How Cryptography is used in Bitcoin

Taylor Adams, CS 548, and tkadams1@crimson.ua.edu

Abstract— **Over the past decade Bitcoin and blockchain technology have ballooned into a major world currency. At the time of writing Bitcoin's market cap is at $777 billion dollars, with the global cryptocurrency market cap at $1.9 trillion dollars. How is all this capital protected? What cryptographic measures take place in the bitcoin algorithm to ensure the authenticity and ownership of every Satoshi (Bitcoin cent)? This paper aims to provide the answers to these questions and provide the reader with an understanding of the cryptography behind the Bitcoin network.**

Keywords: Bitcoin, Cryptocurrency, Cryptography, ECDSA, RSA, SHA-256,

I. A Brief Introduction to the History of Bitcoin

The first real-world application of blockchain technology was discussed in 2008 by Satoshi Nakamoto in his whitepaper titled, "*Bitcoin A Peer-to-Peer Electronic Cash System*". Nakamoto lays the foundation for a decentralized way to transfer funds between untrusted sources, Bitcoin. While Bitcoin is a transaction system; it is decentralized and therefore relies heavily on public key cryptography and many other supporting cryptographic measures.

II. Defining a Blockchain

A blockchain in its most basic form is a database. It stores data and keeps records. Blockchain differentiates itself from traditional databases in it is a distributed ledger. While a distributed ledger might sound complex it is just two parts: a ledger (a book or other collection of financial accounts of a particular type) and that ledger is distributed among many different computers. This is the general idea of what a blockchain is: a ledger of records copied across many different computers. Having multiple copies of a ledger is important for data availability. This is where the core of blockchain's benefits lie: validating the authenticity of a transaction. Having a ledger distributed among many computers, allows the ledger to validate transactions and ensure they are real when an entry is placed on a blockchain. For example, how do you validate someone owns a particular bitcoin on the bitcoin blockchain? Every specific coin has a unique identifier stored on the blockchain and every person who owns a Bitcoin has a Bitcoin wallet (Each wallet has a unique address). Every time a transaction occurs on the Bitcoin blockchain the transaction is stored, and a record of the transfer is kept on the blockchain. This allows the ownership of a Bitcoin to be verified throughout its lifetime, from its creation to its most recent holder.

III. Cryptographic Algorithms in Bitcoin

A. Elliptic Curve Digital Signature algorithm (ECDSA)

On the Bitcoin Blockchain ECDSA is utilized as the digital signature algorithm. ECDSA utilizes elliptic curve cryptography. The implementation of ECDSA in Bitcoin utilizes the curve order Secp256k1, which refers to the parameters of the actual elliptic curve that is used in the bitcoin blockchain. The general format of an elliptic curve algorithm can be mathematically defined as:

$$y^2 = x^3 + ax + b$$

Fig 1: Generalized mathematical function of an elliptic curve [7]

For the ECDSA implementation in Bitcoin the following mathematical equation is used:

$$y^2 = x^3 + 7$$

Fig 2: Elliptic curve function in bitcoin [7]

The goal of ECDSA is to guarantee "Unique and unrepeatable signatures for each generation set private keys and public" [19] and make it practically impossible to falsify digital signature (or computationally secure).

The following is a generic example of how the ECDSA algorithm generates private keys on the Bitcoin blockchain.

<u>Steps of ECDSA on the bitcoin blockchain</u>

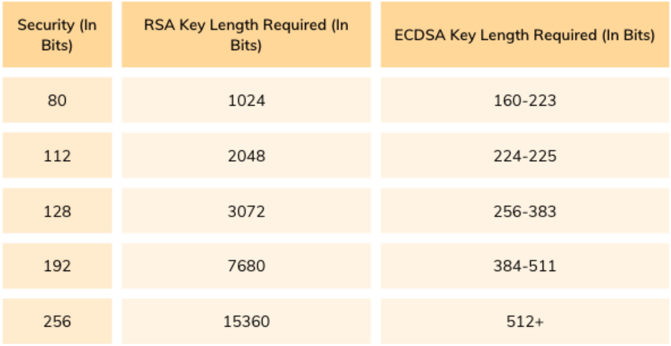
1. An elliptic curve is generated using the mathematical function for the ECDSA implementation
2. A random point is chosen on the elliptic curve.
3. A random number of 256 bits is generated, which is the private key.
4. Using the private key and point of origin, another equation is performed, giving the public key.

