Chapter 10
The Blockchain in IoT

Carlos Davila and Jacob Tarnow

10.1 Introduction

The role of centralized governance over networks and entities has allowed for the mass control of digital media and private life. As the Internet has evolved, researchers and developers have looked for new ways to distribute control and trust. Blockchain technology was first introduced in 2008 with the famous Bitcoin whitepaper by pseudonym Satoshi Nakamoto. Since then, we have seen a global wave of interest and investments into the world of cryptocurrencies and digital assets. While some are just trying to invest into cryptocurrencies, others believe more in the underlying technology behind it—blockchain.

Through the use of blockchain technology, one can decentralize an entire network—never relying on a central entity—and can place trust across all users instead of one central node. By distributing the data throughout the network, any one person or computer can contact their closest node to retrieve information residing on a common ledger.

Many expect that blockchain technology has the potential to transform a range of different industries. Because of this, blockchain is already being used and researched by many of the leading companies in technology. While many efforts are still in their infancy, and there are many challenges to solve, it is expected that blockchain has the power to propel significant transformations in the IoT sector.

Cisco estimates that there will be roughly 26 billion devices connected to the Internet by 2020. Server-client models will struggle to scale to such demand. Centralized models mean high maintenance costs for the manufacturer, and limited consumer trust in devices that are always connected to the Internet [3]. Blockchains facilitate the sharing of services and assets like never before. These types of possibilities have led companies like IBM, Cisco, and Intel to contribute to blockchain in IoT efforts.

A. Rayes, S. Salam, *Internet of Things From Hype to Reality*,
https://doi.org/10.1007/978-3-319-99516-8_10

There are countless digital currencies and innovative applications being developed on top of blockchain. The impact of these efforts will be hard to predict. In IoT, blockchains can facilitate things like M2M transactions, automated firmware updates, or even the tracking of food quality and control. Imagine cars automatically negotiating rates for parking spaces, or drones automatically reserving and paying for a landing pad. These are just a few possibilities, and in this chapter, we explore further how the blockchain can impact the IoT domain.

The chapter is organized in the following way. Section 10.2 defines the blockchain. We describe the difference between Bitcoin and blockchain and provide an overview of how blockchain has evolved over time. In Sect. 10.3, we dive into how blockchains work and review the features that make the technology important. Section 10.4 introduces how the blockchain may impact notable use cases in IoT and reviews the advantages and disadvantages of blockchain technology. Lastly in Sect. 10.5, we go over security considerations within blockchain and IoT.

10.2 What Is the Blockchain?

Before learning what a blockchain is, we should first understand why Bitcoin and the blockchain were introduced together in the original Bitcoin whitepaper. Bitcoin was presented as the *peer-to-peer electronic payment system*, and blockchain was the proposed *mechanism* that allowed it to work. A peer-to-peer digital currency needs a mechanism that allows its users to trust each other without the need for a central authority (like a bank). It is in the Bitcoin whitepaper that Satoshi Nakamoto proposes such a mechanism. More specifically, Nakamoto proposes the blockchain as the solution to the double-spending problem—how to tell if a user, or device, has spent the same digital coin more than once. Double spending is particularly hard to detect in a distributed system like Bitcoin, because there is no central authority tracking balances. This means that without a solution like the blockchain ledger, it is easy for a user to send the *same* coin to different users before anyone in the network learns of the fraudulent transactions. Blockchain is therefore what allows Bitcoin to be a trustless system and is the key innovation responsible for the success of Bitcoin and other cryptocurrencies that later emerged.

> *What is needed is 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...In this paper, we propose a solution to the double-spending problem using a peer-to-peer distributed timestamp server...* (— Nakamoto 2008)

10.2.1 Bitcoin and Blockchain

It is important to make a clear distinction between Bitcoin and the blockchain. As mentioned earlier, the blockchain is the *mechanism* that allows Bitcoin to work. Thus, Bitcoin can be considered to be an *application* that uses blockchain—but

Fig. 10.1 Bitcoin vs. blockchain

blockchain can be used on its own. It can be used to enable other cryptocurrencies, or as we will see in the next section, blockchain can also enable an array of different applications beyond Bitcoin and other cryptocurrencies (Fig. 10.1).

A simple analogy we can use is that of the *car* and the *combustion engine*. A car uses a combustion engine to function, but the combustion engine can be used to power other systems such as buses, trucks, boats, electrical generators, etc. Thus, we can think of the blockchain as the combustion engine and Bitcoin as the car. Bitcoin is just the first example of many possible applications of blockchain technology.

10.2.2 Evolution of Blockchain

Since its introduction in 2008, the blockchain has evolved as it has been adapted in a wide range of applications and industries. In the table below, we break down the different categories of blockchain as proposed by Melanie Swan in the book *Blockchain: Blueprint for a New Economy* (Table 10.1).

Blockchain 1.0: Blockchain 1.0 consists of the use of blockchain in digital *currency* applications for the decentralization of money or payment systems. This includes Bitcoin, other cryptocurrencies, and payment systems. In the beginning, these were the first applications to employ blockchain as a technology.

Blockchain 2.0: The next major innovation in blockchain, considered Blockchain 2.0, is a technology known as *contracts*. Beyond peer-to-peer payment systems, Blockchain 2.0 includes the transfers of other property such as stocks, bonds, and smart property. It also includes *smart contracts*, which are described later in this section.

Blockchain 3.0: Blockchain 3.0 consists of all applications beyond currency and markets. This includes the use of blockchain in areas like healthcare, governments, and commercial settings. In Sect. 10.5 of this chapter, we cover a couple of these segments and the potential use cases of blockchain in IoT.

Table. 10.1 Categories of blockchain

Categories	Description
Blockchain 1.0	Blockchains used for currencies
Blockchain 2.0	Use of smart contracts within blockchains
Blockchain 3.0	Applications beyond currency and financial markets

10.2.3 Defining Blockchain

A blockchain is composed of a distributed digital ledger that is immutable—cannot be edited—and is shared among all participants in a blockchain network. More specifically, a blockchain is a data structure composed of time-stamped and cryptographically linked blocks. Each block has a cryptographic hash, a list of validated transactions, and a reference to the previous block's hash. Through this mechanism, nodes can verify that a participant owns an asset without the need for a central governing authority. The key characteristics behind the success of blockchain are:

1. Decentralized architecture
2. A "trustless" system
3. Consensus mechanism
4. History of transactions
5. Ensured immutability

We consider these as the key factors that have made the technology transformational. The blockchain allows for participants to engage in trustless peer-to-peer transactions. In short, it is said that decentralized, trustless transactions are the key innovation of the blockchain [1].

10.3 How Blockchains Work

A blockchain is just what the name implies, a group of blocks linked, or chained, together cryptographically. It also keeps record of all transactions that have ever been executed by nodes on the network. In this section, we provide an overview of how blockchains work by using Bitcoin as an example. We examine how transactions are created, how they are broadcasted, how they are recorded into blocks, and how they are accepted into the distributed network of nodes.

Important Definitions

Nodes: Any computer or device connected to a blockchain network.
Ledger: A shared and distributed history of all transactions and balances.
Mining/Miners: In Bitcoin, mining is the process of generating a new legitimate block by applying proof-of-work. There are people that dedicate their nodes to "mine" new blocks. These nodes are considered "miners."

Consensus: A consensus algorithm is the mechanism by which all nodes in the network agree on the same version of the truth. A consensus algorithm allows nodes on the system to trust that a given piece of data is valid and that it has been synchronized with all other nodes.

Cryptocurrency: A digital currency built upon cryptographic protocols.

Decentralized Application (DAPP): A decentralized application built on top of a blockchain-based system.

