficient from a consumer protection point of view. This is a major missing element in order to rely on the security of blockchain-based systems as a true social foundation.

4.3 The Moving Target: Open Security Challenges of Blockchains

4.3.1 Longevity Requirements for Security of Blockchains

Longevity of security is critical in blockchain security design and implementation to ensure sustainability of the blockchain and its data in the long run. Future threats to the underlying security mechanisms, such as the quantum threat to standardized cryptography and technological obsolescence of blockchain software, and their long-term implications on longevity of blockchains, should be considered and planned for now. The challenge here is to design a system that is going to resist all future attacks and the creative ways adversaries are going to use to try to undermine the security of blockchain systems, as well as be secure against the exigencies of time. This is a near impossible task. Instead, a more practical approach should plan and design for agility so that we can switch between algorithms when a new one is necessary, or migrate seamlessly and without disruption to new software protocols.

We use cryptographic techniques to achieve integrity, authenticity, and confidentiality. Quantum computing may someday defeat critical components of areas of cryptography that are widely used in blockchain implementations. We had the industry-wide SHA1 to SHA2 migration in 2015. Change management aspects of this migration were very costly and time consuming. In the context of blockchain security, the natural question would be to ask: What happens if SHA256 is also deprecated? Another example is digital signatures that are used for integrity and trust

in the system. Quantum computers can solve the underlying mathematical problem, i.e., Integer Factorization Problem and Discrete Logarithm Problem (Shor 1994). This breaks our most commonly used digital signature schemes, such as ECDSA and DSA. Although scaleable quantum computers are not yet available, we need to plan for the eventuality that these will be available in the near future.

Any cryptographic migration might entail a fork of a blockchain. Managing forks is not a straightforward task and adds another complexity layer to the longevity requirement of blockchain security.

4.3.2 Regulation, Operation and Security

When considering the security of IT systems, there exist many ISO/IEC and other standards as comprehensive frameworks. For cryptographic technology, ISO/IEC JTC1 SC27/WG2 makes many standards in terms of underlying cryptographic mechanisms. For the verification of cryptographic protocols, ISO/IEC 29128 is the standard to verify and evaluate the level of its security.

To cover the security of hardware/software implementation, ISO/IEC 15408 is the standard to evaluate and certify each product. ISO/IEC 15408 and ISO/IEC 29128 define the levels of certification from a loose to a rigorous level. Each nation has its product certification program which aligns with the ISO/IEC 15408 framework, then certifies each product for use in the nation.

In the records space, ISO/IEC 15489—*Information and Documentation—Records Management* is the predominant standard, while ISO/IEC 30300—*Information and Documentation—Management Systems for Records* provides additional requirements for recordkeeping and ISO/IEC 14721—*Space Data and Information Transfer Systems—Open Archival Information System (OAIS)—Reference Model* provide the basis for long-term preservation of records.

The ISO/IEC 27000 series is well-known as the Information Security Management System (ISMS), in securing operations and lifecycles of IT systems. ISO standards are generally referred to when the government designs any system. This is mandated by the World Trade Organization/Technical Barriers to Trade agreement.

The above is the general and existing regulatory and standard framework in term of security of IT systems. Unfortunately, at the time of writing this book, most of the blockchain implementations and blockchain-based IT systems do not comply with these standards and frameworks. The standards and frameworks are not well-known to young open-source engineers, and the fact that these standards were developed for centralized systems makes them difficult to apply to decentralized systems such as blockchains. Being compatible with these frameworks also requires large budgets that small start-ups cannot cover. However, such standards and frameworks are essential to securing blockchain-based systems and making them transparent to consumers and the government. Standard structures and operations are required for securing blockchain-based IT systems.

4.3.3 Trade-off Between Security and Usability

User experience can be the deciding factor between the success or failure of systems, and balancing security in a way that does not restrict a user's ability to interact with a system is critical. Absolute security and usability are unattainable; the focus should therefore be on a system providing adequate levels of both, given the respective constraints. Past experiences and impressions influence a user's decisions and the more expected and normal a situation appears to be, the more it is trusted. Such *situational normality* is however difficult to attain, particularly for blockchains, where users are constantly confronted with new use cases and terminology. The resulting technology, while innovative, is often hard to use, commonly leading to challenges and errors.

The limited number of user studies in the domain focus on digital currencies and suggest that users appear to be facing hard to overcome usability barriers. It was shown that users of Bitcoin do not necessarily understand the technology, in particular when it comes to privacy implications and the underlying cryptography (Gao et al. 2016). Certainly, one might argue that a user does not necessarily need to understand how the technology works in order to be able to use it, but given the evidence of monetary losses due to self-induced errors (Krombholz et al. 2016; Voskobojnikov et al. 2020) the importance of intuitive software becomes apparent. While these findings are not generalizable to the whole domain, it appears that there is an underlying concern of inadequate mental models, meaning that the user's interpretation of the external reality might lead to dangerous behavior. For example, wallet files might be deleted by unsuspecting users, possibly revoking their access and making the system unusable. Given the wide range of available software, it becomes extremely difficult to define what *usable* actually means in the context of blockchain; thus, we need to look at usability in the general context prior to developing guidelines for the blockchain domain.

Traditionally, usability is defined as the extent to which a user can achieve their goals effectively, but depending on the application such goals can be wide-ranging. Trade-offs between security and usability are therefore contextual and need to be made individually, on a case-by-case basis. Computer security is rarely offered as an option in consumer applications; it is more so a system property that the respective user is not necessarily aware of. Security has to be practically invisible to prevent impediment of workflow efficiency. Notifications, warnings, and options therefore should only be displayed in case of significant risks that the user is exposed to at that point in time, e.g., when making irreversible transactions. The fewer security-critical decisions a user is offered, the fewer potential errors can be made. Here, prioritizing intuitiveness can help in ensuring that existing users do not have to re-learn how to interact with a system and newcomers do not face high entry barriers. Innovative technology can only go so far without usable interfaces and both are equally important in facilitating mass adoption.

