A Taxonomy of Blockchain Threats and Vulnerabilities

Ayman Alkhalifah 1,†, Alex Ng 2,† A. S. M. Kayes 3,†,* Jaleb Chowdhury 4,† Mamoun Alazab 5 and Paul Watters 6,†

1 La Trobe University; 19545133@students.latrobe.edu.au
2 La Trobe University; alex.ng@latrobe.edu.au
3 La Trobe University; a.kayes@latrobe.edu.au
4 La Trobe University; m.chowdhury@latrobe.edu.au
5 Charles Darwin University; mamoun.alazab@cdu.edu.au
6 La Trobe University; p.watters@latrobe.edu.au
* Corresponding Author: A. S. M. Kayes, Email: a.kayes@latrobe.edu.au; Tel.: +61 03 94795772
† Address: Computer Science Information Technology, La Trobe University, VIC 3086, Australia

Abstract: Blockchain technology has become one of the most popular technologies for maintaining digital transactions. From the foundation of Bitcoin to the now predominant smart contract, blockchain technology promises to induce a shift in thought about digital transactions in many fields, such as energy, healthcare, Internet of Things, cybersecurity, financial services and the supply chain. Despite blockchain technology offers many cryptography advantages such as immutability, digital signature and hashing; it has suffered from several critical cybersecurity threats and vulnerabilities.

In this paper, we build upon the previous studies on vulnerabilities and investigates over 60 real cybersecurity incidents that have been happening on the blockchain networks between 2009 and 2019. We categorise those incidents against the key cybersecurity vulnerabilities in blockchain technologies; and have developed a taxonomy that captures five types of cybersecurity threats and vulnerabilities based on five main players in blockchain. The outcome of this research prompted concerns and research direction in developing countermeasures to alleviate these risks.

Keywords: Blockchain; Cybersecurity; Attack; Threat; Vulnerability; Taxonomy

1. Introduction

Blockchain Technology (BT), which is the underlying technology of Bitcoin, has emerged with a number of promising potential applications. In less than a decade, BT has seen investment by many companies, stimulated the establishment of a range of consortia and raised more than US$3.1 billion in total venture capital [1]. The cryptocurrency markets created by BT is estimated to reach a total market capitalisation of more than US$143 billion [2]. Furthermore, BT has been applied to the public sector, academic institutes [3] and governments have been planning and developing the use of blockchain in the public sector [4]. KPMG [5] reports the first-half of 2018, investments in the U.S fintech companies is $14.2 billion. Furthermore, BT has been applied to the public, and private sectors with over 53 per cent of Deloitte’s survey respondents say that BT is a priority for their organisations [6].

BT promises a new dimension of conducting business transactions among untrusted entities; its features that support verification, identification, authentication, integrity and immutability are guaranteed through cryptography, transparency and decentralised smart contracts and smart ledgers. BT offers chronologically linked and replicated digital ledgers in a decentralised database and a sharing
of transactions in an extensive network of untrusted entities. BT provides independent verification guarantees which eliminate the need to rely on a central authority. Furthermore, given the absence of central authorities, blockchain services are able to provide better security properties for systems that are distributed among different entities and can apply immutability against abuse and supervision even if there is a malicious insider.

Given that BT is a cutting edge technology with many promises, there are concerns about its robustness [7]. If such an authority exists in a system, tampering with blockchain or hindering the broadcast of its contents is possible with a collusion between the most powerful entities. There have been many reported cyber attacks, and several cybersecurity vulnerabilities have been identified in blockchain implementations [8]. A recent example is the dusting attack on the Litecoin blockchain network which shows attackers are able to break the privacy and anonymity of Bitcoin users by sending tiny amounts of dust coins to the personal wallets of the target victims. The attackers are able to track down the transnational activity of these wallets by performing a combined analysis of the addresses which is able to identify the owner of each wallet [9].

These defects raise questions about whether BT can deliver the security guarantees - in practice - that it promises. The growing use of BT as a service delivered by governments or large firms, such as the financial technology industry, has raised users’ concerns about its security. Recently, several reports have been published about cyber attacks and cybersecurity vulnerabilities in BT. For instance, 8,833 existing Ethereum smart contracts are vulnerable, and their total balance is 3,068,654 million Ethers, which is equal to about US$30 million [8]. Financial losses are possible because of the vulnerabilities in the smart contracts. For example, an attacker attacked Mt Gox in 2014—the largest platform for Bitcoin trading—and stole Bitcoins equal to US$450 million, which led to the collapse of Mt Gox. Another example is when a hacker managed to exploit a vulnerability and steal Ethers which were equal to more than US$60 million in 2016 from the DAO, a smart contract in Ethereum blockchain [10]. Taylor et al.[11] showed that recent researches on BT in terms of cybersecurity concept are mostly concerned with how blockchain can provide security to current and future systems. There is a paucity of studies on the cybersecurity vulnerabilities of BT. This research has the the following questions in mind:

- Is blockchain a proven technology which delivers what it promises?
- What vulnerabilities are usually exhibited in the BT design?
- How have the theoretical vulnerabilities in blockchain been exploited between 2009 and 2019?

In this paper, we extend the previous works from different entities and conduct a systematic analysis on identifiable incidents and vulnerabilities reported so far and as such to establish a direction for the future development of BT. We adopt the following methodology to conduct our research:

- We started with reviewing news available from sources which are specialising in Bitcoin, Ripple, Ethereum and other digital currencies. The purpose is to collect information about incidents reported in the news and blogs. CoinDesk, CoinTelegraph, Bitcoin.com, and CCN are some of the sites that we have visited.
- The second stage is to analyse the incidents to look for clues that can lead us to the answers that we have set in the questions above.
- The third stage is another round of review in search of any news and articles which give explanations to those incidents allowing us to conclude the questions set in this research.

The contributions of our study are as follows:

- We have identified 65 blockchain-related cybersecurity incidents between 2011 and 2019;
- We have conducted a detailed analysis of the attack vectors, victims and damage in each incident and have categorised the attack vectors;
We have identified a list of major concerns for blockchain system designers and developers to consider; and
We have developed a taxonomy which classifies all the known threats and vulnerabilities identified so far under five main type of attack vectors.

The rest of this paper is organised as follows. Section 2 introduces some of the distinctive features exhibited by BT. Section 3 studies the threats and incidents that have been happening on the blockchain network since 2011, Section 4 presents our findings in categorising the threats and vulnerabilities of BT and finally, future work and conclusions are presented in Section 5 of this paper.