Secure Cryptographic Hash Functions: A secure cryptographic hash function is a hash function that preserves one-wayness—easy to compute— but virtually impossible to reverse engineer.

Cryptographic Keys: The use of symmetric (same) keys and asymmetric (public-private) key pairs for the use of signing and verifying transactions.

Merkle Tree Root: The root of a Merkle tree (binary hash tree). The root is the result of all leafs hashed together to a single hash.

10.3.1 Anatomy of the Blockchain

Components of the block's header:

1. *Version:* The version of block validation rules it follows
2. *Previous Block Hash:* The hash of the previous block in the blockchain
3. *Merkle Root Hash:* The root of all transaction hashes in a block
4. *Timestamp:* The Unix epoch time the block was mined
5. *Bits:* Encoded version of the target threshold
6. *Nonce:* Arbitrary number that can only be used once
7. *Transaction Count:* Total count of transactions contained within this block

In Fig. 10.2, we show the basic architecture of the blockchain. A blockchain is very similar to a linked list—each block contains a pointer to the previous block. A key difference in blockchain is that each block contains a *hash pointer* to the previous block. A hash pointer contains two things: a pointer, or reference to the location of the previous block, and the cryptographic hash of that block. Storing the cryptographic hash of the previous block allows us to verify that the block we are pointing to has not been tampered with. To verify a block, we simply compare our stored hash pointer with the previous block's hash and make sure they are equal.

10.3.2 Understanding a Block's Hash

Cryptographic hash functions are an important aspect of blockchain's security. For this reason, let's take a look at how block hashes are calculated and how they are used in preventing an attack. To calculate the hash, three inputs are used: previous block hash, the Merkle root hash, and the nonce. These values are processed by the

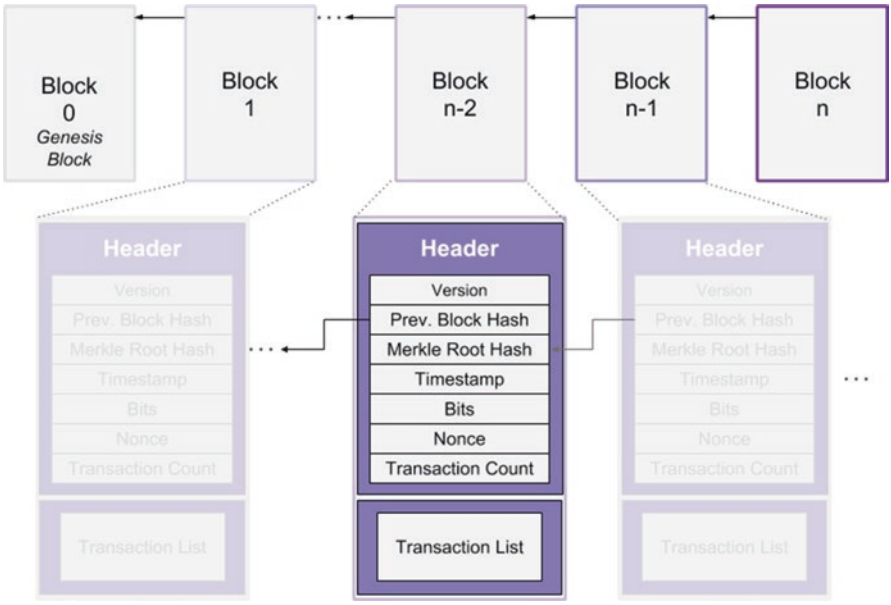

Fig. 10.2 Anatomy of the blockchain

SHA-256 cryptographic hashing algorithm. The output is the block hash—a fixed size output that uniquely represents all of the block's contents.

In Bitcoin, hashing is performed by miners, and the hash produced must be lower than the target hash set by the network. To find a hash meeting this criteria, miners try different nonce values and check if the output hash is lower than the target, while the previous block hash and Merkle root hash remain the same. Miners do this iteratively until a valid hash is found. Because of this, the mining process consumes a lot of power and compute resources. This procedure is how the miners create *proof-of-work*. In Fig. 10.3, we illustrate how the block hashes are calculated.

To understand how it works, consider a scenario where an attacker attempts to pay themselves some Bitcoins by modifying one of the blocks in the chain. Imagine they attempt to add a fake transaction to block 1, claiming that someone has sent them some coins. Upon changing the transaction list, the hacker will be forced to update the Merkle root hash. Because the block's hash is dependent on the Merkle root hash, if the Merkle root hash is altered, then we must recalculate the block's hash. But that is not so easy. In Bitcoin, it takes considerable compute power to mine one block. So the attacker would then have to invest power and time recalculating the block they maliciously altered. Once the attacker has calculated the new hash, then they have to figure out a way to make the block a legitimate part of the blockchain.

This is where hash pointers play a key role. For the attacker to alter any block in the chain, they also have to change every other block that follows! Why? Because every subsequent block points to the previous block—block 2 contains a

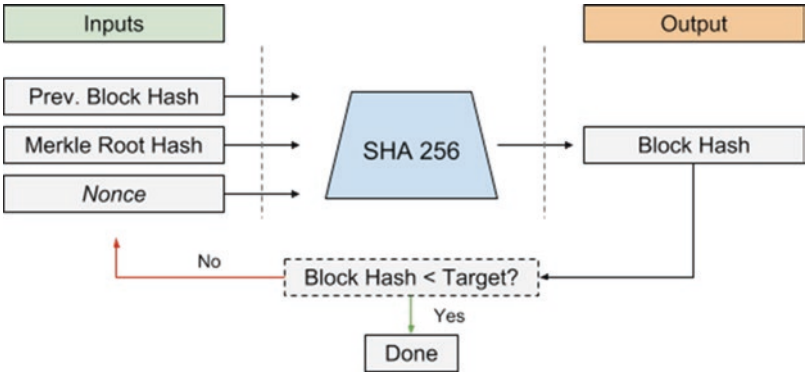

Fig. 10.3 Block SHA-256 calculation

hash pointer to block 1. But our attacker was forced to recalculate the hash for block 1, so a comparison with the hash pointer in block 2 will fail. To avoid this, the attacker must change the hash pointer in block 2 to match the block hash of the new (malicious) block 1. But changing the hash pointer in block 2 changes block 2's hash. Thus, the attacker would have to recalculate the hash for block 2 as well. Once they change block 2, same would have to be done for block 3, block 4, etc. During this time, the network has still been progressing, while the attacker is spending time altering past blocks. Time and cost used in such an attack is expensive and pointless as long as the attacker holds less than 51% of the network's compute power. It is this combination of proof-of-work and hash pointers that trumps 51% attacks and is considered to be the fundamental security feature of Bitcoin's blockchain.

10.3.3 Life Cycle of a Transaction (Fig. 10.4)

To understand how transactions are executed in a blockchain, let's consider an example scenario where Alice sends Bob 0.5 Bitcoin (BTC). In order for the transaction to take effect and be accepted into the blockchain, the following main steps need to be completed:

Let's assume Alice's current balance is 10BTC and Bob's is 2BTC.

1. *Alice agrees to send Bob .5BTC.*
   (a) Alice initiates a transaction using Bob's Bitcoin address. While Bob's identity is not linked to his Bitcoin address, Bob may create a new address for every new transaction to minimize tracking of his activity.
   (b) Bitcoin is pseudo-anonymous, meaning Bob's transactions are not fully obfuscated, and if his address is exposed in connection to his identity, then there are tools that can potentially track all of his past activity on Bitcoin.

