A survey of blockchain from security perspective

Abstract
This report provides an overview of blockchain security, highlights security threats, and summarizes recent research on the subject. It emphasizes that blockchain is a growing technology with disruptive potential but faces significant security challenges. The paper classifies these concerns and explores current trends in blockchain development.

Potential vulnerabilities in blockchain

1. Vulnerability of cryptographic operations

A. Cryptographic key vulnerability

There is skepticism surrounding the NIST P-256 curve due to potential manipulation and intentional weaknesses. The report highlights the example of Dual_EC_DRBG, a random number generator, where a backdoor was suspected. The secp256k1 curve used in Bitcoin and Ethereum is considered non-backdoored, but has deficiencies. Curve25519 is preferred in some blockchain implementations due to well-explained parameters and fewer limitations. The section also mentions the susceptibility of elliptic curve cryptography (ECC) to side-channel attacks and the importance of secure random number generation. It briefly discusses the vulnerability of SHA-256 to length extension attacks and birthday attacks, advising against the use of broken algorithms like MD5 and SHA-1 in cryptography.

B. Hashing operation vulnerability

SHA-256 is commonly used in blockchain for hashing operations, along with other algorithms like Ripemd160 and sCrypt. While SHA-256 is considered strong, it is vulnerable to length extension attacks. To mitigate this, using double SHA-256 has been suggested. Hash functions, including SHA-256, are also susceptible to birthday attacks, as demonstrated with SHA-1. Therefore, broken algorithms like MD5 and SHA-1 should never be used for cryptographic operations.

2. Identity vulnerability

A. Replay attack

Replay attacks, where an attacker impersonates valid parties and gains unauthorized access, pose a common threat to the blockchain community. Key pair-based exchange protocols are effective in mitigating these attacks. Some blockchains utilize one-time private-public key pairs or elliptic curve-based encryption to detect and protect against replay attacks.

B. Impersonation attack

Impersonation of legitimate users is another method used to gain unauthorized access. The use of ECDSA algorithm can provide defense against such attacks, along with other proposed methods like distributed incentive-based approaches. BSeiN utilizes attribute-based signatures for user validation.

C. Sybil attack

Sybil attacks involve creating multiple fraudulent identities controlled by a single entity, aiming to isolate a target node in a peer-to-peer network. In blockchain networks, Sybil attacks can be used to launch various attacks such as refusing to relay transactions, conducting double spending attacks, filtering transactions, or compromising anonymity protocols. TrustChain, a blockchain framework, addresses this issue by creating an immutable chain based on user interactions and computing trustworthiness. It employs a system called "netflow" to ensure resource sharing among network nodes.

3. Manipulation based attacks

A. Eclipse attack

Eclipse attacks involve isolating a target by monopolizing its incoming and outgoing connections in order to manipulate its view of the blockchain or misuse its compute power. The attacker aims to corrupt the target's network communication. This attack primarily targets nodes that accept incoming connections.

B. Transaction malleability

Transaction malleability is a design flaw in Bitcoin where transactions can be altered before being added to a block, resulting in a different transaction ID (TXID). Attackers exploited this flaw at Mt. Gox, a Bitcoin exchange, by modifying TXIDs and fraudulently withdrawing funds. While Mt. Gox cited this flaw as a primary reason for their bankruptcy, subsequent research has questioned its sole attribution to their downfall.

C. Timejacking

Timejacking is an attack where a malicious node skews the timestamp of a target node by providing incorrect time information. By manipulating the network time, the attacker can isolate the target and send fraudulent transactions. This attack can also be used to manipulate mining pools. Countermeasures such as network encryption, randomized port negation, UDP heartbeats, retrieving block data from multiple nodes, diversifying node connections, and round-trip time monitoring can help mitigate these attacks.

4. Quantum vulnerability

The advent of quantum computing poses a threat to popular cryptographic algorithms like ECC and ECDSA, which rely on the elliptic curve discrete logarithm problem (ECDLP). Quantum computers using modified versions of Shor's algorithm can potentially break ECC. New post-quantum cryptographic algorithms based on supersingular elliptic curves, lattice-based constructions, multivariate polynomials, and hash functions are being developed to address this vulnerability. Blockchain implementations like Quantum Resistant Ledger aim to avoid this weakness. Wallet addresses in cryptocurrencies are hashes of public keys, providing some resistance against quantum attacks. While hash functions like SHA-256 are not entirely quantum-resistant, breaking their preimage resistance still remains computationally difficult.