2. Blockchain Technology Explained

BT comprises tamper-proof and tamper-evident digital ledgers executed without a central repository, as a distributed system, and often without a central authority, such as a government, bank or firm. BT allows users in a community to store transactions in a shared ledger within that community. The transactions cannot be changed when they are published in the blockchain network’s normal operation. A new cryptocurrency based on blockchain was created in 2008 by combining BT idea with other computing concepts and technologies. In 2009, BT became very famous following the start of the Bitcoin cryptocurrency, which allowed digital cash to be transferred within a distributed ledger. In Bitcoin, users’ digital rights can be digitally signed and transferred to another Bitcoin user. The Bitcoin blockchain announces this transfer publicly to all the network users to independently verify the transaction’s validity; moreover, a distributed group of users independently manage and maintain the Bitcoin blockchain, and this together with cryptographic mechanisms creates BT’s resilience towards subsequent attempts to modify the ledger by counterfeiting the transaction or altering the blocks. BT has enabled the development of numerous cryptocurrency systems, such as Ethereum and Bitcoin, and that is why some people tend to restrict BT to cryptocurrency solutions only, however, a range of different industry sectors are contemplating using BT in their applications [12].

The Nakamotos white paper [13] introduced the concept of electronic cash and the launch of the Bitcoin cryptocurrency in 2009 makes BT became one of the widely talked about technologies. Blockchain is a database of blocks that are linked together with a cryptography hash function with replicated information stored at all the participants’ server. The data in the BT database is immutable. It can grow only by appending new block (data) at the end of the chain by authenticated users (miners) with strong cryptography capability can add the new block through a competitive mining scheme. Bitcoin is not blockchain. Bitcoin (BTC) is just one of the many applications utilising BT to support the Bitcoin cryptocurrency network which allows digital cash to be transferred within a distributed ledger. There are many other cryptocurrencies such as Ripple (XRP), Ethereum (ETH), Bitcoin Cash (BCH), Litecoin (LTC) and Binance Coin (BNB). In BTC, users ownership of a BTC can be digitally signed and transferred to another BTC user. The BTC blockchain announces this transfer publicly to all the network users to independently verify the transaction’s validity; moreover, a distributed group of users independently manage and maintain the BTC blockchain, and this together with cryptographic mechanisms creates BT’s non-repudiation capability towards attempts to modify the ledger by counterfeiting the transaction or altering the blocks.

There are three main types of BT: private, public or permissionless, and federated or consortium blockchain. Private and consortium blockchains are both considered as permissioned; a permission management entity is required to grant access rights to trusted and known participants. Examples of private blockchain include Multichain, Monax and Quorum. A consortium blockchain is controlled by more than one organisation. The group of organisations that control the consensus mechanism have predetermined nodes in the network. Examples are Ripple, R3 (Banks) and B3i (Insurance). In contrast to the previous two types, public blockchain allows anyone can write or read the data stored in the blockchain network without any permission from any authority, and the operation is entirely
decentralised and anomalous. Some examples are Monero, Etherum and Bitcoin. Public blockchain often uses a consensus-based system. Consensus mechanisms determine which user submits the next block. Consensus mechanisms are designed to allow distrusting users in a blockchain network to cooperate. Many consensus mechanisms have been used in BT, and these include proof of work (PoW), proof of stake (PoS), round-robin, proof of authority (PoA) or proof of identity (PoI) and proof of elapsed time (PoET).

Figure 1 shows the BT categorisation of private, consortium, public and permission versus permissionless. The categorisation is based on how many organisations are involved in maintaining the ledger and validated trusted participants are required or not.

Figure 1. Blockchain Categorisation

2.1. Blockchain Consensus Mechanism

The PoW mechanism is based on solving a computationally intensive puzzle that requires carrying out resource-intensive computations. A user who can solve this puzzle first can publish the next new block in the blockchain network. The checking of the correctness of the solution is done by all other miners to verify any new blocks before adding them to the blockchain and reject any block that does not satisfy the solution. In the PoW mechanism, it is hard to carry out a denial-of-service attack by flooding the blockchain network with malicious blocks. In theory, PoW enables an open non-monopoly environment for all participants to contribute; however, due to the difference in computational power and cost of electrical power, this creates an unfair situation among users. The PoW implementation domain is public cryptocurrency, such as Bitcoin and Ethereum.

The PoS consensus mechanism is based on investors in a network. The likelihood that a PoS user will publish a new block in the network is based on the percentage of their stake in the total staked cryptocurrency in the blockchain network. When they invest in a huge stake, which is usually an amount of cryptocurrency, the possibility that they will destroy the network diminishes, and they will be more likely to help the system succeed. When the cryptocurrency is staked, the user will not be able to spend it. A user’s stake is used by PoS blockchain networks to determine who can publish new blocks. The PoS mechanism has several advantages, one of which is that, compared to PoW, it is less computationally intensive. The PoS mechanism allows any interested user to stake on cryptocurrency; besides that, the system that uses PoS is controlled by stakeholders, although this is
sometimes considered a disadvantage in case they format a pool to create a centralised power base. In contrast, the PoS mechanism is known for its unique issues, such as the ‘nothing at stake’ problem. This problem arises when at some point, due to a temporary ledger conflict, multiple blockchains that are competing against one another appear. The appearance of these blockchains will cause different results from different blockchain versions to be published at almost the same time. In this case, the staked users can perform on every chain to increase their reward earning likelihood, which might grow in different branches in the blockchain network for a period without rejoining a single branch. Additionally, the PoS mechanism is vulnerable to a 51% attack by having sufficient financial power. The PoS implementation domain is public cryptocurrency, such as Casper and Krypton.

The round-robin mechanism allows each user takes his or her turn to create the next block, which might include a time limit for each user to avoid halting the publication of blocks. This mechanism is used by some private and federated blockchain networks. This mechanism prevents users from making the blocks plurality, which is easy to understand and does not require high computational power due to a lack of cryptographic puzzles. On the other hand, this mechanism requires a considerable amount of trust between users; therefore, it is useless in the public blockchain domain, which dominates the majority of current blockchain implementations. The great need for trust in this mechanism is because a malicious user might increase the number of users contributing to the network to raise the likelihood of publishing malicious new blocks to ruin the network. The round-robin implementation domain is a private and federated blockchain, such as MultiChain.

