IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 20, 2018, accepted September 18, 2018, date of publication November 9, 2018, date of current version December 3, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2874539

Bitcoin Concepts, Threats, and Machine-Learning Security Solutions

MOHAMED RAHOUTI[1,2], KAIQI XIONG[2,3,4], (Senior Member, IEEE), AND NASIR GHANI[1,4], (Senior Member, IEEE)

[1]Department of Electrical Engineering, University of South Florida, Tampa, FL 33620, USA
[2]Intelligent Computer Networking and Security Lab, University of South Florida, Tampa, FL 33620, USA
[3]Department of Mathematics and Statistics, University of South Florida, Tampa, FL 33620, USA
[4]Cyber Florida, University of South Florida, Tampa, FL 33620, USA

Corresponding author: Kaiqi Xiong (xiongk@usf.edu)

This work was supported in part by the National Science Foundation (NSF) under Grants 1633978, 1620871, 1620862, and 1651280, and in part by the BBN/GPO Project 1936 through NSF/CNS Grant.

ABSTRACT The concept of Bitcoin was first introduced by an unknown individual (or a group of people) named Satoshi Nakamoto before it was released as open-source software in 2009. Bitcoin is a peer-to-peer cryptocurrency and a decentralized worldwide payment system for digital currency where transactions take place among users without any intermediary. Bitcoin transactions are performed and verified by network nodes and then registered in a public ledger called blockchain, which is maintained by network entities running Bitcoin software. To date, this cryptocurrency is worth close to U.S. $150 billion and widely traded across the world. However, as Bitcoin's popularity grows, many security concerns are coming to the forefront. Overall, Bitcoin security inevitably depends upon the distributed protocols-based stimulant-compatible proof-of-work that is being run by network entities called miners, who are anticipated to primarily maintain the blockchain (ledger). As a result, many researchers are exploring new threats to the entire system, introducing new countermeasures, and therefore anticipating new security trends. In this survey paper, we conduct an intensive study that explores key security concerns. We first start by presenting a global overview of the Bitcoin protocol as well as its major components. Next, we detail the existing threats and weaknesses of the Bitcoin system and its main technologies including the blockchain protocol. Last, we discuss current existing security studies and solutions and summarize open research challenges and trends for future research in Bitcoin security.

INDEX TERMS Bitcoin, blockchain, security, machine learning (ML), anomaly detection.

I. INTRODUCTION

Bitcoin was originally introduced in 2008. Since then, it has emerged as the most successful cryptographic currency among many competitors, boosting the economy with billions of dollars a few years after being launched. As a type of cryptocurrency, Bitcoin exists in the form of sets of computer codes that virtually hold a monetary value. With that context, all transactions and payments are accomplished over the Internet. Note that Bitcoin differs from traditional on-line banking, as it utilizes a peer-to-peer (P2P) network that does not associate with a centralized third-party organization, e.g., such as an e-bank, a notary, or any other traditional on-line financial service provider that conducts and approves electronic payments activities. Instead, Bitcoin users have full control of what they want to do with their own money by means of freely ordering how and when to use digital money without any constraints. Bitcoin is increasingly drawing public attention and moving more and more customers towards using this payment system in a variety of businesses. Fast, convenient, tax-free, and revolutionary are commonly used to describe Bitcoin.

However, things can never be perfect. In particular, the security, confidentiality and reliability of Bitcoin have also been controversial topics since they pose added vulnerabilities being away from consolidated governance and law enforcement. Additionally, to guarantee a reliable and trusted distributed system of monetary transactions, it is critically important for all Bitcoin holders and operators to have a safe environment of monetary operations as well as personal property protection. For the past several years, there have been several research studies that have considered anomaly detection for the Bitcoin system and these studies have considered a variety of techniques including, but not limited to, machine learning (ML) and network analysis methods. Along these lines, in our paper, we conduct an intensive survey study that focuses on ML-based techniques and approaches in the detection of threats and anomalous activities in the Bitcoin and blockchain systems.

As noted in [23], blockchain provides the fundamental framework for all kinds of Bitcoin's operations. In particular, it provides a novel decentralized consensus scheme that stores transactions, money transfers, and any other data records in a secure manner without any involvement from third party authorities. In Bitcoin, every transaction is broadcasted to all peers into the network and processed to verify its integrity, authenticity, and correctness by a group of nodes called *miners*. In particular, instead of mining a single transaction, miners bundle a number of transactions that are waiting in the network to get processed in a single unit called a *block*. A miner then advertises a block across the whole network as soon as it completes its processing (or validation) in order to claim a mining reward. This block is then verified by the majority of miners in the network before it is successfully added in a distributed public ledger called a *blockchain*. The miner who mines a block receives a reward when the mined block is successfully added in the blockchain.

Since the Bitcoin blockchain is a distributed (non-centralized) system, it does not require any permission from a trusted third party (TTP) to handle Bitcoin transactions. Specifically, networking nodes can communicate with each other in a collaborative manner to establish the blockchain without any central authority. However, a single entity can still crash or even behave abnormally. Such a crash or abnormality may lead to communication interruption. Therefore, in order to guarantee an uninterrupted communication service, all entities need to run a fault-tolerant consensus protocol to guarantee that they all agree on the order in which entries are pushed to the blockchain.

As the Bitcoin system and its network infrastructure have been proven vulnerable to a tremendous amount of malicious activities and attacks in the past [67], there are numerous existing studies on dealing with particular security challenges in Bitcoin and blockchain systems such as [51], [52], and [68]. These studies range from anomaly detection to market return and volatility forecasting in the Bitcoin system. Several studies have also focused on utilizing ML techniques for anomaly detection in Bitcoin networks, such as fraud detection and anomalous activities/transactions. For example, Pham and Lee [51] utilize unsupervised learning methods to deal with anomalous activities (i.e., transactions) in Bitcoin systems. In order for us to fix or even isolate malicious parts of the network until they are debugged, various ML techniques, such as support vector machine (SVM) and clustering, can be deployed to help with identifying those parts that behave abnormally.

In this work, we conduct an intensive survey that mainly focuses on the deployment of ML techniques for security threat detection and/or mitigation in Bitcoin and blockchain systems/infrastructures, along with an analysis of their related concepts. Moreover, we further discuss and investigate existing work and state-of-art threat vectors, classify them, and present their limitations whenever applicable. These attack vectors embody a variety of abnormal user behaviors and anomalous Bitcoin transactions that threaten smooth functionalities and operations in real-time monetary services and applications. Furthermore, we study and present some common existing ML-based solutions and countermeasures to combat serious anomalous activities and threats with regard to the major components of Bitcoin and blockchain. Finally, we discuss future research trends and related open security research problems. We also discuss the effectiveness of various security proposals and efforts that have been introduced in the past several years to solve existing or common security challenges for the Bitcoin infrastructure.

The rest of our survey paper is organized as follow and it is also shown in Figure 1. Section II presents an explanatory and detailed overview of Bitcoin along with its functionalities. Section III then gives a taxonomic discussion of vulnerabilities that threaten the security, implementation, and resiliency of the Bitcoin network and blockchain. In Section IV we present ML-based state-of-the-art studies and efforts aiming to counteract a specific security threat or even improve the current resiliency in the Bitcoin system. Finally, Section V provides an overview of future research trends and directions. We then conclude our surveying study in Section VI.

II. BITCOIN INFRASTRUCTURE AND DESIGN

In this section, we discuss the overall design and infrastructure of the Bitcoin system from a contextualized overview and a technical overview as follows.

A. A CONTEXTUALIZED OVERVIEW

We refer interested readers to existing surveys on the wave of cryptocurrency research [13], [15]. In 1983, cryptographic currency was first introduced as a system for bank-issued cash, where coins were blindly signed and unblinded version of coins were conveyed among clients and traders. These coins were redeemed once the bank validated them [21]. Similarly, the Bitcoin system also granted blind signatures to block Bitcoin-enabled banks from binding coins to clients [13]. All the way through the 1990s, several infrastructural extensions and adjustments to the Bitcoin system were introduced [13], including, but not limited to, removing the requirement for a Bitcoin bank to be on-line during a purchase operation [22] and the period of enabling the splitting of coins into small units [50].

In the early 1990s, the concept of a smart contract was introduced [63] to allow different parties in the Bitcoin system to officially appoint a cryptographically-mandatory agreement and foresee the scripting capabilities of Bitcoins.

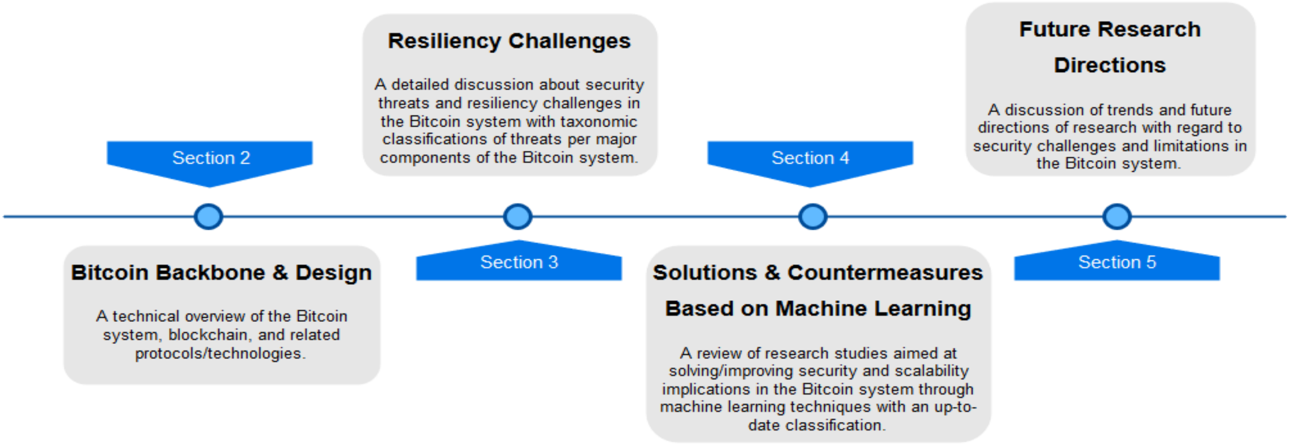

Section 2
Bitcoin Backbone & Design
A technical overview of the Bitcoin system, blockchain, and related protocols/technologies.
Resiliency Challenges
A detailed discussion about security threats and resiliency challenges in the Bitcoin system with taxonomic classifications of threats per major components of the Bitcoin system.
Section 3
Section 4
Solutions & Countermeasures Based on Machine Learning
A review of research studies aimed at solving/improving security and scalability implications in the Bitcoin system through machine learning techniques with an up-to-date classification.
Future Research Directions
A discussion of trends and future directions of research with regard to security challenges and limitations in the Bitcoin system.
Section 5

FIGURE 1. **Overview of the paper.**

The final major revolution took place in 2008, where Bitcoin was announced and an unknown white paper was posted under the pseudonym *Satoshi Nakamoto* in the *Cypherpunks* mailing list [49]. This release included the source code of the authentic reference client as well. Next, the original block of the Bitcoin system was mined early in 2009 and the first official release of the system is believed to have taken a place in May 2010, where a user placed an order for a pizza delivery by trading off 10000 bitcoins. After this stage, the number of users started to dramatically increase on a daily basis as well as reliable services, such as goods trade-off on site.

