CHAPTER 26

Blockchain Databases

Solutions for the Practice Exercises of Chapter 26

Practice Exercises

26.1

Answer:
A fork occurs when a block is added to a block other than the most recent one in the chain. A soft fork does not invalidate prior blocks, but a hard fork does.

26.2

Answer:

a. collision resistance: No. Given a hash value y, it is easy to compute as many values as one wishes for x such that $x \bmod 2^{256} = y$.

b. irreversibility: Not in a strong sense. Given a hash value y, the set of values for x such that $x \bmod 2^{256} = y$ is a small fraction of the domain from which x was chosen. So, unless the realistic set of possible values for x in the real-world application is virtually unbounded, this function fails the irreversibility test.

c. puzzle friendliness: NO. Concatenating a bit string to another creates and easily computed new numeric value. Thus, finding a nonce is trivial computation problem.

26.3

Answer:
The single biggest reason is the energy consumption of proof-of-work.

26.4

Answer:
Proof-of-work is easier to tune for mining rate and less susceptible to control by a relatively small group of large stakeholders.

26.5

Answer:
There is no central control over a public blockchain. In a permissioned blockchain there is a controlling organization for at least membership and identity management.

26.6

Answer:
The high degree of replication of a blockchain means that a successful tamperer must alter a prohibitively large number of copies. Not only is this hard, but also any significant failed attempt is easily detected by the network. The hash-pointer structure of the chain means that all subsequent blocks to an altered block must be altered as well. With a traditional database, theft of the access password can led to arbitrary changes anywhere in the database without detection.

26.7

Answer:
Data mining of the blockchain and separately of real-world data might lead to correlations being discovered. Linkage of an ID to some other via a transaction can lead to the contruction of a relationship graph that may related a user ID to some already de-anonymized ID.

26.8

Answer:
Gas represents a payment to miners for running a smart contract in Ethereum. By charging for code execution, Ethereum is able to place a bound on total execution time and disincent the construction of computationally consumptive smart contracts.

26.9

Answer:
Nonmaliciousness allows us to assume there are no Sybil attacks. Rather we need be concerned only about arbitrary (not just fail-stop) failures. The Byzantine failure model offers that generality while 2PC makes the fail-stop assumption.

26.10

Answer:
Sharding allows parallelism in mining but also divides the set of miners into smaller sets that might be more susceptible to attack.

26.11

Answer:
Enterprise blockchains usually store more than just funds-transfers transactions among accounts, but instead store data of a more general-purpose nature.