The PoA mechanism is based on publishing users’ partial trust, which relies on their real identities which must be verified, proved and included on the blockchain network. The central idea behind this mechanism is that a publishing user’s reputation or identity is being staked to publish new blocks. A publishing user’s reputation is affected by other users in the network based on his or her behaviour. If the blockchain network users disagree with the actions of a publishing user, which must adhere to the agreed manner on the network, the user will lose reputation, minimising the user’s chance of publishing a block. The confirmation time in this mechanism is fast, and the block production rates are dynamic. Moreover, this mechanism can be used in hybrid systems, which draws on another consensus mechanism. In contrast, this mechanism assumes that the validating user is not compromised. Additionally, the user’s reputation is subject to possible high-risk exposure that can compromise the user at any time. This mechanism also leads to a single point of failure. The implementation domain of proof of authority or proof of identity mechanism is a private and federated blockchain and includes hybrid systems such as Ethereum, Kovan Testnet and POA Chain.

The PoET mechanism designates each publishing user requests a wait-time within the network from a hardware time source, which is secured and installed in the user computer system. In this mechanism, each publishing user requests wait time within the network from a hardware time source, which is secured and installed in the user computer system. The hardware produces a random wait time for the publishing user, and the user becomes inactive during the period of the wait time. When the publishing user is reactivated, the user creates and publishes a block on the network, and all the users who are in the inactive state will stop waiting, and the whole process will begin again. In this mechanism, the given wait time must be random; otherwise, the malicious user might dominate the system by keeping the wait time at a minimum amount. Besides, the publishing user in this mechanism must not start early by waiting for the actual given time. These challenges are resolved by executing an application in a trusted processing environment, such as Intel’s Software Guard Extensions, which cannot be modified by external applications. This mechanism is less computationally expensive than PoW; however, the disadvantages of this mechanism are that it relies on hardware to produce random time and assumes that the hardware is not compromised. The implementation domain of the PoET mechanism is a private and federated blockchain, such as Hyperledger Sawtooth.
2.2. Blockchain Block

A block consists a header and a list of transactions and events ledger. Figure 2 shows the structure of a block in a blockchain. It is explained in the following:

2.2.1. Block Header

The block header consists of three different sets of metadata:

- Version and Previous Block Hash
  
  The Version field (4 bytes) is for tracking of software and protocol updates. The Previous Block Hash field (32 bytes) is a reference to the hash of the previous block. The cryptographic hash algorithm is by applying SHA256 twice each time. In a blockchain, every block is linked from the previous block. The previous block’s hash is used to create the new block’s hash. The first block in the blockchain is known as the genesis block.

- The Mining competition scheme

  This set of metadata contains the Timestamp (4 bytes), the Nonce (4 bytes) and the Difficulty Target (4 bytes). The timestamp is the creation time of the block. The nonce ("number only used once") is a number added to a hashed block that, when rehashed, meets the difficulty target level restrictions. The nonce is the first number a blockchain miner needs to discover before solving for a block in the blockchain.

- The Merkle Tree root (32 bytes)

  This is the hash of the root of the Merkle tree which contains a data structure of the transactions in the block. The Merkle tree root is created by repeatedly hashing pairs of transaction nodes until there is only one hash left. The process is performed from the bottom up, from hashes of individual transactions.

2.2.2. Block Data

A Merkle tree is a digital fingerprint of the entire set of transactions in a block enabling a user to verify whether or not a transaction is included in a block. It helps verify that later versions of the
event log include all the previous version and that all data is recorded and presented in chronological order. Block data contains a ledger of events and transactions list included inside the block as well as any other data for all the authentic and validated transactions which have been published on the blockchain network.

2.2.3. Chaining of Blocks

The adding of a new block is performed by the miners. A block in the chain may come from any miner who gets the right to add the new block. The miner picks up the hash of the last block in the chain, combines it with its own set of messages and creates a new hash for the newly created block. This newly created block now becomes the new end for the chain. With this, transaction integrity and non-repudiation is guaranteed as it can be rejected and detected of any altered blocks. Figure 2 shows the generic chaining process in BT.

2.3. Smart Contracts

Ethereum was launched in 2015 and the virtual coin "Ether" was proposed as the Ethereum cryptocurrency [14]. Similar to fiat currencies, Ether values and balances have standardized denominations for smaller units such as Wei, Kwei (1K Wei), Mwei (Mega Wei), and one Ether is (10 ** 18 Wei). Ethereum miners maintain the state of the network and resolve possible conflicts due to for example attacks or failure by using a consensus mechanism. The current consensus mechanism used in Ethereum is PoW consensus which relies on the assumption that miners are willing to follow the mechanism rather than attacking it because the mechanism will pay them for performing the computation needed to maintain the network and users will pay execution fees for every transaction [8]. Ethereum supports all types of computational structures including loops which can run any decentralized application (dApp) programming code utilising the back end Ethereum "smart contracts".

Smart contracts are digital contracts that are executed by themselves when precise circumstances are met and can be developed and implemented on top of Ethereum blockchain. The smart contract can symbolize the digital property ownership and enable everything of value such as shares, property and money to be exchangeable [15]. All the transactions are stored on the Ethereum blockchain and the transactions sequence locates every user’s balance and every smart contract’s state on the blockchain. The state of smart contracts comprises the balance which contains the amount of Ether they hold and private storage which is 256-bit values and a key-value store with 256-bit keys. The storage is private means that it cannot be modified or read by other contracts. A transaction deploys the Ethereum Virtual Machine (EVM) bytecode to the Ethereum blockchain to create the primary state with the constructor and to transfer the code of the contract. Users of smart contracts utilize a contract-invoking transaction with the address of the target smart contract as a recipient to invoke the smart contract once the smart contract is deployed. Every smart contract obtains an independent address to interact within the Ethereum network. The contract is stored on the ledger if both the smart contract initiation and the deployment transaction succeeded. There are two approaches to make one contract interact with another contract. The first approach is one of users directly create transactions to the second contract which has a known address carrying the contract required shape. The second approach is one of the users makes a new instance of the second contract by creating a new contract account with the same functionality of the contract class [16].

