## Q.1 Blockchain implementation challenges

### Lack of adoption

Blockchains function more effectively and efficiently when used across a wide network of users. A blockchain ecosystem, for example, would not only need users to join the network but also its suppliers. APQC, on the other hand, has discovered that just 29% of organizations are experimenting with or fully utilizing blockchain. Without broad adoption, blockchains will remain ineffective and scalable.

However, there are some indications that blockchain adoption will continue to spread. Businesses are gradually coming together and forming collaborative blockchain working groups to tackle similar problems and provide solutions that may benefit everyone without giving away proprietary information.

For example, the Blockchain for Clinical Supply Chain Industry Working Group was formed in 2016 by several large pharmaceutical companies and Deloitte to use blockchain technology. The group developed KitChain with blockchain developer LedgerDomain and worked with them on an app called KitChain. The program also offers other advantages, including keeping track of packaged pharmaceuticals throughout transit. This helps ensure the supply chain's security and minimize reliance on paper logs and secure medical trial data.

### The rising cost of blockchain implementation

It's all about early financial outlays. For some companies, implementation expenditures may be prohibitive. Even if most current solutions are free, a significant investment is required when hiring skilled software engineers who specialize in blockchain development, licensing fees in case of switching to a chargeable software version, overall administration, and more. It is one of the most important blockchain implementation challenges.

If businesses aren't prepared to devote a significant amount of money, perhaps it's better to put off the blockchain introduction.

### Scalability

The main difficulty with its implementation is scalability. Though transaction networks can handle hundreds of transactions per second without failure, when it comes to Bitcoin (roughly 3-7 transactions per second) and Ethereum (about 15-20 transactions), processing the transactions slows down considerably, making blockchain unviable for large-scale apps.

Lightning Network and Plasma for Ethereum are scaling technologies that allow fast, low-fee transactions. For widespread adoption to take place, blockchain must improve its speed.

The inability to serve many users is one of the most significant drawbacks of blockchain technology and, by extension, enterprise blockchain technology.

Organizations that can scale their enterprise blockchain platforms successfully will profit successfully due to the rising demand for enterprise blockchain and associated apps.

**Security and privacy challenges**

What are some of the most pressing questions you should consider? What about the numerous security and privacy concerns? While cryptocurrencies provide pseudonymity, many potential blockchain applications need smart transactions and contracts to be indisputably linked to real identities, raising significant privacy and data security issues.

Many businesses today operate under the constraints of legislation. Their customers put their trust in them with important information. However, if all of this data is kept on a public ledger, it will not be truly private. Private or consortia blockchain technology may be utilized here. You would only have enough access, and your confidential information would remain secure.

Another vital element is security. Only a few scenarios have effective protocols that can handle this, though. Hackers may still break into apps, systems, and businesses based on blockchains even though they are more secure than conventional computer systems.

The answer is not simply to have the government safeguard our privacy. Self-sovereign identities on blockchain will allow us to collect and manage our data. While there’s a lot of effort to create new privacy protocols, such as proof of zero-knowledge, we’re still a long way from a new identity structure. You can go to the blockchain and AI secure data processing article to understand how blockchain and AI can be leveraged for secure data storage. It is one of the most important blockchain implementation challenges.

**Regulations**

The lack of regulation will be the next area where you may face difficulties. Scams and market manipulation that might trigger a global economic collapse are not out of the question. As a result, Bitcoin has been getting lots of negative attention from all around the world.

Some countries have outright banned bitcoin, while others attempt to regulate blockchain networks with little success.

**Criminal activities**

The absence of stringent legislation and the fact that blockchain is still a developing technology have fueled the rise of fraudulent projects and other bad actors seeking to profit from inexperienced investors. There have also been several high-profile cryptocurrency exchange thefts, including Mt. Gox's infamous bitcoin theft in 2014, nearly destroying the entire cryptocurrency industry.

**Energy consumption**

Another worry is that Proof of Work, the most widely used consensus algorithm, is energy-intensive. This restricts entry for regular people into PoW networks, encourages the formation of big mining pools, and prevents decentralization by pushing individuals to join large mining pools, and it also raises environmental concerns.

**51% attacks**

The architecture of blockchain technologies is distinct. Some are more secure than others. The decentralized blockchains, for example, are more vulnerable to 51% attacks than the centralized ones. This has introduced a few issues for crypto investors who want to store their assets on decentralized chains.

