

Object Relational Mapper

OPENDNSSEC ENFORCER

René Post, rene@xpt.nl

January 11, 2011

Contents

1	Overview	2
2	Key and Signing Policy Entity-Relationship Diagram	2
3	Object Relationship Mapping	3
3.1	Data Access Layer	4
3.2	Object Relational Mapper	4
4	Performance	4

1 Overview

The enforcer uses several entities and relationships between them to reason about the state of the DNSSEC keys in the zones it manages.

Multiple databases need to be supported by the enforcer e.g. MySQL,SQLite and Postgress. To get a feeling for what that entails, 2 days were used to investigate the different available ORM solutions. The resulting ORM that was selected for the proof of concept is the BSD licensed litesql object relationship manager.

2 Key and Signing Policy Entity-Relationship Diagram

The key and signing policy ER diagram contains only the operational data needed by the enforcer for its day to day operation. The flattened XML policies that were previously stored in tables in the enforcer database have been collapsed into a single blob value in the policy entity.

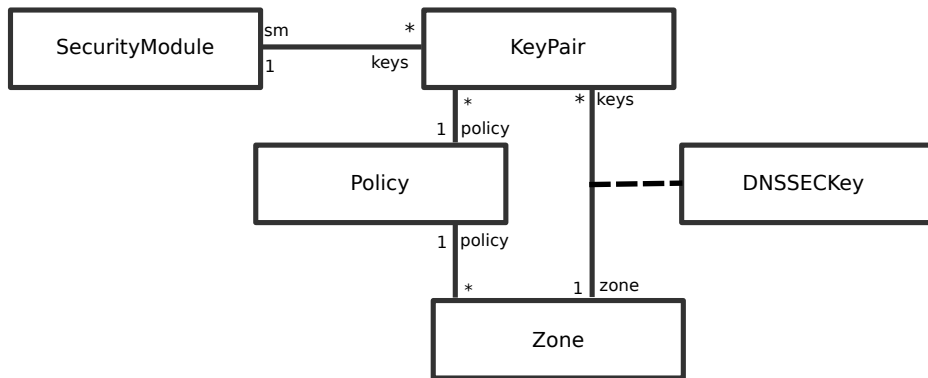


Figure 1: KASP ER Diagram

Policy Entity that represents the policies known to the enforcer. The actual policy parameters are stored as a blob value in the policy record.

SecurityModule Entity that represents the security modules (HSMs) known to the enforcer.

KeyPair Entity that represents the keypairs present in the security modules.

Security module keys Relationship that relates a keypair to the security module that contains it.

Keypair conforms to policy Relationship that relates a keypair to the policy that dictates its lifecycle.

Zone Entity that represents the zones handled by the enforcer

Zone conforms to policy Relationship that relates a zone to the policy that dictates its lifecycle.

DNSSECKey Relationship that stores information about which keypairs are used by which zones. This contains information about the state of the key and the times that changes to the state occurred/will occur.

3 Object Relationship Mapping

The object relationship mapping has been generated with the (BSD licensed) liteSQL package. Because the generated wrappers depend on a run-time support library, we need to decide whether we want to allow our code to depend on this library or if we want to replace it with something we write ourselves.

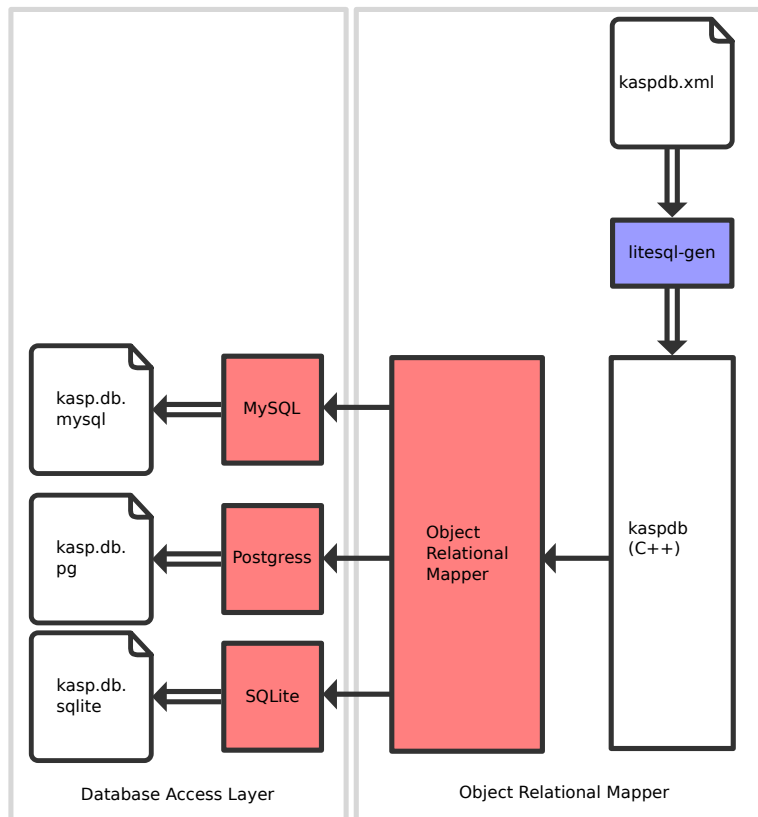


Figure 2: XML to Protocol Buffer

The colours in the diagram have the following meaning.

blue: This is not a current dependency of OpenDNSSEC. The module is needed in order to generate code from a specification for the enforcer.

red: This is not a current dependency of OpenDNSSEC. The module is needed in order to build the enforcer.

3.1 Data Access Layer

kasp.db.mysql MySQL database.

kasp.db.pg Postgress database.

kasp.db.sqlite SQLite database.

MySQL MySQL access layer contains code specific to MySQL that is needed by the ORM to work correctly.

Postgress Postgress access layer contains code specific to Postgress that is needed by the ORM to work correctly.

SQLite SQLite access layer contains code specific to SQLite that is needed by the ORM to work correctly.

3.2 Object Relational Mapper

kaspdb.xml XML file describing the entities and relations.

litesql-gen Generator from the liteSQL package used to generate the database tables.

kaspdb (C++) The C++ classes representing the entities and relations stored in the database. Allows the creation of tables, records and relations by working with C++ classes. The use of SQL is pushed down to lower levels, but queries can be executed using C++ expressions that translate into SQL database queries.

Object Relational Mapper The C++ run-time library used by the generated wrapper code to access the database.

4 Performance

Using the generated wrappers a testcase was created that writes 50000 records to the zonelist. When every zone is inserted in its own transaction this takes 80 seconds. When the zones are all written in a single transaction this takes less than 5 seconds on a 2.4GHz MacBook Core 2 Duo using an Intel 160GB SSD drive.