In Solidity, several primitives are provided to access the information of the block and the transaction. For instance, msg.value is used to access the Wei amount transmitted by a transaction invoking the method. Another example is msg.sender is used to access the account address which invoked the method. The exact signature of the smart contact’s function must be indicated if anyone wants to call a specific function in the smart contract. Any smart contract has a fallback function that
handles the request from transactions that indicate incompatible or no function. The transaction will be executed by executing the code of the smart contract in the contract instance context and every instruction will consume a predefined amount of gas. The transaction’s sender sets a gas limit and if the gas limit exceeded or a runtime error occurred the whole transaction is cancelled and the ledger will not be affected except the sender will lose the used gas. The transaction is handled as an exception if the gas finished before the transaction attains a regular stopping point. In case of one smart contract sends a message to another contract, a portion of the sender gas can only be offered to the receiver. If the gas finished from the receiver, the control will be returned to the sender who can utilize its remaining gas to straighten and treat the exception [17]. The following Program Listing 1 shows a simple smart contract program:

```
pragma solidity >=0.4.0;
contract Aeth {
    // "public" makes variables accessible from other contracts
    address public minter;
    mapping (address => uint) public balances;
    // Events allow clients to react to specific
    // declare contract changes
    event Sent(address from, address to, uint amount);
    // Constructor to create the contract
    constructor() public {
        minter = msg.sender;
    }
    // Sends an amount of newly created ETH to an address
    // To be called by the contract creator
    function mint(address receiver, uint amount) public {
        require(msg.sender == minter);
        require(amount < 1e60);
        balances[receiver] += amount;
    }
    // Sends an amount of existing ETH
    // from any caller to an address
    function send(address receiver, uint amount) public {
        require(amount <= balances[msg.sender], "Insufficient balance.");
        balances[msg.sender] -= amount;
        balances[receiver] += amount;
        emit Sent(msg.sender, receiver, amount);
    }
}
```

3. Cybersecurity Threats and Incidents on Blockchain Network

We have identified 65 real-world cybersecurity incidents occurred between 2011 and first half-year 2019 that have adversely impacted blockchain systems. We calculate the impact figures reported from the source which are based on the price of the lost coins at the time the attacks were discovered. The reported cases may not be complete since our research is based on publicly available information on forums, news feeds and other journal articles. Most incidents are lacking in details about the real circumstances surrounding the incidents. Thus, we provide a high-level classification of three types, namely hack, scam and smart contracts flaws. The total impact of the cybersecurity incidents between 2011 and 2019 has been more than US$3 billion. The highest loss relates to hacking, which is equal to
more than US$1.6 billion followed by the scam, which is equal to more than US$1.1 billion and smart contracts flaws, which is equal to more than US$289 million.

Figure 3. Blockchain Incidents 2011 to 2019 (6 Months)

Figure 3 shows on average, there are seven incidents per year gradually decreased to 2015 reached the bottom with two incidents only. This drop coincides with the drop in the price of Bitcoin in that year. The price of Bitcoin gradually climbed up after 2015 and so as the number of attack incidents which increased gradually to twelve incidents in 2018. The amount of loss to the incidents followed the same trend from US$7 million in 2015 increase to a peak of US$1.6 billion in 2018. The first six months in 2019 has already reached seven incidents with the amount loss is US$131 million only. However, we believe this number will increase in the second half-year of 2019.

Figure 4. Top Ten Most Affected Blockchain Incidents

Figure 4 shows the top ten cybersecurity incidents that had happened on the blockchain networks between 2011 and 2019 in terms of financial loss. It shows the highest loss was due to a Ponzi Scam suffered by victims in Vietnam as many as 32,000 investors investing in Ifan and Pincoin in April 2018 with a loss of total US$660 million [12]. Victims were unable to withdraw their profits in cash. It is a typical pyramid scam which boasts the investment as a “risk-free activity” with profits of “up to
40% monthly” through a range of bonus structures that allow early investors gain profits over later ones. The second-highest loss is due to Coincheck suffered from an external hack into its system in January 2018 for a record 530 million loss in cryptocurrency [18]. Coincheck provides bitcoin wallet and exchange service with 68% dominance in the Bitcoin market. The Coincheck staff failed to implement the exchange’s NEM multisig contract security feature recommended by the NEM developers and stored all of the NEM blocks in a single hot wallet. It is recommended for exchanges today to use a hybrid hot/cold wallet system, which stores the majority of the value in the cold wallets and secured via multisig contract security.

The third highest loss is the Mt. Gox hack happened in February 2014 with a loss of US$480 million, and another US$27.4 million missing from its bank accounts. Hackers had broken into the Mt. Gox exchange and had taken a huge portion of the cryptocurrency money controlled by the company. The incident had caused Mt. Gox into insolvency [15]. Mt. Gox was the largest bitcoin exchange in the world at that time which had suffered three separate attacks in June 2011 (US$8.75 million), October 2011 (US$8.35 million) and February 2014 (US$480 million). The failure of Mt. Gox was ascribed to poor management, lack of software development and security control. Transaction malleability is a vulnerability in the Bitcoin blockchain network, which enables the adversary to alter the transaction identifier (TXID) without revoking the transaction. Modifying the TXID will deceive the victim into believing that the transaction has failed, although it is later confirmed. Currency exchanges are the common targets for this attack. The adversary withdraws from an exchange and then republishes the same transaction with a different TXID, and one of them will show on the network. Because of delays, it is highly probable that the altered transaction will win rather than the original withdrawal. The currency exchange will not locate the original transaction on the network and will think that the transaction has failed if the exchange relies on TXIDs only. Thus, the adversary can continuously withdraw In the Mt. Gox attack; the attackers performed a transaction malleability attack to steal coins from the exchange, which forced the exchange to freeze users’ account and halt withdrawals [19].

Table 1. Blockchain Incident Classification

<table>
<thead>
<tr>
<th>Type</th>
<th>No. Incident</th>
<th>Total Amount Loss ($ million)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hack</td>
<td>48</td>
<td>1,621</td>
</tr>
<tr>
<td>Scam</td>
<td>10</td>
<td>1,126</td>
</tr>
<tr>
<td>Smart Contract Flaws</td>
<td>7</td>
<td>289</td>
</tr>
</tbody>
</table>