Ease of use, or the lack thereof, not only negatively influences existing users but also newcomers who, while eager to learn the technology, are often overwhelmed

during the onboarding process. This is particularly interesting for technology adoption where ease of use, among others, has been identified as an influencing factor (Abramova and Böhme 2016). It therefore raises the question of how to design software systems that are perceived as usable by both users and non-users. Here, the subjectively perceived situational normality can be a deciding factor between technology acceptance and rejection on the user's end and naturally, a system designer's goal should be software that is in accordance with past experiences of the respective user. An interesting example highlighting this was an investigation of the unbanked population in Mexico (Larios-Hernández and Ortiz-de-Zarate-Béjar 2019). Besides the lack of trust in institutions, it is argued that participants were accustomed to informal, face-to-face transactions and that blockchain technology could provide an alternative, but only if it would adhere to existing social norms.

It is clearly infeasible to investigate all possible user groups when designing a system; however, taking existing software solutions that already address a given goal as a benchmark can be of help. For instance, cryptocurrency wallets should resemble conventional online banking software and similar analogies can be found for distributed file systems, supply management, and others. For consumer applications technological features should *never* be the main selling point, as the vast majority of users simply would not be able to process such information. User satisfaction hinges on usable interfaces that allow the completion of tasks, and not on the number of buzzwords used in the pitch. Less might therefore be more when it comes to paving the way towards adoption, independent of application area and use case.

4.3.4 Decentralizing Responsibility for Data Security

Blockchain holds significant promise to enhance data security through decentralization. However, as discussed above, there exist inherent trade-offs between security and usability of blockchain technologies for individual users. Given that public blockchain protocols are still in a relatively early stage of emergence, excessive decentralization of responsibility for security could serve to hamper user adoption. Although users could enjoy enhanced security by taking direct control over their personal data, many may actually prefer centralized third parties to hold custodianship of their data and access to this through custodial wallets. By consequence, one of the core benefits of blockchain technologies—enhanced user privacy and security—may be left unrealized, and user adoption in a partially decentralized system could conversely present new security challenges.

In this section, we expand on the security-usability trade-off, and explore how this might shift in the public perception over time. Specifically, we question whether and how users might begin to take security more seriously, and even begin to sacrifice usability and convenience for this. To do so, we situate the emergence of blockchain within a broader trajectory of information governance and cultural awareness, and explore the roles of users, corporations, and hackers in this trajectory. From this, we argue that initially compromising security may enable adoption in the

short term; however, creating decentralized solutions may subsequently improve the literacy and practices of these users in the longer term.

To begin, we can simply conceptualize the emergence of blockchain technology along Rogers' (2003) technology adoption curve that depicts the diffusion of innovations. Blockchain-based assets and applications have diffused amongst the "innovators" (the initial 2.5%) and the "early adopters" (the next 13.5%). However, it can take significantly longer for innovations to diffuse amongst the next group, the "early majority", due to this group's different expectations around usability and limited technical literacy. To "cross the chasm" (Moore 2014) between early adopters and the early majority requires innovators to cater to mainstream end users by smoothing the behavioral shifts necessary for adoption.

While this model may be rudimentary, it points to an important qualitative consideration in the diffusion of technologies from tech-savvy users to mainstream adopters: the need to ground innovations in existing understandings. To understand new technologies we tend to lean on comparisons with established products and technologies (Hargadon and Douglas 2001; Navis and Glynn 2010), recruiting metaphor and analogy (Cornelissen and Clarke 2010; Überbacher et al. 2015), and often convey these through narrative and storytelling (Lounsbury and Glynn 2001; Martens et al. 2007; Navis and Glynn 2011; Rosa et al. 1999). Moreover, this is rarely one-directional from innovator to end user, but rather involves dialogue and iteration, in which both groups (along with others such as media organizations) interact and co-create new meanings over time (Navis and Glynn 2010; Rosa et al. 1999).

Applying this to the decentralization of personal data security, we can begin to anticipate challenges by comparing this vision to the existing systems. Over the past two decades we have witnessed the institutionalization of an arrangement in which corporations (such as digital platforms) collect, store, and render their users' personal data. This unique control over user data has become a core part of the business models of many digital platforms (Boudreau and Hagiu 2009; Constantinides et al. 2018). User data and metadata can have "generative" properties for organizations (Tilson et al. 2010; Yoo et al. 2012), meaning that data can be analyzed, aggregated and rendered in ways that help them to improve user experience, create switching costs, and even manipulate user behavior. For instance, in 2014 Facebook purposively filtered user news feeds in a successful attempt to manipulate users' emotional states (Panger 2016). Although this arrangement has been labelled exploitative by some (e.g. Malin and Chandler 2017; Rey 2012; Rogers 2016), individual users arguably benefit by not having to concern themselves with data security. Users can rely upon these organizations when they forget their passwords or accidentally delete their data.

Against this backdrop of an arrangement where organizations assume virtually full responsibility for users' personal data, which remains taken-for-granted and largely unquestioned by users, the complete decentralization of responsibility for data security seems an ominous endeavor. Full decentralization of responsibility is essentially the opposite end of the spectrum to current arrangements. Perhaps then, for blockchain-based assets and applications to break into the mainstream, the early

majority may prefer for some dimensions of data security to remain centralized, so that these do not deviate too heavily from existing arrangements.

Such hybrid arrangements appear to be the preferred approach for profit-seeking organizations seeking to promote widespread adoption, for example in the provision of custodial wallets by Facebook (via its Calibra Wallet) in cryptocurrencies and Dapper Labs in online gaming. Especially incumbent organizations may be prone to centralizing aspects of data ownership, custodianship, and/or access as they cling to their existing business models (Barr et al. 1992). However, although centralizing aspects of data custodianship and access may help to forge a path of least resistance to mainstream adoption, such arrangements serve to weaken the security of the system and put individuals at risk. We have witnessed this already in the blockchain space, through the numerous high-profile attacks on cryptocurrency exchanges.