Manuscript received April 30, 2022

   a. The relationship between the point of origin and the public key provides the relationship between the public and private key.
   b. This process is a one-way function and does not provide any (known) meaningful way to derive the private key with the knowledge of the public key and point of origin, therefore establishing ECDSA as a secure digital signature algorithm
5. The public key is then condensed using a base59 encoding, SHA256, and RIPEMD160.
   a. This allows for shorter public keys when conducting transactions.

Benefits of ECDSA over RSA

Typically, RSA is used as a standard for digital signatures, given its history. However, ECDSA provides several benefits over RSA, hence why it was chosen to be utilized in the bitcoin blockchain. These advantages include shorter key lengths for higher security (see table in fig 3 below), increased complexity due to elliptic curve cryptography, and ECDSA offers better performance at higher security levels.



Security (In Bits)	RSA Key Length Required (In Bits)	ECDSA Key Length Required (In Bits)
80	1024	160-223
112	2048	224-225
128	3072	256-383
192	7680	384-511
256	15360	512+

Fig 3: Security relative to key size for RSA and ECDSA [8]

B. *AES-256-CBC*

AES is utilized by Bitcoin in a limited capacity. One of the core principles of a blockchain is that all transactions are made transparent. This is partly why public key cryptography is so import for bitcoin and blockchains in general. While the bitcoin blockchain does not utilize AES, Bitcoin Core does.

AES-256-CBC is utilized for wallet protect in Bitcoin Core. This essentially protects wallet.dat files which contain information like “your private keys, public keys, scripts, key metadata, and transactions related to your wallet." [2]. It is critical this data is hidden from prying eyes. If a private key were made public, all unspent transaction outputs (bitcoin) associate with that public-private key pair would be at risk of being stolen.

AES-256-CBC was chosen for this role due to its track record for being a known secure symmetric key encryption algorithm.

C. *SHA-256*

The SHA-256 hashing algorithm is utilized in Bitcoin mining (see section VI of this paper for an understanding of proof-of-work) as well as in hashing public keys into bitcoin addresses. When mining blocks Bitcoin uses SHA in the following equation:

$$A = \text{SHA-256}(\text{SHA-256}(\text{Block\_Header}))$$

Fig 4: SHA-256 use case in mining blocks [5]

SHA-256 is also used as seen below in Fig 5 with the RIPEMD160 algorithm to hash bitcoin public keys.

D. *RIPEMD160*

RIPEMD160 is another hashing algorithm which is utilized in Bitcoin. It is based on the Merkle—Damgård construction. To summarize the algorithm, “The compression function is made up of 80 stages made up of 5 blocks that run 16 times each. This pattern runs twice with the results being combined at the bottom using modulo 32 addition” [18]. The sole purpose of this hashing algorithm is to convert public keys to a bitcoin address. When converting a public key to a bitcoin address, the public key is initially hashed with SHA-256 to provide an additional layer of security to the public key. The output of the SHA-256 algorithm is then passed to the RIPEMD160 hashing algorithm resulting in the bitcoin address which is 160 bits long. Combining the use of both SHA-256 and RIPEMD160 produces greater security than if SHA-256 had been omitted from the algorithm.

$$A = \text{RIPEMD160}(\text{SHA-256}(K))$$

Fig 5: RIPEMD160 algorithm use in bitcoin [16]

The reason RIPEMD160 is used is to securely shorten the bitcoin address. RIPEMD “produces the shortest hash function whose uniqueness is still sufficiently assured.” [17] Essentially, RIPEMD160 was chosen because it produces the smallest hash which avoids collisions. This allows for transactions to be more user friendly as the user is only required to type a 160-bit address to send bitcoin to, rather than a 256-bit address.

IV. Public Key Cryptography In Use

As mentioned earlier in this paper ECDSA with the secp256k1 elliptic curve is the digital signature algorithm used on the bitcoin blockchain. This section of the paper aims to convey how these public and private key pairs are used on the bitcoin blockchain.

In order for a transaction to occur on the blockchain network both the sender and receiver need a public private key pair generated by the ECDSA algorithm. Sending a transaction on the bitcoin network requires the receiver to give the sender their public key hash. The sender then creates a Pays-to-Pubkey hash (P2PKH) transaction output as seen above. The sender then broadcast this transaction (This becomes an Unspent Transaction Output or UTXO) on the network, and anyone with the matching private key can claim the output of the transaction. Essentially, "pubkey scripts and signature scripts combine secp256k1 pubkeys and signatures with conditional logic, creating a programmable authorization mechanism."[15]. See figures 6 and 7 below for a visual representation of both a UTXO and a P2PKH.