Precisely, the Bitcoin system is taxonomically regarded as a network of nodes that continuously maintain an electronic ledger called blockchain [64]. This system utilizes tokens called *bitcoins* to inveigle and motivate users/clients to participate via transactions. New bitcoins are minted whenever a new block is generated in an incentive form, which is important to drive the system of transactions [39]. These transactions are to store Bitcoin owner information, along with the amount of bitcoins (i.e., funds) stored in the blockchain [64]. Moreover, each user has the ability to check the content of the blockchain that is cloned on all network nodes.

B. A TECHNICAL OVERVIEW

1) DECENTRALIZATION

Bitcoin is the very first distributed crypto-currency system and it is a fully decentralized digital currency system where the monetary power is not controlled by any party [35]. Imparting from the washout of a centralized economic system, the unknown Bitcoin inventor created it in a decentralized manner. However, this decentralized architecture still has several major limitations [54]:

- The transactions ledger needs to be publicly preserved by every single node.
- Ledger transactions need to be checked and legitimized by a distributed entity but not by a centralized authority or party.
- Unlike centralized economic systems, new bitcoins can be generated by any connected entity.
- Exchange operations of bitcoins' values are completely dynamic and no centralized control is required to handle such operations.

Bitcoin is a distributed digital currency system based on a P2P networking system and a decentralized probabilistic consensus protocol where all payments and exchanges are accomplished electronically through the transaction between clients [46]. Addresses in the Bitcoin system are also created through the execution of consecutive irreversible cryptographic hash schemes using users' public key. Namely, a single client can generate several public keys in order to have more than one addresses where each single address will be assigned to one or more wallets [23]. The client's private key is necessary to expend bitcoins through digitally-signed bitcoin transactions. These ledger transactions are also performed to check and validate their authenticity and integrity by a set of distributed networking-enabled entities, i.e., miners. These miners are in charge of bundling numerous ledger transactions, which await to access the network to be performed by a single entity/unit named a *block*. In order to announce a mining reward, a miner node will declare a block in the Bitcoin network once the block processing is completed and validated. Next, the rest (or majority) of networking-enabled miner nodes in the system will check and validate this mining block before it is appended to a publicly decentralized ledger named a *blockchain*. The block winning miner node will obtain a reward once the mined block is completely appended to this blockchain.

2) TRANSACTIONS AND SCRIPTS

Bitcoin transactions are used to transfer digital coins between different client wallets. Specifically, these coins are transferred in form of a transaction or consecutive series of transactions, as depicted in Figure 5. Overall, the list of transactions is continuously increasing and there are no built-in higher-level concepts in the system to manage the balances of active accounts or even the identities of clients. Instead, this information can only be imputed from the entry list of published transactions.

- Transaction format: Each Bitcoin transaction has a multi-dimensional list or an array of input entries and an array of outputs entries as shown in Figure 4. The transaction is entirely hashed by the *SHA-256*, and the produced hash value basically serves as a unique global identifier of the transaction. Next, the transaction is advertised by an ad-hoc-based binary format. Moreover, the output entries constitute a set of integers which reflect the amount of Bitcoin currencies. These output entries also constitute a concise code in the form of a particular scripting language named a *ScriptPubKey*, which reflects the parameters required to validate the redemption of transactions, which will be appended to a later transaction input.
- Transaction Script: In the transactional context, the *ScriptPubKey* appoints the hash of a public key using an Elliptic Curve Digital Signature Algorithm (ECDSA)-based public key along with a signature validation routine. Briefly, as shown in Figure 3, *ScriptPubKey* is a short script illustrating what conditions need to be fulfilled to purport the ownership of bitcoins. An example of *ScriptPubKey* is given in Figure 3. This entire operation is referred to as a *pay-to-pub-key-hash* transaction, while the redeeming transaction also needs to be signed by a key with this appointed hash. Furthermore, the deployed scripting language is specified by its implementation in *bitcoind*, an ad-hoc-based stack language that has about 200 commands named *opcodes*. This language supports cryptographic-based operations, such as hashing information and validating signatures. Furthermore, this scripting language also allows transaction input entries to point to previous transactions using their corresponding hash. Now in order to claim a redemption of a previously-validated transaction, both *ScriptSig* and *ScriptPubKey* need to be executed receptively through the same stack. However, for transactions of the form *pay-to-pubkey-hash*, *ScriptSig* will be a public key combined with a signature.
- From transaction to ownership: To better understand the transactional process in the Bitcoin system as shown in Figure 2, we give an illustrative example here. Suppose client *i* wants to transfer *n* bitcoins to client *j*. In order to make a payment to client *j*, client *i* must utilize a Bitcoin client-side software as well as its own private key and client *j*'s Bitcoin address. Although any client in a Bitcoin network might transfer funds (i.e., money) to a particular a Bitcoin address, bitcoins are released from the account only by a unique signature generated with a private key. In this case, in order for client *i* to prove ownership of those transferred coins, client *i* needs to sign the transaction with its cryptographic key. Once client *i* broadcasts the transaction, all miners in the Bitcoin network will be informed about the new transaction. Last, the miner nodes must verify that client *i* has an enough funds to complete this transaction and validate the correctness of the digital signature. Moreover, the transaction format has a few key properties. Foremost, there is inherently no ingrained concept of identity or no singular client account that own the bitcoins. Ownership in the Bitcoin context solely refers to knowing a private key to grant the capability to make a signature for a specific output redemption (i.e., redemption validation). Furthermore, public key hashes are regarded as pseudonymous client identities, as specified in pay-to-pub-key-hash transactions. These client identities are also referred to as addresses, where no authentic names or real identifications are needed. Overall, since the Bitcoin system integrates cryptographic applications along with P2P networks to guarantee a decentralized digital cryptocurrency environment, it preserves a complete transaction history within the public blockchain. Therefore, Bitcoin clients can be exposed to the leakage of financial information. However, this is mainly due to existing approaches and solutions that aim at de-anonymizing and matching real user identities with the public transactions history [17].

3) BLOCKCHAIN CONSENSUS PROTOCOL AND MINING

Blockchain is basically a public and append-only-based structure that saves a transaction history in the distributed Bitcoin system in the form of individual blocks by using Merkle tree (along with a secure timestamp and the hash of a previously validated transactions block). A new block is successfully appended to the blockchain only if miner nodes validate them through solving a difficult *proof-of-work* (PoW) puzzle. The blockchain also allows for traversing these blocks to find the ownership of a Bitcoin in an efficient way as blocks are saved in a given order. Moreover, manipulating blocks is impossible because tampering with only one block will modify the entire hash value of the block and the tampering will be detected since each single block contains the hash value of the previous one [15].

On another hand, in a blockchain, the block validation process has a distributed nature, which implies that more than one valid solution can potentially be found at the same time. However, this will result in valid blockchain forks with a similar length. Now when more than one fork occurs, miner nodes will be enabled to pick up a fork and continue to mine on top of it. Since this situation might be common, a blockchain with a longer version is likely to exist in the network (and the rest of the miners nodes will eventually begin building their corresponding blocks on the top of this longer version, as depicted in Figure 5).

Furthermore, since the mining process in the network is of a continuous nature, the blockchain will always increase in size. Therefore, the operation of appending a new block will be as follows: (1) a miner node will append the new block in its local blockchain and advertise the solution once the valid hash value is determined (i.e., its hash value is same or smaller