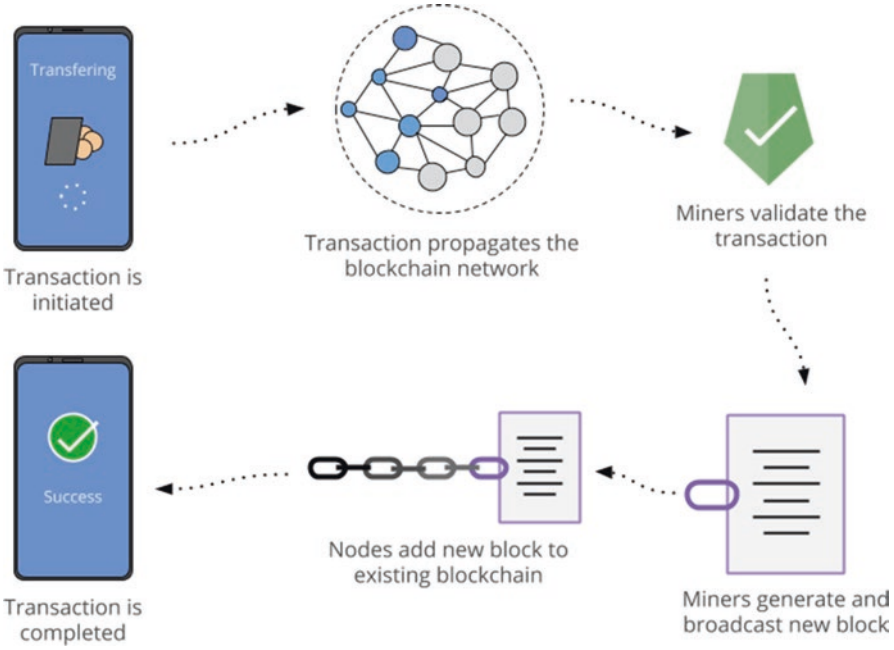

Fig. 10.4 How a Bitcoin transaction is executed

2. *Alice generates a transaction.*

   (a) When Alice broadcasts a transaction to the blockchain network, the message notes that Alice should now have 0.5 less BTC and Bob should gain .5BTC. In reality, no coin or asset is actually transferred (there is no digital coin that actually exists in the form of bits), instead, only records of transactions are recorded in the blockchain's ledger. In order for the transaction to be broadcast securely, Alice signs that the transaction is legitimate. This verifies that no one is trying to withdraw coins out of her wallet without permission. Alice signs the message using her private key to Bob's public key; thus, only Bob can spend these coins.

3. *Alice's wallet or interface into the Bitcoin network will now propagate the transaction to known peers.*

   (a) Once the transaction has been generated and is valid, Alice's wallet or interface to the network will propagate the transaction to her known peers. These nodes will in turn propagate it to their peers upon validating the transaction. This mechanism is called *flooding*.

4. *Miners receive the transaction, and validate it, ensuring that it has not been corrupted or tampered with.*

(a) Miners will use the consensus rules to validate the transactions making sure there is no double spending and that each address associated to the transaction exists.

5. *Miners include the transaction into a block and apply the consensus algorithm (proof-of-work in case of Bitcoin) to mine a new block.*

   (a) Transactions are then added to the new block in order of precedence. Transactions are added in descending order based off of their fees. Each transaction usually contains a fee that is paid to the miner. Once the miner receives the previous block in the network, they will start mining the newest block.

6. *Once a new block is mined, miners then broadcast the new block to be added to the blockchain by all other nodes in the network.*

   (a) The miner will propagate its new block to the network and begin the process all over again with new transactions.

10.4 Features of Blockchain

A blockchain provides key benefits that have never been possible before. These benefits stem from the clever combination of novel and existing technologies that allow the community to build innovative blockchain-based solutions. In this section, we cover some of the important features that a blockchain provides and discuss why they are important in IoT.

10.4.1 Consensus Algorithms in IoT

Blockchains can be considered "trustless" because they provide a mechanism to validate that data being added to the blockchain is legitimate. To achieve this, all nodes need a way of agreeing on the correct version of the truth. The algorithms used to reach an agreement are referred to as "consensus algorithms." For example, Bitcoin uses the *proof-of-work* (PoW) algorithm, but as we will see in this section, PoW is not the only algorithm that exists; there are many, and all of them offer different advantages and disadvantages. In IoT, it is essential that the consensus algorithms used can meet certain security, energy consumption, and computational requirements. In this section, we introduce a short list of the most prominent consensus algorithms and examine their viability in IoT solutions.

Byzantine Generals Problem
Before diving into different consensus algorithms, let's further define the goal of a consensus algorithm. On July 5, 1982, Leslie Lamport, Robert Shostak, and

Marshall Pease published a paper named "The Byzantine Generals Problem." From the original paper:

> *...imagine that several divisions of the Byzantine army are camped outside an enemy city, each division commanded by its own general. The generals can communicate with one another only by messenger. After observing the enemy, they must decide upon a common plan of action. However, some of the generals may be traitors, trying to prevent the loyal generals from reaching agreement. The generals must decide on when to attack the city, but they need a strong majority of their army to attack at the same time. The generals must have an algorithm to guarantee that (A) all loyal generals decide upon the same plan of action ... (B) A small number of traitors cannot cause the loyal generals to adopt a bad plan...The loyal generals will all do what the algorithm says they should, but the traitors may do anything they wish. The algorithm must guarantee condition A regardless of what the traitors do. The loyal generals should not only reach agreement, but should agree upon a reasonable plan.*

In the case of blockchain, the generals are the nodes in the distributed network, and the messages are the communications, or transactions, across the blockchain network. In short, how do all truthful network nodes reach a consensus on the validity of a new transaction even if there exists a certain percentage of malicious or faulty nodes? A *Byzantine Fault Tolerant* system is one that can tolerate the Byzantine Generals Problem.

Proof-of-Work (PoW): Proof-of-work algorithms require "miners" to solve a very complex cryptographic puzzle to try to prove that the current transactions on the blockchain are valid. This is the consensus algorithm used in Bitcoin. All miners receive transactions and begin a race to "mine" a new block. The first "miner" to solve this puzzle correctly wins and receives an incentive in return. In Bitcoin, "miners" receive Bitcoins as a reward. The reward is halved every 210,000 blocks. In PoW, nodes trust the longest chain—the one with the most blocks added to it by other miners. Thus PoW is safe as long as 51% of the compute power is owned by honest miners.

In PoW, solving the puzzle consumes a lot of computational power and takes considerable amount of time to complete. Thus, adding new blocks translates to high energy costs and low amount of transactions per second. In IoT, both present a big challenge. The main gateway and fog domain will most likely be in charge of computation and consensus as they can manage memory and power in a more sustainable fashion. Sensors will primarily rely on sending information to the fog and dealing with identity management between peers. PoW could potentially work with IoT devices, but there would have to be a strict separation of compute nodes and light clients (sensors) throughout the network. We argue that while a feasible algorithm for IoT solutions, it is not a good choice due to the large computational and energy requirements.

Proof-of-Stake (PoS): Proof-of-stake does not require expensive compute resources to mine blocks. Instead, PoS uses a validation process based on the amount of coins that you already own. If you own 1% of the stake in the blockchain, then you will have a 1% chance of getting chosen to create, or "mint," a block. Thus,

simply by having a stake in the system, you can be chosen to "mint" a block. The idea is that the more value you have at stake in the system, the less likely you will be willing to create a malicious block. If a block is invalidated by the rest of the network, then you lose your stake. This action will fall into an invalidation period, where the consensus for that transaction may be taken over by fellow peers, but your validity will drop among the nodes.