Table 1 shows hacking fraud tops with 48 incidents with a total loss of $1.6 billion. Based on the information from the source, hack vectors include compromising computer system, compromising cloud service, compromising email account, compromising private key, compromising third party, compromising server, compromising website, compromising cold and hot wallet, compromising platform, phishing attack, social engineering, masquerade attack, malicious insider and SIM swapping attack. Coinbase successfully detected and blocked what would have been a hack on June 20th. The hackers are believed to have exploited a Firefox zero-day bug, targeting employees by spear-phishing [20]. There are different types of flaws in blockchain users’ software, such as runtime, concurrency, memory, security, performance, configuration, GUI, compatibility, build and hard fork flaws. Flaws in the blockchain users’ software, which is used in the blockchain network, might lead to the exposure of users’ private keys. In 2014, Blockchain.info, which is a hybrid wallet provider, made a mistake during their software update in that when their users generated a new key pair on their local computer using the affected software, the ECDSA algorithm inputs were not adequately random, which caused an adversary operated the software to compromise the users’ private keys by only viewing the public address. There are 0.0002% of users affected, and the issue was detected and resolved within two and a half hours, although some Bitcoins were stolen. Software flaws might lead to a leak of users’ private
The number of Blockchain incidents due to scam is 10. The secrecy of identity property provided by blockchain has enabled itself to become the platform of choice for scams. The table shows the second-highest loss is due to scams such as the Ponzi scheme and Pyramid scheme causing one billion dollars in loss. Scam includes all the incidents from which the target owner disappeared with the funds.

Table 2. Blockchain Fraud Victim Classification

<table>
<thead>
<tr>
<th>Victim Type</th>
<th>No. Incident</th>
<th>Amount Loss ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bitcoin bank</td>
<td>1</td>
<td>502,029</td>
</tr>
<tr>
<td>Bitcoin payment service provider</td>
<td>1</td>
<td>1,800,000</td>
</tr>
<tr>
<td>Bitcoin stock market</td>
<td>2</td>
<td>6,741,039</td>
</tr>
<tr>
<td>Blockchain network</td>
<td>1</td>
<td>7,700,000</td>
</tr>
<tr>
<td>Cryptocurrency project</td>
<td>1</td>
<td>40,000</td>
</tr>
<tr>
<td>Blockchain project</td>
<td>1</td>
<td>500,000</td>
</tr>
<tr>
<td>Cloud services provider</td>
<td>1</td>
<td>228,845</td>
</tr>
<tr>
<td>Darknet market</td>
<td>1</td>
<td>100,000,000</td>
</tr>
<tr>
<td>Digital currency trading platform</td>
<td>1</td>
<td>4,100,000</td>
</tr>
<tr>
<td>Distributed autonomous organisation</td>
<td>1</td>
<td>74,124,000</td>
</tr>
<tr>
<td>Exchange</td>
<td>36</td>
<td>1,568,184,876</td>
</tr>
<tr>
<td>Individual</td>
<td>4</td>
<td>6,646,944</td>
</tr>
<tr>
<td>Margin trading service</td>
<td>2</td>
<td>441,760</td>
</tr>
<tr>
<td>Mining marketplace</td>
<td>1</td>
<td>64,931,534</td>
</tr>
<tr>
<td>Online poker room</td>
<td>1</td>
<td>15,543</td>
</tr>
<tr>
<td>Ponzi scheme</td>
<td>4</td>
<td>1,014,500,000</td>
</tr>
<tr>
<td>Pyramid scheme</td>
<td>1</td>
<td>2,300,000</td>
</tr>
<tr>
<td>Smart contract coding company</td>
<td>2</td>
<td>182,210,733</td>
</tr>
<tr>
<td>Stock exchange</td>
<td>1</td>
<td>10,000</td>
</tr>
<tr>
<td>Bitcoin stock market</td>
<td>2</td>
<td>2,200,000</td>
</tr>
</tbody>
</table>

Table 2 shows the victims of blockchain incidents range from individuals, bitcoin banks, Bitcoin service providers such as Wallet service, currency trading platform and exchange. Cryptocurrency Exchanges are the prime targets of attack. There are 36 incidents with a total of $1.56 billion loss. The transaction malleability vulnerability in BTC was the cause of the Mt. Gox incident in 2014. There are incidents where hackers make use of cryptojacking to install the malware in the target machine or mobile device to utilise their computational power to mine a block, which consumes a large amount of electricity and might compromise the target’s system functionality. In February 2018 alone, researchers launched a cryptojacking campaign, which affected more than 4000 websites, including the UK and US government pages; the other campaign targeted millions of Android devices. Additionally, Radiflow—a critical infrastructure security company—found cryptocurrency mining malware on the European water utility operational network, which had a huge impact on the systems [22].
4. Categorisation of Blockchain Threats and Vulnerabilities

Figure 5 shows a taxonomy of blockchain threats and vulnerabilities that we have identified so far. This taxonomy is built upon the findings from 65 real-world cybersecurity incidents occurred between 2011 and first half-year of 2019 and produce a classification scheme according to the threats vectors and vulnerabilities of BT.

We classify BT threats and vulnerabilities into the following five categories:

- Clients' Vulnerabilities
- Consensus Mechanism Vulnerabilities
- Mining Pool Vulnerabilities
- Network Vulnerabilities
- Smart Contract Vulnerabilities

Each of the threat and vulnerability types are discussed in the sections followed.

4.1. Clients' Vulnerabilities

4.1.1. Digital Signature Vulnerability

All Bitcoin asymmetric cryptography is based on Elliptic Curve Cryptography (ECC). The addresses in Bitcoin are derived from public keys of ECC, and the authentication of the transaction uses digital signatures, which are generated by Elliptic Curve Digital Signing Algorithm (ECDSA). The use of ECC is inadequate because it does not have the requisite randomness, which might compromise the user private key. A random value must be used with the private key to create a digital signature where the random value must be different for each transaction. For example, in Bitcoin blockchain, 158 unique public keys were found which used the same random value (nonce) in more than one signature, which made it possible to compromise the users’ private keys [23].
4.1.2. Hash Function Vulnerability

The operation in some of the blockchain networks, such as Bitcoin blockchain, relies on cryptographic primitives to ensure the correctness and accuracy of the operation. With the rapid development in the computational power and advanced cryptanalysis, these primitives have become breakable [24]. One of these primitives is the hash function. For example, SHA256 is the hash function used in Bitcoin blockchain, which is vulnerable to different cybersecurity threats, such as preimage and collision attacks [25]. A preimage attack is when the attacker is given an output Y from hashing an input m; the attacker attempts to find an input m* so that hashing m* equals Y; however, the attacker’s attempt to find two inputs providing the same hash is considered a collision attack [26]. The potential impact of performing the preimage attack on Bitcoin blockchain might lead to uncovering an address or the complete failure of the blockchain while the impact of the collision attack might be stolen coins or repudiate payment. Although enormous computation power is needed to perform such attacks, attacks might be possible if the adversary has quantum computing or dominates a huge mining pool [24].