51% attacks, in which hackers acquire more than half of the network's computational power, are an issue that has plagued many blockchain systems. They exploit an inherent loophole in decentralized systems, allowing users to control a chain by controlling over 51% of the processing power. This typically happens on networks that use the proof-of-work (PoW) framework.

**Low workforce availability**

Over the last year, the nonfungible token (NFT) and DeFi industries have seen a tremendous rise in nonfungible tokens and projects, leading to labor market issues. According to recent data, demand for blockchain talent has increased by over 300% among both established businesses and startups as they look for top-tier personnel.

The scarcity of experienced developers in the blockchain industry has worsened due to high competition between firms offering highly competitive compensation packages to attract and retain their employees. As a result, some businesses in the crypto sector are paying more than $1 million per year to workers in specific job categories. Businesses are calling

coders to fill the blockchain talent gap. One of the blockchain implementation challenges is getting the closest to being solved day by day.

**Interoperability**

One of the most important issues that must be addressed is interoperability, as this is one of the primary reasons businesses are yet hesitant to embrace blockchain technology. Most blockchains are maintained in isolation and do not communicate with other peer networks since they cannot transmit and receive data from a different blockchain-based system.

They grant access to all of the data in a wallet once generated. It jeopardizes both confidential information and money if stolen. Wallet access is lost for good if lost or destroyed. It is one of the most dangerous blockchain implementation challenges.

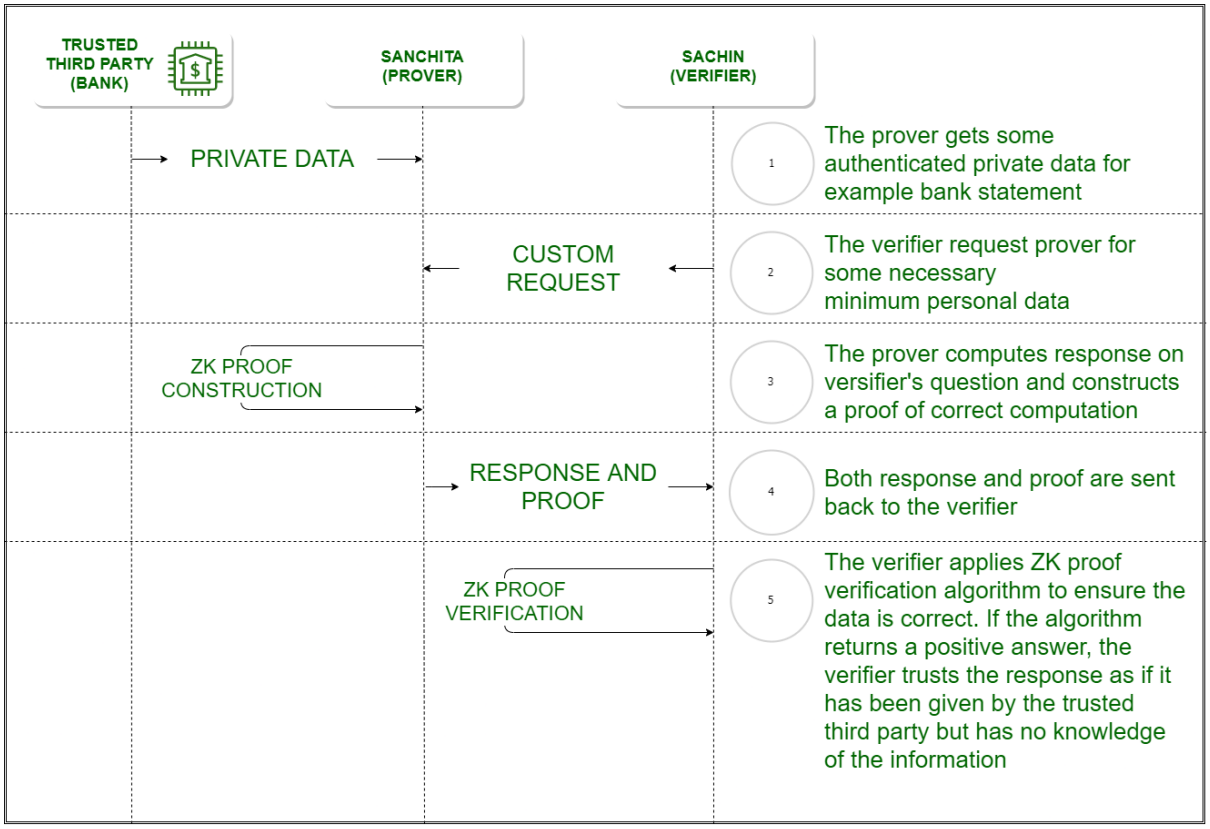
## Q.2 Explain about Zero Knowledge proofs and protocols in Blockchain

**Zero Knowledge Proof**

Zero Knowledge Proof (ZKP) is an encryption scheme originally proposed by MIT researchers Shafi Goldwasser, Silvio Micali and Charles Rackoff in the 1980s.

