Caleb Walter Bowden

U2166829

CHS2401-2324 Advanced Cyber Security Assignment

The University Of Huddersfield

Abstract 2
Introduction 3
Related / Background work 4
Scalability In Blockchains 4
State of Blockchain Scalability 4
Solutions for Scalability 4
Future of Scalability in Blockchain 4
PoW Complexity 5
Understanding PoW Complexity 5
PoW Impact on Blockchain Efficiency 5
Innovations Reducing PoW complexity 5
Security Threats in Blockchain Systems 6
A Catalog of Security Threats 6
Mitigating Security Threats 6
Evolving Nature of Security In Blockchains 7
Method 7
Scalability Of Blockchains 7
Figure 1: code to create multiple transactions 7
Figure 2: Data Collecting 8
Impact Of PoW Complexity 8
Figure 3: PoW complexity code 8
Figure 4: Changing hashing algorithm 8
Security Threats In Blockchain Systems 9
Figure 5: Modifying blockchain 9
Figure 6: Finding hash collision vulnerability code 10
Experimental Analysis 10
Scalability Of Blockchains 10
Table Showing Impact Of Transaction Volume On Blockchain Performance - Difficulty 1 (figure 7) 10
Graph showing the RAM Usage in MB vs Number of Transactions Per 100 - Difficulty 1 (figure 8) 10
Duration to compute hash (MS) vs Number of Transactions Per 100 - Difficulty 1 (figure 9) 11
CPU usage (%) vs Number of Transactions Per 100 - Difficulty 1 (figure 10) 11

Impact Of PoW Complexity 12
Table Showing Time Taken For Transactions to be Verified At PoW difficulty 1 - SHA256 (fig 11) 12
Graph Showing Average Time For Transactions to be Verified at PoW difficulty 1 - SHA256 (fig 12) 12
Table Showing Time Taken For Transactions to be Verified At PoW difficulty 1 - SHA512 (fig13) 13
Graph Showing Average Time For Transactions to be Verified at PoW difficulty 1 - SHA512 (fig 14) 13
Table Showing Time Taken For Transactions to be Verified At PoW difficulty 1 - MD5 (fig 15) 14
Graph Showing Average Time For Transactions to be Verified at PoW difficulty 1 - MD5 (fig 16) 14
Table Showing Time Taken For Transactions to be Verified At PoW difficulty 2 - SHA256 (fig 17) 15
Graph Showing Average Time For Transactions to be Verified at PoW difficulty 2 - SHA256 (fig 18) 15
Table Showing Time Taken For Transactions to be Verified At PoW difficulty 2 - SHA512 (fig 19) 16
Graph Showing Average Time For Transactions to be Verified at PoW difficulty 2 - SHA512 (fig 20) 16
Table Showing Time Taken For Transactions to be Verified At PoW difficulty 2 - MD5 (fig 21) 17
Graph Showing Average Time For Transactions to be Verified at PoW difficulty 2 - MD5 (fig 22) 17
Security Threats In Blockchain Systems 18
Showing how the blockchain reacts to the modification of blocks (figure 23) 18
Showing the MD5 hashing algorithms collision vulnerability to 4 characters (figure 24) 18
Discussion **18**
Scalability Of Blockchains 18
Impact Of PoW Complexity 19
Security Threats In Blockchain Systems 19
Conclusion **20**
References **20**

Abstract

Within this paper, three main themes are covered about blockchain technology. Scalability, PoW complexity & Security threats that Blockchains are susceptible to. The paper starts with an introduction to blockchain technologies and how they are used. It then expands to related background research that has taken place about the three categories that this paper covers about blockchain technologies. This study then goes on to explain the methods that were chosen by the researcher for the experimental analysis it was decided for the scalability method the researcher would create transactions in blocks of 100 and measure the CPU and RAM usage, in addition to the time required to mine these blocks. The researcher then goes on to explain their method for gathering data for the PoW complexity experiment. The decision

was made to use the same code that was used to collect data on the scalability experiment. However this time the PoW difficulty would be changed along with the different types of hashing algorithms to see how this would affect the difficulty and time complexity of mining hashes. The next experiment was checking for security threats within a blockchain system, this method consisted of 2 experiments. Experiment 1 was to check if the blockchain would be valid if the researcher were to modify the blockchain and experiment 2 was to check for MD5 hash collision within the network. The study subsequently progresses to the experimental analysis where data & graphs are shown from the previous experiments. Next, the investigator discusses their findings and concludes with what they have found. The key findings of this paper are that blockchains are very secure and shrug off most attacks, PoW complexity & scalability are correlated, with scalability being an issue that blockchains could face using the PoW algorithm.

Introduction

The emergence of Blockchain technology with Bitcoin was a milestone in the history of decentralized security because it gave users the opportunity for trustless interactions across a network. The most visible impact of the innovative cryptocurrency technology is perhaps the idea of the use of distributed ledgers for recording transactions in a permanent and verifiable way, which has more far-reaching effects than just finance, laying the foundations for a new perspective on industries such as supply chain management. Blockchain is fundamental to the distributed database, secure cryptography, time stamps, and transaction data, lined in blocks to be chained. This architecture maintains the sequence and ultimate trust in the process. The idea of a distributed network of peers, in which there is no controlling entity handling everything, is the core of Blockchain. Hence, its stability is ensured from the remaining censorship and fraud. The attributes of Blockchain are being transparent and having the characteristics of immutability along with its consensus mechanisms of decentralization, which have undoubtedly brought on trust among its users.

Scalability is always an essential issue in the development of blockchain tech, The challenge is to find the optimal infrastructure that will meet the growing needs of the blockchain in terms of throughput, speed, and cost-related issues. The existing problems, including the capacity to process transactions and bottlenecks, may hinder their adoption and raise concerns about whether the scaling solutions will keep pace to allow sustainable growth for Blockchain, Also, the fact that scalability is carried out agilely helps to bring about evolution without compromising on the decentralized character that is such an attractive element of the whole venture. The core of PoW is blockchain security, which makes it difficult to access the system when it is controlled by no one and without any centralized authority. It also prevents double spending at the same time. In Pow, players compete to solve the most complex problems. The lucky one who can verify and solve the block is rewarded. This process certainly minimizes risk but requires intensive computer work, and as such it raises some questions about efficiency and ecology.

Nevertheless, attack vectors including 51% attacks where an actor gains the majority of the mining power of the remaining network, can be problematic for the network. Intelligent contract vulnerabilities, code exploits, and quantum computing make breaching the algorithms even more accessible, but they are also significant threats. These concerts focus on keeping the technology from losing the investigation and keeping up with continuous updates in Blockchain security to counter emerging risks. This paper explores the current impact level of blockchain scalability, gets underneath the cover of the Proof of Work, and finally accesses the dangerous side of the blockchain ecosystem. This paper also shows the attack methods and includes an experimental analysis with a discussion explaining how this method is used. By identifying these core issues, we want to analyze how prepared technology is for wide-range implementation and also emphasize the areas that require more diligence to make blockchain technology a great success.