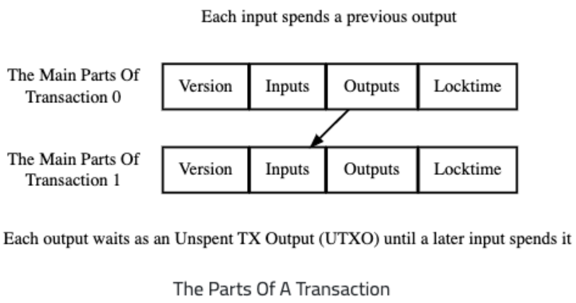

Each input spends a previous output
The Main Parts Of Transaction 0
Version
Inputs
Outputs
Locktime
The Main Parts Of Transaction 1
Version
Inputs
Outputs
Locktime
Each output waits as an Unspent TX Output (UTXO) until a later input spends it
The Parts Of A Transaction

Fig 6: Unspent Transaction Output (UTXO) [15]

On the blockchain whenever a UTXO is spent it is destroyed. In its place one or two additional UTXOs are created depending on if the full amount of the original UTXO was spent or not. This ensures Bitcoin is not duplicated on the network.

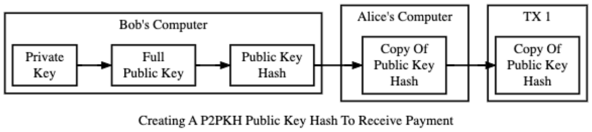


Fig 7: Pays-To-Public-Key-Hash (P2PKH) [15]

V. Bitcoin Mining (Proof-Of-Work)

The goal of mining on a blockchain is twofold. It provides a way to mint new bitcoin fairly across the network, and acts as a method to secure transactions that occur on the network. This “proof-of-work” is used to create fair competition among miners on a blockchain. Every new block to be added on a blockchain will be given a target.

Every miner who enters the network will set up a mining node on the network. Each miner must vary the nonce of the block and hash it until they hit a value less than or equal to the target value. When the miner gets a hash value below the target it is allowed to add a block to the blockchain. The miner who got the hash value below the target first receives the block reward. This block reward incentivizes individuals to utilize their computer’s processing power for the blockchain, establishing a distributed network. See figure 8 and 8 for visual representations of a block on a blockchain.

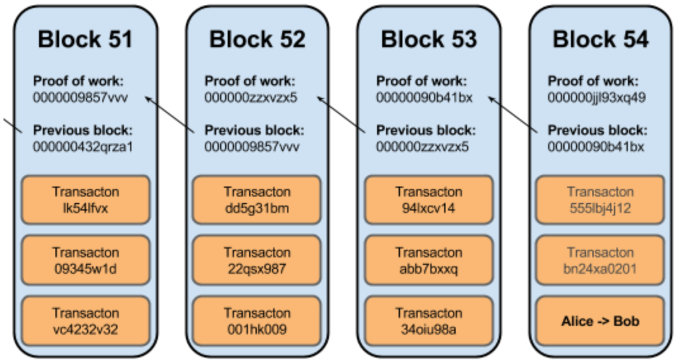


Fig 8: Simplified Block example on the bitcoin blockchain [13]

See additional figure below for a more detailed view of a block on the bitcoin blockchain.

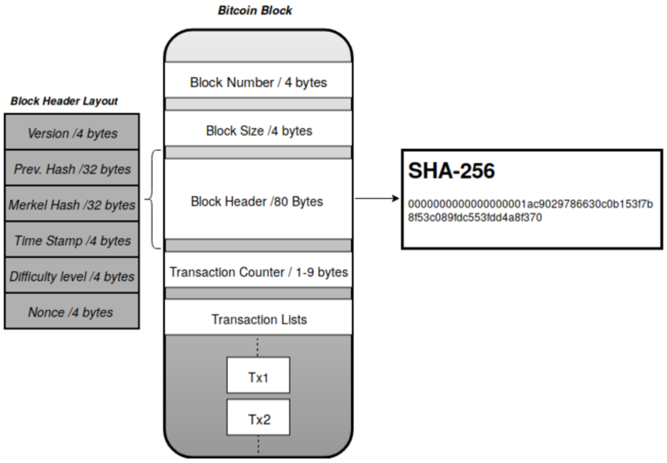


Fig 9: Detailed Block on the bitcoin blockchain [1]

VI. Discussions

Cryptography’s role in bitcoin is of vital importance to not only the bitcoin network, but blockchain as a technology. The digital signature algorithm ECDSA, is a fundamental building block for transactions to occur in a decentralized manner. SHA-256 provides the backbone for proof-of-work and drives the ability of new blocks to be mined to the bitcoin blockchain. SHA-256 not only regulates the supply of bitcoin available, but also plays a role in validating transactions through mining. AES-256-CBC adds a layer of security to Bitcoin Core where local protection of private keys is of the upmost importance. RIPEMD160 provides a quality-of-life improvement to users of Bitcoins blockchain, in securely shortening the length of Bitcoin address which need to be typed in every transaction.