Zero-knowledge protocols are probabilistic assessments, which means they don't prove something with as much certainty as simply revealing the entire information would. They provide unlinkable information that can together show the validity of the assertion is probable.

Currently, a website takes the user password as an input and then compares its hash to the stored hash. Similarly a bank requires your credit score to provide you the loan leaving your privacy and information leak risk at the mercy of the host servers. If ZKP can be utilized, the client's password is unknown the to verifier and the login can still be authenticated. Before ZKP, we always questioned the legitimacy of the prover or the soundness of the proof system, but ZKP questions the morality of the verifier. What if the verifier tries to leak the information?




Example-1: A Colour-blind friend and Two balls :
There are two friends Sachin and Sanchita, out of whom Sanchita is colour blind. Sachin has two balls and he needs to prove that both the balls our of different colour. Sanchita switches the balls randomly behind her back and shows it to Sachin who has to tell if the balls are switched or not. If the balls are of the same colour and Sachin had given false information, the probability of him answering correctly is 50%. When the activity is repeated several times, the probability of Sachin giving the correct answer with the false information is significantly low. Here Sachin is the “prover” and Sanchita is the “verifier”. Colour is the absolute information or the algorithm to be executed, and it is proved of its soundness without revealing the information that is the colour to the verifier.

Example-2: Finding Waldo :
Finding Waldo is a game where you have to find a person called Waldo from a snapshot of a huge crowd taken from above. Sachin has an algorithm to find Waldo but he doesn’t want to reveal it to Sanchita. Sanchita wants to buy the algorithm but would need to check if the algorithm is working. Sachin cuts a small hole on a cardboard and places over Waldo. Sachin is the “prover” and Sanchita is the “verifier”. The algorithm is proved with zero knowledge about it.

Properties of Zero Knowledge Proof :

Zero-Knowledge –
If the statement is true, the verifier will not know that the statement or was. Here statement can be an absolute value or an algorithm.

Completeness –
If the statement is true then an honest verifier can be convinced eventually.

Soundness –
If the prover is dishonest, they can’t convince the verifier of the soundness of the proof.

Types of Zero Knowledge Proof :

Interactive Zero Knowledge Proof –
It requires the verifier to constantly ask a series of questions about the “knowledge” the prover possess. The above example of finding Waldo is interactive since the “prover” did a series of actions to prove the about the soundness of the knowledge to the verifier.

Non-Interactive Zero Knowledge Proof –
For “interactive” solution to work, both the verifier and the prover needed to be online at the same time making it difficult to scale up on the real world application. Non-interactive Zero-Knowledge Proof do not require an interactive process, avoiding the possibility of collusion. It requires picking a hash function to randomly pick the challenge by the verifier. In 1986, Fiat and Shamir invented the Fiat-Shamir heuristic and successfully changed the interactive zero-knowledge proof to non-interactive zero knowledge proof.

## Q.3 SNARKs in blockchain

Even though cryptocurrency transactions aren't tied to anyone's identity, they're normally traceable because they're publicly recorded on a blockchain. If you want to avoid this, you'll need a specific type of cryptocurrency called a privacy coin. One of the methods privacy coins use to keep transactions anonymous is with a technology called zk-SNARKs.

Blockchain crypto technology symbolizing chain of block in digital ledger for cryptocurrency like bitcoin or ethereum. Data security and encryption. Connected nodes, fintech. Abstract background.
For those who are considering privacy coins for personal use or as a cryptocurrency investment, it's important to understand the technology behind them. While zk-SNARKs is somewhat complicated, we'll cover exactly what it does in this guide.

**What is zk-SNARK?**
A zk-SNARK is a zero-knowledge proof protocol where one can prove they possess certain information without revealing it and without any interaction between the parties proving and verifying the information.