Related / Background work

Scalability In Blockchains

State of Blockchain Scalability

The scalability of Blockchain remains the major hindrance to the mass adoption of blockchain technology, primarily due to the limitations in throughput, storage, and network capacity as the major issues. Scientists have examined many blockchain mechanisms, including cryptography authentication, transaction tracking, and governance, which are central to blockchain systems. Xie et al. (2019) give an exhaustive analysis of throughput, providing a holistic approach to explain the crucial need for a sharp and simultaneous increase in transaction rates and decentralized integrity without harming each other's nature. Various scalability reiterations were the subject of the survey throughout its entirety, encompassing the creation of new consensus algorithms to integrate the most advanced data structures. Although a few transactions per second constrain traditional blockchains such as Bitcoin, groundbreaking solutions are re-designing the blockchain infrastructure to cater to the rapidly increasing user base and transaction volumes. This research study also acknowledges scalability as a non-linear path. Requiring a delicate balancing act of elevating the network's scale and maintaining its initial ideals of security and decentralization (Xie et al., 2019)

Solutions for Scalability

The struggle to eliminate the blockchain scalability hurdle has sped up breakthroughs in developing solutions that aim to boost transaction throughput. Khan et al. (2021) demonstrated many Onchain and Offchain strategies but prioritized the Segmented Witness (SegWit) method since it increases the block capacity and the Lightning Network. In these cases, the main chain is not burdened due to the velocity, which makes the transaction speedy. A systematic literature review of the authors' features has dissected these methods. Now, it is time to have an honest look at the results. Authors present us with an extensive evaluation of their efficiency in scaling blockchain backbones (Khan et al., 2021). Zhong et al. (2020) further expand these thoughts when they deliberately classify scalability strategies and express in their outcomes the extent of their performance advancements. The authors also deal with each technique, the research offers vital information about the unlimited capabilities of these scaling techniques to raise blockchain systems to meet real-world demands and keep the basic principles of decentralization and security in situ (Xhou et al., 2020).

Future of Scalability in Blockchain

While blockchain technology improves, the future of its scalability is wide open, with promising innovations that will build more scalable architectures. The appearance of more alternative blockchain protocols like HTNZ ushered in this transformation. As Sohrabi and Tari (2020) point out, this is believed to be due to auxiliary concepts like "sideBlcok," which will drastically increase transaction processing capacities. This method leverages a novel channel for scaling and maintains the decentralization aspect of which Blockchain prides itself. The emerging reality of scalability for Blockchain now looks on the verge of solidifying such groundbreaking methodologies into operational layers, cementing them as practically viable and allowing the transformational models to break the usage barriers. It is a fact that current research is witnessing the emerging trend of adopting the practical approach, in which theoretical constructs are applied. In this way, new methods integrate breakthroughs into the practical ecological system. The transition here that spans from theoretical to applicable attests to the fact that the blockchain community has been dedicated to solving the scalability problem. With “Researchers” and “practitioners” coming together to make these processes possible, the Blockchain will proudly pass through a new and magnificent level and become one of the most essential digital infrastructures in the world. It is worth highlighting that the combination of different approaches will be decisive in surpassing the existing constraints, enabling the knowledge of what Blockchain is capable of.

PoW Complexity

Understanding PoW Complexity

The Proof Of Work (PoW) consensus mechanism is one of the specifies in many blockchain networks, bringing security and decentralization to a new level. In PoWs, miners work difficult math solving to ensure the transactions are validated and a new block is created. The validation of transactions can significantly cut into crypto life because it often causes delays in energy consumption and traction speed. However, PoW has adequately been dismissed by the researchers for its opposite nature: It implies that there should be an increase in energy use for data safeguarding. On the one hand, the PoW modeling form presented by Chin et al. (2020) reveals the Innards arrangement scheme that plays a crucial role in management.

PoW Impact on Blockchain Efficiency

The height of complexity contained in the PoW consensus mechanism, which is to provide network robustness while demanding a considerable amount of natural resources, is also the main challenge regarding energy efficiency. However, the authorities will decide to look ahead more successfully and re-design the PoW process. This would need the same “security level”, but the functionality should improve. The calibration of these factors represents the very significant mechanisms for regulating the mining difficulty and stable operation of the system. On this note, Chin et al. (2022) study in foresight has argued for automatic difficulty adjustments in blockchains using Genetic Algorithms. This flexible methodology involves the automatic fine-tuning of the mining tasks, which could result in shorter block-generating times (thus improving network rendering). Such a way of a genetic algorithm's computational load fine-tuning could be considered. The turning point in how blockchain networks allocate computational power. Thus their distributed computation abilities fit the network's current processing capacity and workload well. The results of this research are significant as they may be able to reduce the environmental impact of blockchain operations while preserving all the cryptographic security aspects of PoW. Adopting adaptive complexity mechanisms enables us to move to a more sustainable and scalable blockchain framework. That is desired for blockchain technology. It is a step towards long-term adaptation and success (Chin et al., 2022). This concrete example demonstrates the innovative advancements in this direction, as the conflict between competition demand and network security is an ever-lasting problem in blockchain systems.

Innovations Reducing PoW complexity

Quantum-resistant blockchain algorithms became a groundbreaking countermeasure to the soon-to-be problems posed by quantum computing on blockchain integrity. Recent research relies on post-quantum cryptography, including those algorithms developed by NIST. The latter has proved to be much more advanced in terms of security compared to classical cryptography. The research published by Thanakakshmi et al.(2023) highlights a significant improvement in the efficiency of the system with the integration of post-quantum signatures with the blockchain systems rather than trying to defend existing protocols when they are under threat of quantum attacks (Thanalakshmi et al., 2023). This research publishes a hologram of signatures and public keys inside the chain, with the processing content storage on OPFS that would be unquestionable in line with overall blockchain function and agreeable with all optimistic quantum era.

Security Threats in Blockchain Systems

A Catalog of Security Threats

Blockchain networks have an elementary reputation for reliability and security due to their decentralized structure and cryptographic basis. However, as Cheng et al. (2020) demonstrated in their study, the solutions are not impervious to security flaws. The study meticulously categorizes potential threats into five main areas based on the architecture of blockchain technology: the problems associated with poor anonymity, the issues in the P2P network, and the exploits of the consensus mechanism. The attacks hit the network's topology and the intelligent contact flaws. Moreover, 51% of attacks, Sybil attacks, and the vulnerability of intelligent contracts detected are mainly caused by numerous attacks, making the network and its decentralized consistency highly unstable. Security and reliability are the backbone of the blockchain system's integrity, continuous research and development to improve the encryption algorithms ensure that all efforts on both sides are the source of the robust system's confidence in a user. As Cheng et al. (2020) argued in their general classification, a current universal categorization is viewed as a fundamental base for recognizing and dealing with the diversified security challenges peculiar to blockchain network systems.

Mitigating Security Threats

The blockchain technology community has created and accepted many security measures to ensure blockchain systems' resilience and security. According to Sapna (2021), academics and thinkers in the area for decades have maintained that wallets and smart contracts must be secure since they could threaten the system for a while. They comprise the sector that offers a secure channel for both parties to communicate throughout the transaction execution process, making them the leading targets for adversaries. The steps to safeguard these crucial elements start with developing advanced cryptographic techniques, and the employment of stringent security methods comes last. In the meantime, Siddiqui et al. (2020) centrally discuss the dangers of Blockchain by giving a wide range of attacks and reviewing the sector of security where the integration of Blcockhain into the secure system is conifers complex. The comprehensive evaluation underlines the significance of ongoing changes and addressing existing blockchain security issues. However, security hearts have changed, and BDS has secured blockchain technology's digital transactions and applications.