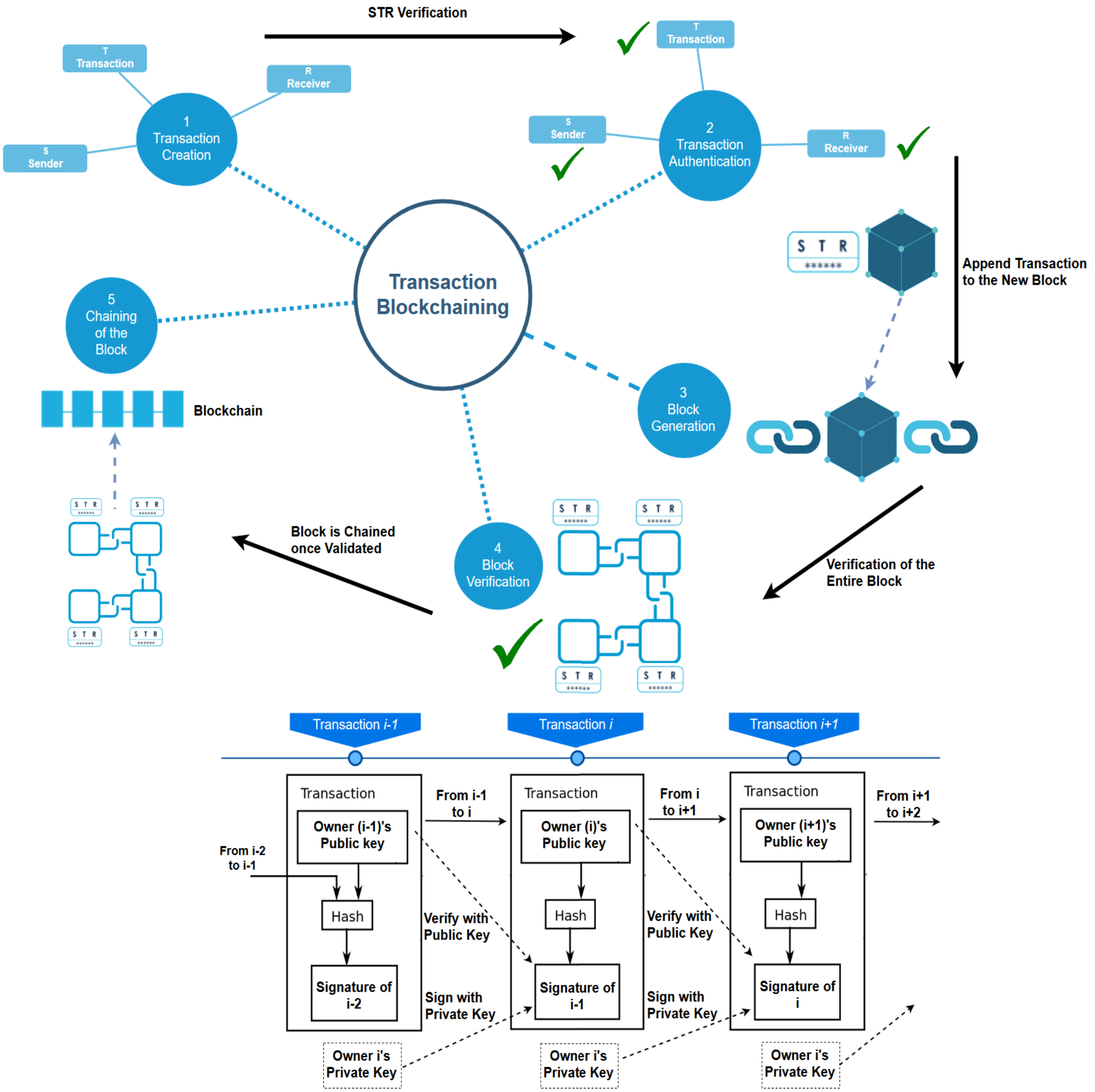
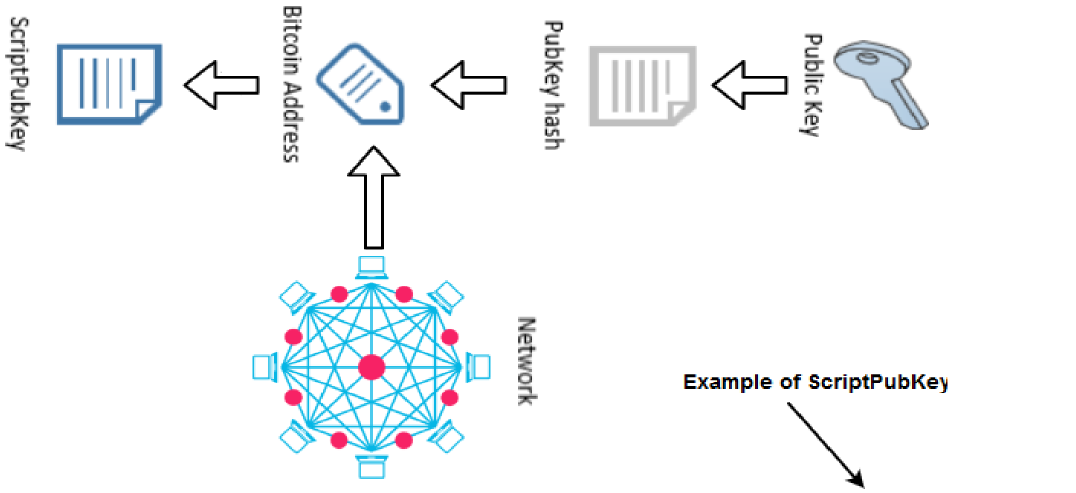


FIGURE 2. Creation and addition of blocks and a transactional process in a blockchain.

than the target's hash value), and (2) miner nodes will rapidly verify the received solution of a valid block and update their local blockchain if the advertised solution is valid; otherwise, it will be discarded. Regarding rewards, a winning miner node will always be rewarded with a set of newly-minted bitcoins as an *incentive* while the winning hashed block will be advertised and published in the public distributed ledger.

Due to the decentralized nature of the Bitcoin blockchain, authorization is not imposed on any TTP prior to any transaction processing. Specifically, active nodes in the network can establish a blockchain in a collaborative way and is independent of any centralized authority. Now, distributed entities can crash, perform malignantly, or even poor network communications between these entities can lead to a service interruption. Hence, in order to guarantee an non-interruptible service, network nodes need to provide a guarantee that they all concur and add valid entries to the blockchain. All miner nodes are forced to abide by the rules appointed in the distributed consensus protocol to append a new blockchain block, and the PoW algorithm is used to enable the Bitcoin system to realize a completely distributed consensus.



```
OP_DUP OP_HASH160 14836dbe7f38c5ac3d49e8d790af808a4ee9edcf OP_EQUALVERIFY OP_CHECKSIG
```

FIGURE 3. ***ScriptPubKey*** **generation from Bitcoin addresses to identify recipients.**

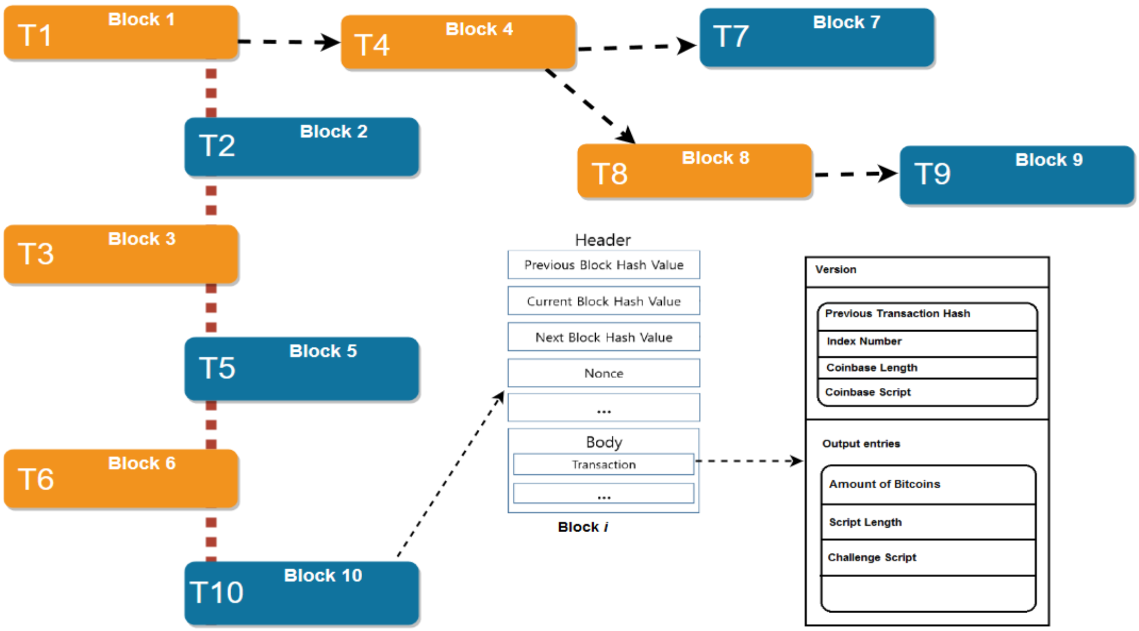


FIGURE 4. A depiction of consensus model of the blockchain with a block and transaction structures.

This PoW-based consensus protocol implies a few key rules: (1) transactions are enabled to spend only unspent valid outputs, (2) output and/or input entries are rational, (3) spent inputs are assumed to have correct signatures, and (4) the outputs of a *coinbase* that are a unique type of bitcoin transaction created by a miner node are not permitted to be spent inwards exactly 100 blocks of their creation.

Many existing Bitcoin studies focus on consensus algorithm as they open up various questions and research problems, including, but not limited to, (1) stability [13], (2) damage of computation resources [43], and (3) scalability [62]. On another hand, from the perspective of power and resource consumption, the PoW consensus protocol in blockchain is considered to be inefficient. Therefore, several research efforts have also been addressing this concern by presenting new consensus protocols such as Proof-of-Stake (PoS), practical byzantine fault tolerance (PBFT) [20], Proof of Storage [59], and so on. These newly-proposed protocols differ from the PoW with regard to the resources expensed, and are driven by the internal resources consumed rather than the external usage (as in PoW). Note that this infrastructural change establishes a completely different series of incentives and therefore mutates the security models in the Bitcoin system [23].