The term "zk-SNARK" is an acronym that stands for "Zero-Knowledge Succinct Non-Interactive Argument of Knowledge." Each part of the name refers to a characteristic of zk-SNARKs, so it helps to look at each component separately:

**Zero-Knowledge:** The prover can show the verifier that they have a piece of information without providing the information itself.
Succinct: The proof can be verified within a few milliseconds since the proof length is only a few hundred bytes at most.
Non-Interactive: The proof consists of a single message from the prover to the verifier.
Argument: Argument is the term used for these proofs because they don't quite fit the traditional definition of proofs, but they effectively serve the same purpose.
Knowledge: Knowledge refers to the information possessed by the prover.
In cryptocurrency, zk-SNARKs are a way for transactions to be private and fully encrypted on the blockchain while still being validated using the network's consensus rules. zk-SNARKs can show that the sender has the amount of funds they want to transfer without making that information public.

**How zk-SNARKs work**
With most types of cryptocurrency, a transaction is validated by the network checking that certain conditions have been met. Specifically, the conditions are that the sender has the funds available and that they've provided the correct private key to show that the funds are theirs.

zk-SNARKs allow the sender of a transaction to prove all that without revealing any of the addresses or amounts involved. To offer this, the blockchain network encodes some of its consensus rules in zk-SNARKs.

During the transaction process, zk-SNARKs turn the information that needs to be proved into equations. These equations can be evaluated and solved without disclosing the information itself.

**Pros and cons of zk-SNARKs**
The primary benefits of zk-SNARKs are the privacy and efficiency they offer. They shield sensitive information, it takes just milliseconds to verify them, and they don't require extended interaction between the parties involved. All that's needed is one message from the prover to the verifier.

Although zk-SNARKs don't have any glaring drawbacks, if someone has the private key used to set up the protocol, they'd be able to create false proofs and counterfeit funds. Privacy coins that use zk-SNARKs must take steps to ensure no single party has access to that private key.

**zk-SNARK vs. zk-Rollup**
zk-SNARK is far from the only type of zero-knowledge proof used by cryptocurrencies. Another common technology is the zk-Rollup, a scaling solution to help Ethereum (CRYPTO:ETH) process transactions more efficiently.

A rollup bundles a large group of transactions and validates them off-chain, meaning all the computations are handled off the main Ethereum blockchain. They're then rolled up into a single transaction that's sent to Ethereum's execution layer.

There are different types of rollups available. For example, Optimistic rollups assume transactions are valid until they're proven false. zk-Rollups, on the other hand, instantly verify transactions and generate cryptographic validity proofs. The proofs can be generated using zk-SNARKs or another type of zero-knowledge proof technology, zk-STARKs.

**zk-SNARK examples**
The most well-known example of zk-SNARKs is their use in shielding cryptocurrency transactions. Zcash is the first widespread application of zk-SNARKs. This privacy coin allows users to choose between private and transparent addresses. When a user chooses a private address, zk-SNARKs shield the transaction data.

Other blockchain projects are also using zk-SNARKs. Ethereum started working on integrating Zcash and zk-SNARKs in 2017. That same year, Zcash partnered with JP Morgan Chase (NYSE:JPM) on building a blockchain-based payment system with zk-STARKs.

While zk-SNARKs are currently used to privatize financial transactions, that's not their only application. In the future, we could see them used to protect people's data during online activity.

## Q.4) Blockchain – Elliptic Curve Cryptography

Cryptography is the study of techniques for secure communication in the presence of adversarial behavior. Encryption uses an algorithm to encrypt data and a secret key to decrypt it. There are 2 types of encryption:

**Symmetric-key Encryption (secret key encryption):** Symmetric-key algorithms are cryptographic algorithms that employ the same cryptographic keys both for plaintext encryption and ciphertext decoding. The keys could be identical, or there could be a simple transition between them.
**Asymmetric-key encryption (public key encryption):** Asymmetric-key algorithms encrypt and decrypt a message using a pair of related keys (one public key and one private key) and safeguard it from unauthorized access or usage.

The following topics of Elliptic Curve Cryptography will be discussed here:

Introduction to Elliptic Curve Cryptography
History of Elliptic Curve Cryptography
Components of Elliptic Curve Cryptography
Elliptic Curve Cryptography Algorithms
Application of Elliptic Curve Cryptography
ECC vs RSA
Elliptic Curve Diffie-Hellman Protocol Implementation
Types of Security Attacks
Benefits of Elliptic Curve Cryptography
Limitations of Elliptic Curve Cryptography

**Conclusion**

Introduction to Elliptic Curve Cryptography

ECC, as the name implies, is an asymmetric encryption algorithm that employs the algebraic architecture of elliptic curves with finite fields.