Evolving Nature of Security In Blockchains

The mutually reliant relationship between technical innovation and safety safeguards is illustrated by blockchain security. Zaghloul et al. (2020) underline how the existing blockchain technology offers new ground, particularly in network security and privacy, independent of the domain in which it is employed. Quantum computing, which could challenge blockchain encryption solutions, may complicate this innate inclination toward security. The quantum computing technological disruption is anticipated to produce a huge issue with blocks in cryptographic methods, the security of which is "easy" to breach. This worrying change has sparked a lot of effort to develop better post-quantum cryptographic approaches that provide quantum-resistant signatures, protecting the Blockchain network from new types of potentially damaging attacks that could take advantage of quantum computing capabilities. The need for improvements has sparked a "cold war" of types between hackers and businesses in today's world. Blockchain technology deepens today's digital backbone, raising more complex security difficulties. Innovation and intelligence are needed to stay ahead of these emerging issues. Such a constantly shifting landscape demonstrates how much developers and researchers must continue to develop and deploy security solutions to make blockchains failsafe and stable.

Method

Scalability Of Blockchains

The scalability of a blockchain is pivotal in its success, the blockchain needs to accommodate the growing amount of transactions & blocks on the blockchain network. The method that has been used creates multiple transactions and measures a variety of data like CPU usage, RAM usage in MB, and the time it takes to find and validate the hash in MS. These data points give us a good indication of how the blockchain scales and whether it shall be successful or not.

```
for (int transactionCount = 100; transactionCount <= 1000; transactionCount += 100)
{
    BlockChain blockChain = new BlockChain(proofOfWorkDifficulty: 1, miningReward: 10);
    const string minerAddress = "miner1";
    const string user1Address = "A";
    const string user2Address = "B";

    // Add transactions for the current level
    for (int i = 0; i < transactionCount; i++)
    {
        blockChain.CreateTransaction(new Transaction(user1Address, user2Address, 5));
    }
```

Figure 1: code to create multiple transactions

As you can see above I have modified the code to include a for loop at the start of a main function which will loop through and create 100 transactions up to the threshold of 1000 transactions this is done in increments of 100 which allows us to simulate the different levels of a network load. Doing this helps mimic real-world data where transactions can vary significantly. After the transaction has been created the mining process will begin and will try to validate this transaction and add it to the blockchain during this phase performance metrics like CPU usage & RAM usage are collected using the inbuilt system diagnostic namespace which is part of the .NET framework.

```
// Begin Mining
Console.WriteLine();
Console.WriteLine("--------- Start mining ---------");

Stopwatch stopwatch = Stopwatch.StartNew();
blockChain.MineBlock(minerAddress);
stopwatch.Stop();
```

```
TimeSpan cpuTime = currentProcess.TotalProcessorTime;
double memoryUsage = currentProcess.WorkingSet64;
```

Figure 2: Data collecting

The stopwatch function is used to “timestamp” or to accurately measure the duration of the code execution within the C# blockchain example this function helps us to calculate and measure how long it takes for each block to be mined by the computer while providing insight to the blockchains scalability. The current process is also used to measure the CPU usage % and the Memory Usage. These inbuilt functions provide us with the resources needed to get the data required for the experimental analysis.

Impact Of PoW Complexity

The proof of work algorithm is mostly used within cryptocurrency mining for verifying transactions making sure that no fraudulent transaction gets approved or confirmed within the blockchain. To check the complexity of the PoW algorithm within the C# blockchain example changes can be made to the proof of work difficulty variable change it from 1 to 2 this will increase the complexity or difficulty to mine a block. This modification will be done 2 times, to collect data from when the difficulty is set at 1 to the difficulty level 2. The same code used for the scalability experiment to collect data is also used here.

```
BlockChain blockChain = new BlockChain(proofOfWorkDifficulty: 2, miningReward: 10);
```

Figure 3: PoW complexity code

To get an average time that the blockchain takes to validate transactions. 5 runs on each PoW difficulty will be executed. This will help us to get a more stable figure. Like in the scalability experiment 100 transactions up to the threshold of 1000 transactions will be created keeping the transaction amount the same will help us to see any correlation within the figures. The block.cs file will also be modified to change the hashing algorithm that is used by default the hashing algorithm used is sha256. We will change this algorithm 3 different times. SHA256, SHA512 & MD5 while also changing the PoW difficulty and see how this affects the complexity of the blockchain. To modify the hashing algorithm changes to lines 40 and 51 need to be modified to use the hashing algorithm of choice.

Before:

```
public string CreateHash()
{
    using (SHA256 sha256 = SHA256.Create())
    {
        string rawData = PreviousHash + _timeStamp + _nonce; //c

        var binFormatter = new BinaryFormatter();
        var mStream = new MemoryStream();
        binFormatter.Serialize(mStream, Transactions);
        List<byte> toProcess = new List<byte>();
        toProcess.AddRange(Encoding.UTF8.GetBytes(rawData));
        toProcess.AddRange(mStream.ToArray());

        byte[] bytes = sha256.ComputeHash(toProcess.ToArray());
        return Encoding.Default.GetString(bytes);
    }
}
```

After:

```
public string CreateHash()
{
    using (MD5 md5 = MD5.Create())
    {
        string rawData = PreviousHash + _timeStamp + _nonce;

        var binFormatter = new BinaryFormatter();
        var mStream = new MemoryStream();
        binFormatter.Serialize(mStream, Transactions);
        List<byte> toProcess = new List<byte>();
        toProcess.AddRange(Encoding.UTF8.GetBytes(rawData));
        toProcess.AddRange(mStream.ToArray());

        byte[] bytes = md5.ComputeHash(toProcess.ToArray());
        return Encoding.Default.GetString(bytes);
    }
}
```

Figure 4: Changing hashing algorithm

Security Threats In Blockchain Systems

Blockchain technology is mostly very secure as it uses a trustless system by distributing ledgers across multiple nodes eliminating the need for a central authority that people in the blockchain must trust. Within the C# Blockchain example, we can try and change a previous block in the blockchain by calling the blockchain chain and modifying its contents directly, as shown in the provided code snippet.

```
blockChain.Chain[1].Transactions = new List<Transaction> { new Transaction(user1Address, minerAddress, 150) };
```

Figure 5: Modifying blockchain

Here you can see that we are trying to modify chain 2 and change its data to reflect a new set of transactions falsely trying to send 150 tokens from user1Address to the minerAddress. We can then check if the blockchain is valid after the illegal transaction by calling the IsValidChain function which will return true or false depending on whether the blockchain is valid.