5. Reputation based attack

Changing user reputation from negative to positive and manipulating the blockchain framework is a significant concern in the blockchain community. This can be achieved by hiding negative transactions or creating new accounts. Currently, there are limited approaches to address this issue. Trustcoin has proposed some detection-based measures, but prevention-based solutions are still lacking.

6. Vulnerability in service

A. Race attack

The race attack involves an attacker creating two transactions, one genuine and one fraudulent. By sending the fraudulent transaction to a target node and the legitimate transaction to a mining pool, the attacker aims to deceive the target into accepting the fraudulent transaction before seeing the legitimate one. It is advisable to wait for a minimum number of confirmations before considering a transaction valid to mitigate this attack.

B. DDoS attack

DDoS attacks pose a threat to blockchain and asset exchange networks. Attackers exploit hijacked devices to overwhelm the network with excessive requests, causing service disruptions. While blockchain networks may have some protection against DDoS attacks, they remain vulnerable to more sophisticated forms of attack. Incidents like the DDoS attack on Bitcoin Gold's launch and the high number of DDoS attacks on Bitcoin-related sites highlight the prevalence of this threat.

C. Double spending attack

Bribery attacks involve creating a legitimate transaction, waiting for confirmations, and then introducing a fraudulent conflicting transaction in a new block. The attacker attempts to extend the fraudulent branch by bribing or renting mining power until it becomes the longest branch, validating the fraudulent transaction.

D. Finney attack

The Finney attack is a double-spending attack where a pre-mined block contains a fraudulent transaction. The attacker simultaneously uses the same coins for a legitimate transaction with a target node. Once the target accepts the transaction, the fraudulent block is broadcasted, rendering the transaction to the target invalid. Waiting for additional confirmations can prevent this attack.

E. vector76 attack

The vector76 attack combines the race and Finney attacks to exploit targets with only one confirmation requirement. The attacker establishes connections with the target and a mining pool. A large fraudulent transaction is made to deposit tokens into the target's E-Wallet, while a small transaction with the same tokens is sent to the miners. Once a block is found, both the fraudulent transaction and the small transaction are broadcasted. The attacker quickly withdraws the credited tokens from the E-Wallet while the rest of the network accepts the small transaction, invalidating the large fraudulent transaction.

F. Collusion attack

The 51% attack occurs when a single entity or group controls over 50% of the mining power in a blockchain network. This control allows them to manipulate the blockchain's history, including accepting fraudulent transactions through double spending. By outpacing the mining output of the rest of the network, the colluding group can create a longer fraudulent chain, overriding the legitimate chain. This attack undermines the integrity and consensus of the blockchain.

7. Malware attacks

The rise in cryptocurrency value has led to the emergence of various malware targeting cryptocurrency theft and blockchain network disruption. One notable malware is covert cryptocurrency mining software hidden in JavaScript applications executed by web browsers. This software, like Coinhive, has affected numerous businesses, including popular platforms like YouTube. Malicious mining payloads have been found in browser extensions and even delivered through leaked NSA malware. Another concern is the immutability of the blockchain, allowing malware and unwanted content to be stored indefinitely. Miners are also targeted, with instances like the Antbleed backdoor introduced by Bitmain, which could potentially disable a significant portion of mining equipment and be exploited through attacks like MITM or DNS hijacking.

8. Application vulnerability

A. Use case design flaw

The Ethereum blockchain is known for its smart contract and dapp capabilities. Decentralized Autonomous Organizations (DAOs) allow individuals to contribute funds and vote on proposals. "The DAO" was a prominent DAO that was hacked due to a bug, resulting in the theft of over 3.6 million ether. The bug allowed recursive splitting of the DAO into a child DAO, enabling the attacker to drain funds. To address this, Ethereum underwent a fork, resulting in two chains: Ethereum, where the hack was rolled back, and Ethereum Classic, where the hack remains in the chain's history.

B. Coding error

In November 2017, a bug in the Parity wallet software led to the accidental destruction of 513,743 ether (worth around $355 million). The bug allowed someone to take ownership of the library contract, which was used by all multisig wallets in Parity. By initializing and subsequently killing the library contract, the funds became irretrievable. This incident is comparable to walking into an open bank vault and intentionally destroying all the money.