**Elliptic Curve Cryptography (ECC)** is an encryption technology comparable to RSA that enables public-key encryption.

While RSA's security is dependent on huge prime numbers, ECC leverages the mathematical theory of elliptic curves to achieve the same level of security with considerably smaller keys.

Victor Miller and Neal Koblitz separately proposed elliptic curve ciphers in the mid-1980s. On a high level, they are analogs of actual public cryptosystems in which modular arithmetic is substituted by elliptic curve operations.

History of Elliptic Curve Cryptography

Neal Koblitz and Victor S. Miller independently proposed the use of elliptic curves in encryption in 1985.

Elliptic curve cryptography algorithms entered wide use from 2004 to 2005.

In the mid-1980s, researchers found that examining elliptic curves could lead to the discovery of new sources of difficult problems. Elliptic Curve Cryptography (ECC) introduced a new degree of security to public key cryptosystems, that provide combined encryption and digital signature services.

The security of elliptic curve cryptosystems, like that of all public-key cryptosystems, is based on tough mathematical issues at the core. Given two elliptic curve points G and Y, where Y = kG.

The term "elliptic curve" is derived from the ellipse. Elliptic curves were discovered in the form of the Diophantine equation for c, after the 17th century. Furthermore, while calculating the surface of the ellipse is simple, calculating the circumference of the ellipse is difficult. The equation can be simplified to an integral:

Components of Elliptic Curve Cryptography

**Below are the components of elliptic curve cryptography:**

**1. ECC keys:**

Private key: ECC cryptography's private key creation is as simple as safely producing a random integer in a specific range, making it highly quick. Any integer in the field represents a valid ECC private key.

Public keys: Public keys within ECC are EC points, which are pairs of integer coordinates x, and y that lie on a curve. Because of its unique features, EC points can be compressed to a single coordinate + 1 bit (odd or even). As a result, the compressed public key corresponds to a 256-bit ECC.

**2. Generator Point:**

ECC cryptosystems establish a special pre-defined EC point called generator point G (base point) for elliptic curves over finite fields, which can generate any other position in its subgroup over the elliptic curve by multiplying G from some integer in the range [0…r].

The number r is referred to as the "ordering" of the cyclic subgroup.

Elliptic curve subgroups typically contain numerous generator points, but cryptologists carefully select one of them to generate the entire group (or subgroup), and is excellent for performance optimizations in calculations. This is the “G” generator.
Elliptic Curve Cryptography Algorithms
Based on the arithmetic of elliptic curves over finite fields, Elliptic-Curve Cryptography (ECC) provides numerous sets of algorithms:

**Digital signature algorithms:**

Elliptic Curve Digital Signature Algorithm. (ECDSA): ECDSA, or Elliptic Curve Digital Signature Algorithm, is a more highly complicated public-key cryptography encryption algorithm. Elliptic curve cryptography is a type of public key cryptography that uses the algebraic structure of elliptic curves with finite fields as its foundation. Elliptic curve cryptography is primarily used to generate pseudo-random numbers, digital signatures, and other data.
Edwards-curve Digital Signature Algorithm (EdDSA): The Edwards-curve Digital Signature Algorithm (EdDSA) was proposed as a replacement for the Elliptic Curve Digital Signature Algorithm for performing fast public-key digital signatures (ECDSA). Its primary benefits for embedded devices are higher performance and simple, secure implementations. During a signature, no branch or lookup operations based on the secret values are performed. Many side-channel attacks are foiled by these properties.
Encryption algorithms:

Elliptic Curve Integrated Encryption Scheme (ECIES): ECIES is a public-key authenticated encryption scheme that uses a KDF (key-derivation function) to generate a separate Medium Access Control key and symmetric encryption key from the ECDH shared secret. Because the ECIES algorithm incorporates a symmetric cipher, it can encrypt any amount of data. In practice, ECIES is used by standards such as Intelligent Transportation Systems.
EC-based ElGamal Elliptic Curve Cryptography: ElGamal Elliptic Curve Cryptography is the public key cryptography equivalent of ElGamal encryption schemes that employ the Elliptic Curve Discrete Logarithm Problem. ElGamal is an asymmetric encryption algorithm that is used to send messages securely over long distances. Unfortunately, if the encrypted message is short enough, the algorithm is vulnerable to a Meet in the Middle attack.
Key Agreement algorithm:

Elliptic-curve Diffie–Hellman (ECDH): Elliptic-curve Diffie-Hellman (ECDH) is a key agreement protocol that enables two parties to establish a shared secret over an insecure channel, each with an elliptic-curve public-private key pair. This shared secret can be used directly as a key or to generate another key. Following that, the key, or the derived key, can be used to encrypt subsequent communications with a symmetric-key cipher.
Fully Hashed Menezes-Qu-Vanstone(FHMQV): Fully Hashed Menezes-Qu-Vanstone is an authenticated key agreement protocol based on the Diffie-Hellman scheme. MQV, like other authenticated Diffie-Hellman schemes, protects against an active attacker. The protocol can be adapted to work in any finite group, most notably elliptic curve groups, in which it is recognized as elliptic curve MQV (ECMQV).
Application of Elliptic Curve Cryptography

**Diffie-Hellman:** The basic public-key cryptosystem suggested for secret key sharing is the

Diffie-Hellman protocol. If A (Alice) and B (Bob) initially agree on a given curve, field size, and mathematical type. They then distribute the secret key in the following manner. We can see that all we need to build the Diffie-Hellman protocol is scalar multiplication.

Elliptic Curve Digital Signature Algorithm (ECDSA): ECC is one of the most widely utilized digital signature implementation approaches in cryptocurrencies. In order to sign transactions, both Bitcoin and Ethereum use the field inverse multiplication, but also arithmetic multiplication, inverse function, and modular operation.

Online application: Moreover, ECC is not limited to cryptocurrencies. It is an encryption standard that will be utilized by most online apps in the future due to its reduced key size and efficiency. Most commonly used in cryptocurrencies such as Bitcoin and Ethereum, along with single-way encryption of emails, data, and software.

Blockchain application: The cryptocurrency Bitcoin employs elliptic curve cryptography. Ethereum 2.0 makes heavy use of elliptic curve pairs with BLS signatures, as stated in the IETF proposed BLS specification, to cryptographically ensure that a specific Eth2 validator has really verified a specific transaction.

## Q.5 Zcash - attacks on Blockchains

Zcash, a privacy-focused proof-of-work blockchain, has been undergoing a spam attack.

Analysts say the blockchain has processed a large volume of transactions that aim to disrupt the network. To do this, the attacker misused Zcash's "shielded transactions" meant for privacy.

Here, the attacker has been adding hundreds of output values within shielded transactions that are proving to be very data-intensive. As a result, the blockchain size has grown dramatically, going from 31 GB in mid-June to more than 100 GB currently, according to data from Blockchair.

The situation is placing high demands on the network. While Zcash has not had any downtime, the blockchain's rapidly growing size has led to troubles for nodes' ability to sync with the network. On-chain records indicate that while the situation has persisted for months, it has been brought to public attention only recently.

"At this point, there only seems to be two problems with the spam: it's bloating the chain size, and it's making it harder for wallets to sync," Sean Bowe, an engineer at Zcash's core development firm Electric Coin Company said.

Others pointed out that Zcash doesn't have spam prevention systems in place. A transactor pays a small fee of 0.0001 ZEC (worth a few cents) for shielded transactions with hundreds of output values, Zcash's block explorer shows.

"It's sad to see. There was always a risk of DoS given Zcash's deliberate lack of a fee market," security researcher Ian Miers, who has previously worked at Zcash, noted in a tweet Wednesday. "The proofs are much larger and slower to verify, making the attack worse."

THE DAILY
Stay up to date on the most influential events and analysis happening across the digital asset

ecosystem.
Enter Email
Also receive our FREE weekly Data & Insights Newsletter
By signing-up you agree to our Terms of Service and Privacy Policy

The motive for this attack has not been determined, but industry commentators have offered varying theories.

Nick Bax, head of research at Convex Labs, speculated that someone is trying to profit off harming the network. He also floated a possible theory that the spam attack is an attempt to "make it harder for people to run nodes" as this can potentially allow them to surveil users' activity on the blockchain.

What are shielded transactions?
Across blockchains that use the UTXO model like Zcash, transactions are validated by linking the sender address, receiver address, and input and output values. But these values can be easily traced on a public blockchain. To enforce transactional privacy, Zcash uses cryptographic proofs called zk-SNARKs to obfuscate the input-output values of transactions.

Zcash transactions that make use of this technique are called shielded transactions and are meant for legitimate usage. However, the lack of a fee market can unintentionally let malicious actors spam the network if they use a large number of output variables. On this point, Solana Labs CEO Anatoly suggested that the project should increase the fees by 100 times for each output aiming to reduce spam.

In August, Zcash researchers proposed a new fee mechanism that aimed to incorporate fees based on transaction size. In such a scenario, fee rates would keep increasing when the network is under extreme usage.