We argue that PoS would be a good fit for IoT because it does not suffer from PoW energy drawbacks and does not require high computational capabilities. With PoS, a possible drawback is that a node with more stake has more control of the network; and this control can continue growing because the node with the most stake is more likely to be chosen to mint a block. In permissioned blockchains, this should not be a problem, but more research is needed to understand the effects of PoS in permissioned and permissionless IoT blockchains.

Proof-of-Activity (PoA): Similar to PoW, proof-of-activity requires miners to mine a new block, the only difference being that the transactions on the network are not required to be part of the new block, and the mining is done for the sole purpose of solving a cryptographic puzzle. Once a new block is found, a similar validation to PoS is performed. The block is broadcasted to a group of chosen validators for them to sign the new block. The likelihood a new validator is chosen is similar to that of PoS, the more stake they own in the network, the more likely they will be chosen to sign the new block. Proof-of-activity suffers from the same drawbacks as PoW. Because of this, it is probably not a good choice for IoT applications.

Proof-of-Elapsed-Time (PoET): Proof-of-elapsed-time is a bit different than the other consensus algorithms mentioned so far. PoET was developed by Intel and is a proposed contribution to the open-source Hyperledger blockchain project. At a high level, PoET essentially works by assigning each node a random wait time, the validator with the shortest wait time "wins" and gets to mine the next block. The algorithm is considered to be "lottery algorithm"—the probability of being selected is proportional to the amount of resources contributed. This consensus algorithm has advantages in that it is much more energy efficient than PoW and does not require expensive hardware. On the other hand, it requires Intel processors to run it (requires trusted execution environment on the CPU), in which case it requires trust in Intel's hardware, which many say goes against the decentralization of trust concept. As far as IoT devices are concerned, we believe that PoET would be a good option for private IoT blockchains. This is because there is no need to have high compute power, or expensive hardware, and is also power efficient (Table 10.2).

While not an exhaustive list of consensus algorithms (and there are many), it is easy to see that at the heart of a blockchain is the consensus algorithm that glues the whole system together. Each consensus algorithm will have its own advantages and disadvantages depending on the use case; different industries and applications will apply different consensus depending on requirements such as scalability, transactions per second, and if the system will be permissioned or permissionless.

Table. 10.2 Consensus algorithms in IoT

Consensus algorithm	Description	IoT compatibility
Proof-of-Work	Computation is needed to solve cryptographic puzzle to ensure consensus	No
Proof-of-Stake	Ability to mint a new block is proportional to the stake in the blockchain network	Yes
Proof-of-Activity	Computation is needed to solve cryptographic puzzle to only known validators who are active	No
Proof-of-Elapsed-Time	Use of random time intervals that determine which node is the current miner	Yes

10.4.2 Cryptography

What makes blockchains trustworthy and secure is its underlying mechanisms based on cryptography, signed keys, and digital signatures. While Bitcoin has been exposed to various attacks in the past, it is worth noting that the ledger itself, or the blockchain, has never itself been knowingly hacked. In the past, Bitcoin hacks targeted Bitcoin wallets or Bitcoin exchange websites instead. Let's consider Bitcoin's cryptographic elements as an example and see how they are used to maintain the blockchain's integrity. Bitcoin's cryptographic components are mainly composed of the following:

- *Secure Hash Algorithm (SHA-256):* Cryptographic hash functions are a set of mathematical functions that output unique outputs for unique inputs. The input can be of any size, and the output is always a fixed size—256 bits (32 bytes) in the case of SHA-256. If any one bit of the input is changed, the cryptographic hash function outputs a completely different and unpredictable output. Secure cryptographic hashing preserves one-wayness, that is, you can easily produce a hash from a given input, but it is extremely difficult to generate the input to the hash by only knowing the hashed output value. SHA-256 is used for most functions including integrity, block-chaining, and hashcash cost function calculations.
- *Elliptic Curve Digital Signature Algorithm (ECDSA)*: ECDSA is used to create cryptographic keys that can derive addresses for use within the blockchain. Each ECDSA algorithm calls a specific curve to be used for key generation, which enables efficient computation.

Cryptography is at the heart of why the blockchain is so revolutionary. Everything from consensus algorithms, to encryption, to the immutability aspects of the blockchain are due to the underlying cryptography. This is a fundamental key in unlocking the potential to IoT, as different devices need to engage in transactions with trustless entities and devices on a constant basis.

10.4.3 Decentralized

Having a decentralized architecture can propel IoT applications to be realized at a wide scale. Currently, IoT systems mostly depend on client/server or publish subscribe architectures [7]. Centralized architectures require expensive infrastructure with high compute and storage capabilities. In addition, they present a form of centralized control that can be act as a single point of failure or the target of a security attack. Publish subscribe architectures can also have a few drawbacks with scalability and security. If devices could perform secure transactions using a peer-to-peer paradigm, it would greatly reduce the cost, transaction time, and probability of service interruption.

The blockchain is composed of a decentralized, distributed network of nodes that participate in transactions and maintenance of the network. This is the core concept behind blockchain. All transactions are peer-to-peer and are tracked by all of the participating nodes in a network. Blockchain networks have a reliability factor of $(n - 1)$—if any node fails, or drops from the network, there is no interruption to service. The network always maintains availability and fault tolerance. Decentralization in IoT is a very attractive alternative to previous architectures, but there are still many challenges, and no clear consensus on how to best take advantage of blockchains decentralized nature in IoT (Fig. 10.5).

10.4.4 Transparency and Trust

The use of a public ledger allows all nodes on the network to see the entire history of the given blockchain. This opens access to the history of data on the chain, giving transparency to all transactions. The trust that is built within the network is maintained through the use of the public ledger and gossip protocol. Each node always

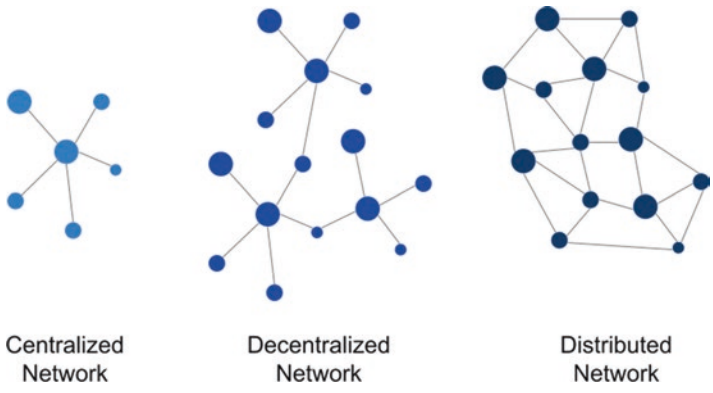

Fig. 10.5 Network types

knows of its nearest neighbors and new nodes. Through each node gossiping to one another, they learn of new transactions. Utilizing the public ledger and protocol instills trust within each node as each node is responsible for one another. This decentralization mechanism holds the nodes responsible for the integrity of the network.

10.4.5 Permissioned, Permissionless, and Consortium

Permissionless blockchains, such as Bitcoin, are designed so that anyone can join and participate in the network without having to establish their identity. There is no need to verify a given user through some sort of identity management system. The only identity needed is the user's public key. In contrast, permissioned ledgers are primarily used in private applications where strong indicators of identity are required to join the network. Permissioned ledgers are preferred among B2B and B2C enterprises. There are usually multiple layers of validation before enrollment to the network is verified. The use of regulators, as seen in IBM and Linux Foundation's Hyperledger, is used to ensure all users meet various requirements on the network. Other blockchains such as Ethereum give one the option to set up the network as permissioned, permissionless, or consortium. Consortium blockchains are very similar to permissioned blockchains. The key difference being that in a consortium, new participants are authenticated by a predetermined group of private entities.