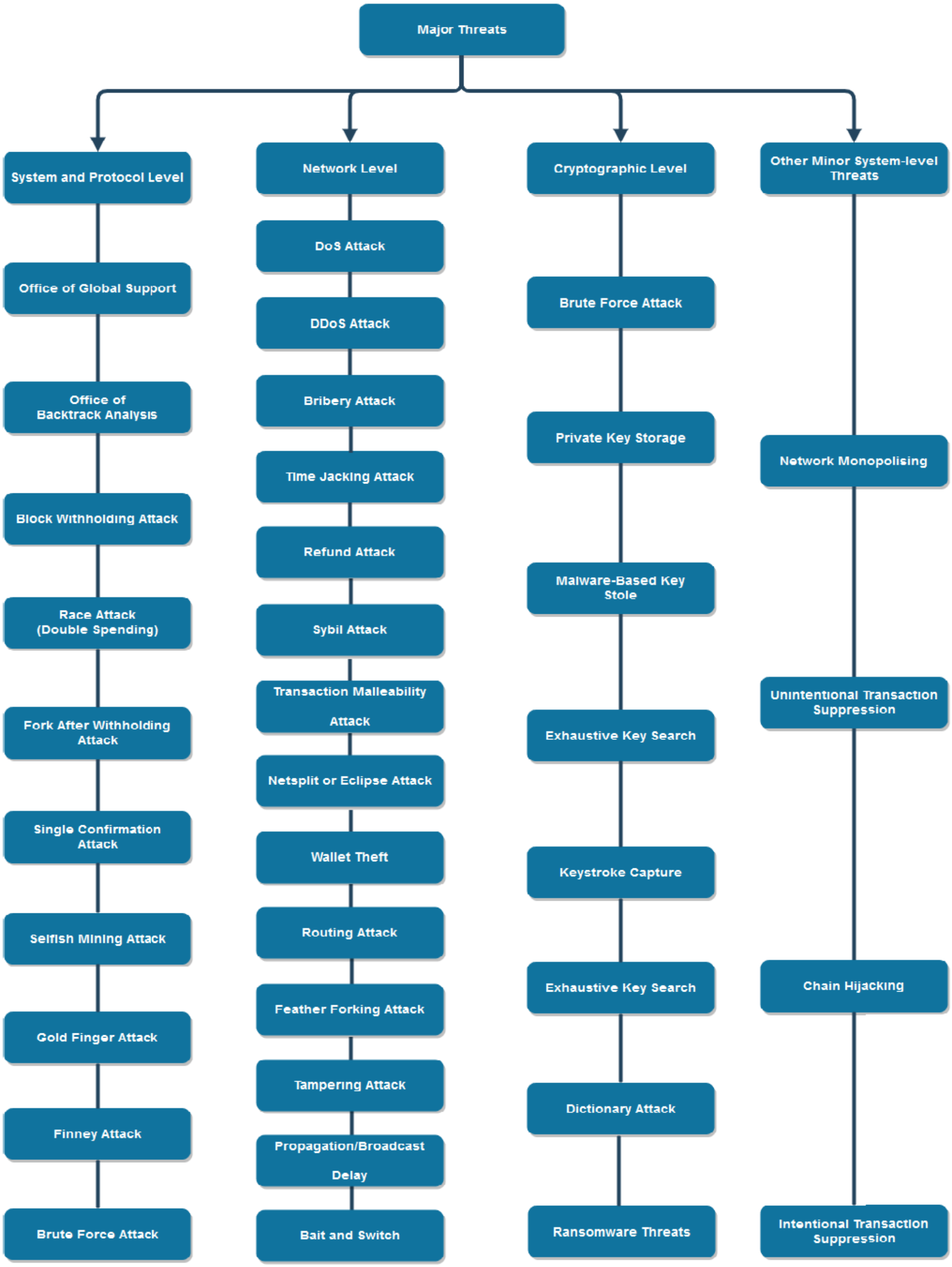


FIGURE 5. Classification of resiliency threats in the Bitcoin system.

4) PEER-TO-PEER (P2P) COMMUNICATION NETWORK

Bitcoin uses an unstructured P2P-based network along with a reliable TCP-based connection transport. An unstructured network suits Bitcoin due to the need for a rapid distribution of data to attain the blockchain consensus. However, various challenges are exposed here and they can be overcome through several means, e.g., using *mainnet*, a live network for Bitcoin system connectivity [23]. Here, all entities (i.e., nodes) must keep the IP address list of their prospect peers. This list will be bootstrapped through the DNS server where further IP addresses are enabled for exchange between other connected peers. All peers seek to keep at least 8 unencrypted TCP-based connections within the overlay network and utilize port 8333 for inbound connections listening. Once an incoming connection is detected, the peers need to run a layer handshake-based application, where the exchanged packets/ messages consist of a synchronization timestamp, the protocol version, and the IP address of the node. By default, each peering node choses its relative peers randomly and an updated list of peers will be elaborated after a constant time interval. This configurative process helps reduce the risk of a *netsplit* threat, i.e., where an adversary can establish a conflicting view of the blockchain through a compromised entity.

The mining nodes hearken to incoming announcements of new blocks that are advertised using Inventory Vector (INV) messages constituting the hash value of a newly-mined block. A *GETDATA* packet will be transmitted to a contiguous node in case the miner figures out that it does not have a recently advertised block and therefore the contiguous node will reply with the desired data through a *BLOCK* packet. The desired block is expected to arrive within 20 minutes at most. Otherwise, the miner node requesting the information will disconnect from the non-responding contiguous miner and demand the same information from another contiguous miner. The benefit of using an unstructured-based P2P network in Bitcoin is to allow for fast data distribution in the entire Bitcoin network.

From an infrastructural resiliency perspective, Bitcoin is very dependent upon the consistent and efficient state of PoW-based consensus and blockchain in general. A discordant state of blockchain is very likely to lead to a double spending vulnerability in case it is successfully exploited by users. For this reason, the P2P communication network must guarantee reliable scalability. Moreover, in Bitcoin, typical users use the Simplified Payment Verification (SPV) scheme to check whether a specific transaction is appended to a particular block, i.e., without downloading the complete block. However, this operation is very costly and it can lead to other security threats, in particular, Denial of Service (DoS) attacks [23], [29].

5) ETHEREUM: CONQUERING BITCOIN's INFRASTRUCTURE LIMITATIONS

Ethereum was first introduced by Buterin *et al.* [18] as a public and open source distributed operating system and computing platform in the form of a blockchain. This proposed system supports an adjusted *Nakamoto* protocol version and it provides efficient features and functionality of smart contracts. In addition, this blockchain-based platform resolves a few critical challenges and limitations in the blockchain structure and scripting language of Bitcoin, such as Turing-completeness. Thereafter, it upholds the state of the transaction for every possible kind of computations including, but not limited to loops [65]. Lastly, Ethereum offers an absolute layer to allow users to build and design their own transaction format, state transition functions, and ownership rules through the built-in Turning-Complete feature [18]. However, these privileges are given through the involvement of the smart contract concept in the Ethereum and cryptographic rules are only executed as long as specific instructions are fulfilled.

III. RESILIENCY CHALLENGES

A. BLOCKCHAIN CONSENSUS PROTOCOL AND MINING

Bitcoin users are not required to authenticate themselves before acceding to the network in the PoW consensus protocol. This process renders the consensus protocol in Bitcoin very scalable in supporting a massive amount of users. However, the PoW-based consensus protocol is prone to particular attacks, where an attacker is estimated to gain control over more than 51% of the mining power [23]. Under this threat scenario, attackers might create their own transaction block or even fork the local blockchain to converge with the primary blockchain at a later time. This security violation can apparently foster various attacks against Bitcoin, including, but not limited to, DoS and double spending [23]. Since Bitcoin is a completely decentralized cryptocurrency system uncontrolled by any party or authority, many adversaries can exploit such a system as simple means to commit fraud and hack transactions. Hence, in the following subsections, we present a taxonomic overview of existing security threats and their countermeasures in Bitcoin along with underlying technologies.

B. DOUBLE SPENDING

The Bitcoin validation process for each transaction block will eventually result in either acceptance or rejection of a questioned block. Namely, the acceptance will be signaled by prolonging a blockchain (i.e., a new block is added to the blockchain), whereas rejection will be signaled by discarding the transaction block and solely maintaining the recently-updated blockchain by other entities [58]. For instance, consider the following Bitcoin communication scenario where entity $E1$ can send a transaction block to entity $E2$, i.e., $E1 \rightarrow E2$, and then to entity $E3$, i.e., $E1 \rightarrow E3$, with the same bitcoin. In this transactional scenario, it is infeasible for other entities in the Bitcoin network to find out what entity is doing a double spending-based attack [34]. Now, according to Bitcoin system rules, all active entities are required to prolong the longest blockchain, and therefore under this double spending scenario, both entities $E2$ and $E3$ will possess a blockchain of exactly the same length. This will result in two typical scenarios:

- Entity $E2$ might await for a confirmation from a contiguous sincere entity to append the questioned transaction $E1 \rightarrow E2$ into the local blockchain. This operation will preclude any possibility of receiving a confirmation for another transaction such as the transaction $E1 \rightarrow E3$ (to be appended to the local blockchain in $E2$).
- Entity $E2$ might just orphan the previously-received transaction block $E1 \rightarrow E2$ immediately after it finds out about the transaction block $E1 \rightarrow E3$, i.e., since all transaction blocks in the Bitcoin network are advertised (broadcasted).

However, the chances of having double spending vulnerabilities drastically decrease as the number of confirmation messages increases (which is eventually the case as there are up to 6 confirmation messages in the current Bitcoin network).

It is expected that a confirmation message is also received by a Bitcoin entity within a fixed time frame before the entity makes a decision about a received blockchain transation.

C. MINING POOL VULNERABILITIES

Bitcoin mining pools are an aggregation of resources used by mining entities that share their own processing power in the P2P network. These pools help evenly divide the reward based on the work rate they contribute to the probability of discovering a transaction block. Now, the current Bitcoin infrastructure is composed of individual miners, open pools which enable any miner node to join the network, and a closed pool which demands a private/closed relationship to accede. These mining pools are designed to augment any computation-processing power that promptly impacts the probing time of a transaction block, thereby increasing the probabilities of winning a mining reward [23]. Generally, many mining pools have been created in the last several years and are being controlled by pool managers, who send unresolved units off to particular pool members/miner nodes. In turn, pool members create full proofs-of-work (FPoWs) and partial proofs-of-work (PPoWs), and then transmit them back to the pool manager in the form of shares. Once a pool member finds a new transaction block, it will transmit it to the pool manager accompanied by its full proofs-of-work. Next, the pool manager will advertise the explored transaction block in the network to acquire the mining reward. The acquired reward will then be split among the contributing miners according to their contributed portion of shares. Hence, the partial proofs-of-work is only used to determine the way rewards are distributed among clients.

1) POOL HOPPING ATTACK

Rosenfeld [57] introduced a pool hopping attack, in which an adversary exploits information about the number of transmitted shares in the mining pool to carry out selfish mining (i.e., the process of mining bitcoins where a set of miners collude to augment their revenue). Under this threat model, the attacker attempts to execute a continual and uninterrupted analysis over the amount of shares transmitted by contiguous miner nodes to their pool manager for the purpose of exploring a new transaction block. The promise here is that rewards are distributed according to the transmitted shares. Therefore, if a huge amount of shares is transmitted and does not have any new transaction block, the attacker will eventually get a relatively small portion of the reward. Hence, it could be more beneficial for an attacker to turn into another mining pool or even mine distinctly [23]. Conversely, a ''sponsored block withholding attack'' was also identified by Buterin *et al.* [16], in which a selfish miner could complicity incorporate with another pool to gain a reward from this connivance pool and attack another pool.

2) BRIBERY ATTACK

Bonneau [12] details an attack model called the bribery, in which an adversary can acquire most of the processing resources for a fixed period of time through bribery. In particular, J. Bonneau describes three methods to insert bribery into the Bitcoin network. (1) Negative-Fee Mining Pool where an adversary creates a pool through paying a higher return, (2) Out-of-Band Payment where an attacker pays the owner with processing resources and directly makes these tricked owners mine transactions' blocks appointed by the attacker, and (3) In-Band Payment through a forking operation where an adversary tries to bribe via Bitcoin itself (by originating a fork that has a bribe in the form of free money for any miner node taking over the fork, see [23]). If an adversary possesses most of the hashing power, they might establish various types of attacks, including, but not limited to, DDoS and double spending [27]. Furthermore, miner nodes accepting bribes will acquire only a short-lived profit, which could be undermined by the losses under the existence of Goldfinger and DDoS attacks, or even through an exchange rate-based crash [23].