4.1.3. Mining Malware

Cryptojacking is when the adversary installs malware in the target machine or mobile device to utilise their computational power to mine a block, which consumes a large amount of electricity and might compromise the target’s system functionality. In February 2018 alone, researchers launched a cryptojacking campaign, which affected more than 4000 websites, including the UK and US government pages; the other campaign targeted millions of Android devices. Additionally, Radiflow—a critical infrastructure security company—found cryptocurrency mining malware on the European water utility operational network, which had a huge impact on the systems [22].

4.1.4. Software’s Flaws

There are different types of flaws in blockchain users’ software, such as runtime, concurrency and hard fork flaws [27]. Flaws in the blockchain users’ software, which is used in the blockchain network, might lead to the exposure of users’ private keys. In 2014, Blockchain.info, which is a hybrid wallet provider, made a mistake during their software update in that when their users generated a new key pair on their local computer using the affected software, the ECDSA algorithm inputs were not adequately random, which meant that an adversary could operate the software to compromise the users’ private keys by only viewing the public address [21].

4.1.5. Users’ Addresses Vulnerability

Addresses in Bitcoin blockchain are vulnerable to identity theft threat because these addresses are not certified. For instance, a man-in-the-middle attack might be performed by an adversary to change the target Bitcoin address to the adversary address. The adversary might vandalise the target website to obtain payments destined for the target. The impact of the attack is disastrous because, in the Bitcoin blockchain, it is impossible to return the payment if the nodes in the network accept and register it in the ledger [28].

4.2. Consensus Mechanisms Vulnerabilities

4.2.1. 51% Vulnerability

Establishing mutual trust in BT is based on the shared consensus mechanism. However, the attackers might control the whole blockchain network by exploiting the 51% vulnerability, which is built in the mechanism. For example, if a single user or a group of users have more than 50% of the total hashing power in the blockchain networks, which are based on the PoW mechanism, then the user or the group of users can exploit the 51% vulnerability. Therefore, gathering the mining power
under a few mining pools might lead to this issue. Recently, GHash.io alone dominated 54% of all
Bitcoin network processing power for a day [19]. Additionally, the blockchain networks, which are
based on the PoS mechanism, also have the 51% vulnerability. The vulnerability can be exploited when
a single miner has more than 50% of the total coins in the network; a 51% vulnerability leads to a 51%
attack, which allows the attacker to do the following [29]:

- Injecting deceptive transactions,
- Manipulating the blockchain network,
- Outstripping all other users in the blockchain network,
- Performing a double-spending fund and
- Stealing other users’ assets.

4.2.2. Alternative History Attack

In this attack, the attacker sends a payment transaction to the target while he or she mines another
blockchain fork which includes a deceptive double-spending transaction. After the confirmation, a
product or service will be received by the attacker from the target. If the attacker succeeds in finding
more blocks than the genuine chain, he or she propagates his or her malicious fork and recovers the
coins; otherwise, he or she must extend his malicious fork to reach the fork of the honest miners. If the
attacker cannot catch up with the other nodes, the attack will fail [30].

4.2.3. Finney Attack

In this attack, one transaction is pre-mined in a block, and a duplicated version of this transaction
is sent to a user by the attacker. After the transaction is accepted and the product is delivered by the
receiver, the attacker propagates the block, which contains the initial transaction. Thus, the transaction
which is sent to the user will be invalid, and the attacker will succeed in producing a double-spend
transaction [31].

4.2.4. Race Attack

This attack is easy to launch in blockchain networks, which are based on the PoW mechanism;
this is mainly because an attacker can exploit the time between the creation transaction and the
confirmation transaction to carry out the attack. Before mining the confirmation transaction, the
attacker has obtained the creation transaction results, which leads to double spending [32].

4.2.5. Vector76 Attack

This attack originally came from the BitcoinTalk forums where a user named Vector76 described
an attack against the MyBitcoin e-wallet, which resulted in double-spending issues. In this attack, the
attacker needs not mine two consecutive blocks—mining one block is sufficient to perform this attack.
The attacker needs to observe the blockchain network to determine the timing of the propagating
transactions of network nodes and how they are broadcasting over the network. The attacker then
identifies the nodes that are earlier in the propagating transactions than the target and sets up a direct
connection with the target. After that, the attacker initiates a transaction that makes a legitimate
deposit into the target and mines it into a block without broadcasting it to the network. The attacker
mines the block like other nodes except that he or she adds an extra transaction that is not broadcast.
When the attacker succeeds in initiating a valid block, he or she does not broadcast it until some other
nodes mine a block. Once a node mines a block, the attacker immediately broadcasts his or her block
to the target, and if the target receives the attacker block before the other block, the target will accept
the attacker block, and the transaction will gain one confirmation. In this situation, the blockchain
and the target and other nodes connected to the target will be forked mainly because the target that
passed on the transaction quickly will consider the attacker block legitimate whereas the other nodes
in the network will consider the other fork valid. The attacker directly transfers the coins to a different
address that is controlled by the attacker, and the target will generate the transaction because the target believes that it is a legitimate transaction. The attacker also double spends the inputs by transferring the coin to himself or herself. The network nodes that did not receive the attacker first block will accept the transaction as a genuine transaction, and they will include it in the next block. If the attacker first block wins when the blockchain has forked, the attacker will not lose anything; however, if the first block loses, then the deposit to the target will become invalid, although the withdraw transaction will still be valid [33].

4.3. Mining Pool Vulnerabilities

4.3.1. Block Withholding (BWH) Attack

In this attack, the attacker joins a mining pool to assist the pool members in mining blocks; however, the attacker will never broadcast any block to decline the pool anticipated income. This attack is also called ‘a Sabotage Attack’ because the scoundrel miner does not obtain anything but causes everyone to lose [34]. Although the attacker does not gain any revenue from this attack, [35] assert that the attacker might be able to earn income from this attack.

4.3.2. Bribery Attack

This attack is based on bribing miners to mine on precise forks or blocks. The attacker can validate random transactions and publish them because he or she has paid to dishonest nodes to verify them. The attacker might gain the majority of the computing resources by using three ways of bribing, namely out-of-band payment, negative-fee mining pool and in-band payment. In the out-of-band payment, the computing resources owner is directly paid by the attacker to mine the attacker’s blocks. In the negative-fee mining pool, the attacker creates a pool by rewarding the higher return. Finally, in in-band payment, the attacker pursues to bribe the blockchain itself by making a fork which includes free bribe money to any miner endorsing the attacker fork [36].