10.4.6 Smart Contracts

Originally introduced by Nick Szabo in 1994, smart contracts consist of small computer programs that contain—embedded in their code—an agreement between two entities. This contract is then distributed across the blockchain and is responsible for facilitating the execution, verification, and enforcement of an agreement between seller and buyer. Essentially, a smart contract is just a digital, auto-enforceable version of a traditional paper-based contract. Ethereum is the most popular blockchain system with embodied smart contracts. It has a current market cap of more than $35 billion as of November 2017. As we will see in later sections, smart contracts allow devices in IoT to negotiate and execute previously agreed actions automatically, enabling a new set of functions and use cases for IoT solutions.

10.4.7 Advantages and Disadvantages

There has been much debate over the use of blockchain technology and its possible applications. In most use cases, traditional back end infrastructures offer a good solution to existing problems. Yet, the industry is beginning to move to a more

decentralized infrastructure to improve security and trust between users and the rest of the network. While blockchain presents a lot of promise, it is not a silver bullet—blockchain does not solve all security and privacy concerns, it is only part of the solution. With every new technology, there are advantages and disadvantages, and blockchain is just one part of a complex technology stack.

Blockchain technology has multiple disadvantages that have decreased its adoption rate. An often-overlooked challenge is that the technology is initially difficult to understand and adopt. Getting people to use blockchain applications is a difficult task, which brings disadvantages as people believe it's an unnecessary precaution for a network.

Scalability is another widely debated challenge. As an example, there has been much debate over scaling in regard to Bitcoin, which brought about a fork in the chain to allow larger than 1 MB block sizes. People felt that this size limitation doesn't scale with the adoption of Bitcoin, and transactions will take longer and longer to be validated and added to the main chain. There has also been other discussions about the scalability of Ethereum with the nature of storing everything within various Merkle roots, where over time, downloading the full chain will be much larger than Bitcoin's full chain (as of April 2018 its around 180GB). To avoid similar storage issues, people—especially users on mobile devices—use simplified payment verification (SPV) nodes which allow them to not run a full node and use filters to only grab the information that they need. This will rise over the next years as well as the use of Lightning Network and other off-chain protocols.

Other disadvantages include the size of the network and limiting the control of nodes. Whether one is building a permissionless, permissioned, or consortium blockchain, limits will have to be set on the admin privileges of nodes, so that the network does not gravitate towards a "centralized" paradigm. With this, there is always the risk of a Sybil (51%) attack on the network. As these are definitely important disadvantages, there are also a great deal of positives from the technology.

Blockchains bring about a new way to enable privacy and security between parties through cryptographic principles. The cryptographic principles employed ensure the handling of assets are controlled only by the one who holds the private key. The decentralized nature enables all users to share the responsibility for the integrity of the network. Blockchain uses an immutable ledger, once something is added into the ledger, cannot be changed or altered. This allows for a fully trustworthy system that we can trust will not be manipulated. There is no more "middleman" or centralized authority that holds all of the information. Every node on the network holds a copy of the ledger which allows for confirmations, for validity, and for a truly trustless system to survive. Blockchains are fairly simple to bootstrap once they are implemented. Furthermore, it is a new way to envision technology and the next frontier of Internet.

10.5 Blockchain Applications in IoT

From financial services, to government services, to peer-to-peer transactions, companies around the world are working to integrate blockchain into our everyday lives. Currently, there is no consensus on how exactly blockchain might transform different industries. Thus, in this section we introduce different IoT applications and examine how integrating a blockchain might transform these use cases.

10.5.1 M2M Transactions

According to Cisco, it is estimated that there will be 26 billion connected devices on the Internet by 2020. M2M interactions are essential for the true potential of IoT to be realized. Multiple challenges still need to be addressed for M2M interactions to truly flourish in IoT, including connectivity standards, lightweight security protocols, and ensuring data privacy, aspects which are covered in Chap. 4 of this book. While there are technical challenges in implementing M2M interactions, "smart contracts" introduce a solution to a fundamental M2M challenge: what protocol do the devices in the IoT utilize to negotiate and execute M2M transactions?

Smart contracts will be heavily used in IoT. Imagine a vending machine that can automatically order certain items and pay for the transaction through the agreement of a smart contract. All of this can be accomplished without the need for a central server, or other central entity. The contract would be automatically negotiated, executed, and enforced by the blockchain network.

10.5.2 Energy Management

Blockchains and smart contracts show potential promise in the energy sector. As mentioned in the IoT Verticals chapter of this book, IoT energy use cases include energy monitoring through smart meters and IoT energy management in the connected home. Through these mechanisms, power providers can collect more data on energy patterns and adjust power plant performance and predictability.

As the grid gets smarter and more capable, homes will be able to not only consume energy but also provide energy that they generate through solar, wind, or any other means. Potentially, homes could use smart contracts on a blockchain to negotiate energy exchanges and execute energy transfer from one home to another—automatically. Payments for the renewable energy transfer would be bought and sold via a blockchain network.

10.5.3 Supply Chain Management

The supply chain is often a complex set of interactions among a long chain of different vendors. Tracking a set of shipped goods is a convoluted task that requires information from many parties. IoT has already begun to provide insights that enable companies to collect data as their goods travel the globe. Sensors provide temperature data, location data, and more, giving companies newfound control and quality assurance that did not exist before.

Blockchain can be used to simplify the expensive logistics involved when shipping products around the world. By using smart contracts, shipments can be tracked at each stage. Every time a product arrives to a location, that product can be scanned, and a contract would be executed between the two vendors exchanging goods. This would enable an open and verifiable history of where the product was handed off, its condition, and if the contract terms were met (time, date, temperature, etc.). This eliminates the need for each stakeholder to independently track an asset in their own database, a database that provides no transparency, collaboration, or verification with all other stakeholders in the supply chain.

In addition, using a blockchain network to track products can provide more transparency and accountability in trading. Consumers will be able to track where their products came from and how the product arrived to their doorstep. For example, according to the Mintel Press Office, only 26% of consumers trust organic food labels, and only 13% believe that organic foods are highly regulated. Having better insight into where food was grown, how it was processed, and how it arrived to the store is important to consumers. There are already exist IoT solutions in agriculture that aim to improve quality of food by means of yield monitoring, optimal seeding, optimal water usage, and more. There also exist IoT solutions in supply chain management to monitor conditions and track of goods as they are transported from the source to the store. Adding all of the information collected via IoT to a consumer-accessible blockchain would provide the consumer a secure, trackable, and tamper-proof way of understanding where their goods were sourced from and how they got to their store, increasing the trust between consumer and producer (Fig. 10.6).

10.5.4 Healthcare

Healthcare is considered one of the most important verticals for IoT. Intelligent wearable devices present new ways to monitor noncritical patients remotely, while clearing up room in hospitals for more critical patients. The healthcare industry is already adopting real-time tracking of medical devices, personnel, and patients. That said, there are still critical challenges in the collection, management, and distribution of patient data that blockchain has potential to provide solutions for.