Another method that can be used to check security threats within a blockchain system is an MD5 hash collision when two different inputs produce the same output hash value using a specific hash function. In cryptographic terms, the MD5 hash function could be susceptible to this attack. To do this we will prototype some code that creates a “findHashCollision” function that will save the original hash and nonce value, and then iterate through possible nonce values for each nonce it will compute a hash, and once the hash is computed the function will then check if the new hash matches the original hash. The “findHashCollision” will be called at the end of the program.cs file and the last chain hash will be pasted into it. Because it can take a long time to find a nonce with the same hash, we will focus on finding a hash with the first 4 characters the same as the original one. We will also use the stopwatch to see how long it takes to brute-force 4 characters. Below you can see the prototype code that was used.

```
blockChain.Chain.Last().FindNonceForHashCollision();
```

```
1 reference
public void FindNonceForHashCollision()
{
    var originalHash = this.Hash;
    var originalNonce = this._nonce;
    int characters = 4;

    Console.WriteLine($"Searching for hash collision with the first {characters} characters...\nOriginal nonce: {originalNonce}\nOriginal hash: {originalHash}\n");

    Stopwatch stopwatch = Stopwatch.StartNew();

    bool found = false;
    long testNonce = 0;

    while (!found)
    {
        if (testNonce != originalNonce)
        {
            this._nonce = testNonce;
            var testHash = this.CreateHash();
            if (testHash.StartsWith(originalHash.Substring(0, characters)))
            {
                stopwatch.Stop();
                Console.WriteLine($"Match found! New nonce: {testNonce}\nNew hash: {testHash}\nTime taken: {stopwatch.ElapsedMilliseconds} ms\n");
                found = true;
            }
        }
        testNonce++;
    }
}
```

Figure 6: Finding hash collision vulnerability code

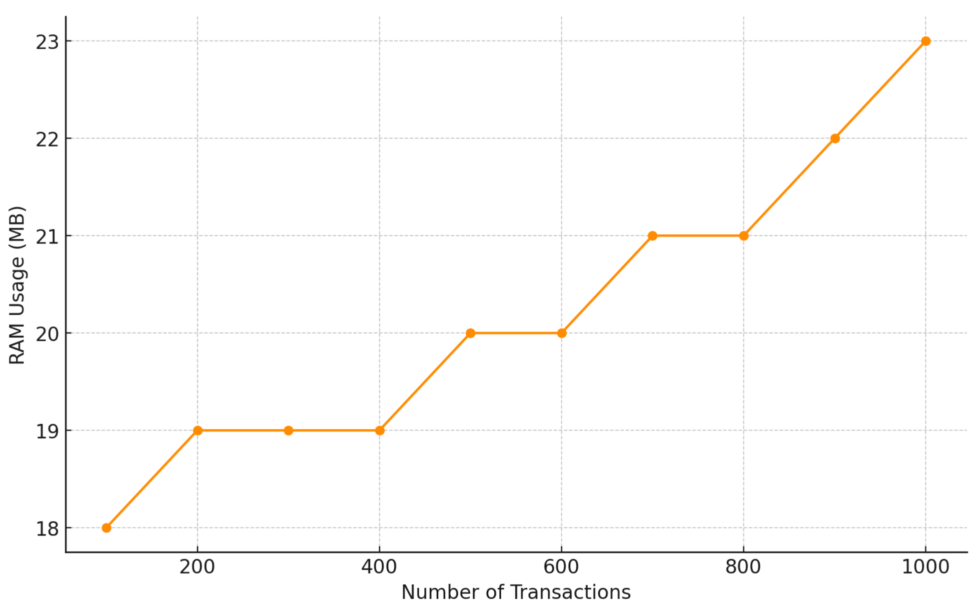
Experimental Analysis

Scalability Of Blockchains

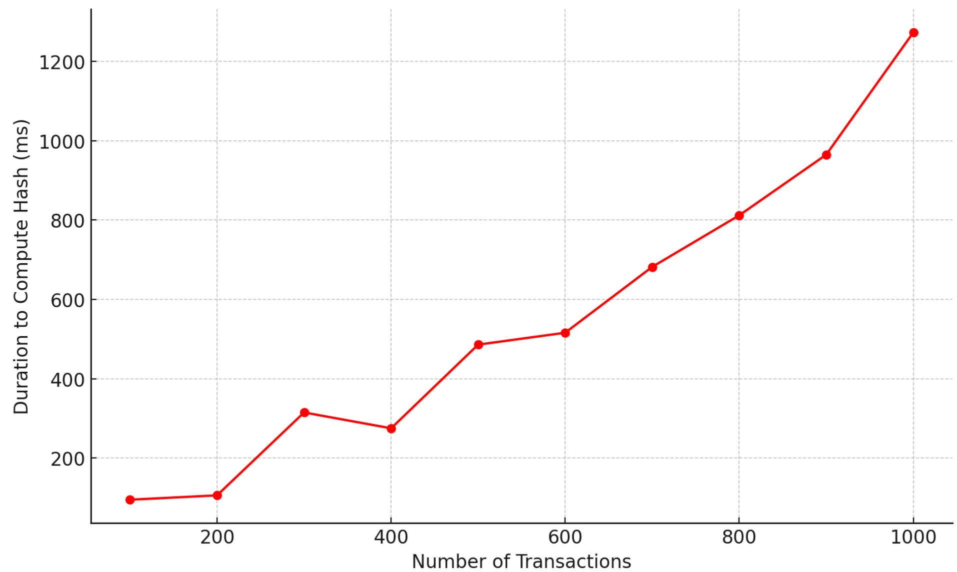
Table Showing Impact Of Transaction Volume On Blockchain Performance - Difficulty 1 (figure 7)

No. Transactions	CPU Usage (%) Rounded	RAM Usage (MB) Rounded	Duration to compute hash (MS)
100	11%	18 MB	95 MS
200	11%	19 MB	106 MS
300	13%	19 MB	315 MS
400	11%	19 MB	275 MS
500	14%	20 MB	486 MS
600	14%	20 MB	516 MS
700	14%	21 MB	682 MS
800	16%	21 MB	812 MS
900	14%	22 MB	965 MS
1000	18%	23 MB	1273 MS

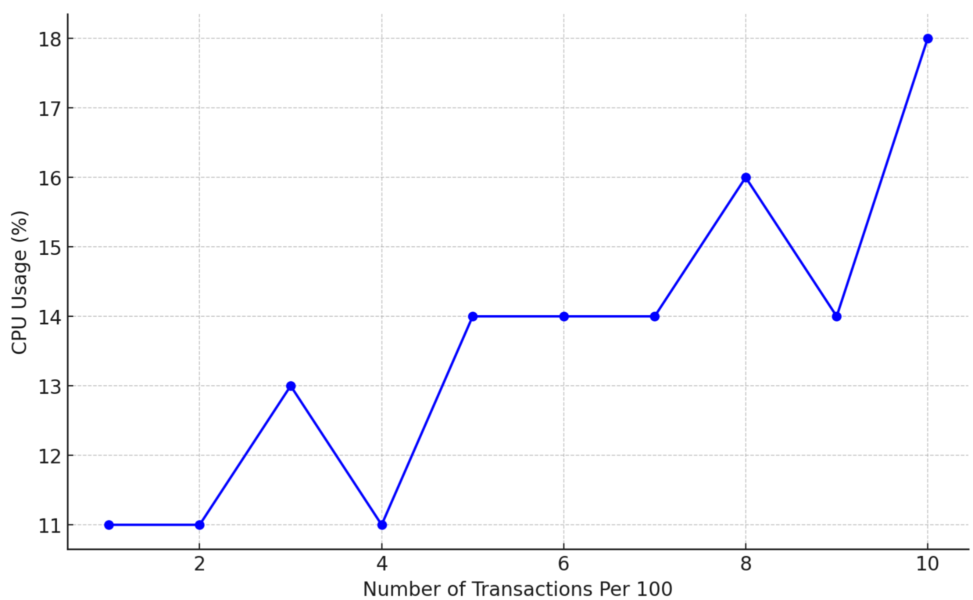
Graph showing the RAM Usage in MB vs Number of Transactions Per 100 - Difficulty 1 (figure 8)



