Question 1

In Hyperledger Fabric, CouchDB is enabled as the state database by changing the state Database configuration option from goleveldb to what?

Options:

A- gocouchdb

B- setCouchDB

C- goCouch

D- CouchDB

Answer:

D

Explanation:

In Hyperledger Fabric, to enable CouchDB as the state database, the stateDatabase configuration option in the core.yaml file must be changed from goleveldb to CouchDB. This change directs the peer to use CouchDB instead of the default LevelDB for storing the world

state, which can then be queried using rich queries and provides additional features like indexing.

Question 2

What allows users to update channels, or invoke new Smart Contracts?

Options:

A- CouchDB

B- Private data collections

C- The peer CLI

D- Endorsement policies

Answer:

C

Explanation:

In Hyperledger Fabric, the peer CLI (Command Line Interface) tool is used to update channels or invoke new smart contracts. This utility allows users to perform various administrative tasks on the network, such as updating the ledger, changing endorsement policies, and managing chaincode installations and instantiations .

Question 3

What happens if the user submits a transaction with no matching function?

Options:

A- The peer will have rejected the transaction.

B- Nothing, the transaction is ignored.

C- The first transaction function is called.

D- The unknownTransaction function is called.

Answer:

D

Explanation:

In Hyperledger Fabric, if a user submits a transaction with no matching function specified in the smart contract, the unknownTransaction function is invoked by default. This function serves as a catch-all method that can be used to handle cases where the transaction type is not recognized, providing a mechanism to manage or log these occurrences effectively .

Question 4

In Raft, there are two places for configuration. One is local configuration. What is the other?

Options:

A- Global configuration

B- Channel configuration

C- Consensus configuration

D- Peer configuration

Answer:

B

Explanation:

In Raft, configuration occurs at two levels: local and channel. The local configuration pertains to node-specific settings such as TLS communication and file storage. The channel configuration, on the other hand, defines the membership and operational parameters of the Raft cluster for a specific channel, including settings like heartbeat frequency and leader timeouts. This dual-level configuration allows for precise control over the behavior of Raft nodes within the Hyperledger Fabric network .

Question 5

Question Type: **MultipleChoice**

What is the only way to recover from a failed migration from Kafka to Raft?

Options:

A- Retry the migration from Kafka to Raft.

B- Restore the state from a backup.

C- Revert to the previous version of Kafka.

D- Rebuild the network from scratch.

Answer:

B

Explanation:

Recovering from a failed migration from Kafka to Raft in Hyperledger Fabric involves restoring the state from a backup. This is necessary because once the ordering service is migrated to Raft and begins committing transactions, reverting to Kafka is not supported without a previous backup. If a migration is botched, and a backup was made at the appropriate point, administrators can roll back to this backup to restore the system to its pre-migration state .

Question 6

Question Type: **MultipleChoice**

Which of the following sources provide block events?

Options:

A- Membership service

B- Peers

C- Ordering service

D- CouchD

Answer:

B

Explanation:

In Hyperledger Fabric, block events are provided by peers. Whenever a block is committed to a peer's ledger, that peer generates a corresponding event. These events can include the full block content or a summary of the transactions within the block, allowing applications to register and receive notifications about these events. This setup is crucial for applications that need to respond to changes recorded on the blockchain in real-time .

Question 7

Which actions are combined into a single function by the Fabric Gateway client APIs to support transaction submission with a single line of code?

Options:

A- Validation and Commitment

B- Endorsement, Submit, and Commitment

C- Ordering and Commitment

D- Endorsement and Commitment

Answer:

B

Explanation:

The Fabric Gateway client APIs in Hyperledger Fabric streamline the transaction submission process by combining endorsement, submission, and commitment into a single function. This allows client applications to efficiently manage transaction submissions with a

single line of code, specifically through the SubmitTransaction() function. This consolidation simplifies the interaction model for applications, enabling a focus on business logic rather than the underlying blockchain infrastructure details .