Cryptographically Enforced Access Control

Consider a Personal Health Record (PHR) system that is used by the following types of parties: patient, doctor, insurance, health club, employer, and hospital.

Design and implement a demonstrator that supports the following functionality:

1. A patient can insert personal health data into his *own* record.
2. A patient can provide his doctor, his insurance, and his employer with read access to (parts of) his health record.
3. A hospital can insert patient health data for any patient that has been treated by *that* hospital.
4. A health club can insert training-related health data to any patient record that is a member of *that* club.

Your report on the design of the demonstrator should include:

1. A definition of the data model.
2. The description of the access control model of your choice. The access control model should be selected based on how well-suited it is for distributed environments, and especially the environment described by the aforementioned system requirements.
3. The definition of the access policies for each type of parties.

All design choices should be motivated.

The implementation should support the four abovementioned system requirements, while ensuring that nothing beyond this functionality can be provided to any type of party.

Note: *You need to use the (cryptographic) techniques taught in the lectures to enforce access control and not to implement classical access control. This is because in the scenario under consideration, the data held in the PHR is private patient information, which should remain confidential. Thus, it is a prerequisite that the data is encrypted in the first place.*

Search in Encrypted Data

Consider a financial consultant that uses a cloud storage service to store the financial data of his clients. The cloud storage server is **honest-but-curious**. To prevent data leakage, the consultant stores all data on the cloud server in encrypted form.

Design and implement a demonstrator that supports the following functionality:

1. The consultant can insert financial data for all of his clients in the storage server.
2. The consultant can search for *specific* information for any *specific* client of his in the encrypted data on the server.
3. The client can insert data in *his* own encrypted record on the storage server.
4. The client can search for specific data in *his* own record on the storage server.

To achieve the aforementioned functionality, apart from a suitable encryption scheme, you need to deploy a key distribution method, during the system setup.

Your report on the design of the demonstrator should include:

1. A definition of the data model.
2. The description of an encryption scheme and a key distribution method of your choice. The combination of the encryption scheme with the key distribution method should allow the consultant to search in any client's record and the clients to search **only** in their *own* records.
3. The system should be designed in such a way that clients can NOT search in other clients' data, while the consultant can search in all data. This should be demonstrated in your report, by a usage scenario of the implemented system.

All design choices should be motivated.

Note: *We have assumed that the cloud storage service provider is honest-but-curious. Hence, although we trust it to follow the protocol honestly, we assume that it wishes to learn as much information as possible. Your demonstrator should guarantee that the cloud storage service provider is not able to learn about the actual encrypted data it holds.*

Blockchain for Diamond Tracing

Consider a global digital registry that traces diamonds to provide verification of their authenticity and provenance. The registry uses a blockchain as a global, public ledger for the diamonds, providing a certificate for each one of them.

Design and implement a demonstrator that supports the following functionality:

1. The registry can put a piece of diamond into the system by creating an ID of it. The ID is based on the physical features of the diamond, and is supposed to be unique.
2. The system registers the place of origin of each registered diamond, and whether it is synthetic or natural.
3. The system tracks ownership of each registered diamond by using a blockchain.
4. Any client can apply for its diamond to be entered in the system (after authentication and verification).
5. Any client can sell its diamond to one, and only one other client.
6. Tampering a diamond's owner, geographic origin, and the way it was created is not possible.
7. Entering the same diamond twice is not possible.

In addition to achieving the aforementioned functionality, measurements for protecting the privacy of the clients should also be implemented.

Your report on the design of the demonstrator should include:

1. A definition of the data model.
2. The description of the blockchain you implement. What is in the public ledger, what are the transactions, what is the consensus protocol used, how do you Incentivise miners (if there are any), what is the size of the blocks in the chain, etc.
3. The description of how to enter a diamond in the ledger, how the ledger shows that a diamond is traded, how the privacy of the clients is protected, how selling the same diamond to two different clients is prevented, and how the ownership of a diamond is traced.

All design choices should be motivated.