Duration to compute hash (MS) vs Number of Transactions Per 100 - Difficulty 1 (figure 9)



CPU usage (%) vs Number of Transactions Per 100 - Difficulty 1 (figure 10)



Impact Of PoW Complexity

Table Showing Time Taken For Transactions to be Verified At PoW difficulty 1 - SHA256 (fig 11)

No. Transactions	Run 1	Run 2	Run 3	Run 4	Run 5	Average
100	95 MS	118 MS	117 MS	139 MS	153 MS	124.4 MS
200	106 MS	74 MS	127 MS	156 MS	29 MS	98.4 MS
300	315 MS	103 MS	326 MS	214 MS	421 MS	275.8 MS
400	275 MS	218 MS	123 MS	318 MS	274 MS	241.6 MS
500	486 MS	633 MS	46 MS	369 MS	133 MS	333.4 MS
600	516 MS	726 MS	492 MS	379 MS	538 MS	530.2 MS
700	682 MS	672 MS	271 MS	1377 MS	752 MS	750.8 MS
800	812 MS	117 MS	840 MS	962 MS	1419 MS	830 MS
900	965 MS	1459 MS	1139 MS	1266 MS	1588 MS	1283.4 MS
1000	1273 MS	974 MS	1398 MS	826 MS	926 MS	1049.4 MS

Graph Showing Average Time For Transactions to be Verified at PoW difficulty 1 - SHA256 (fig 12)

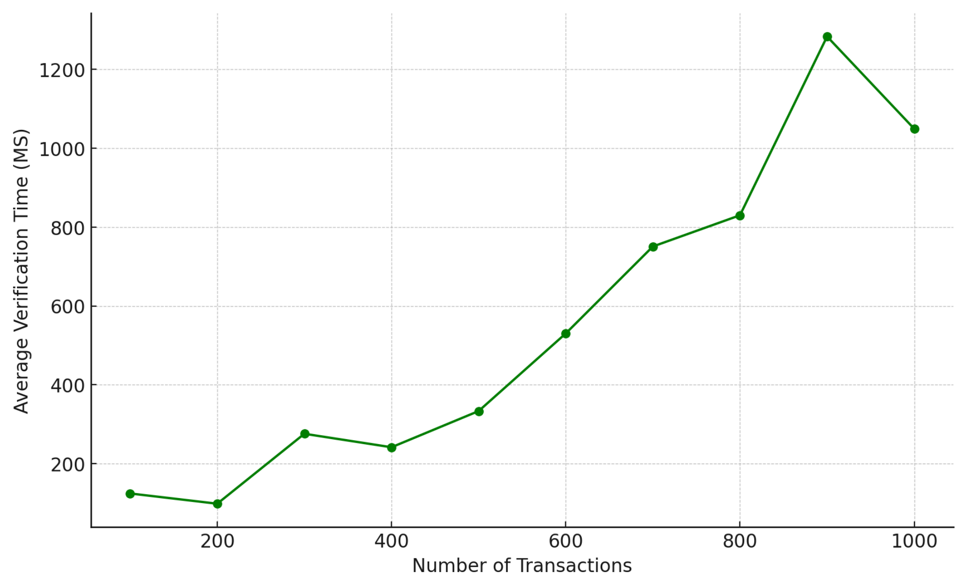


Table Showing Time Taken For Transactions to be Verified At PoW difficulty 1 - SHA512 (fig13)

No. Transactions	Run 1	Run 2	Run 3	Run 4	Run 5	Average
100	99 MS	143 MS	130 MS	64 MS	330 MS	153.2 MS
200	142 MS	465 MS	120 MS	246 MS	461 MS	286.8 MS
300	317 MS	1514 MS	704 MS	584 MS	97 MS	643.2 MS
400	294 MS	32 MS	476 MS	80 MS	186 MS	213.6 MS
500	518 MS	400 MS	481 MS	666 MS	868 MS	586.6 MS
600	769 MS	629 MS	512 MS	681 MS	1247 MS	767.6 MS
700	1010 MS	840 MS	1019 MS	927 MS	999 MS	959 MS
800	1002 MS	1258 MS	1662 MS	1384 MS	809 MS	1223 MS
900	1184 MS	946 MS	711 MS	1160 MS	1815 MS	1163.2 MS
1000	1366 MS	1174 MS	1963 MS	2370 MS	473 MS	1469.2 MS

Graph Showing Average Time For Transactions to be Verified at PoW difficulty 1 - SHA512 (fig 14)

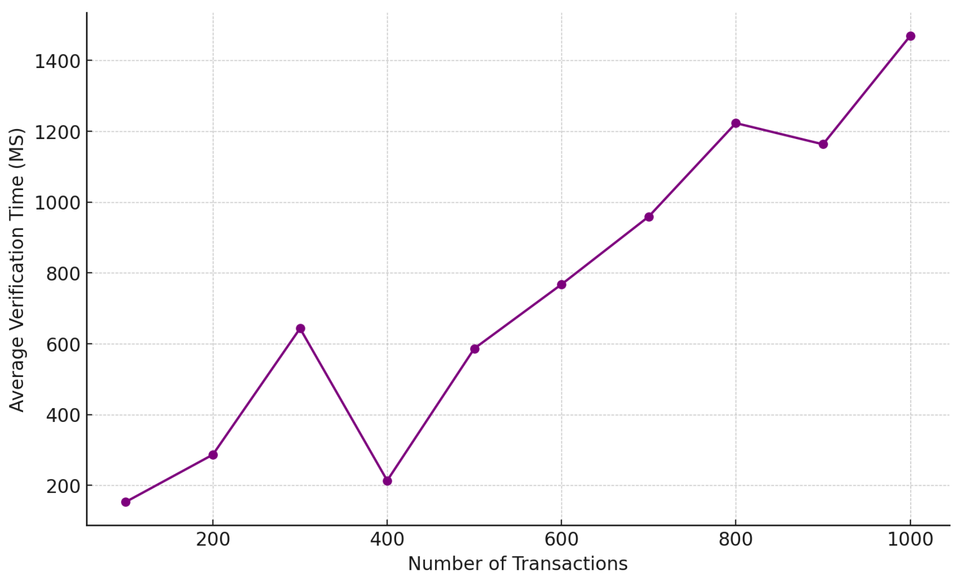


Table Showing Time Taken For Transactions to be Verified At PoW difficulty 1 - MD5 (fig 15)

No. Transactions	Run 1	Run 2	Run 3	Run 4	Run 5	Average
100	136 MS	95 MS	92 MS	91 MS	73 MS	97.4 MS
200	243 MS	179 MS	234 MS	51 MS	153 MS	172 MS
300	62 MS	85 MS	102 MS	55 MS	75 MS	75.8 MS
400	168 MS	239 MS	437 MS	363 MS	614 MS	364.2 MS
500	109 MS	143 MS	221 MS	835 MS	287 MS	319 MS
600	1298 MS	172 MS	434 MS	136 MS	1274 MS	662.8 MS
700	398 MS	364 MS	355 MS	335 MS	418 MS	374 MS
800	346 MS	437 MS	461 MS	482 MS	333 MS	411.8 MS
900	201 MS	518 MS	499 MS	349 MS	497 MS	412.8 MS
1000	525 MS	641 MS	572 MS	1382 MS	850 MS	784 MS