Cryptography is the major machinery which drives Bitcoin, a secure blockchain currently storing $777 billion dollars of wealth.

References

[1] S. Sayeed and H. Marco-Gisbert, “Assessing Blockchain Consensus and Security Mechanisms against the 51% Attack,”

Applied Sciences, vol. 9, no. 9, Art. no. 9, Jan. 2019, doi: 10.3390/app9091788.

[2] “Bitcoin,” Wikipedia. Apr. 19, 2022. Accessed: Apr. 19, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Bitcoin&oldid=1083647464

[3] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” p. 9.

[4] R. J. Rybarczyk, “Bitcoin P2PKH Transaction Breakdown,” Coinmonks, Aug. 31, 2020. https://medium.com/coinmonks/bitcoin-p2pkh-transaction-breakdown-bb663034d6df (accessed Apr. 19, 2022).

[5] “Block hashing algorithm - Bitcoin Wiki.” https://en.bitcoin.it/wiki/Block_hashing_algorithm (accessed Apr. 30, 2022).

[6] “Cryptocurrency Prices, Charts And Market Capitalizations,” CoinMarketCap. https://coinmarketcap.com/ (accessed Apr. 13, 2022).

[7] “ECDSA | River Glossary,” River Financial. https://river.com/learn/terms/e/ecdsa/ (accessed Apr. 29, 2022).

[8] “ECDSA vs RSA: Everything You Need to Know,” InfoSec Insights, Jun. 09, 2020. https://sectigostore.com/blog/ecdsa-vs-rsa-everything-you-need-to-know/ (accessed Apr. 19, 2022).

[9] “Elliptic Curve Digital Signature Algorithm - Bitcoin Wiki.” https://en.bitcoin.it/wiki/Elliptic_Curve_Digital_Signature_Algorithm (accessed Apr. 29, 2022).

[10] “How Bitcoin Uses Cryptography | River Learn - Bitcoin Technology,” River Financial. https://river.com/learn/how-bitcoin-uses-cryptography/ (accessed Apr. 19, 2022).

[11] “How bitcoin works - Bitcoin Wiki.” https://en.bitcoin.it/wiki/How_bitcoin_works (accessed Apr. 19, 2022).

[12] “Promo Only (for guides iframe).” https://www.pluralsight.com/utilities/promo-only?noLaunch=true (accessed Apr. 19, 2022).

[13] “Thoughts on Bitcoin Block Size Economics | Bitcoinist.com,” Feb. 02, 2015. https://bitcoinist.com/thoughts-bitcoin-block-size-economics/ (accessed Apr. 29, 2022).

[14] “Token Security: Cryptography - Part 2,” BlockchainHub, Sep. 10, 2018. https://blockchainhub.net/blog/blog/cryptography-blockchain-bitcoin/ (accessed Apr. 19, 2022).

[15] “Transactions — Bitcoin.” https://developer.bitcoin.org/devguide/transactions.html (accessed Apr. 19, 2022).

[16] Patrick, “What Is SHA-256 And How Is It Related to Bitcoin?,” Mycryptopedia, Apr. 24, 2022. https://www.mycryptopedia.com/sha-256-related-bitcoin/ (accessed Apr. 29, 2022).

[17] D. Schwartz, “Answer to ‘Why does Bitcoin use two hash functions (SHA-256 and RIPEMD-160) to create an address?,’” Bitcoin Stack Exchange, Apr. 05, 2013. https://bitcoin.stackexchange.com/a/9216 (accessed Apr. 30, 2022).

[18] “RIPEMD-160 - Bitcoin Wiki.” https://en.bitcoin.it/wiki/RIPEMD-160 (accessed Apr. 30, 2022).

[19] “What is the ECDSA signing algorithm?” https://academy.bit2me.com/en/que-es-ecdsa-curva-eliptica/ (accessed Apr. 30, 2022).