4.3.3. Pool Hopping Attack

In this attack, the attacker mines based on the appeal rate. If the rate is high, the attacker mines, otherwise the attacker leaves the pool. The attacker utilises the information about the number of the submitted shares in the target mining pool to understand how many shares have been submitted and how many blocks have been found. Using this information, the attacker stops mining in the target pool and contributes elsewhere. The central idea behind this attack is that the attacker chooses various pools to mine to gain maximum income [34].

4.3.4. The Block Discarding Attack

In this attack, compared to the honest nodes, the attacker must possess an adequate number of network connections and dominate multiple slave nodes to increase his or her network superiority. Once the attacker is informed of newly mined blocks, he or she immediately publishes his or her own block, which must be faster than the rest in the network; therefore, when a node publishes a block, the attacker can instantly propagate his or her own blocks to discard honest nodes’ blocks [37].

4.3.5. The Selfish Mining Attack

In this attack, a group of attackers conspire to create a mining pool to negate the honest miners’ work and acquire better income for themselves. The attackers mine in their private blockchain and broadcast it based on the length difference between the public and the private blockchains to influence the rewards [38].
4.3.6. Fork After Withholding (FAW) Attack

The FAW attack income is equal or greater than a BWH attack income, and the attack is four times more fruitful, usually per poll, than the BWH attack. This attack has two types: single-pool FAW attack and multi-pool FAW attack. This attack combines the selfish mining attack and the BWH attack. In the single-pool FAW attack, the attacker joins the target mining pool and performs the attack against it whereas, in the multi-pool FAW attack, the attacker aims to increase his or her income by expanding the attack against several pools. The attacker computing power is divided in this attack into infiltration mining and innocent mining. When the attacker infiltration part locates a full proof of work (FPoW), the attacker keeps the block and does not broadcast it. Based on the next steps, the attacker might publish his or her private block to the manager of the target pool hoping that a fork is created identical to the selfish mining attack or the attacker discards the block which is identical to the BWH attack [39].

4.4. Network Vulnerabilities

4.4.1. Partition Attacks

In this attack, the attacker isolates a group of nodes from the rest of the Bitcoin blockchain network, and the network is partitioned into disjoint components. The adversary hijacks the most specific prefixes, which host each of the isolated nodes’ IP address to redirect the traffic destined to them. The traffic is intercepted by the adversary when he or she is on the path and determines which connections cross the partition that the adversary attempts to create. If the connection does not cross the partition, the adversary drops the packets; otherwise, the connection is contained within the isolated nodes. The adversary tracks the exchanged messages to determine the leakage points; these are nodes in the isolated group, which maintain connections with the external nodes and the adversary cannot intercept. The adversary finally isolates the leakage points from other nodes in the isolated group [40].

4.4.2. Delay Attacks

In the previous attack, the adversary needed to gain full control over the target’s traffic to perform the attack effectively. In contrast, the delay attack can cause significant delays in block publishing even when the adversary intercepts only one of the target’s connections. First, the attacker alters the content of specific messages to delay the block delivery; this is achievable because of a lack of integrity checks and encryption of Bitcoin messages. Additionally, the adversary makes use of the fact that nodes first send block requests to the peer that propagated each block and wait for 20 minutes to deliver it before requesting it from another peer. Thus, the adversary delivers a block to a target node by a 20-minute interval, which makes the target unaware of the most recently mined blocks and makes the target unable to contribute to the network [40].

4.4.3. Distributed Denial-of-Service (DDoS) Attack

Nowadays, the DDoS attack is one of the most common and inexpensive attacks on the Internet [19]. Despite being a peer-to-peer technology, BT is still vulnerable to the DDoS attack. BT networks, such as Ethereum and Bitcoin, have frequently undergone these attacks. For instance, 40 Bitcoin services have suffered from 142 DDoS attacks over two years, and the targets have included 7% of all popular operators [19]. Most of these attacks targeted large mining pools and currency exchange services because of a larger revenue possibility. These attacks have forced firms such as BitQuick and CoinWallet to shut down their service after a few months since their start [19].

4.4.4. Sybil Attack

In this attack, the adversary sets up fake assistant nodes and attempts to expose part of the blockchain network. The adversary might use a group of exposed nodes to perform the attack to
isolate the target and disconnect the transactions created by the target, or the attacker will make the user choose only the blocks that are maintained by him or her [19]. The adversary with malicious nodes will surround the target. The target will think that he or she still connects to the network through different honest nodes; however, the reality is that the target has limited access to the network because the adversary controls all the nodes to which he or she connects. Once the adversary surrounds the target, he or she can refuse to relay the target’s transactions. Besides, the adversary can feed misleading information to the target of the network state [21]. A successful Sybil attack can incapacitate the consensus algorithm functionality and cause potential double-spending attack [19].

4.4.5. Timejacking Attack

This attack is a specific attack on the Bitcoin blockchain network. The network time in this network is maintained by full nodes. The network time is acquired by obtaining a version message from the neighbouring nodes. The median is calculated, and if the median time of all neighbouring nodes exceeds 70 minutes, the network time counter returns to the node system time by default. When the adversary is connecting to the target node, he or she attempts to reveal imprecise timestamps. Once the adversary modifies the node network time counter, the misled node might adopt a substitute blockchain. This attack will isolate the target node from the network or decrease the transaction confirmation rate on the whole network [31].

4.4.6. Transaction Malleability Attack

Transaction malleability is a vulnerability in the Bitcoin blockchain network, which enables the adversary to alter the transaction identifier (TXID) without revoking the transaction. Modifying the TXID will deceive the victim into believing that the transaction has failed, although it is later confirmed. Currency exchanges are the common targets for this attack. The adversary withdraws from an exchange and then republishes the same transaction with a different TXID, and one of them will show on the network. Because of delays, it is highly probable that the altered transaction will win rather than the original withdrawal. The currency exchange will not locate the original transaction on the network and will think that the transaction has failed if the exchange relies on TXIDs only. Thus, the adversary can continuously withdraw [41]. Mt. Gox was one of the largest exchanges in Bitcoin history, which declared bankruptcy due to losing coins valued over US$450 million. The attackers performed a transaction malleability attack to steal coins from the exchange, which forced the exchange to freeze users’ account and halt withdrawals [19].