D. VULNERABILITIES IN CRYPTOGRAPHIC APPLICATIONS

The management of private keys in Bitcoin and blockchain is still an unresolved problem [24], [72]. Current Bitcoin applications utilize the private key to validate the identities of users and accomplish a payment-based transaction. However, the trustworthiness of the private key only depends on the assumption that information cannot be tampered with. In Bitcoin, unlike classical public key cryptography, clients are held accountable for their private keys, and therefore each client is responsible for producing their own private keys and maintaining it (rather than a third-party). If the private keys are lost, the owning clients will not be able to access their own digital assets in a Bitcoin network. Moreover, cryptographic algorithm-based applications might present unbeknown backdoors and threats. Namely, since cryptographic algorithms such as, *Rivest-Shamir-Adleman* (RSA) and *Elliptic-Curve Cryptography* (ECC) are being widely deployed in blockchain, backdoors or threats can appear in such algorithms themselves or even during their implementation phase. Therefore, while a Bitcoin wallet guarantees the flexibility to manage and maintain a user's private key, saving all private keys on local networking-enabled entities could pose tremendous amount of security risks in case of theft [6].

E. GOLDFINGER ATTACKS

Kroll *et al.* [37] also present a novel attack model called the goldfinger attack, where the intention of the majority of mining entities is to explicitly break down Bitcoin network stability. For instance, such an attack scenario could occur when a connected entity in the network attempts to harm Bitcoin in order to avert a competition with its own currency. Additionally, a single user may try to invest in a competing currency. However, these typical vulnerabilities have been addressed through altcoin infanticide [13], where a deep-forking attack was contrasted to a competing currency of a very low mining power, which was effectively mounted using

Bitcoin mining nodes. As a result, this experiment proved the Goldfinger attack profitable.

F. FEATHER-FORKING THREAT

Miller [45] introduced an attack model called feather-forking, where miner nodes try to monitor a blacklist for transactions while publicly warranting if a particular transaction is appended into the blacklist in the blockchain. The adversary will take revenge by discarding the block that possess the targeted transaction and then try to fork the blockchain. The adversary's fork is expected to proceed until outperforming the master branch and earning, or just dropping behind by n number of blocks and thus giving up by advertising the targeted transaction block [13]. In general, adversaries with less than 50% of the mining power are expected to forfeit currencies; yet, they can only block a particular blacklisted transaction block with a certain level of probability. Furthermore, if adversaries have the capability to convincingly prove the seriousness about retaliatory forking, they might also be capable of imposing their designated blacklist without any eventual cost/price. That is, as long as the remaining miner nodes ratify that the adversary is intending to carry out a costly retaliation-based feather-forking attack.

G. NETWORK VULNERABILITIES

Presumptive attacks in the Bitcoin network are estimated to consume over 50% of the processing power held by a malignant entity. However, several offensive trials might be established [35] as follows.

- Stealing coins from particular addresses of other entities: It is unlikely to steal coins from other nodes as doing so will demand breaking down cryptographic algorithms. However, for coins stealing, a malignant entity needs to originate a transaction block through the use of the target entity's private key. Deriving a private key matching a public key will be cryptographically difficult with existing computation abilities [35].
- Restraining transaction blocks in the blockchain: Malignant entities may feasibly evade transactions that supply payments to a particular miner (i.e., address). However, while a P2P network is adopted in Bitcoin, this type of attack will eventually be detected as all blockchain transactions are advertised to each miner and such a detection is achieved due to the existence of sincere miners in the Bitcoin network who will properly append this transaction when creating a new block [35].
- Tempering with a block reward: This type of attacks is also infeasible because a malignant miner node cannot control the copy of software distributed within the entire Bitcoin network. When developers update the software copy, the change will be visible to all clients in the global network. However, such an attack might lead to clients loosing their trust in the system and the bitcoin price will definitely be impacted without any attack establishment by malignant entities. Hence, this attack is practically feasible and the key challenge is that such an attack demands significant investment to outnumber hash power, but it is difficult to accomplish in practice [35]. While numerous threat models have also been identified for Bitcoin networking infrastructures for the past several years, many significant attack models have been ignored. For example, Apostolaki *et al.* [7] investigated currency threats at the level of Internet routing infrastructures, where it was induced that portions of Bitcoin flows could be feasibly manipulated by intercepting network flows or manipulating routing advertisements, e.g., Border Gateway Protocol (BGP).

IV. MACHINE LEARNING-BASED EFFORTS AND COUNTERMEASURES AGAINST THREATS

Network infrastructures have existed for many decades along with users who behave maliciously within these network systems. These particular users are usually referred to as malignants [52]. Now with regard to networks carrying financial transactions, malignants contain users who carry out deceitful transactions. Under the scope of these financial and transactional networks, the intended and perceived objective is to stop these malignants from carrying out illegitimate activities [52]. Hence, in Bitcoin networks, it is vital to reveal suspicious behaviors concerning all users due to the drastically growing nature of thefts.

Blockchain technology hinders two particular types of malicious activities conducted on-line; record hacking and double-spending [67]. As previously discussed, due to network delay and/or propagation delay in the Bitcoin P2P network, the *double-spending* problem might probably occur as a Bitcoin client attempts to participate in more than one transaction with the same bitcoin (i.e., or same quantity of bitcoins). This is realistic because of the delay in broadcasting pending payments through the network, which in turn results in particular nodes being given unvalidated transactions at different times [67].

The anomaly detection problem can also be generalized or applied to other networks which do not necessarily encompass financial and monetary transactions. Indeed, there are many studies and research solutions proposed in the past to deal with anomaly detection [51], [52]. These efforts deploy a broad range of techniques including network analysis and ML methods. However, in this survey paper, we mostly concentrate on recent ML solutions and proposals to address the problem of revealing malignants and suspicious activities and actions in Bitcoin and blockchain, accordingly. Tables 1 and 2 depict a taxonomic presentation and classification of security solutions based on various ML techniques. For example, Smith *et al.* [61] used particular clustering methods to capture malignant activities in a network and were able to classify legitimate system users separately from malicious users, i.e., via *k-means*-based clustering along with self-organizing maps to design a detection solution.

ML techniques have also been used in several studies to address the aforementioned security threats such as [51] and [52]. Namely, using the *k-means* clustering metric

TABLE 1. Taxonomic classification of proposed security solutions using machine learning.

Reference	Year	Method	Contribution
Pham and Lee [51]	2016	Unsupervised ML techniques	Anomaly detection in the Bitcoin network where clients and transactions are considered suspicious
Yin and Vatrapu [69]	2017	Supervised ML technique	Demonstrate how large is the share of cybercrime-related nodes in the Bitcoin network and nodes/addresses relevant to malicious activities
Monamo, et al. [48]	2016	ML-based multifaceted approach	Bitcoin fraud detection where the fraud is studied from both global and local perspectives using trimmed *k-means* and *kd-trees*
Zambre and Shah [70]	2013	ML-based clustering and classification	Identifying peculiar properties of clients performing anomalous behavior through clustering clients who exhibit suspicious activities
Portnoff, et al. [53]	2017	ML classifiers	Designing a ML-based classier to differentiate between ads posted by the same author vs. different authors and also a linking technique, which uses leakages from the Bitcoin network and sex ad site to link a set of sex ads to Bitcoin transactions and public wallets
Harlev, et al. [31]	2018	Supervised ML technique	Minimizing anonymity in the Bitcoin network through a Supervised ML technique deployment to predict the type of unidentified nodes/entities
Bartoletti, et al. [10]	2018	Data mining technique	Detecting Bitcoin addresses in the network that are related to a Ponzi scheme
Zhdanova, et al. [71]	2014	ML-based micro-structuring technique	Revealing fraud chains through developing a new technique for detection of fraud chains in Mobile Money Transfers (MMT) systems
A. Bogner [11]	2017	ML adoption for graphical threat detection	Providing human operators with an intuitive way to derive insights about the blockcahin system through in-gathering of the system's features into a group of characteristics which are graphically rendered
Remy, et al. [56]	2017	ML-based network analysis techniques	Tracking the clients' activities in the Bitcoin system using community detection on a network of weak signals
Hirshman, et al. [32]	2017	Unsupervised ML technique	Applying ML methods on a Bitcoin transactions dataset to explore anonymity guarantees in the network by clustering the dataset
Monamo, et al. [47]	2016	Unsupervised ML technique	Exploring the use of trimmed *k-means* for simultaneous objects clustering and fraud detection in a multivariate configuration in order to detect fraudulent behaviors in blockchain blocks
Pham and Lee [52]	2016	ML-based superior method	Detecting anomalies in Bitcoin networks through exploring clients where transactions seem to be most dubious where a malignant behavior is regarded as a proxy for dubious activity

in [66], Pham and Lee [51] tried to detect abnormal behaviors in Bitcoin transaction networks by using multiple unsupervised ML techniques, such as *k-means* clustering and Unsupervised Vector Machine (SVM) on two Bitcoin transactions' graphs. Specifically, one graph presents client transactions as entities and the other clients as entities. Meanwhile, Pham and Lee [52] utilize power degree laws along with Local Outliers Factor (LOF) and densification techniques. Similar to [51], Pham and Lee [52] also applied two transactions' graphs to the Bitcoin network; one presents clients transactions as entities and the other clients as entities.

Additionally, Harlev *et al.* [31] introduced a novel technique to decrease anonymity in Bitcoin networks using a ML-based supervised approach [19] to predict unidentified network nodes. Caruana and Niculescu-Mizil [19] used a sample of 434 nodes with about 200 million transaction blocks whose identities were exposed. The experimental results demonstrated the efficiency of the proposed ML

TABLE 2. Taxonomic classification of proposed security solutions using machine learning.

