Zerocash differs from Bitcoin in that it does not disclose full transparency on the ledger. Bitcoin broadcasts information to the blockchain with each transaction. Type of information is time, address, the value of transaction, recipient, etc. This information might be used in a malicious way and makes it impossible for a fully private payment system. The discussion of whether this is something we want or not is another question.

Zcash is the first widespread application of zk-SNARKs, a novel form of zero-knowledge cryptography. The strong privacy guarantee of Zcash is derived from the fact that shielded transactions in Zcash can be fully encrypted on the blockchain, yet still, be verified as valid under the network's consensus rules by using zk-SNARK proofs. As a spender (prover), you might decide whether you want your transaction to be transparent or not.

The acronym zk-SNARK stands for "Zero-Knowledge Succinct Non-Interactive Argument of Knowledge," and refers to a proof construction where one can prove possession of certain information, e.g. a secret key, without revealing that information, and without any interaction between the prover and verifier.

On a technical level, I struggle to fully understand the mechanics, but the sender proves that they have the private spending keys of the input notes, giving them the authority to spend. The private spending keys of the input notes are cryptographically linked to a signature over the whole transaction, in such a way that the transaction cannot be modified by a party who did not know these private keys.

Compared to Bitcoin which tracks unspent transaction outputs (UTXOs) to determine what transactions are spendable. In Zcash, the shielded equivalent of a UTXO is called a "commitment", and spending a commitment involves revealing a "nullifier". Zcash nodes keep lists of all the commitments that have been created, and all the nullifiers that have been revealed. Commitments and nullifiers are stored as hashes, to avoid disclosing any information about the commitments, or which nullifiers relate to which commitments.

For each new note created by a shielded payment, a commitment is published which consists of a hash of: the address to which the note was sent, the amount being sent, a number "rho" which is unique to this note (later used to derive the nullifier), and a random nonce. *Commitment = HASH(recipient address, amount, rho, r)*

ZK is technically about minimizing the probability of cheating or luck to a degree that makes it trustworthy. Think of multiplying a fraction. This was first proposed in 1985 with the proposal of completeness, soundness, and zero-knowledge for transactions to be trustworthy.

The ZK-proof has later been modified with SNARKS so that it might also be non-interactiv.

Why this technology was first implemented in a functional way with Zerocash has yet to take off is likely because of the fair of full transparency. KYC and AML -regulations do not like full transparency.