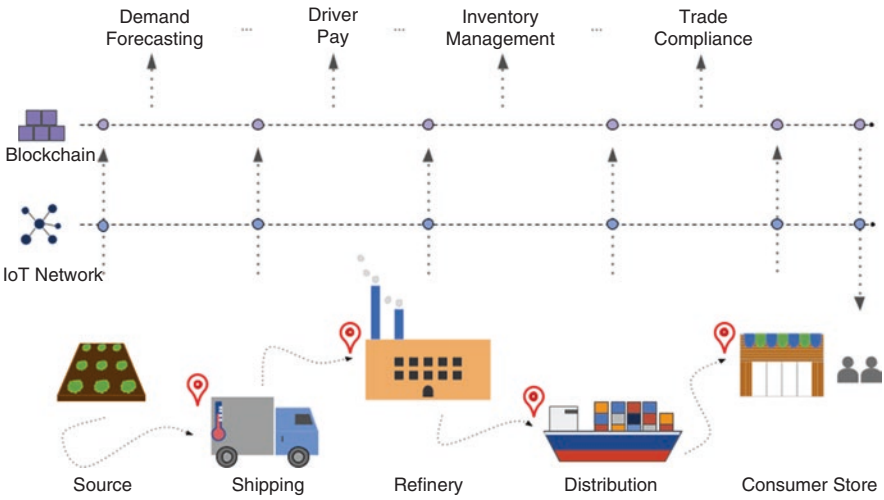

Fig. 10.6 Blockchain in supply chain

The main limitation blockchains can help improve is around the collection and storage of patient data. According to the centers for Medicare and Medicaid services, these records hold information such as demographics, progress notes, problems, medications, vital signs, past medical history, immunizations, laboratory data, and radiology reports. Currently, these electronic health records (EHRs) operate largely in silos; each medical facility collects, maintains, and stores its own medical records for each patient. This creates a high potential for duplication of data while also preventing the cross validation, verification, and data accuracy. Blockchain may allow for all medical records to be stored and shared in a decentralized manner, ensuring one verifiable, non-immutable source of information on any patient, to any authorized provider.

Having EHRs on a blockchain would provide a mechanism that would enable:

1. *IoT Data Exchange*: M2M medical data management would open the doors for a secure and viable way for patient's data to be monitored remotely by medical staff. The blockchain allows for these M2M interactions to happen automatically and would ensure a secure data transfer while preventing duplication of data. In addition, when IoT sensors can exchange data through the blockchain, data is protected from tampering and single sources of failure can be eliminated.
2. *Data Interoperability*: The potential to create a single EHR system is an opportunity that the entire industry is excited about. It is so important that according to the Premier Healthcare Alliance, sharing data across organizations could save hospitals about 93 billion dollars over 5 years alone. A blockchain based system would contain a single version of patients records and would be shareable, traceable, and anonymized and would put the patient in control of what records could be accessed and by whom.

3. *Drug and Treatment Management*: In 2015, a study conducted by *The American Journal of Managed Care* found that 76.9% of patients that participated in the study had at least one medication discrepancy in their medication lists. In addition to errors, there are also issues with ensuring that controlled substances such as opioids are not abused or that a patient is not a victim of fraud. A shared EHR system would allow pharmacies and medical staff to ensure that a patient is not prescribed more than once, and would provide a clean record of substances taken in the past. The power of shared data, combined with new smart labels that leverage the IoT to remind and track a patients' prescription drug intake, would provide very useful data not just for doctors, but also for the machine learning algorithms that are trying to provide more specialized care.

There are of course a lot of challenges that still need to be addressed such as privacy and access management, to name a few. Additionally, blockchain's adoption in the healthcare sector will largely depend on the cooperation of healthcare providers, who currently depend on a large array of proprietary software solutions and established IT infrastructures.

10.5.5 Retail

IoT is already used in retail—enabling the tracking of products in stores, automating and tracking product delivery, and allowing for more beneficial loyalty programs. The blockchain can further these advances, which can result in a better customer experience by increasing consumer trust and improving consumer reward programs.

Product Authenticity: The authenticity of a product is difficult to identify and can result in damage to brands and declining sales. An IP Commission Report on US intellectual property mentions that the cost of counterfeit goods to the US economy could be anywhere between $225 and $600 billion annual US dollars. Blockchain, along with IoT solutions, would provide the consumer with a clear and direct insight into the entire history of the product—from where their product was manufactured, to how it arrived to the store. Such transparency drastically lowers the possibility of a merchant or consumer unknowingly buying a product that is not genuine.

Loyalty Programs: Currently, many loyalty programs work in silos and do not work together to benefit the consumer. IoT solutions are allowing retailers to enhance the customer experience by collecting data on customer patterns and behaviors. Blockchain could allow for one universal loyalty program that a consumer can use at any store or to buy any service. This way, all the companies get access to consumer behavior while providing more value and savings for the customer.

Inventory Tracking: IoT solutions have allowed retailers full visibility into their products and merchandise—along with the ability to track product performance and stocking levels through digitized inventory and supply chain. Like in other use cases, blockchain introduces a way to track a product at every point in the supply chain and in-store, providing an accurate, and up-to-date, trail of where the product is.

10.5.6 Automotive and Transportation

The automotive industry is going through transformations unlike ones seen in the last few decades. From electric vehicles to autonomous vehicles, the industry is going through significant technological shifts. These changes represent vast opportunities for drivers, manufacturers, and other stakeholders such as insurance providers and dealerships. Companies have already discovered how IoT can improve services and efficiency, and can even provide real-time visibility into vehicle functions. As in other use cases, blockchain can augment IoT to create an array of potential benefits.

M2M Microtransactions: One of the most important use cases that blockchain can securely enable is M2M microtransactions. Vehicles would be able to automatically negotiate and pay for a wide array of services: services like finding or reserving a parking spot automatically, or negotiating a faster lane if the person in the car is in a hurry, or automatic payments at a gas station or charging station, just to name a few. All of these M2M interactions could be negotiated and automatically executed through smart contracts. As the industry moves towards autonomous vehicles, these M2M transactions will become ever more crucial.

Vehicle Dynamic Ecosystem: IoT, analytics, artificial intelligence, and blockchain are redefining how vehicles will be owned and cared for. IoT is enabling manufacturers to collect and track more data about their vehicles, improving in-vehicle experience, maintenance downtime, and quality. Logging sensor data in vehicles to a blockchain-based system enables the automobile ecosystem to view all of the same data about a particular vehicle, a set of vehicles (specific model), or even a brand of vehicles. This means that regulators, manufacturers, insurance providers, etc. all see the same exact data on a vehicle. Data that cannot be modified is reliable, opening the door for opportunities for new business models. Insurance providers could automatically provide dynamic pricing based on driving behaviors and even automate the insurance claim process as soon as an accident is detected. Manufacturers could automatically use that same data to run analytics on their vehicles to extract patterns and possible issues early (allowing for a more proactive recall and maintenance schedule). Even auto financing and title transfers could be done in a much faster, transparent, and verifiable way through the blockchain.

10.5.7 Smart City

According to a past World Urban Prospects report, 54% of the world's population lived in cities as of 2014, and that number is expected to grow to about 66% by 2050. With such population growth, cities have already begun developing smart cities to cope with growing challenges and provide more benefits for their citizens. Around the world, there are hundreds of smart city pilots taking place. IoT solutions are being used to digitize the world around us and improve things like transportation, air/water quality, energy management, and public safety.

Blockchain and IoT: To support the evolution of smart cities, the blockchain can be combined with current IoT solutions. Blockchain can accelerate the adoption of energy microgrids by providing a billing system for automatic negotiation and execution of energy distribution. It can be used to automate water supply management by implementing smart contracts that continuously track and manage water distribution so that it happens in the most efficient manner. Air and water quality can also be improved by implementing blockchain systems to record and share data from sensors installed all around the city.

Governance and Digital Services: Another way that blockchains can influence smart cities is through the digitization of citizen records and government services—with the potential to practically eliminate paperwork across government agencies and services. Here are some examples of potential services or solutions:

- *Civil Registration:* Can be used for record keeping of each citizen. A blockchain would make these records secure, tamper-proof (reducing fraud), and shareable among a variety of stakeholders with needed access to the data.
- *Citizen Identity:* Holding the digital identities for each citizen on a blockchain. Digital management of one's identity through blockchain could eliminate a lot of paperwork and make government services much faster and more efficient.
- *Governance:* Digitizing all records and transactions would transform the efficiency of government agencies. Currently, all records are maintained in silos, making sharing of data cross agencies hard and inefficient. Not only would a blockchain improve efficiency but also would increase transparency and visibility into processes.