So how could this tension be addressed? Are we destined to continually sacrifice security for usability in ways that blunt one of the cornerstone advantages of distributed ledger technologies? Again, we may be able to shed light on this question by examining the broader trajectory of personal data and its governance. The discourses around how data is collected, stored, and accessed, appear to be shifting in recent years. Whereas in the early- to mid-2000s, the general public was largely unaware that organizations were collecting extensive data about them (let alone that there existed security concerns around this), frequent, high-profile security breaches in recent years have brought data security into the public consciousness. Continued hacks of centralized organizations such as Facebook, Uber, and Equifax, have brought data privacy and security into the forefront of public attention. Those people with a user profile on Ashley Maddison (an online matchmaking service to facilitate extramarital affairs) when it was hacked in July 2017 may understand the importance of data security more than most. In July 2019, Facebook was fined $5 billion US dollars by the Federal Trade Commission over repeated privacy violations, including the Cambridge Analytica scandal in 2016. Together, these high-profile security breaches are bringing data security into focus for the mainstream user.

Bringing these points together, we posit a possible path for the decentralization of responsibility for data security to individual users. In the near-term, centralized organizations might accelerate the adoption process by smoothing the transition for individual users through their hybrid model. This helps users become used to some parts of the new technology whilst masking others. At the same time, however, they establish themselves as “honeypots” for hacks, in the same way that cryptocurrency exchanges have in the past 10 years. Therefore, hacks may actually push people off these custodial wallets and enable them to take full control over their personal data. In short, and counterintuitively, corporations that centralize some parts of data governance may actually be important stepping stones to reach full decentralization.

To conclude, we propose that the history of centralized data ownership and control must be taken into account when projecting the diffusion of blockchain technology and the decentralization of responsibility for data security. However, counterintuitively, the very organizations that have helped to mold existing arrangements may be integral to their replacement, since centralizing data security carries

inherent disadvantages and presents them as targets for attack. Over time, we may see a continued trajectory where high-profile security breaches of centralized organizations continue to raise public awareness and literacy around data security in ways that encourage further decentralization.

4.3.5 More Complexity Means Less Security

Complexity—and in this case we mean system complexity—is hard to handle for humans. We experience this every day and it is not only true for the blockchain environment. Let us take a look at several examples where we experience complexity in blockchains and let us use the above-defined layers, or dimensions, to tie the discussion together.

4.3.5.1 Social Layer/Dimension

For the vast majority of the population blockchain technology is a closed book. Even for some people dealing with cryptocurrencies like Bitcoin the underlying technology is not comprehensible.

Wallets are a good example of this. They store the private keys which are needed for accessing blockchain addresses. This is required because the handling of a 64-character key is too complex for humans to remember. Therefore, we use wallets or QR codes to reduce this complexity, but this approach introduces several security issues. What if the wallet application sends the private key to some third person? What if the QR code does not represent the intended address? What if another application pretends to be the wallet app and steals the password of the proper wallet?

Another example is trust relations. Blockchain is said to allow trust even if the counterparty is not known. This is because one can trust the immutability of transactions, the identity behind an address, and the transparency of entries to name a few. But how do we verify this? Do we vet the number of miners and understand their relationship enough to exclude a 51% attack?

And finally, in a legal sense, the complexity of our legal system in combination with a complex technology like blockchain has reached such a high degree that legal compliance for many use cases cannot be guaranteed. Even lawyers are often overwhelmed and have to wait for court decisions to be on the safe side of things.

4.3.5.2 Data Records Layer/Dimension

If we take a look at the complexity-security relation in the data records layer, or dimension, we can observe that trustworthiness of data is a complex issue, too. A good illustration of this issue can be found in the context of IoT devices.

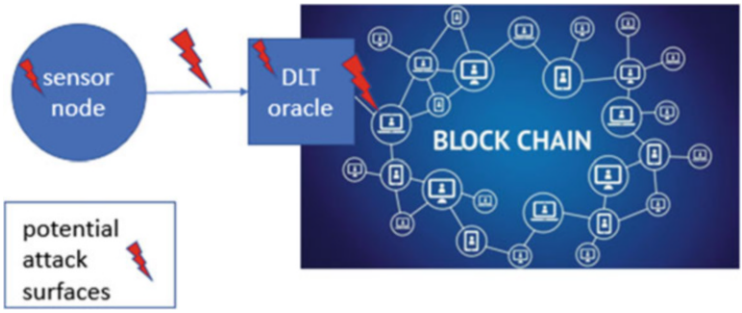

Fig. 4.3 Attack surfaces of sensor connections and DLT-oracles for blockchains

IoT suffers from multiple different definitions. In a very loose and wide definition provided by Farooq et al. (2019), which is itself derived from Singh et al. (2014), the IoT, "provides internet-based services that involve human-to-thing, thing-to-thing, and thing-to-things communications." Therefore, things are involved which are usually considered as sensors or sensor networks. Other terms such as smart fog or smart dust in contrast to high-performance cloud-systems are used (Skwarek et al. 2016).

A basic property of such IoT-devices in terms of sensors, sensor systems, or sensor networks is their simplicity, which is intended to achieve:

- low size;
- low power-consumption;
- low cost; and
- long operating times.

These main design parameters are mostly achieved with low-performance micro-controllers. At the same time, such devices are used as real-world-data sources (=sensors) for trusted systems such as blockchains. This arrangement creates a potential for data trustworthiness issues.

In common setups, the sensors—or IoT-devices—are connected as singular devices via access-points (=gateways) to the blockchain as a data-source (=DLT-oracle) (see Fig. 4.3), sending their data via an (un)trusted software into the blockchain. Given the fact that the sensor data itself must be considered as correct, because this is the defined trust-anchor and will not be questioned (unwisely), the communication channel of the potentially wireless sensor and the operation of the DLT-oracle software might be subject to an attack.

The scenario gets even more complicated when a multi-sensor-network is attached to the DLT-oracle, e.g., for monitoring goods during transport. Many sensors may be distributed among the load sending all their "trustful" data to a wireless gateway, also working as a DLT-oracle server into a blockchain. In this scenario a blockchain is required as the data is required for later inspection (e.g., as evidence of how the shipment was handled). To assure trusted communication, the communication channel is usually encrypted.