Reference	Year	Method	Contribution
Kurtulmus and Daniel [38]	2018	Intelligent problem solving based on ML aspect	Propose DanKu, a byproduct protocol utilizing the distributed nature of smart contracts along with a ML-based intelligent problem solving to solve crowd-sourcing funds for computational research and to efficiently provide a new marketplace without a need for a middlemen
S. Dey [25]	2018	Supervised ML algorithm and algorithmic game theory	Presenting a methodology based on intelligent software agents that supervises the stakeholders' activities in the Bitcoin system in order to detect abnormal behaviors using a supervised ML algorithm along with algorithmic game theory
Liu, et al. [42]	2017	Immune ML-based model	Proposing a ML-based solution to capture double-spending activities in fast Bitcoin payments, where the solution consists of several immune-based blockchain nodes that embrace a detection component
Shaukat and Ribeiro [60]	2018	ML-based Ransomware detection	Proposing a ML solution based on an extensive analysis of Ransomware dataset families in order to provide a layered defense system against cryptographic ransomware in a cryptocurrency system
K. Baqer [8]	2016	Clustering-based method/stress test	Introducing an empirical analysis of a spam campaign/stress test that caused DoS attacks. Namely, a clustering based technique is deployed to detect spam transactions in the Bitcoin network
COINHOARDER [33]	2018	NLP and ML-based phishing ring DNS style detection	Introducing a detection scheme for phishing ring DNS style through ML and NLP techniques where the detection mechanism relies on the observation of the newly registered and/or launched domains
Ermilov, et al. [44]	2017	Address-based clustering methods	Introducing an off-chain information solution along with blockchain information for Bitcoin address separation and classification in order to detect and filtrate errors in users' inputed data and therefore evade insecure Bitcoin usage patterns

scheme in predicting the entries of a yet-unidentified nature. Hirshman *et al.* [32] also proposed an unsupervised ML technique to detect abnormal conducts in the Bitcoin transaction network. Further, their ML approach was applied on a dataset of Bitcoin network transactions to discover anonymity guarantees in the Bitcoin system through dataset clustering.

Meanwhile, Monamo *et al.* [47] investigated the use of the trimmed *k-means* method to capture deceitful behaviors in the Bitcoin network. Namely, trimmed *k-means* was used for simultaneous clustering of entities and fraud capture in a multivariate configuration. Also, Monamo *et al.* [48] described various fraud activities in the Bitcoin network from both local and global perspectives by mean of trimmed *k-means* and *kd-trees*. Inspired by the paradigm of anomaly detection joint with Numenta Anomaly Benchmark (NAB) [40], the spheres in [48] were examined using random forests, binary regression, and maximum likelihood methods. However, based upon the experimental evaluations, the global outliers perspective seems to provide better accuracy than the local perspective.

Bartoletti *et al.* [10] utilized data mining and ML-based methods to investigate and capture Bitcoin addresses relevant to Ponzi schemes. In such particular attacks, an adversary shares a falsified transaction block which can threaten investment in Bitcoin. Bartoletti *et al.* [10] used a dedicated dataset possessing real-world features for their Ponzi schemes that was built by analyzing transactions carrying out scams. Zhdanova *et al.* [71] also used micro-structuring, an often practiced ML approach, to develop a novel technique for fraud chain detection in Mobile Money Transfer (MMT) systems. While traditional detection techniques are built using data mining and ML, Zhdanova *et al.* [71] designed their solution using Predictive Security Analysis at Runtime (PSA@R), which uses an event driven process analysis approach.

Moreover, Yin and Vatrapu [69] provided an accurate estimation of the portion of cybercriminal nodes in the Bitcoin network. Namely, they utilized a large Bitcoin dataset composed of more than 800 observations classified into about 12 categories including observations relevant to cybercrime and uncategorized ones. The utilized dataset was acquired from a third party (provider) who priorly designated three classes for clustering of Bitcoin transactions, i.e., co-spend, intelligence, and behavior-based clustering.

Yin and Vatrapu [69] re-classified the observations via a supervised ML technique which outputted four prevailed classifiers with very high cross-validation accuracy. Finally, with regard to the weighted averages and per class precisions in cybercrime-related categories, Yin and Vatrapu [69] only used Bagging and Gradient Boosting classifiers from the top four classifiers. Experimental evaluation in [69] showed that the share of cybercrime-related activities was about 29.81% and 10.95%, respectively, according to the Bagging and Gradient Boosting techniques.

As Bitcoin is an open-source cryptocurrency system, its design is public, ensuring that no party or authority controls it and all users can participate [1]. However, security violations can occur very frequently. For example, [2] details reports from Bitcoin clients regarding their compromised wallets. Also, Zambre and Shah [70] took advantage of the readily-available information about all Bitcoin transactions to trace back the flow of money within compromised client accounts (once these clients reported robbery or heist). Zambre and Shah also tried to uncover bizarre client behaviors performing such robberies which lead them to protrude. Namely, Zambre and Shah [70] investigated three different types of heists and robberies, Stone Man Loss (SML), All in Vain (AIV), and Mass MyBitcoin Theft (MMT). In SML, Bitcoins are stolen from the original key whereas, in AIV, a trickster Bitcoin client fabricates a massive amount of transactions right after the heist in an attempt to taint the currency [55]. Finally, in MMT, the heist takes place when clients use the same passwords for MyBitcoin and Mt.Gox [4], i.e., due to the password leakage problem in Mt.Gox. In the case of MMT, the thief steals bitcoins from clients with compromised keys to their own key. The authors extracted the classifying features for each Bitcoin client in the dataset and applied *k-Means* clustering to classify them, i.e., defining K centroids and matching each one with observations from the dataset in order to define the objective function.

It should be noted that various illicit activities can also be carried out between Bitcoin clients, e.g., human trafficking, drug sales, etc. Hence, distinguishing such advertisements is a very challenging task. To address this challenge, Portnoff *et al.* [53] presented some ML-based mechanisms and tools which could be used either in an independent or conjunctive way to classify and categorize sex advertisements based upon the genuine owner (not the alleged owner) in the advertisement. Namely, the proposed ML-based classifiers utilize *stylometry* to differentiate advertisements published by the same client from the ones posted by different owners with 90% true positive rate (TPR) and 1% false positive rate (FPR). Furthermore, Portnoff *et al.* [53] explored the Bitcoin mempool leakage to develop a mechanism that matches groups of sex advertisements with Bitcoin wallets and relative transactions. The mechanism's performance was evaluated through a four-week proof of concept trait over Backpage [5] as the site hosts sex advertisements.

Over and above, online machine learning-based security platforms that aim at detecting abnormal and violative client behaviors in open-source systems, including cryptocurreny systems, are very limited. For example, Bonger [11] introduced an online optimized interpretability using an unsupervised ML mechanism for detecting anomalous client activities. This work combined characteristics of an open-source system with sets of graphical presentations and features. Moreover, this proposed mechanism can be flexibly applied to any time series of numerical data. Bonger [11] evaluated the performance of the proposed solution using the public Ethereum blockchain.

Overall, blockchain technology allows for the efficient creation of contracts guaranteeing remuneration in interchange for a well-trained ML model for a specific data set. Hence, this will warrant Bitcoin clients to train ML models for remuneration in an efficient trust-less way such that smart contracts will utilize the blockchain to autonomously verify and legitimize the trained model. Therefore, assumptions about the correctness of the solution are not needed as these solutions submitted by clients do not possess any counterparty threat or the risk of not receiving a payment for provided work.

Buterin *et al.* [18] also discussed the idea of on-chain decentralized marketplaces through the adoption of a reputation and identity system. However, they did not emphasize and clarify their proposal from the implementation perspective. Building on top of the findings in [18], Kurtulmus and Daniel [38] presented a novel protocol on top of the Ethereum blockchain such that reputation and an identity system are not imposed in order to elaborate transactions of the marketplace. Specifically, the proposed protocol in [38] aims at creating a marketplace for exchanging (i.e., providing) ML models with entrants in a secure and automated way. However, the proposed ML models are only evaluated on the Ethereum Virtual Machine in a forwarding pass mode (where verification and training stages are carried out in an independent way in order to restrain over-fitting challenges).

In cryptocurrency systems such as Bitcoin and Litecoin [3], majority-based attacks [26] may not form a grand security threat. However, when it comes to consortium (i.e., collaboration among multiple public and private institutions) blockchain-based networks, majority-based attacks could be an epidemic impendence once a collusion between two or more institutions occurs. To address such a challenge, Dey [25] introduced a novel mechanism that deploys intelligent software-based agents in order to supervise and regulate the activities of clients in the Bitcoin network. The proposed solution utilizes a supervised ML approach along with algorithmic game theory to detect abnormal activities and prevent them from re-occurrence (specifically, collusions and majority based-attacks).

As discussed in Section III, the detection of double-spending violations is very challenging, i.e., since the fast payment strategy adopted by the Bitcoin system is based on the fact that the service is not guaranteed until the transaction of payment is appended to the vendor's wallet. Thus, it becomes worthless to capture double-spending attacks. In order to address this security challenge, Liu *et al.* [42] presented an immune-based mechanism [30] to capture double-spending attacks in the Bitcoin network. The proposed solution consists of several Bitcoin entities that have a detection module implemented. This solution uses a capturing module to remove an antigen character from the transaction prior to generating the initial detectors. Next, the initial detectors detect double-spending attacks in collaboration with the memory detectors. Once a double-spending attack is matched by an initial detector, it is promoted to a memory detector and thus delivered to other entities in the network.

As also previously discussed, cryptographic ransomware is a form of malicious software that can infect Bitcoin networks. Namely, it can hold user files as hostage by encrypting them and then requesting some ransom payment prior to providing the users' decryption key. Shaukat and Ribeiro [60] considered a comprehensive analysis of a large ransomware dataset of various families. Next, they presented Ransomwall, a layered-based detection and mitigation system against cryptographic ransomwares. The proposed solution deploys a hybrid mechanism of dynamic and static analysis combined together in order to identify and characterize different behaviors of these cryptographic ransomwares. Ransomewall [60] utilizes ML to unearth 0-day intrusions. First, the tagging layer tags a process that correlates with a malicious behavior. Next, all user files adjusted by this process are backed up for the purpose of maintaining users' data in order to cluster and classify it as either benign or ransomware.

V. FUTURE RESEARCH DIRECTIONS

While the previous sections have presented an intensive study on the resiliency aspects of the Bitcoin system and an overview of recent research solutions, this section provides a summary of our take-aways. However, prior to discussing future research challenges and trends, we briefly recall that the employment of the PoW-based consensus protocol guarantees an efficient settlement to the Byzantine generals problem in the Bitcoin network [23]. Nevertheless, in order to build a completely distributed consensus protocol, Bitcoin renders its network and system vulnerable to various resiliency vulnerabilities, e.g., double spending, which are very feasible and attainable in the Bitcoin network.

