Realizing Privacy-Preserving Features in Hippocratic Databases
Yasin Laura-Silva and Walid G. Aref

The Privacy problem and Hippocratic Databases (HDB)
- Companies need to comply with privacy laws
- How to manage/share information without violating privacy policies and data owner preferences?
- HDB has privacy as a core principle. It allows automated, fine-grained data disclosure at the database level
- There are still several problems that need to be addressed before HDBs can support efficiently the requirements of real-world systems
  1. Inadequate support of policy retention time
  2. Lack of support of policy versions
  3. Lack of an effective and flexible way to ensure that users only use purposes and recipients that they are supposed to use
  4. Lack of a way to restrict access to DML operations other than SELECT

2. Support of multiple DML operations
- Previous work only focuses on the SELECT operation
- Our contribution to support multiple DML operations includes:
  - The study of the semantics of privacy rules and preferences for other DML operations
  - The algorithms to implement these operations
- With the two first extensions, we are able to enforce restrictions like:
  1. User Mary should use only recipient Doctors when accessing table Patients for the purpose Treatment
  2. For purpose Treatment and recipient Doctors, allow sysadmin to access all the columns of table Patient, and doctors a subset of them
  3. For purpose Treatment and recipient Doctors, allow sysadmin to perform SELECT and UPDATE over table Patient but only SELECT to doctors!

4. Support of policy versions
- 80% of organizations use different privacy policies for employees and clients, 42% have multiple policies for clients, and 76% require support of policy versions
- Different cases of multiple versions/policies requirements:
  1. Multiple policies
  2. Single policy, multiple data owners
  3. Multiple policies over time
  4. Multiple versions. Two cases:
     a) The policy for patients is updated only for new patients
     b) Two policy versions for different groups of patients are simultaneously used
This last case requires the use of two policies associated with the same database entity Patient, this case is not supported by the frameworks for limiting disclosure proposed in previous work.

3. Support of retention time
- Data should be retained only as long as necessary for the fulfillment of the purposes for which it was collected
- The original HDB architecture suggests the deletion of all data items that have outlived their purpose
- Our approach to support retention time is similar to the one used to support opt-in/opt-out preferences
- It does not require deleting the information after the allowed retention time
- It uses SQL conditions, which constitutes a flexible mechanism to express complex restrictions
- It uses the element Retention of P3P privacy rules. This element can have several predefined values: no-retention, stated-purpose, legal-requrement, etc.

5. Support of generalization hierarchies
- Previous support of opt-in/opt-out choices is very limited; data owners can only give either full access to the data or deny it completely; there is not the option to give access to a generalized version of the data
- We propose the study of the integration of HDB and anonymization/generalization techniques
- We present a design to introduce generalization hierarchies into the limiting disclosure framework for HDBs