10.5.8 Identity, Authentication, and Access Management

Most application stacks require a form of authentication. This topic has been researched and implemented through handshake protocols, key escrow, and various cryptographic modules. For IoT, with respect to blockchain technology, there will have to be a primary use of identity management. When a new device is added to the

network, the use of key escrow will have to establish their identity on the network. This exchange and generation of keys can be determined within the secure enclave of the device's hardware—removing the risk of attacks via ports, wireless, and Bluetooth capabilities.

Once the identity of the device has been set up, it would broadcast its public key/address to the network for others to know of its presence. The use of public key/asymmetric cryptography adds a benefit to the network, where all we need to know is your public key. Similar to Bitcoin, there could be time-sensitive intervals set in place, where the device generates new addresses—thus never using the same address twice. This can promote anonymity to malicious observers and sway predictive analysis by attackers.

Another alternative to identity management is to use an escrow or auditing node. This node can be in charge of asset management and communicating to others when a new node has joined or established itself to the network. In a sense, they will work as a directory server similar to BitTorrent-type peer-to-peer networks. This allows for easily addressing the key-value store of asset management which could be mapped to public addresses.

As blockchain evolves and starts being used in IoT frameworks, the identity of each device and model on the network will become significantly more important. Each individual device will be granted access via identity and key management. The key management will need to be controlled via the hardware on each device where the actual access is done through software. As IoT networks are quite large, the entire infrastructure will need to uphold strong cryptographic modules to maintain identity management. While actual key storage is done through HSMs on each device.

10.5.9 Other Blockchain IoT Applications

While it is still too early to tell which IoT solutions the blockchain will revolutionize, the ones mentioned in earlier sections constitute the use cases with the most notable traction. Other notable use cases include:

Decentralized DNS: Provides a more secure Internet that is decentralized and not easily hijackable, potentially preventing past attacks on IoT devices. Examples include Namecoin and EmerDNS, already available through browser extensions.

Legal Contracts: Provides a system where things such as ownership registries, notary services, taxes, and even voting could be performed on a blockchain.

Insurance: Insurance on a blockchain could affect multiple industries. In automotive, insurance can turn into an on-demand and dynamic policy system based on information that is retrieved in real time from sensors in your car. The same could be done for other property like your home.

Sharing Economy: Companies like Slock.it have created a platform that allows anyone to share anything with others. Using blockchain, they can lock and unlock physical assets based on predetermined smart contracts, giving anyone temporary access to any physical asset.

10.6 Blockchain Security in IoT

When it comes to blockchain technology, there are normal security risks that modern day technological infrastructures face every day. Yet, blockchain technology also holds an important security risk which involves key management. As mentioned in earlier sections, one's private keys are the ultimate key/password to obtain your information and assets. Whoever holds those private keys holds your identity. Throughout this section, we will discuss the advantages and disadvantages of security within blockchains and how that relates IoT.

10.6.1 Trust Between Nodes

Decentralization allows for a trustless mechanism to perform consensus among nodes while adhering to one's privacy and truthfulness. Do you have what you say you have? Based off of your connection, known past activity, can we correctly identify you? All of these questions and more need to be asked when building a blockchain-based system. The elimination of a single point of failure is a huge win for all stakeholders. However, this now brings attack vectors to all nodes. Especially in the realm of IoT, we need to carefully consider the protocols between applications and nodes. All messaging between nodes should be secure and private. There should be no possibility of 51% attacks or node compromises. Sybil attack, also known as a 51% take-over of a network, is one of the attacks that is looked at when it comes to consensus and node propagation. The use of keys, messaging systems, and gossip protocols can help protect against this as there are multiple layers of verification before any information that a node posts to the network is accepted and added to the chain. To ensure all nodes are safe, we need to maintain trust between them.

A node first joins the network by bootstrapping off of some discovery peers. These peers are hard coded into the blockchain code base. These nodes may be the major nodes that help uphold the network, or just the main nodes that we started with. Once they connect, they start gossiping between one another to propagate information throughout the network and add blocks to the chain. If an attacker could control a node within this network, they could potentially take over the entire network or propagate faulty information to sway or alter the chain in some malicious manner. By posting invalid transactions and possibly using another malicious node to accept it could be catastrophic. Luckily, most security measures will be built into

the blockchain network when developed. The trustless nature of a decentralized network allows for a consensus to take place among nodes before things are committed to the chain or propagated. If a faulty transaction is propagated, another node will realize that this doesn't match the chain and has not been seen by other nodes. Nodes could add in delays between propagation to make sure that n of m other nodes have verified this transaction or information that was propagated to validate the peer.

10.6.2 Malicious Activity and Cryptographic Principles

If malicious activity does happen to take place within a blockchain, the hope is that nodes and users will easily be able to verify given information based off of the secure cryptographic nature of information through the use of secure cryptographic hash functions or elliptic curve cryptography. If faulty information is introduced into the network, then the actual hash of that information will be different than what is recorded in the main chain. From basic verification of hashes, we can easily distinguish the integrity of the data (as seen in Sect. 10.3). Also, the specific curves that are used in the elliptic cryptography modules are specific and used for a reason. It is highly advised to use NIST approved and known cryptographic modules. Never attempt to write proprietary cryptographic libraries. Most of the time, these will not be tested as in-depth as NIST approved libraries and have the potential for collision. Collisions within cryptographic modules can lead to stolen keys and overall compromise of the blockchain.

Attacks and hacks have been taking place within the blockchain industry over the past few months of 2017. Most have been from ICOs or "Initial Coin Offerings" for Ethereum's ERC20 tokens. Others have been from exchanges and hardware wallets. These attacks mainly occur from exploiting bugs in smart contract code or finding flaws in the safeguarding of private keys. As we have mentioned throughout this chapter, private keys are the holy grail of wallets and blockchain identity. To maintain security and privacy when it comes to keys is to ensure a proper key management and escrow process. Key management can be taken care of in software or hardware. The use of HSMs (hardware security modules) can move the overhead of key escrow and processing to a hardware device to ensure privacy, security, and proper authentication mechanism for nodes. When maintaining your key management and escrow in software, there are more attack vectors exposed. Some core wallets within the blockchain space keep keys in "keyfiles" or a file that is held in local storage. This can be attacked from any sort of malware, from phishing attacks to visiting a malicious website that installs loggers onto your system. A way to protect keys when they are stored through software is by using multi-signature wallets. Multi-signature wallets need more than one user to have access to the wallet. By using a n of m or a majority of the users to allow access to a wallet means that if one key is compromised, the entire wallet is not lost.

10.6.3 IoT Security and Blockchain Advantages

Most IoT devices rely on a central entity to send them information or alert them of security risks. By moving these responsibilities to individual nodes that are decentralized, it theoretically makes these devices "smart" devices. By using a blockchain for IoT, the security level and fundamentals will greatly increase, and it will put the messaging and alerting functions within each devices protocol layer. When dealing with the multiple layers within blockchain technology, we commonly focus upon consensus. Consensus algorithms are what allow the decentralized network to obtain a "trustless" model. As mentioned in earlier sections, there are various types of consensus algorithms to base your blockchain on when building its network and infrastructure. The types of attacks vary in regard to consensus algorithm. For example, PoW deals with miners providing enough computation to gain rewards, enabling the blockchain to grow. In this type of blockchain, the attacker would have to perform a Sybil attack to compromise the network. In a PoS blockchain, the attacker would have to take control of the actual digital asset or sway the market, as nodes use the asset as a proving point within the network. Attacks within PoS blockchains deal much more with attack vectors of change in currency, whereas PoW deals with Dos/DDoS and Sybil attacks. PoS takes a different approach to the normal attack vectors that the security industry has seen over the years. In regard to IoT, the use of PoS would be a great benefit as each device could "mint" its own token in order to pay into the blockchain or protocol of their nature. This will protect them from Sybil attacks which could happen on a specific protocol layer and maintain consensus among n + m IoT devices.