4.5. Smart Contract Vulnerabilities

Two of the major vulnerabilities of Ethereum are discussed in the following: Ethereum Virtual Machine (EVM) Bytecode Vulnerabilities and Ethereum Solidity Vulnerabilities.

4.5.1. EVM Bytecode Vulnerabilities

The EVM is a virtual machine that runs the bytecode, which is the result of compiling the source code of a smart contract. Each operation in the EVM expends a specific amount of gas. The gas represents the code execution cost. Table 3 shows three vulnerabilities under the EVM bytecode umbrella [42].

4.5.2. Solidity Vulnerabilities

Solidity is the smart contracts high-level programming language in Ethereum which the programmer uses to write the smart contract source code. There are six known vulnerability types in the smart contracts source codes, which are already exploited and represent the highest portion of the smart contracts’ vulnerabilities number. Most of these vulnerabilities emanate from a misalignment
Table 3. EVM bytecode vulnerabilities.

<table>
<thead>
<tr>
<th>Vulnerability Name</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ether lost in transfer</td>
<td>Any ether sent to an orphan address, which is not associated with any user lost forever, and there is no way to check if the address is orphan or not.</td>
</tr>
<tr>
<td>Immutable bugs</td>
<td>Published smart contracts are immutable against any alteration.</td>
</tr>
<tr>
<td>Stack size limit</td>
<td>The call stack limitation which can be leveraged by the attacker.</td>
</tr>
</tbody>
</table>

Table 4. Solidity vulnerabilities.

<table>
<thead>
<tr>
<th>Vulnerability Name</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>Call to the unknown</td>
<td>The recipient fallback function is invoked because the called function does not exist.</td>
</tr>
<tr>
<td>Exception disorder</td>
<td>Anomaly in exceptions handling.</td>
</tr>
<tr>
<td>Gasless send</td>
<td>Invoking the out-of-gas exception.</td>
</tr>
<tr>
<td>Keeping secrets</td>
<td>Compromise a private field in the smart contract.</td>
</tr>
<tr>
<td>Reentrancy</td>
<td>Re-enter a non-recursive function before its termination.</td>
</tr>
<tr>
<td>Type casts</td>
<td>No exception is thrown when the caller made an error.</td>
</tr>
</tbody>
</table>

between the programmers’ insight and the Solidity semantics [42]. These vulnerabilities are shown in Table 4.

Ethereum Smart Contract Coding flaws represents the third-highest impact of the incident type. The main cause is due to the reentrancy vulnerability in smart contracts as shown in the case of the DAO incident in 2016. Reentrancy is a kind of vulnerability exhibited in Ethereum Smart Contract only. As the name suggests, an attacker first deposits an amount X to a multi-party smart contract. The attacker then executes a function to withdraw an amount Y, which is more than X before the balance of funds deposited and withdrawn has been settled. The effect is the attacker essentially stealing the money of other parties in the contract. Our study reveals incidents related to Ethereum Smart Contract Flaw has risen from being one incident in 2016, two incidents in 2017 to four incidents in 2018.

5. Conclusions

This paper analysed 65 cybersecurity incidents and discussed some of the attack mechanisms in BT and have developed a taxonomy for blockchain attack issues have arisen because concerns in blockchain conceptualisation, blockchain network implementation, the functionality of smart contracts, the process of mining and consensus mechanisms. Some of these vulnerabilities might become obsolete in the future, but new ones may be discovered; however, we believe the taxonomy we have developed describes a more general set of vulnerabilities that are likely to persist into the future. Additionally, this paper also catalogued real cybersecurity incidents in the BT systems. The outcome of this paper provides a clear direction for BT designers and implementers guidance on how to implement a more secure BT system.

The followings are suggested for the future research direction and guidance on how to implement secure BT.

- No actors on the blockchain network should immune to attacks. Blockchain exchanges are the prime targets who suffered most of the attacks with a significant amount of loss. Apart from the NEM multisig security, enhanced cryptography techniques for integrity, dealing with malware and other active threats, are required in working towards future implementations of blockchain-based access control required for the exchanges and the network.
Blockchain network is being used actively for scamming nowadays. Thus, there is an upward trend in attacks targeting the Ethereum Smart Contract Flaw from 2016. We assert that the Ethereum smart contract vulnerability will be the next prime target for the attackers. Future researchers should improve blockchain by directly addressing these categories of vulnerability, through the use of more advanced context-aware access control systems working towards future implementations of blockchain-based access control for secure data access, and maintaining data sharing between different parties [43–45].

Fraud prevention and detection mechanisms are required as an add-on to BT to ensure trustworthy and privacy-preserving attributes of the users are maintained [46].

References

   in Practice. Financial Cryptography and Data Security; Christin, N.; Safavi-Naini, R., Eds.; Springer Berlin
24. Giechaskiel, I.; Cremer, C.; Rasmussen, K.B. On Bitcoin Security in the Presence of Broken Cryptographic
25. Hoch, J.J.; Shamir, A. On the Strength of the Concatenated Hash Combiner When All the Hash Functions
   Are Weak. Automata, Languages and Programming; Aceto, L.; Damgård, I.; Goldberg, L.A.; Halldórsson,
   616–630.
   Security; Boureanu, I.; Owesarski, P.; Vaudenay, S., Eds.; Springer International Publishing: Cham, 2014;
   pp. 80–96.
30. Mechkaroska, D.; Dimitrova, V.; Popovska-Mitrovikj, A. Analysis of the Possibilities for Improvement
35. Courtois, N.T.; Bahack, L. On subversive miner strategies and block withholding attack in bitcoin digital
   attacks on bitcoin consensus 2016.
37. Bahack, L. Theoretical Bitcoin attacks with less than half of the computational power (draft). arXiv preprint
   (FAW) Attacks on Bitcoin. Proceedings of the 2017 ACM SIGSAC Conference on Computer and
   IEEE Communications Surveys Tutorials 2016, 18, 2084–2123.
42. Atzei, N.; Bartoletti, M.; Cimoli, T. A Survey of Attacks on Ethereum Smart Contracts (SoK). Principles of
43. Kayes, A.; Han, J.; Rahayu, W.; Dillon, T.; Islam, M.S.; Colman, A. A policy model and framework for
   context-aware access control to information resources. The Computer Journal 2018, 62, 670–705.
44. Kayes, A.; Rahayu, W.; Dillon, T.; Chang, E.; Han, J. Context-aware access control with imprecise context