In order to enhance resiliency for the Bitcoin network and blockchain technologies and protocols, researchers considered the altcoins concept in a large testing environment [23]. Efficient techniques such as machine learning [14] can also be deployed to enforce and enhance security aspects in such a cryptocurrency system. Since the Bitcoin network will continue to evolve in the near future, we therefore present some potential open research problems and future research directions here.

- Scalability and blockchain protocol: Miner entities can act in a selfish manner by continuing to carry particular blocks of transactions and unleashing them whenever they wish in order to increase their revenue. Such selfish activities will likely create a game theoretic challenge among egocentric miner nodes and the network [36]. As a result, several suggestions have been presented, such as [28] and [41], that prove the usefulness of game-theoretic techniques in terms of information about the impacts of egocentric mining and holding of blocks of transactions. Overall, these typical techniques are quite accurate with regards to modeling the various issues and delivering efficient solutions for challenges related to mining pools.
- Cryptography techniques: The deployment of clustering techniques based on specific thresholds are designed to address a broad range of threats (e.g., specifying a cluster head and obtaining an extra signature on each single transaction, or utilizing trusted paths based upon particular machinery resources to enable clients to read and write some cryptographic data [9]). However, there are only a few number of methods that use string searching filters for protecting wallets (e.g., AhoâĂŞCorasick and Bloom filter).
- Incentives for miners: The Bitcoin incentive is either constant or inconstant according to the complexity of the miner nodes resolving the puzzle. Namely, an inconstant incentive could eventually augment the competition among miner nodes and assist with solving very challenging puzzles. Hence, malignant miners could conduct an illegitimate activity through the Bitcoin network to acquire extra awarding coins, which will augment the amount of sincere entities in the network. Hence, it is very important to address this challenge by making miner nodes settle to a currency in this cryptocurrency network.
- Preventing backtracks: Smart contracts are of a particular interest to financial applications, which incarnate self-enforcing-based contracts entities in financial networks such as Bitcoin. Therefore, this concept could be adopted in Bitcoin infrastructures as the blockchain system drives out the need to depend on authenticated third parties to handle contracts. However, Bitcoin support for such contracts is still very restricted.

VI. CONCLUSIONS AND FUTURE WORK

Blockchain has demonstrated its potential to transform and mutate classical financial and transactional market models with its key distinctive features, including decentralization, anonymity, and auditability. Hence, in this survey paper, we presented an intensive and comprehensive discussion overviewing Bitcoin and blockchain infrastructures along with relevant key components.

The increasing popularity and important capital in the financial market render Bitcoin system attractive to attackers to establish and elaborate a variety of security attacks. Although the Bitcoin infrastructure is established using the PoW and consensus protocols to protect client transactions and activities, these protocols themselves remain a point of vulnerability and exploitation for cyber threats, starting from the sniffing of network packets to the double spending activities.

In this work, we first presented an overview of the Bitcoin network and related blockchain technologies and protocols. We then analyzed the common blockchain consensus protocol followed by a discussion of its characteristics, including advantages and limitations. Next, we presented a taxonomic classification and precise discussion of existing solutions and proposals that use machine learning (ML) techniques to solve common security threats and anomalous behaviors in Bitcoin networks and blockchain. Finally, we detailed some open research questions and future research directions followed by concluding remarks.

APPENDIX

Table 3 gives a list of acronyms used in this paper.

TABLE 3. A summary of acronyms used.

Acronym	Description
AIV	All in Vain
DDoS	Distributed Denial of Service
DoS	Denial of Service
ECC	Elliptic-Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
FPoWs	Full Proofs-of-Work
IDS	Intrusion Detection System
LOF	Local Outliers Factor
ML	Machine Learning
MMT	Mobile Money Transfer
MMT	Mass MyBitcoin Theft
P2P	Peer-to-Peer Network
PBFT	Practical Byzantine Fault Tolerance
PoS	Proof-of-Stake
PoW	Proof-of-Work
RSA	Rivest-Shamir-Adleman
SML	Stone Man Loss
TPR	True Positive Rate
TTP	Trusted Third Party

ACKNOWLEDGMENT

The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied of NSF.

REFERENCES

[1] *Bitcoin*. Accessed: Jul. 2018. [Online]. Available: https://bitcoin.org/en/

[2] *List of Bitcoin Heists*. Accessed: Jul. 2018. [Online]. Available: https://bitcointalk.org/index.php?topic=83794.0#post_ubitex_scam

[3] *Litecoin*. Accessed: Jul. 2018. [Online]. Available: https://litecoin.org/

[4] *Coindesk*. Accessed: Jul. 2018. [Online]. Available: https://www.coindesk.com/

[5] *Ebackpage*. Accessed: Jul. 2018. [Online]. Available: https://www.ebackpage.com/

[6] Y. Liu *et al.*, "An efficient method to enhance Bitcoin wallet security," in *Proc. Int. Conf. Anti-Counterfeiting, Secur., Identificat. (ASID)*, Oct. 2017, pp. 26–29.

[7] M. Apostolaki, A. Zohar, and L. Vanbever, "Hijacking Bitcoin: Routing attacks on cryptocurrencies," in *Proc. Symp. Secur. Privacy (SP)*, May 2017, pp. 375–392.

[8] K. Baqer, D. Y. Huang, D. McCoy, and N. Weaver, "Stressing Out: Bitcoin 'stress testing,'" in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* Christ Church, Barbados: Springer, 2016, pp. 3–18.

[9] S. Barber, X. Boyen, E. Shi, and E. Uzun, "Bitter to better—How to make Bitcoin a better currency," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* Bonaire, The Netherlands: Springer, 2012, pp. 399–414.

[10] M. Bartoletti, B. Pes, and S. Serusi. (2018). "Data mining for detecting Bitcoin Ponzi schemes." [Online]. Available: https://arxiv.org/abs/1803.00646

[11] A. Bogner, "Seeing is understanding: Anomaly detection in blockchains with visualized features," in *Proc. Int. Joint Conf. Pervasive Ubiquitous Comput., Int. Symp. Wearable Comput.*, 2017, pp. 5–8.

[12] J. Bonneau, "Why buy when you can rent?" in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* Christ Church, Barbados: Springer, 2016, pp. 19–26.

[13] J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. A. Kroll, and E. W. Felten, "SoK: Research perspectives and challenges for Bitcoin and cryptocurrencies," in *Proc. Symp. Secur. Privacy (SP)*, May 2015, pp. 104–121.

[14] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[15] F. Tschorsch and B. Scheuermann, "Bitcoin and beyond: A technical survey on decentralized digital currencies," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2084–2123, 3rd Quart., 2016.

[16] S. Bag, S. Ruj, and K. Sakurai, "Bitcoin block withholding attack: Analysis and mitigation," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 8, pp. 1967–1978, Aug. 2017.

[17] Y. Liu, X. Liu, C. Tang, J. Wang, and L. Zhang, "Unlinkable coin mixing scheme for transaction privacy enhancement of Bitcoin," *IEEE Access*, vol. 6, pp. 23261–23270, 2018.

[18] V. Buterin *et al.*, "A next-generation smart contract and decentralized application platform," White Paper, 2014.

[19] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2006, pp. 161–168.

[20] M. Castro and B. Liskov, "Practical byzantine fault tolerance and proactive recovery," *ACM Trans. Comput. Syst.*, vol. 20, no. 4, pp. 398–461, 2002.

[21] D. Chaum, "Blind signatures for untraceable payments," in *Advances in Cryptology*. Springer, 1983, pp. 199–203.

[22] D. Chaum, A. Fiat, and M. Naor, "Untraceable electronic cash," in *Proc. Conf. Theory Appl. Cryptogr.* Davos, Switzerland: Springer, 1988, pp. 319–327.

[23] R. Beyah, B. Chang, Y. Li, and S. Zhu, Eds., *Secure and Privacy in Communication Networks*, vol. 255. Springer, 2018.

[24] F. Dai, Y. Shi, N. Meng, L. Wei, and Z. Ye, "From Bitcoin to cybersecurity: A comparative study of blockchain application and security issues," in *Proc. Int. Conf. Syst. Inform. (ICSAI)*, Nov. 2017, pp. 975–979.

[25] S. Dey. (2018). "Securing majority-attack in blockchain using machine learning and algorithmic game theory: A proof of work." [Online]. Available: https://arxiv.org/abs/1806.05477

[26] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable," *Commun. ACM*, vol. 61, no. 7, pp. 95–102, 2018.

[27] A. Feder, N. Gandal, J. Hamrick, and T. Moore, "The impact of DDoS and other security shocks on Bitcoin currency exchanges: Evidence from Mt. Gox," *J. Cybersecur.*, vol. 3, no. 2, pp. 137–144, 2018.

[28] B. Fisch, R. Pass, and A. Shelat, "Socially optimal mining pools," in *Proc. Int. Conf. Web Internet Econ. (WINE)*. Bangalore, India: Springer, 2017, pp. 205–218.

[29] A. Gervais, G. O. Karame, S. Capkun, and V. Capkun, "Is Bitcoin a decentralized currency?" *IEEE Secur. Privacy*, vol. 12, no. 3, pp. 54–60, May/Jun. 2014.

[30] M. Glickman, J. Balthrop, and S. Forrest, "A machine learning evaluation of an artificial immune system," *Evol. Comput.*, vol. 13, no. 2, pp. 179–212, Jun. 2005.

[31] M. A. Harlev, H. S. Yin, K. C. Langenheldt, R. Mukkamala, and R. Vatrapu, "Breaking bad: De-anonymising entity types on the Bitcoin blockchain using supervised machine learning," in *Proc. 51st Hawaii Int. Conf. Syst. Sci.*, 2018, pp. 1–10.

[32] J. Hirshman, Y. Huang, and S. Macke, "Unsupervised approaches to detecting anomalous behavior in the Bitcoin transaction network," Stanford Univ., Stanford, CA, USA, Tech. Rep., 2013.

[33] A. Holub and J. O'Connor, "COINHOARDER: Tracking a ukrainian Bitcoin phishing ring DNS style," in *Proc. APWG Symp. Electron. Crime Res. (eCrime)*, May 2018, pp. 1–5.

[34] G. O. Karame, E. Androulaki, M. Roeschlin, A. Gervais, and S. Čapkun, "Misbehavior in Bitcoin: A study of double-spending and accountability," *ACM Trans. Inf. Syst. Secur.*, vol. 18, no. 1, 2015, Art. no. 2.