The required long battery operating times of the sensors are usually achieved by low power consumption—a combination of low-performing processor capabilities and sleep intervals. Especially the sleep intervals are security-critical regarding the "A" (i.e., availability) of the CIAAA principle. During this time, an attacker is able to execute multiple attack-schemes:

- **Man-in-the-middle-replay**: The attacker can replay earlier recorded data on behalf of a suspended sensor, even if the channel is encoded, as the encryption scheme or -key may not have changed.
- **Sybil-attack**: The attacker could become a silent listener to the decoded channel and generate its own data under the identity of the suspended sensor. In real idle-phases, the suspended sensor may not even realize that its identity has been captured and abused by an attacker.

Many attacks can be listed and considered as sensor nodes and networks are not capable of complex operation modes due to their simplistic design principles. To illustrate in terms of the CIA-triad:

- **Confidentiality**: Complex channel encryption requires too much computational effort and energy at the expense of the operating time, therefore most channels in wireless sensor communication are not highly secured.
- **Integrity**: An integrity check can principally be hash-or checksum-based. But as the checksum algorithm has to be known in order for the check to be performed, the value can easily be generated by an attacker. Consequently, using single sensor values, the integrity has simply to be taken as a trust anchor.
- **Availability**: As already discussed, unavailability is a part of the design-principles of an IoT-network, which opens the door to various types of attacks.

These security weaknesses are not insurmountable, however, as discussed in detail in Box 4.1.

Box 4.1 Securing IoT Sensor Communications in the Context of Blockchains Using the Sensorchain Concept

A higher level of security can be achieved by the application of methods of Byzantine fault tolerance, such as those used in many distributed ledgers. This enables the sensors to become reliable and a trusted part of a network by repeating longer time series of messages and performing redundant checks, whether such messages have been received on redundant communication paths or not. As an example see Fig. 4.4. Such communication principles are known from systems such as Hashgraph (Baird 2016), Iota (Popov 2016) or sensorchain (Skwarek 2017) and evolve from research into practical applications. Short snipplets or time series of past sent and received communication are repeated in a new message to show other participants that the IoT-node knows about some history of communication.

(continued)

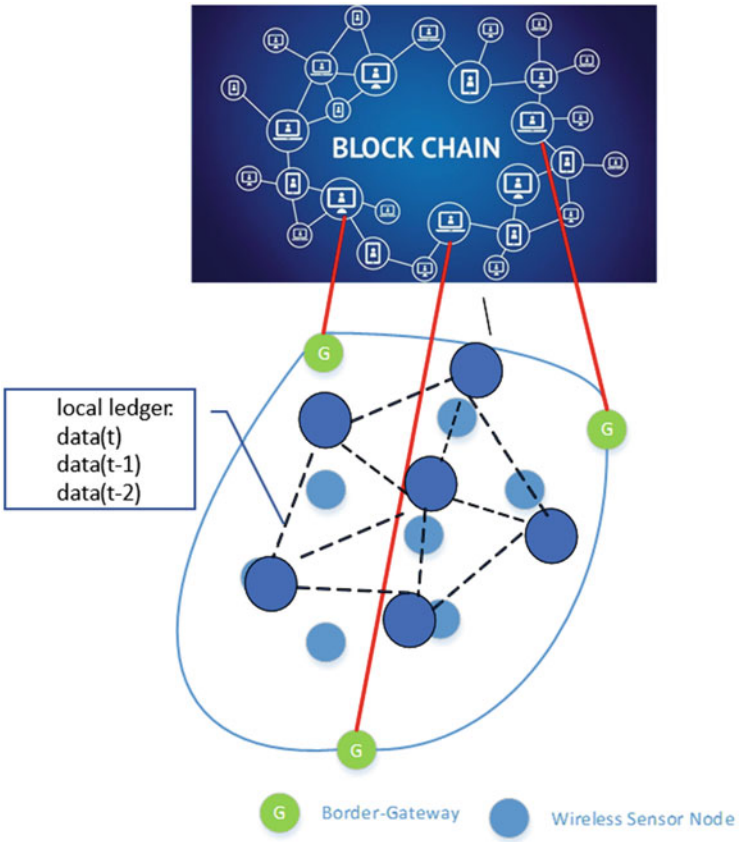

Fig. 4.4 The sensorchain concept

Box 4.1 (continued)

Other participants receiving these BFT-messages are now able to decide about the correctness of the messages in terms of integrity. Moreover, missing availability can be bridged by this method. Although idle-intervals may still lead to data gaps in the time series of BFT-messages, similarities to earlier communication epochs can be detected by other nodes allowing for a means of determining if missing data due to unavailability of a node are plausible and trustworthy.

If the identity of a node is not only generated by some static ID-number, but is generated according to the location of a node such as described by Bornholdt et al. (2019), the identity can also be verified independently by other nodes.

(continued)

Box 4.1 (continued)
Therefore the man-in-the-middle or sybil-attacks can be detected and countered by the network or by the gateway.

Consequently decentralized communication of IoT-systems can be secured by methods of DLT-like BFT protocols.

The anonymity of a blockchain is achieved by avoiding usernames and instead using addresses. The cryptic and often randomly generated address, in combination with using addresses only once, gives us the apparent safety of anonymity. But this pseudonymity only lasts as long as no one can make the link between an address and a user. Once this link is established, former transactions can be viewed due to the immutability of the blockchain. In this case the complexity of the address makes us believe in its security. The desire for data anonymity, moreover, may be in direct conflict with the need we have in some cases (e.g., transfer of property rights) to establish the identity (legal or at least social) of a transacting party to establish the authenticity of records.

One argument for using a blockchain is transparency. In permissionless blockchains everybody can have a look at the data. Therefore, it is said to be transparent. But have you ever had a look at this data? Did you understand the semantics? The maximum degree of transparency we usually check are some webpages showing the transactions and even this is not comprehensible to ordinary citizens. Can we handle this kind of transparency or is it already too complex?