Graph Showing Average Time For Transactions to be Verified at PoW difficulty 1 - MD5 (fig 16)

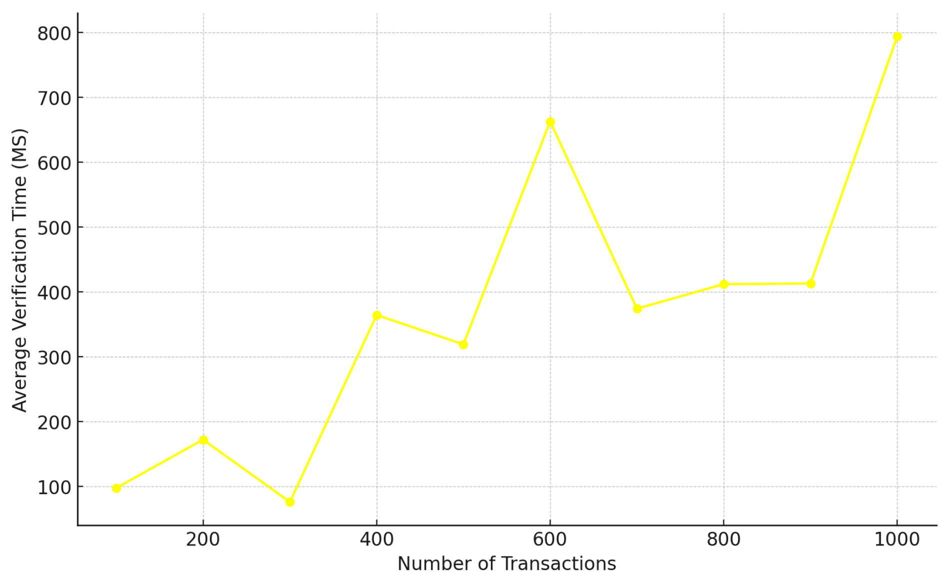


Table Showing Time Taken For Transactions to be Verified At PoW difficulty 2 - SHA256 (fig 17)

No. Transactions	Run 1	Run 2	Run 3	Run 4	Run 5	Average
100	13982 MS	25682 MS	19328 MS	29160 MS	8784 MS	19387.2 MS
200	652 MS	21770 MS	30414 MS	30601 MS	7150 MS	18117.4 MS
300	3761 MS	54923 MS	64687 MS	7561 MS	58415 MS	37869.4 MS
400	562525 MS	2271 MS	10099 MS	35267 MS	16295 MS	125291.4 MS
500	84099 MS	41034 MS	37508 MS	49910 MS	2398 MS	42989.8 MS
600	164634 MS	86829 MS	144493 MS	153844 MS	34239 MS	116807.8 MS
700	17455 MS	25126 MS	106700 MS	39392 MS	90259 MS	55786.4 MS
800	28679 MS	19474 MS	162352 MS	136299 MS	15419 MS	72444.6 MS
900	5631 MS	19046 MS	149552 MS	6804 MS	26023 MS	41411.2 MS
1000	28888 MS	113799 MS	10853 MS	46783	7012 MS	41467 MS

Graph Showing Average Time For Transactions to be Verified at PoW difficulty 2 - SHA256 (fig 18)

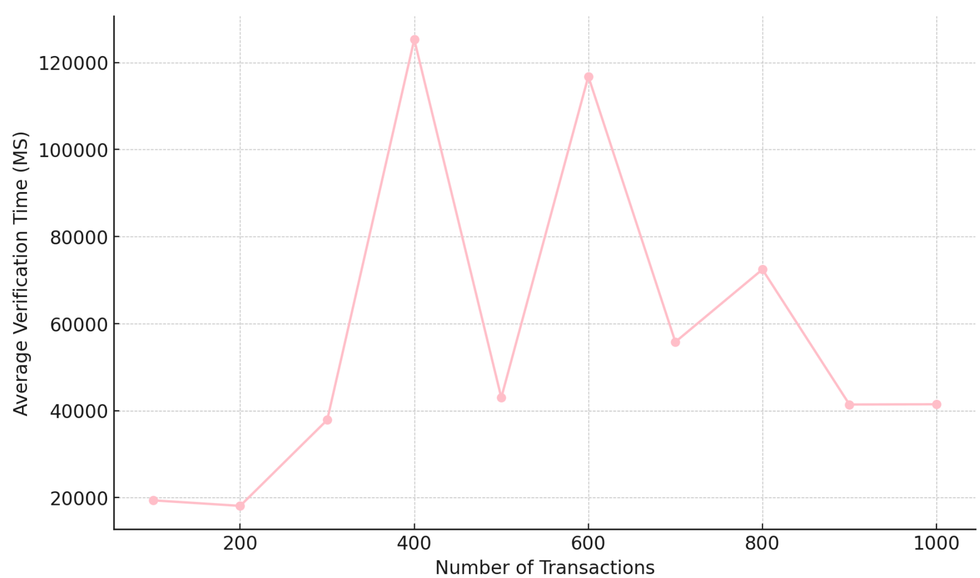


Table Showing Time Taken For Transactions to be Verified At PoW difficulty 2 - SHA512 (fig 19)

No. Transactions	Run 1	Run 2	Run 3	Run 4	Run 5	Average
100	16222 MS	50394 MS	15645 MS	37666 MS	546 MS	24094.6 MS
200	73918 MS	48366 MS	5490 MS	17626 MS	48835 MS	38847 MS
300	2698 MS	98387 MS	23797 MS	19840 MS	9991 MS	30942.6 MS
400	95095 MS	55202 MS	38130 MS	51216 MS	56234 MS	59175.4 MS
500	21352 MS	336499 MS	50710 MS	97163 MS	35671 MS	108279 MS
600	60707 MS	8938 MS	3055 MS	282715 MS	234364 MS	117955.5 MS
700	68693 MS	78308 MS	84982 MS	48593 MS	15770 MS	59269.2 MS
800	18217 MS	427918 MS	197258 MS	78758 MS	68638 MS	158157.8 MS
900	59212 MS	20402 MS	261408 MS	82768 MS	41372 MS	93032.4 MS
1000	551532 MS	115978 MS	9327 MS	321088 MS	26294 MS	204843.8 MS

Graph Showing Average Time For Transactions to be Verified at PoW difficulty 2 - SHA512 (fig 20)