Blockchain offers many security advantages for any desired application or system. Yet blockchain technology is not the end-all-be-all answers for all applications. Blockchain technology should only be used for use cases that require high security, privacy, and a peer-to-peer nature in regard to networking. IoT can greatly benefit from blockchain technology as it will be able to secure the protocol layer and information that is broadcasted between devices and networks. As blockchain technology grows, so will the attack vectors. There will always be phishing attempts, punycode domains, and smart contract hacks. As time progresses, the security space will evolve to build out standards and proper testing methodologies for blockchain technology. The importance of key management, node propagation, messaging, and consensus is what upholds the privacy and security within blockchain technology. Attackers will always try to outsmart your system, so be aware of your technology when building and implementing it in both a secure manner in regard to IoT and blockchain technology.

10.7 Summary

Blockchain is expanding to new industries everyday and has the possibility to propel IoT forward. This potential is greatly due to the technology's foundation in cryptography and the mechanisms by which it addresses the Byzantine Generals

Problem. Blockchain presents key features such as decentralization, security, and trust—all important aspects in IoT solutions. A handful of use cases in M2M, energy management, supply chain management, healthcare, retail, and transportation display a picture of a fast-emerging technology within various industries. Lastly, it is important to consider the challenges being faced by blockchain, such as scalability, privacy, and anonymity. While blockchain is not the answer to all the challenges in IoT, it should be clear to appreciate why the hype exists—the technology presents many new possibilities that are only beginning to gain traction.

Problems and Exercises

1. What is the double-spending problem in digital currencies?
2. Describe what a "Merkle tree" is? How is it used in Bitcoin?
3. In Sect. 10.3.2, we mention hash pointers and how they are key to immutability of the blockchain. Keeping that in mind, what are other features of blockchains that work with hash pointers to maintain immutability?
4. What are they key characteristics provided by the blockchain? Explain what they are and why they are important for adoption in IoT solutions.
5. What is a hash function and how does it work? What is the difference between a hash and a cryptographic hash function? Provide an example of how cryptographic hashes are used in a blockchain (any blockchain will suffice as an example).
6. What is a hash collision? Does Bitcoin suffer from the probability of hash collisions?
7. Consider a scenario where there a potential double-spend attempt by a malicious actor in Bitcoin. Explain how the blockchain works to reject such attempt and what the malicious actor would have to do in order to fool all other honest nodes.
8. In table format, describe centralized, decentralized, and distributed network architectures.
9. Perform a search and mention five companies that are currently working on blockchain + IoT solutions. Describe their solutions and how IoT and blockchain is being combined. Make sure to include at least one start-up and at least one established company.
10. What type of records can be kept in a blockchain?
11. In Sect. 10.4, we describe some consensus algorithms. Research consensus algorithms for blockchain and name an algorithm that we did not mention in this section. Is it good for IoT? Explain why or why not.
12. What is elliptic curve cryptography and how does it benefit the use of keys within blockchain technology?
13. Describe a Sybil attack and other types of attack vectors that could take place on a blockchain.

14. Blockchains all start from a genesis block and then maintain a block height as the chain grows. Describe the importance of block heights as timestamps and lookups within Merkle trees.
15. Describe the difference between permissioned, permissionless, and consortium blockchains. What type do you think best fits a blockchain involving IoT devices.
16. What is the difference between a smart contract and multi-sig address?

References

1. M. Swan, *Blockchain: Blueprint for a New Economy* (O'Reilly Media, Inc., 2015)
2. S. Nakamoto, *Bitcoin: A Peer-to-Peer Electronic Cash System*. (2008), www.bitcoin.org
3. K. Christidis, M. Devetsikiotis, Blockchains and smart contracts for the internet of things. IEEE Access **4**, 2292–2303 (2016)
4. D. Chaum, Blind signatures for untraceable payments. in *Advances in Cryptology-CRYPTO,* (1983) pp. 199–203.
5. V. Gupta, A brief history of blockchain. Harv. Bus. Rev. (2017). hbr.org/2017/02/a-brief-history-of-blockchain
6. N. Szabo, Formalizing and securing relationships on public networks. First Monday **2**(9) (1997). https://doi.org/10.5210/fm.v2i9.548
7. K. Croman, et al., On scaling decentralized blockchains. in *Financial Cryptography and Data Security Lecture Notes in Computer Science* (2016), pp. 106–125. https://doi.org/10.1007/978-3-662-53357-4_8
8. B. Dickson, Decentralizing IoT networks through blockchain. in *TechCrunch, TechCrunch* (2016). techcrunch.com/2016/06/28/decentralizing-iot-networks-through-blockchain/
9. I. Crigg, K. Griffith, A quick history of cryptocurrencies BBTC — before Bitcoin. in *Bitcoin Magazine* (2014). [Online]. Available: https://bitcoinmagazine.com/articles/quick-history-cryptocurrencies-bbtc-bitcoin-1397682630/
10. A.M. Antonopoulos, *Mastering Bitcoin: Programming the Open Blockchain* (Sebastopol, CA: O'Reilly Media Inc., 2017)
11. N. Kshetri, Can blockchain strengthen the internet of things? IT Professional **19**(4), 68–72 (2017)
12. V. Nordahl, M. Rao, Blockchain cryptography. in *My Blockchain Blog* (2017). [Online]. Available: https://www.myblockchainblog.com/blog/blockchain-cryptography
13. K. Lewis, Blockchain: four blockchain use cases transforming business. in *Internet of Things blog* (2017). [Online]. Available: https://www.ibm.com/blogs/internet-of-things/iot-blockchain-use-cases/
14. N. Murty, S. Ananthasayanam, A. Singh, R. Malhotra, V. Vaid, A. Madan, *Blockchain: The Next Innovation to Make our Cities smarter* (PWC, 2018), pp. 22–30
15. A. Castor, A (short) guide to blockchain consensus protocols – coindesk. in *CoinDesk* (2017). [Online]. Available: https://www.coindesk.com/short-guide-blockchain-consensus-protocols/
16. Z. Witherspoon, A hitchhiker's guide to consensus algorithms – hacker noon. in *Hacker Noon* (2018). [Online]. Available: https://hackernoon.com/a-hitchhikers-guide-to-consensus-algorithms-d81aae3eb0e3
17. C. Hammerschmidt, Consensus in blockchain systems. In short. – Chris Hammerschmidt – medium. in *Medium* (2017). [Online]. Available: https://medium.com/@chrshmmmr/consensus-in-blockchain-systems-in-short-691fc7d1fefe

18. D. Mingxiao, M. Xiaofeng, Z. Zhe, W. Xiangwei, C. Qijun, A review on consensus algorithm of blockchain. in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (2017)
19. L. Lamport, R. Shostak, M. Pease, The byzantine generals problem. ACM Trans. Program. Lang. Syst. **4**(3), 382–401 (1982)
20. F. Tschorsch, B. Scheuermann, Bitcoin and beyond: A technical survey on decentralized digital currencies. IEEE Commun. Surv. Tut. **18**(3), 2084–2123 (2016)
21. A. Bahga, V. Madisetti, Blockchain platform for industrial internet of things. J. Softw. Eng. Appl. **09**(10), 533–546 (2016)