4.3.5.3 Technological Layer/Dimension

On the technological layer or dimension, the complexity of blockchain platforms rises with the use of smart contracts. Those self-executing programs allow control over data and assets. The more complex they become, the more likely an error might be included. Since smart contracts cannot be altered once deployed to the blockchain any error might result in the loss of assets.

Finally, the blockchain platforms themselves are highly complex systems which are understood only by a few. Errors in the code of such platforms cannot be fixed like in ordinary computer applications since they are distributed over many nodes. Each node has to agree on an update and perform this update in the same time period to ensure the functionality of the blockchain. Disagreements over changes result in a fork of the blockchain. In this case the security and longevity of the data (and associated records) cannot be guaranteed anymore.

How should we deal with this dilemma? Can blockchains as a complex system be saved at all? The answer may be in nature. As an adaptable highly complex multi-agent system, nature deals with complexity in an excellent manner. Techniques applied are:

- Distribution and redundancy, which we already use in blockchain technology;
- Adaptation to change, which some blockchain platforms like Tezos are attempting to implement;
- Limited life expectancy;
- Resilience or the ability not to fail completely in the event of disturbances or partial failures, but to maintain essential services; and/or
- No stasis, but evolution and survival of the fittest.

Will we find a way to make our blockchain systems robust and secure even though they may become more and more complex? And can nature be a role model for this?

4.4 The Constant

As discussed throughout the previous sections of this chapter, security in the context of blockchain-based and distributed ledger technologies cannot be generalized. However, despite the wide range of application areas and influencing factors, two themes can be observed that can serve as guidance for security in the context of this rapidly evolving technology. Here, *agile security* becomes an integral part of the technology, particularly when considering how costly changes are. Further, the role of the alleged *weakest link*, i.e., the end users, and their influence on a blockchain ecosystem. Certainly, the two themes are not unique to the blockchain space, however, both become increasingly important due to the inherent properties of the technology, such as immutability and decentralization.

4.4.1 Designing for the Future

Only few could have guessed the rapid development of the blockchain domain since Bitcoin's inception in 2009. With thousands of cryptocurrencies and tokens, various sectors making use of the technology and millions of users worldwide, there appears to be great potential in this technology. However, making predictions of how the future might unfold and how the domain might look in 10 years' time is not the goal of this chapter. Here, the focus is on security considerations that can help in designing solutions flexible enough to meet the requirements of the future.

Traditionally, a security goal refers to an asset and defines the security objective, i.e., what attribute of the asset is at stake and needs to be protected. While we cannot predict what assets and stakeholders might emerge over time, the definition of what confidentiality, integrity, and availability mean will likely remain relatively stable. The means with which these attributes (i.e., the CIA-triad) can be protected will change of course, with quantum cryptography being one challenge in the near future. Rather than tackling security challenges as they come, we argue for pre-emptively

creating software that is expandable enough to mitigate issues as they arise in the first place. For public blockchains, such updates come with great cost due to the large number of participating nodes. In this case, updates can only take place once a consensus is reached, and past disagreements resulting in hard forks have shown how difficult this might be. Changes can have grave implications and have therefore to be considered carefully. While disagreements cannot be prevented, update protocols and transparent decision making can help in addressing some of the concerns that have arisen in the case of the biggest hard forks in the past, namely Ethereum and Bitcoin.

For permissioned blockchains however, the argument for expandability is easier to make due to the limited number of participating nodes and their willingness to cooperate. Still, clear governing guidelines are needed to reach consensus and must be created from the very beginning to avoid disagreements later on.

Overall, it appears that in a rapidly evolving domain there is simply no place for stagnant computer security. Risks are changing and so are security requirements. *Agile security* should therefore be a core element of solution design and not merely an afterthought, as is often the case.

4.4.2 The Weakest Link

The human factor is often labeled as the weakest cybersecurity link in both academia and industry but is this assessment truly fair? We argue that the end users are doing their best to adapt to rapidly evolving technology and might simply fall short while doing so. Lowering the cognitive load that users endure during interaction should therefore be the primary goal of solution designers and architects.

The first users of blockchain technology were the select few on Bitcoin forums in 2009. Since then, the user base has grown but has the technology and the user experience also changed? While hundreds of new wallets were developed, the resemblance to the original Satoshi client is clearly there. Users still have to deal with public key cryptography, key management, and confusing terms in interfaces that have already existed more than 10 years ago. While cypherpunks in 2009 were more or less comfortable with this software, it is doubtful that the common computer user nowadays will be as well. Several studies suggest that public key cryptography is hard to use (Whitten and Tygar 1999; Sheng et al. 2006; Ruoti et al. 2015) and this is not surprising. Back in 1999, the average user was struggling with public key cryptography and evidently this is still the case, with the user still being unable to use Bitcoin wallets (Eskandari et al. 2015). Cryptography is hard to grasp and expecting end users to adjust to the technology that is being thrown at them is unjust. While several improvements have been made to enhance user experience, users continue to struggle. High barriers to entry and switching costs are a hindrance to adoption and the technological advancements of blockchains will not matter as long as such barriers exist. If there is no user base, what value is there in a groundbreaking technology? A product becomes successful through its users and user experience.

Great user experience has always been one of the deciding factors in technology adoption, with leading examples from Apple or Facebook. It is hard to disrupt existing technologies when the user experience is lacking. The average user will not choose a platform because of its technological features: a platform will be chosen if it addresses a need without putting exceptional amounts of cognitive load on the respective end user. A technology has to be both *usable* and *useful* to be adopted in the long run and both of these attributes clearly rely on the perceptions of users.

Innovations are adopted over time and while early adopters might already be on board, the early majority is not there yet. Designers should take existing solutions as a benchmark when thinking about interfaces for blockchain technology. If possible, the users should not even be aware of the underlying technology. Usable security research in the space is in its infancy and deserves more attention as technological innovation alone can only partially pave the way towards mass adoption.