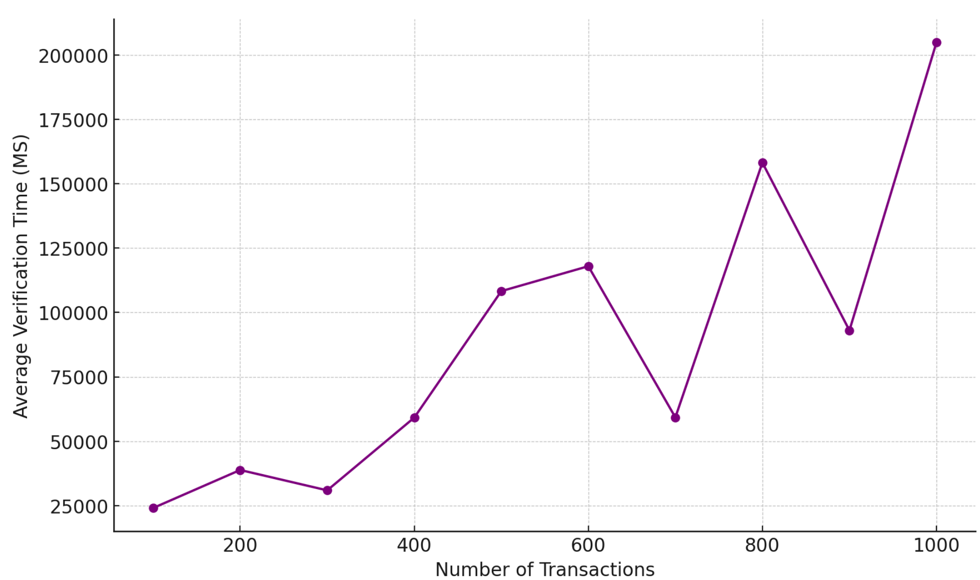
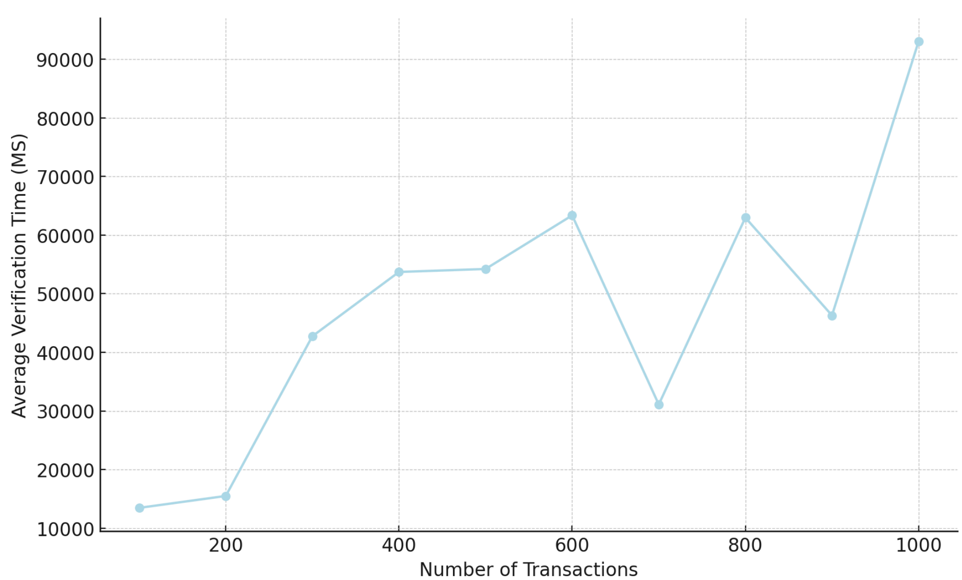


Table Showing Time Taken For Transactions to be Verified At PoW difficulty 2 - MD5 (fig 21)

No. Transactions	Run 1	Run 2	Run 3	Run 4	Run 5	Average
100	6033 MS	4177 MS	32410 MS	9391 MS	15304 MS	13463 MS
200	25554 MS	12363 MS	89 MS	24597 MS	14869 MS	15494.4 ms
300	30532 MS	13048 MS	63335 MS	105376 MS	1448 MS	42747.8 MS
400	49859 MS	120090 MS	76904 MS	3120 MS	18595 MS	53713.6 MS
500	17759 MS	94969 MS	121746 MS	2087 MS	34555 MS	54223.2 MS
600	19757 MS	130688 MS	32091 MS	88864 MS	45446 MS	63369.2 MS
700	19757 MS	60332 MS	22667 MS	48932 MS	3829 MS	31103.4 MS
800	48454 MS	96720 MS	143276 MS	996 MS	25458 MS	62980.8 MS
900	99371 MS	34764 MS	25617 MS	24759 MS	46982 MS	46298.6 MS
1000	1345 MS	17801 MS	375468 MS	36578 MS	34005 MS	93039.4 MS

Graph Showing Average Time For Transactions to be Verified at PoW difficulty 2 - MD5 (fig 22)



Security Threats In Blockchain Systems

Showing how the blockchain reacts to the modification of blocks (figure 23)

```
Is the blockchain valid? : True
Trying to modify Blockchain...
Blockchain modified
Is the blockchain valid? : False
```

Showing the MD5 hashing algorithms collision vulnerability to 4 characters (figure 24)

```
Searching for hash collision with the first 4 characters...
Original nonce: 0
Original hash: 04E41E2F43621B128A224E7D654ED9C8

Match found! New nonce: 58511
New hash: 04E45E2BF85866C1B1A9CD551BB8FF17
Time taken: 102787 ms
```

Discussion

Scalability Of Blockchains

Looking at the data that was collected during the scalability tests. They clearly show that the more amount of transactions that are in the blockchain the longer it takes to compute their hash values. The data also shows an increase per 100 transactions in RAM & CPU usage however these statistics cannot be properly validated as there were too many different variables at stake during the test, for example, other programs being open, multi-threading processes different types of CPU cores (performance, efficiency) & the impact that the PoW difficulty is at because the test was run using the PoW difficulty 1 the C# blockchain example ran very quickly meaning it was hard to collect accurate data because of how much the CPU/RAM statics will fluctuate in such a short amount of time. However, the graphs created show a solid trend that increases per transaction. The reasoning behind the number of transactions per data point (100) was that we needed to populate the blockchain with lots of transactions like in a real-world blockchain. A mere 1 to 10 transactions would need to sufficiently reflect the actual operational conditions of a blockchain. As previously observed the increase in computations like time and resource usage as transactions increase underscores a scalability challenge facing blockchains. As the transaction volume grows so does the strain on the system. In the long term, this can lead to longer transaction processing times with the increased amount of money spent on operational costs for the upkeep of the systems responsible for “mining”. Which could potentially decrease the attractiveness of high-volume applications, However, because blockchains are decentralized, the scalability tests do bring to light a unique advantage. Blockchains can be distributed across the globe with individual third-party miners taking on the computations needed and the operation costs in exchange for a mining reward. This mechanism ensures that blockchains can be scaled by incentivizing more participants to join the network as there is now a monetary value for the number of hashes “mined” This in turn allows for the blockchain to be very scalable if the incentive is greater than the cost of mining hashes. The use of SegWit also allows the blocks to increase in size while listening to the network although this technology is still in its early stages.