[35] P. K. Kaushal, A. Bagga, and R. Sobti, "Evolution of Bitcoin and security risk in Bitcoin wallets," in *Proc. Int. Conf. Comput., Commun. Electron. (Comptelix)*, Jul. 2017, pp. 172–177.

[36] A. Kiayias, E. Koutsoupias, M. Kyropoulou, and Y. Tselekounis, "Blockchain mining games," in *Proc. Conf. Econ. Comput. (EC)*, 2016, pp. 365–382.

[37] J. A. Kroll, I. C. Davey, and E. W. Felten, "The economics of Bitcoin mining, or Bitcoin in the presence of adversaries," in *Proc. Workshop Econ. Inf. Secur. (WEIS)*, 2013, pp. 1–21.

[38] A. B. Kurtulmus and K. Daniel. (2018). "Trustless machine learning contracts; Evaluating and exchanging machine learning models on the ethereum blockchain." [Online]. Available: https://arxiv.org/abs/1802.10185

[39] A. Laszka, B. Johnson, and J. Grossklags, "When Bitcoin mining pools run dry," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* San Juan, PR, USA: Springer, 2015, pp. 63–77.

[40] A. Lavin and S. Ahmad, "Evaluating real-time anomaly detection algorithms—The numenta anomaly benchmark," in *Proc. Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2015, pp. 38–44.

[41] Y. Lewenberg, Y. Bachrach, Y. Sompolinsky, A. Zohar, and J. S. Rosenschein, "Bitcoin mining pools: A cooperative game theoretic analysis," in *Proc. Int. Conf. Auto. Agents Multiagent Syst. (AAMAS)*, 2015, pp. 919–927.

[42] Z. Liu *et al.*, "Double-spending detection for fast Bitcoin payment based on artificial immune," in *Proc. Nat. Conf. Theor. Comput. Sci. (NCTCS)*. Wuhan, China: Springer, 2017, pp. 133–143.

[43] L. Luu, R. Saha, I. Parameshwaran, P. Saxena, and A. Hobor, "On power splitting games in distributed computation: The case of Bitcoin pooled mining," in *Proc. Comput. Secur. Found. Symp. (CSF)*, Jul. 2015, pp. 397–411.

[44] D. Ermilov, M. Panov, and Y. Yanovich, "Automatic Bitcoin address clustering," in *Proc. Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2017, pp. 461–466.

[45] A. Miller, "Feather-forks: Enforcing a blacklist with sub-50% hash power," Bitcointalk, Tech. Rep., 2013.

[46] D. Mingxiao, M. Xiaofeng, Z. Zhe, W. Xiangwei, and C. Qijun, "A review on consensus algorithm of blockchain," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2017, pp. 2567–2572.

[47] P. Monamo, V. Marivate, and B. Twala, "Unsupervised learning for robust Bitcoin fraud detection," in *Proc. Inf. Secur. South Africa (ISSA)*, Aug. 2016, pp. 129–134.

[48] P. M. Monamo, V. Marivate, and B. Twala, "A multifaceted approach to bitcoin fraud detection: Global and local outliers," in *Proc. Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2016, pp. 188–194.

[49] N. Satoshi, "Bitcoin: A peer-to-peer electronic cash system," Tech. Rep., 2008.

[50] T. Okamoto and K. Ohta, "Universal electronic cash," in *Proc. Annu. Int. Cryptol. Conf.* Santa Barbara, CA, USA: Springer, 1991, pp. 324–337.

[51] T. Pham and S. Lee. (2016). "Anomaly detection in Bitcoin network using unsupervised learning methods." [Online]. Available: https://arxiv.org/abs/1611.03941

[52] T. Pham and S. Lee. (2016). "Anomaly detection in the Bitcoin system—A network perspective." [Online]. Available: https://arxiv.org/abs/1611.03942

[53] R. S. Portnoff, D. Y. Huang, P. Doerfler, S. Afroz, and D. McCoy, "Backpage and Bitcoin: Uncovering human traffickers," in *Proc. Int. Conf. Knowl. Discovery Data Mining (SIGKDD)*, 2017, pp. 1595–1604.

[54] D. Puthal, N. Malik, S. P. Mohanty, E. Kougianos, and G. Das, "Everything you wanted to know about the blockchain: Its promise, components, processes, and problems," *IEEE Consum. Electron. Mag.*, vol. 7, no. 4, pp. 6–14, Jul. 2018.

[55] F. Reid and M. Harrigan, "An analysis of anonymity in the Bitcoin system," in *Proc. Int. Conf. Privacy, Security, Risk Trust (PASSAT), Int. Conf. Social Comput. (SocialCom)*, 2011, pp. 1318–1326.

[56] C. Remy, B. Rym, and L. Matthieu, "Tracking Bitcoin users activity using community detection on a network of weak signals," in *Proc. Int. Workshop Complex Netw. Appl. (COMPLEX NETWORKS)*. Lyon, France: Springer, 2017, pp. 166–177.

[57] M. Rosenfeld, "Mining pools reward methods, presentation at Bitcoin 2013 conference," in *Proc. Bitcoin Conf.*, San Jose, CA, USA, May 2013.

[58] M. Rosenfeld. (2014). "Analysis of hashrate-based double spending." [Online]. Available: https://arxiv.org/abs/1402.2009

[59] B. Sengupta, S. Bag, S. Ruj, and K. Sakurai, "Retricoin: Bitcoin based on compact proofs of retrievability," in *Proc. Int. Conf. Distrib. Comput. Netw. (ICDCN)*, 2016, Art. no. 14.

[60] S. K. Shaukat and V. J. Ribeiro, "RansomWall: A layered defense system against cryptographic ransomware attacks using machine learning," in *Proc. Int. Conf. Commun. Syst. Netw. (COMSNETS)*, Jan. 2018, pp. 356–363.

[61] R. Smith, A. Bivens, M. Embrechts, C. Palagiri, and B. Szymanski, "Clustering approaches for anomaly based intrusion detection," in *Proc. Intell. Eng. Syst. Through Artif. Neural Netw.*, 2002, pp. 579–584.

[62] Y. Sompolinsky and A. Zohar, "Secure high-rate transaction processing in Bitcoin," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* San Juan, PR, USA: Springer, 2015, pp. 507–527.

[63] N. Szabo, "Formalizing and securing relationships on public networks," *First Monday*, vol. 2, no. 9, 1997.

[64] V. Vallois and F. A. Guenane, "Bitcoin transaction: From the creation to validation, a protocol overview," in *Proc. Cyber Secur. Netw. Conf. (CSNet)*, 2017, pp. 1–7.

[65] D. Vujičić, D. Jagodić, and S. Randic, "Blockchain technology, Bitcoin, and ethereum: A brief overview," in *Proc. Int. Symp. INFOTEH-JAHORINA (INFOTEH)*, Mar. 2018, pp. 1–6.

[66] H. Xiong, J. Wu, and J. Chen, "K-means clustering versus validation measures: A data-distribution perspective," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 2, pp. 318–331, Apr. 2009.

[67] J. J. Xu, "Are blockchains immune to all malicious attacks?" *Financial Innov.*, vol. 2, no. 1, p. 25, 2016.

[68] S. Y. Yang and J. Kim, "Bitcoin market return and volatility forecasting using transaction network flow properties," in *Proc. Symp. Ser. Comput. Intell. (IEEE-SSCI)*, Dec. 2015, pp. 1778–1785.

[69] H. S. Yin and R. Vatrapu, "A first estimation of the proportion of cybercriminal entities in the Bitcoin ecosystem using supervised machine learning," in *Proc. Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 3690–3699.

[70] D. Zambre and A. Shah, "Analysis of Bitcoin network dataset for fraud," Stanford CS 224W Project Final Rep.-Group 30, Dec. 2013.

[71] M. Zhdanova, J. Repp, R. Rieke, C. Gaber, and B. Hemery, "No smurfs: Revealing fraud chains in mobile money transfers," in *Proc. Int. Conf. Availability, Rel. Secur. (ARES)*, Sep. 2014, pp. 11–20.

[72] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *Proc. Int. Congr. Big Data (BigData Congress)*, Jun. 2017, pp. 557–564.

MOHAMED RAHOUTI received the M.S. degree in statistics from the University of South Florida in 2016, where he is currently pursuing the Ph.D. degree in electrical engineering. He holds numerous academic achievements. His current research focuses on computer networking, software-defined networking, and network security with applications to smart cities.

KAIQI XIONG received the Ph.D. degree in computer science from North Carolina State University. He was with IT industry for several years. He is currently an Associate Professor with the Intelligent Computer Networking and Security Laboratory, University of South Florida, where he is also with the Florida Center for Cybersecurity, the Department of Mathematics and Statistics, and the Department of Electrical Engineering. His research was supported by the National Science Foundation (NSF), NSF/BBN, the Air Force Research Laboratory, Amazon AWS, the Florida Center for Cybersecurity, and the Office of Naval Research. His research interests include security, networking, and data analytics, with applications such as cyber-physical systems, cloud computing, sensor networks, and the Internet of Things. He was a recipient of U.S. Ignite Application Summit with his team in 2015. He received the Best Demo Award at the 22nd GENI Engineering Conference. He also received the Best Paper Award at several conferences.

NASIR GHANI received the Ph.D. degree in computer engineering from the University of Waterloo, Canada, in 1997. He was a Faculty Member with Tennessee Tech University from 2003 to 2007. He was an Associate Chair of the Electrical and Computer Engineering Department, The University of New Mexico, from 2007 to 2013. He also spent several years in industry at large Blue Chip organizations (IBM, Motorola, and Nokia) and hi-tech startups. He is currently a Professor of electrical engineering with the University of South Florida, where he is also the Research Liaison with Cyber Florida, a state-based center focusing on cybersecurity research, education, and outreach. He has authored over 230 peer-reviewed articles and holds several highly cited U.S. patents. He also received the U.S. National Science Foundation (NSF) Career Award in 2005. His research interests include cyberinfrastructure networks, cybersecurity, cloud computing, disaster recovery, and Internet of Things/cyberphysical systems. He has served as an Associate Editor for the IEEE/OSA JOURNAL OF OPTICAL AND COMMUNICATIONS AND NETWORKING, IEEE SYSTEMS, and IEEE COMMUNICATIONS LETTERS. He has also been a guest editor of the special issues of the *IEEE Network and IEEE Communications Magazine* and has chaired symposia for numerous flagship IEEE conferences. He was also the Chair of the IEEE Technical Committee on High Speed Networking from 2007 to 2010.