References

Abramova, S., & Böhme, R. (2016). Perceived benefit and risk as multidimensional determinants of Bitcoin use: A quantitative exploratory study. In *Proceedings of the 37th International Conference on Information Systems (ICIS 2016)* (pp. 233–252). Atlanta, GA: Association for Information Systems. https://aisel.aisnet.org/icis2016/Crowdsourcing/Presentations/19/

Atzei, N., Bartoletti, M., & Cimoli, T. (2017). A survey of attacks on Ethereum smart contracts (SoK). In M. Maffei & M. Ryan (Eds.), *POST 2017: Principles of security and trust* (Lecture notes in computer science) (Vol. 10204, pp. 164–186). Berlin: Springer. https://doi.org/10.1007/978-3-662-54455-6.

Baird, L. (2016). The Swirlds hashgraph consensus algorithm: Fair, fast, byzantine fault tolerance: SWIRLDS-TR-2016-01. Retrieved from https://www.swirlds.com/downloads/SWIRLDS-TR-2016-01.pdf

Barr, P. S., Stimpert, J. L., & Huff, A. S. (1992). Cognitive change, strategic action, and organizational renewal. *Strategic Management Journal, 13*(S1), 15–36. https://doi.org/10.1002/smj.4250131004.

Batista, D., & Lemieux, V. (2019). Bounded and shielded: Assessing security aspects and trustworthiness of smart contracts. In *Proceedings of the Annual Conference of the Canadian Association for Information Science (CAIS), University of Alberta Libraries, AB, June 4, 2019*. Retrieved from https://journals.library.ualberta.ca/ojs.cais-acsi.ca/index.php/cais-asci/

Beldad, A., de Jong, M., & Steehouder, M. (2010). How shall I trust the faceless and the intangible? A literature review on the antecedents of online trust. *Computers in Human Behavior, 26*(5), 857–869. https://doi.org/10.1016/j.chb.2010.03.013.

Bitcoin Core 0.11 (ch 4): P2P network. (2018). In *Bitcoin Wiki*. Retrieved August 27, 2019, from https://en.bitcoin.it/wiki/Bitcoin_Core_0.11_(ch_4):_P2P_Network#Peer_discovery

Blockchain Transparency Institute. (2018). *December 2018: Exchange volumes report*. Originally retrieved February 19, 2019, from https://www.blockchaintransparency.org/december-2018-rankings (now renamed Market Surveillance Report – December 2018 and available at https://www.bti.live/reports-december2018/)

Bornholdt, L., Reher, J. & Skwarek, V. (2019). Proof-of-location: A method for securing sensor-data-communication in a Byzantine fault tolerant way. In *Mobile communication – Technologies and applications; 24. ITG-Symposium* (pp. 1–6). IEEE. Retrieved from https://ieeexplore.ieee.org/document/8731780

Boudreau, K. J., & Hagiu, A. (2009). Platforms rules: Multi-sided platforms as regulators. In A. Gawer (Ed.), *Platforms, markets and innovation* (pp. 163–191). Cheltenham: Edward Elgar

Braden, R. (Ed.). (1989). *Requirement for internet hosts – Communication layers*. Internet Engineering Task Force: Network Working Group: RFC 1122. Retrieved August 23, 2019, from https://tools.ietf.org/pdf/rfc1122.pdf

Chen, L. (2018, November 15). Peer discovery in Harmony network. *Medium*. Retrieved August 27, 2019, from https://medium.com/harmony-one/peer-discovery-in-harmony-network-6a07f9401c61

Collomosse, J., Bui, T., Brown, A., Sheridan, J., Green, A., Bell, M., Fawcett, J., & Thereaux, O. (2018). ARCHANGEL: Trusted archives of digital public documents. In *DocEng '18: Proceedings of the ACM Symposium on Document Engineering 2018* (Article 31, pp. 1–4). New York: Association for Computing Machinery. https://doi.org/10.1145/3209280.3229120.

Constantinides, P., Henfridsson, O., & Parker, G. G. (2018). Platforms and infrastructures in the digital age. *Information Systems Research, 29*(2), 381–400. https://doi.org/10.1287/isre.2018.0794.

Cornelissen, J. P., & Clarke, J. S. (2010). Imagining and rationalizing opportunities: Inductive reasoning and the creation and justification of new ventures. *The Academy of Management Review, 35*(4), 539–557. https://doi.org/10.5465/amr.35.4.zok539.

Duranti, L. (1998). *Diplomatics: New uses for an old science*. Lanham, MD: Scarecrow Press

Er-Rajy, L., El Kiram, M.A., El Ghazouani, M., & Achbarou, O. (2017). Blockchain: Bitcoin wallet cryptography security, challenges and countermeasures. *Journal of Internet Banking and Commerce, 22*(3), 294. Retrieved August 24, 2019, from http://www.icommercecentral.com/peer-reviewed/blockchain-bitcoin-wallet-cryptography-security-challenges-and-countermeasures-86561.html

Eskandari, S., Barrera, D., Stobert, E., & Clark, J. (2015). *A first look at the usability of Bitcoin key management*, presented at USEC '15, San Diego, CA, February 8, 2015. https://arxiv.org/abs/1802.04351

Farooq, U., Ul Hasan, N., Baig, I., & Shelzad, N. (2019). Efficient adaptive framework for securing the Internet of Things devices. *EURASIP Journal on Wireless Communications and Networking, 2019*, 210. https://doi.org/10.1186/s13638-019-1531-0.

Finney, H. (2011). *The Finney attack (the Bitcoin Talk forum), 2011*. Retrieved April 7, 2020, from https://bitcointalk.org/index.php?topic=3441.msg48384

Gao, X., Clark, G.D., & Lindqvist, J. (2016). Of two minds, multiple addresses, and one ledger: Characterizing opinions, knowledge, and perceptions of Bitcoin across users and non-users. In: *CHI'16: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (pp. 1656–1668). New York: Association for Computing Machinery (ACM). http://doi.acm.org/10.1145/2858036.2858049.

General Data Protection Regulation 2016/679 Article 17: Right to erasure ('right to be forgotten') (EU). Retrieved November 23, 2019, from https://gdpr-info.eu/art-17-gdpr/

Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. In *STOC '96: Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing* (pp. 212–219). New York: Association for Computing Machinery (ACM). https://doi.org/10.1145/237814.237866.