Impact Of PoW Complexity

Reviewing the results obtained from the PoW complexity test it is evident that as the difficulty increases the time to mine a hash significantly increases as shown in Figure 14 where the difficulty was set at 1 the longest time it took for a hash to be mined was 1400ms were as in Figure 18 were the hashing algorithm was the same and the difficulty was set at 2 the longest time it took for a hash to be mined was 120000ms this shows a significant increase in time as the complexity gets higher. Looking at the raw table data from Figures 11 - 22 from all the runs 1 - 5 it is apparent that the hashing algorithm selected has an impact on the time taken to validate blocks as shown in Figure 22 where the hashing algorithm used was MD5 comparing this to the SHA family of algorithm 256 & 512. The MD5 algorithm is much more “efficient” time-wise however this comes with some setbacks as MD5 is vulnerable to attacks where the SHA algorithms are not as they are more complex. This shows in the data gathered as SHA algorithms are much slower at confirming blocks. As predicted SHA512 takes a much longer time to validate than SHA256 you can see this in Figures 17 and 19 this is because SHA512 has a much larger block size which will require more computational resources as explained in the naming scheme 512 is much bigger than 256. This will translate into an increased validation time for each block that is being processed. The experiment shows that increasing the PoW difficulty by even a factor of 1 significantly increases the time to mine a block this is because if the difficulty is set at 2 the miner will have to find a hash that starts with two “0”s instead of just 1 “0”. This could be a challenge when trying to scale the blockchain infrastructure however because blockchains are decentralized multiple different parties can collaborate to mine a block. If the difficulty gets too hard for a traditional computer to compute the hash within a viable time frame. Pool mining can be introduced this is where multiple computers on the blockchain network will work together to find a block and the reward is shared among them depending on their hashing power. The same issue that was faced within the scalability test happened within this test as well with the fluctuation in data so it was decided to complete 5 runs of each hashing algorithm and difficulty and get an average this resulted in more consistent data however there are still outliers within the data that could not be addressed. The tests clearly show that the harder the difficulty the higher the time complexity is.

Security Threats In Blockchain Systems

Examining the data gathered from the analysis of security threats in blockchain systems it is apparent that blockchain systems are very secure as shown in Figure 23 any type of illegal modification to the blockchain results in the chain becoming invalid. This shows us that malicious actors cannot easily compromise the integrity of the blockchain as any unauthorized modifications to the chain are detected by the isvalid method within the blockchain example by recalculating the hash based on the current state of the block and comparing sed hash to the stored hash value if these 2 hashes do not match then the block data has been modified which will invalidate the blockchain. This method also checks if the current block hash properly matches the previous block hash. This feature makes sure that not only individual blocks are unaltered but the whole chain remains intact. However, there are a few vulnerabilities that blockchains are susceptible to as shown in Figure 24 the blockchain is susceptible to a hash collision attack if and only if the hashing algorithm used within the chain is MD5. This algorithm has a known vulnerability where 2 different nonces can have the same hash for demonstration purposes the test that was created will find a hash that has the first 4 characters the same and then print out the nonce of that hash. This demonstration shows the vulnerability of the MD5 hashing algorithm within Figure 6 if we change the character variable to the size of the hash it will eventually find a nonce that has the same hash as the original nonce this is an issue because it compromises the unique identity that hashes are supposed to provide to the blockchain allowing for an attacker to potentially insert a fraudulent block that will appear legitimate to the system. This is why most blockchains do not use MD5 and use other hashing algorithms like sha256 which is significantly more resilient. Another vulnerability that blockchains have is the 51% attack which is when an attacker gains control of the majority of the network mining power which allows them to manipulate transactions and double spend coins. However, this attack is very unlikely due to the blockchain's decentralized nature as multiple different entities are mining making this attack very computationally expensive, especially on large networks. Because blockchain technology is fairly new it is considered to be very secure however blockchains are not immune to future threats like quantum computers which pose a significant risk to chain technologies because of their potential to break cryptographic algorithms that were originally considered secure which would allow a malicious actor to double spend and reverse the transaction once it has already been verified. The implementation of Quantum hashing algorithms to the blockchain will be crucial in ensuring security within the blockchains. But for now, blockchains are very secure.

Conclusion

In conclusion, the overall methods that were used to run tests on blockchain scalability, PoW complexity & Security threats were a success as they provided a variety of data on blockchains from how transactions impact the computational power needed for validation to how the PoW algorithm and various hashing algorithms take effect on the time to validate a block, and finally to the security threats that blockchain technology face. Some key findings that were discovered were the impact that transaction volume has on computational power. Shows a direct correlation between the number of transactions and the increase in CPU/RAM usage also in how long it takes to validate a block. Another finding was how unsecure the MD5 hashing algorithm is when comparing it to other algorithms like sha256. There could be some improvements however to how the tests were executed, mainly in the test environment making sure that nothing else is running on the computer making sure that the program has access to the full potential of the CPU using multiple threats. For future work, Investigating the resilience that blockchains have against DDOS attacks (Distributed Denial Of Service) would be a good area to study. Prototyping some code that could create false or fake transactions that could flood the network making it much slower as miners will have to invalidate numerous transactions before getting to “real” transactions. It would be interesting to see how the blockchain reacts to this type of attack. Overall I find that the empirical analysis that the paper gave of blockchain technology was a success.

References

Cheng, J., Xie, L., Tang, X., Xiong, N., & Liu, B. (2020). A survey of security threats and defense on Blockchain. *Multimedia Tools and Applications*, *80*(20), 30623–30652. **https://doi.org/10.1007/s11042-020-09368-6**

Chin, Z. H., Yap, T. T. V., & Tan, I. K. T. (2022). Genetic-Algorithm-Inspired Difficulty Adjustment for Proof-of-Work Blockchains. *Symmetry*, *14*(3), 609. **https://doi.org/10.3390/sym14030609**

Khan, D., Jung, L. T., & Hashmani, M. A. (2021). Systematic Literature Review of Challenges in Blockchain Scalability. *Applied Sciences*, *11*(20), 9372. **https://doi.org/10.3390/app11209372**

Sapna, D. P. (2021). Analysis of Blockchain Vulnerabilities & Attacks on Wallet. *2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*. **https://doi.org/10.1109/icac3n53548.2021.9725403**

Siddiqui, S. T., Ahmad, R., Shuaib, M., & Alam, S. (2020). Blockchain Security Threats, Attacks, and Countermeasures. *Advances in Intelligent Systems and Computing*, 51–62. **https://doi.org/10.1007/978-981-15-1518-7_5**

Sohrabi, N., & Tari, Z. (2020, April 1). *On The Scalability of Blockchain Systems*. IEEE Xplore. **https://doi.org/10.1109/IC2E48712.2020.00020**

Thanalakshmi, P., Rishikhesh, A., Marceline, J. M., Joshi, G. P., & Cho, W. (2023). A Quantum-Resistant Blockchain System: A Comparative Analysis. *Mathematics*, *11*(18), 3947–3947. **https://doi.org/10.3390/math11183947**

Xie, J., Yu, F. R., Huang, T., Xie, R., Liu, J., & Liu, Y. (2019). A Survey on the Scalability of Blockchain Systems. *IEEE Network*, *33*(5), 166–173. **https://doi.org/10.1109/mnet.001.1800290**

Zaghloul, E., Li, T., Mutka, M. W., & Ren, J. (2020). Bitcoin and Blockchain: Security and Privacy. *IEEE Internet of Things Journal*, *7*(10), 1–1. **https://doi.org/10.1109/jiot.2020.3004273**

Zhou, Q., Huang, H., Zheng, Z., & Bian, J. (2020). Solutions to Scalability of Blockchain: A Survey. *IEEE Access*, *8*(1), 16440–16455. **https://doi.org/10.1109/access.2020.2967218**