Hargadon, A. B., & Douglas, Y. (2001). When innovations meet institutions: Edison and the design of the electric light. *Administrative Science Quarterly, 46*(3), 476–501. https://www.jstor.org/stable/3094872/

Heilman E., Kendler A., Zohar A., & Goldberg S. (2015). Eclipse attacks on Bitcoin's peer-to-peer network. In J. Jung (Ed.), *Proceedings of the 24th USENIX Conference on Security Symposium (SEC'15)* (pp. 129–144). Berkeley, CA: USENIX Association. https://dl.acm.org/doi/10.5555/2831143.2831152.

iFour Technolab Pvt. Ltd. (2019, April 11). *Blockchain and architecture* [Blog post]. Retrieved August 24, 2019, from https://www.ifourtechnolab.com/blog/blockchain-history-and-evolution

International Organization for Standardization (ISO). (2012). *Information technology—security techniques—guidelines for cybersecurity* (ISO/IEC 27032:2012)

Javeri, P. (2019). Blockchain architecture. *Medium*. Retrieved August 28, 2019, from https://medium.com/@prashunjaveri/blockchain-architecture-3f9f1c6dac5e

Krombholz, K., Judmayer, A., Gusenbauer, M., & Weippl, E. (2016). The other side of the coin: User experiences with Bitcoin security and privacy. In J. Grossklags & B. Preneel (Eds.), *Financial cryptography and data security* (pp. 555–580). Berlin: Springer. https://doi.org/10.1007/978-3-662-54970-4_33.

Larios-Hernández, G. J., & Ortiz-de-Zarate-Béjar, A. (2019). Blockchain entrepreneurship and the struggle for trust among the unbanked. In H. Treilblmaier & R. Beck (Eds.), *Business transformation through blockchain: Vol. II* (pp. 259–283). Cham: Springer International. https://doi.org/10.1007/978-3-319-99058-3_10.

Lee, J. (2016). What the right to be forgotten means to companies: Threat or opportunity? *Procedia Computer Science, 91*, 542–546. https://doi.org/10.1016/j.procs.2016.07.138.

Leffew, K. (2019). A brief overview of Kademlia, and its use in various decentralized platforms. *Medium*. Retrieved August 27, 2019, from https://medium.com/coinmonks/a-brief-overview-of-kademlia-and-its-use-in-various-decentralized-platforms-da08a7f72b8f

Lemieux, V. L., & Sporny, M. (2017, April). Preserving the archival bond in distributed ledgers: A data model and syntax. In *Proceedings of the 26th International Conference on World Wide Web Companion* (pp. 1437–1443). https://doi.org/10.1145/3041021.3053896.

Lemieux, V. L., Hofman, D., Batista, D., & Joo, A. (2019). *Blockchain technology and recordkeeping* (Report prepared for the ARMA International Education Foundation). Retrieved from http://armaedfoundation.org/wp-content/uploads/2019/06/AIEF-Research-Paper-Blockchain-Technology-Recordkeeping.pdf

Li, X., Jiang, P., Chen, T., Luo, X., & Wen, Q. (2018). A survey on the security of blockchain systems (arXiv:1802.06993). *arXiv.org*. Retrieved August 24, 2019, from http://arxiv.org/abs/1802.06993

Lounsbury, M., & Glynn, M. A. (2001). Cultural entrepreneurship: Stories, legitimacy, and the acquisition of resources. *Strategic Management Journal, 22*(6–7), 545–564. https://onlinelibrary.wiley.com/doi/10.1002/smj.188

Malin, B. J., & Chandler, C. (2017). Free to work anxiously: Splintering precarity among drivers for Uber and Lyft. *Communication, Culture & Critique, 10*(2), 382–400. https://onlinelibrary.wiley.com/doi/abs/10.1111/cccr.12157

Martens, M. L., Jennings, J. E., & Jennings, P. D. (2007). Do the stories they tell get them the legitimacy they need? The role of entrepreneurial narratives in resource acquisition. *The Academy of Management Journal, 50*(5), 1107–1132. Retrieved from https://www.jstor.org/stable/20159915

Mayer, H. (2016). ECDSA security in Bitcoin and Ethereum: A research survey. *CoinFabrik*. Retrieved from https://blog.coinfabrik.com/ecdsa-security-in-bitcoin-and-ethereum-a-research-survey/

Maymounkov, P., & Mazières, D. (2002). Kademlia: A peer-to-peer information system based on the XOR metric. In P. Druschel, F. Kaashoek, & A. Rowstron (Eds.), *Peer-to-peer systems: First international workshop, IPTPS 2002, Cambridge, MA, USA, March 7–8, 2002. Revised papers. Lecture notes in computer science* (Vol. 2429). Berlin: Springer. https://doi.org/10.1007/3-540-45748-8_5.

Moore, G. A. (2014). *Crossing the chasm: Marketing and selling disruptive products to mainstream customers* (3rd ed.). New York: HarperCollins

Nakamoto, S. (2008). *Bitcoin: A peer-to-peer electronic cash system*. Satoshi Nakamoto Institute. Retrieved from https://nakamotoinstitute.org/bitcoin/

Navis, C., & Glynn, M. A. (2010). How new market categories emerge: Temporal dynamics of legitimacy, identity, and entrepreneurship in satellite radio, 1990–2005. *Administrative Science Quarterly, 55*(3), 439–471. https://doi.org/10.2189/asqu.2010.55.3.439.

Navis, C., & Glynn, M. A. (2011). Legitimate distinctiveness and the entrepreneurial identity: Influence on investor judgements of new venture plausibility. *The Academy of Management Review, 36*(3), 479–499. Retrieved from https://www.jstor.org/stable/41319182

Panger, G. (2016). Reassessing the Facebook experiment: Critical thinking about the validity of Big Data research. *Information, Communication & Society, 19*(8), 1108–1126. https://doi.org/10.1080/1369118X.2015.1093525.

Pearce-Moses, R. (ed.) (2018). *InterPARES trust terminology*. InterPARES Trust. Retrieved from https://interparestrust.org/terminology/

Pfleeger, C. P. & Pfleeger, S. L. (2002). *Security in computing* (3rd ed.). Prentice Hall Professional Technical Reference

Popov, S. (2016). The tangle whitepaper. *IOTA.org*. Originally retrieved May 25, 2017, from https://www.iotatoken.com/IOTA_Whitepaper.pdf. Now available at http://www.tangleblog.com/wp-content/uploads/2016/11/IOTA_Whitepaper.pdf

Rey, P. J. (2012). Alienation, exploitation, and social media. *American Behavioral Scientist, 56*(4), 399–420. https://doi.org/10.1177/0002764211429367.

Riegelsberger, J., Sasse, M. A., & McCarthy, J. D. (2005). The mechanics of trust: A framework for research and design. *International Journal of Human-Computer Studies, 62*(3), 381–422. https://doi.org/10.1016/j.ijhcs.2005.01.001.

Rogers, E. (2003). *The diffusion of innovations* (5th ed.). New York: The Free Press

Rogers, B. (2016). The social costs of Uber. *University of Chicago Law Review Online, 82*(1), 85–102. Retrieved from https://chicagounbound.uchicago.edu/uclrev_online/vol82/iss1/6

Rosa, J. A., Porac, J. F., Runser-Spanjol, J., & Saxon, M. S. (1999). Sociocognitive dynamics in a product market. *Journal of Marketing, 63*, 64–77. Retrieved from https://www.jstor.org/stable/1252102

Rosenfeld, M. (2011). Analysis of Bitcoin pooled mining reward systems (arXiv:1112.4980). *arXiv.org*. Retrieved from https://arxiv.org/abs/1112.4980

Ruoti, S., Andersen, J., Zappala, D., & Seamons, K. (2015). Why Johnny still, still can't encrypt: Evaluating the usability of a modern PGP client (arXiv:1510.08555). *arXiv.org*. Retrieved from https://arxiv.org/abs/1510.08555

Sheng, S., Broderick, L., Koranda, C. A., & Hyland, J. J. (2006). *Why Johnny still can't encrypt: Evaluating the usability of email encryption software*. Poster session presented at the meeting of SOUPS 2006: Symposium on Usable Privacy and Security, Pittsburgh, PA, July 12–14, 2006. Abstract. Retrieved from http://www.chariotsfire.com/pub/sheng-poster_abstract.pdf

Shor, P. W. (1994). Algorithms for quantum computation: Discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science* (pp. 124–134). IEEE. Retrieved from https://ieeexplore.ieee.org/document/365700

Singh, D., Tripathi, G., & Jara, A. J. (2014). A survey of Internet-of-Things: Future vision, architecture, challenges and services. In *2014 IEEE World Forum on Internet of Things (WF-IoT)* (pp. 287–292). Retrieved from https://ieeexplore.ieee.org/document/6803174

Skwarek, V. (2017). Blockchains as security-enabler for industrial IoT applications. *Asia Pacific Journal of Innovation and Entrepreneurship 11*(3), 301–311. https://doi.org/10.1108/APJIE-12-2017-035. Retrieved January 6, 2019, from http://www.emeraldinsight.com/doi/10.1108/APJIE-12-2017-035

Skwarek, V., Kistler, T., Rawer, M., & Schauer, S. (2016). IoT und sensornetzwerke: entwurf und programmierung von niedrigstenergiesystemen anhand einer metaarchitektur [IoT and sensor networks: Design and programming of lowest energy systems based on a meta-architecture]. In H. C. Mayr & M. Pinzger (Eds.), *Lecture Notes in Informatics (LNI), Proceedings – Series of the Gesellschaft fur Informatik (GI)P-259 – INFORMATIK 2016*, (pp. 1917–1925)

Stallings, W., & Brown, L. (2018). *Computer security: Principles and practice* (4th ed.). New York: Pearson

Tilson, D., Lyytinen, K., & Sørensen, C. (2010). Digital infrastructures: The missing IS research agenda. *Information Systems Research, 21*(4), 748–759. Retrieved from https://www.uio.no/studier/emner/matnat/ifi/INF5210/h14/pensumliste/articles/tilson-et-al-2010.pdf

Überbacher, F., Jacobs, C. D., & Cornelissen, J. P. (2015). How entrepreneurs become skilled cultural operators. *Organization Studies, 36*(7), 925–951. https://doi.org/10.1177/0170840615575190.

Voskobojnikov, A., Obada-Obieh, B., Huang, Y., & Beznosov, K. (2020, February). Surviving the cryptojungle: Perception and management of risk among North American cryptocurrency (non) users. In J. Bonneau J. & N. Heninger (Eds.) Financial cryptography and data security. FC 2020. Lecture notes in computer science (Vol 12059, pp. 595-614). Cham: Springer. https://doi.org/10.1007/978-3-030-51280-4_32.

Whitten, A., & Tygar, J. D. (1999). Why Johnny can't encrypt: A usability evaluation of PGP 5.0. In: *SSYM'99: Proceedings of the 8th Conference on USENIX Security Symposium* (Vol. 8, pp. 14–14). Berkeley, CA: USENIX. https://dl.acm.org/doi/abs/10.5555/1251421.1251435

Wu, J., & Tran, N. (2018). Application of blockchain technology in sustainable energy systems: An overview. *Sustainability, 10*(9), 3067. Retrieved August 24, 2019, from http://www.mdpi.com/2071-1050/10/9/3067

Yoo, Y., Boland, R. J., Lyytinen, K., & Majchrzak, A. (2012). Organizing for innovation in the digitized world. *Organization Science, 23*(5), 1398–1408. https://doi.org/10.1287/orsc.1120.0771.

Zimmermann, H. (1980). OSI reference model - the ISO model of architecture for open systems interconnection. *IEEE Transactions on Communications, 28*(4), 425–432. https://doi.org/10.1109/TCOM.1980.1094702.